



ESCUELA SUPERIOR POLITÉCNICA DEL LITORAL CENTRO DE INVESTIGACIÓN CIENTÍFICA Y TECNOLÓGICA



Diseño e Implementación de una Solución de VoIP para un Call Center que Brindará Soporte a Usuarios

Diego Jiménez Plúas ⁽¹⁾ Daniel Díaz Brito ⁽²⁾ Gabriel Astudillo Brocel ⁽³⁾

Facultad de Ingeniería en Electricidad y Computación (FIEC) ^{(1) (2) (3)}

Escuela Superior Politécnica del Litoral (ESPOL)

Campus Gustavo Galindo, Km 30.5 vía Perimetral

Apartado 09-01-5863. Guayaquil, Ecuador

diarjime@espol.edu.ec ⁽¹⁾, dediaz@espol.edu.ec ⁽²⁾ gastudil@espol.edu.ec ⁽³⁾

Resumen

El presente trabajo consiste en la implementación de un Call Center y una central telefónica basada en Asterisk, que una vez realizada, permitió demostrar que es posible brindar un servicio con alta disponibilidad, mediante el uso de herramientas de código abierto. Al inicio de este proyecto se estableció que la implementación de la infraestructura del call center, nos permitiría otorgar a los clientes todas las facilidades para hacer uso de nuestro servicio, manejando un control de llamadas y manteniendo un registro de tiempos de servicios a través de un sistema de alta disponibilidad.

Para cumplir con estos aspectos, se determinó que las herramientas a usar, debido a sus características y aporte al propósito, serían Elastix junto con su módulo de call center, clustering y heartbeat. Para lograr esta implementación se hizo uso de Elastix como herramienta de administración gráfica, además de la adecuación de los distintos archivos de configuración pertenecientes al sistema operativo donde fue realizada. Logrando demostrar que mediante el uso de herramientas de código abierto, es posible brindar un servicio con alta disponibilidad.

Palabras Claves: Call Center, Asterisk, Elastix, Clustering, Hearbeat

Abstract

The work consists on the implementation of a Call Center and a Private Branch Exchange, Asterisk based; once completed, sustain that it is possible to offer a highly available service, with open source tools. It was established, at the beginning of the project, that the infrastructure would allow the costumers of the Call Center with access to all the benefits of the solution, so they can gain control of calls and maintain a service usage log.

To accomplish the goals, it was determined that the tools to be used, based on their characteristics and inputs to the project, would be Elastix together with its call center module, clustering and heartbeat. To achieve this implementation, was necessary to adapt certain operating system configuration files as well as the configuration of Elastix, as the graphic user interface for the administration of the solution. Signifying that it is possible to offer a high available service, with the use of open source tools.

1. Introducción

Toda empresa privada tiene un fin lucrativo por lo que es importante implementar herramientas de bajo costo que le permitan cumplir con sus estándares de calidad.

Cuando se habla de empresas de servicios, se toman en cuenta productos intangibles cuya calidad se verá medida y valorada por la satisfacción del cliente.

Al tratarse de un call center, esta satisfacción se ve afectada por varios factores como la atención de los operadores, el tiempo de respuesta y la solución de problemas.

Para los aspectos cuantificables es importante implementar herramientas de control, que nos permitan evaluar resultados concretos, con el propósito de corregir o mejorar el servicio.

Siendo Asterisk una herramienta de código abierto, se puede realizar la implementación de un módulo de call center integrado a través de Elastix, para cumplir así con las necesidades del usuario a un bajo costo para la empresa.

2. Metodología

Se utilizarán tres equipos físicos, dos serán los servidores en los cuales estará instalado Elastix, y a su vez Asterisk y el module de Call Center y el tercero será el Gateway que nos proporcionará la conexión hacia la red de telefonía pública conmutada. Los dos servidores se encontrarán formando un clúster con el servicio Asterisk y Call Center, el cual tendrá una IP Virtual, al cual los usuarios se conectarán. Estos servidores contarán con una base de datos sincronizada y compartida en MySQL, la cual nos asegurará que siempre tendrán ambos servidores la misma información. El diseño se muestra en la Figura 1.

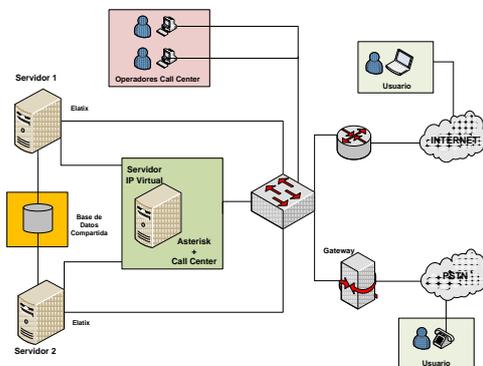


Figura 1 Diseño de Red Call Center

3. Asterisk

Asterisk es un software libre de código abierto que convierte un computador en una central telefónica IP. Soporta una amplia gama de protocolos de telefonía IP que incluye alta compatibilidad con el manejo y transmisión de voz sobre interfaces de telefonía tradicional en particular las líneas analógicas, líneas RDSI-BRI y troncales digitales T1/E1. Soporta una amplia gama de protocolos de (VoIP) como SIP, IAX2, H.323 y MGCP (Fig. 2).

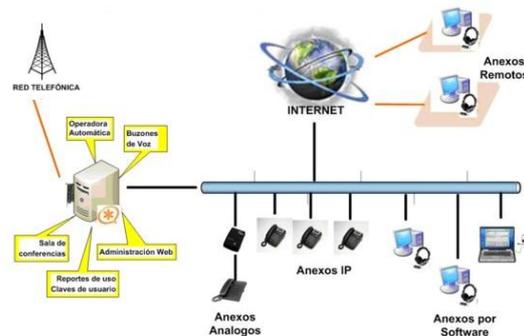


Figura 2 Conexión de Asterisk

Asterisk nos permite conectar teléfonos analógicos convencionales, a través del uso de tarjetas conectadas al servidor con puertos FXO o FXS, a la red IP.

Los puertos FXS tienen la capacidad de generar una señal eléctrica que hace que se genere un timbre en el teléfono, estos emulan las mismas señales que en las líneas telefónicas tradicionales analógicas, por lo que se conectan a ellos todo tipo de dispositivos que necesitan de ese timbre: teléfonos analógicos y faxes.

Por otro lado los puertos FXO se comportan como terminales, estos necesitan del timbre que desarrollan el comportamiento de las llamadas. Se conectan a ellos las líneas analógicas de telefonía tradicional, también extensiones analógicas de centrales PBX.

En la Figura 3 se puede apreciar los distintos tipos de dispositivos que se pueden conectar a los puertos FXS y FXO.

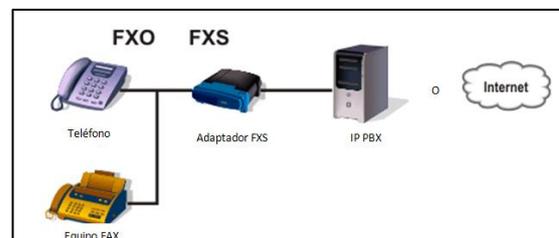


Figura 3 Puertos FXO / FXS

Las principales características de Asterisk se nombran a continuación:

- Funciona como un servidor dedicado VoIP.
- Soporta hasta 150 extensiones por cada sistema Asterisk y más de 150 cuando se trabaja con varios servidores (clustering), dependiendo de la robustez del equipo.
- Ofrece la capacidad de interconectar la red telefónica convencional hacia nuestra red IP y viceversa.
- Soporta líneas de telefonía analógicas, líneas RDSI, VoIP (voz a través de Internet).
- Ofrece opciones como: No molestar y llamada en espera.
- Capacidad para realizar conferencias (2 o más usuarios simultáneamente).
- Identificador de llamadas (CallerID).
- Música en espera y en transferencia de llamadas con la posibilidad de introducción de archivos actualizables por el usuario.
- Buzones de voz.
- Videoconferencia con protocolos SIP e IAX2.
- Funcionalidades para trabajar con Bases de Datos (MySQL).

Entre las principales ventajas que se pueden nombrar se muestran a continuación:

- Puede disponer desde funcionalidades básicas de una central como desvíos de llamadas, capturas, transferencias, multi-conferencias, hasta las más avanzadas tales como Buzones de voz, etc.
- Permite escalar más fácilmente una gran cantidad de canales de voz entre los puntos finales y nos ayuda a crecer en interconexión añadiendo otros servidores con planes de marcados diferentes sucursales.
- Es mucho más bajo en lo que a costo compete debido que es un sistema de código abierto, utiliza desde una plataforma de hardware estándar hasta potentes servidores y tarjetas para las interfaces de telefonía ya sean estas analógicas o digitales, lo que ha ido abaratando los costos de esta tecnología.

4. Clustering, Heartbeat y DRBD

Un servicio que debería estar operativo 24/7 debe mantener un esquema de alta disponibilidad estructurado correctamente. Implementaremos este esquema a través de dos herramientas que nos aseguran la alta disponibilidad. La primera será realizar clustering a través de Heartbeat y la segunda será utilizar DRBD el cual es un sistema de sincronización entre clusters para que los servidores estén constantemente actualizados y sincronizados.

El concepto de clustering se aplica a un grupo de computadoras construidas mediante la utilización de componentes comunes de hardware y que se comportan como si fuesen una única computadora.

Los clúster son empleados para mejorar el rendimiento y/o la disponibilidad por encima de la que se espera de un solo computador.

Un clúster puede brindar los siguientes servicios:

- Alto rendimiento
- Alta disponibilidad
- Balanceo de carga
- Escalabilidad

La creación del clúster es fácil y económica debido a su flexibilidad, estos se pueden clasificar de la siguiente forma:

- Clúster homogéneo: todos tienen la misma configuración de hardware y sistema operativo.
- Clúster semi-homogéneo: diferente rendimiento pero con arquitecturas y sistemas operativos similares.
- Clúster heterogéneo: tienen diferente hardware y sistema operativo.

En resumen un clúster es un grupo de varios computadores unidas por redes de alta velocidad, de tal forma que el grupo es visto como una sola computadora, más potente que los comunes de escritorio.

HeartBeat es una aplicación de fuente abierta que permite configurar sistemas de Alta Disponibilidad a través de un clúster, mediante el cual se ofrece procesos de comunicación y monitoreo de los nodos que están formando el clúster.

Como se observa en la Figura 4, éste funciona configurando dos o más servidores como maestro y el resto como esclavo.

HeartBeat usa una dirección IP lógica (Alias) la cual es configurada dentro de los parámetros y a la cual los clientes enviarán sus requerimientos, gracias a esto, como ya fue indicado anteriormente, los clientes lo verán como un servidor más en la red. La comunicación entre los servidores puede ser punto a punto y la comunicación puede ser realizada a través de un cable de red o serial.

HeartBeat envía pequeños paquetes de control, los cuales verifican que el servidor maestro esté funcionando. Estos paquetes requieren que el servidor en cuestión responda, caso contrario HeartBeat determinará que el servidor maestro está caído y automáticamente levanta el servidor esclavo para que asuma el control de la red.

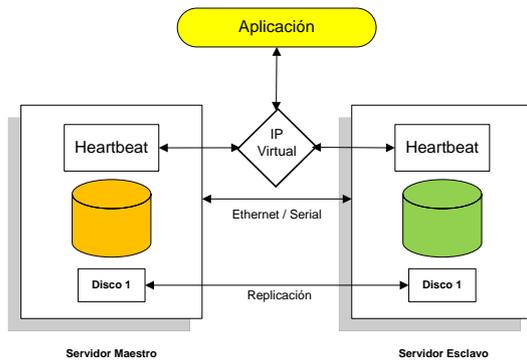


Figura 4 Esquema del funcionamiento Hearbeat

DRBD se refiere a dispositivos de bloque concebido como un bloque de construcción para formar clusters de alta disponibilidad (HA). Esto se hace por reflejo de un dispositivo de bloque entero a través de una red asignada. DRBD se puede entender como una red basada en RAID-1.

En la Fig. 5, las dos ilustraciones de color naranja representan dos servidores que forman un clúster HA. Las ilustraciones contienen los componentes habituales de un kernel Linux: sistema de archivos, cache, planificador de disco, controladores de disco, protocolo TCP / IP y una tarjeta de interfaz de red (NIC). Las flechas en negro ilustran el flujo de datos entre estos componentes.

Las flechas de color naranja muestran el flujo de datos, como espejos DRBD los datos de un servicio de alta disponibilidad desde el nodo activo del clúster HA al nodo en espera del clúster HA.

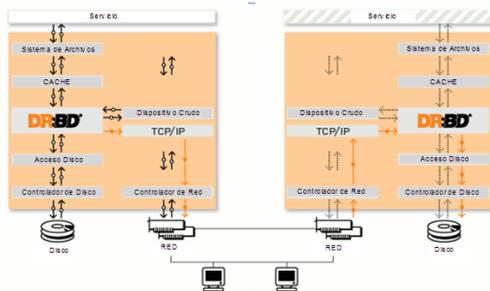


Figura 5 Esquema del funcionamiento Hearbeat

5. Elastix

Elastix fue creado y actualmente es mantenido por la compañía ecuatoriana PaloSanto Solutions. Elastix es una distribución libre de Servidor de Comunicaciones Unificadas que integra en un solo paquete:

- VoIP PBX
- Fax

- Mensajería Instantánea
- Email
- Colaboración

Elastix implementa gran parte de su funcionalidad sobre 4 programas de software muy importantes como son Asterisk, Hylafax, Openfire y Postfix. Estos brindan las funciones de PBX, Fax, Mensajería Instantánea e Email, respectivamente. (Fig. 6)

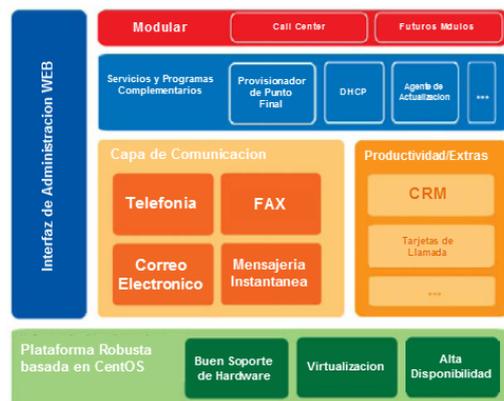


Figura 6 Componentes de Elastix

Elastix posee un módulo de Call Center (Fig. 2.6), que permite la integración con herramientas de CRM (Customer Relationship Management) basadas en servicios web.

6. Hardware

Para esta implementación los requisitos del hardware podrían considerar el tema de escalabilidad en el tiempo, para esto consideramos equipos que cumplan con características de rendimiento óptimas. Asterisk, por ser en si un sistema liviano, podría ser implementado con equipos de características mínimas, pero para este proyecto utilizaremos un hardware robusto.

Para nuestra implementación utilizaremos 2 equipos con las siguientes características:

Tabla 1 Características de Servidores

Procesador	Intel Quad Core 4 GHz
Memoria RAM	4 GB
Disco Duro	250 GB
Disco Duro	120 GB
Tarjeta de Red	10/100/1000 Mbps

Para la comunicación con la red telefónica pública conmutada utilizaremos un Gateway analógico GrandStream GXW4108.

Este dispositivo está equipado con 8 puertos FXO. Cada uno de los puertos puede tomar la señal de una línea fija analógica.

Aparte utilizamos el teléfono IP GrandStream BT200 que nos permite utilizar todas las funcionalidades que brinda Asterisk.

7. Componentes para la implementación

Previo a la instalación de Elastix, se recomienda haber creado tres particiones, debido que una va a ser utilizada por Elastix y las otras dos por DRBD.

Las características de estas particiones son las siguientes:

Tabla 2 Características de Particiones de Disco

Partición	Tamaño Mínimo	Tipo de Sistema de Archivos
Disco 1 Partición 1	20 Gb	Ext3
Disco 2 Partición 1	400 Mb	Ext3
Disco 2 Partición 2	160 Mb	Ext3

7.1. Configuración de Red

Como segundo paso en el proceso de configuración de Elastix, es necesario configurarle la dirección IP a nuestros servidores. Este sencillo paso se lo realiza editando el archivo de red del servidor que se lo encontrará en la siguiente ruta: `/etc/sysconfig/network-scripts/ifcfg-eth0`.

El archivo debería quedar con los siguientes parámetros:

```
# Advanced Micro Devices [AMD] 79c970
[PCnet32 LANCE]
DEVICE=eth0
HWADDR=00:0C:29:87:F5:D3
ONBOOT=yes
IPADDR=192.168.27.121
NETMASK=255.255.255.0
NETWORK=192.168.27.0
GATEWAY=192.168.27.254
DNS1=200.25.197.8
```

Para nuestro caso, el segmento de red a utilizar es la red 192.168.27.0/24.

Editar luego el archivo `"/etc/hosts"` y agregar las siguientes líneas:

```
192.168.27.121    elaxtix-voip
192.168.27.122    elaxtix-voip-2
```

Las cuales representan el nombre de host que va a tener cada servidor.

```
elaxtix-voip      = servidor maestro
elaxtix-voip-2   = servidor esclavo
```

7.2. Configuración DRBD

El nodo 1 será **elaxtix-voip** cuya IP local es 192.168.27.121. Utilizaremos las dos particiones creadas (sdb1 y sdb2) una para las copia de los datos y la otra como meta disco para guardar los datos que DRBD crea al momento de conectarse y actualizarse con el segundo nodo.

El nodo 2 será **elaxtix-voip-2** cuya IP local es 192.168.27.122.

Ahora creamos los meta-datos en la partición xvdd de ambos servidores:

```
drbdadm create-md data
```

Ahora modificamos el script de arranque automático de DRBD:

```
vi /etc/init.d/drbd
```

Modificando esta línea:

```
$MODPROBE -s drbd `DRBDADM sh-mod-params` $ADD_MOD_PARAM || {
```

Para que quede:

```
$MODPROBE -f drbd `DRBDADM sh-mod-params` $ADD_MOD_PARAM || {
```

Terminamos configurando el sistema para que el script se inicie al iniciarse el sistema operativo:

```
chkconfig --add drbd
```

```
chkconfig drbd on
```

Para crear una primera sincronización entre los datos de las dos particiones de los dos servidores cargamos el módulo DRBD en el Kernel:

```
modprobe -f drbd
```

Confirmamos que efectivamente se cargó:

```
lsmod
Module      Size Used by
drbd        247200 0
```

Ahora creamos la conexión entre los dos servidores:

```
drbdadm attach data
```

```
drbdadm syncer data
```

```
drbdadm connect data
```

Efectuamos una primera sincronización de los datos desde el servidor primario:

```
drbdadm -- --overwrite-data-of-peer primary data
```

Esta operación puede durar bastante tiempo dependiendo del tamaño de los discos. Podemos controlar en cualquier momento a que punto ha llegado con el comando:

```
cat /proc/drbd
```

Una vez terminada la sincronización de los dos recursos reiniciamos ambos servidores:

Reboot



7.3. Configuración Heartbeat

Para asegurarse de que la aplicación funcione correctamente se debe seguir al pie de la letra las siguientes indicaciones [3].

- 1- Ejecutar los siguientes comandos para descargar e instalar la aplicación:

```
#yum -y install heartbeat
```

- 2- Realizamos copias de los siguientes archivos:
 - Cp /usr/share/doc/heartbeat-2.x.x/authkeys /etc/ha.d/
- 3- Editamos el archivo /etc/ha.d/authkeys
 - #vi /etc/ha.d/authkeys

Auth 3

3 md5 password

- 4- Editamos el archivo ha.cf , /etc/ha.d/ha.cf, el cual es muy importante ya que aquí le diremos al programa los tiempos que debe esperar para levantar el equipo esclavo en caso de fallo y la interfaz por la que transmitirá, entre otros. Agregamos las siguientes líneas:

```
Logfile /var/log/ha-log
logfacility local0
keepalive 2
deadtime 30
initdead 120
ucast eth0 192.168.27.XX
udpport 694
auto_failback on
node elaxtix-voip
node elaxtix-voip-2
```

- 5- La parte final de la configuración de Heartbeat es decirle que “demonios o servicios” queremos que estén con alta disponibilidad, en este caso para nosotros será “Asterisk y mysqld”. Editamos el archivo /etc/ha.d/haresources y agreguemos la siguiente línea:

```
elaxtix-voip IPaddr2::192.168.27.120/24
drbddisk::data
Filesystem::/dev/drbd1::data::ext3 asterisk
mysqld
```

Definimos como servidor Master elaxtix-voip e indicamos la IP virtual. drbddisk es el script que se encarga de montar la partición donde hemos guardado los archivos de configuración de Asterisk, en mi caso /dev/drbd1. Por último indicamos los servicios que Heartbeat tiene

que arrancar, en nuestro caso Asterisk y MySQL.

7.4. Configuración de MySQL

Antes que nada para comenzar a migrar las bases de datos de un sistema de archivos a otro hay que detener el servicio de MySQL con la consola de administración de la misma, pero esto lo puede hacer solo un usuario con los permisos adecuados [4].

Al momento no hay ningún usuario que pueda hacer esto, por lo que necesitamos asignar estos privilegios al usuario root. Lo hacemos ejecutando lo siguiente:

```
[root@elaxtix-voip ~]# mysql -u root -p
Enter password:
Welcome to the MySQL monitor. Commands end
with ; or \g.
Your MySQL connection id is 9
Server version: 5.0.77 Source distribution
Type 'help;' or '\h' for help. Type '\c' to clear the
buffer.
mysql>
mysql> GRANT SHUTDOWN ON *.* TO
'root'@'localhost' IDENTIFIED BY 'diegormando'
Query OK, 0 rows affected (0.01 sec)
mysql> exit
```

Debido que se está migrando una instalación existente de MySQL, detener el servicio con el siguiente comando: shell> `mysqldadmin shutdown`

Copiar el archivo my.cnf dentro del dispositivo DRBD y darle los permisos adecuados:

```
[root@elaxtix-voip ~]# mkdir /data/mysql
[root@elaxtix-voip ~]# cp /etc/my.cnf /data/mysql
[root@elaxtix-voip ~]# chmod 644 -R /data/mysql
[root@elaxtix-voip ~]# chmod 644 /data/mysql/my.cnf
```

Copiar el directorio de datos de MySQL hacia el dispositivo montado de DRBD.

```
[root@elaxtix-voip ~]# cp -R /var/lib/mysql
/data/mysql/data
```

Editar el archivo de configuración de MySQL para reflejar el cambio de directorio de datos de éste.

```
datadir = /data/mysql/data
log-bin = mysql-bin
```

Cambiar el nombre del archivo original para mantenerlo como respaldo y crear un enlace simbólico dentro de “/etc” que apunte a la nueva ruta del archivo “my.cnf”:

```
[root@elaxtix-voip ~]# cd /etc
[root@elaxtix-voip etc]# mv my.cnf my.cnf.old
[root@elaxtix-voip etc]# ln -s /data/mysql/my.cnf
/etc/my.cnf
```

Cambiar la configuración del script de inicio del servicio “/etc/init.d/mysqld” para que apunte a la ubicación por default del socket, puesto que si no se le

edita este parámetro, el servicio no iniciará correctamente:

Antes:
`get_mysql_option mysqld socket "$datadir/mysql.sock"`

Después:
`get_mysql_option mysqld socket "/var/lib/mysql/mysql.sock"`

Ahora por último iniciar el servicio de MySQL y chequear que los datos que se copiaron estén presentes en el sistema de archivos creado.

`[root@elastix-voip etc]# /etc/init.d/mysqld start`

Nótese que no se puede acceder desde el servidor secundario al recurso compartido de DRBD, sin embargo la información ya está replicada en éste.

7.5. Creación de Extensiones

Las extensiones nos facilitan la comunicación hacia los miembros de la red corporativa que dispongan de un teléfono IP.

Pasos a seguir:

1. En la pestaña PBX – PBX Configuración.
2. Del menú Device se escoge el tipo de dispositivo que se espera conectar a la nueva extensión. Para este ambiente seleccionamos Generic SIP Device y presionamos el botón Submit (Fig. 7).



Figura 7 Creación de extensión

7.6. Creación de troncales

Pasos a seguir:

1. En la pestaña PBX – PBX Configuración - Trunks.
2. Se escoge Add Trunk
3. Se completan los campos del formulario.

8. Pruebas de la consola de administración de Elastix

Con los dos escenarios propuestos se puede comprobar la diferencia de accesos que existen entre un usuario administrador y uno con menos privilegios.

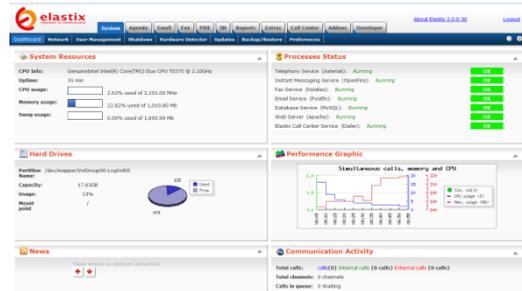


Figura 8 Consola de administración Elastix

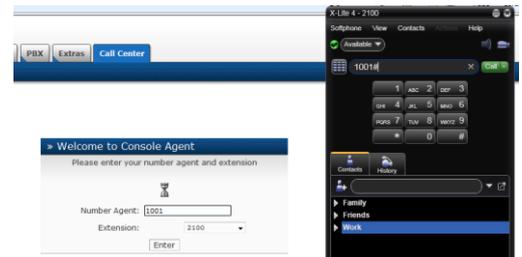


Figura 9 Consola de administración agentes Call Center

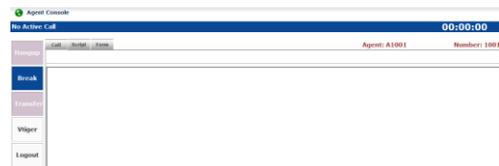


Figura 10 Consola de agentes



Figura 11 Lista de Colas de llamadas

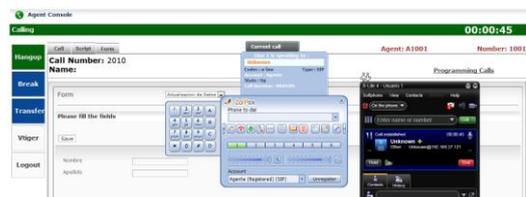


Figura 12 Campaña activa



ESCUELA SUPERIOR POLITÉCNICA DEL LITORAL

CENTRO DE INVESTIGACIÓN CIENTÍFICA Y TECNOLÓGICA



9. Recomendaciones

1. Determinar qué equipo utilizar y sus características de acuerdo a las necesidades del cliente, puesto que puede ser un entorno con 100 usuarios a uno con 1000 usuarios y todas las funcionalidades posibles que este pueda ofrecer.
2. Contratar el número suficiente de líneas telefónicas que puedan ser usadas por el Call Center, para que no exista una pelea continua por los recursos de este al momento de querer realizar llamadas entrantes como para salientes.
3. Utilizar en lo posible servidores con los mismos recursos de hardware, a pesar de no ser un requerimiento.
4. La partición creada que será utilizada por DRBD, tiene que ser estrictamente igual, la misma cantidad de sectores, tamaño y tipo de sistema de archivos en ambos equipos.
5. Realizar chequeos rutinarios al recurso compartido y mantener siempre los servidores sincronizados para que no exista un problema muy común en DRBD llamado "Split-Brain", el cual es causado por una mala sincronización del recurso compartido por ambos servidores.

10. Bibliografía

- [1] Vivek Gite, Linux: How Do I Create a New ext3 File System If a Disk Was Added To The Server/Desktop, <http://www.cyberciti.biz/faq/redhat-centos-linux-ext3-file-system-format-command/>, 6 de marzo del 2008
- [2] Andrea Sannucci, Asterisk alta disponibilidad - DRBD - Raid1 vía TCP en Linode, <http://www.voztovoice.org/?q=node/276>, 13 de abril del 2010
- [3] OpenVZ Wiki, HA cluster with DRBD and Heartbeat, http://wiki.openvz.org/HA_cluster_with_DRBD_and_Heartbeat, 1 de septiembre del 2010
- [4] Md. Abdullah Al Mamun, How to configure MySQL High Availability with DRBD and Heartbeat on Centos 5.3, <http://almamunbd.wordpress.com/2009/05/28/how-to-configure-mysql-high-availability-with-drbd-and-heartbeat/>, 28 de mayo del 2009
- [5] Juan Oliva, Elastix Callcenter "La guía total", <http://jroliva.wordpress.com/howto-elastix-callcenter-%C2%A8la-guia-total%C2%A8/>, 18 de abril del 2009