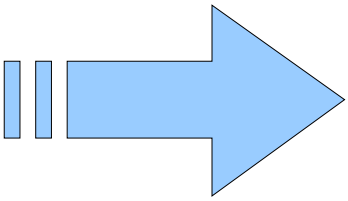
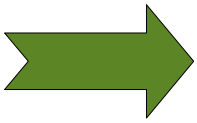
PFFPEE

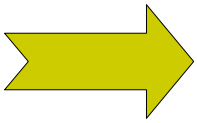
X0809003 Despregue dunha rede corporativa de VoIP



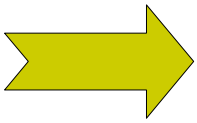
Acceso a WEB



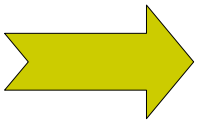
Instalación e configuración de ASTERISK



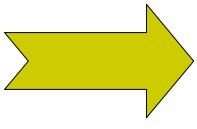
Practica 1



Practica 2



Practica 3



Esquema de rede

Instalación e Configuración dun PBX Asterisk en Debian Lenny

De ASIRodeira

Índice

Introducción

O obxectivo principal é a instalación dun PBX funcional mediante Asterisk nun equipo informático con Debian Lenny, actualmente en estado "testing". O Equipo dispon do seguinte hardware:

- **CPU**: Pentium III (Coppermine) 1Ghz
- **RAM**: 512Mb SDRAM
- **Disco Duro**: IDE 40Gb
- **Tarxeta Rede**: 100M Realtek RTL8139
- **Tarxetas de Telefonía:**
  - Cologne Chip Designs GmbH ISDN (HFC-PCI) rev 02 (Beronet Billion BN250 1 porto BRI)
  - Tiger Jet Network Inc. Tiger 3XX (OpenVox A400P)

O kernel actual é o 2.6.26-1-686, e ten incorporados os drivers para RDSI de isdn4linux (http://www.isdn4linux.de) , que cargan o módulo **hisax** para as dúas tarxetas. En futuras versións do kernel, o sistema a empregar será o máis moderno mISDN (http://www.misdn.org/index.php/Main_Page) que incorpora o driver **misdn** útil para a tarxeta *Cologne Chip Designs*. Para a tarxeta *OpenVox A400P* se emprega habitualmente o driver **zaptel**.

O módulo **hisax** está soportado por asterisk para tarxetas BRI, aínda que parece que é necesario un parche (http://www.voip-info.org/wiki/view/Asterisk+i4l+channels)

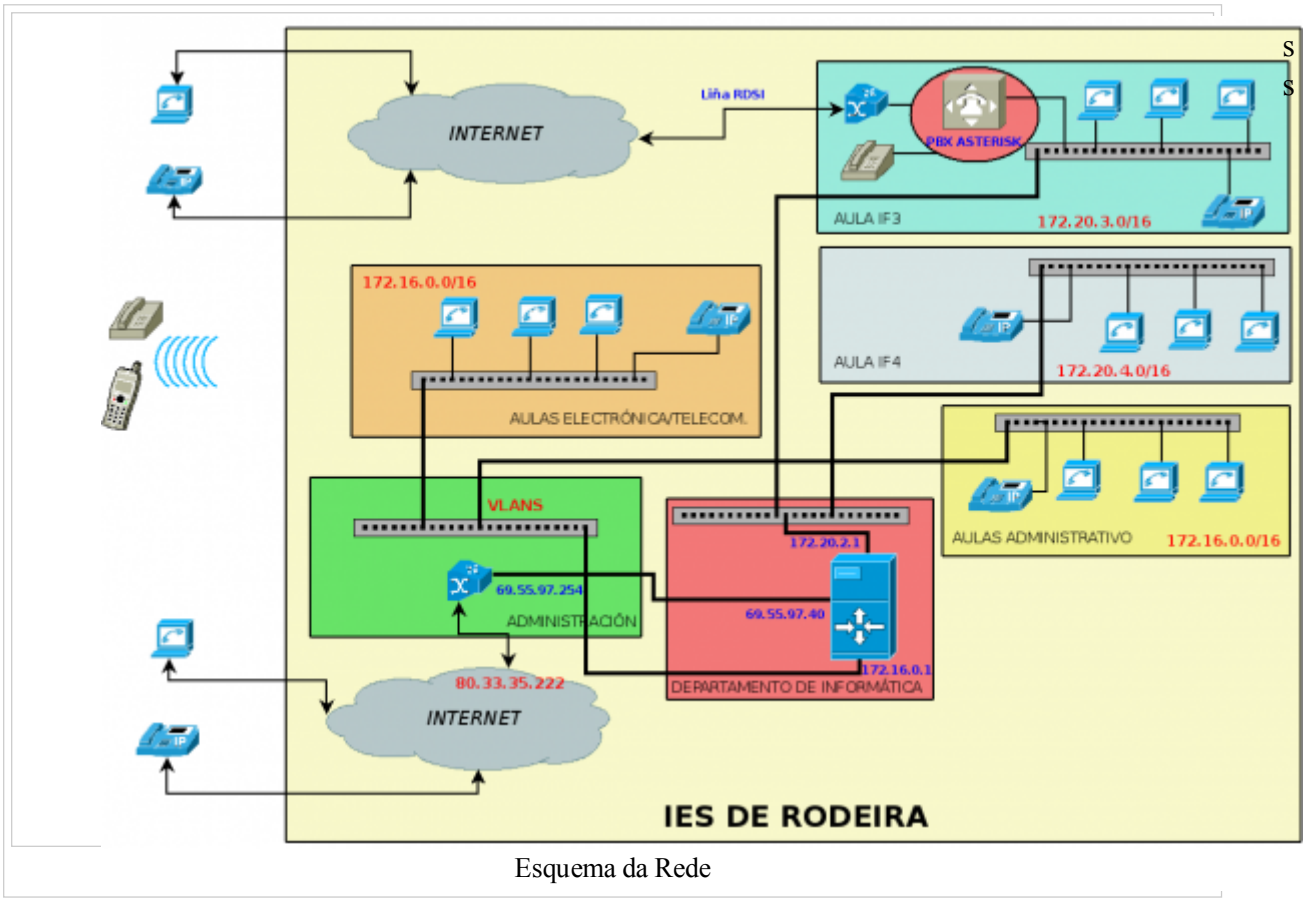
A tarxeta RDSI Cologne Chip Design e soportada normalmente polo driver bristuff (http://www.junghanns.net/en/download.html) de *junghanns*, que parece ser o de máis calidade, pero que precisa de parchear o Asterisk (existen scripts automatizados). Neste caso (Beronet Billion BN250) parece que o driver ideal e o **misdn** coa ventaxa de pasar a formar parte do próximo kernel Linux. A partir do kernel 2.6.27 ven incluido *mISDN 2.0*, non dispoñible en Debian neste momento, o instalaremos para probalo con Asterisk. A versión 1.x de misdn pode instalarse perfectamente no kernel de Lenny, mentras que a versión 2.x é necesario utilizala mediante **chan_lcr**, proporcionada pola aplicación *Linux Call Router*.

Mais drivers que poden soportar a tarxeta RDSI son:

- zaphfc (http://www.voip-info.org/wiki/view/Asterisk+zaphfc)
- libpri (http://www.voip-info.org/wiki/view/Asterisk+libpri) (a partir de Asterisk 1.6)

A versión de Asterisk que imos empregar é a última Asterisk 1.6, e o faremos dende 0, xa que non está dispoñible ningún tipo de paquete Debian para esta versión. A versión 1.6 presenta notables cambios na súa arquitectura que iremos abordando a medida que avancemos na instalación. Un dos máis importantes é o abandono dos drivers zaptel en favor de **DAHDI** (*Digium Asterisk Hardware Device Interface*) que consta de dúas partes, os módulos para o kernel (dahdi-linux) e as utilidades de configuración (dahdi-tools), reemplazando */etc/asterisk/zapata.conf* por */etc/asterisk/chan_dahdi.conf* e */etc/zaptel.conf* por */etc/dahdi/system.conf*. Ainda que este cambio parece moi drástico, a rama 1.4 tamén vai abandonar o driver zaptel.

Esquema de Rede



Esquema da Rede

Informática, conectada a Rede Telefónica mediante unha liña RDSI por medio do interface BRI, e tendo doble acceso a Internet, un directo a través da ADSL sobre RDSI da propia IF3 e outro a través da liña ADSL dos ciclos utilando Gandalf como pasarela e router. O acceso principal a Internet e o acceso dende fora do centro o PBX Asterisk se fará a través da liña ADSL dos ciclos e de Gandalf. En caso de fallo da ADSL dos ciclos, o PBX empregaría a ADSL de IF3.

Unha posibilidade a estudiar sería tamén o balanceo das chamadas VoIP sobre as dúas liñas ADSL, de xeito que se utilicen simultáneamente si se atopan dispoñibles e además contemplar a posibilidade de un servidor Asterisk de seguridade que tomaría o relevo en caso de caída do principal. Electrónica dispón tamén doutra liña ADSL que sería perfectamente posible utilizar do mesmo xeito.

Será posible conectar e utilizar o servidor PBX Asterisk:

- Dende IF3, IF4 e o Departamento de Informática
- Dende as aulas de Administrativo, Electrónica e Telecomunicacións, aula de Informática da ESO e
- Aula Mentor.
- Dende Internet, tanto a través da liña ADSL de IF3 como a liña ADSL dos ciclos. Tamén podería
- utilizar a ADSL de Electrónica

O acceso principal será a través do ADSL dos Ciclos, para o que Gandalf terá que facer o rutado necesario.

Os casos de uso poderían ser os seguintes:

Chamada dende a Rede Telefónica o número da liña RDSI de IF3

- Dependendo do marcado podemos encamiñala hacia: Telefono analóxico conectado o PBX-Asterisk, Teléfono IP conectado en calquera punto da rede, Softphone de calquera punto da rede, Teléfono IP ou softphone conectado a Internet fora do centro e rexistrado no noso PBX-Asterisk, chamada mediante un proveedor VoIP externo a outro número conectado a rede telefónica dentro ou fora de España ou a un softphone ou teléfono IP rexistrado con ese proveedor.

Chamada dende Internet (externa mediante VoIP) a través do PBX-Asterisk.

- Igual que o punto Anterior

Chamada dende un teléfono interior (Softphone, teléfono IP ou teléfono de liña telefónica)

- Igual que o primeiro punto

Como en realidade dispoñemos de dúas liñas telefónicas (a da IF3 e a de Electrónica) a configuración poderá ser máis adiante moito máis completa, pero este traballo deberá estudiarse con posteridade a instalación, xa que a liña de Electrónica será xestionada por outro PBX con Trixbox.

Servicios Planeados

Contestador Automático
: Permitirá a cada extensión utilizar un contestador automático.

Recepción de Correos
: Permitirá que se poda escoitar o correo electrónico de cada usuario vía telefónica.

Chamadas dende telefonía fixa
: As chamadas recibidas a través de liña telefonica conmutada se poderán redirixir a través de varios proveedores de VoIP, a liñas de teléfono VoIP directamente rexistradas co servidor Asterisk, ou co servidor Trixbox, a outras liñas telefónicas fixas ou os teléfonos conectados o servidor Asterisk ou Trixbox.

Chamadas a través de VoIP
: Os encaminamentos serán similares o punto anterior.

Avisos
: Poderanse programar avisos telefónicos que se produciran no momento indicado. Por exemplo, sms para aviso de claustros ou reunións de departamentos.

Chamadas entre proveedores VoIP
: As chamadas entre os clientes dun mismo proveedor IP son normalmente gratuitas, o Asterisk será capaz de encamiñar as chamadas dun proveedor IP a outro co que as chamadas entre varios proveedores IP serán posibles e gratuitas. Para instalar este servicio se terá que abrir contas en varios proveedores IP xa sexan gratuitas ou de pago.

Servicios Suxeridos

Salas de reunións
: Para realizar charlas entre varias persoas, restrinxido a un grupos de extensións por departamentos. ¿Reunión de Departamento telefónica?, ¿Clases vía telefónica? ¿Clases con voz sintetizada programadas vía telefónica?.

Encamiñamento de teléfonos móbiles
: Dispoñendo de tarxetas GSM sería posible facer chamadas fixo<->móbil<->VoIP o precio da

chamada máis barata posible entre dous números do proveedor de telefonía móbil elexido. Incluso e posible que existan ofertas de chamadas gratuitas móbil->móbil que poderías aproveitarse para facer chamadas gratuitas móbil<->fixo e móbil<->VoIP.

Consultas de calificacións vía telefónica
: De notas ou outro tipo de información persoal. A resposta pode ser vía voz ou vía sms.

Consultas da Wikipedia ou apuntes
: Permitirá realizar consultas que se responderán via voz, por exemplo termos dun diccionario ou da wikipedia. A dificultade maior e atopar un sistema de recoñecimento de voz libre e de calidade.

Integración de Asterisk nunha plataforma de e-learning
: Permitirá utilizar os servicios de asterisk integrados nunha plataforma e-learning como Dokeos ou Moodle.

Solicitude de documentación vía telefónica
: Será posible solicitar documentación vía telefónica que será enviada a unha conta de correo electrónico (por exemplo apuntes ou pdf's coas notas).

Interface Web de Configuración
: Creación dunha interface que permita os usuarios configurar os distintos servicios os que teñan acceso.

Instalación de Debian

A instalación será a habitual dunha Debian Lenny, para elo baixamos o CD de instalación por rede da páxina oficial de Debian (http://www.debian.org/devel/debian-installer/) e instalaremos so o sistema base.

O primeiro que temos que facer unha vez actualizado o sistema base é actualizar o sistema completo e instalar aplicacións adicionais, para facer esto iniciamos a sesión como **root** e modificamos o ficheiro */etc/apt/sources.list* para incluir:

```
deb http://www.debian-multimedia.org lenny main
```

Posteriormente actualizamos o sistema:

```
apt-get update
apt-get dist-upgrade
```

O sistema base lle engadiremos os seguintes paquetes:

ssh
: Para acceder e configurar remotamente o equipo Asterisk.

iptraf
: Ferramenta de supervisión de rede.

nmap
: Aplicación de revisión de portos.

rcconf
: Ferramenta de configuración de servicios.

vim
: Editor de texto de consola

wireless-tools
: Ferramentas de configuración de tarxetas wireless.

gnome
: Entorno gráfico de usuario, por si queremos utilizar ferramentas gráficas (consultar a web, descargar co navegador ...). En principio non estará activo.

apache
: Para as ferramentas de configuración web de asterisk.

bridge-utils
: Ferramentas de configuración de rede, particularmente para facer pontes de rede.

ebtables
: Ferramenta de firewall e rutado para pontes de rede.

Executamos:

```
apt-get install ssh iptraf rcconf vim gnome apache2 wireless-tools gnome bridge-utils ebtables
```

Configuración Inicial do Sistema Operativo

En primeiro lugar, desactivaremos o inicio automático do xestor gráfico de entrada de usuario (**gdm**) mediante a ferramente **rcconf**, e posteriormente facemos uns pequenos axustes ó editor **vim** editando */etc/vim/vimrc*:

- Descomentamos *syntax on* (Os comentarios comezan con ")
- Engadimos *set noai*

Agora poñemos un nome o host, editando /etc/hostname. Neste caso ASTERISK-RODEIRA

Instalación de Asterisk

Asterisk 1.4.21 está dispoñible para Debian lenny, polo que a súa instalación é tan sencilla como facer:

```
apt-get install asterisk
```

Sen embargo imos instalar Asterisk 1.6 e configuralo para o noso hardware.

Descarga de Software

apt-get install ntp

```
--- NO ---
apt-get install ntpdate
crontab -e
1 5 * * * /usr/sbin/ntpdate-debian
----------
```

En primeiro lugar descargaremos a última versión de Asterisk 1.6 e paquetes relacionados:

```
wget http://downloads.digium.com/pub/asterisk/releases/asterisk-1.6.0.5.tar.gz
wget http://downloads.digium.com/pub/asterisk/asterisk-addons-1.6.0.1.tar.gz
wget http://downloads.digium.com/pub/telephony/dahdi-linux/dahdi-linux-2.1.0.4.tar.gz
wget http://downloads.digium.com/pub/telephony/dahdi-tools/dahdi-tools-2.1.0.2.tar.gz
wget http://downloads.digium.com/pub/libpri/libpri-1.4.9.tar.gz
wget http://downloads.digium.com/pub/telephony/sounds/asterisk-core-sounds-es-alaw-current.tar.gz
wget http://downloads.digium.com/pub/telephony/sounds/asterisk-core-sounds-es-g722-current.tar.gz
wget http://downloads.digium.com/pub/telephony/sounds/asterisk-core-sounds-es-g729-current.tar.gz
wget http://downloads.digium.com/pub/telephony/sounds/asterisk-core-sounds-es-gsm-current.tar.gz
wget http://downloads.digium.com/pub/telephony/sounds/asterisk-core-sounds-es-sln16-current.tar.gz
wget http://downloads.digium.com/pub/telephony/sounds/asterisk-core-sounds-es-ulaw-current.tar.gz
wget http://downloads.digium.com/pub/telephony/sounds/asterisk-core-sounds-es-wav-current.tar.gz
```

De aquí se poden descargar os drivers *mISDN* versión 1 e 2:

```
wget http://www.misdn.org/downloads/releases/mISDN-1_1_9.1.tar.gz
wget http://www.misdn.org/downloads/releases/mISDNuser-1_1_9.1.tar.gz
```

```
wget http://www.colognechip.com/download/mISDN/socket/mISDN.tar.gz
wget http://www.colognechip.com/download/mISDN/socket/mISDNuser.tar.gz
```

A versións do kernel a partir do 2.6.27 xa inclúen os drivers mISDN v2, de modo que é a que utilizaremos. Este kernel non está dispoñible en Debian, de modo que baixamos o kernel *vanilla* máis actual (*vanilla* quere decir que é o kernel orixinal, sin parches de ningunha distribución) que xa os inclúe. O descargamos no directorio **/usr/src**.

```
wget http://www.kernel.org/pub/linux/kernel/v2.6/linux-2.6.28.5.tar.gz
```

Instalamos o software necesario para compilar o kernel, os módulos e Asterisk:

```
apt-get install kernel-package libncurses5-dev zlib1g-dev linux-headers-2.6-686 automake
```

Descomprimimos e configuramos o kernel. Pode facerse dende un terminal en gnome ou dende a consola (texto). No primeiro caso faremos a configuración do kernel con *make gconfig*, no segundo caso con *make menuconfig*.

```
cd /usr/src
tar -xvzf linux-2.6.28.5.tar.gz
cd linux-2.6.28.5
make menuconfig
```

Poñemos as seguintes opcións:

```
Enable The Block Layer -> I/O Schedulers -> Default I/O Scheduler: Deadline
Processor Type And Features -> Preemption Model - Low Latency Desktop
Processor Type And Features -> Timer Frecuency: 1000Hz
Networking Support -> Phonet Protocols Family: M
Device Drivers -> ISDN Support -> Modular ISDN Driver: *
Device Drivers -> ISDN Support -> CAPI 2.0 subsystem -> (Todo seleccionado)
Device Drivers -> Telephony Support: (Todo M)
```

Creamos o paquete co núcleo e o instalamos, reiniciando co novo núcleo:

```
make-kpkg --initrd --rootcmd fakeroot kernel-image kernel-headers
cd ..
dpkg -i linux-image-2.6.28.3_2.6.28.3-10.00.Custom_i386.deb
ln -s linux-2.6.28.3 linux
shutdown -r now
```

Instalación dos módulos hardware de telefonía

Imos utilizar *mISDNv2* para o módulo RDSI e *dahdi 2.1* para o módulo RTB.

mISDNv2

probamos o módulo mISDN

```
modprobe hfcpci
modprobe mISDN_dsp
```

A saída en */var/log/kern.log* debería ser algo similar a esto:

```
hfcpci 0000:00:09.0: PCI INT A -> Link[LNKB] -> GSI 11 (level, low) -> IRQ 11
mISDN_hfcpci: found adapter CCD/Billion/Asuscom 2BD0 at 0000:00:09.0
mISDN: HFC-PCI driver 2.0
HFC-PCI: defined at mem 0xe1b2c000 fifo 0xdf558000(0x1f558000) IRQ 11 HZ 1000
HFC 1 cards installed
DSP modul 2.0
mISDN_dsp: DSP clocks every 64 samples. This equals 8 jiffies.
mISDN_dsp_element_register: hwec registered
```

A continuación instalaremos o paquete mISDN_utils versión 2

```
mkidr mISDNuser
cd mISDNuser
tar -xvzf mISDNuser.tar.gz
make
make install
```

Desgraciadamente Asterisk non soporta a versión 2 de mISDN incluída no kernel mediante chan_misdn (so mISDN v1), nin parece que teña plans de facelo. A solución é ou ben utilizar mISDN v1, sobreescribindo as versións que traerán os kernel, ou utilizar *chan_lcr*...

DAHDI

O driver *dahdi* e o sustituto de *zaptel* nas últimas versións de asterisk. O ficheiro equivalente a */etc/zaptel.conf* vai ser */etc/dahdi/system.conf* e o sustituto de */etc/asterisk/zapata.conf* e */etc/asterisk/chan_dahdi.conf*.

O compilamos para soportar a tarxeta cos módulos FXO e FXS.

```
tar -xvzf dahdi-linux-2.1.0.4.tar.gz
cd dahdi-linux-2.1.0.4
make
make install
```

Agora necesitaremos compilar as utilidades de manexo do driver dahdi.

```
 tar -xvzf dahdi-tools-2.1.0.2.tar.gz
 cd dahdi-tools-2.1.0.2
 ./configure
 make menuconfig
 make
 make install
```

No momento de facer o *./configure*, o sistema e posible que visualice a seguinte mensaxe:

```
checking for GENERIC_HDLC_VERSION in linux/hdlc.h... no
configure: GENERIC_HDLC_VERSION version 4 not found, disabling sethdlc.
```

Esto é debido a un bug no configure que podemos solucionar do seguinte xeito:

- Editamos *configure.ac* e buscamos a liña que pon *AST_C_DEFINE_CHECK([HDLC], [GENERIC_HDLC_VERSION], [linux/hdlc.h], [4])* e a cambiamos por *AST_C_DEFINE_CHECK([HDLC], [GENERIC_HDLC_VERSION], [linux/hdlc/ioctl.h], [4])*

- Executamos *autoconf*
- Editamos *configure* e buscamos a liña que pon *#include <linux/hdlc/ioctl.h>* e lle poñemos diante *##include <net/if.h>*
- Executamos: *./configure*, *make menuconfig* e *make install*

Insertamos o novo módulo con *modprobe dahdi* e a saída no */var/log/kern.log* debera ser similar a esta:

```
dahdi: Telephony Interface Registered on major 196
dahdi: Version: 2.1.0.4
Freshmaker version: 71
Freshmaker passed register test
Module 0: Installed -- AUTO FXS/DPO
Module 1: Installed -- AUTO FXO (FCC mode)
Module 2: Not installed
Module 3: Not installed
Found a Wildcard TDM: Wildcard TDM400P REV E/F (2 modules)
dahdi: Registered tone zone 6 (Spain)
```

Debemos editar /etc/dahdi/system.conf e cambiar *loadzone* e *defaultzone* a es ...

LCR

LCR (Linux Call Router) e un pequeno PBX que imos utilizar únicamente para que Asterisk poda acceder a tarxeta RDSI. En primeiro lugar o descargamos e o descomprimimos.

```
wget http://www.linux-call-router.de/download/lcr-1.3/lcr_20090107.tar.gz
tar -xvzf lcr_20090107.tar.gz
cd lcr
```

E necesario parchear o arquivo /usr/include/mISDNuser/mbuffer.h, xa que os includes están mal postos, deben ser:

```
#include <mISDNuser/mISDNif.h>
#include <mISDNuser/mlayer3.h>
```

Compilamos e instalamos LCR

```
./configure --with-asterisk --with-ssl
make
make install
ln -s /usr/local/lcr /etc/lcr
```

Comprobamos as chamadas:

```
 lcr fork
 lcradmin status
```

Facemos unha chamada o número da RDSI vía telefono PSTN, comprobando a recepción no terminal.

Editamos /etc/lcr/routing.conf e rediriximos TODO a Asterisk

```
[main]
       : remote application=asterisk
```

Nota para Asterisk 1.6.1.0

A versión de LCR que estou a probar non compila correctamente contra Asterisk 1.6.1.0, dando os seguintes erros:

```
make  all-am
make[1]: Entrando no directorio `/root/lcr'
gcc  -I/usr/include/mISDNuser -DCONFIG_DATA="\"/usr/local/lcr\"" -DSHARE_DATA="\"/usr/local/lcr\"" -DLO
-DEXTENSION_DATA="\"/usr/local/lcr/extensions\""  -D_GNU_SOURCE -fPIC -c chan_lcr.c -o chan_lcr.po
chan_lcr.c: In function ‘lcr_write’:
chan_lcr.c:1998: error: incompatible type for argument 2 of ‘bchannel_transmit’
chan_lcr.c: In function ‘lcr_read’:
chan_lcr.c:2038: error: incompatible types in assignment
chan_lcr.c: In function ‘load_module’:
chan_lcr.c:2535: warning: passing argument 2 of ‘ast_register_application2’ from incompatible pointer t
make[1]: *** [chan_lcr.po] Erro 1
make[1]: Saíndo do directorio `/root/lcr'
make: *** [all] Erro 2
```

O problema e o cambio no campo *data* da estructura *ast_frame* en */usr/include/asterisk/frame.h*, que agora é:

```
union { void *ptr; uint32_t uint32; char pad[8]; } data;
```

A solución consiste en poñer na liña 1998 de chan_lcr.c:

```
bchannel_transmit(call->bchannel, f->data.ptr, f->samples);
```

E na liña 2038:

```
call->read_fr.data.ptr = call->read_buff;
```

Asterisk

Descargamos *libresample* e *nbs* do árbol de proxectos de asterisk, e os compilamos e instalamos:

```
svn co http://svn.digium.com/svn/thirdparty/libresample/trunk
svn co http://svn.digium.com/svn/nbs/trunk/
cd nbs
make&&make install
cd ../libresample
make&&make install
```

Descomprimimos o *tar.gz* do asterisk:

```
tar -xvzf asterisk-1.6.0.5.tar.gz
cd asterisk-1.6.0.5
```

Descargamos e instalamos o codec *ilbc* (**low bit rate codec**) en codecs/ilbc dentro da carpeta de asterisk

```
wget http://www.ietf.org/rfc/rfc3951.txt
wget http://www.ilbcfreeware.org/documentation/extract-cfile.awk
awk -f extract-cfile.awk rfc3951.txt > codecs/ilbc
```

Tamén se pode facer co script situado en *contrib/scripts/get_ilbc_source.sh* dentro do directorio de instalación de Asterisk

Instalamos as dependencias de Asterisk e FreePBX:

```
apt-get install libasound2-dev festival-dev speech-tools festvox-ellpc11k ffmpeg libavcodec-dev libspan
apt-get install libjack-dev libsamplerate0-dev libosptk3-dev libssl-dev libtonezone-dev unixodbc-dev li
apt-get install libltdl3-dev libsqlite0-dev freetds-dev portaudio19-dev libiksemel-dev libopenh323-dev
apt-get install libogg-dev libvorbis-dev libcurl4-gnutls-dev libcurl4-openssl-dev libgtk2.0-dev liblua5
apt-get install libfreeradius-dev libglade2.0-dev libgmime-2.0-2-dev libfreeradius-dev libsqlite3-dev
apt-get install libmysqlclient15-dev libbluetooth-dev
apt-get install cvs libssl-dev
apt-get install doxygen doxygen-gui doxygen-doc
wget http://forja.guadalinex.org/repositorio/frs/download.php/153/festvox-palpc16k_1.0-1_all.deb
dpkg -i festvox-palpc16k_1.0-1_all.deb
```

Configuramos e compilamos Asterisk:

```
./configure
make menuconfig
make
make install
make samples
make progdocs
```

Non configurar "module embedding" (enlazados estáticamente), xa que a compilación falla nesta versión de Asterisk, de todos os xeitos non se obtén ningunha ventaxa práctica sobre os módulos dinámicos.

Actualización de Asterisk 1.6.0.5 a Asterisk 1.6.0.6

Para compilar Asterisk 1.6.0.6 se sigue o mesmo procedimento descrito anteriormente, pero presentanse dous problemas:

```
In file included from app_dahdiras.c:50:
/usr/include/dahdi/user.h:736: error: expected specifier-qualifier-list before __s32
/usr/include/dahdi/user.h:939: error: expected specifier-qualifier-list before __s32
```

Que se soluciona editando app_dahdiras.c e engadíndolle

```
#include <asm/types.h>
```

Outro erro posible e por non instalar *ilbc* e seleccionar a construcción do codec:

```
codec_ilbc.c:40:30: error: ilbc/iLBC_encode.h: Non hai tal ficheiro ou directorio
codec_ilbc.c:41:30: error: ilbc/iLBC_decode.h: Non hai tal ficheiro ou directorio
```

E se soluciona colocando o codec en *codecs/ilbc* do xeito indicado no apartado anterior.

Unha vez compilado Asterisk (ou antes si se quere) se fai unha copia de seguridade da configuración anterior (*/etc/asterisk*) e se desinstala dende o directorio da versión anterior (*make uninstall*). Logo podemos instalar a nova versión do modo habitual e recuperar a configuración.

E importante tamén recordar copiar *chan_lcr.so* e *chan_lcr.po* a */usr/lib/asterisk/modules*.

Envío e recepción de Fax

NVFaxdetect

AsterFax

apt-get install iaxmodem hylafax-server hylafax-client

Conexión con Google Talk (Gtalk)

Google Talk utiliza o protocolo Jabber, e Jingle para transportar voz e vídeo. A partir da versión 1.4 Asterisk da soporte a estes dous protocolos, polo que é posible integrar telefonía de este servicio. A configuración realízase mediante os seguintes ficheiros:

- gtalk.conf

```
[general]
context=GtalkLine               ;;Context to dump call into
bindaddr=172.20.100.100         ;;Address to bind to
allowguest=yes                  ;;Allow calls from people not in list of peers
[guest]                         ;;special account for options on guest account
disallow=all
allow=ulaw
allow=alaw
context=GtalkLine
connection=gtalk_asterisk
```

- jabber.conf

```
[general]
debug=no                                ;;Turn on debugging by default.
autoprune=no                            ;;Auto remove users from buddy list.
autoregister=no                         ;;Auto register users from buddy list.
[gtalk_asterisk]                        ;;label
type=client                             ;;Client or Component connection
serverhost=talk.google.com              ;;Route to server for example, talk.google.com
username=rodeiravoip@gmail.com/Talk     ;;Username with optional resource.
secret=********                         ;;Password
;priority=1                             ;;Resource priority
port=5222                               ;;Port to use defaults to 5222
usetls=yes                              ;;Use tls or not
usesasl=yes                             ;;Use sasl or not
statusmessage="Asterisk IESRODEIRA VoIP Server" ;;Have custom status message for Asterisk.
timeout=100                             ;;Timeout on the message stack.
```

- extensions.conf

```
....
[GtalkLine]
exten => s,1,NoOp(Call from Gtalk)
....
```

Para chamar a un contacto de Gtalk dende, por exemplo, a extensión 300 empregaríamos:

```
....
exten => 300,1,Dial(gtalk/gtalk_asterisk/contacto_a_chamar@gmail.com)
....
```

Festival

Configuramos festival para Asterisk:

- En /usr/share/doc/examples/festival.scm, engadimos o final:

```
(set! voice_default 'voice_JuntaDeAndalucia_es_pa_diphone)
```

```
(define (tts_textasterisk string mode)
  "(tts_textasterisk STRING MODE)
  Apply tts to STRING. This function is specifically designed for
  use in server mode so a single function call may synthesize the string.
  This function name may be added to the server safe functions."
  (let ((wholeutt (utt.synth (eval (list 'Utterance 'Text string)))))
  (utt.wave.resample wholeutt 8000)
  (utt.wave.rescale wholeutt 5)
  (utt.send.wave.client wholeutt)))
```

- Copiamos /usr/share/doc/examples/festival.scm a /etc
- Copiamos /usr/share/doc/examples/festival.init /etc/init.d/festival
- chmod 744 /etc/init.d/festival
- Creamos /etc/default/festival con RUN_FESTIVAL=yes
- Habilitamos o inicio do servicio con *rcconf*

FreePBX [Sen Rematar]

wget http://puzzle.dl.sourceforge.net/sourceforge/amportal/freepbx-2.5.1.tar.gz

```
adduser asterisk --home /dev/null --shell /bin/false --gecos "asterisk PBX" --disable-password
adduser www-data asterisk
```

Poñemos

```
export APACHE_RUN_USER=asterisk
export APACHE_RUN_GROUP=asterisk
```

en /etc/apache2 envvars

Flash Operator Panel [Sen Rematar]

Asterisk-GUI [Sen Rematar]

Consideracións de Seguridade

Seguridade da Rede

- Asterisk abre alguns portos para chamadas VoIP entrantes. Debemos verificar os portos e as direccións IP nos ficheiros de configuración do canal.
- Si se habilita o interface *manager* en *manager.conf* debemos asegurarnos que so se pode acceder de modo seguro, a través dunha VPN ou dun túnel SSH.
- Para todas as conexións TCP/IP de Asterisk se poden deseñar listas ACL (Access Control Lists) que permitan ou deneguen o acceso os servicios Asterisk. Debemos verificar as opcións de configuración "permit" e "deny" en *manager.conf* e nas configuracións dos canais VoIP, como *iax.conf* e *sip.conf*.
- O protocolo IAX2 soporta autenticación RSA e encriptación AES para voz e sinalización. SIP non, para as versións actuais de Asterisk.

Seguridade do dialplan

Debemos utilizar contextos de extensión para aillar a saida ou os servicios de pago de calqueira conexión entrante

Deberíamos considerar que si calqueira canal, liña de entrada, etc. pode acceder a un contexto de extensión, terá a capacidade de acceder a calquera extensión dentro de ese contexto. Polo tanto, NON se debería permitir o acceso a servicios saíntes ou de pago en contextos que son accesibles (sobre todo si non é necesaria unha password) dende os canais de entrada, xa sexan IAX, FX ou outros *trunks*, ou incluso dende estacións da propia rede nas que non teñamos plena confianza. En particular *nunca debemos poñer servicios de saída de pago no contexto* ***default****. E posible incluir o contexto por defecto en outros contextos mediante* ***include*** *=>* ***default*** *na sección apropiada.*

Non debemos olvidarnos de colocar o contexto de demostración (*demo*) fora do contexto por defecto (*default*).

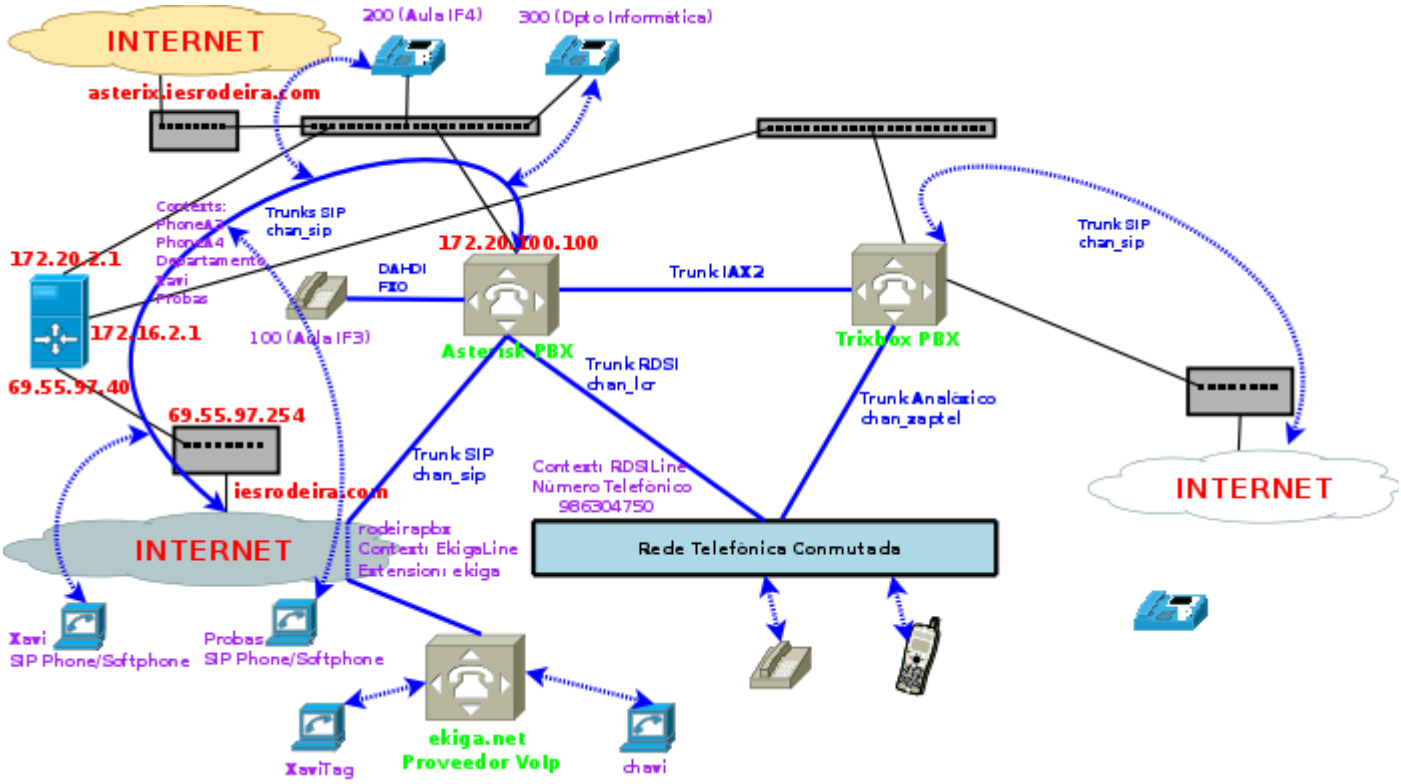
Seguridade dos Logs

Os ficheiros de log de Asterisk poden ter información confidencial, como passwords, polo que os debemos protexer axeitadamente.

Configuración da Rede Local

Mapa Telefónico

Nesta sección se expoñerá cal é o esquema da rede onde se vai a implementar o servicio de VoIP, os problemas surxidos e as solucións aportadas. O mapa telefónico será o seguinte:



NAT

O protocolo SIP é especialmente problemático co NAT, e a configuración na que estamos a traballar é complicada, xa que temos un doble NAT: No router que da acceso a Internet (69.55.97.254) e no router Linux (gandalf) que da acceso as redes 172.20.0.0 e 172.16.0.0. O router que da acceso a a Internet (69.55.97.254) pola rede 69.55.97.0 ten todos os portos de entrada mapeados a gandalf (69.55.97.40).

gandalf(69.55.97.40) ten habilitado o forwarding e mapeados os portos RTP, SIP e IAX o servidor Asterisk mediante DNAT para posibilitar a recepción de chamadas.

Regras iptables

No servidor gandalf (69.55.97.40) e necesario configurar as seguines reglas iptables para que Asterisk teña acceso a internet e tamén para ser accesible:

- SNAT para a saída a Internet do servidor Asterisk

```
iptables -t nat -A POSTROUTING -o $INTERNET_ETHERNET -j SNAT --to-source 69.55.97.40
```

- Servicio STUN

```
iptables -A FORWARD -p tcp --dport 3478 -j ACCEPT
iptables -A FORWARD -p udp -m udp --dport 3478 -j ACCEPT
```

- Sinalización SIP. Alguns servidores utilizan tamén o porto TCP 5060

```
iptables -A FORWARD -p udp -m udp --dport 5060 -j ACCEPT
iptables -A FORWARD -p udp -m udp --dport 5060 -j ACCEPT
```

- RTP

```
iptables -A FORWARD -p udp -m udp --dport 10000:20000 -j ACCEPT
```

- Sinalización IAX2

```
    iptables -A FORWARD -p udp -m udp --dport 4569 -j ACCEPT
```

- Sinalización IAX

```
    iptables -A FORWARD -p udp -m udp --dport 5036 -j ACCEPT
```

- MGCP: *media gateway control protocol*

```
    iptables -A FORWARD -p udp -m udp --dport 2727 -j ACCEPT
```

Alternativamente tamén podemos abrir completamente o acceso o servidor Asterisk (mellor así para o soporte de GTalk):

```
    iptables -A FORWARD -d 172.20.100.100 -j ACCEPT
    iptables -A FORWARD -s 172.20.100.100 -j ACCEPT
```

- DNAT

```
    iptables -t nat -A PREROUTING -i $INTERNET_ETHERNET -p udp -m udp --dport 5060 -j DNAT --to-destina
    iptables -t nat -A PREROUTING -i $INTERNET_ETHERNET -p udp -m udp --dport 10000:20000 -j DNAT --to-
    iptables -t nat -A PREROUTING -i $INTERNETETH -p udp -m udp --dport 4569 -j DNAT --to-destination 1
    iptables -t nat -A PREROUTING -i $INTERNETETH -p udp -m udp --dport 5036 -j DNAT --to-destination 1
    iptables -t nat -A PREROUTING -i $INTERNETETH -p udp -m udp --dport 2727 -j DNAT --to-destination 1
```

Configuración Inicial

En primeiro lugar imos crear os *trunk* principais (salvo o IAX2, que veremos máis adiante) e crear un dialplan básico. Comenzaremos polo trunk para a liña RDSI que manexará *LCR* mediante *chan_lcr*. Precisamos editar o ficheiro */usr/local/lcr/routing.conf* e facer que o seu contido sexa:

```
# Linux-Call-Router routing configuration "routing.conf"


# Ruleset: MAIN
# Calls with different origins will be processed in different rulesets.

[main]
        : remote application=asterisk
```

Por último editamos */usr/local/lcr/interface.conf* e facemos que teña:

```
[RDSILine]
portnum 0
```

Con esto conseguiremos un contexto **RDSILine** para utilizar no dialplan que se referirá as chamadas que cheguen por este trunk.

- */etc/asterisk/modules.conf*. Cargamos todos os módulos e seleccionamos o sistema de son *alsa*

```
;
; Asterisk configuration file
;
; Module Loader configuration file
;

[modules]
autoload=yes
;
; Any modules that need to be loaded before the Asterisk core has been
; initialized (just after the logger has been initialized) can be loaded
; using 'preload'. This will frequently be needed if you wish to map all
; module configuration files into Realtime storage, since the Realtime
; driver will need to be loaded before the modules using those configuration
; files are initialized.
;
; An example of loading ODBC support would be:
;preload => res_odbc.so
;preload => res_config_odbc.so
;
; res_phoneprov requires func_strings.so to be loaded:
preload => func_strings.so
;
; Uncomment the following if you wish to use the Speech Recognition API
;preload => res_speech.so
;
; If you want, load the GTK console right away.
;
noload => pbx_gtkconsole.so
;load => pbx_gtkconsole.so
;
load => res_musiconhold.so
;
; Load one of: chan_oss, alsa, or console (portaudio).
; By default, load chan_oss only (automatically).
;
;noload => chan_alsa.so
noload => chan_oss.so
noload => chan_console.so
```

- */etc/asterisk/rtp.conf*. Seleccionamos os portos RTP dende 10000 a 20000.

```
;
; RTP Configuration
;
[general]
;
; RTP start and RTP end configure start and end addresses
;
; Defaults are rtpstart=5000 and rtpend=31000
;
rtpstart=10000
rtpend=20000
```

- */etc/asterisk/sip.conf*. *Trunks* sip para a conexión dos teléfonos IP e os softphones. Definimos o seguinte:

PhoneIP
: Teléfono IP/Softphone de probas. Utilizará o contexto **PhoneIP** no dialplan.

PhoneA4
: Teléfono IP/Softphone situado no Aula IF-3. Utilizará o contexto **PhoneA4** no dialplan.

Departamento
: Teléfono IP/Softphone situado no Departamento de Informática. Utilizará o contexto **Departamento** no dialplan.

rodeirapbx
: Enlace SIP con *ekiga.net*. O contexto a utilizar no dialplan será *EkigaLine*, e a extensión *ekiga*.

xavi
: Entrada para un softphone que poderá estar situado en calqueira lugar. O contexto para o dialplan a utilizaŕ será **xavi**.

Probas
: Igual que a entrada anterior, pero co contexto do dialplan **Probas**.

```
[general]
context=default
srvlookup=yes
videosupport=no
nat=yes
localnet=172.16.0.0/255.255.0.0
externip=69.55.97.40
fromdomain=iesrodeira.com
qualify=yes
realm=asterix.iesrodeira.com

disallow=all
allow=alaw
allow=ulaw
allow=speex
allow=ilbc
allow=gsm
allow=h261

register => rodeirapbx:**********@ekiga.net/ekiga

[xavi]
type=friend
defaultuser=xavi
context=xavi
secret=xxxxxxxxx
;canreinvite=yes
canreinvite=nonat
;canreinvite=no
host=dynamic
dtmfmode=rfc2833
qualify=3000

[Probas]
type=friend
defaultuser=Probas
context=Probas
secret=xxxxxxxxx
;canreinvite=yes
canreinvite=nonat
;canreinvite=no
host=dynamic
dtmfmode=rfc2833
qualify=3000

[PhoneIP]
type=friend
defaultuser=PhoneIp
context=PhoneIP
secret=xxxxxxxxx
;canreinvite=yes
canreinvite=nonat
;canreinvite=no
host=dynamic
dtmfmode=rfc2833
qualify=3000

[PhoneA4]
type=friend
defaultuser=PhoneA4
context=PhoneA4
secret=xxxxxxxxx
;canreinvite=yes
canreinvite=nonat
;canreinvite=no
host=dynamic
dtmfmode=rfc2833
qualify=3000

[Departamento]
type=friend
defaultuser=Departamento
context=Departamento
secret=xxxxxxxxx
;canreinvite=yes
canreinvite=nonat
;canreinvite=no
host=dynamic
dtmfmode=rfc2833
qualify=3000

[rodeirapbx]
type=peer
context=EkigaLine
```

Tests a realizar

O plan de marcado para realizar os test será a base de un *IVR* coas seguintes extensións:

- 500 - Servicio ECO de *ekiga.net*
- 600 - Servicio ECO de Asterisk

Rexistramos Asterisk nun proveedor VoIP

Coa configuración anterior, Asterisk se rexistra correctamente en *ekiga.net*.

Rexistramos un cliente Teléfono IP no Asterisk dende a rede local

Rexistramos un cliente softphone no Asterisk dende a rede local

Se rexistran sin problemas os softphones *ekiga* utilizando a conta *Probas*. A configuración SIP non utiliza nin servidor STUN nin proxy SIP.

Rexistramos un cliente Teléfono IP no Asterisk dende Internet

Rexistramos un Teléfono IP na conta *PhoneIP* sen maiores problemas, utilizando o servidor STUN *stun.ekiga.net*, e activando Enable Strict Proxy, Auto Detect Server e Enable Via rport.

Rexistramos un cliente softphone IP no Asterisk dende Internet

Se rexistran sin problemas os softphones ekiga utilizando a conta Probas. A configuración SIP non utiliza como servidor STUN *stun.ekiga.net*.

Chamadas dende liña telefónica convencional (PSTN)

O servicio de ECO de Asterisk
: A chamada se realiza correctamente coa configuración inicial.

O servicio de ECO de *ekiga.net* a través do Asterisk
: A chamada se realiza correctamente coa configuración inicial.

A un softphone/Teléfono IP rexistrado en Asterisk situado en Internet
: A chamada se realiza correctamente coa configuración inicial.

A un softphone/Teléfono IP rexistrado en Asterisk situado na rede local
: A chamada se realiza correctamente coa configuración inicial.

A un softphone rexistrado en *ekiga.net*
: A chamada se realiza correctamente coa configuración inicial.

Chamadas dende un softphone/Teléfono IP situado en Internet

Sin rexistrar o Asterisk (contexto *default*)
: As chamadas funcionan correctamente (do mesmo xeito que os softphones rexistrados) e se dirixen o contexto *default* do dialplan.

O servicio de ECO de Asterisk
: A chamada funciona correctamente coa configuración inicial.

O servicio de ECO de *ekiga.net* a través do Asterisk
: E posible realizar a chamada mediante *packet2packet bridge* ou sin bridge (*t* no Dial) e os dous casos funcionan dun xeito similar. Se reciben interrupcións intermitentes debido a problemas de rede.

A un softphone/Teléfono IP rexistrado en Asterisk situado en Internet
: En principio debería funcionar o modo *native bridge*, pero o test realizado e cos dous softphones na mesma rede externa a da Asterisk, o que imposibilita (polo menos en principio) o *native bridge*. Nos outros dous modos (*packet2packet bridge* e sen bridge) funciona correctamente cos problemas de saturación de rede xa comentados.

A un softphone/Teléfono IP rexistrado en Asterisk situado na rede local
: A chamada se procesa en modo *native bridge*. Si a chamada é dende un softphone (*ekiga*) a chamada se interrumpe e se recupera continuamente, pero o colgar tamén o fai o softphone local. Si a chamada é dende un teléfono IP a chamada se establece correctamente, pero o colgar o teléfono IP, o softphone local non colga. Outros softphones probados (*X-Lite* e *twinkle*) tamén se comportan do mesmo modo que o teléfono IP, a chamada se realiza correctamente, pero o colgar o softphone a parte remota non colga. Si colgamos o softphone da rede local o lado remoto remata correctamente. Si o softphone local está dentro do firewall (mesma rede que o servidor Asterisk) é necesario utilizar *packet2packet bridge*.

A un softphone rexistrado en *ekiga.net*
: A chamada funciona correctamente (*bridge packet2packet*), pero presenta os problemas descritos con anterioridade en este tipo de *bridge*. O softphone *X-Lite* funciona con menos interrupcións que *ekiga* ou o teléfono IP. Si utilizamos o modo a través do Asterisk sen bridge ('t' no Dial) funciona de maneira similar.

A un teléfono da liña telefónica convencional (PSTN)
: A chamada funciona correctamente coa configuración inicial.

Chamadas dende un softphone/Teléfono IP situado na rede local

O servicio de ECO de Asterisk
: Funciona correctamente coa configuración inicial.

O servicio de ECO de *ekiga.net* a través do Asterisk
: Funciona correctamente co *packet2packet bridge*, pero cos problemas de rede descritos.

A un softphone/Teléfono IP rexistrado en Asterisk situado en Internet
: A chamada se efectua correctamente en modo *native bridge* coa configuración inicial dende a rede 69.55.97.x e con *packet2packet bridge* dende 172.20.x.x e 172.16.x.x.

A un softphone/Teléfono IP rexistrado en Asterisk situado na rede local
: A chamada se efectua correctamente en modo *native bridge*. Si forzamos o *packet2packet bridge* os problemas son os mesmos descritos con anterioridade.

A un softphone rexistrado en *ekiga.net*
: Funciona correctamente co *packet2packet bridge*, pero cos problemas de rede descritos.

A un teléfono da liña telefónica convencional (PSTN)
: Funciona correctamente coa configuración inicial.

Chamadas dende un softphone rexistrado no proveedor *ekiga.net*

O servicio de eco de Asterisk
: A chamada funciona correctamente coa configuración inicial.

A un softphone/Teléfono IP rexistrado en Asterisk dende Internet
: A chamada funciona correctamente coa configuración inicial mediante *packet2packet bridge* cos problemas xa indicados.

A un softphone/Teléfono IP rexistrado en Asterisk dende a rede local
: A chamada funciona correctamente coa configuración inicial mediante *packet2packet bridge* cos problemas xa indicados.

A un teléfono da rede telefónica convencional (PSTN)
: A chamada funciona correctamente coa configuración inicial

Conclusións e Problemas

O modo *native bridge* parece funcionar correctamente nos casos nos que se pode empregar, os modos *packet2packet* e sen bridge parecen producir problemas intermitentes de conectividade na rede, con tempos de ping a intervalos aleatorios que poden chegar a 6 ou 7 segundos. As posibilidades a explorar son as seguintes:

- Problema de configuración do kernel do sistema Asterisk.
- Problema de parámetros de rede no sistema Asterisk.
- Trafico UDP en xeral na rede local.
- Problemas cos switches e os segmentos de rede a Gigabit Ethernet e de 100Mbits.

NOTA 1: De momento inclínome a considerar a última posibilidade, xa que parece que o segmento de rede que se satura e o comprendido entre o firewall (172.20.2.1) e o equipo Asterisk (172.20.100.100), que consta dun segmento de rede a Gigabit Ethernet e outro a 100 Mbits e dous switches intermedios.

NOTA 2: Probando con dous softphones locais no mesmo segmento de rede (switch común) os problemas co *packet2packet bridge* continúan, o que me leva a pensar que o problema está no propio switch.

NOTA 3: Fago un test con un softphone en 172.20.3.100 e outro en 172.20.3.6 forzando *packet2packet bridge*. O servidor asterisk esta en 172.20.100.100 e todos conectados o mesmo switch. A porta de saída da rede é 172.20.2.1 (dirección "externa" 69.55.97.40) e se atopa dous switches por debaixo. Un tramo 100Mb (dos ordenadores o switch) e dous tramos 1Gb de switch a switch e de switch o servidor. O retraso (intermitentes entre 2 e 12 segundos) detéctase curiosamente nos ping entre 172.20.3.100 e 172.20.2.1 (onde teóricamente non temos circulación UDP, e con *iptraf* non se observa) mentras que os ping entre 172.20.3.100 e 172.20.3.6 son normais, e entre 172.20.3.100 e 172.20.100.100 (O servidor Asterisk) tamén. Curiosamente os pings dende 172.20.2.1 os outros equipos da rede tamén son normais.

NOTA 4: Os paróns no ping (2 a 8 segundos) danse entre 172.20.2.1 e as partes que interveñen na comunicación: 172.20.3.100, 172.20.3.6 e 172.20.100.100, coincidindo cos cortes de voz. O resto de pings é normal.

Integración do Asterisk Debian con Asterisk Trixbox

DUNDI

O servicio "DUNDI" permite a un servidor Asterisk facer públicas e accesibles certas extensións propias a outros servidores mediante varios protocolos. No noso caso empregaremos IAX para enlazar este servidor Asterisk con outro servidor (*obelix.iesrodeira.com*) con Trixbox. En primeiro lugar é necesario crear o Trunk IAX para enlazar os dous servidores, no servidor que estamos a instalar (*asterix.iesrodeira.com*) editamos */etc/asterisk/iax.conf", e no servidor Trixbox (*obelix.iesrodeira.com*) /etc/asterisk /iax_custom.conf:*

```
;asterix.iesrodeira.com
[general]
language=es
jitterbuffer=no
forcejitterbuffer=no
autokill=yes

[dundi]
type=user
dbsecret=dundi/secret
context=DUNDI-calls
disallow=all
allow=alaw
allow=ulaw
allow=speex
allow=ilbc
allow=gsm
allow=h261
```

```
;obelix.iesrodeira.com
language=es

[dundi]
type=user
dbsecret=dundi/secret
context=DUNDI-calls
disallow=all
allow=alaw
allow=ulaw
allow=speex
allow=ilbc
allow=gsm
allow=h261
```

Como se pode observar, o contexto do dialplan o que se vai a acceder a través deste *trunk* chámase *DUNDI-calls*. Agora modificamos o dialplan, en *asterix* editamos */etc/asterisk/extensions.conf* e en *obelix* */etc/asterisk/extensions_custom.conf* para engadir o contexto *[DUNDI-calls]* coas extensións que desexemos facer públicas. No noso caso, dividiremos as extensións a facer públicas en tres categorías:

- *[DUNDI-voip]* - Incorpora as extensións dos teléfonos IP rexistrados permanentemente (Departamentos, Aulas ... etc)
- *[DUNDI-dinamico]* - Incorpora as extensións dos teléfonos IP rexistrados de cando en vez dende sitios distintos (Profesores, alumnos ... etc)
- *[DUNDI-pstn]* - Incorpora as extensións para facer chamadas a través da PSTN.

Deste xeito quedaría logo o contexto *[DUNDI-calls]*:

```
[DUNDI-calls]
include => DUNDI-voip
include => DUNDI-dinamico
include => DUNDI-PSTN
```

Tamén creamos un contexto *[DUNDI-switch]* que permite facer as consultas DUNDi o outro servidor. Para eso bastará cun *include => DUNDI-switch* nos contextos nos que queiramos ter acceso os números importados a través de DUNDi:

```
[DUNDI-switch]
switch => DUNDI/dundi-trunk
```

dundi-trunk fai referencia o mapeo de extensións que se realiza en */etc/asterisk/dundi.conf* en ambos servidores:

asterix.iesrodeira.com

```
[general]
department=Departamento de Informática
organization=IES de Rodeira
locality=Cangas do Morrazo
stateprov=GZ
country=ES
email=xavi@iesrodeira.com
phone=986304750
ttl=32
autokill=yes

; Contextos exportados
[mappings]
dundi-trunk => DUNDI-voip,0,IAX2,dundi:${SECRET}@${IPADDR}/${NUMBER},nopartial
dundi-trunk => DUNDI-dinamico,100,IAX2,dundi:${SECRET}@${IPADDR}/${NUMBER},nopartial
dundi-trunk => DUNDI-PSTN,400,IAX2,dundi:${SECRET}@${IPADDR}/${NUMBER},nopartial

; Servidor o que podemos consultar (obelix.iesrodeira.com)
; O número de abaixo é a MAC do primeiro interface de rede, pero pode cambiarse en dundi.conf
[00:80:5A:4A:88:3D]
model = symmetric
host = obelix.iesrodeira.com
inkey = obelix.iesrodeira.com
outkey = asterix.iesrodeira.com
include = dundi-trunk
permit = dundi-trunk
qualify = yes
```

obelix.iesrodeira.com

```
[general]
department=Departamento de Electrónica
organization=IES de Rodeira
locality=Cangas do Morrazo
stateprov=GZ
country=ES
email=voip@iesrodeira.com
phone=986304750
ttl=32
autokill=yes

[mappings]
dundi-trunk => DUNDI-voip,0,IAX2,dundi:${SECRET}@172.16.23.99/${NUMBER},nopartial
dundi-trunk => DUNDI-dinamico,100,IAX2,dundi:${SECRET}@172.16.23.99/${NUMBER},nopartial
dundi-trunk => DUNDI-PSTN,400,IAX2,dundi:${SECRET}@172.16.23.99/${NUMBER},nopartial

; asterix.iesrodeira.com
[00:05:1c:00:b6:ae]
model = symmetric
host = asterix.iesrodeira.com
inkey = asterix.iesrodeira.com
outkey = obelix.iesrodeira.com
include = dundi-trunk
permit = dundi-trunk
qualify = yes
```

Neste caso colocamos a IP nos *mappings* porque a variable ${IPADDR} ten a outra IP do servidor (a *pública*), que non me sirve para enlazar con *asterix*.

Por último so e necesario crear as chaves de encriptación e poñelas na outra máquina. Estas chaves se crean en ambos servidores en */var/lib/asterisk/keys* executando **astgenkey -n asterix.iesrodeira.com *no servidor* asterix *e*** *astgenkey -n obelix.iesrodeira.com'* en *obelix*, a continuación é preciso copiar a chave *obelix.iesrodeira.com.pub* a carpeta */var/lib/asterisk/keys* de *asterix*, e a chave *asterix.iesrodeira.com.pub* a carpeta */var/lib/asterisk/keys* de *obelix*.

Podemos comprobar a información dende a consola Asterisk cos comandos *dundi query* e *dundi lookup*

Síntesis de voz: Festival

Unha vez configurado Festival como indica o apartado anterior, para utilizalo bastará con utilizar a función do dialplan *Festival* pasandolle como argumento a cadea de texto que se quere leer. Ollo, non se deben pasar acentos, xa que Festival non os procesa:

```
exten => menu,n,Festival(Para chamar o departamento de informatica pulsa uno)
```

Recoñecimento de Voz

Sphinx,

Buzón de Voz

Para activar o buzón de voz e necesario utilizar o arquivo */etc/asterisk/voicemail.conf*, e necesario ter cuidado en ter o parámetro *delete=no* si non queremos perder as mensaxes que se envíen por correo. Na configuración inicial deste Asterisk (configuración estática por medio de ficheiros) e necesario que o módulo *app_voicemail* esté compilado sen opción ODBC nin IMAP. Un erro no buzon de voz fai que se busque a voz **y** na carpeta de *digits* en lugar de *letters*, para solucionalo establecemos un enlace simbólico dende *letters* a *digits*.

```
ln -s /var/lib/asterisk/sounds/es/letters/y.alaw /var/lib/asterisk/sounds/es/digits/y.alaw
```

E o mesmo cos codecs que se estime necesario.

Configuración Real Time

Plan de Marcado

DISA

Extensións a Configurar e Id's

Chamadas "Internas" (teléfonos rexistrados)

- 1: Menú IVR do IES de Rodeira
- 500: Test de ECO
- 600: Buzon de Voz
- 1000: Departamento Informática
- 2000: Aula 3
- 3000: Aula 4
- 3100: Softphone/IP Phone de Xavi
- 5xxx: Chamadas a usuarios Ekiga.net
- 6xxx: Chamadas a usuarios de Gtalk
- Extensións recollidas por DUNDI de obelix.iesrodeira.com

Chamadas "Restrinxidas" (so teléfonos rexistrados, e non todos)

- inclúe as chamadas "Internas"
- _0X.: Chamada a un número da PSTN

Chamadas "Públicas" (Chamadas dende calqueira teléfono VoIP non rexistrado, dende a PSTN, dende ekiga.net ou dende Gtalk

- Menu IVR do IES de Rodeira, permitindo teclear un número de extensión das chamadas "Internas"

Chamadas "Especiais" (so algúns teléfonos rexistrados)

- 6666: Gravador de voz, a voz queda gravada en */tmp/prompt%d.wav*

Todos os teléfonos rexistrados (que poden recibir chamadas) terán buzón de voz.

Plan de Marcado

Batería de Probas

Chamada PSTN (RDSI-IF3)<->IF-3 (Softphone, Teléfono IP, Teléfono convencional)
Chamada VoIP<->IF-3 (Softphone, Teléfono IP, Teléfono convencional) vía Asterisk IF-3
Chamada VoIP<->VoIP (distinto proveedor) vía Asterisk IF-3
Chamada IF-3<->PSTN via servicio VoIP co Asterisk IF-3

Chamada PSTN (RDSI IF-3)<->Electrónica
Chamada PSTN (Liña Electrónica)<->IF-3 (Softphone, Teléfono IP, Teléfono convencional)
Chamada IF-3<->Teléfonos Electrónica
Chamada IF-3<->PSTN via servicio VoIP en Trixbox Electrónica
Chamada Servicio VoIP<->Servicio VoIP (distinto proveedor) pasando por Trixbox Electrónica

ACTIVIDADE DE AULA
O ESTUDO DO PROTOCOLO SIP

OBXECTIVOS

Coñecer o funcionamento interno que permite o envío de VoIP a través dunha rede de ordenadores. Polo tanto, estudaranse en detalle os diferentes protocolos implicados na transmisión de información VoIP.

1 A PILA DE PROTOCOLOS DE INTERNET

Previo á descrición do protocolo SIP,
comezaremos repasando a torre de protocolos
de xeito que posteriormente saibamos situar onde vai o protocolo RTP e SIP.

A torre de protocolos que usan os ordenadores conectados a Internet ten os seguintes niveis:

- **Nivel de acceso á rede:** este nivel abarca os niveis físico e de enlace do coñecido modelo OSI:
    - **Nivel físico**: refírese ás sinais que se envían a través do medio que usamos para a transmisión e recepción da información. Por exemplo se estamos nunha rede ethernet que usa un cable de par trenzado refírese ás sinais eléctricas que transmitimos a través dos pins 1 e 2 e que no outro extremos recibimos nos pins 3 e 5; se estamos falando dunha liña ADSL, refírese as sinais eléctricas que se transmiten pola liña telefónica; ...
    - **Nivel de enlace**: o seu obxectivo é converter un enlace entre dous equipos conectados directamente a través dun mesmo medio, nun enlace seguro e fiable libre de erros. Neste nivel determínanse o formato das tramas que se van a usar para transmitir a información, e doutra para implantar mecanismos de control da conxestión, de detección de erros, ... Por exemplo, nunha rede ethernet os diferentes campos que forman unha trama ethernet permiten identificar a tarxeta orixe, a tarxeta destino, ... tal como se mostra a continuación:

Preámbulo	**SOF**	**Destino**	**Orixe**	**Lonxitude**	**Datos**	**Recheo**	**FCS**
7 bytes	1 byte	6 bytes	6 bytes	2 bytes	0 a 1500 bytes	0 a 46 bytes	4 bytes

Preámbulo

Un campo de 7 bytes (56 bits) cunha secuencia de bits usada para sincronizar e estabilizar o medio físico antes de iniciar a transmisión de datos. O patrón do preámbulo é:

10101010 10101010 10101010 10101010 10101010 10101010 10101010

Estes bits transmítense en orde, de esquerda a dereita e na codificación Manchester representan unha forma de onda periódica.

SOF (Start Of Frame) Inicio de Trama

Campo de 1 byte (8 bits) cun patrón de 1s y 0s alternados e que termina con dous 1s consecutivos. O patrón do SOF é: 10101011. Indica que o seguinte bit será o bit máis significativo do campo de dirección MAC de destino.

Aínda que se detecte unha colisión durante a emisión do preámbulo ou do SOF, o emisor debe continuar enviando todos os bits de ambos até o fin do SOF.

Dirección de destino

Campo de 6 bytes (48 bits) que especifica a dirección MAC de tipo EUI-48 cara a que se envía a trama. Esta dirección de destino pode ser dunha estación, dun grupo multicast ou a dirección de broadcast da rede. Cada estación examina este campo para determinar se debe aceptar a trama (si é a estación destino).

Dirección de orixe

Campo de 6 bytes (48 bits) que especifica a dirección MAC de tipo EUI-48 dende a que se envía a trama. A estación que deba aceptar a trama coñece por este campo a dirección da estación orixe coa cal intercambiará datos.

Tipo

Campo de 2 bytes (16 bits) que identifica o protocolo de rede de alto nivel asociado coa trama ou, no seu defecto, a lonxitude do campo de datos. A capa de enlace de datos interpreta este campo. (Na IEEE 802.3 é o campo lonxitude e debe ser menor ou igual a 1526 bytes.)

Datos

Campo de 0 a 1500 Bytes de lonxitude. Cada Byte conten unha secuencia arbitraria de valores. O campo de datos é a información recibida do nivel de rede (a carga útil). Este campo, tamén inclúe os H3 e H4 (cabeceiras dos niveis 3 e 4), provenientes de niveis superiores.

Recheo

Campo de 0 a 46 bytes que se utiliza cando a trama Ethernet non alcanza os 64 bytes mínimos para que non se presenten problemas de detección de colisións cando a trama é moi corta.

FCS (Frame Check Sequence - Secuencia de Verificación de Trama)

Campo de 32 bits (4 bytes) que conten un valor de verificación CRC (Control de redundancia cíclica). O emisor calcula o CRC de toda a trama, dende o campo destino ao campo CRC supoñendo que vale 0. O receptor recalcúlao, se o valor calculado é idéntico a trama é valida.

- **Nivel de rede**: este nivel introduce un concepto moi importante, xa que obvia os diferentes medios que ten que atravesar unha información para chegar dende un equipo orixe até outro equipo remoto, e convertelo nun único medio virtual a través do cal eses dous equipos conectan. Dito de outra forma, é como se todos os ordenadores que hai no mundo estivesen conectados entre si usando un mesmo medio virtual, obviando os diferentes medios que ten que

atravesar esa información: ethernet, ADSL, Frame Relay, RTB, WiFi, ... Os protocolos máis coñecidos do nivel de rede son: IP, IPX (de Novell) ou NetBEUI (de Microsoft). De todos estes protocolos o máis importante e o máis estendido é o IP, ademais de ser o que usa Internet. Pola súa importancia no noso obxectivo de explicar o protocolo SIP, dedicaremos posteriormente un apartado específico.

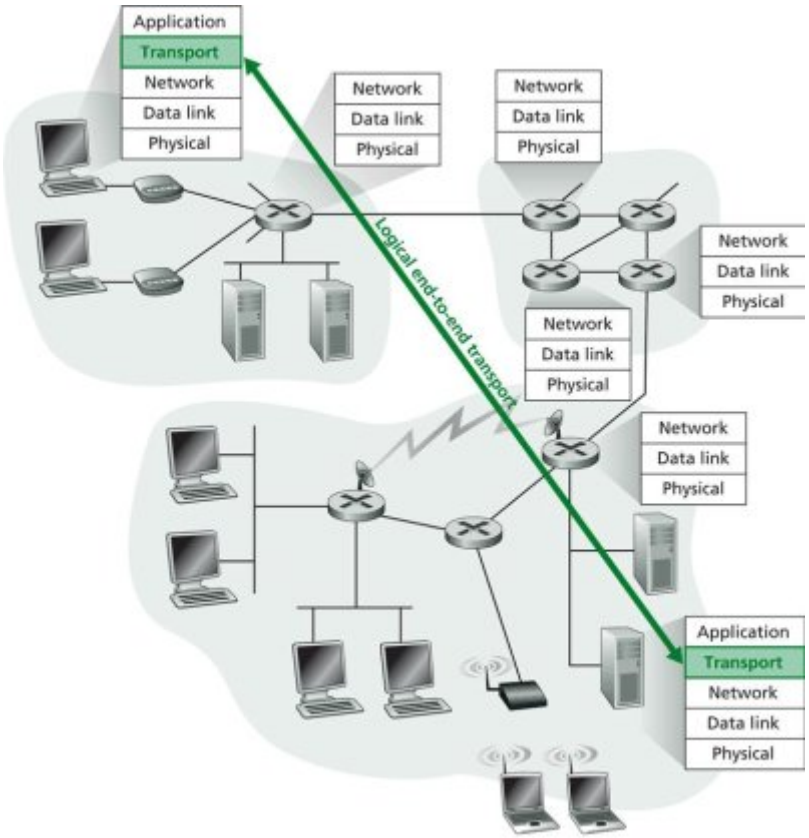
- **Nivel de transporte**: se co protocolo IP acadabamos unha comunicación entre dous equipos, co nivel de transporte acadamos unha comunicación entre dúas aplicacións remotas. Estamos a falar do que habitualmente chámanse *portos*. Pódese facer un simil moi bo, entre todo o explicado e o envío dunha carta: a carta envíase a un piso concreto dun edificio; polo que é levada dende o piso do edificio do destinatario até a oficina de correos onde a deposita; despois a carta é levada a través de diferentes oficinas de correos, atravesando diferentes medios, até chegar á oficina de correos máis próxima ao destinatario; entón a carta é levada polo carteiro até o edificio onde vive o destinatario (dirección de rede ou dirección IP) é finalmente depositada no caixón de correos do destinatario (dirección de transporte ou porto destino). Existen dous protocolos de transporte importantes en Internet: TCP e UDP, o primeiro establece un circuíto virtual de xeito que os paquetes chegan en orde ao destinatario, mentres que UDP utiliza un mecanismo de datagramas onde cada paquete é enviado ao destinatario a través de diferentes camiños co cal non se asegura que cheguen na mesma orde en que foron enviadas.
  É importante destacar que hai servizos servidor que se atopan sempre en portos estándar como o www que está no porto 80, o smtp (correo saínte) que está no porto 25, o POP3 (correo entrante) que escoita no porto 110, ... e por outra parte están as aplicacións clientes que tamén van usando portos que non están sendo usados por outra aplicación e que se atopan sempre por riba do porto 1024. Así, se un cliente web (ou sexa un Internet Explorer ou un Mozilla Firefox) está executándose unha instancia nunha máquina, pode estar usando o porto 1395; pero se abrimos outra instancia do mesmo programa, terá que usar outro porto distinto, de xeito que cando se solicite unha páxina web o servidor ao cal lle solicitamos esa páxina saiba non só a que máquina identificada coa súa dirección IP ten que contestar, senón tamén a que porto desa máquina ten que enviarlle esa contestación.

- **Nivel de aplicación**: neste nivel móntanse as diferentes aplicacións ou protocolos que conteñen a información específica da aplicación informática que estamos a usar: correo electrónico, páxinas web, VoIP, ...

En resumo, no nivel de enlace os datos envíanse entre equipos dentro da mesma LAN, conectados por un *switch* fisicamente. No nivel de rede os datos envíanse a través de distintas redes conectadas fisicamente mediante *routers*. No nivel de transporte hai unha conexión directa entre os dous extremos que interveñen na comunicación pero é unha **conexión lóxica**.

Por conexión lóxica entendemos que hai unha comunicación directa entre os programas ou equipos que interveñen na comunicación, independentemente da rede onde se atopen pero **non existe conexión física directa entre eles**. Existe, polo tanto, un **circuíto virtual** entre o emisor e receptor. A capa de transporte proporciona comunicación

lóxica extremo a extremo. O nivel de transporte non é consciente, nin debe selo, de como están interconectados fisicamente os dous equipos (por unha LAN, unha WAN ou unha combinación de múltiples redes de ambos tipos). Que a comunicación sexa extremo a extremo quere dicir que non se coñecen os detalles da subrede, polo tanto, o nivel de transporte non sabe nada de *routers*, fragmentación, *switches*, *hubs*, etc.



2 ENCAPSULAMENTO DA INFORMACIÓN

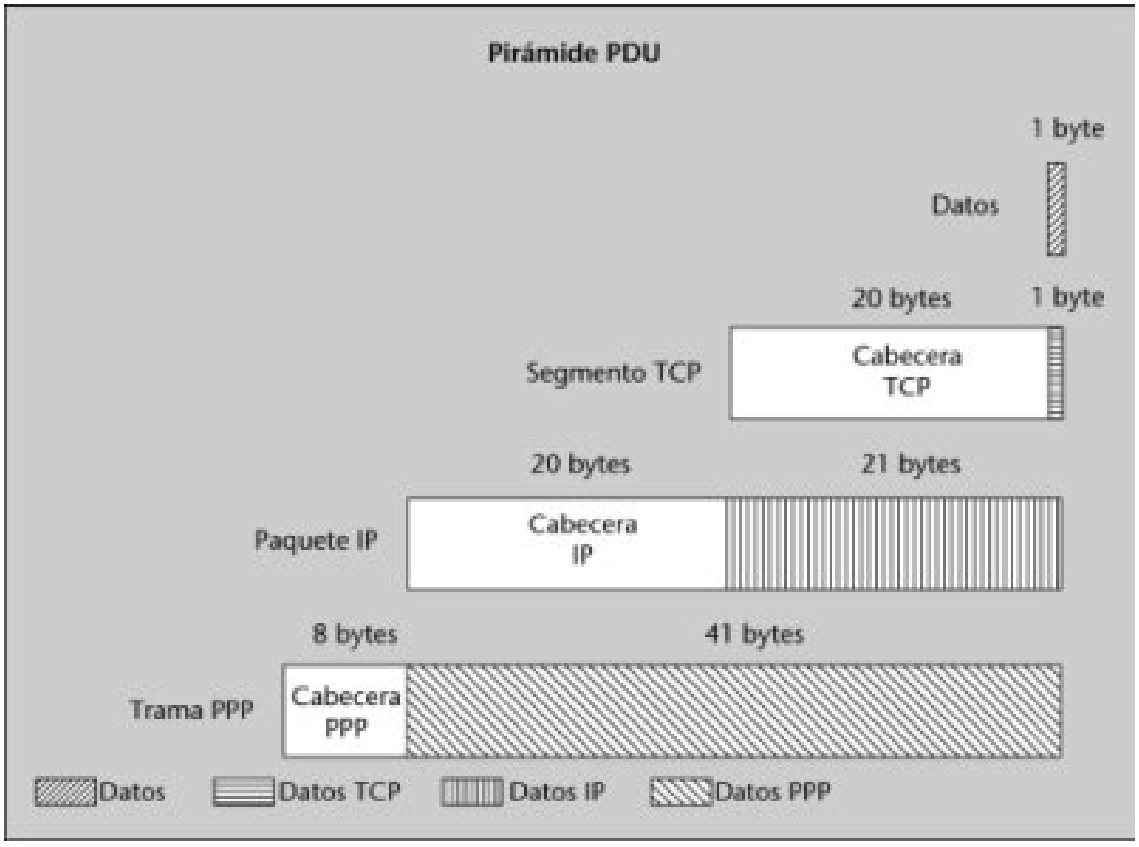
Unha vez estudada a torre de protocolos,
como se mete a información dun nivel superior dentro dun nivel inferior.
Para o protocolo de nivel inferior, esa información son datos que do nivel superior.

Cada protocolo funciona cunhas estruturas fundamentais que se chaman PDU (Unidade de Datos do Protocolo). En cada nivel da pila a PDU ten un nome distinto:

- Nivel de enlace/físico: **Tramas**
- Nivel de rede: **Paquetes**
- Nivel de transporte: **Segmentos (TCP) e datagramas (UDP)**
- Nivel de aplicación: **Mensaxes** ou **datos**

Os datos que xere unha aplicación dividiranse en varios partes que se incluirán en segmentos (PDU do nivel de transporte) que á súa vez se dividirán en varios paquetes (nivel de rede), e estes, en varias tramas (nivel de enlace). Este mecanismo coñecese como **encapsulamento** ou empaquetado e é similar ao das famosas monecas rusas.

Polo tanto, cada nivel engade información do protocolo á PDU do nivel superior producíndose unha cascada de PDU que vai descendendo ata o nivel inferior, que finalmente é quen transmite fisicamente os bits en forma de impulsos eléctricos.



3 O FORMATO DO PAQUETE IP

Á **PDU** (*Unidade de Datos do Protocolo*) do nivel de rede chámaselle paquete ou datagrama. Para entender moitas das funcionalidades do IP hai que estudar o formato do paquete. Este, igual que a trama Ethernet, ten unha cabeceira e uns datos (**payload**). O formato da cabeceira é o seguinte:

E o significado dos campos é o que segue:

- **Versión**: Versión do protocolo IP do datagrama. Normalmente será IPv4.
- **HLEN**: Lonxitude da cabeceira medida en palabras de 32 bits (1 palabra de 32 bits é igual a unha fila do debuxo).
- **Lonxitude total**: Medido en bytes. Inclúe os bytes de cabeceira e dos datos. O campo ten 16 bits, polo tanto, o paquete IP poderá ter, como máximo, 2^16 = 65536 bytes ou 64KBytes
- **Tipo de servizo**: Para especificar a **prioridade** do datagrama, **fiabilidade**, **retardo**, etc. Os routers non fan moito caso deste campo. **Moi importante este campo en VoIP xa que logo falaremos del, cando esteamos no apartado de "Garantir a Calidade do Servizo"**
- **Tempo de vida** (*Time To Live*): Especifica o tempo que o datagrama pode estar na rede. Ao pasar polos routers, estes van decrementando este valor. Se chega a 0 e o datagrama non chegou ao destino descartarase.
- **Protocolo**: O protocolo do nivel de transporte que encapsula (TPC, UDP, ICMP)
- **Checksum**: Úsase para comprobar que a cabeceira chegou correctamente.
- **Direccións IP**: De orixe e destino do paquete.
- **Opcións**: Úsase para probar de rede, depuración, etc.
- **Datos**: Contén a PDU do nivel superior, por exemplo, un segmento TCP.

4 PROTOCOLOS USADOS PORVOIP

Para implantar un sistema de VoIP, precisamos dos seguintes protocolos:

- O Protocolo de Transporte de voz en tempo Real (**RTP**) que opera enriba do protocolo UDP, e que se encarga de transmitir a información da comunicación de voz.
- O Protocolo de Control de Transporte de voz en tempo Real (**RTCP**) que tamén opera enriba do protocolo UDP, e que se encarga de informar

periodicamente aos participantes da calidade e outra información entre os interlocutores.

- O Protocolo de Inicio de Sesión (**SIP**), tamén opera sobre UDP (aínda que pode usar TCP), e é un protocolo de sinalización que se encarga do establecemento, modificación e terminación das comunicacións de voz.
  Existen outros protocolos similares o SIP, como o H.323, aínda que o máis estendido é o SIP. De feito, os coñecidos programas Messenger de Microsoft, ou o iChat de Apple utilizan o SIP.
- Finalmente, enriba de todos estes protocolos, está a aplicación final que interactúa entre o usuario e estes tres protocolos.

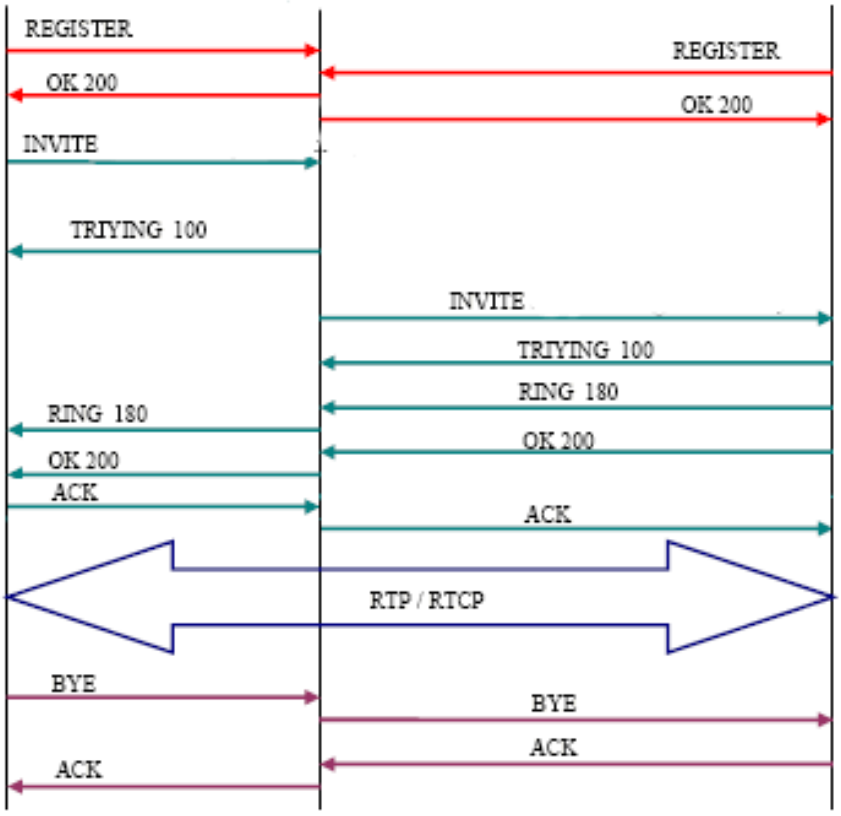
O funcionamento é o seguinte:

1) Utilizase o **protocolo SIP** *para establecer a comunicación co destinatario*.

2) A continuación *envíanse a información de voz* utilizando o **protocolo RTP**.

3) Ao mesmo tempo que se envía a información de voz, e cada 5 segundos *envíase unha información de control entre os interlocutores* (que serve para medir o control da calidade da comunicación, identificar os nomes das persoas que interveñen na comunicación, ...) utilizando o **protocolo RTCP**.

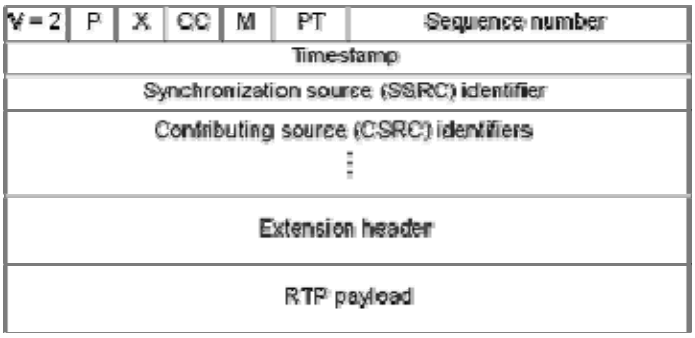
4) Finalmente, vólvese a usar o protocolo SIP para finalizar a comunicación.



5 PROTOCOLO DE TRANSPORTE DE VOZ EN TEMPO REAL (RTP)

O protocolo RTP utilízase para

A seguinte figura mostra o *header* utilizado polo protocolo RTP:

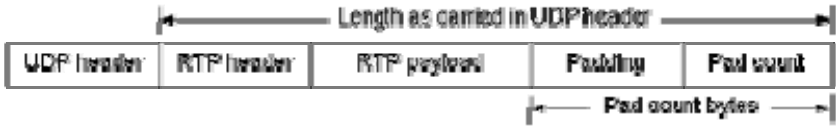


Os primeiros 12 octetos (é dicir, os campos V, P, X, CC, M, PT, sequence number, timestamp e SSRC) SEMPRE están presentes, en tanto que os identificadores de "fontes contribuíntes" (nodos que xeran información ao mesmo tempo para, supoñamos, unha videoconferencia) son utilizados só en certas circunstancias. Despois do *header* "básico" pode terse extensións opcionais para o *header* (Extensión header).

Finalmente o header é seguido polos datos (payload) que transporta RTP e o seu formato é definido pola aplicación.

O deseño do *header* de RTP busca levar só aqueles campos que son necesarios para diversos tipos de aplicacións.

- **V (versión), 2 bits:** Os primeiros dos bits identifican a versión do protocolo.
- **P (padding), 1 bit:** O seguinte bit identifica o *padding*. Informa que os datos de RTP levan un "recheo" para completar un bloque de certo tamaño. O último byte na mensaxe UDP di de qué tamaño é o *padding*



  Con isto cúmprese o obxectivo de ter un *header* RTP pequeno. A lonxitude dos datos calculase a partir da información do *header* do protocolo da capa inferior (UDP neste caso).

- **X (extensión), 1 bit:** O bit de extensión é utilizado para indicar a presenza dun *header* de extensión que pode ser definido para unha aplicación específica e segue ao *header* principal. Ese tipo de *headers* son utilizados en raras ocasións xa que é posible definir un *header* dentro dos datos (*payload*) como parte da definición do formato dos datos para unha aplicación en particular.

- **CC (CSRC count), 4 bits:** O bit X é seguido por 4 bits (CC) que contan o número de "fontes contribuíntes" incluídas no *header* de RTP (en caso de que existan ditas fontes).
- **M (marker), 1 bit:** Este bit é utilizado para indicar o *frame*. Por exemplo, pode indicar o inicio dunha conversación en RTP: o primeiro frame.
- **PT (payload type), 7 bits:** Os seguintes 7 bits indican qué tipo de dato multimedia se está transportando (*payload type*), é dicir, **identifica ao codec utilizado para codificar a información que vai na carga útil**. Isto é moi importante, xa que o sistema de codificación utilizado para enviar a información indícanos por exemplo a frecuencia de mostraxe, e polo tanto o tempo que están separadas cada mostra que envíamos na carga útil do paquete.
  Unha lista deste códigos pódese atopar no IANA que é o organismo que xestiona esta lista. Así por exemplo: o codec "0" utiliza o sistema de codificación G.711 lei μ (PCMU), neste caso a frecuencia de mostraxe é de 8kHz, o que equivale a 125 μs, o que quere dicir que se nun paquete RTP envíanse 5 mostras de voz, entre a primeira mostra deste paquete e a primeira mostra do seguinte paquete transcorreron 625 μs.

- **Sequence number, 16 bits:** O número de secuencia é utilizado para permitir ao receptor dun *stream* RTP detectar paquetes perdidos ou que cheguen en desorde. Observe que RTP non indica qué facer cando se perde un paquete (moi diferente a TCP que corrixe a perdida -por retransmisión- e interpreta a perdida como un indicador de conxestión -que pode levar a reducir o tamaño da ventá en TCP-). Polo contrario, RTP deixa que a aplicación decida qué é o mellor que pode facer cando o paquete se perde.
- **Timestamp, 32 bits:** O campo de *timestamp* permite ao receptor reproducir (playback) as mostras nos intervalos de tempo apropiados e permite que diferentes *media streams* se podan sincronizar. RTP non especifica en que unidades se debe enviar este *timestamp* -as aplicacións e os seus formatos requiren diferentes granularidades en tempo-. O *timestamp* ven a ser un contador de "ticks" onde o tempo entre "ticks" depende do formato de codificación da aplicación (a granularidade do reloxo é un dos detalles que se especifica no *profile* de RTP ou nos datos -*payload*- da aplicación).
- **SSRC, 32 bits:** O identificador de *fonte de sincronización* (SSRC) é un número de 32 bits que identifica de maneira única unha soa fonte nun *stream* RTP. Nunha conferencia multimedia, cada emisor escolle un SSRC aleatorio. O identificador de fonte é diferente da dirección IP ou do número de porto, que permite, por exemplo, que un nodo con múltiples fontes (un equipo con varias cámaras) distinga cada unha das fontes. Cando un só nodo xera diferentes *media streams* (por exemplo, audio e vídeo ao mesmo tempo), non é necesario que utilice o mesmo SSRC en cada *stream* xa que RTCP ten un mecanismo para facer sincronización intermedia.
- **Lista CSRC, de 0 a 15 elementos, cada un de 32 bits:** O identificador de *fonte contribuínte* (CSRC) é utilizado *só* cando varios *streams* RTP pasan a través dun mesturador (*mixer*). Un mesturador pode ser utilizado para reducir os requirimentos de ancho de banda para unha conferencia recibindo datos de moitas fontes e enviando estas como un só *stream*. O número de identificadores incluídos no *header* RTP ven colocado no campo CC (CSRC count). Se hai máis de 15 fontes contribuíntes, só 15 poden ser identificadas.

6 PROTOCOLO DE TRANSPORTE DE CONTROL DE VOZ EN TEMPO REAL (RTCP)

RTCP utilizase para intercambiar información relativa á comunicación entre os interlocutores, unha vez que está establecida a comunicación entre eles. Esta información é complementaria da que se envía a través do protocolo RTP, que é o protocolo que se usa para enviar a información de voz propiamente dita. Polo tanto, a través do protocolo RTCP podemos coñecer o nome das persoas que participan nunha comunicación, o jitter da comunicación (ou flutuación do retardo), cantidade de paquetes perdidos, ...

Existe os seguintes paquetes RTCP:

- SR (Sender Report): serve para a transmisión e recepción das estatísticas dos participantes que son emisores **activos**.
  O paquete SR ten as seguintes partes:
  - Unha cabeceira de 64 bits cos seguintes campos:
    - 2 bits para indicar a versión (V) do protocolo RTP utilizada. Nestes momentos a 2.
    - 1 bit de recheo (P, *padding*), para indicar se ao final do paquete hai bytes de recheo xa que o total ten que ser múltiplo de 32 bits.
    - 5 bits para indicar cantos informes de resposta (RR) imos incluír neste paquete en resposta aos paquetes RTP recibidos dos outros emisores dende o último informe enviado.
    - Tipo de paquete (8 bits): para o paquetes RTCP de tipo SR, este código é 200.
    - Lonxitude do paquete (16 bits): indica o número de grupos de 32 bits que ten o paquete sen contar a cabeceira.
    - SSRC do remitente deste paquete (32 bits).
  - Un informe de emisor de 160 bits, onde se inclúen os seguintes campos:
    - Contador de tempo NTP (64 bits), que indica o instante en que se emitiu o informe, contado dende o 1 de xaneiro de 2000 (GMT), contados en segundos (os 32 bits máis significativos) e en fraccións de 200 picosegundos (os 32 bits menos significativos).
    - Contador de tempo RTP (32 bits), que indica o tempo, no mesmo formato que usa o protocolo RTP.
    - Contador de paquetes RTP enviados polo emisor até este momento (32 bits).
    - Contados de octetos do emisor, contendo só a cantidade útil de información incluída en todos os paquetes enviados até este momento (32 bits).
  - Un informe de receptor de 192 bits, por cada fonte emisora que se escoitara dende o último informe. O formato deste campo é o mesmo que ten os paquetes RR que comentamos a continuación quitándolle a cabeceira.

- RR (Receiver report): serve para a recepción de estatísticas dos participantes que no son emisores activos.
  O paquete RR ten as seguintes partes:
  - Unha cabeceira de 64 bits cos mesmos campos que os paquetes SR agás que no campo Tipo de paquete (PT) toma o valor 201 en lugar de 200.
  - Por cada fonte emisora a que queremos informar, incluímos o seguinte informe:
    - SSRC (32 bits) da fonte a cal queremos informar das estatísticas dos seus paquetes RTP.
    - Fracción de paquetes perdidos (8 bits) en porcentaxe relativo a 256.
    - Total acumulado de paquetes perdidos (24 bits) dende o comezo da comunicación.
    - Ultimo número de secuencia recibido (32 bits, aínda que só hai que ter en conta os 16 bits de menor peso)
    - Flutuación do retardo (ou *jitter*) (32 bits), indica unha estimación da variación que se produce no retardo con que chegan os paquetes RTP. É medido nas mesmas unidades que os contado de tempo dos paquetes RTP, o cal dependía do codecs utilizado.
    - Últimos 32 bits máis baixos transmitidos no campo NTP do último paquete SR recibido desa fonte.
    - Tempo transcorrido entre a recepción do último informe SR recibido e a transmisión deste informe RR de resposta enviado. Este tempo é expresado en fraccións de 1/65536 segundos.

- SDES (Source description): describe a fonte, inclúe o CNAME. O CNAME é o que se chama nome canónico, e identifica a cada usuario mediante un identificador tipo *nomeusuario@88.8.208.68*, indicaría que o usuario *nomeusuario* atópase no ordenador 88.8.208.68. Desta forma o usuario pode ser identificado pola rede e aínda que cambie de ordenador, ser identificado e polo tanto estar localizado para recibir chamadas.
  O paquete SDES ten as seguintes partes:
  - Unha cabeceira de 64 bits cos mesmos campos que os paquetes SR agás que no campo Tipo de paquete (PT) toma o valor 202 en lugar de 200; e en lugar do campo RC indicando o número de informes de resposta que incluímos, pois o substituímos por SC que indica o número de bloques de información que se inclúen neste informe.
  - Por cada bloque de información:
    - Un campo de tipo de información SDES enviada neste bloque de información (8 bits), que pode ser o nome canónico (CNAME) (código 1), o correo electrónico (código 3), o nome do usuario (código 2), número de teléfono (código 4),...
    - Un campo de 8 bits para indicar o tamaño expresado en bytes do texto da información contida neste bloque de información (xa que o tamaño total do bloque de información ten que ser múltiplo de 32 bits)
    - A información de texto propiamente dita.

- BYE: indica un fin da participación no grupo.
  O paquete BYE ten as seguintes partes:
  - Unha cabeceira de 32 bits cos mesmos campos que os paquetes SR agás que no campo Tipo de paquete (PT) toma o valor 203 en lugar de 200; e en lugar do campo RC indicando o número de informes de resposta que incluímos, pois o substituímos por SC que indica o número de bloques de información que se inclúen neste informe. Obsérvese que esta cabeceira non ten 64 bits como no caso das cabeceiras SR e RR, xa que non se inclúe o SSRC do emisor, xa que, en realidade se específica o SSRC de tódalas fontes que deixan de estar activas.
  - SSRC de tódalas fontes que deixan de estar activas (32bits por cada fonte que deixa de estar activa)
  - Lonxitude da texto que explica o motivo polo que deixan de estar activas estas fontes (8 bits)
  - Motivo polo que deixa de estar activa (mínimo 24 bits mais múltiplos de 32 bits).

- APP: funcións especificas de aplicación.
  O paquete APP ten os seguintes campos:
  - Unha cabeceira de 64 bits cos mesmos campos que os paquetes SR agás que no campo Tipo de paquete (PT) toma o valor 204 en lugar de 200; e en lugar do campo RC indicando o número de informes de resposta que incluímos, por iso substituímos por Subtipo cuxo valor é definido polas diferentes aplicacións que usan este tipo de paquetes.

7 PROTOCOLO DE INICIO DE SESIÓN (SIP)

O protocolo SIP utilizase para establecer ou iniciar sesións de comunicación entre dous ou mais participantes.

As mensaxes do protocolo SIP son de dous tipos: solicitudes e respostas. Funcionan de xeito moi parecido o www, é dicir, hai uns clientes que fan unha solicitudes, e uns servidores que responden a estes clientes; pero ao igual que ocorre con protocolo www, tanto as solicitudes como as respostas, fanse utilizando textos facilmente interpretables polas persoas.

O formato das mensaxes é o seguinte:

- Unha **liña de inicio**, que define o tipo de mensaxe de que se trata.
- Unha **cabeceira** formada por un ou máis campos, e que se utiliza para incluír información adicional relativa á solicitude ou resposta.
- Unha **liña en branco** para indicar o final da cabeceira.
- O **corpo da mensaxe, SDP**, que se utiliza para describir as características do tipo de sesión que estamos tentando establecer. É opcional.

7.1 LIÑA DE INICIO

No caso de mensaxe de solicitude existen seis tipos:

- INVITE: para invitar a participar a outro interlocutor.

- ACK: para confirmar ao interlocutor chamado que recibiu a súa resposta de aceptación da chamada.
- OPTIONS: utilizase para preguntarlle a un servidor polas características de chamadas que soporta un servidor SIP (tipos de codecs, ...)
- BYE: para finalizar unha chamada.
- CANCEL: para anular unha solicitude previa.
- REGISTER: para rexistrar a localización actual dun usuario nun servidor SIP.

Por exemplo para comezar unha chamada : INVITE sip:pepe@voiprodeira SIP/2.0

e no caso de mensaxe de resposta, a sintaxe é a seguinte:

- VersiónSIP Código Texto: onde *VersiónSIP* é a versión do protocolo SIP utilizado, *Código* é o código de resposta do estado SIP e *Texto* é un campo opcional que serve para aclarar a resposta.

Por exemplo para indicar timbre soando no destino: SIP/2.0 180

Os códigos de resposta pódense ver na seguinte táboa:

100 – Trying 180 - Ringing 181 - Call Being Forwarded 182 - Call Queued 183 - Session Progress 200 - OK 202 - Accepted 300 - Multiple Choices 301 - Moved Permanently 302 - Moved Temporarily 305 - Use Proxy 380 - Alternative Service 400 - Bad Request 401 - Unauthorized 402 - Payment Required 403 - Forbidden 404 - Not Found 405 - Method Not Allowed 406 - Not Acceptable 407 - Proxy Authentication Required 408 - Request Timeout 409 - Conflict 410 - Gone 411 - Length Required 413 - Request Entity Too Large 414 - Request URI Too Long 415 - Unsupported Media Type 416 - Unsupported URI Scheme 420 - Bad Extension 421 - Extension Required 423 - Interval Too Brief 480 - Temporarily Unavailable

481 - Call/Transaction Does Not Exist
482 - Loop Detected
483 - Too Many Hops
484 - Address Incomplete
485 - Ambiguous
486 - Busy Here
487 - Request Terminated
488 - Not Acceptable Here
491 - Request Pending
493 - Undecipherable
500 - Server Internal Erro
501 - Not Implemented
502 - Bad Gateway
503 - Service Unavailable
504 - Server Time-Out
505 - Version Not Supported
513 - Message Too Large
600 - Busy Everywhere
603 - Declined
604 - Does Not Exist Anywhere
605 - Not Acceptable

7.2 CABECEIRA

A cabeceira das mensaxes está formada por campos de xeito que cada un específica unha característica da comunicación. De entre tódolos campos os máis destacables son os seguintes:

- *Call-ID* Identificador de chamada. Identifica unha chamada de forma que se evite duplicados de paquetes, por exemplo de tipo INVITE
  Exemplo: *Call-ID:123456789@voiprodeira.com*
- *Cseq* Identificación da solicitude. Identifica cada solicitude de xeito que a resposta inclúa o mesmo Cseq.
  Exemplo: *Cseq:1 INVITE*
- *From* Desde. Aparece en todas as mensaxes de solicitude e de resposta para identificar a orixe da solicitude.
  Exemplo: *From:pepe<sip:jose.martinez@vopiprodeira.com>*
- *To* Para. Aparece en todas as mensaxes de solicitude e de resposta para identificar o destino da solicitude.
  Exemplo: *To:lolo manuel.garcia@vopiprodeira.com>*
- *Via* Rexistra a ruta que segue a solicitude. Cada *proxy* polo que pasa engade un campo Via.
  Exemplo: *Via:SIP/2.0/UDP voiprodeira.com*
- *Content-Length* Lonxitude de contido. Lonxitude do corpo da mensaxe en bytes.
  Exemplo: *Content-Length:232*
- *Content-Type* Tipo de contido. Indica o tipo de contido incluído no corpo da mensaxe.
  Exemplo: *Content-Type:application/sdp*

7.3 CORPO DA MENSAXE. SDP

Esta información non é analizado polo *proxys* intermedios da rede. SDP utiliza a seguinte nomenclatura:

campo=valor1 valor2

SDP non está pensado para traballar exclusivamente con SIP, polo que algúns campos non teñen sentido con SIP. Así mesmo, os campos deben ser usados seguindo unha orde:

- En primeiro lugar os relacionados co nivel de sesión, cuxa orde é: ***v, o, s, i, u, e, p, c, b, t, r, z, k, a***.
- En segundo lugar os relacionados co de nivel do medio, cuxa orde é: ***m, i, c, b, k, a***.

Moitos destes campos son opcionais, mentres que os campos ***v, o, s, t, m*** son obrigatorios.

O significado dos campos é o seguinte:

- ✓ v Versión del protocolo (obrigatorio). Exemplo: *v=0*
- ✓ o Identifica ao remitente (obrigatorio). Ten seis subcampos: nome do remitente, identificador da sesión, versión da sesión, tipo de rede, tipo de dirección e dirección.
  Exemplo: *o=lolo 45678 001 IN IP4 servidor.vopiprodeira.com*
- ✓ s Nome de sesión (obrigatorio, pero sen sentido en SIP) Exemplo: *s=*
- ✓ i Información da sesión (obrigatorio, pero sen sentido en SIP) *i=Motivo da chamada*
- ✓ u URI da descrición (sen sentido en SIP). Exemplo: *u=*
- ✓ e Dirección de correo. Exemplo: *e=pmartinez@iesrodeira.com*
- ✓ p Número de teléfono. Exemplo *p=986303933*
- ✓ c Información de conexión. Inclúe tres subcampos: tipo de rede, tipo de dirección e dirección de conexión; co mesmo significado que os equivalentes do campo o, pero relativos ao destinatario.
  Exemplo *c=IN IP4 servidor.electrovoz.com*
- ✓ b Ancho de banda en kbps. Exemplo: *b=3*
- ✓ z (sen sentido en SIP). Zona horaria utilizada.
  Exemplo: *z=1* para a zona GMT+1 (Madrid, París)
- ✓ k Clave de encriptación
- ✓ a Atributos
- ✓ t Tempo de sesión(Start e stop separadas por un espazo) (obrigatorio, pero sen sentido en SIP) *t=0 0* (Os valores 0 simboliza duración indefinida)
- ✓ r Tempo de repetición (sen sentido en SIP)
- ✓ m Información do protocolo de transporte (media) (obrigatorio). Ten catro subcampos: tipo de medio, porto destino, protocolo de transporte utilizado e formatos codecs aceptados. Por cada codecs inclúese unha liña deste tipo, por orde de preferencia.
  Exemplo: *m=audio 4444 rtp/avp 2*
  a=rtpmap 2 G726-32/8000

8 GARANTIR A CALIDADE DO SERVIZO

Existen dous mecanismos básicos para garantir a calidade do servizo nunha comunicación de voz en tempo real: os servizos integrados e os servizos diferenciados.

No caso dos servizos integrados, faise uso dun protocolo específico de sinalización que reserva recursos da rede para que se estableza un camiño extremo a extremo a través do cal vanse enviar todos os paquetes de datos. Todos os routers intermedios, comprométense a garantir que van ser capaces de garantir un fluxo mínimo de datos a través deles. Actualmente, existe un protocolo para implantar esta metodoloxía de traballo, estamos a falar do protocolo RSVP.

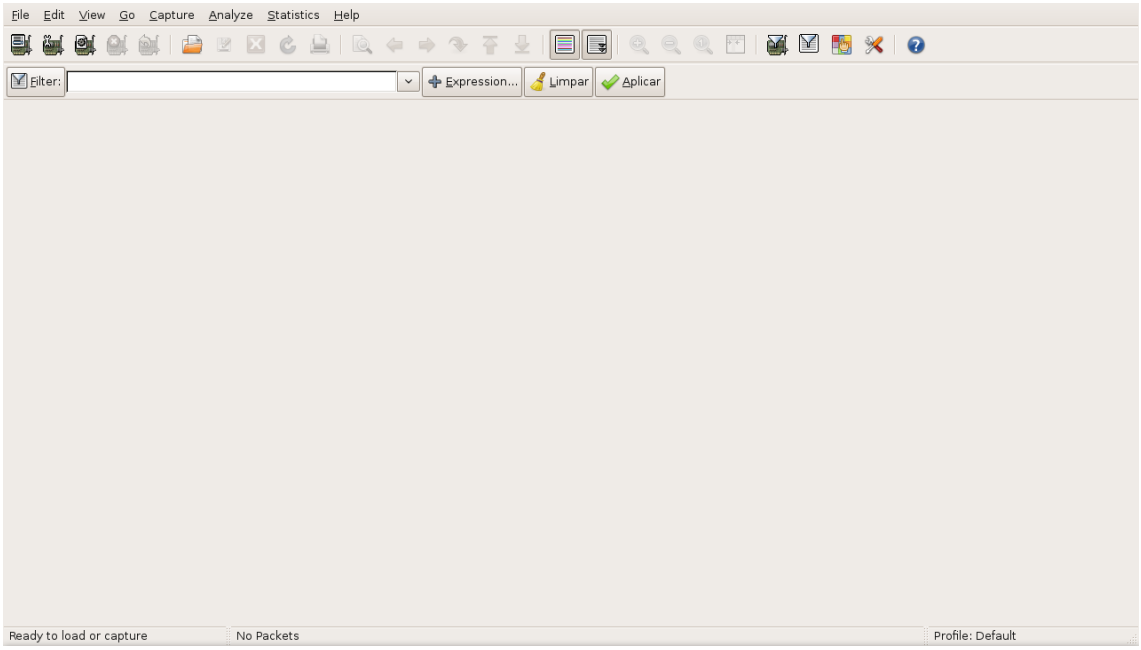
No caso dos servizos diferenciados, o sistema é máis sinxelo de implantar, aínda que non garante a calidade do servizo. Fai uso do campo ToS (tipo de servizo) do protocolo IP, de xeito que a maior número ToS, maior prioridade ten ese paquete, e polo tanto o router dálle maior prioridade.

9 USO DO SNIFFER WIRESHARK

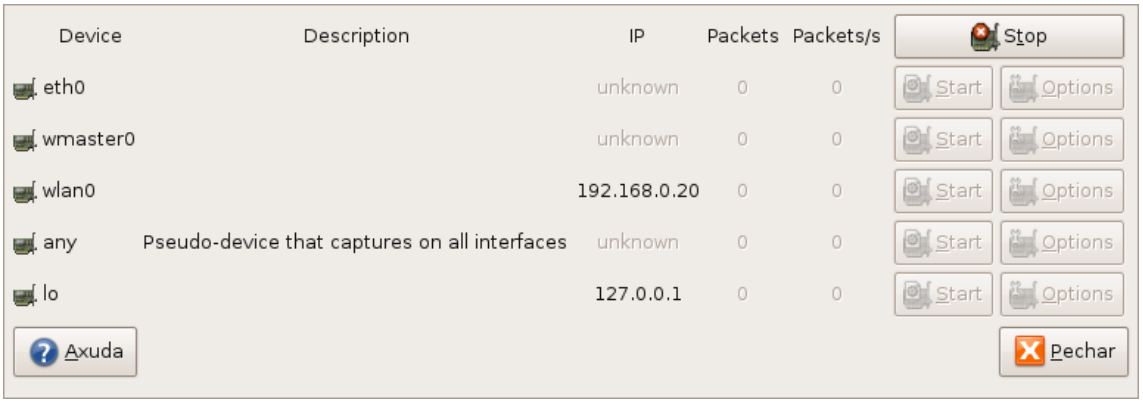
Un sniffer é un programa para monitorizar e analizar o trafico dunha rede de ordenadores. Captura todos o que se transmite pola rede para ser analizado polo usuario, non obstante, e tendo en conta a gran cantidade de información que se recibe é posible filtrar a información por protocolo, por dirección IP, etc.

De entre todo os sniffers cremos o máis axeitado Wireshark. Os motivos son os seguintes: trátase dunha aplicación gratuíta que está dispoñible tanto para Windows como para Linux. Ademais no caso de Windows, existe unha versión portable que pode ser usada por calquera sen necesidade de ser instalada. A páxina web oficial atópase en http://www.wireshark.org.

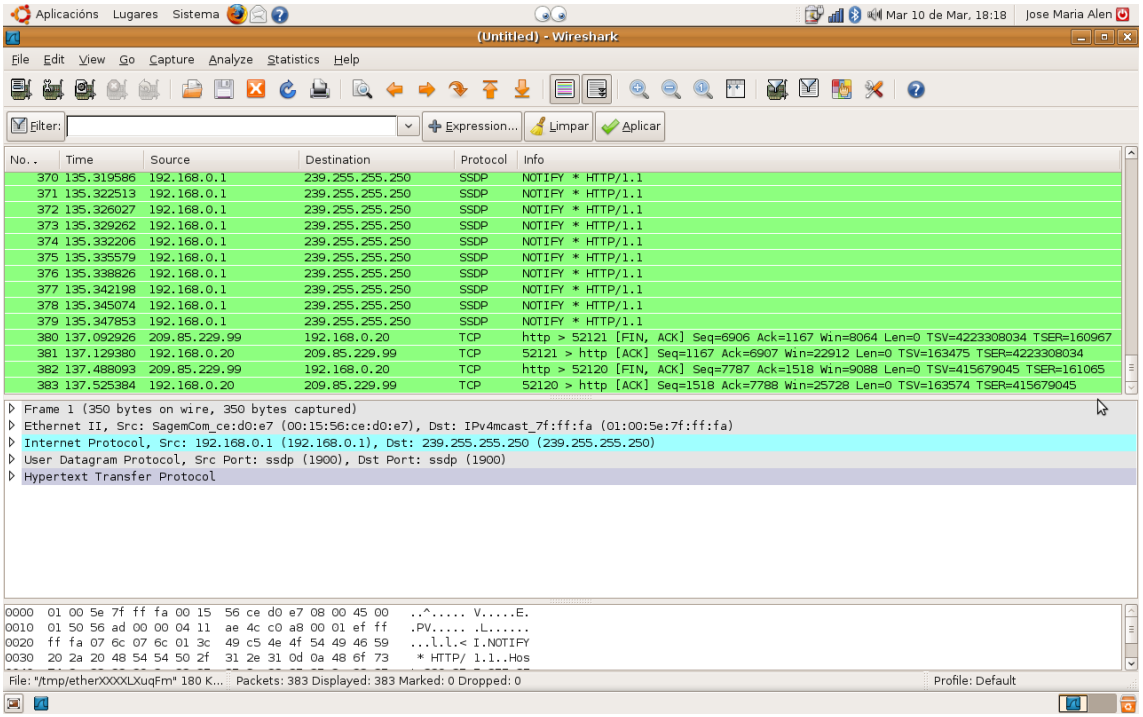
Ao arrancar o programa presenta o seguinte aspecto (en Linux hai que executar o programa como usuario *root*):



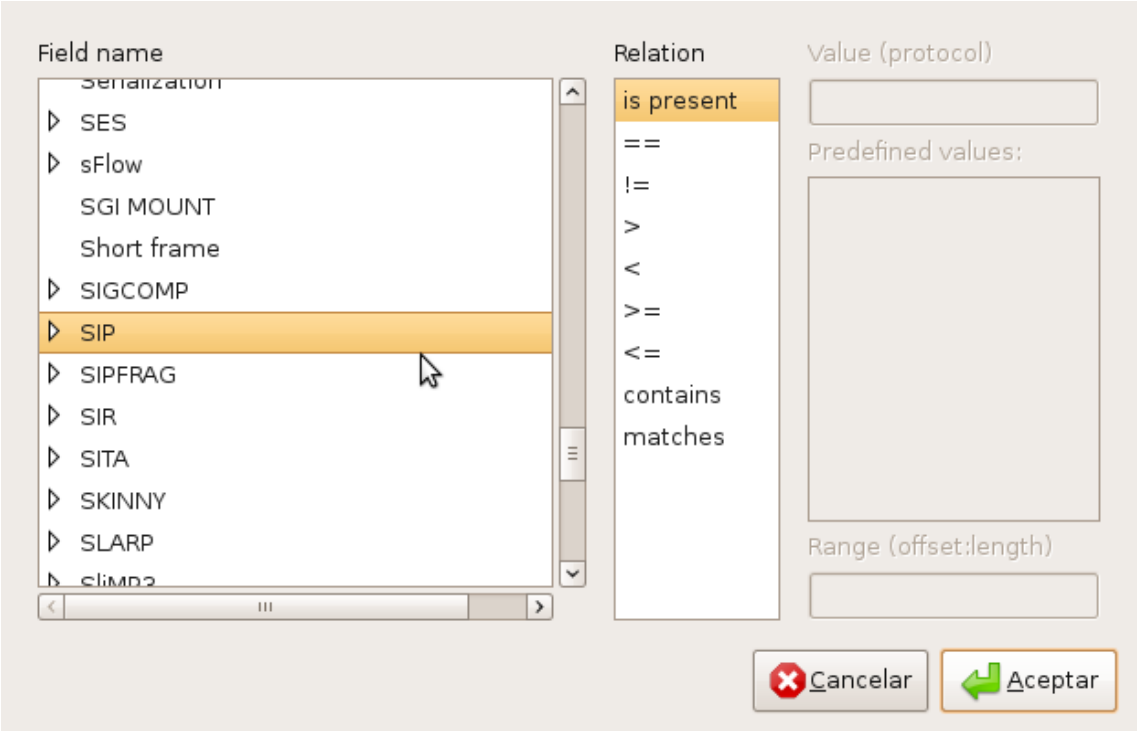
o primeiro que temos que facer é activar a tarxeta de rede a través da cal vanse capturar os paquetes, para isto imos á opción do menú Capture->Interfaces, e unha vez que esteamos nesta pantalla seleccionamos a tarxeta de rede correspondente:



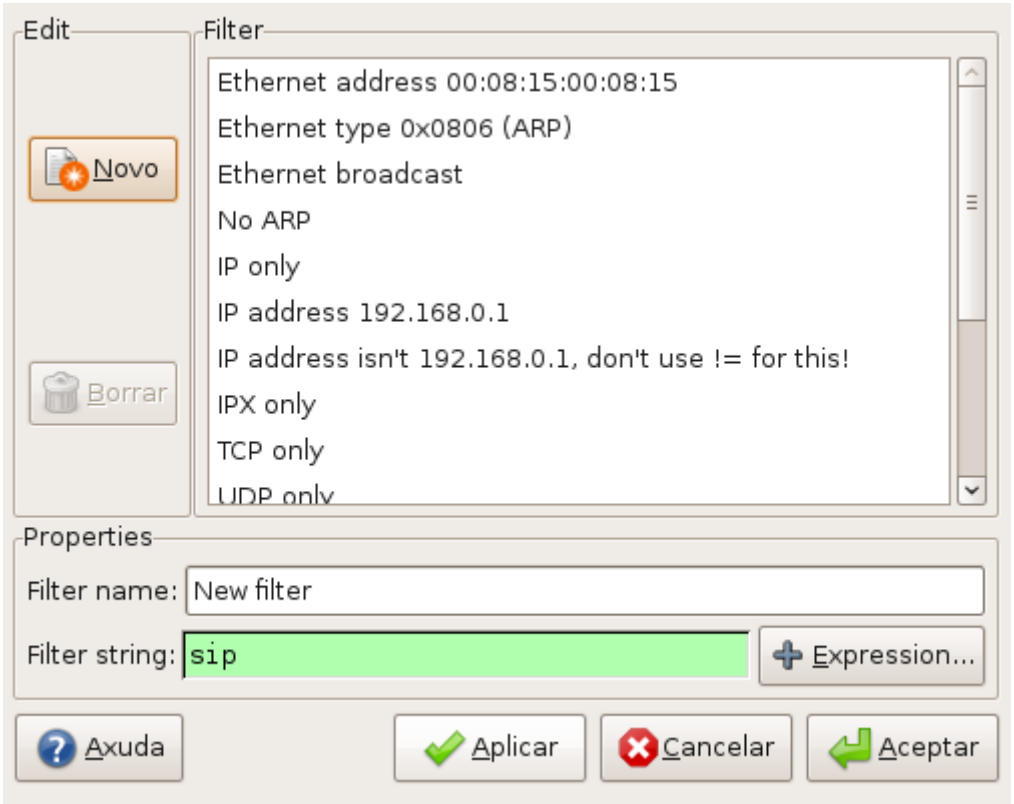
A partir deste momento o programa empeza a capturar paquetes que vai mostrando no programa:



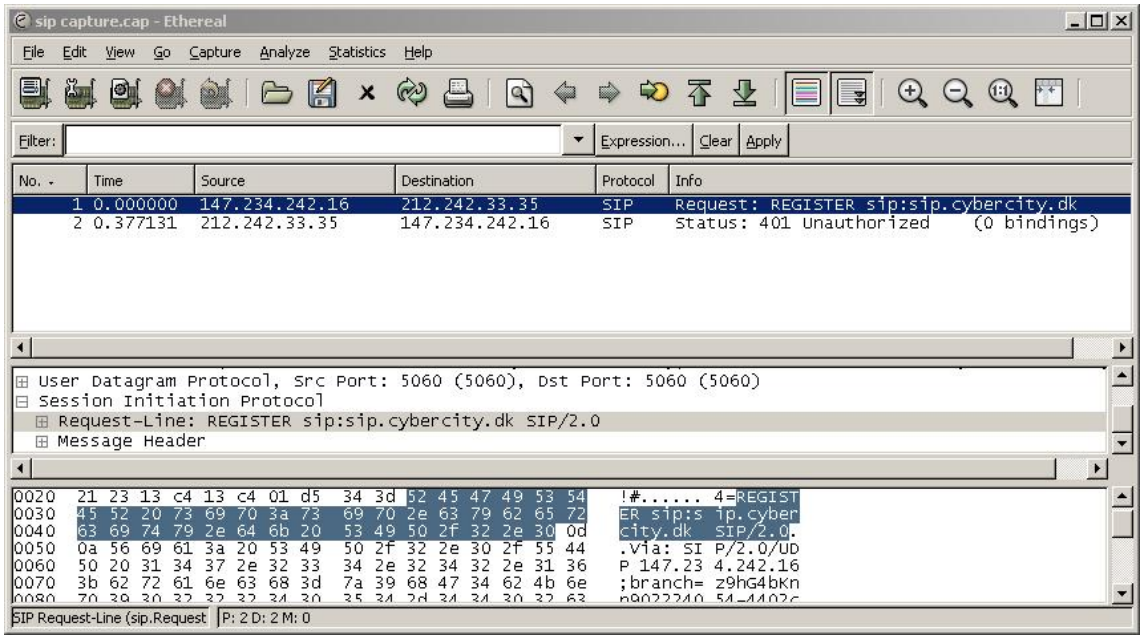
Como queira que a cantidade de información é enorme, e só estamos interesados na información pertencente ao protocolo SIP, teremos que filtrar a información mostrada. Existen diferentes formas de filtrar a información que recibimos. Se queremos filtrar a información recibida correspondente ao protocolo SIP, prememos sobre o botón “Expresións” e seleccionamos SIP.



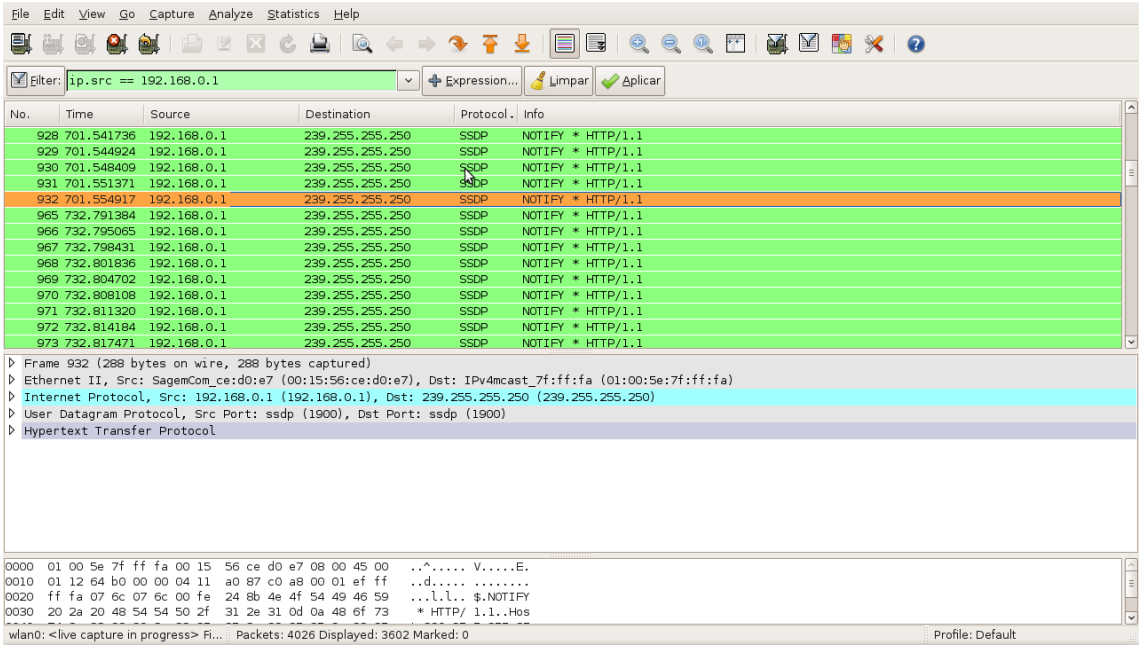
e despois prememos sobre o botón Aplicar:

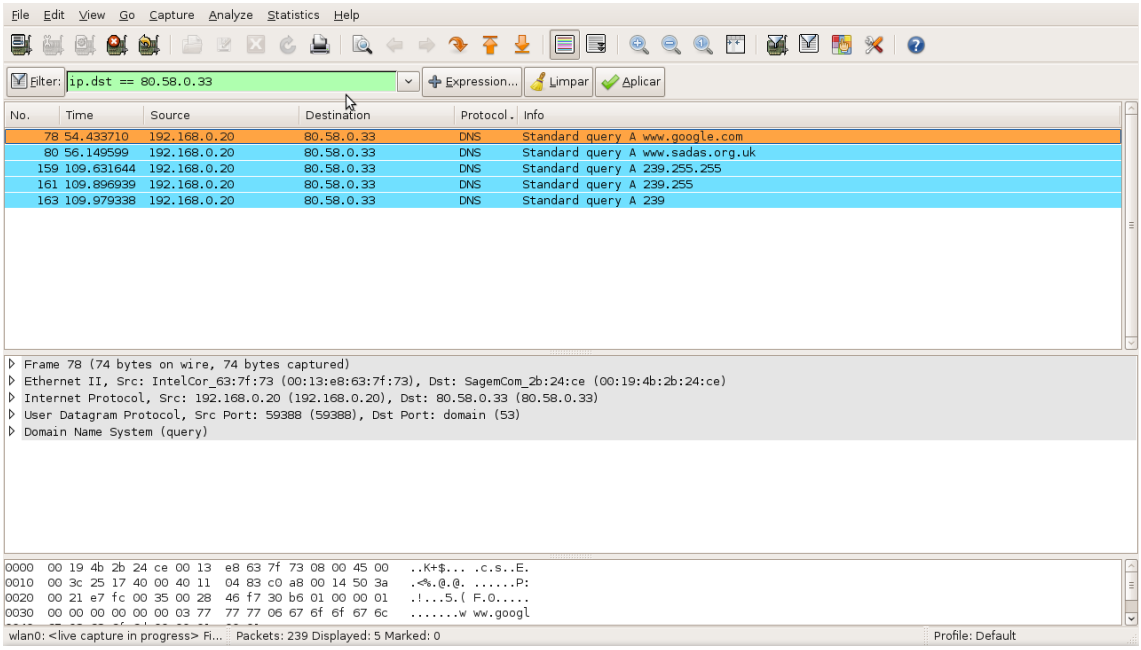


co cal a pantalla principal do programa quedaría da seguinte forma:



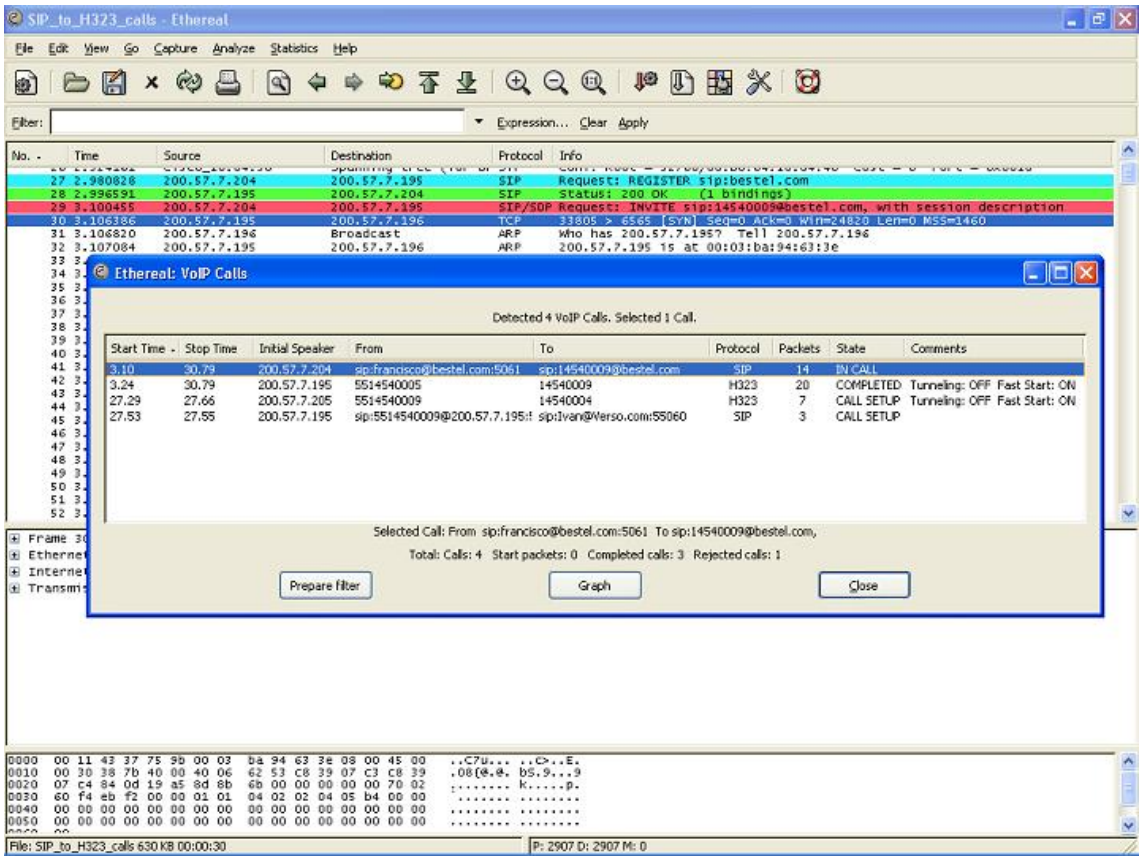
Por outra parte, tamén podemos filtrar por dirección IP orixe ou destino, para isto, poñémonos enriba da dirección IP orixe (ou destino) debaixo da columna correspondente, prememos no botón dereito do rato, e escollemos a opción do menú despregable “Applied as filter->Selected”.



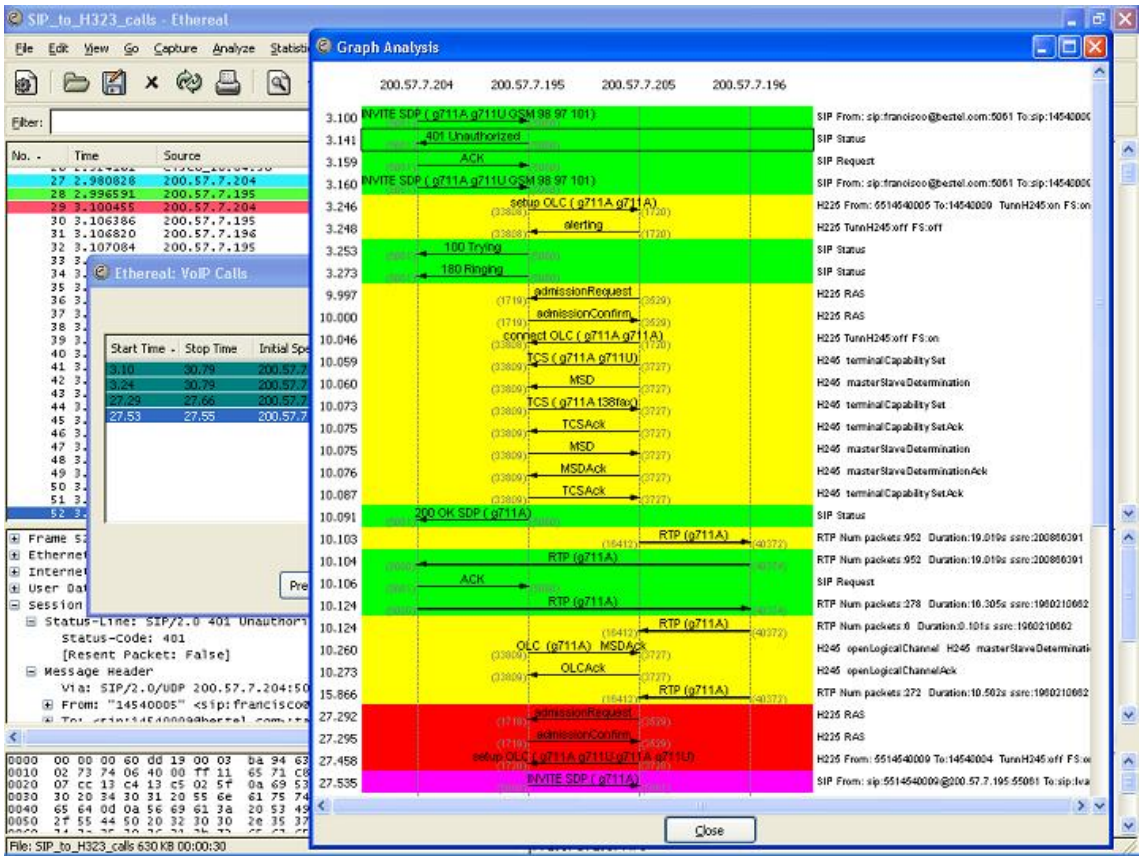


Ao mesmo tempo, o propio Wireshark nos permite traballar directamente con comunicacións VoIP. Así, se prememos na opción do menú Statistics observamos as seguintes opcións relativas a VoIP:

- SIP -> Create Stat ofrécenos unha estatística das comunicacións SIP
- VoIP Call devólvenos as estatísticas de chamadas VoIP detectadas
- RTP devólvenos os paquetes RTP detectados, os cales poden seleccionarse e despois analizarse.
  Esta última opción é moi interesante xa que, dunha parte podemos ver as comunicacións de voz establecidas:



e se seleccionamos unha delas e prememos en Graph, obtemos todo o relativo a esa comunicación:



Ao mesmo tempo, o Wireshark permite arquivar tramas capturas para ser despois analizadas. Para isto temos que usar a opción File->Save.

Finalmente, indicar que para examinar estes arquivos almacenados, é posible facelo sen entrar como usuario "*root*"; no caso de querer capturar tramas hai que entrar como "root". De feito, en Linux instálase dous ítems, "Wireshark" e "Wireshark (como root)". No caso de Windows, tamén hai que executalo como "Administrador" para poder capturar tramas; ademais nesta caso hai que instalar WinPcap. Doutra forma, só poderemos analizar tramas gravadas nun arquivo, para o cal non é necesario actuar como "Administrador".

10 PRÁCTICA 1: USO DO SNIFFER WIRESHARK

Obxectivos:
Adquirir experiencia no uso avanzado dunha ferramenta de análise de tráfico de rede. Identificar os diferentes tipos de protocolos, ver os campos incluídos no mesmo, filtrar os paquetes que nos interesan, etc.
Desta forma o alumno estará preparado para facer o mesmo despois cos protocolos usados por VoIP.

Coñecementos teóricos previos:
Requírese ter claros os diferentes niveis da torre de protocolos de Internet. Así mesmo, fai falla ter un coñecemento básico do uso da ferramenta software Wireshark.

Tempo da práctica:
O tempo estimado para a realización desta práctica é de 2 horas.

Posta a punto:

- Crea no teu directorio local (no teu "home") un subdirectorio chamado "**P1-LecturaTramas**". Nel gardaremos os arquivos que usemos nesta práctica.
- Copia nese cartafol o arquivo chamado "*p1.libcap"*. Este arquivo conten capturas de paquetes que se analizaran nesta práctica.

Recomendase que durante a realización dos exercicio, apuntes os vosos comentario se respostas ás preguntas plantexadas nun arquivo de texto. Este documento de texto serviravos para comparar os resultados obtidos coa captura da práctica nº1.

Exercicio 1: Lectura de paquetes e información asociada en Wireshark.

1) Abre o arquivo "**p1.libcap**" con Wireshark. ¿Qué protocolos interveñen nesta captura? Sendo así, ¿de qué crees que é a captura?

2) ¿Cantos paquetes foron capturados? ¿Canto tempo levou a captura? ¿Qué máquinas intercambian paquetes (véxase Statistics->Endpoint List->IPv4)?

3) Analiza un pouco máis a fondo o primeiro paquete. Observa se ten ou non datos. Se tes activada a función de coloreado de paquetes, poderás ver que Wireshark o identifica como un paquete HTTP, aínda que se te fixas ben o protocolo TCP non encapsula ningún contido de HTTP. ¿Por qué crees que aínda así Wireshark o colorea como un paquete HTTP? Busca o primeiro paquete que conten datos HTTP. Se non hai datos nos anteriores, ¿cál é o seu cometido?

4) Ordena os paquetes capturados por protocolos para poder analizar mellor aqueles que se identificaron como HTTP (non só no relativo á cor, senón que conteñen o protocolo en si). Observando a secuencia dos paquetes do protocolo HTTP pódese ver claramente as peticións HTTP que se fan ao servidor e as respostas que este proporciona. ¿Qué páxina é a que se pide no paquete 4?

5) Observa no espazo dedicado a mostrar o paquete capturado en hexadecimal, ao

fondo da ventá. Nese espazo, hai tres columnas. A primeira é un contador. ¿Canto suma en total para o paquete 4? ¿Qué relación ten esta suma co tamaño do paquete? A segunda columna mostra o contido en hexadecimal do paquete. Cando o pode interpretar, móstranse os caracteres correspondentes na terceira columna; en caso contrario, pódese ver un punto. ¿Qué ocorre cos caracteres "\r" e "\n"? ¿Qué significan eses caracteres? ¿Conséguense interpretar ou se mostran como un punto?

6) O seguinte paquete, o número 15, conten a resposta do servidor á nosa petición. Observa que baixo "Hypertext Transfer Protocol" danos o resultado da nosa petición (un "200 OK") xunto co resto de cabeceiras. Se investigamos o que hai dentro de "line-based text data: text/html", veremos que se trata dunha páxina HTML. ¿Cál é o tamaño do paquete número 15 (en bytes)? ¿Cómo pode ser que se enviara todo ese texto nun número tan pequeno de bytes? Pista: defragmentación. Investiga os paquetes afectados pola defragmentación, en especial, fixándote no seu tamaño, se levan contidos HTTP ou non e se hai algunha indicación de que o contido se encontra noutro paquete. ¿Quén realiza a defragmentación?

7) Identifica o momento no que chega a resposta da petición da páxina HTML. Observa que nos instantes seguintes se piden unha serie de arquivos adicionais ao servidor. ¿Qué son estes arquivos? ¿Ten éxito a petición de todos eles? ¿Cál cres que é o seguinte paquete que conten un GET provocado por unha acción do usuario (humano) que está utilizando o navegador?

8) Observa que o paquete 111 conten unha nova petición doutra páxina HTML. Así mesmo, o paquete 116 volve a pedir a folla de estilo (/docencia/style.css), aínda cando podemos ver que até o paquete 129 non obtemos a páxina HTML completa. ¿Cómo pode ser iso?

9) Volve a ordenar os paquetes segundo o seu número. Se agrupáramos por tamaños os paquetes, cantos grupos poderías identificar a *grosso modo* e con qué se corresponderían.

10) Aplica un filtro para que se vexan só os paquetes HTTP (p.ex. Analyze->Display Filter e logo seleccionando HTTP, que ten idéntico resultado a poñer "http" directamente no campo de Filter no menú). Edita o filtro para que só mostre os paquetes HTTP que ademais van á dirección IP 193.147.71.64. ¿Poderías encontrar a maneira de filtrar para que só se visualizaran os paquetes HTTP cuxo destino é a IP 193.147.71.64? ¿E de que se mostren os paquetes HTTP da IP 193.147.71.64 e todos os paquetes UPD á vez?

11) No menú "Statistics", selecciona "Protocol Hierarchy" para observar a xerarquía de protocolos. ¿Qué protocolo utilizouse máis: TCP ou UDP? Observa para qué utilizouse UDP. Fíxate tamén no tamaño dos datos intercambiados entre o cliente e o servidor web e cánto supón iso do total do intercambio de datos. ¿Qué porcentaxe do total de bytes de TCP corresponde a datos (texto e imaxes)? ¿E cál é a porcentaxe sobre o total de bytes de todos os paquetes? Unha vez que o teñas calculado, pecha esta ventá de estatísticas.

12) No menú de “Statistics”, elixe agora a entrada “IO Graphs”. Verás o ritmo de intercambio de paquetes nunha gráfica onde o eixo horizontal é o temporal, mentres que o vertical indica o número de paquetes. Cambia o intervalo do tick do eixo X a 0.10 segundos para analizalo con maior claridade. ¿Podes identificar as peticións web que fai o cliente? ¿Por qué? Así mesmo, hai outros picos, máis pequenos, que se poden identificar (por exemplo ás 3,5 segundos ou aos 10 segundos). ¿A qué se deben eses picos? ¿Teñen algunha relación coa visita das páxinas HTML?

13) Unha maneira sinxela de ver cánto do tráfico total é debido a HTTP é introducindo unha regra de filtrado no gráfico anterior. Para ilo, nun dos *Graph*s indica no campo de texto “http”.

14) En “Statistics” comproba qué nos ofrece a opción de “Conversations” e “Endpoints”, en particular na pestana de IPv4. Observa a relación de paquetes e bytes enviados dende o cliente e a do servidor. Lembra que na web é un servizo cliente-servidor típico, onde o que ofrece información é o servidor mentres que o cliente é só consumidor da mesma. Saca conclusións ao respecto.

Avaliación

- ✓ ¿Cál é a lonxitude total da parte correspondente a IP do paquete 373? (inclúe unidades)
- ✓ ¿Cál é a lonxitude dos datos de TCP do paquete 373? (inclúe unidades)
- ✓ ¿Cál é o tamaño do paquete 373? (inclúe unidades)
- ✓ ¿Cál é o número do próximo paquete coa páxina web de resposta do servidor web?
- ✓ ¿Cál é o porto fonte do paquete 373?
- ✓ ¿Cál é a dirección IP do servidor web do paquete 373?
- ✓ ¿É o paquete 373 un paquete petición?
- ✓ ¿Cómo se chama o proceso mediante o cal o contido da resposta aparece todo no último paquete?

11 PRÁCTICA 2: USO DO SNIFFER WIRESHARK CON VOIP

Obxectivos:
Adquirir experiencia no uso dunha ferramenta avanzada para análise de tráfico na rede. Identificar os distintos tipos de protocolos, ver as tramas incluídas no mesmo, filtrar os paquetes que son de interese, etc. Comparar unha secuencia de tramas multimedia cunha non multimedia vista na práctica 1.

Coñecementos teóricos previos:
Todos los necesarios para a práctica 1. Ademais, ter realizado a práctica 1.

Tempo da práctica:
O tempo estimado para a realización desta práctica é de 2 horas.

Posta a punto:

- Crea no teu directorio local (no teu "home") un subdirectorio chamado "**P2-LecturaTramas**". Nel gardaremos os arquivos que usemos nesta práctica.
- Copia nese cartafol o arquivo chamado "*p2.libcap"*. Este arquivo conten capturas de paquetes que se analizaran nesta práctica.

Recomendase que durante a realización dos exercicio, apuntes os vosos comentario se respostas ás preguntas plantexadas nun arquivo de texto. Este documento de texto serviravos para comparar os resultados obtidos coa captura da práctica nº1.

Exercicio: Seguindo unha conversa de VozIP con Wireshark.

1) Abre o arquivo "**p2.libcap**" con Wireshark. ¿Qué é o que capturouse? ¿Cantos paquetes foron capturados? ¿Canto tempo levou a captura? Compara estes valores cos obtidos da navegación por varias páxinas web da práctica 1.

2) A conversa da comezo no cuarto paquete; analízalo con maior detemento. Observa os protocolos utilizados e lembra en qué se diferencian dos utilizados na primeira práctica. Tamén poderás observar qué protocolo de iniciación de sesión estase utilizando e incluso a dirección á que se está chamando.

3) Ordena los paquetes capturados por protocolos para poder analizar mellor os tipos de protocolos utilizados durante a captura. ¿Qué protocolos teñen relación coa conversa sobre voz IP (o sinal non oso caso)? ¿Cales non teñen relación algunha? ¿Qué IPs están involucradas na conversa e qué portos se están utilizando? ¿Qué problemas podería plantexar a utilización destes portos (p.ex. nunha rede corporativa)?

4) Observa no espazo dedicado a mostrar en hexadecimal un paquete RTP capturado. ¿Qué información podes interpretar e cal non? En particular, ¿podelo facer co *payload* (a carga)? ¿Por qué non? ¿En qué formato está o *payload*? ¿De qué se trata entonces? Consulta a Wikipedia para obter detalles.

5) Observa os valores de secuencia de número e *timestamp*, así como o de

S*ynchronization Source Identifier* na conversa. ¿Poderías identificar cales corresponden a cada un dos participantes na conversa? ¿Cal é o intervalo de tempo entre *timestamps* de paquetes consecutivos? ¿Qué quere dicir iso?

6) Cambia a temporización dos paquetes para que se mostre o tempo dende o último paquete recibido (primeiro capturado e logo mostrado). ¿Qué relación ten este valor co campo *timestamp*? ¿Qué crees que pode pasar se o valor desvíase moito?

7) Observa o tamaño de cada paquete SIP y RTP. ¿Cómo son os paquetes RTP? ¿Cánto do tamaño total é polo contido e canto é debido ás cabeceiras dos distintos protocolos que encapsulan o contido? (Si fóra necesario, conta os bytes do hexadecimal para coñecer o número exacto).

8) No menú “Statistics”, selecciona “Protocol Hierarchy” para observar a xerarquía de protocolos. ¿Qué protocolo utilizouse mais: TCP ou UDP? Observa para qué utilizouse TCP e para qué UDP. Fíxate tamén no tamaño dos datos intercambiados entre o cliente e o servidor web e canto supón iso do total do intercambio de datos. ¿Saberías dicir qué porcentaxe do total de bytes de UDP corresponde a datos? ¿E cál é a porcentaxe sobre o total de bytes de todos os paquetes? Se comparamos estes resultados cos da práctica 1, ¿qué conclusións podemos sacar? Unha vez que os teñas calculado, pecha esta venta de estatísticas.

9) No menú de “Statistics”, elixe agora a entrada “IO Graphs”. Verás o ritmo de intercambio de paquetes nunha gráfica onde o eixo horizontal é o temporal, mentres que o vertical indica o número de paquetes. Cambia o intervalo do tick do eixo X a 0.10 segundos para analizalo con maior claridade. Unha rápida ollada fainos ver que se poden identificar nitidamente cándo empeza e cando termina a conversa. Así mesmo, hai outros picos, posteriores, case ao final da captura. Intenta indagar a qué se deben eses picos analizando as súas tramas.

10) Unha maneira sinxela de ver canto do tráfico total é debido a RTP é introducindo unha regra de filtrado no gráfico anterior. Para isto, en Graph 2 indica no campo de texto “rtp” e pulsa “Graph 2”.

11) Finalmente en “Statistics” comproba qué nos ofrece a opción de “Conversations” e “Endpoints”, en particular na pestana de IPv4. Observa a relación de paquetes e bytes enviados dende o cliente e a do servidor. ¿Qué diferenzas podemos atopar con respecto á práctica 1?

Avaliación

- ✓ ¿Qué é G.711?

  Escolle unha opción

  - a. encapsular
  - b. transmisión
  - c. compresión de audio
  - d. protocolo

- ✓ ¿Qué método se utiliza en SIP para que o cliente estableza unha conversa?
- ✓ ¿Cada canto tempo envía o emisor un novo paquete RTP? (inclúe unidades)
- ✓ ¿Cal é o tamaño dos paquetes RTP? (inclúe unidades)
- ✓ ¿Cánto ocupa a carga (payload) dos paquetes RTP? (inclúe unidades)
- ✓ O número de secuencia do primeiro paquete RTP enviado é 0.

  Verdadeiro Falso

- ✓ ¿Qué protocolo de transporte usa RTP nesta captura?
- ✓ ¿Cómo se chama á variación involuntaria (e se é moi grande, molesta) da periodicidade do sinal recibida?
- ✓ ¿Cánto ocupa a cabeceira RTP? (incluíndo unidades)
- ✓ ¿Cál é a dirección SIP do que efectúa a chamada?

12 REFERENCIAS

- Libro “VoIP. La telefonía de internet”. José Antonio Carballar. Editorial Thomson (Paraninfo) 2008.

- Páxina web oficial do programa gratuíto “Wireshark”. http://www.wireshark.org/ *Nesta páxina atoparemos documentación acerca do programa Wireshark: versións para descargar, manuais, etc.*

- Curso de “Información audiovisual en redes de ordenadores” impartido pola Universidade Rey Juan Carlos. *Deste curso foron extraídas as prácticas que se presentan nesta unidade didáctica.*

INTRODUCIÓN Á TELEFONÍA SOBRE IP

TITULO DA PRÁCTICA	TEMPO ESTIMADO
Configuración de Extensións en Trixbox	30min

OBXECTIVOS

- Configurar varias extensión no Trixbox para que poidan ser usadas polos softphones ou polos teléfonos IP.
- Instalar e probar softphones

MATERIAL MÍNIMO NECESARIO

- Un rede de area local con 2 o mais ordenadores con Virtual Box e acceso a internet.
- A existencia dentro de cada PC dunha máquina Virtual con Trixbox correndo.
- Auriculares e micrófono. Un xogo por cada PC.

REQUISITOS PREVIOS

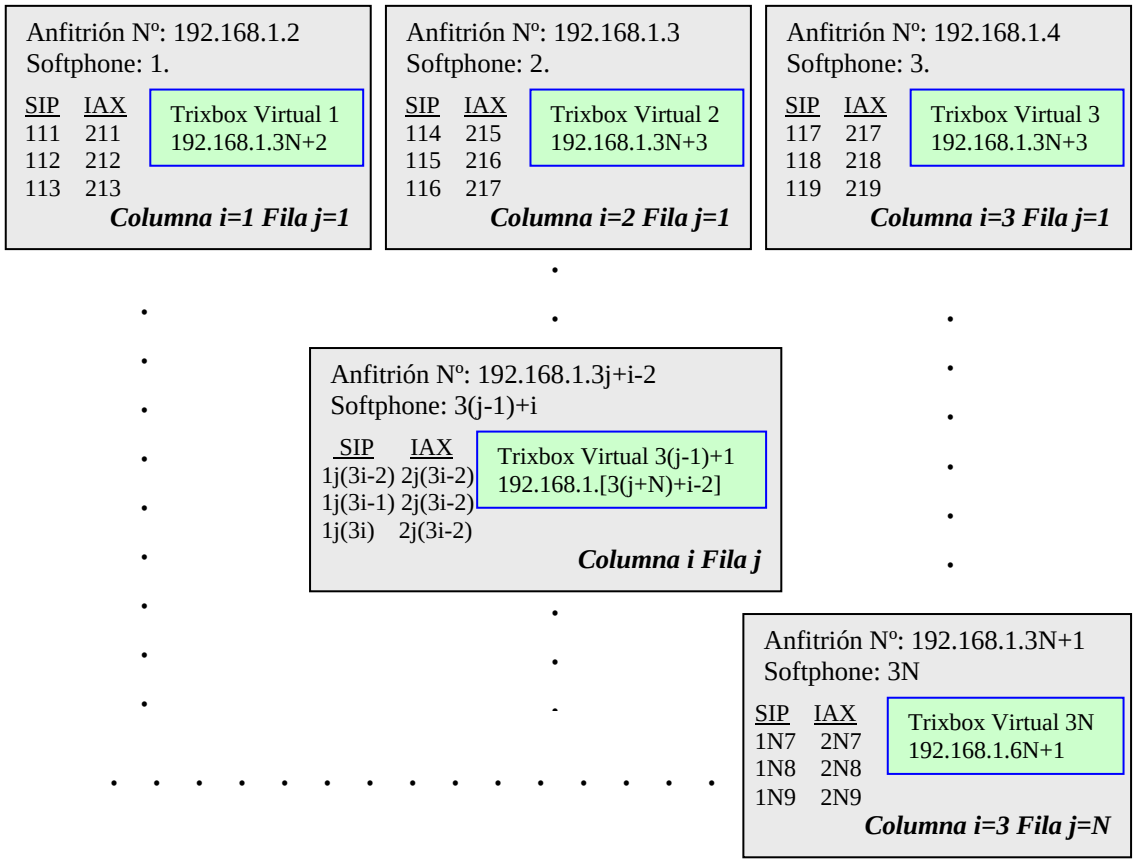
- Ter instalado e configurado o servidor Trixbox.
- Ter acceso a través do navegador de internet a cada Trixbox virtual.
- Todas as máquinas virtuais e anfitrións deben estar na mesma subrede.
- Todos os PCs deben ter a tarxeta de son operativa.
- Como máximo haberá 60 PCs para unha subrede IP de tipo C, para que existan suficientes IPs na mesma subrede para os anfitrións e as máquinas virtuais.

PROCEDEMENTO

1. Descargar e instalar o softphone zoiper na máquina anfitrión:
   http://www.zoiper.com/free.php
2. Entrar no servidor IP-PBX como administrador (usuario *maint*)
3. Dar de alta as extensións SIP e IAX dende FreePBX: 3 extensións SIP e 3 IAX:
   a. Unha SIP e unha IAX para o softphone da máquina anfitrion.
   b. As outras 2+2 extensións se reservarán para 2 dos softphones instalados noutros anfitrións que están na mesma rede de area local, para logo poder facer chamadas entre os 3.
4. Actualizar os cambios do Trixbox e deixalo correndo.
5. Configurar o softphone cos datos das extensións SIP e IAX. Haberá 3 contas para cada protocolo:
   a. 1+1 na IP-PBX da que somos anfitrión.
   b. As outras nas IP-PBX dos anfitrións “veciños” na rede.
   c. Xa que está é unha versión gratuíta de Zoiper, *ten a limitación de só soportar ata 2 contas*. Polo que teremos que seguir unha orde:
      i. Primeiro os 3 softphones definen as contas nunha mesma IP-PBX das tres que hai en rede.
      ii. Fanse as probas 6.a e 6.b
      iii. Volvese ao paso 5.c.i pero mudando de IP-PBX.
6. Faremos chamadas de xeito diferente:
   a. Probando coa conta de SIP no IP-PBX do que somos anfitrión cos outros 2 compañeiros.
   b. Probando coa conta de IAX no IP-PBX do que somos anfitrión cos outros 2 compañeiros.

O mesmo pero probando nas IP-PBX virtuais dos dous anfitrións veciños.

Na seguinte figura indícase como deberían asignarse as direccións IP entre os anfitrións e as máquinas virtuais, nunha rede de area local con mascara 255.255.255.0. Deste xeito contemplase ata 60 anfitrións, é dicir N=20 grupos de 3 PCs. Se fixeran falta mais anfitrións entón habería que proceder de xeito análogo pero cunha mascara do tipo 255.255.0.0
Tamén suxírese un xeito de repartir as extensións SIP e IAX en cada Trixbox Virtual[1]. Realmente non é necesario chegar a este reparto nos números de extensión, pero ao facelo así estaremos deixando aberta a posibilidade de poder facer chamadas entre todos os softphones da rede.



Así o procedemento sería:

1. Os anfitrións de primeira columna (i=1), configuran as extensións sip e iax correspondente sip: 1j1,1j2 e 1j3 iax:2j1,2j2 e 2j3.
2. Os anfitrións configuran os seus softphones coa conta sip e iax do anfitrión da súa primeira columna i=1, ip 192.168.3j-1 :
    a. Ao anfitrión da primeira columna (i=1) correspóndenlle as extensións sip 1j1 e iax 2j1
    b. Ao anfitrión da segunda columna a (i=2) correspóndenlle as extensións sip 1j2 e iax 2j3
    c. Ao anfitrión da terceira columna (i=3) correspóndenlle as extensións sip 1j3 e iax 2j3

[1] Se hai mais de 3x9=18 anfitrións na rede, as extensións dalgúns deles serán de catro cifras. Neste caso o mais recomendable sería empregar para as filas da 1 a 9, extensións do tipo x0jx

3. Agora fanse chamadas entre os 3 para probar os dous protocolos:
    a. O softphone da columna i=1 chama por sip ao da columna i=2
    b. O softphone da columna i=2 chama por iax ao da columna i=3
    c. O softphone da columna i=3 chama por iax ao da columna i=1
    d. O softphone da columna i=1 chama por iax ao da columna i=2,.....
4. Unha vez feitas as chamadas volveríamos ao primeiro paso e repetiriamos do 1 ao 3, pero nesta ocasión a Trixbox Virtual será a do anfitrión da segunda columna (i=2) e teremos que ter en conta que mudan as extensións e as ips segundo recóllese na figura da páxina anterior. Logo o da terceira (i=3)

Ao remate da experiencia todos os Trixbox virtuais deberán haber soportado chamadas entre os 3 participantes de cada fila tanto en iax como en sip.

Softphone: Zoiper

O Softphone que será utilizado é Zoiper, o cal permite os protocolos de comunicación SIP e IAX, soporte de STUN (moi útiles para contas SIP baixo NAT), cancelación de eco, entre moitas outras cousas mais.

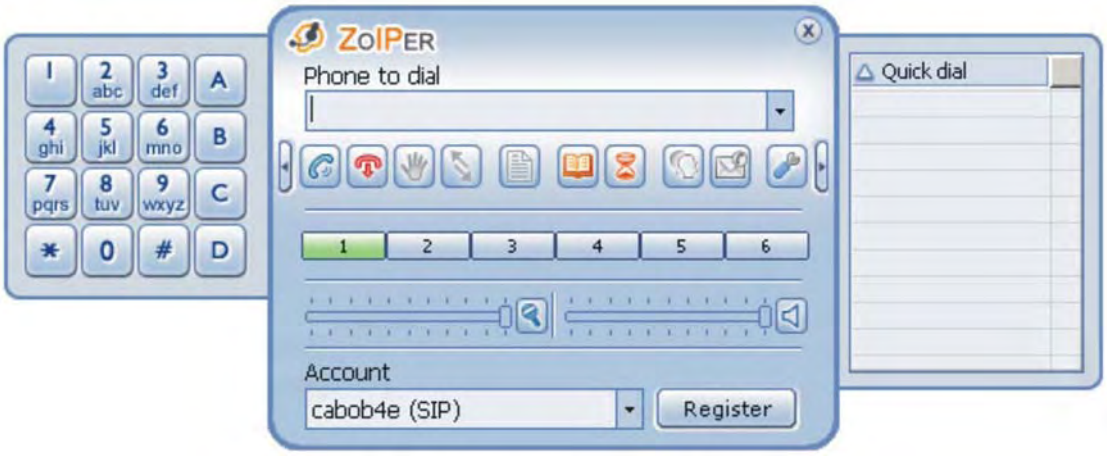


Figura 1: Softphone Zoiper

Extensións

Son cada unha das liñas que se atopasen conectadas a Trixbox. A continuación ensinásese paso a paso a configuración de extensións xa sexa con Softphone ou Teléfonos IP. É recomendable que antes de comezar a definir as extensións se faga unha lista coas extensións que necesitarían, así como definir a numeración de cada unha destas. Recoméndase evitar os números de extensión da Táboa 1, xa que están reservados para usos internos do sistema.

Extensión	Uso Reservado
200	Notificación de Parqueo
300 a 399	Marcación Rápida
666	Probas de Fax
70 a 79	Chamadas en Espera
700 a 799	Chamadas en Espera
7777	Simulación de Chamada Entrante

Táboa 1: Extensións Reservadas

Configuración de extensións en Trixbox

As extensións en Trixbox créanse da mesma xeito tanto para teléfonos IP como para Softphone. As extensións en Trixbox son configuradas en FreePBX, para lograr acceder a este lugar, no menú modo administrador de Trixbox seleccione asterisco e logo FreePBX, co cal debería aparecer a pantalla que posúe as opcións de administración de FreePBX.
Unha vez dentro da administración de FreePBX, pode cambiar o idioma a español na parte dereita da pantalla, logo realice os pasos mostrados nas follas posteriores, que permiten configurar unha nova extensión.
De acordo aos pasos mostrados nas figuras, débese realizar o seguinte.

1. Elixir a opción “configuración” de FreePBX.
2. Como se configurasen extensións, seleccione a opción “extensions” do menú da esquerda da pantalla.
3. Faga clic no enlace de “engadir extensión” (add extension). En caso de querer modificar os datos dunha extensión, elixa a extensión que desexa editar.
4. Seleccione o tipo de dispositivo a usar (SIP, IAX ou outro).
5. Prema o botón “Submit” (enviar) para comezar coa configuración da nova extensión.
6. Dado que se configurará unha extensión básica, é dicir, que inicialmente só se poidan realizar chamadas sen a activación de caixa de correos de voz, gravación de chamadas, entre outras cousas; os datos que se introducirán serían o nome, número e contrasinal da extensión.
7. Logo prema no botón “Submit” (enviar) para crear e gardar a extensión.
8. Finalmente prema a barra de cor vermella (Apply Configuration Changes) que aparece na parte superior da páxina que permite aplicar os cambios definitivamente.

Cos pasos anteriores, tense creada e configurada unha extensión no servidor, agora débese configurar no dispositivo desexado, softphone ou teléfono IP.

Configuración de extensións no Softphone Zoiper

A extensión é configurada nas opcións do softphone, onde se pode configurar unha conta SIP ou IAX. A continuación móstranse os pasos para configurar unha extensión básica SIP ou IAX en Zoiper.

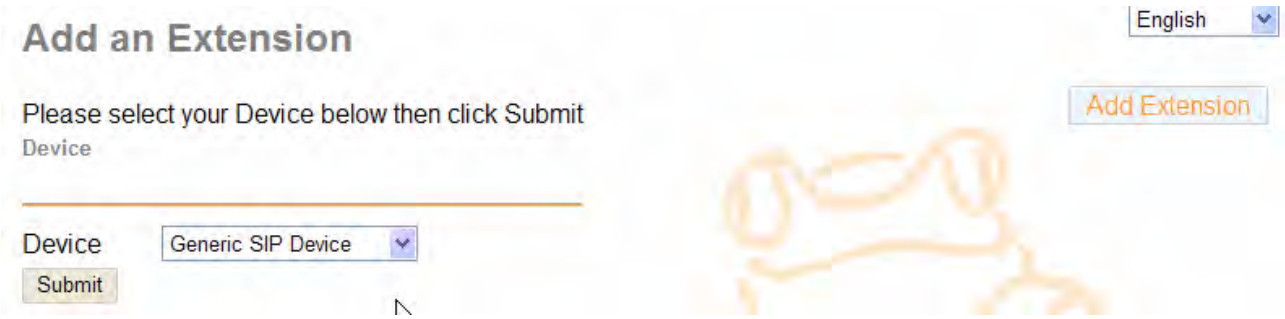
1. En Server Hostname/IP (Domain en SIP) introduza a IP do servidor Trixbox.
2. En Username é o número da extensión con que foi creada en Trixbox.
3. En Password débese introducir o contrasinal da extensión.
4. En Caller ID Name escríbese o nome da extensión, o cal sería mostrado no visor de quen recibe a chamada.

O Caller ID Number non será cuberto, xa que polo momento non cumpre maior funcionalidade, este será usado e explicado mais adiante.
Finalmente aplique os cambios e rexistre a conta recen creada no Softphone na pantalla principal do teléfono. Con isto, o teléfono xa se atopa rexistrado e listo para o seu funcionamento.

Crear Extensiones

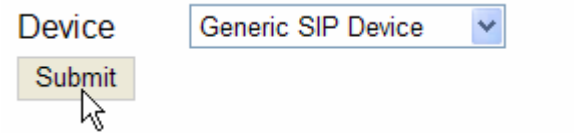
Por defecto, FreePBX nos permite llamadas a extensiones internas de acuerdo a nuestro plan de numeración.
El plan de numerción NO se define en ningún sitio, sino que se crean reglas y se asignan rutas de marcado, de forma que todo, en conjunto, constituye un DIAL PLAN.

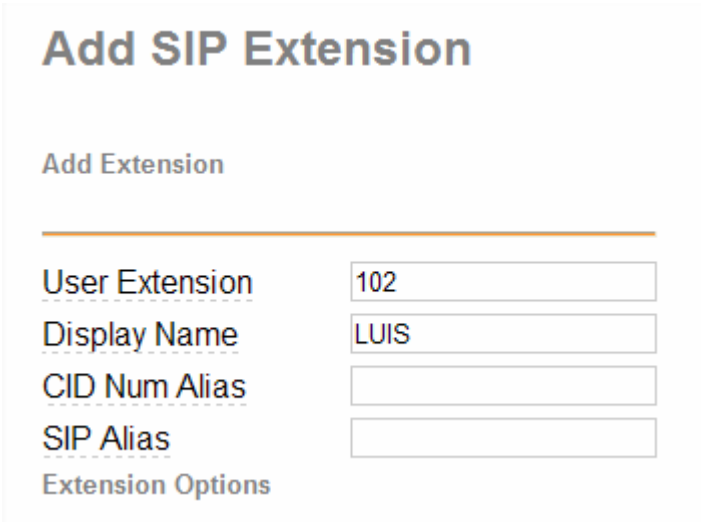
Extensiones para SIP



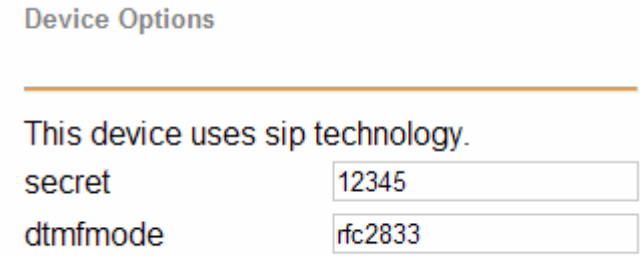
Seleccionamos tipo SIP (Generis SIP Device)

Cubrimos Número de Extensión y Nombre, en nuestro caso seleccionamos extensiones de 3 dígitos.

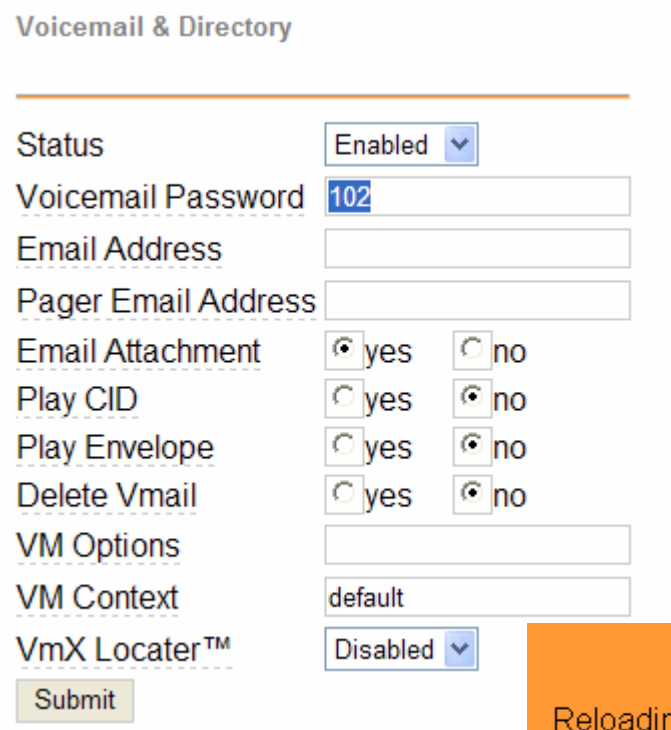




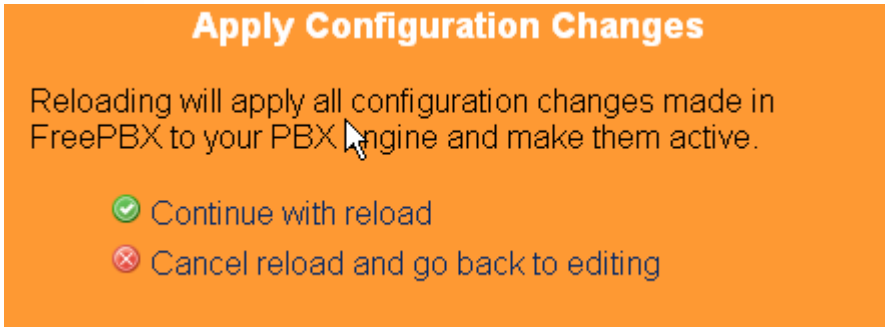
Tambien el SECRET (que es la password)



Habilitamos Buzón de voz con redirección al mail (si lo deseamos). Cubrimos password del buzón de voz.

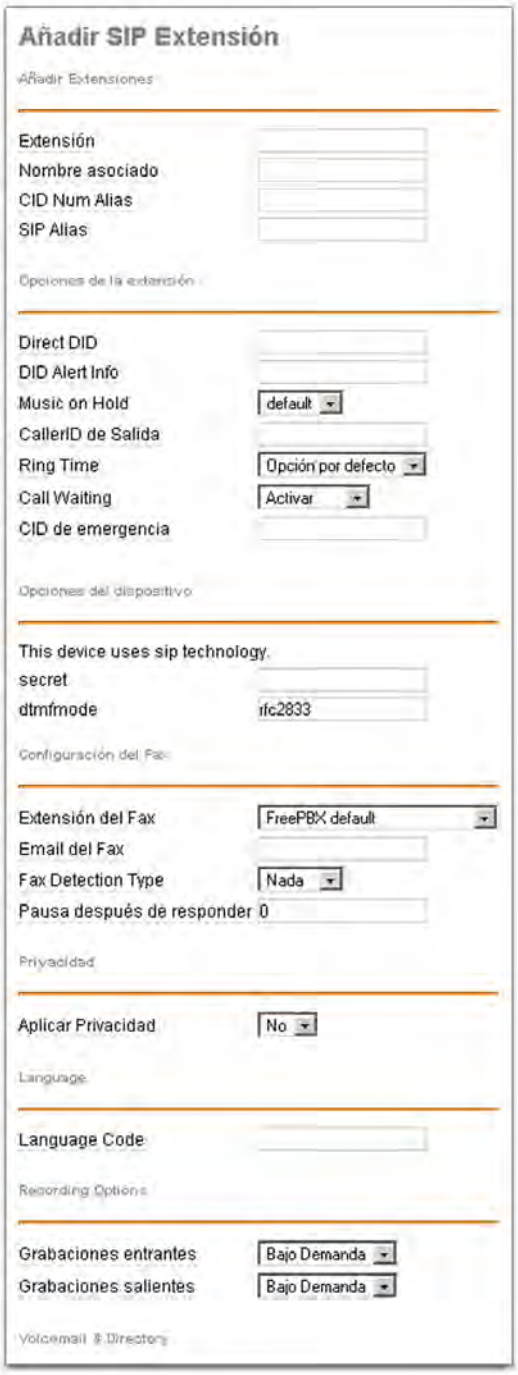


Guardamos los cambios realizados



Extensión.

El valor ingresado en este campo debe ser único. Este es el número que puede ser marcado desde otra extensión cualquiera, ó directamente desde la recepcionista digital (IVR), si está activado este módulo. Este número puede tener cualquier largo, pero convencionalmente es utilizado un número de tres ó cuatro dígitos.



Parámetros a configurar en el caso de haber seleccionado un dispositivo SIP.

Nombre asociado.

Esto es el texto del identificador de llamada que se presenta a los que son llamados. Debería ser cualquier línea ISO8859-1(ó ASCII). Pero aceptará UTF-8. Nótese que algunos teléfonos tendrán problemas con nombres de varios bytes, por ejemplo los UTF-8, posiblemente colapsando los teléfonos. Si utiliza nombres de varios bytes, experimente cautelosamente.

CID Num Alias.

El valor colocado en este campo, sobrescribe el ya configurado en "nombre asociado" solo cuando se realizan llamadas internas. Por ejemplo, puede colocar como alias el número de la cola a la que pertenece el interno y así, las llamadas devueltas, serán hacia la cola y no directamente al interno.

SIP Alias

Direct DID

Aquí se coloca el número directo (PSTN o VoIP) tal como es recibido por la central, al que se quiere asociar con este interno. Por ejemplo, si tenemos una línea con el número 11-5555-1234 y las llamadas a este que suenen directamente en este interno, debemos colocar 1155551234.

DID Alert info.

Esta función es para configurar los ringtones de los teléfonos IP. No tiene efecto si no se configura un número en **Direct DID**.

Music on Hold.

Esta opción especifica que categoría de música usará este interno cuando necesite enviar música en espera a la parte llamante. Las categorías se configuran previamente en el módulo "Music on Hold"

Outbound CID

Debe ingresarse un valor con el formato "Nombre" <#########>. Este valor sobrescribirá el Caller ID de la línea cuando este interno realice una llamada al exterior.

Ring Time

Aquí se configuran los **segundos** de espera antes de derivar al llamante al voicemail. La opción ***default*** toma el valor ingresado en el módulo "***Configuraciones Generales***".

Call Waiting

Permite activar o desactivar la función de llamada en espera para el interno que se está creando. Esta función puede activarse o desactivarse posteriormente utilizando *70 o *71.

CID de emergencia.

El valor que se ingrese en este campo, sobrescribirá todos los ajustes anteriores referentes a la identificación de llamada cuando se utilice una ruta saliente marcada como "***Llamadas de emergencia***".

Contraseña (secret).

Esta es la contraseña (password) utilizada por el dispositivo telefónico para autentificarse al servidor de Asterisk cuando se crea una extensión SIP o IAX2. Esto es generalmente configurado por el administrador antes de dar el teléfono al usuario, y comúnmente no conocida por el usuario. Si el usuario esta utilizando un softphone, entonces necesita saber esta contraseña para configurar el programa.

Extensión del Fax

Puede seleccionarse el interno donde se harán llegar las llamadas originadas desde un Fax. Si se selecciona FreePBX default, se utilizarán los valores indicados en el módulo "Configuraciones Generales". Si en cambio, es seleccionado "sistema", los faxes recibidos serán enviados por mail

Email del Fax

En este campo se indica el mail de destino de los faxes recibidos para el interno que se está creando. Esta fnción tiene efecto solo si el parámetro "***Extensión del Fax***" es configurado a "***sistema***".

Fax detection type.

Seleccionar el modo en que **Asterisk** intenta determinar si el origen de la llamda es de un equipo de fax. Colocar **Zaptel** si el interno está vinculado a una línea directa de protocolo Zap. Si el interno está configurado para contestar llamadas originadas de troncales VoIP (SIP o IAX2) colocar **NVFax**.

Pausa después de responder

Ajustar el tiempo en segundo que desea reproducir el sonido de señal de fax a la parte llamante.

Aplicar privacidad.

Si la persona que llama no tiene identificador de llamada (número privado), se le pedirá que ingrese los 10 números de su número telefónico.

Language Code.

Aquí puede especificar en idioma de los paquetes de sonido que Asterisk utilizará para este interno. Por ejemplo, en para ingles, es para español, it para italiano, etc.

Grabación entrante.

Opciones para grabar las llamadas recibidas en la extensión. Existen tres opciones:

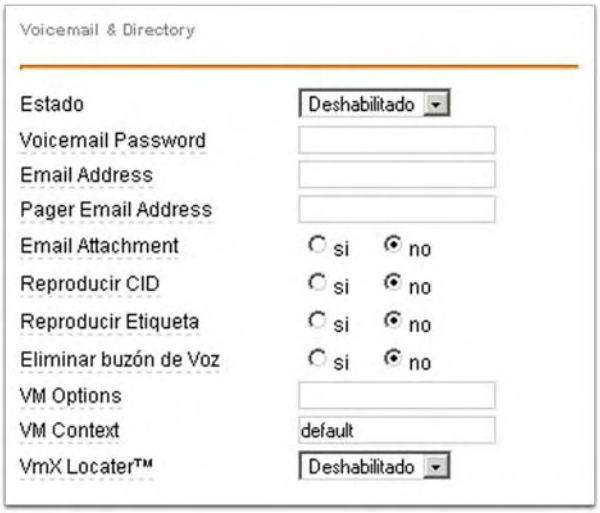
- Siempre
- Nunca
- A pedido (el usuario puede presionar '*'1 para activarlo durante cualquier llamada)

Grabación saliente.

Funciona de la misma manera que el anterior, pero con llamadas salientes.

Buzón de voz (voicemail) y directorio.

Al seleccionar "habilitado", se activa la casilla de mensajes para el interno que se está creado. Si la casilla ya estaba habilitada y se la deshabilita, se borrarán todos los valores de configuración ingresados.



Configuración del voicemail.

Contraseña del correo de voz (voicemail password).

Esta es la contraseña para acceder al sistema de correo de voz (voicemail).
Puede ser cambiada por el usuario cuando ingresa en su buzón de voz marcando *98. Para hacer esto, luego de ingresar, se debe presionar cero y luego cinco.

Dirección de e-mail.

Las direcciones a las que el correo de voz (voicemail) enviará las notificaciones cuando haya un nuevo correo almacenado.

Dirección de e-mail de pager.

Esta es la dirección de e-mail a la que se enviará una pequeña notificación al momento de registrarse un nuevo mensaje en la casilla (voicemail), adaptable para un servicio de e-mail a pager.

Reproducir CID.

Reproduce el número que llamó antes de reproducir el mensaje, e inmediatamente después anuncia la fecha y hora (envelope) en la que fue grabado el mensaje.

Reproducir fecha y hora (envelope).

Esta opción controla si el sistema reproducirá ó no la fecha y hora (envelope) del mensaje antes de reproducir el mensaje. Esta configuración no tiene efecto sobre la operación de la opción de envelope en el menú "advance" del buzón de voz (voicemail).

Borrar buzón de voz (voicemail).

Si está seleccionado en "yes" el mensaje será borrado de la casilla de correo de voz (voicemail) después de que se haya enviado por e-mail. Esta función provee la funcionalidad que le permite al usuario recibir su correo de voz (voicemail) únicamente por e-mail, en lugar de recuperar el mensaje desde la web ó la extensión.

Precaución: Usted debe tener el la configuración de e-mail en "yes" si no quiere que el sistema le envíe una notificación que diga "Usted tiene un correo de voz" e inmediatamente después borre el buzón de voz. Asegúrese de haber revisado voicemail-to-e-mail antes de activar este comando.

Objetivo

Este documento le guiará en el proceso de Obtención, Instalación y Configuración del Softphone[1] ZoIPer (Cliente IAX2[2]), con el cual podrá hacer y recibir llamadas dentro de su Red VoIP.

Obtención del programa e instalación

La primera tarea es obtener ZoIPer desde Internet en la siguiente página:

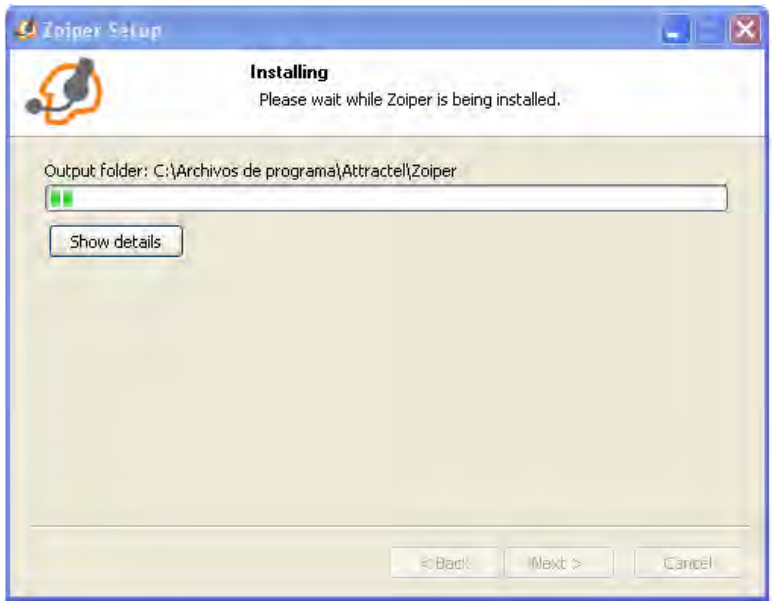
http://www.zoiper.com/

Desde aquí podemos descargar una versión gratuita del Sofphone. Existen versiones para diferentes plataformas; por ejemplo: **MacOS-X**, **Linux** y **Windows**. Descargué la más adecuada para su sistema.

Al momento de escribir este documento la versión mas reciente es la 2.18.

Ahora sólo basta con ejecutar el instalador que acabamos de descargar y seguir las instrucciones en pantalla.

[1] -Un Softphone es un programa que funciona como un teléfono convencional desde su propia PC.
[2] Un cliente IAX2 es un programa o dispositivo que actúa como terminal (aparato tel efónico) dentro se su Red VoIP y que utiliza el protocolo IAX v.2.



Configuración rápida de una cuenta IAX2[3]

Recuerde que...

Antes de iniciar este proceso es importante que tenga a la mano la siguiente información:

- Número de extensión;
- Nombre de Usuario para autenticarse;
- Contraseña y;
- Dirección IP o Nombre de Dominio del Servidor VoIP (En este caso, del servidor PBX-UNAM).

Esta información le deberá ser facilitada por el administrador de la Red VoIP.

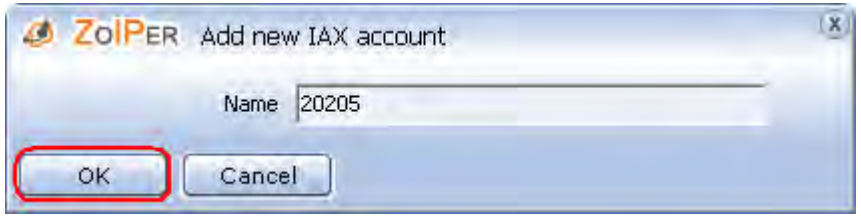
[3] Una cuenta IAX2 es nuestro identificador de abonado al sistema VoIP.

1...

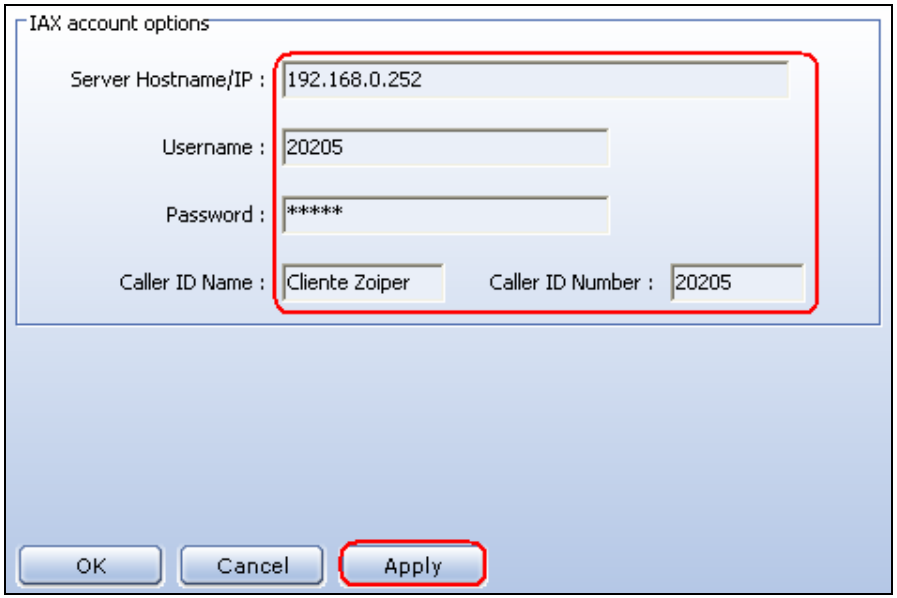
Debemos entrar en la sección "**Options**" mediante el botón correspondiente y aparecerá la siguiente ventana:

2...

ZoIPer puede funcionar tanto como cliente SIP o IAX2, el procedimiento es el mismo en ambos casos. Clic en la sección (Add a new IAX account). Ingrese un nombre para esta cuenta y de clic sobre el botón **Ok**.



3...

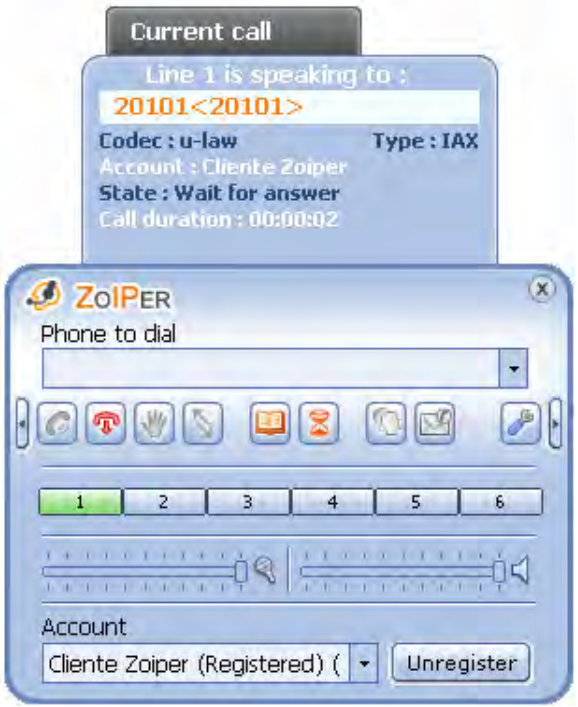


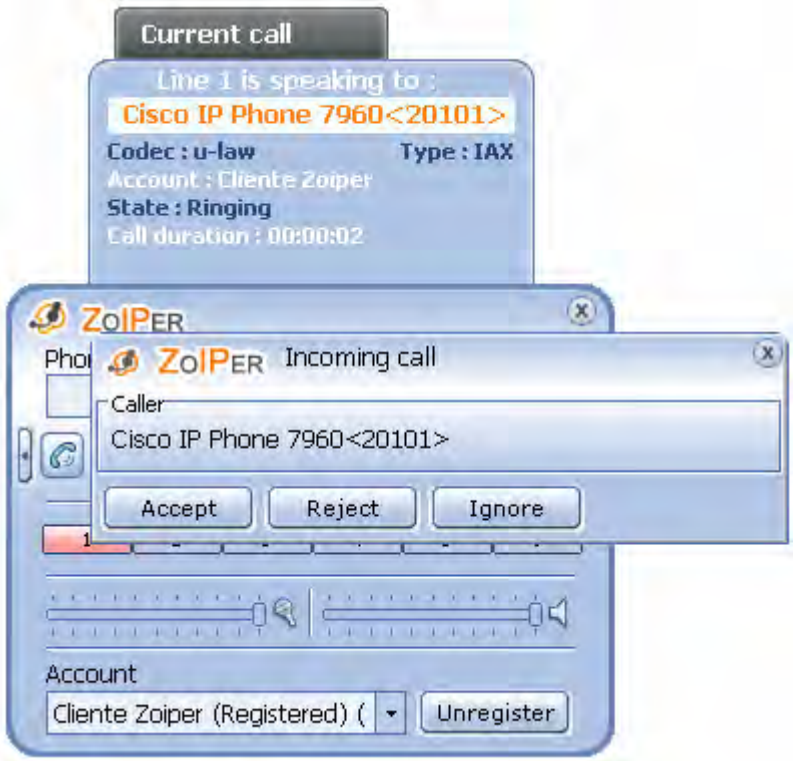
En esta ventana escribimos los datos de nuestra cuenta IAX2. **Server Hostname/IP** es la Dirección IP o nombre del servidor IAX2, **User name** (nombre de usuario), **Password** (Contraseña), nombre para desplegar en pantalla (**Caller ID Name**) y el (**Caller ID Number**) ó número a desplegar en pantalla junto con el Caller ID. ¡Listo! Esta es la configuración básica. No olvide aplicar los cambios hechos. Con esto seremos capaces de hacer y recibir llamadas.

Ahora debemos dar click en botón **Register** (Registrar) para activar nuestra cuenta. El estado de la cuenta debe cambiar a Registrado (**Registered**).

Si no aparece esto, debe verificar que la información introducida haya sido la correcta.

Ahora debe ser capaz de hacer y recibir llamadas.





INTRODUCIÓN A ToIP

TITULO DA PRÁCTICA	TEMPO ESTIMADO
Instalación do software de centraliña PBX de telefonía IP: Trixbox	100 minutos

OBXECTIVOS

- Instalar e configurar unha IP-PBX con Trixbox CE.
- O sistema resultante debe ser seguro e robusto.

MATERIAL MÍNIMO NECESARIO

- Aula con ordenadores en rede local, preferiblemente con IP estática.
- Un ordenador especifico para ser configurado como servidor trixbox con 1 lector de CD, 2 Tarxetas de rede e 7GB de disco duro
- Tarxetas de telefonía (preferiblemente PCI Express).
- Router de acceso a internet con NAPT (NAT con port forwarding)

REQUISITOS PREVIOS

- Ter o ordenador totalmente dedicado para funcionar como centraliña.
- Ranuras de expansión PCI (PCI Express) valeiras para as tarxetas de telefonía.
- No ordenador os buses da tarxeta gráfica e de rede deberían ser distintos do das tarxetas de telefonía para mellorar o rendemento destas.
- CD de instalación coa derradeira versión de Trixbox CE.
- Acceso á configuración do NAT do router de acceso a internet.

PROCEDEMENTO

1. Instalamos o trixbox CE, derradeira versión partindo do CD de instalación.
2. Deixar que o CD instale[1] e particione o disco duro (coa rede desconectada). Durante a instalación hai que decidir a password do root.
3. Hai que esperar a que termine e non reiniciar o equipo. Unha vez rematado, reiniciamos e facemos login como root no centOS. Configuramos a dirección de rede e os parámetros para acceder a internet. Para iso entrando no CentOS como root e editaremos os arquivos[2] *ifcfg-eth0* e *ifcfg-eth1* no cartafol */etc/sysconfig/network-scripts/*. É preferible que ambas sexan de IP estática, e só unha delas pode ter definida a porta de enlace. Por exemplo para a tarxeta eth0 habería que facer *nano /etc/sysconfig/network-scripts/ ifcfg-eth0*
   e escribir as seguintes liñas:
   DEVICE=eth0 (a interface a substituir)
   ONBOOT=yes (para arrancar a tarxeta en cada reinicio)
   IPADDR=192.168.1.99 (a dirección IP estática na LAN do router)
   NETMASK=255.255.255.0 (a mascara da rede do router de acceso)
   GATEWAY=192.168.1.1 (a porta de enlace, ou IP do router de acceso)
   TYPE=Ethernet
   A tarxeta *eth1* editaríase do mesmo xeito coa IP da outra LAN á que estivera conectada a outra tarxeta de rede pero omitindo o parámetro GATEWAY.
   Agora definimos as direccións IP dos servidores DNS, editando *resolv.conf*
   nano /etc/resolv.conf Unha vez aberto escribimos:
   nameserver 80.58.61.254
   nameserver 62.36.225.150
4. Actualizamos o sistema operativo con *yum –y update* dende centOS.
5. Comprobamos que as tarxetas telefónicas están ben instaladas. No caso de ter

[1] Vexa o apartado "Instalar Trixbox nunha Placa Base Asus M2N-MX-SE"
[2] Ou os que corresponda cada ifcfg-ethx corresponde a unha tarxeta de rede distinta do equipo

unha tarxeta RDS I de Acceso Básico, é necesario a instalación dos drivers.

6. Comprobamos que podemos acceder ao GUI de Trixbox co navegador web dende outros ordenadores na mesma LAN *http://192.168.1.99*
7. Pulsamos onde pon *switch* para entrar no panel de administración. O nome de usuario é *maint* e o contrasinal é *password* . Veremos o estado de Trixbox.
8. Actualizamos Trixbox dende o GUI. Non é necesario rexistrarse como usuario de Trixbox. Para iso vamos primeiro ao menú *Settings -> Repositories* Eliximos: *trixbox Stable, trixbox Addons* e *CentOS Base Packages* e prememos no botón que pon *Submit*
9. Vamos agora ao menú *Packages* Unha vez cargue a lista de actualizacións dispoñibles para Trixbox, seleccionaremos todas as que non estean xa instaladas e as que poñan *Upgrade*.
10. Rematada a actualización, iremos ao menú *PBX->PBX Settings* , para entrar no FreePBX. O nome de usuario é *admin* e o contrasinal é *amp111* . Dende aquí premeremos en *Module Admin* A continuación premeremos en *Check for updates online* . Unha vez FreePBX nos amose todas as actualizacións dispoñibles, as seleccionaremos todas e premeremos en *Process*
11. Habilitamos SSL e modificamos a configuración para permitir só SSH2.
12. Probamos dende outro ordenador na mesma LAN se podemos acceder por SSH ao servidor Trixbox. Para iso empregaremos *winscp* e *putty* con usuario *root* e o contrasinal que elixiramos para o CentOS na instalación (paso 2).
13. Dende outro ordenador descargamos e descomprimimos os arquivos coas voces en Español. Con *winscp* crearemos un cartafol no equipo de Trixbox onde trasladaremos estes arquivos e mais o script *script-trixbox.sh*[3]
14. Agora dende o *putty* nos conectamos como root remotamente a través de SSH na máquina Trixbox, e nos diriximos ao cartafol onde está o *script-trixbox.sh*
    chmod 0755 script-trixbox.sh
    sh script-trixbox.sh
15. Editamos /etc/amportal.conf *nano /etc/amportal.conf* Buscamos a liña que pon *AUTHTYPE=none* e o substituimos por *AUTHTYPE=database*
16. Entramos en FreePBX , menú *PBX->PBX Settings* e prememos en *Administrators* . Creamos o usuario *admin* e mudamos o contrasinal de *maint*.
17. Mudamos os contrasinais que teñen por defecto o usuario maint de Trixbox, Flash Operator Panel, Web Meetme, FreePBX, MySQL, o ARI e o servidor de rede de Asterisk. Esta operación é complexa e esixe editar dende o *putty* de xeito manual varios arquivos de configuración, como explícase no anexo.
18. Entramos no router de acceso a internet, para configurar o NAT, a opción mais habitual é *Virtual Server* ou *Port Fordwarding* . Faremos o seguinte:

Servizo	*Porto Externo*	*IP interna*	*Porto interno*	*Tipo*
SIP	5060-5061	192.168.1.99	5060-5061	UDP
SSH	22	192.168.1.99	22	TCP
HTTP	527	192.168.1.99	80	TCP
IAX	4569	192.168.1.99	4569	UDP
RTP	8000-20000	192.168.1.99	8000-20000	UDP

19. Configuramos Trixbox para que SIP funcione nunha rede con NAT.
20. Corriximos alguns erros de programación da versión 2.6.2, ver mais adiante.
21. Quitamos unha copia de seguridade do sistema, entrando en Trixbox como usuario *maint* , menú *System -> Backup*

[3] Este script contén unha serie de ordes básicas do shell de Linux para mover os arquivos de voces en castelán aos cartafoles indicados, para crear enlaces simbólicos e para mudar os permisos por defecto dos cartafoles do CentOS. Tamén corrixe algún erro da versión 2.6.2. e desactiva servizos de Linux non necesarios. Mais adiante detállase o seu contido.

Instalar Trixbox nunha Placa Base Asus M2N-MX-SE

Neste caso ao instalar Trixbox bloquease a instalación coa seguinte mensaxe:

Kernel Panic - not syncing:IO-APIC + timer doesn't work! Boot with apic=debug and send a report. Then try booting with the 'noapic' option.

A solución pasa por entrar na BIOS do PC e desactivar a opción *MCP61 ACPI HPET TABLE* no apartado correspondente ao *Chipset Southbridge.*

Configuración para NAT

Se a rede LAN onde atópase o IP-PBX accede a internet cun router que emprega NAT e mais un firewall, iso pode supoñer un problema de cara ao emprego do protocolo SIP para comunicarse con terminais ou outras IP-PBX situadas fora da LAN e só accesibles a través de internet. A solución é a seguinte:

1. Accedemos a Trixbox como administrador.
2. Seleccionamos PBX->"Config File Editor", para mudar o contido que por defecto ten o arquivo */etc/asterisk/sip_nat.conf*
    a. *Se a dirección IP externa é estática:*
    nat=yes
    externip=xx.xx.xx.xx
    localnet=yy.yy.yy.yy/mm.mm.mm.mm
    b. Se a dirección IP externa é dinámica, débese crear un dominio en http://dyndns.org , instalar o cliente *ddclient* en CentOS[4] () , e xa para rematar o arquivo *sip_nat.conf* sería:
    nat=yes
    externhost=xxxxx.dyndns.xxx ; o noso dominio en *dyndns*
    externrefresh=120
3. Edítase o arquivo */etc/asterisk/sip_general_custom.conf*. Engádese*:*
    port=5060
    bindaddr=0.0.0.0
    nat=yes
    externip=xxxxxx.dyndns.xxxxx ; o mesmo que externhost
    localhost=yy.yy.yy.yy/mm.mm.mm.mm ;a ip local do trixbox
4. Agora dende CentOS executamos *nano /etc/asterisk/rtp.conf*
5. Quitamos (se haino) o punto e coma antes de [general]
6. En *rtpstart* poñemos 8000[5] e en *rtpend* poñemos 20000
    rtpstart=8000
    rtpend=20000
7. En *System->Network* comprobamos a configuración e a completamos se é necesario: en *Host* poñeríamos o dominio externo no caso de telo definido.
8. En *Settings -> General Settings* , en FQDN poñeríamos o nome do dominio externo no caso de telo definido.

[4] Nalgúns routers ADSL xa ven o cliente incorporado só habería que insertar os datos de acceso e o nome do dominio no apartado da configuración apropiado.

[5] Os portos para RTP soen ser do 10000 ao 20000, pero moitos softphones como zoiper non poden empregar portos de mais de 4 díxitos. No caso particular de Zoiper emprega por defecto o 8000 para RTP

9. No paso anterior pódese dar a situación que na nosa LAN teñamos dous equipos ou mais que funcionen sobre SIP e RTP e teñan que ter acceso externo6. Neste caso ao configurar o NAT para que os portos externos 5060-5061 8000-20000 os asocie a IP local dun deles, o outro quedará sen acceso externo. A solución e empregar portos SIP e RTP diferentes para cada equipo. Iso deberemos telo en conta ademais de na configuración do NAT do router de acceso externo, na configuración dos equipos e do Trixbox, ao definir as extensión e nos arquivos sip_general_custom.conf (port=xxxx)e rtp.conf
10. Dende o shell de CentOS, ao que podemos acceder remotamente por SSH co *putty*, editamos o arquivo /etc/hosts, *nano /etc/hosts* e engadiremos unha liña por cada IP local e externa (xx.xx.xx.xx) e *dominio* coa sintaxe seguinte:
<IP> <HOSTNAME>.<DOMAIN> <ALIAS>.
127.0.0.1 localhost localhost.localdomain dominio alias
xx.xx.xx.xx dominio alias localhost localhost.localdomain

Creación dun dominio en dyndns.org

Se temos acceso a internet por medio dunha IP externa dinámica, isto será un problema de cara á configuración do NAT en Trixbox e mais para configurar un servidor smtp externo como gmail. A solución e dar de alta de xeito gratuito un dominio dinámico en dyndns.org (hai mais provedores de este servizo de xeito gratuíto como por exemplo no-ip.com).

1. Obtemos a nosa IP externa de http://whatismyip.com
2. Creámonos un dominio dinámico de tipo dyndns.info, para iso primeiro rexistrámonos en:
http://dyndns.com/services/dns/dyndns/ pulsamos "*Create Account*"
3. Creamos un dominio do *tipo http://o meu_nome_de_dominio.dyndns.info* desde *https://www.dyndns.com/account/services/hosts/add.html*
   - Activamos os wildcards
   - Servizo de tipo *Host with IP adress*
   - Introducimos a IP dinámica externa no campo IP adress
   - Non marcamos *configure mail routing*
   - Ao rematar hai que pulsar *next* e *activate*.
4. Instalar o cliente de dyndns.com *ddclient* no IP-PBX. Este ocuparase de conectarse periodicamente á páxina web de dyndns e renovar a nosa ip dinámica, para que non nos dean de baixa. Neste caso empregaremos o programa ddclient que está en:
http://cdn.dyndns.com/ddclient.tar.gz
5. Descomprimirlo e copiar os contidos para linux do arquivo recien descomprimido no CentOS:
chmod 0755 /mnt/compartido/ddclient-3.7.3/ddclient
chmod 0755 /mnt/compartido/ddclient-3.7.3/sample-etc_rtc.d_init.d_ddclient
mv /mnt/compartido/ddclient-3.7.3/ddclient /usr/sbin/
mkdir /etc/ddclient
mv /mnt/compartido/ddclient-3.7.3/sample-etc_ddclient.conf /etc/ddclient/ddclient.conf
mv /mnt/compartido/ddclient-3.7.3/sample-etc_rtc.d_init.d_ddclient /etc/rtc.d/init.d/ddclient

[6] Ollo! isto as veces ocurre por instalar un servidor Trixbox nunha rede de area local con acceso a internet a través dun operador que no router soporta VoIP sobre SIP. Neste caso o mellor é empregar no Trixbox o porto 5070 ou outros, e deixar os 5060 para o servizo VoIP do noso operador.

6. Editamos *ddclient.conf* coa configuración de acceso e de actualización da nosa ip dinámica en dyndns.org: *nano /etc/ddclient/ddclient.conf* . Escribimos o seguinte:

   daemon=600
   cache=/tmp/ddclient.cache
   pid=/var/run/ddclient.pid
   use=web, web=checkip.dyndns.com/, web-skip='IP Address'
   login= Usuario_de_dyndns
   password=contraseña-De_dyndns
   protocol=dyndns2
   server=members.dyndns.org
   wildcard=YES
   o meu_nome_de_dominio.dyndns.info

   Se o router de acceso a internet tivera integrado un cliente de dyndns, podemos empregar en tornas de *use=web, web=checkip.dyndns.com/, web-skip='IP*, o seguinte:

   fw-login=admin, fw-password=contrasina_do_router
   use=fw, fw=ip_do_router/status.htm, fw-skip='IP Address'

   Neste caso, tamén debe mudarse *daemon=600* por *daemon=60*
7. Engadimos *ddclient* para que ao arrincar o Trixbox, se autoarranque (Ollo! se reiniciamos a IP-PBX con frecuencia, dyndns.org detectará un exceso de actualizacións da nosa IP dinámica e pódenos cancelar a conta. Isto é só recomendabel se a IP-PBX apágase pouco. Se apágase moito é mellor activalo "a man" cada vez que se necesite *service ddclient start*):

   /sbin/chkconfig --add ddclient

Mudar os contrasinais por defecto

Nos seguintes arquivos almacenanse os contrasinais dos diferentes compoñentes e usuarios de Trixbox. Ainda que algúns poden mudarse dende o portal de Trixbox, outros non e deben revisarse para evitar erros. Os contrasinais corresponden aos seguintes módulos de Trixbox:

- Usuario maint de Trixbox , valor por defecto *password*
- Usuario root de mysql , valor por defecto *passw0rd*
- Administrador wmm@localhost de web-meetme , valor por defecto *passw0rd*
- Usuario tim@localhost de web-meetme, valor por defecto *1234*
- Asteriskuser , valor por defecto *amp109*
- Usuario admin de FreePBX valor por defecto *amp111*
- Administrador de Flash Operator Panel, valor por defecto *passw0rd*
- Contrasinal de acceso ao ARI, valor por defecto *passw0rd*

1. Antes de editar manualmente cada un destes arquivos faremos: *passwd-maint* para mudar o contrasinal de acceso como administrador a Trixbox do seu valor por defecto *password*
2. A continuación mudaremos os contrasinais dende mysql:

   yum install mod_auth_mysql
   mysql –u root –p
   use mysql;
   update user set Password=’contrasinal_asterisk’ where User=’asteriskuser’;

flush privileges;
use meetme;
update user set password=’contrasinal_admin_meetme’ where email=’wmm@localhost’;
update user set password=’ contrasinal_user_meetme’ where email=’tim@localhost’;
quit
mysqladmin -u root -p password contrasinal_novo_mysql

3. Agora editaremos co editor *nano* os seguintes arquivos e buscaremos os contrasinais que ainda non estén mudados ao seu valor axeitado:

   /etc/amportal.conf
   /etc/asterisk/cdr_mysql.conf
   /etc/asterisk/cbmysql.conf
   /etc/asterisk/extensions_additional.conf
   /etc/asterisk/phpagi.conf
   /etc/asterisk/manager.conf
   /var/lib/asterisk/agi-bin/user_login_out.agi
   /var/www/html/admin/modules/core/agi-bin/user_login_out.agi
   /var/www/html/admin/functions.inc.php
   /var/www/html/admin/modules/framework/htdocs/admin/functions.inc.php
   /var/www/html/classes/endpoint.php
   /var/www/html/framework/system/config/database.php
   /var/www/html/framework/modules/trixbox/libraries/Trixpbx.php
   /var/www/html/maint/modules/endpointcfg/libs/trixbox.php
   /var/www/html/maint/modules/cdrreport/config/database.php
   /var/www/html/maint/modules/asterisk_info/asterisk_info.php
   /var/www/html/maint/modules/sysmaint/libs/trixbox.php
   /var/www/html/phpmyadmin/config.inc.php
   /var/www/html/recordings/includes/main.conf.php
   /var/www/html/web-meetme/trixbox-files/cbmysql.conf
   /var/www/html/web-meetme/lib/defines.php
   /var/www/html/web-meetme/lib/database.php
   /var/www/html/xmlservices/include/xmlservices_libs.php

Configurar e Activar o Servidor DHCP

Para poder empregar na LAN asignación dinámica de IPs, para facilitar a configuración dos teléfonos IP que poidera haber na rede local, podo facelo dende o IP-PBX xa que Trixbox incorpora un servidor DHCP.

1. Executamos dende CentOS *setup-dhcp*
2. Accedemos a Trixbox como administrador.
3. Seleccionamos “Config Edit” para mudar o contido do arquivo *etc/dhcpd.conf* cos intervalos de direccións IP que o servidor asignará.

Script script-trixbox.sh

O script script-trixbox.sh contén ordes do shell de CentOS necesarias para:

- Instalar o paquete de voces en castelán do Trixbox
- Configurar os permisos dos cartafoles
- Desactivar servizos do CentOS non necesarios para Trixbox
- Protexer o Flash Operation Panel de accesos non autorizados
- Corrixir alguns erros da versión 2.6.2

Este script copiase nun cartafol do equipo xunto cos arquivos de voces descomprimidos, accediendo remotamente por ssh co winscp. Despois accédese a consola como root (remotamente por SSH co Putty) para facer executable o script e executalo. Dende o cartafol onde está o script facemos:

chmod 0755 script-trixbox.sh
sh script-trixbox.sh

CONTIDO DO SCRIPT SCRIPT-TRIXBOX.SH

```
#!/bin/bash
#previamente descomprimimos os dous arquivos tar de voces no mesmo cartafol
# que este script.
#colocamos os arquivos de voces no directorio de asterisk
mv es /var/lib/asterisk/sounds/
mv dictate /var/lib/asterisk/sounds/es/
mv digits /var/lib/asterisk/sounds/es/
mv followme /var/lib/asterisk/sounds/es/
mv letters /var/lib/asterisk/sounds/es/
mv phonetic /var/lib/asterisk/sounds/es/
mv silence /var/lib/asterisk/sounds/es/
cp /var/lib/asterisk/sounds/es/vm-tempgreetactive.gsm /var/lib/asterisk/sounds/es/vm-
tempexists.gsm
cp /var/lib/asterisk/sounds/es/dir-intro.gsm /var/lib/asterisk/sounds/es/dir-intro-
oper.gsm
#enlaces simbolicos
ln -s /var/lib/asterisk/sounds/digits/es /var/lib/asterisk/sounds/es/digits
ln -s /var/lib/asterisk/sounds/dictate/es /var/lib/asterisk/sounds/es/dictate
ln -s /var/lib/asterisk/sounds/followme/es /var/lib/asterisk/sounds/es/followme
ln -s /var/lib/asterisk/sounds/letters/es /var/lib/asterisk/sounds/es/letters
ln -s /var/lib/asterisk/sounds/phonetic/es /var/lib/asterisk/sounds/es/phonetic
ln -s /var/lib/asterisk/sounds/silence/es /var/lib/asterisk/sounds/es/silence
# ln -s /var/lib/asterisk/sounds/es/digits /var/lib/asterisk/sounds/digits/es
# Correccion de bug de version 2.6
ln -s /var/lib/asterisk/moh /var/lib/asterisk/mohmp3
# Configura os permisos mínimos para os cartafoles claves de Asterisk
# Modifica os cartafoles baseándote no contido do teu arquivo /etc/asterisk.conf
# Fai os arquivos propiedade do root e do grupo de asterisk, de xeito que
# un usuario de asterisk non teña permiso de escritura nos arquivos de
# configuración
chown -R root:asterisk /etc/asterisk /usr/lib/asterisk/ /var/lib/asterisk/agi-bin
/var/spool/asterisk /var/run/asterisk /var/log/asterisk
# Da ao root permisos r/w/x e da permisos de grupo (asterisk) r/x
chmod 750 /etc/asterisk /usr/lib/asterisk/ /var/lib/asterisk/agi-bin
```

/var/spool/asterisk /var/run/asterisk /var/log/asterisk
 # Nestes cartafois tamén é necesario permiso de escritura para o grupo asterisk
chmod 770 /var/spool/asterisk /var/run/asterisk /var/log/asterisk
 # desactivamos os servizos do SO non necesarios
chkconfig kudzu off
chkconfig rawdevices off
chkconfig pcmcia off
chkconfig portmap off
chkconfig rpcidmapd off
chkconfig haldaemon off
chkconfig mdmonitor off
chkconfig netfs off
chkconfig isdn off
chkconfig rpcgssd off
chkconfig iptables off
chkconfig irqbalance off
chkconfig vsftpd off
chkconfig auditd off
chkconfig smartd off
chkconfig readahead off
chkconfig microcode_ctl off
chkconfig cpuspeed off
chkconfig messagebus off
chkconfig readahead_early off
chkconfig nfslock off
chkconfig lm_sensors off
chkconfig ircd off
chkconfig autofs off
 # Proteccion para que nadie que acceda a mi IP pueda ver el FOP
mkdir /var/www/protect
mv /var/www/html/panel/.cfg /var/www/protect*
cd /var/www/html/panel
ln -s ../../protect/op_server.cfg op_server.cfg
ln -s ../../protect/op_style.cfg op_style.cfg
ln -s ../../protect/op_astdb.cfg op_astdb.cfg
ln -s ../../protect/op_buttons.cfg op_buttons.cfg
ln -s ../../protect/op_buttons_custom.cfg op_buttons_custom.cfg
ln -s ../../protect/op_buttons_additional.cfg op_buttons_additional.cfg
ln -s ../../protect/op_lang_es.cfg op_lang_es.cfg
ln -s ../../protect/op_lang_en.cfg op_lang_en.cfg
ln -s ../../protect/op_lang_de.cfg op_lang_de.cfg
ln -s ../../protect/op_lang_it.cfg op_lang_it.cfg
 # Correccion de bug de version 2.6
*chmod a+x /usr/local/sbin/**
 # instalación de herramienta de backup y de soporte de SSL
yum --enablerepo=trixboxbeta install tbm-backup
yum –y install mod_ssl
reboot

Instalar os paquetes de voces en Castelán

Para que as mensaxes de audio que ten Trixbox incorporado para os distintos acontecementos que sucedan na IP-PBX , estean en castelán (en galego non hai nada dispoñible polo momento) hai que seguer este procedemento:

1. Descargamos dende http://www.voipnovatos.es/voces/ os arquivos comprimidos co audio codificados en gsm (por exemplo). Colleremos o *core* e o de *extras*.
2. Dende CentOS descomprimos os arquivos con *tar xvfz nome_arquivo.tar.gz*, e dende windows con programas como winrar ou similar. Ao descomprimir crearanse os seguintes cartafoles: es , dictate, digits, followme, letters, phonetic y silence.
3. Copianse os cartafois anteriores xunto co *script-trixbox.sh* a un mesmo cartafol do CentOS. Para isto pódese empregar o winscp.
4. Ao executar o *script-trixbox.sh* completase a instalación das voces.
5. Engadimos *language=es* aos seguintes arquivos en /etc/asterisk:
   zapata.conf, iax_custom.conf, sip_general_custom.conf

Erros de Trixbox CE 2.6.2.

Debemos corrixir os seguintes erros:

- En */etc/trixbox/httpdconf/trixbox.conf* comentar con # a sección do arquivo seguinte:

  #Password protect /var/www/html/admin
  #AuthType Basic
  #Authname “Restricted Area”
  #AuthUserFile /usr/local/apache/passwd/wwwpasswd
  #Require user wwwadmin maint

- No arquivo */var/www/html/web-meetme/user_add_sqldb.php* veñen dous erros:
  - Ven ‘userPassr’ cando o correcto é ‘userPass’
  - Na liña 94 pon $userPass=”password=’$userPass”; cando o correcto é $userPass=”$userPass”;
- En */etc/asterisk/phpagi.conf* hai que engadir as seguintes liñas:

  [asmanager]
  server=localhost
  port=5038
  username=admin
  secret=amp11

Acceso remoto ao servidor con seguridade: SSH

Para poder acceder ao servidor de xeito remoto empregando SSH2 en exclusividade, previamente quitar as # das primeiras liñas do arquivo */etc/ssh/sshd_config* :

Port 22
Protocol 2
ListenAddress 0.0.0.0 #Poderei conectarme a calquera das IPs do Pc

Aumentar o límite de Memoria para executar scripts

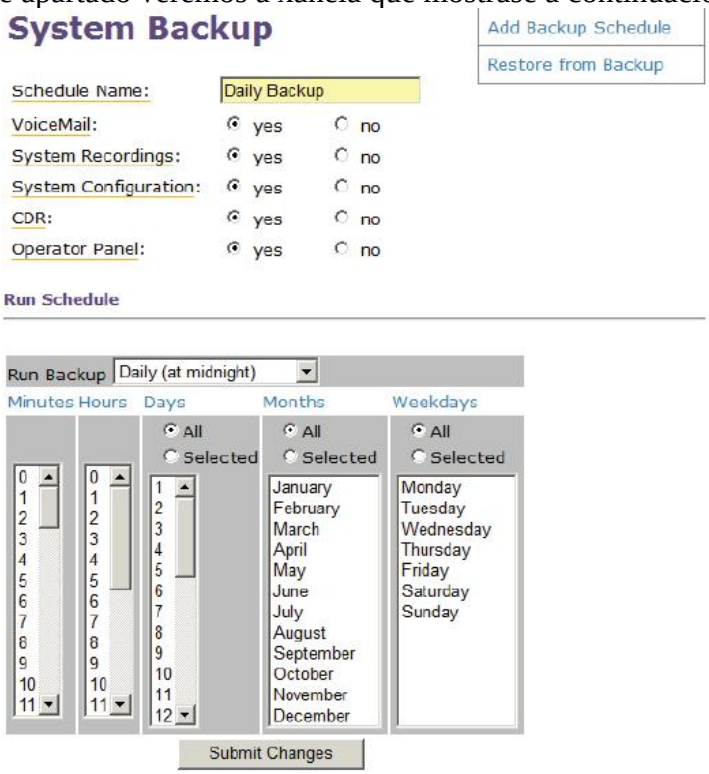
As veces Trixbox consume demasiada RAM, e é aconsellable aumentar ata 100MB o límite cun script PHP pode consumir (ven 64MB por defecto). Para iso editamos o arquivo */etc/php.ini* e trocamos o parámetro memory_limit para que poña o seguinte:

memory_limit = 100M

Configurar as Copias de Seguridade

Para configurar, restaurar e planificar as copias de seguridade do sistema, FreePBX ten un apartado específico, en *Tools->Backup and restore*

Ao acceder a este apartado veremos a xanela que mostrase a continuación:



- **Schedule name**: Dálle a esta copia de seguridade un nome amigable (exemplo: "diario" ou "Correo de voz") para identificar correctamente que é o que está nesa copia de seguridade. Isto facilitará as futuras restauracións.
- **Voicemail**: Inclue as mensaxes de correo de voz na copia de seguridade. Isto pode incrementar seriamente o tamaño da súa copia de seguridade por estar cargando arquivos de audio potencialmente longos.
- **System recordings**: Para resgardar as gravacións do sistema personalizadas que se creen para o recepcionista virtual ou colas. Novamente, isto pode incrementar notablemente o tamaño da copia de seguridade.
- **System Configurations**: Con esta opción se resgardarán datos de configuración de Asterisk e FreePBX, incluíndo o MySQL e as bases de datos de Asterisk. É recomendable que isto estea activo en todas as copias de seguridade.
- **CDR**: Garda o rexistro das chamadas.
- **Operator panel**: Para resgardar o FOP (Flash Operator Panel)
- **Run Backup**. Pódese elixir unha programación preconfigurada desde o menú de selección, ou configurar unha programación personalizada (*Ad Hoc*) utilizando minutos, horas, días, meses e fins de semana desde a caixa de selección. As opcións preconfiguradas son: *Diary* (ao mediodía), *Weekly* (Domingo ou Sábado), *Monthly* (No primeiro día de cada mes á medianoite) ou *Anual* (no primeiro día de Xaneiro á medianoite). Ao premer en *Add Backup Scheme* gárdase a programación seleccionada.

- **Restore from Backup**. Isto mostrará todos as copias de seguridade que estean presentes no sistema (situados en */var/lib/asterisk/backups*). A continuación poderase seleccionar a copia de seguridade que se queira restaurar.

Para acabar coa instalación de Trixbox, e despois de tódolos cambios efectuados convén facer unha copia de seguridade do sistema, para poder recuperalo en caso de avaría.

1. Dende FreePBX vaia ao apartado *Tools->Backup and restore*
2. Tódalas opcións deberán estar en *no*, excepto *System configurations* que deberá estar en *yes*.
3. Poña un nome o campo *Schedule name* , por exemplo *copiainicial*
4. Seleccione na lista *Run Backup* a opción *Now* (agora).
5. Prema en *Add Backup Scheme*
6. Unha vez que o sistema teña feita a copia, acceda outra vez no FreePBX ao apartado *Tools->Backup and restore*
7. Comprobe premendo en *Restore Backup* como se creou o arquivo coa copia de seguridade.
   Tamén pode velo dende o shell de CenOS co comando *ls /var/lib/asterisk/backups*

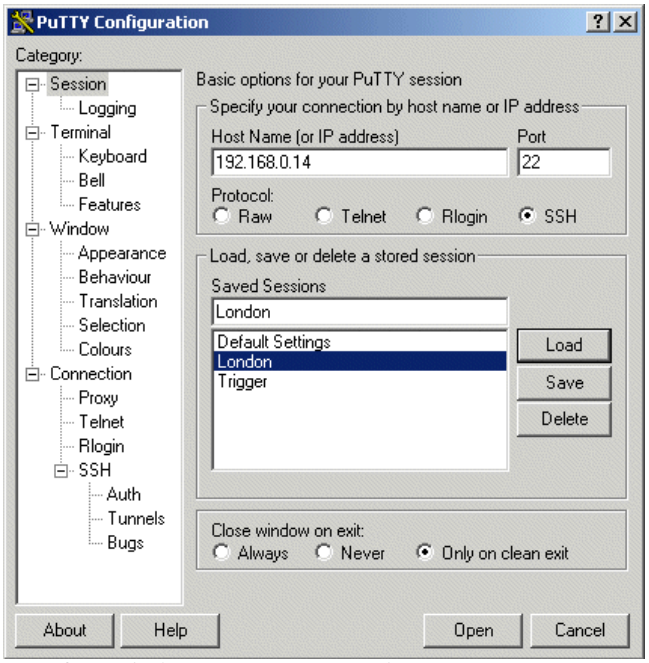
Comandos dende o shell de CentOSde utilidade

- *df-h:* Amosa o espazo dispoñible no disco duro
- *asterisk-r*: Executa o interprete de comandos de interface de Asterisk (CLI) sendo root.
- *quit*: Sae do Asterisk CLI.
- *amportal restart*: Reinicia Asterisk e Flash Operator Panel (FOP)
- *grep amp1 –R /etc/*>>/ficheros.txt* Busca o texto amp1 en todolos arquivos do cartafol /etc/ e nos dos seus subcartafoles. Por cada vez que o atopa engade ao final de /ficheros.txt o contido da liña onde o atopou así como o nome do arquivo.
- *echo grep passw0rd –R /var/www/html/*>/ficheros.txt Crea ou s*obreescribe o arquivo /ficheros.txt coa liña de texto: *grep passw0rd –R /var/www/html/**
- *sed "s/passw0rd/meu_contrasinal/g" /etc/amportal.cfg>/etc/amportal.cfg* Reempraza o texto *passw0rd* do arquivo */etc/amportal.cfg* por *meu_contrasinal*
- *netstat –nr* Amosa as tablas de enrutamento (IPTABLES)
- *shutdown -h now*: Pecha o sistema inmediatamente.
- *shutdown -r now*: Reinicia o equipo instantaneamente.
- *service mysqld restart* Reinicia mysql
- *amportal restart* Reinicia Asterisk
- *service httpd restart* Reinicia o servidor web Apache
- *mysql –u root –p* Entrar en mysql como usuario *root* (nos pide o contrasinal)
- *show databases* Amosar as bases de datos existentes en mysql
- *use nombre_base_datos_elixida*; Seleccionar unha das bases de datos
- *show tables*; Amosar as táboas da base de datos seleccionada
- *show columns from nome_de_taboa* Amosa os campos da táboa seleccionada
- *select * from nome_de_taboa where campo_da_tába='valor_desexado_do_campo';* Selecciona a posición da táboa que nun campo determinado teña un valor determinado.
- *update nome_de_taboa set password ='novo_contrasinal' where campo_da_táboa ='valor_desexado_do_campo';* Actualiza o contrasinal
- *quit* sair de mysql

Putty

Putty es un software gratuito SSH, Telnet, rlogin y ciente TCP raw. Originalmente estaba disponible únicamente para Windows, pero ahora está disponible para varias plataformas Unix, con puertos de trabajo en progreso (work-in-progress) a Mac OS y Mac OS X. Otras personas han contribuido con puertos no oficiales para otras plataformas, como teléfonos móviles potenciados por Symbian. Su versión beta es escrita y mantenida primordialmente por Simón Tatham, es de código abierto y está licenciado bajo una licencia del MIT.

Puede descargarse desde: http://www.putty.org

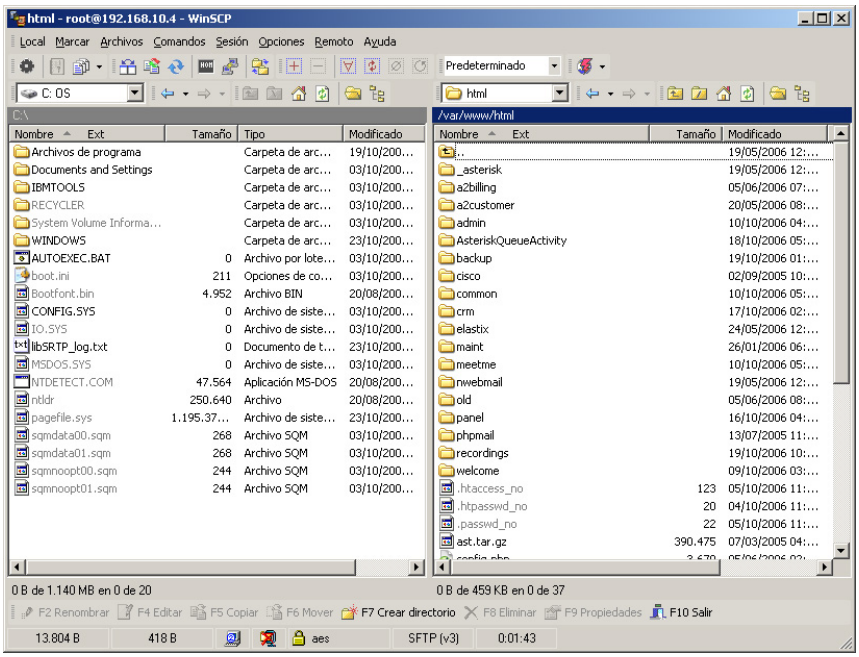


Configuración de Putty para conectarse a la ip 192.168.0.14 por SSH.

WinSCP

WinSCP es una aplicación de Software Libre. WinSCP es un cliente SFTP gráfico para Windows que emplea SSH. El anterior protocolo SCP también puede ser empleado. Su función principal es facilitar la transferencia segura de archivos entre dos sistemas informáticos, el local y uno remoto que ofrezca servicios SSH.

Puede descargarse desde *http://winscp.net*



TRIXBOX - ISDN

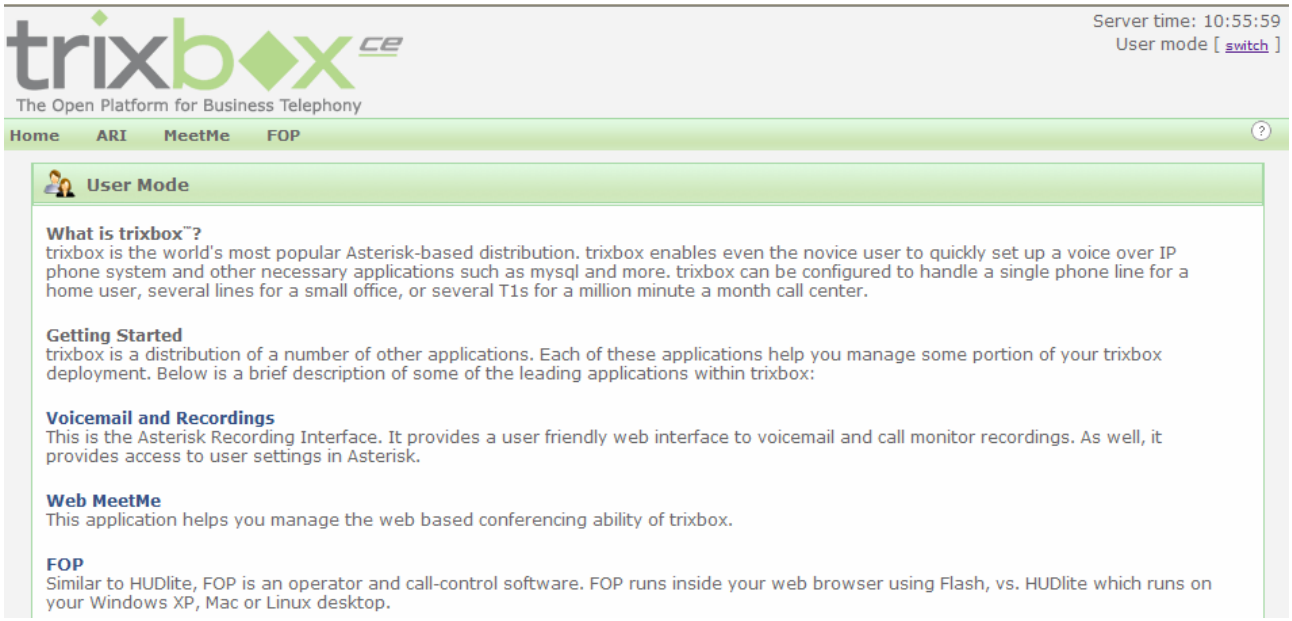
trixbox-update
yum -y update
yum -y install kernel-devel
cd /usr/src
wget beronet.com/download/install-misdn-mqueue.tar.gz
tar xzfv install-misdn-mqueue.tar.gz
cd install-misdn-mqueue
make
make install
reboot
/etc/init.d/misdn-init scan
/etc/init.d/misdn-init config
/etc/init.d/misdn-init start

Una vez realizado estos comandos, tendremos que editar el archivo **/etc/asterisk/misdn.conf** :

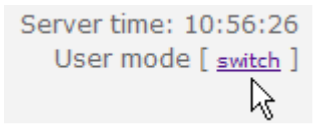
```
[general]

debug = 0
method=standard
bridging=no
stop_tone_after_first_digit=yes
append_digits2exten=yes
dynamic_crypt=no
crypt_prefix=**
crypt_keys=test,muh

[default]
context=misdn
language=es
musicclass=default
senddtmf=yes
far_alerting=no
allowed_bearers=all
nationalprefix=0
internationalprefix=00
rxgain=0
txgain=0
te_choose_channel=no
pmp_l1_check=no
reject_cause=16
need_more_infos=no
nttimeout=no
method=standard
dialplan=0
localdialplan=0
cpndialplan=0
early_bconnect=yes
incoming_early_audio=no
nodialtone=no
callgroup=1
pickupgroup=1
presentation=-1
screen=-1
echocancel=yes
jitterbuffer=4000
jitterbuffer_upper_threshold=0
hdlc=no

[isdn]
ports=1
context=from-pstn
msns=*
```

Acceso a la configuración con GUI

Tecleamos la Dirección IP de TrixBox en un navegador



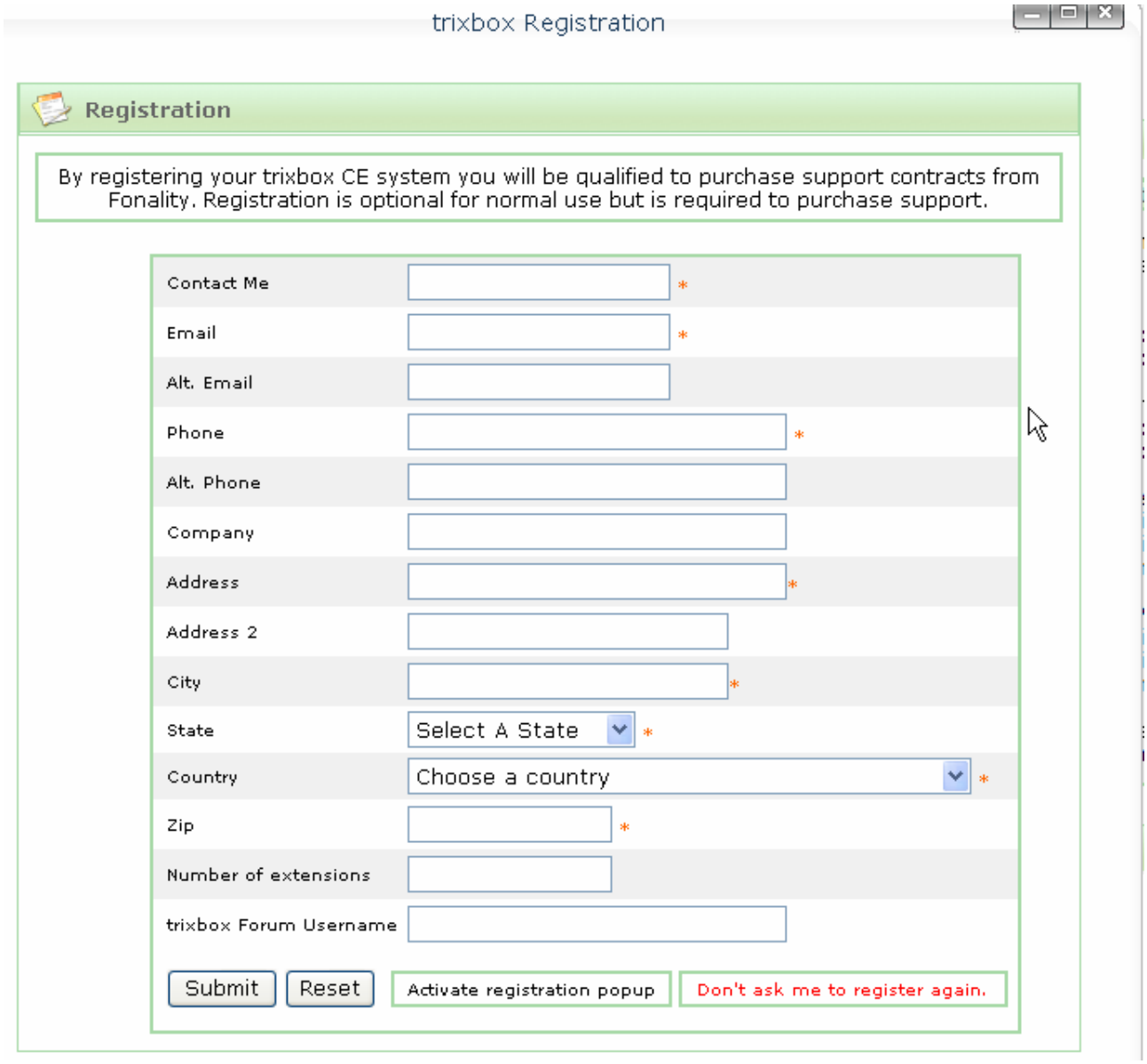
Pasamos a modo administrador



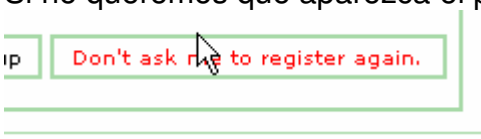
Tecleamos el usario y la contraseña (el usuario es `maint` la contraseña es password)

Nos aparece un formulario de registro. El registro es INDISPENSABLE si queremos actualizar o instalar módulos o el propio TrixBox desde el GUI y nos da acceso a los foros de soporte[2].

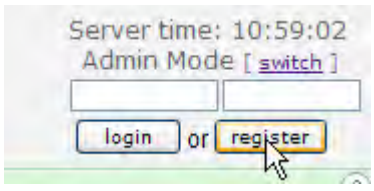
[2] http://www.trixbox.org/forum

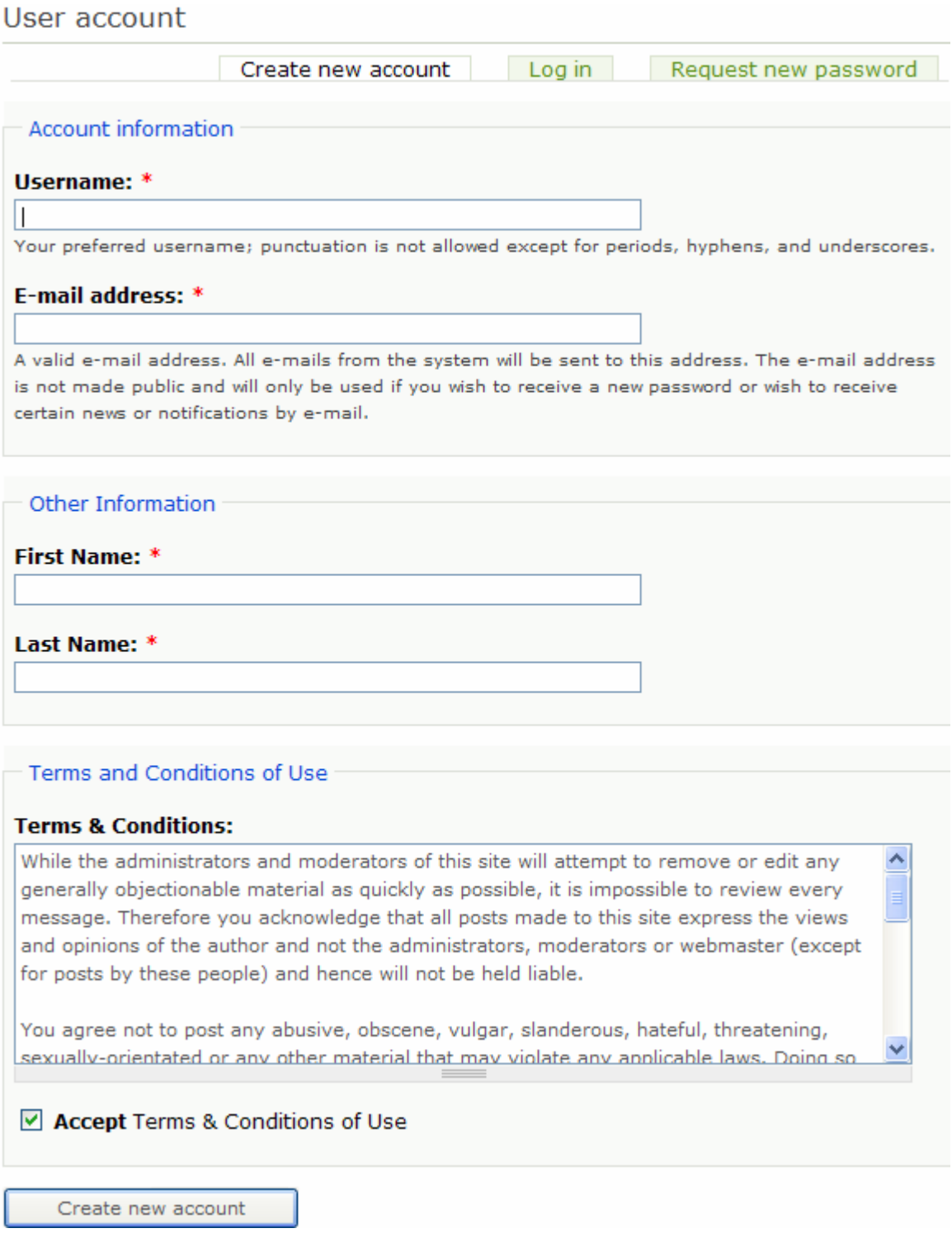


Si no queremos que aparezca el popup de registro

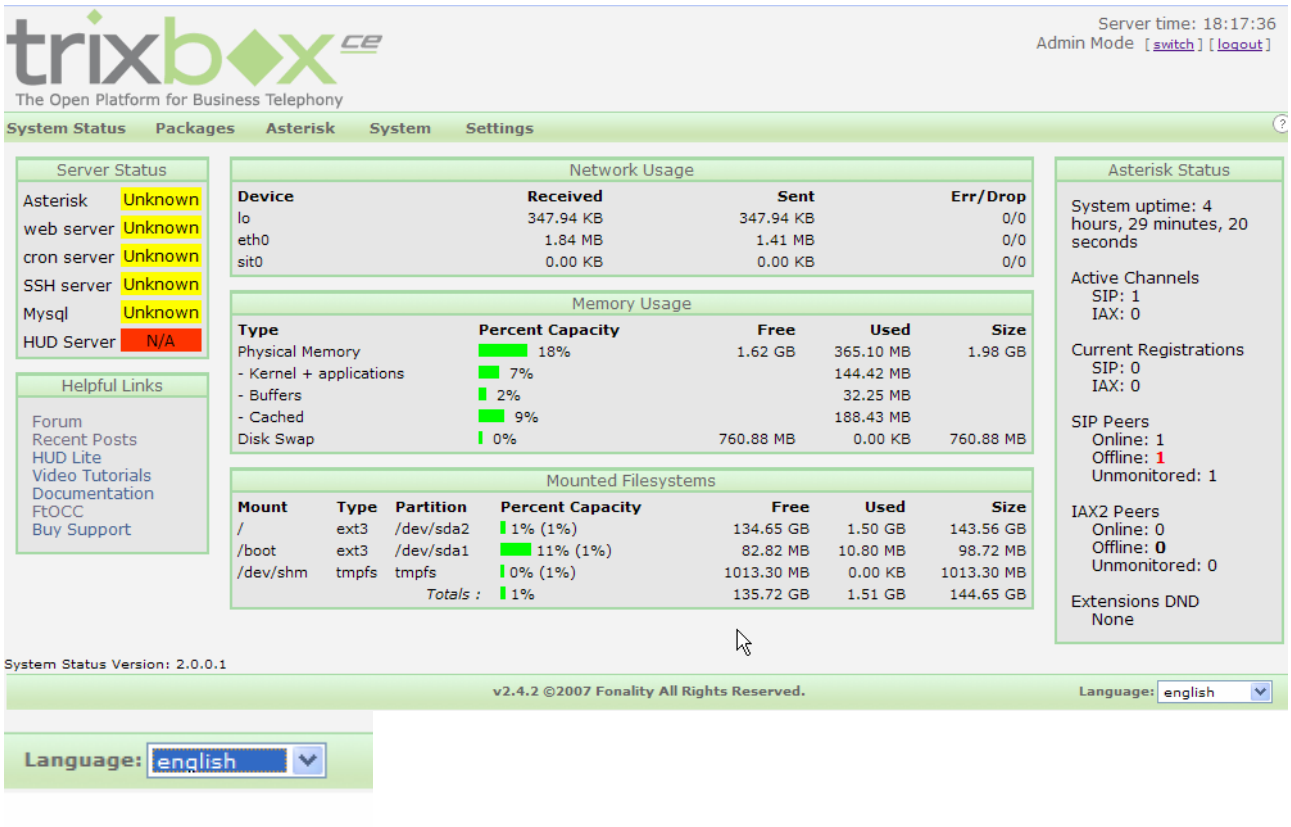


También podemos registrarnos dándole a register.

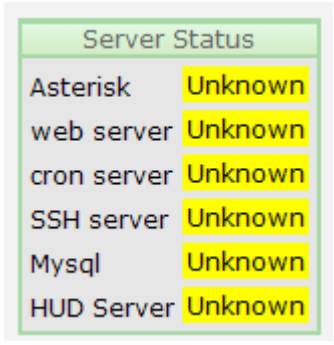




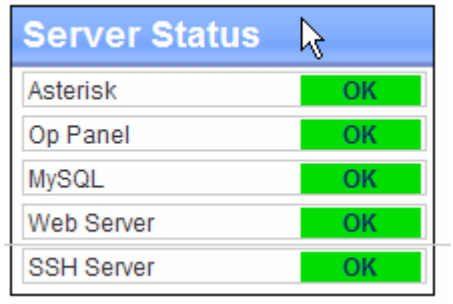
Si queremos podemos cambiar el idioma



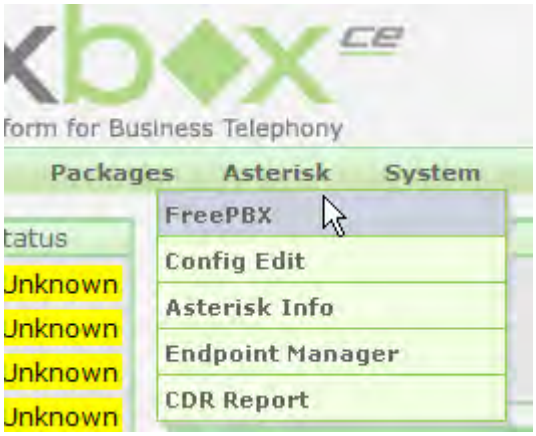
Estado del Sistema



En la pantalla de TrixBox no aparece correctamente reflejado el estado del sistema. Ve remos que aparece correctamente en FreePBX más adelante.



Para Abrir FreePBX



Obtendremos una nueva ventana:

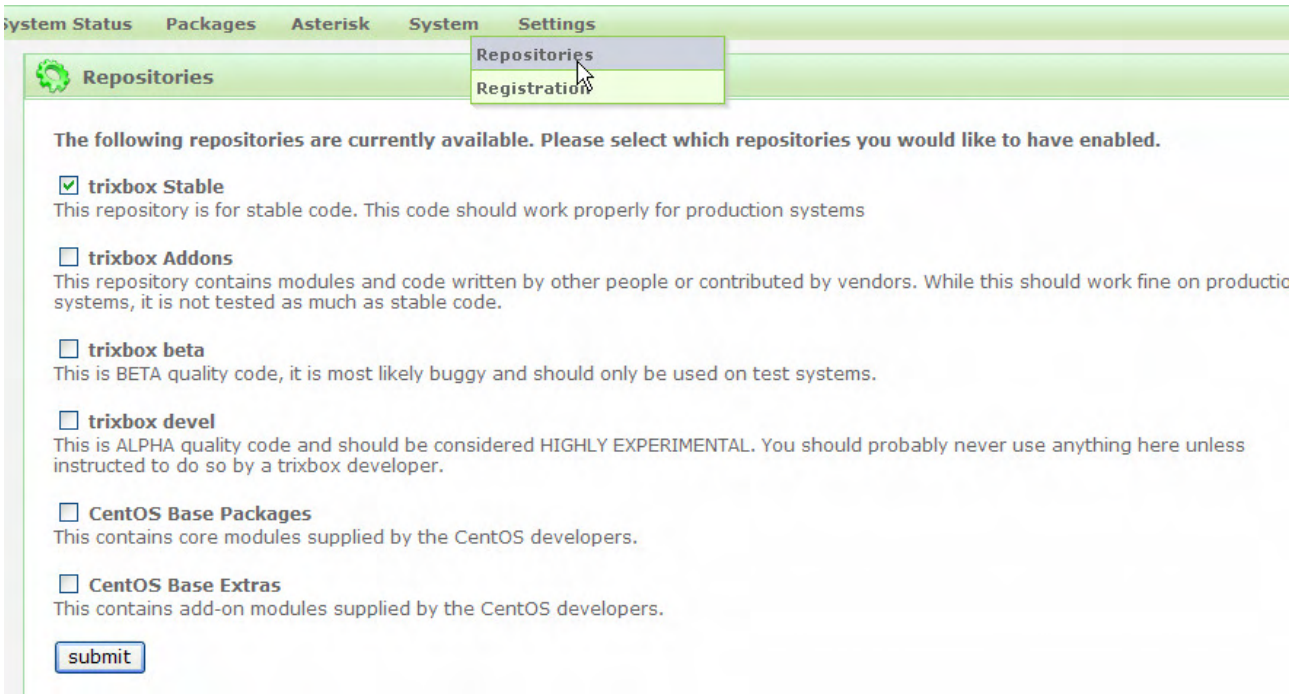
Actualizaciones e Instalaciones

Actualizar el sistema (centos 5)

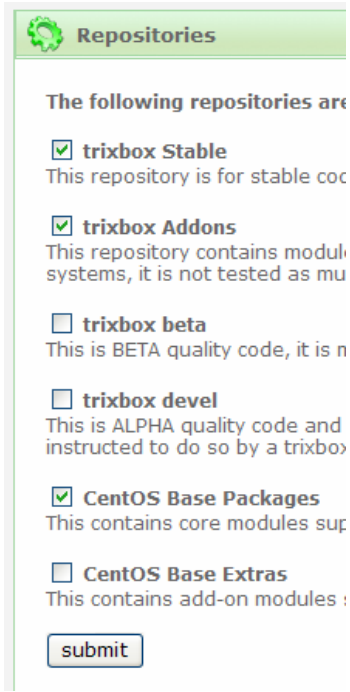
Tecleamos `yum -y update`

Actualizar el sistema desde el GUI

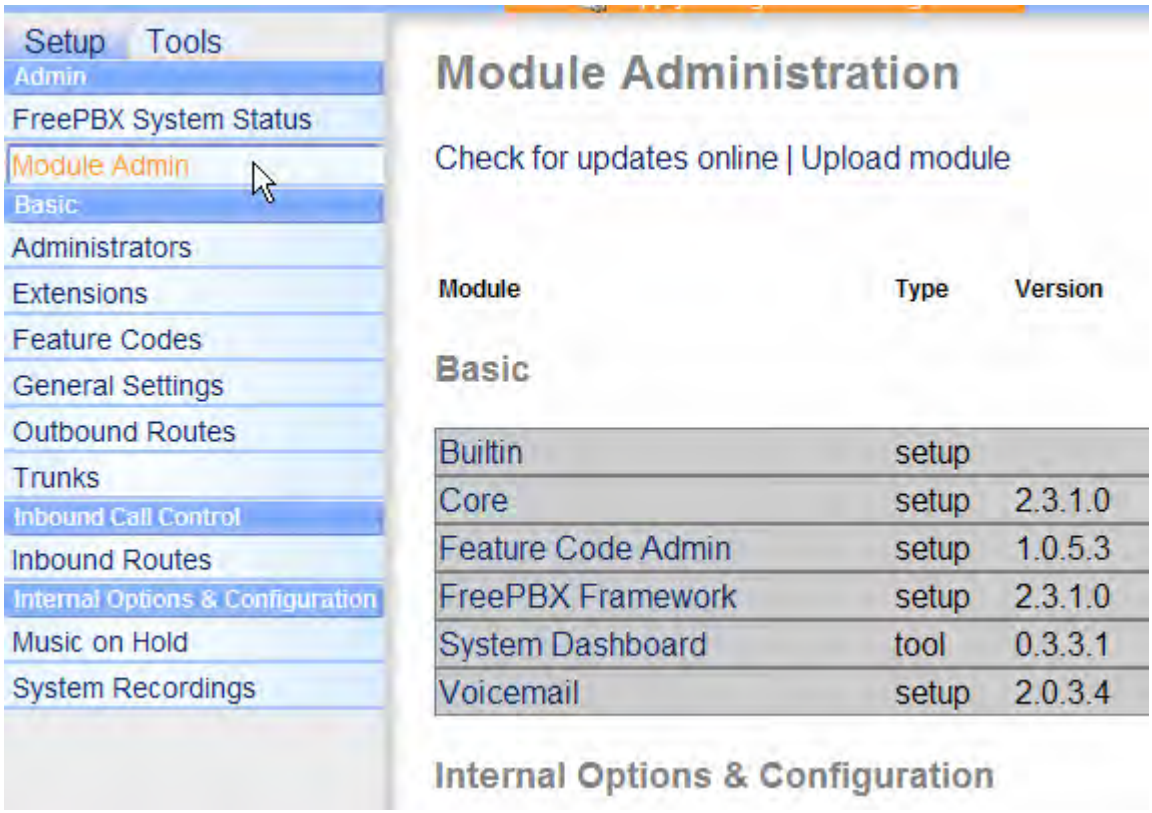
Selección de repositorios. Según las necesidades podemos seleccionar diferentes repositorios de paquetes del sistema TrixBox y CentOS.



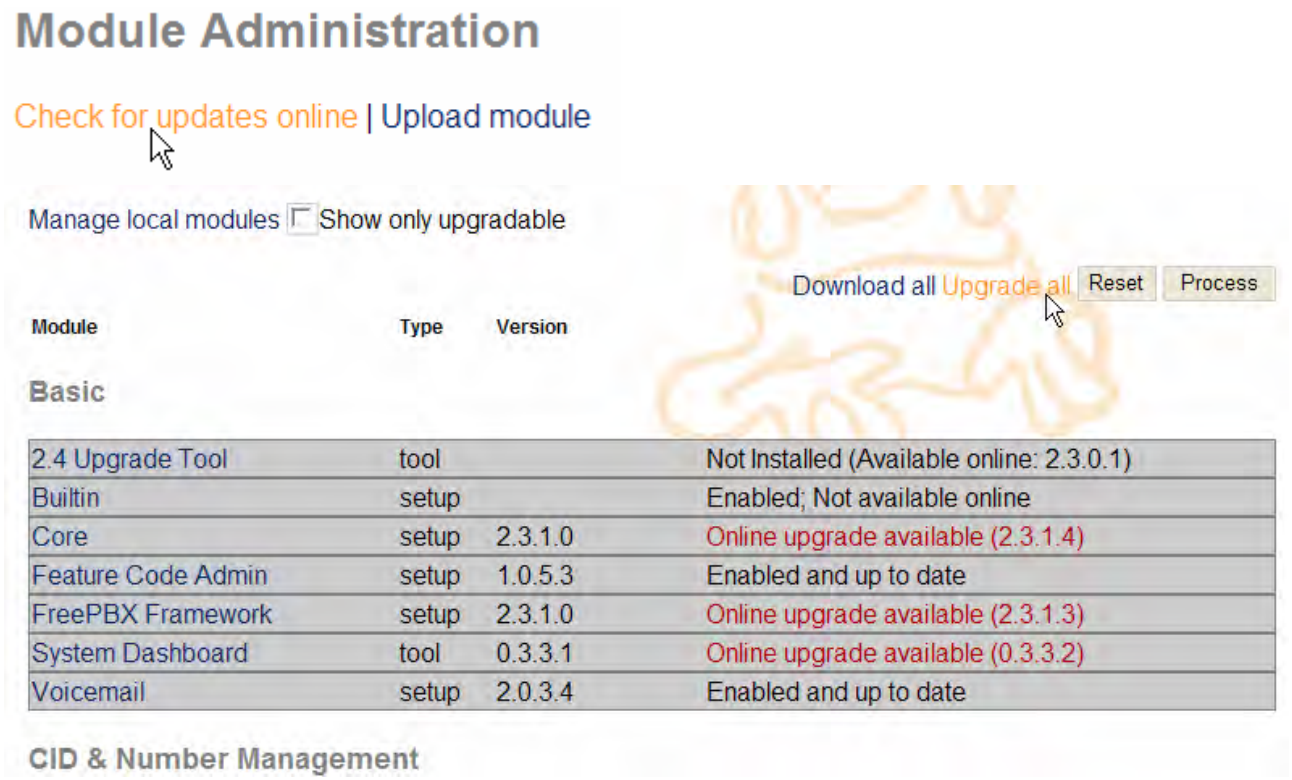
Recomendamos:

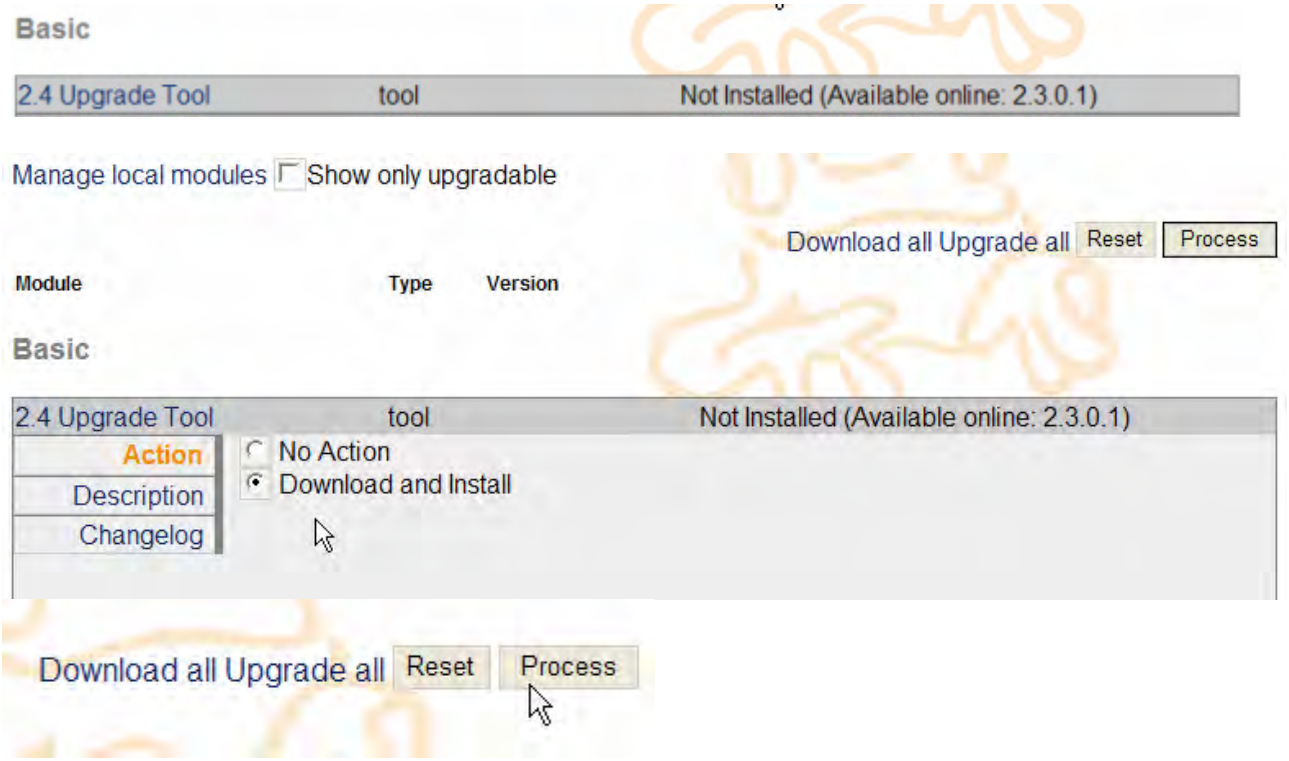


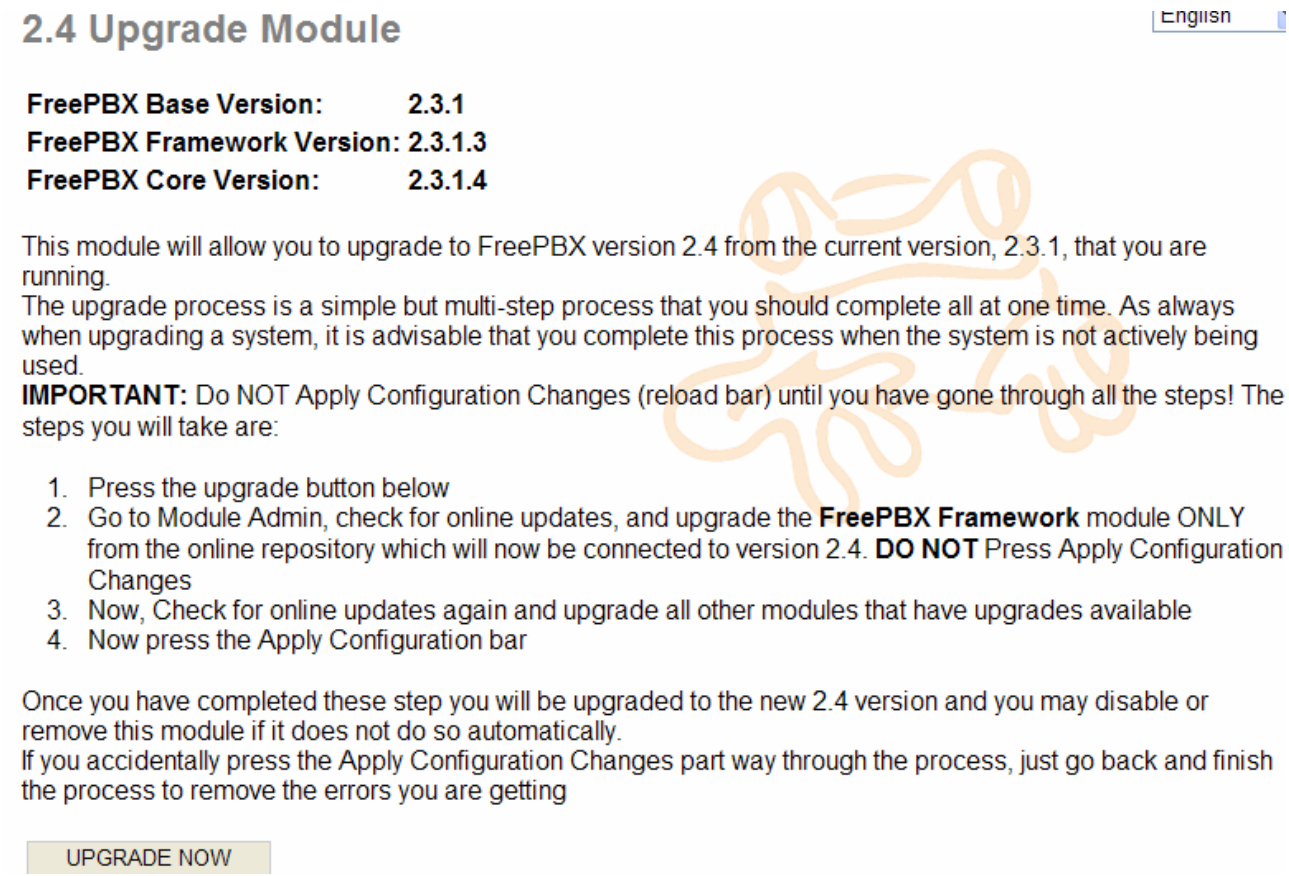
Actualizar FreePBX



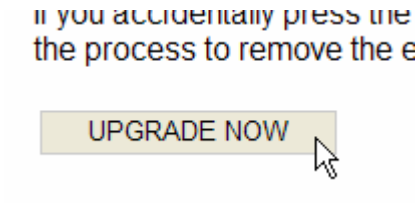
El primer módulo que instalaremos será el de ACTUALIZACIONES
Necesitamos haber realizado login/password y seleccionamos las actualizaciones on-line





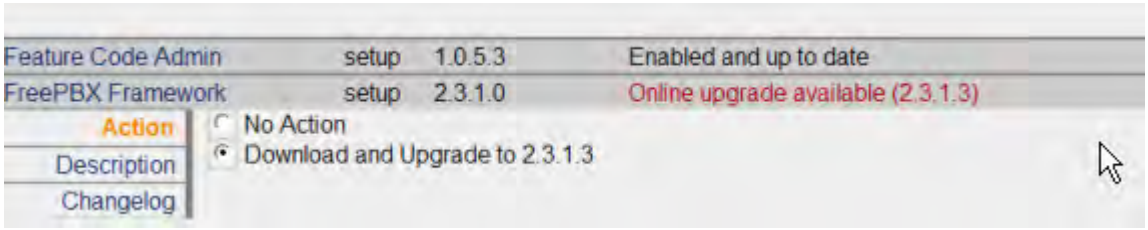


Realizamos el UPGRADE.



Ahora el FRAMEWORK

1. Go to Module Admin, press check for online updates, and upgrade the **FreePBX Framework** module from the online repository. Do not upgrade other modules. **DO NOT** press Apply Configuration Changes



Ahora el resto de Actualizaciones

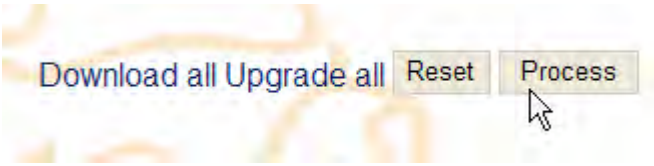
1. Go to Module Admin, press check for online updates, and upgrade the **FreePBX Framework** module from the online repository. Do not upgrade other modules. **DO NOT** press Apply Configuration Changes

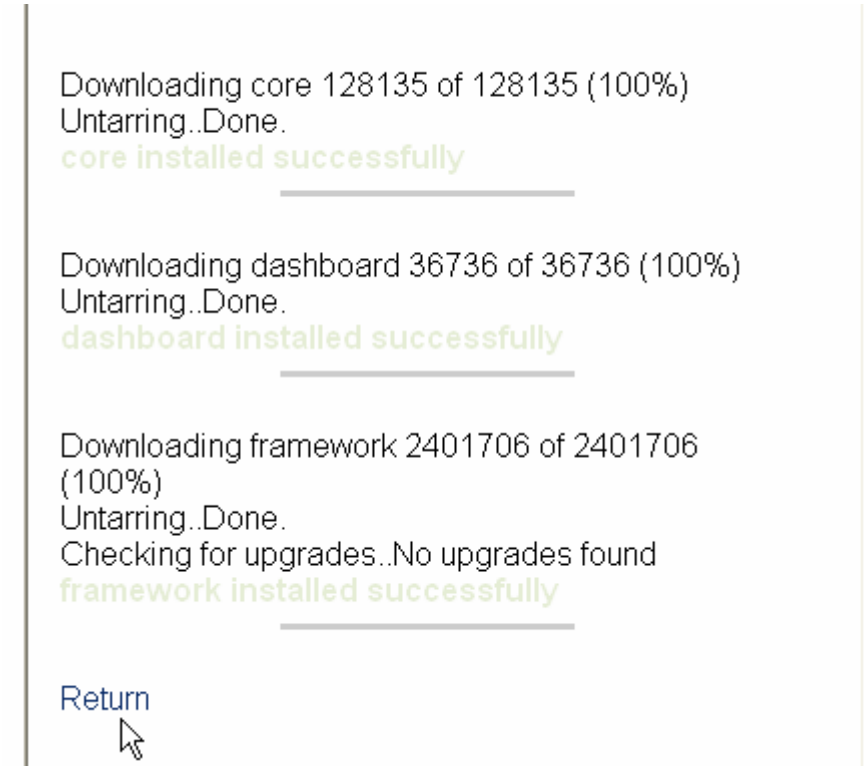
Manage local modules Show only upgradable

Download all Upgrade all Reset Process

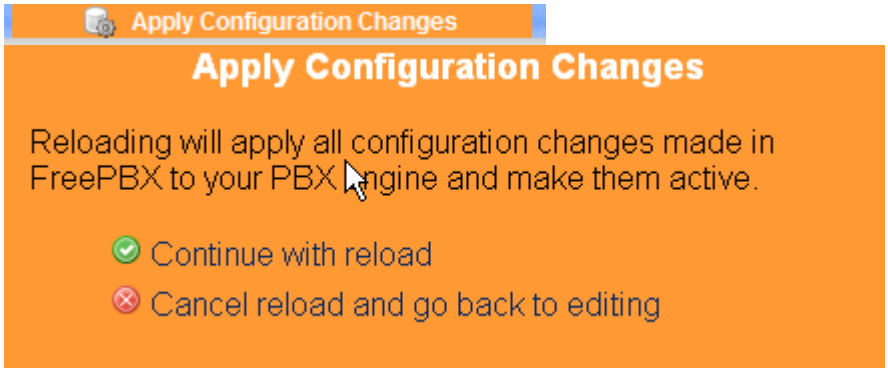
Module	Type	Version	
Basic			
2.4 Upgrade Tool	tool		Not Installed (Available online: 2.3.0.1)
Builtin	setup		Enabled; Not available online
Core	setup	2.3.1.0	Online upgrade available (2.3.1.4)
Feature Code Admin	setup	1.0.5.3	Enabled and up to date
FreePBX Framework	setup	2.3.1.0	Online upgrade available (2.3.1.3)
System Dashboard	tool	0.3.3.1	Online upgrade available (0.3.3.2)
Voicemail	setup	2.0.3.4	Enabled and up to date
CID & Number Management			

Es importante actualizar todos los módulos del FreePBX del apartado "Inbound Call Control"





Ahora ya podemos APLICAR LOS CAMBIOS



Planilla de Datos da Instalación de Trixbox CE

É unha boa medida apuntar tódolos parámetros empregados na configuración do sistema, iso facilitará o seu mantemento e xestión. Como axuda nesta tarefa proponse encher os datos na planilla seguinte:

Rexistro de Hardware

Fabricante de Pc e modelo

Modelo: ..... Nº serie: ..... Identificador:	Web Fabricante: http://www..... Data compra: ..../..../....	□ Portátil □ Torre/Sobremesa

Características do PC:

Bios (fabricante e versión):	Chipset: .....

Procesador	Velocidade interna	Velocidade FSB
	..... MHz	..... MHz

RAM instalada:

	Tipo de RAM	Tamaño (MB)	Tipo de ranura	Velocidade Reloxo
Modulo 1				..... MHz
Modulo 2				□ Síncrona
Modulo 3				□ Asíncrona

Total: ..... GB

Buses ATA

		Dispositivo Master	Dispositivo Slave	Configuración
PATA	Bus Primario	□ HDD ..... GB □ DVD / CD .....x □ DVR ... R / CD-RW ...x / ...x	□ HDD ..... GB □ DVD / CD .....x □ DVR ... R / CD-RW ..x / ..x	□ ATA .... □ PIO ....
	Bus Secundario	□ HDD ..... GB □ DVD / CD .....x □ DVR ... R / CD-RW ...x / ...x	□ HDD ..... GB □ DVD / CD .....x □ DVR ... R / CD-RW ..x / ..x	□ ATA ... □ PIO .....
SATA	Bus Primario	□ HDD ..... GB □ DVD / CD .....x □ DVR ... R / CD-RW ...x / ...x	□ HDD ..... GB □ DVD / CD .....x □ DVR ... R / CD-RW ...x / ...x	
	Bus Secundario	□ HDD ..... GB □ DVD / CD .....x □ DVR ... R / CD-RW ...x / ...x	□ HDD ..... GB □ DVD / CD .....x □ DVR ... R / CD-RW ...x / ...x	

Tarxeta Gráfica

Nome / Modelo	Memoria	Conector saída	Tipo de ranura
Web fabricante: http://www.....	□ Integrada ..... MB □ Compartida	□ DVI □ DB □ RCA □ S-Vídeo	□ Vesa □ Integrada en placa □ PCI □ PCI Express □ AGP x

Teclado		**Rato**	
Nº serie: ..... Fabricante: ..... Modelo:	Conector: □ PS/2 □ DIN □ USB □ Integrado □ IrDA □ Inalámbrico	Nº serie: ..... Fabricante: ..... Modelo: ..... Tipo: □ bóla □ óptico	Conector: □ PS/2 □ Serie □ USB □ Integrado □ IrDA □ Inalámbrico

Tarxetas:

	Fabricante-Modelo		Páxina web
□ Placa Base			
□ Tarxeta de Rede	□ PCI □ Inalámbrica □ Integrada □ Externa	Ethernet MAC:	
□ Tarxeta de son	□ Integrada □ En ranura	□ minijack. Nº..... □ Ópticas. Nº	
□ Outras:			
□			

Ranuras de expansión	Nº	Portos E/S	Nº
□ PCI □ PCI Express □ AGP ...x □ ISA □ VESA □ Outras:		□ USB 1.1 / 2.0 □ Paralelo □ Serie □ Firewire □ PCMCIA □ Outros:	

Tarxetas VoIP	Ranura	Portos	Modelo/Marca	Páxina web
□ Tarxeta POTS	□ PCI □ PCI Express	□ FXS Nº:..... □ FXO Nº:		
□ Tarxeta RDSI PRI	□ PCI □ PCI Express	Nº.....		
□ Tarxeta RDSI BRI	□ PCI □ PCI Express	Nº.....		
□ Tarxeta GSM	□ PCI □ PCI Express	SIM Nº		
□	□ PCI □ PCI Express			

Outros accesorios do Pc:

□ Altofalantes	□ Tarxeta sintonizadora TV	□ .....
□ Micrófono	□ Web Cam	□ .....
□ Módem	□ Monitor LCD/CRT	□ .....

Rexistro da Configuración do IP-PBX

Trixbox PF/CE Versión:.....

Aplicación/Dispositivo	Nome de usuario	Contrasinal
CentOS	root	
Trixbox	maint	
FreePBX	admin	
MySQL	asteriskuser	
	root	
Flash Operator Panel		
Servidor Asterisk amp	admin	
Meetme	tim@localhost	
	wmm@localhost	
Firewall/router		
BIOS PC		
Proxy		
dyndns.org		
gmail		

Configuración de Rede

IP-PBX	Acceso a Internet	Dirección IP	Tipo de IP	IP/Mascara	Porta de enlace: ..... DNS1:..... DNS2:.....
□ En LAN □ Monoposto	□ Sen Acceso □ Acceso con NAT □ Acceso sen NAT	Local	□ Dinámica □ Estática	...../.....	
		Externa	□ Dinámica □ Estática	...../.....	□ Proxy □ Servidor STUN

Configuración de Firewall/NAT

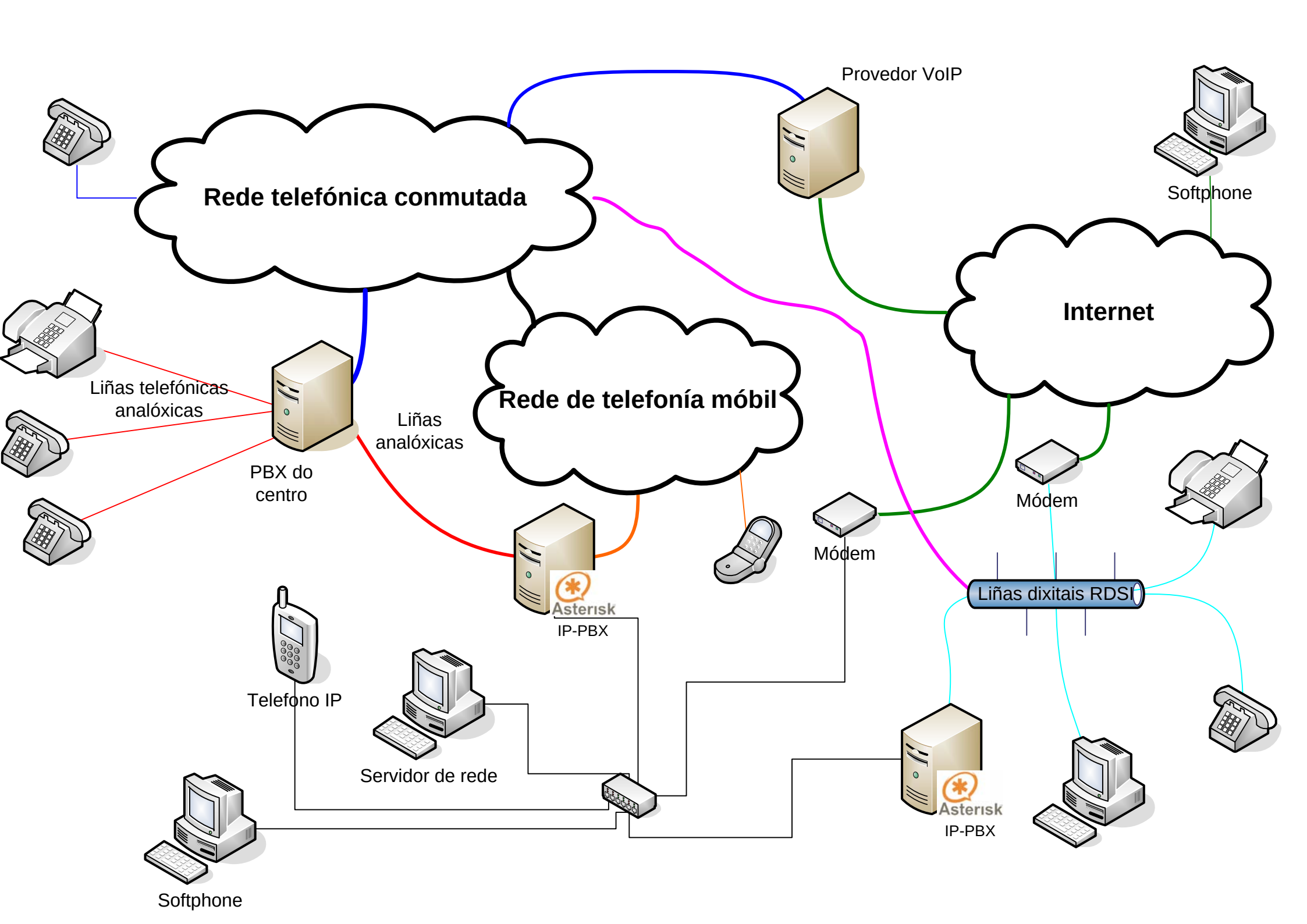
Servizo	IP orixe/externa	P. UDP/TCP orixe/externos	IP destino/local	P. UDP/TCP destino/locais
IAX				
SIP				
RTP				
SSH				
HTTP				

Acceso Inalámbrico

BSSID	Cifrado	Contrasinal	Canal	802.11	Observacións
	□ Ningún □ WEP □ WPA.....			□ a □ b □ g	

Servizo DHCP en CentOS

Servidor DHCP	Intervalo de IPs	Mascara
□ Sí □ Non	.....-.....	.....


Provedor VoIP
Softphone
Rede telefónica conmutada
Internet
Rede de telefonía móbil
Liñas telefónicas analóxicas
PBX do centro
Liñas analóxicas
Módem
Módem
Liñas dixitais RDSI
Asterisk
IP-PBX
Telefono IP
Servidor de rede
Asterisk
IP-PBX
Softphone