trixbox CE 2.6

Implementing, managing, and maintaining an Asterisk-based telephony system

Kerry Garrison

PACKT PUBLISHING

BIRMINGHAM - MUMBAI

trixbox CE 2.6

First published: February 2009

Production Reference: 1190209

Published by Packt Publishing Ltd.
32 Lincoln Road
Olton
Birmingham, B27 6PA, UK.

ISBN 978-1-847192-99-8

www.packtpub.com

Cover Image by Damian Carvill (damianc@packtpub.com)

Credits

Author
Kerry Garrison

Reviewers
Alan H Miller
Bruce Letterle
David Merel
Jan Bjorkhaug
Kristian Kielhofner
Michael D Mosier

Acquisition Editor
Rashmi Phadnis

Development Editor
Dhiraj Chandiramani

Technical Editor
Mithun Sehgal

Editorial Team Leader
Akshara Aware

Project Manager
Abhijeet Deobhakta

Project Coordinator
Lata Basantani

Indexer
Monica Ajmera

Proofreader
Chris Smith

Production Coordinator
Aparna Bhagat

Cover Work
Aparna Bhagat

About the author

Kerry Garrison has spent over 20 years in the IT industry and was introduced to Asterisk by a long-time friend. After getting Asterisk running at home and finding the **Asterisk@Home** project to make doing installations even easier, Kerry began writing technical articles about using the system to run a business-grade phone system. After seeing the response to his talk about Asterisk@Home at Astricon in 2005, Kerry decided to write the first book about the system. Before the book went to print, the project was renamed to trixbox. After many rewrites to try to stay current, the first *Trixbox Made Easy* book was finally published with some additional help from Barry Dempster.

Shortly after the first book was published, Fonality, the sponsor of the trixbox project, hired Kerry as the trixbox project manager to work alongside Andrew Gillis to keep the project moving forward, and he worked as the Community Director at Fonality until January of 2009. Today, Kerry is doing consulting work again, and working on his new project (`http://pbxtutorials.com`).

The following people have contributed code, their writing, and/or their inspiration to help make this new book possible:

My Family

Without the support of my wife and family putting up with my rants about "I swore I would never write another book", this new book would never have happened. Credits to the family would not be complete without mentioning my dog Lola, who sat by my side during the entire writing of the book and provided a nice break whenever she needed to go for a walk.

Andrew Gillis

While not his intention to help create a revolution in the telephony market, Andrew's little basement project made it simple for millions of users to learn how to use these powerful tools.

Tim Yardley

Known to the world as "Engineer Tim", Tim worked on the trixbox CE project during 2008 as the build engineer. His contribution of information on how to secure a trixbox system was a huge contribution to this book.

About the reviewers

Alan H Miller earned a Bachelor's degree in Chemistry from Northwestern University, Evanston, IL. He worked briefly in the materials science research field before moving into the area of computer application design, development, and product management. The telecommunications industry has provided him with stimulating and rewarding opportunities for the majority of his professional career.

The following is the Alan's list of employers:

- Tel Aviv University, Ramat Aviv, Israel
- Northwestern University, Chicago, IL, US
- Scientific Information Retrieval, Inc., Evanston, IL, US
- ROLM Corporation, Chicago, IL
- IBM Corporation, Chicago, IL
- Siemens, Chicago, IL; San Jose, CA; Munich, Germany

Bruce Letterle has worked in the technology industry since 1979 with a variety of system integration firms. He founded Creative Web Technologies in 1996, a software integration firm that specializes in open source products.

David Merel is the founder and CEO of ThinkBright LLC (`www.thinkbright.net`), a cutting-edge Voice over IP carrier providing businesses of all sizes with sophisticated and low cost VoIP solutions.

David started ThinkBright in 2005 and continues to manage the company and its employees, all of whom are dedicated IT professionals. He acts as the company's chief architect, designing new technologies that have added significant revenues to the company's operations. During his years at ThinkBright, David has worked with the latest Voice over IP technology, including all VoIP equipment from major manufacturers such as Cisco, Polycom, Grandstream, and Aastra. He also works with customers ranging from small businesses to Fortune 500 companies, and interacts with system integrators and IT consultants who call ThinkBright on a daily basis for assistance with all of the VoIP solutions that Thinkbright offers. ThinkBright manages hundreds of PBX systems, providing customers with PBX features such as Auto Attendants, Waiting Rooms, and Ring Groups, and also provides SIP or IAX services for customers managing their own PBX system.

David has many years of experience with trixbox and Asterisk, and has installed countless custom configurations and deployments using those solutions.

David earned a Bachelor of Arts triple majoring in philosophy, politics, and law from SUNY Binghamton. David holds a CCNA (Cisco Certified Network Associate) certificate and is proficient in over 10 programming languages and databases, various operating systems, VoIP and related communication solutions, and other open-source business applications.

Jan Bjorkhaug was born in Bergen on the west coast of Norway, 49 years ago. He is happily married with four children ranging from 17 months to 25 years of age. He is the CEO of Omnipro, Netsecur, and VoIPtel.

During the last 15 years, Jan has established five IT ventures located in Norway, the Philippines, and most recently the VoIPtel Distribution Central located in Breda, the Netherlands.

He, Bruce, and Adelina, they are currently responsible for the VoIPtel CE and SE firmware used by the embedded open-source series of IP PBXs manufactured by Atcom Technology Co. Ltd. These small but powerful PBXs are made available through their webshop as well their network of international retailers.

For more information, please visit `http://voiptel.no` or `http://blog.voiptel.no`.

> I would like to thank my son Bruce for his invaluable input during my review of this book, as well as my wife Aileen Joy and the rest of my family for their enormous patience and understanding when my work kept me away from them.

Kristian Kielhofner is VP, Systems Engineering for Star2Star Communications, developer of an end-to-end VoIP architecture. Kristian is responsible for the design and implementation of Star2Star's VoIP services. When he's not consulting or working for Star2Star, he enjoys tinkering with one of his FOSS projects—AstLinux, Runnix, or Recqual. You can find out what else Kristian is up to on his blog at `http://blog.krisk.org`.

Michael D Mosier started messing around with computers when he was 8 years old, on a TRS 80. He has a BFA in both Drawing/Painting and Graphic Arts and a minor in Computer Science. He has been in the IT business for over 20 years, working with setting up the first networks in several Graphic Design firms and Printing companies.

He started learning Asterisk and trixbox about four years ago as a way to teach himself IP-based phone systems. Knowing that technology was moving toward all IP-based systems, he thought it would be a great way to get his company and himself a jump on the new technology.

He is one of the Principals at Select Phone Solutions in Houston, Texas. They have trixbox work all over the states of Texas and Louisiana. They have been successfully installing trixbox CE for over two years. He also holds the technician FtoCC.

Table of Contents

Preface

trixbox CE is an open-source IP PBX system based on the Asterisk Open Source **PBX** (**Private Branch eXchange**) software. Using trixbox CE, you can set up a PBX system to run your business that will have features previously only available to companies with huge budgets. With the flexibility of **VoIP** (**Voice over Internet Protocol**), a trixbox CE system can be extremely versatile and provide features such as remote users and branch office support.

This book will show you all of the features of trixbox CE with information about how to best utilize those features to create a system that will serve your needs as well as keep your incoming callers from being annoyed.

This book starts with the basics and works up through all the features, building on what you are learning, to enable you to build a feature-rich and well-laid-out phone system.

What this book covers

Chapter 1 introduces you to the basics of open source PBX systems, describes how they handle calls, how they connect to the existing telephone networks, and what the software is like under the hood.

Chapter 2 delves specifically into what trixbox CE is and how it is different from Asterisk or other open source PBX platforms.

Chapter 3 shows you how to install trixbox CE and goes through all of the main sections of the user interface that we will be using to configure the system.

Chapter 4 gets into the configuration of all of the primary features of trixbox CE including setting up of phones, extensions, and IVR menus.

Chapter 5 takes you through the planning of a successful deployment and how to properly structure a system for best results.

Chapter 6 takes you through hardware configurations such as TDM cards and hard phones.

Chapter 7 covers the different aspects of network issues that you need to take into account and how different network designs can affect the performance of your system.

Chapter 8 talks about advanced trunking methods for connecting multiple systems together for connecting business units, branch offices, and so on.

Chapter 9 is a complete overview of the PBX functions that are built into the system and shows how to add your own functions if you need to.

Chapter 10 covers the PBX features in detail with best practices on using the features for best results.

Chapter 11 covers general maintenance tasks and reporting functions.

Chapter12 takes you through the steps of troubleshooting your system when things are not working the way they should.

Chapter 13 goes through some of the utilities that are included with trixbox CE to make using your system easier.

Chapter 14 will help you design a good call flow that accomplishes the needs of the company and keeps callers from hanging up.

Chapter 15 explains how to secure your trixbox CE system and keep unwanted people out of the system.

Chapter 16 shows you some of the features of the upcoming HUD (Version **3**) that is coming to trixbox CE soon.

Chapter 17 introduces you to trixbox Pro, the big brother to trixbox CE that is a commercial PBX package.

What you need for this book

Anyone with enough technical knowledge to figure out how to burn a CD can install and configure a basic trixbox system; however, to properly manage and maintain a system, you will need a base level of Linux skills. Throughout this book, we will look at many of the Linux commands you will need to get started.

Who this book is for

If you want to learn how to install and configure either trixbox CE systems or Asterisk-based PBX systems, without struggling with confusing configuration files and cryptic scripts, this is "the" book for you. This book will also benefit readers who use trixbox CE and want to learn all its features, and how best to use them. It's ideal for any user wishing to set up a telephony system for small business usage.

No previous knowledge of trixbox or networking is required, although some basic knowledge of PBX and Linux would be an advantage.

Conventions

In this book, you will find a number of styles of text that distinguish between different kinds of information. Here are some examples of these styles, and an explanation of their meaning.

Code words in text are shown as follows: "Once we are in the Asterisk CLI, we can see what accounts are registered by using the `sip show registry` command."

A block of code will be set as follows:

```
A if [ ! "$TOS_0" == "" ]; then
for i in `echo $TOS_0 | tr ',' ' '`; do
i=`echo $i | tr '_' ':'`
$IPT -t mangle -A PREROUTING -p tcp --sport $i -j TOS --set-tos 0
done
fi
```

When we wish to draw your attention to a particular part of a code block, the relevant lines or items will be made bold:

```
if [ ! "$TOS_0" == "" ]; then
for i in `echo $TOS_0 | tr ',' ' '`; do
i=`echo $i | tr '_' ':'`
$IPT -t mangle -A PREROUTING -p tcp --sport $i -j TOS --set-tos 0
$IPT -t mangle -A PREROUTING -p udp --sport $i -j TOS --set-tos 0
done
fi
```

Any command-line input and output is written as follows:

```
trixbox1*CLI> sip show registry
Host                          Username       Refresh     State
inbound3.vitelity.net:5060    kgarrison1     45          Registered
```

New terms and **important words** are introduced in a bold-type font. Words that you see on the screen, in menus or dialog boxes for example, appear in our text like this: "Add a new SIP account by clicking on the **Add...** button. This will bring up the SIP Account Properties page."

> Warnings or important notes appear in a box like this.

> Tips and tricks appear like this.

Reader feedback

Feedback from our readers is always welcome. Let us know what you think about this book, what you liked or may have disliked. Reader feedback is important for us to develop titles that you really get the most out of.

To send us general feedback, simply drop an email to `feedback@packtpub.com`, making sure to mention the book title in the subject of your message.

If there is a book that you need and would like to see us publish, please send us a note in the **SUGGEST A TITLE** form on `www.packtpub.com` or email to `suggest@packtpub.com`.

If there is a topic that you have expertise in and you are interested in either writing or contributing to a book, see our author guide on `www.packtpub.com/authors`.

Customer support

Now that you are the proud owner of a Packt book, we have a number of things to help you to get the most from your purchase.

Errata

Although we have taken every care to ensure the accuracy of our contents, mistakes do happen. If you find a mistake in one of our books—maybe a mistake in text or code—we would be grateful if you would report this to us. By doing this you can save other readers from frustration, and help to improve subsequent versions of this book. If you find any errata, report them by visiting `http://www.packtpub.com/support`, selecting your book, clicking on the **let us know** link, and entering the details of your errata. Once your errata are verified, your submission will be accepted and the errata added to the list of existing errata. The existing errata can be viewed by selecting your title from `http://www.packtpub.com/support`.

Piracy

Piracy of copyright material on the Internet is an ongoing problem across all media. At Packt, we take the protection of our copyright and licenses very seriously. If you come across any illegal copies of our works in any form on the Internet, please provide the location address or website name immediately so we can pursue a remedy.

Please contact us at `copyright@packtpub.com` with a link to the suspected pirated material.

We appreciate your help in protecting our authors, and our ability to bring you valuable content.

Questions

You can contact us at `questions@packtpub.com` if you are having a problem with some aspect of the book, and we will do our best to address it.

1

Introduction to open source PBX systems

In this chapter we will look at:

- What a PBX is
- What types of PBX systems are available
- The role of open source in today's PBX systems

What is a PBX?

Private Branch Exchange: A Private Branch eXchange (also called PBX, Private Business eXchange, or PABX for Private Automatic Branch eXchange) is a *telephone exchange* that serves a particular business or office, as opposed to one that a *common carrier* or telephone company operates for many businesses or for the general public.—*Wikipedia*

A PBX is the middleman between the phone company and/or **Internet Telephone Service Provider** (**ITSP**) and the extensions within your office. The PBX system provides all of the telephony features for your organization such as extensions, voicemail, music on hold, call transfers, call parking, and many other functions. Early on, the routing of calls was done by banks of workers (usually female) that would connect calls by plugging and unplugging wires to connect a caller to a phone that was connected to the system. Eventually technology improved and moved to automated systems to handle the call routing and extension management.

As the telephone network evolved and companies became more reliant on their telephone connections, the routing functions moved from the phone company to being inside the individual companies, first using the same switchboard technology, and eventually moving into large rooms full of switches and relays.

Traditional PBX

It isn't hard to spot a traditional PBX system; it's usually a large box full of mechanical switches and relays mounted on a wall in 'the phone room'. When a company needs additions, moves, and changes, they need to call out 'the phone guy' to make changes to the system.

With a traditional PBX system you would purchase your phone system and be locked into a very small number of telephone sets, always from the same manufacturer of the PBX system, and usually only a very small number of devices to choose from. Adding features like voicemail is usually an expensive add-on to the base system. Although some legacy PBXs now have options for network access and VoIP functionality, these options are often very expensive upgrades.

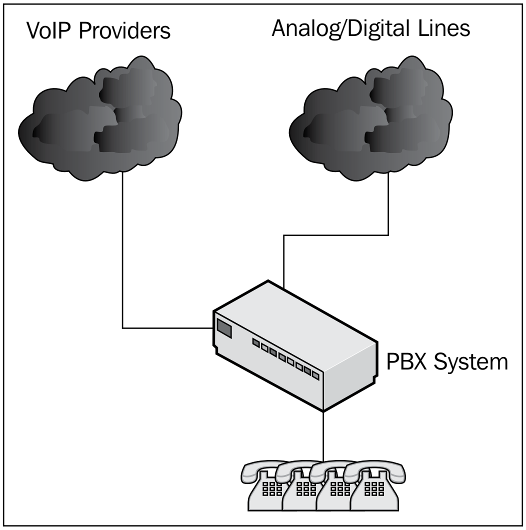
Hybrid PBX

A hybrid PBX combines the features of a traditional PBX system with **VoIP** (**Voice over IP**) functionality. In some cases the VoIP functionality may just be the way the PBX communicates with the phones, other VoIP functions may include the ability to have remote extensions or softphones, and still other VoIP functionality may include being able to use internet telephone service providers and not just the traditional public telephone network.

The Asterisk PBX system is a full hybrid system combining numerous types of connections to the public telephone network as well as VoIP functionality including:

- Use of industry-standard SIP-compliant phones
- Remote extensions using either SIP-compliant phones, or SoftPhones
- Use of Internet Telephone Service Providers
- Bridging of remote Asterisk systems together to act as a single system

Every business, large or small, needs some kind of phone system but many small businesses have never been able to afford all of the features and functions that a good PBX can provide.



With the advent of IP PBX systems based on open source software, the cost of creating a business-class PBX system has plummeted over the last few years. Since an IP PBX such as **Asterisk** or **SIPx** incurs only the cost of the PC hardware and the labor to install it, virtually any small business can now afford a PBX system with all of the features and functionality that only Fortune 500 companies could afford before.

Open telephony projects

There are a number of open source telephony projects available these days besides trixbox/Asterisk. A few examples of other open source projects are the following:

- **SIPx**: (`http://www.sipfoundry.org`) SIPx is a pretty complete SIP-based PBX system with its own web interface for configuration. While SIPx is pretty powerful, it doesn't have the following that Asterisk-based systems have.
- **Call Weaver**: (`http://www.callweaver.org`) Call Weaver started as a fork of Asterisk **1.0.9** and is working on replacing all of the problems that developers feel were wrong with Asterisk. Some of these items include the SIP stack, voicemail, queues, and other primary components. The Call Weaver core is very streamlined and works very efficiently.
- **FreeSwitch**: (`http://www.freeswitch.org`) FreeSwitch is more of a complete softswitch solution that is currently geared towards high-end switching versus being a PBX platform although people are working on adding the applications that are needed for full PBX capability.

The open source dilemma

Using open source software is not for everyone; there are exceptionally good open source packages available and there are some really bad ones as well. Two of the biggest problems with open source software are a lack of commercial support and the project going stale because of lack of interest or lack of funding.

When looking at an open source package for use in your business, you should always look for a commercial support option as well as a means by which the project is funded. With trixbox, you have commercial support from not only *Fonality* but from many third-party support organizations. Secondly, since the project is funded by *Fonality*, the project has a dedicated development team that works on the project full time to create new features, fix bugs, and ensure system reliability.

While I am a big proponent of using open source software whenever I can, there is a time and a place for it. You have to use the right tool for the job. Unfortunately, projects like OpenOffice.org don't have enough compatibility with Microsoft Office products to really be a true alternative, while the GIMP, which is an image processing tool, is more than sufficient for most people who can't afford Adobe Photoshop.

Asterisk and trixbox are specific tools that interface very nicely with traditional phone circuits taking compatibility out of the loop. With the features it offers and the substantial cost savings, a trixbox telephony solution is a viable alternative to a traditional PBX system.

The PSTN

The **Public Switched Telephone Network** is the backbone of the world's communication infrastructure. This is the network that is controlled by the traditional telephone companies. This remains the primary network for calls placed and received around the globe and there are numerous interfaces into this network.

- **POTS**: POTS stands for **Plain Old Telephone Service**; this is the standard analog lines that we all have grown up with in our homes and businesses. POTS is an analog system that is controlled by electrical loops. It is delivered to the client over copper wires running in the homes and offices and is one of the easiest and cheapest forms of telephone service available.
- **ISDN**: **Integrated Services Digital Network** is a faster and more robust connection that is quite a bit more expensive. This type of service gained popularity within small to medium-sized businesses as a cost-effective way of connection to the PSTN and getting some advanced services, like multiple lines to one office or being able to combine voice and data on one circuit. ISDN is a digital service and offers more features over POTS lines.

- **T1/E1 PRI**: **Primary Rate Interface** circuits offer large numbers of voice channels delivered over a digital T1 or E1 circuit; since these lines are fairly expensive, they are usually only used in companies requiring 15-20 lines or more.

The trouble with the PSTN is that it is very static and in most countries it is strictly controlled by the telecommunications companies and even governments. If a business wants to make a lot of internal calls using the PSTN, it is by no means a cheap way to communicate. ISDN/T1/E1 circuits are most commonly found at the external interface of a company's communication network, with all the internal communications going through internal lines that are controlled by an internal telephone system.

Voice over IP

Now that we understand the basics of a PBX system it's time to look at how **VoIP** fits into the equation. In a modern IP-PBX system, VoIP can be used in the following areas:

Connecting phones to the PBX system:

This is usually done over the existing data network that the computers are using. This can be with either local phones or telecommuters.

Connecting to a VoIP service provider:

This provides the ability to get 'dialtone' provided over the Internet to your PBX system. The VoIP service provider then terminates your calls to the PSTN network off of their network.

Connecting PBX systems together:

This is used for connecting branch offices together to allow interoffice communications without the cost of running phone lines between the offices.

It is important to note that VoIP connections for telephone service are not yet a complete replacement for traditional PSTN lines. A VoIP service is certainly an alternative way to connect to the PSTN as you can make and receive calls to the PSTN network through your VoIP service provider.

As the Internet grew, became faster and more reliable, it became apparent that it was possible, and in many cases, preferable to use the Internet for carrying voice as well as data. There were more than a few problems in making voice communication a viable alternative over the Internet. For example, data connections can tolerate a certain amounts of lag, packet loss, and jitter while voice communication will suffer severe quality issues under the same conditions. Network lag and jitter can result in symptoms such as echo, jitter, and garbled communication. If you have ever watched a news broadcast with an overseas reporter using a sat-phone and heard the drop-outs and choppiness, while this may be acceptable for a brief news report, having that same quality on your business-to-business phone calls is simply not going to be acceptable.

While we think of VoIP as being a new technology, we have been using VoIP communication over the Internet for well over two decades. The reason for the increased popularity is that being able to have remote workers with phones tied directly to the PBX system is now possible; the ability to connect remote offices is relatively painless, and being able to use VoIP service providers for call origination and termination is now a viable alternative.

The biggest problem with VoIP service over the Internet is that Internet Service Providers do not allow **Quality of Service** (**QoS**) packets to prioritize voice traffic over other data traffic. Due to this, VoIP over the Internet is very prone to problems that are outside of the control of yourself or your VoIP provider.

On the plus side, VoIP circuits can be very cost-effective, especially for long distance calls through large providers, which may mean your calls can be routed over the Internet to a termination point physically close to the location you are trying to call. Because of these kinds of network reach, very often long distance rates from VoIP service provides can be substantially less than when using PSTN circuits.

Asterisk—the brain of the PBX

Asterisk is the core piece of software that handles the call flow and PBX functions within the system. In essence, Asterisk is a toolkit that can be used to create different types of telephony-based applications such as a security system, conference room system, PSTN termination system, as well as the obvious PBX and IVR (interactive voice response) system that we will build with trixbox. With currently available PCI cards and external devices it is possible to create an enterprise-class PBX system using Asterisk, commodity PC components, and a few pieces of extra gear. Because most of the components are off-the-shelf computer parts and open source software, the cost of building an extremely powerful business phone system is only a fraction of what a traditional phone system would cost.

Asterisk is, simply put, 'just software'. Instead of using banks of dedicated processors, all of the call-flow and routing is done via software. This is really what makes Asterisk so popular; what used to take a wall full of mechanical switches and relays is all done on a regular PC using a piece of open source software. Since traditional systems are closed and proprietary, finding someone to manage and maintain a legacy system can be quite expensive.

Implementing the phone system functions entirely through software opens up countless possibilities for even more advanced features such as predictive dialing, database integration, and virtually anything we can imagine interfacing with or even controlling with a phone system. All of this and much more can be accomplished with a little ingenuity and a little bit of programming. We don't have to be a programmer to use an Asterisk-based PBX system, but being able to do programming and create custom dialplans can greatly increase the types of features you can create and implement. If you are looking for a very specific feature that hasn't been created yet, you can post bounties to the message boards to find someone who can create that functionality for you. Being able to easily find developers and coders to help add functionality is one of the key advantages of the open source model. This type of flexibility is basically unheard of with traditional systems. Developing new functions and features and giving them back to the community is a great way of contributing to the project and helping other people to share in what you have created. This is one of the core philosophies of the open source model. Not everyone will contribute their code that they develop back, nor is it a requirement (more on this later), and many companies have developed closed source software around the open source Asterisk code. Some VoIP service providers use Asterisk for their infrastructure and switching, and companies like Fonality and Digium have closed source commercial packages that use the Asterisk Open Source engine under the hood.

With so many companies using Asterisk to run their businesses on, a wide range of common functions and external features have already been developed and are available for everyone to use.

What Asterisk isn't

Asterisk by itself is not an easy system to set up and configure. Just getting basic services up and running requires an in-depth knowledge of Asterisk configuration files that need to be edited and maintained. By default, there is no built-in interface available to set up and configure Asterisk. Moreover, the few available management tools typically have to be manually installed and configured separately. However, the benefit of trixbox is that much of this is hidden away from the user, and with the aid of this book, it should be very easy to get a system up and running with a very little effort. Asterisk on its own is not:

- A plug-and-play telephone or IVR system
- Easy to configure without training and/or adequate documentation
- A full hardware solution
- The complete solution to every business's communication needs
- Useful without telephony experience or knowledge

While Digium does offer a Business Edition of Asterisk, the open source version does not come with any technical support. Being an open source project, we need either to be able to troubleshoot ourselves, turn to community forums for support, or hire a consultant to help us out.

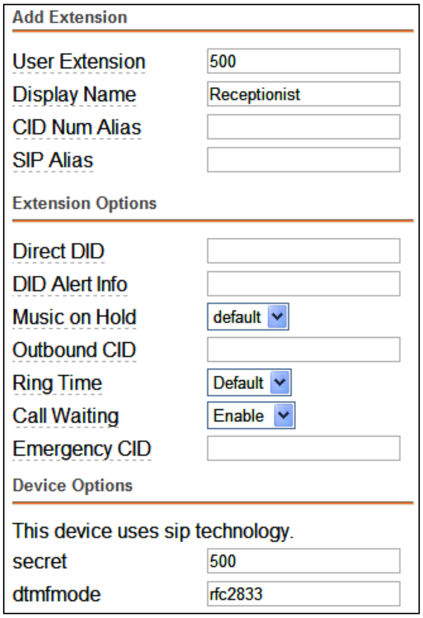
If we are fairly proficient with Linux, we are well ahead of the crowd that typically looks at the trixbox system. While Asterisk is just another software package that runs on almost every Linux distribution, trixbox is a distribution of its own, which abstracts some of these often complicated layers from us. So at some point, Linux skills may be helpful for optimizing the system or troubleshooting; certainly many people with no previous Linux experience have been very successful in rolling out trixbox systems. As we will see, trixbox can make many of the potential issues much less of a concern and more straightforward.

You can't install Asterisk on your standard PC and then plug into any PSTN interface you choose without the correct hardware. If we want to access the PSTN (which is not a requirement if you want a VoIP-only system), we may want to get services from an ITSP (Internet Telephone Service Provider) or install hardware in our server in order to provide access to our POTS, ISDN, or other line types.

Config files—the barrier to entry

One of the biggest hurdles that Asterisk faced in getting massive deployment was that you have to learn how to set up all of the different configuration files. It is true that if you are a `config` file master, you can make Asterisk do things that are simply not possible in the trixbox GUI as trixbox is limited to only those features it has been programmed to configure for you. The good news is that very few companies ever need features that aren't included in trixbox.

The following example shows the screen used in trixbox to create a simple extension and the corresponding configuration file.




```
[500]
type=friend
secret=500
record_out=Adhoc
record_in=Adhoc
qualify=yes
port=5060
nat=yes
mailbox=500@device
host=dynamic
dtmfmode=rfc2833
dial=SIP/500
context=from-internal
canreinvite=no
callerid=device <500>
call-limit=50
```

While trixbox CE does make using Asterisk dramatically easier, it is certainly a good idea to really get in and learn all you can about the Asterisk configuration files. The more you know about how the system works under the hood the easier it will be for you to troubleshoot problems and even add features to your systems that aren't available in trixbox CE.

Additional Asterisk information

Without a doubt, the most complete source of information on using and configuring Asterisk is the book *Asterisk: The Future of Telephony* written by Jim Van Meggelen, Leif Madsen, and Jared Smith. The printed version of the book weighs in at a hefty 444 pages and covers everything you would need to know to configure an Asterisk system from scratch using manually-coded configuration files.

As a bonus, the book is also available as a free PDF download which can be found at `http://asteriskdocs.org`.

The history of Asterisk

Asterisk is the brain child of Mark Spencer and began life as a simple project to use traditional PC hardware to interface to regular phone lines. A few years and countless lines of code later, Asterisk is currently the leading open source PBX software.

Asterisk is written in C for Linux-based systems and has a wealth of features that any business owner would love to have. The following is a very incomplete list of some of the Asterisk's many features.

- **Automated Attendant**: An automated system for answering incoming calls and routing them based on the caller's responses to voice prompts.
- **Blacklists**: The ability to easily add numbers into a central database that will prevent calls from the blacklisted phone number from being processed by the system.
- **Call Detail Records**: Detailed call reports and usage statistics to show an administrator the activity of the phone system.
- **Call Forward on Busy**: Automatically forward a call to another extension if the called extension is busy.
- **Call Forward on No Answer**: Automatically forward a call to another extension if the called extension does not answer.
- **Call Parking**: Placing a call into a holding state so that it can be picked up at another extension.
- **Call Queuing**: A system that allows inbound callers to sit in a holding room listening to music on hold until the next available agent is available to speak to them.
- **Call Recording**: The ability to record inbound or outbound calls to `.wav` files.
- **Call Routing**: Based on the phone number that was dialed (**DID**), or the number that was called from (**ANI**), a call can be routed to a specified extension, group, queue, and so on.
- **Call Transfer**: The ability to transfer an existing call to another extension.
- **Caller ID**: Caller ID is used to display the phone number and other available information of the user that is calling into the system.
- **Conference Bridging**: Asterisk has the ability to create conference rooms that multiple people can attend at one time for group meetings.
- **Interactive Directory Listing**: A company directory system that can look up users by first or last name.
- **Interactive Voice Response** (**IVR**): This system used pre-recorded voice menus to prompt callers to make selections via their phone such as 'press 1 for sales, 2 for support'.

- **Music On Hold**: Asterisk can play MP3 files to callers that are on hold or waiting in a queue.
- **Remote Office Support**: Because Asterisk uses Internet Protocols for communication, users can be at remote locations and have access to the system over broadband internet access.
- **VoIP Gateways**: Using new Internet Telephone Service Providers allows an Asterisk system to receive telephone network access without having to use access from a normal analog service provider.
- **Voicemail**: Each user in an Asterisk system can have their personal extension and voicemail account. Using trixbox, the voicemail can be retrieved via their phone, from a remote location, sent via email, or accessed via a web browser.

Current versions of Asterisk support some very interesting technologies such as storing voicemail in IMAP folders or ODBC connections, Dictation and Follow-Me applications, and much improved jitter buffers and many other new things being added all the time.

The problem is that Asterisk is not easy to install and get running and many people got turned off early on by the complexity of the `config` files. It was only a matter of time before people started putting together graphical user interfaces to create the complex dialplans. The first of these was the **Asterisk Management Portal** (**AMP**), which later became FreePBX. Digium and many Asterisk purists always believed that the *right* way to build a system was using `config` files and not be bound by the design of a graphical user interface.

With the popularity of trixbox CE, even Digium has created its private GUI for Asterisk, which can easily be installed with the AsteriskNow package. At the time of this writing, the AsteriskNow project is in its first full release, and while lacking in many features, it is a fairly nice AJAX-type web interface.

Summary

In this chapter we have looked at different types of PBX systems and how Asterisk plays a role in open source PBX implementations. Asterisk-based systems have numerous features, and since Asterisk has a built-in programming language, many new features can be and have been added by end users and programmers. We will see some of these customizations in the later chapters. We also have an idea of the history of Asterisk and why it came into fairly popular use. We have information on web sites and books for further information on Asterisk, if necessary. However, from now onwards we will be very trixbox-specific with our discussions.

2
What is trixbox?

trixbox began life in November of 2004 when Andrew Gillis built the first ISO installer, which he named **Asterisk@Home**. His only goal at the time was to build a simple installer that he himself and a few friends could use to make installing an Asterisk system faster, easier, and not prone to human error. Over the next year Asterisk@Home got new features such as the **Asterisk Management Portal** (**AMP**, which later became **FreePBX**). Little did Andrew know at the time that his little project would become one of the key components in getting Asterisk-based systems into businesses around the globe.

In this chapter we will look at the core pieces that make up the trixbox CE project, hardware requirements, circuit types, and optional equipment you will need for your first installation.

Asterisk@Home to trixbox—the name change

As already mentioned, trixbox started out being called Asterisk@Home; eventually the name was changed because the word Asterisk, when used in the context of the PBX system, is a trademark of *Digium Ltd*. The name change was quite fortunate for many people who were selling systems to clients as the '@Home' moniker did not convey a lot of confidence in many businesses nor did it speak to the robust feature set that the product has.

trixbox, trixbox CE, Pro, SE, EE, CCE?

When the trixbox name began, there wasn't any concept of additional products that would carry the trixbox name, so the product was simply called trixbox. In the middle of 2007, Fonality launched trixbox Pro, which is a hybrid-hosted product based on the technology developed for Fonality's PBXtra product. trixbox Pro will be discussed in more detail in Chapter 17. To avoid confusion, the existing open source product was renamed to trixbox CE (community edition). Therefore this book is mostly about trixbox CE although there are some concepts that cover all versions and there is a complete chapter about what trixbox Pro is all about.

Prerequisite skills

While anyone with enough technical knowledge to figure out how to burn a CD can install and configure a basic trixbox system, to properly manage and maintain a system, you will need a base level of Linux skills. Throughout this book we will look at many of the Linux commands you will need to get started.

trixbox components

The trixbox system is made up of a number of components each of which is released under the GPL open source license. The main benefit of trixbox is that these components are preinstalled and configured to run. This reduces the effort involved in setting up these applications compared to trying to accomplish this manually. At the time of this writing, trixbox is at version **2.6** and contained the following components:

- **CentOS 5.2**: CentOS is a community-supported version of the Red Hat Enterprise Linux distribution. This is the core operating system used by trixbox.
- **Asterisk 1.4**: Asterisk is the PBX engine under the hood. Version **1.4** is the current version of the Asterisk software.
- **FreePBX 2.5**: FreePBX is the graphical user interface package that is used to manage the PBX functions by creating the configuration files used by Asterisk.
- **Flash Operator Panel** (**FOP**): The Flash Operator Panel is a switchboard application that a receptionist can use to see the status of all of the extensions and telephone circuits.
- **trixbox CE Dashboard**: The Dashboard is the user interface that is developed by the trixbox CE development team. The Dashboard contains various tools and utilities to help manage and maintain the entire system.

- **Automated Installation Tools**: All of the tools, operating system, scripts, and `config` files are automatically installed and configured for use by the trixbox setup scripts.
- **Festival Speech Engine**: Many of the functions within Asterisk require some text-to-speech capability. The Festival Speech Engine provides this functionality.
- **Digium Card Auto-Config**: During the installation process, Digium cards are automatically detected and configured.

What is trixbox CE all about?

Originally, trixbox CE (then Asterisk@Home) was a simple ISO-based installer that would format your drives, install CentOS, and then compile Asterisk from source and install the additional third-party tools. This made setting up a system as easy as burning a CD and booting a machine off the CD. This concept evolved into trixbox CE being an application platform with tools and utilities designed to make managing the entire system as well as the PBX functionality easier for the end users. Some of these tools are:

- Package Manager to help manage installing packages and updates
- Endpoint Manager to help provision phones
- Web MeetMe to manage conference rooms
- RAID monitoring and maintenance module
- DHCP configuration tool
- Enhanced Call Data Reports
- Backup/Restore module

trixbox CE also makes it easy to install third-party tools and utilities such as:

- Phone firmware
- Drivers for PCI cards
- Munin system monitoring
- Webmin management tool
- LumenVox drivers
- IVR Graphing tool

As you can see, trixbox is more than just a simple-to-install Asterisk package like so many others, instead it has evolved into its own application platform with PBX/Telephony services being at its core. The future plans for trixbox take this concept even further.

Difference between Asterisk and trixbox

I like to describe the relationship between Asterisk and trixbox as Asterisk being the engine and trixbox the car; you need an engine to power the car but an engine isn't that useful without all the other components. You can certainly build the 'car' by editing the `config` files, but that would be like having bare metal and having to fabricate the chassis, doors, roof, and so on by hand versus buying a car that is all assembled for you.

The core strength of trixbox is its simplified installation and easy configuration. Although we will cover many other tools, a large amount of this book is spent inside the PBX Settings interface, which is based on the FreePBX project code.

The limitations of trixbox

One of the more common complaints I hear about trixbox is that it is too limiting for some people. While it is true that you can make Asterisk do more if you hand code configuration files than you can do in the web interface, you can still accomplish virtually everything most companies would need. Secondly, you can always add custom dialplan code in the `config` files if you really need to do something that isn't supported in the web interface. We will be looking at several examples of this in different areas of the book.

Fonality support of trixbox CE

When Fonality acquired the trixbox project, the goal was to support the trixbox project in order to give back to the community. This support allowed the team to hire dedicated engineers who work on the project full time. This support also allowed the team to create products and services to help enhance the product and provide training to users. These include:

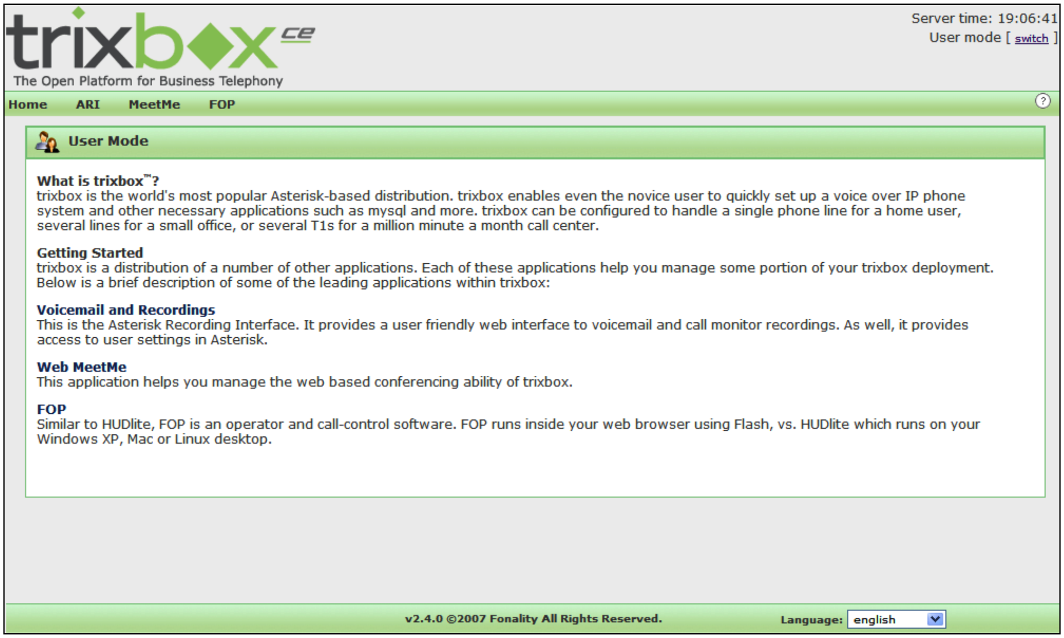
- **The trixbox appliance**: A 3u server with mirrored hard drives and LCD screen pre-loaded with either trixbox CE or trixbox Pro
- **Paid Support options**: Fonality offers both hourly and yearly support options for trixbox CE and trixbox Pro users
- **FtOCC training**: Fonality trixbox Open Communication Certification provides both administrative and technical training for trixbox CE and trixbox Pro users

trixbox CE features

Besides the simple installation that we will discuss in the next chapter, trixbox CE provides a simple web interface to help manage and maintain your system. The trixbox CE dashboard function is split between User functions and Admin functions. The User section is accessible to all users (although some sections are protected) and the Admin functions are protected by a single master administrator password.

User functions

If you point a web browser to the IP address of the trixbox CE server, you will get redirected to the User section of the dashboard. Within the User section are the following modules:

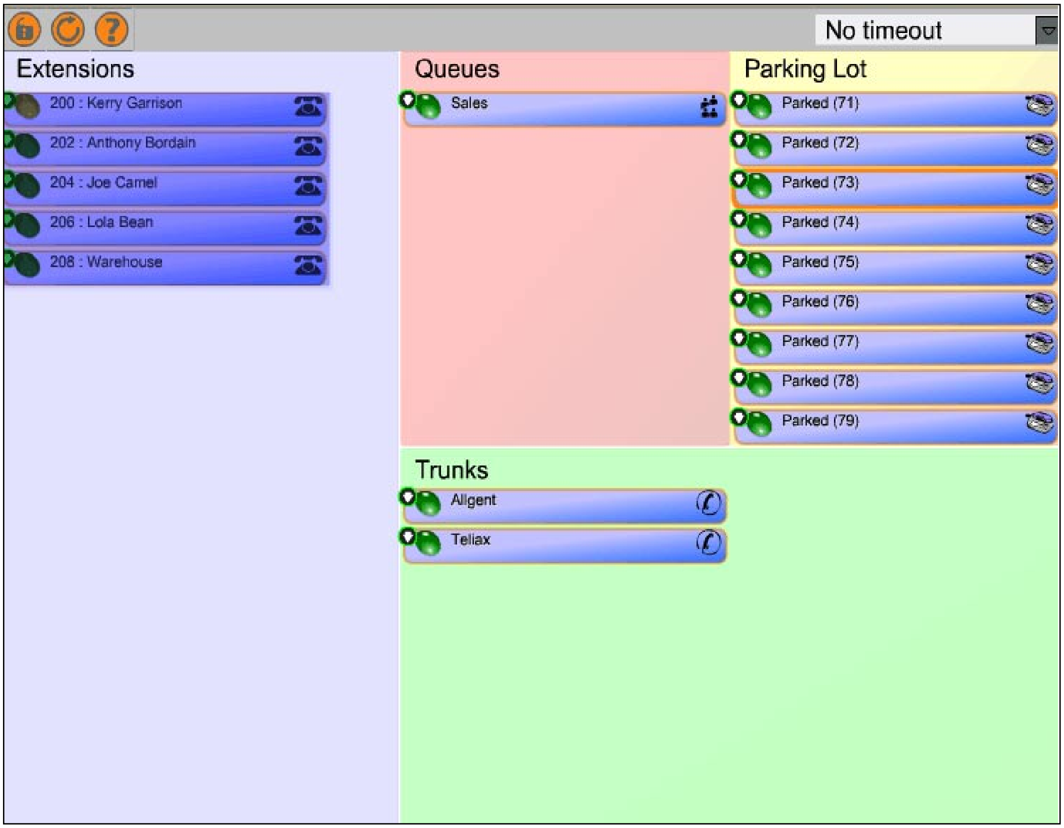


FOP (Flash Operator Panel)

The Flash Operator Panel is a Flash-based display that shows a real-time display of your extension, trunks, queues, and parking lots. This not only shows you the status but can allow you to transfer calls between extensions by dragging-and-dropping icons.

The main drawback to the Flash Operator Panel is that it can't support large numbers of extensions; it was really designed with a small business in mind.

Since you can see the status of all of the extensions and do drag-and-drop from extensions to parking lots, to other extensions, and so on, the Flash Operator Panel can work really well for a receptionist to manage calls coming into a business.

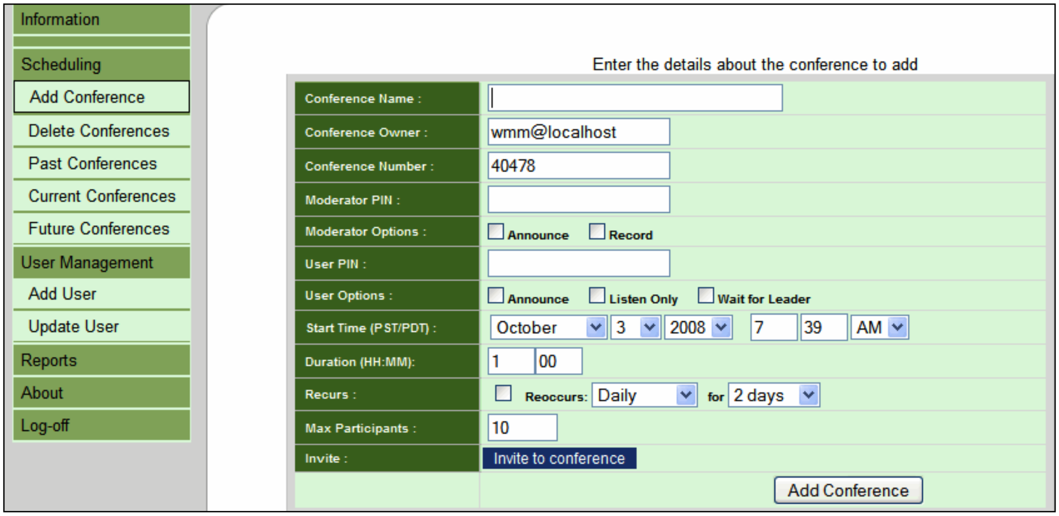


MeetMe

Under the **MeetMe** link you will find **Web MeetMe 3**, which is a powerful conference room management system. Within trixbox CE we will see how we can set up pre-defined conference rooms but with Web MeetMe 3 you now have access to a powerful conference room management system.

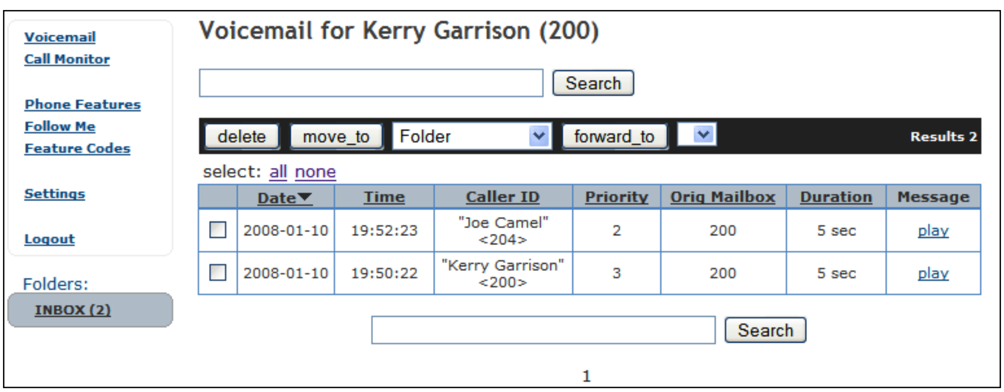
Web MeetMe 3 allows you to create users with different levels of access and then create scheduled conference rooms that are only available during the specified time. This prevents abuse of pre-defined conference rooms. You can also view reports that show past and present conference rooms.

If a conference is currently active you can see a list of participants and have the ability to kick out or mute individual users.



ARI

ARI stands for **Asterisk Recording Interface**, which is a web portal for users to access their stored voicemails and set basic options for their voicemail account. Later in the book, we will look at the VmX Location functionality of ARI that users can use to manage their own inbound call flow.

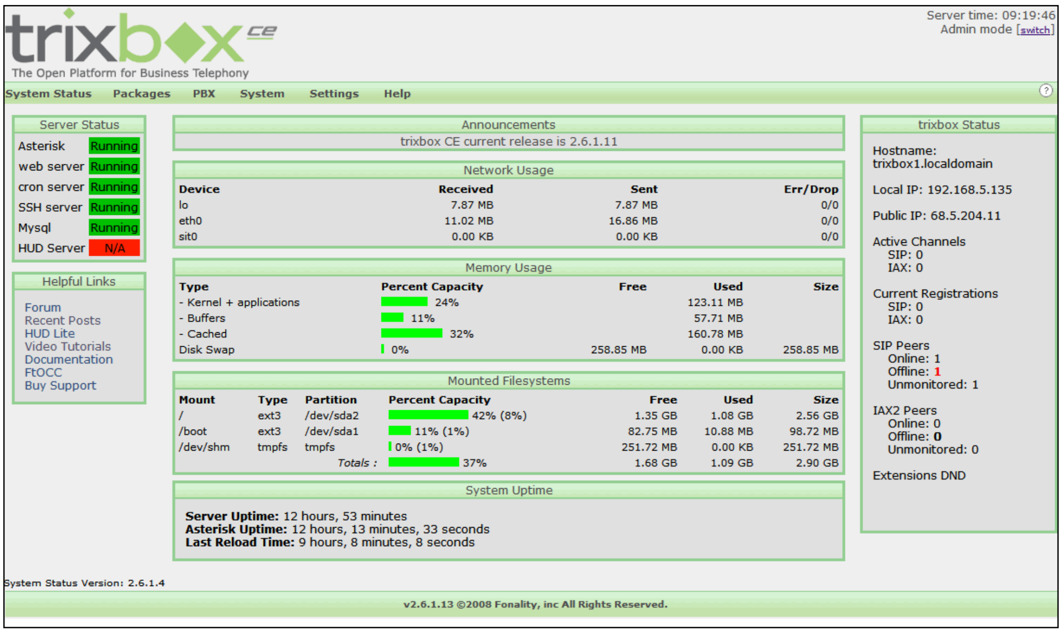


Admin mode

In the upper right-hand corner of the window is a link to **switch** to the Admin mode. Once you log into the admin section, you will see a handful of menus as well as the main system status display that gives you a quick snapshot of the health of your system.

System Status

The primary screen for the Admin section is called the **System Status** screen. This screen shows the status of the most critical services, graphs for available disk space, memory usages, disk utilization, and reports on uptime and extension status information. This snapshot can quickly help you determine if there is something wrong with your system.



Package Manager

The Package Manager is a web interface that is used to upgrade and install different components of the trixbox CE system. Under the hood the Package Manager is a web interface to a yum repository. While advanced users can use yum commands from the command line, less experienced users can simply use the Package Manager to keep their systems up to date. We will cover this in great detail in the chapter on maintaining your trixbox CE system (*Maintenance and reporting*).

#	Package Source	Package Name	Package Info	Version	Installed	Upgrade	Delete
1	trixbox	GConf2	A process-transparent configuration system	2.14.0-9.el5	installed	No Update	
2	trixbox	MAKEDEV	A program used for creating device files in /dev.	3.23-1.2	installed	No Update	
3	trixbox	ORBit2	A high-performance CORBA Object Request Broker	2.14.3-4.el5	installed	No Update	
4	trixbox	SysVinit	Programs which control basic system processes.	2.86-14	installed	No Update	
5	trixbox Update	aastra-xml-scripts	Aastra Telecom XML scripts	2.0.0-2	installed		Upgrade
6	trixbox	alsa-lib	The Advanced Linux Sound Architecture (ALSA) library.	1.0.14-1.rc4.el5	installed	No Update	
7	trixbox	alsa-utils	Advanced Linux Sound Architecture (ALSA) utilities	1.0.14-3.rc4.el5	installed	No Update	
8	trixbox	anacron	A cron-like program that can run jobs lost during downtime.	2.3-45.el5.centos	installed	No Update	
9	trixbox	app_flite	Flite text-to-speech module for Asterisk	0.5-1	installed	No Update	
10	trixbox	apr	Apache Portable Runtime library	1.2.7-11	installed	No Update	
11	trixbox	apr-util	Apache Portable Runtime Utility library	1.2.7-7.el5	installed	No Update	
12	trixbox	aspell	A spelling checker.	0.60.3-7.1	installed	No Update	
13	trixbox	aspell-en	English dictionaries for Aspell.	6.0-2.1	installed	No Update	
14	trixbox	asterisk	Asterisk PBX with Sample Configs	1.4.21.2-2	installed	No Update	
15	trixbox	asterisk-addons	Addons for Asterisk	1.4.7-2	installed	No Update	
16	trixbox	asterisk-perl	C implementation of the Primary Rate ISDN specification	0.09-1	installed	No Update	
17	trixbox	asterisk-sounds	Sounds addon package for the Asterisk PBX	1.2.1-2	installed	No Update	
18	trixbox	atftp	Advanced Trivial File Transfer Protocol (TFTP) client	0.7-6	installed	No Update	
19	trixbox	atftp-server	Advanced Trivial File Transfer Protocol (TFTP) server	0.7-6	installed	No Update	
20	trixbox	atk	Interfaces for accessibility support	1.12.2-1.fc6	installed	No Update	
21	trixbox	atl1	atl1 kernel modile	1.2.40.2-1	installed	No Update	
22	trixbox	audiofile	A library for accessing various audio file formats.	0.2.6-5	installed	No Update	
23	trixbox	audiofile-devel	Development files for Audio File applications.	0.2.6-5	installed	No Update	

PBX menu

Under the **PBX** menu are modules that help to manage the PBX-related components of the system. These tools include:

- **PBX Settings**: This is the main tool used to configure PBX features
- **Gizmo5**: This is a tool for easily purchasing and setting up SIP trunks from Gizmo5
- **Config File Editor**: This is a web interface that is used to edit configuration files manually
- **PBX Status**: This tool provides detailed information about the status of your trixbox CE installation
- **Endpoint Manager**: The Endpoint Manager is used to create configuration files for provisioning supported SIP phones
- **Bulk Extensions**: This tool allows you to create a large number of extensions by uploading a delimited text file
- **CDR Report**: This module provides enhanced call data reports to show you reports of calls that have come in to and out of the system

System menu

The **System** menu provides access to modules that help manage the overall system. These tools include the following:

- **System Info**: This module provides advanced system information on the hardware and operating system of your trixbox CE installation
- **System Maint**: The System Maintenance module allows you to restart Asterisk based on different parameters or to do a complete reboot of the system
- **Network**: The network module allows you to manage the TCP/IP settings of your system such as setting IP addresses, DHCP configuration, DNS settings, and Gateway settings

Settings menu

The **Settings** menu contains modules used to manage different settings within the system. The modules that you will find here are the following:

- **Repositories**: This module allows you to select which repositories are used to populate the Package Manager lists. By changing settings here you can choose to install modules that are currently in beta test.
- **Registration**: If you plan on purchasing support from Fonality you will need to register your system with the registration module. If you do not plan on getting support from Fonality then registration is optional.
- **General Settings**: This contains settings for use when having to configure an external mail server for relay purposes.

The history of trixbox

In late 2004 Andrew Gillis had been experimenting with Asterisk and had become frustrated with how tedious it was to install Linux, MySQL, Asterisk, AMP, and all of the other associated tools to get a system really set up well. Since he figured other people who were playing around with Asterisk might want his easy-to-install system, he created an open source project and called it Asterisk@Home. The first public version was unleashed on the world in November of 2004. The following is a brief history of the evolution of trixbox.

- November 2004
  Asterisk@Home 0.2
  First public release, RedHat Linux, Asterisk 1.0, a web interface, SQL call logging, and a web stats package.

- January 2005
  Asterisk@Home 0.3
  Added AMP, Flash Operator Panel, Cusco XML Services, Music On Hold, FAX Support, and xPL support.
- April 2005
  Asterisk@Home 1.0
  Just bug fixes to take it to the first production release.
- April 2006
  Asterisk@Home 2.8 (arguably one of the most popular versions ever)
  Better install of SugarCRM with Asterisk Plug-In, FreePBX replaced AMP, added phpMyAdmin, asks for root password on install.
- June 2006
  trixbox 1.0
  Name change, new user interface, update script added.
- January 2007
  trixbox 2.0
  Introduction of the trixbox dashboard, lots of new tools, and the Package Manager to manage the system components.
- March 2007
  trixbox CE 2.2
  Improved interface, System Status screen, enhanced call reports, network settings module.
- September 2007
  trixbox Pro product line launches.
- December 2007
  trixbox CE 2.4
  Completely new architecture, based on CentOS 5.1, Asterisk 1.4, FreePBX 2.3.
- February 2008
  trixbox CE 2.6
  Better integration with different tools to provide a more cohesive interface.

Is trixbox CE for me?

If you are looking at installing a phone system to handle anything from 2 – 200 workstations, then you should take a serious look at trixbox CE. There isn't a 200 station limitation but with larger installations I do recommend hiring an experienced integrator to make sure everything goes smoothly.

In the early days, the number of phones that were available was quite limited and many suffered from major problems that turned a lot of people off from using them. Today, you have high quality phones from name brand companies such as Cisco, Linksys, Polycom, Aastra, and SNOM, as well as entry-level phones from companies like Grandstream. PCI cards are now available with optional hardware echo cancellation.

trixbox CE, being an IP PBX system, has features that simply aren't available on traditional PBX systems, such as being able to easily tie multiple machines together, having telecommuter users, desktop applications for call management and presence, or even easy, web-based configuration. If these are just some of the features you are looking for, then trixbox is probably going to be an excellent fit for you.

The future of trixbox CE

It's always a little hard to predict where trixbox is going in the future. The project goals are to make using the system even easier, provide a more professional interface, and add features to the dashboard to minimize ever having to use the command line. There are people who want to see trixbox CE include functions like mail servers, file servers, routers, firewalls, instant message servers, and numerous other things. While adding functionality is a good thing, having a single 'God Box' to run your entire business is certainly putting all of your eggs in one basket. Since trixbox CE is a community-driven project, what trixbox becomes in the future will largely be determined by the requests of members of the community and what they feel are the features that they need to be successful in implementing trixbox CE-based solutions.

Hardware we will need to have

The actual hardware requirements for trixbox are actually very minimal. Many people have run their office systems on basic 1 GHz P4 systems. For testing purposes you can usually use just about anything better than a PIII 500 MHz system with 384MB RAM. This would be OK for a couple of extensions. For a production machine in a business environment the recommended system is a P4 2.0 GHz machine or better, 512MB RAM or more, and at least an 80GB HD.

One key point when selecting a processor is to not use a Hyperthreading CPU. While you can use multiple processors or dual-core processors, Hyperthreading processors do not work well with Asterisk.

Even if you don't plan on using any traditional phone circuits, it is recommended to always have an analog line card installed. While you may never use the line card, it will provide a very reliable timing source for Asterisk to use; this will prevent certain issues with sound quality on audio playback and music on hold. It is highly recommended to actually always have an analog line connected to the system so that you can always have 911 service as a backup to your primary circuit.

Choosing a motherboard is one of the more difficult issues these days. The tendency is to buy cheap desktop system motherboards; the main problem is that the IRQ handling on the low end motherboards isn't as good as on server motherboards. This poorer IRQ handling can cause interrupt conflicts across the PCI bus. IRQ conflicts can cause issues such as choppy sound or even dropped calls. Digium's website has a list of currently recommended motherboards, and resources like the trixbox wiki and forums are great places to look for hardware compatibility information.

Timing problems can really cause problems when the system has to transcode (convert) from one codec (audio format) to another. For example, if our PBX is connecting to a provider using G711 and our phones are connecting to the server using G729, and if we have a timing problem then Asterisk can lose synchronization between the two codecs resulting in sound quality issues. If you have a TDM card installed into your system then it will provide the proper timing the system needs in order to function properly.

The best way to think of this timing signal is to think of it as a system-generated heartbeat that allows different processes to stay in sync. While software-only solutions are coming out based on the latest CentOS kernel, a good hardware solution is always going to be preferable when possible.

Connecting a phone

The first thing we typically want to do is to connect a phone to the system so that we can make sure the system is up and running and that we can make calls from extension-to-extension. The three primary types of phones we can use are going to be VoIP phones using the SIP protocol, an ATA connected to an analog phone, or a softphone which is a software phone that we run on our computer and then use a microphone and speakers to talk with.

Hard phones

For the best results and features, a SIP-based hard phone is recommended. Most of these have multiple lines, LCD displays, speakerphones, programmable keys, and other standard business functions. While these may be overkill in a house, it's almost a necessity to use these business-grade phones when installing a production system into a business.

SIP-based hard phones are available from companies like Aastra, Linksys, Cisco, Polycom, Snom, and others.

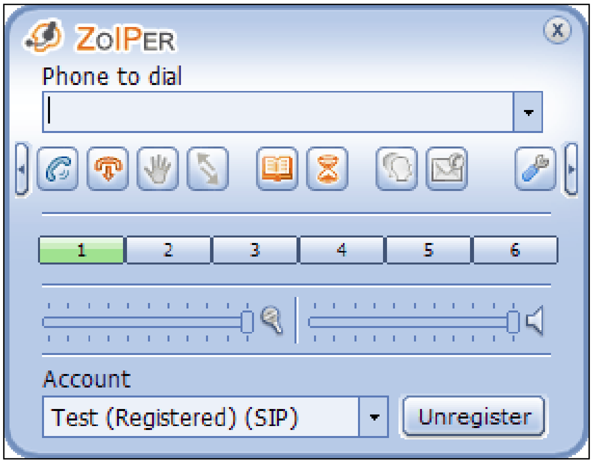
ATAs

An ATA is an analog telephone adapter. These small devices are connected to the PBX as a SIP device just like a hard phone but have can have one or more FXS ports on them so that you can plug in a regular analog phone. The disadvantage of using an ATA is that you do not have programmable buttons for functions such as accessing voicemail, call transfers, hold, making conference calls, and other similar features, while virtually all of these functions can still be used—if you can remember all of the function codes. The primary advantage is that they cost less than buying a SIP phone and allow you to use an existing analog phone. Good ATAs are available from Linksys and Grandstream.

Softphones

A softphone is very convenient for people who travel with a laptop a lot and they are great for testing systems. You can install a softphone on your computer and use an analog mic, Bluetooth headset, or USB device for the microphone and/or speakers. One thing that is really nice about softphones is that some of the best ones available are actually free. Counterpath's X-Lite is a really good choice for SIP connections and Zoiper (`http://zoiper.com`) is excellent for both SIP and IAX connections.

A large list of available softphones can be found at `http://www.voip-info.org/wiki/view/VOIP+Phones`.



Connecting to an analog phone line

Usually the first thing you want to do is to connect your system to a regular analog phone line. In order to connect a phone line, we need a device that will provide us with an FXO port. This can be accomplished either with PCI add-in cards or with an external gateway device.

FXO versus FXS ports

Before we look at what kind of port we need, let's answer the question of what the difference between an FXO port and an FXS port is. If you can remember what the acronyms for FXO and FXS stand for then it will be much easier to remember what their purpose is. The main thing to keep in mind is that the port is labeled based on what plugs into it; many people end up getting this backwards.

FXO (Foreign Exchange Office)

Since we said that the port is labeled based on what the port connects to, then an FXO port connects to the remote office (basically the telephone company). So if we have a standard analog phone line from our local phone company, we would plug it into an FXO port.

FXS (Foreign Exchange Station)

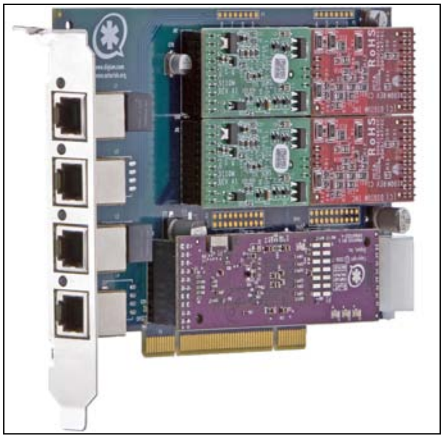
While Office may be a little hard to remember, 'Station' is much easier. This port makes more sense, since an analog phone is easily considered to be a station so it shouldn't be hard to remember that if we want to use an analog phone, we need to plug it into a station port, or FXS port.

Add-In cards

The most common way to provide an FXO port is with an internal PCI interface card. These can range in price from around $30 to more than $2000 depending on brand, features, and number of ports. The lowest cost cards are the single FXO port X100P cards from `http://X100P.com`.

> Beware of the cheap X100P clone cards that you can get from on-line auction sites; these are usually sold for $12 or less and are simply cheap Intel WinModem cards. These cheap imitations will cause you all kinds of problems from IRQ issues, to major echo problems, to simply not working.

For a 1-4 line installation you can get good quality cards from companies such as Digium, Sangoma, Rhino, and OpenVox. For an actual production system it is highly recommended that you spend the extra money to get a card that has on-board hardware echo cancellation.



SIP gateways

Besides using an internal card, an external SIP gateway can be a good solution as well. A SIP gateway will provide an FXO port to connect to your POTS line and then connects to your local area network. Calls on the POTS line are converted to digital packets and sent as VoIP to the PBX system. Some of the advantages to using external SIP gateways include:

- No issues with IRQ conflicts
- Phone line interface does not have to be physically close to the PBX system
    - The gateways can be placed in the phone room and connect to the PBX in a server rack somewhere else.
    - The PBX doesn't even have to be in the same building; once the call is converted to VoIP, the PBX can be anywhere on the Internet.

Some common SIP gateways include the Linksys SPA-3102, which not only has an FXO port to connect your POTS line to, but also has an FXS port so that you can connect an analog phone to it and use it as an extension to your PBX. Other SIP gateway products that work well with trixbox include the four-port Linksys SPA400 and the four-port Grandstream GXW4104.

VoIP providers

Certainly a more and more common way to get phone service into your phone system is with the use of a VoIP service provider, also known as an Internet Telephone Service Provider (ITSP). Using SIP or IAX protocols an ITSP can provide you with a **DID** (**Direct Inward Dial**) phone number and allow you to place calls to regular phone numbers through its service.

While using an ITSP is often less expensive than using regular phone circuits, the downside is that you are reliant on your Internet access for your phone service and even the best Internet connections are less reliable than regular phone lines. While in many places Internet reliability has improved dramatically over the years, you can't control the quality of the connection when your calls are going out over the Internet. Activities like file transfers, watching videos online, or even sending large email attachments can affect call quality (more on this later).

Even if you aren't using an ITSP for your primary business circuit, it can make sense to have an ITSP for certain calls such as long distance or overseas calls as well as to handle peak time call overflow.

ITSPs are very simple to set up but you may need to do some port forwarding and adjust some network settings to get incoming calls to work and for audio to work correctly.

Where to get more information

The following are the companies and links to their websites to get more information.

- **Aastra**
  SIP Phones
  `http://www.aastratelecom.com`
- **Linksys**
  SIP Phones and ATAs
  `http://www.linksys.com`
- **Grandstream**
  SIP Phones and ATAs
  `http://www.grandstream.com`
- **Counterpath**
  Softphone
  `http://www.counterpath.com`
- **Zoiper**
  Softphone
  `http://www.zoiper.com`
- **Sangoma**
  Telephony Interface Cards
  `http://www.sangoma.com`
- **VoIP Info**
  General VoIP Information Wiki
  `http://www.voip-info.org`

Where to buy

The hardware devices listed above aren't available at your local electronics store (yet) but are available from reputable online resellers.

- **Voiplink**
  SIP Hardware and trixbox servers
  `http://www.voiplink.com`

Summary

In this chapter, you have learned about the different modules that make up the trixbox CE platform and how trixbox CE is different than installing the different components by yourself. We have also taken a look at the history of trixbox since its beginnings from when Andrew Gillis was hacking the ISO installer together in his basement to how Fonality's support is helping to keep development of the project going. We have also looked at the basic requirements for setting up a trixbox CE system. We looked at popular models of SIP phones and ATAs and some common softphones. We also learned the difference between FXO and FXS ports and what devices we need to connect phone lines and analog phones. We also learned about VoIP providers and the role they play.

3

Installing trixbox

By now you are probably chomping at the bit to get to actually installing your trixbox CE system. You should have all your hardware lined up, a server ready to go, and now all we need to do is install the software.

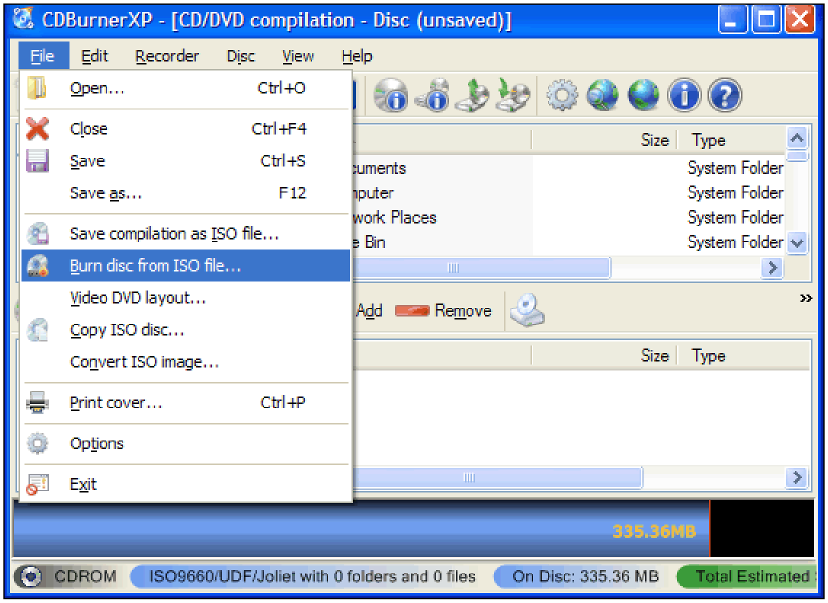
Download the ISO image

The trixbox CE software comes as an ISO image, which is file that is a complete image of a CD-ROM. You can download the current software by going to `http://www.trixbox.org/downloads`. The ISO image file is going to be about 670 megabytes.

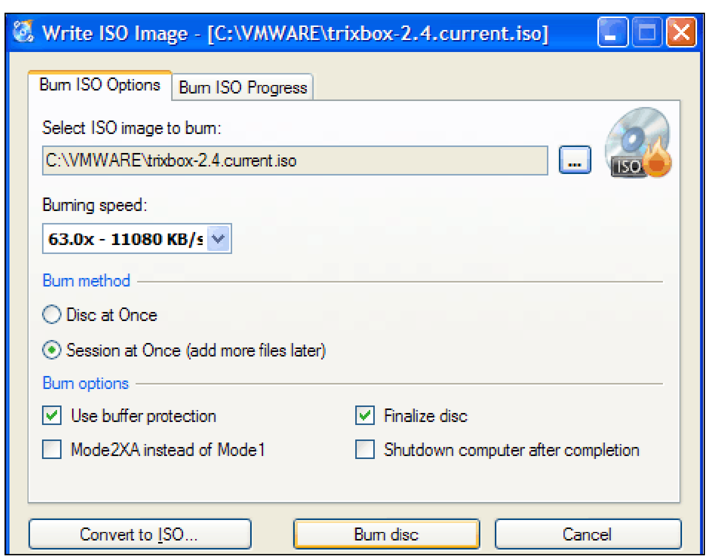
Burning the CD

Most CD-ROM burning software can burn the ISO image to a blank CD; the thing to make sure of is that you don't simply burn the ISO file onto the disc like any other file you would copy when doing a backup. If you don't burn the file onto the CD as an image, it won't work properly. If you don't have any CD-ROM burning software, we will walk through the process of burning the image using the free program called CDBurnerXP, which is available for download at `http://cdburnerxp.se`. When you first run CDBurnerXP, you will need to click on the **Create Data CD/DVD** button.

When the next screen comes up, go to the **File** menu and select **Burn disc from ISO file...**.



When the next screen comes up, click on the ... button to browse to where you downloaded the ISO file to, select the file, and click the **Open** button. If you have a blank CD in your CD-ROM burner, click on the **Burn disc** button to begin burning the image to the CD.



Once the process completes you will have a bootable CD that you can use to install trixbox CE onto your target machine.

Preparing for installation

You will need a computer onto which you are going to use to install trixbox CE. The installation process will format the hard drives and install a new operating system and all of the trixbox CE software.

> Installing trixbox CE will format ALL of the hard drives connected to the machine, and ALL of the data on ALL of the attached hard drives will be lost. Do not have any hard drive including external USB hard drives connected if you mind losing all of the data that is stored on those drives.

Depending on your system you may either be able to hit a specific key to boot off the CD or you may have to go into the system's BIOS menus to set the computer to boot off of the CD. Refer to the manual for your motherboard for instructions on how to boot from the CD-ROM drive.

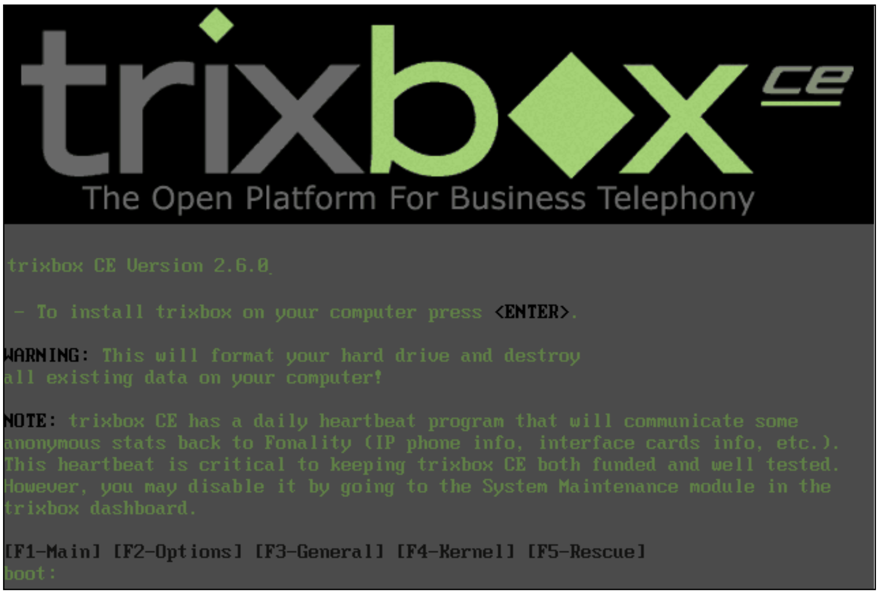
Installing trixbox CE

We are finally ready to go, we have the trixbox CE CD in the drive; now all you need to do is fire up the computer and boot off the CD. We will now walk through the installation screens and look at the different installation options. When the first screen comes up, you can usually just hit the *Enter* key to proceed with a normal installation. If you hit the F2 key, you will get a screen of advanced functions. Some of these advanced functions that you should know are the following:

- **default mediacheck**: Using this command will run a media check during the installation process that checks the CD media against a checksum on the CD to make sure your download and burn worked properly. If you have problems installing the trixbox CE software, you should try rebooting and using this command.
- **sataraid**: If you have two SATA drives connected to your system you can use this command to automatically configure software mirroring between the two drives to ensure that your system can survive a hard drive failure.
- **sataraidstore**: This command will install trixbox CE onto a 20GB partition and format the rest of the system as a single partition that is mounted as `/store`. This is used to set up a trixbox CE system as a phone system as well as a file server.

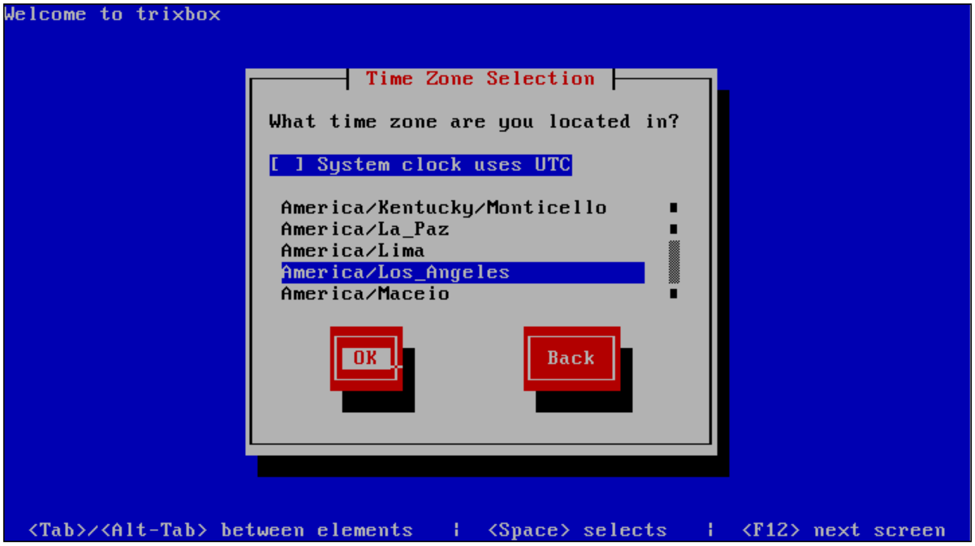
- **advanced**: If you are familiar with Linux and want to set up your own drive partitioning then you can use this command to have more control over your trixbox CE installation.

For now, let's just hit the *Enter* key to begin the installation process.

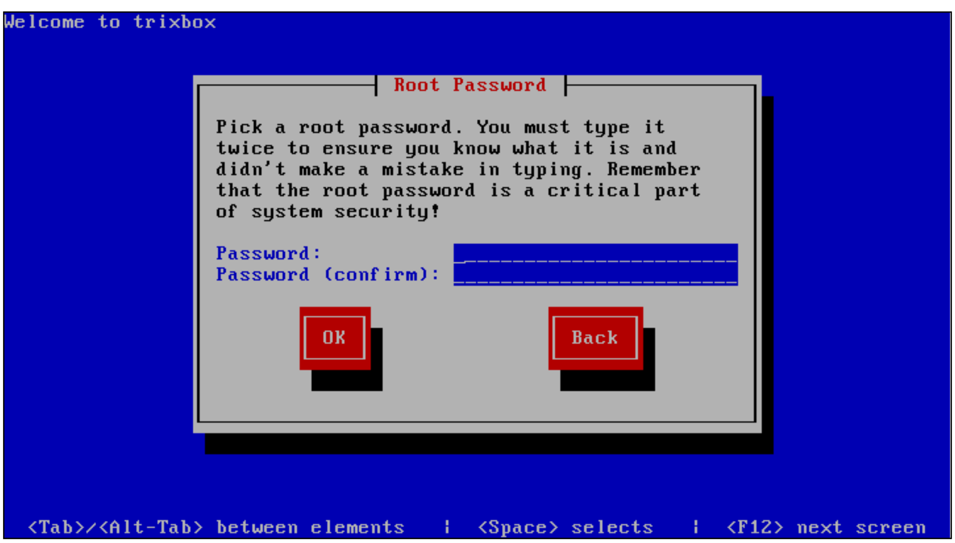


The second screen that comes up will ask us which default language for the operating system we want to use.

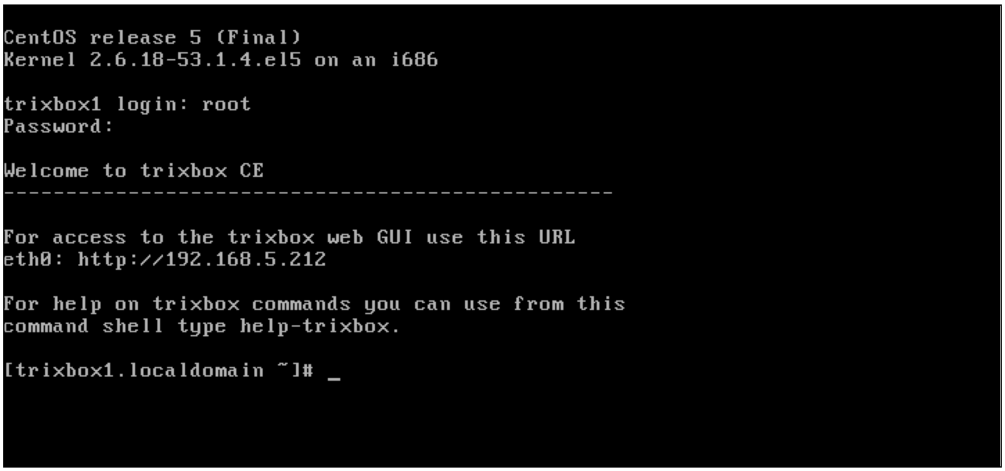
Moving right along, the next screen will prompt you to set the time zone for your system. Generally speaking, you can use the **UTC** option as this will account for daylight savings time. However, if you are using dual boot into another operating system like DOS or Windows, this can cause time issues for the other operating system.



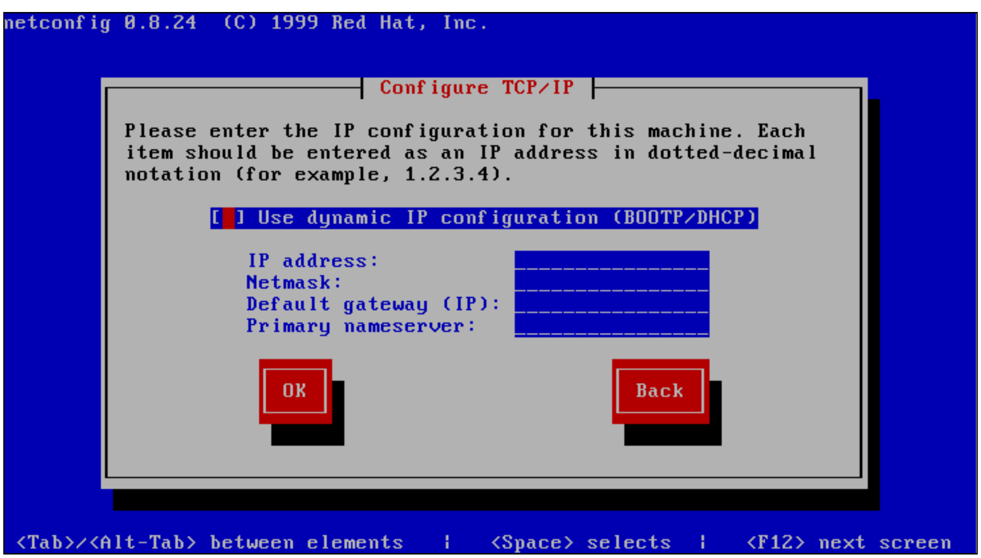
The final screen in the install process will prompt you to select a password. This password is used to log-in to the console or through SSH as root. Don't forget this password or you will have to wait till the end of the book to learn how to reset a lost root password.



After entering a password for the root account, the rest of the installation process will begin. Depending on the speed of your system, this installation process will take between 10 and 20 minutes. When the installation process is complete, the system will reboot and be waiting at the login prompt.



Once you log in, the system will show you the IP address that was assigned to your system from your DHCP server. If you need to change to a static IP address, you can browse to the URL shown on your screen and use the Network Configuration module to configure your network, or you can run the **system-config-network** command to set up your networking.



SSH Access

When the trixbox CE installation is complete, we can also log in to the system using SSH (a secure shell protocol). A popular open-source tool for accessing a system with SSH is PuTTy (`www.chiark.greenend.org.uk/~sgtatham/putty/`).

Basic system configuration

Now is a good time to look at a few tools we can use to do some basic system configuration. (PBX configuration will be covered in the next chapter.) For many installations you are actually ready to begin setting up the PBX functionality at this point. If your system is on a public IP address, is in a DMZ, or has any ports forwarded to it from the public Internet, you will want to keep your system up-to-date with the latest security updates. Regardless of what you may have heard, there are always vulnerabilities being found in different components and you need to make sure your system is secured against these issues.

Updating trixbox CE from the command line

As we dig into the trixbox CE dashboard we will see how we can use the Package Manager to update our system. Since we are currently sitting in front of the machine and logged into the console, this is a good time to learn about how to upgrade your system from the command line in case something ever goes wrong with the Package Manager and you need to do an update manually.

Getting all yummy with it

The maintainers of CentOS keep several repositories of packages to make it simple to install and upgrade packages. Fonality maintains repositories that are specific to the trixbox CE code base. Using the `yum` command will allow you to do quick updates to all aspects of your system with one command:

```
yum update
```

Running the command will give you some output such as the following:

```
[trixbox1.localdomain ~]# yum update
Dependencies Resolved

=========================================================================
====
 Package                 Arch       Version          Repository
Size
=========================================================================
====
Updating:
 php-pear                noarch     1:1.5.0-3        trixbox
406 k
 php-pear-DB             noarch     1.7.13-1.el5.centos  extras
142 k

Transaction Summary
=========================================================================
====
Install      0 Package(s)
Update       2 Package(s)
Remove       0 Package(s)

Total download size: 549 k
Is this ok [y/N]:
```

Select Y to begin installation. Usually you do not need to reboot your system after installing updates but since you can't always tell if you need to or not, it is usually a good idea to issue a `reboot` command to restart the system.

trixbox CE command-line tools

For a list of available trixbox CE command-line tools, use the `help-trixbox` command. This command will give you the following output:

```
[trixbox1.localdomain ~]# help-trixbox

trixbox - HELP

Commands                 Descriptions
---------------------------------------------------------------------
system-config-network    configure ethernet interface
passwd-maint             set master password for web GUI
```

```
passwd                  set root password for console login
setup-cisco             create a SIPDefault.cnf in /tftpboot
setup-aastra            create a aastra.cfg in /tftpboot
setup-grandstream       setup for autoconfiguration of Grandstream
setup-linksys           setup for configuration of Linksys phones
setup-polycom           setup for polycom phones
setup-snom              setup for snom phones
setup-dhcp              set up a dhcp server
setup-samba             set up a Samba server (Microsoft file sharing)
setup-mail              configure sendmail
setup-pstn              detect and setup supported PSTN interface cards
asterisk -r             Asterisk CLI
install-hudlite         Install hudlite server
install-postfix         Install postfix mail server
install-sendmail        Install sendmail mail server
```

In later chapters we will look at these commands in more detail as we learn about phone provisioning and securing our server.

Using the web interface

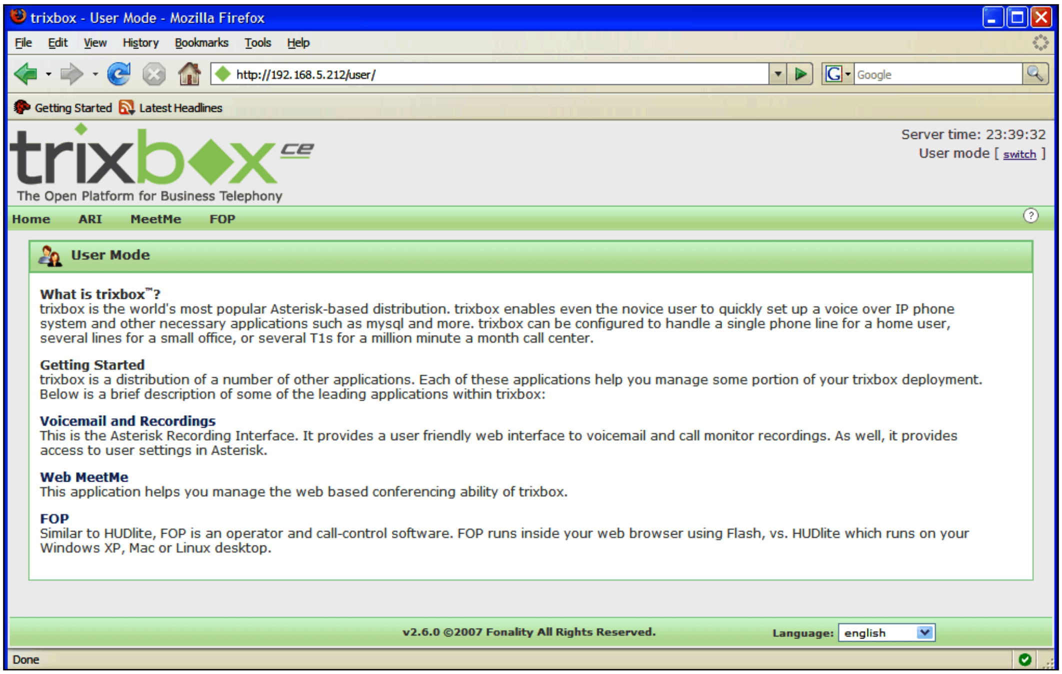
Our systems are up and running and now it's time to get into the web interface and start seeing exactly what our system is capable of doing. To get started we need to use a different machine than the one onto which we installed trixbox CE because it does not have a desktop system or web browser installed on it.

From the trixbox CE system you installed, once you log in as 'root' with the password you created during the installation, the system will display the IP address to use to access the web interface.

```
Welcome to trixbox CE
-------------------------------------------------

For access to the trixbox web GUI use this URL
eth0 http://192.168.5.250

For help on trixbox commands you can use from this
command shell type help-trixbox.

[trixbox1.localdomain ~]#
```

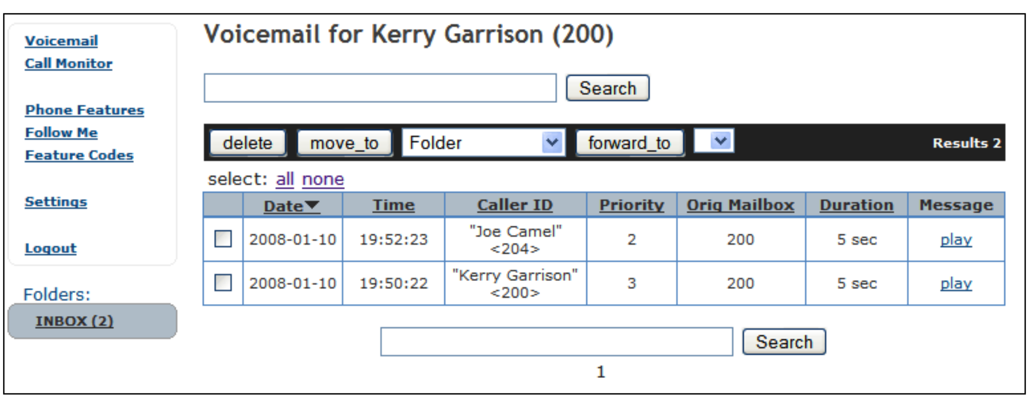
User mode

The first screen you will get is the **User Mode** screen. Since the tools here are designed for regular users there is no password required to access this system although some passwords are required in some of the individual tools themselves.

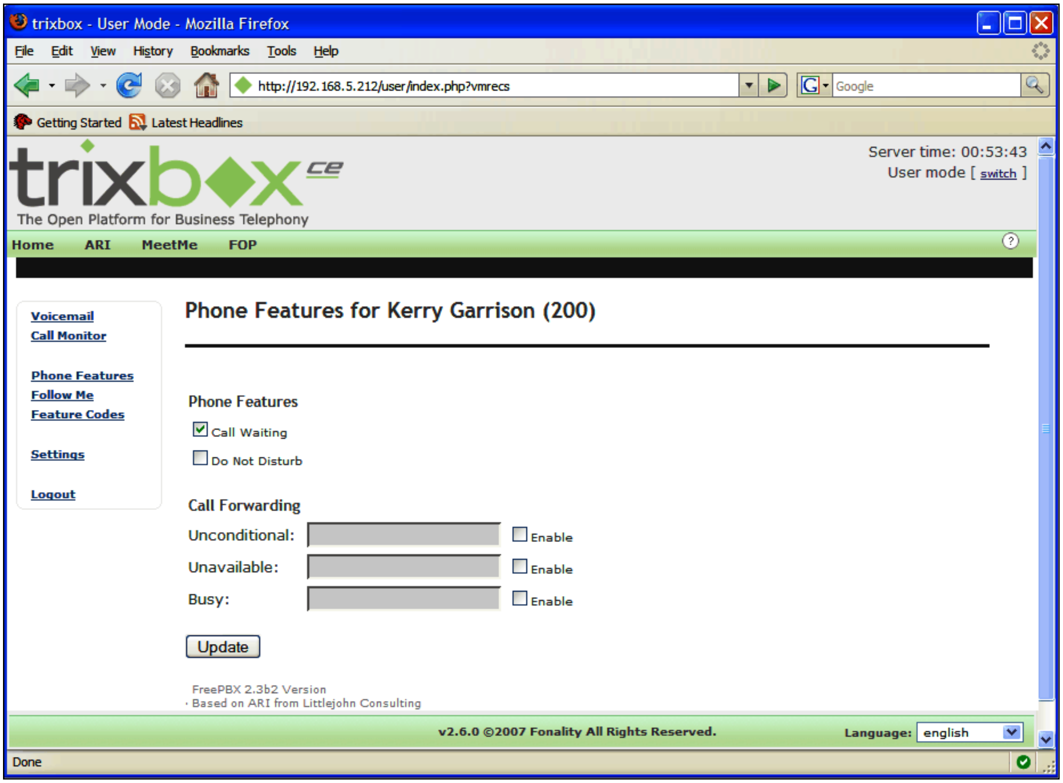


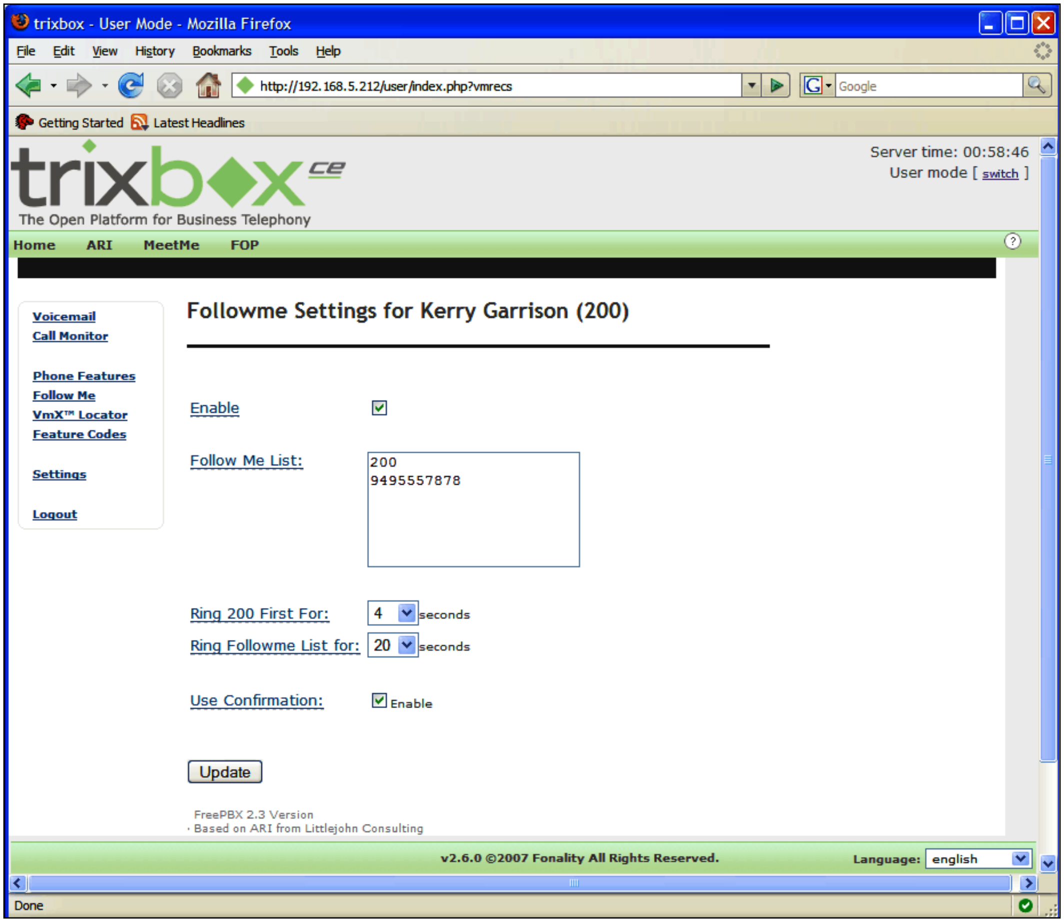
Asterisk Recording Interface (ARI)

The ARI module is a web interface used by users for web access to voicemail and recordings. A user logs into ARI with their extension and voicemail password.

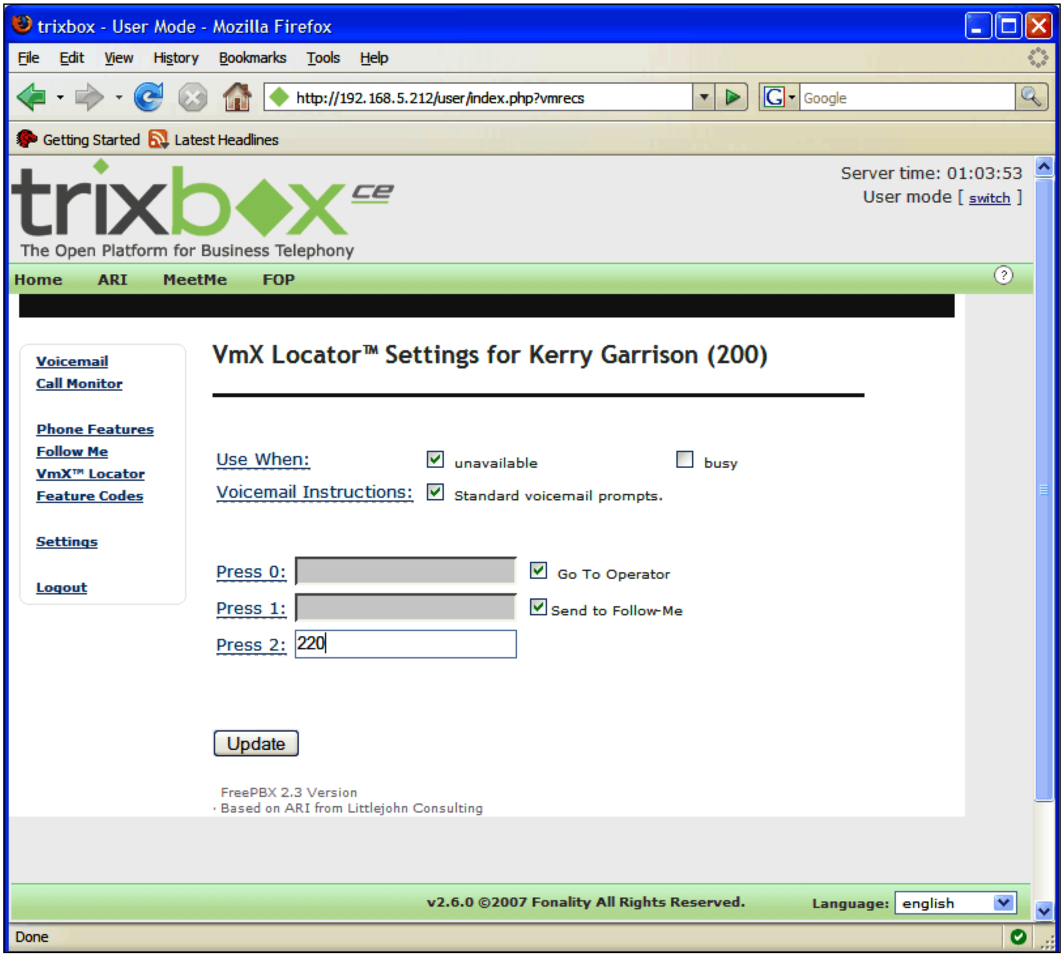


The most common function of ARI is simply to allow users to check their voicemails and recordings. Users who are used to older versions will appreciate some of the newer features that are now available, such as being able to modify their phone settings and adjust certain **Follow Me** functions if the administrator has set it up for them.





Using trixbox CE **2.4** and higher, a new feature has been added called the VmX Locator. If the VmX Locator is enabled by the administrator for the user, then a new menu for this will appear within the ARI interface.



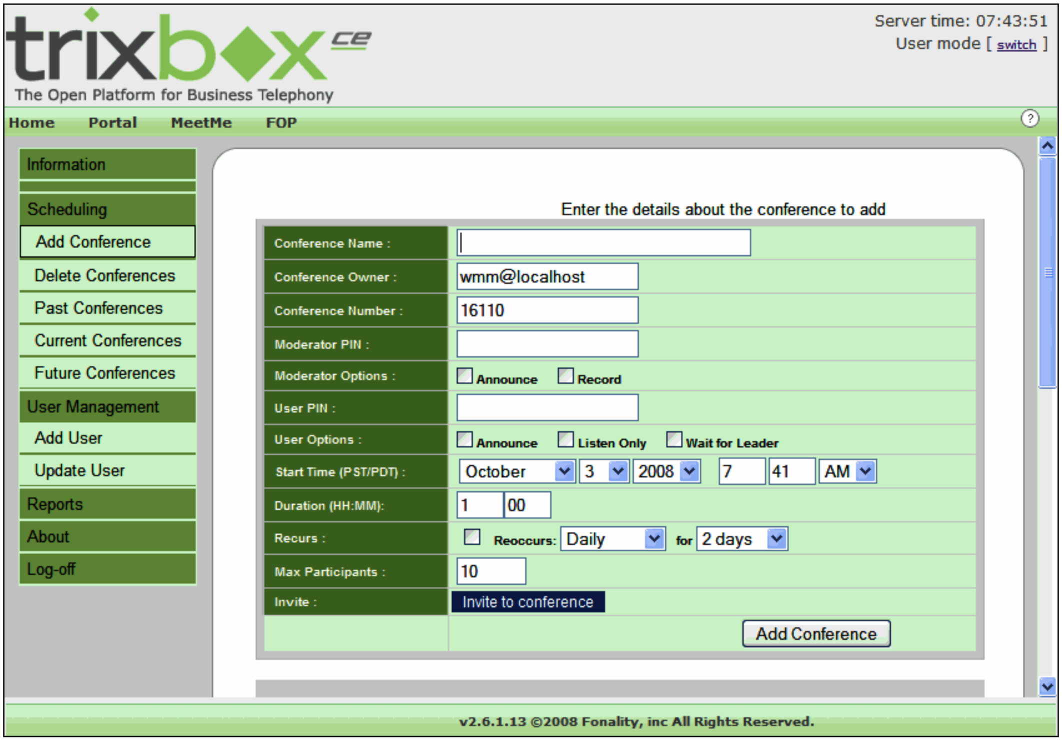
Web MeetMe (MeetMe)

As we saw earlier, Web MeetMe is a powerful conference-room management system. With the ability to schedule conferences and manage conferences that are in session, the new Web MeetMe takes conference-room management to a new level.

Unfortunately the user database does not tie into the other databases in the system, so if you want to create users to have access to this control, you have to create them within Web MeetMe.

To actually use the conference rooms requires some setup within the PBX Configuration tool that we will set up when we talk more about conference rooms. To use Web MeetMe you will need to log on with one of the two pre-configured accounts:

- Admin Access
  Username: wmm@localhost
  Password: wmmpw
- Standard user
  Username: tim@localhost
  Password: 1234



During a conference call you can see a list of participants and selectively kick them from the conference or mute them if needed.

Flash Operator Panel (FOP)

The Flash Operator Panel gets its name because it was written in Flash. FOP is designed as a receptionist tool to help manage calls within the company. What works really well in a receptionist setup is to have FOP running full screen on a second monitor so the receptionist always has instant access to the screen.

The following list describes the different status states that buttons can be in:

- **Red**: Channel is currently busy (has a call on it)
- **Green**: Channel is currently available (no call on it)
- **Flashing**: Channel has an incoming call ringing on it
- **All Ovals Flashing**: FOP has lost connection to the server

If there is current activity on a channel, you can hover the mouse cursor over that button to get additional details for that call.

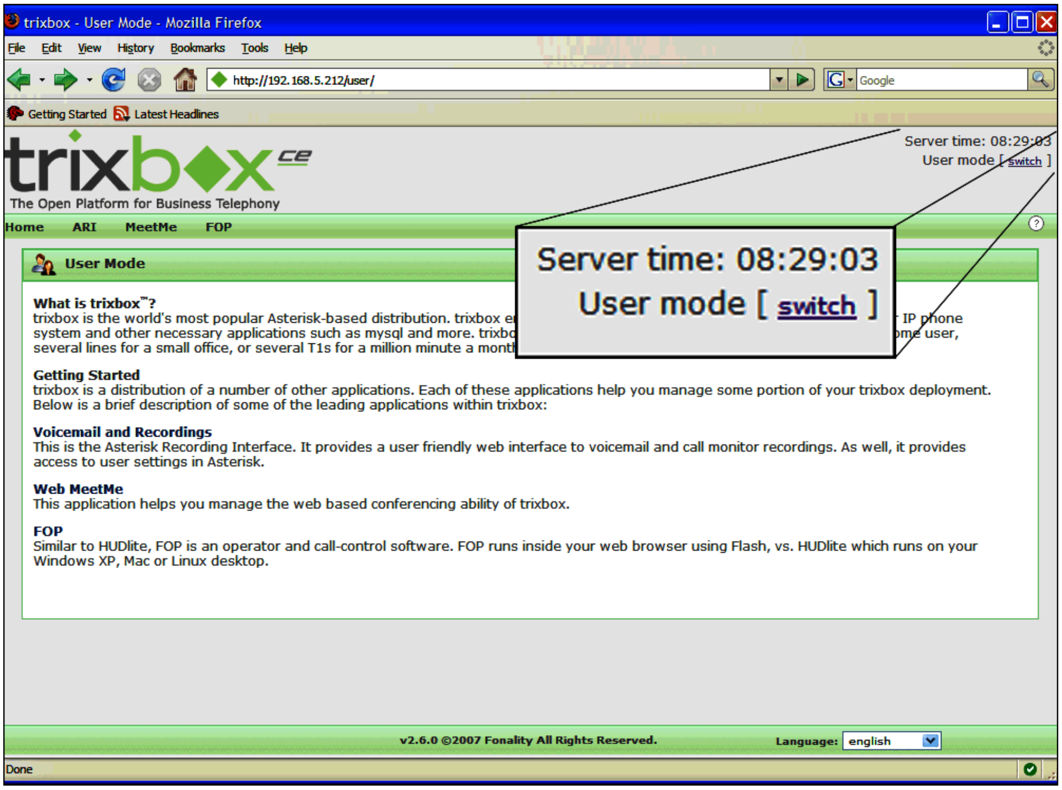
If you try to perform any actions within FOP, a dialog box will appear asking for a password. The default password for FOP in trixbox CE is passw0rd (that's a zero, not an 'o'). Once you have entered the password you can perform the following functions:

- **Hangup a channel**—double-click the colored dot on the button
- **Transfer a call**—drag the phone icon on a talking button to another button
- **Originate calls**—drag the phone icon from one available button to another available button
- **Barge in on a call**—drag the phone icon from one available button to a bridged/busy one

Admin mode

To switch to **Admin Mode** click on the **switch** link in the upper right-hand corner of the screen. Clicking on the link will bring up a dialog box asking for a login and password. Use the following credentials to log into the Admin section:

- **Login**: **maint**
- **Password**: **password**



System Status

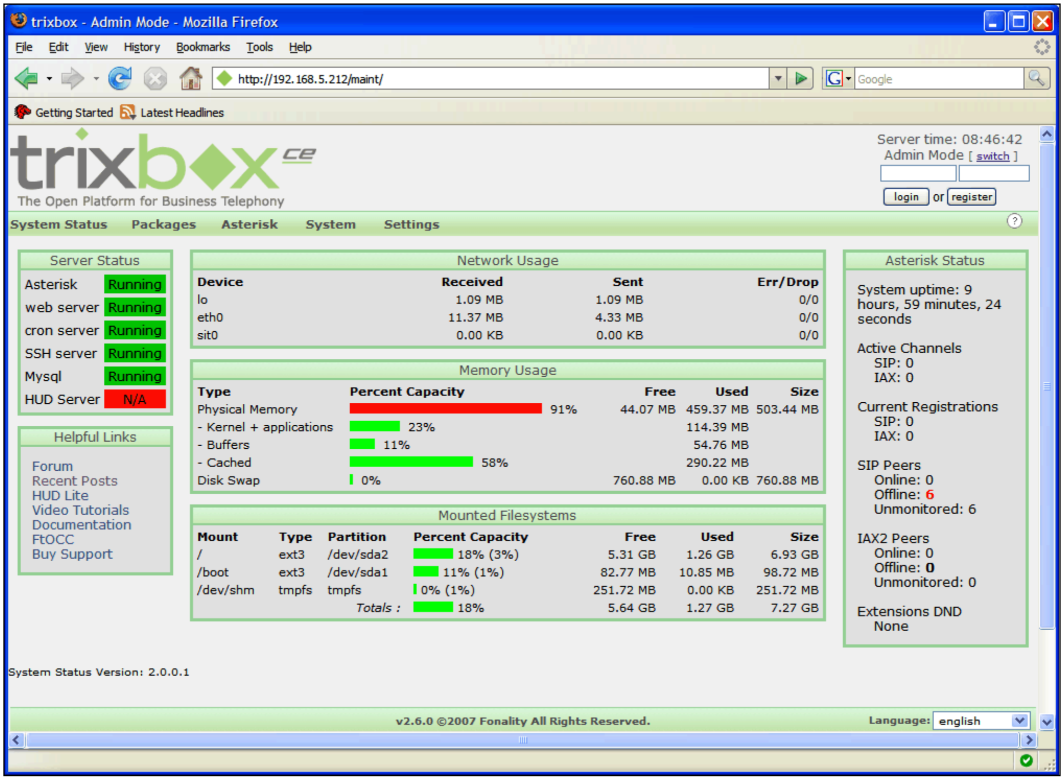
The first screen you get is the System Status display; this screen will give you a snapshot of the current health of your system. On the left side of the screen is the status of common server processes that should be running on your system. Unless you have installed the HUDLite Server, it will show up as **N/A** (Not Available). Down the middle of the screen you will see the **Network Usage** box; if you start seeing errors or dropped packets then you may be having a network problem that needs to be looked at.

The only time to really be concerned here is if the **Disk Swap** graph starts to show usage—this means you have run out of physical RAM and the system is swapping out to disk. This disk swapping can affect call quality.

At the bottom of the status stack is the **Mounted Filesystems** box. This graph will monitor the available and used disk space of each partition in your system.

In the right-hand column is information about the status of Asterisk and associated connections; the following list will help explain what each of the monitors is really showing you:

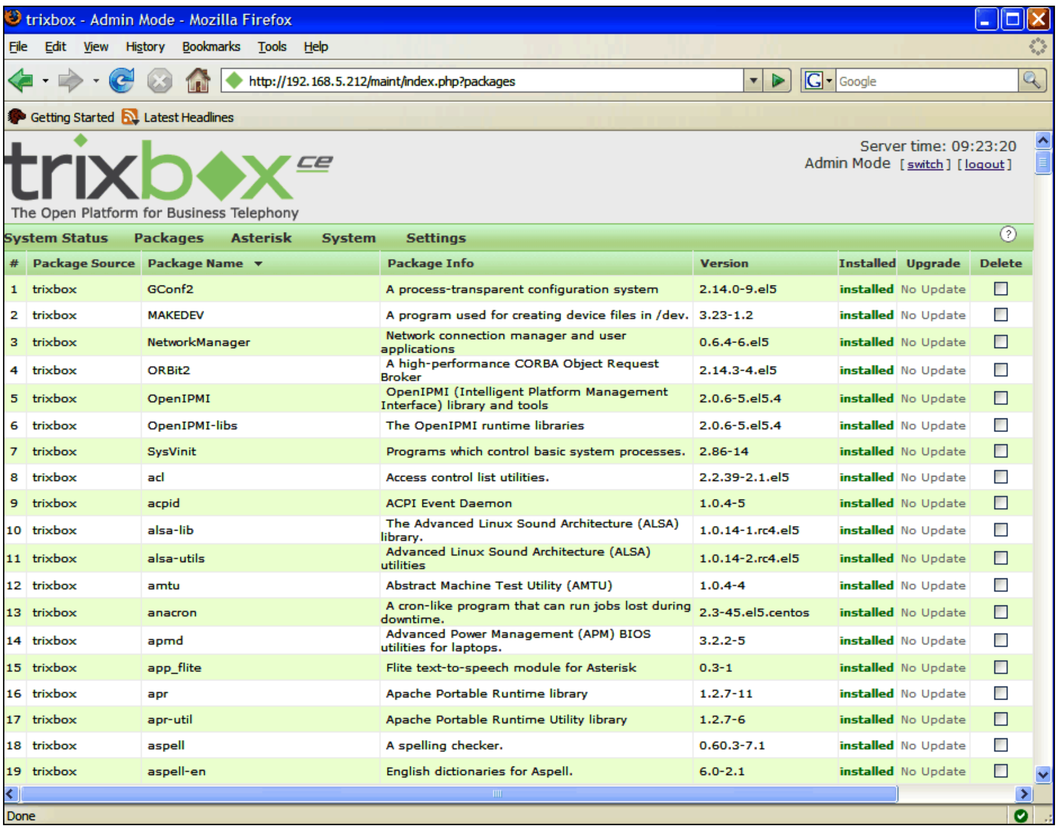
- **System Uptime**: This will tell you how long the system has been running since the last time it was restarted. This is helpful to see if the system has been restarted without you knowing about it.
- **Active Channels**: The **Active Channels** monitor tells you how many calls are currently in progress on both IAX and SIP trunks.
- **Current Registrations**: When you are using an ITSP for your phone service, the system will need to register with the provider. This monitor will show you if you have current registrations that are active.
- **SIP Peers**: When using SIP phones and ATAs, these devices act as Peers to the phone system. The Peers monitor will show you the status of SIP devices that are configured for the system.
- **IAX2 Peers**: Just like **SIP Peers**, IAX Peers shows the status of IAX devices that are configured for the system.



Package Manager

The Package Manager is a web interface to the repositories that contain system updates and additional tools and utilities. At a glance you can see what packages are installed, which ones have updates, and which are available to install.

Just because there are new packages all the time does not mean you should always upgrade to the most recent version. For example, upgrading to a newer version of Zaptel may break certain PCI interface cards if their manufacturers have not released updated drivers yet.



PBX menu

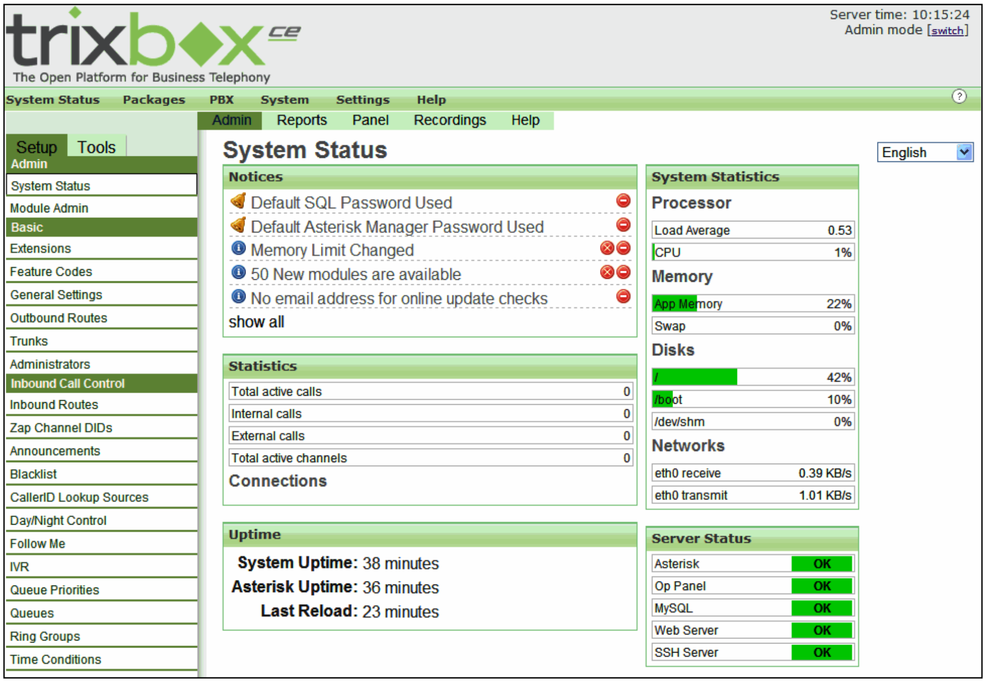
Under the **PBX** menu are PBX configuration and reporting tools. These include:

- **PBX Settings** for managing all of the PBX-related configurations
- **Gizmo5** tool for purchasing and managing Gizmo5 SIP trunks
- **Config File Editor** tool for editing configuration files
- **PBX Status** to provide detailed information about your trixbox installation
- **Endpoint Manager** to provision phones
- **Bulk Extensions**, which allows you to create large numbers of extensions by uploading a delimited text file
- **CDR Report** to see the system call logs

As we progress through the book we will look at some of these tools in greater detail and use them to configure the phone system, set up phones, and make custom configuration options. For now, let's continue our walkthrough of the tools and see what each one of them does.

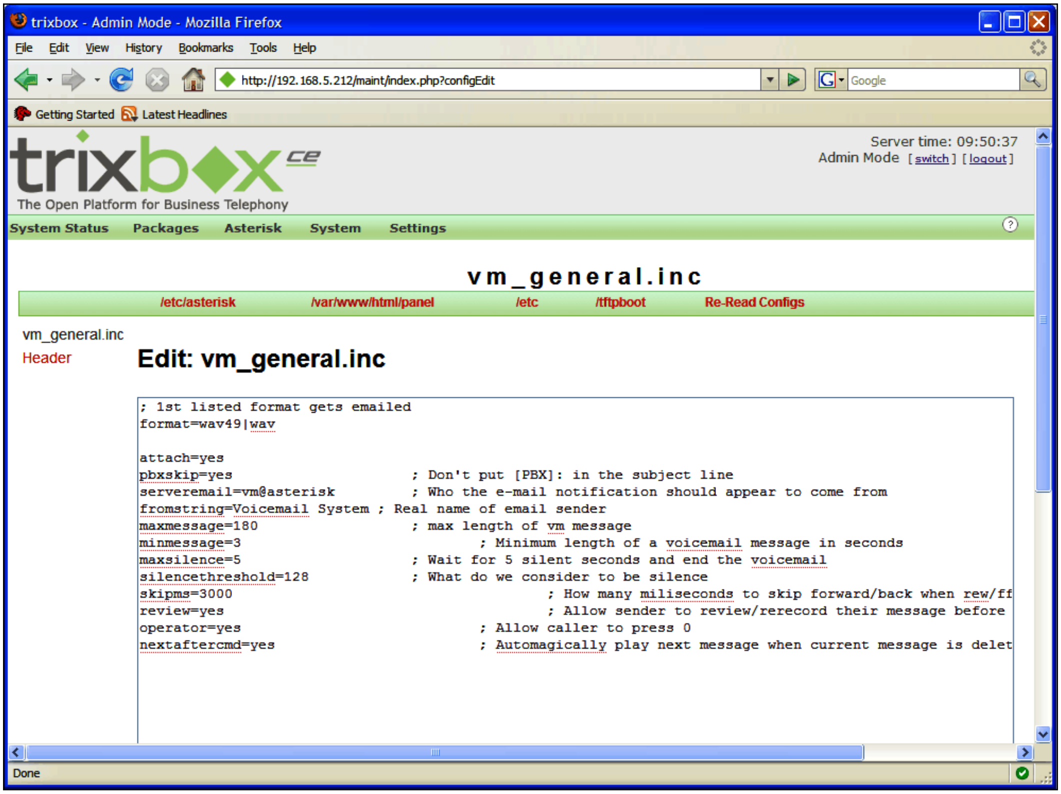
PBX Configuration

As you will see in the next chapter, the PBX Configuration tool is the primary tool for managing the PBX functionality of the system. Using the PBX Configuration tool, we configure the extensions, trunks, call menus, and other aspects of the phone system.



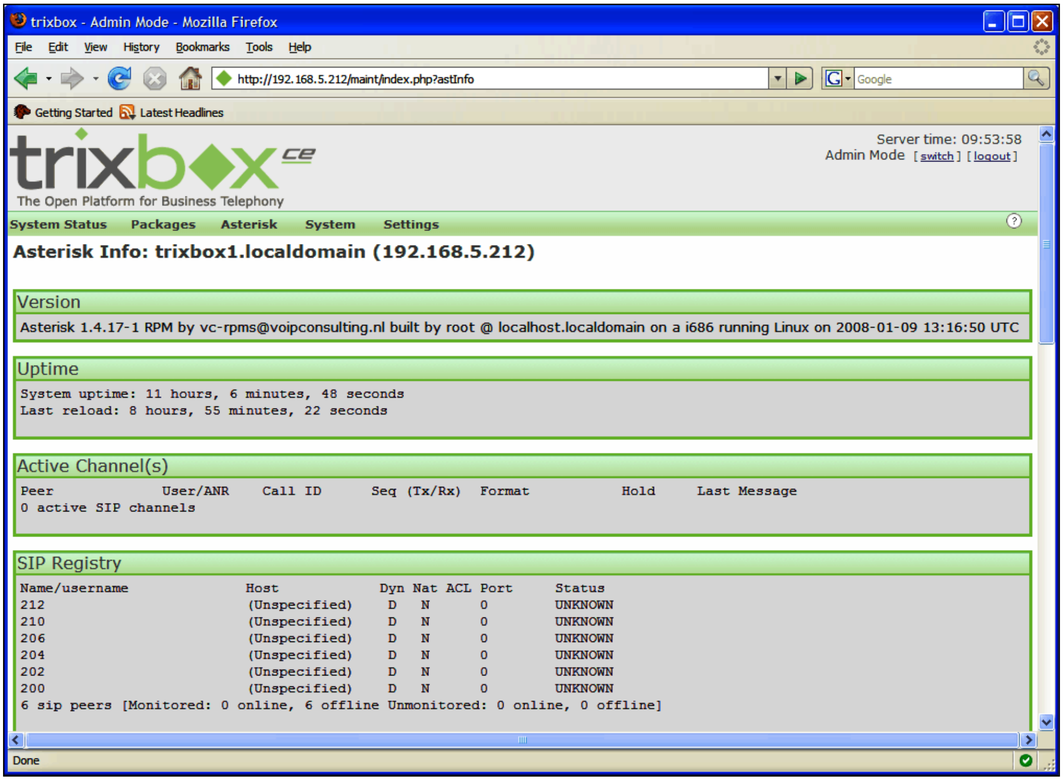
Config File Editor

The Config File Editor tool is basically a simple web-based text editor. Using the Config Edit module we can edit configuration files that we may not have a tool within the trixbox dashboard to manage. We will use this tool throughout this book to edit different files to fine-tune the system for our needs and to add features that aren't included by default in trixbox CE.



PBX Status

While the System Status display shows some basic information about your trixbox CE installation, the PBX Status module provides very detailed information that can help troubleshoot issues that you see on the System Status page.

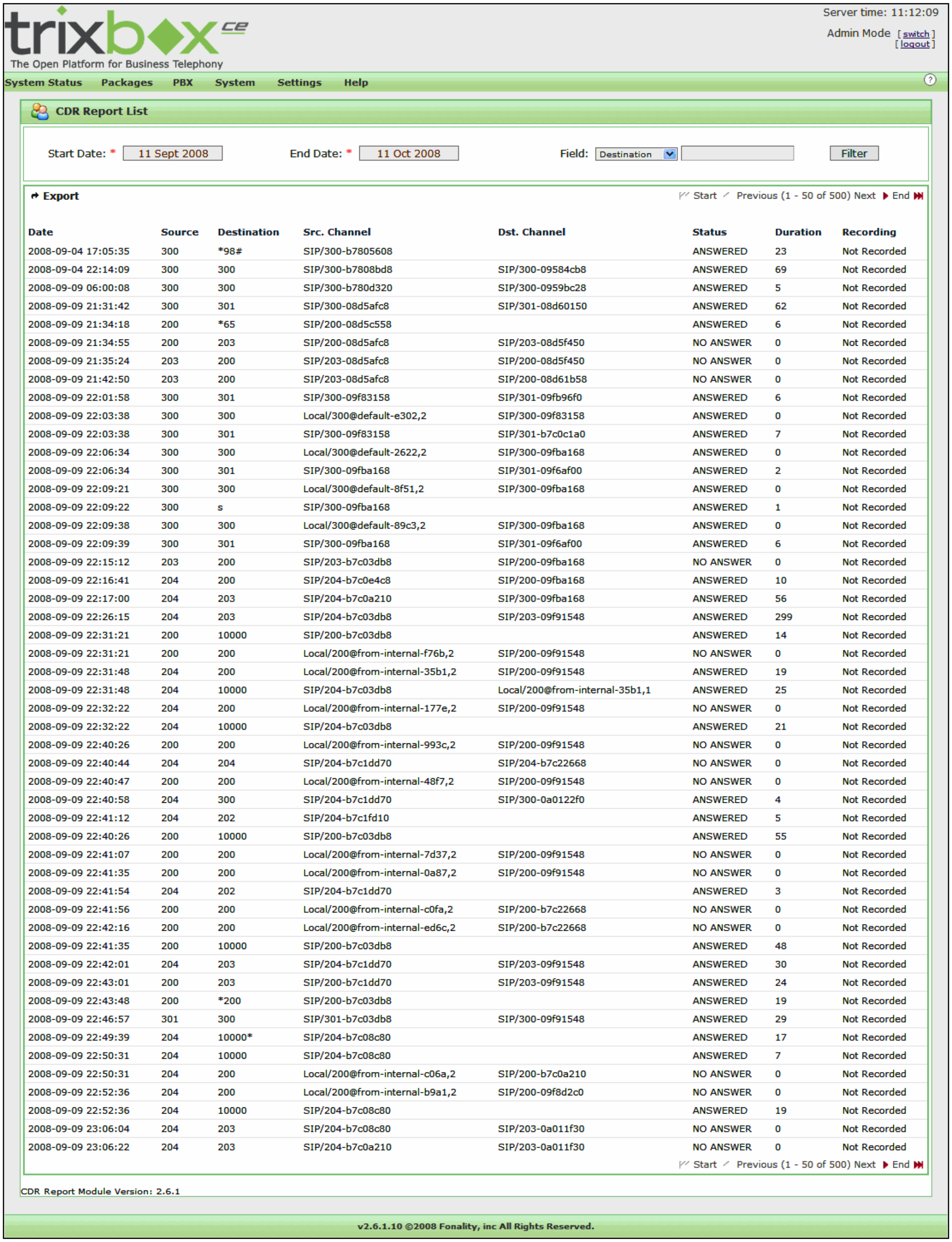


Endpoint Manager

One of the things that really separate trixbox CE from other similar products is the Endpoint Manager. By using the Endpoint Manager you can easily set up supported phones by scanning the network to find available phones, and with only a few mouse clicks, the Endpoint Manager will create configuration files for each device.

CDR Reports

In order to view the calls that are placed into and out of the system, you can use the CDR Reports tool. If you have recorded calls you will also be able to click on a link to play back the recorded sound file.

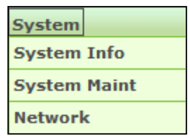


Date	Source	Destination	Src. Channel	Dst. Channel	Status	Duration	Recording
2008-09-04 17:05:35	300	*98#	SIP/300-b7805608		ANSWERED	23	Not Recorded
2008-09-04 22:14:09	300	300	SIP/300-b7808bd8	SIP/300-09584cb8	ANSWERED	69	Not Recorded
2008-09-09 06:00:08	300	300	SIP/300-b780d320	SIP/300-0959bc28	ANSWERED	5	Not Recorded
2008-09-09 21:31:42	300	301	SIP/300-08d5afc8	SIP/301-08d60150	ANSWERED	62	Not Recorded
2008-09-09 21:34:18	200	*65	SIP/200-08d5c558		ANSWERED	6	Not Recorded
2008-09-09 21:34:55	200	203	SIP/200-08d5afc8	SIP/203-08d5f450	NO ANSWER	0	Not Recorded
2008-09-09 21:35:24	203	200	SIP/203-08d5afc8	SIP/200-08d5f450	NO ANSWER	0	Not Recorded
2008-09-09 21:42:50	203	200	SIP/203-08d5afc8	SIP/200-08d61b58	NO ANSWER	0	Not Recorded
2008-09-09 22:01:58	300	301	SIP/300-09f83158	SIP/301-09fb96f0	ANSWERED	6	Not Recorded
2008-09-09 22:03:38	300	300	Local/300@default-e302,2	SIP/300-09f83158	ANSWERED	0	Not Recorded
2008-09-09 22:03:38	300	301	SIP/300-09f83158	SIP/301-b7c0c1a0	ANSWERED	7	Not Recorded
2008-09-09 22:06:34	300	300	Local/300@default-2622,2	SIP/300-09fba168	ANSWERED	0	Not Recorded
2008-09-09 22:06:34	300	301	SIP/300-09fba168	SIP/301-09f6af00	ANSWERED	2	Not Recorded
2008-09-09 22:09:21	300	300	Local/300@default-8f51,2	SIP/300-09fba168	ANSWERED	0	Not Recorded
2008-09-09 22:09:22	300	s	SIP/300-09fba168		ANSWERED	1	Not Recorded
2008-09-09 22:09:38	300	300	Local/300@default-89c3,2	SIP/300-09fba168	ANSWERED	0	Not Recorded
2008-09-09 22:09:39	300	301	SIP/300-09fba168	SIP/301-09f6af00	ANSWERED	6	Not Recorded
2008-09-09 22:15:12	203	200	SIP/203-b7c03db8	SIP/200-09fba168	NO ANSWER	0	Not Recorded
2008-09-09 22:16:41	204	200	SIP/204-b7c0e4c8	SIP/200-09fba168	ANSWERED	10	Not Recorded
2008-09-09 22:17:00	204	203	SIP/204-b7c0a210	SIP/300-09fba168	ANSWERED	56	Not Recorded
2008-09-09 22:26:15	204	203	SIP/204-b7c03db8	SIP/203-09f91548	ANSWERED	299	Not Recorded
2008-09-09 22:31:21	200	10000	SIP/200-b7c03db8		ANSWERED	14	Not Recorded
2008-09-09 22:31:21	200	200	Local/200@from-internal-f76b,2	SIP/200-09f91548	NO ANSWER	0	Not Recorded
2008-09-09 22:31:48	204	200	Local/200@from-internal-35b1,2	SIP/200-09f91548	ANSWERED	19	Not Recorded
2008-09-09 22:31:48	204	10000	SIP/204-b7c03db8	Local/200@from-internal-35b1,1	ANSWERED	25	Not Recorded
2008-09-09 22:32:22	204	200	Local/200@from-internal-177e,2	SIP/200-09f91548	NO ANSWER	0	Not Recorded
2008-09-09 22:32:22	204	10000	SIP/204-b7c03db8		ANSWERED	21	Not Recorded
2008-09-09 22:40:26	200	200	Local/200@from-internal-993c,2	SIP/200-09f91548	NO ANSWER	0	Not Recorded
2008-09-09 22:40:44	204	204	SIP/204-b7c1dd70	SIP/204-b7c22668	NO ANSWER	0	Not Recorded
2008-09-09 22:40:47	200	200	Local/200@from-internal-48f7,2	SIP/200-09f91548	NO ANSWER	0	Not Recorded
2008-09-09 22:40:58	204	300	SIP/204-b7c1dd70	SIP/300-0a0122f0	ANSWERED	4	Not Recorded
2008-09-09 22:41:12	204	202	SIP/204-b7c1fd10		ANSWERED	5	Not Recorded
2008-09-09 22:40:26	200	10000	SIP/200-b7c03db8		ANSWERED	55	Not Recorded
2008-09-09 22:41:07	200	200	Local/200@from-internal-7d37,2	SIP/200-09f91548	NO ANSWER	0	Not Recorded
2008-09-09 22:41:35	200	200	Local/200@from-internal-0a87,2	SIP/200-09f91548	NO ANSWER	0	Not Recorded
2008-09-09 22:41:54	204	202	SIP/204-b7c1dd70		ANSWERED	3	Not Recorded
2008-09-09 22:41:56	200	200	Local/200@from-internal-c0fa,2	SIP/200-b7c22668	NO ANSWER	0	Not Recorded
2008-09-09 22:42:16	200	200	Local/200@from-internal-ed6c,2	SIP/200-b7c22668	NO ANSWER	0	Not Recorded
2008-09-09 22:41:35	200	10000	SIP/200-b7c03db8		ANSWERED	48	Not Recorded
2008-09-09 22:42:01	204	203	SIP/204-b7c1dd70	SIP/203-09f91548	ANSWERED	30	Not Recorded
2008-09-09 22:43:01	200	203	SIP/200-b7c1dd70	SIP/203-09f91548	ANSWERED	24	Not Recorded
2008-09-09 22:43:48	200	*200	SIP/200-b7c03db8		ANSWERED	19	Not Recorded
2008-09-09 22:46:57	301	300	SIP/301-b7c03db8	SIP/300-09f91548	ANSWERED	29	Not Recorded
2008-09-09 22:49:39	204	10000*	SIP/204-b7c08c80		ANSWERED	17	Not Recorded
2008-09-09 22:50:31	204	10000	SIP/204-b7c08c80		ANSWERED	7	Not Recorded
2008-09-09 22:50:31	204	200	Local/200@from-internal-c06a,2	SIP/200-b7c0a210	NO ANSWER	0	Not Recorded
2008-09-09 22:52:36	204	200	Local/200@from-internal-b9a1,2	SIP/200-09f8d2c0	NO ANSWER	0	Not Recorded
2008-09-09 22:52:36	204	10000	SIP/204-b7c08c80		ANSWERED	19	Not Recorded
2008-09-09 23:06:04	204	203	SIP/204-b7c08c80	SIP/203-0a011f30	NO ANSWER	0	Not Recorded
2008-09-09 23:06:22	204	203	SIP/204-b7c0a210	SIP/203-0a011f30	NO ANSWER	0	Not Recorded

System menu

The **System** menu utilities are designed to report and manage non-PBX functions such as settings at the operating system level. These modules include:

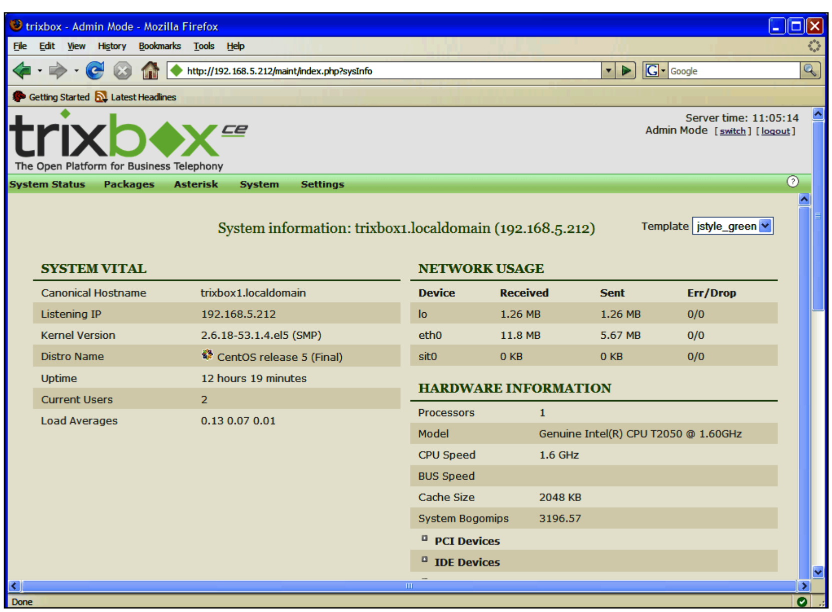
- **System Info** for advanced system information
- **System Maint** to restart asterisk, reboot the system, and disable statistics
- **Network settings** to allow you to change your IP address information on the system.



While there are only a few tools available by default, we will see in later sections of the book that we can add additional tools to the menus. Some optional additional modules that can be installed later include modules that configure a DHCP server, monitor RAID arrays, and provide backup and restore functions.

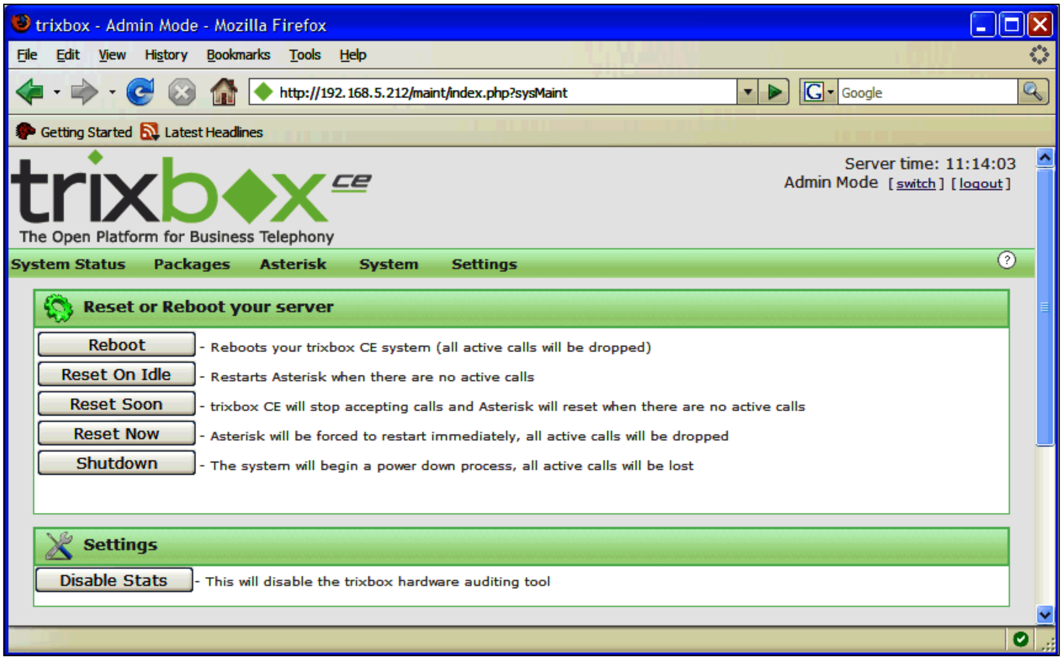
System Info

The System Status page provides some good information on your system but the System Info modules provide some very detailed information about your processor, IDE devices, USB devices, SCSI devices, and system speed.



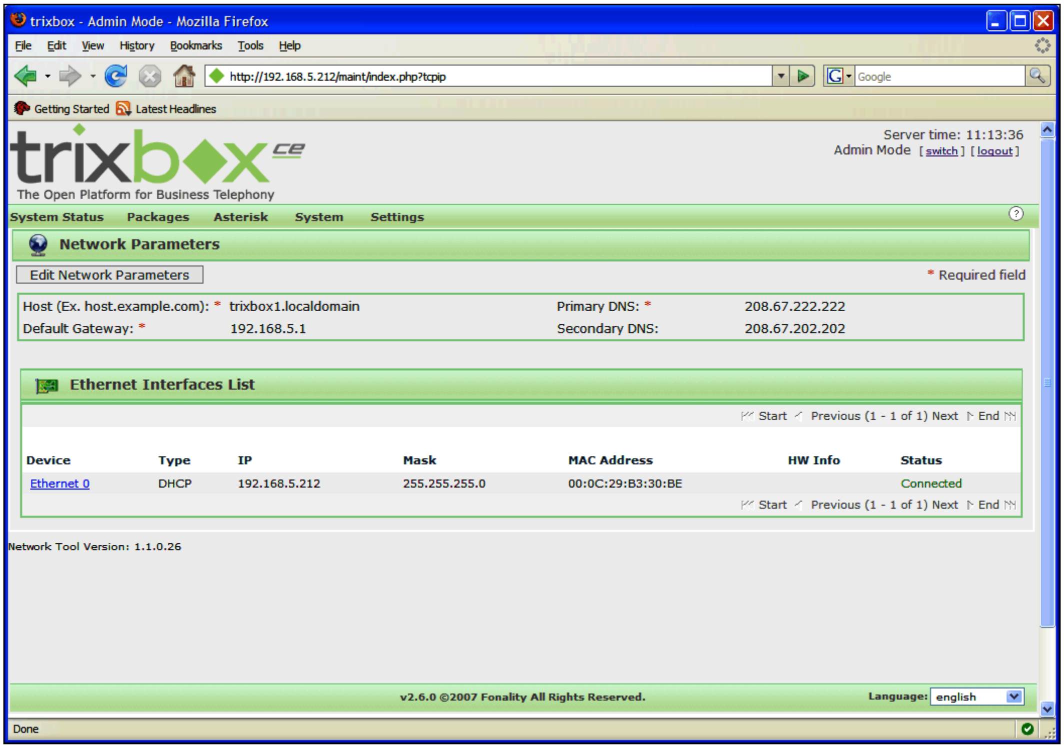
System Maint

Sometimes you may need to restart Asterisk or reboot the system or even shut it down for maintenance. Using the System Maint module you have access to all of these functions.



Network

Sometimes you will want to modify the network settings on the system, such as assigning a static IP address, changing the gateway settings, or setting the DNS servers. Using the Network module will allow you to make these changes without having to know all of the Linux command-line equivalents.



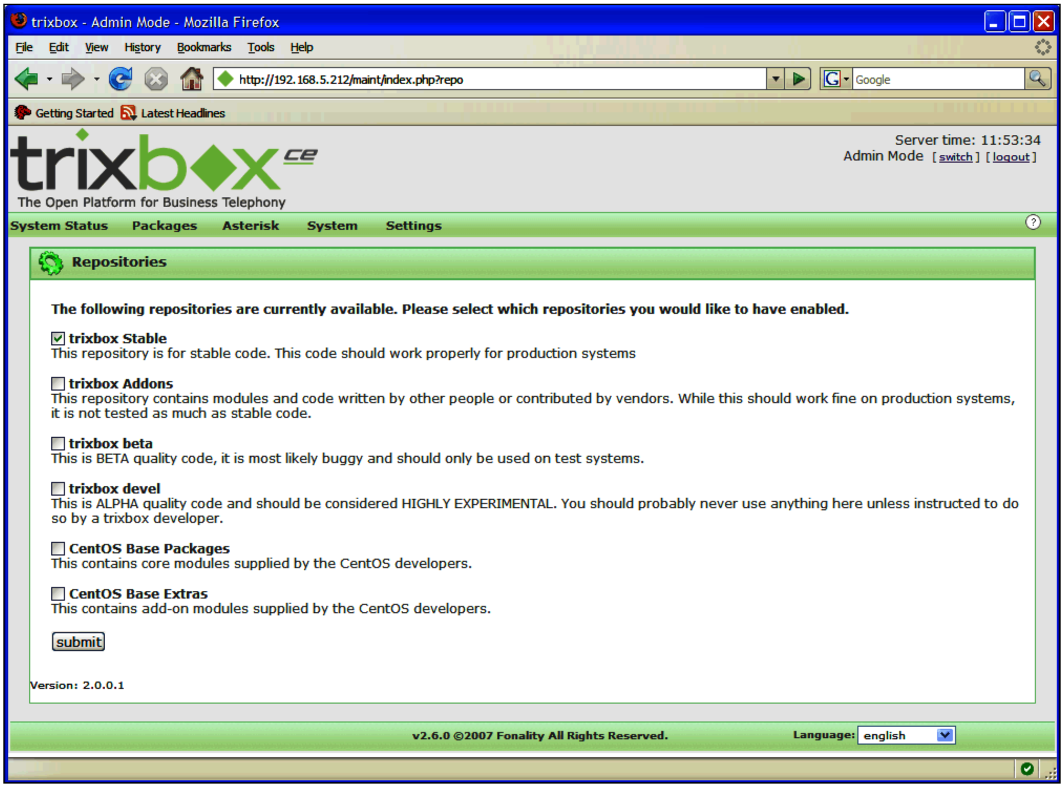
Settings menu

The **Settings** menu is the last of the **Admin Mode** menus and contains tools that control trixbox CE-related settings. The two settings packages that you will find here include:

- **Repositories** for selecting which set of files you would like the Package Manager to look in for updates and new modules
- **Registration**, which allows you to register your system with Fonality if you plan on purchasing paid support options from them

Repositories General Settings currently allows you to modify settings that are used for outbound SMTP mail relays. Another option on the System Maint page is a function to disable sending system statistics to Fonality. These system stats are anonymous and optional but Fonality does provide information about the types of equipment that is used on trixbox CE systems for manufacturers to help fund the trixbox CE development. No personal information, call logs, passwords, or voicemails are ever sent to Fonality. To see the actual list of commands that are run to create the stats that are sent, you can view the command file at `/var/adm/bin/registry_commands`.

There are several different trixbox CE repositories to choose from. By default the **trixbox Stable** repository is used, which will contain all of the standard modules and packages. Optionally, you can enable the Beta repository, which contains modules that are currently in beta test. If you really want to be adventurous you can enable the Development repository. The Development repository should never be used on a production machine unless instructed to do so by a developer to help you solve a specific problem you are having.



Summary

You should now have a good grasp of the different tools and their functions as well as where they are located within the trixbox Dashboard. You can also see how trixbox CE is more than just a collection of other tools all rolled up together as the trixbox CE Dashboard provides a lot of extra functionality that isn't provided by other tools.

4
Configuring trixbox CE

So far we have installed our trixbox CE system and covered all of the tools we will be using to set up a basic configuration. In this chapter we are going to set up a very basic system that can be used to make calls between extensions, make and receive phone calls, and utilize the basic PBX functionality. To create our first system we will be covering the following topics:

- PBX Settings
- Extensions
- Trunks
- Inbound Routing
- Outbound Routing
- IVR Setup

In later chapters we will get into more advanced setup options.

PBX Settings tool

If you recall from the previous chapter, the PBX Settings tool is the tool that we will use to set up the PBX functionality. To access the PBX Settings tool, go to the **PBX** menu and select **PBX Settings**.

The PBX Settings tool uses a modular plug-in system so that you can add, remove, and upgrade individual components. Once we go into the PBX Settings tool interface we should select which modules we want to use and check for any available updates.

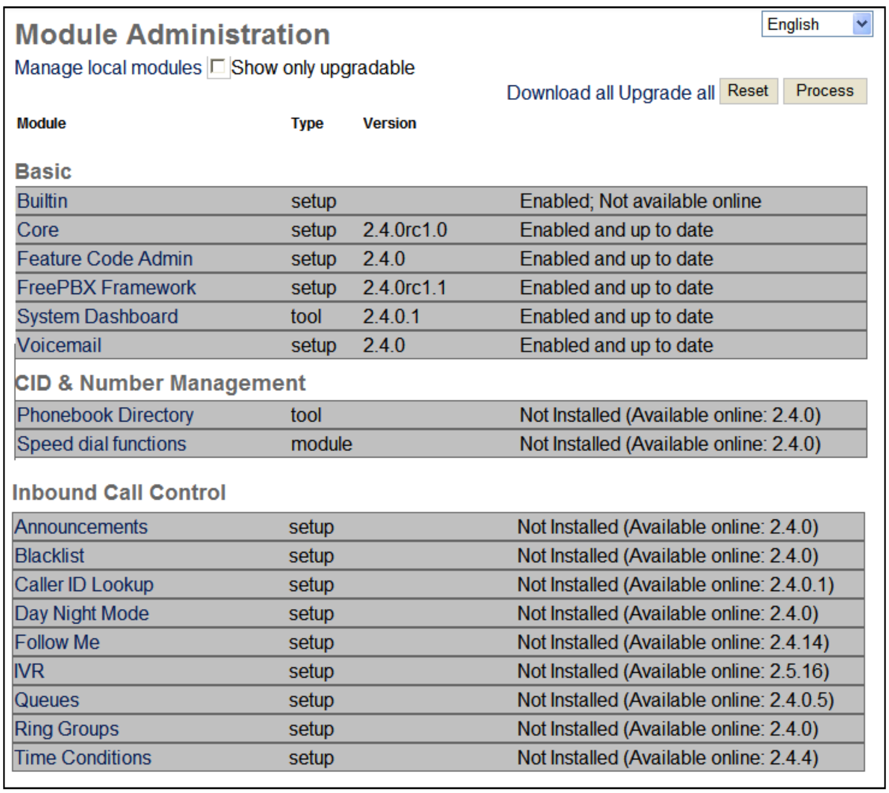
For the remainder of this chapter we will want to have the following modules enabled:

- Voicemail
- IVR
- Ring Groups

Installing PBX Configuration tool modules

After getting your system up and running you will need to install any modules into the PBX Configuration tool that you will want to use on your system. From the left-hand navigation menu of the PBX Configuration tool, select the **Module Admin** tool. The Module Admin tool will allow you to enable or disable different modules. To get a list of available modules that aren't already installed locally, use the **Check for updates online** link at the top of the page.

Once you get the list you can choose from a large list of available modules. For the rest of this chapter you will need to install the **Ring Groups**, **Voicemail**, and **IVR** modules.



Module Administration

English

Manage local modules ☐ Show only upgradable

Download all Upgrade all Reset Process

Module	Type	Version	
Basic			
Builtin	setup		Enabled; Not available online
Core	setup	2.4.0rc1.0	Enabled and up to date
Feature Code Admin	setup	2.4.0	Enabled and up to date
FreePBX Framework	setup	2.4.0rc1.1	Enabled and up to date
System Dashboard	tool	2.4.0.1	Enabled and up to date
Voicemail	setup	2.4.0	Enabled and up to date
CID & Number Management			
Phonebook Directory	tool		Not Installed (Available online: 2.4.0)
Speed dial functions	module		Not Installed (Available online: 2.4.0)
Inbound Call Control			
Announcements	setup		Not Installed (Available online: 2.4.0)
Blacklist	setup		Not Installed (Available online: 2.4.0)
Caller ID Lookup	setup		Not Installed (Available online: 2.4.0.1)
Day Night Mode	setup		Not Installed (Available online: 2.4.0)
Follow Me	setup		Not Installed (Available online: 2.4.14)
IVR	setup		Not Installed (Available online: 2.5.16)
Queues	setup		Not Installed (Available online: 2.4.0.5)
Ring Groups	setup		Not Installed (Available online: 2.4.0)
Time Conditions	setup		Not Installed (Available online: 2.4.4)

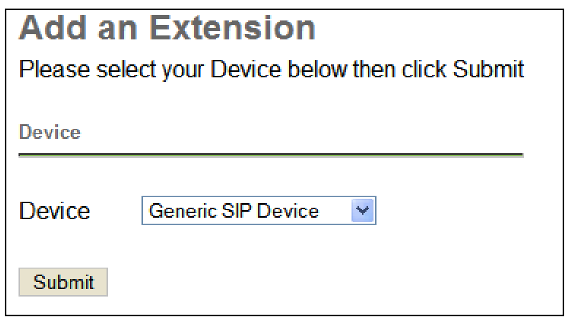
Setting up an extension

The first thing we want to do is to set up some extensions and then make some calls between them to ensure that the system is working properly.

> One important thing to keep in mind is that when you are saving changes in the PBX Configuration tool, you are only saving the changes into a database. The new changes aren't saved as the Asterisk configuration files until you click on the **Apply Configuration Changes** bar at the top of the page.

Let's start with creating a few new extensions by clicking on the **Extensions** link in the left-hand menu.

The first screen that we will see will ask us to select the device type of the new extension. For our first few extensions we will select the default type of **Generic SIP Device**.



Once we get to the new extension screen we are ready to enter the information we want to use for this extension. As you can see on the Extension Setup screen, there are a large number of options we can use to customize each extension. For now, the only options we want to set are the following:

- **Extension number**: This will be the number we dial to reach this particular extension.
- **Display Name**: This is the name that will be shown on internal calls within the office and is also used by the *dial-by-name* directory system.
- **Secret**: The Secret is the password used by the device to authenticate to the PBX system. While the examples in this book show the Secret as being the same as the Extension number, we highly recommend using much stronger passwords. Having the Secret being the same as the Extension number is a major security risk if you allow extensions to connect to the Internet.

With these three basic pieces of information we can set up a basic extension. We will dig deeper into the other extension options in Chapter 12. Now is also a good time to look at the voicemail options as most people will have voicemail set up on their phone.

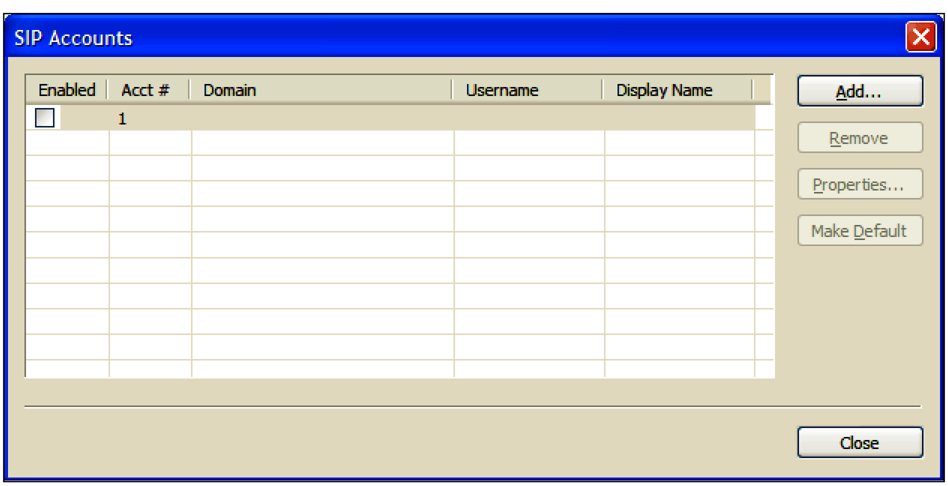
At a minimum you will need to enable the voicemail for the extension and provide a password that will be used to access the voicemail system. You can also set an email address to which to send the voicemail files. Other voicemail options we want to look at right now include:

- **Play CID**: Turning on this option will play the caller ID of the phone number that left the voicemail
- **Play Envelope**: If this option is enabled, the date and time that voicemail message was left will be played back when you listen to the message
- **Delete Vmail**: If you are sending the voicemail files to an email address, this option will determine if you want to delete the message from the system after it sends the email to the specified email address

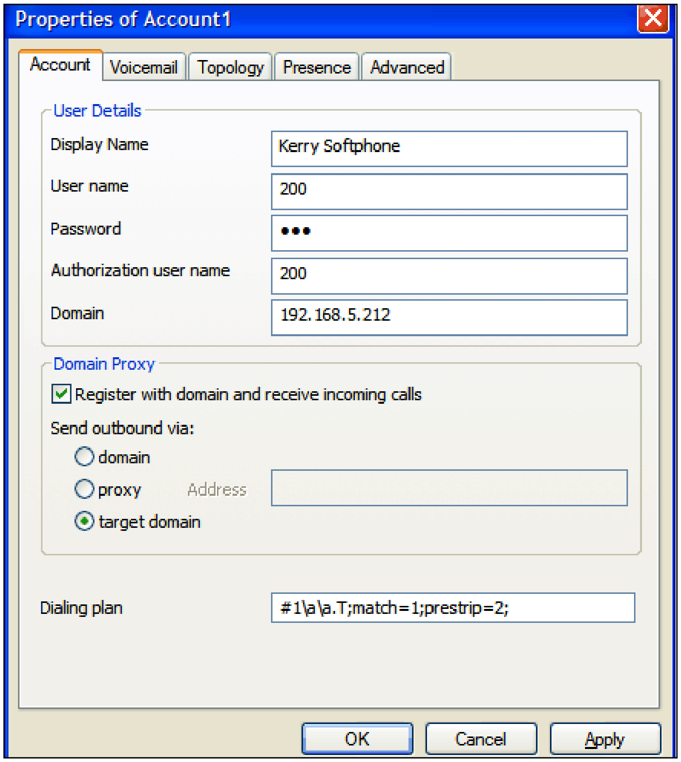
With the settings described here, we should have an extension ready to go, and we are now ready to set up a phone to connect to it. For this first device setup we will use the X-Lite softphone, which is available for free from `http://www.counterpath.com`.

Setting up a softphone

Make sure that you have X-Lite installed and let's run it for the first time and get it set up to connect to our new trixbox CE system. The very first time you run X-Lite, it should come up and ask for new account settings. If it doesn't do that, then simply right-click on the X-Lite logo to open up the SIP Account Settings dialog box.



Add a new SIP account by clicking on the **Add...** button. This will bring up the SIP Account Properties page.



On the Account Properties page we need to know what information to put into the fields in order to get the softphone to connect properly.

- **Display Name**: The display name is for reference only. This has nothing to do with the phone registering with the PBX system.
- **User Name**: This field needs to contain the extension number for the extension you want this phone to connect as.
- **Password**: The password field on the softphone needs to contain the password you entered into the Secret field when you created your extension.
- **Authorization User Name**: Just like the **User Name** field above, this field needs to contain the extension number of the extension we are setting up.
- **Domain**: On most phone devices this field is called the Proxy Server. This is the IP address or host name of the PBX system we are connecting to.
- **Domain Proxy**: You should leave the default settings in this section alone as they are correct for connecting to a trixbox CE system.

To test our softphone setup we can dial *65 on the phone, and if everything is working properly, then the phone system should read back the extension number.

**65 is a feature code that is built into the system along with a number of other feature codes. Later in this chapter we will look at a larger list of available feature codes.*

Set up a couple of extensions and test making calls between them.

Basic troubleshooting

As long as the computer you are putting the softphone on is on the same subnet as the PBX system then there is very little to go wrong. Usually when a softphone won't connect to the system and they are on the same network, the most common cause is going to be a typo in the username, password, or host IP address.

Trunks

At this point we have a basic system running with internal extensions. Now it's time to communicate with the outside world. If this is your first installation, you might not have any hardware to use to connect to a PSTN line, so we will set up a VoIP trunk for our first setup.

Setting up a VoIP account

If this is your first trixbox or Asterisk-based system, you may not have any hardware to connect a phone line to, so our first trunk may be a VoIP account to an ITSP. We can select from a large number of ITSPs, and the configuration is relatively similar for all of them with only slight differences. For our first VoIP trunk, we will do an example of setting up a trunk with Vitelity. I have been using Vitelity for several years and highly recommend it. Once you have created an account at Vitelity.com, make sure it is set to route to your SIP PBX. This will set up the Vitelity side properly. If you ever need to look up the settings to use for your account, just go to the **Support** tab and you can view the examples. For this example, we will put in all the settings and cover some explanation of what the settings do.

Creating the trunk

From within the PBX Configuration tool interface, we need to select **Trunks** from the left-hand navigation menu, and then select the link to add a new SIP trunk.

General Settings

The settings in this section control the basic behavior of the trunk.

- **Outbound Caller ID**: For each trunk you can specify the outbound caller ID. With analog lines you cannot override the caller ID; however, some PRI providers and some ITSPs will allow you to override caller ID. In your choice of providers, you may need to find out if your provider will allow this in case you will need to use this feature.
- **Never Override Caller ID**: Some providers will drop calls if you try to override the caller ID, to prevent caller ID spoofing. If your provider will drop calls if you change the caller ID, then check this box to prevent any part of the system from trying to send out a different caller ID down this trunk.
- **Maximum Channels**: This setting sets the maximum number of available channels on this trunk. If the system knows how many channels are available per trunk, then failing over to another trunk will not require getting failures before trying a different trunk. For our Vitelity account, this should be set to 2.

- **Disable Trunk**: If you need to disable the trunk but don't want to delete it and lose all the settings, you can check this box and prevent this trunk from being used.
- **Monitor Trunk Failures**: If you have this option checked then any errors on this trunk will be sent to the specified AGI script that can do things such as log, report, or email the errors.

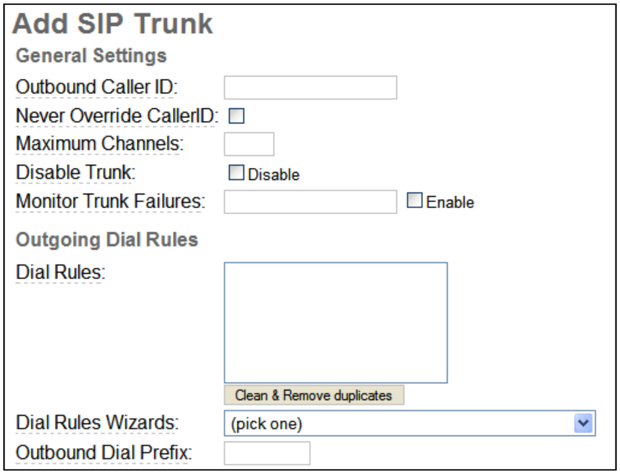
Outgoing Dial Rules

Outgoing rules determine how calls are dialed on this trunk. This can be used to add digits to or remove digits from the phone number dialed. You may need to do this for certain ITSPs or if you are sending calls through another PBX system.

With most ITSPs, their systems do not know what area code is local to where you are calling from since they service companies all over the country, or even around the world. Because of this we usually need to send the area code as a prefix when someone tries to call a local phone number. We can use the Dial Rules Wizard to walk us through the most common settings. As an example of this, since my phone system is in the 949 area code, I want to send *1949* in front of any number that I dial that is only a regular seven digit phone number. To accomplish this, I will need the following dial rule:

1949+NXXXXXX

- **Outbound Dial Prefix**: If you need to dial a specific prefix on every call on this trunk, then it is easier to use the Outbound Dial Prefix. This is most often used to dial a 9 first if dialing through another PBX, or for putting a *w* (slight delay) when using analog lines if you have lines that don't provide dialtone fast enough.

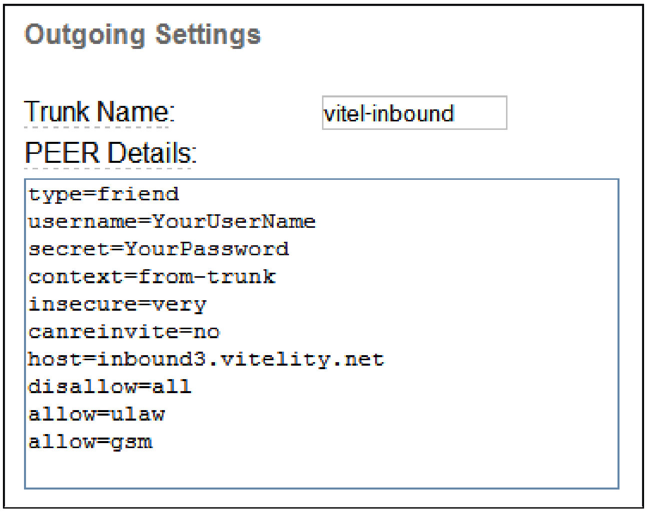


Outgoing Settings

A SIP trunk has settings for both incoming and outgoing settings as sometimes you may use a specific trunk for inbound calls only, outbound calls only, or both. These settings provide for setting up outbound calls to your ITSP.

- **Trunk Name**: You will need to find out if you need to use some specific trunk name that your provider is looking for when dialing out. In the case of Vitelity, we will use *vital-inbound* as our trunk name.
- **PEER Details**: The PEER Details provide settings and authentication for connecting to your provider. The following settings are used for connecting to Vitelity:
    - **type=friend**: The `type` setting determines the direction of this trunk. The available options are `Peer` (outbound), `User` (inbound), `both` (inbound and outbound).
    - **Username=<your username>**: This field contains your SIP account username.
    - **Secret=<your password>**: This field contains your SIP account password.
    - **Context=from-trunk**: The `context` defines the section of the dialplan where the call will be placed when a call comes in. Normally we want this to be set to `from-trunk` as that is a built-in context that manages inbound calls.
    - **Insecure=very**: This setting determines the type of security that is used on the trunk. Using the `very` option allows calls to come in from a registered host.
    - **Canreinvite=no**: Allowing reinvites to be on will cause the media path to run directly between the two endpoints. This does not work with all endpoints or all providers, so leaving this set to `no` is the most compatible setting.
    - **host=inbound3.vitelity.net**: The `host` setting is the proxy server you are trying to connect to.
    - **disallow=all**: The `disallow` setting determines which codecs should be disabled. If we want to use only specific codecs, we can disable all available codecs and then only specify the ones that we want to use.
    - **Allow=ulaw / Allow=gsm**: The `Allow` setting determines which specific codecs will be enabled. In the case of Vitelity, we will allow G711 (Ulaw) and GSM.

Depending on your SIP provider you may need to use some of the same settings in the **Incoming Settings** section. With Vitelity, you do not need to put anything into the **USER Details** section.



Registration string

The registration string is used to register with a provider in order to receive calls. The registration string is typically in the format of `<username>:<password>@hostname:port`. An example of this with Vitelity would be:

```
camelco:password123@inbound3.vitelity.net:5060
```

Basic troubleshooting

In order to see if the trunk is working properly we can log into the system using either SSH or from the console. First we need to go into the Asterisk CLI:

```
[trixbox1.localdomain ~]# asterisk -vr

Asterisk fon_o_1.2.17, Copyright (C) 1999 - 2006 Digium, Inc. and others.

Created by Mark Spencer <markster@digium.com>

Asterisk comes with ABSOLUTELY NO WARRANTY; type 'show warranty' for
details.

This is free software, with components licensed under the GNU General
Public License version 2 and other licenses; you are welcome to
redistribute it under certain conditions. Type 'show license' for
details.

=========================================================================
```

```
Connected to PBXtra Core fon_o_1.2.17 currently running on trixbox100402
(pid = 3904)
Verbosity is at least 41
trixbox100402*CLI>
```

Once we are in the Asterisk CLI, we can see what accounts are registered by using the `sip show registry` command. We can also see the status via the **PBX Status** module located under the main **PBX** menu.

```
trixbox1*CLI> sip show registry
Host                              Username       Refresh      State
inbound3.vitelity.net:5060        kgarrison1     45           Registered
```

Setting up an outbound route

An outbound route plan can be anything from a simple setup to send all calls out from a specific trunk to a complex, multi-point least-cost-routing plan using a variety of trunk types and dial patterns. With other systems you might have set up least-cost routing rules so that local calls went to one provider, long-distance in-state calls would go to another provider, and yet another provider might be used for out-of-state long-distance calls. With the PBX Configuration tool you can still set up rules like this and use as many providers as needed. Some companies that want to stick with traditional lines, like a PRI for their regular calls, may want to use a VoIP provider for international calls to save money.

We also use outbound routes to set up failover routes for capacity overflow and trunk failure conditions.

There is a pre-set outbound route included in the default trixbox CE installation. With this configuration, the system will take any number that starts with 9 and send it out through the specified route. If you created your VoIP trunk earlier in this chapter, you just need to change the route sequence to specify that route, save your changes, click on the **Apply Bar** and you should be ready to make phone calls by dialing 9 + phone number.

Route options

There are a few route options you should know about when setting up outbound routes.

- **Route Name**: This is for reference purposes only.
- **Route Password**: If you have a password set for a route then the system will prompt the user before dialing out. This could be used for blocking 900 numbers or international calls.

- **Emergency Dialling**: If this trunk is used for emergency calls, you can set this option and the system will use the emergency caller ID setting of the extension that is dialing out. This is useful for telecommuters; you could set their emergency caller ID to their home phone number so that if they dial 911 from their remote phone, the emergency center would get the address where the phone is physically located.
- **Intra Company Route**: If this option is selected then the device's internal caller ID is used instead of using the outbound caller ID. If you have branch offices connected together on this trunk, you want this option set to make sure the called party sees the extension of the person who called instead of the company outbound caller ID setting.
- **Music On Hold**: You can choose which Music On Hold category will be used for calls that go out on this route.
- **Dial Patterns**: Only calls that match this list of patterns will be allowed to go out on this route.
- **Trunk Sequence**: This is the sequence of trunks that are used to place this call.

Pattern matching

What if we wanted to match on some specific number patterns? We can use the Dial Patterns section to specify what patterns will match for this trunk. In order to create a working pattern we need to know a few things about how to create one. The following characters can be used to create a pattern:

- **X**—matches any digit from 0 – 9.
- **Z**—matches any digit from 1 – 9.
- **N**—matches any digit from 2 – 9.
- **[12347-9]**—matches a number or sequence within the brackets. This example would match on 1,2,3,4,7,8,9.
- **.**—acts as a wildcard character matching any combination of characters.
- **|**—separates the prefix from the number to be dialed. As you can see if the pre-configured outbound route uses 9|., it would strip off the 9 and send all remaining digits to the trunk.

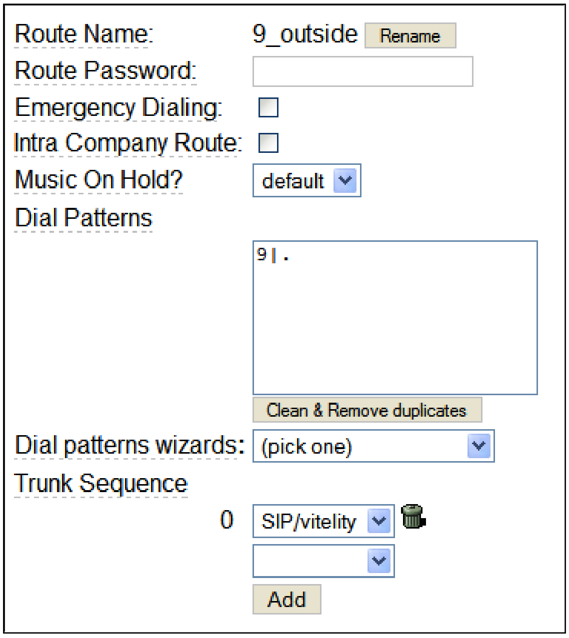
Examples

Pattern matching can be a little daunting for newcomers but with a little practice, designing a good call flow shouldn't be too difficult. As we already saw with the default setup **9|.**, we'll take the dialed string **917145551212** and strip the 9 off and then match the remaining characters (because of the use of the **.** as a wildcard) and send the matched number out through the selected trunk.

For companies that want to send international calls through a specific route, you could use a match such as **011.**, which would then match on any number of any length that begins with **011**.

Another common example is when setting up interoffice dialing between different systems. If Office A used extensions in the 300 range and Office B used extensions in the 400 range then Office A would set the interoffice route to match on **[4]XX** and Office B would create a pattern to match on **[3]XX**.

One nice feature of the PBX Configuration tool is the **Dial patterns wizard** which will populate the **Dial Patterns** field just by selecting your choice from the pull-down menu.



Setting up a ring group

A ring group is simply a list of extensions that can be grouped together so that they all ring at the same time. A ring group is a good choice when you have plenty of people available to answer your incoming calls. If you are in a position where you sometimes get more inbound calls than you have available people to handle those calls, then a Queue may be a better choice. We will look at Queues in the *PBX settings in detail* chapter.

For our first ring group, we will create a basic group using the default options. When we set up an IVR in the next section, we will be able to send calls to this ring group by pressing a key during a message playback.

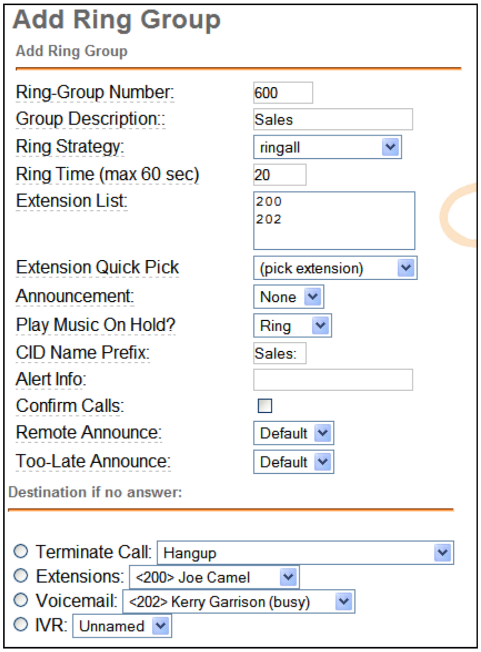
From the left-hand navigation menu, select **Ring Groups** to create your first group.

Ring group options

A ring group is used to ring a list of extensions. While there are quite a few options available, we only usually use just a few. For our first ring group we will only use a few of the possible options.

- **Ring-Group Number**: The ring group number is the extension number that is used to access the ring group.
- **Group Description**: The description is an optional field that allows you to use a descriptive name for the group.
- **Ring Strategy**: The ring strategy allows you to set how the extensions in the list will ring when a call is sent to the ring group. The available options include:
    - **Ring All**—rings all of the available extensions until one device answers or the timeout expires. This is the default setting for a new ring group.
    - **Hunt**—take turns ringing each available extension until a device answers or the timeout expires.
    - **Memory Hunt**—starts by ringing the first extensions and then adds each additional extension on each ring. This will cause the first extension to ring and then the first and second extension to ring, then the first, second, and third extensions to ring, and so on.
    - ***-prim**—the modes that end with *–prim* act as the previously mentioned modes unless the first extension is in use which will cause the other extensions to not ring.
    - **First Available**—only ring the first available channel.
    - **First Not On Phone**—ring only the first channel that is not off-hook.

- **Ring Time**: This sets the length of time for which a call will ring before expiring and being sent to the failover destination.
- **Extension List**: This is the list of the extensions that are part of the ring group.
- **Announcement**: If you have a prerecorded announcement, you can play it when a call is sent to the ring group.
- **Play Music On Hold**: While the caller is waiting for the call to be answered, you can select if you would like the caller to hear ringing, music, or neither.
- **CID Name Prefix**: You can optionally set a prefix that will be prepended to the caller ID name. This will then show the available phones what ring group the call came in on. This is good for knowing that a call is meant for a sales department versus a support department.
- **Alert Info**: If your phone device has a setting for Alert Info then you can use this field to set a distinctive ring for each ring group.
- **Confirm Calls**: This option is used to play a message to external numbers asking if they would like to accept the call or not. This option should always be used if calling out to a cell phone, to prevent the call from going to the voicemail of the cell phone if the cell phone is offline for any reason.
- **Remote Announce / Too-Late Announce**: These options are used to select prerecorded messages in conjunction with the Confirm Calls setting.
- **Destination if no answer**: If the timeout expires, then the call will be sent to the selected destination.



Setting up an IVR

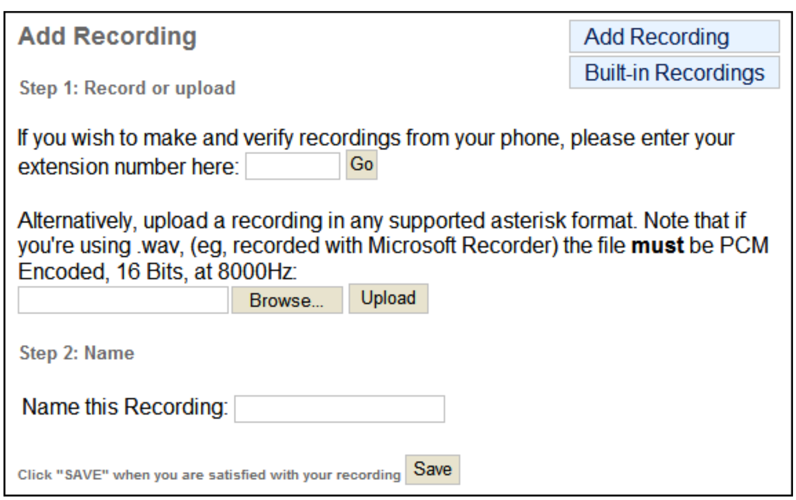
The **IVR** (**Interactive Voice Response**) system, also known as a digital receptionist, is the system by which a message is played to the caller and then the caller is allowed to use a keypress to navigate a menu and have their call routed properly. Designing a good call flow is so important to a successful installation that we have included an entire chapter on it later on in the book (Chapter 14). For now, we will create a simple message to play to callers and then set up keypresses to route the calls.

Recording messages

Before we can program our IVR, we should create the message we want to play when someone calls in. From within the PBX Configuration tool, click on the **System Recordings** in the left-hand navigation menu. This will bring up a screen that will allow you to upload a recorded file or select an extension that will be used to record the message. You can actually get excellent results using a good quality phone to do your recordings with. If you want to have someone record your messages for you, there are several companies around that offer voice talent.

For our first setup, let's use the extension that we created and use our softphone to create a message—this of course requires that you have a microphone attached to the computer.

Enter your extension into first field and click on the **Go** button.



On the second page, the system will tell you to dial ***77** on the phone to begin recording. When you dial *77, you will hear a quick beep sound telling you to start recording.

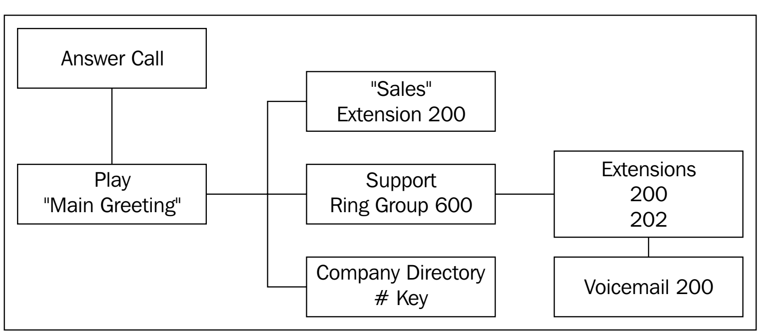
A simple example of a basic message would go something like this:

Thank you for calling the Acme Widget company; if you know your party's extension, you may dial it at any time. For a company directory, press the pound key; for sales, press 1; for support, press 2. Thank you for your call!

Once you are satisfied with your recording, give the recording a name and click on the **Save** button. The name should be descriptive so we can select the correct one when we need to use it in other areas.

Designing the IVR menu

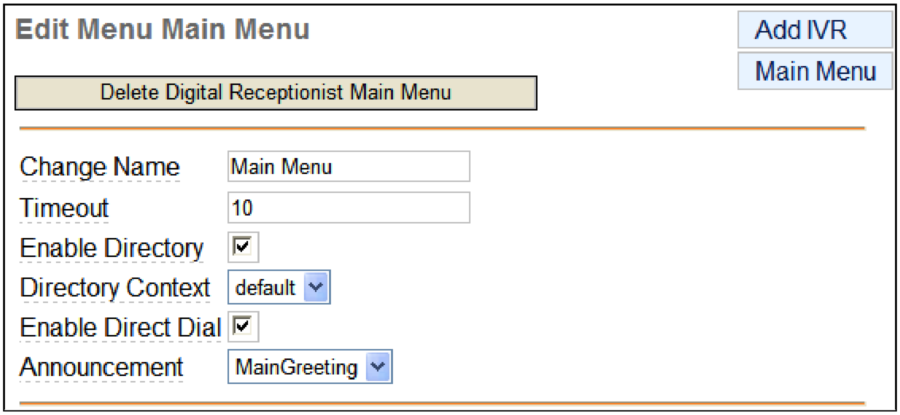
We are going to go into much more detail about IVR design in the next chapter; right now we just want to focus on getting a basic understanding of how the system works. It is always a good idea to design a flowchart for your system; the following illustration shows the diagram of the system we are going to create:



To set up IVR, select **IVR** from the left-hand navigation menu within the PBX Configuration tool. On the first screen, select the button to create a new IVR. The IVR creation page is broken up into general settings for this specific IVR and a section for keypresses and actions on the button.

In the general settings area, you have the following settings:

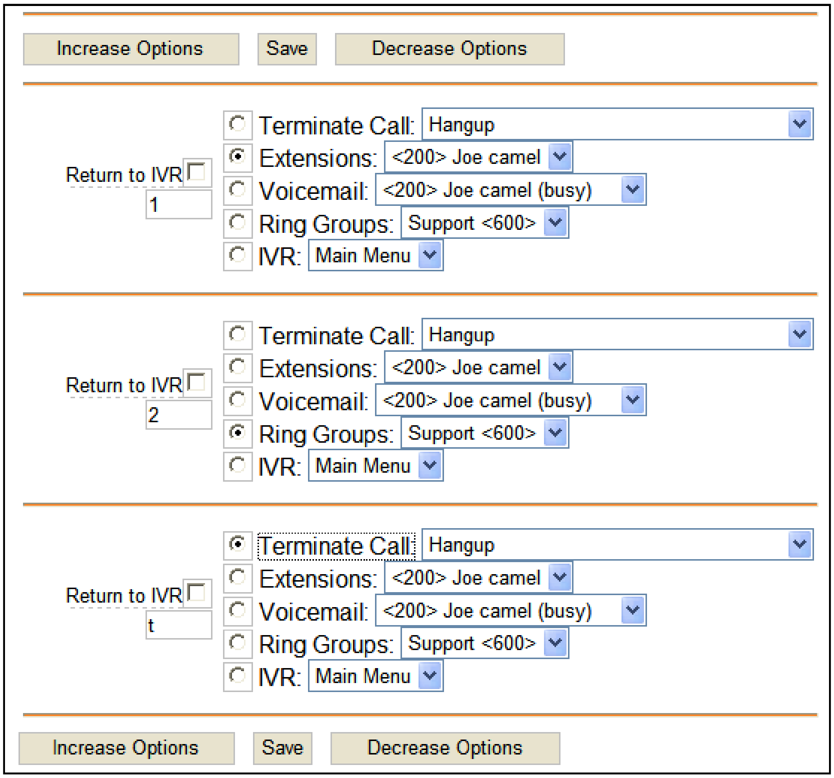
- **Change Name**—allows you to select a descriptive name for this IVR
- **Timeout**—how long to wait after playing the announcement before timing out
- **Directory Context**—the voicemail directory context used when a caller presses the # key. This should usually just be set on **default**
- **Enable Direct Dial**—leave this checked if you want callers to be able to dial extensions directly
- **Announcement**—this is the used to select the pre-recorded announcement to be played when the caller hits this IVR



In the keypress selection area, we can now map keypresses to different destinations. At this point there are only a few possible destinations; however, as we make the system more complex with announcements, call queues, and other features, more destinations will be available for us to choose. We can also use 't' to be able to set a specific destination if no keypress is detected before the timeout value expires, and we can use 'i' to go to a specific destination if someone presses an invalid key.

For our example we are going to have the following mapping:

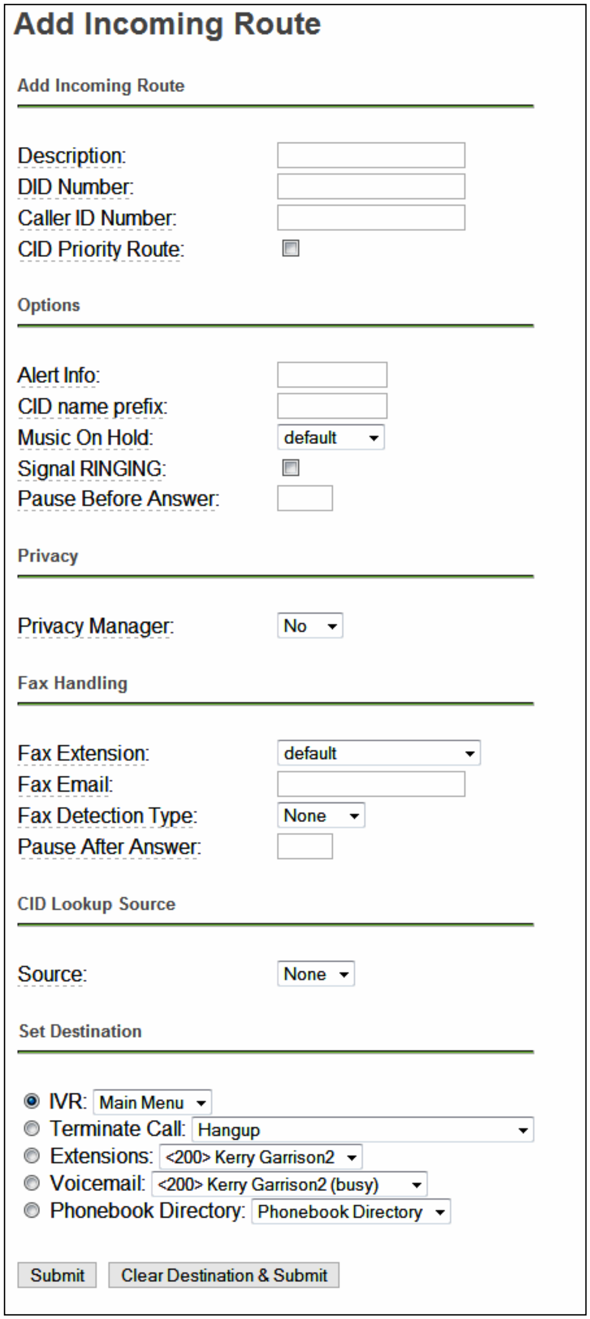
Keypress	**Destination**
1	Extension 200
2	Ring Group 600
Timeout	Hangup
#	Company Directory



Setting up an inbound route

We now have all of the basic building blocks put together and many people get to this point and end up posting in the forums that they can make calls but can't receive any. This is because you need to have at least a default blank inbound route—this tells the system how inbound calls are to be handled by the system.

For a basic configuration, you only need to create an inbound route with all of the fields blank; select **IVR** as the destination, select **Submit**, and your system will be ready to receive calls.



Summary

At this point we have gone through all the basic steps in getting a simple system up and running—from getting the modules you need installed, to setting up an extension, VoIP provider, IVR, and inbound and outbound routes. You now have all of the basic building blocks to get your initial system going. In the next chapter, we will take these concepts further and look at best practices for setting up an efficient system.

5
Planning your trixbox deployment

When planning your production environment, the more time you spend on planning the less time you will spend in actually programming the system. A poorly designed system will cause you to make numerous changes after the system is up and running. Most businesses will not tolerate too many changes after their system is up and running. Having to redo the system over and over to try to meet a client's needs is the surest way to an unhappy client. Proper planning can also reduce the administrative burden by having a known layout that makes sense. We should ensure that we properly plan the deployment and configuration of the system so that it meets the needs of the client as well as properly utilizes the features of the phone system.

To help plan and deploy our system, we will look at each segment of a good design to see how it relates to the overall design.

During this chapter:

- We will develop flowcharts and spreadsheets to help us design our systems so that we ensure we have covered all of the bases and that we have a well thought-out plan.
- We will create some spreadsheets to help us during the design process; these finished spreadsheets can be downloaded from the book's web site at `http://www.packtpub.com/support`.

The design plan

There are a number of elements to the design process that we need to consider when planning a deployment, such as the physical environment and infrastructure, which will control the stability and physical security of the system, make sure there is adequate ventilation and cooling, control access to phone lines, and so on. Most of this can be very specific to individual locations. Besides physical planning, the more important area which we will discuss in this chapter is around the configuration of the phone system itself. We will begin our planning by looking at the following components:

- Extension Planning
- Ring Groups
- Call Queues
- Telco Connectivity
  - PSTN (PRI / POTS)
  - VoIP / ITSPs
  - DID Planning
- Telephone Selection
  - Hard Phones
  - Softphones
- IVR (Interactive Voice Response menus)
- Additional Requirements

Although a complete system may have many more components, these are the most common options to begin working with. In the chapter *PBX settings in detail*, we will go into even more detail with other configuration options.

Planning your extensions

The issue of how to plan your extensions comes up in forum topics on a regular basis, mostly because there is no real rule on how you should do it. If you ask ten people, you may get ten different answers as to what the best approach is. Some people think you should always use the shortest number of digits possible to support the number of users you have, others use DID numbers, while other people use things like building floor plus cube/office number, and still others use some unknown mysticism to conjure up the optimal extension length. Whatever method you finally choose, you should be sure to think through it carefully to make sure that your plan includes the ability to grow beyond the number of employees you currently have.

The approach we will use here is based on my personal preferences but takes a number of key points into consideration that you can use to come up with a plan that is right for you.

How many users?

Even though I have seen it done, I would never recommend that a company use single digit extensions. This is extremely limiting as it locks you into having 9 or less extensions, or even fewer if you want any kind of voice menu prompt when callers call into the system. I like to compare using an Asterisk-based phone system to the World Wide Web in the mid-90s. The Web allowed small companies to appear to the world no different than huge global corporations by being able to have a nice web presence on the Internet. With an Asterisk-based phone system, you have the ability to sound much bigger than you really are. If you have a voice menu for callers that says *Thank you for calling the Acme Widget Company; press 1 for Kerry, 2 for Andrew, 3 for Chris*, this gives the caller the sense that you are a very small operation.

Even if you only have the same three employees, you could have a voice menu that says *Thank you for calling the Acme Widget company; if you know your party's extension, you may dial it at any time. For the company directory, press the pound key, for sales, press 1; for billing, press 2; or press 3 for support*. This can make a big difference in the impression that the caller will get when calling you. When people call you for the first time, you want to convey an air of professionalism for whatever business or trade you are in. This also makes it easier for us to change or move employees around as we don't have to re-record the main greeting when there is an employee change.

This actually goes even further; consider how a small extension number looks on a business card. Looking at the following two examples, the first gives the impression that the company is very small and could give a potential client a bad impression, while the second does not give away the size of the company based on the extension numbering.

1. Kerry Garrison
(949) 555-1212 x2
Versus
2. Kerry Garrison
(949) 555-1212 x200

We should also keep in mind the final design of the IVR menus and how this may affect the extensions. Typically we should try to avoid extension numbers that begin with a number that we will use in the IVR menus. If we use menus like *press 1 for sales, press 2 for support, press 3 for billing*, it is often best to avoid using extensions beginning with 1, 2, or 3. If we really want to use those, it will not be a major impact on the system, but if someone is a little too slow in dialing an extension then they may be inadvertently forwarded to the wrong location. It is also a good idea to group extension number where possible so that we have specific ranges for specific functions within the phone system. We will look at this more as we create functions like ring groups and call queues.

Departmental considerations

When planning our extensions, we need to know that other components of the phone system use extension numbers besides the actual phone devices. Components such as ring groups and call queues also use extension numbers. From an organizational point of view, it is a good practice to group extensions based on departments and functions, such as 2xx for sales, 3xx for marketing, and so on. This same approach should be used when creating ring groups and queues. If we were using three digit extensions, I may switch to four digit extensions such as 2000 for sales, 3000 for marketing, or designate a specific extension in the same range (as the extensions) as the group for that department such as using 299 for sales, 399 for marketing, and so on.

While splitting extension groups up by department may not be an efficient use of number ranges, it can be very useful in terms of organization, growth, and flexibility.

Location considerations

Another factor to consider should be to take into account any different locations that may have to be addressed. You may have users that have phones at home and/or softphones for when they travel. I personally like to group remote phone extensions into a different numbering scheme to make them easier to manage. If Tim has extension 305, I may make his remote phone 505—then when I get a call from Tim, I know by his extension if he is in the office or not. Also, if I know that Tim is working from home today, I can just dial his home phone directly.

If you have branch offices, it is very important to come up with a good extension plan for call routing purposes. If you using the 300 – 399 range for extensions at both locations, it can become difficult to route calls back and forth and avoid duplicate extension numbers. In this case, I would make sure that each office starts with a unique number to avoid these kinds of problems. For example, office A would only use extensions in the 7000 – 7999 range and office B would use extensions in the 8000 – 8999 range, and so on. This would make linking the offices and providing a simple routing pattern between them much easier. In the *PBX settings in detail* chapter, we will look at linking multiple systems together, and you can see where having overlap would cause an administrative headache.

Planning exercise

In this exercise we will look at grouping extensions within our company into groups by department, and create a list of ring groups and queues if we are going to use them. The following table shows the users, extensions, primary departments, and which ring groups each person belongs to:

User	Extension	Department	Sales Group (2000)	Support Group (3000)	Marketing Group (4000)
Kerry	200	Support		X	
Andrew	210	Support		X	
Karen	305	Marketing			X
Arnold	310	Marketing	X		X
David	405	Sales	X		

There are no rules that say you should set up your extensions groups one way or the other, but we have used the guidelines that we have already discussed in this chapter to create some logical groups based on departments, and created appropriate ring groups using four digit numbers for the groups while using three digit numbers for extensions. This should work pretty well for a basic design, although again, this is purely for maintenance reasons and not because the system requires it to be this way.

A successful deployment will require very few changes after it goes in; customers have very low tolerance for too many changes, so good design is imperative. Most trixbox installations don't use more than three or four digit extensions with three being the most common. We must also take into consideration any future growth changes, changes to different locations, and other possible company changes. For example, if you are expanding into multiple territories soon, or expect that there may be a corporate merger in the future, these are important things to consider when planning your deployment.

The following table represents the extensions part of our spreadsheet that will provide us with the information we need to have in order to get the extensions configured. In this spreadsheet we will record the following:

- User's full name
- Extension number
- Inbound DID number (optional)
- Outbound CID (optional)
- Email address
- Email to voicemail (optional)

Name	Extension	Inbound DID	Outbound CID	Email address	Email to VM
Kerry Garrison	200	(949) 555-2200	Default	`kgarrison@mydomain.com`	Yes
Andrew Gillis	210		Default	`andrew@mydomain.com`	Yes

If we want to provide different voicemail to email options for users then we may also want to record the following additional information:

- Voicemail password
- Email address Attachment yes/no (if set, voicemail will be sent to an email address attached as a WAV file)
- Play Caller-ID yes/no (if set, will play the caller ID of the person who left the voicemail)
- Play Envelop yes/no (if set, will play the date and time of the voicemail that was left)
- Delete Voicemail yes/no (if set, voicemail will be deleted from the system after emailing it to the user)

Ring groups

A ring group is a group made up of a list of extensions that will all be called based on the selected ring strategy (for example, Ring All, Hunt, Memory, and so on) when the extension number assigned to the ring group is called. This is often used to send calls to any available sales or support person, allowing multiple people to act as the company receptionist, or to ring multiple devices for the same person.

If a company has more people available to take calls than the number of calls that come in, then a ring group is an appropriate solution to dealing with these inbound calls as it will ring all available phones thus minimizing the wait time for a caller to reach an agent. We can also use an external number within the ring group, such as a cell phone number, so that a call can ring both an extension and a cell phone at the same time.

Ring groups have different methods or strategies for how the extensions in the group should ring. Usually, this is set to **Ring All** to ring all available phones, or to **Hunt** to take turns calling each extension in order.

When designing our spreadsheet for creating our ring groups, we need to record the following information:

- Ring Group number (extension)
- Description
- Ring Strategy
- Extension list
- Announcement (if used)
- Destination if no one answers the call

Group Name	Group #	Ring Strategy	Ring Time	Announcement	CID Prefix	N/A Destination	Members
Sales	2000	Ring All	20	sales_main	Sales	VM310	310, 405
Support	3000	Ring All	20	support_main	Support	VM200	200, 210

Call queues

Call queues are one of the most advanced features of Asterisk and are almost as easy to set up as a ring group. In most legacy PBX systems, call queues are only available as very high-priced options, if they are even available at all.

While a ring group is designed to take calls and route them immediately to an available agent, a call queue is designed to put callers on hold and wait for an agent to become available. Call queues are more suited to situations where there are sometimes more calls coming in than there are people available to take the calls. While the caller is waiting, they can be given a message about their place in the queue and the estimated wait time. While they are on hold, they can listen to music in between announcements. Because they can accommodate peak times when calls exceed the number of agents, queues can be very useful for sales and support organizations. This is because they ensure that callers do not get a busy signal and agents don't have to juggle multiple calls while ensuring that calls are answered in the order they came in.

When we call large organizations, we usually get to experience the call queue. This is usually preceded with a message like *We are experiencing larger than normal call volume, please stand by and your call will be answered in the order that it was received*.

When designing our call queues we should consider what we want the caller to hear, how much wait the users can tolerate, and how many agents we need online at a particular time. Our spreadsheet for call queues will look like the following table.

Queue Name	**Queue #**	**Password**	**Max Wait Time**	**Ring Stategy**	**Wrap Time**	**Annc. Position**	**Annc Hold Time**	**Join Msg**	**Static Agents**
Sales	2500	232	90	Ring All	10	Yes	Yes	sales_ queue	200, 210
Support	3500	222	120	Round Robin	10	Yes	Yes	support_ queue	300, 305

Agents

Agents are the extensions that are used within a call queue to determine where the next call will be sent to. With call queues, agents can be either statically set by listing the extensions when setting up the queue, or they can be dynamic, which means anyone can log into or out of the queue to determine when calls are sent to them.

Connectivity

Once we know how many users we will have, we can try to factor what our typical outbound calling ratio will be, and then we need to estimate the number of peak inbound calls. Once we have a feel for the number of total possible concurrent calls, we then need to figure out what kind of phone circuits we will need to provide enough service.

Typically, if we are going to have eight calls or less, then it is usually cheaper to use analog lines than moving to digital lines such as PRI connections.

Before considering moving to a pure VoIP solution, be sure to have someone fully check out your network, firewall, Internet connection, and route to the selected service provider to make sure the quality is up to the standards you are looking for.

PSTN

The most common connection to the PSTN (Public Switched Telephone Network) is a POTS (Plain Old Telephone Service) line, which is the typical analog phone line coming into your house or business that you can plug a regular phone into to get service. Small businesses may have a handful of individual POTS lines coming into their office.

When using POTS lines, we need to have some way of getting the phone line into the system. Whether using an interface card or SIP gateway device, we need to have FXO ports available.

PCI interface cards are available from companies such as Digium, Rhino, Sangoma, PIKA, and several others, and range from 2 – 48 ports. SIP gateways are available from other companies and range from 1 – 24 ports.

With T1 PRI interfaces, you can get cards with 1 – 4 ports allowing up to 96 concurrent calls. In Europe, this would be an E1 which would get you up to 128 channels.

VoIP

While Asterisk typically uses Voice over IP between the PBX and the phones, that doesn't mean the primary method of getting a dialtone should always be through a VoIP service provider. For many companies, the reliability of the existing phone system circuits outweighs any possible cost savings.

VoIP connectivity is among the many things that makes Asterisk such a compelling solution. By using Internet services, some companies can realize substantial cost savings over regular PSTN lines.

Using VoIP, phone calls can be placed over a broadband connection using Internet Telephone Service Providers (ITSPs). These ITSPs connect our phone system to the traditional phone circuits. For the most part, ITSPs are the most economical telephone service available. Pricing for VoIP service can be anywhere from ½ cent to 2 cents per minute, sometimes without monthly fees and usually with cheap long distance calls.

The bandwidth usage of an ITSP will vary dramatically based upon the codec used between the ISP and your phone system. The following chart outlines the most common codecs and their typical bandwidth usage:

Codec	Single Call	Two Calls	Additional Calls	Calls per megabit
G.711 (ulaw/alaw)	81.1 kbps	158 kbps	65.0 kbps	15
iLBC	28 kbps	49.3 kbps	21.2 kbps	47
G.729	30 kbps	39.7 kbps	9.6 kbps	103
GSM	35.4 kbps	50.2 kbps	14.7 kbps	68

While there are other codecs available, these are the most common to Asterisk and all but G.729 are included in a default installation. To use G.729, you will need to purchase a license from Digium for each channel at a cost of $10 per channel.

With a standard T1 PRI circuit you get 23 channels for voice communication, while a full T1 (1.5mbs) will max out around 20 calls using G.711 because of the extra overhead of the TCP/IP protocol and SIP or IAX headers. In comparison we can get around 150 calls on the same T1 circuit using G.729. While you may think that it is best just to use the best compression available to maximize your circuits, using compression will affect call quality and will put a much bigger resource load on your system. Running 150 channels of G.729 will bring all but the top-end computers to their knees, although if you are using G.729 throughout the entire system, then very little, if any, transcoding will take place. If you are going to plan a deployment like this, thorough testing and evaluation will be required for a successful deployment.

As you can see, the local network bandwidth usually isn't even a consideration on most small-medium sized installations. On larger installations, isolating network traffic onto its own network or by using VLANS may be warranted. Many small office DSL circuits are just not up to the task since although they may have fast download speeds, the upload speed is often too slow.

ITSP connectivity

There are literally hundreds of ITSPs around the world these days offering countless different calling plans, features, services, and prices. The best recommendation is to do some research in forums and by getting referrals in your area to find an ITSP that has good service and fits the needs of your business.

If you are planning on using regular PSTN circuits for your primary lines, you may still want to consider an ITSP for capacity overflow, specific long distance calling, or failover. Some companies find that by using an ITSP for capacity overflow they can reduce the number of PSTN lines thus realizing a cost saving from avoiding paying for additional lines or a reduction in long distance fees.

Some ITSPs offer only inbound calling, some offer only outbound calling, while most offer both inbound and outbound. Using an ITSP can also be an inexpensive way of adding additional DID numbers or even toll-free numbers for your customers to call.

If we are going to plan on using a large number of channels, then we also need to take into account which codecs are supported by the ITSP. If we want to utilize a lot of channels and want to maximize our bandwidth, then we may want to consider an ITSP that supports G.729 and then purchasing enough licenses to put on our PBX. If you do not need to do any transcoding, as mentioned earlier, then you will not need to purchase any licenses.

Primary circuit?

The big debate is about whether or not you should use an ITSP as a primary business circuit. This is a decision that you need to think through very carefully. My normal recommendation is that it is perfectly reasonable to use VoIP as a primary business line if the VoIP service is being provided over a dedicated Internet connection to the ITSP. Some companies offer hybrid services that combine data and voice service. These services allocate more bandwidth when less phone calls are in use, and reduce bandwidth when more voice channels are needed. When there is only one hop to the ITSP that is terminating your call to the PSTN, your calls never go out over the public Internet thus dramatically reducing issues with QoS that can affect call quality.

In a recent survey on trixbox.org, the results showed that 42% of new installations were using VoIP trunks as primary business circuits. This is certainly showing that the acceptance levels, along with quality and reliability, are growing rapidly. Before committing to a VoIP-only solution, test the connections and call quality during peak times to ensure that you will have a minimum number of issues.

DID numbers (Direct Inward Dial)

A DID number is simply the phone number that is dialled in order to make your phone ring. An analog or POTS number will usually only have a single DID assigned to it (depending on services provided by your telco), while PRI lines can have numerous numbers assigned to them. VoIP trunks can also have any number of DIDs assigned to them. When considering DID numbers, we should determine if we need more than the primary phone number, and whether we want direct phone numbers for different services like conference rooms, different departments, or even individual users.

When planning our DID requirements, we should take into account the following:

- The name we will give to the trunk (usually the name of the provider and a number if we have more than one trunk from the same provider)
- The technology we will use for the trunk (which protocol such as IAX, SIP, PRI)
- How many channels this particular trunk provides
- The order in which these trunks will be used for outbound calls

The following table shows the spreadsheet section of our worksheet for deploying our trunks:

Trunk Name	Tech	Channels	DID	Outbound Order
SBC T1	PRI	24	949-555-1000 – 949-555-1200	2
Vitelity	SIP	4	949-555-5000	1
PSTN1	Analog	1	949-555-1234	3/911

Telephones

Certainly the choice of phones is going to be one of the most important decisions you will make regarding your phone system. Since the phone is the primary interface to the entire system, care should be taken when choosing the devices that you will use. The cost of phones will vary greatly from the low end to the high end, with the primary differences being the call quality, echo suppression, speakerphone quality, display type, and other features.

Hard phones

A hard phone is a physical device that works like any other physical phone. The most common phones to use with trixbox are SIP-compliant phones; these are available from companies such as Aastra, Cisco, Polycom, Linksys, Grandstream, SNOM, and several others. These phones work well with trixbox and have business features like multiple lines, call transfer buttons, conference buttons, voicemail buttons, message waiting indicator lights, and other typical business phone features.

Along with the standard desk-type phones, new wireless phones are becoming more widely available using different connection methods such as WiFi and DECT.

Using **Analog Telephone Adapters** (**ATAs**) and channel banks, you can also use a regular analog phone to connect to your phone system. While this will work well, you will not have all of the features that are available in a SIP phone.

Softphones

A softphone is a software phone that runs on your computer. This will emulate all of the functionality of a regular phone but works as a software application and will use the microphone and speaker on your computer as the handset. For best results, it is recommended to use a USB or Bluetooth mic/headset as these provide the best audio quality.

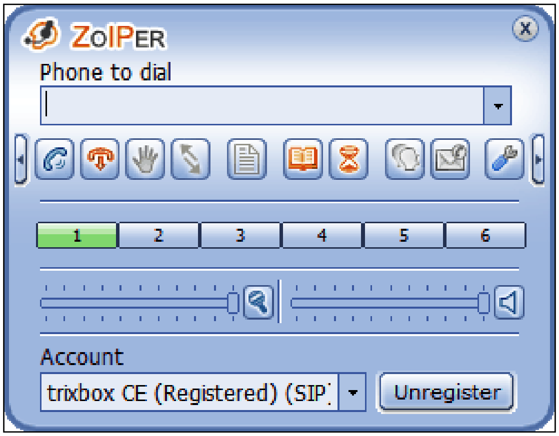
There are a number of softphones available, with the two most common being shown next; in the next chapter we will go into detail about how to configure these softphones.

Counterpath X-Lite

Counterpath's X-Lite (`http://www.counterpath.com`) is a very popular softphone as it has a traditional phone look and feel.

Zoiper

Zoiper (`http://zoiper.com`), formerly known as IDEFISK, is one of the most popular softphones available. Users like it because of its small screen real estate, and because it supports both SIP and IAX extensions.



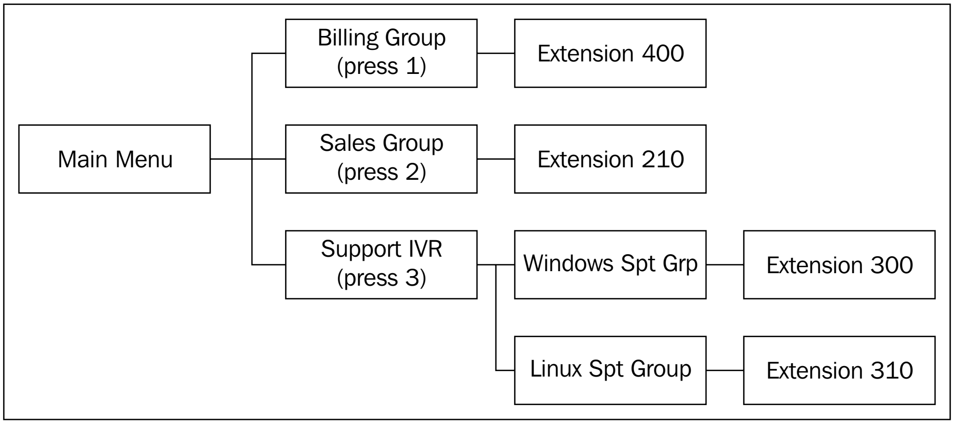
IVR menu

The IVR (Interactive Voice Response) acts as a digital receptionist for your phone system. A well-designed IVR menu tree is one of the key features of a successful installation. An example of an IVR menu would go like this:

Thank you for calling American Widgets; if you know your party's extension, you may dial it anytime. For sales, press 1; for billing, press 2; for a company directory, press the pound key.

The PBX Configuration tool system inside of trixbox CE allows you to easily build complex, multi-branching voice menus to route callers (based on a valid keypress) to appropriate destinations such as extensions, queues, ring groups, or another IVR menu.

Designing our IVR menus in advance will not only allow for you and the client to work out exactly how the system will work, it will also be your roadmap for the actual configuration of the system. Again, a well designed flowchart of the IVR system will save you lots of programming time as well as make it much easier to explain to people how the system works should they want to make changes later. The following diagram shows a typical IVR menu:



It can be very frustrating for callers to get a poorly designed IVR system. The IVR system is the first interaction a caller has with a company and a poor experience with your phone system can leave a lasting negative impression. I am sure that many readers have called into a company and sat there holding up fingers trying to decide what the best menu option for them would be. If that ever happens, then you have a fairly poor IVR design.

The following are some general rules to keep in mind when designing your IVR menus:

- **Keep it simple**: Some experts say that humans remember things better in groups of three. Bearing that in mind we should keep any level of our menus to only three items or less. It is far better to break things down into simpler menus with submenus than to have a smaller tree but have eight or nine menu options.
- **Don't play 'Find the human'**: There are bound to be times when even the best designed IVR system will fail to properly direct a caller to who they are looking for. You should always make it fairly simple to direct the caller to a live person. If you have a closed system with no obvious path to get to what you are looking for, some callers will resort to trying random extensions to try to reach someone that can then transfer them to someone that can help them.
- **Don't ask for useless information**: As much as I hate typing security codes, social security codes, pin numbers, and so on into an IVR system, the thing I hate worse is typing all that in and then being asked for it multiple times or being asked again by the person who takes the call. If you aren't using the information properly, don't ask for it.

- **Let them go back**: Humans are calling your system and yes, humans do often make mistakes. If someone pushes the wrong key and gets to the wrong submenu, make sure they have a way of moving back to the previous menu. Few things are as frustrating as being forced to hang up and call back because you got stuck in an IVR somewhere.
- **Have good system recordings**: Even the best laid-out IVR system can become a source of embarrassment if the call recordings do not sound good. If you don't have someone willing and capable of doing clear, professional sound recordings, consider using a professional service. Alison Smith, who is the voice of all the built-in prompts, is even available for hire at `http://theivrvoice.com`.

When planning out our IVR, we should note each entry with the following information:

- The name of the menu
- The selection number (keypress) for that command
- The destination for each command (ring group, call queue, extension, and so on)

While using a program like Visio is the preferred way to layout an IVR menu, we can still do this in our worksheet using the following table:

IVR Name	Keypress	Destination
Main	1	Sales Ring Group (600)
Main	2	Finance (ext 410)
Main	3	Support (IVR Support)
Support	1	Windows Support Group (610)
Support	2	Linux Support Group (620)

Fax requirements

Regardless of whether we want it or not, faxing is still a requirement for many businesses. In the chapter *PBX settings in detail*, we will go in much more depth on the different options available for faxing. At this time, inbound faxing and routing PDF files to email boxes works pretty well over analog and PRI circuits and outbound faxing is available with additional add-on tools.

Case studies

Let's take a look at a few examples of typical business situations and see how we would go about planning their deployments.

American Widgets Consulting Services

AWCS is a full-service IT consulting business in southern California. With a small office and several technicians out in the field all day, communication is a key requirement to keep in touch with the technicians and to provide service in the office. One big item on the to-do list was to find ways to cut back on the enormous costs of cell phones. Sales people wanted to ensure that they would get calls to their desks and to their cell phones and managers wanted to stay in touch by having remote phones at home.

trixbox CE is an ideal solution for this company as all of its requirements can easily be satisfied. The typical number of concurrent calls is fairly small, so it would like to use a few POTS lines for most of its calls and use a VoIP provider for capacity overflow.

The following are AWCS's requirements:

Basic Design:

- **Extensions**: Three digits.
- **Ring groups**: Sales could use a ring group to help make sure calls get to a salesperson quickly. Since there are always more sales people available than inbound calls, a ring group is an appropriate choice.
- **Call queues**: Since there is no problem with more callers than agents, there is no need to use call queues.
- **Sales requirement**: Using the Find-Me/Follow-Me functionality, calls to sales people can be made to ring both their desk and their cell phones at the same time.
- **Management requirement**: Since SIP phones work over UDP, placing remote phones at people's homes is fairly simple and should work without any real changes. Some firewalls may require you to do some basic port forwarding for the phone to receive calls.

Connectivity:

- An internal PCI card with 4 FXO ports will handle the majority of call volume. Outbound calls can use the POTS lines and then use the VoIP provider for extra capacity.
- **VoIP**: SIP trunk to ITSP.

Telephones:

- **Hard phones**: One per desk, one for each home user
- **Softphones**: Installed on tech's laptops

IVR:

- Simple menu system to go to sales and support groups

FAX:

- Single POTS line connected to FAX machine

Acme Collections Company

ACC is a medium-sized company with a single call center, which employs about 50 people with 30 sales agents on the phones making and receiving a large volume of calls at the same time. It also has a few small departments for HR, IT, and Accounting.

Their requirements could be met as follows:

Basic Design:

- **Extensions**: Three digits
- **Ring groups**: The IT and Accounting departments don't get a lot of calls at any one time, so ring groups for these departments will work well
- **Call queues**: The sales department sometimes gets overwhelmed with calls, so setting up a call queue for inbound calls makes the most sense

Connectivity:

- An internal PCI card with a single span PRI card will handle all of the company's call volume.

Telephones:

- **Hard phones**: One per desk, one for each home user
- **Softphones**: Installed on tech's laptops

IVR:

- Simple menu system to go to sales queue, HR manager, IT, and Accounting ring groups

FAX

- Sales extensions are configured to send inbound faxes sent to their DIDs to their email accounts

Summary

In this chapter, we have dug into the design and deployment plans needed for a successful deployment. We also took a look at gathering requirements by looking at a couple of case studies. We have gone through the basic building blocks of most typical systems and learned when to use different features as well as how to design a good IVR menu.

6
Hardware configurations

If you are going to install business systems, you will need to learn about the different hardware interfaces and devices that you will be using. From phones, to TDM cards, to gateways, there is a plethora of different devices that you will end up working with. This chapter will cover the most common devices and walk you through all the steps to configure each one.

TDM cards

The most common way to connect to analog or digital phone circuits is to use an internal PCI interface card. The most common manufacturers of TDM cards include:

- Digium
- Sangoma
- Rhino

One of the goals of the trixbox CE development team is to create an automatic setup tool for all different manufacturers' cards; while as of this writing that is not complete yet, it is scheduled for spring of 2009.

Digium cards

Since the Zaptel driver is written by Digium, all of Digium's cards are supported by default within trixbox CE making them very easy to get up and running. In fact, if you have Digium cards installed in your system, during the install process they will be automatically detected and configured for you.

If you need to set up a card after you have already installed trixbox CE and then need to configure a Digium TDM card, shut down the system, install the card, boot the system back up, log in, and then run the following command:

```
setup-pstn
```

This command will shut down Asterisk, do a hardware detection, and set up the appropriate configuration files. When it completes, you should get something like the following output:

```
Chan Extension  Context         Language    MOH Interpret
pseudo          default         en          default
1               from-pstn       en          default
2               from-pstn       en          default
3               from-pstn       en          default
4               from-pstn       en          default
```

At this point the card will be configured and Asterisk will be seeing the cards. Since the PBX Settings tool has a default Zap trunk set up and outbound route, you should be ready to make calls by dialing 9 + the phone number you want to dial. Before you can get incoming calls you will need to set up an inbound route.

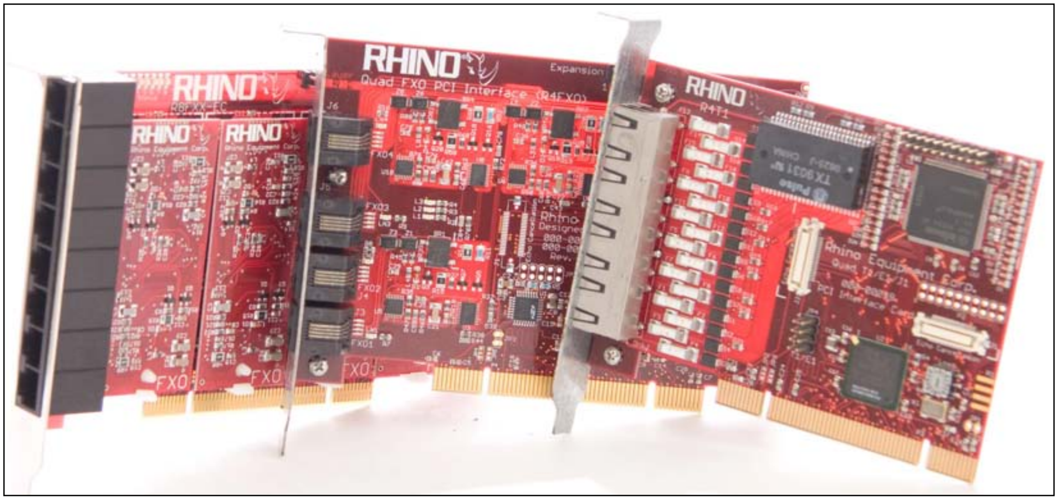
Rhino Equipment cards / channel banks

One of my favorite card manufacturers is *Rhino Equipment*. I have been using its cards for several years. At the time of this writing, trixbox CE doesn't auto-detect Rhino cards although this will be coming in an upcoming release. A big advantage of Rhino cards is that all of the analog cards from Rhino come with on-board hardware echo cancellation—this makes for far less messing around with tweaking settings to solve echo problems—just use these cards and the echo cancellation is automatic.

Rhino also makes a selection of channel banks; these are devices that take multiple FXO or FXS ports into them and then connect to your phone system using a PRI connection. These are most often used when a company may want to switch to a PRI later but has a number of analog phone lines in use currently. When it's time to switch to a PRI, you simply unplug the channel bank, plug in the cable for the PRI and you are back up and running.

Another use of channel banks is for connecting analog phones. Each channel bank can connect up to 24 analog phones as extensions. This is often used in places like hotels, or for connecting FAX machines.

At the time of this writing, Rhino offers a full line of cards from 4-port FXO, to 8-and 24-port modular FXO/FXS cards, as well as Single, Dual, and Quad port T1/E1 cards. The channel banks come in either FXO or FXS configurations or a modular version that allows you to use its modules to determine what port types to use.



Rhino PCI cards

A script has been developed to make configuring a Rhino card quite simple. From the command line, simply run `setup-rhino` and the script will figure out the correct driver to install, download it, and install it. After the script is finished, reboot your trixbox system, log back in, and run `setup-pstn`.

Rhino channel banks

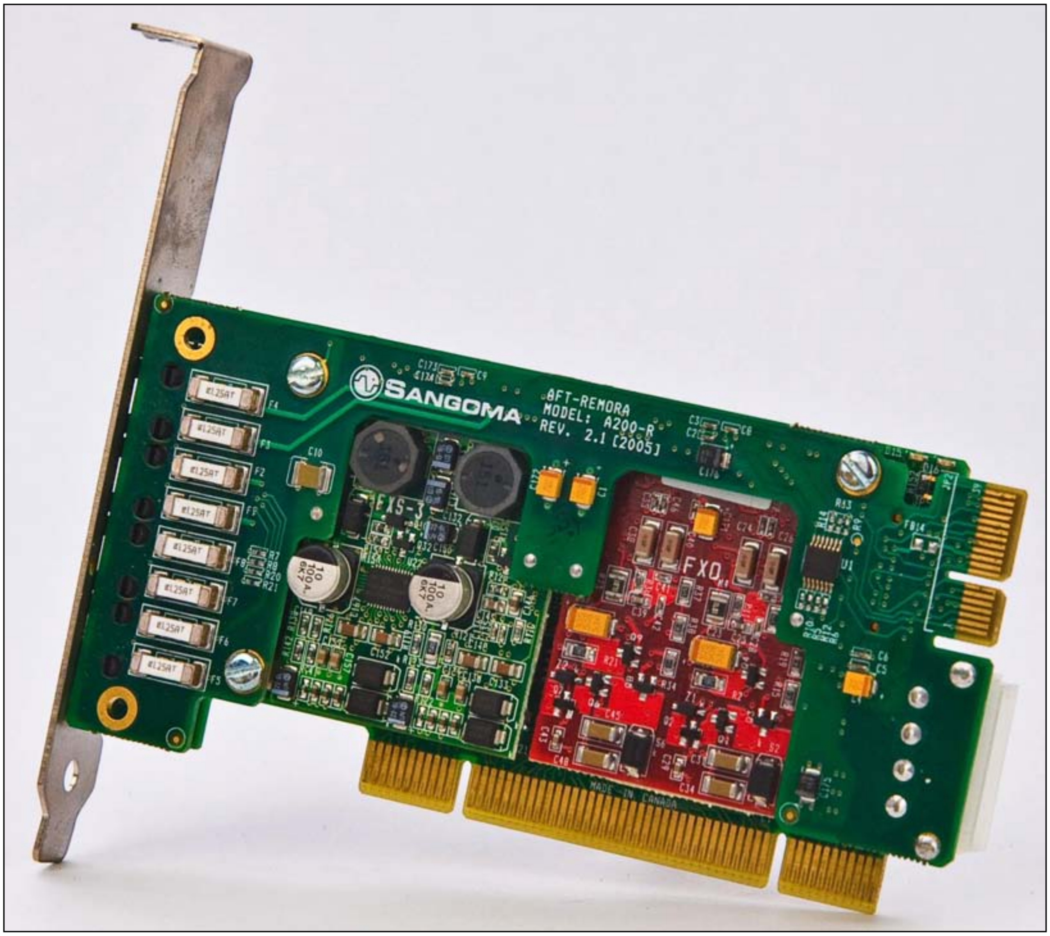
The channel banks connect to analog lines or phones either through a standard Amphenol cable or via an RJ-11 breakout box; then they connect to your trixbox system via an RJ-45 connector to a PRI card in your trixbox system. To your trixbox system, the device simply acts like a T1 PRI.

In most cases, the Rhino channel banks are very simple to install as well. From the keypad, select the Auto-T1 function, which will configure the connection to your trixbox system and you should be ready to go.

Sangoma cards

Another very popular option is the cards from Sangoma. The Sangoma analog cards use 2-port FXO or FXS modules. Sangoma also has a full line of T1/E1 cards as well. An interesting feature of the Sangoma analog cards is that it allows you to add daughter boards to the basic card configuration to expand the number of ports available. With the 4-port A200 card, you can start with only 2 ports and expand up to 24 ports. With the A400 card, you can start with only 2 ports and expand up to 48 ports.

Sangoma's digital line includes single, dual, quad, and even an 8-port T1/E1 card, and it has recently released a 2 – 24-port ISDN BRI card.



To install a Sangoma card into your trixbox system is fairly simple. From the trixbox CE Package Manager you can install the wanpipe-modules and wanpipe-util, or go to the command line and run the following command:

```
#yum install wanpipe*
```

After the drivers are installed, reboot your trixbox CE system, log back into the system, and run the following command:

```
#setup-sangoma
```

The script will walk you through the configuration of the cards.

Manual configurations for TDM cards

While the scripts we have seen will normally get everything working, there are times when you just have to go in and set things up manually. Some of the reasons you will want to manually edit files are:

- Custom channel grouping
- Using less ports than available on the card—you only want to use 2 spans on a 4-span PRI card
- Adding custom settings specific to your telephone company
- Setting up the system as a media gateway

For these reasons, as well as general troubleshooting, you should understand the files that are used to configure these cards. The files you will want to be familiar with are as follows:

- `/etc/zaptel.conf`: This file defines the spans and ports that are used on each card
- `/etc/asterisk/zapata.conf`: This file defines the groups, signaling, and channel-specific settings
- `/etc/asterisk/zapata-channels.conf`: This file is automatically generated from the `setup-pstn` script

Zaptel.conf

The first file that's used is `zaptel.conf`—each card and span will need to be defined. There are two basic settings we will see in this file:

- Analog line settings—used to define the ports for analog lines
- PRI settings—used to define T1/E1 ports

Analog line settings

Since we can have both analog lines and analog extensions, we would set either FXO or FXS ports depending on the modules that we have installed. The following is an example of a card with 2 FXO ports and 2 FXS ports:

```
#Zaptel Channels Configurations (zaptel.conf)
#
loadzone=us
defaultzone=us
#Sangoma A200 [slot:4 bus:2 span:1]
```

```
fxsks=1
fxsks=2
fxoks=3
fxoks=4
```

The important thing to note here is that the first two ports are FXO (phone line) ports and the second two are the FXS (analog phone) ports. However, in the file it appears to be the opposite; the reason is that our phone system has to provide proper signaling to the port, which is the opposite of what the port is called.

Digital line settings

A PRI line is configured quite differently from an analog line as multiple channels come in over a single connection. A PRI in US will have 1 D channel for signaling and 23 B voice channels. In other parts of the world, it may be an E1 with 32 voice channels. The other options on the span setting are dependent on the settings from your circuit provider. A typical US PRI setting is shown below:

```
#Rhino R1T1
span=1,1,0,esf,b8zs

bchan=1-23

dchan=24
```

Zapata.conf

The `zapata.conf` file contains additional settings and channel groupings for each line. Let's dissect the file and look at the different settings that are used in the file.

A typical file will look like the following example. In this example, no lines are set up yet; we will look at the line configurations separately.

```
[channels]
language=en
context=from-zaptel
rxwink=300
usecallerid=yes
hidecallerid=no
callwaiting=yes
usecallingpres=yes
callwaitingcallerid=yes
threewaycalling=yes
transfer=yes
cancallforward=yes
callreturn=yes
echocancel=yes
echocancelwhenbridged=no
;echotraining=800
```

```
rxgain=0.0
txgain=0.0
group=0
callgroup=1
pickupgroup=1
immediate=no
faxdetect=incoming
```

- **Language**: The language setting defines the default language for what version of prerecorded sounds Asterisk will use for this channel.
- **Context**: This defines which section of the dialplan the call is sent to.
- **RXWink**: Sets the receive wink timing. This will usually never need to be changed.
- **Usercallerid**: Determines if caller ID is going to be used on this channel.
- **Hidecallerid**: On PRI lines, this setting will determine whether or not to send caller ID on outbound calls.
- **Callwaiting**: If enabled, Asterisk will generate 'call waiting pips' when you are already in a conversation on your FXS handset and someone tries to call you. If the channel has call waiting by default, you can temporarily disable it by lifting the handset and dialing *70, whereupon you will get a dialrecall tone and may then dial the intended number. There is no corresponding way to temporarily enable call waiting for channels that have it off by default.
- **UseCallingPres**: On PRI lines, this determines whether or not to use the Caller ID presentation for the outgoing call that the calling switch is sending.
- **Callwaitingcallerid**: Sets whether Asterisk will send Caller ID data to the handset during call waiting indication. Also requires setting `callwaiting=yes`.
- **Threewaycalling**: If enabled, you can place a call on hold by pressing a hook flash, whereupon you get a dialrecall tone and can make another call.
- **Transfer**: This option has effect only when `threewaycalling=yes`. If `threewaycalling=yes` and `transfer=yes`, then once you've placed a call on hold with a hook flash, you can transfer that call to another extension by dialing the extension and hanging up.
- **Cancallforward**: If this is set then you can use 'call forward immediate' by dialing *72 followed by the number you want to forward the call to. You can disable call forwarding with *73.
- **Callreturn**: If this is enabled then you can dial *69 to have Asterisk return the call from the last number that called you.

- **Echocancel**: Disable or enable echo cancellation. If you don't have hardware echo cancellation then this is one of the settings you may need to tweak in order to solve echo problems. The default setting of `yes` is equal to 128 taps. You can use a number such as 16, 32, 64, 128, or 256. Each tap is one sample from the data stream, so on a T1 this will be 1/8000 of a second.
- **Echocancelwhenbridged**: Typically, bridged TDM calls do not require echo cancellation but sometimes audio performance is improved with this option turned on.
- **Echotraining**: This setting adjusts the time used for the echo canceller to train itself. In current versions of trixbox CE, this setting is commented out because it will cause audio problems with the OSLEC echo cancellation software that is included with trixbox CE **2.4** and above.
- **RXGain / TXGain**: These settings adjust the gain of the audio for the Receive and Transmit sides of the call. The gain is specified as a number from -100 to 100 representing dB. You will typically never go above or below -10 thru 10. These settings are also used to adjust volumes to help solve echo problems.
- **Group**: This setting allows you to group together a set of channels. This can be used to create different trunk groups for different dialing routes.
- **CallGroup**: Each channel can belong to zero or more callgroups. A callgroup can specify who can answer the phone when it is ringing. If a channel is ringing, people in the same callgroup can answer the call by dialing *8#.
- **Pickupgroup**: A channel can also belong to one or more pickupgroups. Pickupgroups specifiy whose phones you may answer.
- **Immediate**: When set to `no`, this will provide an FXS extension with a dialtone and will listen to keypresess until a pattern is matched before dialing. If this is set to `yes` then Asterisk will start executing the commands for this channel's 's' extension. This can be used for a lobby phone or 'batphone' mode.
- **Faxdetect**: This determines if fax detection should be used on incoming calls, outgoing calls, or both.

Channel configuration

The other settings used in `zapata.conf` (or in `zapata-channels.conf` if `setup-pstn` is used) are for configuring the different channels.

Analog card configuration

The following shows a basic analog card configuration:

```
context=from-zaptel
group=0
signalling = fxs_ks
channel => 1-2
```

In this case, the signaling is FXS kewlstart, which is the standard for FXO ports and the channel setting shows this has two ports on it.

Digital card configuration

The following shows a basic analog card configuration:

```
switchtype=national
context=from-pstn
signalling=pri_cpe
group=1
channel => 1-23
```

The settings for a PRI are quite a bit different from analog lines. The `switchtype` is determined from the type of circuit that is provided by the phone company and the signaling is `pri_cpe` standing for PRI/Customer Premise Equipment.

Configuring endpoints

To configure an endpoint there are three main pieces of information required for a phone to register with Asterisk:

- IP address of proxy server (trixbox CE server)
- Username
- Device Password

Each device does things a little differently and calls different fields by different names. Learning the nuances of each device will allow you to figure out how to configure any type of device.

In this section we will go through setting up phones manually. We can also use the trixbox CE Endpoint Manager to configure common phones; we will look at how to use the Endpoint Manager later on.

Aastra phones

Aastra phones are fairly simple to get up and running. The easiest way to set up an Aastra phone is from the web interface of the phone. The first step in setting up an Aastra phone is to get the IP address of the phone. While there are slight differences between the different models, they are similar enough that we can use one model as our example. In this case, we will use the Aastra 480i as our example phone. To find the IP address of the phone, use the following instructions:

1. Press the *Options* key. (This is a button labeled with the Aastra **A** logo on older models.)
2. Scroll down to **Network**.
3. Press the *Show* button.
4. Enter **22222**.
5. Scroll down to **IP Address**.
6. Press the *Show* button

When you are ready to configure the phone, browse to the IP address that you found from a web browser. When prompted, enter the following information:

Login: **admin**

Password: **22222**

In the left-hand menu, select the **Global SIP** link. In this section, we will just set the IP address or host name that our phone will point to in order to register. All you need to do here is change the **Proxy Server** and **Registrar Server** to point to your trixbox CE system.

Basic SIP Network Settings	
Proxy Server	192.168.5.49
Proxy Port	0
Backup Proxy Server	0.0.0.0
Backup Proxy Port	0
Outbound Proxy Server	0.0.0.0
Outbound Proxy Port	0
Registrar Server	192.168.5.49
Registrar Port	0
Backup Registrar Server	0.0.0.0
Backup Registrar Port	0
Registration Period	0

After you have the first set of settings finished, click on the **Line 1** link in the left-hand navigation; this is where we will configure the account settings. The **Screen Name**, **Phone Number**, **Caller ID**, and **Authentication Name** will all be the extension number we set up in the PBX Settings tool. The Password is **Secret** that we set for the extension. After you have saved your settings, click on the **Reset** link in the left-hand navigation to restart the phone.

Basic SIP Authentication Settings	
Screen Name	200
Phone Number	200
Caller ID	200
Authentication Name	200
Password	•••••
BLA Number	
Line Mode	Generic

As long as you used the correct information for the extension that you created in the PBX Settings tool, the phone should register to the phone system. If you are attempting to connect to a system that is not on the same subnet as the phone system, such as a remote phone, then you may have to figure out issues with port forwarding or network routing. Because it can be more difficult to set up a remote phone, it is recommended that you set up phones on the same network as the phone system first, to make sure you understand the concepts and methods used to configure the devices.

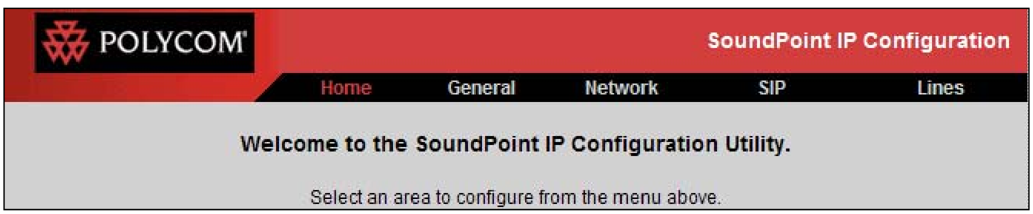
Polycom phones

Polycom has been around for a long time and is a well respected brand name for business users. Known for their call quality, the phones are actually fairly difficult to set up properly. While we can do a basic configuration through the web interface on the phone, we cannot configure all of the settings without using configuration files that get pulled from the server. In this chapter, we are only doing basic configurations and will look at the configuration files in the *PBX settings in detail* chapter.

To get started with the Polycom phones, we first need to figure out the IP address that the phone is on. The following steps on the phone will get you the IP address that the phone was given by your DHCP server:

1. Press the **Menu** key.
2. Press **2** for the Status menu.
3. Press **2** for the Network menu.
4. Press **1** for TCP/IP parameters.

Our phone was configured as 192.168.5.195 so we can open a web browser and point to `http://192.168.5.195`. The first thing we will see will be the main menu for the phone.



To configure a basic extension, we need to click on the **Lines** option. When you do it, a dialog box will appear asking for a login and password; use the following credentials to login:

Login: **Polycom** (note: this is case sensitive)

Password: **456**

There are three sections on this page we need to configure:

- Identification
- Server
- Message Center

Identification

This section specifies the login credentials for the device to the phone system; use the following guide to fill out the fields:

Display Name: Extension Number

Address: Extension Number

Auth User ID: Extension Number

Auth Password: Extension Secret

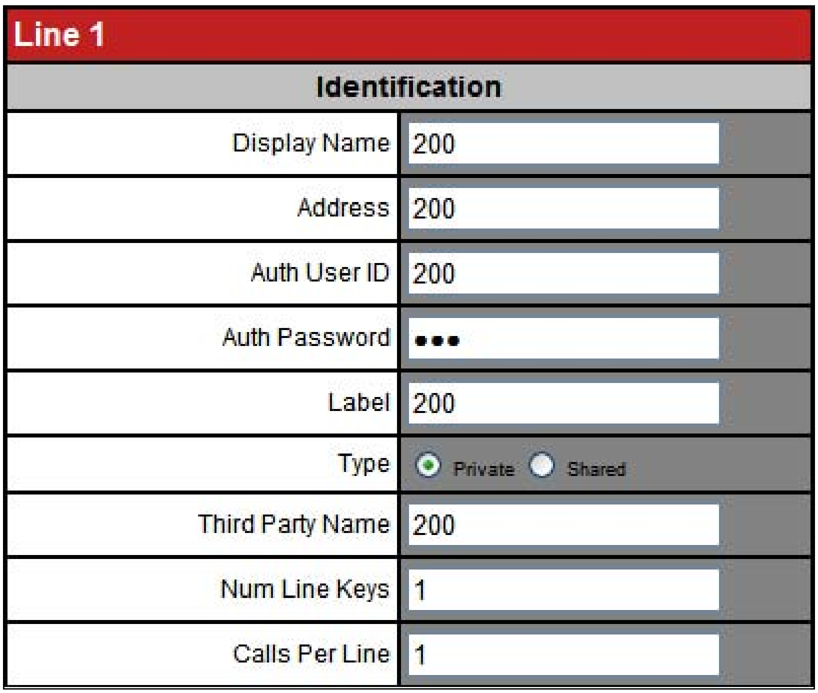
Label: Extension Number

Type: **Private**

Third Party Name: Extension Number

Num Line Keys: **1**

Calls Per Line: **1**



Server information

The server information tells the phone how to communicate with the trixbox system. Use the following guide for configuring this section:

Address: IP Address of trixbox system

Port: **5060**

Transport: **UDPOnly**

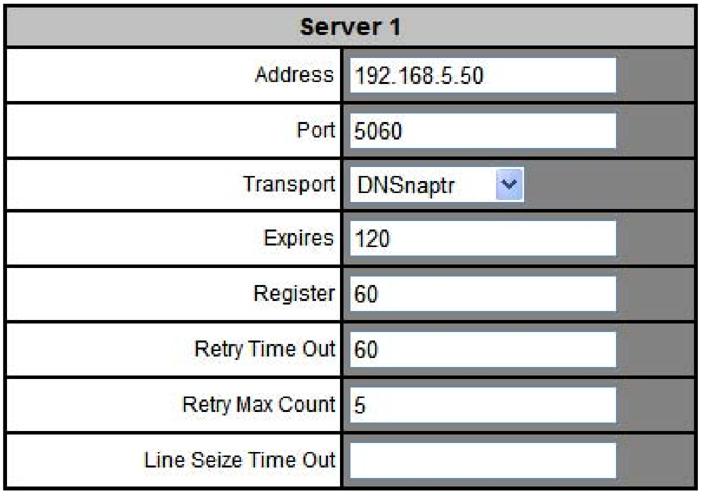
Expires: **120**

Register: **60**

Retry Time Out: **60**

Retry Max Count: **5**

Line Seize Time Out: <blank>



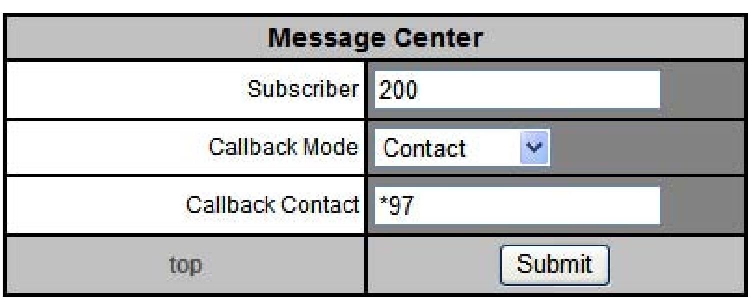
Message Center

The message center sets up the Message Waiting Indicator to point to the correct mailbox and configures the voicemail button on the phone to connect to the trixbox voicemail system. Use the following guide to configure the Message Center:

Subscriber: Extension Number

Callback Mode: **Contact**

Callback Contact: *97



When finished, press the **Submit** button and the phone will reboot with the new settings applied.

Other devices

Seeing how several different phones are configured, and knowing the three basic pieces of information that any device will need to connect to the system, figuring out how to configure any other device should be pretty straightforward. With an ATA, you may need to refer to the manual to figure out how to get the IP Address information in order to configure it.

Summary

In this chapter we have looked at configuring our basic connectivity options with different types of interface cards as well as configuring popular brands of phones. This will give you a basic understanding of how different devices and endpoints work, and gives you the basic information needed to get any compatible device to function properly.

7

Network considerations

Since a trixbox CE phone system runs over standard networking cables and uses regular TCP/IP protocols, we need to take certain things into consideration when designing our systems. Things to think about when designing the system:

- Do we want voice traffic and data traffic on the same network?
  - How do we want to separate traffic?
- Are we going to use VoIP providers?
  - Will firewalls and port forwarding be a problem?
  - Do we have adequate QoS?
- Will we have remote users?
  - Will NAT be an issue?
  - Is there enough bandwidth?
  - Do we have adequate QoS??
- Will all the phones be on the same subnet as the PBX?
  - Will we have routing issues?
  - Will we have NAT issues?

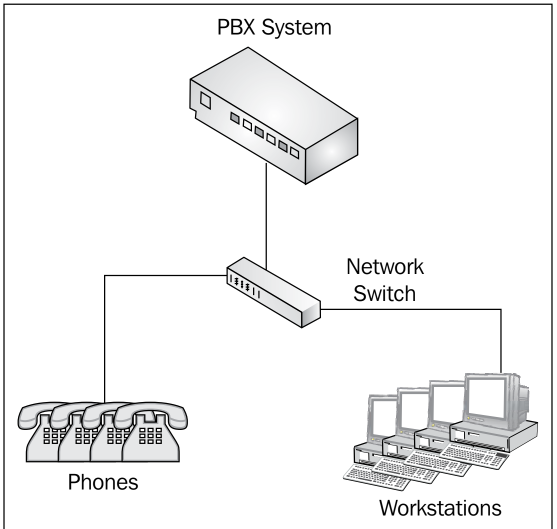
As we go through the different issues we will face, we will look at how to solve them to ensure that we don't run into major obstacles during the installation.

Combined or segmented network

The first thing to consider is if we want to use a combined network or a segmented network. In a combined network, the trixbox system will sit on the same network as any other server you have and will share the network infrastructure with all of the computers in the office. In a segmented network, we would put the phone system and all of the phones onto their own network in order to isolate the voice traffic from the data traffic.

Combined network

In a combined network, the voice traffic and data traffic go across the same Ethernet network.



Combined Network

To determine if a combined network is the right choice for our deployment, we need to understand the pros and cons of this type of architecture.

Pros

- No need for additional cabling
- Most phones have a pass-thru port, so only one network jack is needed
- In an existing office, you can use an existing network
- No need for additional network switches
- Less infrastructure (switches, UPSs)

Cons

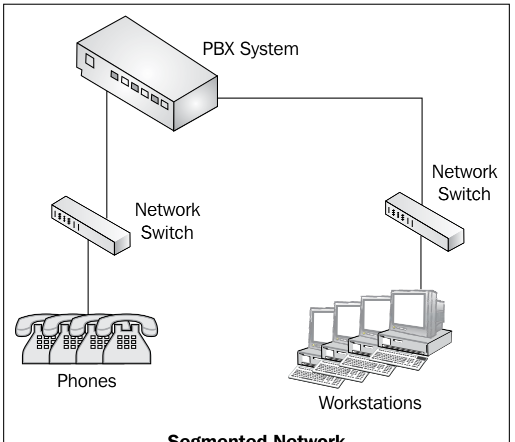
- Heavy traffic load may impact call quality
- Less security of data
- Possible interference with defective devices on the network

Since the only real downside to a combined network is that in a heavily congested network you could have call quality issues, let's take a moment to examine the possible issues we could have here. If we have a typical 100 megabit office network, and we refer back to our matrix of bandwidth usage for different codecs, we see that the G.711 codec will support 15 concurrent calls per megabit. If we have 100 users on our network, we will usually never go above 30% of the phones being used at once, so our bandwidth requirement would be around 2 megabits. Even if half of the phones were in use at one time, we are still only just over 3 megabits of usage on our 100 megabit connection.

Should you become concerned that the data traffic is impacting the voice quality, almost all of the phones on the market today support VLAN tagging. If you have network switches that support VLANs, then this is an effective way to isolate the phone traffic from the data traffic and still use a combined network. Configuring VLANs is an advanced networking concept and is outside of the scope of this book.

Segmented network

In a segmented network, we completely isolate the phones from the workstations by using completely different physical networks. This is accomplished by adding a second network card to the trixbox system so that one network card is connected to the primary network for maintenance and Internet connectivity, and the second network card is connected to a separate network connected to the phones.



Segmented Network

As we did with the combined network, let's take a look at the pros and cons of a segmented network.

Pros

- Completely isolated network
    - No conflict with data traffic
    - Network problems on the data network do not affect the phone network
    - Workstation users cannot snoop voice traffic
    - Data network can go offline without affecting phones
- Phone system can run a separate DHCP server for phone network

Cons

- Additional network is required
- Additional switches are required

While there are certainly some advantages to having a segmented network, the additional cost for adding more switches and network cabling usually makes this a less popular choice.

> For the sake of simplicity, this book will be based on using a combined network for all of our examples.

Network services

In any network there are some basic network services that are provided somewhere within the network infrastructure. The primary services that we are concerned about with regards to designing a phone system network are:

- DHCP Server
    - Provides IP addresses to devices
    - Can provide location of TFTP server for auto configuration
    - Can provide location of time server to devices
- DNS Server
    - Provides host name resolution for devices

- NTP Server
    - Provides a reliable time source to devices

In most cases, the DHCP and DNS services are provided by the network router or firewall, or sometimes by a primary server such as a Windows 2003 server. The main problem with most firewall/routers being used as the DHCP server is that most routers/firewalls do not have the functionality to set scope options, such as the boot server address, that will help in configuring our phones. Although not required, it is recommended to use the trixbox system to provide DHCP services to the network. This is easily done by adding the DHCP Manager module from the Package Manager.

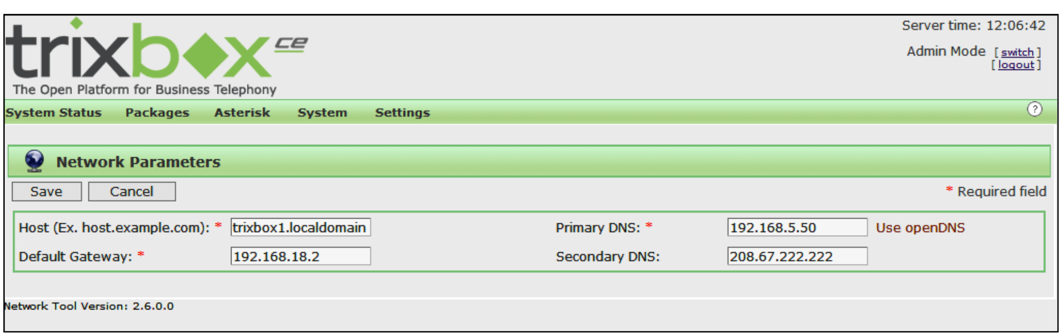
> Enabling the DHCP server on your trixbox CE system without disabling the existing DHCP server on your network will cause numerous network problems. Do not do this unless you have a good understanding of how to enable/disable the DHCP services on your existing device.

If you use your ISP's DNS servers for name resolution, it is recommended that you also run a DNS server. The reason we may want to do this is that, if there is a problem communicating with our ISP's DNS servers, we don't want the phone system to have a problem.

To set up the DNS server on the trixbox CE system, you can use the Package Manager to install the Bind DNS server, or from the command line run the following command:

```
#yum install bind
```

Once Bind is installed, you need to tell the system to use itself as a primary DNS server. Bind is automatically installed with trixbox CE versions **2.6** and above. To configure your system to use itself as the DNS server, from the trixbox CE **Admin** dashboard, go to the **System** menu and select the **Network** module. For the primary DNS server, enter the IP address of the trixbox CE system; for the secondary DNS server, use your ISP's DNS server or a third-party DNS service like OpenDNS.



An NTP server is also automatically configured on trixbox CE by default. If you use the trixbox CE DHCP manager, it will automatically configure your DHCP server for TFTP and NTP settings.

Remote users / VoIP services

Whenever you need to access a remote VoIP service such as a SIP provider, or if you are providing access to remote users, then you need to take this information into consideration. If we do not have our network settings configured properly within Asterisk, we can end up with several different issues the most notable being one-way audio.

In most cases, your trixbox CE system will be assigned a private IP address within your corporate network but will be communicating to the outside world through your network's public IP address. In order for everything to work right, we need to tell Asterisk more about our network configuration.

To set up the network settings, we need to edit the `/etc/asterisk/sip_nat.conf` file using the following command:

```
#nano /etc/asterisk/sip_nat.conf
```

After an installation, this file doesn't exist but is set up to be included once you create it. A typical file will look like the following example:

```
;sip_general_custom.conf
externip=10.10.18.48  ;Edit to match your external/public IP address
localnet=192.168.5.0/255.255.255.0 ;Edit to match your
                                internal/private IP address
nat=yes ;Lets the system know you are behind a NAT address
```

The `externip` setting needs to be set to the public IP address that is provided by your ISP. If you do not know what your public IP address is, you can use a web browser and go to `http://whatismyip.com`. If you do not have a static IP address and you are using an ITSP for your telephone service, then your service may be interrupted if your ISP changes your IP address. To avoid this problem, you can use a service like DynDNS.org. Most routers these days can be set to automatically update DynDNS.org when your IP address changes. You can then use `externhost` instead of `externip`.

```
externhost=pbx.dyndns.org
```

If you are using `externhost`, then Asterisk will do a DNS lookup every 10 minutes to make sure it always knows the correct IP address to use.

Port forwarding

With Asterisk ready to listen for traffic, we next need to make sure that the correct traffic is going to get to the system; for this, we need to go into our firewall and set up some port forwarding. Since every firewall does this differently, it is beyond the scope of this book to cover exactly how to set this up for different devices; refer to your firewall's manual to learn how to configure port forwarding. The following chart shows the different ports that need to be forwarding to your trixbox CE system.

Port Range	Type	Description
5060 - 5061	UDP	SIP Registration
10,000 – 20,000	UDP	RTP (voice transmission)
4569	UDP	IAX

QoS

Although technically this is ToS, most people refer to packet prioritization as QoS. QoS stands for **Quality of Service** and is a means of ensuring that voice packets are prioritized over other types of traffic over your network. When using a VoIP service provider, or when you have remote phones for telecommuters, QoS can be absolutely critical to having good quality voice communication.

It is quite common to experience call quality problems when using an ITSP or a remote phone if you are using a low-end router. Fortunately, some of the lower end routers now, such as the Linksys WRT54G, have QoS systems built-in that can set up and will work fairly well for a small system.

Larger systems may require more robust QoS solutions such as more expensive firewall/routers, or an actual QoS device, which is available from companies like *Kentrox* (`http://kentrox.com`).

One of the basic problems today is that you do not maintain QoS tagging across the open Internet, so once your voice packets are outside of your network, you cannot guarantee that your voice packets will have priority. For the most part, however, simply prioritizing voice packets from your network to your ISP, and at your remote locations to the ISP they connect to, will dramatically increase call quality.

Summary

This chapter has introduced the basic information you will need to configure your network for use with trixbox. We have looked at the pros and cons of segmented versus combined networks and have gone over the network considerations for using VoIP services and supporting remote users. We have also explored the NAT issues and learned about port forwarding and the settings that Asterisk needs to communicate properly. Finally, we learned how QoS can help to improve call quality when calls are going out over the Internet.

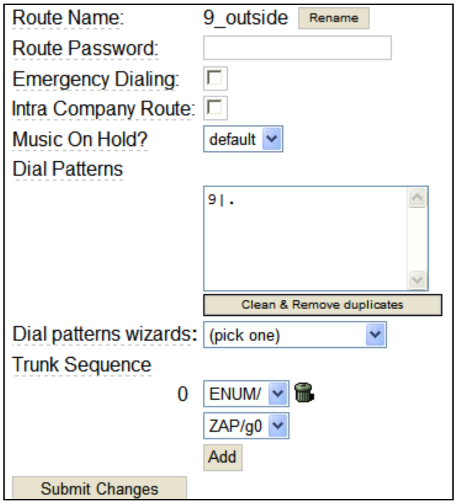
8
Advanced trunking

We saw in earlier chapters how to create a SIP trunk to a VoIP service provider and how to create a PRI and Analog trunk using TDM cards. This chapter is going to cover other trunk options as well as show how to connect two trixbox CE systems together and route calls between them.

ENUM

ENUM is a protocol that uses the Internet DNS system to translate E.164 (that is, standard phone number format) telephone numbers into IP addresses and allows Asterisk-based PBX systems to call each other directly over the Internet and bypass traditional telco circuits.

To set up trixbox CE to place calls using ENUM, you go to the **PBX** menu and select **PBX Settings**, then go to trunks, add an ENUM trunk, and save the changes. Next you go to the outbound routes and make the ENUM trunk the first trunk in the outbound sequence.



To receive calls over ENUM requires a little extra work. First, you need to go to `http://e164.org` and create an account. Secondly, you need to register your phone numbers. When you add your phone numbers, make sure your inbound route is set to go directly to an extension so that you can get the verification call and pin number you will need to verify your number.

If you have the ports forwarded as discussed in the previous chapter, then anyone else that is set up to make outbound calls via an ENUM trunk should be able to go directly to you.

What happens during an ENUM call? The system making the outbound call will do a query of the ENUM database; if there is no match then the call goes out on the next available trunk. If there is a match, the system will return the IP address of the remote system and the calling system will initiate a VoIP connection directly to the remote PBX system.

If you have business partners or friends that also have trixbox CE systems, then using ENUM will bypass your telephone service providers and the calls between your systems will be free.

Connecting two trixbox CE systems together

At some point you may want to connect branch offices together, connect directly to a business partner's system, or may want to have multiple systems in different parts of your organization. There are numerous methods to accomplish this but we will focus on a simple method that will accomplish the following goals:

- Allow extension to extension calling
- Properly pass caller
- Be simple to set up

First off, let's create a table that gives the details of the two systems that we will connect together.

Parameters	System 1	System 2
Extension Range	3000 – 3999	4000 – 4999
Public IP	10.10.45.50	24.7.72.71

Within the PBX Configuration tool, we can now create a very simple trunk that will allow us to place calls between the two systems.

System 1

- Create an IAX trunk:

  Trunk Name: **InterOffic**e

  PEER Details:

  - **Qualify=yes**
  - **type=friend**
  - **host=24.7.72.71**

  Leave all the remaining fields blank.

- Create a new Outbound Route:

  Dial Patterns: **4xxx**

  Trunk Sequence: **IAX2/InterOffice**

System 2

- Create an IAX trunk:

  Trunk Name: **InterOffice**

  PEER Details:

  - **Qualify=yes**
  - **type=friend**
  - **host=10.10.45.50**

  Leave all the remaining fields blank.

- Create a new Outbound Route:

  Dial Patterns: **3xxx**

  Trunk Sequence: **IAX2/InterOffice**

After saving and applying the configurations on both systems, you should now be able to place a call between the two systems.

DUNDi

From the Asterisk website:

> *DUNDi is a peer to peer system for locating Internet gateways to telephony services. Unlike traditional centralized services (such as the remarkably simple and concise ENUM standard), DUNDi is fully distributed with no centralized authority whatsoever.*
>
> *DUNDi is not itself a Voice over IP signalling or media protocol. Instead, it publishes routes which are in turn accessed via industry standard protocols such as IAX, SIP, and H.323.*
>
> *DUNDi can be used within an enterprise to create a fully federated PBX with no central point of failure, and the ability to arbitrarily add new extensions, gateways, and other resources to a trusted web of communication servers, where any adds, moves, changes, failures, or new routes are automatically absorbed within the cloud with no additional configuration.*

When you are connecting more than two systems together, DUNDi may be a more appropriate method, as it is not only used for creating a simple trunk, but can be used to create branch offices that place their phone calls through one central phone system.

Since DUNDi is rather difficult to configure, and there is no web interface available to configure it at this time, we will not cover DUNDi trunks in this chapter.

Handling emergency calls

Something to always consider is how to handle calls to emergency service (911 in US). In cases where we don't have a PRI circuit with plenty of extra capacity, I simply recommend having a single POTS line that has its own trunk and whose only purpose is for 911 calling. For many people, the monthly cost of a dedicated phone line that is reserved only for emergency use is a little expensive. For companies with remote users, it doesn't take into account a phone at a remote location that dials 911. In this case, emergency services would be dispatched to the address of the company and not where the remote user is located.

One built-in solution is fairly simple to set up although it does require that your telephone service provider will allow you to override outbound caller ID. While some SIP trunk and PRI/BRI providers will allow you to override the outbound caller ID, you should check with your provider to see if this is possible with your service.

Let's say that Andrew works at home in New York and we want to make sure that if someone grabs his office phone and dials 911, then it gives the emergency service his actual home phone number instead of the phone number of the company, which is in California.

We edit his extension and in the Emergency CID field, we enter his phone number of <2125558534>. We then go to our Outbound Routes configurations and add a new route. Let's use the following settings to create an Emergency routing table:

Route Name: **Emergency**

Emergency Dialing: **Yes**

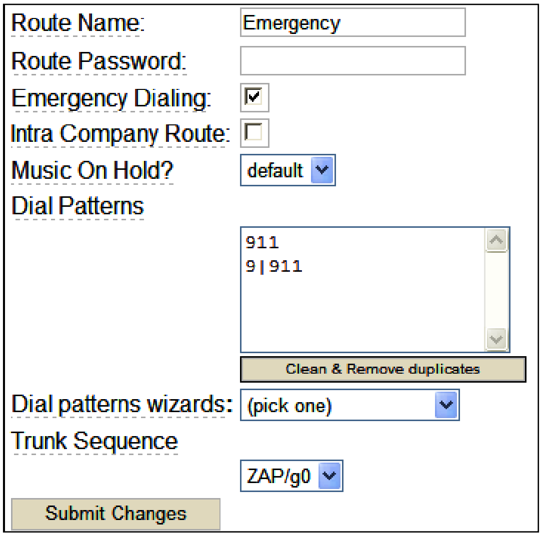
Dial Patterns: **911, 9|911**

Trunk Sequence: Select the trunk to dial out on.

Third-party services

There are also third-party 911 services that can help manage remote and mobile users. One service like this is 911Enable (`http://911enable.com`). With a service like this, you create a SIP trunk to its system and route all 911 calls through it. You then have to manage a database of your users through its web interface to let their system know where all of your users are located. Mobile users can even log in and update their information themselves if they are travelling.

When a 911 call is routed to it, it is routed to the appropriate emergency services provider in the area that is local to where the caller is located. This will provide the best quality emergency service available.



Summary

While this has been a fairly shorter chapter, we have covered some advanced trunking techniques and concepts, and have seen how to connect two trixbox CE systems. We have also looked at how to configure 911 emergency services for local and remote users.

9

PBX functions and features

Every commercial PBX system has its own set of advanced features that are activated by key commands, also known as **vertical service activation codes**. trixbox CE has its share of standard and advanced features, and we will even look at how to add our own custom features to the system.

Standard features

The following sections will break down the list of available features by category. While the codes listed are the default settings, they can be modified in the PBX Configuration tool using the **Feature Codes** module. These features are invoked by dialing the code from a registered SIP or IAX endpoint, or via an analog extension plugged into an FXS port. Some of the following features require the appropriate PBX Configuration tool module to be installed.

Call forwarding

The call forwarding mechanism is both powerful and flexible. With the different options, you can perform a number of different functions or even create a basic find-me/follow-me setup when using a feature like call *forward on no answer*, or send callers to your assistant if you are on a call using *call forward on busy*.

Function	**Code**
Call Forward All Activate	*72
Call Forward All Deactivate	*73
Call Forward All Prompting	*74
Call Forward Busy Activate	*90
Call Forward Busy Deactivate	*91
Call Forward Busy Prompting Deactivate	*92
Call Forward No Answer/Unavailable Activate	*52
Call Forward No Answer/Unavailable Deactivate	*53

Call waiting

The call waiting setting determines whether a call will be put through to your phone if you are already on a call. This can be useful in some call center environments where you don't want agents to be disturbed by other calls when they are working with clients.

Function	Code
Call Waiting Activate	*70
Call Waiting Deactivate	*71

Core features

The core features control basic functions such as transfers and testing inbound calls. Simulating an inbound call is useful for testing a system without having to call into it. If you don't have any trunks hooked up, it is the easiest way to check your call flow. Once you have telephone circuits connected, you can still use the function to test your call flow without having to take up any of your circuits.

Function	Code
Call Pickup	**
Dial System FAX	666
Simulate Incoming Call	7777

Active call codes

These codes are active during a call for features like transferring and recording calls. While some phones have some of these features built into the device itself, others are only available via feature codes. For example, you can easily do call transfers using most modern SIP phones, like Aastra's or Polycom's, by hitting the transfer button during a call.

Function	Code
In-Call Asterisk Attended Transfer	*2
In-Call Asterisk Blind Transfer	##
Transfer call directly to extension's mailbox	*+Extension
Begin recording current call	*1
End Recording current call	*2
Park current call	#70

Agent features

The agent features are used most often in a Call Center environment to monitor different calls and for agents to log in and log out of queues.

Function	Code
Agent Logoff	*12
Agent Logon	*11
ChanSpy (Monitor different channels)	555
ZapBarge (Monitor Zap channels)	888

Blacklisting

If you have the PBX Configuration tool Blacklist module installed, then you have the ability to blacklist callers from being able to call into the system. This is great for blocking telemarketers, bill collectors, ex-girl/boyfriends, and your mother-in-law.

Function	Code
Blacklist a number	*30
Blacklist the last caller	*32
Remove a number from the blacklist	*31

Day / Night mode

If you have the PBX Configuration tool Day/Night mode module installed, then you can use a simple key command to switch between day and night IVR recordings. This is great for companies that don't work off a set schedule everyday but want to manually turn on and off an off-hours greeting.

Function	Code
Toggle Day / Night Mode	*28

Do not disturb

Usually, do-not-disturb functions are handled at the phone level. If you do not have phones with a DND button on them, then you can install this module to enable key commands to toggle Do Not Disturb on and off.

Function	Code
DND Activate	*78
DNS Deactivate	*79

Info services

The info services are some basic functions that provide information back to you without changing any settings. These are most often used for testing and debugging purposes.

Function	**Code**
Call Trace	*69
Directory	#
Echo Test	*43
Speak your extension number	*65
Speaking Clock	*60

Intercom

If you have a supported model of phone then you can install the PBX Configuration tool module to enable paging and intercom via the telephone's speakerphones.

Function	**Code**
Intercom Prefix	*80
User Allow Intercom	*54
User Disallow Intercom	*55

Voicemail

If you want to access your voicemail from any extension then you need to choose 'Dial Voicemail System', otherwise using 'Dial My Voicemail' will use the extension number you are calling from and only prompt for the password.

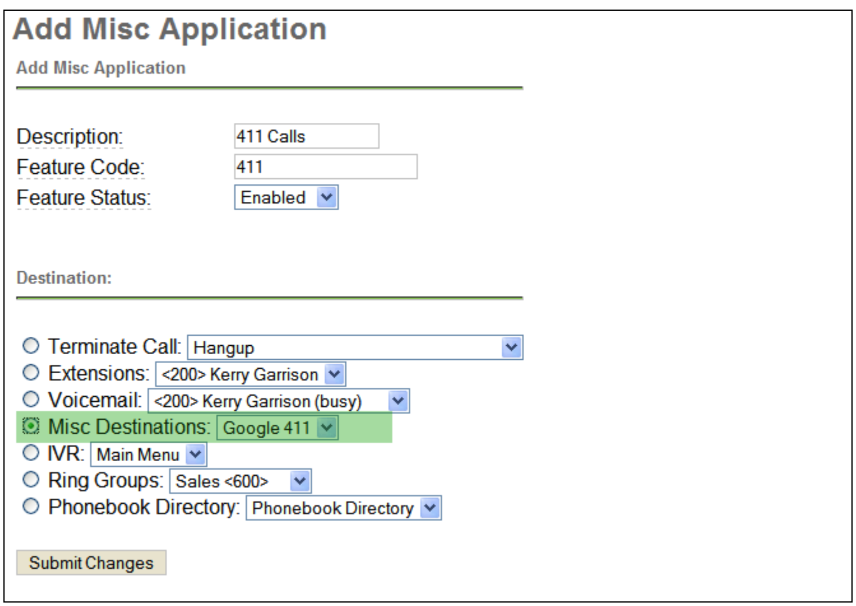
Function	**Code**
Dial Voicemail System	*98
Dial My Voicemail	*97

Adding new features

The ability to add new features is built into the system. One common thing to do is to redirect 411 calls to a free service like Google's free service. The following steps will walk you through how to add a custom feature like this to your system.

Begin by going to the **Misc Destination** module and enter a **Description** of the destination you want to create.

Next, go to **Misc Application** to create the application. Here we will enter another **Description** and the number we want to use to dial the application, make sure the feature is enabled, and then point to the destination that we created in the previous step.



As you can see, any code can be assigned to any destination and a custom destination can consist of anything you can dial. This allows you to create many different types of custom features within your system.

Voicemail features

trixbox CE comes with the **Asterisk Mail** voicemail system. Asterisk mail is a fairly robust and useful voicemail system. The Asterisk Mail voicemail system can be accessed by any internal extension or by dialing into the main IVR system.

As we saw earlier in this chapter, there are two ways of accessing the voicemail system, 'Dial Voicemail' and 'Dial My Voicemail'. To access the main voicemail system, we can dial *98 from any extension; we will then be prompted for our extension and our voicemail password.

If we dial *97 for the 'My Voicemail' feature, the system will use the extension number you dialed in from and only prompt you for your voicemail password.

The following tables will show you the basic structure of the voicemail menu system:

Voicemail main menu options

Press:

- 1 to Listen to (New) Messages
- 2 to Change Folders
- 0 for Mailbox Options
- * for Help
- # to Exit

Listen to messages

Press:

- 5 to Repeat Message
- 6 to Play Next Message
- 7 to Delete Message
- 8 to Forward to another user

  Enter Extension and press #

  - 1 to Prepend a Message to forwarded message
  - 2 to Forward without prepending
- 9 to Save Message
- 0 for New Messages
- 1 for Old Messages

- 2 for Work Messages
- 3 for Family Messages
- 4 for Friends Messages
- * for Help
- # to Cancel/Exit to Main Menu

Change folders

Press:

- 0 for New Messages
- 1 for Old Messages
- 2 for Work Messages
- 3 for Family Messages
- 4 for Friends' Messages
- # to Cancel/Exit to Main Menu

Mailbox options

Press:

- 1 to Record your Un-Available Message
- 2 to Record your Busy message
- 3 to Record your Name
- 4 to Change your Password
- # to Cancel/Exit to Main Menu

User area

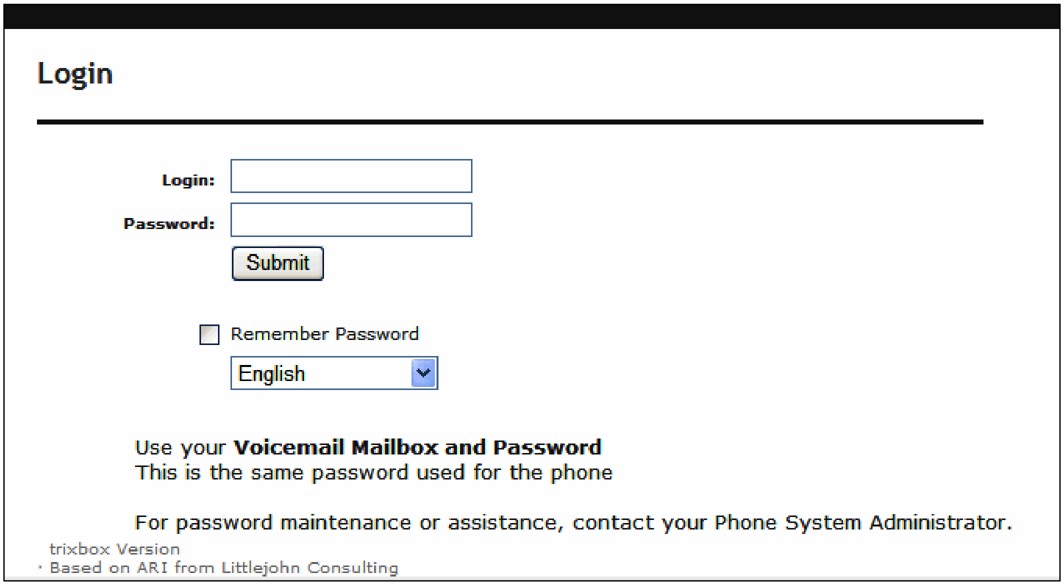
trixbox CE features a user area, which has several tools that are designed for non-administrators. These tools include:

- **User Portal**:
    - This is for the individual users to manage their own voicemail and settings.
- **MeetMe**:
    - Users with access can create scheduled conference bridges.

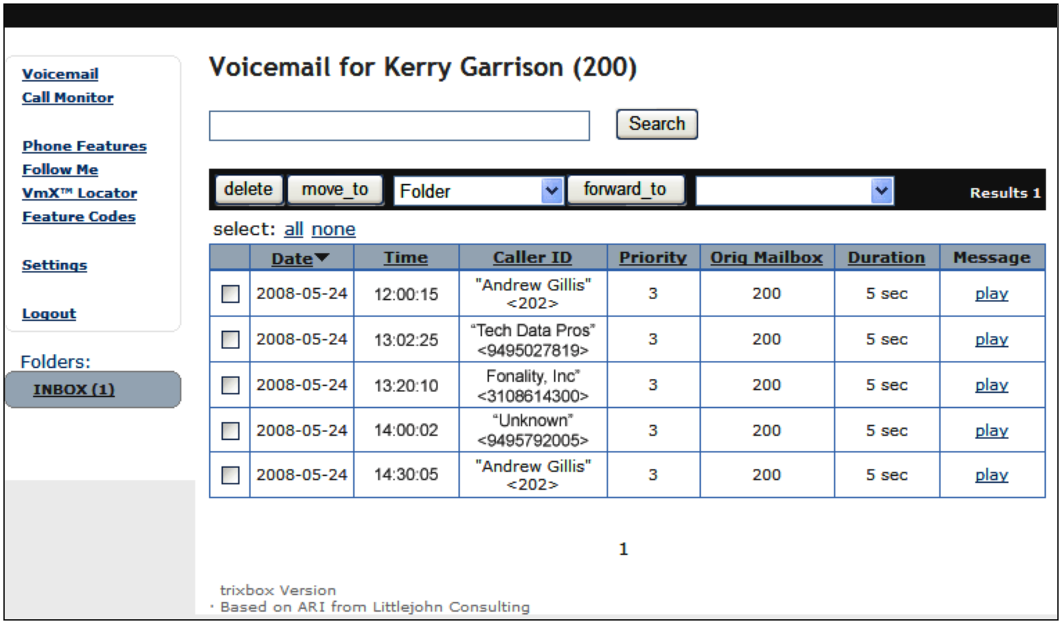
- **Flash Operator Panel**:
    - This is a web-based tool that will show the status of the extensions, trunks, queues, and parked calls. Users that know the password can transfer calls between extensions.

User portal

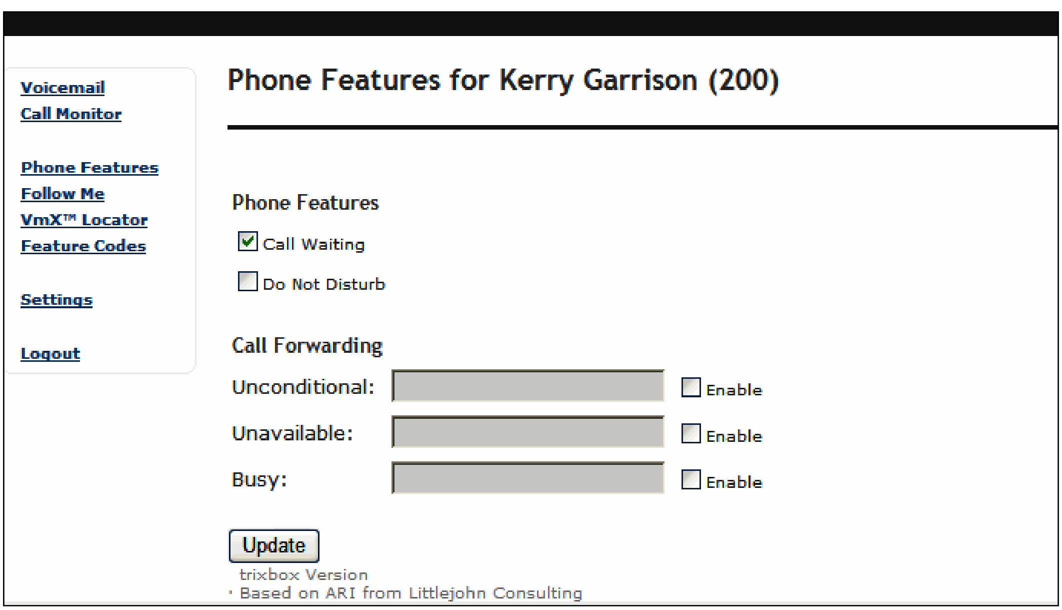
Any user can log into his/her user portal with his/her extension number and voicemail password. When you first access the user portal, it will ask you for your **Login** and **Password**.



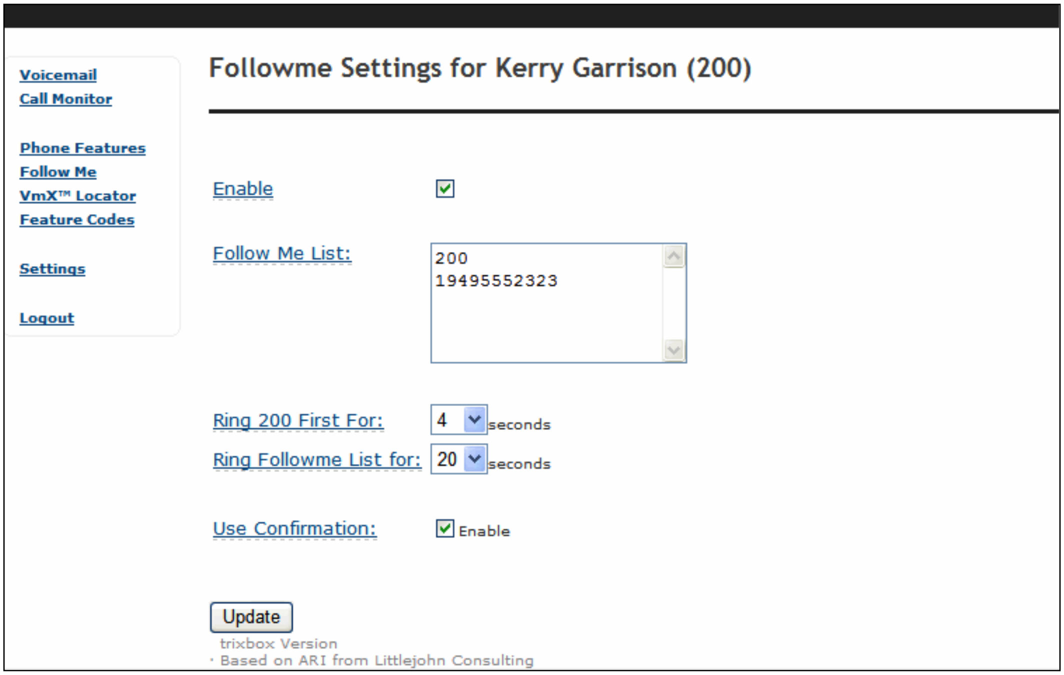
Once you have logged in, you will see a menu on the left and a list of available voicemails in the middle. You can select multiple messages and then use the pull-down menu to delete, move, or forward the messages. If you have recorded calls, the interface is very much the same and you access those recorded calls though the **Call Monitor** link in the left-hand menu.



Under the **Phone Features** menu, you will find options to enable and disable **Call Waiting** and **Do Not Disturb**. You can also configure different types of **Call Forwarding** from the same screen.

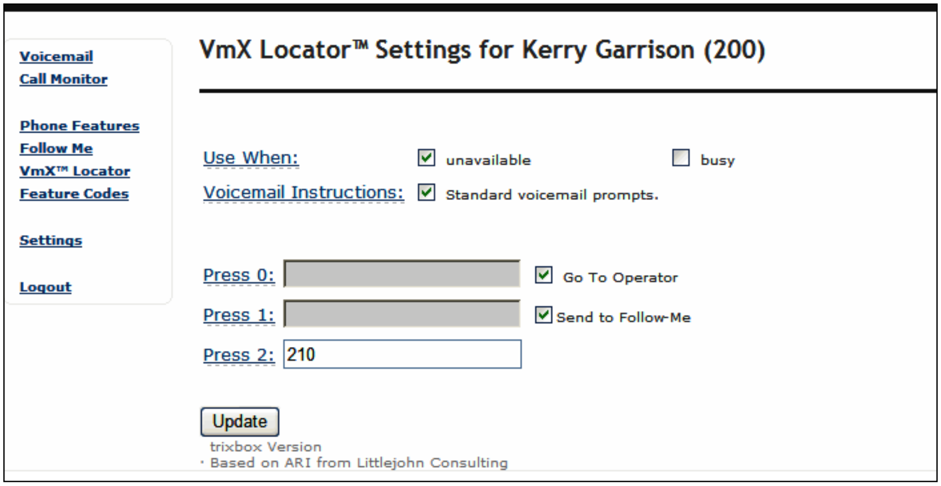


If an administrator has already enabled and set up Follow-Me options for an extension, then the user can modify his or her own options. A user can enable or disable his or her Follow-Me settings, change the list of where a call will go during the Follow-Me process, set the ring times and whether or not to **Use Confirmation** (which will require the user to press 1) to accept a call when the call is answered from an outside line. This will help keep calls from going into a user's cell phone's voicemail and ensure that all voicemails remain on the phone system.

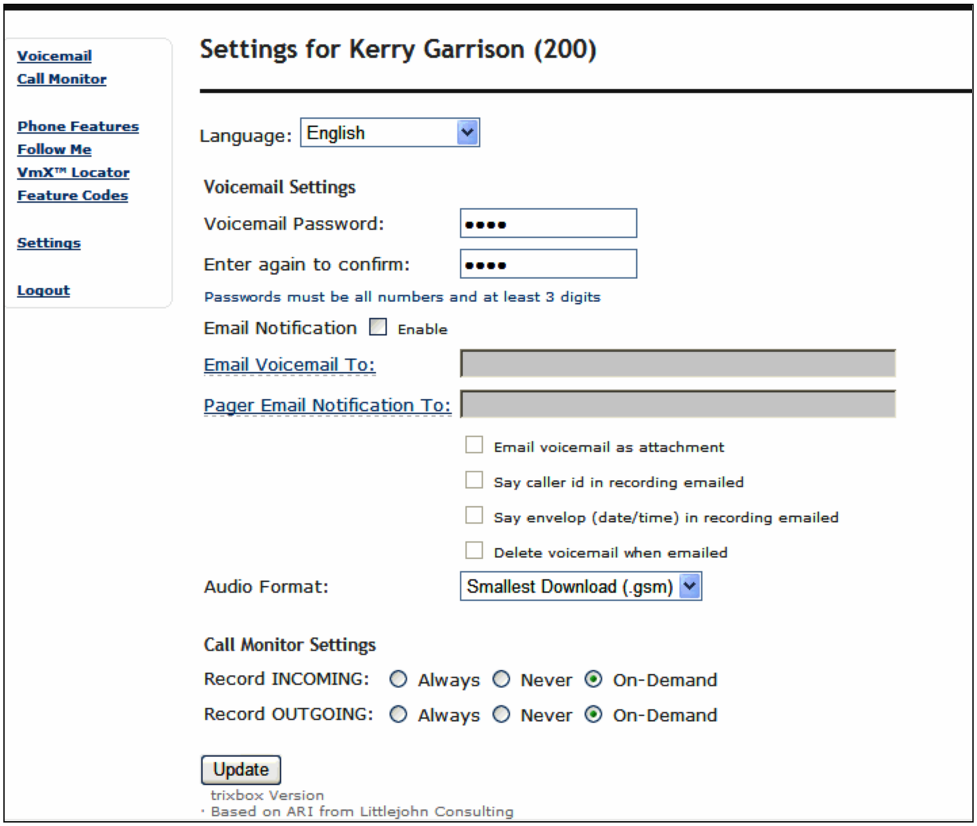


The **Locator** system is more like a personal digital receptionist as it allows each user to create his or her own call menu that can route users to different places. The **0**, **1**, and **2** keys can be programmed to either call another extension or phone number, as well as go to the company operator (configured in General Settings) or to send the call to the Follow-Me system.

If the user is going to use the Locator function, then their voicemail greeting has to tell the caller about the options and what keys are available to press.



The **Settings** menu contains the settings for the user to change his or her **Voicemail Password**, and set up **Email Notification**, **Audio Format**, and **Call Monitor Settings**.

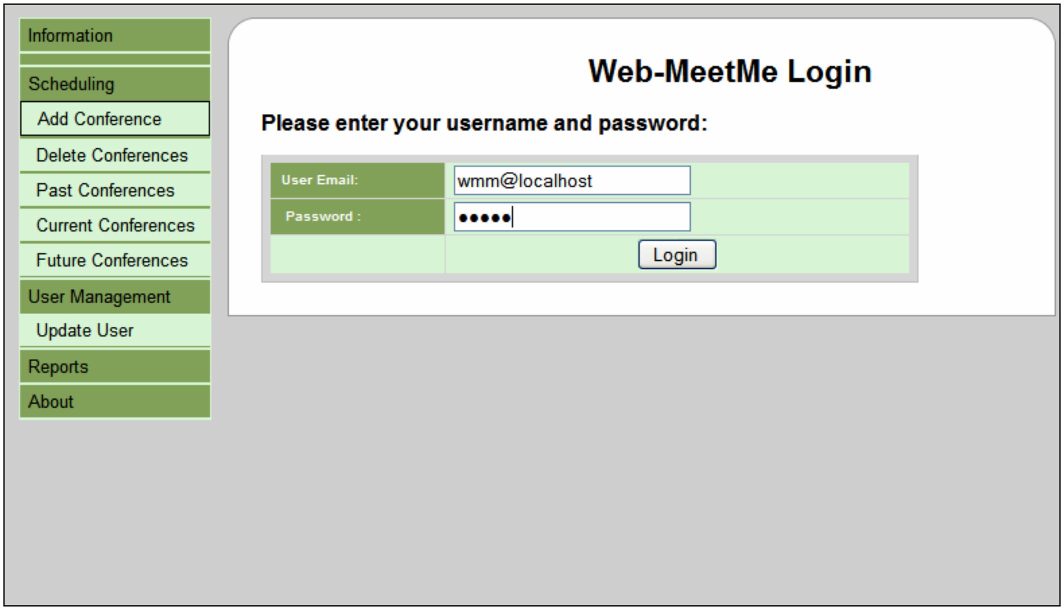


MeetMe

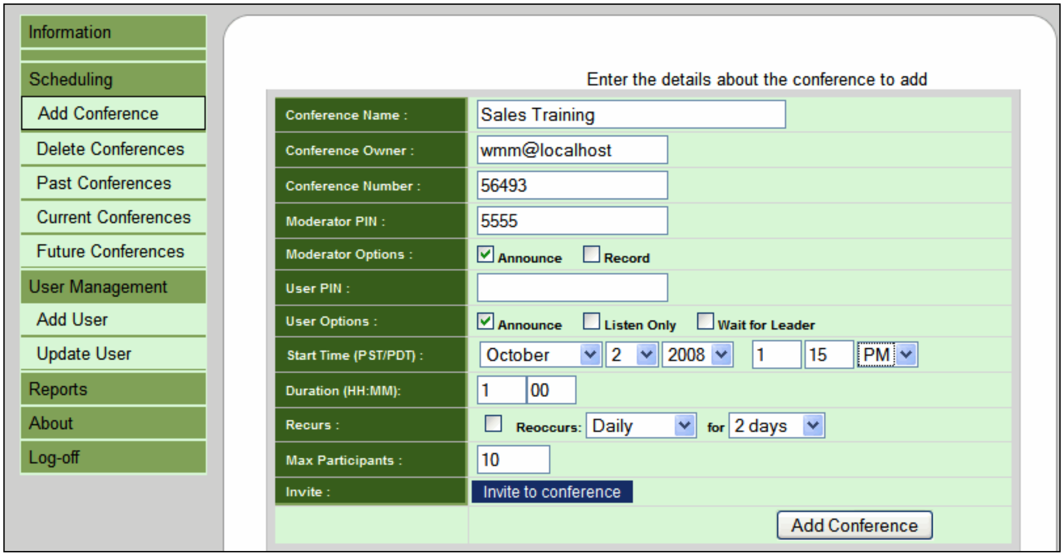
The **Web-MeetMe** control is for creating scheduled conference rooms and being able to prepare reports on previous conferences. The MeetMe tool is part of the user mode section since regular users can be given accounts in it. To log in, initially you need the default login and passwords:

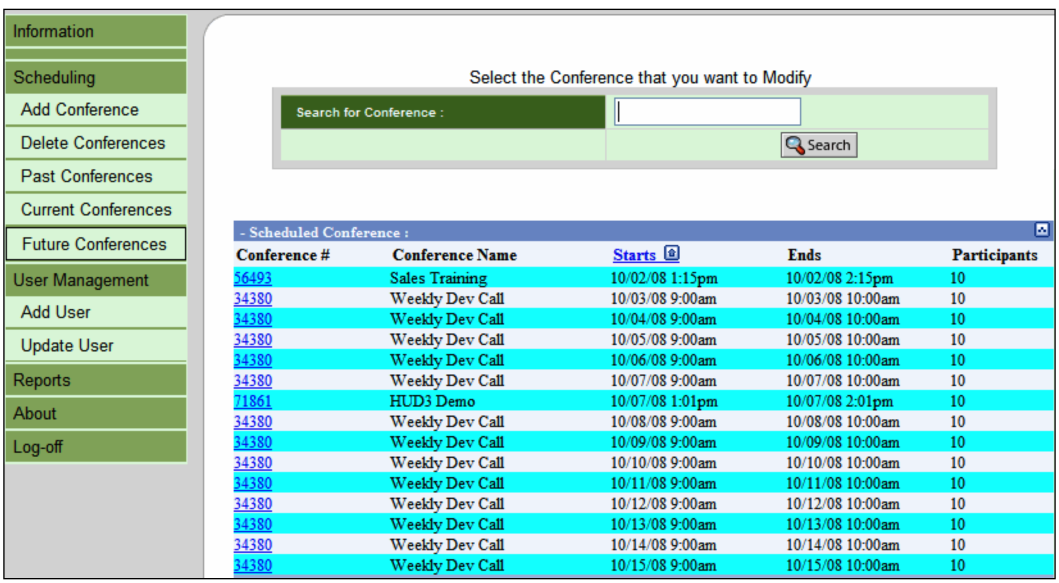
Admin Mode:
Username: wmm@localhost
Password: wmmpw

Standard user:
Username: tim@localhost
Password: 1234



Once logged into the MeetMe system, you have access to create new conference rooms as well as **Delete Conference**s, view **Past Conferences**, manage **Current Conferences**, and see upcoming conferences. You can also manage users who have access to the MeetMe system.

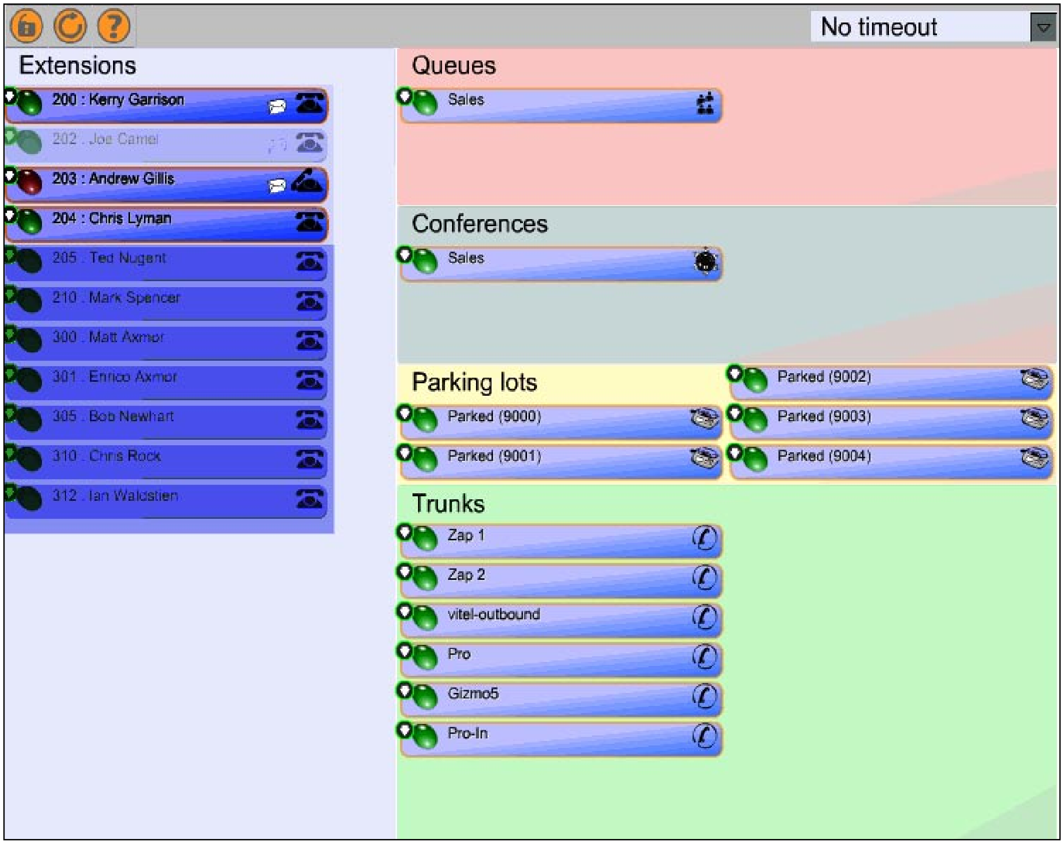




FOP (Flash Operator Panel)

The Flash Operator Panel is a simple tool to view the status of your system as well as to manipulate calls somewhat. The first time you try to perform an action (such as dragging one extension to another), it will prompt you for a password. The default password is: passw0rd (that's a zero, not a capital O).

A limitation in FOP is that it is difficult to manage large numbers of extensions as you quickly run out of available slots that can be used. Even so, it is still a fairly useful tool for a receptionist to see who is available for calls and for quickly transferring calls to people.



Summary

So far we have covered the basic features of a trixbox CE system, and now we should be ready to look at all of the other features that are available. This chapter covered the basic feature codes and extra functions, as well as how to create custom features like redirecting calls for 411 to a free provider. The following chapter goes into detail on all of the PBX settings and what their options are.

10
PBX settings in detail

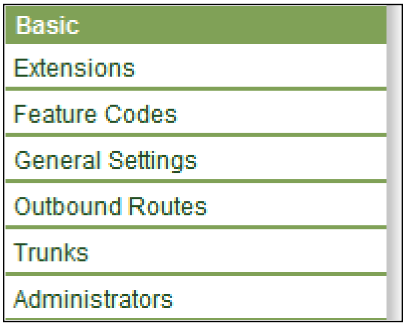
This chapter goes into the meat of the PBX settings and shows you what each feature does and how to configure it. Along the way, we will look at some best practices and see how to best configure our systems.

For this chapter, we will have installed every module that is available in the PBX Settings tool under the Admin module. We also want to go online and check for updates and make sure everything is current.

Basic settings

The Basic settings consists of the following items:

- Extensions
- Feature Codes
- General Settings
- Outbound Routes
- Trunks
- Administrators



Extensions

Extensions are the combination of the user and the device that is assigned to the user. Every endpoint must be assigned as an extension in order for it to place and receive phone calls.

The Extensions page is quite long, so we are breaking it up into different sections in order to make it easier to understand.

Edit Extension

The first section of the Extension page contains the following settings:

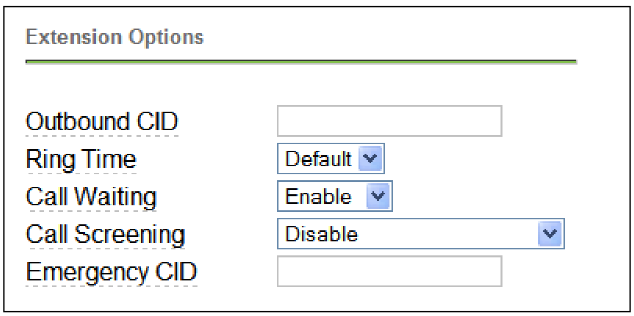
- **Display Name**: Used as a description of the extension as well as being used by the dial-by-name directory.
- **CID Num Alias**: Used internally to masquerade this extension as a different user. An example of this is an IT support department made up of multiple people but that only wants a single caller ID number to be displayed when calling people within the company.
- **SIP Alias**: Defines a friendly name that is used for direct SIP dialing. This feature is rarely used.

Extension Options

The second section, under the Extension Options heading, contains the following settings:

- **Outbound CID**: This is used to override the global outbound caller ID information with a different caller ID as specified here. While company employees have individual DIDs, the company may want to show a Main Listed Number to called parties. This field uses the standard caller ID format of *caller name* <##########>.
- **Ring Time**: This value specifies the number of seconds to ring this extension before going to voicemail. If this is set to **Default**, it will use the value in the General Settings section.

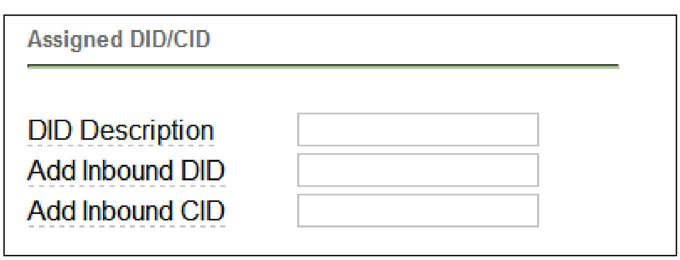
- **Call Waiting**: This sets the initial/current call waiting state for this extension.
- **Call Screening**: This setting requires that external callers say their name before the call will be sent to the extension. Screening with memory verifies against the inbound caller ID only once. Screening without memory always requires a caller to say his or her name.
- **Emergency CID**: If a number is specified here, it will be used as the outbound caller ID when dialing out on a trunk that is specified as an emergency trunk.



Assigned DID/CID

In this section, you have additional control over inbound calls. You can create a combination that will even trap certain types of calls with blocked caller ID. The following options are available:

- **DID Description**: This is a text description of the DID you are specifying, such as "Kerry's DID".
- **Add Inbound DID**: This field allows you to specify a DID that is assigned to this extension. You need to use the exact format as provided by your phone service provider; this can typically be of 4, 7, or 10 digits.
- **Add Inbound CID**: This field allows this extension to also trap specific caller ID numbers in addition to the inbound DID number.

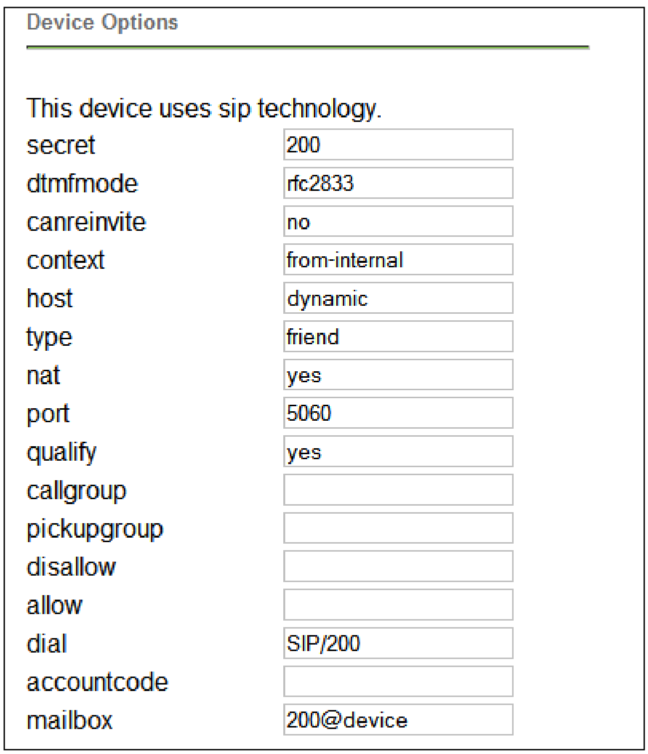


Device Options

The following set of fields is used to configure the device functions. In most cases you will only set the device secret and use the rest as defaults. There are some exceptions that we will cover. The available settings are as follows:

- **secret**: This is the password that the device uses to authenticate to the PBX. It is highly recommended that you use strong passwords here and not the simple ones as shown in the examples in this book.
- **dtmfmode**: The default setting is **rfc2833**, which should work for most devices. If this setting is not allowing DTMF tones to be sent properly, other possible settings are **Inband** and **Auto**.
- **canreinvite**: This setting, under the right conditions, can be used to take Asterisk out of the call and redirect the audio path to go directly between the two endpoints. This setting is usually set to **no** for greater hardware compatibility.
- **context**: One can think of contexts for extensions as a type of group to which the extension belongs. Normally, this is set to **from-internal**, and it is not recommended that you change it unless you know exactly what you are doing.
- **host**: This specifies the IP address or host name on which the system should expect to see this endpoint registering. This will prevent an endpoint assigned to this extension from registering on an unassigned IP address. If the device for this extension is on DHCP, or may change because it is remote, then this setting will need to be **dynamic**.
- **type**: Extensions are almost always set as **friends**. Only in special circumstances, when doing custom extensions, would you need to change this.
- **port**: This specifies the UDP port that this extension will use for SIP signaling.
- **qualify**: If this is set to **yes**, or a specific number, Asterisk will monitor the extension and not allow calls to go to it if the ping time exceeds the threshold that is set.
- **callgroup**: This is used to specify a numeric (**0 – 63**) group that this extension is in; this is then used by the **pickupgroup** setting to determine who can pick up this call if it is ringing.
- **pickupgroup**: The setting specifies what callgroups can be picked up from this device. If a ringing device is in callgroup 5, and this device has a pickupgroup value like 1-4, 5, 7, then this device can use the *8# command to answer calls that are ringing to that device since the pickupgroup definition would cover callgroup 5.

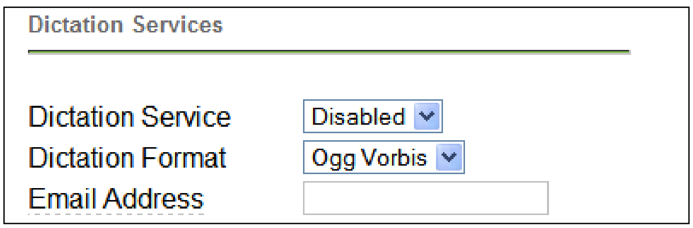
- **disallow**: This field specifies codecs that you do not want to be used by this extension.
- **allow**: This field specifies codecs that you want to be used by this extension. This field also overrides the disallow value, so setting **disallow = all** and **allow = iLBC** would allow only iLBC calls to come through.
- **dial**: This specifies how Asterisk will dial to reach this extension. You should almost never change this unless you are very familiar with how Asterisk works.
- **accountcode**: Adds an account number entry into the Call Detail Record (CDR) reports.
- **mailbox**: This specifies the voicemail box that is associated with this extension. Normally, you will want to keep this as the default.



Dictation Services

This section will only appear if you have the Dictation module installed. If you do, then you can use your phone to record a memo and then have it emailed to a specific person when finished. To use dictation, you dial *34 to begin dictating, and then dial *35 to email the completed dictation to the specified email address. Use the following fields to configure the dictation module:

- **Dictation Service**: **Enabled** or **Disabled**.
- **Dictation Format**: This specifies the file format used to record the dictation. This can be **Ogg Vorbis**, **GSM**, or **Wav**.
- **Email Address**: This is the email address that will receive the dictation file when the user dials *35.



Call Recording Options

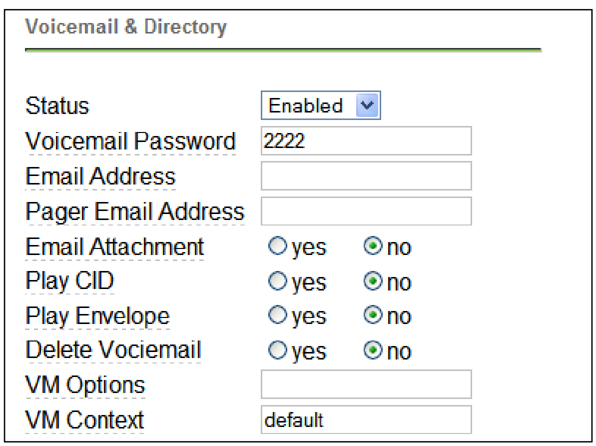
You can set the call recording options for each extension to never record calls, record calls on demand, or always record calls. This can be set separately for inbound and outbound calls and will require the proper setting in the **Dial commands** on the **General Settings** page. The settings needed to record calls are covered in that section.

Voicemail and Directory

This section sets all of the options that are used with the voicemail system and how the extension fits into the company directory. The following settings are available:

- **Status**: **Enabled/Disabled**.
- **Voicemail Password**: This is a numeric password that is used to authenticate into the voicemail system.
- **Email Address**: This is the address that is used to notify the user of a new voicemail message.
- **Pager Email Address**: This email address is used to send short voicemail notifications to a pager or mobile phone.

- **Email Attachment** (**yes**/**no**): This determines if a WAV file of the voicemail message is emailed to the address specified in the **Email Address** field.
- **Play CID** (**yes**/**no**): If this is set to **yes**, then the caller ID number of the calling party will be read to you when you listen to the voicemail.
- **Play Envelope** (**yes**/**no**): If this is set to **ye**s, then the date and time that the voicemail was left will be read to you when you listen to the voicemail.
- **Delete Voicemail** (**yes**/**no**): If this is set to yes, then the voicemail file will be deleted once it is emailed to the **Email Address** location.
- **VM Options**: This is used to set options otherwise not available via the GUI. The field accepts a pipe-delimited list of options such as "review=yes| maxmessage=60".
- **VM Context**: Using this you could create different company directories. Do not change it from the default setting unless you fully understand what this will affect.

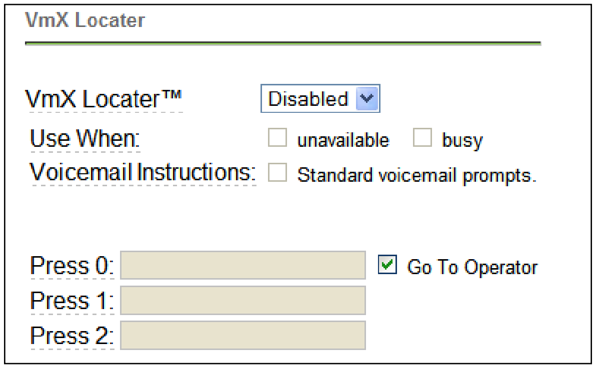


VmX Locater

The VmX Locater can create a mini-IVR for each extension. While this can be set up here, each user can manage these settings via the ARI system in the User Portal. The following settings are available:

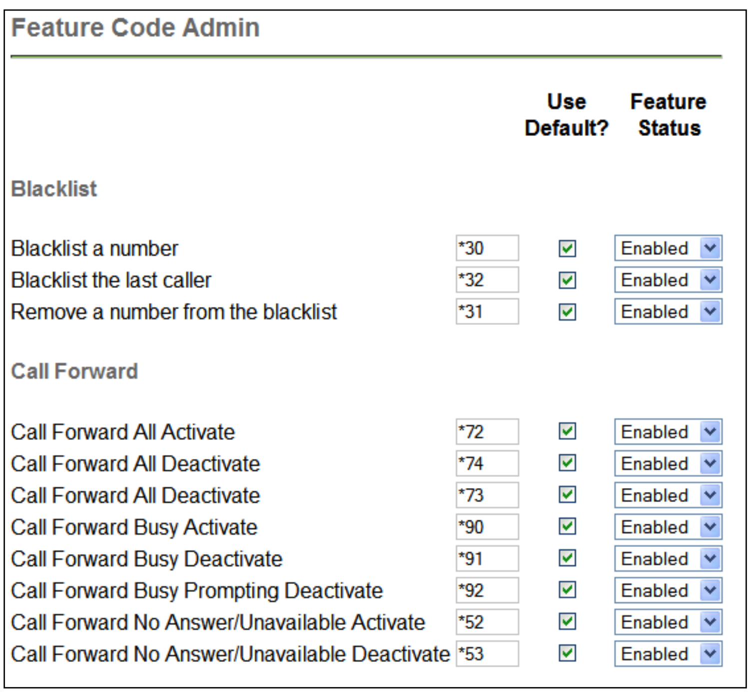
- **VmX Locater**: **Enabled**/**Disabled**.
- **Use When**: You can select to use the VmX Locater when the extension is **unavailable**, **busy**, or both.
- **Voicemail Instructions**: If you want to use the standard voicemail prompts, you can select this, otherwise only beeps will be heard and your greeting will have to explain to callers what to do.

- **Keypress Options** (**0**, **1**, **2**): You can specify numbers to call when the caller presses **0**, **1**, or **2**. This can be your cell phone or another extension, for example.



Feature Codes

The Feature Codes module allows you to change the default settings for different internal feature codes. You can also enable or disable different codes on a global setting if there are codes that you don't want your users to be able to use.



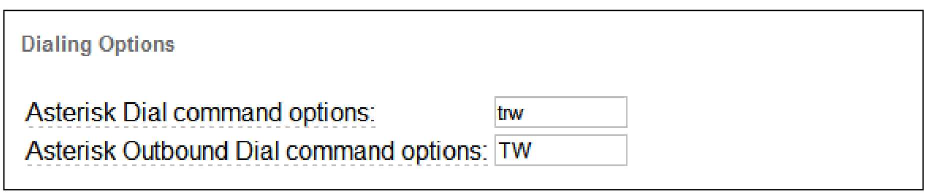
General Settings

This module sets a number of global system settings. We will go through the page and explain each feature and how to configure it.

Dialing Options

The dialing options control a number of different things that can happen during the call, such as recording and transferring. The different options that are available include:

- **T**: allows the *calling user* to transfer the call
- **r**: generates a ringing tone for the calling party
- **R**: indicates ringing to the calling party when the called party indicates ringing (only available when using Bristuff)
- **m**: plays music on hold to the calling party until the called channel answers
- **h**: allows the called party to hang up by pressing *
- **H**: allows the calling party to hang up by pressing *
- **w**: allows the called user to start recording by pressing *1
- **W**: allows the calling user to start recording by pressing *1



Call Recording options

This section will set some parameters on how call recordings are handled. The following options are available:

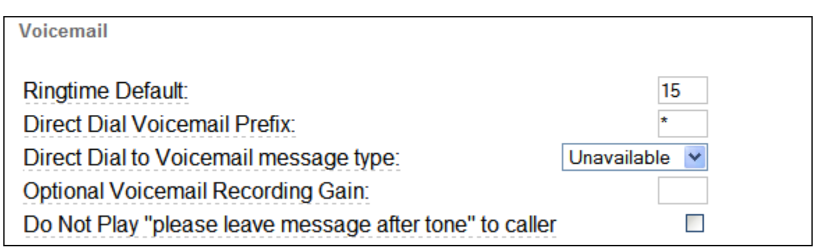
- **Extension Recording Override**: If this is set to **Enabled**, then it checks to see if calls to/from extensions can be recorded. If you do not allow your extensions to do recording, then this is one way to increase performance. This setting does not override the on-demand function that we set in the previous section.
- **Call recording format**: This setting determines the file format for recorded calls. The options available are: **wav**, **ulaw**. **alaw**, **sln**, **gsm**, and **g729**.

- **Recording Location**: This field allows you to set a different location for recorded calls. This is very handy to keep call recordings from using up the system's disk space, which can cause the system to crash if the disk space runs out.
- **Run after record**: This allows you to run a script after the call is hung up to do further processing, such as remixing it to mp3, or updating a database.

Voicemail options

These settings control different options related to how callers are handled and voicemail is recorded. The following options are available:

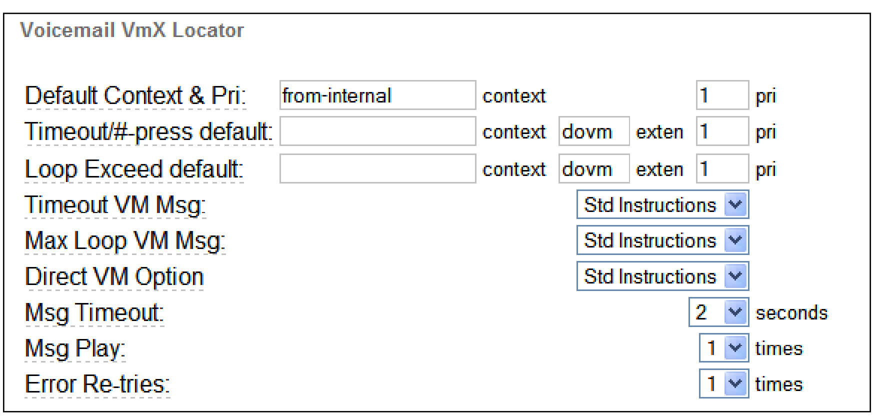
- **Ringtime Default**: This is the number of seconds a phone will ring before going to voicemail. The standard phone will ring every six seconds, and the default setting is 15 seconds, so your phone will ring twice and pause for a few seconds before going to voicemail.
- **Direct Dial Voicemail Prefix**: This character is used to transfer someone directly to voicemail; for example, if you take a call and want to transfer the call to extension 200's voicemail, you would transfer the call to *200 and it would go right into the user's voicemail instead of ringing the phone first.
- **Direct Dial to Voicemail message type**: This selects the **Unavailable** or **Busy** message, or **no message** when dialing directly to a user's voicemail.
- **Optional Voicemail Recording Gain**: If you need to increase or decrease the recorded volume of voicemails, you can adjust the gain by entering a numeric value into this field.
- **Do Not Play "please leave a message after tone" to caller** (yes/no): If this is checked, then the message to leave a message after the tone will not be played.



Voicemail VmX Locater options

These options control how the VmX Locater system functions; they are a little complicated for novice users but can allow for extra control for more advanced, users. The options include:

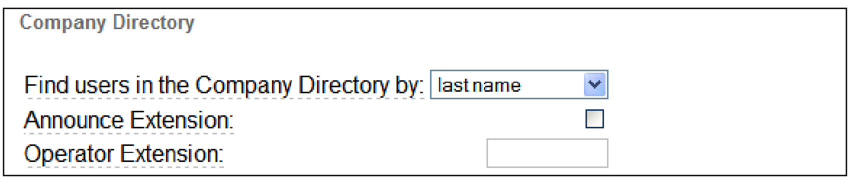
- **Default Context & Pri**: The sets the default context and priority; this should only be changed if you are doing custom things in your dialplan as changing this can easily break how the system functions.
- **Timeout/#-press default**: This is an optional field for sending callers to a different context if they don't press a key or if they press #, which is considered a timeout. Setting the context to **dovm** will send them to voicemail and is normally not changed.
- **Loop Exceed default**: This is an optional field for sending callers to a different context if they press an invalid option too many times. Setting the context to **dovm** will send them to voicemail and is normally not changed.
- **Timeout VM Msg**: This sets whether to play the standard voicemail prompts or only a beep when a timeout occurs.
- **Max Loop VM Msg**: This sets whether to play the standard voicemail prompts or only a beep when an invalid keypress is sent too many times.
- **Max Loop VM Option**: If a user-defined option is to go to voicemail, this is the default option if not specified in the user settings.
- **Msg Timeout**: This is the time to wait after the message in seconds before considering the call to have timed out.
- **Msg Play**: This sets the number of times the message will repeat if the caller does not press any key and it times out.
- **Error Re-tries**: This is the number of times to play invalid options and repeat the message when receiving an invalid keypress.



Company Directory options

These settings control how the company directory (dial-by-name system) functions. The available settings are:

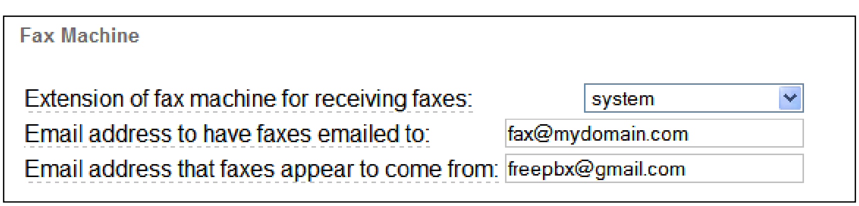
- **Find users in the company direction by** (**last name/first name/first or last name**): This determines what the company directory system will ask the caller for and how it will attempt to find the user in the directory system.
- **Announce Extension**: If this option is checked, then the caller will be told the extension of the user that it found. This is typically turned off in order to keep people from finding out the internal extension numbers.
- **Operator Extension**: This is the extension that callers will be transferred to when they hit **0** when in the directory. This can be any valid number, such as a ring group, queue, or even an external phone number.



Fax Machine options

These settings control how inbound faxes are handled. While the system can be used to receive faxes, it is not recommended, as the success rate is typically not as good as for dedicated fax hardware. The following settings are available:

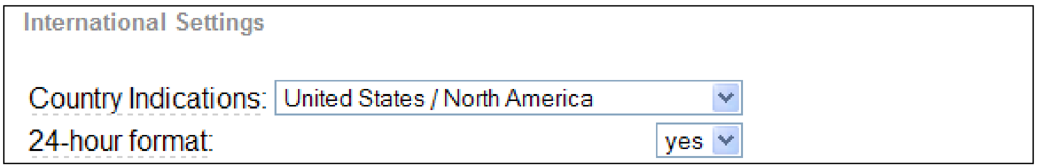
- **Extension of fax machine for receiving faxes**: If **system** is selected, then the system will receive the fax and attempt to process it. If an extension is set to receive the fax, then the system will forward the call to that extension. If this setting is set to **disabled**, then fax detection will be turned off and calls will be answered faster.
- **Email address to have faxes emailed to**: If the system is set to receive faxes, then this field controls where the faxes are emailed to.
- **Email address that faxes appear to come from**: If the system is set to receive faxes, then this field controls who the faxes appear to be emailed from.



International Settings

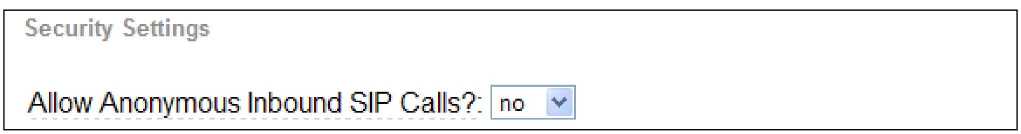
These settings are used to control how the system functions in different countries. The available options are:

- **Country Indications**: This sets certain internal settings based on the country you are in.
- **24-hour format** (**yes**/**no**): This sets how time is used in the system; if set to **yes**, then the system will use a 24-hour format versus a 12-hour format.



Security Settings

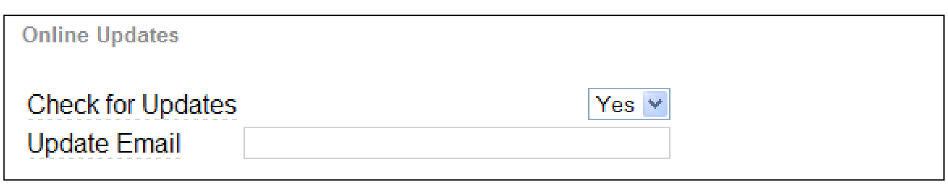
The only option available in this section is **Allow anonymous Inbound SIP Calls**; you will need to set this to **yes** if you wish to use ENUM trunks, allow direct IP calls, or if your SIP provider requires it, otherwise it is best to leave this setting at **no**.



Online Updates

This setting allows the system to periodically check for updates; if they exist then you can be sent an email notification. The following settings are available in this section:

- **Check for updates** (**Yes**/**No**): This determines if update checking is enabled or not.
- **Update Email**: This option sets the email address to send update notifications to.



Outbound Routes

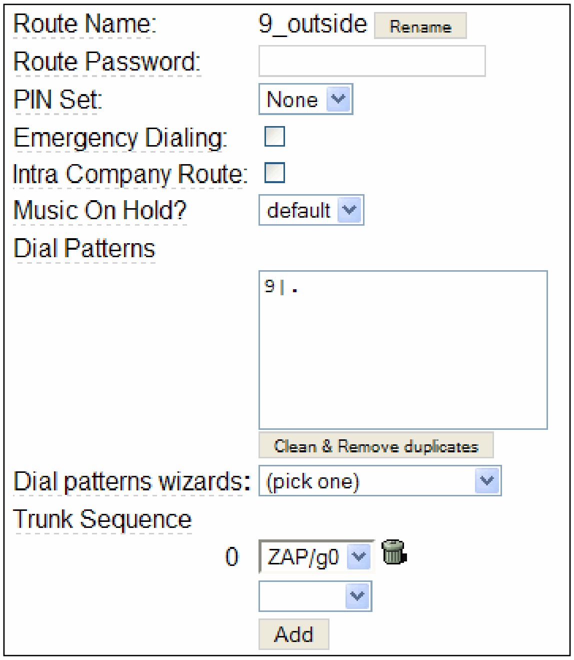
Outbound routes determine how calls going outside the system are handled and how they are routed. This can be used to set up different routes for local versus long distance, or to dial different codes to use different trunks. We will look at the different options to see how we can configure our system to how we want to use it.

The minimum settings for an outbound route are a route name, a dial pattern to match against, and a trunk selected for calls to go out on. By default, a basic route is pre-configured for you with the basic requirements that any number that starts with 9 will have the 9 removed from the number and the call will be placed on the **Zap/g0** trunk.

Information on setting up a route and how to use pattern matching is found in Chapter 4. In this section, we will go through the settings again just as a refresher.

- **Route Name**: This is for reference purposes only.
- **Route Password**: If you have a password set for a route, then the system will prompt the user before dialing out. This could be used for blocking 900 numbers or international calls.
- **Emergency Dialing**: If this trunk is used for emergency calls, you can set this option and the system will use the emergency caller ID setting of the extension that is dialling out. This is useful for telecommuters; you could set their emergency caller ID to their home phone number so that if they dial 911 from their remote phone, the emergency center would get the address where the phone is physically located.
- **Intra Company Route**: If this option is selected, then the device's internal caller ID is used instead of using the outbound caller ID. If you have branch offices connected together on this trunk, you want this option set to make sure the called party sees the extension of the person who called instead of the company outbound caller ID setting.
- **Music On Hold?**: You can choose which music-on-hold category will be used for calls that go out on this route.
- **Dial Patterns**: Only calls that match this list of patterns will be allowed to go out on this route.

- **Trunk Sequence**: This is the sequence of trunks that are used to place this call.



Trunks

Trunks are connections between systems. For example, a connection from your PBX to a SIP provider or to a remote PBX would use a trunk. When connecting to another system, you need to know how you are going to connect the two systems together. When working with Analog lines or PRI circuits, we will be using Zap trunks; when working with a VoIP provider, this could be either SIP or IAX2 depending on the provider; ENUM trunks were discussed in a previous chapter; and DUNDi trunks are used when creating a network of systems, typically a distributed network of systems for one company.

Since the main types of trunk setups have already been covered in Chapter 4, we will not go through them again here.

Administrators

Sometimes you want to give people access to specific parts of the system without giving them full control; the Administrators module allows you to do exactly that. Setting this up is a little tricky though.

First off, you need to go to the command line of the system either on the console or by SSH'ing into the system with a tool like Putty.

Edit the Apache configuration file with the following command:

```
nano /etc/trixbox/httpdconf/trixbox.conf
```

Comment out the entire first section so that it looks like the following:

```
#Password protect /var/www/html/admin
#<Directory /var/www/html/admin>
#AuthType Basic
#AuthName "Restricted Area"
#AuthUserFile /usr/local/apache/passwd/wwwpasswd
#Require user wwwadmin maint
#</Directory>
```

To save the file, type `Ctrl+O`, and then type `Ctrl+X` to exit.

Next, you need to switch the authentication mode of the system by editing the `amportal.conf` file. To do this, edit the file with the following command:

```
nano /etc/amportal.conf
```

Scroll down and find the line that says:

```
AUTHTYPE=none
```

Change it to:

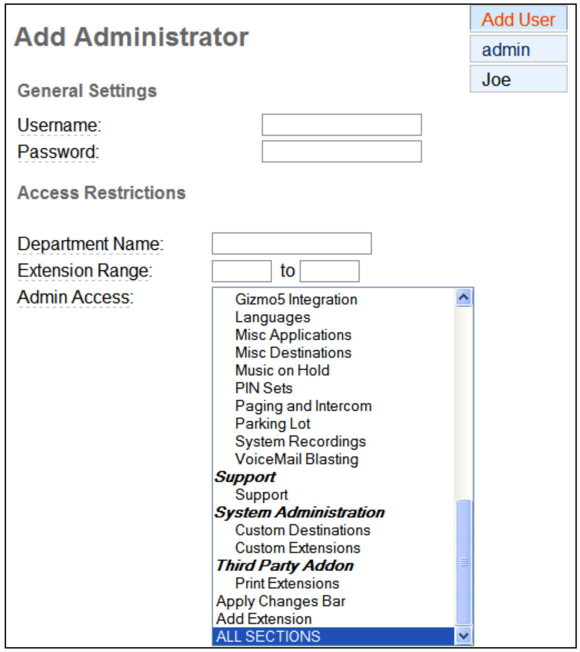
```
AUTHTYPE=database
```

To save the file, type `Ctrl+O`, and then type `Ctrl+X` to exit.

The next step is to restart the Apache service with the following command:

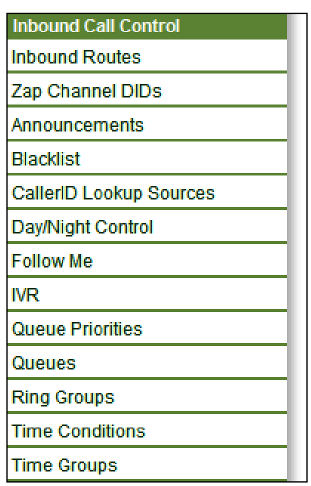
```
service httpd restart
```

You can now use the Administrators module to create different accounts with different levels of access within the PBX Settings module.



Inbound Call Control

Now that we have looked at the Basic functions group of features, we now turn to the next section, which covers what happens to a call when it enters the system. These settings will consist of options like playing announcements, routing based on DID or CallerID, Blacklisting, Day/Time control, Ring Groups, Queues, and IVR (Digital Receptionist).



By understanding how to build good inbound call flow, you can design a very robust system for yourself or your clients.

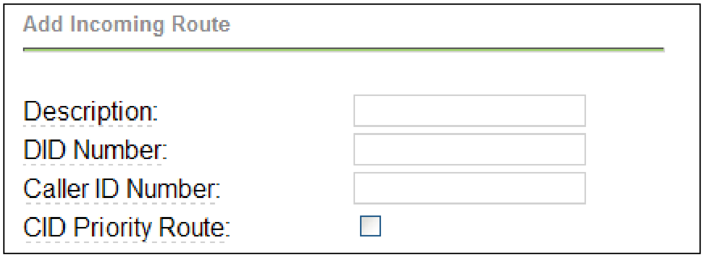
Inbound Routes

Setting up inbound routes is one of the more complex tasks you can do but in many cases it will be one of the easiest. If you simply want all calls that come into the system to go to the same places, such as the IVR, then the only thing you need to do is to select the destination and leave all the other fields blank; this will create an "Any CID/Any DID" inbound route. This is the simplest setup possible. Of course, there are dozens of other available settings that we need to understand for more advanced setups. This section will go through all of those settings.

Add Incoming Route

The first section defines the route name and if the route is based on Caller ID, DID Numbers, or both. DID routing is most often used to send calls directly to a particular department; for example, a support department may have a direct phone number that will go directly to it instead of a caller having to go through the main IVR. Caller ID routing is often used to provide VIP service to clients. A client can call into the system, and based on their caller ID, he or she can be routed directly to an extension or to a priority queue for faster service. The available settings are:

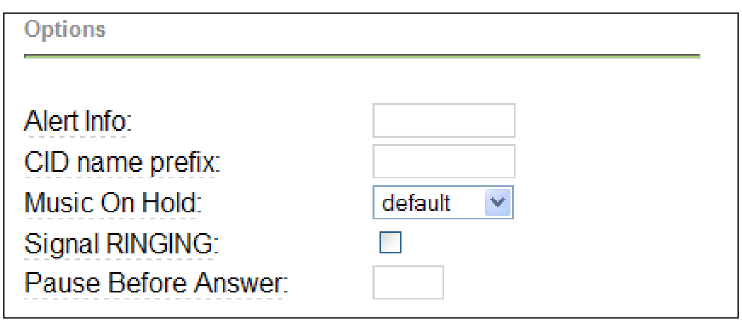
- **Description**: This is a simple text field for you to use to name or describe the new route.
- **DID Number**: If this route is going to be based on the phone number the caller dialed to get to you, then you will enter that number into this field. This will have to match what your phone service provider is sending you, which could be 4, 7, 10, or some other number of digits.
- **Caller ID Number**: If this route is going to be based on the caller ID of the person who called you, then you would enter the complete phone number into this field. In order for this to work, you need to have caller ID service from your telephone service provider.
- **CIF Priority Route** (yes/no): This setting affects CID ONLY routes when there is no DID number set. If this option is selected, then calls with this CID will be sent to this route even if there is a route to the DID that was called.



Options

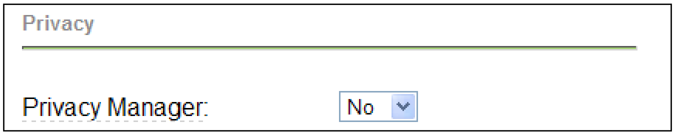
The inbound route options allow you to control some of the behavior of the calls for this particular route; for example, you can change the **CID name prefix** so that the phones will display what department was called, or you can modify the **Alert Info** to create distinctive rings for this route. The options in this section are:

- **Alert Info**: This allows for distinctive rings to SIP devices; you will need to know what **Alert Info** options your particular device supports.
- **CID name prefix**: This will prepend the caller ID name information with whatever text you put into this field.
- **Music On Hold**: This setting allows you to select a playlist category from your music-on-hold selections to play for callers put on hold from this route.
- **Signal RINGING**: This setting is only used with a small number of SIP devices to provide ringing before answer; in almost all cases you do not want to check this option.
- **Pause Before Answer**: This allows you to set a delay before answering in case you have some other equipment that you may want to have pick up the line first.



Privacy

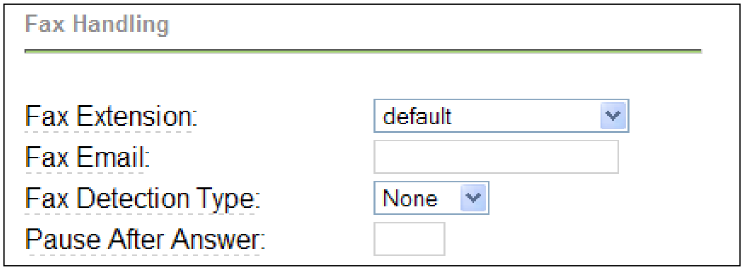
If the **Privacy Manager** is turned on, then all calls to this route will check for the presence of caller ID information. If there is no caller ID, then the system will ask the caller to enter his or her phone number; if the caller fails to enter the number three times, then the system will disconnect him or her. This system should probably not be used anymore and instead the call screening options that we saw earlier in this chapter should be used now.



Fax Handling

As we saw earlier, trixbox is capable of handling inbound faxes; this section sets up some specific fax options for this particular inbound route. The fax related options are:

- **Fax Extension**: Just like in the General Settings, we can select the system to receive faxes, disable faxing, or send the fax calls to a specific extension
- **Fax Email**: If the system is receiving the faxes, then each fax will be converted to a PDF file and emailed to the address specified in this field
- **Fax Detection Type**: Select **None** for no fax detection, **Zap** for Analog or Digital TDM circuits, or **NVFax** for VoIP circuits
- **Pause After Answer**: This allows you to specify a number of seconds to wait for fax tones on the line after answering the call

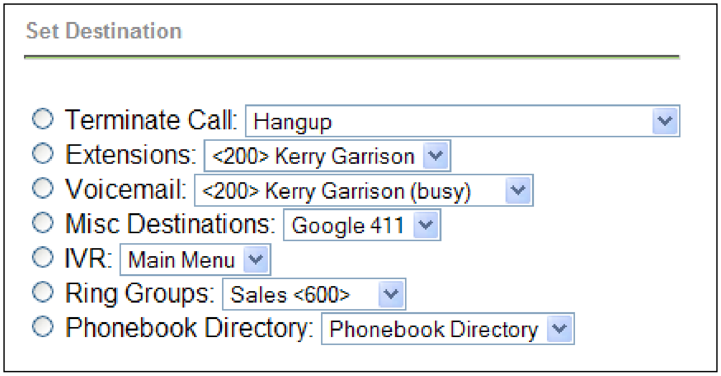


CID lookup source

This function tells the system where to look up caller ID Name information. This will be covered in more detail when we discuss the Caller ID Lookup Sources module later in this chapter.

Set Destination

Once all of the options have been applied, we now need to tell the system what to do with this call. As you will see when we look at other modules, different modules will add items to the list of possible destinations; the example shown here is a fairly basic setup but the actual number of possible destinations can be much larger.



Zap channel DIDs

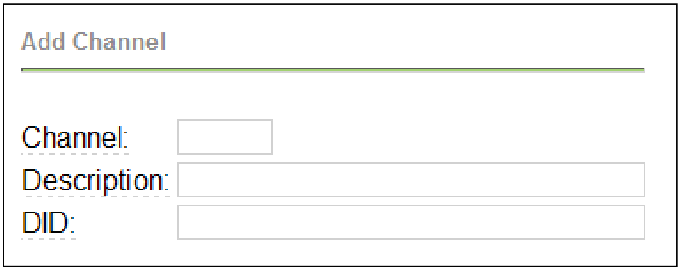
In the case of analog phone lines, there is no actual DID information available to route on; this used to make it very difficult to route calls for different groups of phone lines. With this module, you can now easily assign a particular channel to a DID for use when routing calls via the Inbound Routes module that we just looked at.

The module will tell you some information about making sure that your channel contexts are set to `context=from-zaptel`; you shouldn't need to worry about that because that is the default setting that trixbox sets up when you install your cards.

On a typical analog line card, channel 1 is usually the top port on the card, so if you wanted to assign a DID number to the first channel, we would use 1 as our channel number. The following settings will allow us to set up these analog line channels' DIDs:

- **Channel**—the channel on the card that we want to map to the DID
- **Description**—a description of this DID

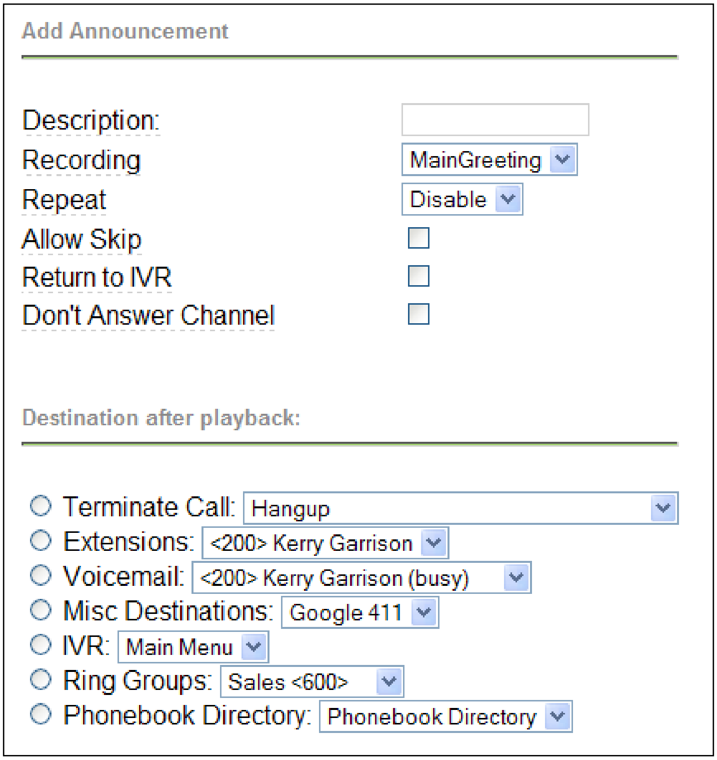
- **DID**—the full phone number that you want mapped to this channel



Announcements

Announcements is a simple tool that is used to play a recorded file to a caller in the IVR menu and then go to another destination. This is most typically used to play office hours or directions to an office and then return back to the IVR menu. This module is divided into a set of options and a destination list to select from after the announcement is played. The available options for announcements are:

- **Description**: This is a text field for entering a description of the announcement.
- **Recording**: This is a pull-down menu that will display the list of available system recordings. We will see how to record new messages when we get to the *System Recordings* module later in this chapter.
- **Repeat**: This allows you to set a keypress that can be used to replay the message for the caller. You would need to add instructions to your recording such as "to hear this message again, please press 9 or wait to be returned to the previous menu".
- **Allow Skip** (yes/no): If this option is checked, then the caller can press a key to skip the message. This is useful for long messages that a caller may hear repeatedly whenever they call in.
- **Return to IVR** (yes/no): If this option is selected, and the announcement was called from an IVR, then the caller will be returned back to the IVR that called it and the destination that is set at the bottom will be ignored.
- **Destination after playback**: This determines where the caller should be sent to after the announcement has been played. This will display a list of all of the available destinations that are currently in the system.

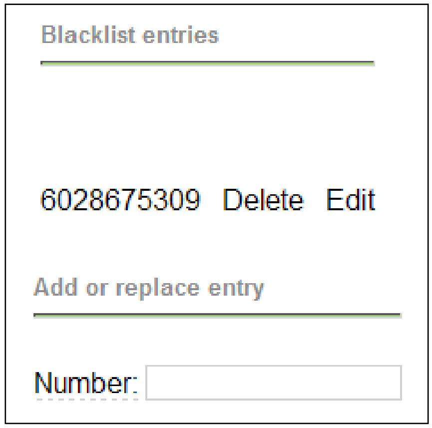


Blacklist

The blacklist is a list of numbers that are blocked from going into the system. If you don't want someone to be able to call you, then you can enter their number into the system; this is great for telemarketers, ex-girlfriends/boyfriends, and crazed groupies. If you have the blacklist module installed, then you can use the module to add, delete, and edit entries, or you can also use the following feature codes:

- *30: blacklist a number
- *32: blacklist the last caller

- *31: remove a number from the blacklist

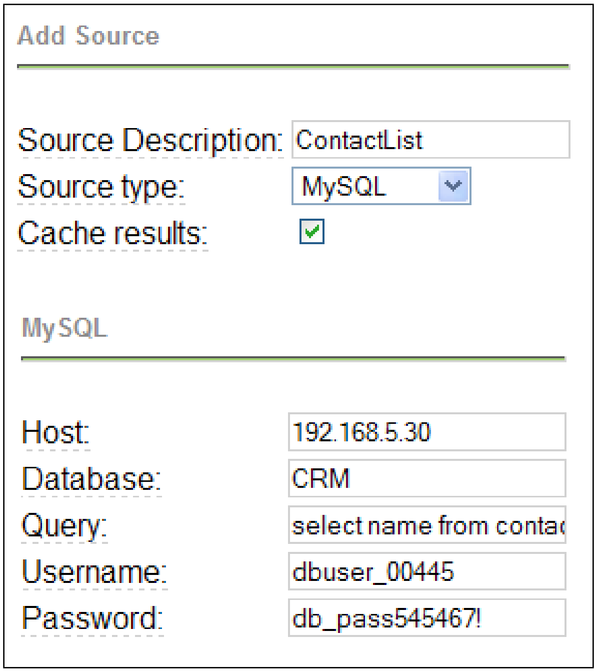


Caller ID lookup sources

With some telephone circuits you may only get the caller ID number and not the caller ID name associated with it; if you have this issue, then you can create different ways for the system to look up a caller's name information. This module allows you to set up different sources. The available options for this module are:

- **Source Description**: This is a text field for creating a description for this source.
- **Source type**: This allows you to choose different source types for the lookup; currently this can be:
    - **Internal**—uses the internal astdb for look ups; this is populated with the phonebook module that we will see later in this chapter.
    - **ENUM**—this will use DNS to lookup caller names. This is typically not very useful.
    - **HTTP**—this will allow you to specify a URL with parameters to query against a web database.
    - **MySQL**—this option will allow you to query against a MySQL database, such as a SugarCRM contact list.

- **Cache Results** (yes/no): If this option is set, then results are cached to the astdb for faster lookups.

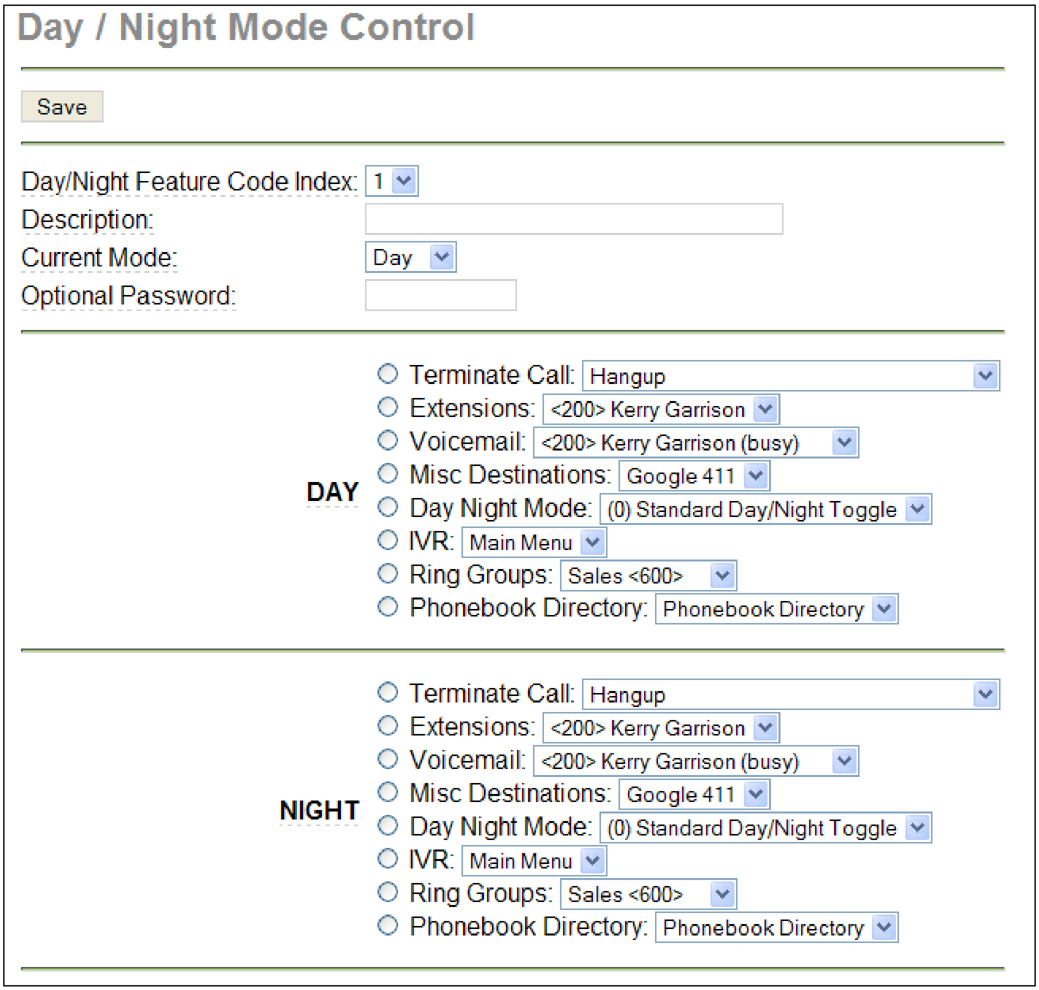


Day/Night Mode Control

While some companies like to automatically switch between day mode and night mode, other companies want a simple way to toggle the system into the different modes. The Day/Night control allows you to create up to ten different configurations that can all be toggled from any phone on the system. To configure Day/Night mode, we need to look at the available features, which include:

- **Day/Night Feature Code Index**: This allows you to set the index number for this particular Day/Night mode set. Once you apply the settings, the system will tell you the feature code to activate it. By default, the first Day/Night control will use ***280** (*28 for the feature code + 0 for the index).
- **Description**: This allows you to set a text description for this Day/Night control.
- **Current Mode**: For testing purposes, you can force a particular Day/Night control into **Day** or **Night** mode setting.

- **Optional Password**: If you don't want anyone to be able to change the settings, you can supply a numeric password that must be entered in order to change the setting.
- **Destinations**: You will be presented with two sets of available destinations so you can set the Day versus the Night behavior of this control.

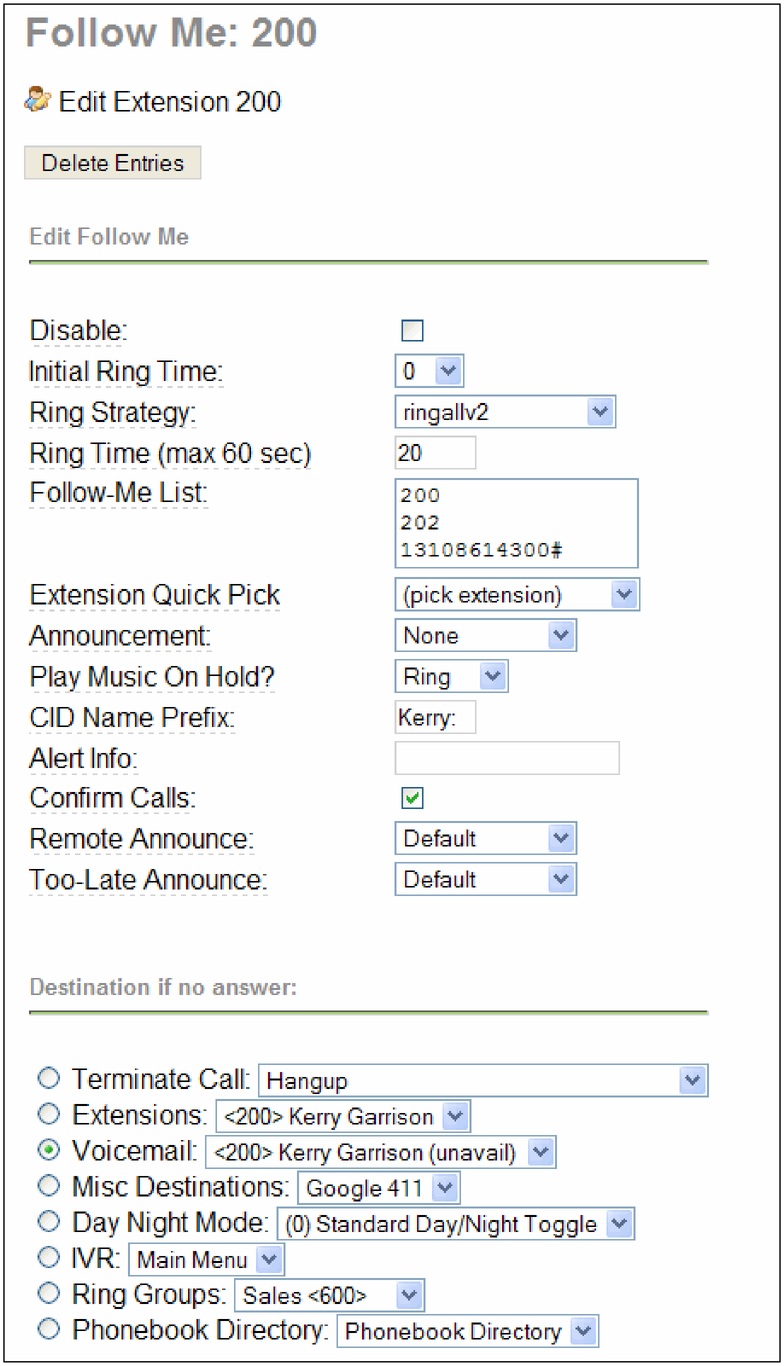


Follow Me

The Follow Me function basically creates a personal ring group for a given extension, which will allow a call to ring in multiple places at the same time when a particular extension is called. By using different ring strategies, you can determine the order that the system will try to find you. To use this feature, you can either select the Follow Me link when editing an extension or select from the list of extensions when you are in the Follow Me module. Let's take a look at all of the options that you have available in the **Follow Me** module.

- **Disable** (yes/no): This simply disables the Follow Me rules if it is selected.
- **Initial Ring Time**: This sets the number of seconds to ring the primary extension before proceeding to the rest of the rules. If you set this number to **0**, then you should be sure to include the primary extension number in the **Follow Me List** or the primary extension will not be called.
- **Ring Strategy**: This option determines how numbers in the Follow Me list are handled. The available options available here are:
    - **RingAllv2**: This will ring the primary extension for the initial ring time followed by all additional extensions until one answers
    - **RingAll**: This will ring all available numbers in the **Follow-Me List** until one answers
    - **Hunt**: This setting will take turns ringing each available extension
    - **Memory Hunt**: This option will ring the first number in the **Follow-Me List**, then it will ring the first and second number, then the first, second, and third number, and so on
    - ***-prim**: If you select any of the available options with the **–prim** extension, then the mode will act as described except if the primary extension is in use, then the other devices will not ring because the system will assume you are at the primary location
    - **FirstAvailable**: This will ring only the first available number in the **Follow-Me List**
    - **First Not On Phone**: This setting will ring only the first channel that is not off hook and will ignore any call waiting setting
- **Ring Time**: This is the time in seconds that the phones will ring. For all of the hunt-style ring strategies, this is the number of seconds for each iteration of the phone(s) that are rung.

- **Follow-Me List**: This is the list of the numbers to ring. You can manually enter extension numbers or use the **Extension Quick Pick** wizard directly below this field. To enter an external phone number, you can enter the phone number followed by a pound sign (for example, 17148675309#). If you are calling out to a cell phone, you will most likely want to enable the **Confirm Calls** option.
- **Announcement**: You can optionally play an announcement before the group of numbers is called. This has to be recorded in the System Recordings module, which limits its usefulness at this time.
- **Play Music On Hold**: This setting determines whether the caller is hearing Music on Hold or ringing while the system is calling the numbers in the **Follow-Me List**.
- **CID Name Prefix**: This allows you to set a prefix to be prepended to the caller ID name field so that you can see on your phone where the call came from. For example, I can put "Kerry:" into this field, and when I get a call from Andrew, my caller ID name field will show "Kerry: Andrew Gillis", so I know that this was a call to my extension.
- **Alert Info**: This allows you to set a distinctive ring for certain SIP devices; you will need to know what options your device supports in order to use this option.
- **Confirm Calls**: If this option is set, then calls to external numbers will require you to answer the phone and push 1 on your phone to accept the call. This feature only works with the ringall/ringall-prim strategies. This is particularly useful for calls going to cell phones as without this feature a call will go straight to your cell phone's voicemail if your phone is not in service.
- **Remote Announce**: If you want a different message played to you when you answer a call from an external number, then you can select a recording from the list of available system prompts.
- **Too-Late Announce**: This message is played to the person receiving the call if the call has already been accepted by another device before they press 1 to accept the call.
- **Destination if no answer**: The list of available system destinations is available for you to choose from to determine where the call should go if there is no answer from any of the numbers in the **Follow-Me List**. Typically, this will go to the unavailable message for the extension this is set up for.


Follow Me: 200
Edit Extension 200
Delete Entries
Edit Follow Me
Disable:
Initial Ring Time:
0
Ring Strategy:
ringallv2
Ring Time (max 60 sec)
20
Follow-Me List:
200
202
13108614300#
Extension Quick Pick
(pick extension)
Announcement:
None
Play Music On Hold?
Ring
CID Name Prefix:
Kerry:
Alert Info:
Confirm Calls:
Remote Announce:
Default
Too-Late Announce:
Default
Destination if no answer:
Terminate Call:
Hangup
Extensions:
<200> Kerry Garrison
Voicemail:
<200> Kerry Garrison (unavail)
Misc Destinations:
Google 411
Day Night Mode:
(0) Standard Day/Night Toggle
IVR:
Main Menu
Ring Groups:
Sales <600>
Phonebook Directory:
Phonebook Directory

IVR (Digital Receptionist)

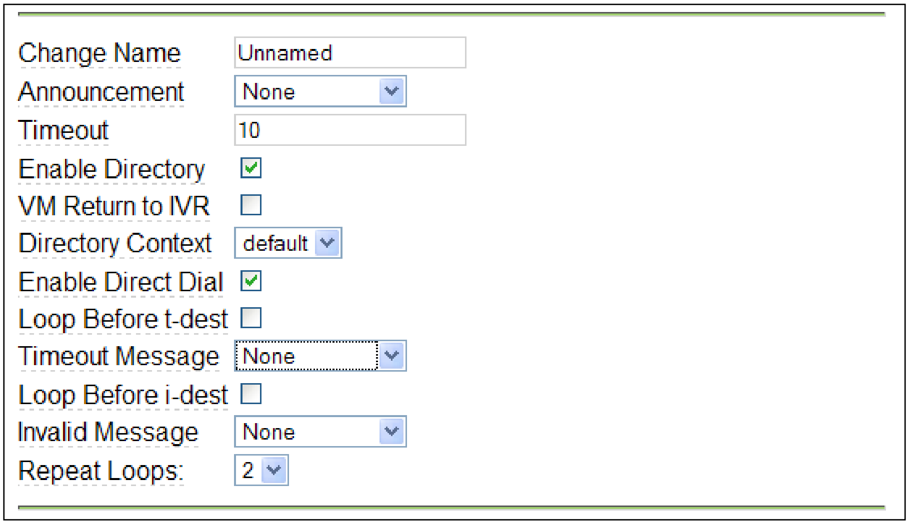
The IVR (Interactive Voice Response) system is the main control for designing the call flow for inbound calls. Getting a good call flow is critical to the successful implementation of a phone system. I consider this so important that Chapter 14 is dedicated entirely to helping you design a good call flow. Since a lot of detail about design is in Chapter 14, this section will simply go over the available options.

IVR Options

The first section of the IVR settings control different aspects of how this particular IVR menu will behave. The available options are:

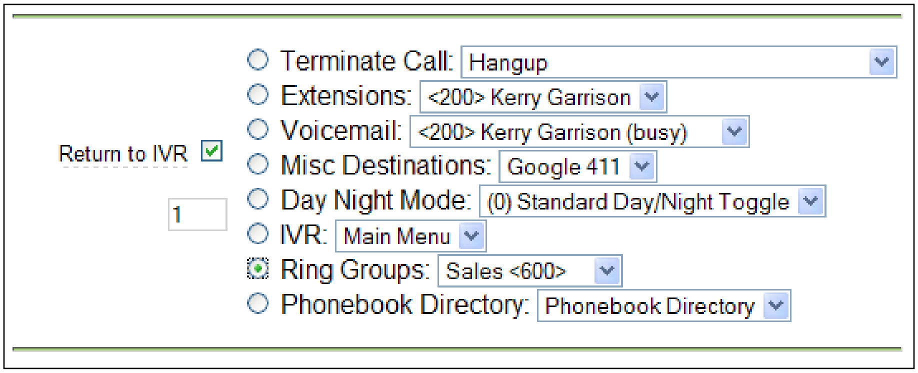
- **Name**: This sets the name of the IVR that will be displayed in the destination lists throughout the system.
- **Announcement**: This allows you to choose from the list of available system recordings to play when a caller is sent to this IVR menu.
- **Timeout**: This is the number of seconds to wait before considering this menu to have timed out. We will see how to route a timed-out call later in this section.
- **Enable Directory** (yes/no): If this option is selected, then the caller can press # to go into the company directory system.
- **VM Return to IVR** (yes/no): If this item is selected, then the caller will be returned to the IVR menu after he or she has left a voicemail.
- **Directory Context**: When the # key is pressed, this will determine the directory context to use. This should only be used by advanced users.
- **Enable Direct Dial** (yes/no): If this item is selected, then callers can directly dial an extension from this IVR menu.
- **Loop Before t-dest** (yes/no): If this item is checked and a timeout ("t") option is set in the keypress options, then the IVR will loop back to the beginning if there is no input. The number of times this will loop is set with the **Repeat Loops** setting.
- **Timeout Message**: This is an optional message from the system recordings that will be played when a timeout occurs.
- **Loop Before i-dest** (yes/no): If this item is checked and a invalid keypress ("i") option is set in the keypress options, then the IVR will loop back to the beginning if there is an invalid input. The number of times this will loop is set with the **Repeat Loops** setting.
- **Invalid Message**: This is an optional message from the system recordings that will be played when an invalid keypress occurs.

- **Repeat Loops**: This is the number of times the system will loop if the **Loop Before t-dest** or **Loop Before i-dest** options are set.



Keypress Options

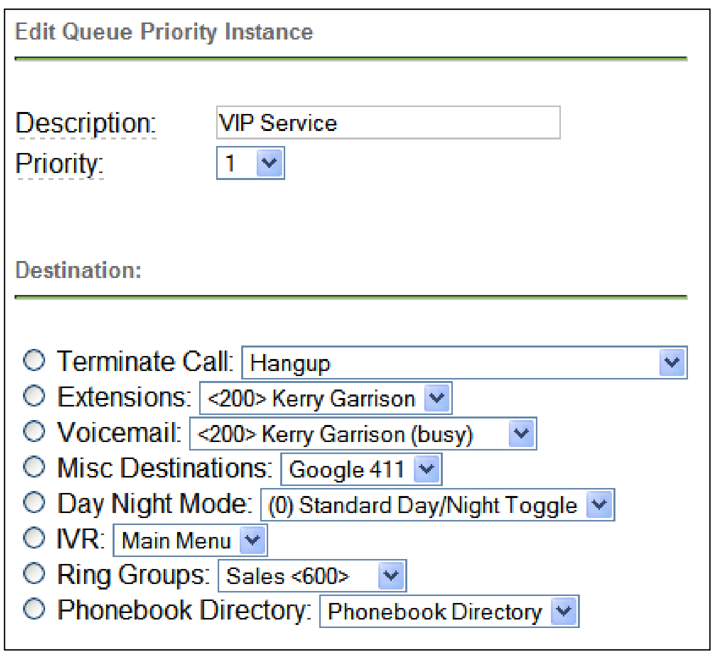
For any given keypress, you can redirect the caller to any available destination within the system. Valid keypress options are 0 – 9, as well as "t" for timeout and "i" for invalid keypress. If you select the checkbox for **Return to IVR**, then the system will ignore any destination selected and will return to the IVR menu that called this particular IVR. This is helpful if you have multiple levels of IVR menus that you are designing. The following screenshot shows a typical configuration that will send callers to the Sales ring group if the **1** key is pressed.



Queue Priorities

Creating queue priorities allows you to preset the priority that a caller will have in any queue that they end up going into. By default, every caller has a priority of **0**. Setting a higher priority level will put that caller ahead of other callers in the queue. Once you create a queue priority, it will appear in the destination lists throughout the system. A common use of this is to provide a priority service for specific customers; if they call in via a particular DID, or if you are using caller ID matching, then you can send their call to the queue priority to have an escalated priority set and then send them into the same queue as other customers—but their call will be answered before those of callers without a priority set.

The only available options here are a textual **Description** for this queue priority rule and a **Priority** setting that allows you to set priorities from **0 – 20**.



Queues

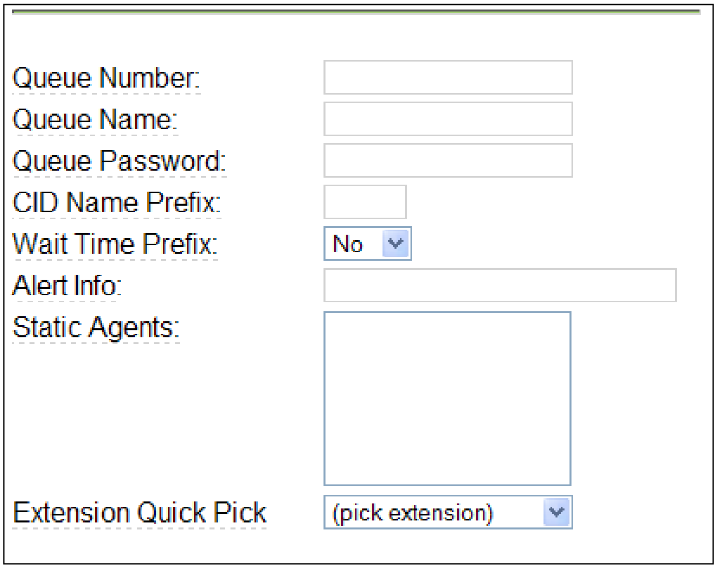
Queues are one of the most powerful features within trixbox CE and are a feature that alone can cost tens of thousands of dollars to purchase for other systems. For inbound call centers, queues are a saving grace as they allow a call center to handle more calls at a time than there are agents available to take calls. This is done by queuing up the inbound calls and releasing them to agents as they become available. This prevents callers from getting busy signals or having to leave messages for a possible callback. Of course, the system can only take as many queue calls as you have available lines. The queue settings are broken up into several sections, and we will look at the different sets of options one set at a time.

Queue Settings

The basic queue settings define the queue itself and the agents that are assigned to the queue. The basic settings consist of:

- **Queue Number**: This number is an extension number that you will use to access this queue. Normally, calls are routed to the queue from the IVR, so you may never need to use this number directly unless an operator is transferring a call into the queue. It is recommended that the queue number be well outside the normal set of extension numbers to prevent confusion, such as using a four digit number for queues if you are using three digit extension numbers or vice versa.
- **Queue Name**: This is just a text field for you to enter a textual description of the queue, such as "Support".
- **Queue Password**: If you want agents to have to log into the queue, you can optionally set a password for them to enter before being logged into the queue.
- **CID Name Prefix**: As we have seen before, we can change the caller ID name prefix so that agents can visually see on their phone which queue this call came in from.
- **Wait Time Prefix** (**Yes/No**): If this option is set, then the number of minutes that the caller has been waiting in the queue will be prepended to the caller ID name so that the agent can see how long the caller has been waiting.
- **Alert Info**: If your SIP device supports **Alert Info** to enable a distinctive ring, then you can set this option; you will need to know which settings your particular device can support.

- **Static Agents**: Extensions listed in the **Static Agents** field will be assumed to always be "in the queue" and the agent will not need to log into and out of the queue.



Queue Options

The queue options determine how calls are handled by the queue and set rules for how calls can come into and out of the queue.

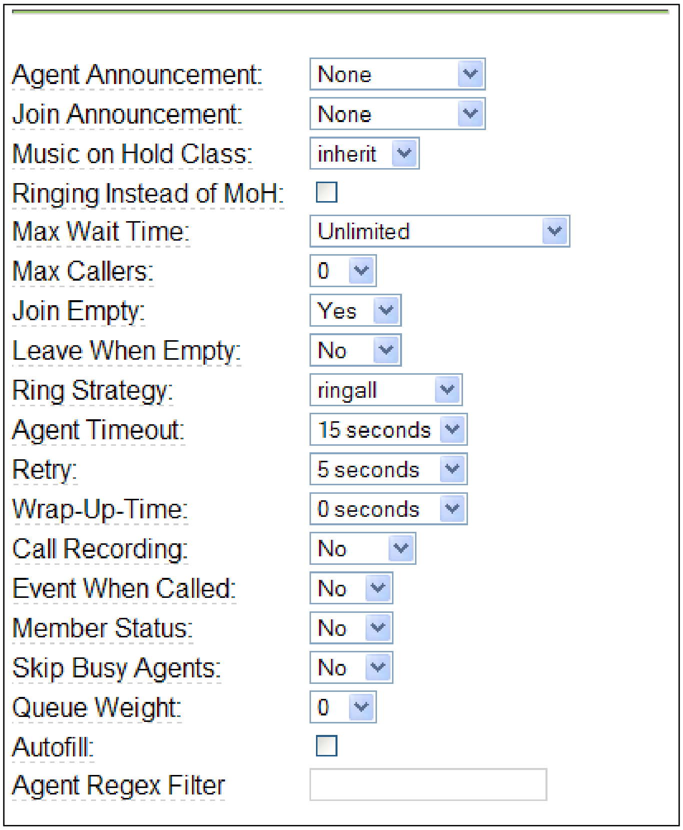
- **Agent Announcement**: This is an optional announcement that is played to the agent before bridging the call. This can be useful for agents that belong to multiple queues so that a message such as "transferring a support call" can be played. This list is populated from the available recordings from the System Recordings module.
- **Join Announcement**: This is an optional announcement that is played to the caller upon joining the queue. This is often used to play a message like "You have reached the support department; your call is very important to us. Your call will be answered in the order that it was received." This list is populated from the available recordings from the System Recordings module.
- **Music on Hold Class**: This setting allows you to play different music-on-hold lists to callers in this queue. If this is set to **inherit**, then the music-on-hold setting will be inherited from an upstream setting such as the inbound route.

- **Ringing Instead of MoH** (yes/no): If this option is selected, then the caller will hear ringing instead of music on hold. It is not recommended to play a ringing sound as callers will typically hang up after hearing a ringing sound more than a few times.
- **Max Wait Time**: This option sets the maximum time that a caller can wait in the queue before the call is considered to be timed out and sent to the **Destination if no answer**.
- **Max Callers**: If you want to limit the number of callers that can be in a particular queue, then you can set this option. A setting of **0** will allow an unlimited number of callers to join the queue.
- **Join Empty**: This setting controls whether or not callers can be sent into the queue if there are no agents signed in. The following options are available:
    - **Yes**—callers can be sent into the queue even if there are no agents available to take the call
    - **No**—callers will not be sent into the queue if there are no available agents
    - **Strict**—callers will not be sent into the queue if the queue has no static agents or if the agents are all unavailable
- **Leave When Empty**: This setting controls whether or not callers will be pulled out of the queue if there are no agents signed in. The following options are available:
    - **Yes**—callers will be pulled from the queue if there are no agents available to take the call
    - **No**—callers will be left in the queue even if there are no available agents
    - **Strict**—callers will be pulled from the queue as soon as the queue has no static agents or if the agents are all unavailable
- **Ring Strategy**: This setting controls how calls are handed out to the available agents. The different ring strategies that are available are:
    - **Ring All**: This setting will ring all available agents until an agent answers the call
    - **Round Robin**: This option will take turns ringing each available agent until the call is answered
    - **Least Recent**: This setting will ring the agent who was least recently called by this queue
    - **Fewest Calls**: This setting will ring the agent who has the fewest completed calls from this queue

- **Random**: This option will randomly ring an agent from the list of available agents
- **Round Robin w/Memory**: This is similar to Round Robin except it remembers where it left off on the last ring pass so that it doesn't start from the beginning each time

- **Agent Timeout**: This sets the amount of time that an agent's phone will ring before it is considered to be timed out. Other timeout values, such as system ringtime or extension settings, may affect this setting.
- **Retry**: This setting controls the amount of time to wait before retrying the agent's phones again. If you set this option to "No Retry" then the system will only take one pass through the list of agents before sending the call to the failover destination.
- **Wrap-Up Time**: If your agents need time to close tickets, or do other paperwork at the end of each call, then you may want to set this to an appropriate value to allow the agent to wrap up from the previous call before another call is released to them.
- **Call Recording**: If you want all calls to this queue to be recorded, then you can change this value to one of the recording formats that are available; these include wav49, WAV, and GSM.
- **Event When Called** (**Yes/No**): If this setting is set to **Yes**, then Asterisk will generate AgentCalled, AgentDump, AgentConnect, and AgentComplete manager events. This is useful for certain third-party tools to use that monitor queue activity.
- **Member Status** (**Yes/No**): If this setting is set to **Yes**, then Asterisk will generate QueueMemberStatus events.
- **Skip Busy Agents** (**Yes/No**): Setting this option to **Yes** will bypass agents whose phones are showing a busy status. Even if a phone has call waiting or is a multi-line phone, the call will not be sent to the phone if a call is in progress.
- **Queue Weight**: This option gives the queue a "weight" option to ensure that a higher priority queue will deliver its calls to agents first if there are agents that are members of more than one queue.
- **Autofill**: If you are using trixbox CE **2.4** or newer, then it is recommended that you enable this option so that if there are multiple agents available, the system can send a call to each waiting agent instead of keeping other calls on hold while the first call in the queue is being processed.

- **Agent Regex Filter**: This is a very advanced feature that can be used to filter against the agent callback feature. This should only be used by very advanced users; otherwise it should be left blank.



Call Announcements

While a caller is waiting in the queue, you have the option of playing different types of announcements to the caller on a regular basis. The most common type is **Caller Position Announcements** that can tell the caller his or her position in the queue and/or the estimated hold time.

Caller Position Announcements

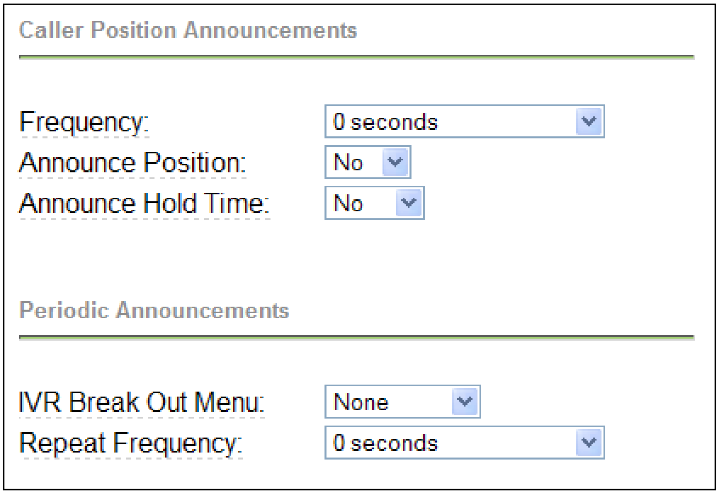
You can set the frequency and hold times for callers into the queue with the following settings:

- **Frequency**: This setting controls how often the position announcements are played to the caller
- **Announce Position** (**Yes**/**No**): If this option is set to **Yes**, then the caller's position in the queue will be played at the frequency set in the previous option
- **Announce Hold Time**: You can set this option to **Yes**, **No**, or **Only Once** to control when the caller hears what the estimate hold time is

Periodic Announcements

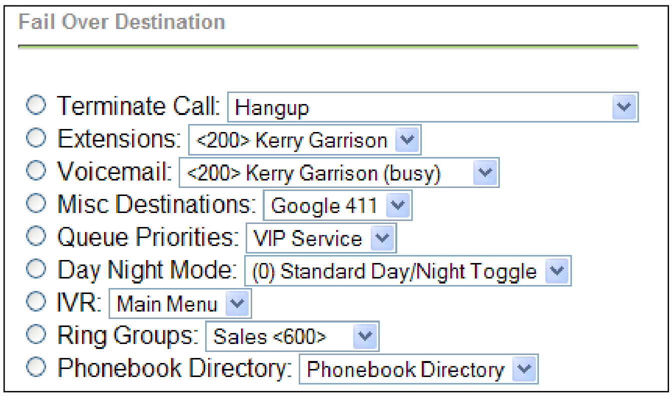
At different points, you may want to give the caller the option to break out of the queue and perform some other action like leaving a voicemail message. You can set how often this option is played to the caller and what IVR menu to present to the caller with the following settings:

- **IVR Break Out Menu**: You can set an optional IVR menu to be a "break out" menu. This will allow the caller to break out of the queue using an existing IVR menu to navigate. The catch here is that the IVR must only contain single-digit keypress options. The announcement that is assigned to the IVR menu will be played with the **Repeat Frequency** set in the next option.
- **Repeat Frequency**: This sets how often the IVR announcement from the previous setting is played.



Fail Over Destination

If the call times out, or matches some other rule that pulls it out of the queue, then the call will go to the fail over destination. This is a list of the available destinations within the system.



Ring groups

While queues are best used when there are more inbound callers than there are agents available to take the call, ring groups are best used when there are more agents available to take calls than there are inbound calls coming in. Unlike a queue, a ring group will not stack up inbound calls; it will simply ring the devices that are listed in the group until the time limit that is set and then send the call to another destination. The available options for ring groups are:

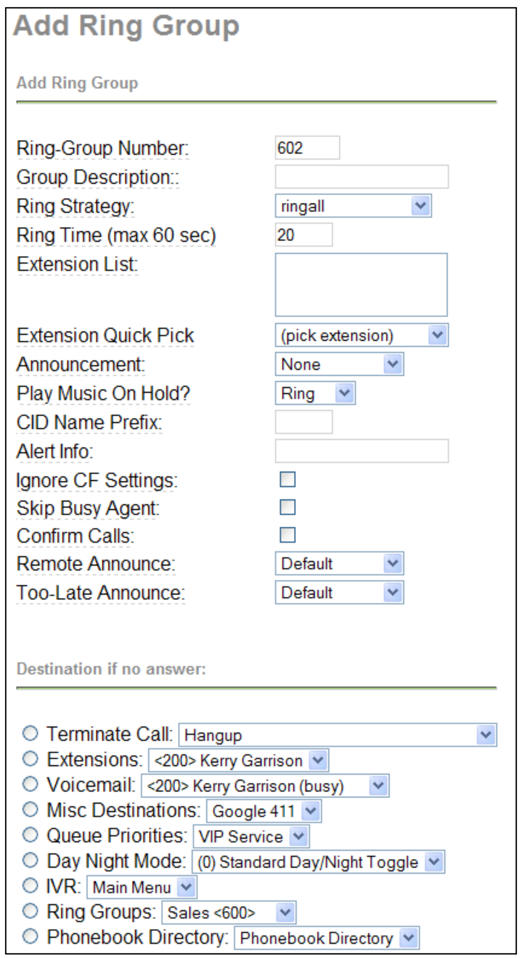
- **Ring Group Number**: This is the extension number of the ring group and can be used to transfer a call to the group from a regular phone. It is recommended that ring group numbers are kept well apart from normal extension numbering plans in order to avoid confusion.
- **Group Description**: As we have seen in many other modules, the description field is a text string that allows you to have a more readable name associated with this particular group.
- **Ring Strategy**: This option determines how numbers in the Follow-Me List are handled. The options available here are:
    - **RingAll**: This will ring all available numbers in the Follow-Me List until one answers
    - **Hunt**: This setting will take turns ringing each available extension
    - **Memory Hunt**: This option will ring the first number in the Follow-Me List, then it will ring the first and second number, then the first, second, and third number, and so on

    - ***-prim**: If you select any of the available options with the **–prim** extension, then the mode will act as described except if the primary extension is in use, then the other devices will not ring because the system will assume you are at the primary location
    - **FirstAvailable**: This will ring only the first available number in the Follow-Me List
    - **First Not On Phone**: This setting will ring only the first channel that is not off hook and will ignore any call waiting setting

- **Ring Time**: This is the time in seconds that the phones will ring. For all of the hunt-style ring strategies, this is the number of seconds for each iteration of the phone(s) that are rung.
- **Extension List**: This is the list of the numbers to ring. You can manually enter extension numbers or use the **Extension Quick Pick** wizard directly below this field. To enter an external phone number, you can enter the phone number followed by a pound sign (for example, 17148675309#). If you are calling out to a cell phone, you will most likely want to enable the **Confirm Calls** option.
- **Announcement**: You can optionally play an announcement before the group of numbers is called. This has to be recorded in the System Recordings module, which limits its usefulness at this time.
- **Play Music On Hold**: This setting determines whether the caller is hearing Music on Hold or ringing while the system is calling the numbers in the Follow-Me List.
- **CID Name Prefix**: This allows you to set a prefix to be prepended to the caller ID name field so that you can see on your phone where the call came from. For example, I can put "Kerry:" into this field, and when I get a call from Andrew, my caller ID name field will show "Kerry: Andrew Gillis", so I know that this was a call to my extension.
- **Alert Info**: This allows you to set a distinctive ring for certain SIP devices; you will need to know what options your device supports in order to use this option.
- **Ignore CF Settings**: If this setting is checked, then the devices that are set for call forwarding will be called regardless of the setting.
- **Skip Busy Agent**: When this is checked, any extension in the list that is on a call will be skipped; this applies even if the device has call waiting or is a multi-line phone.
- **Confirm Calls**: If this option is set, then calls to external numbers will require you to answer the phone and push *1* on your phone to accept the

call. This feature only works with the ringall/ringall-prim strategies. This is particularly useful for calls going to cell phones as without this feature a call will go straight to your cell phone's voicemail if your phone is not in service.

- **Remote Announce**: If you want a different message played to you when you answer a call from an external number, then you can select a recording from the list of available system prompts.
- **Too-Late Announce**: This message is played to the person receiving the call if the call has already been accepted by another device before they press *1* to accept the call.
- **Destination if no answer**: The list of available system destinations is available for you to choose from to determine where the call should go if there is no answer from any of the numbers in the Follow-Me List. Typically, it will go to the unavailable message for the extension this is set up for.



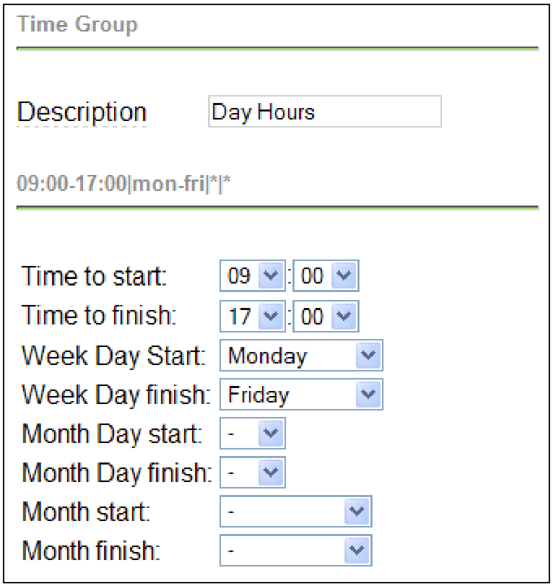
Time conditions

The use of time conditions allows you to have control of the call flow based on the time of day or even the day of the week. This is most often used for playing a different message during non-business hours or special messages when the office is closed for holidays. The setup of time conditions works across two modules, the *Time Conditions* and the *Time Groups* modules; because of that, we will look at both of these modules at the same time.

Creating a time schedule

The first thing we probably want to do is to create a time schedule, which will allow setting the normal office hours versus the non-office hours. To demonstrate how this works, let's think of a company that wants a daytime message played from 9am – 5pm, Monday through Friday, and a different message played when the office is closed in the evenings and at weekends.

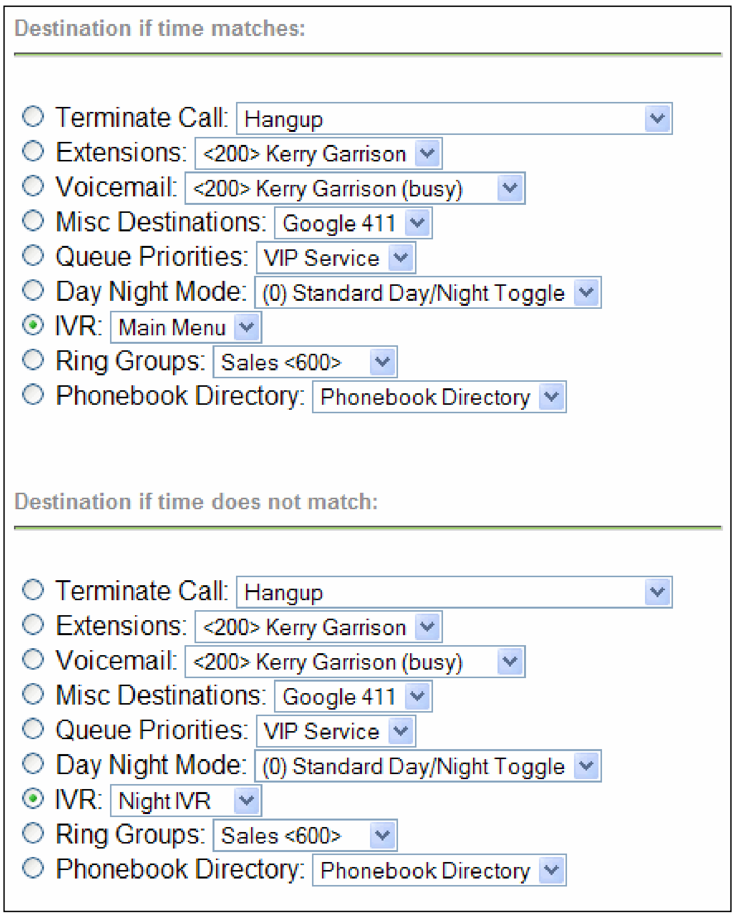
We first need to go to the **Time Groups** module and create a rule set. The set we will use will define the hours when the office is open. We will set the start time to 09:00 and the finish time to 17:00 and this rule will run Monday through Friday. This is shown in the following image:



Once we have the time rule configured, we can then go back to the Time Conditions module to set up how the rule will be used.

Once we are back in the **Time Conditions** module, we need to create a new condition. We start off by giving the condition a name; in this case we will use **Office Hours**, and we will select the **Day Hours** group from the Time Group pull-down menu. Optionally, we could associate this with a Day/Night toggle, which we looked at earlier in this chapter, but for simplicity's sake we will just define how this condition will work.

Since we have a very simple rule that defines what the office hours are, we can select a destination to go to if the time condition matches, such as our main IVR. If the time condition fails, which means the time is outside of the regular office hours, then we can route the call to the night-time IVR menu.



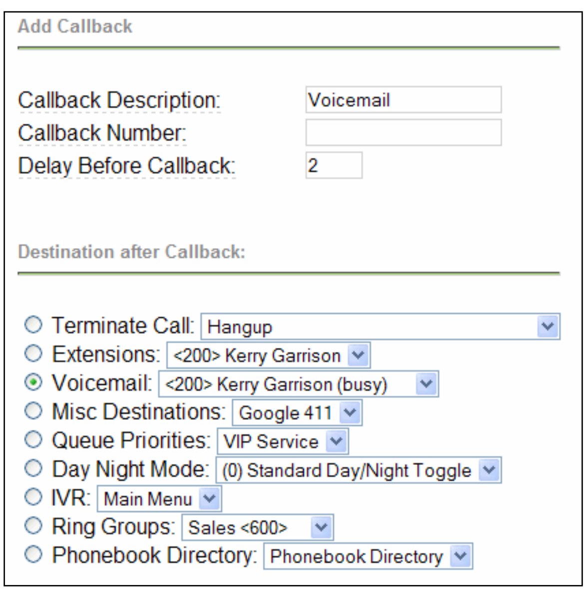
Internal options and configuration

The settings in this section make up different tools and options that can be configured. These include tools like Callback, Music On Hold, Conference Rooms, Paging, Intercom, and several others. Let's take a look at all of the options that are available here.

Callback

The Callback feature is often used by sales people who need to make outbound calls but want the call to be placed by the office phone system, or to check voicemail to help reduce their costs, especially if the caller's cell phone has free inbound calling on it. To configure Callback, we need to set up the following options:

- **Callback Description**: Again, this is just a text description to define this setting.
- **Callback Number**: You can enter a specific number to call the caller back at; this setting is usually used when trying to increase security and prevent other callers from using Callback. If this field is left blank, then the caller ID of the caller will be used for the Callback.
- **Delay Before Callback**: This sets the number of seconds to wait before calling the person back.
- **Destination After Callback**: This sets the destination to send the call to after the Callback is successful.



Conferences

Earlier we looked at the Web MeetMe system, which is a powerful add-on to manage scheduled conferences. If we just want to have some static conference rooms, then the Conferences manager is a much easier way to accomplish this task.

Conference rooms have a set of basic options that define the conference room and a second set of options that control how the conference room functions.

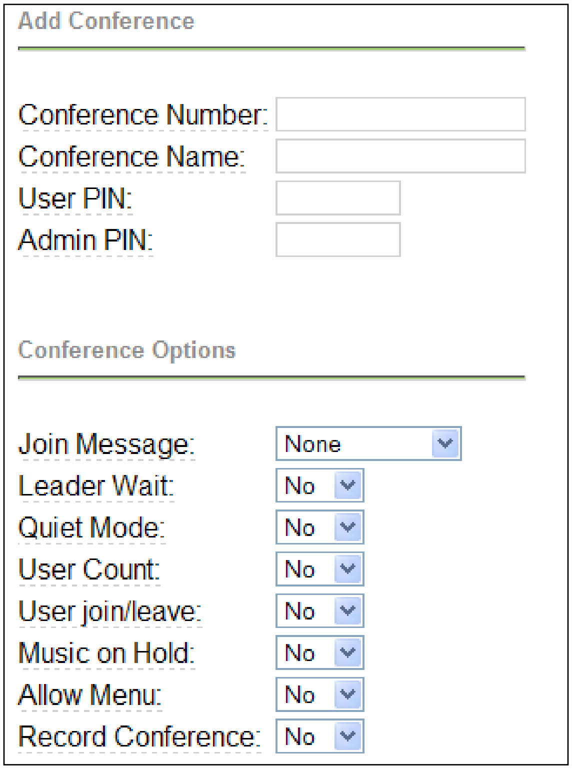
To add a conference, we need to set the following options:

- **Conference Number**: This is the extension number of the conference that can be used to transfer calls into the conference room
- **Conference Name**: This is a text description of the conference room
- **User PIN**: If you want to password-protect the conference from callers being able to join it, you can optionally set a pin code to be used in order to enter the conference room
- **Admin PIN**: You can also optionally set an Admin PIN if you want someone to be able to do the admin functions within the conference

Each conference room can have its own set of options based on how you want the conference room to behave; the options that are available are:

- **Join Message**: This is a message that will be played to callers who join the conference. This may be something like "Welcome to the weekly sales training conference call, please stand by until your host joins the conference." These messages are added via the System Recordings module.
- **Leader Wait** (**Yes**/**No**): If this is set to **Yes**, the conference will not begin until an admin user enters the conference room.
- **Quiet Mode** (**Yes**/**No**): If this is set to **Yes**, then the system will not play entry and exit tones.
- **User Count** (**Yes**/**No**): If this is set to **Yes**, then the user count will be played when a user joins the conference.
- **User join/leave** (**Yes**/**No**): If this is set to **Yes**, then the system will announce when a caller joins or leaves the conference.
- **Music on Hold** (**Yes**/**No**): If this is set to **Yes**, then music on hold will play if there is only a single user in the conference room.
- **Allow menu** (**Yes**/**No**): If this is set to **Yes**, then a menu will be presented to the caller if he or she presses the * key.

- **Record Conference** (**Yes/No**): If this is set to **Yes**, then this conference room will always be recorded.



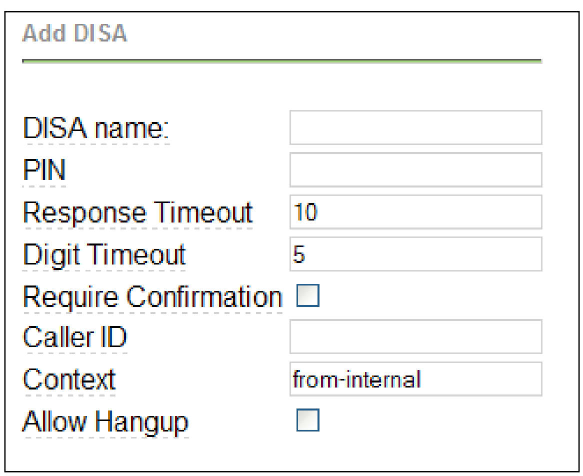
DISA

DISA (**Direct Inward System Access**) allows an external caller to obtain an "internal" system dialtone and dial calls as if he or she were connected to one of the extensions attached to the phone system. The typical setup is for the caller to call into a DID or an IVR menu that puts the caller into DISA mode; the DISA application in turn requires the user to enter his or her passcode, followed by the pound sign (#). If the passcode is correct, the user will hear dialtone telling the user he or she is connected and ready to make calls. It goes without saying that this can be a major security hole and great care should be taken to ensure that it is configured properly.

One good example of when to use DISA came up when my wife was in the hospital; the hospital phone would only allow local calls, and my wife wanted the ability to call her family, which were all long distance calls. By using DISA, my wife was able to call into our home phone system, which was a local call, enter in her code to get dialtone, and then make her outbound calls at will.

The following settings are used to configure DISA:

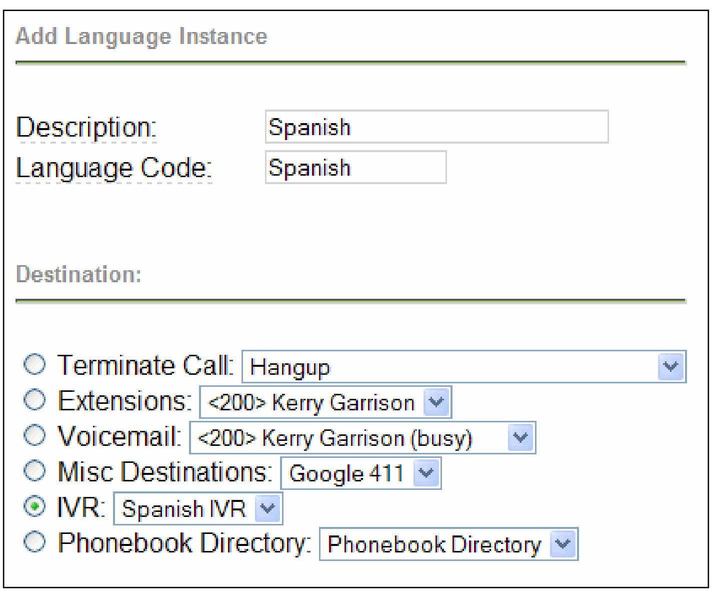
- **DISA name**: This is a simple description used to identify this DISA configuration.
- **PIN**: This is the password used to authenticate the caller. Multiple PIN codes can be entered into this field by separating them with commas.
- **Response Timeout**: This sets the maximum number of seconds to wait between digits when the user is dialing an extension.
- **Require Confirmation** (yes/no): If this is selected, then confirmation will be required before entering the PIN code. This is used if your PSTN connection appears to answer the call immediately.
- **Caller ID**: This is an optional field that can be used to set the outbound caller ID for someone using this DISA configuration. This field uses the standard **Caller ID** format of *User Name* <*xxxxxxx*>.
- **Context**: This setting allows you to change the context that calls will originate from. You will almost always want to leave this at the default setting unless you are an advanced user and understand what changes to this will do.
- **Allow Hangup** (yes/no): If this setting is turned on then a caller can press the * key to hang up the current call and get back to the system dialtone.



Languages

If you wanted to have a multi-language system with complete sets of voice prompts in different languages, it used to be next to impossible to set up. With the new Languages module, this is fairly simple. Let's say you wanted to have English and Spanish prompts; your initial IVR menu could say "For English press 1, para el español, prensa dos". If the caller presses 1, then the system continues normally; if the caller presses 2, then the IVR would send the call to a language destination that sets the system to Spanish. In order for this to work, you would already have to have a set of Spanish prompts installed on your system as only English prompts are included with the base installation. Let's assume that you got a set of Spanish prompts and installed them into the `/var/lib/asterisk/sounds/Spanish` folder. We could then set up the Languages module as follows:

- **Description**: This is a text description for this language set; for our example we could set this to **Spanish**.
- **Language Code**: This is going to match the language folder where we installed the language files; in our case this would be **Spanish**.
- **Destination**: This is the destination to go to after setting the language code; for our example, we are going to send calls to a Spanish IVR tree.

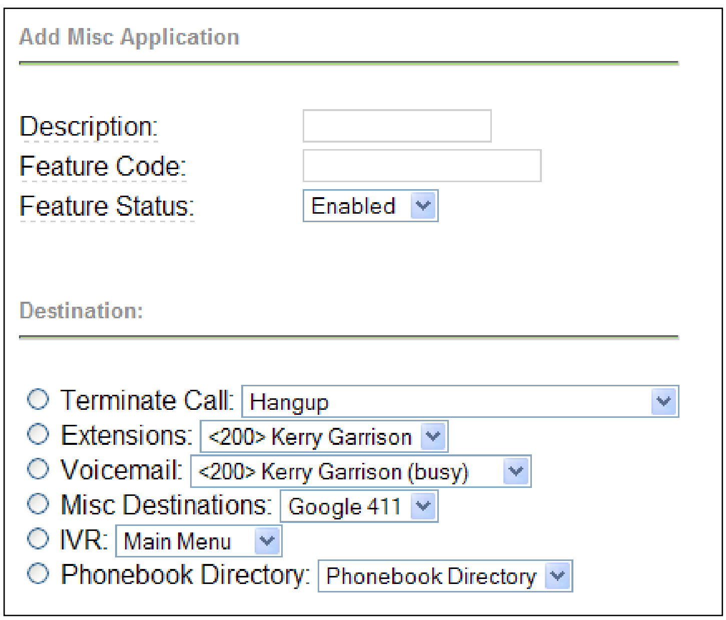


Misc Application

Misc Applications are for adding feature codes that you can dial from internal phones that go to various destinations available within trixbox. We took a look at using Misc Applications and Misc Destinations in Chapter 9 when we set up 411 to dial a free directory service.

Options for Misc Applications are:

- **Description**: This is a description of the Misc Application you are creating
- **Feature Code**: This is the number you are going to dial to access this application
- **Feature Status**: This sets whether this feature is enabled or disabled
- **Destination**: This sets where the caller will be sent when dialling this feature code

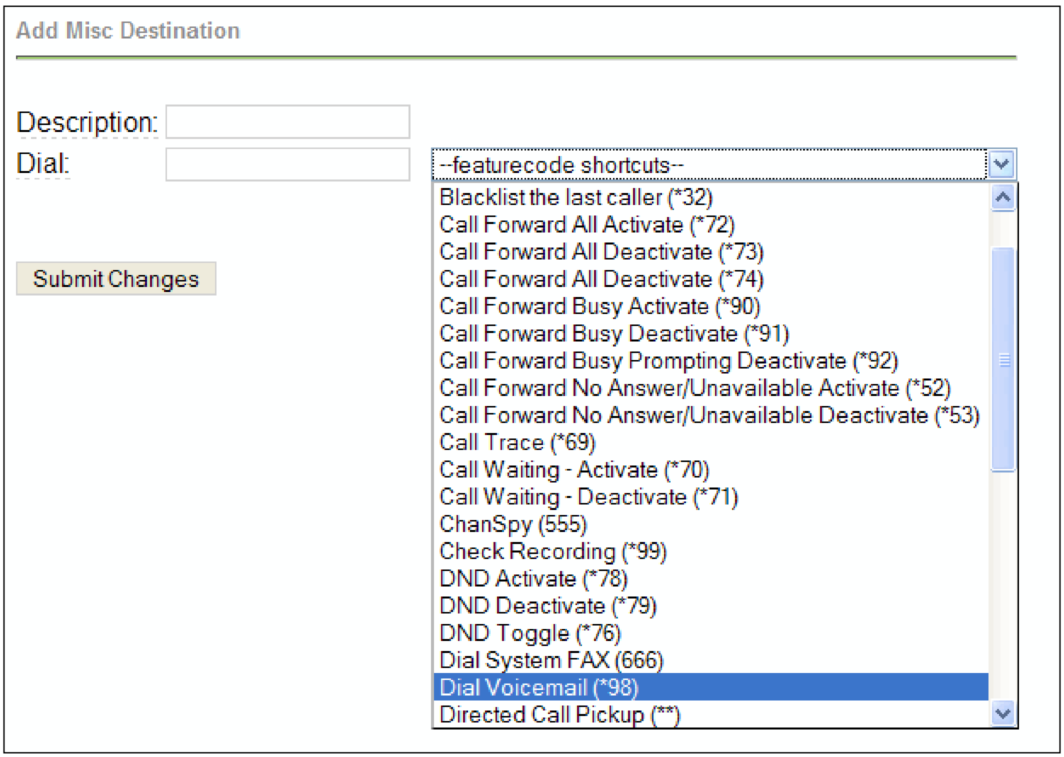


Misc Destinations

This module allows you to create destinations that will appear in the destinations list of the different modules as we have seen throughout this chapter. In Chapter 9, we saw how we could create a destination that would dial the Google directory service. Another common usage is to create a destination that will dial into voicemail; you could then use this destination from either an IVR option or a DID number for faster access to the voicemail system for remote users.

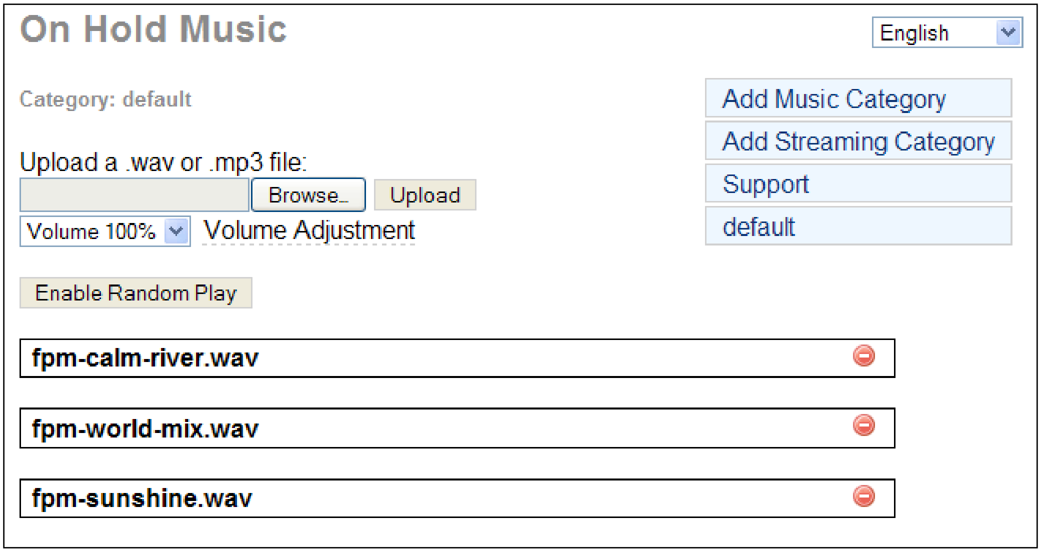
The available options for Misc Destinations are:

- **Description**: This is a text field for giving the destination a friendly or useful name
- **Dial**: This can either be a phone number, an extension, or be selected from the **featurecode shortcuts** pull-down menu



Music On Hold

This module allows you to create and define "playlists" of music-on-hold files that can be utilized throughout the system. The Music On Hold module allows you to upload MP3 files and to create different categories; this allows you to mix music and custom messages to different people. For example, when people are holding in a sales queue, you could play music for a while and then play an advertisement for a special you are having, while in the support queue the message could be how to solve the top five issues people have.

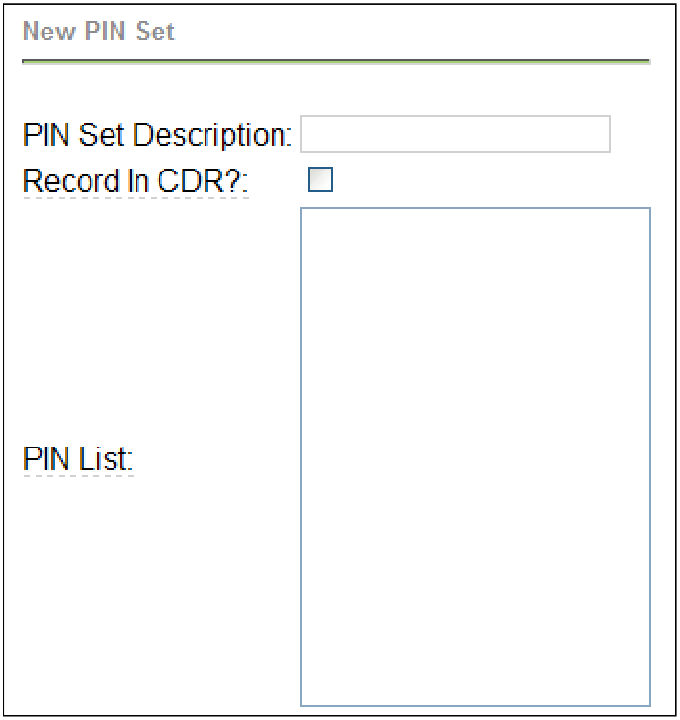


PIN Sets

Within the system, different modules have the option to use passwords or PINs in order to access their functionality. This can be found, for example, with outbound routes. If a PIN code is set for that outbound route, then the caller will need to enter the PIN number in order to complete the call. The available options for PIN Sets are:

- **PIN Set Description**: This is a text description of this particular set of PIN codes.
- **Record in CDR?** (yes/no): If this is set to yes, then the PIN code will be used in the CDR reports. This is handy for billing purposes.

- **PIN List**: This is the list of available PIN numbers assigned to this PIN Set.



Paging and intercom

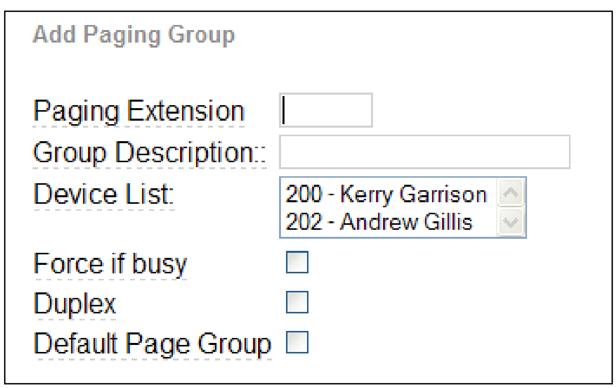
Paging allows you to broadcast a message to a set of phones via the speakerphone on the device. Intercom is almost the same but allows the called party to talk back to the caller.

Example usage:

- ***80nnn**—intercom extension nnn
- ***54**—enable all extensions to intercom you (except those explicitly denied)
- ***54nnn**—explicitly allow extension nnn to intercom you (even if others are disabled)
- ***55**—disable all extensions from intercom you (except those explicitly allowed)
- ***55nnn**—explicitly deny extension nnn to intercom you (even if generally enabled)

The available options while creating paging and intercom groups are:

- **Paging Extension**: This is the number to dial when you want to call this particular group.
- **Group Description**: This is a text description of this paging group.
- **Device List**: This is the list of devices to be included in the paging group; hold down the control key and click on the different devices to select them for inclusion.
- **Force if busy** (yes/no): If this option is set to yes, then the devices will not be checked to see if they are in use. Depending on the device, this could interrupt the call, or some devices will put a call in process on hold in order to process the page. This feature is only recommended to be turned on when using this paging group for emergencies.
- **Duplex** (yes/no): If this is set to yes, the paging group becomes an intercom group and all parties will be joined into a conference.
- **Default Page Group** (yes/no): Each system is allowed a single default paging group. If this is selected, then extensions can be automatically added to this group when creating the extension. If you make one group the default, it will remove the default setting from any previous group that you had set as the default.



Parking Lot

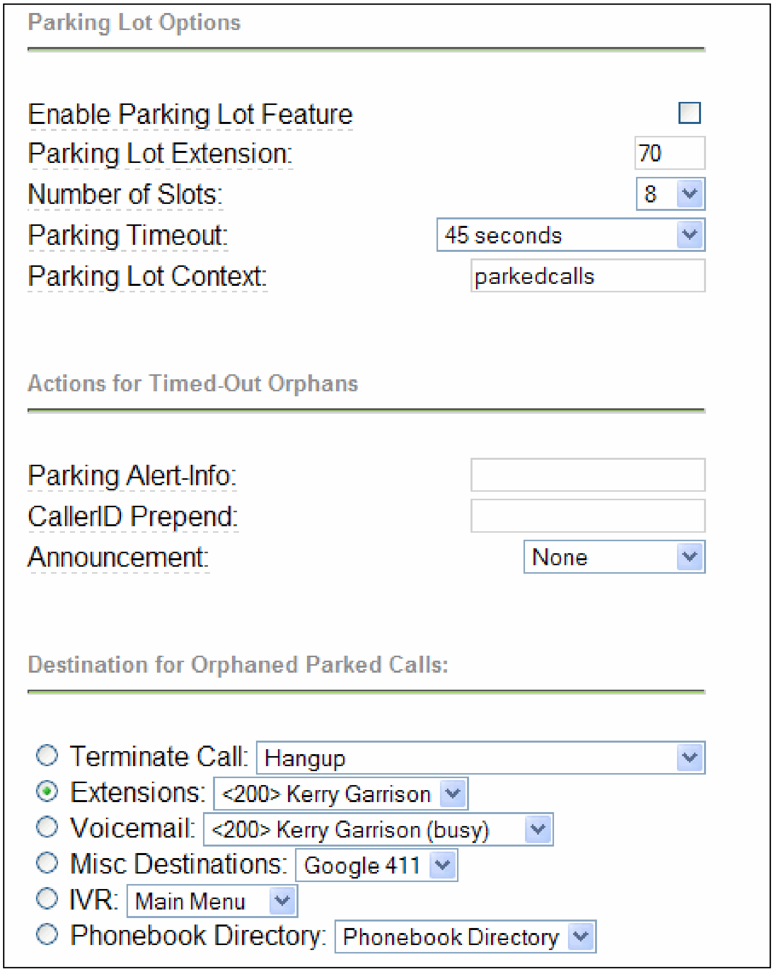
Older phone systems tied all of the incoming lines together to be shown on each phone; this allowed a user to place the call on hold and then yell out "Joe, you have a call on line 2". With a trixbox CE system, you can have so many lines that this simply isn't practical, so instead of putting the line on hold, you park the call and tell Joe what parking lot number the call is in. In some other systems, this feature is referred to as Orbit.

The Parking Lot module options are as follows:

- **Enable Parking Lot** (yes/no): This setting simply enables or disables the parking lot feature.
- **Parking Lot Extension**: This is the number you will transfer the calls to in order to park them.
- **Number of slots**: This setting determines how many parking lot slots you have available. If you have eight slots available, then calls will be placed in slots 71 – 79.
- **Parking Timeout**: This setting determines the amount of time to leave a call parked before timing out and returning the call to the extension that parked the call.
- **Parking Lot Context**: This is useful when you have multiple companies sharing the same system but should not be changed unless you know exactly what the ramifications of doing so are.

Along with the standard options are a set of options to apply to calls that time out after being parked. This is useful for knowing that a call that is ringing back to you is returning after being parked too long. These options are:

- **Parking Alert-Info**: This setting can be used to set a distinctive ring for many SIP devices; you will need to know what settings your particular device supports.
- **CallerID Prepend**: This allows you to prepend some text to the caller ID to visually show that this call is an orphaned parked call.
- **Announcement**: This is an optional message that can be played to the caller when his or her call has timed out and before going to the destination for orphaned calls.
- **Destination for Orphaned Parked Calls**: You may want orphaned calls to go to a different destination than the original parker; typically, you may want them to go to a receptionist for further processing.

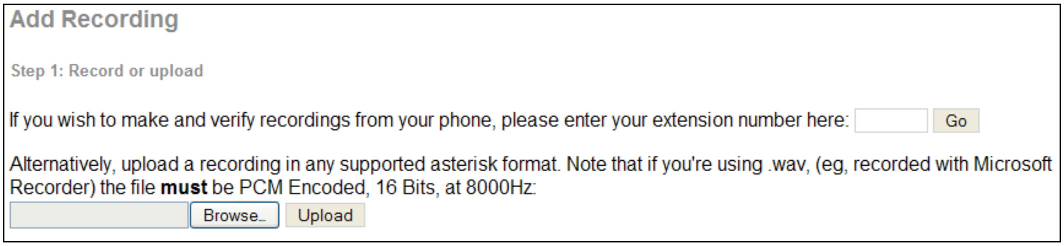


System Recordings

As we have seen in numerous sections within the system, you often have to prerecord system messages for the other modules to use. All of the system recordings are managed from this module. System recordings can be added in two different ways; you can either prerecord your messages and then upload them to the system, or you can record them directly into the system using any phone that is registered as an extension.

Prerecorded sound files must be PCM-Encoded, 16 bits, at 8000Hz.

If you simply wish to upload a file, you can do so from the main screen as shown below, or enter the extension number that you will use to record the message from.



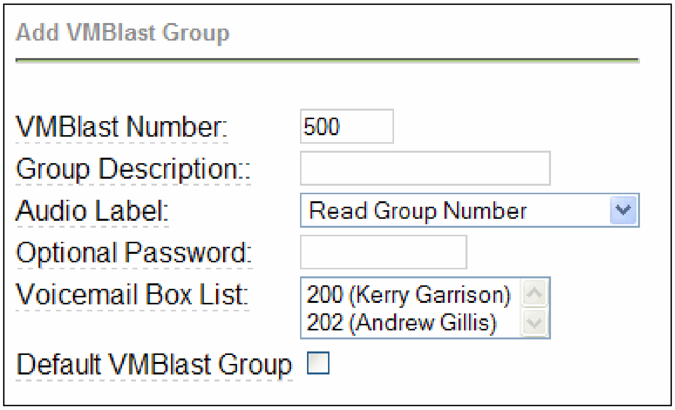
If you chose to record from an extension, then the system will take you to the next screen and tell you to dial *77 to begin recording. You can begin recording when you hear the short tone and then simply hang up when finished. To listen to your recording you can dial *99. If you are happy with the recording then you can give the recording a name and it will then be available from the pull-down menus throughout the system.

Voicemail blasting

A handy feature for managers is the ability to send the same voicemail message to a group of people; this is accomplished by setting up voicemail blast groups. The available option when creating voicemail blast groups are:

- **VMBlast Number**: This is the number you will dial to send the message to the group.
- **Group Description**: This is a text description for this particular voicemail blast group.
- **Audio Label**: This message is played to the caller to assure that he/she dialed the correct blast group number. This can be set to read the number of the group back, a beep only with no confirmation, or to any available system recording.
- **Optional Password**: If you want to restrict usage of the voicemail blast groups, then you can set a password to prevent unauthorized people from using the groups.
- **Voicemail Box List**: From this list, you need to select the mailboxes that will be a part of this particular voicemail blast group.

- **Default VMBlast Group** (yes/no): Each system is allowed a single voicemail blast group. If this is selected, then extensions can be automatically added to this group when creating the extension. If you make one group the default, it will remove the default setting from any previous group that you had set as the default.



Summary

In this chapter, you have learned about each feature that is in the PBX Settings module and how to configure them. Along the way, we have looked at common settings and some best practices for setting up some of the modules and why we may want to use certain features over others depending upon our needs and requirements. This should give you a solid footing for getting your system configured and tailored for your particular needs.

11
Maintenance and reporting

Once a system is up and running, the only thing you usually have to do from then on is the basic maintenance of the system for system updates. The other regular task may be for call detail record reporting. This chapter will show you the primary tasks for both of these typical functions.

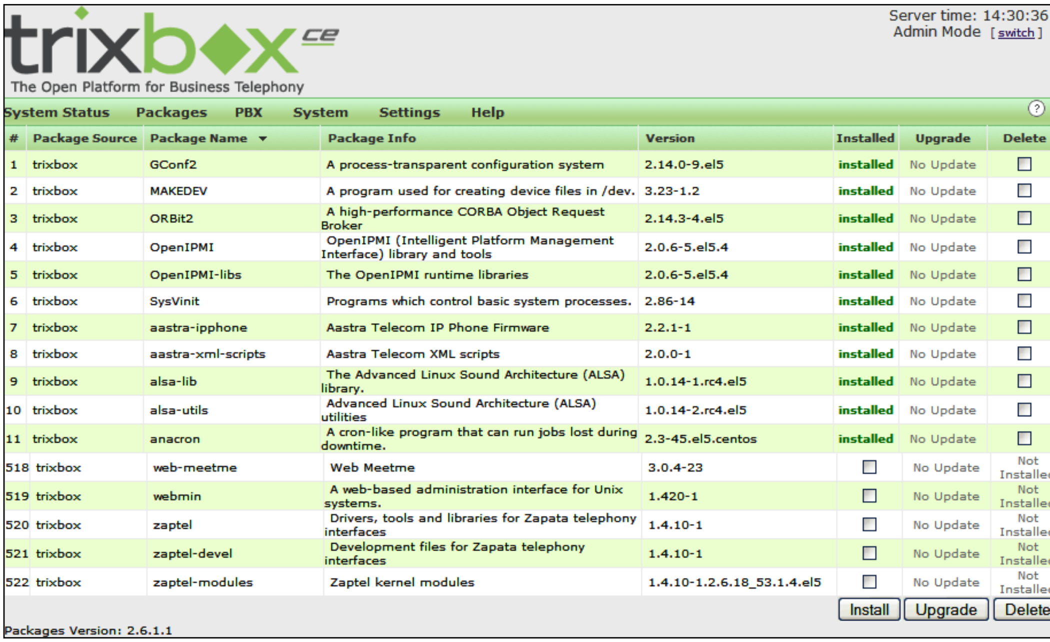
System updates

Once in a while, some system update for the operating system or Asterisk may come out, or upgraded features or bug fixes will become available. The primary system administrator should watch for those updates, especially if your system is exposed to the Internet via any VoIP trunks. The trixbox CE system has a built-in system for managing updates and new packages; this system is called the Package Manager and is available from the trixbox CE dashboard when you are in Admin mode.

To access the Package Manager, click on the **Packages** link in the menu bar.

The Package Manager will then contact the trixbox package repositories to find out which new packages are available for installation or upgrading. This is the easiest means of keeping your system up to date. More advanced Linux administrators can use yum, and the Package Manager is simply a web interface to a set of yum repositories.

When the Package Manager finishes getting the list of available packages, a list will be displayed to let you know which packages are available for installation or upgrade. You can also choose to uninstall existing packages. You do need to be careful here because it is actually possible to uninstall the entire trixbox CE dashboard system.



#	Package Source	Package Name	Package Info	Version	Installed	Upgrade	Delete
1	trixbox	GConf2	A process-transparent configuration system	2.14.0-9.el5	installed	No Update	
2	trixbox	MAKEDEV	A program used for creating device files in /dev.	3.23-1.2	installed	No Update	
3	trixbox	ORBit2	A high-performance CORBA Object Request Broker	2.14.3-4.el5	installed	No Update	
4	trixbox	OpenIPMI	OpenIPMI (Intelligent Platform Management Interface) library and tools	2.0.6-5.el5.4	installed	No Update	
5	trixbox	OpenIPMI-libs	The OpenIPMI runtime libraries	2.0.6-5.el5.4	installed	No Update	
6	trixbox	SysVinit	Programs which control basic system processes.	2.86-14	installed	No Update	
7	trixbox	aastra-ipphone	Aastra Telecom IP Phone Firmware	2.2.1-1	installed	No Update	
8	trixbox	aastra-xml-scripts	Aastra Telecom XML scripts	2.0.0-1	installed	No Update	
9	trixbox	alsa-lib	The Advanced Linux Sound Architecture (ALSA) library.	1.0.14-1.rc4.el5	installed	No Update	
10	trixbox	alsa-utils	Advanced Linux Sound Architecture (ALSA) utilities	1.0.14-2.rc4.el5	installed	No Update	
11	trixbox	anacron	A cron-like program that can run jobs lost during downtime.	2.3-45.el5.centos	installed	No Update	
518	trixbox	web-meetme	Web Meetme	3.0.4-23		No Update	Not Installed
519	trixbox	webmin	A web-based administration interface for Unix systems.	1.420-1		No Update	Not Installed
520	trixbox	zaptel	Drivers, tools and libraries for Zapata telephony interfaces	1.4.10-1		No Update	Not Installed
521	trixbox	zaptel-devel	Development files for Zapata telephony interfaces	1.4.10-1		No Update	Not Installed
522	trixbox	zaptel-modules	Zaptel kernel modules	1.4.10-1.2.6.18_53.1.4.el5		No Update	Not Installed

You can only install groups of new packages, or update groups of existing packages at the same time; you cannot select some packages to install and others to upgrade in the same process.

Select either new packages or upgrades and hit the appropriate button at the bottom of the page. The packages will then be downloaded and processed without any further user intervention.

Should you install updates all the time?

I typically do not recommend installing every update just for the sake of having the latest code bits on your system. I have client systems that are still running on the original installation of **Asterisk@Home 0.8** with zero updates. However, if you have VoIP trunks, there have been several vulnerabilities that require updated versions of Asterisk to fix. The recommendation is that any system that is connected to the Internet should always be properly maintained and updated to keep it as safe and secure as possible.

A phone system is almost always the single most mission-critical system in any business, and installing updates on a system that is working perfectly needs to be thought through and not done without testing and ensuring there will not be any issues.

Checking system health

trixbox CE contains several methods of monitoring system health. The primary system is the main Admin mode screen, which has a number of monitors and graphs that show the current status of different system components.

Server Status

In the Server Status box is the status of common system services running on the system. On a default installation, the HUD Server is not available, so it will appear as N/A until you install the HUDLite Admin package.

Network usage

If you are seeing lots of errors or dropped packets in the last column, this could indicate a bad network card or switch port. This could cause serious call quality issues as errors and dropped packets will result in a jittery sounding or garbled conversation.

Memory usage

Unless you have a lot of RAM in your system, Linux will use as much memory as possible for system caching; this may cause the Physical Memory graph to go to 99% and turn red. Since this is a typical behavior on most systems, the only really important graph in this section is the Disk Swap graph. The Disk Swap graph should always show 0%; if this starts to go up, then the system is swapping RAM out to disk space. Because of the delay while swapping, you can get call quality issues if your system is using disk swap space.

Mounted filesystems

If your system runs out of disk space, all kinds of whacky things can happen from simply not being able to record calls to the system deciding to not answer inbound calls at all. The main cause of disk space usage is voicemail and recorded phone calls. If someone has set Asterisk in debugging mode, this will also cause the Asterisk log file to grow rapidly on a busy system.

From the command line, we can see what the difference directories contain.

Checking voicemail directories:

```
[root@asterisk1 ~]# du -s -h /var/spool/asterisk/voicemail/
126M    /var/spool/asterisk/voicemail/
```

You can see that on this system the voicemail is taking up to 126MB.

Checking recorded calls directories:

```
[root@asterisk1 monitor]# du -s -h /var/spool/asterisk/monitor
4.0K    /var/spool/asterisk/monitor
```

On this system, there is only 4KB taken up, so there is really nothing being recorded on this system.

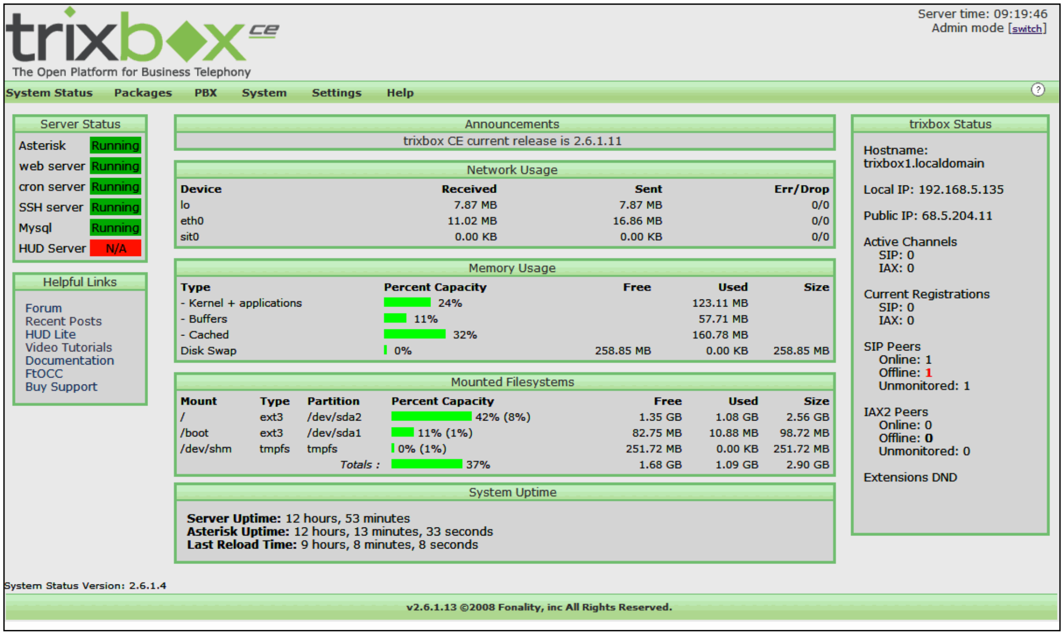
Checking log files directories:

```
[root@asterisk1 monitor]# du -s -h /var/log/asterisk
9.4G    /var/log/asterisk
```

This system is showing 9.4GB of log files; this is fairly typical for a medium-sized call center.

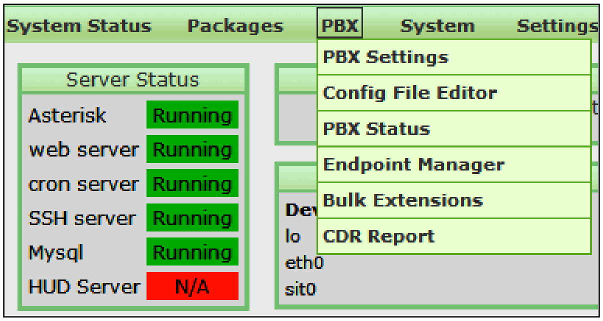
trixbox status

The status section shows you some basic information at the top, such as hostname and IP addresses, and system uptime. Under the basic information, you can see the number of active calls and then the number of configured devices versus how many are online and offline. If you have devices that are offline, these may be softphones that are not connected, or it could be indicative of a problem somewhere.



Reporting tools

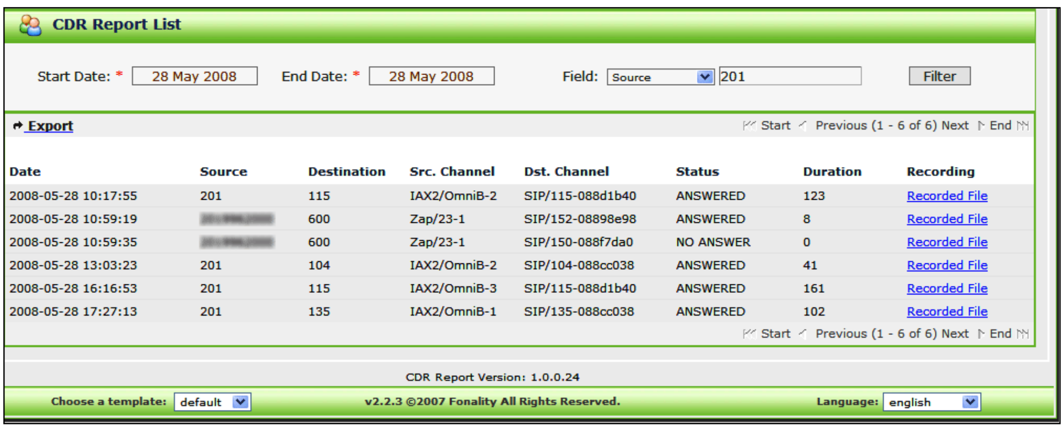
trixbox CE has a tool for doing call reporting to see what the different users on your system are doing. To access the CDR Reports, you select the **CDR Report** menu item from the **PBX** menu in the **Admin** section. This will launch the reporting module and show you the call data for the current day.



You can then change the search criteria to narrow down the scope of the reports. On the opposite page, you will see a complete call data report page showing 50 of 268 calls for the day from a small call center. We can narrow down the amount of data we are looking at by using some filters.

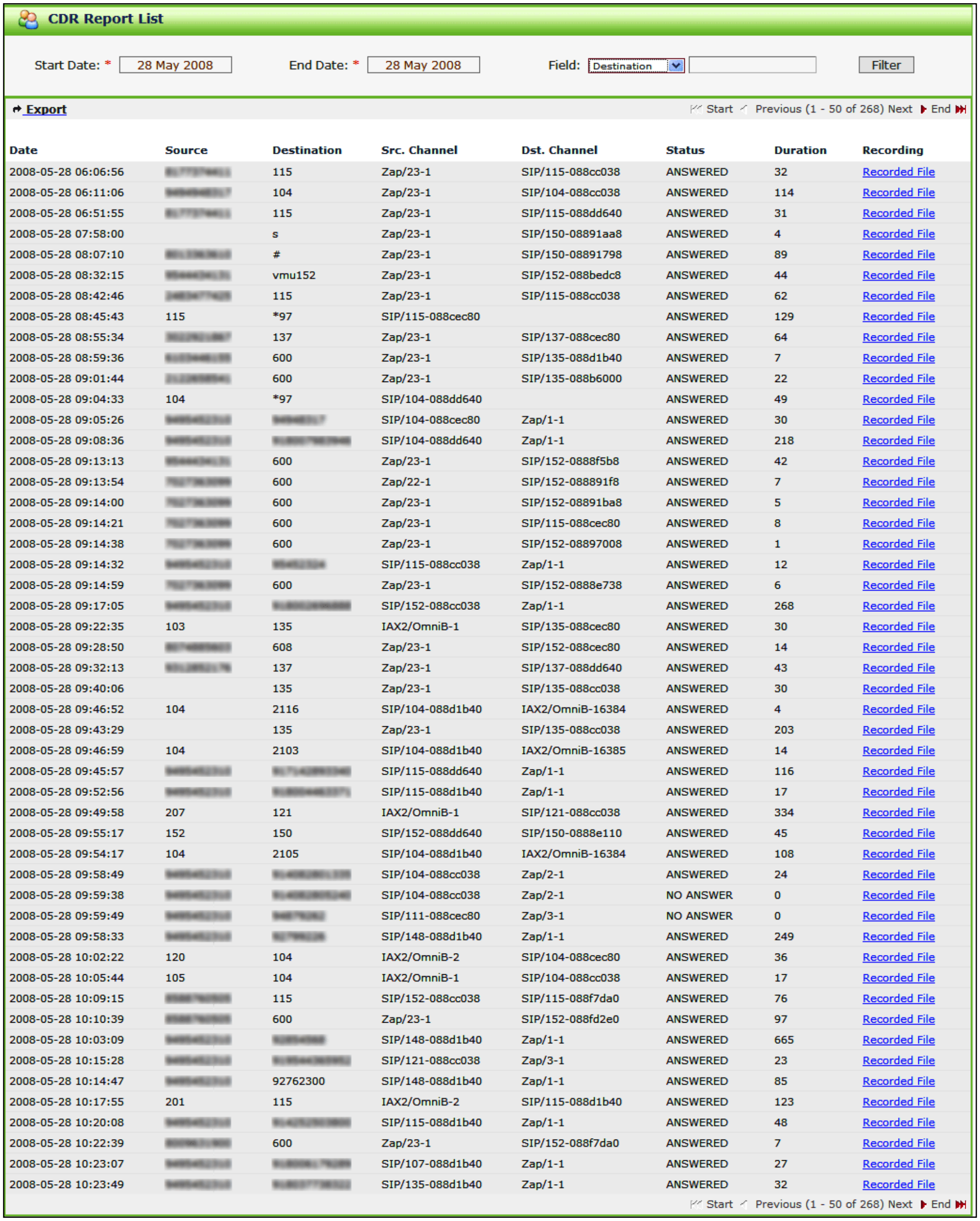
If we want to see only the calls that went to extension 115, we can put 115 into the search field making sure the pull-down menu is on **Destination**. In this case, it narrows down the search results down to 31 entries.

If we want to look at outbound calls, then we would change the pull-down menu to **Source** and put in the extension we want to do a report on. The following image shows an outbound call report:



Date	Source	Destination	Src. Channel	Dst. Channel	Status	Duration	Recording
2008-05-28 10:17:55	201	115	IAX2/OmniB-2	SIP/115-088d1b40	ANSWERED	123	Recorded File
2008-05-28 10:59:19	[illegible]	600	Zap/23-1	SIP/152-08898e98	ANSWERED	8	Recorded File
2008-05-28 10:59:35	[illegible]	600	Zap/23-1	SIP/150-088f7da0	NO ANSWER	0	Recorded File
2008-05-28 13:03:23	201	104	IAX2/OmniB-2	SIP/104-088cc038	ANSWERED	41	Recorded File
2008-05-28 16:16:53	201	115	IAX2/OmniB-3	SIP/115-088d1b40	ANSWERED	161	Recorded File
2008-05-28 17:27:13	201	135	IAX2/OmniB-1	SIP/135-088cc038	ANSWERED	102	Recorded File

The **Src Channel** and **Dest Channel** are used mostly for troubleshooting, or if you have distinct lines coming into different sections of the phone system.



CDR Report List

Start Date: * 28 May 2008 End Date: * 28 May 2008 Field: Destination Filter

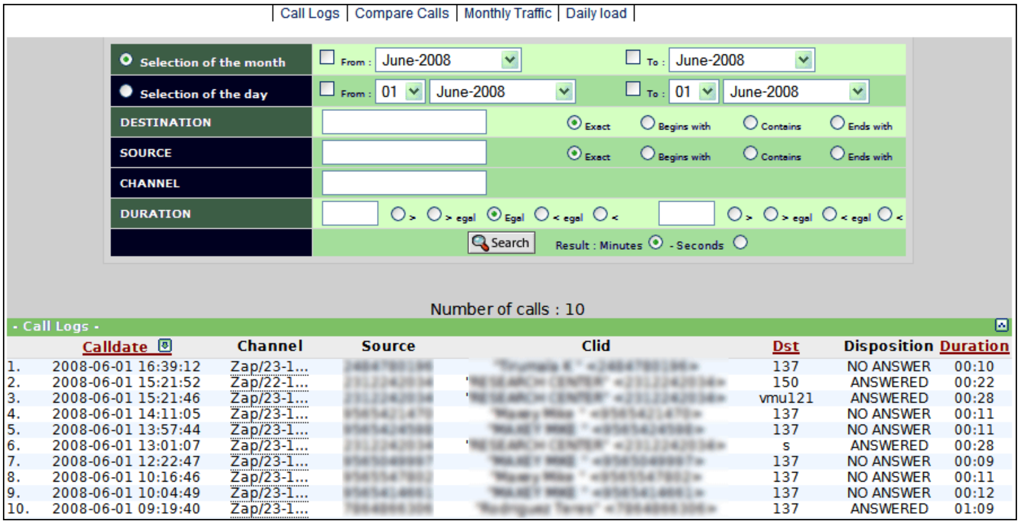
Export Start Previous (1 - 50 of 268) Next End

Date	Source	Destination	Src. Channel	Dst. Channel	Status	Duration	Recording
2008-05-28 06:06:56	[redacted]	115	Zap/23-1	SIP/115-088cc038	ANSWERED	32	Recorded File
2008-05-28 06:11:06	[redacted]	104	Zap/23-1	SIP/104-088cc038	ANSWERED	114	Recorded File
2008-05-28 06:51:55	[redacted]	115	Zap/23-1	SIP/115-088dd640	ANSWERED	31	Recorded File
2008-05-28 07:58:00		s	Zap/23-1	SIP/150-08891aa8	ANSWERED	4	Recorded File
2008-05-28 08:07:10	[redacted]	#	Zap/23-1	SIP/150-08891798	ANSWERED	89	Recorded File
2008-05-28 08:32:15	[redacted]	vmu152	Zap/23-1	SIP/152-088bedc8	ANSWERED	44	Recorded File
2008-05-28 08:42:46	[redacted]	115	Zap/23-1	SIP/115-088cc038	ANSWERED	62	Recorded File
2008-05-28 08:45:43	115	*97	SIP/115-088cec80		ANSWERED	129	Recorded File
2008-05-28 08:55:34	[redacted]	137	Zap/23-1	SIP/137-088cec80	ANSWERED	64	Recorded File
2008-05-28 08:59:36	[redacted]	600	Zap/23-1	SIP/135-088d1b40	ANSWERED	7	Recorded File
2008-05-28 09:01:44	[redacted]	600	Zap/23-1	SIP/135-088b6000	ANSWERED	22	Recorded File
2008-05-28 09:04:33	104	*97	SIP/104-088dd640		ANSWERED	49	Recorded File
2008-05-28 09:05:26	[redacted]	[redacted]	SIP/104-088cec80	Zap/1-1	ANSWERED	30	Recorded File
2008-05-28 09:08:36	[redacted]	[redacted]	SIP/104-088dd640	Zap/1-1	ANSWERED	218	Recorded File
2008-05-28 09:13:13	[redacted]	600	Zap/23-1	SIP/152-0888f5b8	ANSWERED	42	Recorded File
2008-05-28 09:13:54	[redacted]	600	Zap/22-1	SIP/152-088891f8	ANSWERED	7	Recorded File
2008-05-28 09:14:00	[redacted]	600	Zap/23-1	SIP/152-08891ba8	ANSWERED	5	Recorded File
2008-05-28 09:14:21	[redacted]	600	Zap/23-1	SIP/115-088cec80	ANSWERED	8	Recorded File
2008-05-28 09:14:38	[redacted]	600	Zap/23-1	SIP/152-08897008	ANSWERED	1	Recorded File
2008-05-28 09:14:32	[redacted]	[redacted]	SIP/115-088cc038	Zap/1-1	ANSWERED	12	Recorded File
2008-05-28 09:14:59	[redacted]	600	Zap/23-1	SIP/152-0888e738	ANSWERED	6	Recorded File
2008-05-28 09:17:05	[redacted]	[redacted]	SIP/152-088cc038	Zap/1-1	ANSWERED	268	Recorded File
2008-05-28 09:22:35	103	135	IAX2/OmniB-1	SIP/135-088cec80	ANSWERED	30	Recorded File
2008-05-28 09:28:50	[redacted]	608	Zap/23-1	SIP/152-088cec80	ANSWERED	14	Recorded File
2008-05-28 09:32:13	[redacted]	137	Zap/23-1	SIP/137-088dd640	ANSWERED	43	Recorded File
2008-05-28 09:40:06		135	Zap/23-1	SIP/135-088cc038	ANSWERED	30	Recorded File
2008-05-28 09:46:52	104	2116	SIP/104-088d1b40	IAX2/OmniB-16384	ANSWERED	4	Recorded File
2008-05-28 09:43:29		135	Zap/23-1	SIP/135-088cc038	ANSWERED	203	Recorded File
2008-05-28 09:46:59	104	2103	SIP/104-088d1b40	IAX2/OmniB-16385	ANSWERED	14	Recorded File
2008-05-28 09:45:57	[redacted]	[redacted]	SIP/115-088dd640	Zap/1-1	ANSWERED	116	Recorded File
2008-05-28 09:52:56	[redacted]	[redacted]	SIP/115-088d1b40	Zap/1-1	ANSWERED	17	Recorded File
2008-05-28 09:49:58	207	121	IAX2/OmniB-1	SIP/121-088cc038	ANSWERED	334	Recorded File
2008-05-28 09:55:17	152	150	SIP/152-088dd640	SIP/150-0888e110	ANSWERED	45	Recorded File
2008-05-28 09:54:17	104	2105	SIP/104-088d1b40	IAX2/OmniB-16384	ANSWERED	108	Recorded File
2008-05-28 09:58:49	[redacted]	[redacted]	SIP/104-088cc038	Zap/2-1	ANSWERED	24	Recorded File
2008-05-28 09:59:38	[redacted]	[redacted]	SIP/104-088cc038	Zap/2-1	NO ANSWER	0	Recorded File
2008-05-28 09:59:49	[redacted]	[redacted]	SIP/111-088cec80	Zap/3-1	NO ANSWER	0	Recorded File
2008-05-28 09:58:33	[redacted]	[redacted]	SIP/148-088d1b40	Zap/1-1	ANSWERED	249	Recorded File
2008-05-28 10:02:22	120	104	IAX2/OmniB-2	SIP/104-088cec80	ANSWERED	36	Recorded File
2008-05-28 10:05:44	105	104	IAX2/OmniB-1	SIP/104-088cc038	ANSWERED	17	Recorded File
2008-05-28 10:09:15	[redacted]	115	SIP/152-088cc038	SIP/115-088f7da0	ANSWERED	76	Recorded File
2008-05-28 10:10:39	[redacted]	600	Zap/23-1	SIP/152-088fd2e0	ANSWERED	97	Recorded File
2008-05-28 10:03:09	[redacted]	[redacted]	SIP/148-088d1b40	Zap/1-1	ANSWERED	665	Recorded File
2008-05-28 10:15:28	[redacted]	[redacted]	SIP/121-088cc038	Zap/3-1	ANSWERED	23	Recorded File
2008-05-28 10:14:47	[redacted]	92762300	SIP/148-088d1b40	Zap/1-1	ANSWERED	85	Recorded File
2008-05-28 10:17:55	201	115	IAX2/OmniB-2	SIP/115-088d1b40	ANSWERED	123	Recorded File
2008-05-28 10:20:08	[redacted]	[redacted]	SIP/115-088d1b40	Zap/1-1	ANSWERED	48	Recorded File
2008-05-28 10:22:39	[redacted]	600	Zap/23-1	SIP/152-088f7da0	ANSWERED	7	Recorded File
2008-05-28 10:23:07	[redacted]	[redacted]	SIP/107-088d1b40	Zap/1-1	ANSWERED	27	Recorded File
2008-05-28 10:23:49	[redacted]	[redacted]	SIP/135-088d1b40	Zap/1-1	ANSWERED	32	Recorded File

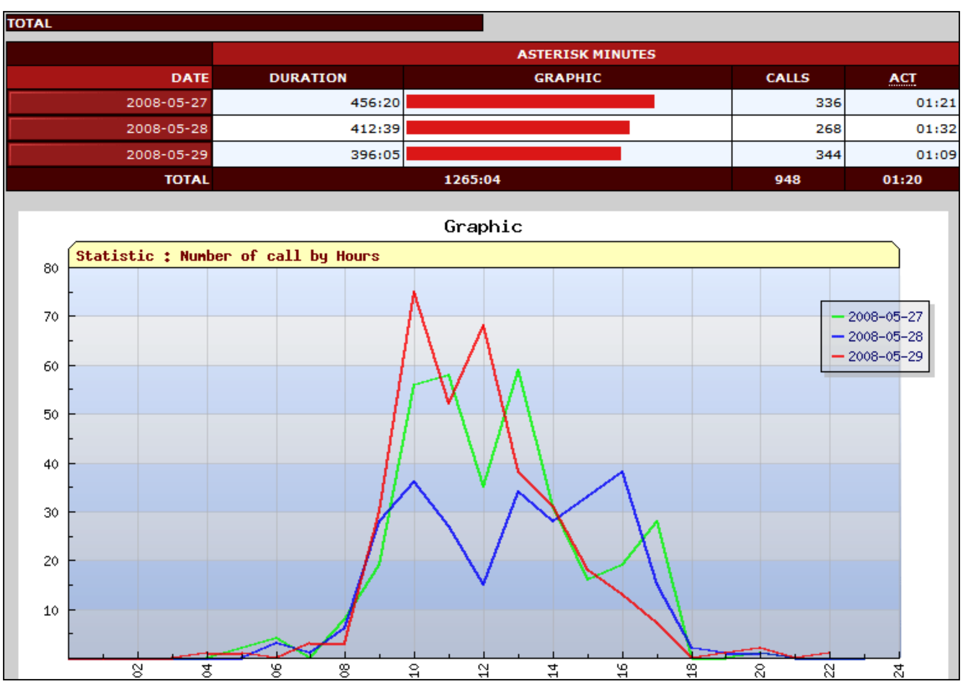
Start Previous (1 - 50 of 268) Next End

There is a more advanced call reporting system that is accessible by going into the **PBX Configuration** tool and selecting the **Reports** tab at the top of the page.

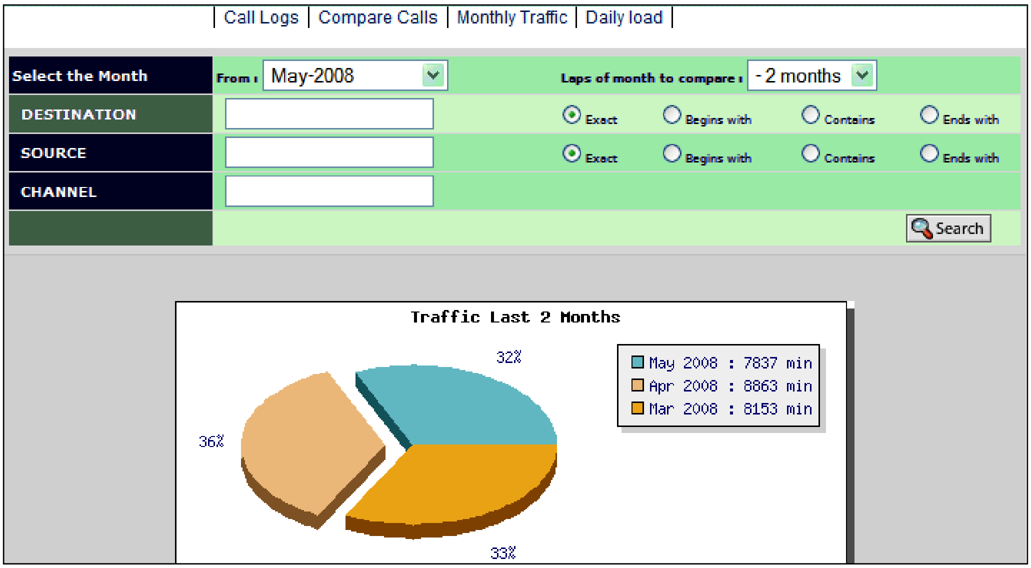
The following is a typical example of call reports from the advanced CDR tool:



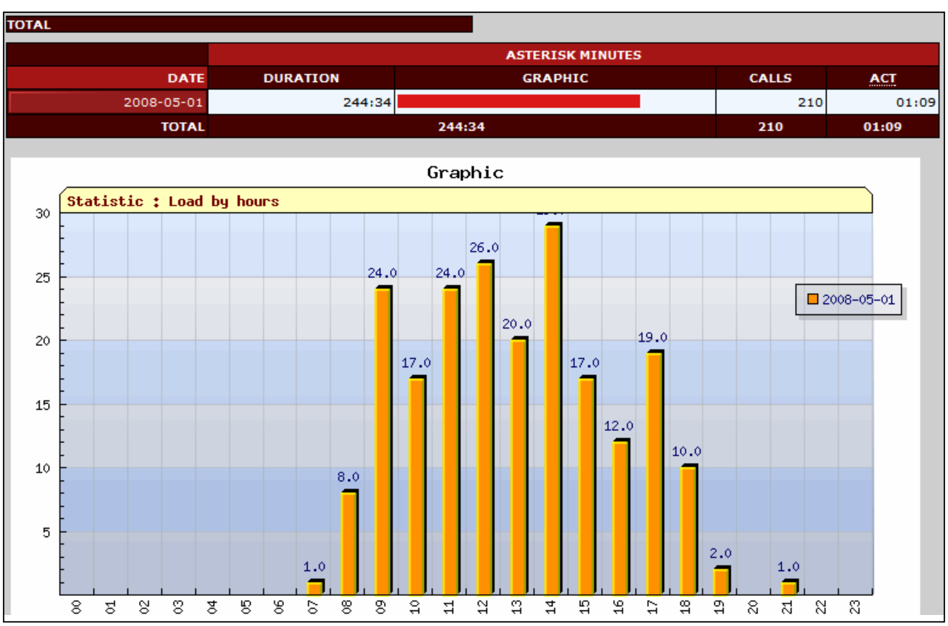
While the basic page is very similar to the standard CDR Reports with some additional controls over the search criteria, there are a few more functions available at the top of the page. Selecting **Compare Calls** will allow you to select a date range and see a report comparing the call volume across that date range.



The Monthly Traffic report will show the call volume over the period of several months. You can select a date range as well as the other normal criteria and get a graph of call usage.



The final report is the Daily Load, which can show the call volume per hour over the course of the day. You can then drill down into the calls for a specific time frame.



Summary

In this chapter, we have taken a look at how to handle basic system maintenance using the Package Manager. We have also taken a look at how to get both basic and advanced call reports using the trixbox CE dashboard CDR Reports tool and the Reports tool that is in the PBX Configuration tool.

12
Troubleshooting

One of the most important skills you can develop if you are going to install and manage trixbox CE systems is good troubleshooting. Although this chapter will give you a basic background in troubleshooting common problems, you should still take the time to become familiar with general troubleshooting of common Linux problems such as networking, hard disk maintenance, command-line tools, and package management. With the information in this chapter, you should be able to work through the most common problems that you are likely to encounter.

Getting to know Asterisk

Since Asterisk is the engine under the hood, you need to know how to get under the hood and see what it's doing and use Asterisk's **CLI** (**Command Line Interface**) to help troubleshoot connections to devices and service providers. To access the Asterisk CLI, we need to log in to the console or SSH into the system using a tool like Putty. Once logged in, we access the Asterisk CLI with the `asterisk -r` command. When we do this, we should get a screen like the following example:

```
[trixbox1.localdomain ~]# asterisk -r
Asterisk 1.4.21.2-2 RPM by vc-rpms@voipconsulting.nl, Copyright (C) 1999
- 2008 Digium, Inc. and others.
Created by Mark Spencer <markster@digium.com>
Asterisk comes with ABSOLUTELY NO WARRANTY; type 'core show warranty' for
details.
This is free software, with components licensed under the GNU General
Public License version 2 and other licenses; you are welcome to
redistribute it under certain conditions. Type 'core show license' for
details.
=========================================================================
Connected to Asterisk 1.4.21.2-2 RPM by vc-rpms@voipconsulting.nl
currently running on trixbox1 (pid = 2755)
Verbosity is at least 3
trixbox1*CLI>
```

Once in the CLI, we can always type `help` for a list of available commands. While this list is quite long, we don't need to understand each and every command, but there are quite a few that will help us to diagnose and troubleshoot problems. This section will break down the commands by category so we can get to know when we will need specific sets of commands.

Stopping, restarting, and reloading Asterisk

Often we need to stop or restart asterisk or to tell Asterisk that we want it to reload its configurations. The main thing is to understand how to do this properly so that you don't cause even more problems for yourself. With trixbox CE, Asterisk runs as the user 'asterisk', so if you simply try to start Asterisk from the Linux command line, it will run as the root user, and all kinds of things will get funky and not work properly. To avoid these issues, make sure to always start and stop Asterisk using the information listed below.

To stop, start, or restart Asterisk, we want to be at the Linux command prompt to issue the following commands:

- `amportal stop`: This command will stop Asterisk immediately
- `amportal start`: This command will attempt to start up Asterisk properly
- `amportal restart`: This command will stop and restart Asterisk properly

If we are in the Asterisk CLI, there are a few commands that we can use depending upon what we are trying to accomplish; these commands are:

- `reload`: Using the `reload` command will tell Asterisk to reload all of its configuration files
- `restart gracefully`: This command will stop any new calls from being processed and will restart Asterisk when any current calls complete
- `restart now`: All calls will be immediately dropped and Asterisk will restart immediately
- `restart when convenient`: Asterisk will wait until there is no call volume before restarting
- `stop gracefully`: This command will stop any new calls from being processed and will stop Asterisk when any current calls complete.
- `stop now`: All current calls will be terminated and Asterisk will shut down immediately
- `stop when convenient`: Asterisk will wait until there is no call volume before stopping

After issuing any of the stop commands, you will need to run `amportal start` to properly start Asterisk back up again.

General channel information

We often need to see if there are any calls in progress and what the status of those calls is. For this, we use the `core show channels` and `core show channel` commands. Let's take a look at what these commands will show us.

The `core show channels` command will list all of the current channels that are in use on the system. The following is an example of the output from this command:

```
Channel (Context Extension Pri) State Appl. Data
SIP/vitelity-081c0498 (incoming 1) Up Bridged Call SIP/300-b7300470
SIP/300-b7300470 (internal 913108614300 5) Up Dial SIP/
vitelity/13108614300
2 active channels
1 active call
```

In this example, we have one channel going to our VoIP service provider and another channel connected to extension 300. Once we have this information, we can use the `core show channel` command to get information specific to a channel. The following example shows how we can do this:

```
trixbox1*CLI> core show channel SIP/300-b7300470

 -- General --
               Name: SIP/300-b7300470
               Type: SIP
           UniqueID: 1223299945.196
          Caller ID: 9495557819
     Caller ID Name: Kerry Garrison
        DNID Digits: 913108614300
              State: Up (6)
              Rings: 0
       NativeFormat: 4
        WriteFormat: 4
         ReadFormat: 4
1st File Descriptor: 33
          Frames in: 1022
         Frames out: 1028
     Time to Hangup: 0
```

```
       Elapsed Time: 0h0m23s
      Direct Bridge: SIP/vitelity-081c0498
    Indirect Bridge: SIP/vitelity-081c0498
 -- PBX --
            Context: internal
          Extension: 913108614300
           Priority: 5
         Call Group: 2
       Pickup Group: 2
        Application: Dial
               Data: SIP/vitelity/13108614300
        Blocking in: ast_waitfor_nandfds
          Variables:
BRIDGEPEER=SIP/vitelity-081c0498
DIALEDPEERNUMBER=vitelity/13108614300
DIALEDPEERNAME=SIP/vitelity-081c0498
EXTENSION=200
SIPCALLID=6fa08e69300ef9c6
SIPUSERAGENT=Aastra 35iCT/2.1.0.2145
SIPDOMAIN=10.10.20.200
SIPURI=sip:00085D10BD1A@192.168.5.159:5060

      CDR Variables:
level 1: clid="Kerry Garrison" <9495557819>
level 1: src=9495557819
level 1: dst=913108614300
level 1: dcontext=internal
level 1: channel=SIP/00085D10BD1A-b7300470
level 1: dstchannel=SIP/vitelity-081c0498
level 1: lastapp=Dial
level 1: lastdata=SIP/vitelity/13108614300
level 1: start=2008-10-06 06:32:25
level 1: answer=2008-10-06 06:32:28
level 1: end=2008-10-06 06:32:28
level 1: duration=0
level 1: billsec=0
level 1: disposition=ANSWERED
level 1: amaflags=DOCUMENTATION
level 1: uniqueid=1223299945.196
```

As you can see, there is quite a bit of information available for different channels.

Troubleshooting SIP extensions and trunks

In most cases, you will be using SIP devices as your endpoints, and knowing how to use Asterisk to provide information on your extensions is one of the most valuable tools available. For SIP trunks to VoIP service providers, a similar set of tools will help troubleshoot those connections. The following is a list of SIP-related Asterisk CLI commands:

- `sip set debug`—enable SIP debugging
- `sip set debug ip`—enable SIP debugging on IP
- `sip set debug off`—disable SIP debugging
- `sip show channels`—list active SIP channels
- `sip show channel`—show detailed SIP channel info
- `sip show peers`—list defined SIP peers
- `sip show peer`—show details on specific SIP peer
- `sip show registry`—list SIP registration status

SIP extensions are referred to as 'peers', so these are the commands we will use to look at the status of our extensions. The `sip show peers` command will give us a list of these peers and some additional information that we can use. Let's take a look at this command to see what it does.

```
Name/username         Host            Dyn Nat ACL Port     Status
301                  (Unspecified)    D   N       0        UNKNOWN
300                  (Unspecified)    D   N       0        UNKNOWN
204/204              192.168.5.248    D   N       5060     OK (10 ms)
203/203              192.168.5.145    D   N       5060     OK (21 ms)
202                  (Unspecified)    D   N       0        UNKNOWN
200/200              192.168.5.244    D   N       5060     OK (61 ms)
6 sip peers [Monitored: 3 online, 3 offline Unmonitored: 0 online, 0
offline]
```

In the first column we see the name of the peer and the username that was used to authenticate into the peer; with extensions, this should show the extension number for both name and username. In the second column we see the IP address that the connection came from. The third column tells us if the IP address was dynamic or not. The fourth column tells us if Nat was enabled or not. The next column gives us the UDP port that was used to connect to the peer. The final column gives us a status of the peer; if the status is OK, then it gives us a report on the connection speed to that peer. Anything above about 150ms is most likely going to cause you problems with poor call quality.

If you have SIP trunks then you will also see entries in there for these connections as well; the following is an example of what a SIP trunk will look like:

```
Name/username              Host             Dyn Nat ACL Port     Status
vitelity/kerryg_demo      64.2.142.28           N      5060     OK (98 ms)
1 sip peers [Monitored: 1 online, 0 offline Unmonitored: 0 online, 0
offline]
```

Here we see much of the same information, but because the name is not an extension number, we know that this is a trunk instead of an extension. Using the `sip show peers` command, we can quickly see if an extension or trunk is connected or not. If we are expecting the device or trunk to be connected and it is not, we need to take further troubleshooting steps.

For most SIP trunks (ones that require a registration), we use the `sip show registry` command to see the registration status. An example of this command in use is as follows:

```
Host                            Username       Refresh     State
inbound3.vitelity.net:5060      kerryg_demo    45          Registered
proxy01.sipphone.com:5060       17476060900    105         Registered
beta.teliax.com:5060            kerry_test     105         Auth Sent
```

This shows us the host we are trying to connect to along with the port we are trying to connect on, the username we are trying to use to register with, the registration interval time, and the current state of the registration attempt. Anything other than `Registered` indicates a problem somewhere. This could be as simple as a typo in the host name, username, or password.

If we need to start troubleshooting an extension or trunk, the next command we will want to get to know is the `sip set debug` command. In order to filter out clutter, it is best to issue the command to look only at messages to/from a specific IP address. If we are having problems with a particular device, this will let us only see messages that are relevant. If we wanted to only look at messages related to IP address 192.168.5.5, we would use the following command:

```
sip set debug ip 192.168.5.5
```

When everything is working properly, and an extension attempts to register and does so successfully, we should see several pages of information that will tell us what is happening. A typical registration attempt will start with a registration request like the following:

```
<--- SIP read from 192.168.5.5:5060 --->
REGISTER sip:192.168.5.135;transport=UDP SIP/2.0
Via: SIP/2.0/UDP 192.168.5.5:5060;branch=z9hG4bK-d8754z-d9aeed14276a5752-
1---d8754z-;rport
Max-Forwards: 70
Contact: <sip:200@192.168.5.5:5060;rinstance=8cb51e856b641bf9>;transport=
UDP
To: <sip:200@192.168.5.135>;transport=UDP
From: <sip:200@192.168.5.135>;transport=UDP;tag=89156d04
Call-ID: ODY1NjllMWYyNjc2NTZhY2JiZDMwZDYzMTkyODg2Y2Q.
CSeq: 1 REGISTER
Expires: 3600
Allow: INVITE, ACK, CANCEL, OPTIONS, BYE, NOTIFY, REFER, MESSAGE,
OPTIONS, INFO
User-Agent: Zoiper for Windows rev.1105
Allow-Events: presence
Content-Length: 0
```

If all goes well, it should end with a success message that looks like the following example:

```
<--- SIP read from 192.168.5.5:5060 --->
SIP/2.0 200 OK
Via: SIP/2.0/UDP 192.168.5.135:5060;branch=z9hG4bK784cfde1;rport=5060
Contact: <sip:200@192.168.5.5:5060>
To: <sip:sip:200@192.168.5.5:5060>;tag=e568af7d
From: "Unknown"<sip:Unknown@192.168.5.135>;tag=as166248a0
Call-ID: MTgyZTVlYzhkMzZjNTQ3YWU3ZTllMGQzNDE0NDJlMjY.
CSeq: 102 NOTIFY
User-Agent: Zoiper for Windows rev.1105
Content-Length: 0
```

One of the most typical problems with an extension or trunk not registering properly is a bad username or password. We can tell that we have a problem such as that if we get a `Forbidden (Bad auth)` message; an example of a bad password message is shown here:

```
<--- Transmitting (NAT) to 192.168.5.5:5060 --->
SIP/2.0 403 Forbidden (Bad auth)
Via: SIP/2.0/UDP 192.168.5.5:5060;branch=z9hG4bK-d8754z-b12d43225da8e917-1---d8754z-;received=192.168.5.5;rport=5060
From: <sip:200@192.168.5.135>;transport=UDP;tag=ff4c6533
To: <sip:200@192.168.5.135>;transport=UDP;tag=as716b551a
Call-ID: NWEwNWM1MmI5MzlmZjhmZTQ5ZGMxMGNlNzMzNGUyYmM.
CSeq: 2 REGISTER
User-Agent: Asterisk PBX
Allow: INVITE, ACK, CANCEL, OPTIONS, BYE, REFER, SUBSCRIBE, NOTIFY
Supported: replaces
Content-Length: 0
```

By being able to analyze our `sip` debug messages, we can begin to figure out what kind of problem we are having and take steps to correct it.

Troubleshooting IAX2 extensions and trunks

If you are using IAX2 extensions, there are also commands for looking at these types of connections. The `iax2 show peers` command is similar to the `sip show peers` command with just a few minor differences. The following is an example showing an inbound and outbound IAX2 trunk as well as an IAX2 extension connected:

```
Name/Username    Host                 Mask             Port      Status
vitel-outbound/  64.2.142.18     (S)  255.255.255.255  4569      OK
                                                                (130 ms)
vitel-inbound/t  64.2.142.28     (S)  255.255.255.255  4569      OK
                                                                (138 ms)
303              192.168.5.5     (D)  255.255.255.255  4569      OK
                                                                (72 ms)
3 iax2 peers [3 online, 0 offline, 0 unmonitored]
```

We can see the name/username as before, the trunk that has a static IP unlike the extension, which is dynamic, the netmask of the IP address, the UDP port that was used to connect, and the status and connection speed.

We can also look at the registration of IAX2 trunks using the `iax2 show registry` command. An example of this is shown as follows:

```
Host                  dnsmgr  Username    Perceived        Refresh State
64.2.142.28:4569      N       tech_demo   <Unregistered>   60 Rejected
```

In this case we can see that the trunk is not registered because the registration was rejected. Debugging IAX2 connections is a little trickier than SIP connections due to the type of information that is outputted. If we issue the `iax2 set debug` command to turn on IAX2 debugging, we can force a re-registration attempt by reloading the IAX2 configurations with the `iax2 reload` command. What we get in the case of this particular failure is shown here:

```
Tx-Frame Retry[000] -- OSeqno: 000 ISeqno: 000 Type: IAX     Subclass:
REGREQ
   Timestamp: 00008ms  SCall: 05520  DCall: 00000 [64.2.142.28:4569]
   USERNAME         : tech_demo
   REFRESH          : 60
Tx-Frame Retry[000] -- OSeqno: 000 ISeqno: 000 Type: IAX     Subclass:
POKE
   Timestamp: 00008ms  SCall: 02609  DCall: 00000 [64.2.142.18:4569]
Tx-Frame Retry[000] -- OSeqno: 000 ISeqno: 000 Type: IAX     Subclass:
POKE
   Timestamp: 00008ms  SCall: 03170  DCall: 00000 [64.2.142.28:4569]
Rx-Frame Retry[ No] -- OSeqno: 000 ISeqno: 001 Type: IAX     Subclass:
PONG
   Timestamp: 00008ms  SCall: 00023  DCall: 02609 [64.2.142.18:4569]
Rx-Frame Retry[ No] -- OSeqno: 000 ISeqno: 001 Type: IAX     Subclass:
REGREJ
   Timestamp: 00007ms  SCall: 00020  DCall: 05520 [64.2.142.28:4569]
   CAUSE            : Registration Refused
   CAUSE CODE       : 29
```

The `Cause Code` of `29` indicates that this is a 'Facility Rejected' error meaning we probably have something wrong with our login or password. Although it isn't easy to figure out what the cause codes are, the cause text is usually going to put you on the right track to figure out your issue.

Troubleshooting zap channels

Zap channels are used when dealing with TDM cards (analog lines or PRI circuits) and are always active, unlike a SIP trunk, which is only available when it is actually in use. If we use the `zap show channels` command, we can see all of the channels that are currently available. This example shows the output from a Digium TDM400 with two FXO modules and one FXS module configured:

```
Chan Extension  Context         Language   MOH Interpret
pseudo          default         en         default
1               from-zaptel     en         default
2               from-zaptel     en         default
3               from-internal   en         default
```

Whether a call is in progress or not, we can use the `zap show channel` command to look at a specific channel. If we have a call in progress on zap channel 1, we can use `zap show channel 1` to see detailed information about the channel. The following example shows the typical information available during a call:

```
Channel: 1
File Descriptor: 13
Span: 1
Extension:
Dialing: no
Context: from-zaptel
Caller ID:
Calling TON: 0
Caller ID name:
Destroy: 0
InAlarm: 0
Signalling Type: FXS Kewlstart
Radio: 0
Owner: Zap/1-1
Real: Zap/1-1
Callwait: <None>
Threeway: <None>
Confno: -1
Propagated Conference: -1
Real in conference: 0
DSP: yes
Relax DTMF: no
Dialing/CallwaitCAS: 0/0
Default law: ulaw
```

```
Fax Handled: no
Pulse phone: no
Echo Cancellation: 128 taps unless TDM bridged, currently ON
Actual Confinfo: Num/0, Mode/0x0000
Actual Confmute: No
Hookstate (FXS only): Offhook
```

To get a quick look at the status of the TDM boards you have installed in your system, you can use the `zap show status` command. On the same TDM400, here is the output from this command:

```
Description                         Alarms    IRQ      bpviol    CRC4
Wildcard TDM400P REV E/F Board 1    OK        0        0         0
```

Troubleshooting VoIP problems

There are several common problems you are most likely to run into, especially when dealing with extensions that are off-site from the main location or dealing with VoIP service providers.

The two most common issues are phones losing registration and users getting one-way audio. Some people will say that the only way to solve this is to use IAX2 for extensions, but a proper setup of your system and network will get regular SIP devices working properly virtually every time.

The first thing we need to know is that SIP extensions and trunks use specific UDP ports, so you will need to make sure that your firewall is not blocking or modifying these ports. The table below shows the different ports that need to be available:

Protocol	Type	Port Range
SIP	UDP	5060 - 5061
RTP (for SIP calls)	UDP	10,000 – 20,000
IAX2	UDP	4569

To make sure our trixbox CE system is set up properly, we need to edit the `/etc/asterisk/sip_nat.conf` file and make sure we have it set up properly for our network.

From the Linux command line, type `nano /etc/asterisk/sip_general_custom.conf`.

We need to have an `externip` setting that correlates to the public IP address that we are using (go to `http://whatismyip.com`, or look on the **System Status** page on your trixbox CE system, for your public IP address), a `nat` setting if we are using network address translation (setting to `yes` is just a safe default), and a `localnet` setting that describes the local network the trixbox CE system is on. A typical configuration may look like the following example:

- `externip=10.10.1.170`—edit to match your public IP address
- `nat=yes`
- `localnet=192.168.2.0/255.255.255.0`—edit to match your local network

If you are having issues with inbound calls not coming into your system, or you have remote phones that are losing registration at times, you will want to use the port forwarding feature of your firewall to route the ports listed in the previous table to your phone system or remote endpoint.

With VoIP trunks, we may often come across call quality issues that we need to figure out; this will typically manifest itself as jitter, pops, dropped speech, and sometimes as echo. Before even setting up a VoIP circuit you can determine if your internet access is likely to even support VoIP calls, by using the same techniques.

To begin, we can use the `Linux ping` command to simulate the same type of traffic that a typical voice call would use. To do this, we can use the following parameters:

```
ping –i 0.02 –s 270 –c 500 <hostname>
```

We can also create multiple instances of the command running at the same time by stringing the commands together with an `&` symbol. If we wanted to simulate five simultaneous calls to Vitelity, we would use the following command:

```
ping -i 0.02 -s 270 -c 500 inbound3.vitelity.net&ping -i 0.02 -s 270 -c 500 inbound3.vitelity.net&ping -i 0.02 -s 270 -c 500 inbound3.vitelity.net&ping -i 0.02 -s 270 -c 500 inbound3.vitelity.net&ping -i 0.02 -s 270 -c 500 inbound3.vitelity.net
```

While this is running, we will be seeing data flow down the screen that looks like the following example:

```
278 bytes from 64.2.142.28.ptr.us.xo.net (64.2.142.28): icmp_seq=498 ttl=52 time=54.6 ms
278 bytes from 64.2.142.28.ptr.us.xo.net (64.2.142.28): icmp_seq=499 ttl=52 time=53.9 ms
278 bytes from 64.2.142.28.ptr.us.xo.net (64.2.142.28): icmp_seq=500 ttl=52 time=59.7 ms
```

What we are looking for is the last item on the line (`time=...`) not to vary very much; strange spikes in the time number can indicate a potential problem.

Once the command completes, you will get a summary of the information like this:

```
500 packets transmitted, 499 received, 0% packet loss, time 11062ms
rtt min/avg/max/mdev = 51.606/56.385/69.116/3.261 ms, pipe 4
```

On the first row we are looking to see if we are getting any packet loss; even a single packet loss during a phone call can be heard and any constant packet loss will make the call quality basically intolerable. On the second line we want to look at the second and third number in the sequence. The second number is the average ping time, which we want to be nice and low, while the third number is the maximum ping time, which we not only want to be low, but to be fairly close to the average so that we know we are not getting wild swings in the numbers.

If you are experiencing issues, it can be helpful to figure out where those issues may be coming from. One of the best tools for this is `mtr` (Matt's traceroute), which will give us a look at the path between us and the remote connection and tell us what it thinks is happening along the way.

We can use `mtr` the same way as we used the `ping` command. Let's take a look at the output from a typical `mtr` command:

```
mtr -i 0.02 -s 270 -c 500 inbound3.vitelity.net
                  My traceroute  [v0.72]
trixbox1.localdomain (0.0.0.0)           Mon Oct  6 08:31:29 2008
Keys:  Help   Display mode   Restart statistics   Order of fields   quit
Packets   Pings
Host                         Loss%   Snt   Last   Avg  Best  Wrst StDev
1. 10.71.128.1               0.0%    19    7.4    8.4   7.1  13.8  1.8
2. ip68.oc.oc.cox.net        0.0%    19    9.8    8.3   6.2  11.8  1.4
3. ip68.oc.oc.cox.net        0.0%    19    8.5    9.9   7.3  19.0  3.1
4. ip68.oc.oc.cox.net        0.0%    19   10.1    9.4   7.7  13.4  1.7
5. ip68.oc.oc.cox.net        0.0%    19    9.5   10.4   6.6  18.7  4.0
6. langbbr01la.cox.net       0.0%    19   12.2   10.8   8.2  15.8  2.1
7. cr1.lay.savvis.net        5.3%    19    9.0   11.9   9.0  16.9  2.0
8. cr2.dv.savvis.net         0.0%    19   66.7   52.9  49.8  66.7  3.9
9. dcr2.Denver.savvis.net    0.0%    19   50.2   52.3  50.0  57.4  2.2
10. 208.172.161.178          0.0%    19   42.0   42.7  40.4  45.3  1.1
11. border8.den.pnap.net     0.0%    19   44.6   44.2  41.2  54.9  3.1
12. 69.25.215.226            0.0%    19   58.9   55.2  52.0  64.2  3.5
13. 64.2.142.28. us.xo.net   0.0%    19   53.7   55.2  53.1  64.5  3.1
```

From this example, we can see that `hop #7` looks to be dropping packets. If this was my connection, I would call Cox and report a problem between its connect and Savvis. It may not get it resolved right away, but continually opening tickets with a provider when you see a problem like this is often the only way to get these types of issues resolved.

Troubleshooting echo problems

Echo issues are actually such a complex problem that an entire book could be written on the topic. Since it is not normal to have echo on a SIP or IAX2 trunk, you can almost always rule out an issue with the trunk or provider, although there are some very rare situations where echo over a VoIP trunk can occur—it's almost always caused somewhere else in the system.

Beginning with trixbox CE **2.4**, the **OSLEC** (**open source line echo canceller**) has been included by default and almost instantly the complaints about echo on analog lines went away. There are times though that there may still be an echo being heard by one of the parties on a call.

The first step is to determine if the issue is caused by acoustic echo. This happens often with cheaper phones if the mic gain on a handset is set to high. Acoustic echo will also happen on an extension to extension call, as well as a call to a remote person, whereas other types of echo will only happen on calls outside the system. If you are getting acoustic echo, try to talk softer to see if that makes the echo go away; if it does, then go into the settings for the phone and look for a mic/microphone or handset gain and adjust it down to solve the problem.

Hybrid echo occurs between the trixbox system and the telephone company and is much harder to solve. The first step is to call the phone company and see if it will come out and test the lines; often a bad line or poor voltage will be the root cause of an echo problem.

Sometimes echo can be removed by adjusting the gains within Asterisk. Using the `ztmonitor` command, we can look at the audio levels in real time to see if we are getting large audio spikes. If we have a phone line on zap channel 1, then we can use the following command:

```
ztmonitor 1 –v
```

The resulting output will look like the following output:

```
Visual Audio Levels.
--------------------
Use zapata.conf file to adjust the gains if needed.
```

```
( # = Audio Level  * = Max Audio Hit )
<-------------(RX)-------------> <-------------(TX)------------->
############# *                  ########## *
```

As you are talking, you will see the graphs move left and right. What we want to do is to make sure that the spikes aren't much past the halfway point on either graph. If they are well past halfway, then we need to make some adjustments in order to correct that.

This is best done using two Putty windows open, one running `ztmonitor` and the other open to edit the files and reload Asterisk.

In the window you will use for editing files, run the following command:

```
nano /etc/asterisk/zapata.conf
```

In this file you will find the following lines:

```
rxgain=0.0
txgain=0.0
```

If the graph is showing that the TX is spiking too high, then start by editing the `txgain` to `-1`; if you need to make further adjustments, only increment or decrement the value by 1 each time, as it may not take much to get to the point that the calls are sounding good. When you have made the change, type *Ctrl+O* to save and *Ctrl+X* to exist. Next we need to tell Asterisk to reload the configuration we just made; we can do this quickly from the command line with the following command:

```
asterisk -rx "reload"
```

Continue testing and you should see the graph not spike as hard anymore. Keep making these adjustments to either the RX or TX gain based on the `ztmonitor` graph until you have tuned out all the echo.

Hardware troubleshooting

In most cases, installing a new card is going to work; however, there are some systems that will end up with IRQ conflicts that can result in the card not being detected or not initializing properly. There are a number of things to try to avoid any conflicts like this.

Clean up the BIOS

The very first thing you should do is disable any unused hardware in your system's BIOS. This would include disabling the serial port, parallel port, floppy controller, on-board audio card, and USB ports—this will free up IRQ's for your add-in cards.

Checking for conflicts

If cleaning up the BIOS didn't help, then the next step is to check the system to see if your cards are detected and have been handed an IRQ properly.

The first command to be typically used is `cat /proc/interrupts` as this will show us a list of equipment installed into a system and the IRQ numbers that are assigned. The following output is from a typical system with a Sangoma analog card installed:

```
           CPU0
  0:  852014312    IO-APIC-edge  timer
  2:          0    XT-PIC  cascade
  8:          1    IO-APIC-edge  rtc
137:         44    IO-APIC-level  eth1
145:  851809694    IO-APIC-level  wanpipe1
153:   59824592    IO-APIC-level  ehci_hcd, ohci_hcd, ohci_hcd
161:    6225357    IO-APIC-level  eth0
169:    5911306    IO-APIC-level  libata
177:          0    IO-APIC-level  libata
NMI:          0
LOC:  851363216
ERR:          0
MIS:          0
```

The Sangoma card used a driver called wanpipe, and we can see that the wanpipe driver is on its own IRQ and should not be having any issues.

Another useful command is `lspci`, which is another command to list the installed devices. The output from the same machine as above is shown as follows:

```
00:00.0 Host bridge: ATI Technologies Inc Radeon Xpress 200 Host Bridge
(rev 01)
00:01.0 PCI bridge: ATI Technologies Inc RS480 PCI Bridge
00:11.0 IDE interface: ATI Technologies Inc 437A Serial ATA Controller
(rev 80)
00:12.0 IDE interface: ATI Technologies Inc 4379 Serial ATA Controller
(rev 80)
00:13.0 USB Controller: ATI Technologies Inc IXP SB400 USB Host
Controller (rev 80)
00:13.1 USB Controller: ATI Technologies Inc IXP SB400 USB Host
Controller (rev 80)
```

```
00:13.2 USB Controller: ATI Technologies Inc IXP SB400 USB2 Host
Controller (rev 80)
00:14.0 SMBus: ATI Technologies Inc IXP SB400 SMBus Controller (rev 82)
00:14.1 IDE interface: ATI Technologies Inc Standard Dual Channel PCI
IDE Controller (rev 80)
00:14.3 ISA bridge: ATI Technologies Inc IXP SB400 PCI-ISA Bridge
(rev 80)
00:14.4 PCI bridge: ATI Technologies Inc IXP SB400 PCI-PCI Bridge
(rev 80)
01:05.0 VGA compatible controller: ATI Technologies Inc RC410 [Radeon
Xpress 200]
02:02.0 Ethernet controller: Realtek Semiconductor Co., Ltd. RTL-
8139/8139C/8139C+ (rev 10)
02:03.0 Ethernet controller: Realtek Semiconductor Co., Ltd. RTL-
8139/8139C/8139C+ (rev 10)
02:04.0 Network controller: Sangoma Technologies Corp. A200/Remora
FXO/FXS Analog AFT card
```

With this information, we can see that the system properly detected the Sangoma A200 card.

If these commands do not show your installed card, or if you are getting IRQ conflicts, the next thing to try is moving the card to a different slot. Even with all this, there is still no guarantee that trixbox CE and the card you chose will work on every motherboard. Fonality has a list of approved hardware available at `http://trixbox.org`.

Whose your Dahdi?

At the time of this writing, Zaptel is being phased out by Digium and its being replaced by Dahdi to avoid some copyright infringements. While all of the base code is still the same, many of the commands are being changed to reflect the new naming. For example, `ztmonitor` will be renamed to `dahdi_monitor` and the `zap` commands in the Asterisk CLI will be renamed to `dahdi`.

Summary

In this chapter, we have looked at the basics of troubleshooting a trixbox CE system. If you find yourself stuck and still having problems, the best place to turn to is the forums at `http://trixbox.org` and someone is bound to answer any question you may have.

13
trixbox utilities

What makes trixbox CE different from other distributions is the extra tools that come with it. There is an Endpoint Manager to help you configure your phones, simple CDR reports, a bulk extension creation tool, a tool to help you configure your network, a tool to help manage a DHCP server if you want to run one on your system, a backup module to help you back up your configuration, and several other utilities and scripts. This chapter will dig into these tools that make trixbox CE unique.

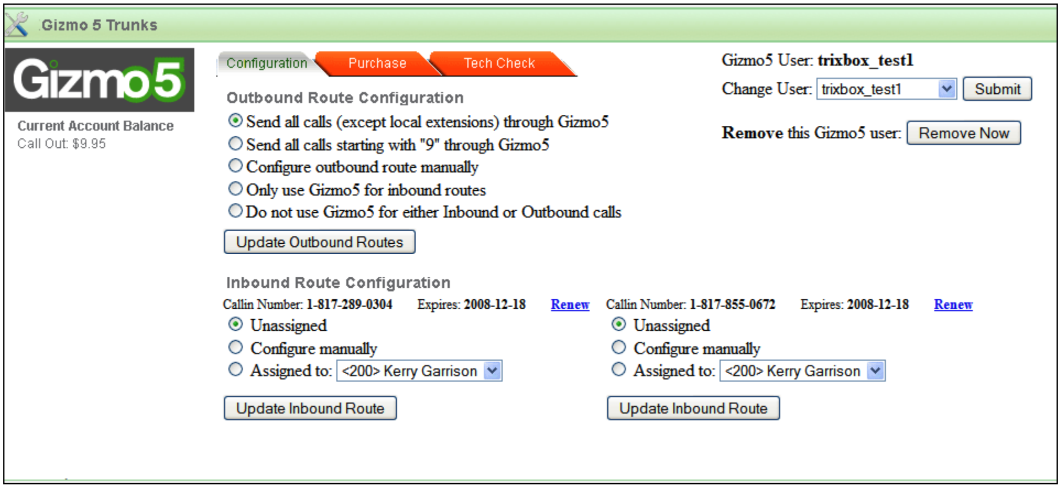
PBX menu

The **PBX** menu contains tools and utilities that help to manage and maintain settings within the PBX system as well as give you Gizmo5 trunk management, Call Data Reports, the ability to edit configuration files directly, bulk extensions creation, and a status tool for getting additional information about how your system is running.

You have already seen everything that is in the PBX Settings module as we dedicated an entire chapter to it. In this chapter, we will now look at the other modules that are available.

Gizmo5 module

Gizmo5 is a service very much like Skype except that, unlike Skype, Gizmo5 uses standard SIP protocols for communication. While you can simply use the Gizmo5 client just like you can with Skype, you can also use Gizmo5 as a SIP trunk provider. The Gizmo5 module that comes with trixbox CE allows you to easily purchase minutes, configure your Gizmo5 trunk to send and receive calls, and test your connection to make sure it is working properly.

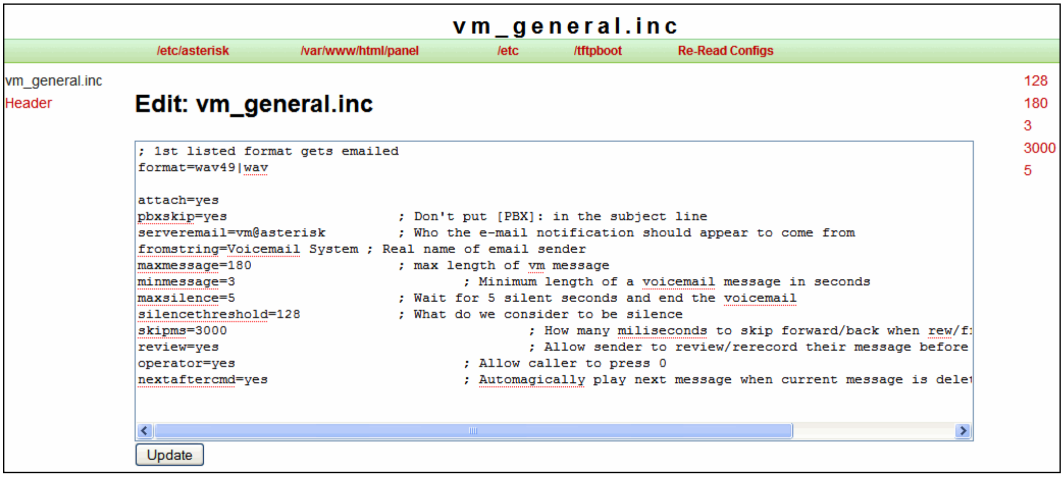


Config File Editor

The Config File Editor module allows you to directly edit many of the configuration files that are used with trixbox. Care must be taken since making a mistake in some of the files can render your system unusable. There are also several files that are overwritten every time you apply a change in the PBX Settings module. There are several groups of related files similar to the list shown here:

- `extensions.conf`
- `extensions_additional.conf`
- `extensions_custom.conf`
- `features.conf`
- `features_applicationmap_additional.conf`
- `features_applicationmap_custom.conf`

The only safe files to edit are that ones that have the word `custom` in them; these files are reserved for custom code and are not overwritten on new changes.



vm_general.inc

/etc/asterisk /var/www/html/panel /etc /tftpboot Re-Read Configs

vm_general.inc
Header

Edit: vm_general.inc

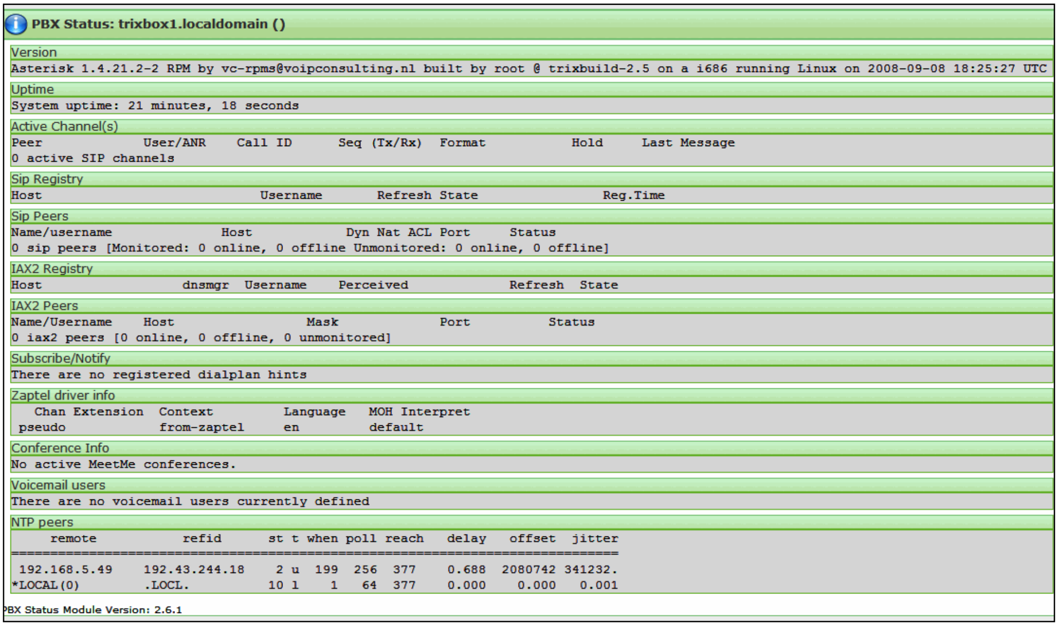
```
; 1st listed format gets emailed
format=wav49|wav

attach=yes
pbxskip=yes                     ; Don't put [PBX]: in the subject line
serveremail=vm@asterisk         ; Who the e-mail notification should appear to come from
fromstring=Voicemail System ; Real name of email sender
maxmessage=180                  ; max length of vm message
minmessage=3                            ; Minimum length of a voicemail message in seconds
maxsilence=5                    ; Wait for 5 silent seconds and end the voicemail
silencethreshold=128            ; What do we consider to be silence
skipms=3000                                     ; How many miliseconds to skip forward/back when rew/f:
review=yes                                      ; Allow sender to review/rerecord their message before
operator=yes                            ; Allow caller to press 0
nextaftercmd=yes                        ; Automagically play next message when current message is delet
```

128
180
3
3000
5

Update

PBX Status

While the System Status page has some basic information about the current status of the PBX system, the PBX Status page gives you even more information. You can get all of this information directly from the Asterisk CLI if you want, but this page will give you the most common items you will normally want to look at, without having to go to the console or SSH into the system. The following is a screenshot of the **PBX Status** page:



PBX Status: trixbox1.localdomain ()

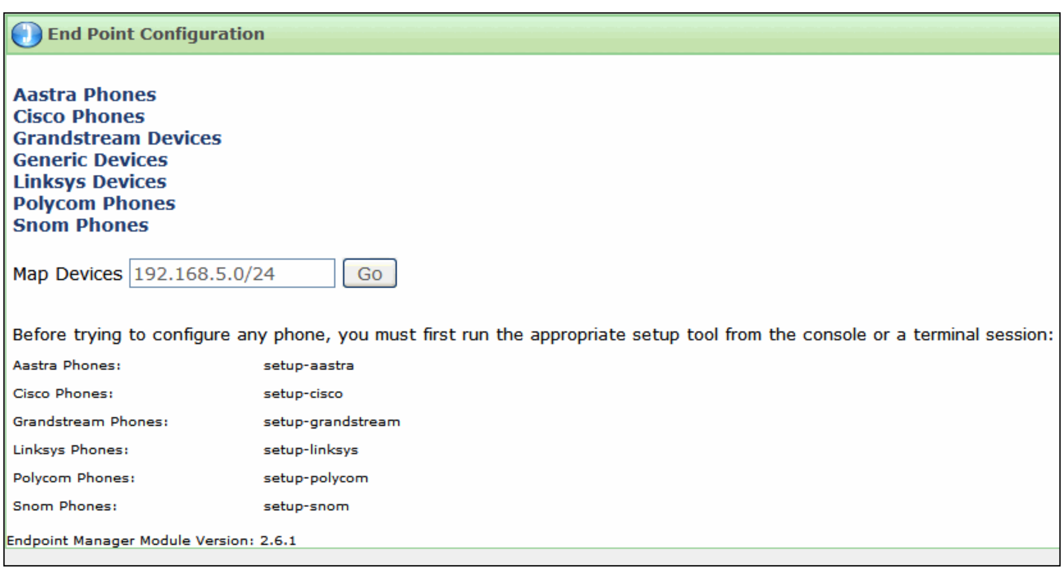
Version
```
Asterisk 1.4.21.2-2 RPM by vc-rpms@voipconsulting.nl built by root @ trixbuild-2.5 on a i686 running Linux on 2008-09-08 18:25:27 UTC
```
Uptime
```
System uptime: 21 minutes, 18 seconds
```
Active Channel(s)
```
Peer             User/ANR    Call ID      Seq (Tx/Rx)  Format           Hold     Last Message
0 active SIP channels
```
Sip Registry
```
Host                                Username       Refresh State                Reg.Time
```
Sip Peers
```
Name/username             Host                Dyn Nat ACL Port     Status
0 sip peers [Monitored: 0 online, 0 offline Unmonitored: 0 online, 0 offline]
```
IAX2 Registry
```
Host                      dnsmgr  Username    Perceived             Refresh  State
```
IAX2 Peers
```
Name/Username    Host                 Mask                Port          Status
0 iax2 peers [0 online, 0 offline, 0 unmonitored]
```
Subscribe/Notify
```
There are no registered dialplan hints
```
Zaptel driver info
```
   Chan Extension  Context         Language   MOH Interpret
 pseudo            from-zaptel     en         default
```
Conference Info
```
No active MeetMe conferences.
```
Voicemail users
```
There are no voicemail users currently defined
```
NTP peers
```
     remote           refid      st t when poll reach   delay   offset  jitter
==============================================================================
 192.168.5.49    192.43.244.18    2 u  199  256  377    0.688  2080742 341232.
*LOCAL(0)        .LOCL.          10 l    1   64  377    0.000    0.000   0.001
```
PBX Status Module Version: 2.6.1

Endpoint Manager

The Endpoint Manager is one of the coolest tools available in trixbox CE; it allows you to quickly and easily set up a large number of phones within your office. The **Endpoint Manager** module is found in the **Admin** section under the **PBX** menu. The Endpoint Manager will scan your network and find the available phones, display them to you in a list, and allow you to assign extensions that you've already created to the phones. After you have set up the phones in the Endpoint Manager, you can then reboot the phones and have them automatically come up configured to the extension that you assigned.

At the time of this writing, the Endpoint Manager will support Aastra, Cisco, Polycom, SNOM, and Linksys phones. To get started with the Endpoint Manager, you first need to run a script from the command line; there is a different script for each phone brand. The following is a list of the scripts that are available that you need to run before the Endpoint Manager will function properly:

Phone Brand	Script
Aastra	`setup-aastra`
Cisco	`setup-cisco`
Grandstream	`setup-grandstream`
Linksys	`setup-linksys`
Polycom	`setup-polycom`
SNOM	`setup-snom`

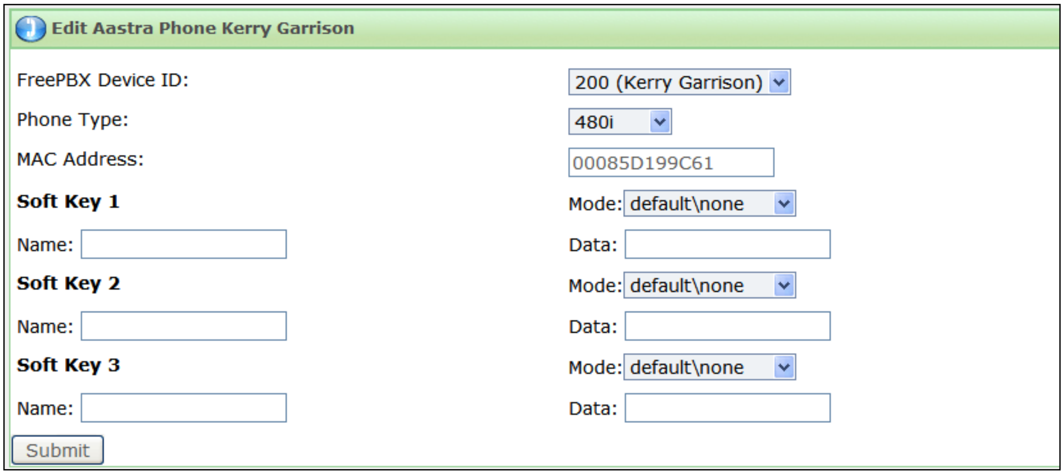


Once you have run one of the initial setup scripts, the next thing to do is go into the Endpoint Manager and click on the **Go** button; this will scan your network and find all of the available phones. When you click on a **MAC Address** in the list of available phones, the Endpoint Manager will then take you to another screen that will allow you to configure that particular phone. If you already have phones that have been preconfigured, you'll see the phone name extension and the phone type listed along with the other phones.

End Point Map

MAC Address	IP Address	vendor	Phone Name	Extension	Phone Type
00085D1981D3	192.168.5.11	Aastra			
000E08CAE25E	192.168.5.40	Linksys/Sipura			
000B82152EEE	192.168.5.145	Grandstream	Andrew Gillis	203	GXP2000
00085D10BD1A	192.168.5.159	Aastra			
0004132A0FF8	192.168.5.202	Snom			
000E08EABCC2	192.168.5.223	Linksys/Sipura			
00085D199C61	192.168.5.244	Aastra	Kerry Garrison	200	480i
000E08DCFCE0	192.168.5.248	Linksys/Sipura		204	SPA942

Different brands of phones will have different types of configuration options available; for example, the Aastra phones have a large number of soft keys that can be used for speed dials, BLF, Do Not Disturb, or other programmable options. When you click on the MAC address of the phone, you'll be given a configuration screen for that particular phone. From there you need to assign the extension number, select the phone type, set up any options that are specific to that phone, and save your changes.



For the Endpoint Manager to be fully effective, you need to make sure that your network is set up properly. You should make sure that you have a DHCP server capable of pushing out option 66. If you are using an inexpensive router then it may not be able to support the option 66 feature. If this is the case, you may be able to use the trixbox CE DHCP module, as this module will set up a DHCP server with all of the necessary configurations. We will take a look at the DHCP module later on in this chapter.

A DHCP server can be configured to issue option 66, and then make sure that setting is pointed to your trixbox CE server. This will allow the phones to find the trixbox CE system and grab their configuration files from it.

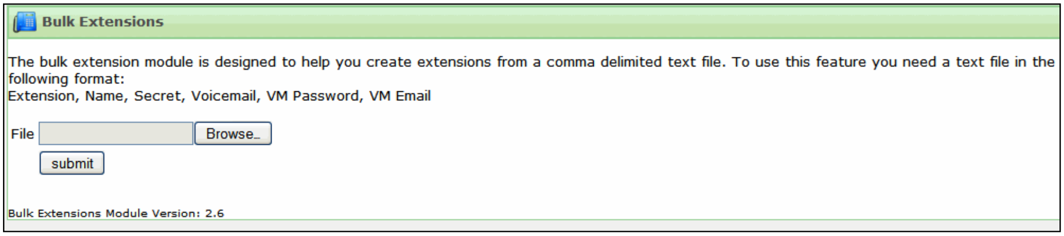
Bulk Extensions

If you need to create a large number of extensions all at once, the Bulk Extensions module is available to help you out. The **Bulk Extensions** module is found in the **Admin** section under the **PBX** menu. In order to use the Bulk Extensions module you will need to create a comma delimited text file using the following format:

Extension, Name, Secret, Voicemail, VM Password, VM Email

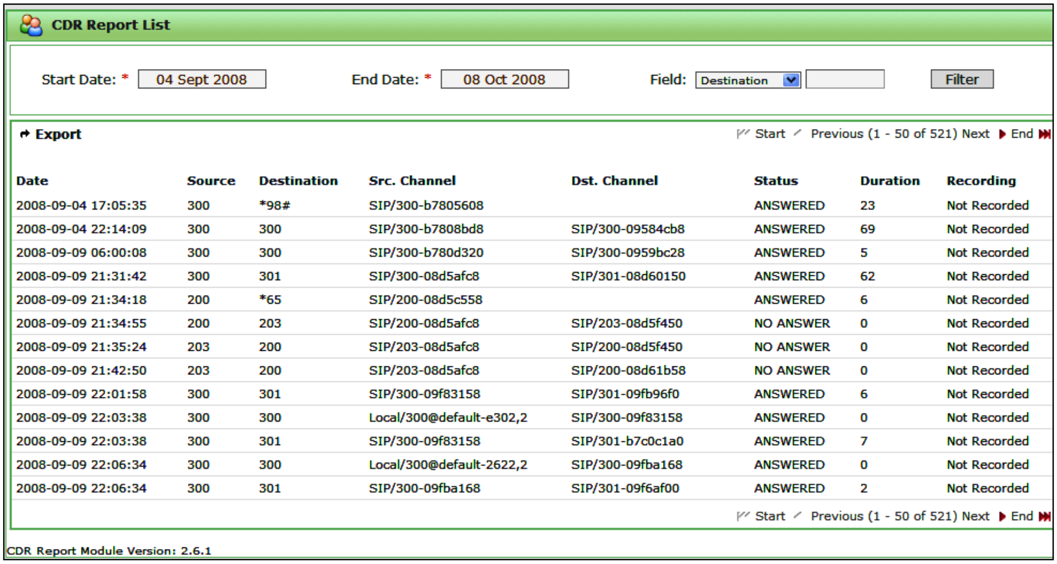
The following is an example of a working CSV file:

```
200,Kerry Garrison,432783,Enabled,2222,kg_email@tdp.com
202,Andrew Gillis,54544,Enabled,2222,ag_email@tdp.com
```



CDR Reports

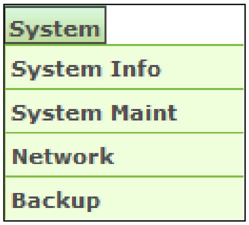
Call Detail Records reports are a key tool to seeing who is doing what on your phone system. The Reports tool in the PBX Settings module is much more advanced, but the CDR Reports module is a fast and simple way to run some basic reports. Secondly, if a call was recorded, this module will display a link to the file for you to play.



Date	Source	Destination	Src. Channel	Dst. Channel	Status	Duration	Recording
2008-09-04 17:05:35	300	*98#	SIP/300-b7805608		ANSWERED	23	Not Recorded
2008-09-04 22:14:09	300	300	SIP/300-b7808bd8	SIP/300-09584cb8	ANSWERED	69	Not Recorded
2008-09-09 06:00:08	300	300	SIP/300-b780d320	SIP/300-0959bc28	ANSWERED	5	Not Recorded
2008-09-09 21:31:42	300	301	SIP/300-08d5afc8	SIP/301-08d60150	ANSWERED	62	Not Recorded
2008-09-09 21:34:18	200	*65	SIP/200-08d5c558		ANSWERED	6	Not Recorded
2008-09-09 21:34:55	200	203	SIP/200-08d5afc8	SIP/203-08d5f450	NO ANSWER	0	Not Recorded
2008-09-09 21:35:24	203	200	SIP/203-08d5afc8	SIP/200-08d5f450	NO ANSWER	0	Not Recorded
2008-09-09 21:42:50	203	200	SIP/203-08d5afc8	SIP/200-08d61b58	NO ANSWER	0	Not Recorded
2008-09-09 22:01:58	300	301	SIP/300-09f83158	SIP/301-09fb96f0	ANSWERED	6	Not Recorded
2008-09-09 22:03:38	300	300	Local/300@default-e302,2	SIP/300-09f83158	ANSWERED	0	Not Recorded
2008-09-09 22:03:38	300	301	SIP/300-09f83158	SIP/301-b7c0c1a0	ANSWERED	7	Not Recorded
2008-09-09 22:06:34	300	300	Local/300@default-2622,2	SIP/300-09fba168	ANSWERED	0	Not Recorded
2008-09-09 22:06:34	300	301	SIP/300-09fba168	SIP/301-09f6af00	ANSWERED	2	Not Recorded

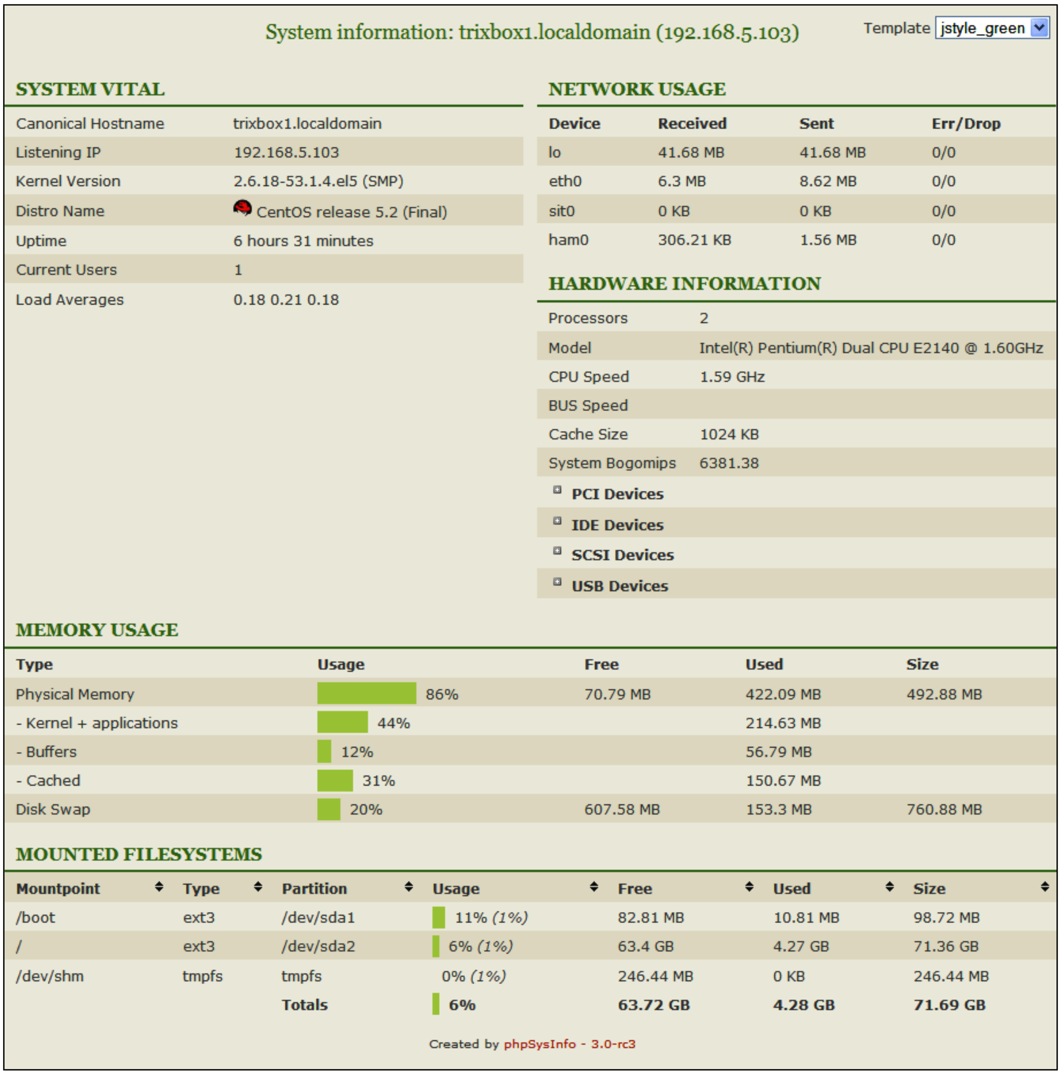
System menu

The **System** menu has tools and reporting on items that are more system oriented rather than specific to the PBX functionality. This is where you will find the System Information modules, a tool to reset/reboot your system, a tool for managing the network interface cards, a backup utility, and if you have installed it separately, a tool for managing a DHCP server.



System Info

The System Status page has some of the information from this section on it, but again, this is a specialized tool that has much more information available if you need to get in and see more information about your system and how it is performing.



System information: trixbox1.localdomain (192.168.5.103)

Template jstyle_green

SYSTEM VITAL

Canonical Hostname	trixbox1.localdomain
Listening IP	192.168.5.103
Kernel Version	2.6.18-53.1.4.el5 (SMP)
Distro Name	CentOS release 5.2 (Final)
Uptime	6 hours 31 minutes
Current Users	1
Load Averages	0.18 0.21 0.18

NETWORK USAGE

Device	Received	Sent	Err/Drop
lo	41.68 MB	41.68 MB	0/0
eth0	6.3 MB	8.62 MB	0/0
sit0	0 KB	0 KB	0/0
ham0	306.21 KB	1.56 MB	0/0

HARDWARE INFORMATION

Processors	2
Model	Intel(R) Pentium(R) Dual CPU E2140 @ 1.60GHz
CPU Speed	1.59 GHz
BUS Speed	
Cache Size	1024 KB
System Bogomips	6381.38
PCI Devices	
IDE Devices	
SCSI Devices	
USB Devices	

MEMORY USAGE

Type	Usage	Free	Used	Size
Physical Memory	86%	70.79 MB	422.09 MB	492.88 MB
- Kernel + applications	44%		214.63 MB	
- Buffers	12%		56.79 MB	
- Cached	31%		150.67 MB	
Disk Swap	20%	607.58 MB	153.3 MB	760.88 MB

MOUNTED FILESYSTEMS

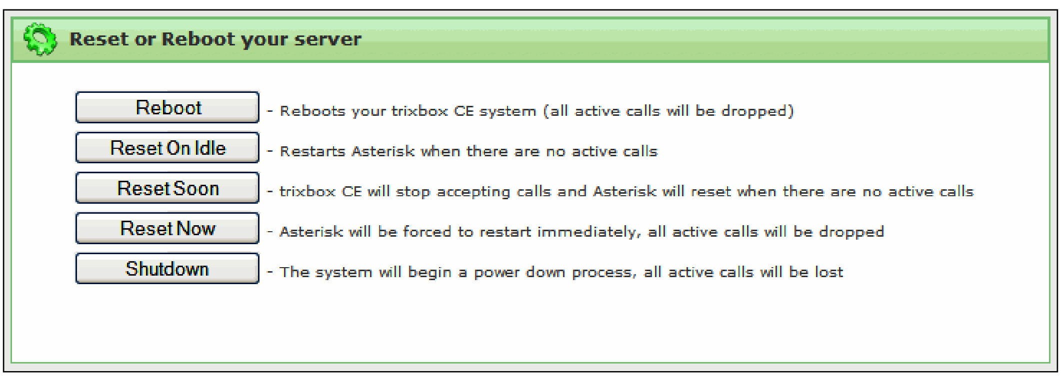
Mountpoint	Type	Partition	Usage	Free	Used	Size
/boot	ext3	/dev/sda1	11% *(1%)*	82.81 MB	10.81 MB	98.72 MB
/	ext3	/dev/sda2	6% *(1%)*	63.4 GB	4.27 GB	71.36 GB
/dev/shm	tmpfs	tmpfs	0% *(1%)*	246.44 MB	0 KB	246.44 MB
		Totals	**6%**	**63.72 GB**	**4.28 GB**	**71.69 GB**

Created by phpSysInfo - 3.0-rc3

System Maint

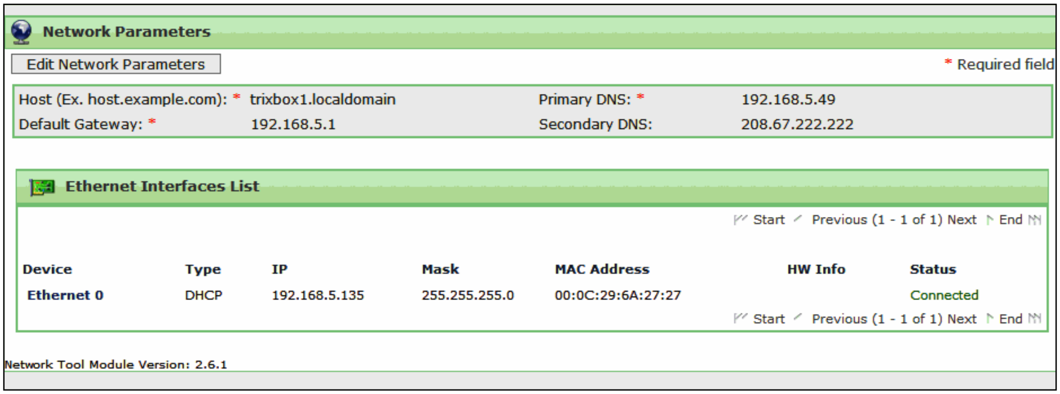
If you need to restart Asterisk, reboot the machine, or even shut down your system remotely, the following options are available to you from this page:

- **Reboot**: Drops all calls immediately and reboots the entire system. This is the equivalent to being at the console and typing `reboot` or `shutdown -r now`.
- **Reset On Idle**: This option will wait until there are no current calls and will then tell Asterisk to restart. This is the equivalent to being in the Asterisk CLI and issuing the `restart when convenient` command.
- **Reset Soon**: This option will cause Asterisk to stop accepting any additional phone calls and will restart as soon as all of the current calls are completed. This is the equivalent to being in the Asterisk CLI and issuing the `restart gracefully`.
- **Restart Now**: This option will cause Asterisk to drop all current phone calls and restart immediately. This is the equivalent to being in the Asterisk CLI and issuing the `restart`.
- **Shutdown**: This option will cause Asterisk to drop all current phone calls and will cause the entire system to shutdown. This is the equivalent to the command line `shutdown -h now`.

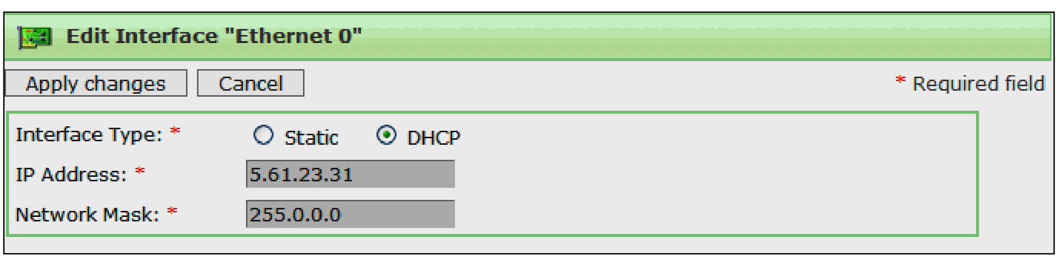


Network module

The Network settings module will allow you to set up and configure network settings for network cards within your trixbox CE system. The **Network** module is found in the **Admin** section under the **System** menu.

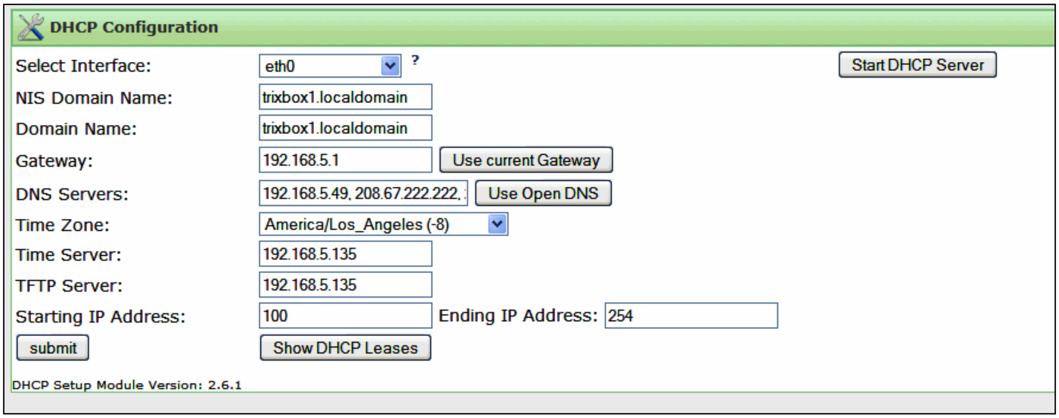


When a new trixbox CE system comes up, it is set to use DHCP by default; to change this you can click on the device that you would like to change—in the previous screenshot, you can see this as **Ethernet 0**. Once you click on the Ethernet device that you want to manage, you can then change it from **DHCP** to **static** and assign an IP address in the network mask.



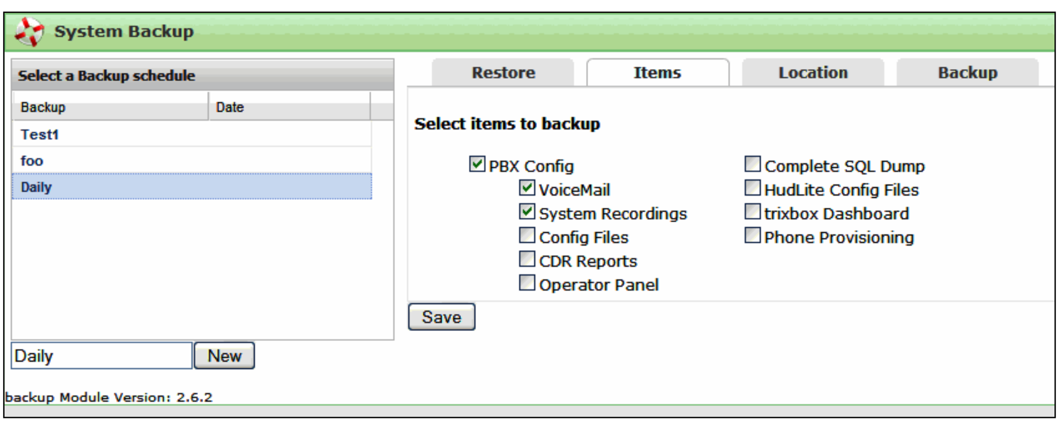
DHCP module

As we saw earlier in this chapter, if you want your phones to be able to use the Endpoint Manager effectively, you need to have a server that can serve out DHCP option 66. The DHCP module for trixbox CE will allow you to do this from the trixbox system itself. You will want to make sure that you do not have any other devices on your network that are serving out DHCP IP addresses otherwise you'll end up with a conflict. The DHCP module is an optional tool and is not included in the normal ISO distribution. To install the DHCP module, you'll first need to go to the Package Manager and install the DHCP manager package. Once installed, you can use the DHCP module to manage your DHCP settings.



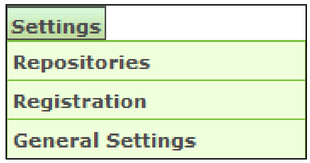
Backup module

As any good IT person would know, you do need a good backup of any system that you are in control of. Just because trixbox is a phone system does not mean you don't need a good backup of all your settings, system recordings, and voicemail files. To use the backup module, go to the **System** menu and select **Backup**. The first thing you'll need to do is create a new backup job; to do this enter a name and click on the **New** button. Once you have created a new backup job, you can then select the backup job from the list and define what items you would like it back up and what schedule you would like to use for this particular backup job.



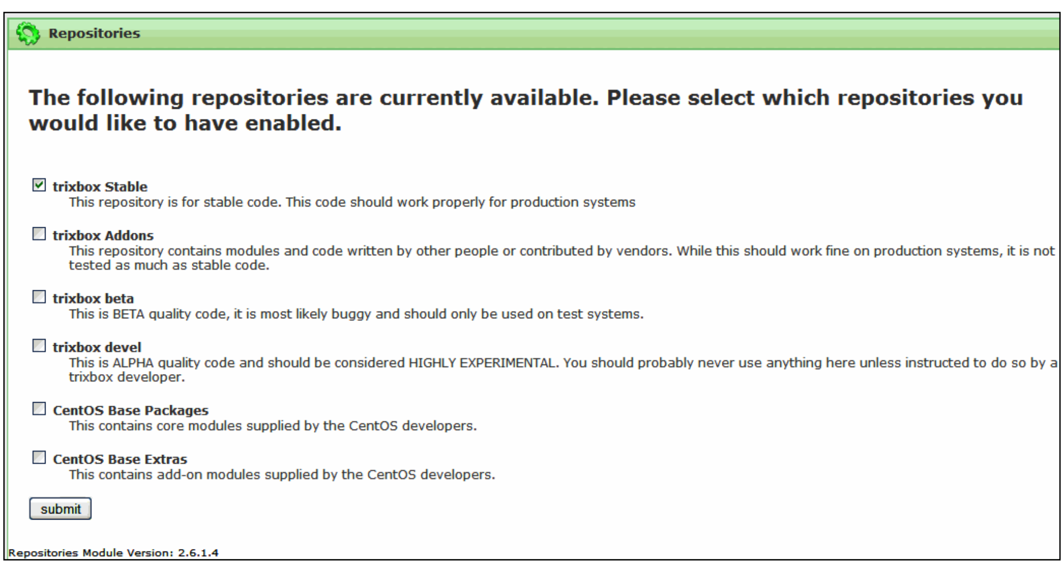
Settings menu

The **Settings** menu has a few modules in it that don't really make sense to have in other areas. The **Repositories** module lets you select which group of packages to make available to install; the **Registration** module allows you to register your system with Fonality, and the **General Settings** has several other options available.



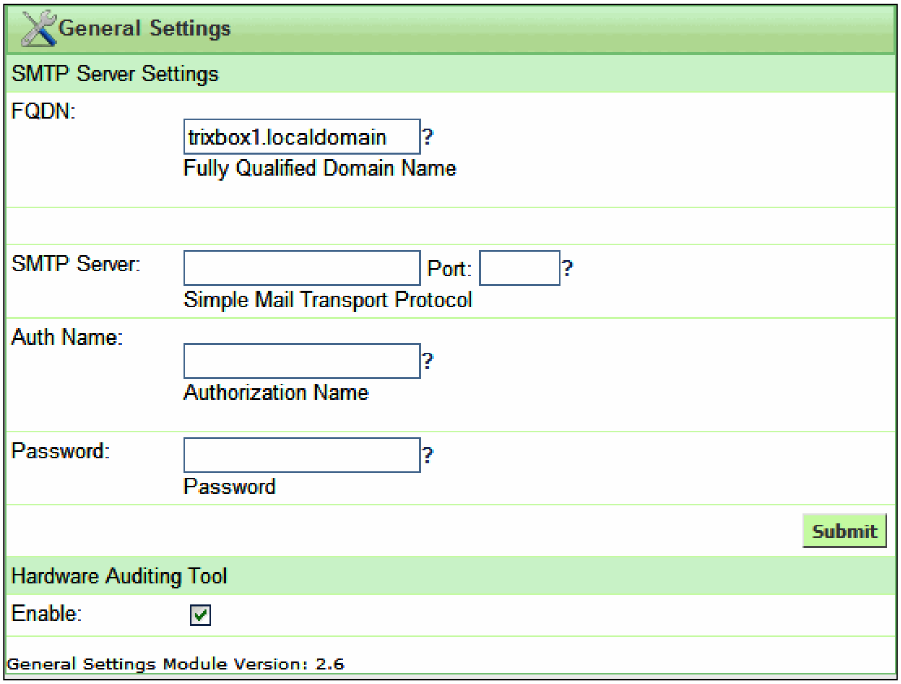
Repositories

The trixbox CE system contains several different repositories you can choose from to download different packages. There is a stable repository, which is the code that should work properly for production systems, an add-ons repository for extra tools that are not part of the normal distribution or are contributed by other people or vendors, a beta repository for beta code that is currently available for people to test, and a development repository for code that is under development and should be considered highly experimental and only be used if instructed to by a trixbox developer. You can choose which repositories you want to use by going to the **Settings** menu and selecting **Repositories**. For most systems, it's probably best to just leave it set for **trixbox Stable** to ensure that you are not using code that is untested.



General Settings

The General Settings module is a place where there are settings that don't quite belong anywhere else; currently the **General Settings** module allows you to set up **SMTP Server** settings for sending outbound mail, and also allows you to enable or disable the **Hardware Auditing Tool**.

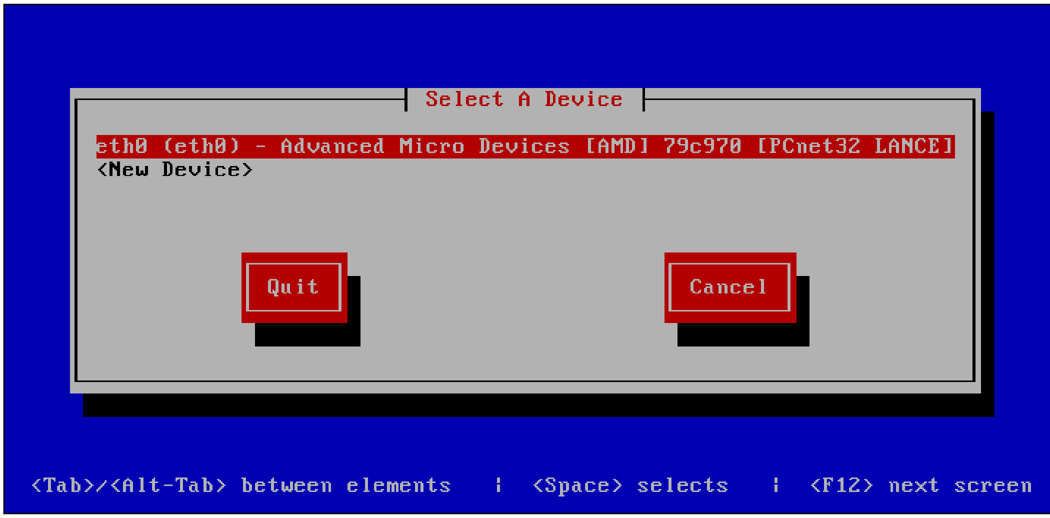


Command-line tools

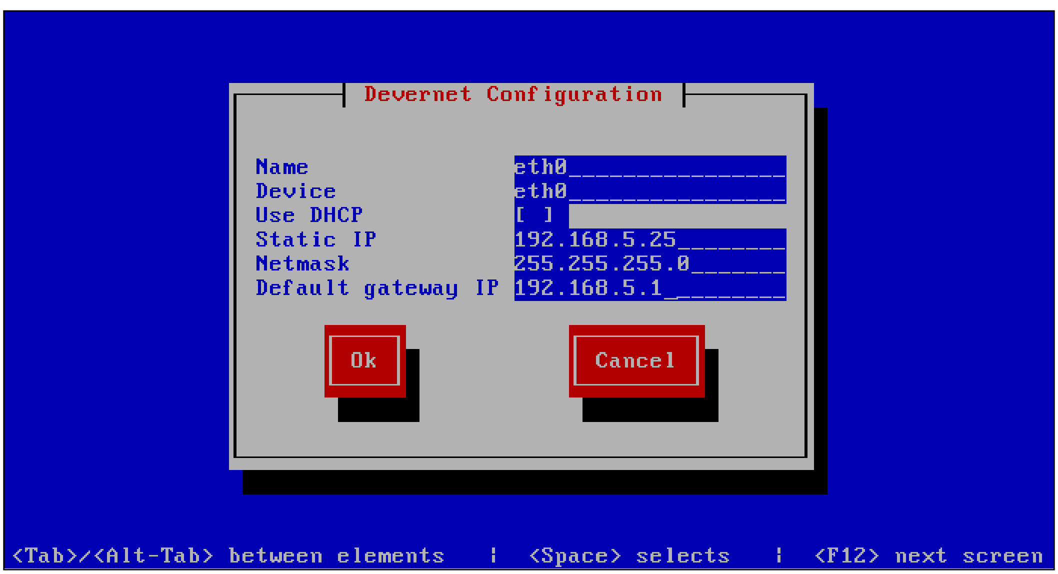
While the goal of trixbox CE is to make a system simple enough not to have to use command-line tools, there are still times when it is easier to use the command line than it is to build a graphic user interface for a simple task. As we saw with the Endpoint Manager, there are several scripts that you need to run in order to set up the configurations for the different phones. There are also command-line tools to set up networking, change passwords, and install the different mail servers that are available. In this section we will look at the different command-line tools and how they are used.

- **system-config-network**: While it is very easy to use the network management tool that we just looked at, you can also use a `system-config-network` script to manage your network settings as well. The first screen you will get will ask you to choose which network device you want to manage.

DNS server entries cannot be changed here; if you need to change what DNS servers your system is using, you need to edit the `/etc/resolv.conf` file manually. Any change to that file takes effect immediately.



The second screen will allow you to change the **Name** of the **Device**, set **DHCP** or a **Static IP**, set the network mask (**Netmask**), and set the **Default Gateway IP**.



After making any changes to your network configuration, you will need to run the `service network restart` command in order for the changes to take effect.

- **passwd-maint**: For security reasons you may want to change the password that is used to get into the Admin section. The `passwd-maint` script will allow you to do this easily. When you run the command, it will prompt you for a new password, ask you to confirm the password again, and then will update the `maint` password if the two passwords you entered match.
- **passwd**: The `passwd` command is used to change the password for system accounts. Running the command with no parameters will change the password for the currently logged in user. This is the easiest way to change the root password.
- **install-hudlite**: This command will install the HUDLite server and admin components. When HUD3 is released, this command will be replaced with a means of installing the upcoming HUD3 components.
- **install-postfix / install-sendmail**: These commands allow you to switch the internal mail system between postfix and sendmail depending on your preference. As of trixbox CE **2.6.1**, the default installation will include postfix; older versions installed sendmail by default.

Summary

This chapter introduced you to the tools that are included within trixbox CE and make it different from other distributions of Asterisk that have much less functionality. With these additional tools, you are easily able to set up, manage, maintain, and get reports on your system. With the ultimate goal of the trixbox CE project being the ability to do everything you need to do without having to use the command line, these tools are a great start and continue to improve with each release.

14 Designing a good call flow

Once you have got past all of the technical issues of building a system, the real make-it-or-break-it of a successful installation is the design of the inbound call flow. If the system has to be modified on a regular basis or causes too much frustration for the caller, then it should have been designed better in the first place. A good design can also make a big difference to the impression people get when they call in. Take a look at the two examples here and see which you think is a better design:

Example 1:

Thank you for calling the Acme Widget Company. For Bob, dial 200; for Sue, dial 201; for David, dial 202.

Example 2:

Thank you for calling the Acme Widget Company. For sales, press 1; for accounting, press 2; for support, press 3; for a company directory, press the pound key.

By listing off all of the employees during the main greeting, you will have to change the greeting anytime people leave the company or if you hire new people. It also announces the direct extension numbers to people whom you may not want, and it also does not allow multiple people to answer a call even if they may have the ability to do so.

In this chapter, we will go over the components that make up a call flow, how to use the different components, and how the components interact with each other to make a good call flow.

Call-flow feature review

In the previous chapters we saw the different call-flow components such as ring groups, queues, announcements, and the IVR tool to help link these pieces together. In this section we'll review the different components to make sure you have a solid understanding of how and when they should be used.

- **Announcements**: This component plays back a system recording and then sends the caller to a predetermined destination
- **Queues**: Queues are best used when there are more incoming calls than agents available to answer the calls
- **Ring Groups**: Ring groups are best used when there are more agents available to answer phone calls than there are calls coming into the system
- **Time Conditions**: Time conditions are used to automatically switch between menus that are played during normal office hours and menus that are played during non-business hours
- **IVR Tool**: We use the IVR tool to create menus and submenus that the callers use to traverse the system, ending up at extensions, rings groups, and so on

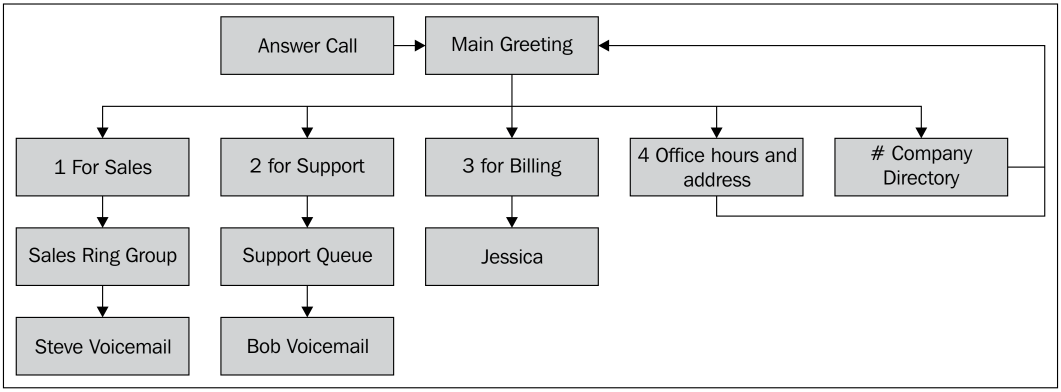
Flowcharting your call flow

The best way to ensure that you are going to have a good call flow is to use a good flowcharting tool to layout the design before you begin programming it. Some good tools to use for flowcharting include:

- Microsoft Visio
- Microsoft PowerPoint
- `http://gliffy.com`
- OpenOffice Draw

Getting your call flow down into a flowchart not only makes it easier to program (because you have a template from which to work) but you also have the ability to have signoff from everybody in your company that has a stake in the design on how the system should work before you even get started. The following flowchart is an example of a basic call flow; you can see that it starts by answering a call, playing a message providing a menu of different options to the caller, and you will see how those options relate to each other.

Once you have your call flow all designed, you begin programming in reverse, that is to say, you need to create all the end pieces and work your way back to the beginning. This methodology is called *Design forward / program in reverse*.



Design guidelines

There are some simple design guidelines you can use that will help you design a good efficient call flow.

- **Keep each level simple**: The human brain remembers things much better in groups of three to four; with this in mind, we should always strive to have no more than three menu choices per level.
- **Have good, clean recordings**: If you are not happy with the sound of the voices of anyone in your company, then by all means go to an outside source. Alison Smith, who recorded all of the system sounds in the system, is available for professional voice recordings at `http://theivrvoice.com`.
- **Never let the caller hear more than three ring tones**: The duration between ringing sounds is approximately six seconds; once someone has waited for more than three rings, or she starts to get impatient, and many will begin hanging up at this point.

Design scenarios

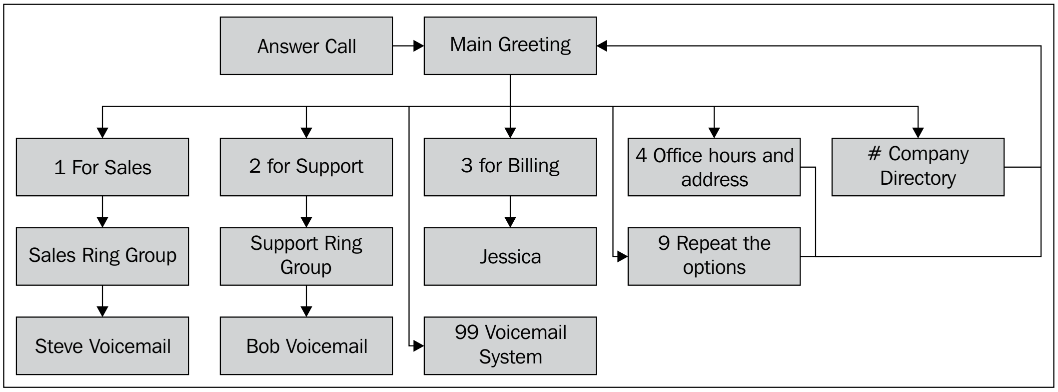
Let's take a look at several different design scenarios; we will start off with a very basic one and slowly build it into a more complex design.

Scenario 1:

This small company has only a few employees and each may take calls for different departments; since they would like to grow, they would like the system to be able to grow with them without having to make constant changes. They would like calls to go to a sales group, support group, a billing person, office hours, and directions announcement, an option to repeat the message, and an option to play the company directory. They would also like a way to quickly call in and check voicemail messages.

- Sales Group
- Support Group
- Billing
- Office Hours Announcement
- Remote access to voicemail
- Repeat the options
- Company Directory

Now that we know what they want, we can begin to design the flowchart. We need to make sure we have included all of the features that they want and that it is laid out in a way that is easy to read. The following flowchart should fit this customer's needs quite nicely.

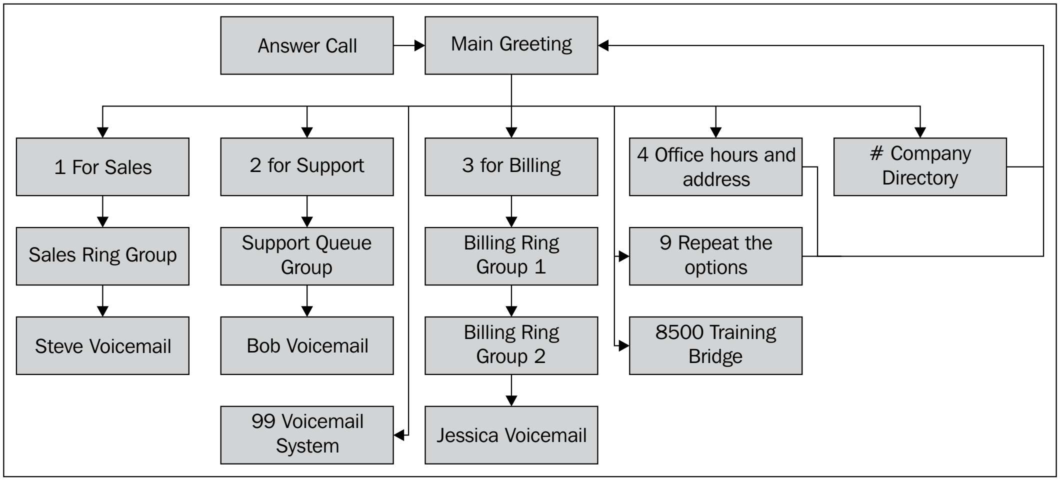


Another way of looking at this is to use a bulleted list:

- Answer Call
    - Play Greeting
        - Press 1 for sales
            - Go to sales ring group
        - Press 2 for support
            - Go to support ring group
        - Press 3 for billing
            - Go to Jessica's extension
        - Press 4 for office hours and address
            - Play a recorded message
            - Return to main menu
        - Press # for company directory
        - Press 9 to repeat these options
            - Return to main menu
- Press 99 to access voicemail system (not announced in greeting)

Scenario 2:

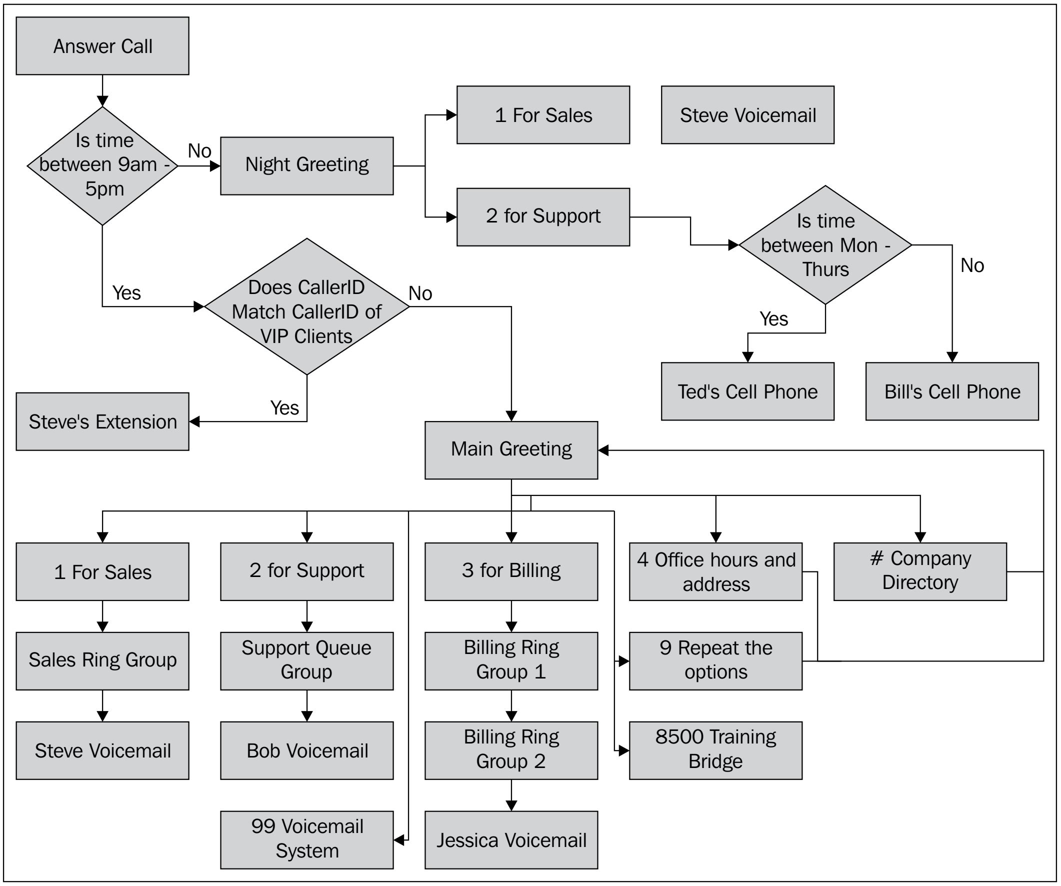
The company is growing and the current phone system needs some modifications in order for it to be more effective. For example, the support group is peaked at times where technicians are overloaded with calls; Jessica has a new assistant and Jessica wants calls to go to Sarah if Jessica doesn't answer the call, but if Sarah doesn't answer the call then Jessica wants the call to go back into her voicemail. The company is now doing training for clients and needs a training conference bridge that customers can call into. Let's take a look at how this changes the flowchart:



At first glance you may not be able to see the differences between the two flowcharts. In order to handle the peak times when technicians are overloaded with calls and cannot answer all the inbound calls, we've changed the support ring group to a support queue; this will allow more calls to come into the system than there are agents available to answer the calls—this should solve the problem for the support group. Solving Jessica's problem was a little more complex; in order to have the system ring her extension several times then ring someone else's extension several times and go back to Jessica's voicemail, we needed to create two different ring groups. The first group, which we called the billing group 1, contains only Jessica's extension with the failover destination set to billing group 2. The second group, which we called billing group 2, contains only Sarah's extension with the failover destination set to Jessica's voicemail. We have also added a conference room at extension 8500.

Scenario 3:

As the company has grown, the need for a more complex call flow has evolved. In this new design, the company would like a different message to be played after business hours so that someone can reach a support tech for emergency support or leave a message for sales. After hours Ted is on call from Monday through Thursday and Bill is on call from Friday night through Sunday night, the sales manager also wants calls from several VIP clients to go directly to the sales manager. So let's take a look at our new call flow design.



As you can see, the design has become much more complicated: to begin with, a time condition checks to see if the time is within the normal business hours; if so, then the system proceeds with the normal greeting and IVR. If the time is outside of normal business hours, then the system plays a night greeting with a different set of options. If somebody selects support during the night hours, then another time condition checks to see whether the system should call Ted or Bill. Back in the daytime menu, inbound routing rules check for the caller ID of the VIP clients; if there is a match then the call is routed directly to Steve's extension.

Summary

As you can see, as the system gets more complicated (and these examples aren't really that complex), the system is basically unmanageable without a good flowchart. By laying out the design in a flowchart, you can easily see where the logic points are, determine how long people will be on hold, how many transfers a person will go through to get to someone, and overall, how easy, or difficult, your system will be to manage.

15
Securing your trixbox server

Even though a trixbox system is a phone system, it is still a basic computer system like any other. One of the problems that we face is that extensions and VoIP service providers typically come into the system over the open Internet; this means that certain aspects of our system are wide open to the outside world. During the week that I was writing this particular chapter, several new scripts came out that allowed people to scan machines over the Internet, find systems that are running Asterisk, get the list of available extensions, and then hack the passwords. These tools allow a malicious hacker to get into your system and start making long-distance phone calls. There were numerous instances of companies with phone bills reaching into the thousands and even tens of thousands of dollars. Because of issues like this, it is more imperative than ever that you understand how to properly secure your trixbox server from the outside world.

Start with a good firewall

Never have your trixbox system exposed completely on the open Internet; always make sure it is behind a good firewall. While many people think that because trixbox is running on Linux, it is totally secure, Linux, like anything else, has its share of vulnerabilities, and if things are not configured properly, is fairly simple for hackers to get into. There are really good open-source firewalls available, such as *pfSense*, *Viata*, and *M0n0Wall*. Any access to system services, such as HTTP or SSH, should only be done via a VPN or using a pseudo-VPN such as Hamachi. The best designed security starts with being exposed to the outside world as little as possible. If we have remote extensions that cannot use VPNs, then we will be forced to leave SIP ports open, and the next step will be to secure those as well.

Stopping unneeded services

Since trixbox CE is basically a stock installation of CentOS Linux, very little hardening has been done to the system to secure it. This lack of security is intentional as the first level of defence should always be a good firewall. Since there will be people who still insist on putting the system in a data center with no firewall, some care will need to be taken to ensure that the system is as secure as possible. The first step is to disable any services that are running that could be potential security vulnerabilities.

We can see the list of services that are used with the `chkconfig –list` command.

```
[trixbox1.localdomain rules]# chkconfig --list
anacron         0:off   1:off   2:on    3:on    4:on    5:on    6:off
asterisk        0:off   1:off   2:off   3:off   4:off   5:off   6:off
avahi-daemon    0:off   1:off   2:off   3:off   4:off   5:off   6:off
avahi-dnsconfd  0:off   1:off   2:off   3:off   4:off   5:off   6:off
bgpd            0:off   1:off   2:off   3:off   4:off   5:off   6:off
capi            0:off   1:off   2:off   3:off   4:off   5:off   6:off
crond           0:off   1:off   2:on    3:on    4:on    5:on    6:off
dc_client       0:off   1:off   2:off   3:off   4:off   5:off   6:off
dc_server       0:off   1:off   2:off   3:off   4:off   5:off   6:off
dhcpd           0:off   1:off   2:off   3:off   4:off   5:off   6:off
dhcrelay        0:off   1:off   2:off   3:off   4:off   5:off   6:off
ez-ipupdate     0:off   1:off   2:off   3:off   4:off   5:off   6:off
haldaemon       0:off   1:off   2:off   3:on    4:on    5:on    6:off
httpd           0:off   1:off   2:off   3:on    4:on    5:on    6:off
ip6tables       0:off   1:off   2:off   3:off   4:off   5:off   6:off
iptables        0:off   1:off   2:off   3:off   4:off   5:off   6:off
isdn            0:off   1:off   2:off   3:off   4:off   5:off   6:off
kudzu           0:off   1:off   2:off   3:on    4:on    5:on    6:off
lm_sensors      0:off   1:off   2:on    3:on    4:on    5:on    6:off
lvm2-monitor    0:off   1:on    2:on    3:on    4:on    5:on    6:off
mDNSResponder   0:off   1:off   2:off   3:on    4:on    5:on    6:off
mcstrans        0:off   1:off   2:off   3:off   4:off   5:off   6:off
mdmonitor       0:off   1:off   2:on    3:on    4:on    5:on    6:off
mdmpd           0:off   1:off   2:off   3:off   4:off   5:off   6:off
memcached       0:off   1:off   2:on    3:on    4:on    5:on    6:off
```

```
messagebus      0:off   1:off   2:off   3:on    4:on    5:on    6:off
multipathd      0:off   1:off   2:off   3:off   4:off   5:off   6:off
mysqld          0:off   1:off   2:off   3:on    4:on    5:on    6:off
named           0:off   1:off   2:off   3:off   4:off   5:off   6:off
netconsole      0:off   1:off   2:off   3:off   4:off   5:off   6:off
netfs           0:off   1:off   2:off   3:on    4:on    5:on    6:off
netplugd        0:off   1:off   2:off   3:off   4:off   5:off   6:off
network         0:off   1:off   2:on    3:on    4:on    5:on    6:off
nfs             0:off   1:off   2:off   3:off   4:off   5:off   6:off
nfslock         0:off   1:off   2:off   3:on    4:on    5:on    6:off
ntpd            0:off   1:off   2:off   3:on    4:on    5:on    6:off
ospf6d          0:off   1:off   2:off   3:off   4:off   5:off   6:off
ospfd           0:off   1:off   2:off   3:off   4:off   5:off   6:off
portmap         0:off   1:off   2:off   3:on    4:on    5:on    6:off
postfix         0:off   1:off   2:on    3:on    4:on    5:on    6:off
rdisc           0:off   1:off   2:off   3:off   4:off   5:off   6:off
restorecond     0:off   1:off   2:on    3:on    4:on    5:on    6:off
ripd            0:off   1:off   2:off   3:off   4:off   5:off   6:off
ripngd          0:off   1:off   2:off   3:off   4:off   5:off   6:off
rpcgssd         0:off   1:off   2:off   3:on    4:on    5:on    6:off
rpcidmapd       0:off   1:off   2:off   3:on    4:on    5:on    6:off
rpcsvcgssd      0:off   1:off   2:off   3:off   4:off   5:off   6:off
saslauthd       0:off   1:off   2:off   3:off   4:off   5:off   6:off
snmpd           0:off   1:off   2:off   3:off   4:off   5:off   6:off
snmptrapd       0:off   1:off   2:off   3:off   4:off   5:off   6:off
sshd            0:off   1:off   2:on    3:on    4:on    5:on    6:off
syslog          0:off   1:off   2:on    3:on    4:on    5:on    6:off
vsftpd          0:off   1:off   2:off   3:off   4:off   5:off   6:off
xinetd          0:off   1:off   2:off   3:on    4:on    5:on    6:off
zaptel          0:off   1:off   2:on    3:on    4:on    5:on    6:off
zebra           0:off   1:off   2:off   3:off   4:off   5:off   6:off
```

The highlighted lines are services that are started automatically on system startup. The following list of services is required by trixbox CE and should not be disabled:

- `Anacron`
- `crond`
- `haldaemon`
- `httpd`
- `kudzu`
- `lm_sensors`
- `lvm2-monitor`
- `mDNSResponder`
- `mdmonitor`
- `memcached`
- `messagebus`
- `mysqld`
- `network`
- `ntpd`
- `postfix`
- `sshd`
- `syslog`
- `xinetd`
- `zaptel`

To disable a service, we use the command `chkconfig <servicename> off`. We can now turn off some of the services that are not needed:

```
chkconfig ircd off
chkconfig netfs off
chkconfig nfslock off
chkconfig openibd off
chkconfig portmap off
chkconfig restorecond off
chkconfig rpcgssd off
chkconfig rpcidmapd off
chkconfig vsftpd off
```

We can also stop the services immediately without having to reboot:

```
service ircd stop
service netfs stop
service nfslock stop
service openibd stop
service portmap stop
service restorecond stop
service rpcgssd stop
service rpcidmapd stop
service vsftpd stop
```

Securing SSH

A very large misconception is that by using SSH to access your system, you are safe from outside attacks. The security of SSH access is only as good as the security you have used to secure SSH access. Far too often, we see systems that have been hacked because their root password is very simple to guess (things like *password* or *trixbox* are not safe passwords). Any dictionary word is not safe at all, and substituting numbers for letters is very poor practice as well. So, as long as SSH is exposed to the outside, it is vulnerable. The best thing to do, if you absolutely have to have SSH running on the open Internet, is to change the port number used to access SSH. This section will detail the best methods of securing your SSH connections.

Create a remote login account

First off, we should create a user on the system and only allow SSH connections from it. The username should be something that only you know and is not easily guessed. Here, we will create a user called *trixuser* and assign a password to it. The password should be something with letters, numbers, symbols, and not based on a dictionary word. Also, try to string it into a sentence making sure to use the letters, numbers, and symbols. Spaces in passwords work well too, and are hard to add in scripts that might try to break into your server.

> A nice and simple tool for creating hard-to-guess passwords can be found at `http://www.pctools.com/guides/password/`.

```
[trixbox1.localdomain init.d]# useradd trixuser
[trixbox1.localdomain init.d]# passwd trixuser
```

Now, ensure that the new account works by using SSH to log in to the trixbox CE server with this new account. If it does not let you in, make sure the password is correct or try to reset it. If it works, continue on.

Only allowing one account access to the system over SSH is a great way to lock out most brute force attacks. To do this, we need to edit the file in `/etc/ssh/sshd_config` and add the following to the file.

```
AllowUsers trixuser
```

I like to edit the `PermitRootLogin` setting so that root can't log in over SSH. Remove the `#` from in front of the setting and change the `yes` to `no`.

```
PermitRootLogin no
```

Change the SSH port

Finally, I would recommend changing the `Port` setting from the standard `22`, which everyone knows as SSH, to something else. Be careful what you change it to; you don't want the port to conflict with a port in use or that might become in use. You can also attract more attention to the server if you put it on another known port than if you left it at `22`. In this example, we will use `2222`. Please decide your own port number to use on your system. The setting we edit is `Port 22` in `/etc/ssh/sshd_config`.

Remove the `#` from in front of the setting and change `22` to `2222`.

```
Port 2222
```

We need to restart `sshd` for the changes to take effect. Please use caution when changing these settings on a remote system that you can't easily get to. If there is an error in the `config`, it could cause `sshd` to not restart. To restart the SSH service for the new settings to take effect, use the following command:

```
service sshd restart
```

Now, test to make sure that you can get into the server over SSH. The root user should be denied access and only the user we created should be allowed to get in. Don't forget to change your SSH port to 2222 when connecting. In Putty, it is listed next to the IP address; on the command line, the flag is `-p port`.

Extension security

Although, in the examples you've seen throughout this book, the extensions use the same secret as the extension number, in practice this is a very big security hole as several scripts that are available look for exactly this setup when trying to attack

Asterisk-based systems. Make sure that you use a very strong password as your secret for each extension. In the next section, we will look at a set of tools that can be used to protect your system against extension attacks.

Additional security

With the advent of hacking scripts, you really cannot be too careful; if you have any remote extensions or VoIP trunks, it is now recommended that you set up tools to capture illegitimate login requests and block those IP addresses from getting into your system. One popular tool among trixbox CE users is **fail2ban**, and there is quite a bit of information in the trixbox forums about how to set it up. For the purpose of this book, we are going to look at APF and BFD as a more robust solution.

The following information is provided courtesy of Tim Yardley, the trixbox CE Build Engineer. Tim's recommendation is to use R-fx Networks, APF, and BFD for firewalling trixbox CE systems.

Links to their software can be found here.

APF: `http://rfxnetworks.com/apf.php`

BFD: `http://rfxnetworks.com/bfd.php`

APF stands for **Advanced Policy Firewall**. This is used to control iptables on the system to allow or disallow ports to be open. APF has additional features that make it stand out above the rest. **Reactive Address Blocking** (**RAB**), QoS (TOS), direct integration with BFD, and much more—see its site for full details.

BFD stands for **Brute Force Detection**. This is used to monitor any failed logins and block IP addresses from getting in. This runs as a cron daemon and works perfectly with APF.

Installing both of these applications is very simple. You can download both of them from the R-fx Networks links, uncompress them, and then run the `install.sh` script. Tim has also created an installer script that can be downloaded to your machine and run. This will install the latest and greatest APF/BFD. To get this script, you will need to use `wget` or another method to pull it off a web server. You will want to be logged into your system as root to use these commands:

- `wget http://engineertim.com/install_apf_bfd.sh`
- `chmod 755 install_apf_bfd.sh`
- `./install_apf_bfd.sh`

This will start the installation process for both APF and BFD. Once the scripts complete, you will be returned to a command prompt.

APF

Configuring APF is pretty easy, and we will look at few of the `config` file options in this section. Two of the options are covered in great detail on its web site and well-commented in the `conf.apf` file.

The `config` file for APF lives in `/etc/apf` and is called `conf.apf`.

We will need to edit the `conf.apf` file. If you have multiple network interfaces on your trixbox setup, you will want to set the IFACE_IN and IFACE_OUT to your external interface. This is the untrusted network interface that is connected to the Internet. If you have a second card, eth1, that is used for internal, trusted network, you can set the IFACE_TRUSTED to this interface.

To begin editing the file, use the following command:

```
nano /etc/apf/conf.apf
```

Please see the comments in the `conf.apf` if you are uncertain.

The setup script will try to properly determine which interface is used for the untrusted network and place it in the appropriate field. It is recommended to set the value of `SET_TRIM` to `0`. This value sets the total number of rules allowed inside of the *deny trust* system. It is designed to save memory and start time. With the default value of 50, the system will start to purge old rules once this number is met. With the inclusion of BFD, this number will generally climb past 50.

Setting this value to `0` will disable this feature.

```
SET_TRIM="0"
```

APF has the ability to do QoS on packets; this is defined with the TOS values in the `conf.apf` file. For SIP and IAX, I set the following:

```
TOS_8="21,20,80,4569,5060,10000_20000"
```

This also requires a small tweak to one of the `config` files, which I will discuss later in the chapter, in order to tag UDP packets.

If you changed the SSH port to a different number, we have to edit the `conf.apf` file to match this new port.

```
HELPER_SSH_PORT="2222"
```

Make sure to replace 2222 with the correct port number on which you decided to run SSH.

Ingress filtering is used to open inbound ports for access; both TCP and UDP have separate settings. For a trixbox setup, the following ports should be open; both TCP and UDP are listed. If you are not using TFTP, then do not have port 69 open. Do not forget to change the SSH port from 22, to the port you choose to run SSH on. Otherwise, you will be locked out; here we are using port 2222 from our last example. I have not included IAX ports in this setup. There is an easy way to ensure that only specific hosts can use IAX, which I will cover later. This is handy if you use IAX to do interoffice trunks, as I do, but don't want IAX ports open for the world to see.

```
IG_TCP_CPORTS="2222,69,80,5060,6600,10000_20000"
IG_UDP_CPORTS="69,5060,10000_20000"
```

Egress filtering is used to allow outbound filtering. I don't use egress filtering, and it will not be covered in this chapter. It is set to `EGF="0"`, or disabled by default. In the section of the `conf.apf` file called Imported Rules, there are settings for various feeds. Feeds are used so that many people can get information about malicious IP addresses as soon as one system reports them; this way if a script from a certain IP is attacking systems, often before the script gets a chance to get to you, your system has already blocked that IP address. Some of these feeds are very handy and I use them all. You can even set up your own custom feed that would allow you to adjust all of your servers with global deny rules. You can disable or enable this feature with the `USE_DS` setting—a `1` is enabled, a `0` is disabled.

We are now ready to start APF for the first time. If you start APF right now and something is wrong, it will disable itself in 5 minutes. This is called `DEVEL_MODE` and is the first setting in the `conf.apf` file. Leave this set to `1` until you are certain you can get in via SSH and things are working.

To save the configuration file, hit *Ctrl+O* to save and *Ctrl+X* to exit.

To see a list of command-line options, run `apf` without any flags.

```
[trixbox1.localdomain apf]# apf
apf(3402): {glob} status log not found, created
APF version 9.6 <apf@r-fx.org>
Copyright (C) 1999-2007, R-fx Networks <proj@r-fx.org>
Copyright (C) 2007, Ryan MacDonald <ryan@r-fx.org>
This program may be freely redistributed under the terms of the GNU GPL
usage /usr/local/sbin/apf [OPTION]
-s|--start ........................ load all firewall rules
-r|--restart ...................... stop (flush) & reload firewall rules
```

```
-f|--stop........ .................. stop (flush) all firewall rules
-l|--list .......................... list all firewall rules
-t|--status ........................ output firewall status log
-e|--refresh ....................... refresh & resolve dns names in trust
rules
-a HOST CMT|--allow HOST COMMENT ... add host (IP/FQDN) to
allow_hosts.rules and immediately load new rule into firewall
-d HOST CMT|--deny HOST COMMENT .... add host (IP/FQDN) to
deny_hosts.rules and immediately load new rule into firewall
-u|--remove HOST .................. remove host from
[glob]*_hosts.rules and immediately remove rule from firewall
-o|--ovars ......................... output all configuration options
```

To start APF, we issue the following command:

```
[trixbox1.localdomain apf]# apf -s
apf(3445): {glob} activating firewall
apf(3489): {glob} determined (IFACE_IN) eth0 has address 192.168.1.31
apf(3489): {glob} determined (IFACE_OUT) eth0 has address 192.168.1.31
apf(3489): {glob} loading preroute.rules
apf(3489): {resnet} downloading http://r-fx.ca/downloads/reserved.
networks
apf(3489): {resnet} parsing reserved.networks into
/etc/apf/internals/reserved.networks
apf(3489): {glob} loading reserved.networks
apf(3489): {glob} SET_REFRESH is set to 10 minutes
apf(3489): {glob} loading bt.rules
apf(3489): {dshield} downloading http://feeds.dshield.org/top10-2.txt
apf(3489): {dshield} parsing top10-2.txt into /etc/apf/ds_hosts.rules
apf(3489): {dshield} loading ds_hosts.rules
apf(3489): {sdrop} downloading http://www.spamhaus.org/drop/drop.lasso
apf(3489): {sdrop} parsing drop.lasso into /etc/apf/sdrop_hosts.rules
apf(3489): {sdrop} loading sdrop_hosts.rules
apf(3489): {glob} loading common drop ports
...........trimmed for this document.........
apf(3489): {glob} default (ingress) input drop
apf(3445): {glob} firewall initalized
apf(3445): {glob} !!DEVELOPMENT MODE ENABLED!! - firewall will flush
every 5 minutes.
```

We can see that APF has started, downloaded some rules from `dshield.org` and `spamhaus.org`, and then told us it is in `DEVELOPMENT MODE`. Now, test connecting to your server over SSH to ensure that you have set up the correct port number ingress. If you can't connect, you will have to wait 5 minutes and then APF will shutdown. Once you are sure you can get in with SSH, we can change the `conf.apf` file from `DEVEL_MODE="1"` to `DEVEL_MODE="0"` and restart/start APF. APF will start and not warn you about being in `DEVELOPMENT MODE`; your firewall should be good to go.

APF additional tweaks

This setup might not be ideal for everyone. If you connect to your provider over IAX, then you will definitely want to add the IAX ports to the `conf.apf`. However, if you have two or more systems that you connect to each other over IAX for interoffice connections, this is the way to go. This will work with static IP addresses and DynDNS setups alike. You can use a fully qualified DNS hostname or IP address. One of the flags for the `apf` command is `-a`, which is allow. This will globally allow a host to connect to this system, bypassing the firewall rules. I can't stress how handy this is. Some examples are allowing an SNMP query, IAX connections, or other ports that you do not want open, but need to allow specific hosts to connect to. To do this, just issue the following command and substitute your remote system IP address for the one I have here.

```
apf -a 192.168.1.216
```

This will allow the system `192.168.1.216` to connect to any port on the firewalled server, thereby bypassing the firewall rules. If you are running APF on both systems, be sure to do the same thing on the other host using the correct IP address.

APF also allows a system admin to block a host or a complete subnet. This is handy if you see someone attempting to connect to your machine over FTP, Telnet, SSH, and so on. To block a specific host, use the following; be sure to use the IP address you want to block.

```
apf -d 192.168.1.216
```

To block a complete subnet (CIDR), the command is very similar:

```
apf -d 202.86.128.0/24
```

This will block the entire subnet. You can sometimes get the subnet (CIDR) listing using a WHOIS on the IP address. You can also look up a CIDR by IP on Google or ripe.net. Be sure that the subnet that you are blocking is not the one you are using or you could lock yourself out.

TOS for UDP packets are not defined for APF. Only TCP packets have the TOS bit set. There is an easy way to fix this. In the /etc/apf/internals folder, there is a file called functions.apf. We need to edit this file manually. It is pretty straightforward as to what we need to change, so don't worry. There are several places we have to add a single line. Look for the TOS_ section in the functions.apf file. It will look like this:

```
if [ ! "$TOS_0" == "" ]; then
for i in `echo $TOS_0 | tr ',' ' '`; do
i=`echo $i | tr '_' ':'`
$IPT -t mangle -A PREROUTING -p tcp --sport $i -j TOS --set-tos 0
done
fi
```

We have to add the settings for UDP. We copy one line and change tcp to udp. A sample is below, highlighted.

```
if [ ! "$TOS_0" == "" ]; then
for i in `echo $TOS_0 | tr ',' ' '`; do
i=`echo $i | tr '_' ':'`
$IPT -t mangle -A PREROUTING -p tcp --sport $i -j TOS --set-tos 0
$IPT -t mangle -A PREROUTING -p udp --sport $i -j TOS --set-tos 0
done
fi
```

This additional line has to be done for all the TOS bits you are using. If you are using only TOS_8, then only worry about doing it for those. Make sure you do the tospostroute and tospreroute sections.

BFD

Brute Force Detection is used to capture illegitimate login attempts for services on the system. We see quite often a large number of SSH attempts into servers that haven't had the SSH port changed. These attempts are often an outside attempt to gain access by running dictionary attacks against common user names. These can now easily be stopped by using BFD.

If you ran the install_apf_bfd.sh, then BFD should be installed. The configuration file for BFD is located in /usr/local/bfd and is called conf.bfd. This file, like the one for APF, is heavily commented and covered in great detail on the R-fx Networks web site. This section will just cover some of the settings. I should first promise this by stating that you can become locked out of your own server if you fail to type your own password correctly. This is another good reason to add a trusted system using the apf -a command. You can also add a host to specifically block by adding the IP address to the /usr/local/bfd/ignore.hosts file.

The `ban` command that BFD uses is tied directly to APF. The command is `apf -d`, which is the same as we saw to manually ban addresses and subnets. The first configuration variable we will look at is `TRIG`; this is the number of failed attempts before becoming banned. The default is 15, and is pretty good. Keep in mind that this is per IP address connections, not account. So if 1 IP address fails 15 times using multiple accounts, it will be banned. Feel free to change this value if you want; I recommend not setting this above 5 to reduce the number of attempts that are allowed.

BFD has the ability to send emails out to alert of brute force attempts. This is a good idea as it will give you notice when attempts to access your system are occurring. To enable email alerts, set the value of `EMAIL_ALERTS` to `1`; then set the address you want emails to be sent to using `EMAIL_ADRESS`. You can define the subject for the email as well. This makes for easy flagging/filtering in email applications.

BFD runs from cron and places a cron entry in `/etc/cron.d` called `bfd`. This runs BFD every 3 minutes. This should be acceptable for almost anyone. You can get a list of offending IP addresses using `bfd` on the command line. This is useful for looking at specific IP subnets that you might want to start blocking, if you see a pattern starting. To get this list, use the following command:

```
bfd – a
```

To start BFD, use the following command:

```
bfd -s
```

Summary

While there are other ways to help ensure the security of your system, we have covered some of the most important in this chapter. Besides a good firewall, changing access to the SSH service and adding login attempt protection to your extensions is going to go a long way in keeping hackers out of your system. Do not underestimate the importance of security; these steps can mean the difference between being secured and having someone log in and start making thousands of phone calls around the country from your phone system.

16
HUD

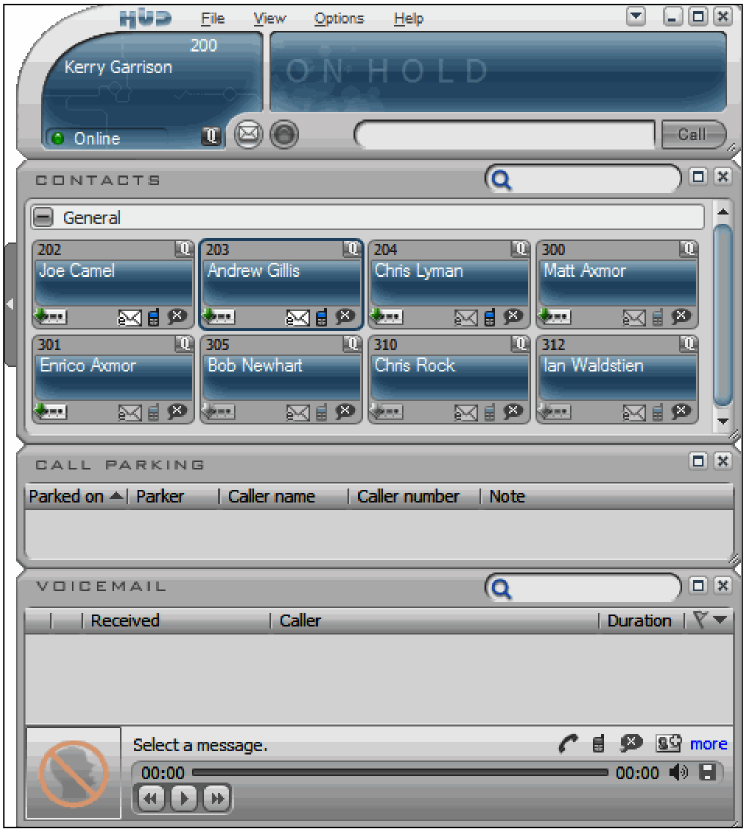
HUD (**Heads Up Display**) is Fonality's Unified Communication tool that is designed to improve the productivity and efficiency of your employees. By using HUD you can virtually look over the cubicle walls to see if other workers are available, if they are on a phone call, or are available for chat. For companies with more than just a few employees, HUD saves you from having to track people down and improves how people communicate.

At the time of this writing, **HUD3** is in a pre-alpha state, so there isn't a lot of information available about the final set of features, but by the time you read this, all of the information should be available on `http://trixbox.org`.

Desktop convergence / improved productivity

HUD integrates features of your phone with your desktop. This means that you can answer calls, place calls, hang up calls, see call status, transfer calls, put calls on hold, record calls, listen to voicemail, and perform other functions without actually having to touch your telephone device. Bringing many of the phone functions to the desktop means less time is spent navigating your phone and you can do many tasks much faster with just keyboard shortcuts and mouse clicks.

The other thing that makes HUD a powerful productivity tool is that you spend less time trying to figure out where people are and how to find them. At a glance you can see if someone is on a call, what kind of call it is, if he or she is available for chat or not, and simple icons to call his or her phone, call his or her mobile phone, chat, or send email. This ability is like being able to look over the cubicle wall to see what a co-worker is doing before trying to call him to talk to him or her.



HUD features

HUD is a very powerful application that can dramatically affect how you communicate with people in the company and how you use your phone system. By being able to keep your eyes on the screen and your hands on your keyboard, there are very few reasons why you ever need to actually touch your phone to manage your calls. HUD may seem like a simple tool, but is loaded with features:

- Operator Panel (w/BLF)
- Call Parking Area

- Drag-and-Drop Call Control
- Color-Coded Call Status
- Drag and Drop to Voicemail
- Extension Sorting
- Enterprise Instant Messaging
- Outlook Integration
- Presence Management
- Click-to-Call Mobile Phones
- Click-to-Email
- Desktop Alerts
- Interactive Desktop Alerts
- Group and User Permissions
- Extension Grouping
- Extension Search
- Extension Search—QuickMenu
- Shortcuts (Hotkeys)
- On-the-Fly Recording

Managing HUD users

Once you have created extensions, you can go to the HUD User Manager to set up additional options for each user. Eventually, this functionality will be integrated directly into the Extensions Manager, but on initial release, it will probably still be a separate module. With the users module, you set the HUD password, cell phone number, email address, and whether or not you want that user to be seen in HUD or not.

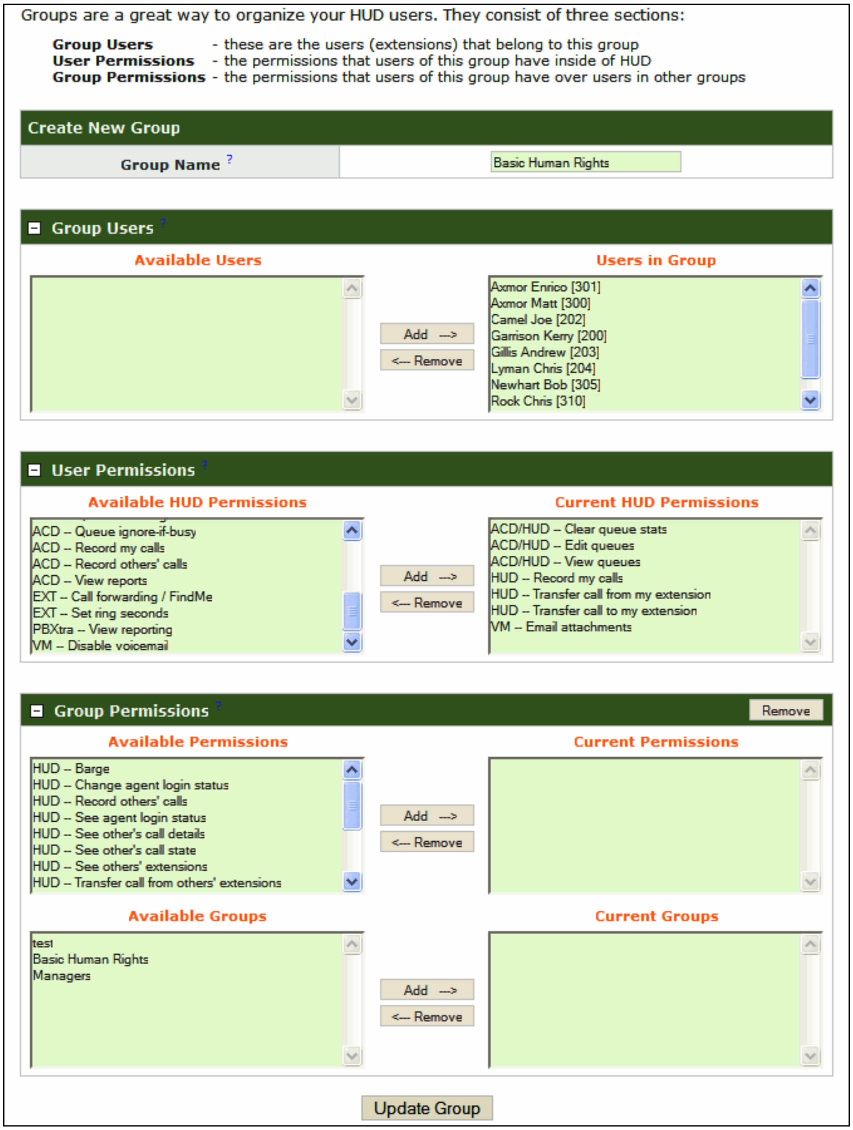
Extension	Ext Name	HUD Login	Password	First Name	Last Name	Cell Phone	Email	In HUD
SIP/200	Kerry Garrison	60_200	abc123	Kerry	Garrison	9492831385	kgarrison@gmail.com	✓
SIP/202	Joe Camel	60_202	abc123	Joe	Camel	6035551212	joe@camel.com	✓
SIP/203	Andrew Gillis	60_203	abc123	Andrew	Gillis	6037144364	agillis@fonality.com	✓
SIP/204	Chris Lyman	60_204	abc123	Chris	Lyman	3104691666		✓
SIP/300	Matt Axmor	60_300	89xpvyhs	Matt	Axmor			✓
SIP/301	Enrico Axmor	60_301	1w8gfbp2	Enrico	Axmor			✓
SIP/305	Bob Newhart	60_305	abc123	Bob	Newhart			✓
SIP/310	Chris Rock	60_310	abc123	Chris	Rock			✓
SIP/312	Ian Waldstien	60_312	abc123	Ian	Waldstien			✓

Submit

PBXconfig 2.4.0

Managing permissions

Like trixbox Pro, which we will look at in the next chapter, HUD for trixbox CE also has a permissions model built in. The HUD Groups module allows you to assign HUD-related permissions to users, such as the ability to record their calls, transfer calls, see other people's status, and much more. You begin with creating a group, assigning users to that group, and then setting the permissions that users in that group have. A second level allows that group to have permissions over other groups, such as barging, monitoring, and recording other people's calls.



Summary

When used to its fullest extent, HUD provides a great improvement in productivity to users and finally brings real unified communications to the trixbox CE platform. Since HUD is still in development at the time of this writing, there is no information about how to install it or what the final list of features will be. By the time this book is published, HUD should be available.

17
Introduction to trixbox Pro

While this book is focused on trixbox CE, which is the completely open-source project, Fonality also has another family of products under the trixbox Pro name. This chapter will delve into the differences between trixbox CE and Pro as well as describing the different versions of trixbox Pro itself.

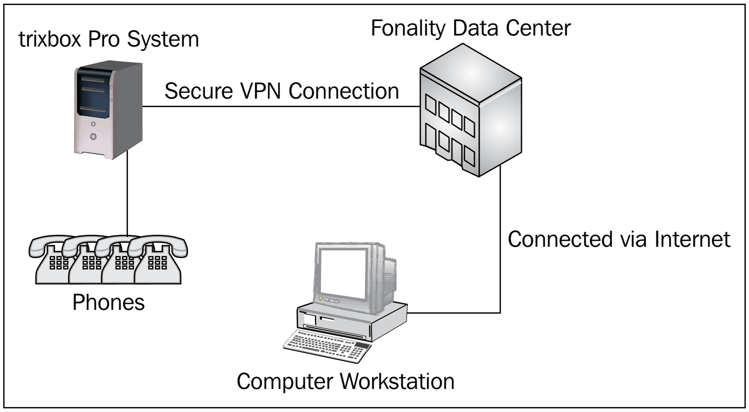
trixbox Pro versus trixbox CE

There are a number of significant differences between the CE and the Pro versions of trixbox, from both licensing and technology points of view. While trixbox CE is a completely customer premises-based solution, trixbox Pro uses Fonality's Hybrid-Hosted technology which places the PBX system at the customer premises but the web interface to manage the system is located in the Fonality data center.

The advantages of the Hybrid-Hosted model include the following:

- Inherent backup of configuration settings since settings are configured in the Fonality data center
- Centralized monitoring of remote systems
- Instant access to new features and bug fixes as only the web interface in the data center has to be upgraded
- New features and bug fixes that can automatically be pushed out to remote servers if needed
- Simplified access to do adds/moves/changes from anywhere with Internet access
- Call reporting and ACD reporting that does not use up any PBX system resources
- Fonality-run dynamic DNS system, which makes setting up remote phones very easy
- Ability to restore a system to a new system by re-syncing existing settings

From a technical point of view, trixbox CE uses a collection of open-source projects such as Asterisk, FreePBX, ARI, Flash Operator Panel, as well as code developed by the trixbox CE development team, all rolled into a single distribution. trixbox Pro also uses Asterisk as a base piece of code but Fonality has added over twice the original amount of code in order to harden the software for use in larger installations, as well as to add features that work the way businesses expect them to. Fonality's engineers have also written their own web-based graphical user interface for managing all of the PBX functions.



trixbox Pro Editions

There are three different versions of trixbox Pro for use in different types of situations.

trixbox Pro Standard Edition

The Standard Edition of trixbox Pro is a great starting point for most small businesses. Since there is no license fee for using the SE version, it is totally free to use just like the Community Edition version; this also gives companies a chance to try it out and see if they like how the trixbox Pro system works before committing to the other paid versions.

Enterprise Edition

The Enterprise Edition takes the Standard Edition even further by adding more features that are needed by larger businesses than the Standard Edition is designed for. Some of these features include items such as extension groups, routing by DIDs, alerts and notifications, and much more. We will look at a chart comparing the different versions later in this chapter.

Call Center Edition

The Call Center Edition of trixbox Pro is designed specifically for larger organizations and inbound Call Centers. If you need multiple queues and conference rooms, you will need to be using the Call Center Edition.

Feature Matrix

This table shows the differences between the different versions of trixbox Pro. The Standard Edition (SE) is available for free, while the Enterprise Edition (EE) and Call Center Editions (CCE) require you to purchase user licenses.

Features	Standard Edition	Enterprise Edition	Call Center Edition
Auto-Attendant (IVR)	Yes	Yes	Yes
Outlook Integration	Yes	Yes	Yes
Voicemail	Yes	Yes	Yes
Voicemail-to-Email	Yes	Yes	Yes
Music-on-Hold	Yes	Yes	Yes
Scheduler	Yes	Yes	Yes
Custom CTI (AGI)	Yes	Yes	Yes
Analog & IP Phones	Yes	Yes	Yes
Ring-All (Blast Group)	Yes	Yes	Yes
Call Forwarding	Yes	Yes	Yes
Name Directory	Yes	Yes	Yes
DIDs	Yes	Yes	Yes
Unlimited VoIP Accounts	Yes	Yes	Yes
PSTN Fallback	Yes	Yes	Yes
Telecommuters	Yes	Yes	Yes
Branch Office Support	Yes	Yes	Yes
Web-based Control Panel	Yes	Yes	Yes
Powerful Reporting	Yes	Yes	Yes
Auto Phone Provisioning	Yes	Yes	Yes
Re-Brandable Interface	co	Full	Full
Smart FindMe	-	Yes	Yes
Conference Bridge	-	Yes	Yes
Extension Groups	-	Yes	Yes
Routing by DIDs	-	Yes	Yes

Features	**Standard Edition**	**Enterprise Edition**	**Call Center Edition**
Paging / Zone Paging	-	Yes	Yes
Intercom / Zone Intercom	-	Yes	Yes
Voicemail Groups	-	Yes	Yes
Advanced Call Forwarding	-	Yes	Yes
Call Return	-	Yes	Yes
Call Out	-	Yes	Yes
Report Exporting (.csv)	-	Yes	Yes
Custom Caller IDs	-	Yes	Yes
IVR Authentication	-	Yes	Yes
SMS/Pager Voicemail Notify	-	Yes	Yes
Upload Voice Prompts	-	Yes	Yes
Alerts & Notifications	-	Yes	Yes
Trunks Status Pages	-	Yes	Yes
Real-Time System Graphs	-	Yes	Yes
Historical System Graphs	-	Yes	Yes
Unlimited Call Queues**	-	-	Yes
Full Featured A.C.D.	-	-	Yes
Graphical Queue Reports	-	-	Yes
Agent Call Recording	-	-	Yes
Agent Variable Log-off	-	-	Yes
Agents on Cell Phones	-	-	Yes
Agents Shared across Sites	-	-	Yes
Real-Time Queue Stats	-	-	Yes

Managing trixbox Pro

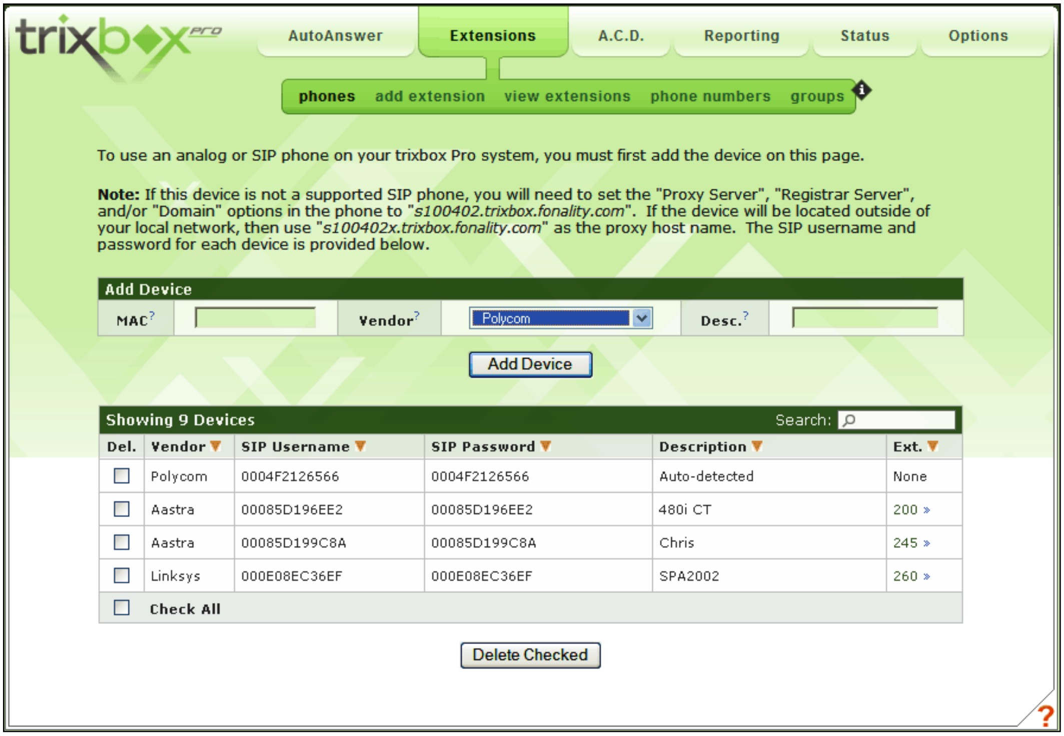
To manage your trixbox Pro installation, you need to log in to the trixbox Pro control panel system, which is located at `http://cp.trixbox.com`. From there, you need to use your login and password to access your configuration management tools.

Devices and users

Unlike trixbox CE, trixbox Pro separates users from devices, so the first thing you should do is adding the phones to the system and then creating users and assigning the phone device to each user. While we are going to go through manually provisioning a phone, trixbox Pro will automatically configure a new phone when plugged into the network.

Auto provisioning of Aastra phones is completely automatic because Aastra phones use a network service called MDNS, which is running on the trixbox Pro system. Polycom and Cisco phones require that your DHCP server is pushing out option 66 to point to the trixbox Pro system in order for these phones to automatically provision. During auto provisioning the phone device is created and a new extension is created that is automatically associated with the new phone.

To create a new phone, you enter in the MAC address of the phone, and select the type of phone. (You can also enter an optional description.) The MAC address will become the login and password of the device and the page tells you what the host name is if you manually configure the device.



When we look at the list of extensions, we can see which extensions are configured, what the current registration status of those extensions is, the user's name, description, the DID that is assigned to that extension, and columns that allow us to see if that extension is part of the blast group, if the extension is available in HUD (Fonality's Head Up Display), or if it is configured with call forwarding.

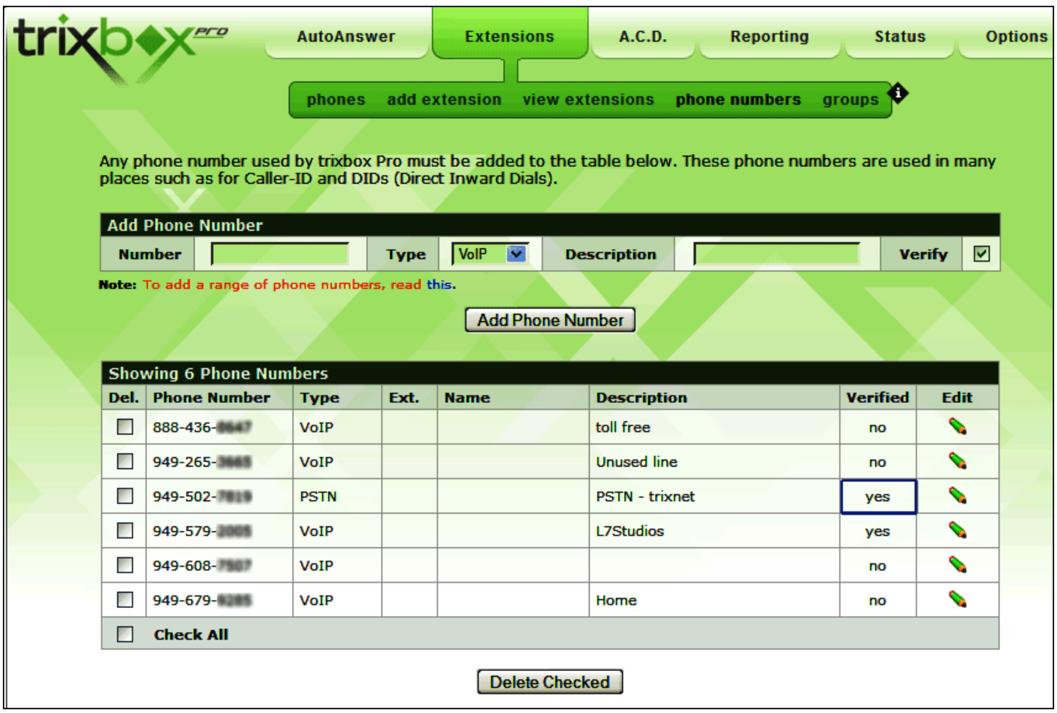
When we add or edit an extension, we are given the extension editor screen; from here, we can set a large number of options for each extension. While most of the options are fairly obvious, we do need to point out a few important fields you need to be aware of.

- **Web Username/Password**: These fields control the login and password that this user can use to log in to the trixbox Pro control panel and access his or her specific user settings and do things like set up his or her FindMe functions and listen to voicemail.
- **Inbound Phone No.**: If this user has a DID assigned to him or her, you use this field to select an available phone number to assign to this user.
- **Queue Settings**: Queue settings are only applicable if you are using the Call Center Edition.
- **In Blast Group**: If you want this extension to ring when a call is sent to the blast group, then this needs to be set to **yes**.
- **In HUD**: If this is not set to **yes**, this extension will not be visible in HUD.

- **Private Extension**: A private extension cannot receive calls directly from external callers even if they know the extension number. This is often used to prevent callers from being able to get directly to the CEO, or for use as a lobby phone.

Phone numbers

Phone numbers are entered into the system through the Phone Number management screen. This allows the pull-down menus in different parts of the system to be pre-populated with the available phone numbers. While adding a new phone number, you have the option of having the phone number verified through trixNet. If you choose to have your number verified, you will need to have that phone number routed to an extension so that you can receive the verification phone call. Once verified, other trixbox Pro systems that are using trixNet and attempt to call your phone number will place a direct phone system to phone system call and bypass the phone company resulting in free phone calls between trixbox Pro users.



Groups and permissions

With trixbox Pro Enterprise and Call Center Editions, there is a complete hierarchical permissions structure. This allows you very granular control over who can do and see what within the system. This is one of the most powerful features of trixbox Pro as it controls not only the permissions of the phone system itself but also permissions within HUD (Heads Up Display).

In the first section, you define the group members and the group itself. These groups are not only used for permissions but can also be used for group paging and group intercom. Once you define the group name and extension, the next section allows you to add members to the group.

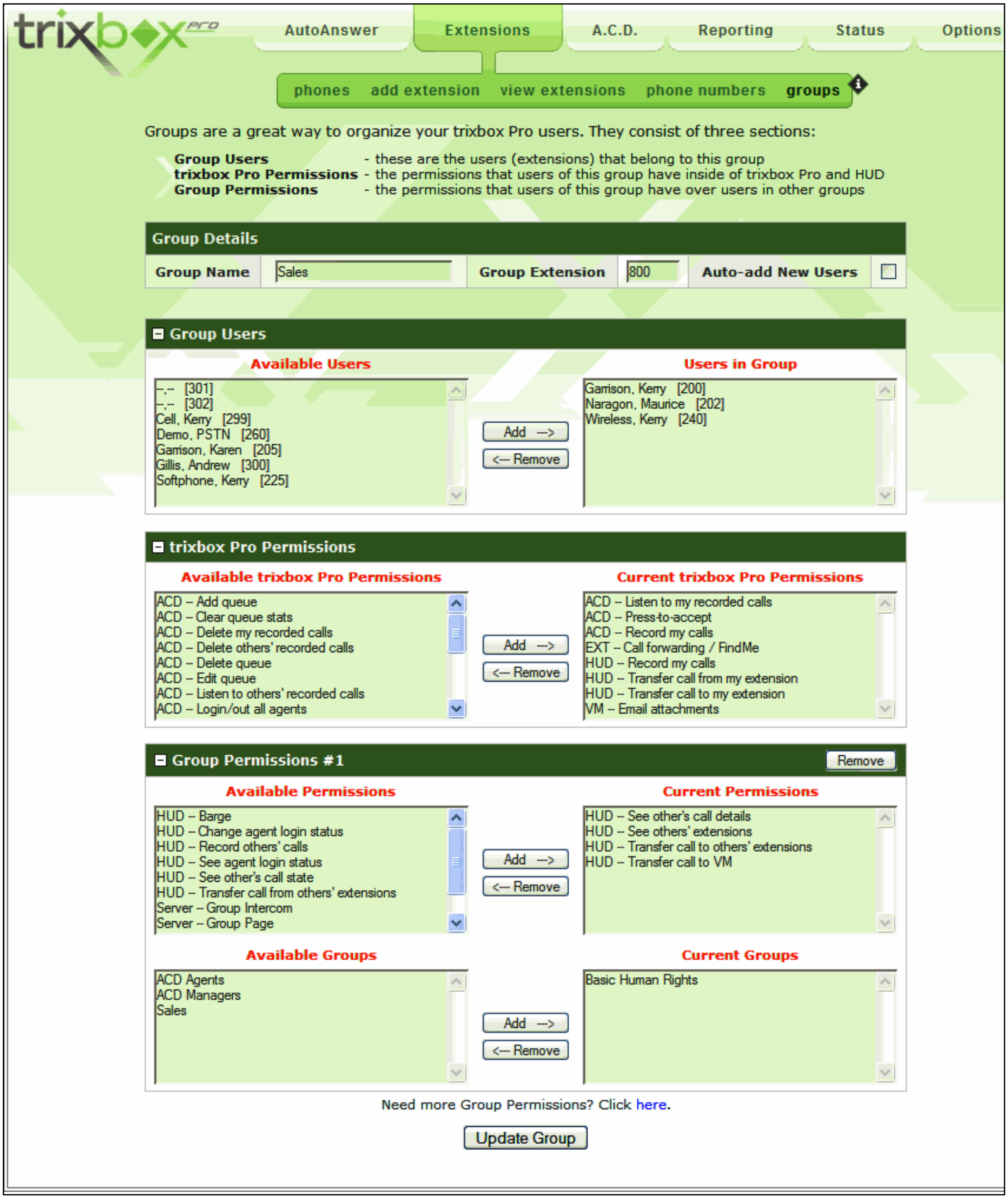
trixbox Pro permissions

The second section defines the permissions that the members of the group have within the system. This section includes things like the ability to record calls, forward calls, transfer calls, view reports, edit queues, record other people's calls, and so on.

Group permissions

The group permissions section defines the permissions that members of the group have over other groups. This allows you to give permissions to managers to barge onto the calls for people in their group. Most of the permissions in the group section are used to control permissions used in HUD.

The second part of the group permissions section determines the groups that the permissions affect. A Sales Manager group would have permissions over a Sales Agents group, for example.


trixbox pro
AutoAnswer
Extensions
A.C.D.
Reporting
Status
Options
phones
add extension
view extensions
phone numbers
groups
Groups are a great way to organize your trixbox Pro users. They consist of three sections:
Group Users - these are the users (extensions) that belong to this group
trixbox Pro Permissions - the permissions that users of this group have inside of trixbox Pro and HUD
Group Permissions - the permissions that users of this group have over users in other groups
Group Details
Group Name
Sales
Group Extension
800
Auto-add New Users
Group Users
Available Users
-,- [301]
-,- [302]
Cell, Kerry [299]
Demo, PSTN [260]
Garrison, Karen [205]
Gillis, Andrew [300]
Softphone, Kerry [225]
Add -->
<-- Remove
Users in Group
Garrison, Kerry [200]
Naragon, Maurice [202]
Wireless, Kerry [240]
trixbox Pro Permissions
Available trixbox Pro Permissions
ACD – Add queue
ACD – Clear queue stats
ACD – Delete my recorded calls
ACD – Delete others' recorded calls
ACD – Delete queue
ACD – Edit queue
ACD – Listen to others' recorded calls
ACD – Login/out all agents
Current trixbox Pro Permissions
ACD – Listen to my recorded calls
ACD – Press-to-accept
ACD – Record my calls
EXT – Call forwarding / FindMe
HUD – Record my calls
HUD – Transfer call from my extension
HUD – Transfer call to my extension
VM – Email attachments
Group Permissions #1
Remove
Available Permissions
HUD – Barge
HUD – Change agent login status
HUD – Record others' calls
HUD – See agent login status
HUD – See other's call state
HUD – Transfer call from others' extensions
Server – Group Intercom
Server – Group Page
Current Permissions
HUD – See other's call details
HUD – See others' extensions
HUD – Transfer call to others' extensions
HUD – Transfer call to VM
Available Groups
ACD Agents
ACD Managers
Sales
Current Groups
Basic Human Rights
Need more Group Permissions? Click here.
Update Group

AutoAnswer

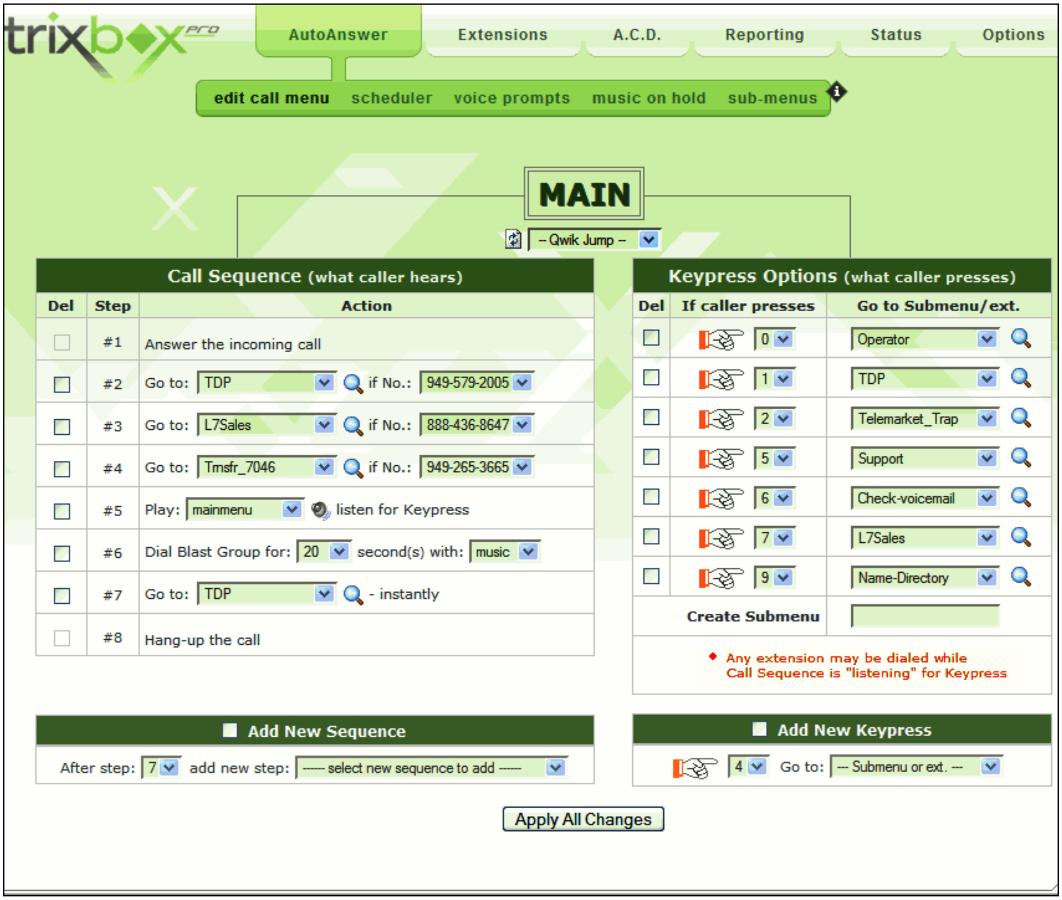
To set up your IVR (Digital Receptionist) you use the AutoAnswer menu; here you create the sequence of steps a call goes through and the key-presses that the system will listen for. This is quite different in design from how trixbox CE works, so it can take some rethinking of how you do things to get the hang of it. A big advantage of the way trixbox Pro handles call flow is that it is very easy to look at the system and figure out exactly what the call flow process is. With trixbox CE, you have to look in multiple locations to figure out that complete call flow.

At any point, you can add, delete, or modify a step in the process. With trixbox CE, if you wanted different DIDs to route to different menus, you would have to create the different menus, and then go to a separate module to set up the inbound routes. With trixbox Pro (EE/CCE), you do all of this from the same page by using a pull-down menu to select the submenu to go to and another pull-down menu to select the DID phone number to match on.

On the opposite side of the screen, you manage the available key-presses and select either a submenu or extension that the call will be sent to. This ease of use not only makes it easier to follow the call flow, but allows even very non-technical people to be able to figure out how to use the system.

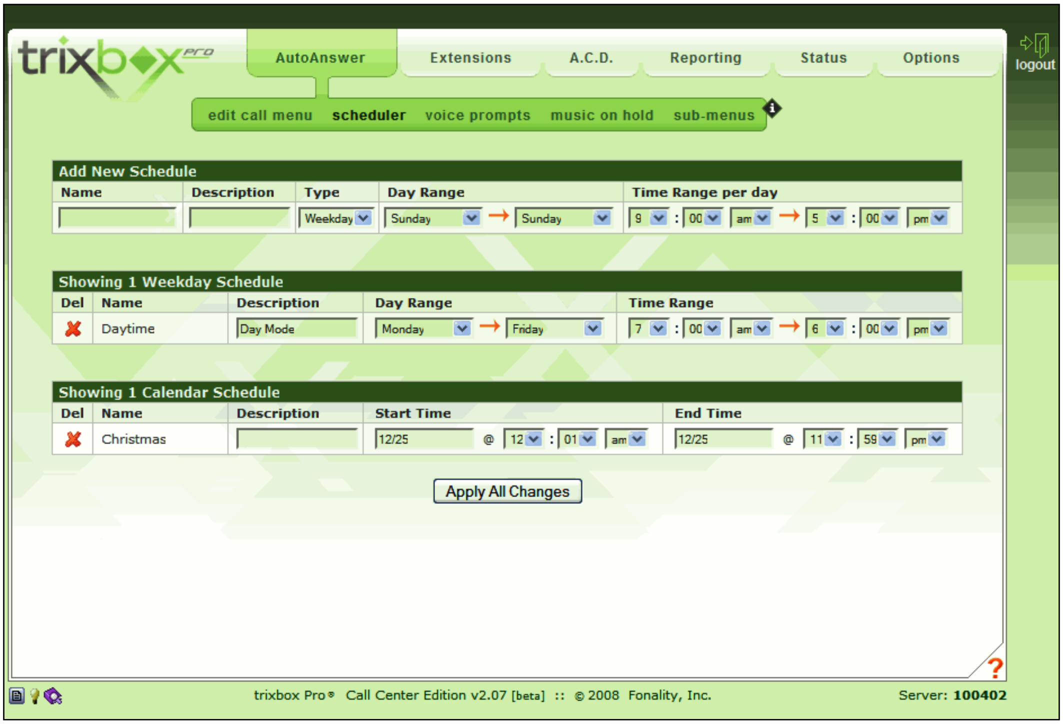
One of the myths about trixbox Pro is that you can't integrate AGI scripts into a call flow. To do this, you simply put the script into the `agi` folder, and then use the **Run Script** option in the call sequence area and give it the name of the `agi` script you want to use. This can be used for things like:

- Custom call menus
- Caller ID Name lookups
- Surveys
- Database lookups
- Trouble ticket systems
- And much more


trixbox pro
AutoAnswer
Extensions
A.C.D.
Reporting
Status
Options
edit call menu
scheduler
voice prompts
music on hold
sub-menus
MAIN
-- Qwik Jump --
Call Sequence (what caller hears)
Del
Step
Action
#1
Answer the incoming call
#2
Go to: TDP if No.: 949-579-2005
#3
Go to: L7Sales if No.: 888-436-8647
#4
Go to: Trnsfr_7046 if No.: 949-265-3665
#5
Play: mainmenu listen for Keypress
#6
Dial Blast Group for: 20 second(s) with: music
#7
Go to: TDP - instantly
#8
Hang-up the call
Keypress Options (what caller presses)
Del
If caller presses
Go to Submenu/ext.
0 Operator
1 TDP
2 Telemarket_Trap
5 Support
6 Check-voicemail
7 L7Sales
9 Name-Directory
Create Submenu
Any extension may be dialed while
Call Sequence is "listening" for Keypress
Add New Sequence
After step: 7 add new step: ----- select new sequence to add -----
Add New Keypress
4 Go to: --- Submenu or ext. ---
Apply All Changes

Scheduler

By using the scheduling tool, you can create different time conditions so that the system can automatically use different AutoAnswer menus at certain times of the day or even on specific days. This is useful for creating an after-hours menu or special messages used on holidays.

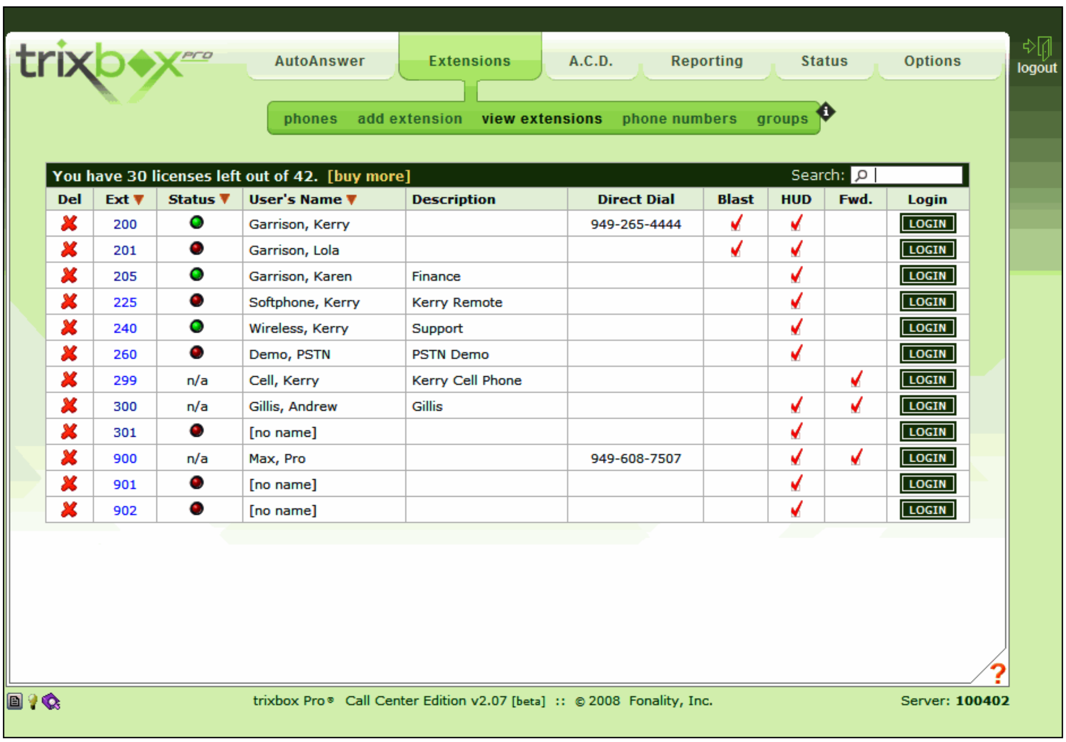


Also within the AutoAnswer menu, you will find a section to record the voice prompts that the system will use in the AutoAnswer menus, a section to manage music-on-hold files, and a section for creating submenus off of the main AutoAnswer menu.

Managing extensions

Unlike trixbox CE where the user and the device are typically combined, trixbox Pro separates the device from the user. Under the **Extensions** menu, there is a section for managing the phones and a separate section for the extensions. If your network is set up properly, both Aastra and Polycom phones will automatically be detected when they are plugged into your network, and the phone device will be automatically created as well a new extension that will be mapped to the new device.

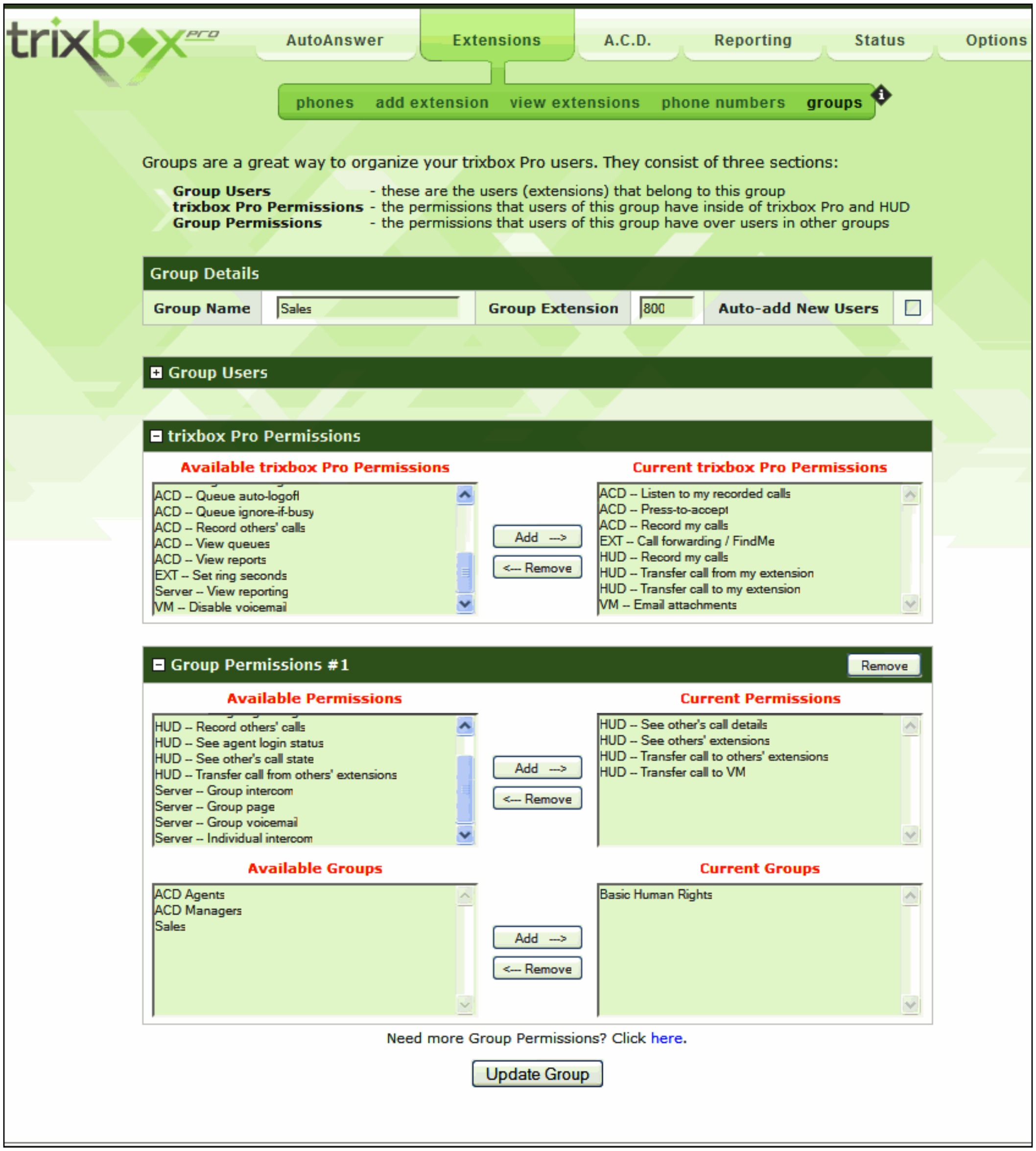
When you look at the list of the extensions, you can see the extension number, the current status of that device, who the extension is assigned to, and other settings such as DID numbers, Blast Group Membership, whether the extension will show up in HUD or not, and if the extension is being forwarded or not.



Del	Ext	Status	User's Name	Description	Direct Dial	Blast	HUD	Fwd.	Login
	200		Garrison, Kerry		949-265-4444	✓	✓		LOGIN
	201		Garrison, Lola			✓	✓		LOGIN
	205		Garrison, Karen	Finance			✓		LOGIN
	225		Softphone, Kerry	Kerry Remote			✓		LOGIN
	240		Wireless, Kerry	Support			✓		LOGIN
	260		Demo, PSTN	PSTN Demo			✓		LOGIN
	299	n/a	Cell, Kerry	Kerry Cell Phone				✓	LOGIN
	300	n/a	Gillis, Andrew	Gillis			✓	✓	LOGIN
	301		[no name]				✓		LOGIN
	900	n/a	Max, Pro		949-608-7507		✓	✓	LOGIN
	901		[no name]				✓		LOGIN
	902		[no name]				✓		LOGIN

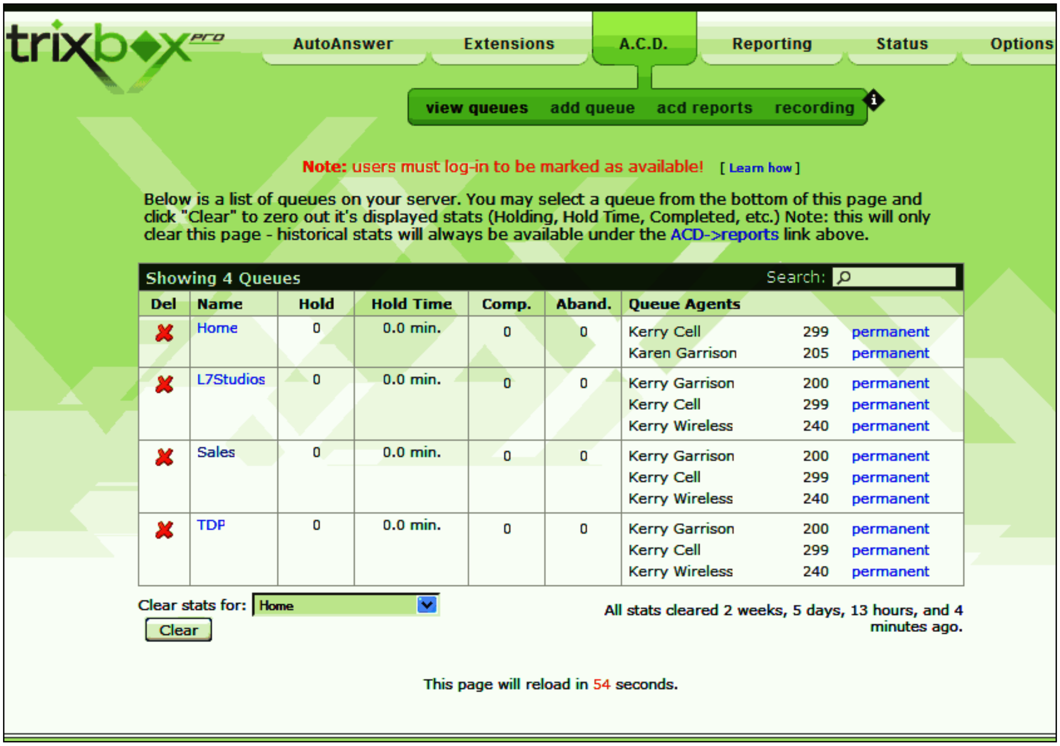
Groups and permissions

One of the biggest features of trixbox Pro (Enterprise and Call Center Editions) is the groups and permissions that can be used to control access to features of the system. The Groups system allows you to create groups that are used for group paging and group intercom, but you can also assign permissions to the groups such as the ability to manage queues, view reports, use FindMe, and the ability to use HUD to transfer calls. You can also assign the group permissions over other groups—this is used to control access to who can barge, record other people's calls, perform intercom functions, and transfer calls around. If you are looking for a phone system with real granular control over permissions, trixbox Pro is well ahead of anything else available.



A.C.D. (Call Center Edition)

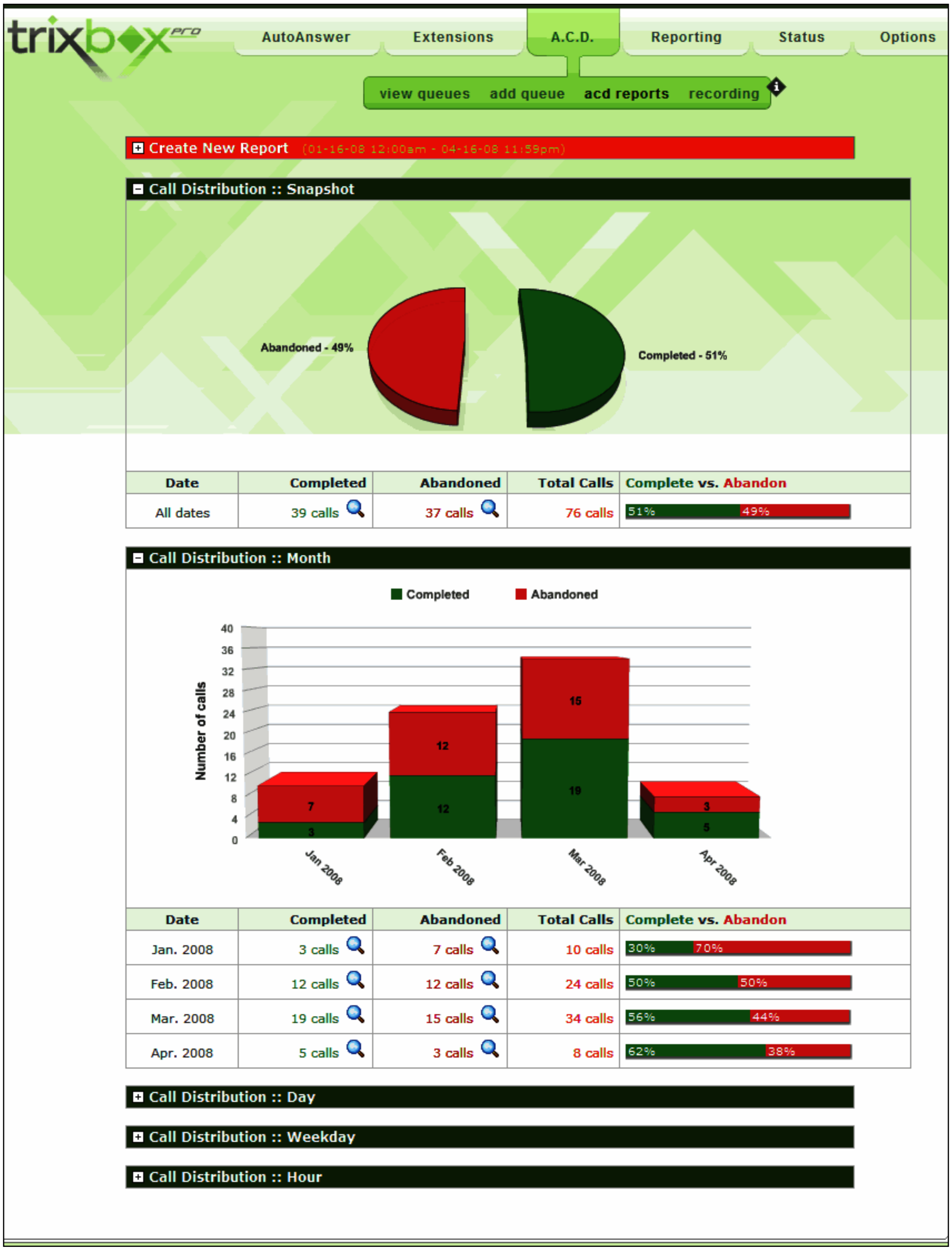
The **A.C.D.** (**Automatic Call Distribution**) system is used to manage call queues. Queues allow callers to be placed into a staging area while they are waiting for an agent to become available to take their calls. The main **view queues** page shows you a list of available queues along with useful information such as the number of calls on hold, average hold time, completion percentage, abandonment percentage, and the agents assigned to the queue.



A.C.D. reports

trixbox Pro also has a nice system for viewing reports for the call queues; with more than 15 different reports and a handful of different criteria, the **A.C.D.** reports give Call Center managers an insight into what's going on with inbound calls, which agents are being effective, and how well your queue is distributing calls.

The following is a pictorial representation of A.C.D. reports:

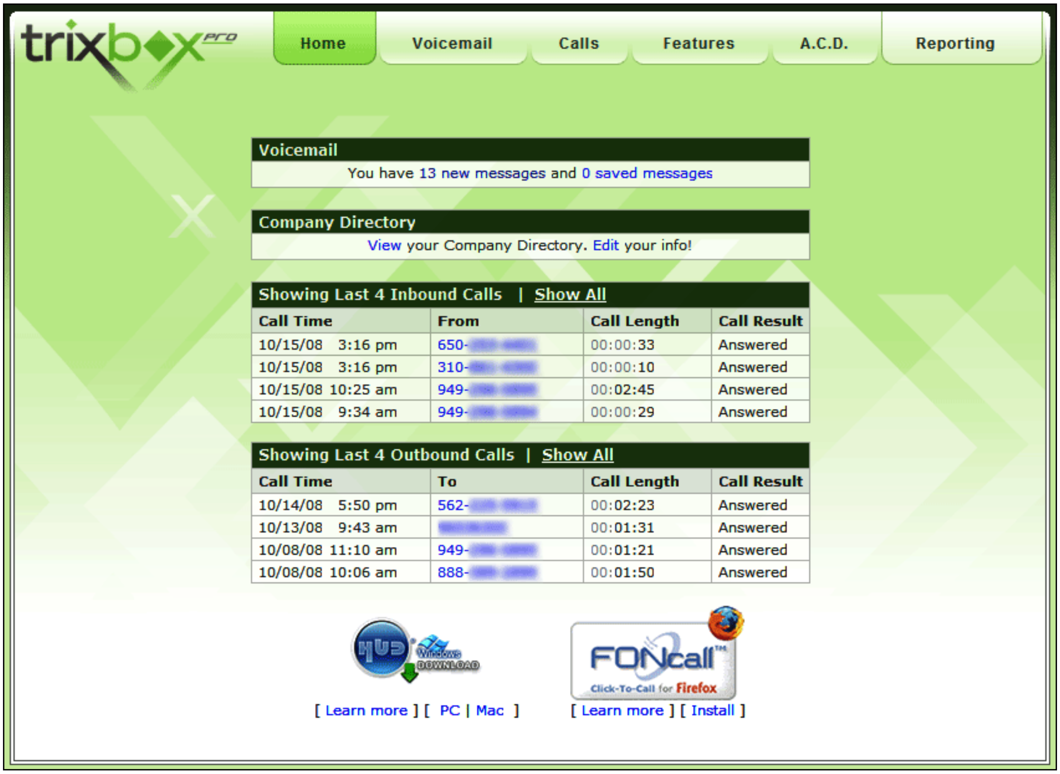


Additional admin features

trixbox Pro has a wealth of other administration features that allow you to set up your interface cards, VoIP accounts, set up alerts for common error conditions, and view real time or historic system performance information. Since this chapter is just an overview of what is available, we can't dig into each and every feature as that would be a book all by itself.

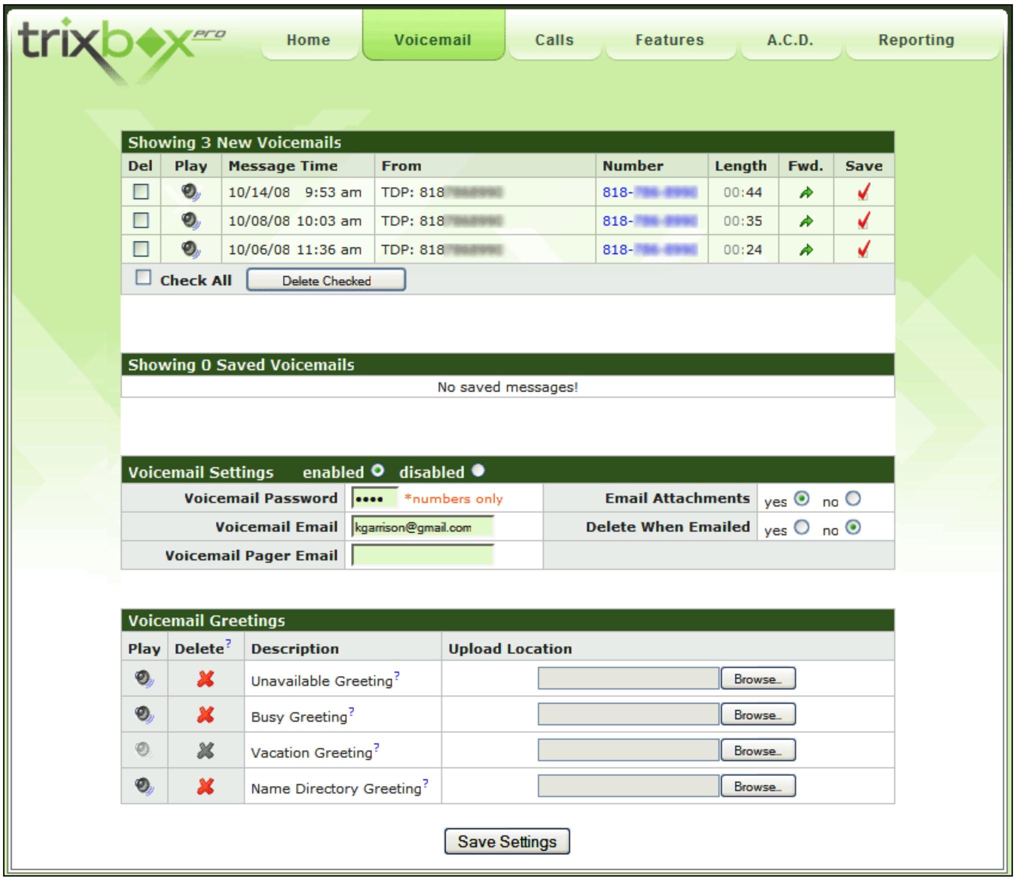
User features

Before we close on this topic, we should take a look at the User Portal that comes with trixbox Pro. Each user has access to his or her own web portal by using the web username and password that is assigned to him or her in the Extensions Manager. When you first log in, you will see how many voicemails you have, have access to the company directory, and see your most recent call history.



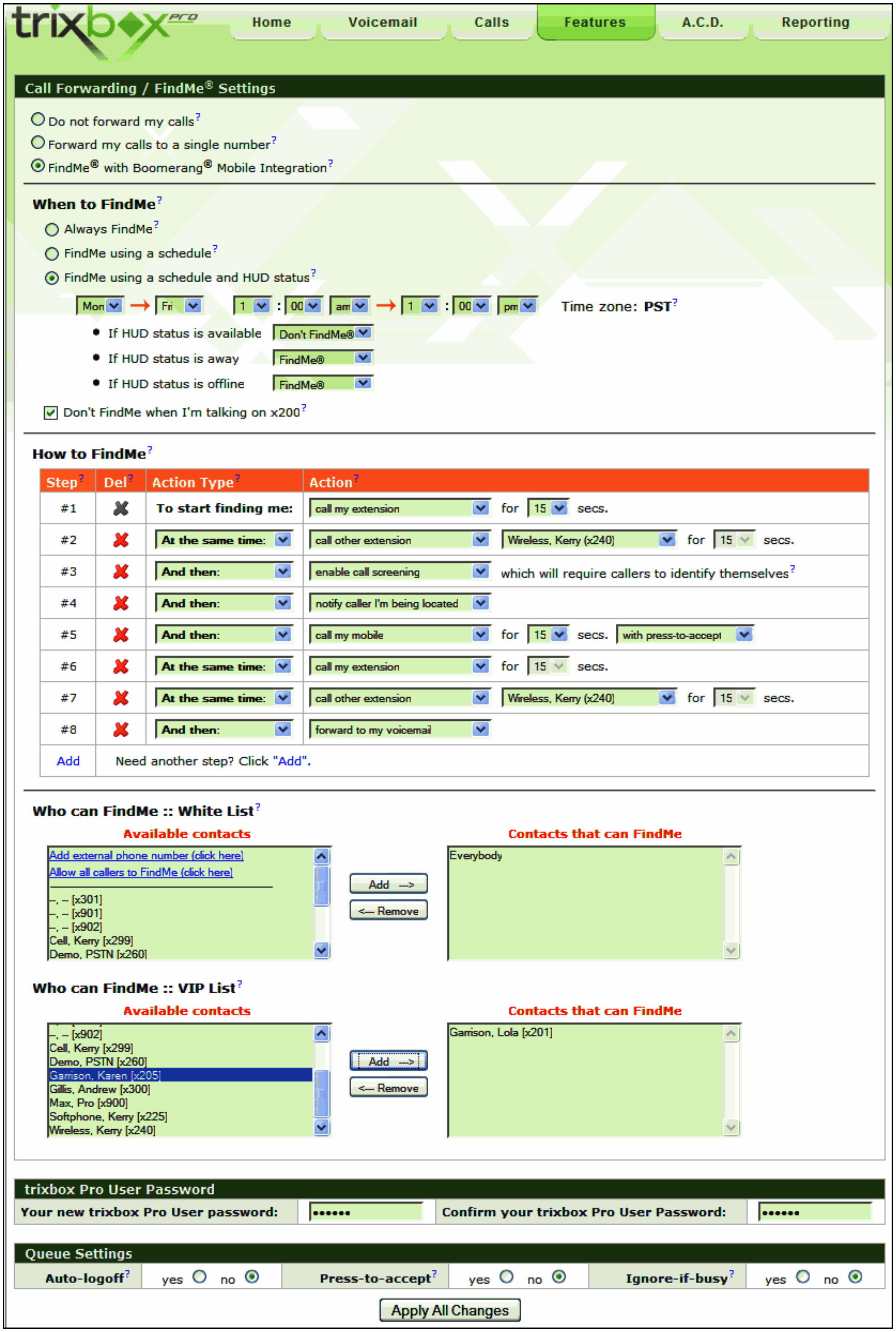
Voicemail

The voicemail system shows you your list of current voicemails and allows you to delete, play, and forward voicemail messages. You also have control over your own voicemail settings and voicemail greetings.



Find-Me

Even though many systems, including trixbox CE, have a Follow-Me type of system, trixbox Pro Enterprise Edition and Call Center Edition have a very robust system called Find-Me. Using the Find-Me tool, users can create very simple to quite complex call flows for how they wish to be contacted. The system can also include a schedule and even use your HUD status. The system allows you to create different steps that determine what extensions to call, and when to call your cell phone, enable call screening, or forward to your voicemail. With this level of control, every user can set up his or her own rules according to how they want to be found.


trixbox pro
Home
Voicemail
Calls
Features
A.C.D.
Reporting
Call Forwarding / FindMe® Settings
Do not forward my calls
Forward my calls to a single number
FindMe® with Boomerang® Mobile Integration
When to FindMe
Always FindMe
FindMe using a schedule
FindMe using a schedule and HUD status
Mon → Fri 1 : 00 am → 1 : 00 pm Time zone: PST
If HUD status is available Don't FindMe®
If HUD status is away FindMe®
If HUD status is offline FindMe®
Don't FindMe when I'm talking on x200
How to FindMe
Step Del Action Type Action
#1 To start finding me: call my extension for 15 secs.
#2 At the same time: call other extension Wireless, Kerry (x240) for 15 secs.
#3 And then: enable call screening which will require callers to identify themselves
#4 And then: notify caller I'm being located
#5 And then: call my mobile for 15 secs. with press-to-accept
#6 At the same time: call my extension for 15 secs.
#7 At the same time: call other extension Wireless, Kerry (x240) for 15 secs.
#8 And then: forward to my voicemail
Add Need another step? Click "Add".
Who can FindMe :: White List
Available contacts
Add external phone number (click here)
Allow all callers to FindMe (click here)
-, - [x301]
-, - [x901]
-, - [x902]
Cell, Kerry [x299]
Demo, PSTN [x260]
Add -->
<-- Remove
Contacts that can FindMe
Everybody
Who can FindMe :: VIP List
Available contacts
-, - [x902]
Cell, Kerry [x299]
Demo, PSTN [x260]
Garrison, Karen [x205]
Gillis, Andrew [x300]
Max, Pro [x900]
Softphone, Kerry [x225]
Wireless, Kerry [x240]
Add -->
<-- Remove
Contacts that can FindMe
Garrison, Lola [x201]
trixbox Pro User Password
Your new trixbox Pro User password: ••••••
Confirm your trixbox Pro User Password: ••••••
Queue Settings
Auto-logoff yes no
Press-to-accept yes no
Ignore-if-busy yes no
Apply All Changes

Summary

Where trixbox CE is completely open source, trixbox Pro is a closed-source frontend that uses a Hybrid-Hosted model to ensure that fixes and updates can be pushed out quickly and easily and, by its very nature, is an inherent backup and recovery system. With a free Standard Edition, trixbox Pro can be tried out to see if it is a better fit than trixbox CE. What you may sacrifice in flexibility, you may make up in ease of use and manageability.

Glossary of VoIP terms

This appendix will cover the common terms and acronyms used throughout this book, which are also common to Asterisk and telephony in general.

ACD

Automatic Call Distribution is a feature used to route calls in a Call Center environment to the appropriate person, based on factors such as availability, call usage, skills, time, and so on.

Agent

An agent is a person on an extension that is a member of a queue.

AGI

The Asterisk Gateway Interface is the interface used by applications to talk to the Asterisk engine.

ARI

The Asterisk Recording Interface is the module that is available from within the trixbox CE User dashboard that provides web access to voicemail and individual user settings.

ATA

An Analog Telephone Adapter allows a traditional analog phone to be used on an Asterisk phone system by converting the analog signals to VoIP and connecting to the Asterisk system as a SIP extension.

BRI

Basic Rate Interface is a communication standard that is used on ISDN connections.

CDR

Call Data Records are the call logs that are kept by the system showing the logging of all of the calls into and out of the system.

Codec

A Codec is a piece of code that encodes or decodes audio using a given type of algorithm.

Conference Bridge

A conference bridge is a feature of the phone system that creates a meeting room that multiple callers can be in to have a multiple-person conference call.

CRM

A Customer Relationship Manager tool is used to track contacts, sales leads, and notes. A popular open-source CRM package is SugarCRM, which is currently no longer included with trixbox CE.

DID

This stands for Direct Inward Dialing, which refers to a phone number that can be dialled to place a call into your PBX system.

DISA

Direct Inward System Access is a method by which you can connect into the phone system from the outside and have access to functions as if you were connected via an extension.

Firewall

A device that exists at the border of your network and applies polices to the traffic that traverses the network.

Flash Operator Panel

This is a component that allows a user to see the different extensions, trunks, queues, and parked calls within a system and drag-and-drop calls around.

Follow-Me

This feature allows users to configure their extension to send calls to multiple destinations, such as their extension and their cell phone.

FOP

See *Flash Operator Panel*.

FreePBX

FreePBX is the component within the trixbox CE Admin dashboard that allows you to configure all of the PBX functionality.

FXO

The Foreign eXchange Office is the port that connects to an analog phone line.

FXS

The Foreign eXchange Station is the port that connects to an analog phone device.

G.711 Ulaw

G.711 is an uncompressed audio format, thereby providing the best call quality but the highest bandwidth usage.

G.729

The G.729 codec provides excellent call quality and very low bandwidth but requires a license for each channel that will be used.

GSM

GSM is a codec that provides good call quality and very low bandwidth usage.

Hard Phone

This is a hardware phone device.

IAX

The Inter-Asterisk-eXchange protocol was developed as a simpler and easier-to-manage alternative to the SIP protocol. IAX can traverse NAT far better than SIP can. The IAX protocol is not an industry standard but is native to Asterisk.

iLBC

The iLBC codec provides very good call quality with decent bandwidth usage.

Intercom

An intercom system allows you to call another extension and have its extension automatically answered on speakerphone for a two-way conversation.

ISDN

This stands for Integrated Services Digital Network. While use of ISDN circuits has become increasingly rare in the US, it is still very popular in other countries and provides some PRI type features to small to medium businesses.

ITSP

An Internet Telephone Service Provider is a company that provides inbound or outbound call termination to your PBX over the Internet rather than requiring you to go through a traditional telephone company.

IVR

The Interactive Voice Response system is the menus that are given to your callers that help route the caller to the appropriate destination.

NAT

Network Address Translation allows for a network to have its own private IP space and still communicate to the Internet through a single public IP address.

Paging

A paging system can use overhead speakers or can page through the speakerphones of the phone devices, and allows the person who initiated the page to broadcast a one-way message.

PBX

The Private Branch Exchange is the phone system in your office that is the interface between the extensions in your office and the different connections to telephone service providers.

POTS

This acronym stands for Plain Old Telephone Service, which is the commonly used analog phone lines found in houses and small-medium businesses. POTS is an analog system and is controlled by electrical loops. It is provided by copper wires run to homes and businesses and is often the cheapest and easiest telephone service to implement.

Predictive Dialer

A predictive dialer is a software application for outbound calling that monitors the agents who are on calls, anticipates when the next agent will be available, and calls the next number in the list trying to maximize the use of the outbound calling agents.

PRI

Primary Rate Interface is a digital communication standard that is brought in over T1 or E1 circuits.

PSTN

The Public Switched Telephone Network is the traditional public phone network that carries all traditional phone calls.

Queues

A call queue is typically a very-high end PBX feature that allows callers to go into a holding area in the order that they called in and wait for an available agent. A queue is typically used when there are peak times and more inbound calls coming in than there are agents available to take the call.

Ring Groups

A ring group is a group of extensions that can be set to ring at the same time. A ring group is usually used when there are typically more agents available to take calls than there are calls coming in.

SIP

Session Initiation Protocol is the industry standard for voice over IP communications.

Softphone

Unlike a hard phone, a softphone is a software program that you run on your computer that emulates all of the functions of a normal phone and allows you to receive and place phone calls from your computer.

Trunk

A trunk is a group of channels between two distinct points. This can be either between PBXs within a corporation or between an organization and a telephone service provider.

T1/E1

This is the common digital circuit for medium to large businesses. T1 and E1 circuits are a digital service and offer more features than ISDN or POTS lines. A T1 circuit can handle 24 voice calls while an E1 circuit can handle 32 voice calls.

VoIP

This is the acronym for any voice traffic that uses Internet protocols. In a typical Asterisk setup, the phones will communicate over VoIP to the phone system and the phone system may communicate over VoIP to an Internet Telephone Service Provider.

Index

A

B

C

D

E

F

G

H

I

Q

R

S

U

V

W

Z

Thank you for buying

trixbox CE 2.6

Packt Open Source Project Royalties

When we sell a book written on an Open Source project, we pay a royalty directly to that project. Therefore by purchasing trixbox CE 2.6, Packt will have given some of the money received to the trixbox project.

In the long term, we see ourselves and you—customers and readers of our books—as part of the Open Source ecosystem, providing sustainable revenue for the projects we publish on. Our aim at Packt is to establish publishing royalties as an essential part of the service and support a business model that sustains Open Source.

If you're working with an Open Source project that you would like us to publish on, and subsequently pay royalties to, please get in touch with us.

Writing for Packt

We welcome all inquiries from people who are interested in authoring. Book proposals should be sent to `authors@packtpub.com`. If your book idea is still at an early stage and you would like to discuss it first before writing a formal book proposal, contact us; one of our commissioning editors will get in touch with you.

We're not just looking for published authors; if you have strong technical skills but no writing experience, our experienced editors can help you develop a writing career, or simply get some additional reward for your expertise.

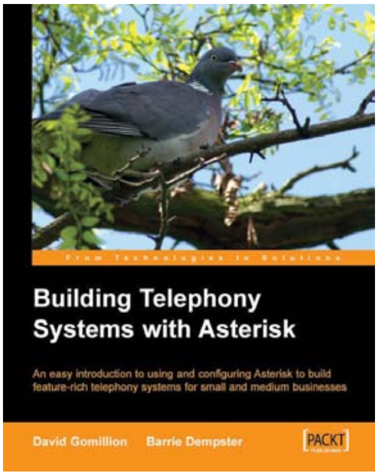
About Packt Publishing

Packt, pronounced 'packed', published its first book "Mastering phpMyAdmin for Effective MySQL Management" in April 2004 and subsequently continued to specialize in publishing highly focused books on specific technologies and solutions.

Our books and publications share the experiences of your fellow IT professionals in adapting and customizing today's systems, applications, and frameworks. Our solution-based books give you the knowledge and power to customize the software and technologies you're using to get the job done. Packt books are more specific and less general than the IT books you have seen in the past. Our unique business model allows us to bring you more focused information, giving you more of what you need to know, and less of what you don't.

Packt is a modern, yet unique publishing company, which focuses on producing quality, cutting-edge books for communities of developers, administrators, and newbies alike. For more information, please visit our website: `www.PacktPub.com`.

PACKT PUBLISHING



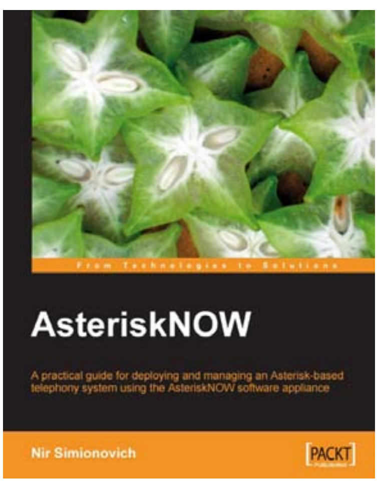
Building Telephony Systems With Asterisk

ISBN: 1-904811-15-9 Paperback: 180 pages

An easy introduction to using and configuring Asterisk to build feature-rich telephony systems for small and medium businesses.

1. Install, configure, deploy, secure, and maintain Asterisk
2. Build a fully-featured telephony system and create a dial plan that suits your needs
3. Learn from example configurations for different requirements

---



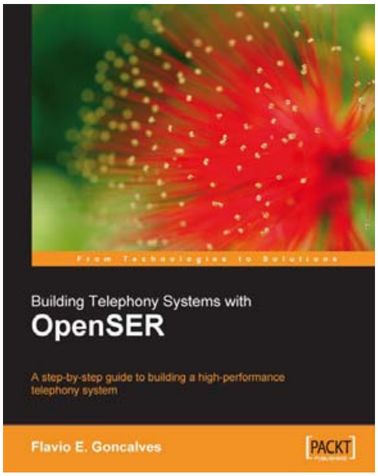
AsteriskNOW

ISBN: 978-1-847192-88-2 Paperback: 204 pages

A practical guide for deploying and managing an Asterisk-based telephony system using the AsteriskNOW Beta 6 software appliance

1. Install an Asterisk-based telephony system fast
2. Build an office PBX using AsteriskNOW
3. Learn the AsteriskGUI web management interface
4. Configure IP phones and connections
5. Configure and use the conferencing system
6. Write your own applications for Asterisk

Please check **www.PacktPub.com** for information on our titles

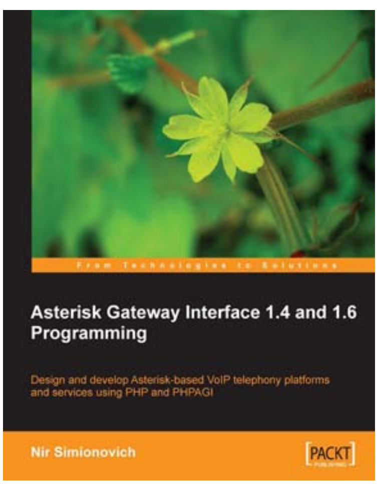


Building Telephony Systems with OpenSER

ISBN: 978-1-847193-73-5 Paperback: 303 pages

A step-by-step guide to building a high performance Telephony System

1. Install, configure, and troubleshoot OpenSER
2. Use OpenSER to build next generation VOIP networks from scratch
3. Learn and understand SIP Protocol and its functionality
4. Integrate MySQL with OpenSER



Asterisk Gateway Interface 1.4 and 1.6 Programming

ISBN: 978-1-847194-46-6 Paperback: 200 pages

Design and develop Asterisk-based VoIP telephony platforms and services using PHP and PHPAGI

1. Develop voice-enabled applications utilizing the collective power of Asterisk, PHP, and the PHPAGI class library
2. Learn basic elements of a FastAGI server utilizing PHP and PHPAGI
3. Develop new Voice 2.0 mesh-ups using the Asterisk Manager
4. Add Asterisk application development skills to your development arsenal, enriching your market offering and experience

Please check **www.PacktPub.com** for information on our titles