



freePBX™

Juan Miguel Alcarria (que pasa churra)

INDICE

ACLARACIONES INICIALES.....	3
INSTALACIÓN DEL SISTEMA OPERATIVO.....	4
INSTALACIÓN DE PAQUETES VARIOS.....	4
INSTALACIÓN DE PAQUETES PARA COMPILAR ASTERISK Y ZAPTEL	5
INSTALACIÓN DE ASTERISK Y ZAPTEL	6
CONFIGURACIÓN ZAPTEL Y ASTERISK	11
INSTALACIÓN DE FREEPBX	15
CONFIGURACIONES PARA EL HARDWARE.....	20
SCRIPTS INICIALES:.....	23
CONFIGURACIÓN PARA PROBAR EL SISTEMA	25
Para añadir una extension:.....	25
Para añadir un trunk para el FXO:.....	25
Para añadir un trunk para RDSI o Billion:	26
Rutas Entrantes:.....	26
Rutas Salientes (FXO):.....	26
Rutas Salientes (Billion):.....	26
AGRADECIMIENTOS:.....	26

En este manual os enseñamos a configurar una centralita telefónica basada en la nueva tecnología **VoIP**. Como siempre montaremos el servicio con el sistema operativo **Debian**, como no podía ser de otra forma.

El documento explica como configurar **Asterisk** para que sea usado con **FreePBX**, portal que nos ayudara muchísimo con la administración de Asterisk. También hablamos de como configurar los drivers para las tarjetas OpenVox (clon de las tarjetas de Digium, módulos FXO y FXS para líneas RTB) y tarjetas RDSI o ISDN compatibles con el driver de Junghanns o también llamado Bristuff, entre estas tarjetas podemos encontrar desde simples tarjetas rdsi, que manejan dos canales, hasta tarjetas controladoras de primarios (30 líneas por primario, puffff).

Todo el sistema de telefonía será compilado y parcheado para su buen funcionamiento, esperemos que de por vida.

Si deseas implantar una centralita telefónica en tu trabajo o simplemente ponerla en tu casa, **lee el documento** y prueba. **El coste es ridículo** y podrás hacer verdaderas virguerías como por ejemplo: listas negras de teléfono, recepcionistas digitales (Si quiere hablar con fulanita pulse el 1...), buzón de voz directamente a tu buzón de correo, condicionantes de tiempo (Si es la hora de la siesta y es la mujer de telefonía preguntandome que si me quiero dar de alta que salte este mensaje...) salas de conferencia y el sinfín de posibilidades que nos otorga la voz ip unida con el magnífico paquete **GNU** llamado asterisk (gracias a **Digium**).

No perdamos más tiempo pongámonos manos a la obra:

ACLARACIONES INICIALES:

Para el código que veréis en el howto utilizo lo siguiente: código que tecleo (**NEGRITA + CURSIVA**).

Para el código que muestra la pantalla (**CURSIVA**).

Este manual compilara asterisk, zaptel y todos los drivers de las tarjetas fabricadas por Junghanns, entre estos drivers se encuentra el que necesitamos, zaphfc, también te instalara los módulos para tarjetas controladoras de primarios tarjetas pci a gsm y un montón mas.

Se compilaran los paquetes, ya que asterisk se lleva mal con las actualizaciones del sistema y después de lo que cuesta configurarlo si me lo fastidia un apt-get upgrade me da algo.

Aclarar que para la instalación de los módulos la Billion (tarjeta pci a rdsi de Junghanns) necesitaremos parchear asterisk y zaptel mas de una vez. Si tu instalación no va a tener Billion o tarjetas rdsi, no tendrás que parchear nada, solo compilar.

Después de leer esto os preguntareis por que no he utilizado una de las compilaciones que existen por ahí, por ejemplo Trxibox 2.2 o AsteriskNOW de Digium, el caso es que lo he intentado, pero ocurre lo de siempre, prefiero debian, montarme paquete a paquete yo y de esa forma saber que tengo en el servidor. Además con esta distribución (yo utilice trixbox) los problemas que me surgieron fueron bastantes, el problema gordo fue la unión del driver Bristuff para las tarjetas rdsi, trixbox tenia un script muy currado para la instalación de ese driver pero las versiones de asterisk zaptel parches y demás no se correspondían y acabaron con un problema de eco en las líneas bastante difícil de aguantar. En fin por una cosa o por otra me decidí a montar los servicios de telefonía en Debian. Si solo vais a usar la tarjeta OpenVox para líneas RTB normales y corrientes podéis probar antes con una de estas distribuciones os resultara menos

arduo y si el tema os sigue molando pasarlo a Debian con el manual que os dejo.

OJO a los Saltos de Linea

INSTALACIÓN DEL SISTEMA OPERATIVO

La instalación del sistema operativo será la estándar.

INSTALACIÓN DE PAQUETES VARIOS

Una vez terminada comprobaremos que nuestro sistema esta al día, para ello editaremos el archivo /etc/apt/sources.list

```
operadora:/# vi /etc/apt/sources.list
```

Retocaremos nuestros repositorios para que los coja de internet, no del cd. Tendréis que dejar algo así:

```
deb http://ftp.rediris.es/debian/ etch main  
deb-src http://ftp.rediris.es/debian/ etch main  
deb http://security.debian.org/ etch/updates main  
deb-src http://security.debian.org/ etch/updates main
```

Actualicemos los repositorios y actualicemos nuestro sistema:

```
operadora:/# apt-get update  
operadora:/# apt-get upgrade
```

Actualizad todo, si os pide actualizar el kernel hacerlo y reiniciar, es lo único que hace reiniciar a Debian. Una vez actualizado y reiniciado instalaremos ssh para acceder remotamente al nuevo servidor:

```
operadora:/# apt-get install ssh
```

Yo también instalo de primeras el paquete vim (VI mejorado) para no perderme con la edición:

```
operadora:/# apt-get install vim  
operadora:/# vi /etc/vim/vimrc
```

y descomento:

```
syntax on  
set background=dark
```

y añado debajo de syntax on la siguiente linea:

```
set noai
```

La primera para que nos pinte de colorines según el código que el lea, la segunda esta clara y la tercera es para eliminar el autoindent, cuando pegas código de otra ventana y tiene tabuladores el vim mete otro mas si no le pones esta opción.

INSTALACIÓN DE PAQUETES PARA COMPILAR ASTERISK Y ZAPTEL

Necesitaremos un montón de paquetes para compilar asterisk zaptel e instalar freepbx, la instalación de este portal nos facilitara la administración futura del servidor de una forma barbara, muy aconsejable. Comencemos:

```
operadora:/# apt-get install linux-headers-`uname -r` build-essential cvs libssl-dev libssl0.9.8  
libssl0.9.7 modconf php4 php4-mysql php4-gd php4-cgi php5 php5-mysql php5-gd php5-cgi mime-  
construct libxml2 libxml2-dev libtiff4 libtiff4-dev apache2 mysql-server mysql-client libmysqlclient15-  
dev php4-pear openssl perl bison libaudiofile-dev libncurses5-dev curl sox hdparm
```

Veréis por ejemplo que instalo php5 y php4, el caso es que si no instalas los dos se hace el lio a la hora de utilizar php-cgi, así que no queda mas remedio que instalar los dos. Podéis ver por ejemplo que instalamos las cabeceras del kernel que utilizamos, mysql que sera necesario para la instalación de freepbx.

Necesitaremos instalar unos paquetes necesarios para PERL, este tipo de paquetes no se instalan con apt-get, se hace con CPAN. Para la instalación de CPAN basta con realizar lo siguiente:

```
operadora:/# perl -MCPAN -e "install Net::Telnet"
```

Cuando demos esta orden al sistema nos freirá a preguntas sobre nuestro servidor, darle a todo por defecto (pulsad intro) hasta que veáis que os pregunta de que continente sois, pues contestad, luego os preguntara de que país, contestad, luego os preguntara que repositorio de PERL queréis utilizad, usad el que mas os guste. Una vez rellenado todo instalara el paquete Net::Telnet. Vamos a por otro par de paquete de perl, pero respirad tranquilo que las preguntas solo las hace una vez.

```
operadora:/# perl -MCPAN -e "install IPC::Signal"  
operadora:/# perl -MCPAN -e "install Proc::WaitStat"
```

Los paquetes que nos vayan haciendo falta a partir de ahora los iremos comentando mas adelante, pasemos a la instalación de asterisk y zaptel.

INSTALACIÓN DE ASTERISK Y ZAPTEL

Desde aquí, el manual cambiara, dependiendo del hardware que tengáis necesitareis una cosa u otra.

La instalación de asterisk que propongo es la versión mas actual dentro de la versión 1.2, ya que por ahora freepbx no esta para asterisk 1.4 y además esta ultima versión no es estable, así que ya sabéis, según el hardware que tengamos descargaremos una u otra versión de asterisk y zaptel pero siempre de la versión 1.2.

Necesitaremos descargar los siguientes paquetes, yo los descargo al directorio /usr/src

Si no vas a utilizar RDSI necesitaras estos dos paquetes, los que vayáis a utilizar rdsi pasad de estas lineas, la versión que pongo es la estable a día de hoy os recomiendo que os paséis por digium y descarguéis al mas actual (siempre de al 1.2):

Solo los que NO vayan a usar la Billion:

```
operadora:/usr/src# wget http://ftp.digium.com/pub/asterisk/releases/asterisk-1.2.18.tar.gz
operadora:/usr/src# wget http://ftp.digium.com/pub/zaptel/zaptel-1.2.17.1.tar.gz
```

Descomprimiremos (Solo los que NO vayan a usar la Billion):

```
operadora:/usr/src# tar xvfz asterisk-1.2.18.tar.gz
operadora:/usr/src# tar xvfz zaptel-1.2.17.1.tar.gz
```

Los que vayan a utilizar la Billion descargar el paquete Bristuff que descargara automáticamente la versión correcta de asterisk y de zaptel (Solo los que utilicen RDSI).

```
operadora:/usr/src# wget http://www.junghanns.net/downloads/bristuff-0.3.0-PRE-1y.tar.gz
```

Descomprimimos (Solo los de Billion)

```
operadora:/usr/src# tar xvfz bristuff-0.3.0-PRE-1y.tar.gz
```

A partir de ahora los paquetes a descargar son comunes, debemos hacernos con los sonidos estándar de asterisk así como con sus aplicaciones extras:

```
operadora:/usr/src# wget http://ftp.digium.com/pub/asterisk/asterisk-addons-1.2.6.tar.gz
operadora:/usr/src# wget http://ftp.digium.com/pub/asterisk/old-releases/asterisk-sounds-1.2.1.tar.gz
```

Descomprimiremos los paquetes que acabamos de bajar:

```
operadora:/usr/src# tar xvfz asterisk-addons-1.2.6.tar.gz
operadora:/usr/src# tar xvfz asterisk-sounds-1.2.1.tar.gz
```

Aquí hay que hacer un parón, os cuento mi problema y como lo he solucionado, no os tiene porque pasar, pero lo que si es seguro es que si tenéis dos tarjetas billion a la vez os pasara casi seguro. Una vez realizados todos los pasos que indica este manual, el demonio de asterisk lanza un mensaje al syslog continuamente (un par de veces al minuto, depende del trafico que tengáis). El mensaje de error es uno de los siguientes, los números de bytes pueden cambiar pero el error es parecido:

```
zaphfc: bchan rx fifo not enough bytes to receive! (z1=6207, z2=6200, wanted 8 got 7), probably a buffer overrun
zaphfc: dropped audio (z1=6312, z2=6295, wanted 8 got 17 dropped9)
```

La solución para este problema es aplicar un parche a código SIN compilar del driver zaphfc, los que no use la rdsi no tienen porque leer esto, pero los que la usen y tengan un problema parecido a este deberán aplicar el parche antes de compilar el driver de zaptel para la tarjeta rdsi Billion. Os cuento la solución:

Descargamos el parche de Florian Zumbiehl (también llamado Florz' Patch for the zaphfc Driver), este parche se encuentra en la página de Florian, <http://zaphfc.florz.dyndns.org>

```
operadora:/usr/src# wget http://zaphfc.florz.dyndns.org/zaphfc_0.3.0-PRE-1o_florz-12.diff.gz
```

En el caso de este manual este parche es el que le conviene a la instalación de asterisk que realizara el paquete que descargamos de Bristuff, este script descargara, compilara e instalara la versión de asterisk numero 1.2.14 con lo cual según dice la pagina de Florian el parche adecuado es zaphfc_0.3.0-PRE-1o_florz-12.diff.gz que va de la versión 0.3.0-PRE-1o a la versión 0.3.0-PRE-1y-b del paquete de Bristuff, recordad que la versión que nosotros hemos bajado del paquete de Bristuff es la 0.3.0-PRE-1y, hemos acertado.

Debemos aplicar el parche de compilar zaptel con lo cual apliquemos el parche:

Nos moveremos al directorio que contiene el código a parchear:

```
operadora:/usr/src# cd bristuff-0.3.0-PRE-1y/zaphfc  
operadora:/usr/src/bristuff-0.3.0-PRE-1y/zaphfc# zcat ../zaphfc_0.3.0-PRE-1o_florz-12.diff.gz | patch  
-p1
```

Perfecto nos debe decir algo así:

```
patching file Makefile  
patching file zaphfc.c  
Hunk #2 succeeded at 33 with fuzz 2 (offset 4 lines).  
Hunk #3 succeeded at 77 (offset 4 lines).  
Hunk #4 succeeded at 114 (offset 4 lines).  
Hunk #5 succeeded at 125 (offset 4 lines).  
Hunk #6 succeeded at 186 (offset 4 lines).  
Hunk #7 succeeded at 205 (offset 4 lines).  
Hunk #8 succeeded at 218 (offset 4 lines).  
Hunk #9 succeeded at 236 (offset 4 lines).  
Hunk #10 succeeded at 458 (offset 4 lines).  
Hunk #11 succeeded at 476 (offset 4 lines).  
Hunk #12 succeeded at 493 (offset 4 lines).  
Hunk #13 succeeded at 506 (offset 4 lines).  
Hunk #14 succeeded at 527 (offset 4 lines).  
Hunk #15 succeeded at 538 (offset 4 lines).  
Hunk #16 succeeded at 586 (offset 4 lines).  
Hunk #17 succeeded at 654 (offset 4 lines).  
Hunk #18 succeeded at 698 (offset 4 lines).  
Hunk #19 succeeded at 737 (offset 4 lines).  
Hunk #20 succeeded at 752 (offset 4 lines).  
Hunk #21 succeeded at 762 (offset 4 lines).  
Hunk #22 succeeded at 776 (offset 4 lines).  
Hunk #23 succeeded at 825 (offset 4 lines).  
Hunk #24 succeeded at 865 (offset 4 lines).  
Hunk #25 succeeded at 872 (offset 4 lines).  
Hunk #26 succeeded at 918 (offset 4 lines).  
Hunk #27 succeeded at 936 (offset 4 lines).  
patching file zaphfc.h
```

Bueno despues de haber descargado todos los paquetes que necesitábamos para compilar y los que van a ser compilados es hora de compilar.

Para la gente que utilice el paquete Bristuff deberán hacer un par de cosillas antes:

Debemos crear un par de enlaces simbólicos a nuestras cabeceras del kernel, uno de ellos le hace falta al paquete de Bristuff y otro al parche de Florz:

```
operadora:/usr/src/bristuff-0.3.0-PRE-1y# ln -s /usr/src/linux-headers-2.6.18-4-686 /usr/src/linux-2.6  
operadora:/usr/src/bristuff-0.3.0-PRE-1y# ln -s /usr/src/linux-headers-2.6.18-4-686 /usr/src/linux
```

Ahora si, basta con ejecutar el install.sh dentro de su directorio:

```
operadora:/usr/src/bristuff-0.3.0-PRE-1y# ./install.sh
```

Es posible que la compilación nos de algún que otro warning, que paquete no lo da?

Bueno si todo ha ido bien, no nos da ningún error nos debe salir algo así:

```
+---- Asterisk Installation Complete -----+  
.....  
*****  
ASTERISK installed.  
Installation finished.  
*****
```

Como nos dice el cartel podemos instalar si queremos ejemplos de archivos de configuración. Yo los instalo, pero antes tenemos que irnos a la carpeta de asterisk para compilar los ejemplos:

```
operadora:/usr/src/bristuff-0.3.0-PRE-1y# cd asterisk  
operadora:/usr/src/bristuff-0.3.0-PRE-1y/asterisk# make samples
```

Ahora es el turno de la gente que no usara Bristuff.

Cread los enlaces simbólicos:

```
operadora:/usr/src/bristuff-0.3.0-PRE-1y# ln -s /usr/src/linux-headers-2.6.18-4-686  
/usr/src/linux-2.6  
operadora:/usr/src/bristuff-0.3.0-PRE-1y# ln -s /usr/src/linux-headers-2.6.18-4-686  
/usr/src/linux
```

Nos meteremos en el directorio de zaptel:

```
operadora:/# cd /usr/src/zaptel-1.2.17.1
```

Y ejecutaremos las siguientes ordenes:

```
operadora:/usr/src/zaptel-1.2.17.1# make clean all
```

..... (esperamos)

operadora:/usr/src/zaptel-1.2.17.1# make install

Una vez instalados los módulos de zaptel nos tiene que decir algo así:

```
***  
*** WARNING:  
*** If you had custom settings in /etc/modprobe.d/zaptel,  
*** they have been moved to /etc/modprobe.d/zaptel.bak.  
***  
*** In the future, do not edit /etc/modprobe.d/zaptel, but  
*** instead put your changes in another file  
*** in the same directory so that they will not  
*** be overwritten by future Zaptel updates.  
***
```

Compilemos ahora asterisk, nos metemos en su directorio:

operadora:/# cd /usr/src/asterisk-1.2.18

y ejecutamos las siguientes ordenes:

```
operadora:/usr/src/asterisk-1.2.18# make clean  
operadora:/usr/src/asterisk-1.2.18# make all  
operadora:/usr/src/asterisk-1.2.18# make install
```

Ya tenemos hecha la mitad del trabajo, tenemos todo compilado, nos falta instalar los módulos (decirle al sistema que los utilice), sobre esto hay que hacer un parón, os explico, las tarjetas RDSI pueden funcionar de 2 modos RDSI modo TE o RDSI NT, si lo que queremos es utilizar nuestra tarjeta RDSI para que recoja llamadas y asterisk las gestione necesitaremos poner la tarjeta en modo TE (es el caso mas normal), si queremos poner la tarjeta para pincharle un teléfono RDSI o un fax rdsi debemos ponerla en modo NT. Os cuento esto porque ahora que vamos a cargar los módulos la gente que quiera ponerlo en modo NT deberán hacerlo de otro forma:

Otra cosa ha tener en cuenta es el tipo de tarjeta que tenemos pinchada en nuestra tarjeta OpenVox, dos tipos son los posible FXO o FXS, exactamente igual que la tarjeta RDSI, FXO para pincharla directamente a la roseta de teléfono y FXS para pincharla a un teléfono o un fax.

El modulo zaptel lo necesitaremos todos (los que usan rdsi y los que no), instalémoslo

wctdm (los que usan OpenVox con FXO)

wcfxo (los que usan OpenVox con FXS, si parece que me he equivocado pero no lo he hecho, la señalización es FXO para módulos FXS)

Para cargar estos tres módulos lo haremos desde modconf:

operadora:/usr/src/asterisk-1.2.18# modconf

Los nuevos drivers están dentro de la carpeta misc

Los módulos de zaphfc (Solo RDSI) lo haremos de otro forma, no me funciona con modconf

Nos vamos al directorio de bristuff:

operadora:/usr/src# cd bristuff-0.3.0-PRE-1y/zaphfc/

Para los que use la tarjeta en modo TE pondremos lo siguiente:

operadora:/usr/src/bristuff-0.3.0-PRE-1y/zaphfc# make load

Para los que use la tarjeta en modo NT pondremos lo siguiente:

operadora:/usr/src/bristuff-0.3.0-PRE-1y/zaphfc# make loadNT

Para otros módulos (usb gsm,...) que tengáis es lo mismo, no me se los modos en los que pueden trabajar pero es fácil, con modconf se puede hacer sin problemas.

Ya tenemos cargados todos los módulos.

Debemos comprobarlo, instalaremos las aplicaciones de zaptel, este vez no tenemos que compilar, nos vale con un simple apt-get

operadora:/# apt-get install zaptel

Podemos utilizar una de las herramientas de zaptel para ver si por lo menos nos ha cogido bien las tarjetas:

operadora:/# zttool

debe salirnos algo así:

Alarms Span

UNCONFIGURED Wildcard TDM400P REV E/F Board 1

UNCONFIGURED HFC-S PCI A ISDN card 1 [TE]

UNCONFIGURED HFC-S PCI A ISDN card 2 [TE]

Como nos esta diciendo zttool las tarjetas están reconocidas pero no configuradas así que vamos a configurar a zaptel y a asterisk.

CONFIGURACIÓN ZAPTEL Y ASTERISK

Empezaremos con zaptel, debéis hacerlo todos, los que uséis rdsi o no, lo único que cambiara las configuraciones. El archivo de configuración de zaptel esta situado en /etc y es el archivo /etc/zaptel.conf, contendrá una información valiosa, así que yo lo que hago es copiarlo antes de modificar nada:

```
operadora:/usr/src/bristuff-0.3.0-PRE-1y/zaphfc# cp /etc/zaptel.conf  
/etc/zaptel.conf.ORIGINAL
```

Echémosle un primer vistazo al fichero de configuración:

```
operadora:/usr/src/bristuff-0.3.0-PRE-1y/zaphfc# cd /etc/  
operadora:/etc# vi zaptel.conf
```

Yo lo que hago es comentar las dos lineas que están si comentar (loadzone y defaultzone) y poner esto al final del archivo:

```
loadzone = es  
defaultzone=es
```

```
# Span 1: WCTDM "Wilcard whit 1 FXO"  
fxsks=1
```

```
# Span 2: ZTHFC1 "HFC-S PCI A ISDN card 1 [TE]"  
span=2,1,3,ccs,ami  
bchan=5-6  
dchan=7
```

```
# Span 3: ZTHFC2 "HFC-S PCI A ISDN card 2 [TE]"  
span=3,1,3,ccs,ami
```

bchan=8-9
dchan=10

Os cuento que significa todo esto, fxsks es mi modulo FXO en la primer banco pci de mi tarjeta OpenVox, las tarjetas PCI van en orden de menor a mayor, normalmente, según estén de cerca del procesador. La mas cercana al procesador según miramos la placa sera la 1 y así hacia arriba, os cuento esto porque a zaptel.conf hay que decírselo.

Diréis, si solo tengo un modulo FXO, porque la primera RDSI utiliza el canal 5, porque zaptel sabe que la tarjeta de OpenVox tiene para poner 4 módulos FXO o FXS y los reserva.

Si os fijáis lo único que cambia entre el Span2 y Span3 (ranuras pci 2 y 3) son estos números y sus canales. Haced vuestras cuentas y configurad como lo tengáis.

Una vez cambiado el archivo zaptel.conf podemos comprobar que todo va bien ejecutando otra de las aplicaciones de zaptel:

operadora:/etc# ztcfg -vvvvvvvvvvvvvv

Zaptel Configuration

=====

SPAN 1: CCS/AMI Build-out: 399-533 feet (DSX-1)

SPAN 2: CCS/AMI Build-out: 399-533 feet (DSX-1)

Channel map:

Channel 01: FXS Kewlstart (Default) (Slaves: 01)

Channel 05: Clear channel (Default) (Slaves: 05)

Channel 06: Clear channel (Default) (Slaves: 06)

Channel 07: D-channel (Default) (Slaves: 07)

Channel 08: Clear channel (Default) (Slaves: 08)

Channel 09: Clear channel (Default) (Slaves: 09)

Channel 10: D-channel (Default) (Slaves: 10)

7 channels configured.

Perfecto, zaptel esta perfectamente configurado. Pasemos ahora a configurar Asterisk.

Lo único que haremos por ahora es editar zapata.conf para comprobar que nos reconoce bien los canales dados por zaptel para despues hacer buen uso de ellos con freepbx.

Editaremos el archivo /etc/asterisk/zapata.conf

Haremos una copia previa antes de modificar nada:

operadora:/# cp /etc/asterisk/zapata.conf /etc/asterisk/zapata.conf.ORIGINAL

Configuremos el archivo:

```
operadora: # vi /etc/asterisk/zapata.conf
```

Yo pongo lo siguiente, es decir dejo en blanco el archivo de configuración y añado lo siguiente:

```
[channels]
language=es
; include zap extensions defined in AMP
#include zapata_additional.conf
; XTDM20B Port #1,2 plugged into PSTN
;AMPLABEL:Channel %c - Button %n
context=from-zaptel
signalling=fxs_ks
faxdetect=incoming
usecallerid=yes
echocancel=yes
echocancelwhenbridged=no
echotraining=800

group=0
channel=1
```

Para los que usen la tarjeta RDSI tienen que añadir esto en el mismo archivo (zapata.conf):

```
;Include BRI-HFC configs
#include zapata-BRI-HFC.conf
```

y tendrán que crear el archivo en cuestión con los siguientes datos:

```
; Basic ISDN Card TE mode config
resetinterval=never
immediate=no
switchtype=euroisdn
signalling=bri_cpe_ptmp
pridialplan=dynamic
prilocaldialplan=local
nationalprefix=0
internationalprefix=00
usecallingpres=yes
echocancel=yes
echocancelwhenbridged=yes
echotraining=100
context=from-zaptel
group=1
channel => 5-6
channel => 8-9
```

Los canales y como queráis ordenar los grupos es cosa vuestra, yo al tener 2 rdsi con 2 canales cada una con el mismo número de teléfono para los 4 canales (un grupo de salto de telefónica) meto a los 4 canales en un mismo grupo, pero vosotros podéis crear tantos grupos como canales, luego en asterisk nos referiremos a grupos y no ha canales, pero bueno todos estos cambios podremos ir haciéndolos después.

Es momento de comprobar si asterisk nos funciona y nos reconoce los canales pasados por zaptel.

```
operadora:/# asterisk  
operadora:/etc# asterisk -rvvvvvvvvvvvvvvvvv
```

```
== Parsing /etc/asterisk/asterisk.conf: Found  
== Parsing /etc/asterisk/extconfig.conf: Found  
Asterisk 1.2.14-BRIstuffed-0.3.0-PRE-1y, Copyright (C) 1999 - 2006 Digium, Inc. and others.  
Created by Mark Spencer <markster@digium.com>  
Asterisk comes with ABSOLUTELY NO WARRANTY; type 'show warranty' for details.  
This is free software, with components licensed under the GNU General Public  
License versión 2 and other licenses; you are welcome to redistribute it under  
certain conditions. Type 'show license' for details.  
=====  
Connected to Asterisk 1.2.14-BRIstuffed-0.3.0-PRE-1y currently running on operadora (pid = 25356)  
Verbosity is at least 16  
-- Remote UNIX connection  
operadora*CLI> zap show channels  
Chan Extension Context Language MusicOnHold  
pseudo from-zaptel es  
1 from-zaptel es  
5 from-zaptel es  
6 from-zaptel es  
8 from-zaptel es  
9 from-zaptel es  
== Primary D-Channel on span 2 down  
== Primary D-Channel on span 3 down  
operadora*CLI>
```

Perfecto, nos reconoce todos los canales, las 2 ultimas lineas donde nos dice Primary D-Channel on span 2 down, nos saldrán siempre que estamos en la consola de asterisk, significa que el canal digital está caído, no pasa nada nosotros utilizamos los otros dos.

Podemos realizar otro tipo de comprobaciones, como por ejemplo:

```
operadora*CLI> pri show span 2  
operadora*CLI> zap show channel 5
```

INSTALACIÓN DE FREEPBX

Bueno esta es la otra mitad del trabajo, cuando ya tenemos todo funcionando debemos instalar freepbx, esta herramienta no es solo un portal para realizar configuraciones, añade muchas herramientas a nuestro asterisk, su modo de funcionamiento es simple, todas las configuraciones que hacemos en sus paginas las guarda en una base de datos y cuando aplicamos las configuraciones escribirá estas en los ficheros de configuración de Asterisk, por ello cuando instalemos freepbx no deberiamos tocar los archivos de configuración, para ello nos dejara un nombre_de_archivo_de_configuración_custom.conf sera ahí donde freepbx no toque e incluirá en los archivos originales el custom.

Comencemos parando a asterisk.

```
operadora:~# kill -9 `pidof asterisk`
```

Para poder instalar freepbx debemos realizar un montón de modificaciones en archivos de configuración varios y modificar un montón de permisos sobre directorios. Comencemos diciéndole a php.ini de apache que acepte archivos de 40 mb (la musica en espera la subiremos por la web):

```
operadora:~# vi /etc/php5/apache2/php.ini
```

Buscaremos la linea que pone:

```
; Maximum allowed size for uploaded files.  
upload_max_filesize = 2M
```

y cambiaremos el valor 2 Megas por **40M**

Sigamos, tendremos que decirle al php.ini del paquete php4-cgi y php5-cgi que utilice la extension mysql.so (con esto le permitimos a este paquete realizar consultas sql contra el motor de bases de datos mysql):

```
operadora:/# vi /etc/php4/cgi/php.ini  
operadora:/# vi /etc/php5/cgi/php.ini
```

Buscaremos la linea en los dos archivos y la **descomentaremos** (quitad el punto y coma), esta es la linea a buscar:

```
;extension=mysql.so
```

Añadamos ahora un usuario que se llame asterisk y un grupo que se llame asterisk:

```
operadora:/# groupadd asterisk  
operadora:/# useradd -c "asterisk PBX" -d /var/lib/asterisk -g asterisk asterisk
```

El home del usuario sera donde están todos los archivos necesarios para que asterisk arranque.

Debemos crear el siguiente directorio para que freepbx no nos casque en la instalación, debian mete todos los procesos arrancados en /var/run y freepbx quiere que se sitúen en un directorio dentro de ese:

```
operadora:/# mkdir /var/run/asterisk
```

Editemos el archivo /etc/asterisk/asterisk.conf para decirle que coloque el proceso en el nuevo directorio:

```
operadora:/# vi /etc/asterisk/asterisk.conf
```

La linea a cambiar es la que dice

```
astrundir => /var/run
```

y la debemos dejar así:

```
astrundir => /var/run/asterisk
```

Es hora de nuevo de compilar paquetes que le hacen falta a Freepbx, si os acordais bajamos el asterisk-addons y el asterisk-sounds, compilemoslos, si no los tenéis descomprimidos tenéis que descomprimir los paquetes antes (tar xvfz nombre_a_descomprimir.tar.gz):

```
operadora:/# cd /usr/src/  
operadora:/usr/src# cd asterisk-addons-1.2.6
```

Tenemos que ejecutar un comando antes de compilar (según nos dice el manual de instalación de Freepbx):

```
operadora:/usr/src/asterisk-addons-1.2.6# perl -p -i.bak -e  
's/CFLAGS.*D_GNU_SOURCE/CFLAGS+=-D_GNU_SOURCE\nCFLAGS+=-  
DMYSQL_LOGUNIQUEID/' Makefile
```

*** OJO AL SALTO DE LINEA**

```
operadora:/usr/src/asterisk-addons-1.2.6# make clean  
....  
operadora:/usr/src/asterisk-addons-1.2.6# make  
....  
operadora:/usr/src/asterisk-addons-1.2.6# make install  
....
```

Compilemos ahora los sonidos:

```
operadora:/usr/src/asterisk-addons-1.2.6# cd ../asterisk-sounds-1.2.1  
operadora:/usr/src/asterisk-sounds-1.2.1# make install
```

Bajaremos el paquete de freepbx, en este momento el actual es la versión 2.2.1, si cogéis otra versión leeros el archivo de instalación y revisad que nos nos falta nada por hacer.

```
operadora:/usr/src# wget http://dfn.dl.sourceforge.net/sourceforge/ampportal/freepbx-  
2.2.1.tar.gz
```

Descomprimimos:

```
operadora:/usr/src# tar xvfz freepbx-2.2.1.tar.gz
```

Lo bajamos porque necesitamos código que tiene para crear las bases de datos, hagamos todo lo relacionado con la base de datos ahora:

Os aconsejo poner a todo el mismo password y así no os liareis, aun así yo os pongo que password deberíais poner por si queréis diferenciarlo.

```
operadora:/usr/src# mysqladmin -u root password 'TU_PASSWORD_MYSQL'  
operadora:/usr/src# mysqladmin create asteriskcdrdb -p  
operadora:/usr/src# mysql --user=root --password=TU_PASSWORD_MYSQL  
asteriskcdrdb < /usr/src/freepbx-2.2.1/SQL/cdr_mysql_table.sql  
operadora:/usr/src# mysqladmin create asterisk -p  
operadora:/usr/src# mysql --user root -p asterisk < /usr/src/freepbx-  
2.2.1/SQL/newinstall.sql  
operadora:/usr/src# mysql --user root -p
```

```
mysql> GRANT ALL PRIVILEGES on asteriskcdrdb.* TO asteriskuser@localhost IDENTIFIED BY  
'TU_PASSWORD_ASTERISK';  
Query OK, 0 rows affected (0.00 sec)  
mysql> GRANT ALL PRIVILEGES on asterisk.* TO asteriskuser@localhost IDENTIFIED BY  
'TU_PASSWORD_ASTERISK';  
Query OK, 0 rows affected (0.00 sec)  
mysql> FLUSH PRIVILEGES;  
Query OK, 0 rows affected (0.00 sec)  
mysql> quit
```

Debemos cambiar el usuario y el grupo con el que se ejecuta apache2, al cambiarlo le daremos permiso al portal para que interactúe con la consola de asterisk y pueda para los servicios y demás. Para ello editaremos este archivo:

```
operadora:/# vi /etc/apache2/apache2.conf
```

Buscaremos estas dos líneas:

```
User www-data  
Group www-data
```

Y **cambiaremos www-data por asterisk** para dejarlo así:

```
User asterisk  
Group asterisk
```

Reiniciaremos el servicio apache2

```
operadora:/# /etc/init.d/apache2 restart
```

Es momento de instalar Freepbx, al comienzo nos hará unas cuantas preguntas ya que no tiene el archivo /etc/amportal.conf

Incluso la instalación nos fallara diciendo que el archivo /var/lib/asterisk/bin/retrieve.conf no tiene permisos para acceder al archivo que el mismo acaba de crear /amportal.conf. Cuando se salga sin habernos instalado daremos permisos y volveremos a lanzar de nuevo la instalación. Ojo con los usuarios y contraseñas prestad atención a cual os pide y cuales son las nuevas que vais creando. El comando para lanzar la instalación es el siguiente:

Ahora si, lancemos la instalación, nos va a pedir si queremos sustituir los archivos de configuración existentes, decidle que si a todo (all):

```
operadora:/# cd /usr/src/freepbx-2.2.1  
operadora:/usr/src/freepbx-2.2.1# ./install_amp
```

Como casca, debemos dar permisos a los ficheros y cambiar propietarios, debemos realizar los siguientes pasos:

```
operadora:/# chmod 777 -R /var/lib/asterisk/bin  
operadora:/usr/src/freepbx-2.2.1# chmod 777 /etc/amportal.conf
```

Antes de lanzar de nuevo la instalación debemos levantar asterisk:

```
operadora:/# asterisk
```

Volvamos a lanzar la instalación:

```
operadora:/usr/src/freepbx-2.2.1# ./install_amp
```

No debería darnos ningún error, a lo mejor alguno de el op_flashpanel, no puede levantarlo, pero no debe darnos ninguno de permiso denegado ni nada parecido.

En el archivo que crea de configuración nos pregunta sobre en que directorio va a levantarse el portal, el por defecto lo deja en /var/www/html, yo lo he cambiado a /var/www/freepbx, tengamos donde lo tengamos debemos decirle que el dueño es el usuario y el grupo asterisk:

```
operadora:/# chown asterisk:asterisk -R /var/www/freepbx
```

Probemos a arrancar amportal y a pararlo:

```
operadora:/# amportal stop  
operadora:/# amportal start
```

Si os fijáis el amportal levanta asterisk, pone permisos a los ficheros que necesita y levanta el Flash Operator Channel.

Dejad levantado el amportal y probar ahora metiendo en un navegador web la siguiente dirección <http://tudireccionip/tudirectoriodeinstalacion>, en mi caso: <http://192.168.1.2/freepbx>. Cuando entremos a la zona de administración nos pedirá usuario y contraseña recordad admin y admin le da igual lo que configuremos en amportal :-).

Para cambiar el password debemos dejar el mismo en estos tres sitios:

```
/etc/amportal.conf (linea AMPMGRPASS=tupassword)  
/etc/asterisk/manager.conf (linea secret = tupassword)  
y en el portal freepbx en modo administrador, en la pestaña setup, en el link  
administrators, cambiad la contraseña para admin.
```

Tenemos que crear el directorio /var/lib/asterisk/sounds/custom con permisos 777 y del grupo asterisk. Se utilizara para el modulo Recordings que instalaremos luego.

```
operadora:/# mkdir /var/lib/asterisk/sounds/custom  
operadora:/# chown asterisk:asterisk -R /var/lib/asterisk/sounds/custom  
operadora:/# chmod 777 -R /var/lib/asterisk/sounds/custom
```

En el portal de Freepbx en la zona de administración y en tools encontraremos "Module Admin" descargar los que querais y actualizad todos. Yo me descargo todos los módulos y los actualizo todos, despues de chequear online los modulos existentes para descargar.

CONFIGURACIONES PARA EL HARDWARE

Bueno tenemos que realizar un par de ajustes en el hardware para dejar el sistema perfecto.

Iniciaremos el sistema con la opción de kernel noapic. Cuando iniciemos el sistema y grub arranque (antes de que lo haga) le diremos "Edit" nos saldrán las opciones de arranque del kernel, le daremos edit y pondremos al final "noapic" (sin comillas), para arrancar con estas opciones le daremos a la tecla b (boot).

Hacemos esto para ver que interrupciones utiliza cada tarjeta, lo mejor es dejar las tarjetas OpenVox en ranuras distintas, juega moviendo las tarjetas hasta que lo consigas.

Para ver las interrupciones utilizadas ponemos este comando:

operadora:~# cat /proc/interrupts

CPU0

0: 46563 XT-PIC timer

1: 8 XT-PIC i8042

2: 0 XT-PIC cascade

5: 175092 XT-PIC wctdm

6: 2 XT-PIC floppy

7: 1 XT-PIC parport0

8: 1 XT-PIC rtc

9: 1 XT-PIC acpi

10: 0 XT-PIC uhci_hcd:usb2

11: 1418 XT-PIC uhci_hcd:usb1, Intel ICH2, eth0

12: 554880 XT-PIC zaphfc, zaphfc

14: 3405 XT-PIC ide0

15: 63 XT-PIC ide1
NMI: 0
LOC: 46536
ERR: 0
MIS: 0

Si os fijáis en la respuesta del comando, es óptima, en la interrupción numero 5 tenemos a la tarjeta OpenVox solita, eso es lo que tenemos que conseguir moviendo la tarjeta por cada ranura PCI que tenemos en el ordenador.

Recordad que cuando hagais estos ajustes debéis volver a configurar los SPAN del archivo /etc/zaptel.conf

Debemos realizar un par de ajustes mas al hardware.

El disco duro debe trabajar en modo DMA, y dentro de modo DMA una configuración UDMA no inferior a 2 y no superior a 3.

Para realizar esto debemos ejecutar los siguientes parametros:

Comprobemos que utilizamos modo DMA:

operadora:~# hdparm /dev/hda

```
/dev/hda:  
multcount = 0 (off)  
IO_support = 0 (default 16-bit)  
unmaskirq = 0 (off)  
using_dma = 1 (on)  
keepsettings = 0 (off)  
readonly = 0 (off)  
readahead = 256 (on)  
geometry = 39546/16/63, sectors = 39862368, start = 0
```

Comprobemos el modo UDMA:

operadora:~# hdparm -i /dev/hda

```
/dev/hda:  
Model=SAMSUNG SV2044D, FwRev=MM101-48, SerialNo=0191J1FN500105  
Config={ HardSect NotMFM HdSw>15uSec Fixed DTR>10Mbs }  
RawCHS=16383/16/63, TrkSize=34902, SectSize=554, ECCbytes=4  
BuffType=DualPortCache, BuffSize=472kB, MaxMultSect=16, MultSect=off  
CurCHS=16383/16/63, CurSects=16514064, LBA=yes, LBAsects=39862368  
IORDY=yes, tPIO={min:120,w/IORDY:120}, tDMA={min:120,rec:120}  
PIO modes: pio0 pio1 pio2 pio3 pio4  
DMA modes: mdma0 mdma1 mdma2  
UDMA modes: udma0 udma1 udma2 udma3 *udma4 udma3 *udma4  
AdvancedPM=no WriteCache=enabled  
Drive conforms to: ATA/ATAPI-4 T13 1153D revision 17: ATA/ATAPI-1 ATA/ATAPI-2  
ATA/ATAPI-3
```

TA/ATAPI-4

** signifies the current active mode*

Cambiamos los modos de trabajo:

operadora:~# hdparm -d 1 -X udma2 -c 3 /dev/hda

/dev/hda:

setting 32-bit IO_support flag to 3

setting using_dma to 1 (on)

setting xfermode to 66 (UltraDMA mode2)

IO_support = 3 (32-bit w/sync)

using_dma = 1 (on)

Volvemos a comprobar:

operadora:~# hdparm -i /dev/hda

/dev/hda:

Model=SAMSUNG SV2044D, FwRev=MM101-48, SerialNo=0191J1FN500105

Config={ HardSect NotMFM HdSw>15uSec Fixed DTR>10Mbs }

RawCHS=16383/16/63, TrkSize=34902, SectSize=554, ECCbytes=4

BuffType=DualPortCache, BuffSize=472kB, MaxMultSect=16, MultSect=off

CurCHS=16383/16/63, CurSects=16514064, LBA=yes, LBAsects=39862368

IORDY=yes, tPIO={min:120,w/IORDY:120}, tDMA={min:120,rec:120}

PIO modes: pio0 pio1 pio2 pio3 pio4

DMA modes: mdma0 mdma1 mdma2

*UDMA modes: udma0 udma1 *udma2 udma3 udma4 udma3 udma4*

AdvancedPM=no WriteCache=enabled

Drive conforms to: ATA/ATAPI-4 T13 1153D revision 17: ATA/ATAPI-1 ATA/ATAPI-2

ATA/ATAPI-3

TA/ATAPI-4

** signifies the current active mode*

Perfecto, disco duro funcionando OK.

Para los que utilicen modulos FXO con su tarjeta OpenVox, pueden llegar a sufrir terribles ecos en sus lineas, la solución es el paquete fxotune, el cual hemos compilado a la vez que zaptel.

La forma de utilizarlo es la siguiente, debemos generar un archivo de configuración, para ello basta con teclear el siguiente comando y esperar un ratito:

operadora:~# fxotune -i 4

Generara el archivo /etc/fxotune.conf.

SCRIPTS INICIALES:

Para que todo inicie en su justo momento y en el orden que queramos necesitamos crear unos scripts.

Crearemos unos archivos en `/etc/init.d/`

Comencemos:

Si habéis reiniciado ya, es posible que os encontréis con que el modulo `zaphfc` no se carga automáticamente (extraña razón que no he logrado descubrir la que provoca este fallo) para solucionarlo lo que hago es crearme un script que lo carga antes de lanzar la carga de `zaptel`. Lo he llamado `carga_zaphfc.sh`

```
operadora:/# vi /etc/init.d/carga_zaphfc.sh
```

```
#!/bin/bash  
insmod /lib/modules/2.6.18-4-686/misc/zaphfc.ko
```

Démosle permisos de ejecución:

```
operadora:/# chmod +x /etc/init.d/carga_zaphfc.sh
```

Creemos ahora el archivo para que levante `fxotune`, deje el disco duro en modo `dma` y `udma 2` y levante `asterisk`:

```
operadora:/# vi /etc/init.d/inicia_asterisk.sh
```

```
#!/bin/bash  
echo "***** CAMBIAMOS MODO HD *****"  
hdparm -d 1 -X udma2 -c 3 /dev/hda
```

```
echo "***** MODO HD CAMBIADO *****"  
echo "***** CARGAMOS FXOTUNE *****"  
fxotune -s  
echo "***** FXOTUNE CARGADO *****"  
amportal start
```

Démosle permisos de ejecución:

```
operadora:/# chmod +x /etc/init.d/inicia_asterisk.sh
```

Tenemos que crear los enlaces simbólicos para que se ejecuten al iniciar para ello debemos irnos al directorio /etc/rcd.2

```
operadora:/# cd /etc/rc2.d/
```

Veamos que tenemos:

```
operadora:/# ls
```

En mi caso el zaptel tiene un orden de arranque de S15, y como necesitamos que cargue el modulo zaphfc antes de arrancar zaptel pues añadimos por ejemplo el link al script de carga de zaphfc en la posición S14:

```
operadora:/etc/rc2.d# ln -s /etc/init.d/carga_zaphfc.sh /etc/rc2.d/S14carga_zaphfc
```

El asterisk hay que levantarlo despues de que todos los servicios de los que depende lo han hecho, en mi caso la posición S92 es perfecta para levantar asterisk:

```
operadora:/etc/rc2.d# ln -s /etc/init.d/inicia_asterisk.sh /etc/rc2.d/S92inicia_asterisk
```

Ahora si que se nos ejecutaran en el orden adecuado cada vez que reiniciemos la maquina

CONFIGURACIÓN PARA PROBAR EL SISTEMA

Ya tenemos todo hecho. Solo nos falta configurar todo lo relacionado con Freepbx, podéis empezar a probar dando de alta una extensión y loguearos con un softphone (snom360 es muy bueno) contra la centralita. Añadid un trunk (salida de teléfonos) ya sea zap, sip, iax (Para las rdsi o fxo los trunks son del tipo zap, para peoplecalls tipo sip y si queréis conectar contra otra centralita lo mejor es iax). Bueno la configuración del portal es otro capítulo, ya que cada uno tendrá un tipo de configuración, es tan difícil hacer una estándar para todo el mundo que lo mejor será ir resolviendo dudas en el foro sobre tipos de configuración, os invito a hacer todas las preguntas que necesitéis, os ayudare en lo que pueda.

Aun así os dejo una configuración mínima para una extensión, un trunk para un fxo, un trunk para una rdsi o billion, para que probeis:

Para entrar a hacer todas las configuraciones entrad en http://ip_operadora/freepbx (si seguisteis al pie de la letra en howto).

Para añadir una extensión:

Id Pinchando donde os diga, Freepbx administration, setup, (meter usuario y contraseña por defecto admin y admin), extensions, add extension (Generic sip device) y rellenamos lo siguiente: User extension (por ejemplo la 100), Display Name (Tu nombre), Secret (Contraseña de la extensión, yo pongo la misma que la extensión es decir 100), submit changes, lo demás por defecto. Si queréis miraros algo, para más adelante.

Para añadir un trunk para el FXO:

Freepbx administration, setup, (meter usuario y contraseña por defecto admin y admin), trunks, Add ZAP trunk, lo único que tenemos que rellenar es el Zap Identifier que se corresponde al grupo que tenemos creado en el archivo /etc/asterisk/zapata.conf, siempre que lo añadamos debe ser con la nomenclatura g0 o g1.... (dependiendo del grupo), , submit changes, en mi caso el FXO está en el grupo 0 con lo cual añado en Zap Identifier g0.

Para añadir un trunk para RDSI o Billion:

Se comporta exactamente igual que el FXO pero sera un grupo diferente, Freepbx administration, setup, (meter usuario y contraseña por defecto admin y admin), trunks, Add ZAP trunk, Zap Identifier (g1), submit changes.

Tenemos que configurar rutas salientes y entrantes para poder probar:

Rutas Entrantes:

Freepbx administration, setup, (meter usuario y contraseña por defecto admin y admin), Inbound Routes, Add Incoming Route y solo rellenamos Set Destination y lo ponemos a Core (Tu nombre, tu extension), submit changes. Nos abra añadido una ruta entrante llamada "any DID / any CID" (Cualquier llamada).

Rutas Salientes (FXO):

Freepbx administration, setup, (meter usuario y contraseña por defecto admin y admin), Outbound Routes, Add Route, Route Name (RTB), Dial Patterns (001|.), Trunk Sequence (ZAP/g0), submit changes. Le acabamos de decir que las llamadas que empiecen por 001 las saque por el ZAP0, en mi caso la linea RTB.

Rutas Salientes (Billion):

Freepbx administration, setup, (meter usuario y contraseña por defecto admin y admin), Outbound Routes, Add Route, Route Name (RDSI), Dial Patterns (002|.), Trunk Sequence (ZAP/g1), submit changes. Le acabamos de decir que las llamadas que empiecen por 002 las saque por el ZAP1, en mi caso la linea RDSI.

Probad con el Softphone las lineas llamando hacia afuera con 001teléfono para salir por la RTB y 002teléfono para salir por la rdsi, probad también las entrantes llaméis a cual llames de vuestro números de teléfono os debería sonar el softphone.

AGRADECIMIENTOS:

Pues como siempre a **Alfonso** de Microalcarria S.L., sin el esto del linux para mi seria mas que complicado, gracias ;-)

A **Juan Carlos Valero** de Capatres Soluciones Soluciones Tecnológicas S.L., me ayudo mucho con el tema del hardware, gracias ;-)

A **vosotros** que, como siempre, con vuestra ayuda mejoraremos este documento.