User’s manual

Version 2.2-Rc4

1. Preface

1.1 The history

Asterisk is an open source software application designed to run on a standard PC under Linux environment. It can vest in a PC the typical features of an hybrid PBX (RTG-VoIP). Mark Spencer, an American computer engineer, wrote the first release of Asterisk to promote Digium's hardware interfaces and choosing a GPL license he tickled up the attention of a lot of users.
Today, Asterisk is a milestone in VoIP. The completeness and the reliability of the software make it an ideal platform for a wide range of applications and it's often used as building-block in a lot of complex solutions (RTG/IP PBXs, Centrex systems, Call Centers management software, and still more.

1.2 The easyAsterisk software

Asterisk is managed through a set of text files whose compilation is not very intuitive and can take time also for highly experienced people. **easyAsterisk** is a package that offers a totally automated installation process and a management portal that can be used to configure and maintain Asterisk's configuration files. The management portal is splitted in two sections: "Admin", normally used by system administrators, and "User", where each end user can manage his own telephone extension adding and configuring services. **easyAsterisk** is today available in two sets, free and professional.

1.3 free easyAsterisk

Free easyAsterisk is tailored to cover the standard requirements of the middle-small sites. It allows to access all the features of **easyAsterisk** and it can handle every type of standard Telco interface (Analogical, basic ISDN, primary ISDN) and IP trunks; using free **easyAsterisk** you can setup a complete hybrid PBX RTG/IP for small enterprise and create a little, private telephone net by interconnecting two free **easyAsterisk** to communicate in VPN for free. free **easyAsterisk** capabilities are limited to a maximum of 5 phone extensions and 2 trunks. Free **easyAsterisk** can be freely downloaded as an ISO image from http://www.easyasterisk.org/.

1.4 easyAsterisk Professional

easyAsterisk Professional is targeted to the professional market: designed to manage the most complex installations, it is equipped with a technical support service directly provided by the authors of **easyAsterisk**. **easyAsterisk** Professional can manage complex requirements, advanced services and high number of users. The technical support carried with **easyAsterisk** Professional let the user to access all the technical sources and software updates, lifetime. Obviously, **easyAsterisk** Professional lets to activate a boundless number of extensions, Phone VPN, VoIP carrier, outbound routing, and so on.

2. Architectures

easyAsterisk can be used to deploy advanced logical configurations to cover pratically all user's requirements. Some of these architectures let the user to exploit a lot of existing hardware (PBX, telephones, answering machines, and so on) adding easyAsterisk's advanced features without hardware replaces.

2.1 Hybrid PBX RTG/IP

easyAsterisk was developed to replicate the standard PBX behaviour adding on top IP trunks and advanced performances (voicemail, IVR, meetme rooms, and so on).
This kind of solution is perfect for any new settlement, when existing hardware recovery isn't a mandatory goal and it's necessary to keep global costs within a reasonable range.
Phone terminals (IP phones) are connected to easyAsterisk via LAN and easyAsterisk can be configured to manage both the traditional TELCO lines (analogical or ISDN) and new IP connectivity trunks.

Figure 2.1 – Hybrid PBX RTG/IP

2.2 Satellite IP

"Satellite" architecture helps to recover existing hardware. This kind of solution needs the existing PBX to be equipped with a dedicated digital interface (usually called ISDN S0 Bus) to connect to the easyAsterisk machine. Then, the existing PBX needs to be "reprogrammed" in order to use the S0 Bus - in red in the figure - to divert to easyAsterisk outgoing IP addressed calls.
This kind of solution can be deployed when it’s mandatory to recover existing hardware or when the PBX isn't expandable.

Figure 2.2 – PaBX Satellite IP

2.3 IP Router

"IP Router" is totally transparent for the PBX that remains unaware of easyAsterisk’s existence. One or more outgoing lines are simply "cut" and diverted to easyAsterisk. The figure shows the simplest setting: easyAsterisk machine is equipped with coherent RTG cards to manage "incoming from PBX" calls (the PBX continues to address outgoing calls toward Telco lines) and diverts calls to IP trunks. The same schema can be used to manage IP incoming calls (the PBX will receive each call as if it was coming from standard Telco lines).
In this way all hardware premises underlying the PBX is totally recovered. Finally, the PBX can preserve a backup connection with the standard TELCO lines for temporary backup purposes (Internet failure).
IP Router is also the configuration that allows the user to build private VPN to talk between company's branches almost for free.

Figure 2.3 – Router IP

3. Installation

3.1 Obtaining the software

easyAsterisk is distributed as a self installing bootable CD, or as an ISO image ready to burn with your favourite burning software. The ISO image is available for the download on the internet site http://www.easyasterisk.org. The installation process is totally automated and pratically requires no interaction with the end user. When booted up, easyAsterisk's CD will automatically install the operating system (CentOS), Asterisk and all its components (compiling when needed the source code included in the installation CD), and our WEB management portal (including all related software such as php, apache server and mysql).
You have to remember that the installation process **will automatically partition and format the hard disk; this means that all the previously stored data will be lost.**

3.2 Hardware requirements

easyAsterisk requires a standard PC equipped with 32 bits processor, hard disk starting from 2 Gbytes minimum and CD-ROM reader. PC hardware requests, for what concerns Cpu and Ram, may depend on installation goal (number of contemporary voice channels, number of trunks and so on). Today's hardware (let's say starting from Pentium IV) is normally able to support even the greater deployments.

Note: some users have reported problems on PC equipped with Pentium 4 with Hyper Threading technology activated (BRI, PRI and Analogical telco interfaces management). We therefore recommend to disable Hyper Threading before installing easyAsterisk.

3.3 Start up

It’s obviously mandatory to cofigure the host system to boot from CD (generally through the BIOS options of the machine, priority to the BOOT from CD) and then reboot the machine. At the end of the boot process the screen will show the *figure 3.1*, which gives general warnings about the installation process (the hard disk will be formatted; the final default IP/Netmask will be 192.168.0.1/255.255.255.0).

Figure 3.1 – Installation boot up

Please press <*enter*> to begin the installation; first of all, it will be necessary to configure the keyboard layout. *(figure 3.2).*

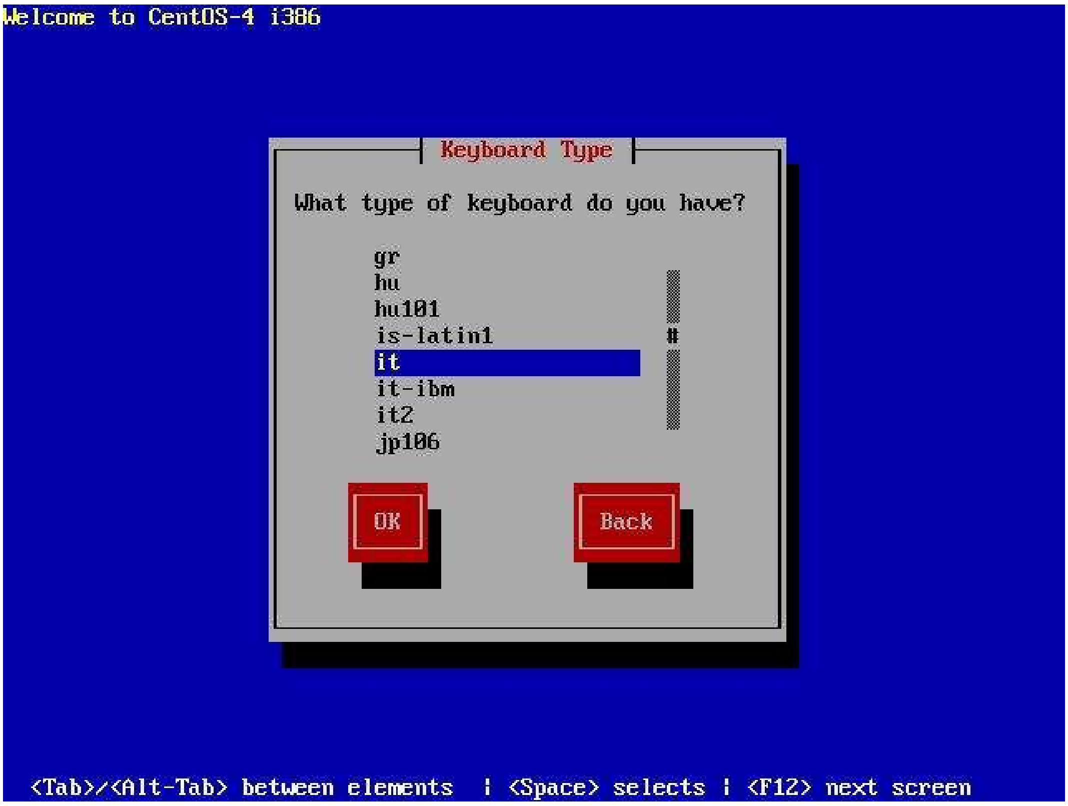


Figure 3.2 – Keyboard layout configuration

Then declare the desired root password *(figure 3.3).*

Figure 3.3 – Root password configuration

After that, the filesystem will be formatted and the installation will begin *(figure 3.4).*

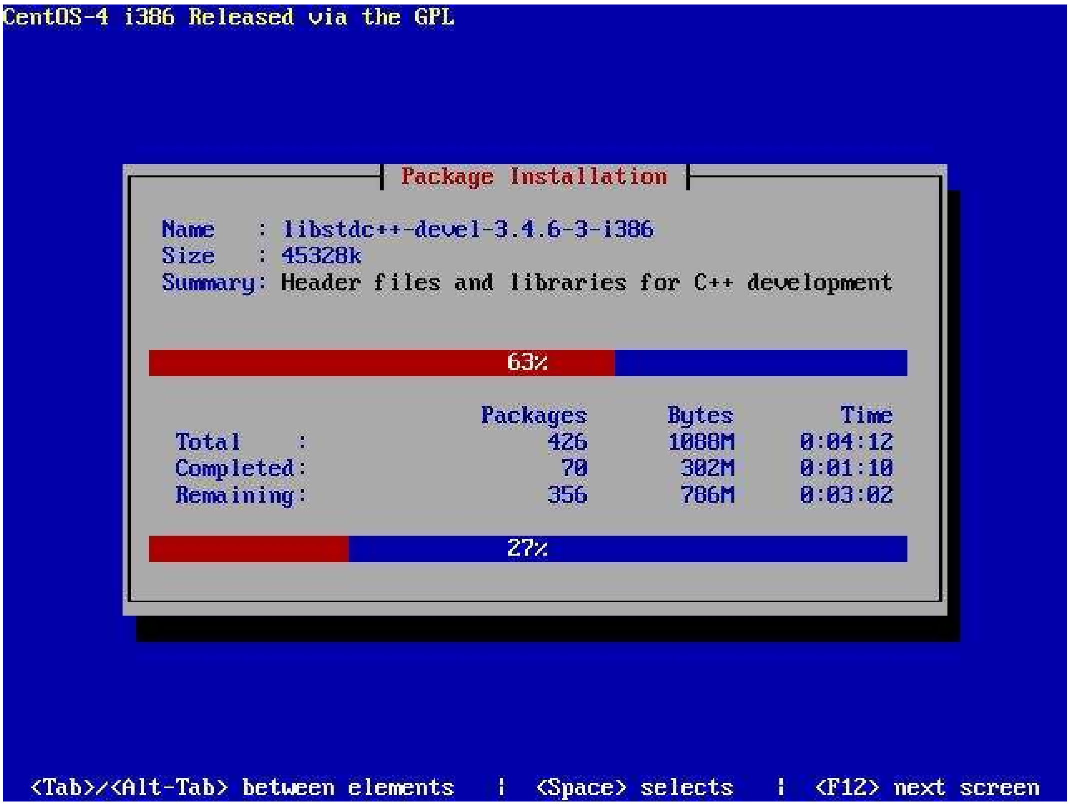


Figure 3.4 – Package copy on hard disk

When the formatting process ends the system will be rebooted (please remember to remove the CD to avoid a new CD reboot) and easyAsterisk's installation second step will begin *(figure 3.5).*
The installation script needs easyAsterisk's CD to complete the process; so it will prompt the user to re-insert it.

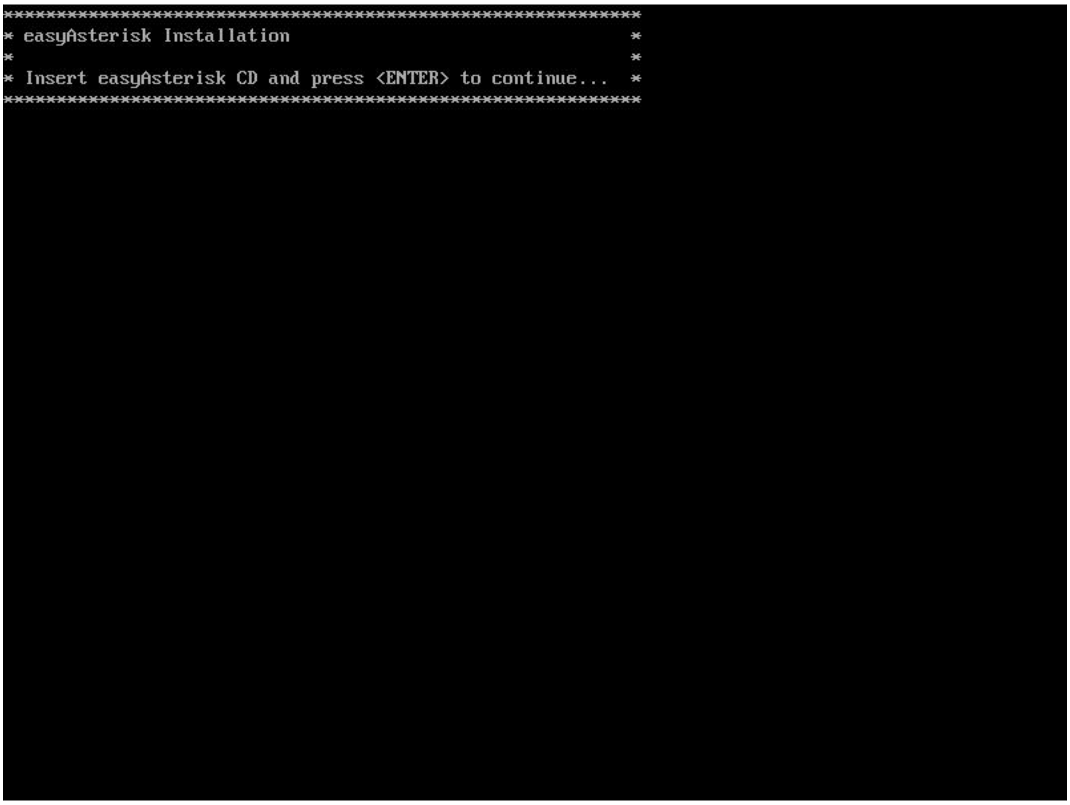
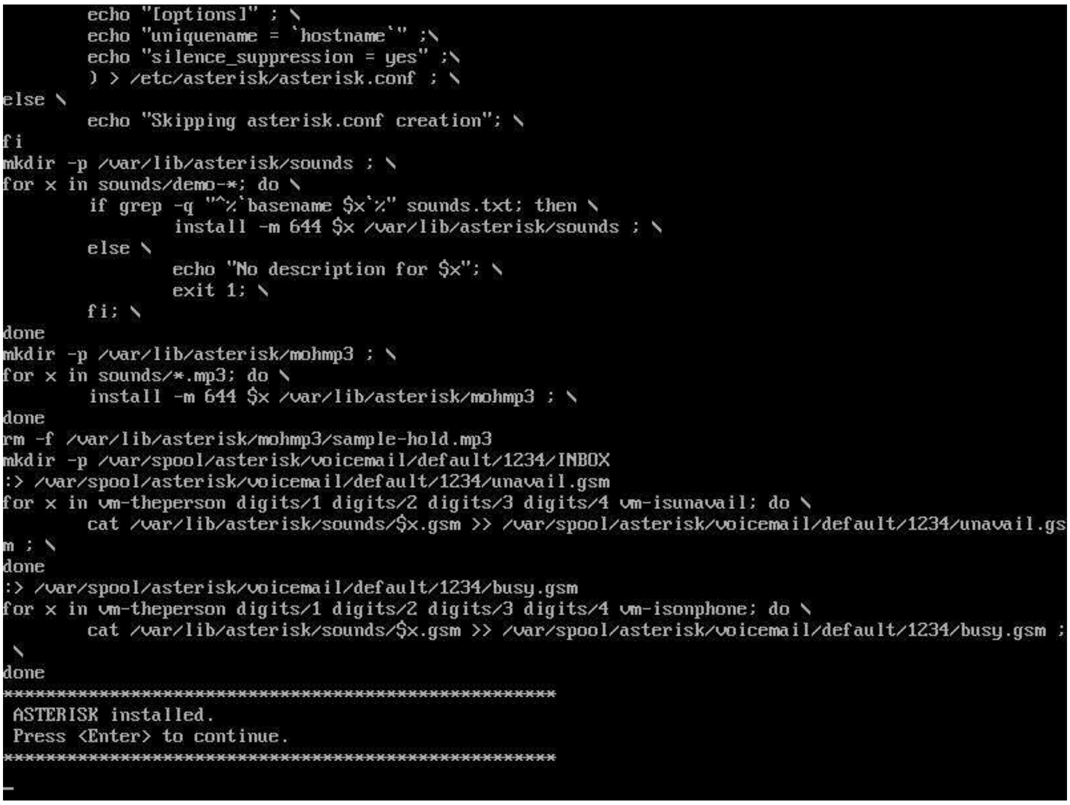


Figure 3.5 – Second installation step

Some Linux elements will be updated and configured, various Asterisk components will be compiled (this phase could take some time according to the power of the system in use). The user will be prompted to press *<enter>* key to proceed with the next step *(Figure 3.6).*



```
        echo "[options]" ; \
        echo "uniquename = `hostname`" ;\
        echo "silence_suppression = yes" ;\
        ) > /etc/asterisk/asterisk.conf ; \
else \
        echo "Skipping asterisk.conf creation"; \
fi
mkdir -p /var/lib/asterisk/sounds ; \
for x in sounds/demo-*; do \
        if grep -q "^%`basename $x`%" sounds.txt; then \
                install -m 644 $x /var/lib/asterisk/sounds ; \
        else \
                echo "No description for $x"; \
                exit 1; \
        fi; \
done
mkdir -p /var/lib/asterisk/mohmp3 ; \
for x in sounds/*.mp3; do \
        install -m 644 $x /var/lib/asterisk/mohmp3 ; \
done
rm -f /var/lib/asterisk/mohmp3/sample-hold.mp3
mkdir -p /var/spool/asterisk/voicemail/default/1234/INBOX
:> /var/spool/asterisk/voicemail/default/1234/unavail.gsm
for x in vm-theperson digits/1 digits/2 digits/3 digits/4 vm-isunavail; do \
        cat /var/lib/asterisk/sounds/$x.gsm >> /var/spool/asterisk/voicemail/default/1234/unavail.gs
m ; \
done
:> /var/spool/asterisk/voicemail/default/1234/busy.gsm
for x in vm-theperson digits/1 digits/2 digits/3 digits/4 vm-isonphone; do \
        cat /var/lib/asterisk/sounds/$x.gsm >> /var/spool/asterisk/voicemail/default/1234/busy.gsm ;
 \
done
****************************************************
 ASTERISK installed.
 Press <Enter> to continue.
****************************************************
_
```

Figure 3.6 – Components compilation

A message will appear confirming the completion of the installation process; it's now time to remove the CD and press *<enter>* to reboot the system *(Figure 3.7).*

Figure 3.7 – Last step of the installation

Fine! It's now possible to start to use easyAsterisk's management portal using an Internet browser pointing to the default IP address (http://192.168.0.1). If you need to change the default IP address you have to login as *root* user (the password has been defined during the installation) and type *setup*

[root@pbx ~] # *setup*

The command will start the tool to configure several system elements, including the network configuration *(Figure 3.8).*

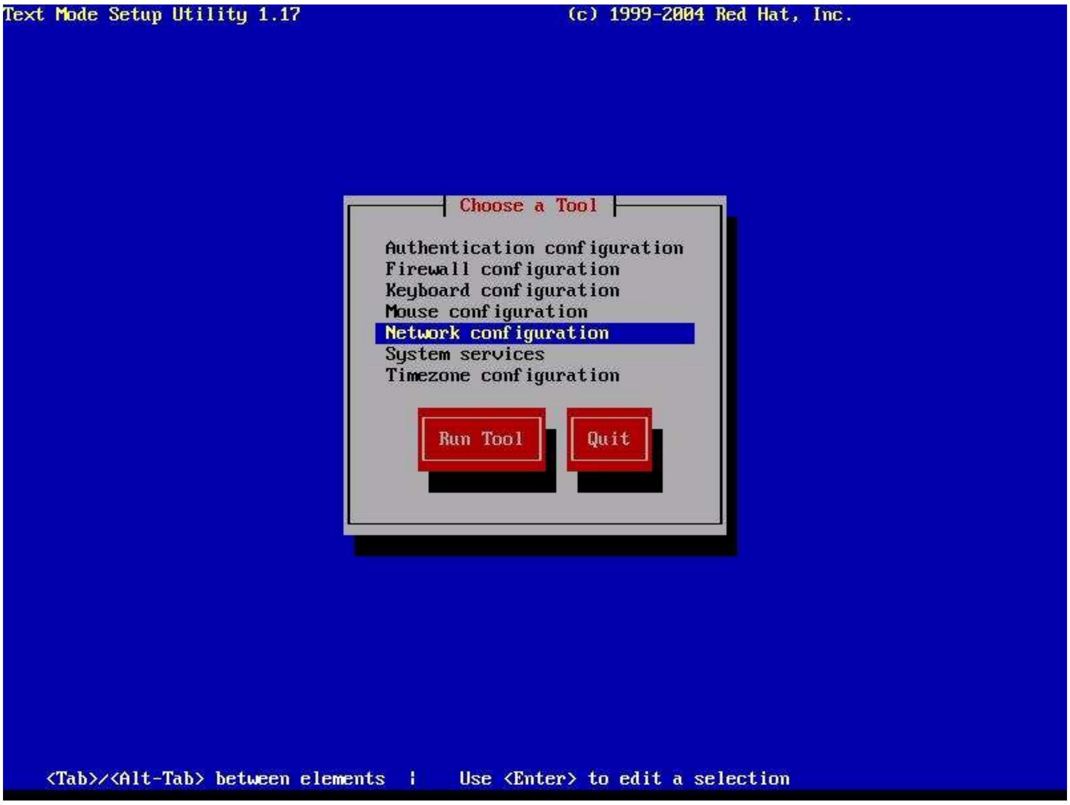


Figure 3.8 – CentOs Setup utility

As usual, please use arrows to select "Network configuration" and press <*enter*> to open the TCP/IP protocol configuration screen *(Figure 3.9).*

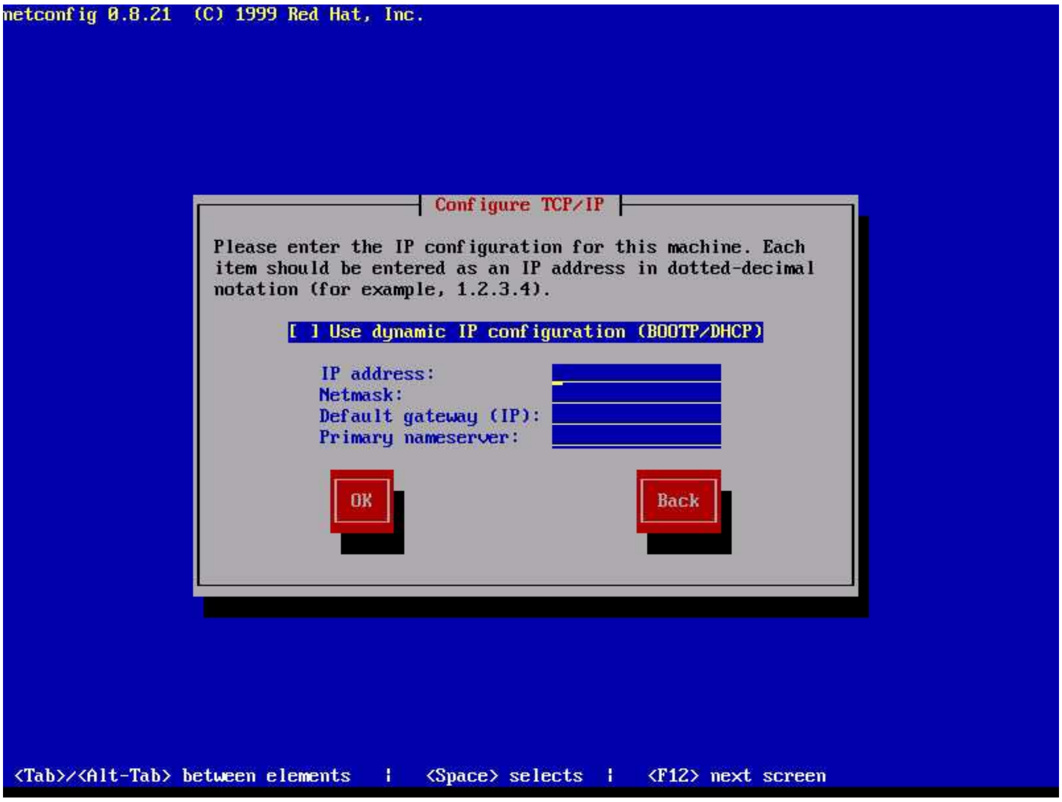


Figure 3.9 – Configure TCP/IP

Enter the desired IP address and confirm the new setting. To make the settings active you have to restart the system typing the command *reboot*

[root@pbx ~] # *reboot*

4. PBX first set-up

4.1 The administration panel

To access easyAsterisk's administration panel please open your browser and point to the default PBX IP address. A simple page will ask for "Username" and "Password" to autenticate the user. The standard settings are:

Username: **admin**
Password: **admin**

When the preferred user interface language is choosen, easyAsterisk's home page is shown:

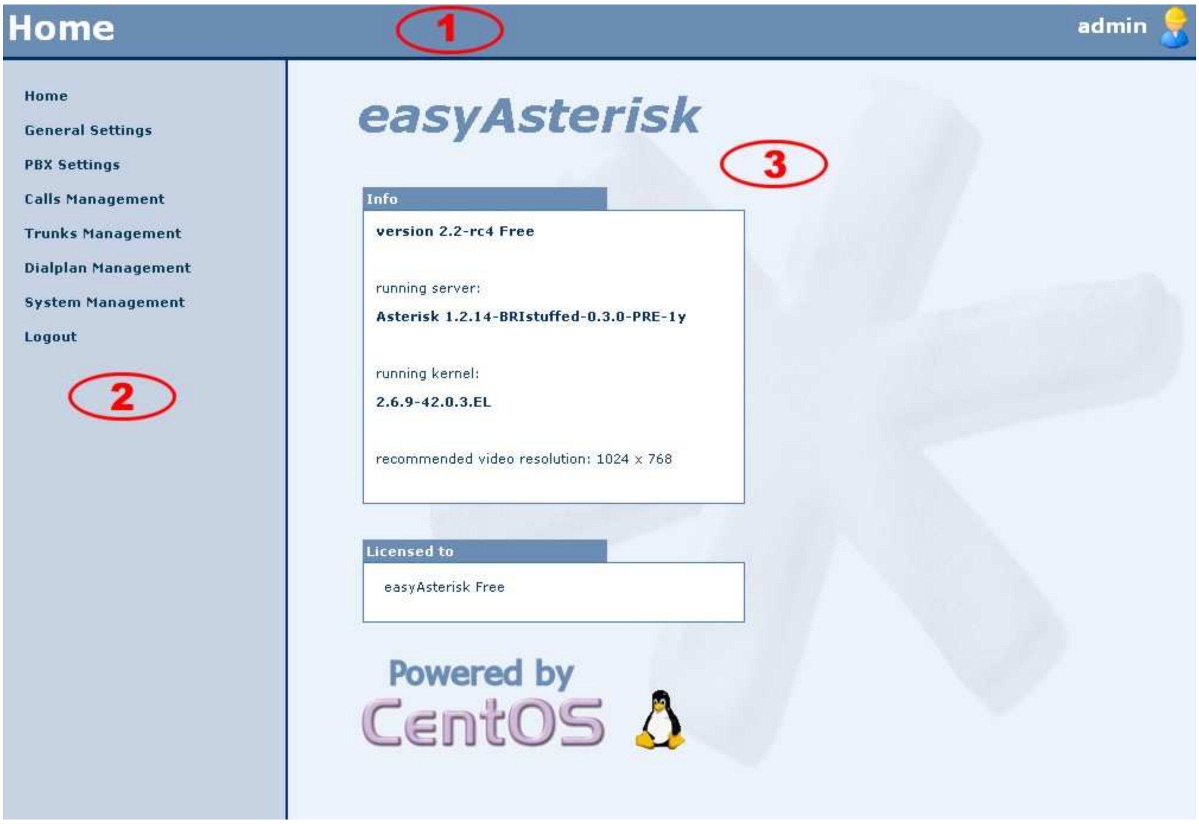


Figure 4.1 – The administration panel home page

The page is splitted into 3 sections, as shown in *figure 4.1*:

1. <u>*Title Section*</u>: The selected menù item is shown on the left side; on the right side the username of the logged in user id shown.
2. <u>*Menù Section:*</u> Located in the left part of the screen, this menu' lets the user to move among the various sections of the software. Select an entry to access the underlying submenus useful to manage the different sections of the configuration process.
3. <u>*Configuration Section:*</u> In this section the various configuration items of each submenu are shown.

In this page you can also find some general informations such as software release level, running Asterisk server release level, running Linux kernel release level, and – for the PRO version - the customer of the software licence.

We will now shortly explore the items of the Main Menù to understand better their contents.

Using ***General Settings*** you can set some PBX's global characteristics; you can, for instance, configure the supported IP protocols (SIP, IAX2 and H323), you can set up the services extensions (call forwarding, voicemail, etc.), you can manage an audio files repository (music on hold, queues, IVR Menus, etc.), you can set the Operator Panel to monitor, in real time, the PBX's elements state and finally manage a system-level address book.
PBX Settings can be used to manage typical PBX elements, such as local extensions, local groups, conference rooms and queues with relative agents, virtual fax (receipt only).
Calls Management lets you define the routing rules of incoming and outgoing calls, and LCR (Least Cost Routing) policy.
In ***Trunks Management section*** are defined some connection parameters related to IP carriers along with some configuration parameters of installed TELCO interfaces.
The ***Dialplan Management*** is dedicated to the high-skilled Asterisk user who wants to merge easyAsterisk generated dialplan with his own contexts and personalized macros.
System Management gives access to some general system setting related sections (hardware and software); backup management, security management, system monitoring and log files inspection.
Logout is obviously used to exit the application.

Every time you modify one or more PBX settings, a button will appear at the bottom of *Title Section* informing you that to make those changes active it is necessary to reload the system configuration (to reload you have to press the button). When you modify the configuration of some particular elements, such as *Hardware Settings*, it is necessary to reboot the Asterisk server. You will see a warning button but, once pressed, you will be prompted to confirm before rebooting the PC. Please note that, during reboot, all running phone calls are lost.

4.2 SIP protocol general settings

easyAsterisk can support SIP, IAX2 and H323 protocols. It is possible to define the general settings of these protocols through the *General Settings* menù. Let's analyze these options in detail, starting from SIP:

Port:	Bind UDP port. For SIP default port is 5060.
Bind address:	Bind IP address. (*0.0.0.0* indicates the bind to all system IP address).
Channel language:	Default language of the protocol. The default installation includes english and italian templates but it's possible to add custom templates *(Chapter 4.9)*..
Realm:	Realm to be used for the authentications procedures. It generally corresponds to host name or to the domain name. The default realm will be "*asterisk*", if not specified.
Srvlookup:	Enable the DNS SRV lookups for outbound calls. When disabled it won't be more possible to effect outbound calls based on domain name.
Maxexpirey:	Maximum duration, given in seconds, for incoming recordings. Default *3600 (1 hour)*.
Defaultexpirey:	Duration of incoming and outgoing recordings, default measure second. Default *120*.
User Agent:	Allows to set the string "User-Agent" used in protocol headers. Default: *Asterisk PBX*.
Nat:	Enable NAT Mode. Use if clients are behind to a NAT device.
Accepts not authenticated calls:	If set allows to accept anonymous incoming calls from not authenticated users.
Tos:	Defines Type of Service. It is possible to select one of the default values or declare a personalized value.
Localnet:	Indicates the subnet address, netmask included (for example *192.168.0.0/255.255.255.0*).

ExternIP:	If Asterisk server has a private address and is behind a NAT, *ExternIp* indicates the public IP address the server uses on the big internet.
Context:	Defines the default SIP protocol context for incoming calls. It is possible to choose among one of the system contexts or to select a personalized context created via the *Dialplan Management* menù. The default value is "incoming" that sends all the incoming calls to the attendant console, or to an alternative destination defined in a DID.
Music on hold:	Define the default class of music on hold.
Checkmwi:	Interval, expressed in seconds, used to check the presence of new messages in the vocal box. On IP phones the MWI allows the notification of the incoming new vocal messages through a warning light.
Default codec:	Indicates the list of the enabled default codecs. The listing order also sets the usage priority.
Advanced:	In this section it is possible to define some advanced settings that will be inserted in the general section of the protocol's configuration files (more precisely, in *"/etc/asterisk/sip-general.conf"* file). The correct syntax for every line is: **key=value**

4.3 IAX2 protocol general settings

Let's now analyze the general settings related to the IAX2 protocol:

Port:	Bind UPD port. For IAX2, default port is 4569.
Bind address:	Bind IP address. (*0.0.0.0* indicates the bind to all system IP address).
Channel language:	Default language of the protocol. The default installation includes english and italian templates but it's possible to add custom templates *(Chapter 4.9)*.
Delayreject:	Enable this option to prevent brute force attacks. It will be delayed the sending of some packets, such as "authentication reject", for the authentication applications.
Tos:	Defines Type of Service. It is possible to select one of the default values or introduce a personalized value.
Jitterbuffer:	Enables or disables the *Jitterbuffer*, that is a portion of memory where are allocated the incoming packets or the outbound packets, with the purpose to compensate possible net latencies and to maximize the audio quality. If Jitterbuffer is enabled, you can configure a set of its particular parameters.
Dropcount:	Maximum frames number that can be eliminated in the last 2 seconds for supply to possible net latencies.
Maxjitterbuffer:	A maximum size for the *Jitterbuffer*.
Maxexcessbuffer:	In case of latency problems *Jitterbuffer* could increase over the necessary. If the maximum threshold overcome the planned value in *Maxexcessbuffer* Asterisk will arrange to bring back the buffer to a value in the norm.
Minexcessbuffer:	Sets up the least value of the *Jitterbuffer* free space. If the free space had to increase over this value, Asterisk will enhance the total of the *Jitterbuffer*.
Jittershrinkrate:	Number of milliseconds required to increase or to decrease the *Jitterbuffer*. It is convenient use short values.
Context:	Define the default context for IAX2 protocol. Logic is the same described for the SIP protocol.
Default codec:	Indicates the list of the default codecs enabled. The order with which the various codecs are listed also represents the priority with which they will be

	used.
Advanced:	Advanced options for IAX2 protocol. Logic is the same described for the SIP protocol.

4.4 H323 protocol general settings

In this release we apply the H323 protocol developed by Jeremy McNamara; let's see its general settings:

Port:	Bind UPD port. Default port value is 1720.
Bind address:	Bind IP address. Unlike SIP and IAX2, it is recommended (mandatory) to specify a valid IP address and not to leave the value "0.0.0.0", because this might cause the protocol malfunction.
Tos:	Defines Type of Service. It is possible to select one of the default values or introduce a personalized value.
Dtmf:	Indicate the type of default Dtmf tones for the protocol.
Context:	Define the default context for H323 protocol. Logic is the same described for the SIP protocol.
Default codecs:	Indicates the list of the enabled codecs. The listing order also represents the codec's usage priority.
Advanced:	Advanced options for H323 protocol. Please consider the same logic described for the SIP protocol.

4.5 Extensions management

easyAsterisk can now support SIP/IAX2 capable IP phones. Standard, analogical phones can be used installing special adapters, or using an analogical card (e.g.: Digium TDM400P with FXS modules). PBX local extensions are managed through *General Settings* menù in *Local Extensions* item (*figure 4.2*). In this page the user can manipulate a table containing the list of the previously configured local extensions. As you can note, a default test local extension (100) is shown: that is the attendant console (the dafault termination of all incoming calls – please see the chapter *INCOMING CALLS MANAGEMENT* of this manual).

Some operations are possible:

- **Create a new extension:** press the "*add*" button to access the page where the new extension properties can be defined.
- **Clone an existing extension:** allows to create a new extension maintaining some characteristics of an existing extension. Useful when extension shares some properties (for instance, outbound calls permissions). Use "*Copy from…*" button and select the extension that you want to clone.
- **Change an existing extension properties:** Simply click on the icon or on the extension number that you want modify to access the properties extension's page.
- **Delete an extension :** Click on the icon to delete an extension.

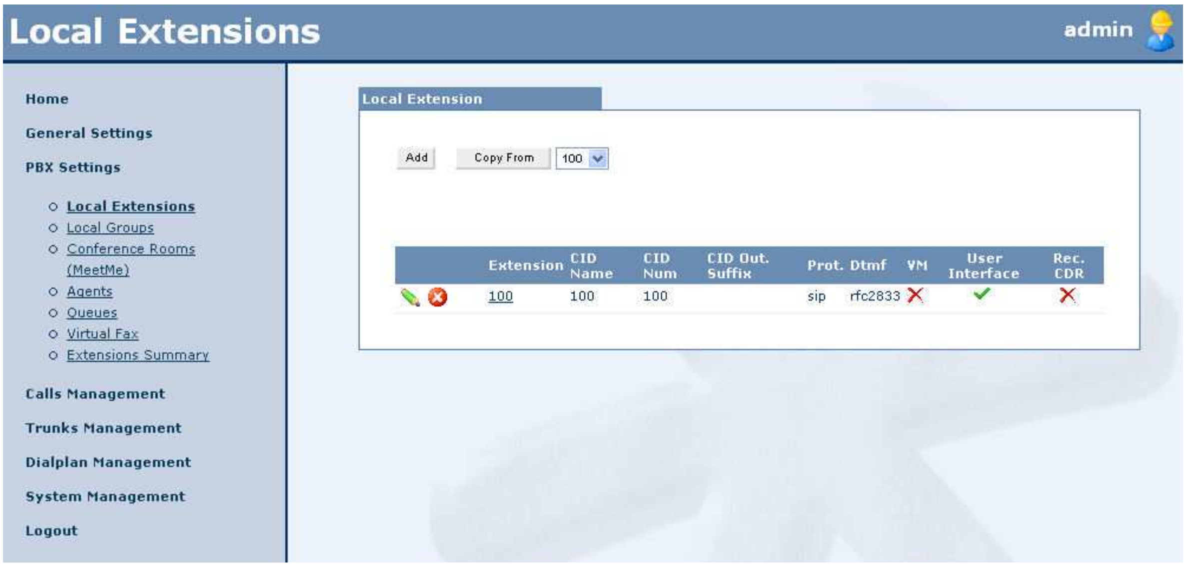


Figure 4.2 – Local extensions management

When adding a new extension you will have to define some parameters to set the new extension properties *(figure 4.3)*.

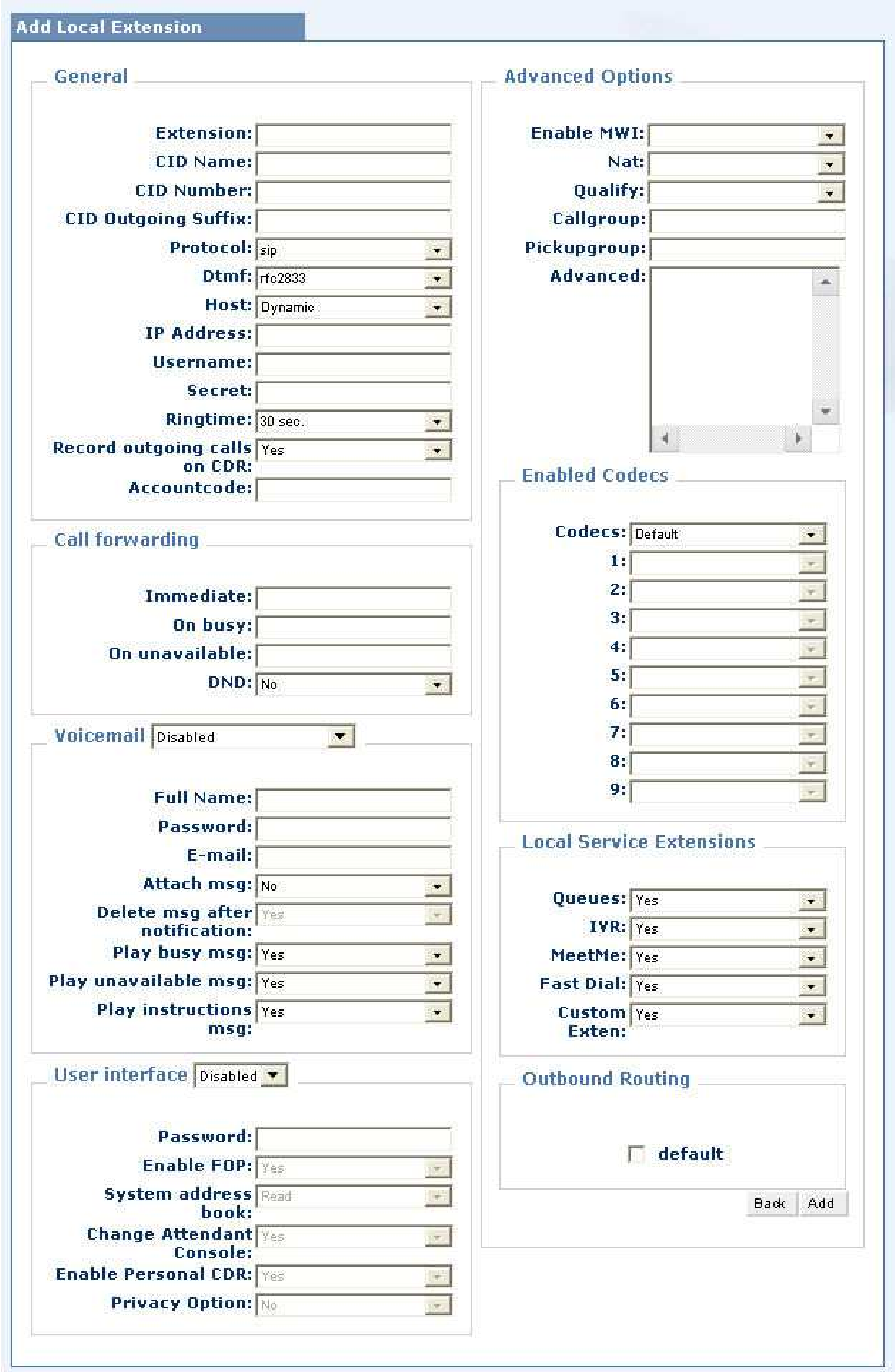


Figure 4.3 – Creation of a new local extension

GENERAL:	
Number:	Local extension's number.
CID Name:	Caller ID in alphanumeric format. It will be visualized on telephone display that support this functionality.
CID Number:	Caller ID in numeric format, generally it corresponds to the extension number.
CID Outgoing Suffix:	It represents the suffix that you desire to add to the Trunk caller ID used for the outbound calls. If left blank, only the caller ID assigned to the trunk will be used. This option is useful when you stipulate a direct inward dial contract service with your telephone carrier and you desire that some local extensions are shown on the public network with their full DID number rather than with the reduced GNR.

Protocol:	Setup the used protocol. If you are configuring an “IP” extension, this value can be SIP or IAX2. If FXS boards are installed, it is possible to choose ZAP (analogical telephones) with the corresponding board-configured hardware channel.
Dtmf (only SIP):	Setup the Dtmf type tones used by the telephone client in use.
Host:	Lets to chosse between static or dynamically assigned IP address. When static, the address must be declared in “*IP Address*” field. When dynamic, it is necessary to choose a username/password pair *(secret)* that should be used for server subscribing.
Ringtime:	Timeout, in seconds, to activate unanswered call policies (voicemail activation, call forwarding, etc).
Record Outgoing Calls on CDR:	Enables the calls recording into a Sql database.
Accountcode:	Account code that will be recorded in CDR for outgoing calls coming from this extension.
CALL FORWARDING:	Shows and sets up the calls forwarding and the DND management. If a call forwarding is enabled the destination extension will appear in this mask.
VOICEMAIL:	Voicemail is an answering machine that allows to record messages in audio files and can be activated for each extension with different activation rules (busy number, unavailable, not responding…). The user can heard the recorded messages later (using the phone, or remotely using easyAsterisk’s dedicated interface). It’s also possible to send audio files as e-mail attachments, with various rules. This field is mandatory also if voicemail is disabled (you can use dummy data).
Full name:	User name that will be reported in the notification e-mail when message recording occurred.
Password:	Access password to the answering machine. User name corresponds to extension number.
Attach msg:	If this option is enabled, a wav file, with the recorded message, will be attached to the e-mail.
Delete msg after notification:	Delete the message from the answering machine database after notification via e-mail.
Play busy msg:	Play a message to inform the caller that the user is busy in an other conversation. It is possible customize this message for each user.
Play unavailable msg:	Play a message to inform the caller that the user is unavailable. It is possible customize this message for each user.
Play instructions msg:	Play an intro/usage message to the caller.
USER INTERFACE:	Enables the user panel. User’s login is similar to the admin’s login (point a browser to server’s IP address).
Password:	Password used to access the User panel. Normally, user name corresponds to the local extension number.
Enable FOP:	Enables the user to access to Flash Operator Panel.
System address book:	Enables the user to READ od READ AND WRITE the system address book.
Change attendant console:	Allows the user to modify the incoming calls management rules.
Enable personal CDR:	Enables the user to view the incoming and outbound calls details (related to its extension).
Privacy option:	When “Privacy option” is enabled, the last 3 digits in CDR will be hidden.

ADVANCED OPTIONS:	
Enable MWI:	Enable *Message Waiting Indicator*. The option allows to receive a warning on the telephone display when a message is leaved in the answering machine. This service can work if the user phone can support the feature and is correctly configured.
Nat:	Enable this option when the client is behind a firewall.
Qualify:	When enabled, Asterisk will periodically check the state of the client. Generally this option is used to keep an UDP session open when clients are behind a nat.
Callgroup:	Pickup groups, listed as a numerical, comma-separated list, whose the local extension belongs. Each item of the list represents a group predefined as a dedicated extension in *"Service Extensions"* / *"General Settings"*.
Pickupgroup:	Groups on which the local extension is enabled to pickup calls.
Advanced:	Advanced options to insert in *"/etc/asterisk/sip-exten.conf"*. The correct syntax for each row is: **key=value**
ENABLE CODECS:	
Default:	Uses the standard settings configured in general protocol settings.
Customize:	Customize the codecs settings.
LOCAL SERVICE EXTENSIONS:	Enables or disables user access to service extensions. Among these, *"Custom Exten"* allows access to *"custom-exten"* context, used by experienced administrators to configure custom extensions.
OUTBOUND ROUTING:	Sets client permissions on outbound calls (for each route and routing).

4.6 Voicemail general option

In this section it is possible to set up some options for voicemail services.

Max message length:	Maximum message duration, expressed in second.
Min message length:	Minimum message duration, expressed in second. When a message duration is shorter than this value that message will be automatically erased from the answering service without any notification to the user.
Playback allowed:	Allow the caller to play the recorded message back before hanging up. If set, lets also the user to delete the leaved message.
e-mail server:	e-mail address that will be used as sender for notification e-mails.
e-mail user name:	User name that will be used as sender for notification e-mails.
VoiceMail Main Extension:	Extension to dial to enter the answering system management service, through witch it is possible to play and erase messages.
e-mail language:	Choose the default language for the notification e-mail.
Maximum number of message:	Maximum number of messages that can be stored in a mailbox.

4.7 Service Extensions

In this section it is possible to configure the numerical codes used to enable or disable some services. Let's briefly analyse them:

Attended transfer:	Allow the call transfer with announcement. After dialing this code, the system will ask for the destination extension using a pre-recorded message.
Blind transfer:	Allow the call transfer without announcement. After dialing this code, the user must immediately enter the destination extension.
Recall:	Allows to recall a call during the transfer process. Useful if the destination user doesn't replies.
Direct pickup:	Allows the pickup of an extension-terminated call. When enabled, to pickup a call the user needs to enter the pickup code followed by the ringing extension.
Group pickup:	Allows the pickup of a group-terminated call. The user needs to enter the pickup code followed by the group extension. If more than one extension in the group is ringing this function can have unpredictable results.
Call forwarding:	If enabled, lets the user to configure a follow-me service. The forwarding process can be immediate, on busy, on unavailable (intended after the delay defined in *ringtime*). To set up a forwarding the user needs to type the forwarding code followed by the destination extension (it's also possible to configure external or mobile numbers). A beep indicates that the follow-me is correctly set up. Simply type in the activation code and wait for the beep to remove the service.
DND (Do Not Disturb):	Enabling "DND" a phone will not receive calls. To enable or disable the service, simply type in the activation code and wait for the beep.

Please note that the greatest part of the standard IP phones also let the user to set up follow-me services through their internal management software.

4.8 Audio section: repository and music on hold

easyAsterisk can to manage a wide set of audio files. These audio files are structured in two categories: wav files, used in queues management and IVR menu management, and *mp3* files, used during music on hold services setup. Wav files can be simply recorded using a phone or can be uploaded from another resource (files must be recorded in mono, 16 bits / 8000 Hz). Extensions used to record and playback Wav files are defined in the *Settings* submenu. The following figure explains how to manage Wav files using the phone: from *Repository* menu (*figure 4.4*) type the desired file name, compose the configured recording extension and speak after the beep (to end and playback press the # key). Press *"Add"* to write the record and the new file should appear in the files list. Use the appropriate icon to delete (), playback () or download () each file.
To upload external files use method 2: assign the name and press *"Browse…"*, to select the desired file, as usual.

Figure 4.4 – Repository audio files creation

Musi On Hold is managed through the following mask *(figure 4.5)*. Please note that the Default class can't be deleted; in this example we have a single class, *default*, with three files recorded. As usual, press *Browse…* button to choose the file to upload in the class or use the icons to delete, playback, download audio files.

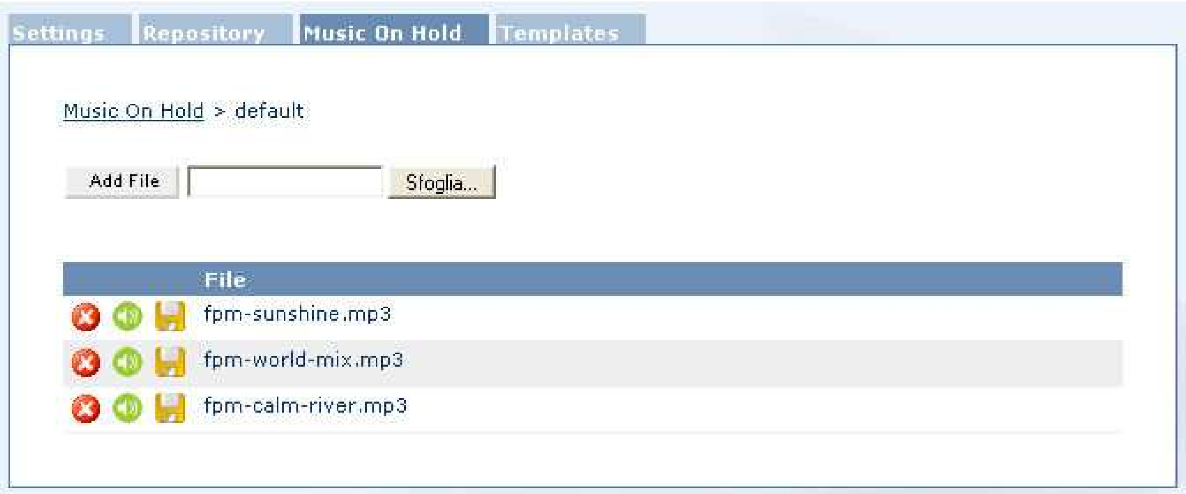


Figure 4.5 –Music on hold class changes

4.9 Audio section: templates

Asterisk uses some audio files to play some application's messages, such as voicemail and queues. English and Italian audio templates are installed by default and it's possible to add custom templates for other languages. Every template is identified by: a *Name*, a string used to identify the language in use, and a *Description*. When a template is created related audio files must be uploaded manually on the server on these directories:

/var/lib/asterisk/sounds/templatename
/var/lib/asterisk/sounds/dictate/templatename
/var/lib/asterisk/sounds/digits/templatename
/var/lib/asterisk/sounds/letters/templatename
/var/lib/asterisk/sounds/phonetic/templatename

Each "*templatename*" subdirectory is automatically created by easyAsterisk. When a template is deleted. all subdirectories are removed. Some "ready to use" audio files in various languages can be founded following this link:
http://www.voip-info.org/wiki/view/Asterisk+sound+files+international

4.10 Groups

A Group is a set of extensions with an associated identifier. When a caller dials thet identifier all the extensions included in the group will ring. The first phone that hook the phone will answer the incoming call.
For each group it is possible to define a fallback destination to define the PBX's behaviour when none of the xtensions in the group hooks the phone when the defined timeout expires. The groups are managed in *PBX Settings*/*Local Groups* menu. The following table describes the settings:

GENERAL:	
Number:	Extension associated with the group.
Ringtime:	Time in second to wait before to activate the fallback policies.
LOCAL EXTENSIONS:	The left list includes all PBX extensions; the right one only those belonging to the current group. To include or remove, simply move an extension using arrows.
FALLBACK DESTINATION:	This is the destination of the fallback policy; more details in the following configuration items.
Destination:	The ement type to activate after *ringtime*. If Hangup is selected the call will be closed and the following parameter (Detail) must be left blank.
Detail:	The destination extension to activate after *Ringtime*. Incoming calls can be router to another extension, an external number, na IVR, and so on.

4.11 Conference rooms (MeetMe)

MeetMe rooms are a sort of phone chat where multiple users can speak all together. Each room is identified by a code and can have an authentication PIN. A room can have a predefined list of allowed users each having its own properties (e.g.: an user can be configured to partecipate in a MeetMe room as a simple listener). Let’s analyze the various options:

GENERAL OPTIONS:	
Room code:	Univocal numerical code that identifies the room. Attention, this isn’t an extension.
Pin:	Optional numerical code that the user must type in to access the conference.
LINKED EXTENSIONS:	This section defines the properties associated with each extension linked to this room. Use the icons to add or remove an extension.
Extension:	Extention number.
Language:	Language to use during audiomessages playback (e.g.: “Please enter your PIN number”).
MOH:	Sets on/off the music on hold used when just one user is logged to the room. When enabled, it’s mandatory to select the desired audio class.
Actions:	Lets choose the permissions granted to the user thet enters a room from the extension being configured (only speak, only listen, both).
Quiet Mode:	When enabled, it suppress the audio warning that Asterisk sends to all logged users when an user enters or leaves the room.
Exit with #:	Enable a user to leave the conference room by pressing #.

4.12 Agents and queues

Queues is a way to keep on hold an incoming call waiting for a free operator. It’s a feature often used in callcenters. They are normally managed using FIFO criterion, in which calls are assigned to a free operator following their arrival order.
The group containing the answering operators can be static (users that constantly belongs to a defined queue) or dynamic (users that decide to partecitate to a queue logging themselves – using username and pin - to the queue for a given time. In this second case we’re talking about Agents). Agents are defined in *PBX Settings /Agents* menu.

GENERAL SETTINGS:	
Login extension:	Extension used by an agent to logon to a queue starting to receive phone calls.
Logout extension:	Extension used by an agent to logout from a queue.When requested, simply type # to disconnect.
Autologoff:	Time to wait, in second, before disconnecting an Agent that seems dead. To disable (wait indefinitely) set it to zero.
AGENTS:	
Id:	Agent identification code (must be a numerical value).
Pwd:	Password (in numeric format).
Name:	Agent name.

Queues are managed via *PBX Settings/Queues* item:

GENERAL:	
Queue name:	The name of the queue.
Extension:	Local extension to enter the queue. Optional parameter.
Prefix CID Name:	Prefix that will "mark" an incoming call as coming from a given queue. This works with phones equipped with a display, of course.
Audio on entering:	Audio message played to the caller normally for presentation purposes. It's possible to use a prerecorded message (repository).
Music on hold:	Class of audio files that callers will listen during the wait.
Ring Strategy:	Defines the ringing policy applied to queue's members: • *Ringall*, all telephones contemporarily ring until an operator answers. • *Roundrobin*, telephones ring up in turns, one at the time, until an operator answers. • *Leastrecent*, makes to ring the most-inactive operator. • *Fewestcalls*, assigns the incoming call to the operator with less completed calls. • *Random*, randomly assigns the call to an operator. • *Rrmemory*, strategy similar to the round robin, but queue starts from the last involved operator.
Timeout ringtone:	Maximum time in second for a phone to ring with no answer (timeout).
Retry:	Inactivity time before retrying to transfer the call to an operator.
Wrapuptime:	Inactivity time to wait before readmitting an operator in queue after a call completion.
Maxlen:	Maximum allowed number of queued calls. (Set to 0 for boundless queue).
Call Recording:	Enables the calls recording in audio files. To enable this function is necessary to specify one of the recording formats: • Wav • Wav49 (wav compressed) • Gsm To manage the recordings files it is necessary to access the queue management main page and select the related item. Created files names include the date and the time of the recording.
STATIC MEMBERS:	Using apposite arrows, move from the global local extensions list (left column) to the right column to define a new static member.
AGENTS:	As for static members, configure in the same way the agents for the queue.
FALLBACK:	Defines the rules applied to the calls that remains in queue, unanswered, until a timeout happens.
Timeout:	Maximum time, in minutes, that a call can wait in a queue. Set it to zero for unlimited wait.
Destination:	Sets the destination type to consider after *Timeout*, as we saw before. If *Hangup* is selected the call will terminate and the following option (**Detail**) must be blank.
Detail:	Select the destination extension.
ANNOUNCES:	
Current position:	Enabling this option Asterisk will periodically inform caller in queue about its position within the queue.
Current position frequency:	Time in seconds to wait between an announce and the next one.
Periodic	Lets select an audio file that will be periodically played to the callers

advertisement:	during their wait in queue.
Periodic Announce Frequency:	Sets the number of seconds between an advertisement and the next one.
Holdtime:	This option will cause Asterisk to inform a caller about the expected waiting time in queue.
Agent Announce:	A massage can be played to an agent just before the transfer of a call; this could be useful to inform the agent about the queue the caller is coming from (originating queue).
AUDIO MESSAGES:	Lets the experienced used to customize some advertisement default audio files. Use with caution, some system files are involved!

4.13 Virtual Fax

easyAsterisk can manage incoming faxes. Once received and stored, faxes can be viewed using via the User Panel; (*PBX Settings/Virtual Fax* item). A fax can also be forwarded via e-mail at the end of the receiving process. Unfortunately, until today Asterisk can't affordably send faxes.

GENERAL:	
Name:	Identifier of the virtual fax.
Type:	Defines the format used to save received faxes. It's possible to choose between "Tif" and "Pdf".
E-MAIL:	
Notify:	Enables (or not) the forwarding of an incoming fax via e-mail.
Name:	Sender's name of the sent forwarding e-mail.
E-mail:	E-mail address used to forward the fax.
Language:	E-mail notification language.
Server e-mail:	E-mail address visualized as e-mail sender.
Signature:	E-mail signature (text string).
Notify failed fax receipt:	When this option is enabled a notification email will be sent when the receiving of a fax fails.
MANAGEMENT ENABLE USERS:	Lets to configure the users granted to receive and see the incoming faxes. Use the same rules previously described (choose an used and move it from the left list to the right one).

4.14 Hardware settings

Asterisk can be connected to the *General Telephone Net* through a certified telephone cards (POTS, Isdn BRI, Isdn PRI). easyAsterisk lets to configure up to 3 cards on the same system choosing within the following comptibility list:

- **Digium TDM400P** (POTS card up to 4 FXS/FXO ports)
- **Digium X100P**, 1 port analogical card *FXO*.
- **Digium TE110P, TE2XXP, TE4XXP**, 1/2/4 port **PRI** card, compatible with T1/E1 protocols.
- **Junghanns QuadBri**, 4 ISDN **BRI** ports.
- **Junghanns QuadGSM,** GSM PCI adapter (1, 2 or 4 ports).
- Generic 1 port **BRI** card with chipset **HFC-S** (very cheap).

Please take care of some limitations: Digium X100P can't be installed with other adapters and up to 2 generic 1 BRI are allowed at the same time. All installed interfaces can't share the IRQs with other peripherals. To visualize the IRQs assignment type in by console:

[root@pbx ~] # *cat /proc/interrupts*

Let's now examine the parameters to configure the various cards (*System Management* menu, *Hardware Settings* item). Please note that some of them, apparently meaningless, are reserved to the experienced user. If you don't know what a parameter is, please accept the default value.

SETTINGS:	
Channel language:	Default language of the protocol. The default installation includes english and italian templates but it's possible to add custom templates
Zone Tone:	"Zone tones" that will be loaded for the hardware interfaces.
Indications:	Normally, some tones are directly created from the PBX. Indications sets the Zone to use for that tones.
CallerID:	Enables or disables the CallerID management on the card.
Usecallingpres	Enables or disables the *Caller ID Presentations*.
Switchtype:	The European standard is *EuroISDN*.
Pridialplan:	Defines the called extension format. Default: *local*
Prilocaldialplan:	Defines the caller extension format. Default: *national*
LBO: *(only PRI)*	Line Build Out. Default: *0*
Protocol: *(only PRI)*	Select E1 for the European standard, T1 for American. Default: E1
Framing: *(only PRI)*	"Framing and coding" defines communication parameters with the connected premise. For E1 it is possible to choose between *cas* or *ccs*, for T1 between *d4* or *esf*. Default: *E1/ccs* and *T1/d4*
Coding: *(only PRI)*	For E1 can be *ami* or *hdb3*, for T1 *ami* or *b8zs*. Default: *E1/hdb3* and *T1/ami*
Enable CRC: *(only PRI)*	If a line E1 is used it is possible to enable the CRC control only if the telephone carrier provides this service.
Busydetect: *(analogic only)*	When using POTS lines, lets to identify the busy tones and to detect the line disconnection signal.
Busycount: *(analogic only)*	If busydetect is enabled, busycount sets the amount of busy tones to get before assuming a busy line. Default: *4*
Telephone num.: *(QuadGSM only)*	Phone number of the used SIM.
PIN: *(QuadGSM only)*	SIM's PIN code (if the SIM haven't PIN, leave the field empty).
Signalling:	Defines the signalling type. Generally use *TE* when connected to a telephone line and *NT* when connected to a PBX.
EC:	*EchoCancel*. Enable the echo cancellation process.
ECWB:	*EchoCancel When Bridged.* Enables the echo cancellation for bridged TDM calls.
ET:	*EchoTraining.* Enables EchoTraining, a feature useful to improve Asterisk's echo recognition. To disable, leave it blank; to enable, sets a value between 10 and 4000 (that is the delay, expressed in milliseconds, Asterisk has to wait before to evaluate the amount of received echo.
Overlap:	Enables the sending of digits in overlap mode.
Timing:	For each port, sets the priority or totally disables the timing. Allowed values are: • 0 don't use this port as a synchronization source • 1 use this port as primary synchronization source • 2 use this port as secondary synchronization source, and so on… Timing is normally enabled on ports conncted to the carrier (POTS, BRI,

	PRI) and disabled on ports connected to another PBX.
Rxgain:	Sets the volume during receipt. Values can vary between +100.0 and -100.0. Default value: *0.0*
Txgain:	Sets the volume during transission. Values can vary between +100.0 and -100.0. Default value: *0.0*

easyAsterisk will show a warning message when a reconfiguration of ZAP local extensions or trunks are needed. This normally happens when a card is removed or added, for instance. This icon (⚠) is shown when an extension or a trunk need a reconfiguration.

4.15 Extensions Summary

In "*PBX Settings*" a table showing all currently configured extensions has been introduced. This table lists configured extensions for local extensions, groups, queues, ivr menù, meetme, fast dial and pbx applications. Using "*Find*" button specific extensions can be quickly found.

5. Incoming calls management

5.1 Attendant console

The "attendant console" is normally a standard telephone station with some services / features added. In some configurations, all incopming calls are terminated to the attendant console that provides in their forwarding and management. With easyAsterisk it is possible to arrange different destination to incoming calls according to days of week or different time slots. easyAsterik's attendant console configuration is made by means of *Calls Management*/*Attendant Console* item *(figure 5.1)*.

Figure 5.1 – Attendant Console configuration

As the image shows, it is possible to define different destinations: for “*Regular Hours*”, that is normally intended as working hours received calls, and "*After Hours*", for all other calls. Generally, when an attendant console is used the greater part of incoming, regular hours calls are diverted to the attendant console operator or to specific services (IVR, for istance); "*After Hours*" are normally diverted to IVR or advertising services.

OPTION:	
Record incoming calls on CDR:	Enables or disables the recording of incoming calls in a database for following inquiries.
REGULAR HOURS:	Defines working hours and days. Up to 3 ranges of values can be defined here. Unset ranges will be considered as “After Hours”.

DESTINATION: REGULAR HOURS:	Defines the destination type element for calls received in “regular hours”.
DESTINATION: AFTER HOURS:	Defines the destination type element for calls received in “after hours”.

5.2 Direct Inward Dial (DID)

Direct Inward Dialing (DID, also called DDI in Europe) is a feature offered by telephone companies for use with their customers' PBX system, whereby the telephone company (telco) allocates a range of numbers all connected to their customer's PBX. As calls are presented to the PBX, the number that the caller dials is also given, so that the PBX can decide which person in the office to route the call to.
In easyAsterisk the DID configuration is made by means of *Calls Management/Direct Inward Dial (DID)* item.

GENERAL:	
DID number:	The local extension to be configured.
Record calls on CDR:	Enables or disables the CDR recording of the conversations of the extension.
DESTINATION:	Defines the destination extension.

5.3 Interactive Voice Response (IVR)

easyAsterisk let users to create a series of IVR menus that will drive the incoming caller to the various services assigned to each extension. IVR menus are managed by *Calls Management*/*Interactive Voice Response (IVR)* item.

GENERAL:	
Name:	Identifyier of the IVR menu’.
Extension:	Local extension associated to the menu.
Audio:	Audio files played to the caller for welcome purposes. Select an audio file from the repository; normally, this file contains a brief instruction on IVR’s usage.
Audio No-Sel:	Audio file to play if the user doesn’t effect any choice until *Response Timeout*.
Audio Invalid-Sel:	Audio file to play if an invalid choice was made.

Response Timeout:	Maximum time, in second, for the user to make a choice.
Digit Timeout:	Timeout to consider during extensions keyin, if the caller is too slow.
Description:	Text field to describe the IVR meaning.

Pressing *"Add"* key a series of options will appear. It is possible to set up to 3 timeslots to consider in IVR management.

HOURS:	Defines the IVR activity timeslot.
AFTER HOURS:	Defines the policy to activate during *After Hours*.
OPTION:	The Option field contains the extension the user has to digit to make the choice. Once configured, press to make it active.

6. Outbound calls management

Outbound calls management involves several operations:

- **Definition of one or more trunks.** You can think to a trunk as a group of communication channels used by the PBX for calls management. For instance, a trunk can be a connection to an IP carrier, a link to another PBX or a group of communication ports on an hardware interface.
- **Configuration of one or more outbound routes.** A route can be defined as a logical group of numerical classes that define what types of calls can be effected by the clients (for instance national numbers, international, mobile phones, and so on). easyAsterisk lets to configure a boundless number of routes, everyone identified by a dialing prefix code (a code that the user must digit before the destination number to select a specific route). Using routes it’s also possible to configure LCR services.
- **Route’s permissions management.** Once the routes are defined, each user can be granted to access or not any route; in this way it’s possible to define a detailed outbound calling permission plan.

6.1 Hardware Trunks (ZAP)

As we said before, it’s possible to connect easyAsterisk to the *General Telephone Net (RTG)* using some hardware interfaces. To do that its’ mandatory to create one or more trunks (*Trunks Management* menu, *Trunks ZAP* item). Let’s see some details:

GENERAL:	
Name:	Textual label identifying the trunk being created.
Caller ID:	Caller ID that must be used when an outgoing call goes across the trunk. If not specified the generating Client Caller ID will be used.
Max channels:	Maximum number of contemporary opened outgoing channels for the trunk. If not set, the physical limit will be assumed.
PORTS:	The following image shows the physical ports available in the system; you can choose which ports must be included in the trunk being defined (e.g.: you can set a new trunk using only two of the four ports provided by a Junghann’s 4BRI card).

Figure 6.1 –ZAP Trunk creation

6.2 Trunks IP SIP and IAX2

An IP trunk defines a connection to a IP telephone carrier or to another PBX. IAX2 protocol, compared to SIP protocol, is more easily manageable behind a firewall/NAT because it uses a single port (default 4569 UDP) for both signaling and RTP voice traffic, while SIP uses (by default) 5060 UDP for the signaling and a variable range port (usually in the range between 10000-20000) for RTP. IP trunks are managed starting from the *Management Trunks* menu that lets the user to define 3 items:

- **Peer,** that defines the parameters to use for outgoing calls. The most important parameters are the destination host (generally the IP server to connect), username and password.
- **User,** used to authenticate incoming calls coming from the host defined in the peer.
- **Register,** mandatory if a dynamic IP is used to let the remote host know easyAsterisk's IP address..

A brief discussion can be made on how to receive incoming calls on a trunk IP. *User* configuration is going to disuse because it's often possible to use only *peer* configuration.
When Asterisk receives a call it evaluates to accept or not using these criteria:

- Tries to find a *user* matching the caller name (from: username field); if found, the call is accepted.
- Tries to find a *peer* matching the caller's IP address; if found, the call is accepted.
- Incoming call is otherwise rejected unless differently specified in *Protocol General Settings* (choosing to accept unauthenticated incoming calls).

GENERAL:	
Name:	Textual label identifying the trunk being created.
Caller ID:	Caller ID that must be used when an outgoing call goes across the trunk. If not specified the generating Client Caller ID will be used.
Max channels:	Maximum number of contemporary opened outgoing channels for the trunk. If not set, the physical limit will be assumed.
PEER:	
Name:	The name assigned to the peer.
Host:	Remote server IP address to which address outgoing calls. If a dynamic IP is used set to "*dynamic*" (without quote).
Username:	Remote server's authentication user name.
Secret:	Remote server's authentication password.

Fromuser: *(only SIP)*	Specifies the user to put in the "from" field in place of callerID (only for SIP protocol).
Fromdomain: *(only SIP)*	Some IP carriers needs to know the originating Domain, for authentication purposes. Set it into this field.
Type:	If set to **type=friend,** the peer also act as an user; therefore it will accept remote host's incoming calls.
DTMF: *(only SIP)*	Here you can define how to manage DTMF tones.
Nat: *(only SIP)*	Enable this option if the peer is behind to a nat device.
Qualify:	Checks periodically that the peer is online.
Canreinvite: *(only SIP)*	Enabling this option a client will try to send media streams directly to the peer. Setting it to off all media streams will cross the Asterisk server.
Notransfer: *(only IAX2)*	Similar as *canreinvite* but applied to IAX2 protocol. Setting *notransfer=yes* Asterisk will prevent the client to send audio flows directly to a peer.
Trunk Mode: *(only IAX2)*	Enable the trunk mode (useful to save bandwidth).
Insecure:	Enable this option it will be possible to receive calls from peer without authentication process.
Context:	When the peer needs to manage incoming calls it is necessary to define a context. Leave it blank if the peer manages only outgoing calls: • **Incoming**: incoming calls will be addressed to the attendant console. It is possible to set routings by means of the direct inward dial rules. • **Local**: the calls will be addressed to the local extensions (extension, groups, etc…). Useful if you want to let clients of two interconnected easyAsterisk (each one with its own extensions range) to call each other. • **Custom**: it is possible to use customized contexts defined by the *Dialplan Management* menu.
Codecs:	Select the codecs used for this trunk. Set to *default* to use the one defined in General Protocol Parameters.
Deny / Permit / Priority:	It is possible to define some rules to limit the access to the peer. It is necessary to define the subnet address with the relative net mask and the priority with which the rules must be activated. For instance: deny = IP 0.0.0.0 NETMASK 0.0.0.0 permit = IP 1.2.3.4 NETMASK 255.255.255.255 priority = deny Denies all access except 1.2.3.4
Advanced:	Section reserved to experienced users to insert customized peer values.
USER:	
Name:	Name to assign to the user that we are creating.
Secret:	Authentication password.
Context:	Incoming calls context. Please consider the same peer's logic.
Codecs:	Defines the enabled codecs.
Deny / Permit / Priority:	Defines restrictions to be applied on incoming calls addresses. Please consider the same peer logic.
Advanced:	Section reserved to experienced users to insert customized user values.
REGISTER:	
Username:	Username.
Password:	Password.
Host:	Remote server.

Contact:	Specific local extension to be addressed when a call is received. Optional parameter.

6.3 Trunks IP H323

H323 protocol configuration is similar to the one used for SIP and IAX2 protocols: it's mandatory to define a *peer* for the outbound calls and an *user* for those incoming calls but it isn't supported a remote host registration process (*register*).
If a binding address isn't defined in Protocol's General Settings, a worning message will be shown when configuring an H323 trunk. Asterisk doesn't include a gatekeeper management: so, it's necessary to add and configure manually a thirdy-party product to do this.

6.4 Outbound routing and LCR

It's now necessary to defines routes and routing rules using *Calls Management/Outbound routing* item. The first parameters to set up are the name of the route and the dialing prefix to address it *(figure 6.2).*

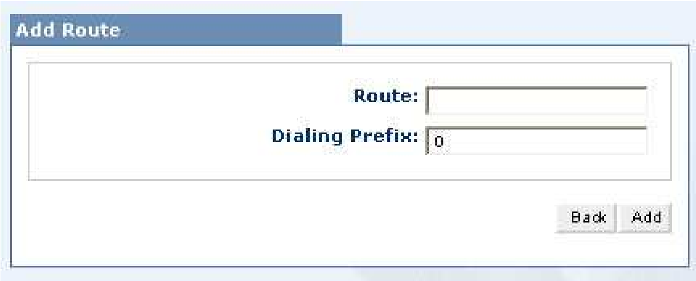


Figure 6.2 – New route adding.

And pressing the "Add" button to access the route configuration page (*figure 6.3*).

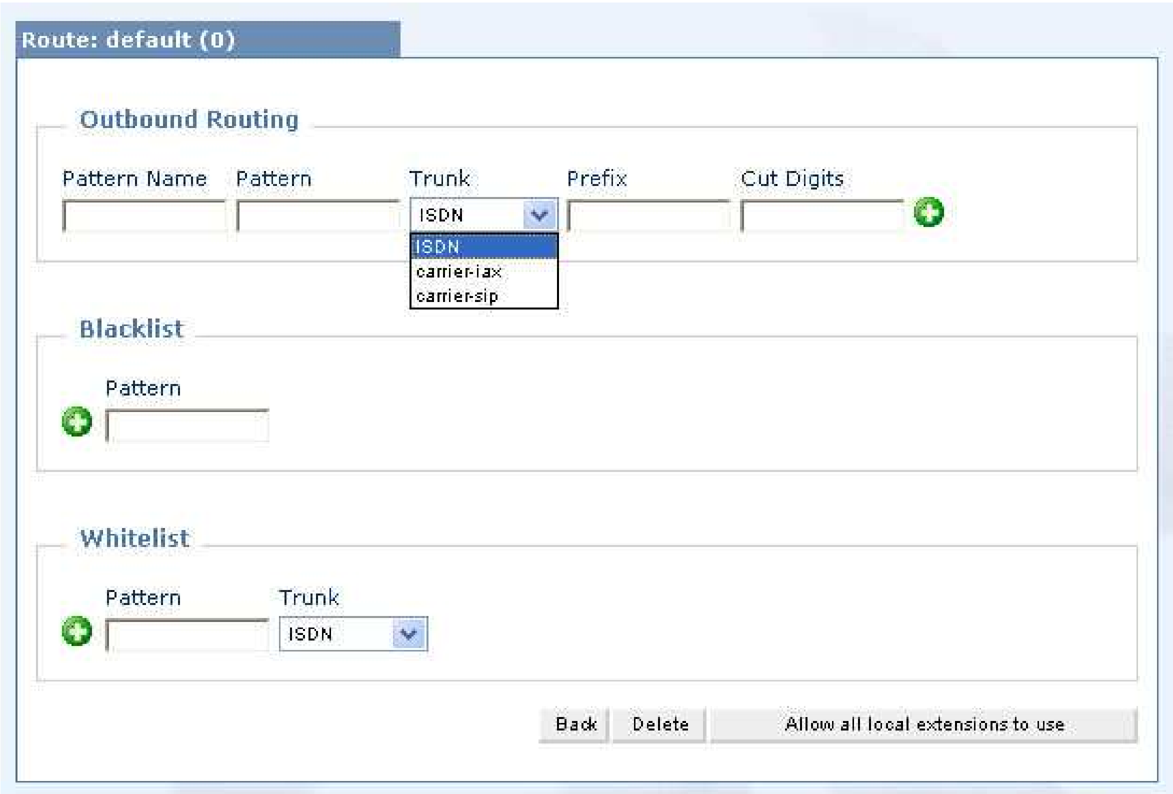


Figure 6.3 – Route attribute setting

The example shows a route named **default** with a dialing prefix code **0**. Please note that 3 trunks are defined (one probably pointing to one or more ports of an ISDN card and 2 IP connections). Using *patterns* it's now necessary to define route's routing rules. Patterns are alphanumerical strings defining outgoing addressable extensions and are composed by these chacarcters:

- **X** indicates an any digit from 0 to 9
- **Z** indicates an any digit from 1 to 9
- **N** indicates an any digit from 2 to 9
- It is possible to use parentheses to indicate digits range:
  [1246] indicates digit 1,2,4 and 6
  Separating two digits with a hyphen will mean all digits in that range:
  [1236-9] indicates digit 1,2,3,6,7,8 and 9
- **.** (dot) indicates any digits for an indefinite number of following digits

Some practical examples will help us to better understand patterns' structure:

PATTERN:	**DESCRIPTION:**
00.	Indicates numbers starting with a double zero and followed from any digit for an indefinite number of digits. This pattern often indicates *international* prefixes.
0Z.	All numbers starting with 0, a second digit from 1 to 9 and then any digit for an indefinite number of digits. Normally indicates *long-distance prefixes.*
3.	Normally indicates a mobile phone.
335.	In this case a particular mobile prefix is declared.
X.	Any type of number, limitsless.
800XXXXXX	Toll-free numbers: in Italy they are starting with an 800 prefix.
[1,8].	All numbers beginning with 1 or 8 followed by any digit for an indefinite number of digits.

Let's now analyze the various options:

OUTBOUND ROUTING:	
Pattern name:	Assign a name to the pattern that we are creating.
Pattern:	Define the pattern according to the criteria explained before.
Trunk:	Select the desired outgoing trunk.
Prefix:	It is possible to add a numerical prefix to select, for istance, a specific telephone carrier.
Cut Digits:	Sometime it's necessary to cut some digits before to make a call. This field declares the number of digits to cut.
BLACKLIST:	It is possible to define some extensions, or some patterns, that will be never dialed (or istance, avoinding international calls).
WHITELIST:	The opposite of blacklists. It's possible to declare a list of numbers that the users will be able to call (for istance, the international number of another company's branch). It is necessary to specify the trunk used to place outgoing calls.

Let's now continue with our example:
We define a route that picks all local calls routing them on the *carrier-sip* trunk. To do this w have to configure:

- **Pattern Name:** short_distance
- **Pattern:** 0Z. (it's also possible to use the alternative notation 0[1-9])
- **Trunk:** carrier-sip
- To add the defined routing it's necessary to press the key

We should see the following status (see *figure 6.4)*.

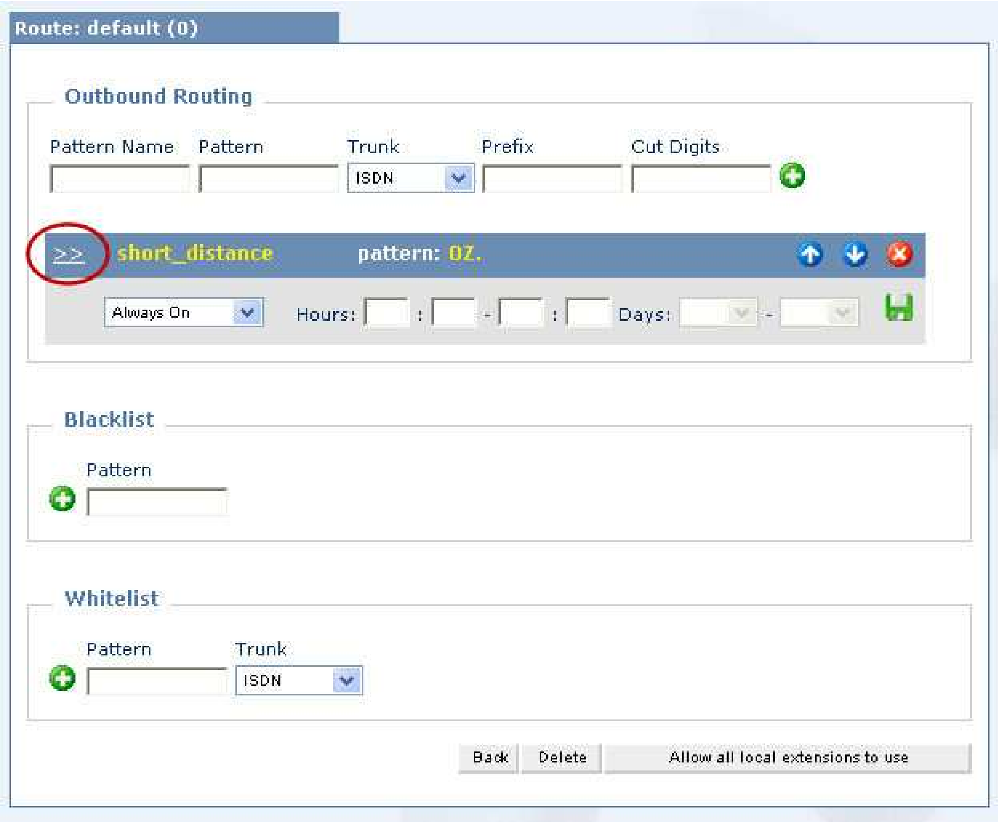


Figure 6.4 – Outbound routing setting

Through the drop-down menu it's possible to set the routing's activity time (always active or only in some days/hours). Please use the *"Scheduled"* menu item to set the preferred hours and days (to confirm the settings, press the icon).
Clicking on the arrows next to the pattern name (in *figure 6.4* it's encircled in red) it's possible to configure some alternative trunks on which the calls fallback if the first in list is not available (*figure 6.5*).

Figure 6.5 – Advanced property of a route object

The order of trunks represents their priority. In *figure 6.5* we add the trunk ISDN as secondary trunk with addition of the “1055” prefix on outbound extensions. To add trunks to the list is analogous but the box inside the table must be used. Through the keys encircled in red in *figure 6.5* is possible to change the trunks priority or eliminate them from the list.
Let’s now analyze the situation in *figure 6.6*.

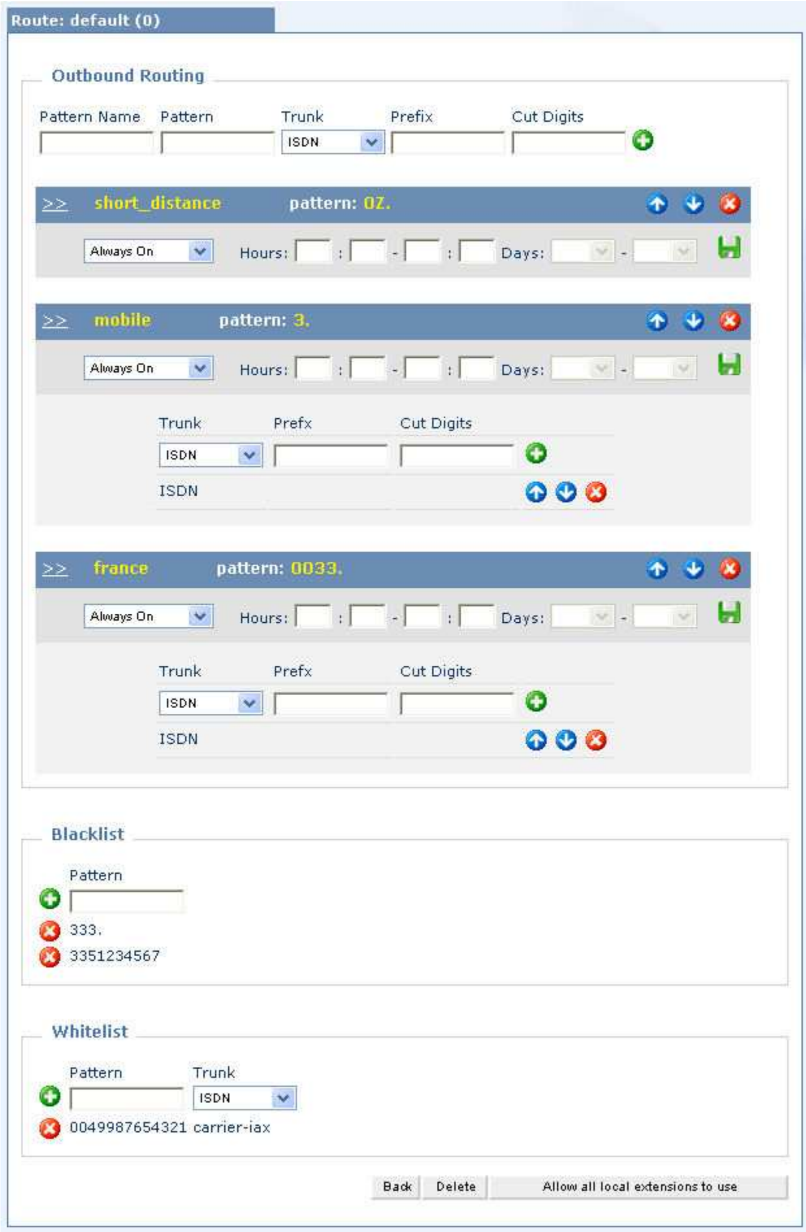


Figure 6.6 – Advanced route setting

Two routings had been added:

- *Mobile*, identifying calls to mobile operators. It used only the ISDN trunk without prefixes and it’s always active.
- *France*, that manages all calls toward France (international prefix 0033). This also uses ISDN trunk without prefixes or cut digits but it is scheduled from 09.00 hours till 18.00 o'clock from Monday to Friday. This simply means that it won't be possible to call France during the remaining hours.

Looking at the *Blacklist* setting it is interesting to note that, despite the mobile operators calls are always allowed, the given blacklist will block calls starting with 333 and the whole number 3351234567. The same for the whitelist; also when a specific route to Germany is not defined, the

user will call the extension 0049987654321 using the trunk *carrier-iax*. Please note that when a number or a pattern of a Whitelist is defined it's not possible to modify outbound routing with prefixes or cut digits.
To begin to use the route it's now necessary to set some permissions to the various local extensions. Pressing *"Allow all local extensions to use"* button, all the configured local extensions will be enabled to use all the outbound routings. As an alternative, it is possible to access to the attributes of every local extension (*PBX Settins* menu, *Local Extensions* item) under *Outbound Routing* and enable the desired accesses *(figure 6.7)*.

Figure 6.7 – Local extension enabling to the routes use

The order with which outbound routing configurations are listed in each definition represents the priority in their management. The following example shows better this concept: let's analyze the situation in *Figure 6.8*.

Figure 6.8 – An example to understand the routings priority

In this route we see 2 routings, *"long_distance"* and *"france"*, with *" long_distance"* listed for first and with priority on *"france"*. The users wants all international calls routed to *"carrier-sip"* and France-addressed calls to *"ISDN"*. This configuration isn't correct because the second definition is a subset of the first one and so it will be never activated.

To make a correct configuration it's enough to give more priority to the second setting using the blue arrows. Alternatively, it's also possible to delete the unused setting pressing on key. The *Figure 6.9* shows the correct configuration.

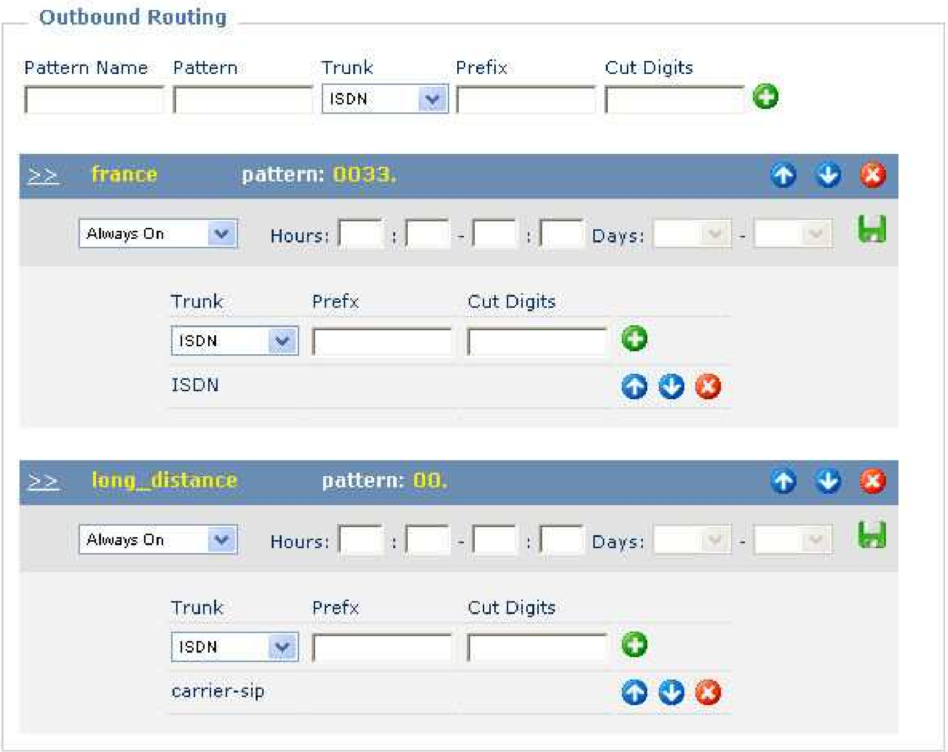


Figure 6.9 – The correct configuration of the proposed example.

6.5 Fast Dial

Fast Dial is a list of short codes, normally composed of a limited digits, easy-to-remember numbers, with some long extensions associated (*Calls Management* menu, *Fast Dial* item). To call the associated extension it's possible to type the short code, allowing a faster and better dial. easyAsterisk allows to configure two categories of short numbers:

- **Trunk type,** defining a correspondence between the short number and a long extension and preferred trunk;
- **IP type,** linking a short code to a specific IP Host.

Let's see the configuration in detail.

TRUNK TYPE:	
Extension:	The short number we need to configure.
Destination:	Associated extension.
Trunk:	Defines the trunk to use to place the call.
Description:	Optional text field to describe the configuration.
IP TYPE:	
Extension:	The short number we need to configure.
Destination:	Real extension to call.

Protocol:	The protocol we want to use.
IP:	Detination host IP address (the host on which that extension is defined).
Description:	Optional text field to describe the configuration.

6.6 Remote PBX

The *Remote PBX* configuration (reachable via *Calls Management* menu) lets the user to call the extensions of another PBX as if they were local extensions. It’s the standard configuration used to make PBXs VPN, letting different remote branches of the same company to communicate pratically with no cost.
Let’s make an example: in a company with two remote branches we have two interconnected easyAsterisk. The first branch used extensions ranging from 100 to 199, the second ranging from 200 to 299.
The configuration is very easy, because it’s only necessary to declare, on each easyAsterisk, the trunk to be used and the range of the other PBX’s extensions.

Pattern:	Defines the pattern of remote extensions (in this example, 2XX for the first PBX and 1XX for the second).
Trunk:	Selects the trunk to use to place calls.
CDR Record:	Enables or disables the details recording the calls in the internal database.

6.7 Hints about Outbound Caller ID management

The management of outbound caller ID is splitted in two sections:

- If a fixed CallerID has been defined on the trunk (*General Options* of trunk properties) that CallerID will always be used. If for the local extension that places the call a *CID Out Suffix* has been defined, that suffix will be added to the outgoing CallerID.
- If no fixed CallerID has been defined, the numerical CallerID of the calling local extension will be used.

Please pay a good attention to the correct configuration between these parameters and the configuration of Carrier’s outbound lines: it’s very often a cause of mistakes.

6.8 Router IP

With "IP Router" architecture it’s possible to route calls coming from traditional PBX on routes configured on easyAsterisk. This is very useful when the user wants to recover all the existing hardware (standard PBX and phones) but also wants to use an IP Carrier. This schema *(figure 6.10)* lets to understand better:

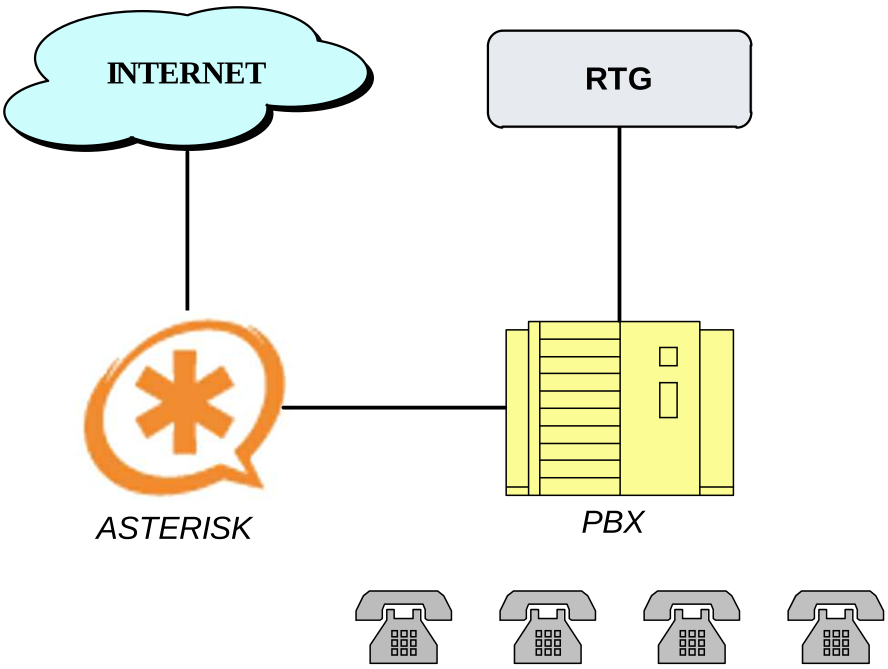


Figure 6.10 – Connection of easyAsterisk to a traditional PBX

This particular configuration often needs a partial PBX reconfiguration. easyAsterisk machine and the PBX are connected through an ISDN interface (Bri or Pri) configured as *NT* in easyAsterisk and *TE* on the PBX.
The existing PBX also must be set to route all outgoing calls to that interface to allow their routing on IP net.

Note: due to specific technical characteristics on ISDN interface it's advisable to disable the timing management on the ISDN PBX's port.

Looking at the configuration from Asterisk side you can see that incoming calls, received by easyAsterisk from the ISDN interface, are converted in outgoing calls via the IP Trunk:

1. Configure a ZAP trunk including the ports used to connect the PBX.
2. Configure the IP trunks to connect easyAsterisk to the IP Carrier.
3. Configure a route to address outbound calls to IP Carrier.
4. Redirect the calls coming from the ZAP trunk to the IP route.

To manage the fourth point (the first three points are described in the previous pages) we will use the *Calls Management* menu under the *Router IP* item *(figure 6.11)*.

Figure 6.11 – Router IP configuration

Name:	The name of the IP Router.
Trunk:	The trunk from the calls to be routed are received from the PBX
Route:	The destination IP route
Emulate ringing:	Setting this option, easyAsterisk will emulate a ringing phone signal, so that the traditional PBX will be aware that the call was correctly placed out.
Saves in CDR as:	Sets the kind of recording used to mark in CRD the outgoing calls (*INCOMING* or *OUTGOING*).

In *figure 6.11* we are creating a IP router called *"from-PBX"* that redirects the calls coming from *"ISDN"* trunk on the route *"default"*.

Another architecture made possible by easyAsterisk IP routing capabilities is the one represented in *figure 6.12*, where an easyAsterisk server can place outgoing calls using the resources of another remote easyAsterisk connected via a SIP or IAX2 trunk. In this case it's only necessary to specify, in *Asterisk 2*, the IP trunk used to connect *Asterisk 1* as the outgoing default trunk.

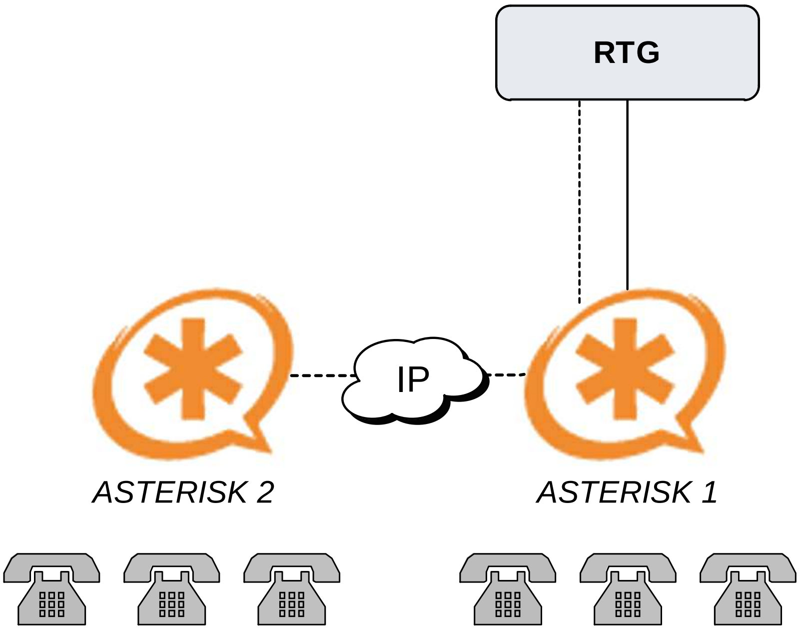


Figure 6.12 – Connection of two servers easyAsterisk through Router IP

7. Calls detail record

easyAsterisk can record a complete list of details on incoming and outgoing calls in a Sql database. To manage the database you can use the menu *Calls Management* under *Calls Detail Record* item *(figure 7.1)*.

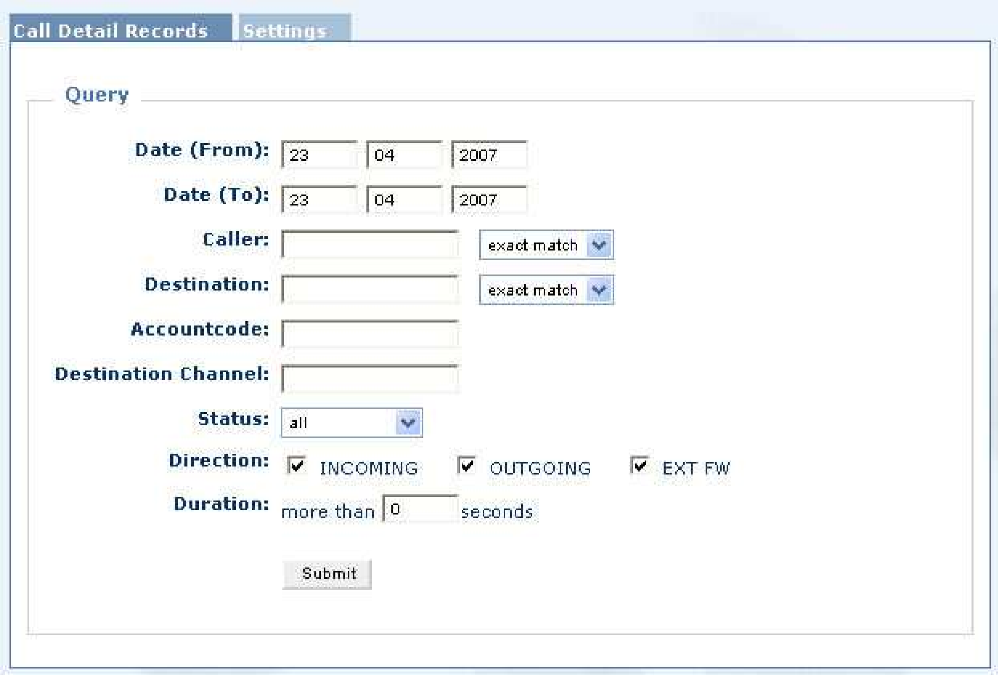


Figure 7.1 – Call Detail Records

Through this panel it's possible to perform some queries to extrapolate data from the database.

Date (From): **Date (To):**	Sets the date range scope for the query.
Caller:	The caller's number (or a portion of that number) to consider in the query.
Destination:	The addressed number (or a portion of that number) to consider in the query.
Accountcode:	Account code to use in the query.
Destination Channel:	Name of the opened channel (during the call).
Status:	Status of the call (all, answered, not answered, busy)
Direction:	Can be INCOMING, OUTGOING, or a external forwarding (EXT FW). In the last case, the forwarding local extension is visualized too.
Duration:	Sets the desired minimum duration for the call.

The dataset reported by the query will be shown; the user can export the data (PDF or CSV format) or he can decide to erase the records from the database.

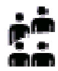
It's now possible to record CDR data on a plain text file. In this case, the resulting filename will be "*/var/log/asterisk/cdr-csv/Master.csv*". It's advisable to enable the *Log Rotation* for this file to avoid unpredictable results.

8. The operator panel

easyAsterisk integrates *Flash Operator Panel*, an application developed by Nicolás Gundiño (http://www.asternic.org/) that allows to visualize through a browser the real time status of local extensions, queues, conference rooms and external lines. FOP configuration is made through *General Settings* menu, *Operator Panel Management* item. In this section there are some submenus: *Settings* lets to manage some general options, the other menus (*Extension*, *Queues*, *MeetMe*, *Peer/Users*) lets to manage various FOP's elements.

Language:	The language to be used in FOP.
Layout:	Sets the dimension of the buttons to be configured.
Security Code:	It's a password (default value: "password") that the operator can use to force the disconnection of a call or to activate a call between two extensions (simply dragging the caller's icon on the addressed user's icon). If these operation are done at runtime, FOP will ask for this password.
Hide Caller ID:	Hides the CallerIDs for the calls in progress.
Show IP:	Visualize the IP address of the extension and of the IP trunks.
Show Agent Code:	Enable this option to visualize the code of the extension-logged agent.
Show Agent Name:	Enable this option to see the logged in Agend name.
Change led color for queue members:	Enable this option to let the extension icon color to change when an agent logs in.
Show external lines (bri and analog):	This option is available only for POTS and ISDN BRI lines; when enabled, shows the status of the external line.

Let's now see some other options:

Show:	Shows/Hides the element in the panel.
Label:	Assigns a label to the element.
Icon:	Assigns an icon to the element. The available icons are: **Icon 1:**  **Icon 4:**  **Icon 2:**  **Icon 5:**  **Icon 3:**  **Icon 6:** 

Admins can open the Operator Panel using *General Settings* menu', *Operator Panel* item. Users will be able to open FOP from their user interface only when granted of Administrator's permissions.

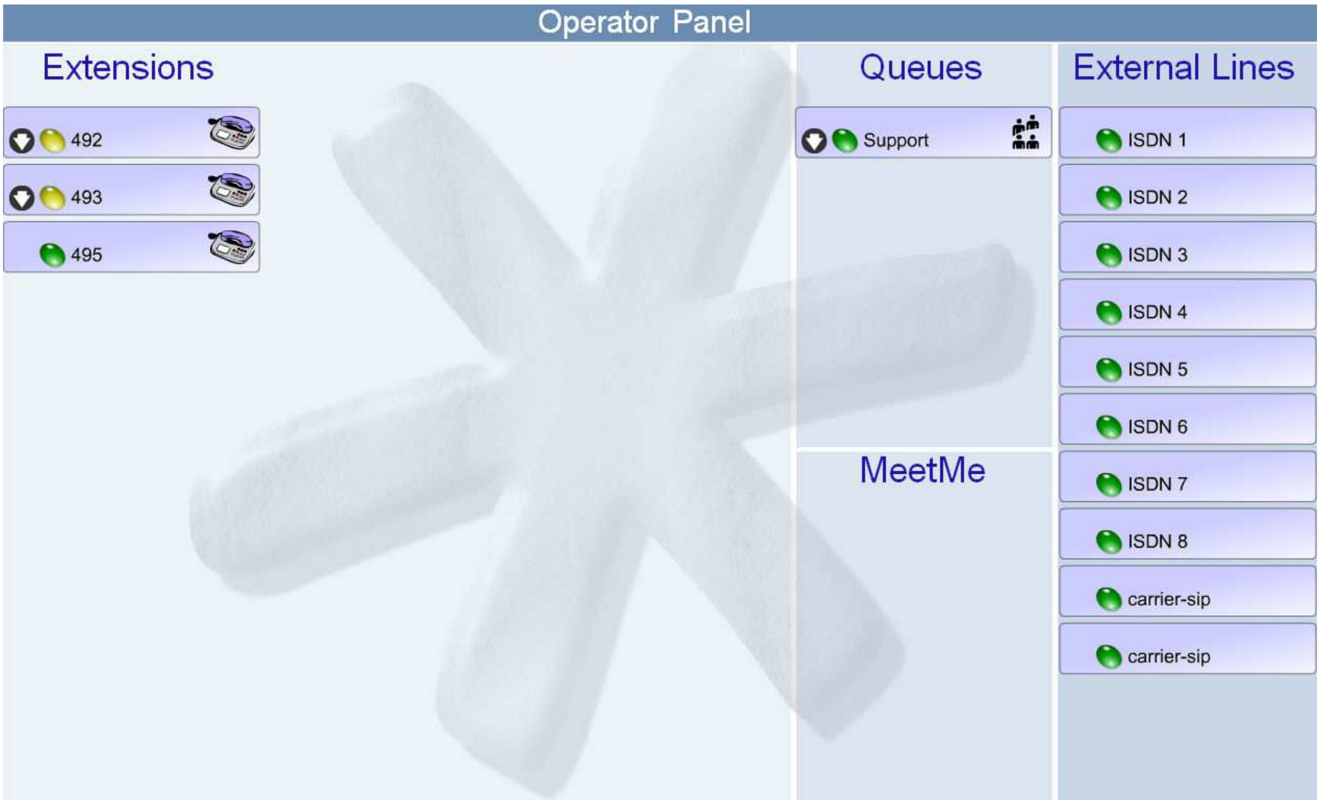


Figure 8.1 – The operator panel

9. PBX management

By means of *System Management* menu it's possible to manage our IP PBX and get information on its status

9.1 Network settings

Using *IP* menu it's possible to modify the parameters related to the net address configuration. Normally, a system reboot is needed to apply the changes, nevertheless it's not necessary when only the DNS address has been changed.

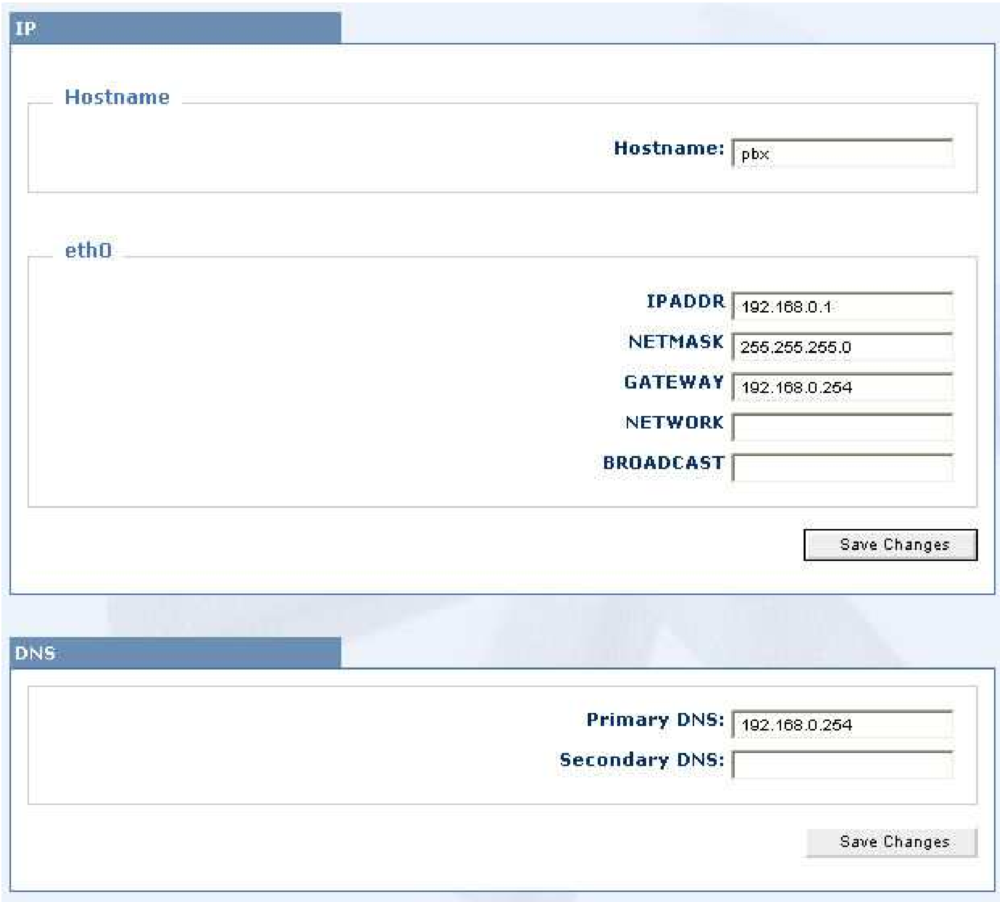


Figure 9.1 – Network settings changes

9.2 Date and time set-up

By default, easyAsterisk activates an NTP service to let IP telephones to syncronize their clock directly with easyAsterisk. NTP takes the current time from the server's system time; it can be modified using the *Date / Time* menu.

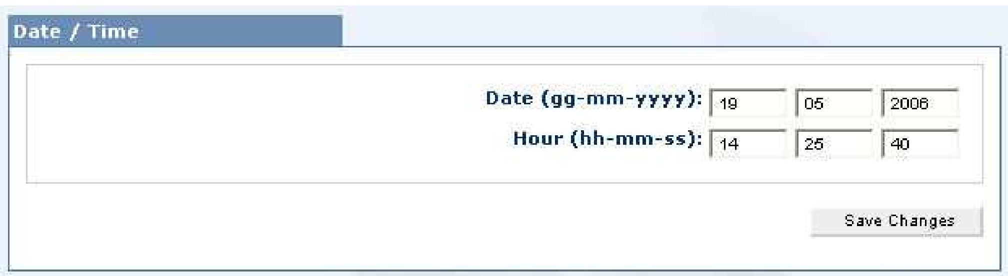
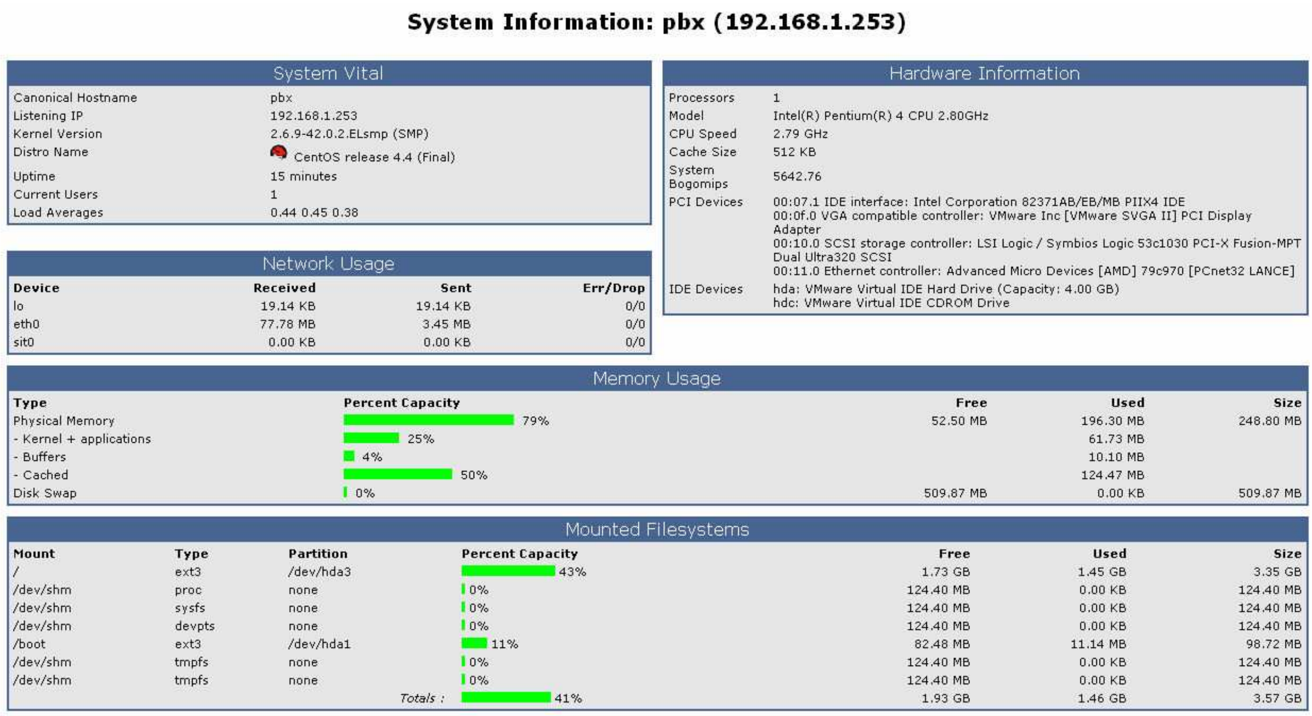


Figure 9.2 – Date and system time settings

9.3 System information

easyAsterisk integrates *Phpsysinfo*, an opensource application that gives a detailed series of information on the hardware status, on mounted filesystems, on the memory usage and on the network usage (http://phpsysinfo.sourceforge.net/).



System Information: pbx (192.168.1.253)

System Vital

Canonical Hostname	pbx
Listening IP	192.168.1.253
Kernel Version	2.6.9-42.0.2.ELsmp (SMP)
Distro Name	CentOS release 4.4 (Final)
Uptime	15 minutes
Current Users	1
Load Averages	0.44 0.45 0.38

Hardware Information

Processors	1
Model	Intel(R) Pentium(R) 4 CPU 2.80GHz
CPU Speed	2.79 GHz
Cache Size	512 KB
System Bogomips	5642.76
PCI Devices	00:07.1 IDE interface: Intel Corporation 82371AB/EB/MB PIIX4 IDE 00:0f.0 VGA compatible controller: VMware Inc [VMware SVGA II] PCI Display Adapter 00:10.0 SCSI storage controller: LSI Logic / Symbios Logic 53c1030 PCI-X Fusion-MPT Dual Ultra320 SCSI 00:11.0 Ethernet controller: Advanced Micro Devices [AMD] 79c970 [PCnet32 LANCE]
IDE Devices	hda: VMware Virtual IDE Hard Drive (Capacity: 4.00 GB) hdc: VMware Virtual IDE CDROM Drive

Network Usage

Device	Received	Sent	Err/Drop
lo	19.14 KB	19.14 KB	0/0
eth0	77.78 MB	3.45 MB	0/0
sit0	0.00 KB	0.00 KB	0/0

Memory Usage

Type	Percent Capacity	Free	Used	Size
Physical Memory	79%	52.50 MB	196.30 MB	248.80 MB
- Kernel + applications	25%		61.73 MB	
- Buffers	4%		10.10 MB	
- Cached	50%		124.47 MB	
Disk Swap	0%	509.87 MB	0.00 KB	509.87 MB

Mounted Filesystems

Mount	Type	Partition	Percent Capacity	Free	Used	Size
/	ext3	/dev/hda3	43%	1.73 GB	1.45 GB	3.35 GB
/dev/shm	proc	none	0%	124.40 MB	0.00 KB	124.40 MB
/dev/shm	sysfs	none	0%	124.40 MB	0.00 KB	124.40 MB
/dev/shm	devpts	none	0%	124.40 MB	0.00 KB	124.40 MB
/boot	ext3	/dev/hda1	11%	82.48 MB	11.14 MB	98.72 MB
/dev/shm	tmpfs	none	0%	124.40 MB	0.00 KB	124.40 MB
/dev/shm	tmpfs	none	0%	124.40 MB	0.00 KB	124.40 MB
		Totals :	41%	1.93 GB	1.46 GB	3.57 GB

Figura 9.3 – System Information

9.4 Asterisk server information

Using *Asterisk Info* menu it's possible to access a series of submenus showing the real-time status of the PBX. To update a page simply press the browser "refresh" button. Let's see in detail what informations are given:

Active Channels:	Lists the opened channels.
SIP/IAX2 Registry:	Shows the status of SIP/IAX2 registrations on remote server (for instance, an IP telephone carrier).
Queues:	Status of queues and operators. For each queue a list of logged in users is shown; some statistics about answered or unanswered calls are given too.
Agents:	Here we can see the status of Agents.
MeetMe:	List of the logged-in users in every meetme room.

9.5 System status

Through *System Status* it's possible to get informations on some main services while they're running. Every service can be stopped or restarted pressing self-explaining buttons; it's not possible with services such as Apache, Mysql, Sendmail and Cron. In this section, we find useful informations such as easyAsterisk's Uptime; finally, *Asterisk Monitor* is a script that checks periodically Asterisk behaviour and, in case of problems, restarts the related services and reloads the TELCO hardware device-drivers.

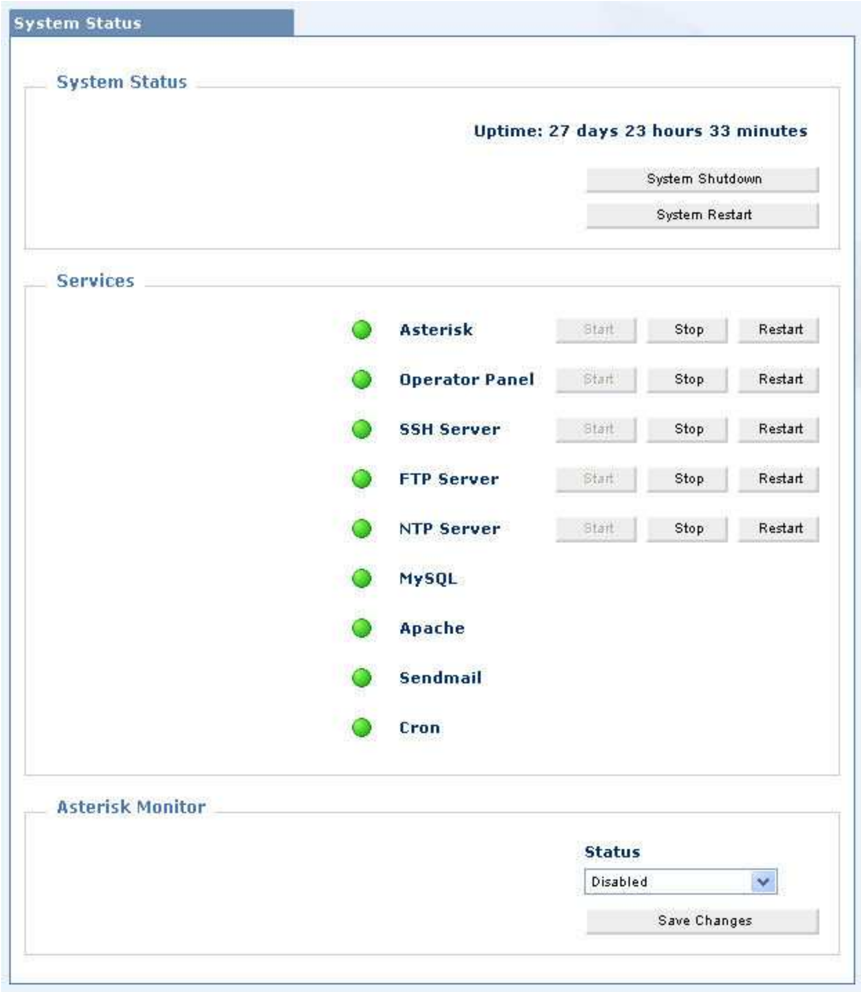


Figure 9.4 –System Status

9.6 Backup and Restore

Using *Backup and Restore* it's possible to manage PBX configuration and CRD backup files. This section is divided in 3 parts:

- **Backup:** a simple, two-buttons panel to instantly perform a backup of the selected element. The resulting file is added into the underlying section and the given name will include the system date/time.
- **System Backup Files:** lists all the previous PBX backup files. All files can be downloaded, deleted or restored using the relative button.
- **CDR Backup Files:** lists all CDR backup files.

To re-upload a file locally saved it's necessary to open an FTP connection with the server using the following username and password:

Username: **backup**
Password: **backup**

When the connection is estabilished two folders are shown:

- *system*, that takes system backup files
- *CDR*, that takes CDR backup files.

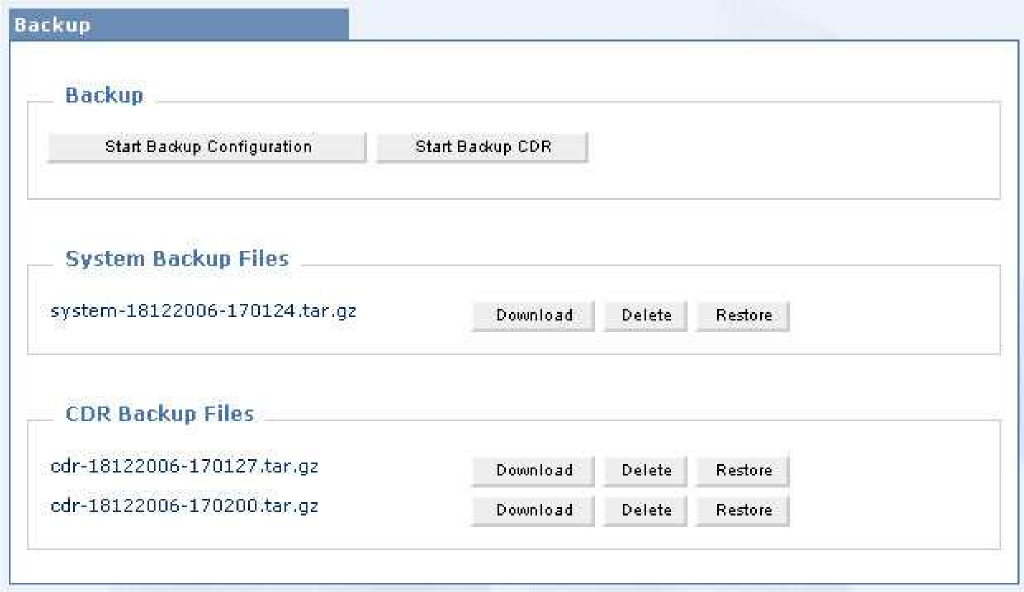


Figure 9.5 – Backup and restore

When restoring a system backup, the user must tell if he wants to update also the database's structure. This feature is normally used when a backup was generated with a previous release. To use this feature it's mandatory to give SQL administrator's permissions (user/password); these fields can be leaved empty if the SQL root default password hasn't been changed.

9.7 Database management via Phpmyadmin

The whole PBX configuration is recorded on a Mysql database, called *astconf*, that represents the source of each easyAsterisk configuration. easyAsterisk integrates Phpmyadmin utility (*System Management* menu) that is useful to manage, when neede, the MySql database. We recommend nevertheless to use this utility only when strictly necessary, because a wrong operation could cause unpredictable results. In that deprecated case, simply restore a previously saved working configuration using the yet-discussed utilities. With this utility is also possible to access the CDR database (*asteriskcdrdb*).
To use Phpmyadmin it's necessary to complete an authentication process using:

Username: **sqladmin**
Password: **sqladmin**

It is also possible to modify the password using a special option of Phpmyadmin's access menu.

Figure 9.6 – Phpmyadmin access menu

9.8 Users management

easyAsterisk can manage up to three levels of user's security:

- **ADMIN:** can access all functionalities with no limitations
- **PBXMANAGER:** can manage all PBX elements but can't modify the system configuration. E.g.: this kind of user can add or remove extensions, groups, conference rooms, agents, queues, virtual fax, can modify the operator panel configuration and manage system-level address book.
- **STATS:** can only access CDR files, normally for accounting purposes.

At least one admin user must be defined. The User's Management panel can be reached through the *System Management* menu (*Users Manager* item).

Figure 9.7 – Users Manager menu

9.9 System address book

easyAsterisk can manage two types of address books: *System-level*, available for all users, and *Personal*, particular and available for each single user. Using the User Panel and browsing the address book simply click on the desired item to place a call. Finally, it's also possible to search within the address book entries (the *contacts*) using the special text box and pressing the *"Find"* button. Please note that the search process spans over all the fields of each record.

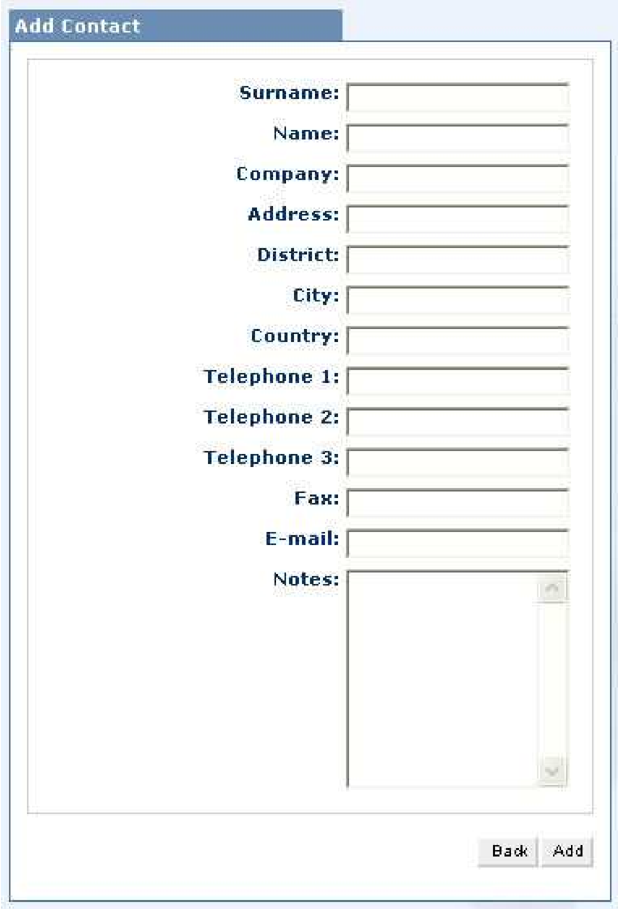


Figure 9.8 – Contact insertion in index book

9.10 API Manager

Asterisk API Manager let authenticated user applications to connect and to communicate with Asterisk using TCP/IP protocol. Their administration is made via *System Management* menu/*API Manager* item. This menù is divided in two sections:

- **Settings:** this section is used to configure the Managers' IP bind address (TCP 5038 port is used by default) and it is possible to choose between 127.0.0.1 (the loopback) or 0.0.0.0. This second option is deprecated for security reasons.
- **Users:** this section is used to define some parameters for each consigured Manager (username, password, permissions and so on) to let the Manager to interact with Asterisk.

More information on Asterisk Manager are available on the following web site: http://www.voip-info.org/wiki-Asterisk+manager+API

9.11 Log management

Asterisk server uses 3 different logfiles, normally located in "*/var/log/asterisk*":

- **event_log :** used to record Asterisk's runtime informations
- **messages :** used to log Asterisk's messages
- **queue_log :** queue's messages logfile

It's possible to visualize these files using a specific men' item (to find it, follow *System Management/Log Viewer)*. Rotation of these LogFiles can be enabled from *Settings* submenu', along with the detail level of messages.

9.12 Update Database Structure

When updating easyAsterisk, a database structure realignment is often needed. To use this feature it's mandatory to give administrator's permissions (user/password); these fields can be leaved empty if the SQL DB's root default password hasn't been changed.

10. Dialplan customizing

This section is addressed to expert users. Using *Dialplan Management* it's possible to modify the easyAsterisk's system macros or to create customized macros and contexts.

10.1 Changing a system macro

Since a wrong operation in this area can seriously damage our PBX, easyAsterisk provides the ability to restore an original macro. When a System macro has been modified an icon will appear on its side to remember that the macro waas modified. (*figure 10.1*)

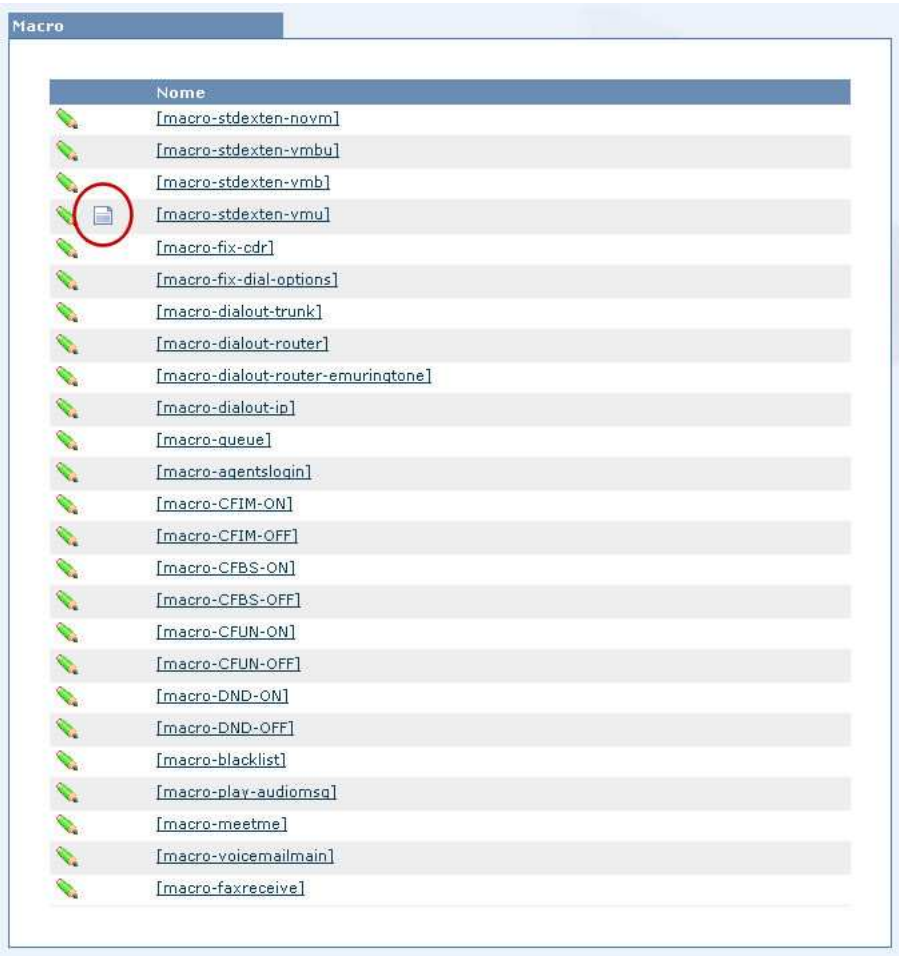


Figure 10.1 – System macro customizing

Let's briefly see system macros:

[macro-stdexten-novm]	Defines an extension with disabled voicemail
[macro-stdexten-vmbu]	Defines an extension with voicemail activated when busy or unavailable.
[macro-stdexten-vmb]	Defines an extension with voicemail activated when busy.
[macro-stdexten-vmu]	Defines an extension with voicemail activated when unavailable.

[macro-fix-cdr]	Modifies the Caller ID recorded in CDR when a call is directly transferred to another extension.
[macro-fix-dial-options]	Sets the call options for the incoming and outgoing calls.
[macro-dialout-trunk]	Macro used when an external call is made.
[macro-dialout-trunk-gsm]	Used for calls made via a QuadGSM linked trunk.
[macro-dialout-router]	Used when a call is received from an IP router. (It is possible to customize this macro only in the professional version).
[macro-dialout-router-emuringtone]	Used when an IP incoming call, having the *"Emulate Ringing"* option enabled, is received. (It is possible to customize this macro only in the professional version).
[macro-dialout-ip]	Called when an *"IP type"* short number is defined.
[macro-queue]	Defines a queue.
[macro-agentslogin]	Defines the agents' login extension.
[macro-CFIM-ON]	Activates an immediate forwarding.
[macro-CFIM-OFF]	Disables an immediate forwarding.
[macro-CFBS-ON]	Activates a on busy-forwarding.
[macro-CFBS-OFF]	Disables a on busy-forwarding.
[macro-CFUN-ON]	Activates a on unavailable-forwarding.
[macro-CFUN-OFF]	Disables a on unavailable-forwarding.
[macro-DND-ON]	Activates *"Do Not Disturb"* service.
[macro-DND-OFF]	Disables *"Do Not Disturb"* service.
[macro-blacklist]	Defines a blacklist route.
[macro-play-audiomsg]	Plays an audio file.
[macro-meetme]	Defines a conference room.
[macro-voicemailmain]	Defines the extension of the voicemail messages center (answering service).
[macro-faxreceive]	Defines a virtual fax.

10.2 Contexts and macro customizing

Customized contexts, as seen before, can be used in the IP trunks configuration, in the general settings of the IP protocols, in the DIDs and in the IVR menu. When a customized context is created, a leading underscore will be added to its name to avoid duplicated names. By default, a customized context called "*custom-exten*" is available after the first installation of the software.
Using Customized context it's possible to define local extensions to which the PBX local extensions can refer to obtain common services or behaviours; let's see a brief example to make things more clear:

```
[custom-exten]
exten => 123,1,Playback(demo-congrats)
exten => 123,2,Hangup
```

In this example, the 123 extension will play the audio file "demo-congrats" just before to hangup the line

10.3 Global Variables

This section lets to set up global dialplan variables that maybe used when compiling custom contexts and macros. For every new global variable it's necessary to type *name* and *value*. Global variables will be inserted in the DialPlan's "*globals*" section .

11. User panel

11.1 General view

easyAsterisk has a specific panel the endusers can use to simplify the management of their own local extension. As we seen in the *chapter 4.5* it is possible to define the extensions allowed to access it, for every extension, enable, or not, the access to it. To login, users have to point the PBX's local IP address using their *extension* as login and their *password* as password.

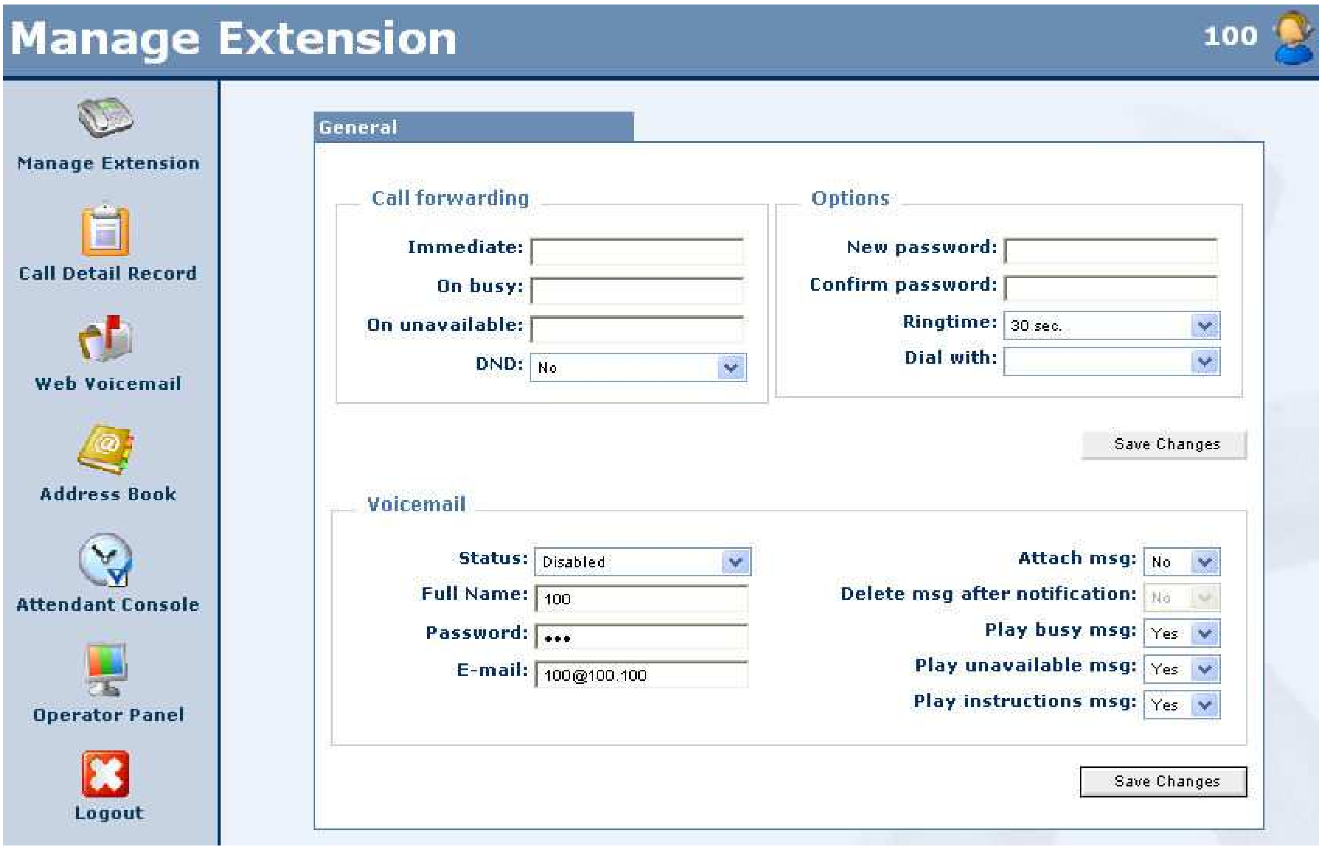


Figure 11.1 – User panel

The screen looks like the Administrator's panel (a top bar with the current menu, a left section with the menu list, a right section with the options).

11.2 Extension Management

Using *Extension Management* esch user can simply configure the behaviour of his extension adding forwardings, configuring voicemail, and other options *(figure 11.1)*. Let’s see it in detail:

CALL FORWARDING:	Using this item it’s possible to set a number (a local etension or an external number provided with all necessary prefixes) to redirect all incoming calls. Another similar service is the DND (Do Not Disturb) that when activated let the user not to receive any kind of call.
OPTIONS:	
New password/Confirm password:	Sets or changes the user’s password.
Ringtime:	Indicates the time, express in second, after which to activate the forwarding or the voicemail services (if enabled).
Dial with:	Sets the route to use when the user calls numbers stored in system or personal address books.
VOICEMAIL:	Enables or disables the voicemail. Settings are the same as described in the *chapter 4.6*.

11.3 Private call detail record

Using *Call Detail Record* menu it’s possible to search the local extenstion’s CDR Database. To make a query the user can follow the same rules described in the *chapter 7;* results can be exported in PDF or CSV document. Records can’t be deleted from the database.

11.4 Web voicemail

Web voicemail is a web interface that lets the user to manage all audio messages recorded in his owm mailbox. (*figure 11.2*).

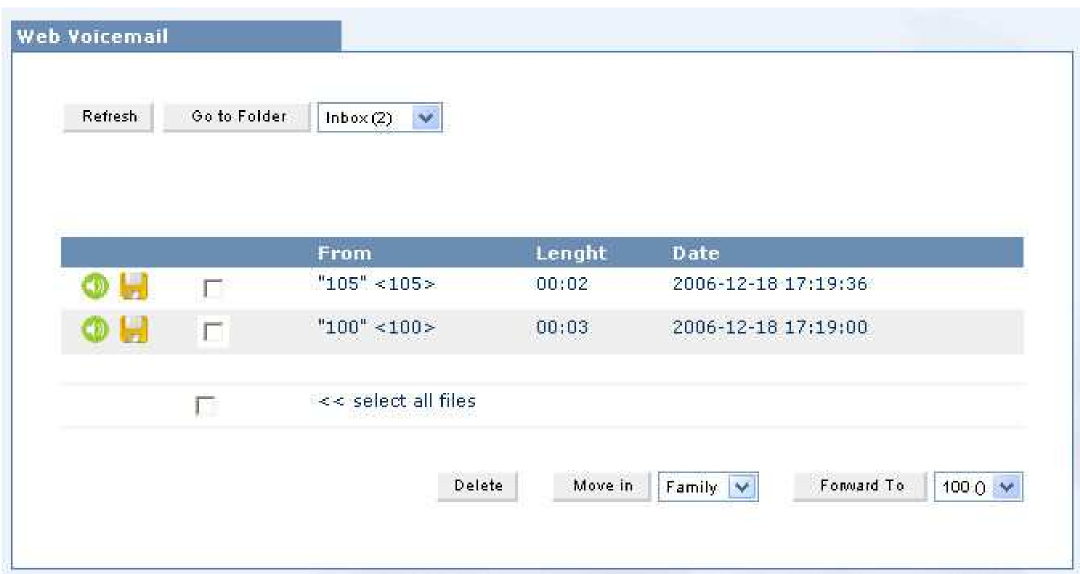


Figure 11.2 – Web Voicemail

Using this interface, messages can be listened, deleted, downloaded, forwarded to other users or moved into folders. The default INBOX is called *"Inbox"* while other available folders are *"Friends"*, *"Family"*, *"Work"* and *"Old”*.

11.5 Managing the address book

Address Book let the user to manage his own personal address book and to access the system-level address book. Items can be called simply clicking on them; the user's extension will start to ring until off-hook and the user can start to talk with the selected number. Using the option *Dial With* in the menu' *Extension Management* it's necessary to set the default route to use for the outgoing calls. Using the *Free Call* item it's possible to digit in a text box the desired number to dial; *Find Contact* searches within the contacts stored in the address books.

11.6 Attendant console and operator panel

Attendant Console and *Operator Panel* menus will appear in the user panel only if they have been enabled by the administrator in the extension options (see *chapter 4.4*). As described before, *Attendant Console* lets modify the routing of incoming calls while *Operator Panel* shows the real time status of extensions, conference rooms, queues and external lines.

11.7 Managing virtual Fax

If Virtual faxes have been configured it's possible to access received documents using a simple interface. When a user has permissions to manage faxes he will see an additiona icon on the left of the panel bar; clicking on it the virtual faxes list will appear (*figure 11.3*).

Figure 11.3 –Fax Management

12. Practical examples

12.1 Two Asterisk PBX connected via trunk ip

In the following example we suppose to have two PBXs we want linked together via IP TRUNK.
PBX1 (IP Address: 192.168.1.1) manages local extensions ranging from 100 to 199
PBX2 (IP Address: 192.168.1.2) manages local extensions renging from 200 to 299.
We need to create an IP link (in this case we will use the sip protocol) to let each local extensions call the other PBX's local extensions. In other words, we want to build a small PBX Virtual Private Network.

PBX1 (192.168.1.1)	**PBX2 (192.168.1.2)**

The configuration of local extensions should follow this scheme:

PBX1 (192.168.1.1)

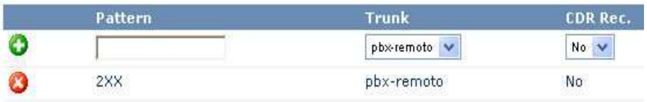
Interno	CID Nome	CID Numero	CID Suffisso Uscente	Prot.	Dtmf	VM	Int. Utente	Reg. CDR
100	100	100		sip	rfc2833	✗	✓	✗
101	101	101		sip	rfc2833	✗	✗	✓
102	102	102		sip	rfc2833	✗	✗	✓
103	103	103		sip	rfc2833	✗	✗	✓
104	104	104		sip	rfc2833	✗	✗	✓

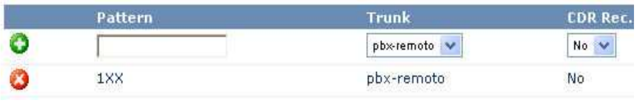
PBX2 (192.168.1.2)

Interno	CID Nome	CID Numero	CID Suffisso Uscente	Prot.	Dtmf	VM	Int. Utente	Reg. CDR
200	200	200		sip	rfc2833	✗	✗	✓
201	201	201		sip	rfc2833	✗	✗	✓
202	202	202		sip	rfc2833	✗	✗	✓
203	203	203		sip	rfc2833	✗	✗	✓
204	204	204		sip	rfc2833	✗	✗	✓

That's the SIP trunk creation:

You can note that we create only the *peer* that handles incoming and outgoing calls. Username and password are not set because the *insecure* option is enabled (that options lets to receive calls without caller authentication). The selected context is *local* to directly address the calls coming from the peer; selecting *incoming* as context, calls are routed to attendant console. We must now to route the calls using *Remote PBX* item/*Calls Management* menu.





Please note that we simply redirect the other machine’s extensions patterns to the created trunk (in both cases the trunk name is pbx-remote).

12.2 "IP Router" set-up

Let’s suppose the following scenario: a traditional PBX equipped with two PRI cards, the first one connected to Telco lines and the second one connected to an easyAsterisk PBX.
easyAsterisk has to be configured as an *IP Router* to accept calls from the standard PBX and to route them to an IP Carrier. To do that, traditional PBX will be configured so that extensions will engage a line using “0” or “9” depending on the user’s will (traditional call, IP call). On easyAsterisk it’s necessary to configure PRI cards by *Hardware Settings (figure 12.1).*

Figure 12.1 – PRI Card Configuration

As you can note, the signalling must be *pri master (NT).* Most often *Overlap* setting must be enabled to instruct Asterisk to consider all the received digits. If echo is experienced, it’s possible to modify *Rxgain* and *Txgain* values making some tests. It’s then necessary to create a ZAP trunk (in our example *called from-pbx, figure 12.2*).

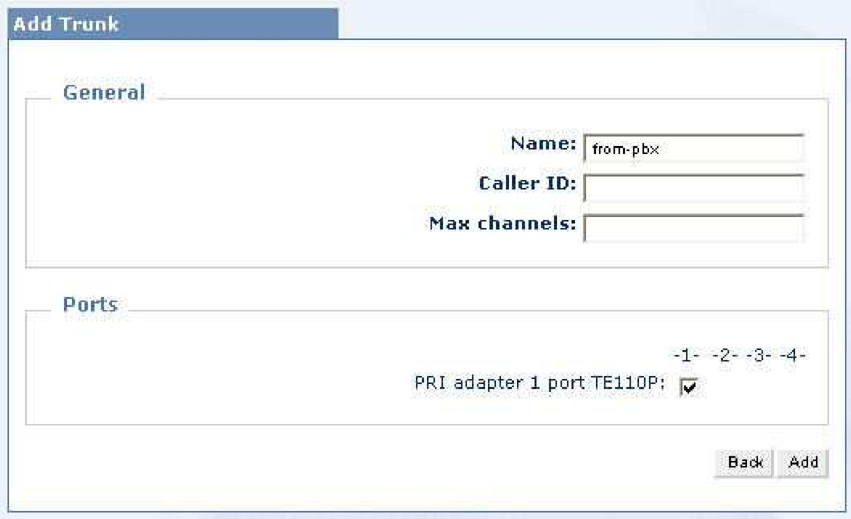


Figura 12.2 – ZAP Trunk creation

We configure two SIP trunks to connect two IP carriers *(figure 12.3)* without setting limits in the outbound channels.

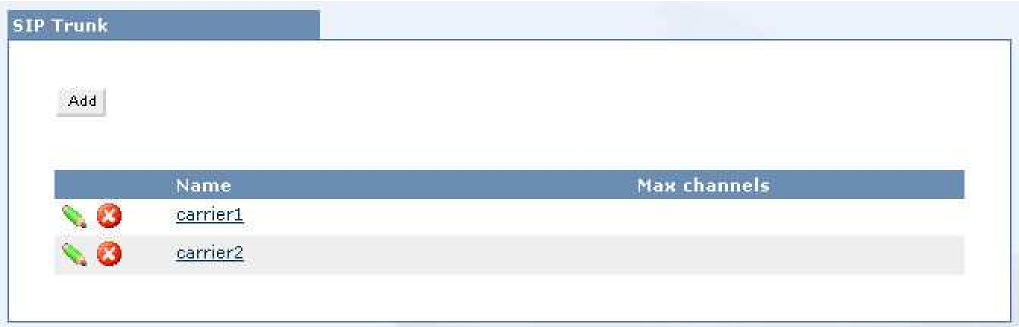


Figure 12.3 – SIP Trunk configuration

We have to set an outbound Route, in this example it's called *Route1*, with a fictitious dialing prefix code.

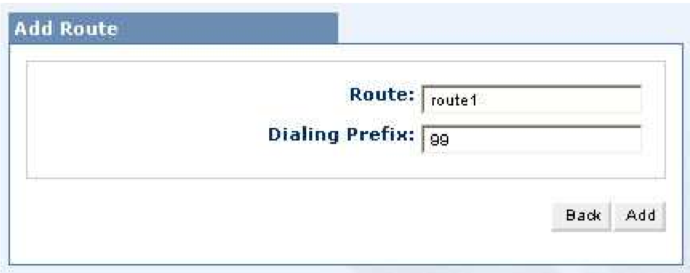


Figure 12.4 – Route configuration

Then we have to configure routings for the just created route. We decide to route all international calls to *carrier1* trunk and local calls to *carrier2 (figure 12.5).*

Figure 12.5 – Configuration routings

The last step is IP Router creation, to instruct easyAsterisk to redirect all the calls coming from the ZAP trunk (*from-pbx*) to *route1 (figure 12.6).*

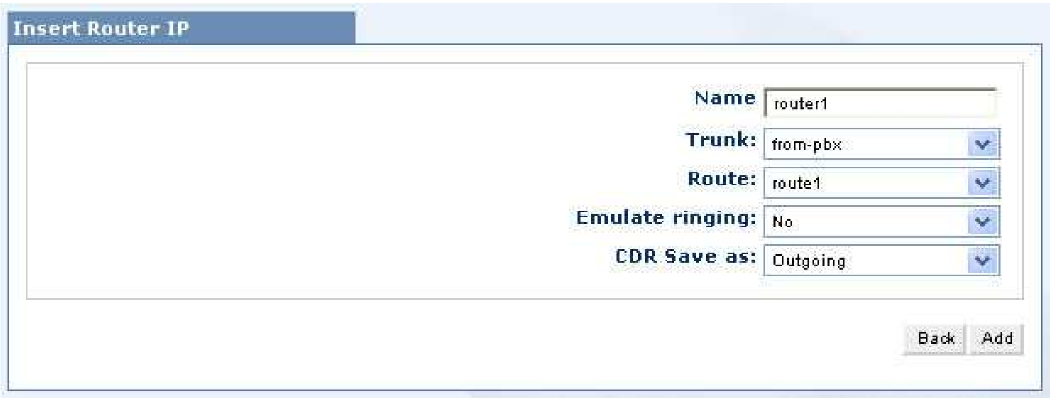


Figure 12.6 – Router IP creation

Well, the configuration should be complete. Local phones can use "0" to place a traditional call and "9" to place an IP call.

13. References

easyAsterisk's reference site is http://www.easyasterisk.org/. On that web site several useful items such as FAQs and FORUM are available.

Please find a small list of some other Asterisk-related internet sites:

http://www.asterisk.org/	Official home page of the famous PBX open source.
http://www.digium.com/	Digium home page.
http://www.junghanns.net/	Bristuff developers, support for ISDN BRI cards.
http://www.voip-info.org/wiki/	Site containing technical informations about Asterisk and the VoIP world in general.