

ESCUELA POLITECNICA DEL EJÉRCITO
DEPARTAMENTO DE ELÉCTRICA Y ELECTRÓNICA
CARRERA DE INGENIERIA EN ELECTRÓNICA Y
TELECOMUNICACIONES

PROYECTO DE GRADO PARA LA OBTENCIÓN DEL TÍTULO
EN INGENIERÍA

IMPLEMENTACIÓN DE UN CALL CENTRE SOBRE UNA
CENTRAL DE VOIP BASADA EN ASTERISK

DANIEL VERDESOTO R

SANGOLQUI – ECUADOR
2008

ESCUELA POLITÉCNICA DEL EJÉRCITO
INGENIERÍA EN ELECTRÓNICA Y TELECOMUNICACIONES

CERTIFICADO

Ing Rubén León, Ing. Carlos Romero

CERTIFICAN

Que el trabajo titulado “Implementación de un Call Center Sobre una Central de VOIP basada en Asterisk”, fue realizado por el señor Carlos Daniel Verdesoto R, ha sido guiado y revisado periódicamente y cumple normas estatutarias establecidas por la ESPE, en el Reglamento de Estudiantes de la Escuela Politécnica del Ejército.

Quito, 29 de febrero del 2008

Ing. Rubén León

DIRECTOR

Ing. Carlos Romero

CODIRECTOR

RESUMEN

En este proyecto se desarrolló una central telefónica IP con el software de código abierto Asterisk, en su versión 1.4 considerada estable, en un servidor de comunicaciones con el sistema operativo Linux en su distribución CentOS 5.

La central telefónica maneja dos troncales analógicas que permiten la entrada y salida de llamadas con la red telefónica pública y una cuenta para llamadas internacionales a través de una conexión al Internet. Se definieron contextos para el manejo de llamadas internas, entrantes y salientes y se entregaron capacidades de call centre.

Entre las capacidades para Call Centre se encuentran la administración de colas y agentes, el desarrollo de un IVR que interactúa con la base de datos de Asterisk y capacidades de transformación de texto a voz.

Se realizó además una revisión de los conceptos más importantes a entender en un Call Centre desde el punto de vista tecnológico y un análisis económico comparativo que muestra la importante diferencia en los costos de implementación de un sistema como este en contraste con los sistemas propietarios actuales, demostrando que el sistema implementado es mucho más económico especialmente para empresas medianas y pequeñas.

AGRADECIMIENTO

A los Ingenieros Rubén León y Carlos Romero quienes han sido un apoyo inestimable para la realización de este proyecto y a todos los maestros, compañeros y amigos que han sido una fuente de inspiración a lo largo de mi carrera.

DEDICATORIA

A mis padres, por su ejemplo de esfuerzo, constancia y entrega.

PRÓLOGO

En la actualidad, las empresas dan gran importancia a las relaciones con sus clientes y buscan dar servicios que reúnan dos cualidades esenciales: calidad y bajos costos. El Call Centre, cuando se encuentra bien manejado, además de estas características provee a las áreas de atención al cliente herramientas de gran utilidad para una comunicación efectiva.

El presente proyecto consiste en el desarrollo de un ambiente de Call Centre a partir de la configuración de un servidor de comunicaciones con el software para telefonía Asterisk, el cual ha cobrado gran importancia debido a sus grandes ventajas de flexibilidad y costos.

En este trabajo se puede encontrar una revisión de los conceptos más importantes a entender en un Call Centre desde el punto de vista tecnológico, una descripción del servidor de comunicaciones desarrollado junto con las funciones que ofrece a los clientes, y un análisis económico comparativo que muestra la gran diferencia en los costos de implementación de un sistema como este en contraste con los costosos sistemas propietarios.

ÍNDICE

Prólogo	6
Índice	7
Capítulo 1	15
Introducción	15
Capítulo 2	16
Call Centre – Contact Centre	16
¿Qué es un Call Centre?.....	16
Componente Humano de un Call Centre	17
Clientes.....	17
Agentes	17
Supervisores.....	18
Componente Tecnológico de un call centre	18
Infraestructura de Aplicaciones	18
PBX	19
ACD Independiente	21
Híbridos	22
Integración Teléfono Computador CTI.....	24
Servidor de Comunicaciones	26

Infraestructura de Red.....	28
Conmutación por Circuitos	28
Comunicaciones digitales	29
Conmutación de Paquetes	30
La voz sobre IP (VoIP).....	31
Protocolo SIP	32
Beneficios de la Voz sobre IP en un Call Centre	33
Herramientas Tecnológicas Adicionales en un Call Centre.....	34
Sistema de Respuesta Interactiva de Voz (IVR).....	34
CRM (Administración de Relaciones con el Cliente)	35
TTS (text to speech)	35
Servidores de Bases de Datos	36
Servidores de Archivos.....	36
Software de Generación de Reportes.....	36
Agentes Remotos	37
Call Centres virtuales.....	37
Distribución del Call Centre	37
Instalaciones.....	38
Dinámica del call centre	39

Capítulo 3	40
Central IP Implementada	40
Red de datos	40
Componentes	40
PBX.....	41
Componentes de Hardware	41
Servidor	41
Tarjeta Para Conexión a la PSTN.....	42
Componentes de Software	43
Sistema Operativo	43
Software Servidor de Comunicaciones.....	43
Asterisk.....	43
Implementación del Servidor de Comunicaciones.....	45
Instalación del Sistema Operativo	45
Descarga de Asterisk.....	45
Instalación de Asterisk.....	46
Instalación de la tarjeta TDM400P	51
Configuración de la Central Telefónica	54
Configuración de Canales	55

	10
Canales ZAP.....	55
Canales SIP.....	56
Configuración de una cuenta con SipDiscount para llamadas internacionales	57
Plan de Marcación.....	59
Contextos.....	60
Extensiones.....	60
Contextos utilizados en la Implementación.....	60
Contextos: [interno]; [supervisor]; [espíar].	61
Contexto [salientes]	62
Macros.....	63
Interfaz de Línea de Comandos de Asterisk	65
Base de Datos de Asterisk	66
Uso de AstDB en el proyecto.....	67
Base de datos de Clientes Externos.....	67
Base de datos de Usuarios.....	67
Capítulo 4	69
Características de Call Centre	69
Configuraciones en Asterisk.....	69

Configuración de los Agentes.....	69
Opciones en el contexto [general]	69
Opciones en el Contexto [agents].....	69
Creación de agentes:.....	71
Configuración de agentes en el Proyecto Implementado:	71
Configuración de las Colas	72
Opciones del contexto [general]	72
Opciones de los Contextos de las colas:	73
Configuración de colas desarrollada en la implementación:.....	76
Configuración del IVR	77
Contextos: [ivr]; [ivr-autenticado]; [cambio-clave]	77
Contexto [ivr].....	78
Contexto [ivr-autenticado].....	78
Contexto [cambio-clave]	78
Instalación y Configuración del Motor TTS.....	79
Festival	79
Configuración de Festival para Asterisk	79
Cepstral	80
Interfaz Web para Operadora y Monitoreo de la central IP	81

Utilización del Panel	82
Capítulo 5	84
Análisis Comparativo de Costos	84
Costo aproximado de la Solución Implementada	84
Consideraciones	84
Servidor, costo y características principales	84
Costos de Software	85
Servicios Profesionales para Configuración e Instalación.	85
Costos aproximados Solución Propietaria.....	86
Central Telefónica.....	86
Call Centre.....	87
Costos Totales Aproximados de la Solución Propietaria	88
Conclusiones y Recomendaciones	89
Conclusiones.....	89
Recomendaciones.....	90
Referencias Bibliográficas	91
Textos	91
Páginas Web.....	91
Anexo 1.....	93

	13
Código de Configuración de la Central	93
Anexo 2.....	110
Configuración de Flash Operator Panel.....	110
Configuración de Asterisk para el uso de FOP.....	111
Configuración de FOP.....	111
Configuración de botones	117
Anexo 3.....	121
Pruebas en el Servidor de Comunicaciones	121
Pruebas de Llamadas Internas.....	122
Pruebas con las troncales	124
Pruebas de Llamadas Entrantes	124
Pruebas al IVR	126
Pruebas de Llamadas Salientes.....	128
Pruebas de Características Manager	132
Prueba TTS.....	132
Anexo 4.....	135
Instalación de CentOS en Modo Texto	135
Procedimientos.....	136
Anexo 5.....	152

Diagramas de Flujo del IVR y Contestadora Automática	152
Menú Principal.....	153
Menú IVR Autenticación de Usuario.....	154
Menú de Usuario Autenticado	155
Cambio de Clave.....	156
Anexo 6.....	157
Listado de Comandos del CLI de Asterisk.....	157
Índice de Figuras	162
Índice de Tablas.....	165
Glosario	166
Siglas:	166
Términos:	167

CAPÍTULO 1

INTRODUCCIÓN

El call centre es una unidad funcional utilizada como nexo entre la empresa y el cliente, y su propósito es manejar grandes volúmenes de llamadas telefónicas para dar soporte a las operaciones cotidianas de una entidad, fortaleciendo de esta manera el servicio de atención al cliente en un negocio.

La mayor parte de la tecnología disponible para implementar un call centre es bastante costosa, por lo que las empresas medianas y pequeñas no pueden acceder a ventajas como la distribución automática de llamadas y la respuesta interactiva de voz, por lo que deben conformarse con soluciones antiguas que no se ajustan a sus necesidades reales, todo debido a un bajo presupuesto.

Es en este punto donde el fuerte desarrollo del software libre y los avances tecnológicos juegan un papel importante al brindar soluciones como el sistema operativo Linux, y la popular central telefónica Asterisk, las cuales llenan un vacío en el mercado.

El principal objetivo de este proyecto es el desarrollo de un Call Centre basado en un Servidor de Comunicaciones Asterisk¹, el cual manteniendo un costo bastante accesible será capaz de dar servicio a pequeñas y medianas empresas que requieren dar un punto de contacto a sus clientes pero no están en capacidad de realizar fuertes inversiones (que pueden superar los 50.000 dólares americanos).

¹ Asterisk Central Telefónica IP basada en Software

El modelo desarrollado fue pensado para dar servicio a un Call Centre pequeño con capacidad para aproximadamente 5 a 10 extensiones, 2 troncales telefónicas analógicas, 2 colas de llamadas, 10 agentes, y uno o dos supervisores, pero rico en funciones y características. Entre las características se incluyen: buzón de mensajes, Contestadora Automática, capacidad de IVR¹, funcionalidades TTS², interacciones con una base de datos, capacidad para multi-conferencias, y una interfaz web para el control de las actividades de la central telefónica.

En este trabajo se deben distinguir tres partes principales, la primera es un análisis de los principales conceptos utilizados a lo largo del proyecto, ¿Qué es un Call Centre?, sus características principales y la tecnología que envuelve el desarrollo de un Servidor de Comunicaciones basado en Voz sobre IP, además de una revisión de otras tecnologías disponibles en el mercado actual.

La segunda parte, dividida en dos capítulos del presente trabajo, explica el desarrollo de la Central Telefónica, instalación y Configuración de sus principales características, conceptos como el plan de marcación, canales, contextos, extensiones, y las características propias de un Call Centre introducidas en la Configuración, tales como colas, agentes, IVR, interacciones con bases de datos, además de un análisis de los resultados obtenidos.

La tercera parte es un análisis de costos, en donde se observa claramente la diferencia entre el costo de una implementación con el Servidor de Comunicaciones Asterisk y otra implementación de tipo Propietario, de la cual se puede concluir la fuerte ventaja económica que entrega la opción propuesta.

¹ IVR Respuesta de Voz Interactiva

² TTS transformación de texto en voz

CAPÍTULO 2

CALL CENTRE – CONTACT CENTRE

¿QUÉ ES UN CALL CENTRE?

El call centre es una unidad funcional u oficina centralizada que se utiliza como nexo entre la empresa y el cliente, y su propósito es manejar grandes volúmenes de llamadas telefónicas entrantes y salientes, con el propósito de dar soporte a las operaciones cotidianas de la entidad. Es una herramienta que se diseña y construye a medida para atender las distintas necesidades planteadas por las áreas comerciales.

Los call centres nacieron de la oportunidad de prestar un servicio inmediato al cliente a través del teléfono y su uso se ha expandido de forma considerable. El avance tecnológico ha ayudado a reducir de forma importante el tiempo de respuesta y los costos de operación, esto hace que la cantidad y tipo de productos y servicios ofrecidos y realizados a través del teléfono aumente de forma considerable, permitiendo el uso de un único número telefónico.

El call centre es operado por una empresa para dar soporte de productos o información requerida por los clientes. También se realizan llamadas salientes para telemarketing, cobro de cuentas, encuestas, fidelidad de clientes, entre otras.

Si a las características de call centre se les agrega el manejo de cartas, faxes, y correo electrónico, entonces esta oficina se puede llamar un contact centre.

En un Call Centre se deben destacar dos partes principales: el componente humano, y la infraestructura tecnológica requerida.

COMPONENTE HUMANO DE UN CALL CENTRE

La administración adecuada de un call centre incluye el balance de eficiencia en costos y nivel de servicio. Los clientes no desean entrar en colas de llamadas excesivamente largas con esperas interminables por lo que debe existir una cantidad razonable de personas y capacidad de atención que asegure la calidad de servicio. Para esto, existen técnicas de dimensionamiento que determinan la cantidad de agentes necesarios para distintos momentos.

Dado que este proyecto se centra en el área tecnológica, y no en la parte administrativa, no se va a profundizar en áreas como la estadística, ni la administración de Recursos Humanos. Sin embargo, es importante conocer a los actores que se ven involucrados en este escenario, así como sus necesidades y requerimientos básicos, para de esta manera brindarles un servicio adecuado, por lo que a continuación se hará una breve revisión de las personas que serán atendidas por la plataforma tecnológica.

Clientes

Toda persona que se comunica con el call centre es un cliente y la percepción que tenga del trato recibido por parte del sistema debe considerarse en todo momento durante el diseño, programación y Configuración de la central telefónica.

Agentes

Un ejecutivo de atención al cliente, o agente, es un profesional encargado de recibir o realizar llamadas. Es el representante de la empresa que

directamente se comunica con el cliente para aclarar sus dudas, darle soporte, o realizar ventas de productos o servicios.

Supervisores

Un supervisor es el encargado de monitorear la calidad del servicio recibido por el cliente, lo que incluye observar a los agentes, revisar quien se encuentra atendiendo una llamada, los tiempos que toma la atención a los clientes, las llamadas atendidas, las llamadas perdidas.

Dependiendo del tipo de control que una empresa desee sobre sus agentes, deberá Configurarse el sistema telefónico para dar a los supervisores características como la posibilidad de escuchar llamadas, intervenir en conversaciones, y obtener estadísticas de calidad de servicio del call centre a su cargo.

COMPONENTE TECNOLÓGICO DE UN CALL CENTRE

Factores importantes para aliviar la carga de los agentes son el área técnica y la automatización, los cuales permiten a los clientes obtener la información requerida de forma más rápida y menos rígida, incluso en ocasiones, sin la necesidad de contactarse con un empleado. Esto además de mejorar la experiencia del cliente, permite a las empresas un importante ahorro en gastos operativos.

A continuación se presenta un resumen de los más importantes componentes tecnológicos y las distintas opciones disponibles para la implementación de un Call Centre.

INFRAESTRUCTURA DE APLICACIONES

Se pueden destacar cinco tipos de Infraestructura para comunicaciones dentro del ámbito de un Call Centre, estos son: PBX, ACD Independiente, sistemas Híbridos, CTI, y Servidores de Comunicaciones.

PBX

La PBX (por sus siglas en inglés Private Brand Exchange) es un computador especializado para la administración de conexiones telefónicas. Tiene una unidad central de procesamiento CPU, memoria, unidades de almacenamiento, y software tanto de sistema operativo como de aplicaciones.

Otros componentes incluyen gabinetes y tarjetas de circuito que proveen conexión a los teléfonos así como a los proveedores de servicio telefónico. Su mecanismo de transporte es la multiplexación TDM¹.

Para su utilización en Call Centres, los proveedores de PBX utilizan software especial (llamado usualmente software ACD²) que se activa dentro de la misma PBX para permitir el uso de colas, enrutamiento de llamadas, monitoreo y administración de agentes, entre otros.

¹ TDM Siglas en Inglés para Multiplexación por División en Tiempo

² ACD Siglas en Inglés para Distribuidor Automático de Llamadas

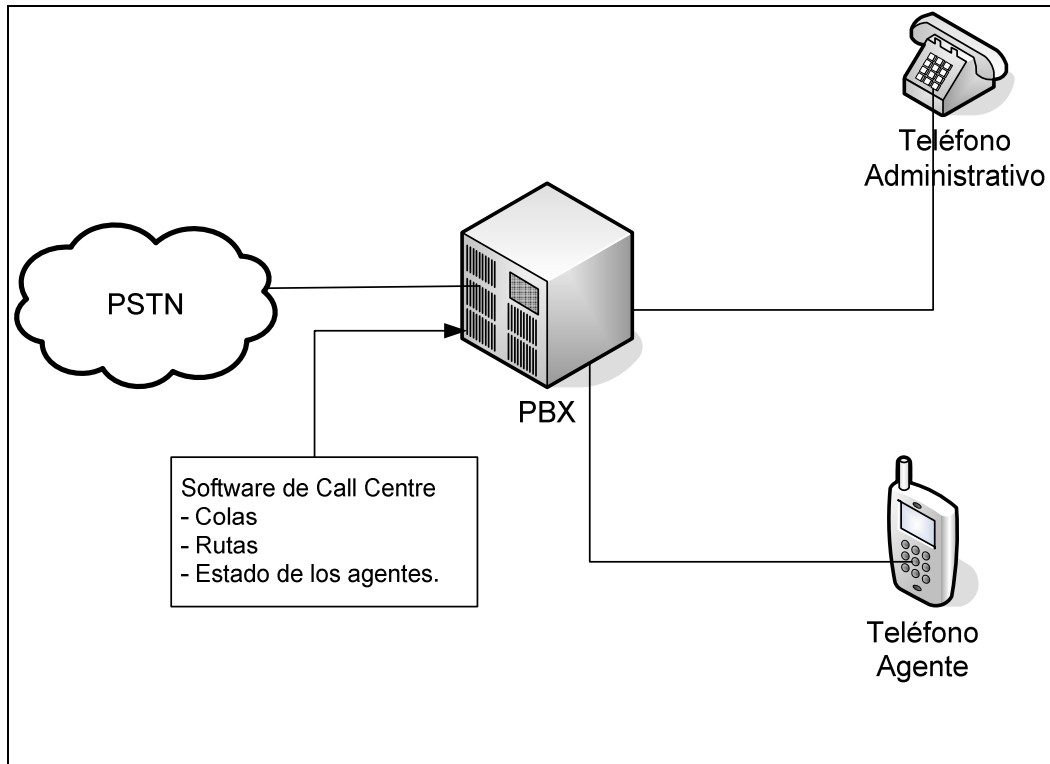


Figura 2. 1 Infraestructura con PBX

Ventajas de una PBX	Desventajas de una PBX
<ul style="list-style-type: none"> • Un sistema es capaz de atender una organización completa • Alta escalabilidad • Buenas herramientas de administración y mantenimiento. • Sets de productos y servicios • Capacidades robustas para el manejo de llamadas. • Tecnología probada. 	<ul style="list-style-type: none"> • Propietario, hardware y software de un solo proveedor. • Costoso por su naturaleza propietaria. • No maneja una arquitectura elegante para multimedia. • No maneja de forma elegante comunicaciones basadas en paquetes.

Tabla 2. 1 Ventajas y Desventajas de la Infraestructura con PBX

ACD Independiente

Un ACD (Siglas en Inglés para un Distribuidor Automático de Llamadas) es un dispositivo o sistema especializado optimizado para el uso en call centres, distribuye las llamadas entrantes a un grupo específico de terminales que utilizan los agentes.

El sistema de ACD independiente tiene gabinetes dedicados, CPU, sistema operativo, almacenamiento y memoria, y sirve al call centre mientras que una PBX atiende al resto del negocio (usuarios administrativos). Los usuarios administrativos se comunican con los agentes entre los diferentes sistemas mediante troncales que los enlazan.

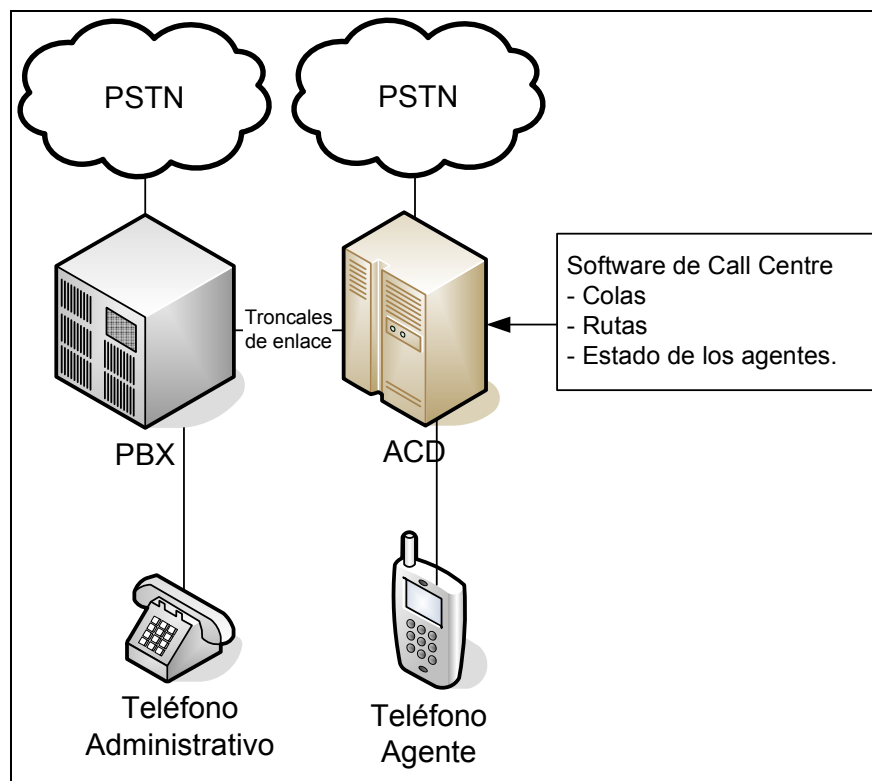


Figura 2. 2 Infraestructura con ACD Independiente

Ventajas de un ACD	Desventajas de un ACD
<ul style="list-style-type: none"> Alta escalabilidad. Alta confiabilidad. 	<ul style="list-style-type: none"> Infraestructuras duplicadas para administración,

<ul style="list-style-type: none"> • Funciones y características avanzadas específicas para las necesidades de un call centre • Sets de productos y aplicaciones. • Enfocado a la industria de Call Centre. • Buenas aplicaciones y herramientas de diseño. 	<p style="text-align: center;">mantenimiento y manejo.</p> <ul style="list-style-type: none"> • Costoso por inversiones dobles. • Arquitectura propietaria. • No es concebido para interactuar con multimedia. • No se maneja elegantemente con comunicaciones por paquetes.
-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------	------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

Tabla 2. 2 Ventajas y Desventajas de la Infraestructura con ACD Independiente

Híbridos

Dado que los clientes han buscado mayores funcionalidades y arquitecturas abiertas, algunos proveedores han migrado algunas de sus aplicaciones para call centre desde su conmutador PBX a servidores estándar con sistemas operativos conocidos.

De esta forma se aprovechan algunas ventajas de los servidores actuales, y se realiza una separación de las funciones de Call Centre del conmutador de la empresa, pero se mantiene la naturaleza propietaria, tanto del software que se instala en el servidor híbrido como de la forma de enlazarlo al sistema PBX.

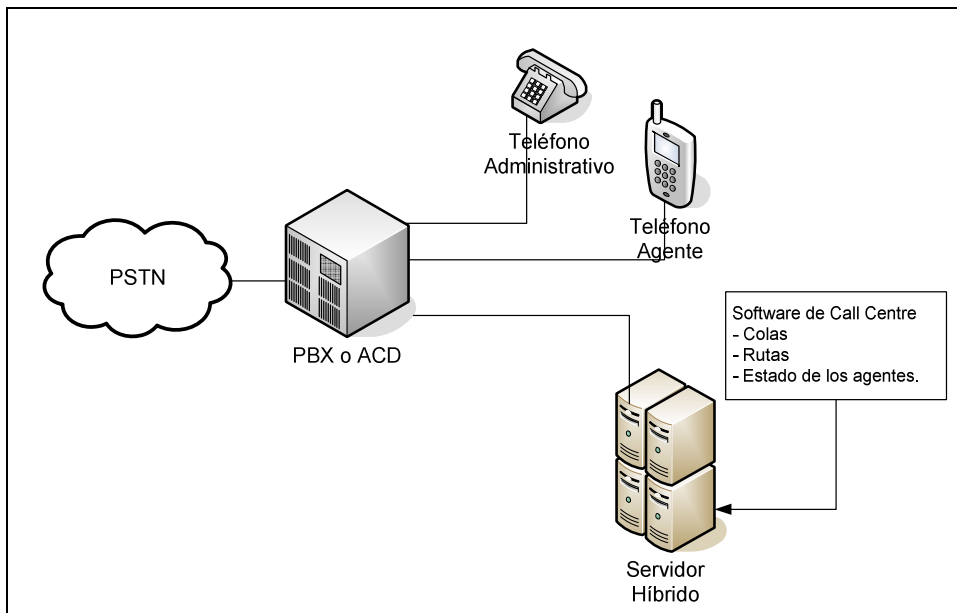


Figura 2. 3 Infraestructura con Servidor Híbrido

Ventajas de los sistemas Híbridos	Desventajas de los sistemas Híbridos
<ul style="list-style-type: none"> • Mayores funcionalidades • Capacidades mejoradas. • Corre sobre una plataforma estándar más abierta. • Administración mejorada del sistema. • Provee un camino para colas y rutas multimedia. 	<ul style="list-style-type: none"> • Se mantiene la naturaleza propietaria. • Varios requieren que el hardware y software se adquieran del mismo proveedor. • Puede requerir doble administración de la PBX o ACD y el servidor externo.

Tabla 2. 3 Ventajas y Desventajas de la Infraestructura con Servidor Híbrido

Integración Telefonía - Computador CTI

La integración Telefonía - Computador (CTI¹ por sus siglas en inglés), crea nexos entre el mundo de la telefonía tradicional con el modelo cliente/servidor. La infraestructura CTI puede agregarse o reemplazar a la infraestructura de PBX y/o ACD.

Algunas aplicaciones CTI agregan funcionalidades del conmutador, ya que una PBX o ACD no puede llevar a cabo algunas tareas sin el apoyo de software externo adicional.

Ejemplos son los softphones², marcación predictiva, integración multimedia, y generación de reportes, que consiguen mejorar las funcionalidades de su infraestructura telefónica.

Otras aplicaciones CTI reemplazan al ACD o PBX en sus funcionalidades de Call Centre, utilizando hardware y software que se encuentran fuera del conmutador de voz, esto no aumenta funcionalidades, pero cambia el lugar en donde encuentran y ejecutan las mismas.

Esta arquitectura es aparentemente bastante similar a la del modelo híbrido, pero existen grandes diferencias de fondo, una es la cantidad de tareas que se mueven desde el conmutador hacia el servidor externo, y la segunda se refiere a la interoperabilidad, ya que CTI soporta una variedad de PBX y ACDs, mientras que el software híbrido trabaja únicamente con equipos de su marca mediante enlaces licenciados por el mismo proveedor.

¹ CTI siglas en inglés para Computer Telephony Integration

² Softphone nombre utilizado comúnmente para teléfonos por software

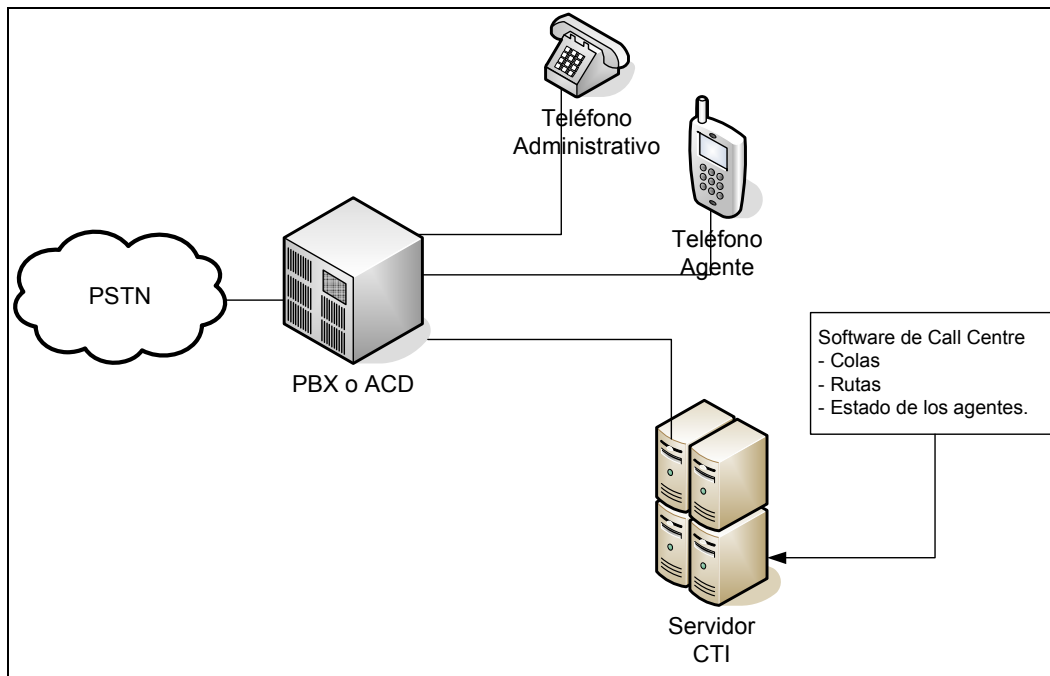


Figura 2. 4 Infraestructura con Servidor CTI

Ventajas CTI	Desventajas CTI
<ul style="list-style-type: none"> • Reutilización del hardware de PBX/ACD existente. • Interoperabilidad con una variedad de PBX/ACD • Soporte para diferentes tecnologías de software. • Soporte de estándares como TAPI, SQL, ODBC, TCP/IP. • Kits de herramientas. • Altamente personalizable. • Opciones para el Sistema Operativo, RDMS, servidor preferido. • Posibilidades de utilización de una variedad de aplicaciones CTI 	<ul style="list-style-type: none"> • Escalabilidad puede limitarse. • Problemas de confiabilidad. • Pueden requerirse altos esfuerzos para personalizar. • Puede requerir doble administración para el conmutador y el servidor externo. • Posibilidad de quedar atrapado con una aplicación de software. • Dificultades en el soporte por la alta personalización. • La calidad de la aplicación depende mucho del enlace entre conmutador y servidor CTI.

<ul style="list-style-type: none"> • Apertura para colas y rutas multimedia. 	<ul style="list-style-type: none"> • El procesamiento de llamadas puede requerir una gran cantidad de puertos IVR.
---------------------------------------------------------------------------------------------	-----------------------------------------------------------------------------------------------------------------------------------

Tabla 2. 4 Ventajas y Desventajas de la Infraestructura con Servidor CTI

Servidor de Comunicaciones

Con esta opción, el sistema telefónico corre sobre un servidor estándar con sistemas operativos conocidos, como NT, Unix o Linux y es bastante orientado a software. Es posible escoger componentes como el sistema operativo, motor de base de datos, hardware para el servidor, y tarjetas de telefonía. O tomar un set de opciones certificadas o aprobadas por un distribuidor.

Esta infraestructura entrega capacidades integradas para funciones de telefonía básicas, así como paquetes completos con capacidades para Call Centre. Además, esta infraestructura tiene capacidades integradas de procesamiento de voz como IVR y correo de voz, reportes, colas, y aplicaciones CTI de escritorio, por lo que se suele llamar una solución todo en uno.

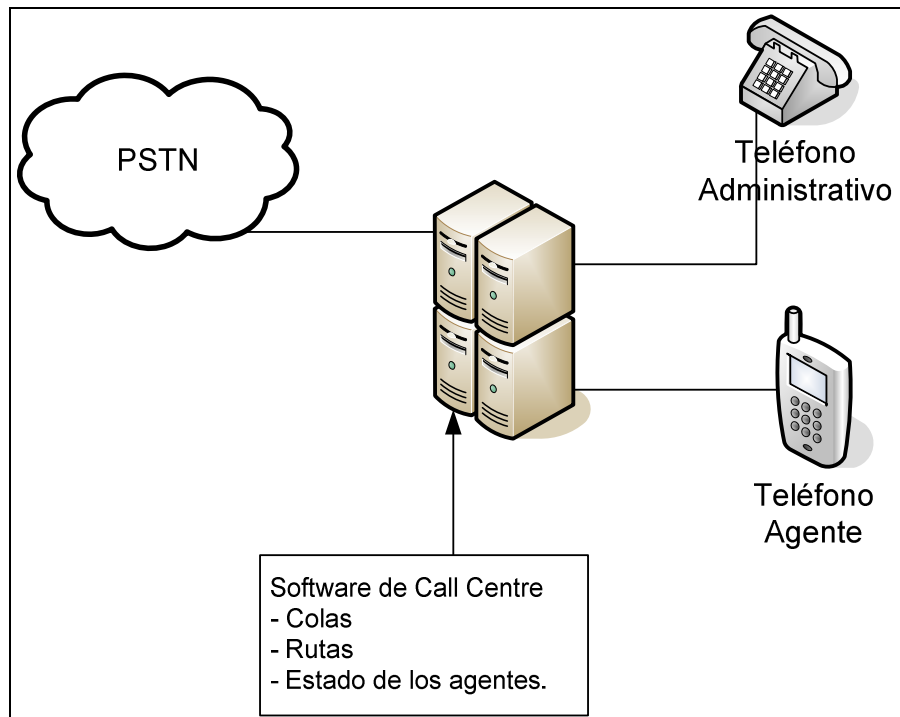


Figura 2. 5 Infraestructura con Servidor de Comunicaciones

Ventajas de un Servidor de Comunicaciones	Desventajas de un servidor de comunicaciones.
<ul style="list-style-type: none"> • Capacidades de enrutamiento ACD, reportes, IVR, CTI, correo de voz y multimedia altamente integradas. • Opciones para elegir componentes de software como OS, servidores web, motor de base de datos. • Opciones para componentes de hardware: (servidores, tarjetas de telefonía, teléfonos) • Sets de herramientas para personalización. • Altas funcionalidades para centros medianos y pequeños. 	<ul style="list-style-type: none"> • La confiabilidad es altamente dependiente del servidor, sistema operativo, software de aplicación. • La escalabilidad puede verse limitada. • Un mercado relativamente nuevo, por lo que las aplicaciones no se encuentran del todo maduras. • Soporte y mantenimiento pueden dificultarse debido a la alta personalización, múltiples componentes y proveedores. • Los costos de personalización

<ul style="list-style-type: none"> • Buena opción para organizaciones que prefieren hacer su propio soporte IT. 	<p>pueden ser altos.</p>
--------------------------------------------------------------------------------------------------------------------------------	--------------------------

Tabla 2. 5 Ventajas y Desventajas de la Infraestructura con Servidor de Comunicaciones

En el proyecto implementado, este es el modelo escogido, Asterisk es un software que al instalarse en un computador bajo el sistema operativo Linux se convierte en un servidor de comunicaciones con todas las características anteriormente mencionadas.

INFRAESTRUCTURA DE RED

Hasta hace pocos años, todo el tráfico de voz era llevado a través de redes con conmutación de circuitos, pero en la actualidad ha aparecido una nueva opción para transportar la voz, la conmutación por paquetes, esta tecnología es la base sobre la cual se construye la voz sobre IP.

A continuación se van a tratar brevemente estas dos opciones para la infraestructura de red sobre la que se puede transportar la voz en un Call Centre.

Conmutación por Circuitos

Con este tipo de conmutación, una conexión dedicada se establece de forma temporal para conectar a las dos terminales que se comunican desde el momento en que el teléfono timbra hasta que se cierra la llamada. Nadie más puede utilizar ese canal mientras la llamada se mantenga.

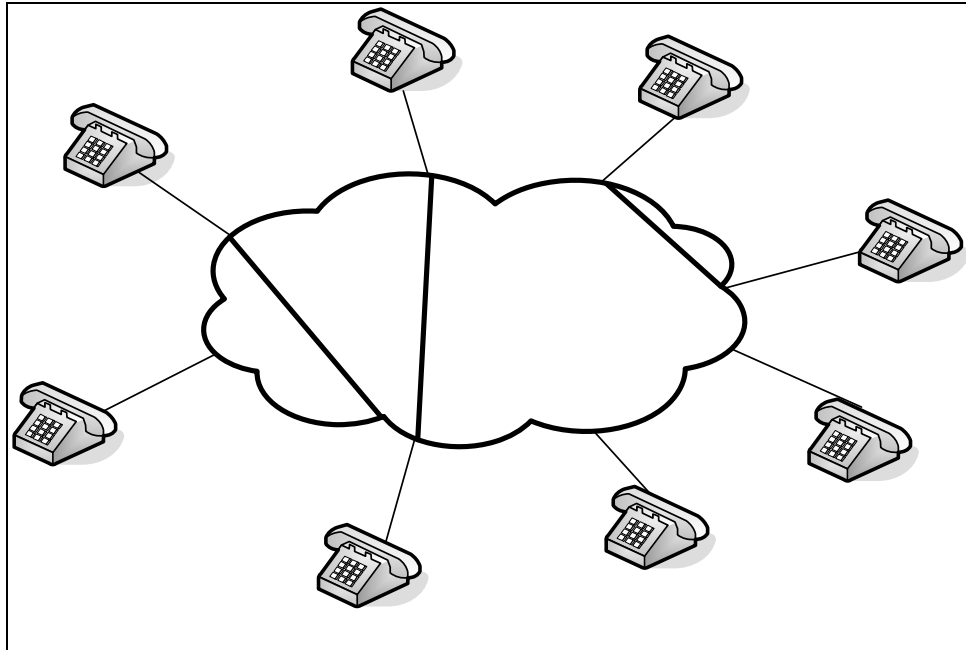


Figura 2. 6 Conmutación de Circuitos

Comunicaciones digitales

Hoy en día, la mayor parte de las comunicaciones de voz son digitales. Los teléfonos transforman la voz en señales digitales mediante modulación por codificación de pulsos (PCM); la conversación se transforma en una cadena a 64 kbits/segundo.

A través de la Multiplexación por División en Tiempo (TDM), múltiples conversaciones se colocan en un mismo circuito. Cada conversación se transmite utilizando espacios de tiempo asignados (time slots). La multiplexación por división en Tiempo es utilizada tanto en proveedores de servicio telefónico como en centrales privadas.

Los caminos utilizados para unir a un proveedor de servicio telefónico al conmutador de una empresa se conocen como “troncales” o líneas telefónicas. Estos troncales pueden utilizar transmisión analógica o digital.

Para aplicaciones de Call Centre es común utilizar líneas digitales especiales llamadas E1 o T1, un T1 (utilizado normalmente en Norteamérica y Japón) es capaz de llevar 24 conversaciones simultáneas, mientras un E1 (Europa, Latinoamérica) puede llevar 31 conversaciones multiplexadas en el tiempo.

Conmutación de Paquetes

Mientras la voz tradicionalmente ha utilizado la conmutación de circuitos, la comunicación de datos utiliza conmutación de paquetes, en donde la información es dividida en pequeños fragmentos, que se envían a una red compartida.

Cada paquete tiene una dirección de remitente, dirección de destino y un número de secuencia en su cabecera. Los equipos de red (routers, switches) interpretan estas direcciones y envían cada paquete a su destino (la ruta de cada paquete puede ser distinta dependiendo de congestión y disponibilidad de las rutas). Finalmente el destinatario recibe y reordena los paquetes.

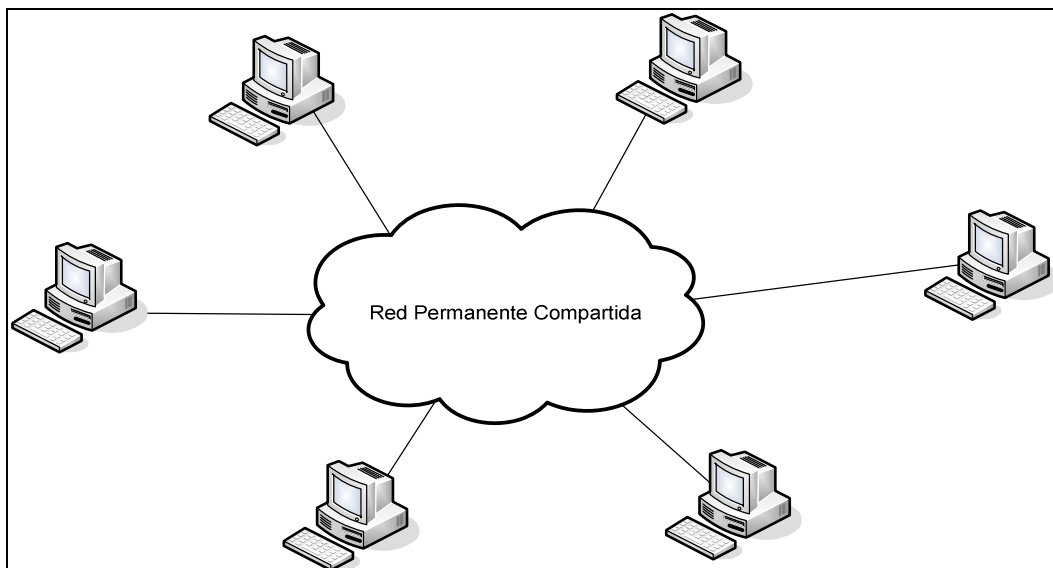


Figura 2. 7 Conmutación de Paquetes

La voz sobre IP (VoIP)

Con el gran crecimiento del tráfico de datos, y la capacidad de las redes actuales, una evolución natural ha sido el interés por transmitir voz y otras aplicaciones en tiempo real a través de las redes de datos, consiguiendo con esto una convergencia que trae beneficios como un mejor manejo de los activos de una empresa y un ahorro considerable.

Una desventaja de la VoIP son algunas de las premisas con las que se desarrollaron las redes de datos. Los protocolos para la transmisión de datos se desarrollaron para redes no confiables.

Las redes de datos utilizan comúnmente el set de protocolos TCP/IP en donde los paquetes dañados son retransmitidos, o pueden llegar en desorden, lo cual no funciona para aplicaciones en tiempo real. Para resolver estos inconvenientes, la VoIP se vale de otros protocolos como UDP y RTP,

Protocolo de Datagramas de Usuario UDP¹. Un rápido protocolo best-effort², esto significa que la red realiza su mejor esfuerzo para entregar los paquetes y si estos llegan dañados son descartados. La voz puede tolerar la pérdida de unos pocos paquetes dañados. Este protocolo funciona bien con la transmisión de voz ya que en este caso es más importante la velocidad que la confiabilidad.

Protocolo de Transporte en tiempo Real RTP³. Ayuda con el ordenamiento de los paquetes. Coloca un time-stamp (marcas de tiempo) en cada paquete de voz. La aplicación de VoIP lee la marca de cada paquete que llega y si este se encuentra fuera de orden, es descartado. Si esto ocurre de forma ocasional, el cerebro humano llena los vacíos y entiende la información sin mayores dificultades, si los problemas son frecuentes, la comunicación se ve degradada.

¹ UDP siglas para User Datagram Protocol

² Best-effort El mejor esfuerzo

³ RTP siglas para Real-time Transport Protocol

Para establecer sesiones de comunicación, la VoIP se basa en las recomendaciones H323 y el protocolo SIP.

H323. Es una recomendación de la ITU-T¹, que define los protocolos para proveer sesiones de comunicación audiovisual sobre paquetes de red, es una parte de la serie de protocolos H.32x, los cuales también dirigen las comunicaciones sobre RDSI, RTC o SS7.

Es utilizado comúnmente para Voz sobre IP y para videoconferencia basada en IP. Es un conjunto de normas ITU para comunicaciones multimedia que hacen referencia a los terminales, equipos y servicios estableciendo una señalización en redes IP.

No garantiza calidad de servicio, y es independiente de la topología de la red, admite pasarelas, permitiendo usar más de un canal de cada tipo (voz, video, datos) de forma simultánea.

Protocolo SIP

Session Initiation Protocol (Protocolo de Inicio de Sesión) es un protocolo desarrollado con la intención de ser el estándar para la iniciación, modificación y finalización de sesiones interactivas de usuario donde intervienen elementos multimedia como video, voz, o mensajería instantánea.

SIP es uno de los protocolos de señalización para voz sobre IP. Los clientes SIP usan el puerto 5060 en TCP (Transmission Control Protocol) y UDP (User Datagram Protocol) para conectar con sus servidores. SIP es usado para iniciar y terminar llamadas de voz y video.

¹ ITU-T International Telecommunication Union – Telecommunication o Unión Internacional de Telecomunicaciones

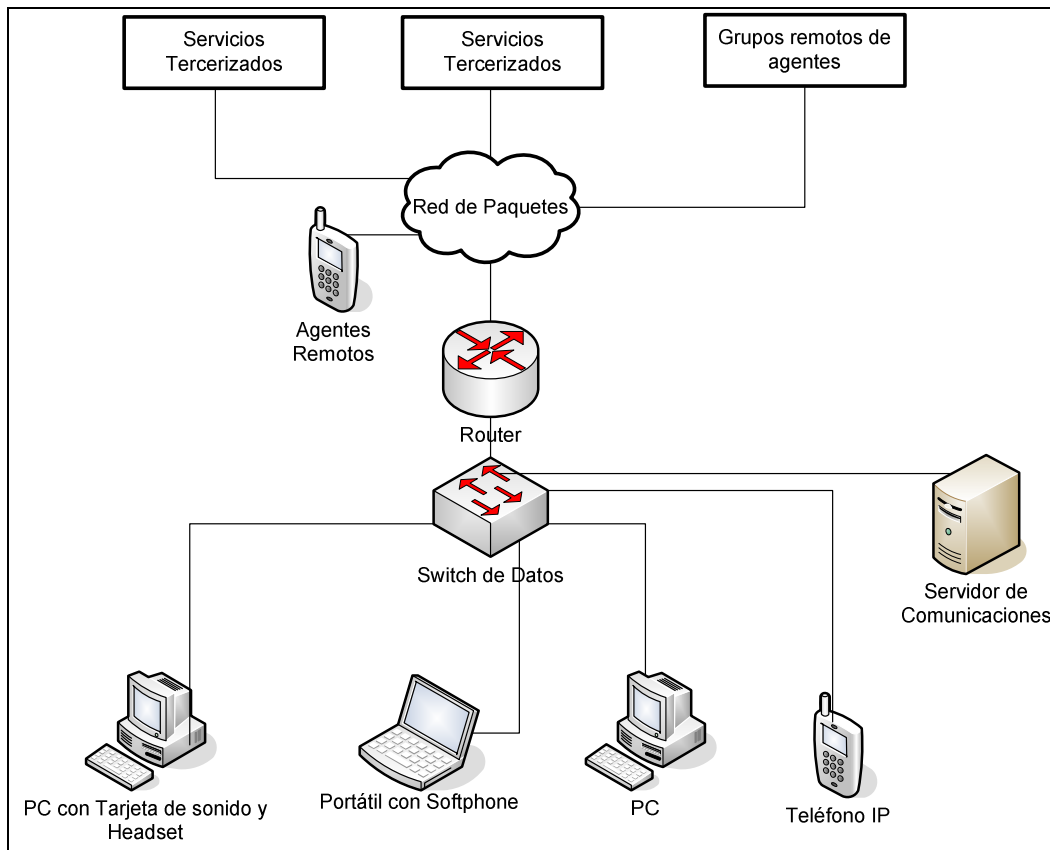


Figura 2. 8 Red Convergente

Beneficios de la Voz sobre IP en un Call Centre

Las ventajas que esta tecnología ofrece a un Call Centre en la actualidad son muchas, entre estas se pueden mencionar las siguientes.

- Mejora la utilización de activos ya que aprovecha la red y equipos de datos para las aplicaciones tanto de voz como de datos.
- Reduce costos al eliminar la infraestructura de conmutación basada en TDM.
- Permite el trabajo con varios locales interconectados de forma económica.
- Permite agentes remotos, como agentes en casa, o grupos de agentes en oficinas distantes.
- Provee la base para un contact centre multimedia.

- Reduce costos internos de mantenimiento.
- Elimina mantenimiento de equipo de conmutación PBX/ACD
- Elimina los costos de actualización o upgrade de PBX/ACD
- Simplifica el cableado de la empresa.
- Facilita los movimientos, cambios, y aumento de dispositivos.
- Permite una reducción de costos en llamadas de larga distancia.

HERRAMIENTAS TECNOLÓGICAS ADICIONALES EN UN CALL CENTRE

Los call centres utilizan una amplia variedad de distintas tecnologías que les permiten manejar el volumen de trabajo a su cargo. Estas tecnologías se aseguran de que los agentes sean más productivos, y las llamadas sean procesadas en el menor tiempo posible, resultando en buenos niveles de servicio. Entre estas se pueden destacar las siguientes:

Sistema de Respuesta Interactiva de Voz (IVR)

La respuesta interactiva de voz se encarga de las llamadas entrantes, las encamina y permite a los clientes realizar labores rutinarias sin la necesidad de interactuar con un agente, por ejemplo, para realizar consultas de saldos, planillas, etc. en donde es suficiente con la generación de mensajes automáticos entregados por el sistema, dejando de esta manera a los agentes la atención especializada y específica.

La respuesta interactiva de voz permite a un computador detectar voz, y tonos utilizados en una llamada telefónica normal. El sistema IVR puede responder con audio pregrabado o generado dinámicamente para indicar al cliente como proceder. Estos sistemas pueden utilizarse para controlar prácticamente cualquier función donde la interfaz pueda ser dividida en una serie de opciones de menú.

Los sistemas de IVR se utilizan para manejar grandes volúmenes de llamadas, reducir costos y mejorar la experiencia del cliente. Ejemplos típicos son banca telefónica, transacciones de tarjetas de crédito, y voto telefónico.

Los call centres utilizan sistemas de IVR para identificar y separar a los clientes, lo cual facilita la personalización de los servicios de acuerdo al perfil del cliente que llama. También se aumenta la opción de escoger servicios automáticos autenticando al cliente por el número telefónico desde el que llama o solicitándole un número pin o una contraseña.

La utilización de DNIS (dialled number information services o información sobre el número marcado) ayuda a personalizar el servicio entregado.

CRM (Administración de Relaciones con el Cliente)

El manejo de relaciones con el cliente CRM, cubre conceptos utilizados por las compañías para manejar temas que incluyen la captura, almacenamiento y análisis de información de clientes, vendedores, socios, y procesos internos

Para el CRM existen varias opciones en software que permiten una manera de organizar toda esta información, y son una herramienta tecnológica de mucha utilidad en un contexto como el Call Centre. En las consideraciones tecnológicas se debe tener en cuenta:

- Bases de datos para almacenar la información de clientes.
- Software de soporte para el agente en la parte operacional.
- Sistemas de manejo de interacciones para el CRM colaborativo.
- Software de análisis estadístico para el CRM Analítico.

TTS (text to speech)

Existen motores que permiten convertir el texto escrito en voz con lo cual es posible entregar datos del sistema a los clientes por medio de una voz sintética.

Servidores de Bases de Datos

Son los repositorios de la información de los clientes de la organización, estos pueden interactuar con los sistemas de PBX e IVR mediante distintas Configuraciones.

Servidores de Archivos

En donde se encuentra la información que comparten los agentes, además de herramientas para brindar el servicio requerido.

Software de Generación de Reportes.

Una característica normalmente utilizada es la generación de estadísticas y reportes de un call centre, con lo que se obtienen pautas cuantitativas sobre el nivel de servicio.

Existen varias mediciones utilizadas para evaluar el desempeño de un call centre. Algunas de ellas son las siguientes:

- Tiempo medio de espera que debe experimentar un cliente desde que ingresa a una cola.
- Tiempo medio de conversación.
- Porcentaje de llamadas contestadas en un período de tiempo.
- Número de llamadas por hora que debe atender un agente.
- Cantidad de tiempo que le toma al agente procesar requerimientos del cliente fuera de conversación.
- Porcentaje de llamadas en las que se resuelve el problema del cliente.
- Porcentaje de abandono de llamadas.
- Porcentaje de tiempo en que los agentes no están listos para recibir llamadas, (Idle time).

Las métricas mencionadas entregan números que son posteriormente analizados por supervisores y gerentes.

La calidad de la atención es también monitoreada por equipos que utilizan grabaciones de las llamadas para realizar un análisis cualitativo del tipo de servicio entregado.

Otra manera de medir niveles de servicio es la utilización de encuestas para obtener una retroalimentación con respecto a la experiencia del cliente.

Agentes Remotos

Esta es una alternativa a tener a todos los agentes en una oficina central. En esta alternativa, los agentes trabajan desde su hogar utilizando un acceso ISDN BRI para comunicarse con una plataforma centralizada de computación. Este tipo de agentes tienen como ventaja la reducción de costos de implementación de la oficina central, así como la reducción de gastos de transporte, sin embargo el call centre debe cubrir los costos de la línea ISDN. Otra alternativa es el uso de VoIP, la cual puede facilitar esta comunicación con un menor costo.

Call Centres virtuales

Los call centres virtuales se construyen utilizando varios call centres de menor tamaño en distintas oficinas y conectándolos entre ellos. La ventaja de estos es que mejoran los niveles de servicio, proveen un respaldo en emergencias y permiten manejar los horarios extendidos de operación en call centres aislados.

DISTRIBUCIÓN DEL CALL CENTRE

Un call centre es usualmente operado en un espacio amplio donde se encuentran las estaciones de trabajo para los agentes y un número de supervisores. Puede operar de manera independiente o conectado a otros

centros en una red corporativa en donde interactúa con Mainframes, computadores y redes de área local. Es también posible tener agentes remotos, lo cual permitiría el trabajo desde casa.

Instalaciones

La centralización del manejo de llamadas apunta a mejorar las operaciones y reducir costos, mientras provee un servicio estandarizado, y uniforme para los consumidores haciendo de este un enfoque ideal para compañías grandes con amplias necesidades de soporte a sus clientes. Para atender una base de clientes tan grande, se adaptan amplios espacios para atender las operaciones de call centre bajo un solo techo.

En oficinas centralizadas, una gran cantidad de trabajadores pueden ser controlados por un número relativamente bajo de supervisores y personal de soporte. Son usualmente apoyados por tecnología que gestiona, mide y monitorea las actividades y rendimiento de los empleados.

En un call centre se monitorea muy de cerca el nivel de conocimientos y servicio al cliente. Operaciones típicas en un call centre se enfocan en disciplinas de manejo de empleados, manejo de colas, monitoreo de calidad y reportes. Los reportes se pueden clasificar en reportes en tiempo real y reportes históricos. Los tipos de información recogida para un grupo de agentes incluye típicamente: agentes registrados, agentes listos para tomar llamadas, duración media de llamadas, llamadas con mayor duración, número de llamadas en la cola, número de llamadas abandonadas y velocidad media para contestar.

Varios Call centres usan un software para el control de personal, el cual utiliza información histórica que junto a las necesidades proyectadas puede generar itinerarios de trabajo. Esto busca proveer suficiente personal capacitado para atender las llamadas.

El alto costo de personal y espacio de oficina, así como las dificultades en el manejo de contrataciones, gerencia, y costos de operación han influenciado para la tercerización del servicio de call centre en muchas empresas.

DINÁMICA DEL CALL CENTRE

Las llamadas que maneja un Call Centre son divididas comúnmente en entrantes y salientes. Las llamadas entrantes son realizadas por el consumidor para obtener información, reportar problemas, o pedir ayuda. Estas tienen un comportamiento bastante distinto a las llamadas salientes, donde los agentes contactan al cliente con la intención de realizar una venta o dar un servicio. El call centre desarrollado en este proyecto tiene como objetivo manejar llamadas entrantes de clientes.

El personal de un Call Centre se organiza usualmente en un sistema de varios niveles para dar un manejo más eficiente a las llamadas. El primer nivel consiste en operadores, que dirigen las consultas o reclamos al departamento a cargo y proveen información general, este primer nivel puede ser reemplazado con un IVR programado de forma adecuada.

Si un cliente necesita mayor asistencia, puede ser enviado a un segundo nivel donde más inconvenientes o dudas pueden resolverse por una persona con mayor conocimiento. En algunos casos, el escalamiento puede continuar con tres o más niveles en donde el cliente tiene contacto con ingenieros, desarrolladores, o soporte de alto nivel técnico del producto.

CAPÍTULO 3

CENTRAL IP IMPLEMENTADA

En la implementación se deben distinguir dos partes básicas: La red de datos, y la PBX IP con todas sus Configuraciones y software.

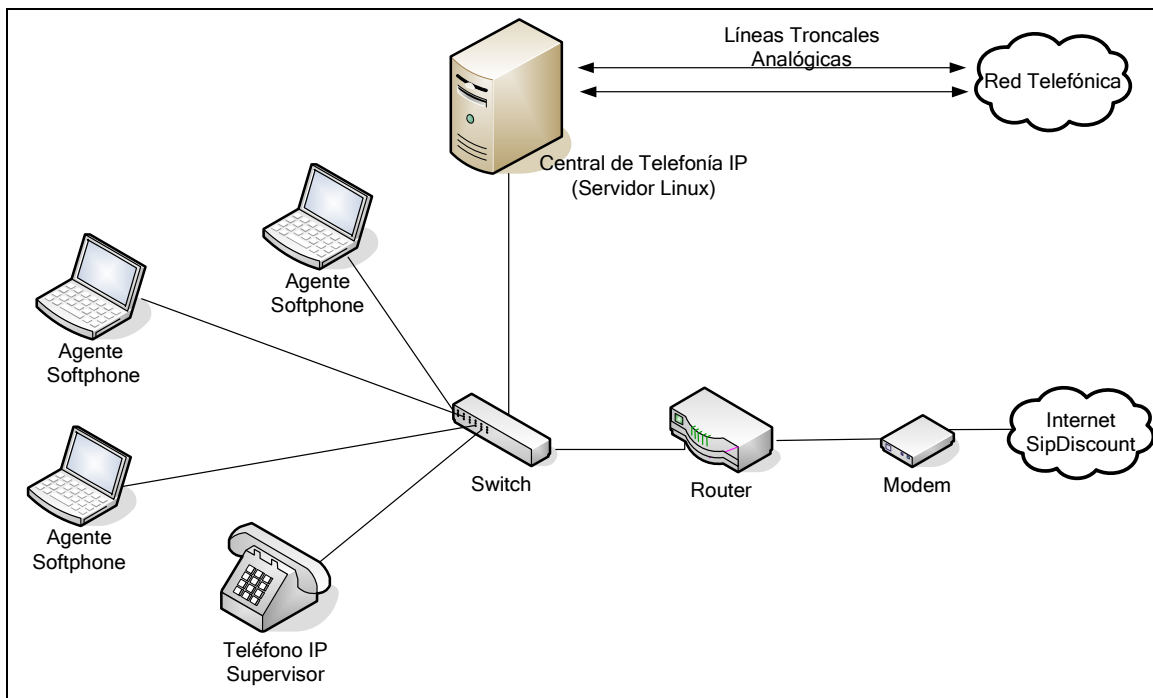


Figura 3. 1 Diagrama de la Implementación

RED DE DATOS

Componentes

- Router Inalámbrico
- Teléfono IP Grandstream Budge Tone-200
- Computadores para los agentes

- Teléfonos IP en software (softphones)

PBX

Componentes de Hardware

Servidor

La selección del servidor de acuerdo al número de canales y el tipo de servicio que va a prestar es un punto de mucha importancia, la tabla 3.1 fue tomada del texto Asterisk The Future of Telephony y contiene consideraciones sobre el Hardware requerido de acuerdo al número de canales que tendrá que manejar el Servidor.

PROPÓSITO	NÚMERO DE CANALES	MÍNIMO RECOMENDADO
Sistema de Hobby	No más de 5	400-MHz x86, 256 MB RAM
SOHO	5 a 10	1-GHz x86, 512 MB RAM,
Negocio pequeño	Hasta 15	3-GHz x86, 1 GB RAM
Sistema mediano a grande	Más de 15	CPU dual, posiblemente arquitectura distribuida con múltiples servidores

Tabla 3. 1 Guía de Requerimientos del Sistema

El computador designado para el uso como Servidor tiene las siguientes características técnicas.

PROCESADOR	INTEL CORE 2 DUO E4300 1.86GHZ
MAINBOARD	INTEL D946GZIS CD,S775,1066MHZ,DDR2
DISCO DURO	SEAGATE 250GB 7200 Sata
MEMORIA	DDR 2 1GB PC-533mhz Corsair.
TARJETA RED	TARJETA DE RED 10/100

Tabla 3. 2 Características del Servidor Utilizado en el Proyecto

Tarjeta Para Conexión a la PSTN

Digium TDM400P con cuatro puertos FXO.

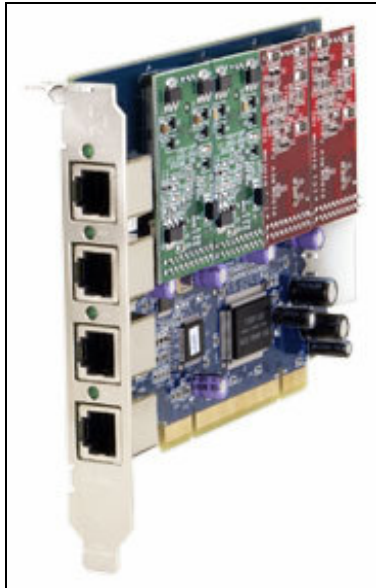


Figura 3. 2 Tarjeta TDM400P

La tarjeta TDM400P es un dispositivo que permite conectar un computador a la PSTN y/o a teléfonos analógicos comunes. Esto se consigue a través de líneas telefónicas conectadas a los puertos FXO (Foreign Exchange Office) o teléfonos analógicos a los puertos FXS (Foreign Exchange Station).

Identificación de los módulos

Las tarjetas TDM400P pueden traer módulos FXS o FXO, estos se identifican por su color:

Los Módulos FXO son de color rojo.

Los Módulos FXS son de color verde.

En la Figura 3.2 se puede ver un ejemplo de esta tarjeta con dos módulos FXO y dos módulos FXS.

Identificación de los puertos

Cada tarjeta tiene cuatro puertos RJ11. Cada uno de estos puertos se relaciona a un módulo y está numerado del 1 al 4. La información de módulos es importante para la posterior Configuración del hardware para su funcionamiento con Asterisk.

Existen varias Configuraciones estándar para la tarjeta TDM400P. Cada Configuración contiene de uno a cuatro módulos FXS y/o FXO. La tarjeta utilizada en este proyecto contiene cuatro módulos FXO de los cuales únicamente se utilizarán dos para las troncales analógicas de la central telefónica, dejando la opción para aumentar posteriormente 2 líneas de ser necesario.

Componentes de Software

Sistema Operativo

Para esta implementación se utilizó el Sistema Operativo Linux en la distribución Centos 5.

Software Servidor de Comunicaciones

Asterisk

Asterisk es una central telefónica IP por software, que se maneja con licencia GPL (Licencia Pública General) y permite el desarrollo de servidores de comunicaciones para empresas.

Es una central telefónica a la cual se puede conectar un número determinado de dispositivos para la realización de comunicaciones internas, permite además la Configuración de cuentas de proveedores internacionales de VoIP y la conexión a través de tarjetas con la red pública PSTN tanto con básicos como primarios y dispositivos de comunicaciones analógicos.

Asterisk fue creado por Mark Spencer de Digium y es actualmente desarrollado por esta empresa y los colaboradores de la comunidad que apoya el software de libre distribución.

Asterisk fue originalmente desarrollado para el sistema operativo GNU/Linux, pero en la actualidad funciona en BSD, MacOSX, y Solaris, aunque la plataforma nativa (GNU/Linux) es la mejor soportada de todas.

Asterisk incluye muchas características que anteriormente estaban disponibles únicamente en costosos sistemas propietarios, entre estas: buzón de voz, conferencias, IVR, distribución automática de llamadas, y más.

La principal característica de Asterisk es la apertura que da a los usuarios que pueden crear nuevas funcionalidades y personalizar su servidor de comunicaciones según las necesidades específicas de su empresa mediante el plan de marcación (dialplan) en el lenguaje de script de Asterisk o añadiendo módulos escritos en cualquier lenguaje de programación soportado por Linux. Lo cual revaloriza el conocimiento real de la plataforma de comunicaciones utilizada por encima del presupuesto de una empresa.

En Asterisk no existen cajas negras, ni soluciones imposibles llenas de licenciamientos para cada característica adicional que desee entregarse a los usuarios.

Asterisk soporta una variedad de protocolos para VoIP como SIP, H.323, IAX y MGCP y puede interactuar con terminales IP actuando como un registrador y como gateway entre ambos.

Para la conexión con dispositivos analógicos y digitales, es necesaria la adquisición de Hardware adicional que debe conectarse al servidor de comunicaciones, ejemplos son las tarjetas telefónicas FXS / FXO fabricadas por Digium, existen además opciones como Sangoma que fabrica tarjetas compatibles con Asterisk.

IMPLEMENTACIÓN DEL SERVIDOR DE COMUNICACIONES

Instalación del Sistema Operativo

El primer paso a seguir es la selección e instalación del Sistema Operativo, en el caso de esta implementación se optó por el sistema Linux en la distribución CentOS, muy similar a la distribución RedHat.

Para mayores detalles sobre la instalación del sistema operativo, Ver el Anexo número 4 Instalación de CentOS en Modo Texto.

Descarga de Asterisk

Asterisk actualmente cuenta con las versiones 1.2, 1.4, 1.6 además de AsteriskNOW con interfaz gráfica y Asterisk Business Edition, un producto comercial. Se optó por la versión 1.4 de Asterisk que se considera estable y permite una gran flexibilidad en el trabajo, además de ser gratuita.

Los paquetes utilizados en la implementación de este proyecto fueron los siguientes:

Asterisk asterisk-1.4.10.tar.gz

Zaptel zaptel-1.4.4.tar.gz

Libpri libpri-1.4.1.tar.gz

Asterisk-Addons asterisk-addons-1.4.2.tar.gz

Asterisk y sus librerías pueden descargarse como archivos .tar.gz desde un navegador web en la página <http://www.asterisk.org/downloads> o directamente desde una Terminal de Linux con el comando wget:

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

```
#wget http://downloads.digium.com/pub/asterisk/releases/asterisk-1.4.10.tar.gz
```

```
#wget http://downloads.digium.com/pub/zaptel/releases/zaptel-1.4.4.tar.gz
#wget http://downloads.digium.com/pub/libpri/releases/libpri-1.4.1.tar.gz
#wget http://downloads.digium.com/pub/asterisk/releases/asterisk-addons-1.4.2.tar.gz
```

Instalación de Asterisk

Una vez obtenidos los paquetes, es importante revisar que los siguientes paquetes de Linux se encuentren instalados:

- Kernel sources y headers
- gcc y sus dependencias preferentemente la versión 3.x o superior
- bison necesario para los archivos de Configuración
- ncurses para la consola o CLI
- openssl y openssl-dev o libssl-dev para las librerías criptográficas
- libnewt para aplicaciones de Zaptel como zttool

Para verificar si están instaladas las fuentes del kernel, se utiliza el comando `rpm -q kernel-devel`, debería verse algo como esto:

```
[root@asterisk ~]# rpm -q kernel-devel
kernel-devel-2.6.18-8.el5
```

Si no están instaladas es posible su instalación con yum de la siguiente forma:

```
yum install kernel-devel
```

Luego se deben ejecutar los siguientes comandos para verificar si se encuentran instalados los paquetes:

```
#rpm -q bison
#rpm -q bison-devel
#rpm -q ncurses
#rpm -q ncurses-devel
```

```
#rpm -q zlib  
#rpm -q zlib-devel  
#rpm -q openssl  
#rpm -q openssl-devel  
#rpm -q gnutls-devel  
#rpm -q gcc  
#rpm -q gcc-c++
```

Si alguno de estos faltase debe instalarse de la siguiente manera:

```
yum install nombre-del-paquete
```

Una vez instalados los paquetes previos, se puede iniciar la instalación de asterisk, la cual se debe realizar en el siguiente orden:

1. zaptel
2. libpri
3. asterisk
4. asterisk-addons

Se inicia con la instalación de Zaptel 1.4:

```
#tar zxvf zaptel-1.4.4.tar.gz  
#cd zaptel-1.4.4  
#./Configure  
#make menuselect
```

La opción Configure ajusta la instalación del paquete al sistema, y la opción menuselect ofrece una interfaz amigable que permite personalizar las opciones de compilación de Zaptel.


```

*****
Zaptel Module Selection
*****
Press 'h' for help.

---> 1. Kernel Modules
      2. Utilities

```

Figura 3. 3 Selección de Módulos Zaptel

Con esta herramienta se pueden seleccionar los módulos requeridos de acuerdo al hardware Zaptel instalado en el sistema y las herramientas o utilidades necesarias. Se debe presionar 'x' para grabar cambios y 'q' o la tecla escape para salir sin grabar. Los módulos seleccionados serán los instalados, y aquellas opciones que aparezcan marcadas con tres letras 'X' indican no disponibilidad debido a falta de dependencias como se muestra en la Figura 3.4

```

*****
Zaptel Module Selection
*****
Press 'h' for help.

[*] 1. fxotune
[*] 2. fxstest
[*] 3. sethdlc-new
[*] 4. ztcfg
[*] 5. ztdiag
[*] 6. ztmonitor
[*] 7. ztspeed
[*] 8. ztttest
XXX 9. ztttool

```

Figura 3. 4 Selección de Módulos Zaptel

Una vez seleccionados los módulos se deben escribir los siguientes comandos:

```
#make clean  
#make  
#make install
```

Eso concluye la instalación de Zaptel.

Para la instalación de libpri, los comandos son los siguientes:

```
#tar zxvf libpri-1.4.1.tar.gz  
#cd libpri-1.4.1  
#make  
#make install
```

En este punto, libpri debe estar disponible en el sistema.

La compilación e instalación de Asterisk tiene los siguientes pasos:

```
#tar zxvf asterisk-1.4.10.tar.gz  
#cd asterisk-1.4.10  
#./Configure  
#make menuselect
```

La herramienta menuselect, de forma similar que con Zaptel, permite personalizar las opciones de compilación escogiendo detalles para: Aplicaciones, CDR, Canales, Códecs, Intérpretes de formato, Funciones, Módulos de PBX, Banderas de Compilación Paquetes de archivos de sonidos, Paquetes de Música en espera y más.

```

*****
Asterisk Module Selection
*****
Press 'h' for help.

---> 1. Applications
      2. Call Detail Recording
      3. Channel Drivers
      4. Codec Translators
      5. Format Interpreters
      6. Dialplan functions
      7. PBX Modules
      8. Resource Modules
      9. Voicemail Build Options
     10. Compiler Flags
     11. Module Embedding
     12. Core Sound Packages
     13. Music On Hold File
         Packages
     14. Extras Sound Packages

```

Figura 3. 5 Selección de Módulos de Asterisk

Se puede continuar con la Configuración predefinida o navegar por estas opciones para personalizar la instalación de acuerdo a las necesidades de un sistema. Para el propósito de este proyecto se instalaron las opciones predefinidas,

Los comandos para concluir la instalación de Asterisk son:

```

# make
#make install
#make samples

```

Para instalar Asterisk-addons se debe ingresar el siguiente código:

```

#tar zxvf asterisk-addons-1.4.2.tar.gz
#cd asterisk-addons-1.4.2
#make clan

```

```
#make
```

```
#make install
```

Con esto se ha concluido la instalación y puede ejecutarse Asterisk en el sistema.

Instalación de la tarjeta TDM400P

Para la instalación de la tarjeta se deben seguir los siguientes pasos:

1. Apagar el computador y desconectarlo de la fuente de poder.
2. Conectar una pulsera anti estática a la muñeca y Abrir el case
3. Insertar la tarjeta en un slot PCI de 3.3 o 5 voltios (PCI 2.2 o mayor)
4. Colocar la cubierta del computador
5. Conectar las líneas telefónicas a los puertos FXO.

Software de la tarjeta

La tarjeta funciona únicamente con Linux, el fabricante recomienda las distribuciones Debian, Fedora, y RedHat. El hardware Digium necesita drivers y librerías que no se han integrado con el kernel de Linux por lo que debe obtenerse el código fuente.

Los requerimientos mínimos son:

Kernel Linux 2.4 o posterior.

Bison 1.84

zlib, libedit, y openssl.

1. El primer paso es revisar la compatibilidad del Bus PCI. Se debe iniciar el computador en Linux y ejecutar el siguiente comando:

```
#lspci
```

Debe identificarse un dispositivo Tiger Jet o Jens Shoenfeld. Si ninguno de los dos aparece, la máquina no es compatible con PCI 2.2 y la tarjeta no funcionará.

2. Obtener e instalar los drivers Zaptel.
3. Cargar los drivers Zaptel en el kernel mediante modprobe, el driver es wctdm.

```
#modprobe wctdm
```

4. Configurar la tarjeta abriendo el archivo zaptel.conf en el directorio /etc

```
vi /etc/zaptel.conf
```

Agregar para los módulos FXO:

```
fxsks=1,2,3,4
```

Escoger un defaultzone para el país. Para Ecuador la tarjeta funciona correctamente con la Configuración predefinida. Guardar el archivo y regresar a la línea de comandos.

```
#modprobe wctdm
```

```
#ztcfg -vv
```

5. Confirmar la Configuración de la tarjeta con un el comando dmesg. El texto debe ser similar al que se presenta a continuación:

```
Found a Wildcard TDM: Wildcard TDM400P REV E/F (4 modules)
Freshmaker version: 71
Freshmaker passed register test
Module 0: Installed – Auto FXO (FCC mode)
Module 1: Installed – Auto FXO (FCC mode)
Module 2: Installed – Auto FXO (FCC mode)
Module 3: Installed – Auto FXO (FCC mode)
Found a Wildcard TDM: Wildcard TDM400P REV E/F (4 modules)
```

Registered tone zone 0 (United States / North America)
Registered Tone zone 0 (United States / North America)
ide-floppy driver 0.99.newide

CONFIGURACIÓN DE LA CENTRAL TELEFÓNICA

La Configuración realizada en el Servidor de Comunicaciones, busca cuatro objetivos básicos.

1. Comunicar todas las extensiones internas (dispositivos SIP) entre sí y brindarles servicios básicos como buzón de mensajes.
2. Recibir y encaminar llamadas entrantes desde las troncales analógicas, mediante el uso de una contestadora automática.
3. Permitir la salida de llamadas externas a través de las troncales adecuadas dependiendo del tipo de llamada (local, nacional o celular)
4. Autorizar y denegar la salida de llamadas de acuerdo a los permisos de cada usuario.

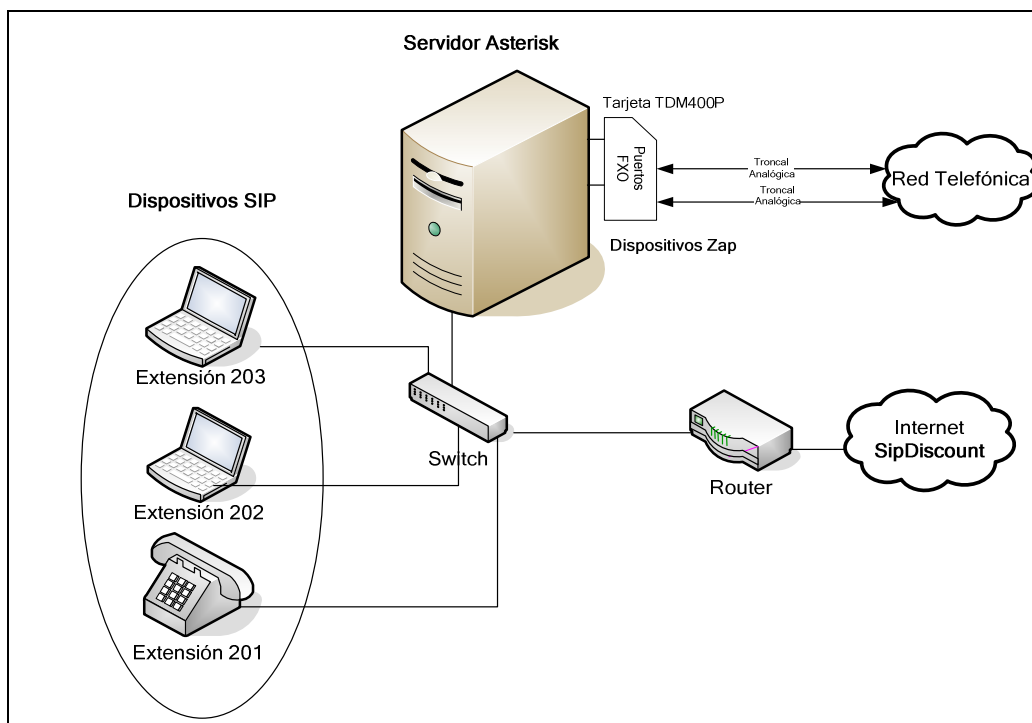


Figura 3. 6 Diagrama de Dispositivos de Comunicaciones

Para cumplir estos objetivos, en la Configuración se han creado los canales que los dispositivos van a utilizar y se desarrolló un Plan de Marcación con varios contextos, para el enrutamiento de las llamadas.

Configuración de Canales

Un Canal es una forma de comunicación entre los recursos internos y/o externos como las líneas telefónicas analógicas o digitales, teléfonos IP o extensiones físicas del sistema. Cada llamada controlada por el sistema viaja por su propio canal.

Canales ZAP

Son los canales usados por los puertos FXS/FXO, PRI y BRI y corresponden al hardware Zaptel. Los parámetros se definen en el archivo zapata.conf y a continuación se nombran algunos de los más importantes.

Language. Lenguaje por defecto para canales ZAP.

Context. Contexto por defecto al que se envían llamadas que entran a un canal zap.

Switchtype. Tipo de switch, este parámetro se utiliza para PRI.

Overlapdial, Modo de marcación traslapado.

Signaling. Método de señalización.

Toneduration. Define la duración de tonos DTMF y MF reproducidos sobre el canal (en milisegundos).

Canales SIP

Los canales SIP ¹son los usados por los dispositivos que utilizan este protocolo, por ejemplo teléfonos IP, teléfonos en software, y clientes de proveedores internacionales de servicio de VoIP.

Se Configuran en el archivo sip.conf, en donde se definen opciones generales a todos los dispositivos, como la dirección IP y el número de puerto del servidor, y características individuales de los dispositivos en sus propias secciones donde se definen parámetros como usuario, contraseña y dirección IP para clientes no registrados.

Configuración sip.conf usada en el proyecto

```
[general]
bindport=5060
context=default
language=es
disallow=all
allow=ulaw

context=entrante
register => miusuario:micontraseña@sip1.sipdiscount.com

; Inclusion del archivo de dispositivos
#include sip_dispositivos.conf

;Ejemplo de un dispositivo

[201]
type=friend
host=dynamic
context=supervisor
username=201
secret=201
callerid= Daniel Verdesoto <201>
dtmfmode=rfc2833
mailbox=201@supervisor
callgroup=1
pickupgroup=1
```

¹ SIP Session Initiation Protocol – Protocolo de Inicio de Sesiones

Configuración de una cuenta con SipDiscount para llamadas internacionales

Una opción para permitir el ahorro en las llamadas internacionales es el uso de un proveedor internacional de Voz sobre IP mediante una cuenta Configurada como un canal SIP en el servidor de comunicaciones Asterisk.

El proveedor seleccionado fue SipDiscount que permite la creación de una cuenta que puede usarse desde cualquier cliente SIP en el que se Configuren el nombre de usuario y contraseña, y permite la conexión desde la central IP Asterisk.

En la página www.sipdiscount.com se puede adquirir un usuario y una contraseña, además de saldo para las llamadas, también es posible obtener información sobre las tarifas aplicables para los distintos destinos.

Los datos proporcionados por el proveedor y necesarios para la Configuración de la cuenta SIP son los siguientes:

Usuario: miusuario

Contraseña: micontraseña

SIP/Proxy registrar: sip1.sipdiscount.com

Domain/Realm: sip1.sipdiscount.com

STUN server: stun.sipdiscount.com

Con estos datos, se procede a crear la cuenta dentro del archivo sip.conf de la siguiente manera:

```
[general]
```

context=entrante

register => miusuario:micontraseña@sip1.sipdiscount.com

En el archivo sip_dispositivos.conf, se crea el dispositivo:

```
[sip1.sipdiscount.com]
```

```
type=friend
```

```
host=sip1.sipdiscount.com
```

```
progressinband=yes
```

```
fromuser=miusuario
```

```
authuser=miusuario
```

```
username=miusuario
```

```
secret=micontraseña
```

```
qualify=yes
```

```
disallow=all
```

```
allow=ulaw
```

```
dtmfmode=rfc2833
```

```
canreinvite=no
```

```
nat=yes
```

```
context=incoming-sipdiscount
```

```
fromdomain=sip.sipdiscount.com
```

```
insecure=port,invite
```

PLAN DE MARCACIÓN

El plan de marcación tiene a su cargo enrutar cada llamada en el sistema, es la parte central en el sistema de telefonía y especifica la manera en que se maneja el flujo entrante y saliente de llamadas.

La Configuración del plan de marcación se realiza en los siguientes archivos-

Extensions.conf. Plan de numeración de extensiones estándar.

Extensions.ael. Plan de numeración de extensiones en sintaxis AEL.

Features.conf. Configuración de características adicionales como extensiones de parqueo, grabación por demanda y otras.

ExtConfig.conf. Utilizado para cargar datos desde una base de datos por medio de la tecnología Asterisk Real Time (ARA)

En este proyecto, el plan de marcación se centra en el archivo extensions.conf, al cual se ha dividido en varios archivos de texto más pequeños para separar las principales funciones con el objetivo de dar un mayor orden a la implementación. Estos son:

extensions.conf: contiene la Configuración de la contestadora automática.

extensions_interno.conf: en este archivo se encuentra la Configuración de llamadas y características internas.

extensions_saliente.conf: con la Configuración para llamadas salientes.

extensions_macros.conf: todas las macros utilizadas en el plan de marcación.

ivr_pruebas.conf: Configuración del IVR.

Contextos

Los contextos definen la organización y alcance de la información o parámetros definidos en cada uno de ellos. Pueden utilizarse para que un sistema conteste de forma distinta cada línea telefónica, o para separar llamadas de diferentes canales, con lo que es posible separar las llamadas internas, entrantes, salientes.

Extensiones

Dentro de cada contexto se definen una o más extensiones, las cuales determinan el flujo de la llamada, de esta forma es posible programar en cada contexto un comportamiento de acuerdo al número marcado, el cual puede tomarse como una extensión interna o como una variable.

Contextos utilizados en la Implementación

Contexto [entrante]

En este contexto se encuentra definido el comportamiento de la central telefónica ante llamadas que ingresan por las troncales analógicas. Es decir, cuando un cliente llama a la central será atendido por esta Configuración.

En el contexto entrante, se define la contestadora automática, la cual consiste en una bienvenida, y un menú.

El menú dirige al cliente a una de las siguientes opciones, según el número marcado:

1. A la extensión del supervisor
2. Al primer Call Centre de pruebas
3. Al segundo Call Centre de pruebas
4. Para ingresar a una multi-conferencia.
5. Para el IVR de pruebas.

Contextos: [interno]; [supervisor]; [espiar].

Estos contextos manejan las llamadas internas, entregan a los usuarios servicios como correo de voz, y dentro del proyecto permiten acceder directamente a opciones como el IVR de pruebas o para comprobar el funcionamiento de Cepstral.

Contexto [interno]

Dentro de este contexto se definen las extensiones utilizadas por todos los usuarios, sean estos agentes o supervisores. Es aquí donde se indica a la PBX qué hacer cuando un usuario interno marca un número desde una de las extensiones.

Extensión Corresponde a

_2XX	Extensión, desde la 200 hasta la 299 están asignadas a dispositivos SIP para los usuarios.
700, 701	Números directos para las colas.
600	Extensión para conferencias
123	Extensión de pruebas para servidor TTS Festival
124	Extensión de pruebas para TTS con Cepstral
125	Extensión directa para pruebas del IVR
*96	Extensión para registro de Agentes
*98	Extensión del Buzón de Mensajes
_66XX	Números del 6600 al 6699 utilizados para grabaciones de voz.

22255516 Número virtual para ingreso al contexto entrante, para la simulación de llamadas de clientes.

_XXXXXX Cualquier número de 6 dígitos, utilizado como clave para permitir la salida de llamadas.

Contexto [espiar]:

321 Número entregado a los supervisores para espiar llamadas conectadas con troncales analógicas.

Contexto [supervisor]

Incluye los contextos interno y espiar.

Contexto [salientes]

Este es el contexto encargado de la realización de llamadas externas. Tras la autenticación del usuario interno con su clave de seis dígitos en el contexto [interno], este es dirigido a este contexto, en donde el sistema le da la bienvenida y solicita el número al que se desea salir.

Una vez obtenido el dato, el sistema se dirige hacia la macro marcar, llevando el número a marcar, y el dato de autorización entregado por AstDB, además de la troncal a utilizar para la llamada.

Los patrones utilizados son los siguientes:

Llamadas locales _NXXXXXX

Llamadas a celular _09XXXXXXXX
 _08XXXXXXXX

Llamadas nacionales _02XXXXXXXX
 _03XXXXXXXX
 _04XXXXXXXX
 _05XXXXXXXX

_06XXXXXXXX
_07XXXXXXXX

Llamadas internacionales _00X.

Macros

Además de estos contextos, se crearon macros que realizan funciones específicas y repetitivas, por lo que permiten reducir de forma considerable el código en el plan de marcación. Estas se encuentran en el archivo `extensions_macros.conf` y son las siguientes:

Macro `extenvm ${ARG1}`

Se encarga del marcado a extensiones internas. Dirige la llamada al dispositivo y, de no recibir contestación, envía la llamada al Buzón de Mensajes.

Recibe un parámetro `${ARG1}`, en el cual se encuentra la extensión a marcar y la tecnología, ejemplo: SIP/201

Código de la macro:

[macro-extenvm]

```
exten => s,1,Dial(${ARG1},20)
exten => s,2,Voicemail(u${MACRO_EXTEN})
exten => s,3,Hangup
exten => s,102,Voicemail(b${MACRO_EXTEN})
exten => s,103,Hangup
```

Macro `colas ${ARG1}`

Cuando una llamada se dirige a una de las colas, esta es la macro encargada, direcciona la llamada a la cola correcta y da una indicación por medio de un beep.

Recibe un parámetro, `${ARG1}`, el cual indica la cola a marcar, ejemplo:
`${COLA1}`

Código de la macro:

[macro-colas]

exten => s,1,Queue(\${ARG1})|tT|||20)

exten => s,n,Playback(beep)

exten => s,n,Hangup()

Macro marcar

Esta macro se encarga del control de llamadas salientes. Comprueba que el usuario tenga autorización para realizar la llamada leyendo la variable de autorización entregada por el contexto [interno] y leído desde AstDB, limpia estas variables y realiza la llamada.

Recibe 4 argumentos:

`\${ARG1}` corresponde a la autorización para realizar la llamada por parte del usuario, el valor puede ser 1 o 0, para llamadas locales, corresponde a **`\${LOCAL}`**, para llamadas nacionales **`\${NACIONAL}`**, para llamadas a celular **`\${CELULAR}`**, internacionales **`\${INTERNACIONAL}`**

`\${ARG2}` en este argumento, se debe indicar la troncal a utilizar para la llamada, por ejemplo: **`\${TRONCAL2}`**

`\${ARG3}` en el tercer argumento, se indica el número a marcar, por ejemplo: **`\${EXTEN:1}`; 911; `\${EXTEN}`**

`\${ARG4}` el cuarto argumento, indica si la llamada es internacional, para diferenciar el tipo de troncal, y utilizar la cuenta de sipdiscount.

Código de la macro:

[macro-marcar]

exten => s,1,GotIlf(\$[\${ARG1} = 1]? 2:i,1)

```

exten => s,2,Set(CDR(accountcode)=${NOMBREM})
exten => s,3,Macro(borrar_variables)
exten => s,4,GotoIf($[${ARG4} = 1]? 5:6)
exten => s,5,Dial(${ARG2}/${ARG3})
exten => s,6,Dial(SIP/${ARG3}@sip1.sipdiscount.com)
exten => s,106,Dial(${ARG2}/${ARG3})
exten => s,107,Congestion()

exten => i,1,Macro(borrar_variables)
exten => i,n,Playback(llamada_no_autorizada)
exten => i,n,Hangup()

```

Macro borrar_variables

Esta macro tiene como única función llevar las variables LOCAL, NACIONAL, CELULAR, INTERNACIONAL, NOMBRE al valor cero para evitar la realización de llamadas por parte de usuarios no autorizados, no recibe argumentos.

Código de la macro:

```
[macro-borrar_variables]
```

```

exten => s,1,SetGlobalVar(LOCAL=0)
exten => s,n,SetGlobalVar(NACIONAL=0)
exten => s,n,SetGlobalVar(CELULAR=0)
exten => s,n,SetGlobalVar(INTERNACIONAL=0)
exten => s,n,SetGlobalVar(NOMBREM=0)

```

INTERFAZ DE LÍNEA DE COMANDOS DE ASTERISK

La interfaz de Asterisk (Asterisk CLI) permite llevar a cabo operaciones de monitoreo y control de la central telefónica.

Para iniciar esta aplicación se debe ingresar el siguiente texto en una Terminal de Linux:

```
#asterisk -vvrc
```

Cada letra v indica un grado de verbosidad, es decir, la cantidad de datos que se van a mostrar con respecto al funcionamiento actual de la central.

Al iniciarse asterisk, se verá la siguiente pantalla:

```
ER again (after 20 seconds)
ServidorAsterisk*CLI>
```

Figura 3. 7 Interfaz de Línea de Comandos de Asterisk

En la cual puede ingresarse cualquiera de los comandos del Anexo 6, los cuales permiten administrar el servidor de comunicaciones.

Esta interfaz además muestra una serie de mensajes que indican las acciones que desarrolla el servidor de comunicaciones con lo cual es posible depurar el plan de marcación desarrollado.

BASE DE DATOS DE ASTERISK

Asterisk trae integrada una Base de Datos denominada AstDB, la cual permite el ingreso y procesamiento de información dentro de la central telefónica, para una variedad de usos. En el presente proyecto, se utilizó AstDB para almacenar la información de clientes externos y usuarios de la central telefónica, los primeros para el desarrollo del IVR y los segundos para la definición de permisos para la salida de llamadas locales, nacionales, a celular e internacionales.

Desde la línea de comandos de Asterisk, se puede trabajar la base de datos con los siguientes comandos:

database del. elimina un valor de la base de datos

database deltree. elimina valores/arbol de la base de datos

database get. obtiene un valor de la base de datos

database put. Agrega/actualiza un valor de la base de datos

database show. muestra los contenidos de la base de datos

database showkey. muestra contenidos de la base de datos de manera más específica.

Uso de AstDB en el proyecto

Para el funcionamiento de la central telefónica del proyecto, la base de datos de Asterisk se utilizó en dos partes principales, la base de datos de clientes externos y la base de datos de los usuarios.

Base de datos de Clientes Externos

La base de datos de los clientes consta de los siguientes valores: apellido, clave, nombre, saldo, sexo.

Para el ingreso de campos en la base de datos, se utilizó el comando database put de la siguiente manera:

```
database put familia clave valor
```

Ejemplo:

```
database put 1708712912 apellido Montero
```

```
database put 1708712912 nombre Carlos
```

```
database put 1708712912 clave 1708
```

```
database put 1708712912 sexo m
```

```
database put 1708712912 saldo 30
```

Base de datos de Usuarios

En esta tabla se almacenó la información sobre claves, nombres y permisos de salida de los usuarios de la central telefónica para los distintos tipos de llamada, el valor 0 significa *no autorizado*, el valor 1 significa *autorizado*.

Ejemplo:

database put 111110 nombre danielverdesoto

database put 111110 local 1

database put 111110 celular 0

database put 111110 internacional 1

database put 111110 nacional 0

CAPÍTULO 4

CARACTERÍSTICAS DE CALL CENTRE

CONFIGURACIONES EN ASTERISK

En Asterisk existen dos archivos importantes para la Configuración cuando la implementación es un Call Centre, estos son: `agents.conf` y `queues.conf` en los cuales se detallan los agentes y las colas respectivamente.

CONFIGURACIÓN DE LOS AGENTES

En el archivo `agents.conf` se crean los Agentes. Estos son usuarios que pueden Configurarse para atender las llamadas entrantes en las colas.

Los agentes pueden ser separados en distintos contextos, para de esta manera tener una mayor flexibilidad en la administración de las colas, ya que es posible escoger que agente debe trabajar en qué cola.

En `agents.conf` existen dos tipos de contextos: `[general]` y `[agents]`. Los nombres de los contextos no pueden cambiarse.

Opciones en el contexto `[general]`

Persistentagents Esta opción define si los registros de los agentes deben guardarse en la base de datos de Asterisk o no. Si se coloca el valor 'yes' entonces se guardará la información de registro y en el caso de un reinicio de Asterisk, los agentes continuarán registrados.

Opciones en el Contexto `[agents]`

Las siguientes opciones se encuentran disponibles:

Autologoff. Con esta opción se indica por cuanto tiempo el teléfono debe timbrar sin respuesta, antes de que se haga un log off al agente. El tiempo se indica en segundos.

Ackall. Esta opción obliga a los agentes a usar # durante el registro.

Wrapuptime. Esta opción se refiere al tiempo que debe transcurrir después de que una conversación termina para que una nueva llamada ingrese.

Musiconhold. Con esta opción se puede determinar la clase de música en espera para los agentes.

Updatecdr. Con esta opción se define si se desea cambiar el originador de una llamada en el CDR o no.

Group. Esta opción permite agrupar los agentes en distintos grupos para facilidad de administración.

Para asignar un agente a un grupo se lo debe agregar después del texto group=número del grupo.

Recordedagentcalls. Esta opción permite la grabación de llamadas de un agente.

Recordformat. Para indicar el formato del archivo de audio.

Createlink. Esta opción permite insertar el nombre de la grabación en el CDR.

Urlprefix. Esta opción agrega un texto al nombre de la grabación. Permite que sea un URL.

Savecallsin. Permite cambiar el directorio donde se guardan los archivos de grabación. El valor por defecto es /var/spool/asterisk/monitor.

Creación de agentes:

Los agentes se agregan de la siguiente forma:

agent => número,contraseña,nombre

número es el número correspondiente al agente.

contraseña es la contraseña del agente, que va a ser solicitada durante el registro con las aplicaciones AgentLogin o AgentCallbackLogin.

nombre es el nombre real del agente.

Configuración de agentes en el Proyecto Implementado:

[general]

persistentagents=yes

[agents]

ackall=no

wrapuptime=5

updatecdr=yes

recordedagentcalls=yes

agent => 5001,1231, Daniel Verdesoto

agent => 5002,1232, Eric Verdesoto

agent => 5003,1233, Maribel Verdesoto

agent => 5004,1234, Virginia Rodríguez

agent => 5005,1235, Karen Verdesoto

agent => 5006,1236, Pablo Coba

agent => 5007,1237, Cristóbal Cordero

agent => 5008,1238, Freddy Garzón

agent => 5009,1239, Paulina Erazo

agent => 5010,1240, Karina Salazar

CONFIGURACIÓN DE LAS COLAS

Las colas son útiles ya que permiten entregar un servicio más profesional y resultan imprescindibles en el ámbito de un call centre. Gracias a las colas, el sistema puede responder a cada llamada de forma inmediata sin importar que existan operadores o agentes disponibles. Si todas las líneas se encuentran ocupadas, la llamada puede mantenerse en una cola hasta que sea posible su atención por parte del próximo agente disponible.

El archivo `queues.conf` se utiliza para crear colas en Asterisk. Dentro de este archivo existen dos tipos de contextos. El primero es el contexto `[general]` y su nombre no puede ser cambiado. Los nombres de los otros contextos pueden ser arbitrarios. Al igual que en los otros archivos de Configuración de Asterisk, el inicio de un contexto es su nombre (entre corchetes) y su fin viene indicado por el siguiente contexto.

Opciones del contexto `[general]`

Persistentmembers. Si esta opción se Configura con 'yes', esto va a provocar que en el sistema se almacene cada agente registrado dinámicamente en una cola en la base de datos de Asterisk. De esta manera, en caso de reiniciar la PBX Asterisk, los agentes van a colocarse automáticamente en sus colas. Por defecto, esta opción está Configurada con 'yes'.

Autofill. (valor por defecto 'no') el comportamiento de una cola es tener un comportamiento serial, en el que la cola va a mantener a todos los clientes en espera hasta que el primer cliente sea atendido aun si existen varios agentes disponibles. Una vez que el primer cliente es atendido, el siguiente cliente en la cola se vuelve la cabeza y es conectado con el siguiente agente disponible los clientes en espera no pueden ser conectados aun cuando existan agentes disponibles. El activar autofill, hace posible que los clientes sean conectados a los agentes de forma paralela hasta que no hayan agentes disponibles o clientes en la fila.

monitor-type. Especifica un formato de monitoreo para permitir el grabado de conversaciones de los agentes, app_queue va a utilizar MixMonitor o Monitor.

Opciones de los Contextos de las colas:

Las siguientes opciones se encuentran disponibles:

music : indica la música que escucharán los clientes en espera dentro de la cola de llamadas

announce: con esta opción se puede dar un anuncio al agente informándole desde qué cola le llega la llamada, es útil para agentes que son miembros de más de una cola de llamadas.

strategy: con esta opción se puede definir la forma en que las llamadas entrantes se distribuyen a los agentes. Las siguientes estrategias están disponibles:

ringall: timbra a todos los agentes disponibles en la cola hasta que uno contesta.

roundrobin: timbra a cada agente disponible uno por uno.

leastrecent: timbra al agente que no ha recibido ninguna llamada en el mayor tiempo.

fewestcalls: timbra al agente que ha atendido la menor cantidad de llamadas.

random: timbra a los agentes de forma aleatoria.

rrmemory: es un round robin con memoria, recuerda el último agente al que entregó una llamada e intenta timbrar al siguiente.

servicelevel: esta opción se utiliza para estadísticas de nivel de servicio. Se Configura un período de tiempo en el cual las llamadas deben contestarse.

context: con esta opción se puede especificar un contexto (del plan de marcación) al cual el usuario puede ser transferido en caso de presionar un dígito. Si el cliente presiona un dígito, va a ser sacado de la cola y llevado a la extensión correspondiente al dígito presionado dentro de ese contexto.

timeout: cuanto tiempo puede permanecer timbrando, antes de considerar que la llamada no va a ser contestada.

retry: cuanto tiempo se debe esperar antes de iniciar el timbrado en otro agente.

weight: cuando un canal se encuentra incluido en mas de una cola, la cola con el menor peso va a ser la primera en manejar una llamada entrante desde ese canal.

wrapuptime: esta opción indica un período de tiempo en segundos, que debe expirar después de que una llamada ha sido terminada, antes de que otra llamada se pase al agente.

autopause: coloca a un agente en pausa cuando no ha atendido una llamada.

maxlen: se puede determinar el número máximo de llamadas que pueden estar en espera dentro de la cola.

setinterfacevar: si se coloca en 'yes', antes de que el cliente se conecte al agente, en la variable MEMBERINTERFACE se escribe el nombre del agente (ejemplo: Agent/5001).

announce-frequency: esta opción define cada cuanto tiempo el sistema debe anunciar la posición actual del cliente en la cola.

periodic-announce-frequency: cada cuanto tiempo realizar un anuncio periódico.

announce-holdtime: al habilitar esta opción, se entrega al cliente anuncios del tiempo de espera estimado. Existen los valores posibles 'yes', 'no' y 'once'.

announce-round-seconds: el sistema anunciará los segundos y minutos para este valor.

periodic-announce: Todos los agentes se encuentran ocupados/ espere al próximo.

Estos sonidos se encuentran en los directorios `/var/lib/asterisk/sounds` y `/usr/src/asterisk/sounds` y pueden cambiarse de acuerdo a la necesidad del usuario.

monitor-format: esta opción tiene dos funciones, la primera es habilitar el monitoreo de conversaciones, la segunda es especificar el formato deseado para los archivos en los que se van a guardar las conversaciones. El nombre del archivo se puede definir con la variable `#{UNIQUEID}`. Los formatos posibles son `gsm`, `wav` y `wav49`.

monitor-join: cuando esta opción se coloca en 'yes', los archivos en que la conversación es grabadas, se mezclan en un solo archivo de forma automática.

joinempty: con esta opción se determina si un cliente puede o no entrar a una cola en donde no hay agentes registrados.

leavewhenempty: con esta opción se puede permitir retirar clientes de una cola si hay nuevos clientes que no pueden ingresar a la cola.

eventwhencalled: con esta opción se activa la generación de los siguientes eventos: `AgentCalled`, `AgentDump`, `AgentConnect` y `AgentComplete`.

eventmemberstatus: genera eventos extra como `QueueMemberStatus`.

reporholdtime: informa al agente que responde una llamada cuanto tiempo ha permanecido en la cola el cliente al que va a atender.

memberdelay: se puede dar un período de tiempo de silencio antes de que el agente se conecte al cliente.

ringinuse: para evitar que una cola envíe llamadas a miembros cuyos equipos están ocupados en ese momento.

timeoutrestart: si esta opción está habilitada, y se envía una señal de Busy o Congestion, el tiempo para contestar de un agente se resetea.

Configuración de colas desarrollada en la implementación:

[general]

persistentmembers = yes

autofill = yes

[cola-ventas]

music=default

announce = anunciouv

strategy = rrmemory

timeout = 15

retry = 5

wrapuptime = 15

maxlen = 0

member = Agent/5001

member = Agent/5002

member = Agent/5003

member = Agent/5004

member = Agent/5005

[cola-tecnico]

```
music=default  
announce = anunciot  
strategy = rrmemory  
timeout = 15  
retry = 5  
wrapuptime = 15  
maxlen = 0  
member = Agent/5006  
member = Agent/5007  
member = Agent/5008  
member = Agent/5009  
member = Agent/5010
```

CONFIGURACIÓN DEL IVR

Dentro de los elementos importantes en la estructura del Call Centre se encuentra la respuesta interactiva de voz (IVR), la cual se obtuvo mediante programación dentro del plan de marcación y la grabación de un conjunto de mensajes similares a los de una contestadora automática.

La característica que diferencia al IVR de una simple contestadora está en la interacción personal e individualizada que se consigue tras identificar al cliente e interactuar con una base de datos, para de esta forma cumplir con tareas que pueden resultar repetitivas y no requieren la atención de un agente especializado, tales como la lectura de un saldo o un cambio de clave.

Dentro de este IVR se ha integrado la entrega de mensajes mediante un motor de TTS (texto a voz digital) para la lectura de datos como el nombre del cliente.

Contextos: [ivr]; [ivr-autenticado]; [cambio-clave]

Este grupo de contextos maneja el IVR y ha sido separado por menús.

Contexto [ivr]

Este contexto maneja el menú principal del IVR, en el cual el usuario debe elegir entre autenticarse como un usuario, salir del IVR, o salir del sistema.

Si se escoge la autenticación como usuario, debe ingresarse el número de cédula, el cual será buscado en la base de datos de Asterisk. De existir este número de cédula en el sistema, será solicitada una clave personal. Si la clave ingresada es correcta, el usuario ingresa al contexto [ivr-autenticado], caso contrario el sistema le informará que ha ingresado datos incorrectos.

Contexto [ivr-autenticado]

Una vez identificado el usuario por su cédula y clave personal, el IVR da la bienvenida al usuario y pronuncia su nombre utilizando TTS (Cepstral), e inmediatamente se inicia un nuevo menú, en el cual el cliente puede:

1. Averiguar el saldo que se encuentra a su favor dentro de la base de datos,
2. Realizar un cambio de clave, para lo cual debe ingresar nuevamente su clave personal, luego es redirigido al contexto [cambio-clave].
3. Salir al menú anterior para autenticarse como un usuario distinto,
4. Terminar la llamada telefónica.

Para opciones inválidas, se le informa al usuario de su falla y la llamada es concluida.

Contexto [cambio-clave]

Cuando en el contexto [ivr-autenticado] el usuario ha elegido la opción cambio de clave y ha entregado su clave antigua de manera correcta, puede ingresar a este contexto, en el cual el sistema solicita una nueva clave de cuatro dígitos, y pide una confirmación de la misma. Si el cliente ha introducido de forma correcta su nueva clave las dos ocasiones, se guarda la misma en AstDB, y se informa que el cambio ha sido exitoso, caso contrario, se informa que

existió un error y el cliente es regresado al menú personal en el contexto [ivr-autenticado]

INSTALACIÓN Y CONFIGURACIÓN DEL MOTOR TTS

Para agregar funciones de Texto a Voz (TTS por sus siglas en Inglés), se realizaron pruebas con Festival, un motor TTS de código abierto, y Cepstral un motor económico de mejor calidad.

Festival

Configuración de Festival para Asterisk

Se agregó el siguiente texto al archivo de Configuración festival.scm ubicado en /usr/share/festival

```
(define (tts_textasterisk string mode)
"(tts_textasterisk STRING MODE)
(let ((wholeutt (utt.synth (eval (list 'Utterance 'Text string))))))
(utt.wave.resample wholeutt 8000)
(utt.wave.rescale wholeutt 5)
(utt.send.wave.client wholeutt)))
```

Configuración de Asterisk para Festival

El archivo de Configuración de Asterisk que se encarga de Festival tiene como nombre festival.conf, dentro del mismo se debe especificar: hostname y puerto del servidor festival, se utilizaron las Configuraciones predeterminadas.

Inicio del Servidor Festival

Para iniciar el servidor, se debe ingresar el siguiente código en la terminal:

```
[root@asterisk ~]# festival --server --language spanish
```

Utilización desde el Plan de Marcación

Una vez que el servidor Festival se encuentra funcionando y no rechaza las conexiones, puede iniciarse y ser llamado desde el plan de marcación, es capaz de leer textos o variables. La aplicación realizada para las pruebas se encuentra en el contexto Interno y se encarga de contestar, dar la bienvenida a la persona que llama, lee su nombre desde el ID de llamada y su número de extensión.

```
exten => 123,1,Answer()
exten => 123,n,Playback(bienvenidom)
exten => 123,n,Festival(${CALLERID(name)})
exten => 123,n,Playback(suextension)
exten => 123,n,SayNumber(${CALLERID(num)})
exten => 123,n,Wait(2)
exten => 123,n,Hangup()
```

Cepstral

Configuración con Cepstral

Después de la realización de Pruebas, el motor TTS Festival funcionó junto con Asterisk, pero la calidad de la voz entregada resultó bastante pobre y robotizada, por lo que se optó por la adquisición del motor TTS comercial Cepstral, el cual permite descargas gratuitas de prueba, se descargó una voz en castellano desde el sitio <https://www.cepstral.com/downloads/> y una vez demostrada su calidad, se adquirió una licencia para el uso de este motor.

Una vez descargada la aplicación, esta se instala de manera sencilla mediante el script "install.sh" (como sudo o root) y puede llamarse desde Asterisk mediante el código siguiente:

```
exten => XXX,n,System(/usr/local/cepstral/bin/swift -o /tmp/swift.wav "Texto")
exten => XXX,n,Playback(/tmp/swift)
exten => XXX,n,System(rm /tmp/swift.wav)
```

El código implementado para realizar pruebas fue:

```
exten => 124,1,Answer()
exten => 124,n,System(/usr/local/cepstral/bin/swift -o /tmp/swift.wav "Bienvenido,
" ${CALLERID(name)} ", su numero de extension es , " ${CALLERID(num)})
```

```

exten => 124,n,Playback(/tmp/swift)
exten => 124,n,System(rm /tmp/swift.wav)
exten => 124,n,Hangup()

```

Este tuvo un resultado bueno, por lo que fue este el motor TTS seleccionado para las interacciones de la base de datos con los clientes.

Gracias a esta función, cuando un cliente realiza una consulta en la base de datos, el sistema es capaz de darle la bienvenida, y leer su nombre, apellido y saldo sin necesidad de poseer grabaciones de audio con los datos de cada cliente almacenado

....

```

exten => s,2,Playback(bienvenidom)
exten => s,3,Goto(5)
exten => s,4,Playback(bienvenidaf)
exten => s,5,System(/usr/local/cepstral/bin/swift -o /tmp/swift.wav ${NOMBRE}
${APELLIDO})
exten => s,6,Playback(/tmp/swift)
exten => s,7,System(rm /tmp/swift.wav)
exten => s,8,Background(menu_personal)
...

```

INTERFAZ WEB PARA OPERADORA Y MONITOREO DE LA CENTRAL IP

Para tener una visualización más sencilla de las distintas actividades que realiza la central telefónica IP, así como la actividad en las colas, las troncales utilizadas, los agentes ocupados o en espera de llamadas, el número de personas dentro de la sala de conferencias, etc. Se instaló una interfaz gráfica web que a través de un servidor Apache permite el monitoreo y ciertas actividades de operadora.

El Software seleccionado fue Flash Operator Pannel, que puede ser obtenido de manera gratuita y con licencias GNU en la página www.asternic.org

Utilización del Panel

Una vez instalado y Configurado el panel (ver Anexo 2 Configuración de Flash Operator Panel), procedemos a su utilización.

Se debe escribir desde un navegador web el directorio en donde se encuentran colocados los archivos html y swf. En esta implementación, la dirección es: 192.168.0.104/panel/flash

En un momento inicial, la página obtenida es la de la Figura 4.1.



Figura 4. 1 Panel Flash correspondiente a la Central Telefónica Implementada

Cuando un canal se encuentra disponible, y no hay nadie en la cola o conferencia, el LED será verde, si un canal está ocupado, o hay alguien en una conferencia o cola, será rojo. Un LED titilando significa que un canal está timbrando.

Una vez que los agentes comienzan a registrarse, los nombres van a aparecer en el lugar del agente que les corresponde, así como también en la extensión en la que se han registrado.

En la Figura 4.2. pueden observarse los nombres de tres agentes registrados, en los que además el LED de estados ha cambiado al color amarillo

para indicar que se encuentran disponibles, y se iniciaron temporizadores que indican el tiempo que cada agente lleva esperando llamadas entrantes.



Figura 4. 2 Panel de Operador con agentes registrados

En el siguiente gráfico se puede observar que el agente “Daniel” realiza una llamada, y esta ingresa a la cola de ventas, a la extensión del agente “Pablo”. Así de manera gráfica es posible monitorear las actividades de la central telefónica y sus principales componentes de Call Centre.



Figura 4. 3 Panel de Operador con llamadas en curso

CAPÍTULO 5

ANÁLISIS COMPARATIVO DE COSTOS

COSTO APROXIMADO DE LA SOLUCIÓN IMPLEMENTADA

Consideraciones

- En la implementación deben tomarse en cuenta los costos en Hardware, Software, y Servicios Profesionales para Configuración e Instalación.
- Este análisis observa una implementación para 4 troncales, 6 dispositivos SIP (5 para teléfonos y 1 para la cuenta de llamadas internacionales), 10 agentes, 1 supervisor, 2 colas, IVR, Consola de Monitoreo, 1 sala de Conferencias, TTS.
- Se asume que el cliente provee una infraestructura de red operativa que permita la instalación de los equipos para Voz sobre IP, y conexión a Internet para el manejo de llamadas Internacionales.

Servidor, costo y características principales.

Case	SUPER POWER
Procesador	INTEL CORE 2 DUO E4300 1.86GHZ
Mainboard	INTEL D946GZIS CD,S775,1066MHZ,DDR2
Disco Duro	SEAGATE 250GB 7200 Sata
Memoria	DDR 2 1GB PC-533mhz Corsair.
Tarjeta de Red	TARJETA DE RED 10/100

Total Servidor	USD 800
-----------------------	----------------

Tabla 5. 1 Costos Servidor

Tarjeta para conexión a PSTN.	Digium TDM400P con cuatro puertos FXO	450
Teléfonos	Teléfono IP Grandstream Budge Tone-200	100 c/u

Tabla 5. 2 Costos Equipos de Telefonía

Costos de Software

Sistema Operativo	Linux en su Distribución CentOS	Gratuito
Software para Servidor de Comunicaciones	Asterisk versión 1.4	Gratuito
Interfaz Web	Flash Operator Panel	Gratuito
TTS	Licencia para 1 voz masculina en español a 8khz para telefonía.	29.99
IVR	Integrado en Asterisk.	Gratuito

Tabla 5. 3 Costos en Software

Servicios Profesionales para Configuración e Instalación.

Estos precios se han considerado tomando en cuenta las horas dedicadas por un Ingeniero especializado a las distintas tareas.

Instalación Hardware*	USD 2000
Agentes:	USD 100 c/u
Supervisores:	USD 100
Colas:	USD 100 c/u
Sala de Conferencias:	USD 100 c/u
IVR:	USD 1000

FOP:	USD 300
Capacitación:	
A usuarios, agentes.	USD 150
A supervisores.	USD 150
Administrador Central.	USD 300
TTS	USD 100
Contrato de Mantenimiento (1 visita trimestral durante 1 año)	USD 1000
Total Solución Propuesta	USD 6500

Tabla 5. 4 Costo Total Implementación con Asterisk

Este precio incluye la instalación de un servidor de comunicaciones, Configuraciones de call centre, IVR, teléfonos, consola de monitoreo.

*** Incluido Sistema Operativo, Tarjetas para Telefonía, Software Asterisk y Configuración Central Telefónica (Dial Plan con contestadora automática para manejar 6 dispositivos SIP, 4 puertos analógicos, cuenta para llamadas internacionales, Buzón de mensajes)**

COSTOS APROXIMADOS SOLUCIÓN PROPIETARIA

Central Telefónica

Solución de Telefonía	1,808.24
Kit de Upgrade de memoria	589.15
Chasis	626.46
Tarjeta para chasis	157.11
Teléfono IP avanzado para Supervisor	239.28
Licencia teléfono supervisor	122.36
Fuentes de poder para teléfonos 120/60-24VDC	93.82
Licencias de conexión con servidores	1,160.18
Softphone licencia de uso	464.06
Total Central Telefónica	5,252.38

Tabla 5. 5 Costos Servidor de Comunicaciones y equipos de Telefonía Propietarios

Call Centre

Licencias	Cantidad	Precio
Solución Contact Centre - Modulo Inicial	1	11,200
Agente	5	4,400
Multimedia	0	0
Call Record	5	550
Scripting	5	550
Supervisor	1	880
Administrator	1	880
IVR	2	880
Smart Dialer	5	2,750
Puertos de ASR/TTS	2	3,200
Diseñador Flow	1	1,000
Diseñador Script	1	1,000
		27,290
Servicios Profesionales		
Instalacion y Configuracion	1	2,100
Implementación y Desarrollo	1	2,000
		4,100
Capacitación		
Operación Administrador	2	800
Operación Supervisor	1	320
Operación Agente	1	320
Curso de Implementación en Flow Designer	1	3,200
Curso de Implementación en Script Designer	0	0
		4,640
Soporte Técnico y Actualización de Licencias - Contrato Anual		
Express 5x8Soporte Técnico Remoto (teléfono, e-mail, web) y Actualización de Licencias de Software en días y horarios hábiles - CONTRATO ANUAL	1	4,094
		4,094
Total Call Centre		40,124

Tabla 5. 6 Costos Infraestructura Call Centre Propietario

Costos Totales Aproximados de la Solución Propietaria

Total Call Centre	USD 40,124
Total Central Telefónica	USD 5,252.38
Total Solución	USD 45,376.38

Tabla 5. 7 Costo Total de una implementación del tipo propietario

CONCLUSIONES Y RECOMENDACIONES

CONCLUSIONES

1. El Call Centre debido a sus características propias requiere una infraestructura muy diferente a la de cualquier central telefónica común, lo cual implica un considerable aumento en los costos de instalación, y el tipo de herramientas tecnológicas necesarias para su desarrollo, principalmente si se utiliza un sistema cerrado o antiguo.
2. Se ha comprobado que Asterisk permite el desarrollo de servidores de comunicaciones con Voz sobre IP con todas las características básicas para la atención a los usuarios, entre estas: llamadas internas, llamadas externas entrantes y salientes, correo de voz, y Configuración de cuentas con proveedores internacionales de VoIP.
3. El plan de marcación de Asterisk, permite el desarrollo de funciones como respuesta interactiva de voz, integración con bases de datos, uso de motores de TTS (texto a voz) entre otros, de forma flexible y sin costos adicionales para los canales de comunicación con estas aplicaciones.
4. Las funciones disponibles para el ambiente de call centre permiten una completa y sencilla Configuración de agentes, tiempos de espera, colas para la distribución de llamadas, y mensajes a clientes y usuarios internos.
5. En el ambiente de un Call Centre, el uso de la Voz sobre IP trae una gran cantidad de ventajas, entre ellas la facilidad de integración de las comunicaciones con sistemas computacionales, generación de reportes y control constante de las actividades de los agentes y la calidad del servicio entregado.

6. La Implementación de servidores de Comunicaciones con el software Asterisk permite una importante reducción en el costo de complejas soluciones que permitirán a empresas de todos los tamaños acceder a las ventajas tecnológicas de la Voz sobre IP.

RECOMENDACIONES

1. A pesar de las facilidades que ofrece Asterisk para la reducción de costos, es importante recordar que este software entrega herramientas para desarrollar una central telefónica desde cero, por lo que se debe prestar gran atención al valor del conocimiento, es decir, los profesionales a cargo de Configurar el servidor de comunicaciones, quienes resultan la parte más costosa en la instalación y mantenimiento de la central. Un ahorro excesivo en este aspecto puede resultar en una central poco confiable y de baja calidad.
2. Debido a la apertura que ofrece el código abierto, existe una gran cantidad de productos que se ofrecen en base al servidor de comunicaciones Asterisk y sistemas operativos Linux. Una de las tareas más importantes para quien desee Configurar su propio sistema es la selección de los paquetes y herramientas necesarias.
3. Es altamente recomendable en el ambiente de un Call Center la adquisición de un software para la generación de reportes y otro de manejo de relaciones con los clientes (CRM) para complementar al trabajo realizado por la central telefónica.

REFERENCIAS BIBLIOGRÁFICAS

TEXTOS

BOCKLUND, Lori, *Call Center Technology Demystified*, 1era Edición 4ta Impresión, ICMI Press, Maryland USA 2006, páginas 19-58 “Infrastructure”.

BERGEVIN, Réal, *Call Centers for Dummies*, 1era Edición, John Wiley & Sons LTD., Canada 2005, páginas 169-189 “Making life better with technology”.

CERTAIN, Alfredo, *Asterisk Comunicaciones de Código Abierto*, Gecko Networks 2006, paginas 19 – 28, Capitulo 3: Instalando Asterisk.

MASI D.M.B., FISCHER M.J., HARRIS C.M., *Numerical Analysis of Routing Rules for Call Centers*, Telecommunications Review, 1998

TDM400P User’s Manual, Digium, Huntsville, 2005, páginas 13-29, Chapter 1 Overview, Chapter 2 Card Installation

VAN MEGGELEN, Jim, *Asterisk The Future of Telephony*, Primera Edición, O’Reilly Media, Estados Unidos, 2005, capítulos 2, 3, 4, 5, 6.

PÁGINAS WEB

<http://www.marketing-xxi.com/rafael-muniz-gonzalez.html>, Capítulo 7.9 “Call Centers”

<http://www.gestiopolis.com/canales/demarketing/articulos/61/callcenter.htm>, ¿Qué es un Call Centre?

<http://cnx.org/content/m13398/latest/>, ¿Qué es un Call Centre?

http://en.wikipedia.org/wiki/Call_center, Call Centre

<http://en.wikipedia.org/wiki/IVR>, IVR

http://en.wikipedia.org/wiki/Automatic_call_distributor, ACD

http://en.wikipedia.org/wiki/Computer_telephony_integration, CTI

<http://www.alfinal.com/consultor/CRM/callcenter.shtml>, CRM

http://www.serintel.cl/in_prof/calidad_de_servicio.html, Calidad de Servicio

<http://www.voipforo.com/>

http://www.ecualug.org/?q=2007/jun/22/comos/asterisk_1_4_4_en_centos_5,
Instalación de Asterisk.

<http://www.digium.com/en/products/hardware/tdm400p.php>, Tarjeta TDM400P.

<http://www.asternic.org/>

http://www.asteriskguru.com/tutorials/agents_conf.html

http://www.asteriskguru.com/tutorials/queues_conf.html

<http://www.linuxparatodos.net/portal/staticpages/index.php?page=instalacion-texto-centos>

ANEXO 1

CÓDIGO DE CONFIGURACIÓN DE LA CENTRAL

Archivo extensions.conf

;Manejo de las llamadas entrantes

[entrante]

;MENU PRINCIPAL PARA LLAMADAS ENTRANTES

```
exten => danielverdesoto,1,Goto(s,1)
exten => s,1,Answer()
exten => s,2,Playback(bienvenida)
exten => s,3,Background(menu)
exten => 1,1,Macro(extenvm,${EXT201})
exten => 2,1,Macro(colas,${COLA1})
exten => 3,1,Macro(colas,${COLA2})
exten => 4,1,Playback(bienvenida-conferencias)
exten => 4,2,Goto(conferencia,s,1)
exten => 5,1,Goto(ivr,s,1)
exten => i,1,Playback(opcion-invalida)
exten => i,2,Goto(incoming,s,3)
exten => t,1,Macro(extenvm,${EXT201})
```

; Inclusion de archivos con los demás contextos

```
#include extensions_interno.conf
#include extensions_saliente.conf
#include extensions_macros.conf
#include ivr_pruebas.conf
#include variables_globales.conf
```

Archivo extensions_interno.conf

[default]

include => agente
include => supervisor

[agente]

include => interno

[supervisor]

include => espiar
include => interno

[interno]

; Extensiones SIP

exten => _2XX,1,Macro(extenvm,SIP/\${EXTEN})

;Directos de las colas

exten => 700,1,Macro(colas,\${COLA1})
exten => 701,1,Macro(colas,\${COLA2})

;Autenticacion para llamadas salientes

exten => _XXXXXX,1,SetGlobalVar(LOCAL=\${DB(\${EXTEN}/local)})
exten => _XXXXXX,n,SetGlobalVar(CELULAR=\${DB(\${EXTEN}/celular)})
exten => _XXXXXX,n,SetGlobalVar(NACIONAL=\${DB(\${EXTEN}/nacional)})
exten =>
_XXXXXX,n,SetGlobalVar(INTERNACIONAL=\${DB(\${EXTEN}/internacional)})
exten => _XXXXXX,n,SetGlobalVar(NOMBREM=\${DB(\${EXTEN}/nombre)})
exten => _XXXXXX,n,SetGlobalVar(CUENTA=\${EXTEN})
exten => _XXXXXX,n,Goto(salientes,s,1)

; Extension para conferencias

exten => 600,1,Goto(conferencia,s,1)

; Extension de prueba Servidor Festival

exten => 123,1,Answer()
exten => 123,n,Playback(bienvenidom)


```

exten => 123,n,Festival(${CALLERID(name)})
exten => 123,n,Playback(suextension)
exten => 123,n,SayNumber(${CALLERID(num)})
exten => 123,n,Wait(2)
exten => 123,n,Hangup()

```

; Extension para pruebas con Cepstral
; Se envia el texto a Cepstral y se guarda la salida en un archivo temporal para luego reproducirlo y borrarlo

```

exten => 124,1,Answer()
exten => 124,n,System(/usr/local/cepstral/bin/swift -o /tmp/swift.wav "Bienvenido,
" ${CALLERID(name)} ", su numero de extension es , " ${CALLERID(num)})
exten => 124,n,Playback(/tmp/swift)
exten => 124,n,System(rm /tmp/swift.wav)
exten => 124,n,Hangup()

```

; Extension de pruebas del IVR

```

exten => 125,1,Goto(ivr,s,1)

```

; Llamadas para registro de agentes

```

exten => *96,1,Answer()
exten => *96,n,Wait(1)
exten => *96,n,AgentCallbackLogin()
exten => *96,n,Hangup()

```

; Llamadas al buzón de mensajes

```

exten => *98,1,Answer
exten => *98,n,Wait(1)
exten => *98,n,VoiceMailMain()
exten => *98,n,Hangup()

```

; Números para grabar mensajes

```

exten => _66XX,1,Wait(2)
exten => _66XX,n,Record(/tmp/prompt${EXTEN:2}.wav)
exten => _66XX,n,Wait(1)
exten => _66XX,n,Playback(/tmp/prompt${EXTEN:2})
exten => _66XX,n,Wait(2)
exten => _66XX,n,Hangup()

```

; Numero para simulacion de llamadas

```

exten => 22255516,1,Goto(entrante,s,1)

```

[espiar]

exten => 321,1,ZapBarge

exten => 321,2,Hangup()

exten => _32NXX,1,ExtenSpy(\${EXTEN:2}@agente,b,w,q,v(4))

exten => _32NXX,2,Hangup()

[conferencia]

exten => s,1,Meetme(600,i)

Archivo extensions_saliente.conf

;Contexto para llamadas salientes

[salientes]

```
exten => s,1,System(/usr/local/cepstral/bin/swift -o /tmp/swift.wav "Bienvenido, "
${NOMBREM})
exten => s,n,Playback(/tmp/swift)
exten => s,n,System(rm /tmp/swift.wav)
exten => s,n,Background(marque_numero)
```

;Llamadas locales

```
exten => _NXXXXXX,1,Macro(marcar,${LOCAL},${TRONCAL},${EXTEN},1)
```

;Llamadas a celular

```
exten =>
_09XXXXXXX,1,Macro(marcar,${CELULAR},${TRONCAL},${EXTEN},1)
exten =>
_08XXXXXXX,1,Macro(marcar,${CELULAR},${TRONCAL},${EXTEN},1)
```

;Llamadas nacionales

```
exten =>
_03XXXXXXX,1,Macro(marcar,${NACIONAL},${TRONCAL},${EXTEN},1)
exten =>
_04XXXXXXX,1,Macro(marcar,${NACIONAL},${TRONCAL},${EXTEN},1)
exten =>
_05XXXXXXX,1,Macro(marcar,${NACIONAL},${TRONCAL},${EXTEN},1)
exten =>
_06XXXXXXX,1,Macro(marcar,${NACIONAL},${TRONCAL},${EXTEN},1)
exten =>
_07XXXXXXX,1,Macro(marcar,${NACIONAL},${TRONCAL},${EXTEN},1)
```

;Llamadas internacionales

```
exten =>
_00X.,1,Macro(marcar,${INTERNACIONAL},${TRONCAL},${EXTEN},0)
```

Archivo extensions_macros

; Contexto para las macros

; Macro para marcar hacia las extensiones

[macro-extenvm]

```
exten => s,1,Dial(${ARG1},20)
exten => s,2,Voicemail(u${MACRO_EXTEN})
exten => s,3,Hangup
exten => s,102,Voicemail(b${MACRO_EXTEN})
exten => s,103,Hangup
```

; Macro para el manejo de colas de llamadas

[macro-colas]

```
exten => s,1,Queue(${ARG1}|tT|||20)
exten => s,n,Playback(beep)
exten => s,n,Hangup()
```

; Macro para llamadas salientes

[macro-marcar]

```
exten => s,1,GotoIf($[${ARG1} = 1]? 2:i,1)
exten => s,2,Set(CDR(accountcode)=${NOMBREM})
exten => s,3,Macro(borrar_variables)
exten => s,4,GotoIf($[${ARG4} = 1]? 5:6)
exten => s,5,Dial(${ARG2}/${ARG3})
exten => s,6,Dial(SIP/${ARG3}@sip1.sipdiscount.com)
exten => s,106,Dial(${ARG2}/${ARG3})
exten => s,107,Congestion()
```

```
exten => i,1,Macro(borrar_variables)
exten => i,n,Playback(llamada_no_autorizada)
exten => i,n,Hangup()
```

; Borrado de variables creadas en llamadas salientes de acuerdo al usuario autenticado

[macro-borrar_variables]

```
exten => s,1,SetGlobalVar(LOCAL=0)
exten => s,n,SetGlobalVar(NACIONAL=0)
```

```
exten => s,n,SetGlobalVar(CELULAR=0)  
exten => s,n,SetGlobalVar(INTERNACIONAL=0)  
exten => s,n,SetGlobalVar(NOMBREM=0)
```

Archivo ivr_pruebas.conf

[ivr]

```

exten => s,1,Answer()
exten => s,2,Background(menu_ivr)
exten => s,3,Background(ingrese_cedula)

```

```

;A continuacion se lee desde Asterisk Database el nombre, cedula y saldo
;correspondientes al usuario

```

```

exten =>
_[012]XXXXXXXX,1,SetGlobalVar(NOMBRE=${DB(${EXTEN}/nombre)})
exten =>
_[012]XXXXXXXX,n,SetGlobalVar(APELLIDO=${DB(${EXTEN}/apellido)})
exten => _[012]XXXXXXXX,n,SetGlobalVar(CLAVE=${DB(${EXTEN}/clave)})
exten => _[012]XXXXXXXX,n,SetGlobalVar(SALDO=${DB(${EXTEN}/saldo)})
exten => _[012]XXXXXXXX,n,SetGlobalVar(CEDULA=${EXTEN})
exten => _[012]XXXXXXXX,n,SetGlobalVar(SEXO=${DB(${EXTEN}/sexo)})
exten => _[012]XXXXXXXX,n,Background(ingrese_clave)

```

```

exten => _XXXX,1,GotoIf($[${EXTEN} = ${CLAVE}]? 2 : i,1)
exten => _XXXX,2,Goto(ivr_autenticado,s,1)

```

```

exten => 3,1,Goto(ivr,s,3)
exten => 4,1,Goto(entrante,s,3)
exten => 5,1,Playback(adios)
exten => 5,2,Hangup()
exten => t,1,Goto(ivr,s,4)
exten => i,1,Playback(opcion-invalida)
exten => i,2,Goto(ivr,s,3)

```

[ivr_autenticado]

```

exten => s,1,GotoIf($[${SEXO} = m]? 2:4)
exten => s,2,Playback(bienvenidom)
exten => s,3,Goto(5)
exten => s,4,Playback(bienvenidaf)
exten => s,5,System(/usr/local/cephstral/bin/swift -o /tmp/swift.wav ${NOMBRE}
${APELLIDO})
exten => s,6,Playback(/tmp/swift)
exten => s,7,System(rm /tmp/swift.wav)
exten => s,8,Background(menu_personal)

exten => 1,1,Playback(elsaldoes)

```

```

exten => 1,2,System(/usr/local/cephstral/bin/swift -o /tmp/swift.wav ${SALDO}
"dolares" )
exten => 1,3,Playback(/tmp/swift)
exten => 1,4,System(rm /tmp/swift.wav)
exten => 1,5,Goto(s,8)

```

```

exten => 2,1,Background(ingrese_clave)
exten => 3,1,Goto(ivr,s,2)
exten => 4,1,Playback(adios)
exten => 4,2,Hangup()

```

```

exten => i,1,Playback(opcion-invalida)
exten => i,2,Hangup()

```

```

exten => _XXXX,1,GotoIf($[${EXTEN}=${CLAVE}]? 2:i,1)
exten => _XXXX,2,Goto(cambio-clave,s,1)

```

[cambio-clave]

```

exten => s,1,Set(TMP=0)
exten => s,2,Background(ingrese_clave_nueva)
exten => _XXXX,1,GotoIf($[${TMP}=0]? :5:2)
exten => _XXXX,2,Set(CLAVE2=${EXTEN})
exten => _XXXX,3,Set(TMP=1)
exten => _XXXX,4,Background(confirme_clave_nueva)
exten => _XXXX,5,GotoIf($[${EXTEN} = ${CLAVE2}]? 6 : i,1)
exten => _XXXX,6,Set(DB(${CEDULA}/clave)=${CLAVE2})
exten => _XXXX,7,Playback(clave_guardada)
exten => _XXXX,8,Goto(ivr_autenticado,s,8)

```

```

exten => i,1,Playback(opcion-invalida)
exten => i,2,Goto(ivr_autenticado,s,8)

```

Archivo variables_globales.conf

;Contexto global

[globals]

;Declaraci3n de variables

;Colas

COLA1 = cola-ventas

COLA2 = cola-tecnico

;Troncales

TRONCAL=Zap/g2

Archivo manager.conf

[general]

displayssystemname = yes

enabled = yes

port = 5038

bindaddr = 127.0.0.1

[daniel]

secret=daniel

deny=0.0.0.0/0.0.0.0

permit=127.0.0.1/255.255.255.0

read = system,call,log,verbose,command,agent,user

write = system,call,log,verbose,command,agent,user

Archivo sip.conf

; Archivo de Configuración

[general]

bindport=5060

context=default

language=es

disallow=all

allow=ulaw

context=entrante

register => danielverdesoto:dani221708@sip1.sipdiscount.com

; Inclusión del archivo de dispositivos

#include sip_dispositivos.conf

Archivo sip_dispositivos.conf

```
[201]
type=friend
host=dynamic
context=supervisor
username=201
secret=201
callerid= Daniel Verdesoto <201>
dtmfmode=rfc2833
mailbox=201@supervisor
callgroup=1
pickupgroup=1
```

```
[202]

type=friend
host=dynamic
context=agente
username=202
secret=202
callerid= Cristóbal Cordero <202>
callgroup=1
pickupgroup=1
mailbox=202@agente
```

```
[203]

type=friend
host=dynamic
context=agente
username=203
secret=203
callerid= Equipo Estudio <203>
callgroup=1
pickupgroup=1
mailbox=203@agente
```

```
[204]

type=friend
host=dynamic
context=agente
username=204
secret=204
callerid= Servidor Asterisk <204>
```

callgroup=1
pickupgroup=1
mailbox=204@agente

[205]
type=friend
host=dynamic
context=agente
username=205
secret=205
callerid= Maribel Verdesoto <205>
callgroup=1
pickupgroup=1
mailbox=205@agente

[sip1.sipdiscount.com]
type=friend
host=sip1.sipdiscount.com
progressinband=yes
fromdomain=sipdiscount.com
fromuser=danielverdesoto
authuser=danielverdesoto
username=danielverdesoto
secret=dani221708
domain=sipdiscount.com
disallow=all
allow=ulaw
dtmfmode=rfc2833
nat=yes
canreinvite=no
qualify=yes
insecure=very
context=entrante

Archivo agents.conf

[general]

persistentagents=yes

[agents]

;

ackcall=no

wrapuptime=5000

musiconhold=default

updatecdr=yes

recordedagentcalls=yes

recordformat=gsm

; Listado de agentes

agent => 5001,1231, Daniel Verdesoto

agent => 5002,1232, Eric Verdesoto

agent => 5003,1233, Maribel Verdesoto

agent => 5004,1234, Virginia Rodriguez

agent => 5005,1235, Karen Verdesoto

agent => 5006,1236, Pablo Coba

agent => 5007,1237, Cristobal Cordero

agent => 5008,1238, Freddy Garzon

agent => 5009,1239, Paulina Erazo

agent => 5010,1240, Karina Salazar

Archivo queues.conf

[general]

persistentmembers = yes
autofill = yes
monitor-type=MixMonitor
;

[default]

; Cola de llamadas

[cola-ventas]

music = default
announce = anunciouv
strategy = rrmemory
timeout = 15
retry = 5
wrapuptime = 15
maxlen = 0

member = Agent/5001
member = Agent/5002
member = Agent/5003
member = Agent/5004
member = Agent/5005

[cola-tecnico]

music = default
announce = anunciot
strategy = rrmemory
timeout = 15
retry = 5
wrapuptime = 15
maxlen = 0

member = Agent/5006
member = Agent/5007
member = Agent/5008
member = Agent/5009
member = Agent/5010

Archivo zapata.conf

[trunkgroups]

[channels]

usecallerid=yes
hidecallerid=no
inmediate=no

group=2
context=entrante
signalling=fxs_ks
echocancel=yes
channel=1,2,3,4

signalling=fxo_ls
;
rxwink=300
usecallerid=yes
hidecallerid=no
callwaiting=yes
usecallingpres=yes
callwaitingcallerid=yes
threewaycalling=yes
transfer=yes
canpark=yes
cancallforward=yes
callreturn=yes
echocancel=yes
echocancelwhenbridged=yes
rxgain=0.0
txgain=0.0
group=1
callgroup=1
pickupgroup=1
immediate=no

ANEXO 2

CONFIGURACIÓN DE FLASH OPERATOR PANEL

CONFIGURACIÓN DE ASTERISK PARA EL USO DE FOP

Requerimientos para la utilización de FOP

- PERL instalado en el sistema.
- Servidor web, puede ser Apache.
- Agregar un usuario en el archivo manager.conf dentro de los archivos de Configuración de asterisk.
- Una versión compatible de Asterisk, puede ser 1.2 o 1.4.

Instalación

Se debe descargar la última versión del Flash Operator Pannel. Descomprimir el código fuente, ingresar al directorio descomprimido y leer los archivos README, FAQ, de la siguiente forma.

```
#wget http://asternic.org/files/op_panel-snapshot.tar.gz
```

```
#tar zxvf panel-snapshot.tar.gz
```

```
#cd panel-snapshot
```

Copiar los archivos dentro del subdirectorio FLASH a un lugar apropiado dentro del servidor web. Por ejemplo si la raíz del servidor web es /var/www/html, se puede crear un subdirectorio con el nombre panel y copiar los archivos a esta carpeta:

```
#mkdir /var/www/html/panel
```

```
#cp flash/* /var/www/html/panel
```

CONFIGURACIÓN DE FOP

Flash Operator Panel consiste en dos partes, un servidor y un cliente. El servidor se conecta al puerto Asterisk Manager y actúa como un proxy entre el cliente

flash y Asterisk. El cliente Flash se conecta al servidor proxy de FOP (op_server.pl) y habla su propio protocolo para reflejar cambios de estado y enviar comandos de control hacia Asterisk.

Op_server.pl conversa con Asterisk a través del puerto por defecto TCP/5038. Los clientes Flash se conectan con op_server.pl a través del puerto por defecto TCP/4445. Ambos puertos pueden ser cambiados, pero en este proyecto fueron utilizados con la Configuración predeterminada.

El archivo de Configuración para el servidor del panel es op_server.cfg en donde se pueden definir aspectos básicos y fundamentales del setup, como credenciales para registrarse, códigos de seguridad, extensiones y contextos, directorios, etc. El archivo de Configuración para el cliente flash es op_style.cfg en donde se pueden definir principalmente aspectos visuales como el tamaño de los botones, posición de los íconos, tipo y tamaño de fuente, etc.

Para que el programa pueda funcionar, se Configuraron algunos aspectos de Asterisk y del panel para ajustarlos a las necesidades planteadas

Se debe editar el archivo /etc/asterisk/manager.conf y agregar un usuario.

```
; Asterisk Call Management support
;
[general]
enabled = yes
port = 5038
bindaddr = 127.0.0.1

[daniel]
secret = daniel
deny=0.0.0.0/0.0.0.0
permit=0.0.0.0/0.0.0.0
read = system,call,log,verbose,command,agent,user
write = system,call,log,verbose,command,agent,user
```

Se deben considerar ACLs si es necesario restringir el acceso al manager. En este proyecto se ha permitido el acceso desde cualquier host en la red. Si el

servidor va a correr dentro de la misma máquina, es mejor conectarse via localhost (127.0.0.1)

Además se pueden dar permisos de lectura y escritura al manager, para tener un panel funcional con todas sus posibilidades deben darse los permisos indicados en las siguientes líneas.

```
read = system,call,log,verbose,command,agent,user
```

```
write = system,call,log,verbose,command,agent,user
```

Configuración del servidor FOP

Se debe editar el archivo `op_server.cfg` y cambiar los parámetros según sea necesario:

manager_host. Es el nombre o dirección IP del host donde se encuentra Asterisk.

manager_user Es el nombre de usuario definido en `manager.conf` (en el caso de este proyecto es daniel)

event_mask. Permite elegir el tipo de evento a recibir desde Asterisk. Los valores posibles son “on”, “off”, “system”, “call”, “log” El valor escogido para recibir todos los eventos en este proyecto fue “on”.

Es posible monitorear varios servidores de Asterisk con el mismo FOP repitiendo varias veces los parámetros `manager_host`, `manager_user`, `manager_secret` y `event_mask`.

listen_addr. Especifica la dirección IP a la que `op_server.pl` se va a ligar. El valor por defecto va a intentar ligarse a toda dirección IP.

listen_port. Es el puerto que el servidor utilizará para conexiones entrantes de clientes flash, el valor predeterminado es 4445.

web_hostname. El hostname utilizado para acceder al servidor web, es opcional.

security_code. Una contraseña definida para autorizar comandos entrantes (transferencias, cierre de llamadas, etc.) desde clientes flash.

flash_dir. Debe ser la dirección exacta del directorio donde los archivos html y swf se encuentran colocados. Para este proyecto: /var/www/html/panel/flash

poll_interval. Es la frecuencia en segundos con la que el servidor va a consultar el estado de SIP e IAX2.

poll_voicemail. Frecuencia de consultas sobre correo de voz, es opcional.

auth_md5. Para habilitar la autenticación utilizando MD5.

Debug. Nivel de debug para el servidor, puede tener valores de 0 a 255.

queue_hide. Para ocultar botones que no están siendo usados.

clid_format. Un patrón para el formato de números en el caller ID.

clid_privacy. Opción para mostrar u ocultar el callerid en los botones del panel.

show_ip. Habilitado con 1 muestra las direcciones IP para dispositivos SIP e IAX

rename_label_wildcard Para nombrar los botones con el nombre real del canal.

rename_label_agentlogin. Para cambiar la etiqueta del botón al nombre del agente conectado.

rename_label_callbacklogin. Cambia la etiqueta del botón al nombre del agente cuando se logea con un AgentCallbackLogin.

rename_to_agent_name. Permite que el nombre colocado en las opciones anteriores sea el definido en agents.conf

rename_queue_member. Renombra sin usar los nombres en agents.conf.

change_led_agent. Si se activa, cuando un agente se registra, el led del botón que le corresponde cambia de color.

conference_context. Es el contexto donde se encuentran las salas meetme.

voicemail_extension. Es la extensión y contexto en extensions.conf para alcanzar la aplicación de voicemail.

Configuración utilizada para FOP en el Proyecto Implementado

En el archivo op_server.cfg la Configuración escrita fue:

```
[general]
manager_host=127.0.0.1
manager_port=5038
manager_user=daniel
manager_secret=daniel
event_mask=on
auth_md5=1
listen_port=4445
flash_dir = /var/www/html/panel/flash
security_code=daniel
poll_interval=20
poll_voicemail=0
kill_zombies=0
debug=1
language=en
barge_muted=1
clid_format=${CLIDNAME} (xxx) xxx-xxxx
clid_privacy=0
show_ip=1
queue_hide=0
rename_label_agentlogin=0
rename_label_callbacklogin=1
rename_label_wildcard=0
rename_to_agent_name=1
agent_status=1
rename_queue_member=0
change_led_agent=1
```

```
reverse_transfer=0  
clicktodial_insecure=1  
enable_restart=0
```

CONFIGURACIÓN DE BOTONES

Se debe editar el archivo `op_buttons.cfg`. La sintaxis es similar a la de los archivos de Configuración de Asterisk

Se deben Configurar los botones que se desea mostrar en el panel, cualquier canal que no se ha definido en este archivo no aparecerá.

El nombre del canal debe ir dentro de corchetes. Dependiendo del nombre y posición del canal se debe escoger el tipo de botón.

La sintaxis para canales SIP es SIP/extensión. Para ZAP es ZAP/numero del canal. Pra IAX2 es IAX2/nombre.

Ejemplo:

```
[ZAP/4]
Position=1
Label="11 Reception"
Extension=11
Context=local
Icon=1
Mailbox=11@localvm
URL=mypage.php
Target=myframe
```

Posición. Es el número del botón, los botones se colocan en el panel de arriba hacia abajo y de izquierda a derecha. Puede escribirse un número, o el valor `n` que quiere decir la siguiente posición disponible.

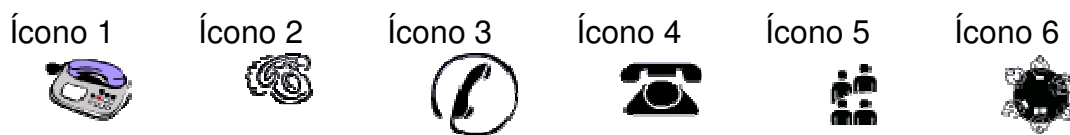
Label. Es la etiqueta del botón, se debe encerrar en comillas.

Server. El servidor Asterisk al que corresponde este dispositivo.

Extensión. El número de extensión, debe estar definido en el dialplan, en caso de no existir extensión para ese botón, como las troncales, se usa el valor `-1`.

Contexto. Si la extensión no es alcanzable en el contexto default del dialplan de Asterisk, se debe especificar el contexto, por ejemplo interno.

Icon. El ícono a utilizar en este botón, existen 6 posibilidades:



Configuración de los botones en el proyecto:

En el archivo op_buttons.cfg, se Configuraron botones para:

Extensiones SIP

Fueron creados 5 botones para las extensiones internas de la 201 a la 205. La Configuración fue como la que se muestra a continuación:

```
[SIP/201]
Position=1
Label="Supervisor"
Extension=201
Context=supervisor
Mailbox=201@supervisor
Icon=1
Privacy=false
Groupcount=1
```

Sala de conferencias

Fue Configurado un botón para la sala Meetme, en el cual se puede observar el número de personas dentro de la conferencia.

```
[600]
Position=7
Label="Meetme"
Extension=600
Context=conferencia
Icon=5
```

Troncales

Para las troncales se crearon cinco botones, cuatro del tipo Zap, y uno del tipo Sip correspondiente a la cuenta de SipDiscount.

```
[Zap/1]
Position=8
Label="TrunkA"
Extension=-1
Icon=2
Context=entrante
[SIP/sip1.sipdiscount.com]
Position=n
Label="SipDisc"
Extension=-1
Icon=2
Context=entrante
```

Colas

Dos botones para las colas, con la siguiente Configuración:

```
[QUEUE/cola-ventas]
Position=13
Label="Ventas"
Extension=-1
Icon=3
```

Agentes

Diez botones para los agentes desde 5001 hasta 5010 con la Configuración:

```
[Agent/5001]
Position=15
Label="Agent1"
Context=Agents
Icon=3
```

Integración con CRM

FOP puede utilizarse en un ambiente de call centre e integrado a un software de CRM basado en Software. Esta característica permite cargar una página web basándose en el caller id de una llamada entrante.

Es posible tener una página web con todos los detalles del cliente antes de responder al teléfono.

Inicio del Servidor

Para que el servidor comience a trabajar, se debe establecer el bit ejecutable en `op_server.pl`, revisar la localización del binario de perl, en el caso de esta implementación `/usr/bin/perl`.

```
#chmod a+x op_server.pl
```

```
#./_opserver.pl
```

ANEXO 3

PRUEBAS EN EL SERVIDOR DE COMUNICACIONES

A continuación se presentan impresiones de las pantallas mostradas del CLI de Asterisk en las cuales se puede observar el comportamiento de la central telefónica para distintas situaciones.

PRUEBAS DE LLAMADAS INTERNAS

Llamadas entre extensiones

La Figura a3.1 indica el comportamiento de la central cuando la extensión 201 perteneciente al contexto supervisor marca a la extensión 202, la cual timbra, luego es atendida y las extensiones son conectadas.

```
-- Executing [202@supervisor:1] Macro("SIP/201-b792aa60", "extenvm|SIP/202") in new stack
-- Executing [s@macro-extenvm:1] Dial("SIP/201-b792aa60", "SIP/202|20") in new stack
-- Called 202CLI>
-- SIP/202-09884b40 is ringing
-- SIP/202-09884b40 answered SIP/201-b792aa60
-- Native bridging SIP/201-b792aa60 and SIP/202-09884b40
== Spawn extension (macro-extenvm, s, 1) exited non-zero on 'SIP/201-b792aa60' in macro 'extenvm'
== Spawn extension (macro-extenvm, s, 1) exited non-zero on 'SIP/201-b792aa60'
```

Figura A-3. 1 Llamadas entre Extensiones

Llamadas al buzón de mensajes

Cuando la extensión 201 en el contexto supervisor marca el número *98 es contestado, y conectado al correo de voz, en donde debe ingresar su número de buzón, y clave para ingresar a un menú en donde puede escuchar sus mensajes, eliminarlos o almacenarlos. Este comportamiento puede observarse en la Figura A3.2.

```

-- Executing [*98@supervisor:1] Answer("SIP/201-098b06a0", "") in new stack
-- Executing [*98@supervisor:2] Wait("SIP/201-098b06a0", "1") in new stack
-- Executing [*98@supervisor:3] VoiceMailMain("SIP/201-098b06a0", "") in new
-- <SIP/201-098b06a0> Playing 'vm-login' (language 'es')
-- <SIP/201-098b06a0> Playing 'vm-password' (language 'es')
-- <SIP/201-098b06a0> Playing 'vm-youhave' (language 'es')
-- <SIP/201-098b06a0> Playing 'digits/1M' (language 'es')
-- <SIP/201-098b06a0> Playing 'vm-message' (language 'es')
-- <SIP/201-098b06a0> Playing 'vm-Olds' (language 'es')
-- <SIP/201-098b06a0> Playing 'vm-onefor' (language 'es')
-- <SIP/201-098b06a0> Playing 'vm-messages' (language 'es')
-- <SIP/201-098b06a0> Playing 'vm-Old' (language 'es')
-- <SIP/201-098b06a0> Playing 'vm-opts' (language 'es')
-- <SIP/201-098b06a0> Playing 'vm-first' (language 'es')
-- <SIP/201-098b06a0> Playing 'vm-message' (language 'es')
== Parsing '/var/spool/asterisk/voicemail/default/201/0ld/msg0000.txt': Found
-- <SIP/201-098b06a0> Playing '/var/spool/asterisk/voicemail/default/201/0ld
[Jan 28 09:43:31] NOTICE[2746]: chan_sip.c:7301 sip_reg_timeout: -- Registrat
1.sipdiscount.com' timed out, trying again (Attempt #106)
[Jan 28 09:43:31] WARNING[2746]: chan_sip.c:7379 transmit_register: Probably a D
danielverdesoto@sipl.sipdiscount.com, trying REGISTER again (after 20 seconds)
-- <SIP/201-098b06a0> Playing 'vm-advopts' (language 'es')
-- <SIP/201-098b06a0> Playing 'vm-repeat' (language 'es')
-- <SIP/201-098b06a0> Playing 'vm-delete' (language 'es')
-- <SIP/201-098b06a0> Playing 'vm-toforward' (language 'es')
-- <SIP/201-098b06a0> Playing 'vm-savemessage' (language 'es')
-- <SIP/201-098b06a0> Playing 'vm-helpexit' (language 'es')
-- <SIP/201-098b06a0> Playing 'vm-goodbye' (language 'es')
[Jan 28 09:43:51] NOTICE[2746]: chan_sip.c:7301 sip_reg_timeout: -- Registrat
1.sipdiscount.com' timed out, trying again (Attempt #107)
[Jan 28 09:43:51] WARNING[2746]: chan_sip.c:7379 transmit_register: Probably a D
danielverdesoto@sipl.sipdiscount.com, trying REGISTER again (after 20 seconds)
-- Executing [*98@supervisor:4] Hangup("SIP/201-098b06a0", "") in new stack
== Spawn extension (supervisor, *98, 4) exited non-zero on 'SIP/201-098b06a0'
ServidorAsterisk*CLI> █

```

Figura A-3. 2 Llamada al Buzón de Mensajes

Registro de Agentes

Al marcar *96, el usuario es atendido, se le solicitan tres datos: usuario, clave y número de extensión, una vez ingresados estos datos exitosamente, se registra al agente, en este caso el usuario 5001 en la extensión 201.

```

-- Executing [*96@supervisor:1] Answer("SIP/201-09dd14c0", "") in new stack
-- Executing [*96@supervisor:2] Wait("SIP/201-09dd14c0", "1") in new stack
-- Executing [*96@supervisor:3] AgentCallbackLogin("SIP/201-09dd14c0", "")
[Nov 21 09:39:26] WARNING[3434]: chan_agent.c:2229 callback_deprecated: AgentCa
[Nov 21 09:39:26] WARNING[3434]: chan_agent.c:2230 callback_deprecated: See doc
[Nov 21 09:39:26] WARNING[3434]: chan_agent.c:2231 callback_deprecated: the sam
-- <SIP/201-09dd14c0> Playing 'agent-user' (language 'es')
-- <SIP/201-09dd14c0> Playing 'agent-pass' (language 'es')
-- <SIP/201-09dd14c0> Playing 'agent-newlocation' (language 'es')
== Setting global variable 'AGENTBYCALLERID_201' to '5001'
-- <SIP/201-09dd14c0> Playing 'agent-loginok' (language 'es')
== Callback Agent '5001' logged in on 201
-- Executing [*96@supervisor:4] Hangup("SIP/201-09dd14c0", "") in new stack
== Spawn extension (supervisor, *96, 4) exited non-zero on 'SIP/201-09dd14c0'
ServidorAsterisk*CLI>

```

Figura A-3. 3 Registro de Agentes

PRUEBAS CON LAS TRONCALES

En la siguiente pantalla (Figura A3.4) se indica el comportamiento de la central ante una llamada entrante a través de la troncal analógica Zap 1/1, se contesta la llamada, se reproduce el mensaje de bienvenida, y se inicia el menú principal.

```

-- Starting simple switch on 'Zap/1-1'
-- Executing [s@entrante:1] Answer("Zap/1-1", "") in new stack
-- Executing [s@entrante:2] Playback("Zap/1-1", "bienvenida") in new stack
-- <Zap/1-1> Playing 'bienvenida' (language 'en')
-- Executing [s@entrante:3] Background("Zap/1-1", "menu") in new stack
-- <Zap/1-1> Playing 'menu' (language 'en')

```

Figura A-3. 4 Llamada entrante desde Troncal Analógica

PRUEBAS DE LLAMADAS ENTRANTES

Cuando una llamada llega a través de la Troncal Zap 1/1, esta es contestada, y se puede ver la secuencia siguiente: Bienvenida, Menú principal, el usuario ingresa la opción 2 en el menú y es llevado a la cola de “Ventas”, en donde escucha música de espera mientras la extensión 201 (donde el agente 5001 registrado en esta cola) timbra, el agente responde la llamada, escucha un mensaje indicando que esta pertenece a la cola “Ventas” y es finalmente conectado con el cliente.

```

-- Starting simple switch on 'Zap/1-1'
-- Executing [s@entrante:1] Answer("Zap/1-1", "") in new stack
-- Executing [s@entrante:2] Playback("Zap/1-1", "bienvenida") in new stack
-- <Zap/1-1> Playing 'bienvenida' (language 'en')
-- Executing [s@entrante:3] Background("Zap/1-1", "menu") in new stack
-- <Zap/1-1> Playing 'menu' (language 'en')
== CDR updated on Zap/1-1
-- Executing [2@entrante:1] Macro("Zap/1-1", "colas|cola-ventas") in new stack
-- Executing [s@macro-colas:1] Queue("Zap/1-1", "cola-ventas|t||20") in new stack
-- Started music on hold, class 'default', on Zap/1-1
-- outgoing agentcall, to agent '5001', on 'Local/201@default-e36f,1'
-- Executing [201@default:1] Macro("Local/201@default-e36f,2", "extenvm|SIP/201") in new stack
-- Executing [s@macro-extenvm:1] Dial("Local/201@default-e36f,2", "SIP/201|20") in new stack
-- Called 201CLI>
-- SIP/201-092cddb8 is ringing
-- Agent/5001 is ringing
-- SIP/201-092cddb8 answered Local/201@default-e36f,2
-- Agent/5001 answered Zap/1-1
-- <Agent/5001> Playing 'anunciov' (language 'en')
== Spawn extension (macro-extenvm, s, 1) exited non-zero on 'Local/201@default-e36f,2' in macro '
== Spawn extension (macro-extenvm, s, 1) exited non-zero on 'Local/201@default-e36f,2'
-- Stopped music on hold on Zap/1-1
== Spawn extension (macro-colas, s, 1) exited non-zero on 'Zap/1-1' in macro 'colas'
== Spawn extension (macro-colas, s, 1) exited non-zero on 'Zap/1-1'
-- Hungup 'Zap/1-1'
ServidorAsterisk*CLI>

```

Figura A-3. 5 Llamada entrante

Pruebas de Salón de Multiconferencias

Cuando el cliente opta por ingresar al salón de multiconferencias, este debe ingresar un pin, y si este es correcto, grabar su nombre para luego ingresar a la conferencia con los otros usuarios presentes.

```

servidorAsterisk*CLI>
-- Starting simple switch on 'Zap/1-1'
-- Executing [s@entrante:1] Answer("Zap/1-1", "") in new stack
-- Executing [s@entrante:2] Playback("Zap/1-1", "bienvenida") in new stack
-- <Zap/1-1> Playing 'bienvenida' (language 'en')
-- Executing [s@entrante:3] Background("Zap/1-1", "menu") in new stack
-- <Zap/1-1> Playing 'menu' (language 'en')
== CDR updated on Zap/1-1
-- Executing [4@entrante:1] Playback("Zap/1-1", "bienvenida-conferencias") in new stack
-- <Zap/1-1> Playing 'bienvenida-conferencias' (language 'en')
-- Executing [4@entrante:2] Goto("Zap/1-1", "conferencia|s|1") in new stack
-- Goto (conferencia,s,1)
-- Executing [s@conferencia:1] MeetMe("Zap/1-1", "600|i") in new stack
== Parsing '/etc/asterisk/meetme.conf': Found
-- Created MeetMe conference 1023 for conference '600'
-- <Zap/1-1> Playing 'conf-getpin' (language 'en')
-- Recording*CLI>
-- <Zap/1-1> Playing 'vm-rec-name' (language 'en')
-- <Zap/1-1> Playing 'beep' (language 'en')
-- x=0, open writing: /var/spool/asterisk/meetme/meetme-username-600-1 format: sln, 0x9
-- User ended message by pressing #
-- <Zap/1-1> Playing 'auth-thankyou' (language 'en')
-- <Zap/1-1> Playing 'vm-review' (language 'en')
-- <Zap/1-1> Playing 'vm-msgsaved' (language 'en')
-- <Zap/1-1> Playing 'conf-onlyperson' (language 'en')

```

Figura A-3. 6 Ingreso al Salón Multiconferencia

PRUEBAS AL IVR

Autenticación

Cuando el cliente ingresa a la opción de “IVR de pruebas” desde el menú principal, en donde se le presenta la posibilidad de autenticarse como usuario, o salir del sistema, debe ingresar su número de cédula, y si esta se encuentra en la base de datos de clientes, tendrá que ingresar su clave personal, con lo cual es llevado al menú “ivr_autenticado”.

```

-- Executing [s@ivr:1] Answer("SIP/201-098b06a0", "") in new stack
-- Executing [s@ivr:2] Background("SIP/201-098b06a0", "menu_ivr") in new stack
-- <SIP/201-098b06a0> Playing 'menu_ivr' (language 'es')
== CDR updated on SIP/201-098b06a0
-- Executing [3@ivr:1] Goto("SIP/201-098b06a0", "ivr|s|3") in new stack
-- Goto (ivr,s,3)
-- Executing [s@ivr:3] Background("SIP/201-098b06a0", "ingrese_cedula") in new stack
-- <SIP/201-098b06a0> Playing 'ingrese_cedula' (language 'es')
[Jan 28 09:48:31] NOTICE[2746]: chan_sip.c:7301 sip_reg_timeout: -- Registration for 'danielverdesoto@
l.sipdiscount.com' timed out, trying again (Attempt #121)
[Jan 28 09:48:31] WARNING[2746]: chan_sip.c:7379 transmit_register: Probably a DNS error for registra
danielverdesoto@sip1.sipdiscount.com, trying REGISTER again (after 20 seconds)
== CDR updated on SIP/201-098b06a0
-- Executing [1708712912@ivr:1] SetGlobalVar("SIP/201-098b06a0", "NOMBRE=carlos") in new stack
== Setting global variable 'NOMBRE' to 'carlos'
-- Executing [1708712912@ivr:2] SetGlobalVar("SIP/201-098b06a0", "APELLIDO=montero") in new stack
== Setting global variable 'APELLIDO' to 'montero'
-- Executing [1708712912@ivr:3] SetGlobalVar("SIP/201-098b06a0", "CLAVE=1708") in new stack
== Setting global variable 'CLAVE' to '1708'
-- Executing [1708712912@ivr:4] SetGlobalVar("SIP/201-098b06a0", "SALDO=30") in new stack
== Setting global variable 'SALDO' to '30'
-- Executing [1708712912@ivr:5] SetGlobalVar("SIP/201-098b06a0", "CEDULA=1708712912") in new stack
== Setting global variable 'CEDULA' to '1708712912'
-- Executing [1708712912@ivr:6] SetGlobalVar("SIP/201-098b06a0", "SEXO=m") in new stack
== Setting global variable 'SEXO' to 'm'
-- Executing [1708712912@ivr:7] Background("SIP/201-098b06a0", "ingrese_clave") in new stack
-- <SIP/201-098b06a0> Playing 'ingrese_clave' (language 'es')
== CDR updated on SIP/201-098b06a0
-- Executing [1708@ivr:1] GotoIf("SIP/201-098b06a0", "1? 2 : i|1") in new stack
-- Goto (ivr,1708,2)
-- Executing [1708@ivr:2] Goto("SIP/201-098b06a0", "ivr_autenticado|s|1") in new stack

```

Figura A-3. 7 Autenticación en el IVR

Una vez en el menú de `ivr_autenticado`, el usuario está en capacidad de escuchar su saldo o realizar un cambio de clave, en el caso de la Figura A3.8, se muestra un cambio de clave, al usuario se le solicita su clave actual, luego debe ingresar una clave nueva de 4 dígitos, y la confirmación de esta clave. Si los datos se han ingresado correctamente, la nueva clave es almacenada en la base de datos, se informa al cliente del cambio exitoso y se lo lleva nuevamente al menú `ivr_autenticado`.


```

== CDR updated on SIP/201-098b06a0
-- Executing [1708@ivr_autenticado:1] GotoIf("SIP/201-098b06a0", "1? 2:i|1") in new stack
-- Goto (ivr_autenticado,1708,2)
-- Executing [1708@ivr_autenticado:2] Goto("SIP/201-098b06a0", "cambio-clave|s|1") in new
-- Goto (cambio-clave,s,1)
-- Executing [s@cambio-clave:1] Set("SIP/201-098b06a0", "TMP=0") in new stack
-- Executing [s@cambio-clave:2] Background("SIP/201-098b06a0", "ingrese_clave_nueva") in n
-- <SIP/201-098b06a0> Playing 'ingrese_clave_nueva' (language 'es')
== CDR updated on SIP/201-098b06a0
-- Executing [8071@cambio-clave:1] GotoIf("SIP/201-098b06a0", "1?:5:2") in new stack
-- Executing [8071@cambio-clave:2] Set("SIP/201-098b06a0", "CLAVE2=8071") in new stack
-- Executing [8071@cambio-clave:3] Set("SIP/201-098b06a0", "TMP=1") in new stack
-- Executing [8071@cambio-clave:4] Background("SIP/201-098b06a0", "confirme_clave_nueva")
-- <SIP/201-098b06a0> Playing 'confirme_clave_nueva' (language 'es')
== CDR updated on SIP/201-098b06a0
-- Executing [8071@cambio-clave:1] GotoIf("SIP/201-098b06a0", "0?:5:2") in new stack
-- Goto (cambio-clave,8071,5)
-- Executing [8071@cambio-clave:5] GotoIf("SIP/201-098b06a0", "1? 6 : i|1") in new stack
-- Goto (cambio-clave,8071,6)
-- Executing [8071@cambio-clave:6] Set("SIP/201-098b06a0", "DB(1708712912/clave)=8071") in
-- Executing [8071@cambio-clave:7] Playback("SIP/201-098b06a0", "clave_guardada") in new s
-- <SIP/201-098b06a0> Playing 'clave_guardada' (language 'es')
-- Executing [8071@cambio-clave:8] Goto("SIP/201-098b06a0", "ivr_autenticado|s|8") in new
-- Goto (ivr_autenticado,s,8)

```

Figura A-3. 8 Cambio de Clave

PRUEBAS DE LLAMADAS SALIENTES

Llamadas Autorizadas (Local, Nacional, Celular)

Cuando un usuario interno desea realizar una llamada, debe ingresar su clave de 6 dígitos, la cual es verificada por el sistema en la base de datos, en esta se localizan los permisos de salida y nombre del usuario.

Una vez identificado el usuario, Asterisk le da la bienvenida, pronuncia su nombre mediante el motor TTS y le solicita ingresar el número telefónico al cual desea salir. Cuando el usuario ingresa el número, el sistema reconoce si esta es una llamada local, nacional, celular, o internacional de acuerdo al patrón numérico y de acuerdo a esta información verifica los permisos, en el caso de la Figura A3.9 esta es una llamada permitida.

```

-- Executing [111110@agente:1] SetGlobalVar("SIP/202-b7920c08", "LOCAL=1") in new stack
== Setting global variable 'LOCAL' to '1'
-- Executing [111110@agente:2] SetGlobalVar("SIP/202-b7920c08", "CELULAR=0") in new stack
== Setting global variable 'CELULAR' to '0'
-- Executing [111110@agente:3] SetGlobalVar("SIP/202-b7920c08", "NACIONAL=0") in new stack
== Setting global variable 'NACIONAL' to '0'
-- Executing [111110@agente:4] SetGlobalVar("SIP/202-b7920c08", "INTERNACIONAL=1") in new stack
== Setting global variable 'INTERNACIONAL' to '1'
-- Executing [111110@agente:5] SetGlobalVar("SIP/202-b7920c08", "NOMBREM=danielverdesoto") in new stack
== Setting global variable 'NOMBREM' to 'danielverdesoto'
-- Executing [111110@agente:6] SetGlobalVar("SIP/202-b7920c08", "CUENTA=111110") in new stack
== Setting global variable 'CUENTA' to '111110'
-- Executing [111110@agente:7] Goto("SIP/202-b7920c08", "salientes|s|1") in new stack
-- Goto (salientes,s,1)
-- Executing [s@salientes:1] System("SIP/202-b7920c08", "/usr/local/cepstral/bin/swift -o /tmp/swift.wav
"Bienvenido| " danielverdesoto") in new stack
-- Executing [s@salientes:2] Playback("SIP/202-b7920c08", "/tmp/swift") in new stack
-- <SIP/202-b7920c08> Playing '/tmp/swift' (language 'es')
-- Executing [s@salientes:3] System("SIP/202-b7920c08", "rm /tmp/swift.wav") in new stack
-- Executing [s@salientes:4] Background("SIP/202-b7920c08", "marque_numero") in new stack
-- <SIP/202-b7920c08> Playing 'marque_numero' (language 'es')
== CDR updated on SIP/202-b7920c08
-- Executing [2255516@salientes:1] Macro("SIP/202-b7920c08", "marcar|1|Zap/g2|2255516|1") in new stack
-- Executing [s@macro-marcar:1] GotoIf("SIP/202-b7920c08", "1? 2:i|1") in new stack
-- Goto (macro-marcar,s,2)
-- Executing [s@macro-marcar:2] Set("SIP/202-b7920c08", "CDR(accountcode)=danielverdesoto") in new stack
-- Executing [s@macro-marcar:3] Macro("SIP/202-b7920c08", "borrar_variables") in new stack
-- Executing [s@macro-borrar_variables:1] SetGlobalVar("SIP/202-b7920c08", "LOCAL=0") in new stack
== Setting global variable 'LOCAL' to '0'
-- Executing [s@macro-borrar_variables:2] SetGlobalVar("SIP/202-b7920c08", "NACIONAL=0") in new stack
== Setting global variable 'NACIONAL' to '0'
-- Executing [s@macro-borrar_variables:3] SetGlobalVar("SIP/202-b7920c08", "CELULAR=0") in new stack
== Setting global variable 'CELULAR' to '0'
-- Executing [s@macro-borrar_variables:4] SetGlobalVar("SIP/202-b7920c08", "INTERNACIONAL=0") in new stack
== Setting global variable 'INTERNACIONAL' to '0'
-- Executing [s@macro-borrar_variables:5] SetGlobalVar("SIP/202-b7920c08", "NOMBREM=0") in new stack
== Setting global variable 'NOMBREM' to '0'
-- Executing [s@macro-marcar:4] GotoIf("SIP/202-b7920c08", "1? 5:6") in new stack
-- Goto (macro-marcar,s,5)
-- Executing [s@macro-marcar:5] Dial("SIP/202-b7920c08", "Zap/g2/2255516") in new stack
-- Called g2/2255516
-- Zap/1-1 answered SIP/202-b7920c08
-- Hungup 'Zap/1-1'
-- Spawn extension (macro-marcar,s,5) exited non-zero on SIP/202-b7920c08! in macro 'marcar'

```

Figura A-3. 9 Verificación de Clave para Llamadas Salientes

Llamadas no Autorizadas

En el caso de una llamada no autorizada, después de la verificación con la información en la base de datos, el sistema informa al usuario que esta no es una llamada permitida, y no realiza la conexión.

```

-- Executing [111110@agente:1] SetGlobalVar("SIP/202-b7920c08", "LOCAL=1") in new stack
== Setting global variable 'LOCAL' to '1'
-- Executing [111110@agente:2] SetGlobalVar("SIP/202-b7920c08", "CELULAR=0") in new stack
== Setting global variable 'CELULAR' to '0'
-- Executing [111110@agente:3] SetGlobalVar("SIP/202-b7920c08", "NACIONAL=0") in new stack
== Setting global variable 'NACIONAL' to '0'
-- Executing [111110@agente:4] SetGlobalVar("SIP/202-b7920c08", "INTERNACIONAL=1") in new stack
== Setting global variable 'INTERNACIONAL' to '1'
-- Executing [111110@agente:5] SetGlobalVar("SIP/202-b7920c08", "NOMBREM=danielverdesoto") in new stack
== Setting global variable 'NOMBREM' to 'danielverdesoto'
-- Executing [111110@agente:6] SetGlobalVar("SIP/202-b7920c08", "CUENTA=111110") in new stack
== Setting global variable 'CUENTA' to '111110'
-- Executing [111110@agente:7] Goto("SIP/202-b7920c08", "salientes|s|1") in new stack
-- Goto (salientes,s,1)
-- Executing [s@salientes:1] System("SIP/202-b7920c08", "/usr/local/cepstral/bin/swift -o /tmp/swift.wav
"Bienvenido| " danielverdesoto") in new stack
-- Executing [s@salientes:2] Playback("SIP/202-b7920c08", "/tmp/swift") in new stack
-- <SIP/202-b7920c08> Playing '/tmp/swift' (language 'es')
-- Executing [s@salientes:3] System("SIP/202-b7920c08", "rm /tmp/swift.wav") in new stack
-- Executing [s@salientes:4] Background("SIP/202-b7920c08", "marque_numero") in new stack
-- <SIP/202-b7920c08> Playing 'marque_numero' (language 'es')
== CDR updated on SIP/202-b7920c08
-- Executing [095000465@salientes:1] Macro("SIP/202-b7920c08", "marcar|0|Zap/g2|095000465|1") in new sta
ck
-- Executing [s@macro-marcar:1] GotoIf("SIP/202-b7920c08", "0? 2:i|1") in new stack
-- Goto (macro-marcar,i,1)
-- Executing [i@macro-marcar:1] Macro("SIP/202-b7920c08", "borrar_variables") in new stack
-- Executing [s@macro-borrar_variables:1] SetGlobalVar("SIP/202-b7920c08", "LOCAL=0") in new stack
== Setting global variable 'LOCAL' to '0'
-- Executing [s@macro-borrar_variables:2] SetGlobalVar("SIP/202-b7920c08", "NACIONAL=0") in new stack
== Setting global variable 'NACIONAL' to '0'
-- Executing [s@macro-borrar_variables:3] SetGlobalVar("SIP/202-b7920c08", "CELULAR=0") in new stack
== Setting global variable 'CELULAR' to '0'
-- Executing [s@macro-borrar_variables:4] SetGlobalVar("SIP/202-b7920c08", "INTERNACIONAL=0") in new sta
ck
== Setting global variable 'INTERNACIONAL' to '0'
-- Executing [s@macro-borrar_variables:5] SetGlobalVar("SIP/202-b7920c08", "NOMBREM=0") in new stack
== Setting global variable 'NOMBREM' to '0'
-- Executing [i@macro-marcar:2] Playback("SIP/202-b7920c08", "llamada_no_autorizada") in new stack
-- <SIP/202-b7920c08> Playing 'llamada_no_autorizada' (language 'es')
-- Executing [i@macro-marcar:3] Hangup("SIP/202-b7920c08", "") in new stack
== Spawn extension (macro-marcar, 1, 3) exited non-zero on 'SIP/202-b7920c08' in macro 'marcar'
== Spawn extension (macro-marcar, 1, 3) exited non-zero on 'SIP/202-b7920c08'
ServidorAsterisk*CLI>

```

Figura A-3. 10 Llamada no autorizada

Llamadas Internacionales (Utilizando SipDiscount)

El comportamiento de las llamadas internacionales es el mismo que el de todas las llamadas salientes hasta el momento en el cual se autoriza la conexión, en este punto, la macro de llamadas salientes realiza la marcación a través del dispositivo Sip en lugar de las troncales analógicas, se debe destacar la siguiente línea:

```

Executing [s@macro-marcar:4] GotoIf("SIP/201-0875ef00", "0? 5:6") in new stack
Goto (macro-marcar,s,6)
Executing [s@macro-marcar:6] Dial("SIP/201-0875ef00", "SIP/0019147087568@sip1.sipdiscount.com")
Called 0019147087568@sip1.sipdiscount.com
SIP/sip1.sipdiscount.com-08758bf8 is making progress passing it to SIP/201-0875ef00

```

Figura A-3. 11 Marcado de Llamada Internacional

```

== Parsing /etc/asterisk/users.com : [Dec 13 11:29:27] NOTICE[3500]: pbx_acl.c:3964 pbx_load_module: Fo
-- Executing [111116@supervisor:1] SetGlobalVar("SIP/201-0875ef00", "LOCAL=1") in new stack
== Setting global variable 'LOCAL' to '1'
-- Executing [111116@supervisor:2] SetGlobalVar("SIP/201-0875ef00", "CELULAR=0") in new stack
== Setting global variable 'CELULAR' to '0'
-- Executing [111116@supervisor:3] SetGlobalVar("SIP/201-0875ef00", "NACIONAL=1") in new stack
== Setting global variable 'NACIONAL' to '1'
-- Executing [111116@supervisor:4] SetGlobalVar("SIP/201-0875ef00", "INTERNACIONAL=1") in new stack
== Setting global variable 'INTERNACIONAL' to '1'
-- Executing [111116@supervisor:5] SetGlobalVar("SIP/201-0875ef00", "NOMBREM=cristobalcordero") in new
== Setting global variable 'NOMBREM' to 'cristobalcordero'
-- Executing [111116@supervisor:6] SetGlobalVar("SIP/201-0875ef00", "CUENTA=111116") in new stack
== Setting global variable 'CUENTA' to '111116'
-- Executing [111116@supervisor:7] Goto("SIP/201-0875ef00", "salientes|s|1") in new stack
-- Goto (salientes,s,1)
-- Executing [s@salientes:1] System("SIP/201-0875ef00", "/usr/local/cephstral/bin/swift -o /tmp/swift.w
-- Executing [s@salientes:2] Playback("SIP/201-0875ef00", "/tmp/swift") in new stack
-- <SIP/201-0875ef00> Playing '/tmp/swift' (language 'es')
-- Executing [s@salientes:3] System("SIP/201-0875ef00", "rm /tmp/swift.wav") in new stack
-- Executing [s@salientes:4] Background("SIP/201-0875ef00", "marque_numero") in new stack
-- <SIP/201-0875ef00> Playing 'marque_numero' (language 'es')
== CDR updated on SIP/201-0875ef00
-- Executing [0019147087568@salientes:1] Macro("SIP/201-0875ef00", "marcar|1|Zap/g2|0019147087568|0") i
-- Executing [s@macro-marcar:1] GotoIf("SIP/201-0875ef00", "1? 2:i|1") in new stack
-- Goto (macro-marcar,s,2)
-- Executing [s@macro-marcar:2] Set("SIP/201-0875ef00", "CDR(accountcode)=cristobalcordero") in new sta
-- Executing [s@macro-marcar:3] Macro("SIP/201-0875ef00", "borrar_variables") in new stack
-- Executing [s@macro-borrar_variables:1] SetGlobalVar("SIP/201-0875ef00", "LOCAL=0") in new stack
== Setting global variable 'LOCAL' to '0'
-- Executing [s@macro-borrar_variables:2] SetGlobalVar("SIP/201-0875ef00", "NACIONAL=0") in new stack
== Setting global variable 'NACIONAL' to '0'
-- Executing [s@macro-borrar_variables:3] SetGlobalVar("SIP/201-0875ef00", "CELULAR=0") in new stack
== Setting global variable 'CELULAR' to '0'
-- Executing [s@macro-borrar_variables:4] SetGlobalVar("SIP/201-0875ef00", "INTERNACIONAL=0") in new st
== Setting global variable 'INTERNACIONAL' to '0'
-- Executing [s@macro-borrar_variables:5] SetGlobalVar("SIP/201-0875ef00", "NOMBREM=0") in new stack
== Setting global variable 'NOMBREM' to '0'
-- Executing [s@macro-marcar:4] GotoIf("SIP/201-0875ef00", "0? 5:6") in new stack
-- Goto (macro-marcar,s,6)
-- Executing [s@macro-marcar:6] Dial("SIP/201-0875ef00", "SIP/0019147087568@sip1.sipdiscount.com") in r
-- Called 0019147087568@sip1.sipdiscount.com
-- SIP/sip1.sipdiscount.com-08758bf8 is making progress passing it to SIP/201-0875ef00
-- SIP/sip1.sipdiscount.com-08758bf8 answered SIP/201-0875ef00
-- Packet2Packet bridging SIP/201-0875ef00 and SIP/sip1.sipdiscount.com-08758bf8
== Spawn extension (macro-marcar, s, 6) exited non-zero on 'SIP/201-0875ef00' in macro 'marcar'
== Spawn extension (macro-marcar, s, 6) exited non-zero on 'SIP/201-0875ef00'
[Dec 13 11:31:14] WARNING[2762]: chan_sip.c:12203 handle_response_register: Got 200 OK on REGISTER that is
ServidorAsterisk*CLI>

```

Figura A-3. 12 Autenticación y Marcado de Llamada Internacional

Pruebas de Características Manager

Un manager tiene la capacidad de escuchar las llamadas en curso, mediante una llamada a la extensión 321, y el ingreso del número de troncal a espiar.

```
danielverdesoto@sip1.sipdiscount.com, trying REGISTER again (after 20 seconds)
-- Executing [321@supervisor:1] ZapBarge("SIP/201-0985b130", "") in new stack
-- <SIP/201-0985b130> Playing 'conf-getchannel' (language 'es')
== Spawn extension (supervisor, 321, 1) exited non-zero on 'SIP/201-0985b130'
ServidorAsterisk*CLI>
```

Figura A-3. 13 Espiar llamadas mediante ZapBarge

PRUEBA TTS

La Figura mostrada a continuación (Figura A3.14) indica una prueba del funcionamiento del motor TTS, en el IVR una vez autenticado el usuario, el sistema le da la bienvenida y pronuncia su nombre el cual es obtenido desde la base de datos y llevado hasta Cepstral, almacenado en un archivo temporal para su reproducción en Asterisk y su posterior eliminación.

```
-- Goto (ivr,1708,2)
-- Executing [1708@ivr:2] Goto("SIP/201-098b06a0", "ivr_autenticado|s|1") in new stack
-- Goto (ivr_autenticado,s,1)
-- Executing [s@ivr_autenticado:1] GotoIf("SIP/201-098b06a0", "1? 2:4") in new stack
-- Goto (ivr_autenticado,s,2)
-- Executing [s@ivr_autenticado:2] Playback("SIP/201-098b06a0", "bienvenidom") in new stack
-- <SIP/201-098b06a0> Playing 'bienvenidom' (language 'es')
-- Executing [s@ivr_autenticado:3] Goto("SIP/201-098b06a0", "5") in new stack
-- Goto (ivr_autenticado,s,5)
-- Executing [s@ivr_autenticado:5] System("SIP/201-098b06a0", "/usr/local/cepstral/bin/swift -o /tmp/swift.wav carlos montero") in new stack
-- Executing [s@ivr_autenticado:6] Playback("SIP/201-098b06a0", "/tmp/swift") in new stack
```

Figura A-3. 14 Reproducción de Audio mediante Cepstral

Para utilizar el servidor TTS Festival, este debe iniciarse desde el Prompt de Linux, en la Figura A3.15 se muestran los mensajes mostrados cuando se realizan conexiones exitosas.

```
[root@ServidorAsterisk ~]# festival --server
server    Mon Jan 14 10:28:57 2008 : Festival server started on port 1314
client(1) Mon Jan 14 10:29:15 2008 : accepted from localhost
client(1) Mon Jan 14 10:29:15 2008 : disconnected
client(2) Mon Jan 14 10:30:45 2008 : accepted from localhost
client(2) Mon Jan 14 10:30:45 2008 : disconnected
█
```

Figura A-3. 15 Servidor Festival

Prueba Festival

La Figura A3.16 indica el procedimiento que Asterisk sigue para realizar una comunicación con el Servidor Festival y la reproducción del audio entregado por este motor TTS en una llamada telefónica de prueba.

```
-- Executing [124@agente:1] Answer("SIP/202-b792aa60", "") in new stack
-- Executing [124@agente:2] System("SIP/202-b792aa60", "/usr/local/cepstral/bin/swift -o /tmp/swift.wav
"Bienvenido| " Cristóbal Cordero | su numero de extension es | " 202") in new stack
-- Executing [124@agente:3] Playback("SIP/202-b792aa60", "/tmp/swift") in new stack
-- <SIP/202-b792aa60> Playing '/tmp/swift' (language 'es')
-- Executing [124@agente:4] System("SIP/202-b792aa60", "rm /tmp/swift.wav") in new stack
-- Executing [124@agente:5] Hangup("SIP/202-b792aa60", "") in new stack
== Spawn extension (agente, 124, 5) exited non-zero on 'SIP/202-b792aa60'
-- Executing [123@agente:1] Answer("SIP/202-b7920c08", "") in new stack
-- Executing [123@agente:2] Playback("SIP/202-b7920c08", "bienvenidom") in new stack
-- <SIP/202-b7920c08> Playing 'bienvenidom' (language 'es')
-- Executing [123@agente:3] Festival("SIP/202-b7920c08", "Cristóbal Cordero") in new stack
== Parsing '/etc/asterisk/festival.conf': Found
[Jan 14 10:20:55] WARNING[3574]: app_festival.c:405 festival_exec: festival_client: connect to server failed
== Spawn extension (agente, 123, 3) exited non-zero on 'SIP/202-b7920c08'
-- Executing [123@agente:1] Answer("SIP/202-b792aa60", "") in new stack
-- Executing [123@agente:2] Playback("SIP/202-b792aa60", "bienvenidom") in new stack
-- <SIP/202-b792aa60> Playing 'bienvenidom' (language 'es')
-- Executing [123@agente:3] Festival("SIP/202-b792aa60", "Cristóbal Cordero") in new stack
== Parsing '/etc/asterisk/festival.conf': Found
-- Executing [123@agente:4] Playback("SIP/202-b792aa60", "suextension") in new stack
-- <SIP/202-b792aa60> Playing 'suextension' (language 'es')
-- Executing [123@agente:5] SayNumber("SIP/202-b792aa60", "202") in new stack
-- <SIP/202-b792aa60> Playing 'digits/200' (language 'es')
-- <SIP/202-b792aa60> Playing 'digits/2' (language 'es')
-- Executing [123@agente:6] Wait("SIP/202-b792aa60", "2") in new stack
-- Executing [123@agente:7] Hangup("SIP/202-b792aa60", "") in new stack
== Spawn extension (agente, 123, 7) exited non-zero on 'SIP/202-b792aa60'
ServidorAsterisk*CLI> █
```

Figura A-3. 16 Prueba de Comunicación Asterisk con Festival

La misma prueba para el motor Cepstral mediante una llamada desde Asterisk para la ejecución de este programa en el sistema se muestra a continuación (Figura A3.17).

```
-- Executing [124@agente:1] Answer("SIP/202-b792aa60", "") in new stack
-- Executing [124@agente:2] System("SIP/202-b792aa60", "/usr/local/cepstral/bin/swift -o /tmp/swift.wav
"Bienvenido| " Cristóbal Cordero "| su numero de extension es | " 202") in new stack
-- Executing [124@agente:3] Playback("SIP/202-b792aa60", "/tmp/swift") in new stack
-- <SIP/202-b792aa60> Playing '/tmp/swift' (language 'es')
-- Executing [124@agente:4] System("SIP/202-b792aa60", "rm /tmp/swift.wav") in new stack
-- Executing [124@agente:5] Hangup("SIP/202-b792aa60", "") in new stack
== Spawn extension (agente, 124, 5) exited non-zero on 'SIP/202-b792aa60'
ServidorAsterisk*CLI>
```

Figura A-3. 17 Comunicación de Asterisk con Cepstral

ANEXO 4

INSTALACIÓN DE CENTOS EN MODO TEXTO

PROCEDIMIENTOS

Se debe Insertar el disco de instalación de CentOS y en cuanto aparezca el diálogo de inicio (boot:), pulsar la tecla **ENTER** o ingresar las opciones de instalación deseadas.

Ingresar **linux text** para iniciar la instalación en modo texto.

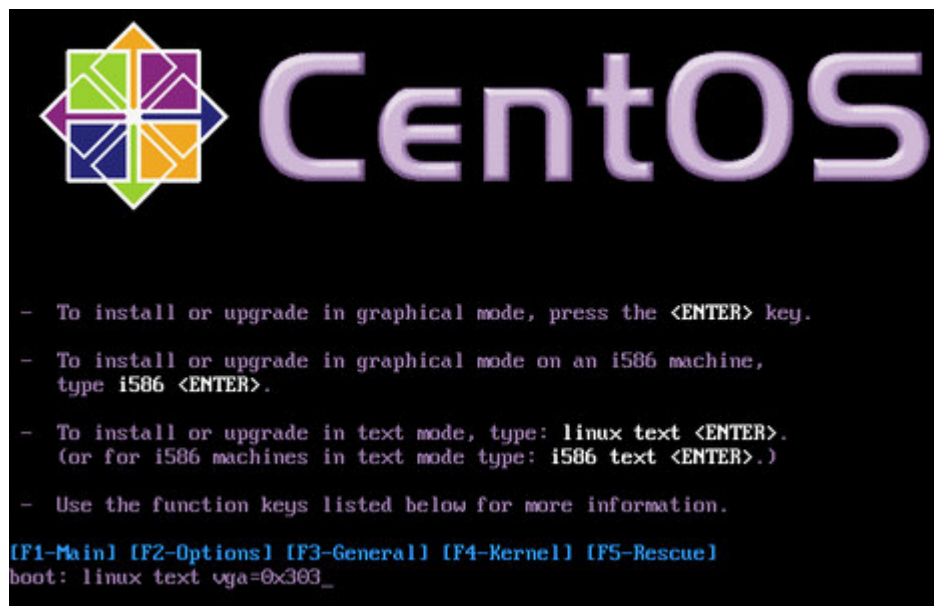


Figura A-4. 1 Pantalla Inicial de Instalación

Para verificar la integridad del disco a partir del cual se realizará la instalación, se debe seleccionar **OK** y pulsar la tecla **ENTER**, este proceso puede tomar varios minutos.

Para saltar esta opción, seleccionar **Skip** y pulsar la tecla **ENTER**.



Figura A-4. 2 Menú para pueba del CD

Pulsar la tecla **ENTER** en la pantalla de bienvenida al programa de instalación de CentOS.

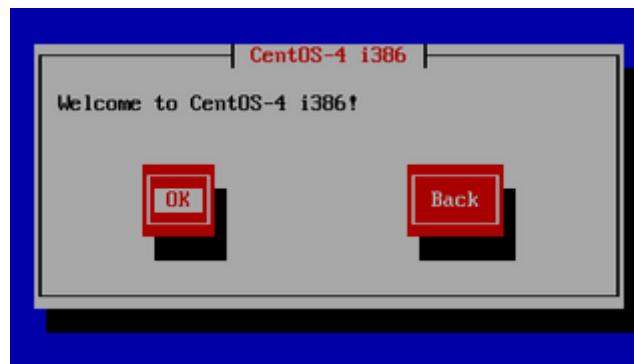


Figura A-4. 3 Pantalla de bienvenida

Seleccionar **Spanish** como idioma para ser utilizado durante la instalación.



Figura A-4. 4 Selección de idioma de instalación

Seleccionar el mapa de teclado que corresponda al dispositivo utilizado. El mapa **es** corresponde a la disposición del teclado Español España. El mapa **latin-1** corresponde a la disposición del teclado Español Latino Americano.



Figura A-4. 5 Selección de Modelo de Teclado

Seleccionar **Custom** y luego pulsar la tecla **ENTER** para realizar una instalación personalizada.

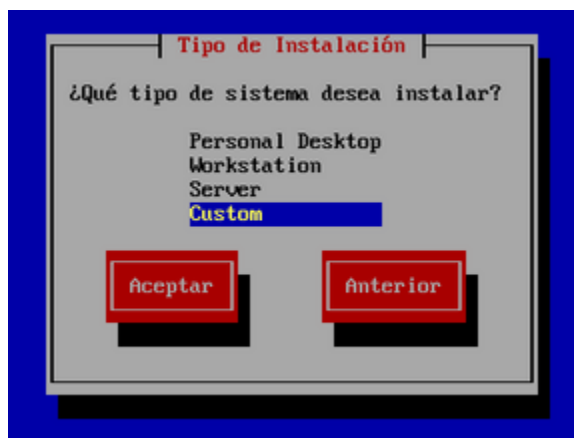


Figura A-4. 6 Tipo de sistema a instalar

Ingresar a la herramienta para particiones del disco duro con Disk Druid

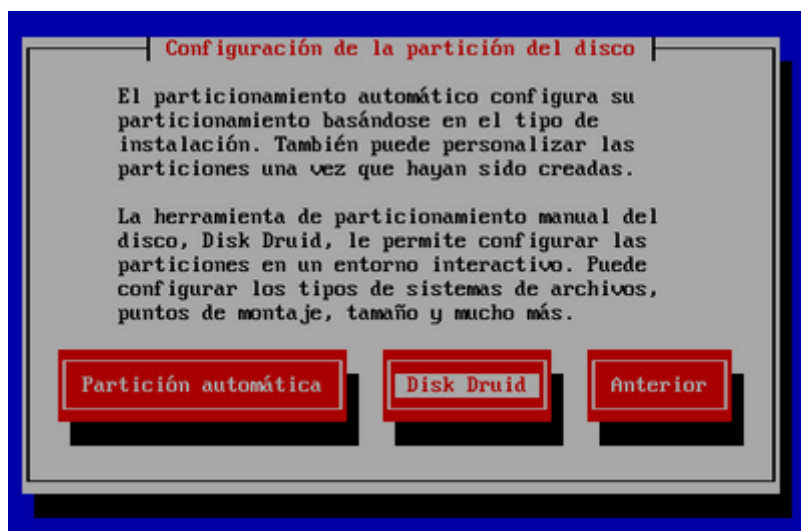


Figura A-4. 7 Herramienta para particiones

Crear una nueva partición seleccionado **Nuevo** y pulsando la tecla **ENTER**.



Figura A-4. 10 Muestra de Partición Nueva

Se debe repetir este procedimiento el número de veces que sea necesario para conseguir las particiones deseadas, el número mínimo recomendado es tres, en el caso de esta implementación se tienen las siguientes particiones:

/
 swap.
 /usr,
 /var,
 /tmp
 /home

Seleccionar otra vez **Nuevo** y crear la siguiente partición.

La partición para la memoria de intercambio no requiere punto de montaje.

Una vez conforme con la tabla de particiones creada, Seleccionar **ACEPTAR** y pulsar la tecla **ENTER** para saltar a la siguiente pantalla.

Seleccionar que se utilizará el gestor de arranque GRUB y pulsar la tecla **ENTER** para saltar a la siguiente pantalla.

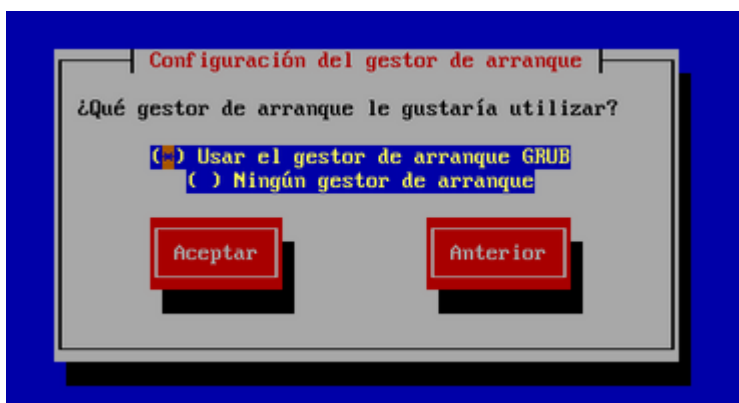


Figura A-4. 11 Selección de Gestor de Arranque

De ser necesario pasar algún parámetro en particular al núcleo (kernel), como por ejemplo, la resolución de pantalla para el modo texto, ingresar en el campo correspondiente aquello que sea necesario. En la mayoría de los casos no se necesita ingresar parámetro alguno.

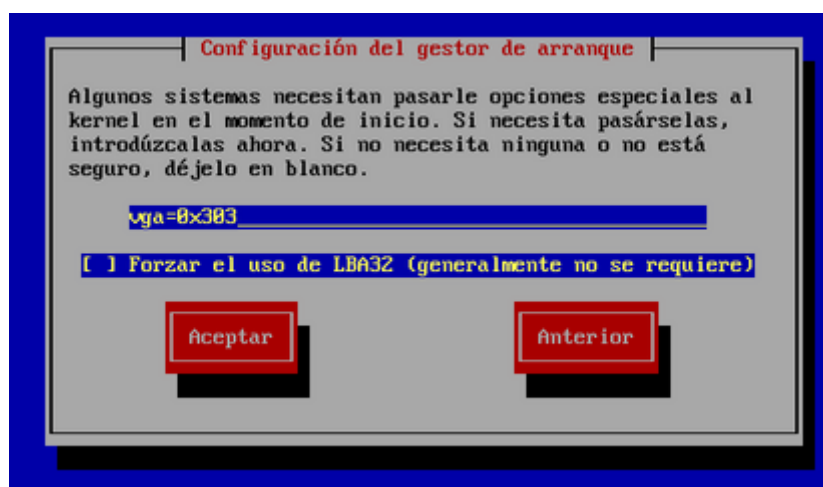


Figura A-4. 12 Configuración del Gestor de Arranque

Por motivos de seguridad, y principalmente con la finalidad de impedir que alguien sin autorización y con acceso físico al sistema pueda iniciar el sistema en nivel de corrida 1, o cualquier otro, asignar, con confirmación, una clave de

acceso exclusiva para el gestor de arranque. Al terminar, pulsar la tecla **ENTER** para saltar a la siguiente pantalla.

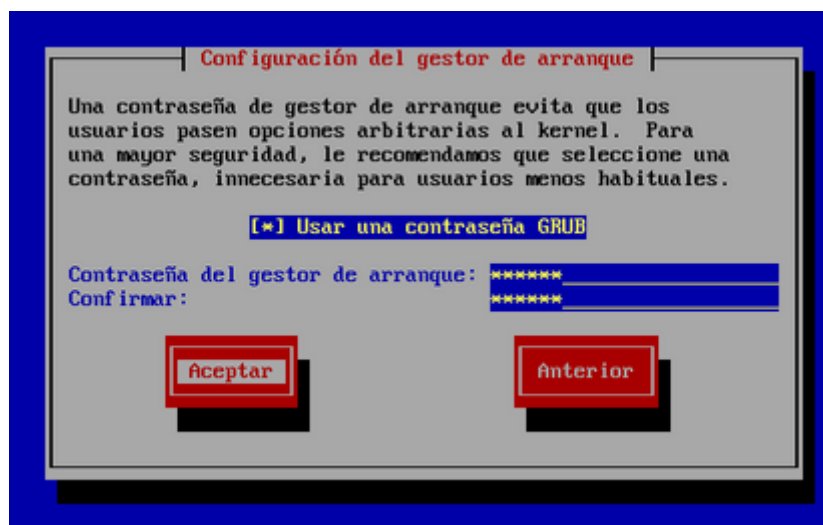


Figura A-4. 13 Contraseña para gestor de arranque

De haber otro sistema operativo instalado en el sistema, Seleccionar el que se utilizará para iniciar de forma predeterminada.



Figura A-4. 14 Selección de Particiones para el Gestor de Arranque

Seleccionar que el gestor de arranque se instale en el sector maestro del disco duro (**MBR** o **Master Boot Record**). Al terminar, pulsar la tecla **ENTER** para saltar a la siguiente pantalla.

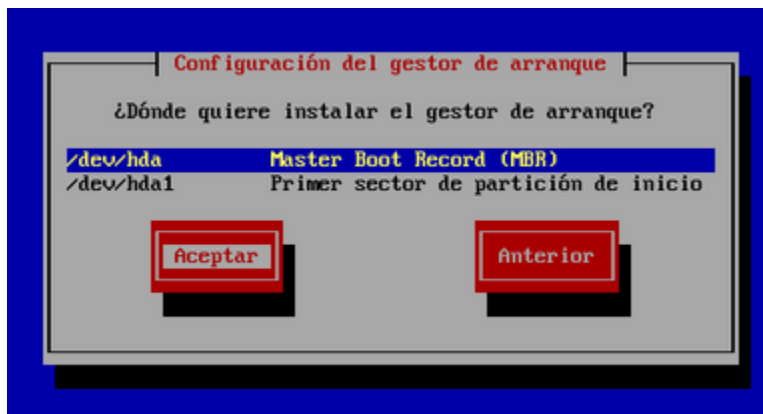


Figura A-4. 15 Ubicación del gestor de arranque

Definir la dirección IP y máscara de subred que utilizará en adelante el sistema.



Figura A-4. 16 Configuración de Red

Definir la dirección IP de la puerta de enlace y las direcciones IP de los servidores DNS de los que disponga.



Figura A-4. 17 Configuración de Gateway y servidores DNS

Asignar un nombre de anfitrión (HOSTNAME) para el sistema. Se recomienda que dicho nombre sea un **FQDN** (Fully Qualified Domain Name) resuelto al menos en un DNS local.



Figura A-4. 18 Nombre del Host

No se Configuró el cortafuegos



Figura A-4. 19 Habilitar Firewall

Saltar la Configuración del cortafuegos.

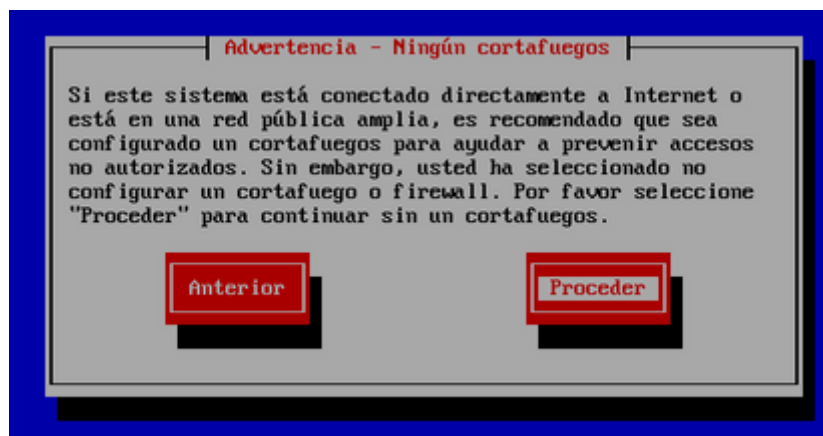


Figura A-4. 20 Advertencia Cortafuegos

Dejar activo SELinux, ya que éste proveerá al sistema de seguridad adicional.



Figura A-4. 21 Selección de SELinux

Agregar el soporte para idiomas adicionales de acuerdo al país donde se hospedará el sistema.



Figura A-4. 22 Soporte de Idiomas

Seleccionar el idioma predeterminado a utilizar en el sistema



Figura A-4. 23 Selección de idioma para el sistema

Seleccionar la casilla **System clock uses UTC**, que significa que el reloj del sistema utilizará **UTC** (Tiempo **U**niversal **C**oordinado), y la zona horaria que corresponda a la región donde se hospedará físicamente el sistema.

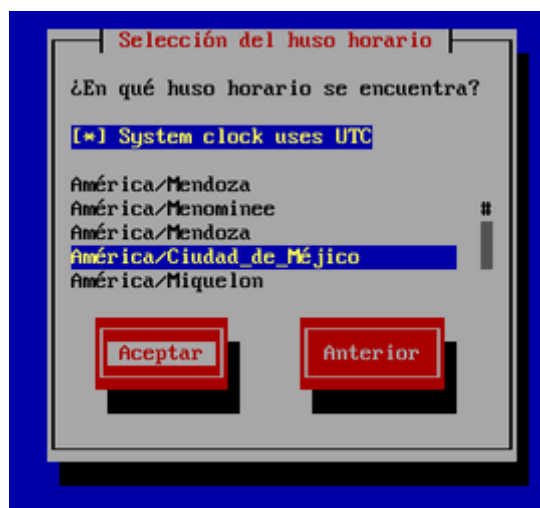


Figura A-4. 24 Selección de Huso Horario

Asignar una clave de acceso al usuario root. Debe escribirse dos veces a fin de verificar que está coincide con lo que realmente se espera.

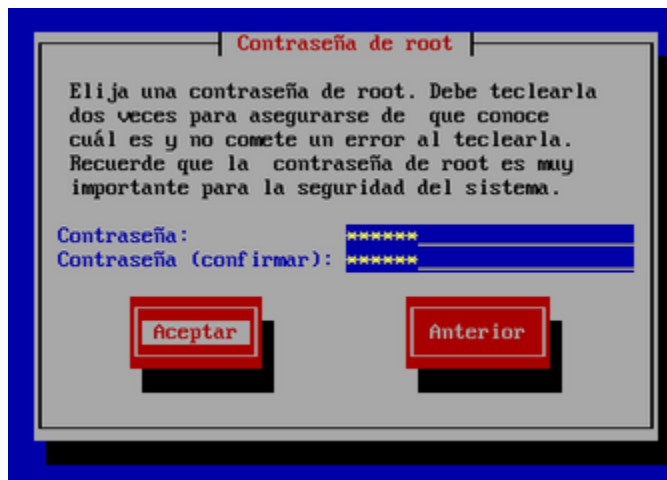


Figura A-4. 25 Selección de contraseña para Root

Es recomendable realizar una instalación con el mínimo de paquetes, desactivando todas las casillas de cada grupo de paquetes. El objeto de esto es solo instalar lo mínimo necesario para el funcionamiento del sistema operativo, y permitir instalar, posteriormente, solo aquello que realmente se requiera de acuerdo a la finalidad productiva que tendrá el sistema.



Figura A-4. 26 Selección de paquetes

Antes de iniciar la instalación sobre el disco duro, el sistema informa respecto a que se guardará un registro del proceso en si en el fichero `/root/install.log`.



Figura A-4. 27 Inicio de Instalación

Si iniciará de forma automática el proceso de formato de las particiones que haya creado para instalar el sistema operativo.

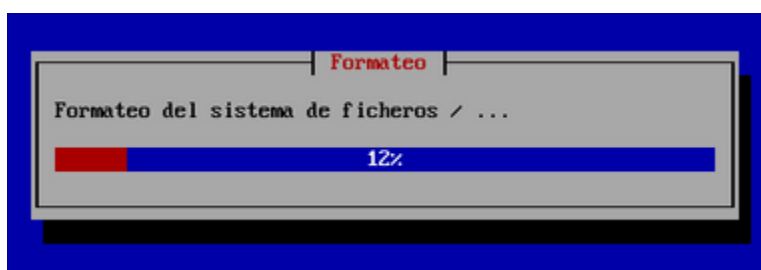


Figura A-4. 28 Formato de particiones

Se realizará de forma automática la transferencia de una imagen del programa de instalación hacia el disco duro, con la finalidad de agilizar el procedimiento.

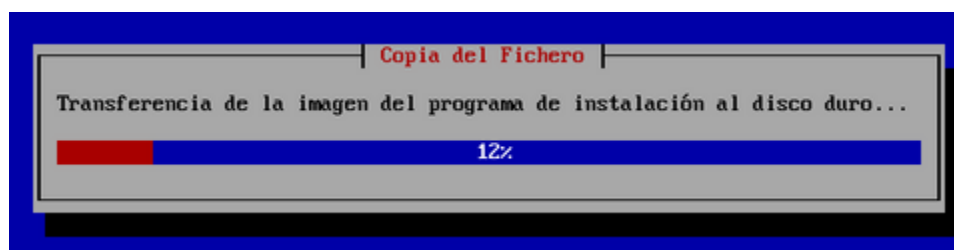


Figura A-4. 29 Transferencia de Imagen del programa de Instalación

Iniciará la instalación de los paquetes necesarios para el funcionamiento del sistema operativo.

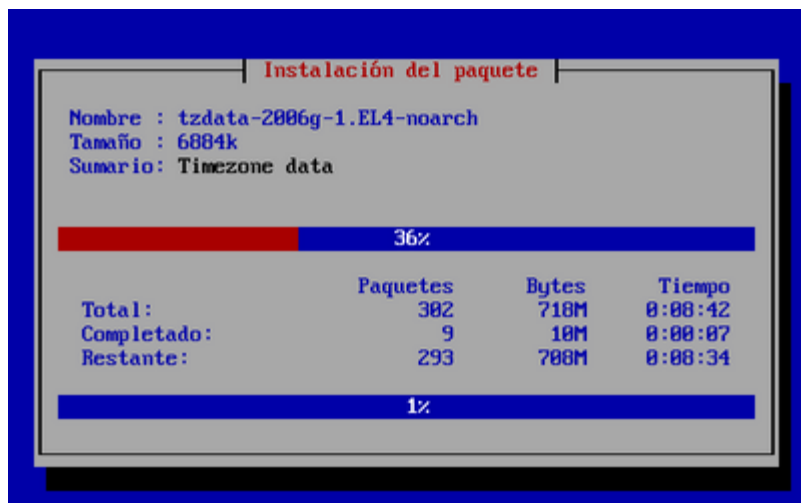


Figura A-4. 30 Instalación de paquetes

Una vez concluida la instalación de los paquetes, pulsar la tecla **ENTER** para reiniciar el sistema.



Figura A-4. 31 Instalación Completa

ANEXO 5

DIAGRAMAS DE FLUJO DEL IVR Y CONTESTADORA AUTOMÁTICA

MENÚ PRINCIPAL

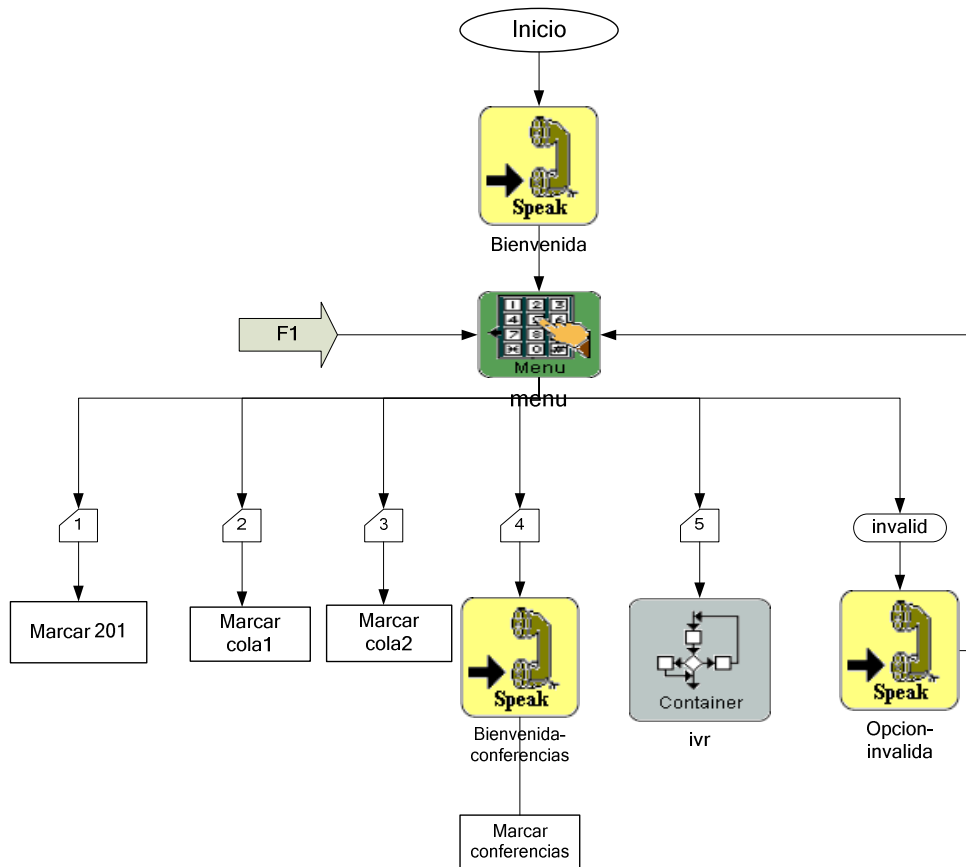


Figura A-5. 1 Diagrama de Flujo Menú Principal

MENÚ IVR AUTENTICACIÓN DE USUARIO

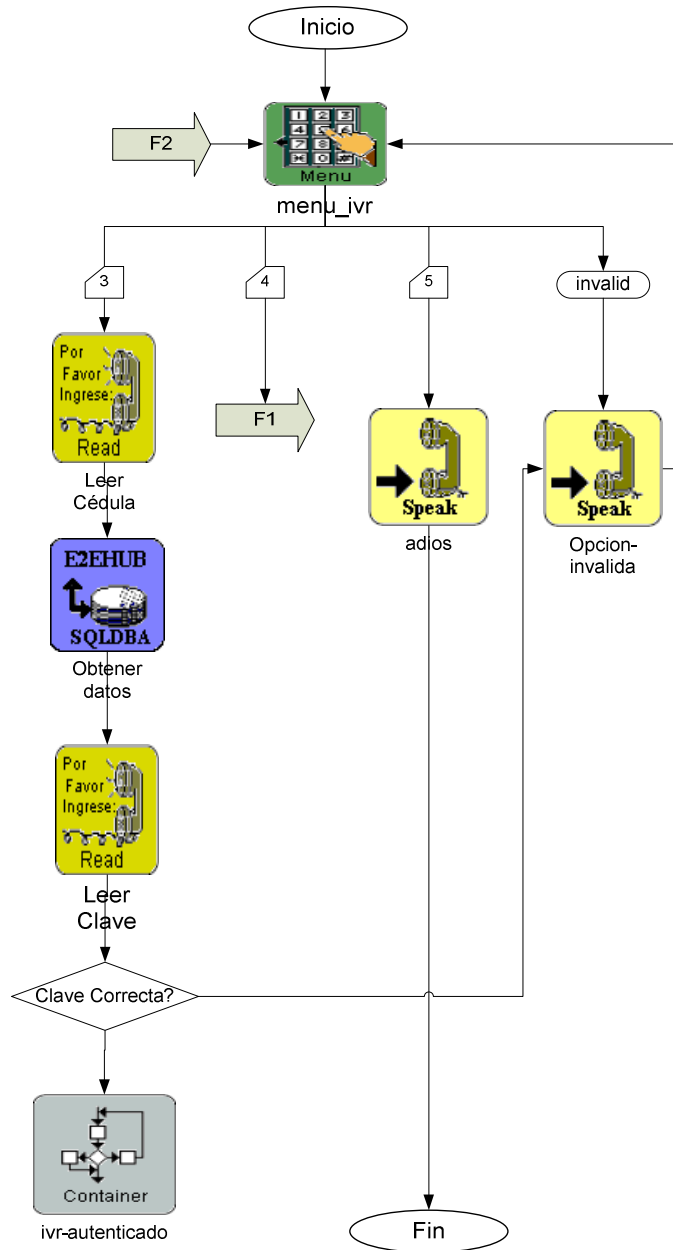


Figura A-5. 2 Diagrama de Flujo Autenticación Usuario IVR

MENÚ DE USUARIO AUTENTICADO

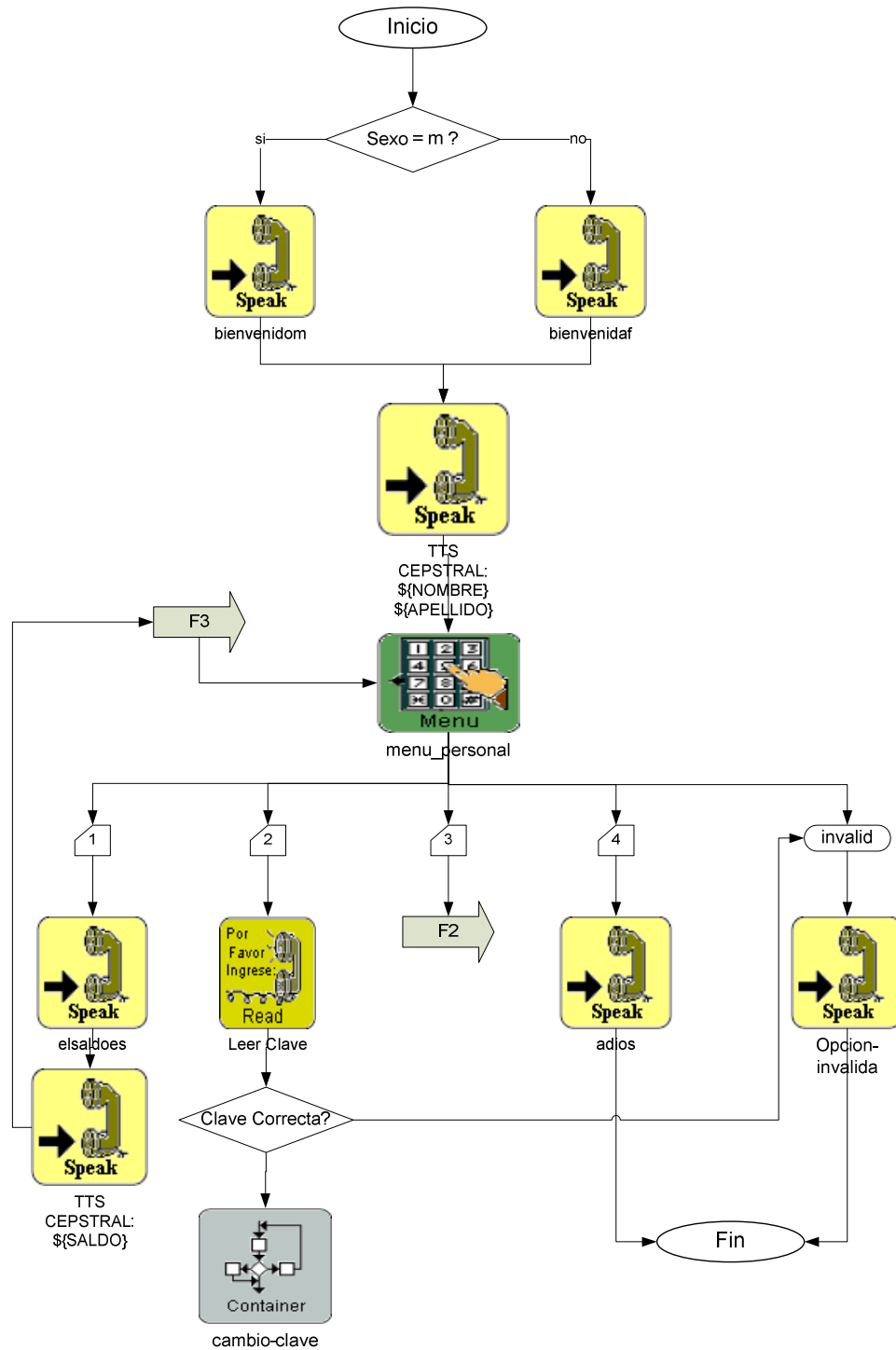


Figura A-5. 3 Diagrama de Flujo Menú de Usuario

CAMBIO DE CLAVE

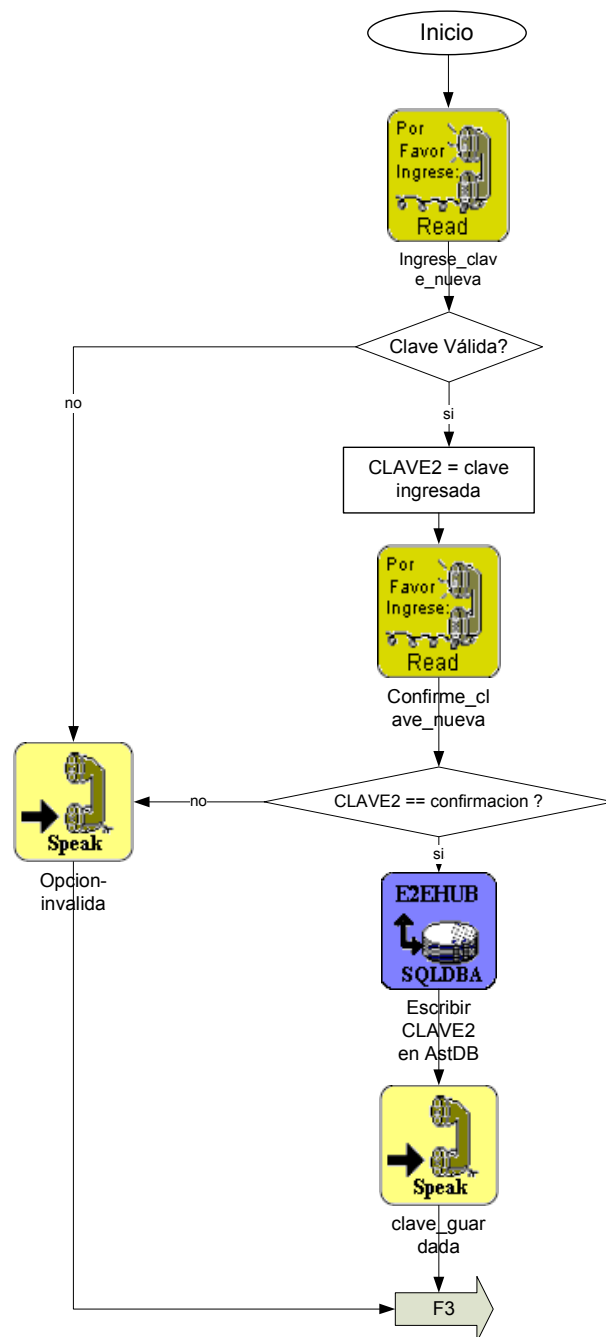


Figura A-5. 4 Diagrama de Flujo Procedimiento Cambio de Clave

ANEXO 6

LISTADO DE COMANDOS DEL CLI DE ASTERISK

Comando

!
 abort halt
 ael debug contexts
 ael debug macros
 ael debug read
 ael debug tokens
 ael nodebug
 ael reload
 agent logoff
 agent show
 agent show online
 agi debug
 agi debug off
 cdr status
 console active
 console answer
 console autoanswer
 console boost
 console dial
 console flash
 console hangup
 console mute
 console send text
 console transfer
 console unmute
 core clear profile
 core set debug channel
 core set debug
 core set debug off
 core set global
 core set verbose
 core show applications
 core show application
 core show audio codecs
 core show channels
 core show channel
 core show channeltypes
 core show channeltype
 core show codecs
 core show codec
 core show Config mappings
 core show file formats
 core show file version
 core show functions
 core show function
 core show globals
 core show hints
 core show image codecs
 core show image formats
 core show license

Función

Ejecutar un comando del shell
 Cancelar un halt en ejecución
 Habilitar depuración de contextos AEL
 Habilitar depuración de macros
 Habilitar depuración de AEL read
 Habilitar depuración de tokens AEL
 Deshabilitar mensajes de depuración AEL
 Recargar Configuración AEL
 Coloca un agente fuera de línea
 Muestra estado de los agentes
 Muestra agentes en línea
 Habilita depuración AGI
 Deshabilita depuración AGI
 Muestra estado CDR
 Establece/muestra consola activa
 Responder a una llamada entrante de consola
 Configura/muestra autorespuesta
 Configura/muestra amplificación de micrófono en db
 Marcar una extensión en la consola
 Flash a una llamada en la consola
 Colgar una llamada en la consola
 Deshabilitar entrada de micrófono
 Enviar texto a un dispositivo remoto
 Transferir una llamada a una distinta extensión
 Habilitar entrada de micrófono
 Vaciar información de perfil
 Habilitar/Deshabilitar depuración de un canal
 Establecer nivel de verbosidad de depuración
 Apagar verbosidad de depuración
 Establecer variable global para plan de marcación
 Establecer nivel de verbosidad
 Muestra aplicaciones registradas del plan de marcación
 Describe una aplicación del plan de marcación
 Muestra una lista de códecs de audio
 Muestra información sobre canales
 Muestra información sobre un canal específico
 Lista los tipos de canales disponibles
 Entrega detalles sobre un tipo de canal
 Muestra una lista de códecs
 Muestra un códec específico
 Muestra mapeo de Configuración
 Muestra formatos de archivos
 Lista versiones de archivos usados para construir Asterisk
 Muestra funciones registradas del plan de marcación
 Describe una función específica del plan de marcación
 Muestra variables globales del plan de marcación
 Muestra pistas del plan de marcación
 Muestra una lista de códecs de imagen
 Muestra formatos de imagen
 Muestra la(s) licencia(s) para esta copia de asterisk

core show profile	Muestra información de perfiles
core show switches	Muestra switches alternativos
core show threads	Muestra threads en ejecución
core show translation	Muestra matriz de traducción
core show uptime	Muestra inforamción de uptime
core show version	Muestra información de versión
core show video codecs	Indica una lista de códecs de vídeo
core show warranty	Muestra la garantía de la copia de Asterisk
database del	Elimina valor de base de datos
database deltree	Elimina valores/árbol de base de datos
database get	Obtiene un valor de la base de datos
database put	Añade/actualiza valor en la base de datos
database show	Muestra los contenidos de la base de datos
database showkey	Muestra contenidos de la base de datos
dialplan add extension	Agrega nueva extensión en un contexto
dialplan add ignorepat	Agrega nuevo patrón para ignorar
dialplan add include	Incluye un contexto en otro
dialplan reload	Recargar extensiones y solo extensiones
dialplan remove extension	Eliminar una extensión específica
dialplan remove ignorepat	Eliminar patrón a ignorar de un contexto
dialplan remove include	Eliminar un include específico de un contexto
dialplan show	Mostrar Plan de Marcación
dnsmgr reload	Recargar Configuración de manager DNS
dnsmgr status	Indica el estado de manager DNS
dundi debug	Habilitar depuración de DUNDi
dundi flush	Borrar cache de DUNDi
dundi lookup	Buscar un número en DUNDi
dundi no debug	Deshabilitar depuración DUNDi
dundi no store history	Deshabilitar registros históricos DUNDi
dundi precache	Precache un número en DUNDi
dundi query	Consultar un EID DUNDi
dundi show entityid	Mostrar ID de Entidad Global
dundi show mappings	Mostrar mapeos DUNDi
dundi show peers	Mostrar pares DUNDi definidos
dundi show peer	Información sobre un compañero DUNDi específico
dundi show precache	Mostrar precache DUNDi
dundi show requests	Mostrar peticiones DUNDi
dundi show trans	Mostrar transacciones DUNDi activas
dundi store history	Habilitar registros históricos DUNDi
feature show	Lista características Configuradas
file convert	Convertir archivo de audio
group show channels	Mostrar canales activos con grupo(s)
help	Mostrar lista de ayuda, o ayuda para un comando
http show status	Mostrar estado de servidor HTTP
iax2 provision	Provisión en un dispositivo IAX
indication add	Add the given indication to the country
indication remove	Remove the given indication from the country
indication show	Display a list of all countries/indications
keys init	Initialize RSA key passcodes
keys show	Displays RSA key information
local show channels	List status of local channels

logger mute	Toggle logging output to a console
logger reload	Reopens the log files
logger rotate	Rotates and reopens the log files
logger show channels	List Configured log channels
manager show command	Show a manager interface command
manager show commands	List manager interface commands
manager show connected	List connected manager interface users
manager show eventq	List manager interface queued events
manager show users	List Configured manager users
manager show user	Display information on a specific manager user
meetme	Execute a command on a conference or conferee
mixmonitor	Execute a MixMonitor command.
module load	Load a module by name
module reload	Reload Configuration
module show	Listar módulos e información
module show like	Lista módulos e información
module unload	Descargar un módulo por nombre
moh reload	Música en espera
moh show classes	Listar clases de MOH
moh show files	Listar clases MOH en base a archivos
originate	Originar una llamada
queue add member	Agregar un canal a una cola especificada
queue remove member	Elimina un canal desde una cola especificada
queue show	Mostrar estado de una cola específica
realtime load	Utilizado para imprimir variables realtime
realtime update	Utilizado para actualizar variables RealTime
restart gracefully	Reiniciar Asterisk con elegancia
restart when convenient	Reiniciar Asterisk cuando no hay llamadas
say load	Configurar/mostrar el modo say
show parkedcalls	Listado de llamadas aparcadas
show queues	Mostrar colas
sip history	Habilitar historial SIP
sip history off	Deshabilitar historial SIP
sip notify	Enviar un paquete de notificación a un peer SIP
sip reload	Recargar Configuración SIP
sip set debug	Habilitar depuración SIP
sip set debug ip	Habilitar depuración SIP en IP
sip set debug off	Deshabilitar depuración SIP
sip set debug peer	Habilitar depuración SIP en Peername
sip show channels	Listar canales SIP activos
sip show channel	Mostrar información detallada de canales SIP
sip show domains	Listar dominios locales SIP
sip show history	Mostrar historial de diálogos SIP
sip show inuse	Listar todos en uso/límites
sip show objects	Listar todas las ubicaciones de objetos SIP
sip show peers	Listar pares SIP definidos
sip show peer	Mostrar detalles de par SIP específico
sip show registry	Listar estado de registro SIP
sip show settings	Mostrar Configuraciones globales SIP
sip show subscriptions	Listar suscripciones SIP activas
sip show users	Listar usuarios SIP definidos

sip show user	Mostrar detalles sobre un usuario SIP específico
stop gracefully	Apagar Asterisk con elegancia
stop now	Apagar Asterisk inmediatamente
stop when convenient	Apagar Asterisk cuando volumen de llamadas vacío
stun debug	Habilitar depuración STUN
stun debug off	Deshabilitar depuración STUN
voicemail show users	Listar buzones de correo de voz definidos
zap destroy channel	Destruir un canal
zap restart	Reiniciar completamente canales zaptel
zap show cadences	Mostrar cadences
zap show channels	Mostrar canales zapata activos
zap show channel	Mostrar información en un canal
zap show status	Mostrar el estado de todas las tarjetas Zaptel

ÍNDICE DE FIGURAS

Capítulo 2

Figura 2. 1 Infraestructura con PBX.....	20
Figura 2. 2 Infraestructura con ACD Independiente.....	21
Figura 2. 3 Infraestructura con Servidor Híbrido	23
Figura 2. 4 Infraestructura con Servidor CTI.....	25
Figura 2. 5 Infraestructura con Servidor de Comunicaciones	27
Figura 2. 6 Conmutación de Circuitos.....	29
Figura 2. 7 Conmutación de Paquetes.....	30
Figura 2. 8 Red Convergente.....	33

Capítulo 3

Figura 3. 1 Diagrama de la Implementación	40
Figura 3. 2 Tarjeta TDM400P	42
Figura 3. 3 Selección de Módulos Zaptel.....	48
Figura 3. 4 Selección de Módulos Zaptel.....	48
Figura 3. 5 Selección de Módulos de Asterisk	50
Figura 3. 6 Diagrama de Dispositivos de Comunicaciones.....	54
Figura 3. 7 Interfaz de Línea de Comandos de Asterisk.....	66

Capítulo 4

Figura 4. 1 Panel Flash correspondiente a la Central Telefónica Implementada.....	82
Figura 4. 2 Panel de Operador con agentes registrados	83
Figura 4. 3 Panel de Operador con llamadas en curso.....	83

Anexo 3

Figura A-3. 1 Llamadas entre Extensiones	122
Figura A-3. 2 Llamada al Buzón de Mensajes	123
Figura A-3. 3 Registro de Agentes.....	124
Figura A-3. 4 Llamada entrante desde Troncal Analógica	124
Figura A-3. 5 Llamada entrante	125
Figura A-3. 6 Ingreso al Salón Multiconferencia	126

Figura A-3. 7 Autenticación en el IVR.....	127
Figura A-3. 8 Cambio de Clave.....	128
Figura A-3. 9 Verificación de Clave para Llamadas Salientes	129
Figura A-3. 10 Llamada no autorizada.....	130
Figura A-3. 11 Marcado de Llamada Internacional	131
Figura A-3. 12 Autenticación y Marcado de Llamada Internacional.....	131
Figura A-3. 13 Espiar llamadas mediante ZapBarge	132
Figura A-3. 14 Reproducción de Audio mediante Cepstral.....	132
Figura A-3. 15 Servidor Festival	133
Figura A-3. 16 Prueba de Comunicación Asterisk con Festival	133
Figura A-3. 17 Comunicación de Asterisk con Cepstral.....	134

Anexo 4

Figura A-4. 1 Pantalla Inicial de Instalación	136
Figura A-4. 2 Menú para prueba del CD	137
Figura A-4. 3 Pantalla de bienvenida.....	137
Figura A-4. 4 Selección de idioma de instalación	138
Figura A-4. 5 Selección de Modelo de Teclado	138
Figura A-4. 6 Tipo de sistema a instalar	139
Figura A-4. 7 Herramienta para particiones	139
Figura A-4. 8 Asignación de particiones	140
Figura A-4. 9 Partición Nueva.....	140
Figura A-4. 10 Muestra de Partición Nueva.....	141
Figura A-4. 11 Selección de Gestor de Arranque	142
Figura A-4. 12 Configuración del Gestor de Arranque.....	142
Figura A-4. 13 Contraseña para gestor de arranque	143
Figura A-4. 14 Selección de Particiones para el Gestor de Arranque.....	143
Figura A-4. 15 Ubicación del gestor de arranque.....	144
Figura A-4. 16 Configuración de Red.....	144
Figura A-4. 17 Configuración de Gateway y servidores DNS	145
Figura A-4. 18 Nombre del Host	145
Figura A-4. 19 Habilitar Firewall.....	146
Figura A-4. 20 Advertencia Cortafuegos.....	146
Figura A-4. 21 Selección de SELinux	147
Figura A-4. 22 Soporte de Idiomas	147
Figura A-4. 23 Selección de idioma para el sistema	148
Figura A-4. 24 Selección de Huso Horario.....	148
Figura A-4. 25 Selección de contraseña para Root	149
Figura A-4. 26 Selección de paquetes	149
Figura A-4. 27 Inicio de Instalación.....	150
Figura A-4. 28 Formato de particiones.....	150
Figura A-4. 29 Transferencia de Imagen del programa de Instalación	150
Figura A-4. 30 Instalación de paquetes	151
Figura A-4. 31 Instalación Completa.....	151

Anexo 5

Figura A-5. 1 Diagrama de Flujo Menú Principal	153
Figura A-5. 2 Diagrama de Flujo Autenticación Usuario IVR	154
Figura A-5. 3 Diagrama de Flujo Menú de Usuario.....	155
Figura A-5. 4 Diagrama de Flujo Procedimiento Cambio de Clave.....	156

ÍNDICE DE TABLAS

Capítulo 2:

Tabla 2. 1 Ventajas y Desventajas de la Infraestructura con PBX	20
Tabla 2. 2 Ventajas y Desventajas de la Infraestructura con ACD Independiente	22
Tabla 2. 3 Ventajas y Desventajas de la Infraestructura con Servidor Híbrido.....	23
Tabla 2. 4 Ventajas y Desventajas de la Infraestructura con Servidor CTI	26
Tabla 2. 5 Ventajas y Desventajas de la Infraestructura con Servidor de Comunicaciones.....	28

Capítulo 3:

Tabla 3. 1 Guía de Requerimientos del Sistema	41
Tabla 3. 2 Características del Servidor Utilizado en el Proyecto	41

Capítulo 5

Tabla 5. 1 Costos Servidor	85
Tabla 5. 2 Costos Equipos de Telefonía.....	85
Tabla 5. 3 Costos en Software	85
Tabla 5. 4 Costo Total Implementación con Asterisk.....	86
Tabla 5. 5 Costos Servidor de Comunicaciones y equipos de Telefonía Propietarios....	86
Tabla 5. 6 Costos Infraestructura Call Centre Propietario.....	87
Tabla 5. 7 Costo Total de una implementación del tipo propietario.....	88

GLOSARIO

SIGLAS:

ACD (Automatic Call Distribution) Distribución Automática de Llamadas

BSD, Berkeley Software Distribution (en español, Distribución de Software Berkeley)

CRM (Customer Relationship Management) Administración de Relaciones con el Cliente.

FOP (Flash Operator Panel) Panel Flash para operadora.

FXO (Foreign Exchange Office) Interfaz que se conecta a la oficina central de la red pública conmutada

FXS (Foreign Exchange Station) Interfaz que se conecta directamente a un teléfono estándar y entrega timbrado, voltaje y tono de marcado.

GNU (GNUs Not Unix) acrónimo recursivo que significa GNU No es Unix.

GPL (General Public License) Licencia Pública General

IAX (Asterisk Exchange Protocol), Protocolo de Intercambio Asterisk.

IP (Internet Protocol), Protocolo Internet

IVR (Interactive Voice Response) Respuesta Interactiva de Voz

MGCP (Media Gateway Control Protocol) Protocolo de control de dispositivos

PSTN (Public Switched Telephone Network) Red Telefónica Conmutada

RJ11 Interfaz física usada para conectar redes telefónicas.

RJ45 Interfaz física usada para conectar redes de datos.

RTP (Real Time Transport Protocol) Protocolo de Transporte en Tiempo Real

SIP (Session Initialization Protocol) Protocolo de Inicio de Sesión

TCP (Transport Control Protocol) Protocolo de Control de Transporte

TTS (Text To Speech) Transformación de Texto a Voz

UDP (User Datagram Protocol) Protocolo de Datagramas de Usuario

VoIP (Voice over IP) Voz sobre IP

TÉRMINOS:

Automatic Call Distributor. dispositivo o sistema especializado que distribuye llamadas entrantes a un grupo específico de terminales que utilizan los agentes.

Agente. Un ejecutivo de atención al cliente, profesional encargado de recibir o realizar llamadas

Call Centre. Instalación con operadores, computadores, y bases de datos de clientes que maneja una gran cantidad de llamadas entrantes y salientes para mercadeo y atención a clientes.

Canal. una forma de comunicación entre los recursos internos y/o externos de una central telefónica como las líneas telefónicas, teléfonos o extensiones.

Case. estuche, caja que contiene el cpu de un computador

Cliente. Toda persona que se comunica con el call centre.

Cola. fila de espera en la cual son colocadas las llamadas de clientes que se comunican con un servicio.

Conmutación por Circuitos. una conexión dedicada se establece de forma temporal para conectar a las dos terminales que se comunican desde el momento en que el teléfono timbra hasta que se cierra la llamada

Conmutación por Paquetes. la información es dividida en pequeños fragmentos, que se envían a una red compartida.

Contact Centre. Centro de atención que abarca las funciones de un Call Centre y añade el manejo de comunicaciones por correo electrónico, fax, vídeo, etc.

Contexto. En Asterisk un contexto es una sección del plan de marcación que permite separar las extensiones definidas en el mismo de las de otras secciones en la central telefónica.

CTI Integración Teléfono Computador. une un conmutador con servicios en computadores externos de manera abierta

Dialplan. ver plan de marcación,

Extensión. en Asterisk, una extensión es una instrucción que el programa debe seguir y es disparada por una llamada entrante o dígitos marcados desde un canal o dispositivo.

Los dispositivos como teléfonos también son conocidos como Extensiones.

Floppy. disco flexible

Híbridos. Sistemas que mantienen un conmutador y servicios específicos en servidores externos.

Linux. Sistema Operativo

MacOSX. Sistema operativo de Macintosh

Macros. Construcciones que permiten el desarrollo de actividades repetitivas, permiten reducir la cantidad de código en el plan de marcación.

Mainboard. Tarjeta Madre, la parte central del computador que permite unir todas las partes que lo componen.

Memoria. Dispositivo electrónico que retiene información.

PBX. (Private Brand Exchange) es un computador especializado para la administración de conexiones telefónicas.

Plan de Marcación. Listado de instrucciones que la central telefónica debe seguir para el manejo de llamadas entrantes y salientes de los distintos canales que maneja.

Procesador. Parte de un computador que controla la realización de instrucciones de programas.

Servidor de Comunicaciones. equipo encargado de manejar un sistema telefónico que usualmente corre sobre un servidor estándar con sistemas operativos conocidos, orientado a software.

Solaris. Sistema Operativo

Unix. Sistema Operativo

Supervisor. es el encargado de controlar la calidad del servicio recibido por el cliente

Tarjeta de Red. Dispositivo encargado de manejar la conexión de un computador a otros equipos en una red.

Telemarketing. Ventas y servicios conducidos a través del teléfono.

VOIP. término utilizado en telefonía para referirse a un conjunto de instalaciones que permiten la transmisión de voz sobre redes de datos.

ESCUELA POLITÉCNICA DEL EJÉRCITO
INGENIERÍA EN ELECTRÓNICA Y TELECOMUNICACIONES

ENTREGADO

Ing. Gonzalo Olmedo

COORDINADOR DE CARRERA

Sr. Daniel Verdesoto R

AUTOR