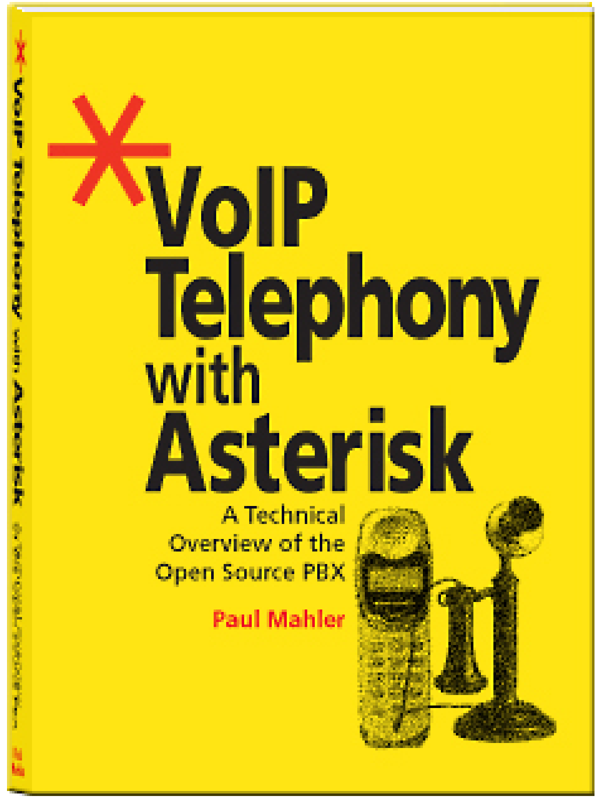
VoIP Telephony with Asterisk



BY Paul Mahle

ISBN 09759992-0-6

Mahler, P.S.

Asterisk and IP Telephony / Paul Mahle

Printed in the United States of America

20 19 18 17 16 15 14 13 12 11

Preface

This book is a beginner's guide to Asterisk and VoIP. This book is a road map to your first successful installation of an Asterisk telephone system. The path you need to take is documented step-by-step The information you need is all here in a single place. This is not a beginner's guide to Linux in that assume you already are a skilled Linux and network administrator. However, you do not need grea expertise in telephony or IP telephony to benefit from this book

Asterisk software turns an inexpensive PC architecture server running Linux or Unix into a reliable, sophisticated, full-featured enterprise telephone system. Because Asterisk is free and runs on an industry standard PC platform, an Asterisk system will cost you far less than any traditional, proprietar PBX. With Asterisk, you can quickly and easily build a sophisticated business telephone system for any enterprise, no matter how large or small. Because it is reliable, free and effective, and because it i based on modernInternet protocols, Asterisk will replace many legacy telephone systems in the marketplace.

Asterisk is far less expensive and much more effective that any competing telephone system. Asterisk provides all the functionality of a traditionalPBX, but it also provides new features and capabilities a legacyPBX can't offer. Because Asterisk is open you can change it and tune it as needed, unlike legacy systems which only provide closed black boxes with closed interfaces. With Asterisk you will neve again get locked into proprietary obsolete equipment from an unappealing single-source vendor.

This book documents the first release of Asterisk. Asterisk is quickly evolving which makes it exceedingly difficult to completely and effectively document. Thus, this book is not a complete guide to all the functionality Asterisk provides. Not every Asterisk feature is covered, not every covered feature i covered completely. None-the-less, this book should help you more quickly come up to speed wit Asterisk. I have tried to write the book I wanted to have while I was learning Asterisk

I have worked extremely hard to assure the accuracy of this text, and others have greatly contributed in their review of this book, but errors are unavoidable. If you find an error, please let me know with mai to*bookbugs@signate.com* or by going to our Web page at *http://asterisk.signate.com* so that we can fix it for the next edition. While this book is the result of the contribution of many people, the errors o omissions are my responsibility alone.

Paul Mahler

asterisk@signate.com

http://www.signate.com

Acknowledgements

There wouldn't be a book without the enormous help and support of Mark Spencer and Digium. James Lyons, Matthew Nicolson, Mat Fredrickson, John Bigelow and Mike Wood at Digium Technical support deserve special thanks for the many hours of patient help. They should get a medal. Gre Vance was always there to help.

Thanks to David Edison and Daryl Jones for making it all possible. Thanks to Warren Woodford for creating an Asterisk ready distribution of Mepis. John Todd contributed very valuable technical material.

The reviewers, Matt Florell, Mike Diehl, and Tom Scott, did an especially good job of finding, and fixing, many of my mistakes and adding new material. This book is much, much better because of thei hard work. I am especially grateful for their help.

Thank you, so much, everyone!

John Bigelow, Bill Boehlke

Malcom Davenport, Mike DiehlÂ

David Edison, Matt Florell Â

Mat Fredrickson, Chris Hariga Â

Dr. Lewis Heniford, Amal Johnson Â

Daryl Jones, James Lyons Â

Matthew Nicholson, Mike Pechner Â

Marcelo Rodriguez, Tom Scott, Â

David Schlossman, Mark Spencer Â

John Todd, Greg Vance, Â

Mike Wood, Warren Woodford

Forward

Telephony uses an old and inefficient model. Academics and researchers have shared their work for centuries. Scientists publish new discoveries in journals. Imagine where mankind would be if peopl had been unable to build on the knowledge of others. Yet this is the mentality on which proprietar telephone systems have depended

Traditional office telephones systems combine proprietary hardware and software. The resulting products have been either low cost and low function, or functional but expensive to purchase, maintain, and change. The developer of proprietary products has no interest in giving customers the ability t enhance or maintain them. Why should he? The proprietary model gives the traditional telephon supplier the ability to charge customers to use the products, charge to fix them, and charge again whe they need enhancement.

The proprietary model gets even better for the telephone supplier and worse for the customers as customers become tied to the vendor's specific methods and capabilities. The cost of switching away from the supplier becomes very large, creating formidable barriers to change.

That's why the open source model of software development is exploding. In the same way shared knowledge propels the whole of society forward, open technology development is showing that it ca drive innovation for an entire industry. Open source returns control to the user. Users can see the cod that makes the product work, change it, and learn from it. Shared problems are more easily found an fixed, without dependence on a single vendor's priorities. If customers don't like how one vendor i serving them, they can choose another without major switching costs.

Now, open source development has come to telephony, in the form of Asterisk, the open source telephony platform. A full-featured private branch exchange with capabilities for call distribution and interactive voice response, Asterisk runs on industry-standard hardware and shares your existing dat network rather than requiring separate lines and interconnection hardware. This combination ca reduce business customers' initial investment in telephony by as much as 90%, and provides the opportunity for equally dramatic reductions in calling costs.

Even better, Asterisk lets customers integrate their telephone system with other applications as easily as they integrate their CRM application with their accounting software. Asterisk can be extende using its APIs, dynamic module loader, and AGI scripting interface, and customers can add their ow applications that run on the system in C or any scripting language of their choice. Asterisk means tha powerful capabilities like call recording and call retrieval will be affordable by the majority of businesses for the first time.

Paul Mahler's book on Asterisk will help you learn how to install, configure and maintain Asterisk so you can begin realizing the benefits of open source telephony. I welcome you to the Asterisk community

William Boehlke

Presiden

Signate, LL

ChapterÂ 1 - Introduction

Asterisk is a PBX and a lot more. Asterisk is revolutionary, reliable, open source, free software that turns an ordinary inexpensive PC running Linux into a powerful enterprise telephone system. Asteris is an open source toolkit for telephony applications and a full-featured call-processing server. Asteris is an open architecture Computerized Telephony Integration platform. Many Asterisk systems are successfully installed around the world. Asterisk technology is working today for many businesses. Asterisk can be used for many things and has features includin

- Private Branch Exchange (PBX)
- Voicemail Services with Directory
- Conferencing Server
- Packet Voice Server
- Encryption of Telephone or Fax Calls
- Heterogeneous Voice over IP gateway (H.323, SIP, MGCP, IAX)

Custom Interactive Voice Response (IVR) system

Soft switch

Number Translation

Calling Card Server

Predictive Dialer

Call Queueing with Remote Agents

Gateway and Aggregation for Legacy PBX systems

Remote Office or User Telephone Services

PBX long distance Gateway

Telemarketing Block

Standalone Voicemail System

Many of the world's largest telephone companies have committed to replacing their existing circuit switched systems with packet switched voice over IP systems. Many phone companies are alread transporting a significant portion of their traffic with IP. Many calls made over telephone compan equipment are already being transported with IP.

Packet switched voice over IP systems are in principle as efficient as a synchronous circuit switched systems, but only recently have they had the potential to achieve the same level of reliability as the public switched telephone network or proprietaryPBX equipment. With the invention and implementation of RTP (real time protocol) and SIP (session initiation protocol,) voice over IP has the technological base to obsolete the circuit switched public switched telephone network.

Scenario - A Small Office

Asterisk can benefit a small office. In this scenario, a small office has four lines from the telephone company, each with its own telephone number. The office ha ten users. There is a fax machine and a conference room. The ten users eac have an IP telephone. There is an IP telephone in the conference room. Th small business can easily afford the inexpensive Asterisk server.

The Asterisk server manages calls for the four lines and all the phones and fax machines in the office. Any incoming call on the fourth line is directed to th fax machine. An incoming caller dialing the first line hears a voice menu There are choices for accessing a company directory, calling the operator, contacting sales, or dialing an extension directly.

The caller wants to speak to someone in sales. They consult the directory for the sales extension. They press 100 on their telephone keypad, the extension for sales Three phones are in the sales department. All three phones ring. There is distinctive ring that lets the sales staff know this is an incoming call from potential customer.

If no phone is answered by the fourth ring, the caller is given the choice of leaving a message or contacting the operator. If the user leaves a message, it is stored i a separate voicemail box for the sales department. Each of the three users i sales is sent an e-mail message letting them know that there is a new sales call.

What is a PBX?

Asterisk is a software implementation of a PABX. A PABX, usually called a PBX, is a Private Automatic Branch Exchange. A PBX is private because the enterprise owns it, not the telephone company. The telephone company can still be a supplier or service provider. Originally,PBX equipment was analog, more recent PBX equipment is digital. A PBX is cost attractive because it is less expensive to use a PBX than a separate phone line for every user in the enterprise and because it provides more services.

With a PBX, lines from the telephone company can be shared instead of having a separate line to the telephone company for each user. APBX provides a place for trunk (multiple phone) lines to terminate at the enterprise. APBX is a telephone system that services an enterprise by switching calls between enterprise users on local lines and by sharing the external phone lines. ThePBX has the intelligence to switch calls within the enterprise and outside the enterprise.

A PBX provides features and capabilities not available with direct connections to the Public Switched Telephone Network (PSTN.) A PBX moves telephone functions from the phone company to the enterprise. APBX provides additional functions and features like interactive voice response, call waiting, conferencing or voice mail, paging, transferring calls, or three way calling that wouldn't be available with separate telephone lines. A PBX usually has a console for use by an operator.

Alternatives to a PBX include Centrex. Centrex provides a pool of lines from the central office to the enterprise.Centrex can provide some of the same functions as a PBX, for example voice mail, call hold, call waiting or call transfer.

Like the PSTN, legacy enterprise telephony (ET) systems are circuit switched. They both use a common infrastructure model. All the control protocols and features are combined into a single model. ET systems usually cannot handle the same volume of traffic asPSTN switches. ET systems usually use proprietary protocols where thePSTN relies on the standard SS7 protocol.

Larger PBX systems typically have more features and abilities than smaller PBX systems. This is the way legacyPBX vendors market their systems. A feature you want may not be available on a PBX you can afford. You can only get the features you need if you are willing to spend more money.

How Does Asterisk Compare to a PBX?

ET systems, and Asterisk, provide interoperability between a local system and the PSTN. Many features in a legacy PBX system are rarely used. Some features may have been developed for a single user to make a single large sale. Because of this, Asterisk does not yet have all the features of allPBX systems from all vendors. Because Asterisk is an open platform features are easy to add and many new features are being added all the time. If Asterisk does not yet have a feature you want it is either already under development or easy to add. Any feature added to Asterisk by any user will be available for yo to use. This is because Asterisk is an open source product distributed under a GPL license.

What is Asterisk?

Asterisk is open source. It implements communications in software instead of hardware. This allows new features to be rapidly added with minimal effort. You can easily make your own changes or additions. With its included support for internationalization, rich set of configuration files, and ope source code, every aspect of Asterisk can be customized to meet your needs.

New interfaces and technologies are easily added to Asterisk. With Asterisk you can take control of your communications. Once a call is in your Linux sever with Asterisk, anything can be done with it Asterisk gives you fine-grained control over every aspect of your communications

Scenario - A Home Office

Julie is an outside sales rep for a company in Chicago. She covers the Southwestern region and lives in Phoenix. Julie has aDSL line coming in to her home office. The head office has an Asterisk server. The head office has a hig speedInternet connection.

Julie has a telephone on her desk that connects to her DSL line. A caller contacts the Chicago office by dialing the Chicago 800 toll free telephone number of th head office. The caller listens to the directory of extensions for the sale department. The directory gives choices for each of the regions. The calle selects the Southwestern region. Asterisk tells them the extension for Julie announces her name, and then announces it will contact her.

The Asterisk server in Chicago rings the telephone on Julie's desk. Since this call is being made over theInternet over Julie's DSL line, there is no long distance charge between Julie and the head office. If Julie doesn't answer within si rings, the caller is given the choice of leaving a message or returning to th Sales directory or talking with the operator.

An Asterisk system is a fraction of the cost of legacy PBX systems. The additional hardware that turns a small Linux server into a telephone system is inexpensive and readily available. Support is availabl from different sources including Signate.

Asterisk is incredibly efficient. A small PC will serve many telephone users. With Asterisk you can easily build a telephone system for the smallest or the largest enterprise, There are Asterisk server running thousands of phones right now. You can easily scale or combine Asterisk systems to serve an number of users in any number of locations.

When combined with low-cost Linux telephony hardware, Asterisk creates a PBX at a fraction of the price of traditionalPBX systens. While an Asterisk system is a fraction of the cost of legacy systems, it provides better functionality than the most expensive proprietary systems. Asterisk includes feature such as voicemail, interactive voice response IVR,) and conferencing which are very expensive in proprietary systems

Scenario - A Large Business

Asterisk can benefit a large business with offices in several locations. In this scenario, there are fifteen hundred employees. The main office is in New York. Distric offices are in Chicago and Los Angeles. Support is done at the Denver office.

Asterisk servers are in separate hosted facilities in New York and Chicago. The Asterisk servers communicate with each other over a high-speed Internet connection. Various Asterisk servers are needed to support this many users. The Asterisk servers communicate witheach other and each of the branch office over a high-speed internet connection. The hosted facilities are hardened an geographically separate from each other and the company offices.

With shared Asterisk servers, if one fails the another takes over. This is much safer for the company as there is no single point of failure. Even in the event of an outage at one of the main offices, telephone communications won't be disrupted.

If there is a problem in the office, employees can take their phones off their desk and move them to their home or another office. If there is a problem at the Chicago office, key employees can relocate to the New York office. They can tak their desk phones with them, or use phones already at the New York office Business goes on.

Users seeking support can call local numbers in any of the regions. These calls are routed to the support center in Denver. The calls are sent over theInternet so there is no long distance charge to the company. The user has called a loca number and has no long distance charge. This is called"toll bypass."

With Asterisk, you can make calls through the telephone company, or make calls over the Internet. With the appropriate hardware, Asterisk supports telephony over thePSTN without any Internet connection. It is much cheaper to send telephone calls over the Internet than through the telephone companies. Asterisk can pay for itself with the money you save on your phone bill.

With Asterisk PBX's and Interactive Voice Response (IVR) applications are rapidly created and deployed. The powerful command line interface and feature rich text configuration files support rapi configuration and real-time diagnostics

Web servers provide easy deployment of dynamic content, for example movie listings or weather reports. Asterisk can deploy dynamic content over the telephone, with the same ease. For example Asterisk can display contact or meeting information on the LCD panel of an IP telephone.

Asterisk's unusually flexible dial plan allows seamless integration of IVR and PBX functionality. Asterisks Features are easily implemented using nothing more than extension logic.

Asterisk supports a wide range of protocols for handling and transmitting voice over traditional telephony interfaces. Asterisk supports US and European standard signalling types used in standard business phone systems. This allows Asterisk to bridge between next generation voice-data integrated networks and existing network infrastructure. Asterisk not only supports traditional phone equipment it provides this equipment with additional capabilities

Scenario - A Busy User

Asterisk can benefit a busy user who travels frequently. A caller contacts the user's Asterisk system. Asterisk prompts the caller for their name. The caller say their name. Asterisk then plays a message asking them to wait for a momen while the called party is located.

The Asterisk server rings the office telephone at the headquarters and at the branch office, the home telephone and the cell phone of the user, all at the same time If any of the phones are busy, the caller is directed to voicemail. If the use doesn't answer any of the phones after six rings, the caller is prompted to leav a voicemail message.

If the user answers any of the phones, the Asterisk server announces the telephone number of the calling party, if caller ID is available. Then the Asterisk serve plays back the name the called party recorded. The user presses one on th keypad of their phone to accept the call, or three to refuse the call. If the use refuses the call, the caller is directed to voicemail. The Asterisk server sends text message to the user's cell phone indicating there is new voicemail.

Inter-Asterisk Exchange (IAX) is a Voice over IP protocol specific to Asterisk. IAX allows Asterisk to merge voice and data traffic seamlessly across disparate networks. When using Packet Voice, data like URL information and images can be sent in-line with voice traffic. This supports advanced integratio of voice and data that is not available in legacy systems

Asterisk provides a central switching core, with four APIs for modular loading of telephony applications, hardware interfaces, file format handling, and codecs[1]. Asterisk provides transparent switching between all supported interfaces. This is how Asterisk ties together diverse telephony systems into single switching network

Scenario - An International Business

An electronics manufacturer has main offices in San Jose, California with international offices in London, Tokyo, Hong Kong and Munich. Asterisk servers are in hosted facilities in San Jose, and Tokyo. Asterisk servers are in th Hong Kong, Munich and London offices.

All the Asterisk servers have high speed connections to the Internet. All the servers have connections to local public telephone systems.

Because the Asterisk servers are connected over the Internet, there are no long distance charges for calls between the offices. Any user in any office can call any user in any other office. These calls are routed over theInternet, that is they are toll bypass calls

The support staff for this company is all at the San Jose headquarters. Instead of having support staff in the London office, management decides to perform all English language support from San Jose. Users in London can call the London telephone number for the company. If they wish to contact support, thei call i routed to the San Jose office over the company's VPN. This is a toll bypass call.

Asterisk is primarily developed with GNU and Linux for x86. It is known to compile and run on GNU and Linux for PPC. Other platforms and standards based UNIX-like operating systems shoul be easy to port. Much work has been done to port Asterisk to BSD.

1. A CODEC is a compressor-decompressor. A CODEC is used to digitize voice into data or convert digitized voice back to an analog signal.

Who Made Asterisk?

Asterisk was originally written by Mark Spencer of Digium, Inc. Code has been contributed from Open Source programmers from around the world. Testing and bug-patches from the communit have proven invaluable in developing Asterisk. Asterisk is now an extremely successful team effort b the open source community.

What it Does

Let's start with a simple description of the way an Asterisk system works and what an Asterisk system can do for you. First is a description of an Asterisk system in your office. Next, larger systems that connect to theInternet are described. Last, there is a description of the connection between your Asterisk system and the phone company

VoIP (Voice Over IP) systems like Asterisk can use a computer to send and receive telephone calls over a data network.Telephone calls are sent over the network as data using IP, the Internet Protocol. Telephone calls are sent from one IP phone to another IP phone as data.

An Asterisk system often services many IP telephones, as many as a thousand or more. Standard analog telephones or other devices like fax machines can be connected with an inexpensive adaptor. With such a system, anyone in the office can call anyone else in the office. Calling outside the office, fo example anyone with a regular telephone, is described below

IP phones are not connected to wires you rent from the phone company, to the telephone company itself, or to telephone wires you have in your office. They are connected to your data network.

You can call from a VoIP phone on your network to any other phone connected to your VoIP system. VoIP calls go over your local data network, not thePSTN (Public Switched Telephone Network,) and not your local telephone wires.

You don't need a connection to the PSTN to make calls to other phones connected your local VoIP system. If you have two different office buildings, or offices on different floors, and they are connecte to your local area network, you call phones, or fax machines, in the other area. Those calls still trave over your data network.

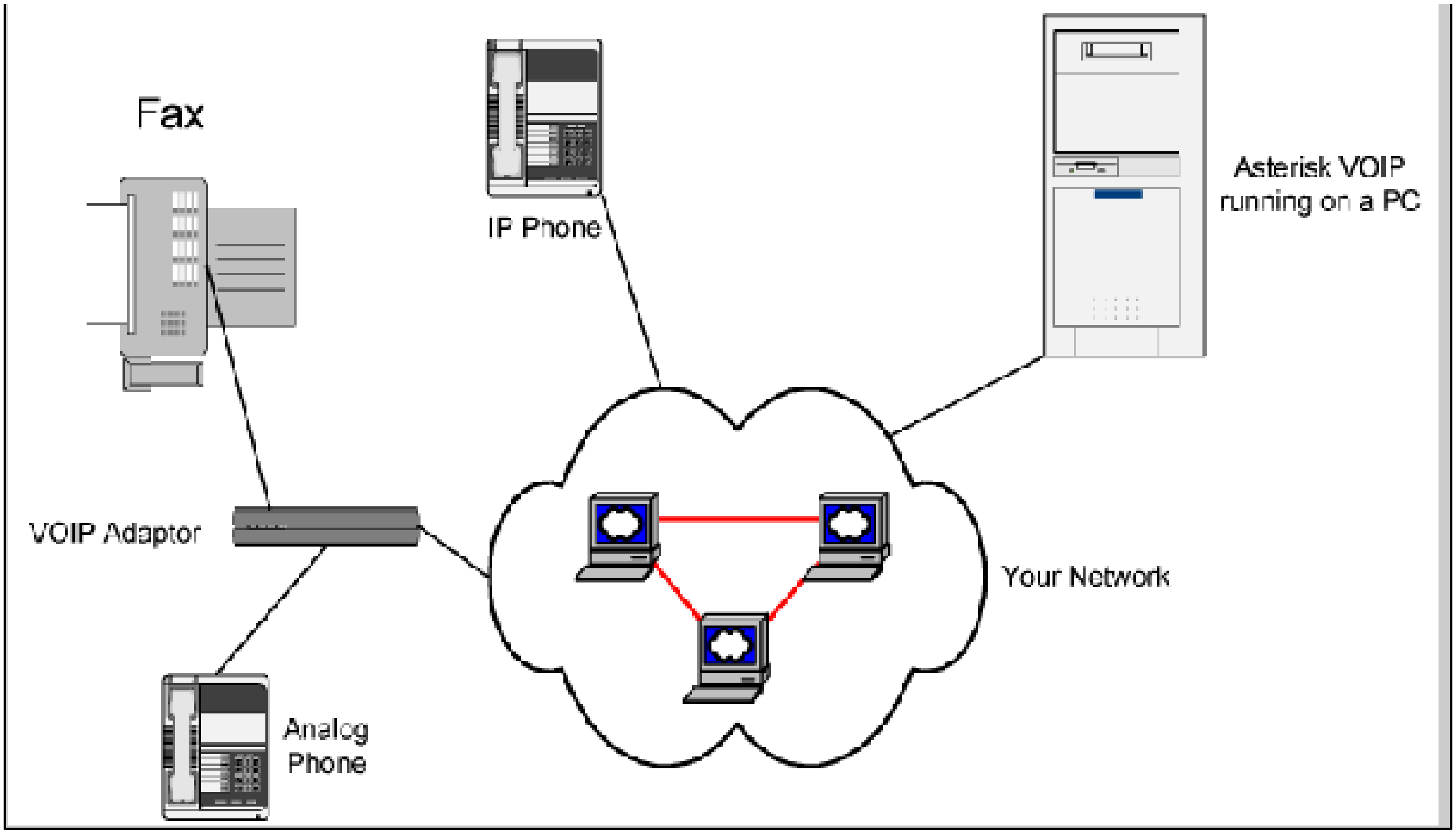


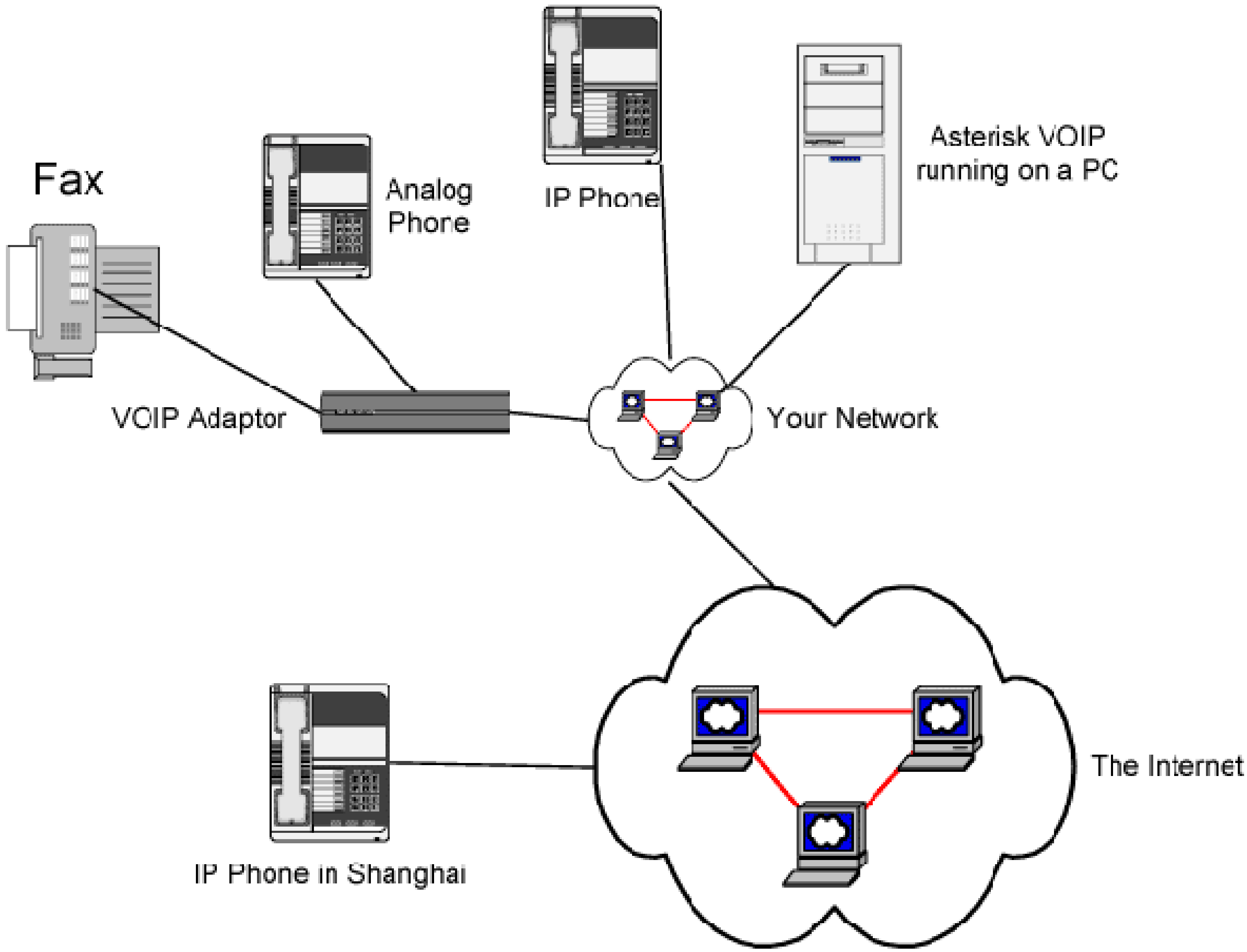
Figure: 01-1 IP Phones in the Office

Connecting your Office Telephone System to the Internet

As shown in the illustration, your Asterisk telephone system can easily be connected to the Internet. Any telephone can be easily connected to theInternet. You can connect an IP phone directly to the Internet. You can connect any standard analog phone or fax machine to the Internet with an inexpensive VoIP adaptor.

If your Asterisk system is connected to the Internet, any VoIP enabled telephone that is connected to theInternet can be allowed to connect to your Asterisk system. You can easily call any other VoIP phone serviced by your Asterisk system, no matter where that phone is. You can easily assure that th connections are secure and that unauthorized users are excluded. Any phone controlled by your Asterisk system can call any other VoIP or analog phone controlled by your Asterisk system.

It doesn't matter where a network connected phone is located. For example, you can have an Asterisk phone system in your office in New York and an office in Shanghai. Your Asterisk system in Ne York is connected to theInternet, and your Shanghai office is connected to the Internet. A phone in Shanghai connects to your New York Asterisk system over theInternet.The phone in your Shanghai office now works exactly like any phone in your New York office. When you dial the number for phone in the Shanghai office from your New York phone, the phone rings in Shanghai.



With a little bit of the right equipment you can install a phone at your home office and plug it into the Internet. Your office phone, now at home, communicates with your office Asterisk system over the Internet. Now, using your phone at home is just like using your phone in your office. No one would be able to tell where you are! You can take your phone on a trip and call from anywhere you have anInternet connection.

You can call anyone who uses a VoIP system, even if it isn't an Asterisk system. Your Asterisk system has to have a connection to their VoIP system. This can be a local network connection, or both system can be connected to theInternet. The call is sent over the data network or Internet, not the PSTN. Both systems must have the correct permissions and configurations.

Because the VoIP telephone call is sent over your data network or the Internet, there is never a long distance charge or a toll charge. The charge for the telephone call is included in the price you pay fo your network orInternet connection. This is one place you save money, no more toll charges or long distance charges!

Connecting Your Asterisk System to the PSTN

As shown in the following illustration, Asterisk users should be able to place calls to telephones connected to the PSTN. This requires a connection to the PSTN. Your Asterisk system has to be connected to the PSTN. This is easy to do.

Asterisk users need a telephone number if calls are to be accepted from the PSTN. You have to rent telephone numbers from a telephone company. You can rent a connection to your telephone company This connection is usually some wires they buried in the ground or wires they hung from poles.

Boards you add to the server running Asterisk connect the server to the connection you rent from the phone company. When someone dials your telephone number from thePSTN, your desk phone rings.

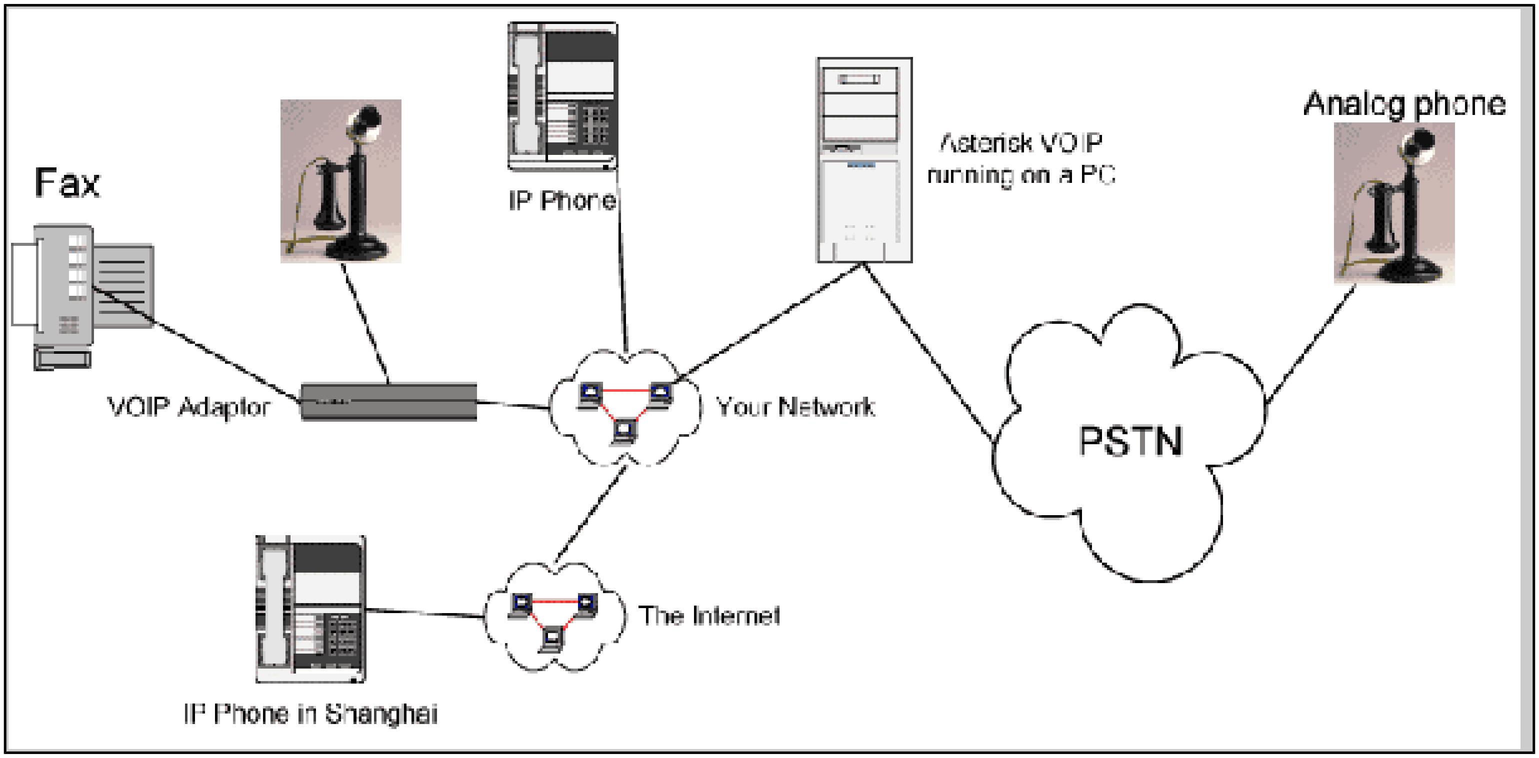


Figure: 01-2 Connecting to the Public Telephone Network

Asterisk Compared to Proprietary Telephone Systems

Various companies make a wide range of telephone systems from small to large. All the components of a proprietary system come from a single manufacturer. The single company designs and builds all th hardware and software for their telephone system. They manufacture the system themselves. None o their equipment will work with systems from other companies. This is how they control the price.

Manufacturers usually sell the largest systems themselves, through a dedicated sales force. A dedicated sales force is, of course, expensive. The cost of this sales force and all the support behind the sales forc is included in the price you pay for your telephone system

Anything smaller than the very largest systems are usually sold through representatives or distributors. The smallest systems are typically available through representatives or distributors

The price you pay for a proprietary telephone systems includes all the costs of manufacturing and distribution. The price has to be high enough to provide a profit for everyone in the distribution chain, the manufacturer, distributor, representative, retailer, etc. The cost of designing and manufacturing i spread over a relatively few systems from a single manufacturer. This makes proprietary systems ver expensive.

Asterisk is built with commodity PC hardware. Event the most sophisticated, industrial strength PC is far less expensive than any traditionallPBX. Since a PC is a commodity, PCs are inexpensive and your Asterisk system is inexpensive.

You may need interface boards to support telephony. For example, you may need a board that will let you hook up to an incoming telephone line. You may want a board that lets you connect fax machine in your office to your Asterisk system. The boards you add to the PC from companies like Digium ar inexpensive. An Asterisk system is far less expensive than any proprietary telephone system you migh consider buying for your business.

Proprietary systems are classified by their manufacturers by features. Do you want voicemail, that's more hardware and more money. Do you need a system that supports more users? That's a larger mor expensive system. A proprietary system will cost more for every feature you want. Features like voice-mail and anInternet connection will be expensive.

Each proprietary system in a manufacturer's product range is limited to a certain number of users. Adding more users requires adding more expensive cards to the system, or buying a more expensiv system. The manufacturer demands much more money for their more capable systems

A small inexpensive PC will run Asterisk and support a surprising number of users. Do you need an Asterisk system to support more users? You can use a larger PC. You can very easily use multipl Asterisk servers. If you ever have too many users for a single Asterisk system, spend a little bit mor money and put in another Asterisk server.

You won't be able to get the features available with an expensive proprietary system if you purchase an inexpensive proprietary system. Manufacturers do not put all the features they support into all th products they sell. There may be a feature you need or want that is only available with a more expensiv system.

Asterisk provides many features. Features only available in a proprietary phone system costing tens or hundreds of thousands of dollars are now available in your free Asterisk software. Asterisk has most o the features found on any high-end proprietary telephone system.

Asterisk is an "open source" product sponsored by Digium. (http://www.digium.com is the digium URL.) No company owns it.

A user community has grown up around Asterisk. When a developer from any organization adds a new feature, you get that feature too. Unlike proprietary systems, you can easily add your own features

As it is new, Asterisk may still lack a few features here and there, but it is easy to add new features to Asterisk. When someone in the Asterisk community adds the feature you want, you won't be charge extra for it. Since the product is open source, you can add you own features.

Asterisk has facilities proprietary telephone systems cannot provide. For example, Asterisk has a scripting system. This scripting system makes it easy to make Asterisk do amazing things. For example, you can write a script to have Asterisk call you in the morning to wake you up. You can write a script t have Asterisk read a weather or traffic report.

The following chapters describe how to design, install, configure, build and maintain an Asterisk system for your enterprise.

Partial Feature List

At the time of writing, Asterisk provides the following features. New features are regularly added.

- Telephony Services
    - Voicemail System Â
        - Password Protected
        - Separate Away and Unavailable Messages
        - Default or Custom Messages
        - Multiple Mail Folders
        - Web Interface for Voicemail Checking
        - E-mail notification of Voicemail
        - Voicemail Forwarding
        - Visual Message Waiting Indicator
        - Message Waiting Stutter Dialtone
    - Auto Attendant
    - Interactive Voice Response
    - Overhead Paging
    - Flexible Extension Logic
        - Multiple Line Extensions
        - Multi-Layered Access Control
        - Direct Inward System Access
    - Directory Listing
    - Conference Bridging
        - Unlimited Conference Rooms

        - Access Contro
  - Call Queuing
  - ADSI Menu System
    - Support for Advanced Telephony Features
    - PBX Driven Visual Menu Systems
    - Visual Notification of Voicemail
  - Call Detail Records
  - Local Call Agents
  - Remote Call Agents
  - Protocol Bridging
    - Provides seamless integration of technologies
    - Offers a unified set of services to users regardless of connection type
    - Allows interoperability of VoIP systems
  - Call Features
    - Music on Hold
    - Music on Transfer
      - Flexible mp3 based system
      - Volume Control
      - Random Play
      - Linear Play
    - Call Waiting
    - Caller ID
    - Caller ID Blocking
    - Caller ID on Call Waiting
    - Call Forward on Busy
    - Call Forward on No Answer
    - Call Forward Variable
    - Call Transfer
    - Call Parking
    - Call Retrieval
    - Remote Call Pickup
    - Do Not Disturb
  - Scalability
    - TDMoE
      - Allows Direct Connection of Asterisk PBX
      - Offers Zero Latency
      - Uses Commodity Ethernet Hardware
    - Voice over IP

        - Allows for Integration of Physically Separate Installations
        - Uses commonly deployed data connections
        - Allows a unified dial plan across multiple offices
    - Voice over IP Interoperability
        - Inter-Asterisk Exchange (IAX)
        - H.323 Session Initiation Protocol SIP)
        - Media Gateway Control Protocol (MGCP)
    - Traditional Telephony Interoperability
        - Robbed Bit Signaling Types
            - FXS and FXO
            - Loopstart
            - Groundstart
            - Kewlstart
            - E&M
            - E&M Wink
            - Feature Group D
        - PRI Protocols 4ES
            - Lucent 5E
            - DMS100
            - National ISDN2
            - EuroISDN
            - BRI (ISDN4Linux)
            - Codec Support
                - GSM
                - G.729 (available through purchase of commercial license(s))
                - G.723.1 (pass through)
                - Linear G.711 Mu-La
                - G.711 A-Law
                - ADPCM
                - ILBC
                - LPC-10
                - MP3 (decode only)

Getting Help

Commercial support for Asterisk development and Digium hardware is available from **http://www.digium.com**. Asterisk training and Asterisk support is available from Signate at **http://www.signate.com**..

Mailing Lists

You can learn a great deal about Asterisk by joining the mailing lists and reading the many messages sent each day or saved in the archives. Participation will help anyone with a serious interest in implementing an Asterisk system or coding on the Asterisk project.

The Asterisk mailings have three lists, asterisk-users, asterisk-dev and asterisk-announce. The asterisk-users and asterisk-dev are for users with implementation and support questions. They are helpful for developers who want to participate in the technological discussions about Asterisk. You can subscribe for individual messages or a daily digest version

Mark Spencer is the author of Asterisk and its primary sponsor Digium, Inc. Mark uses the mailing listasterisk-announce@lists.digium.com for infrequent major update announcements and press releases.

Subscribing & Unsubscribing

Subscribe or unsubscribe to Asterisk mailing lists at

`http://lists.digium.com/mailman/listinfo/asterisk-announce` Â

`http://lists.digium.com/mailman/listinfo/asterisk-users` Â

`http://lists.digium.com/mailman/listinfo/asterisk-dev`

Alternatively, send e-mail to mailman@lists.digium.com with 'help' in the subject or message body. You will get back an e-mail containing information on subscribing and unsubscribe via e-mail. Al administrative requests should be directed to

`mailman-owner@lists.digium.com`.

Modifying Subscriptions

To modify your subscription to an Asterisk mailing list click on the appropriate link above, enter your e-mail address, and click 'Edit Options'. Follow the instructions listed on the website or if you nee further assistance e-mailmailman-owner@lists.digium.com.

Browse & Search

To browse the Asterisk mailing list archives go to http://lists.digium.com/mailman/listinfo

To browse the old <asterisk@marko.net> mailing list archives go to

http://www.marko.net/asterisk/archives/

You can search the archives with the Google link found at

http://www.digium.com/index.php?menu=mailing_list

A wealth of information about Asterisk is available from the Asterisk mailing list found at

http://lists.digium.com

IRC

There is an Asterisk IRC channel available on

Server: irc.freenode.net Â

Port: 6667 Â

Channel: #asterisk

You can easily login to the freenode chat line at

http://www.digium.com/index.php?menu=live_chat

VOIP Forum

The VOIP forum has a large archive of useful technical information. You can access the forum at

http://www.voip-forum.com/

You can easily search the VOIP forum at

http://search.voip-forum.com/

Participating

You can, and should, contribute to Asterisk. Developers can contribute to the Asterisk code base with bug fixes, new features, enhancements, new applications or new channel drivers.

Please send any suggestions about improvements or corrections to this book to asterisk@signate.com

Licensing

Asterisk is generally distributed under the terms of the GNU General Public License, or GPL. This license permits you to freely distribute Asterisk in source and binary forms, with or without modifications, provided that when it is distributed to anyone at all, it is distributed with source code (includin any changes you make) and without any further restrictions on their ability to use or distribute th code. For more information, refer to the GNU General Public License

The GPL does not extend to the hardware or software that Asterisk talks to. For example, if you are using aSIP soft phone as a client for Asterisk, it is not a requirement that program be distributed under GPL. For those applications in which the GNU GPL is not appropriate (because of some sor of proprietary linkage, for example), Digium is the solely capable of licensing Asterisk outside of th terms of the GPL at their discretion. For licensing outside of the GPL contact Digium.

ChapterÂ 2 - Asterisk Architecture

Asterisk is middle ware that connects Internet and telephony technologies with Internet and telephony applications. Asterisk applications connect any phone, phone line or packet voice connection to an other interface or service. Asterisk easily and reliably scales from very small to very large systems Asterisk supports high density, redundant applications

Asterisk supports every possible kind of telephone technology. The technologies include VoIP, SIP, H.323, IAX, and BGCP (for gateways and phone.) Asterisk can interoperate with almost all standards-based telephony equipment. Hardware to connect your Asterisk system is inexpensive. Asterisk supports traditional telephone technologies likeISDN PRI and T-Carrier including T1 and E-1. Telephony applications include calling, conferencing, call bridging, voicemail, auto attendant, custo Interactive Voice Response scripting, call parking, intercom, and many others.

An Asterisk server connected to a local area network can control phones connected to that local area network. These phones can call each other through the Asterisk server. The Asterisk server can contro phones connected to other networks or theInternet, even if those phones or the Asterisk server are behind firewalls.

With Digium FXS interface cards, an Asterisk server can control local analog telephones. FXO and T-carrier interface boards from Digium can connect an Asterisk server to the PSTN. This allows calls to be made to and from thePSTN. PSTN users can call phones controlled by the Asterisk server, Asterisk phones can call users on the PSTN.

Calls can be switched from one Asterisk server to another Asterisk server. A telephone controlled by an asterisk server can call a telephone controlled by a second Asterisk server. A call from a telephon controlled by one Asterisk server can be switched to a second Asterisk server and then on to th PSTN.

As shown in figure one, Asterisk contains engines that perform critical functions. When Asterisk starts, the*Dynamic Module Loader* loads and initializes drivers. The drivers provide channel drivers, file formats, call detail recording backends, codecs, and applications, among others.

The Asterisk *PBX Switching Core* accepts telephone calls from the interfaces. The *Switching Core* handles calls according to the instructions found in a *dial plan*. The *PBX Switching Core* uses the Application Launcher to ring phones, to connect to voicemail, or to dial out on outbound trunks.

The *PBX Switching Core* includes a *Scheduler and I/O manager* that is available to drivers and applictions. The *Codec Translator* seamlessly connects channels that compressed with different codecs. Most of Asterisk's flexibility comes from the applications, codecs, channel drivers, file formats and othe facilities interaction with the various programming interfaces.

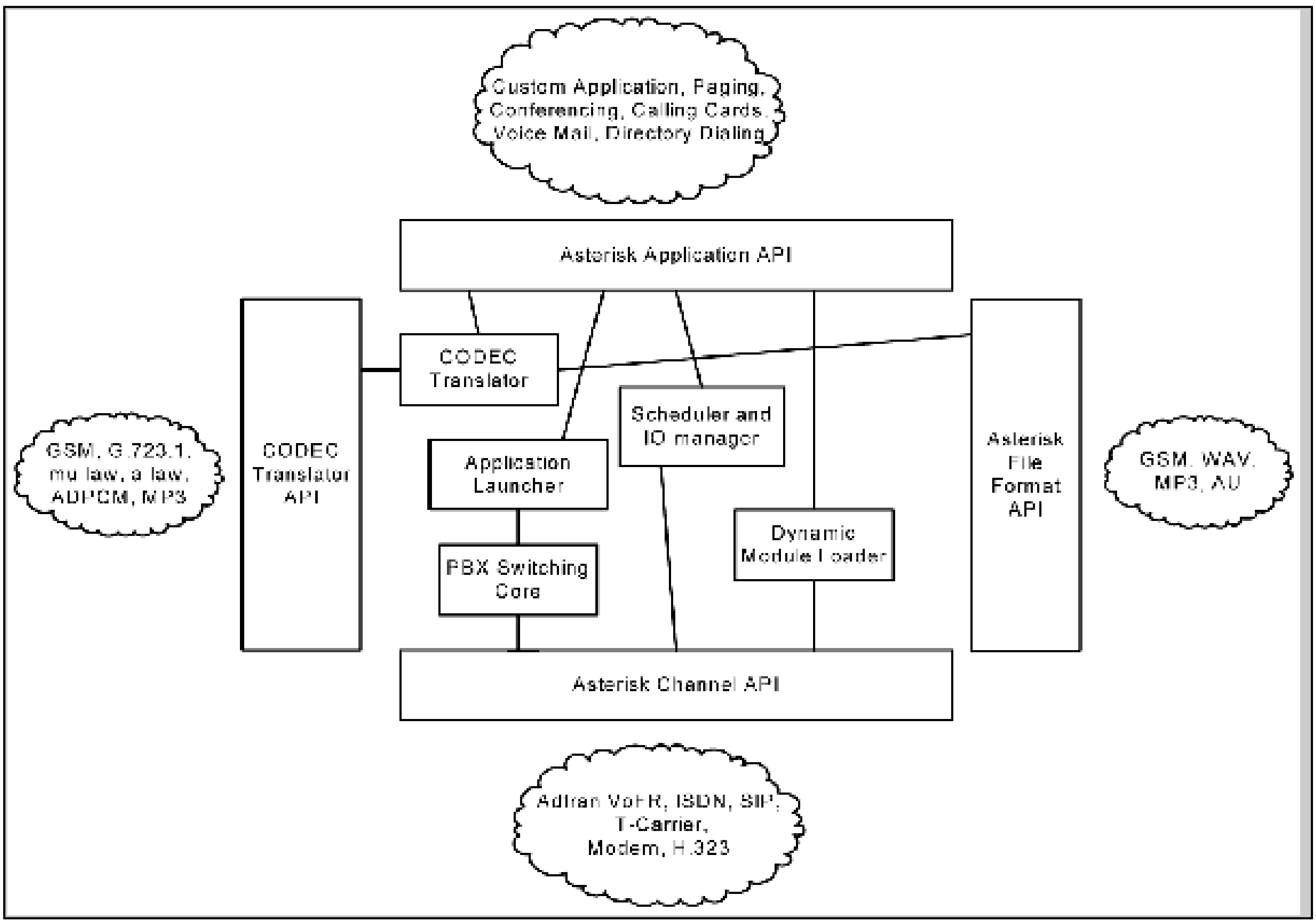


Figure: 02-1 Major Asterisk Subsystems

Interfaces & Channels

You must understand what interfaces are available and how they work to be able to install or configure Asterisk. You will never be successful in configuring or maintaining Asterisk unless you understan interfaces and their interaction with Asterisk

***All* calls arrive at or leave an Asterisk server through an interface, for example SIP , Zaptel or IAX. Any incoming or outgoing call is made through an interface.**

Every call is placed or received over an interface on its own distinct channel. A channel can be connected to a physical channel like a POTS line, or to a logical channel like an IAX or SIP channel.

It is very important to differentiate the arrival of a call on a channel from what is done with that incoming call. When a call arrives at Asterisk over a channel, a dial plan determines what is done wit the call. For example, a call might arrive through aSIP channel. The call could be coming from a SIP telephone, or from aSIP soft phone running on a computer. The dial plan determines if the call should be answered, connected to another telephone, forwarded or directed to voice mail.

Asterisk provides various applications, for example voice mail. These applications are available to the dial plan when processing the incoming call. The dial plan and the applications selected for use withi the dial plan determine what Atersisk does.

Different types of interfaces are associated with different kinds of hardware or protocols. For example, SIP channels are used to route calls in and out of an Asterisk server over IP with Session Initiation Protocol. A call can come in to an Asterisk server through aSIP channel or leave the Asterisk server outbound to theInternet through a SIP channel.

All calls arrive on a channel. Even internal calls. For example, a legacy analog telephone can be directly connected to an Asterisk server with the appropriate Digium interface board. When the user picks u the handset, a channel is activated. The user's call then flows through the activated channel. The dia plan determines what should happen to this call, for example dialing another internal number ove another analog channel, or dialing an outside telephone number, or accessing voice mail.

Asterisk uses a channel driver (typically named *chan_xxx.so*) to support each type of channel. An Asterisk channel is specified in this way

```
/
```

Technology is one of installed channel modules, i.e. SIP, IAX, IAX2, MGCP, or Modem. The format of the Dialstring depends on the type of channel selected. The standard distribution includes the following interface types

```
SIP - Session Initiation Protocol IETF
IAX - Inter-Asterisk Exchange protocol - v1 and v
MGCP - Media Gateway Control Protocol / Megaco IET
ZAP - Zapata channel
Modem - Modem channels (Incl ISDN)
Skinny - Skinny channels (Cisco phones)
Voice over Frame Relay - Adtran styl
```

```
console - Linux OSS console client driver for sound cards /dev/ds
vbp - VoiceTronix Interface drive
local - Loopback into another contex
H.323 - H.323 IT
phone - Linux Telephony channe
agent - ACD Agent channe
```

Outgoing channels, for example for the Dial application, use names with the same format. Later chapters describe how to configure various types of channels.

Hardware Interfaces

Asterisk supports a variety of hardware interfaces for connecting telephony channels through a Linux computer.

Zaptel Pseudo TDM Interfaces

All Digium Hardware shares a common driver suite and uses a common interface library. Digium drivers are based on the Zapata Telephony Driver suite. This set of drivers is often called"Zaptel." Zapata is an open source project available at**http://packages.qa.debian.org/z/zaptel.html.** The zaptel telephony infrastructure was jointly developed by Mark Spencer of Linux Support Services, Inc. an Jim Dixon of Zapata Telephony.

Even if no interface cards are installed, you must install at least one Zaptel driver to enable conferencing. Asterisk does not require a sound board to operate unless you are using a soft phone on the computer running Asterisk.

The zaptel interface uses the host processor to simulate the time division multiplexor (TDM) bus typically built into other telephony hardware interfaces (e.g. Dialogic and other H.100 vendors). The resulting pseudo-TDM architecture requires more CPU power but provides a substantial savings i hardware cost and a substantial increase in flexibility. Zaptel interface cards are available from Digiu http://www.digium.com) for a variety of network interfaces including PSTN, POTS, T1, E1, PRI, PRA, &M, Wink, and Feature Group D interfaces among others.

Traditional TDM hardware resources including echo cancelling, HDLC controllers, conferencing DSP's and DAX's are replaced with software equivalents. With software TDM, switching is still done in near-real-time, and call qualities are excellent. The pseudo-TDM architecture extends the TDM bus across Ethernet networks. Zaptel devices support data modes on clear channel interfaces, including Cisco HDLC, PPP, and Frame Rela

Non-Zaptel Interfaces

Interfaces for connectivity to traditional legacy telephone services that do support Pseudo-TDM switching includ

TABLE: 02-1 Non-Zaptel Interfaces

Interface	Description
ISDN4Linux	Basic Rate ISDN interface for Linux
OSS/Alsa	Sound card interfaces
Linux Telephony Interface (LTI)	Quicknet Internet Phonejack/Linejack

Dialogic	Full-duplex Intel/Dialogic hardware

Packet Voice Protocols

These are standard protocols for communications over packet networks like IP or Frame Relay. These interfaces do not rely on specialized hardware. These interfaces will work without specialized hardware.

- Session Initiation Protocol (SIP)
- Inter-Asterisk Exchange (IAX) versions 1 and
- Media Gateway Control Protocol (MGCP
- ITU H.32
- Voice over Frame Relay (VoFR

Linux Telephony Interface

The LinuxTelephony Interface was developed primarily by Quicknet, Inc. with help from Alan Cox. This interface is geared toward single analog interfaces and provides support for low bit-rate codecs.

The following products are known to work with Asterisk although they may not work as well as Digium devices.

- Quicknet Internet Phonejack (ISA, FXS)
- Quicknet Internet Phonejack PCI (PCI, FXS)
- Quicknet Internet Linejack (ISA, FXO or FXS)
- Quicknet Internet Phonecard (PCMCIA, FXS)
- Creative Labs VoIP Blaster (limited support)

ISDN4Linux

The ISDN4Linux interface is used primarily in Europe to connect lines from BRI interfaces to an Asterisk machine. Any adapter that is supported by ISDN4Linux should work with Asterisk.

OSS/ALSA Console Drivers

The OSS and ALSA console drivers allow a single sound card to function as a "console phone" for placing and receiving test calls. Using auto answer/auto hang up, the console can create an intercom

Adtran Voice over Frame Relay

Asterisk supports Adtran's proprietary Voice over Frame Relay protocol. The following products are known to talk to asterisk using VoFR. You will need a Sangoma Wanpipe or other frame relay interface to talk to them

Adtran Atlas 800

Adtran Atlas 800+

Adtran Atlas 550

Supported VoIP Protocols

Asterisk supports two industry standard and one Asterisk specific VoIP protocols.

Inter-Asterisk Exchange (IAX)

IAX is the Asterisk specific VoIP protocol. It is the standard VoIP protocol for Asterisk networking. It provides transparent interoperation withNAT and PAT (IP masquerade) firewalls. It supports placing, receiving, and transferring calls and call registration. With IAX, phones are totally portable. Just connect a phone or Asterisk server anywhere on theInternet. They will register with their home PBX and instantly route calls appropriately.

IAX is extremely low-overhead. IAX has four bytes of header, as compared to at least 12 bytes of header forRTP based protocols like SIP and H.323. IAX control messages are substantially smaller.

IAX supports internationalization. A requesting PBX or phone can receive content from the providing PBX in its native language.

IAX supports authentication on incoming and outgoing calls. Asterisk provides fine-grained control over access. Limits can be placed on access to only specific portions of the dial plan.

With IAX dial plan polling, the dial plan for a collection or cluster of PBX's can be centralized. Each PBX only needs to know its local extensions, and can query the central PBX for further information as required

Session Initiation Protocol (SIP)

SIP is the IETF standard for VoIP. SIP is described at greater length in a following chapter. SIP control syntax resembles SMTP, HTTP, FTP and other IETF protocols. SIP runs over TCP/IP and manages Real Time Protocol RTP) sessions. RTP transfers the data for a VoIP session. SIP is the emerging standard in VoIP because it is simple compared to other protocols like H.323 and human-readable. The AsteriskSIP interoperates successfully with multiple vendors including SNOM and Cisco

H.323

H.323 is the ITU standard for VoIP. Support for H.323 in Asterisk was contributed by Michael Mansous of InAccess Networks (http://www.inaccessnetworks.com), and is based on the OpenH.323 project http://www.openH323.org).

While H.323 support is present in Asterisk, H.323 is a dying standard. Whenever possible you should use a more modern interface likeSIP or IAX.

Codec and file formats

A codec (compressor/decompressor) is used to compress analog voice into a digital data stream or to decompress the data back into an analog signal. Asterisk can operate with a wide variety of codecs an file formats. Because of its open architecture, it is easy to incorporate additional codecs or file formats.

There are two common 64 kbps PCM compression standards, micro-law and a-law. Both use logarithmic compression to effectively achieve 12 to 13 bits of linear compression in 8 bits. Logarithmic compression reduces higher volumes or frequencies exponentially. Micro-law is slightly better in compressing low level signals and has a slightly better signal-to-noise ratio. Micro-law is commonl used in North America, a-law is commonly used in Europe

Asterisk provides seamless, transparent translation between any of the following codecs.

TABLE: 02-2 Supported Codecs	***Â***
Codec	**Rate**
`16-bit linear`	`128 kbps`
`G.711u (micro-law)`	`64 kbps`
`G.711a (A-law)`	`64 kbps`
`IMA-ADPCM'`	`32 kbps`
`GSM 6.10`	`12 kbps`
`MP3`	`variable, decode only`
`LPC-10`	`2.4 kbps`

In addition, other codecs, such as G.723.1 and G.729 can be passed through transparently.

Note that you should use the alaw, ulaw, or linear codecs to use in-band DTMF. Note that most codecs are too lossy to support fax transmissions.

Note that a codec determines how information is encoded. This is different from a file format. A stream of data compressed with a codec could be saved in different file formats.

File Formats

Asterisk uses files to store audio data including voicemail and music on hold. Asterisk supports a wide variety of file formats for audio files. Supported formats includ

TABLE: 02-3

format	description
raw	16-bit linear raw data
pcm	8-bit micro-law raw data
vox	4-bit IMA-ADPCM raw data
wav	16-bit linear WAV file at 8000 Hz
WAV	GSM compressed WAV file at 8000 Hz
gsm	raw GSM compressed data
g723	simple g723 format with time stamp

Quality of Service

Quality of Service (QoS) is the ability of a network to provide improved service to selected network traffic.QoS support is available in a variety of networking equipment, for example routers. QoS tools can let you manage the end-to-end efficiency of your voice traffic. A detailed discussion ofQoS is beyond the scope of this book. You can pursue this topic elswhere, including RFC3290.

QoS provides priority service to selected traffic to optimize the use of available bandwidth, control jitter and latency and improve loss characteristics. QoS tools provide control over congestion management, queue management, traffic shaping and policing, and link efficiency. This makes it easier for mission-critical applications to co-exist on a network. OptimizingQoS for one data flow should not make other data flows fail. Many routers and switches provide facilities for managing QosS

For example, you may have a small office with a DSL line. The DSL line might have 384 kbps of bandwidth bi-directionally.QoS tools would allow you to dedicate 128 kbps of the bandwidth of the DSL line specifically to telephony. This would mean there would always be bandwidth for telephone calls no matter how busy theInternet connection gets carrying other traffic.

File System Organization

The following table shows where Asterisk related files are stored.

TABLE: 02-4

Directory	Description
/etc/asterisk	All configuration files except /etc/zaptel.conf
/usr/sbin	Asterisk executables and scripts including **asterisk, astman, astgenkey and safe_asterisk**.
/usr/lib/asterisk	Asterisk architecture specific binary objects
/usr/lib/asterisk/modules	Runtime modules for applications, channel driver, codes, file format driver, etc.
/usr/include/asterisk	header files required for building asterisk applications, channel drivers and other loadable modules.
/var/lib/asterisk	Variable data used by Asterisk during normal operation.
/var/lib/asterisk/agi-bin	AGI scripts used by the dial plan AGI application
/var/lib/asterisk/astdb	The Asterisk database, hold configuration information. This file is never changed by hand. Use Asterisk *database* command line functions to change, add to and modify this file.
/var/lib/asterisk/images	Images referenced by applications or by the dial plan.

/var/lib/asterisk/keys	Private and public keys used within Asterisk for RSA authentication. IAX uses keys stored here.
/var/lib/asterisk/mohmp3	MP3 files used for music on hold. The configuration for music on hold is found in the directory **/var/lib/asterisk/sounds**.
/var/lib/asterisk/sounds	Audio files, prompts, etc. used by Asterisk applications. Some applications may hold their files in subdirectories.
/var/run	Runtime named pipes and PID files
/var/run/asterisk.pid	Primary Process Identifier (PID) of the running Asterisk process.
/var/run/asterisk/ctl	Named pipe used by Asterisk to enable remote operation.
/var/spool/asterisk	Runtime spooled files for voicemail, outgoing calls, etc.
/var/spool/asterisk/outgoing	Asterisk monitors this directory for outbound calls. An outbound call results in a file in this directory. Asterisk parses the created file and attempts to place a call. If the call is answered, it is passed to the Asterisk PBX.
/usr/spool/asterisk/qcall	Used by the deprecated qcall application. Don't use.
/var/spool/asterisk/vm	Voicemail boxes, announcements and folders.

Applications

Asterisk includes many applications. These applications perform useful functions like dialing a telephone number or saving a voicemail message. These applications are described at length in the chapter on Asterisk configuration.

ChapterÂ 3 - Connectivity

This chapter describes connections between your Asterisk system and the Internet or the PSTN. You must be familiar with the information in this chapter in order to design, install and configure an Asterisk system.

If you are already familiar with IP Telephony and standard telephony including T-Carrier, you may wish to skip this chapter. For more in-depth information aboutT-Carrier, consult the later T-Carrier chapter. IP telephony protocols, for exampleSIP, are described in a later chapter. There are many excellent books about telephony if you wish more in-depth information, for example*Voice over IP Fundamentals* by Jonathan Davidson.

Two separate networks are available, the PSTN and the Internet. They each provide different services. Telephone numbers are used to address a specific device on the PSTN. IP addresses are used to address a specific device on theInternet.

Because the public telephone network is optimized for voice, it is not well suited for data transmission. Since voice can easily be digitized, theInternet is well suited to transmitting digitized voice. Because of this, the currentPSTN with all its channels is growing obsolete. Over the coming years the PSTN is moving to a new IP Internet Protocol) architecture. Many telephone carriers already have a serious financial commitment to this change

Connecting Asterisk to the PSTN or Internet

With Asterisk, telephone calls can be routed over an IP network including the Internet. If two users are connected to Asterisk, they can communicate over a data network, no telephone company i needed.

Accepting calls from users on the PSTN requires a telephone number. Telephone numbers are only hosted on thePSTN. Telephone numbers are rented from a supplier, a telephone company.

Making or receiving telephone calls from the PSTN requires a connection to the PSTN. Direct connections to the PSTN can be rented from a telephone company.

The PSTN is built with channels, for example the pair of wires that run from your phone to a phone company switch, or the channels that make up a T1 circuit. A channel provides a dedicated connectio between one telephone and another telephone for the duration of the call. Consult the chapter title *T-Carrier* for an in-depth description of T1 lines and an extremely brief introduction to SONET.

When you make a telephone call over the PSTN, you consume a channel for the entire call. Only your telephone call goes over the channel. You and the called party have exclusive use of the channel for a long as the call lasts

A POTS (Plain Old Telephone Service) line has a single telephone number associated with it. Calls to that telephone number are routed over a dedicated circuit. An Asterisk server connected to aPOTS line can send and receive calls over that circuit.

You can rent POTS lines from a telephone company, if they are not out on strike. You can connect thesePOTS lines to your Asterisk system. Digium cards allow you to connect a POTS line to your Asterisk server.

There may be different companies (alternate carriers) in your area that provide telephone numbers and connections. Alternate carriers often rent at least part of their network, for example the wires to you premises, from your local telephone company.

A direct connection to the PSTN can be a larger connection, for example a T-Carrier connection or some other even larger connection. Digium cards interface withT-Carrier lines. Your telephone numbers are associated with this connection. Calls to your telephone numbers are routed to you Asterisk server over theT-Carrier connection.

A T-Carrier connection provides multiple channels. A T1 line provides 24 voice channels. If you have twenty-four users in your office, and twenty-four telephone numbers, and a T1 line, every user has a available line. This means twenty-four incoming or outgoing calls can be placed concurrently.

There can be more telephone numbers, or users, than circuits. You can have more telephone numbers thanT-Carrier channels. If you have fifty telephone numbers and a T1 circuit, calls to any of the fifty numbers can be sent over any of the twenty-three T1 channels to your Asterisk server. The world wid telephone system has many more users than channels. That's why you get a busy signal after an emergency when everyone is trying to get a channel

The service provided with a T-Carrier line signals what number is ringing. This allows Asterisk to appropriately route the incoming call.

In addition to a telephone number and connections, telephone companies provide additional services like local or long distance calling. You can usually get long distance or international calling from a variety or providers.

A new generation of telephone companies provides the best of both worlds. These companies will provide telephone numbers, and route calls over the Internet or PSTN.

You can connect to an Internet telephone company that provides a bridge to the PSTN. Instead of a connection to thePSTN, you use a connection to the Internet. A call placed to your telephone number is sent from that provider to your Asterisk server over theInternet.

A T-Carrier circuit can connect to a telephone company, or to an Internet provider. T-Carrier lines connected to a telephone company use the individual channels for individual telephone calls.

A T1 used for a network or Internet connection uses all the T1 channels to transmit data. Different kinds of data (including voice) share all the channels. Different kinds of data are sent over the connection simultaneously. All the available bandwidth of the line is shared to send data.

A T1 line with a public line interface that is connected to a telephone company can support only twenty-three simultaneous calls. Because voice compresses well, more concurrent calls can be place over a T1 line where all 24 channels are used for a data network connection. The number of call depends on the compression scheme you select. More calls can be sent at the sacrifice of voice quality Good quality networking equipment can help you maintain the quality of service for your telephon calls

Sending voice over the PSTN is expensive compared to sending voice as data over the Internet. Unlike anInternet connection, PSTN channels aren't shared.

Internet Connections

There are a variety of ways to connect to the Internet. The following table compares some of them. Some connections are symmetrical, that is they are just as fast in both directions. Some connections like a satellite connection, are much faster in one direction, for example down from the satellite to you.

TABLE: 03-1

Connection Name	Relative Speed	Connection type	Speed	Monthly Cost	Simultaneous Calls
Modem	1	telephone	56 kbps	$40	one, maybe
Satellite	1 up 5 down	radio	56 kbps up 512 kbps dow	$175	one, maybe
ISDN	2	telephone	128 kbps	4 cents per minute	two
DSL	2-4 up 4-10 down	telephone	128 kbps - 6 mbps	$30 to $300	two
Cable Modem	2-4 up 5-48 down	broadband cable	128Mps or more up to 6 mbps down	$45	one, maybe
T1	25	telephone wire	1.544 mbps	$450 up depending on distance	23 to 40
T3	625	Telephone wire	44.736 mbps	Â	Â

OC-1	2,500	Â	155.5 Mpbs	Â	Â

Most small businesses will do well with a T1 line or a business grade DSL line. The time delay of a satellite link makes them impractical for most business settings. The inexpensive satellite links are ver low bandwidth up to the satellite. The higher speed satellite links are very expensive. The asymmetrical speed of a cable modem makes them impractical for IP telephony in a business setting.

There are various wireless links like 802.11 that can provide high speed data access. These are not listed in the table as they are not commonly available from commercial providers.

Quality of service is a very serious issue. Most businesses rely heavily on their telephones to do business. If your phones are out, you may be out of business. T1 type lines usually come with a service levl agreement (SLA.) If the line goes down, someone fixes it within an agreed upon time. Most of th other connection types, includingDSL, may not have a service level agreement.

Lastly, you may be sharing your data connection with voice and data traffic. In this case, you may want special loadQoS or traffice shaping that pre allocates bandwith for telephone calls. This will assure t hat calls will always get through ahead of data services.

Renting Telephone Network Connections

Over time, because connections are becoming less expensive, Internet connections are becoming less expensive. You should shop to find the best price for a T1 line from a company who may actually sta in business.

Sadly, there is no central location I have found that lists all the companies that sell Internet connections in your area. There are some Internet sites that will refer your inquiry about T1 lines to companies that pay them for the referral. This is annoying because you can't find all the local vendors. Referral agencies will insist on getting your contact information. Worse yet, they will actually try t contact you to sell you a T1 line.

Your local phone company is always a potential source of a T1 line, although they may not be the most cost effective solution.

If you connect to the Internet with a T1 line, the line goes from your office all the way to your Internet provider's facility. When you are connecting to theInternet, the T1 channels will send data instead of telephone calls. If you use anInternet connection for VoIP calls, the calls are sent over the T1 line as data.

You rent a T1 line, usually from a telephone company, by the month. You may pay for it by the mile. The cost often depends on how far it is between the end points. The cost usually depends on th amount of wire that you need to connect between your office and yourInternet provider. The phone company calls this"wire miles." It's the length of the wire in miles between you and them.

T1 connections are usually point-to-point. The T1 line goes from your office to your Internet provider. Usually, the T1 uses wires that your local telephone company owns. That means your T1 goes fro your office to your telephone company and then from your telephone company to yourInternet provider.

The local loop is tariffed. This means the government has approved what the local loop costs. This means that the price for the local loop is usually going to be the same no matter who you buy your T from.

For the part of the T1 line that runs from the local telephone company to your selected end point, you can always get service from an alternate vendor. You pay the alternate vendor for both parts, the loca loop and the remaining connection. When an alternate vendor quotes you a price for your T1 line, yo will most likely be quoted two amounts. One amount will be for the local loop, the other amount wil be for the remaining portion of the T1 line. Here the prices can vary a lot. This is where it pays t shop.

You may not need all of a T1. Part of a T1 may be enough for your application. This is called a fractional T1. You can often rent a fractional T1.

With the right equipment you can share a single T1 between network and PSTN connections. For example, you could devote 12 channels of your T1 to anInternet connection and 11 to telephone calls.

Lastly, if you are cautious and you can afford it, you might want two different connections from two different companies. That way, one connection is always likely to be working.

Other Providers for PSTN Connections

There are providers who will rent you telephone numbers and connect you to the PSTN over a network connection instead of a PSTN connection, for example voicepulse.com. Your Asterisk system connects to their VoIP system over yourInternet connection. They have a connection to the PSTN. They will provide you with telephone numbers and a bridge to thePSTN.

Tie Lines

Consider a business with offices in two different locations. If there is sufficient call volume between the two sites it may be cost effective to rent a tie-line. A tie-line is a permanent circuit between the tw offices. This is often a T1 or E1 or fractional T1 or E1. For a tie-line to be effective it must be les expensive that using thePSTN. This is, of course, a function of call volumes and distance.

Hosted VoIP Systems

You can obtain VoIP service from an outside vendor like Signate, http://www.signate.com. The VoIP system is at their site. Your local phones connect to their system through theInternet or a point-to-point connection. They will maintain the system for you and provide you with the telephone numbers you need. The only equipment you need in your office are your telephones or fax machines.

You may want to host your own VoIP system off site. For example, if you rent space for all your Internet related equipment at a hosting center, you may want to put your VoIP system there. You could share the data connection from your office to your hosting center for voice and data.

The phone company provides this service. It is called Centrex. When you host your own Asterisk server you can get all the facilities ofCentrex at a fraction of the cost.

You may want to share one Asterisk system between several offices. You could use data connections between the offices to share the single Asterisk system.

Sharing a Connection

Many small businesses do not need all of a T1 connection. If you are in a location near other small businesses, you may be able to share a T1 connection with your neighbors. If you are friends with you neighbors at home, you can share a T1 connection to your home. You can connect your neighbors t your T1 line with wireless equipment and share the cost.

Note that there are security concerns surrounding a shared connection. You will need the appropriate hardware and software to share a connection safely. This subject is beyond the scope of this book.

If you are located close to a larger number of other businesses, you could even share a larger connection like a T-3. A T-3 is 28 times bigger than a T1, but it isn't 28 times more expensive. A T-3 is usuall inexpensive compared to 28 T1 lines.

Various types of equipment are available to help you insure that no one user takes more than their share of the line.

Other Types of Connections

There are a few circumstances where you won't need to get a local loop from your local telephone company. If other companies have run wire or fiber optic cables into your neighborhood, you may not need your local telephone company.

If your VoIP system is in a remote hosted facility, a company like AT&T or Sprint may have a high speed fiber optic connection into the facility. You may be able to connect to this circuit with a T1 lin and not need a local loop from your telephone company.

T1 Alternatives

DSL (Digital Subscriber Line) can give you just as fat a pipe to the Internet as a T1 line. DSL usually doesn't have an SLA. This means if yourDSL line goes down, you might have to wait a long time for it to be fixed. A DSL line might be an excellent backup for when your T1 line isn't working. You may be able to get a businessDSL line with a SLA.

Many carriers are now providing DS-1 circuits over HDSL lines with a single pair of copper wires. This is a less expensive alternative toT-Carrier circuits and does not require repeaters.

Frame over DSL is usually less expensive than a T1 line. Frame over DSL replaces the T-

Carrier (described below) portion of the network. It is easier to manage, but the management services that ar available are not as extensive. It is more difficult to get a good SLA with this technology.

This service is becoming more widely available. It was initially used for slower speed connections, but is now becoming more commonly available at T1 speeds. Frame overDSL isn't available in all locations because DSL isn't available at all locations.

There are other connections available as well, for example, 802.11 wireless, "wireless T1" or licensed wireless connections like microwave. You might have fiber optic connections available in your neighborhood from your phone company or another company. These can provide very fast connections.

Some connections like a dialup connection are not as suitable for VoIP. Cable modems usually do not have enough speed from you to theInternet. A cable connection may provide enough bandwidth for a single conversation.

Satellite Connections

A Satellite connection is only palatable when there is no other alternative. Most satellite connections provide little bandwidth from you to the satellite.

There is a very long annoying delay on a satellite call, as much as two or three seconds, between when you say something and when the calling party hears it. This delay comes, in part, from the 22,50 miles the signal has to travel up to and back from the satellite. There are other propagation delays i the system.

The voice quality of a SIP call depends on the available bandwidth and the reliability of the connection. IAX is probably preferable to SIP for Satellite traffic.

ChapterÂ 4 - Designing Your System

This chapter will help you design an Asterisk system for your enterprise. This chapter will assist you in designing your system, sizing your system and selecting the appropriate hardware and communication links

Consulting and Support

You may want help installing, configuring, monitoring and maintaining your Asterisk system. Signate, provides Asterisk design, installation, integration, training and management services anywhere in th world. You can reach Signate at***www.signate.com***, by telephone at 415.442.4011, or my email at ***support@signate.com***

Hardware Vendors

At the time of writing, the following vendors specialize in providing hardware from Digium and other supplies for use with Asterisk Systems.

APB International

APB international specializes in the distribution of high-end technology products including Voice over IP solutions based on Open Standards for converged data and voice communications. The company serves resellers in North an South America.

www.abptech.com

(972) 745-1220

Cylogistics

Cylogistics is a specialty distributor serving the open enterprise reseller community with a special focus on telephony including VoIP, IP, & SIP technology.

www.cylogistics.com

(800) 749-2734

The Map

How do you get to a working Asterisk system? Here is your map. You must:

Find out what the business requirements are--talk to management and users.

Document the current functionality. What does the existing system do? How does it do it

Design an Asterisk installation that meets existing and new requirements.

Design and install any needed infrastructure including a local area network, Internet connection, or telephone network connection

Design and build the Asterisk system including the server and peripheral equipment.

Configure the Aterisk system for your environment.

Install the new system.

Test the new system including all connections and echo suppression.

Document the system including operating procedures and user guides.

Train the users.

Deploy the new system.

Support and maintain the system.

Backup and monitor the system.

Periodically upgrade the system.

Plan for disaster recovery.

Each of these steps is vital. If you get any of these steps wrong, your project will fail.

Requirements

Talk to your users and management to determine your business needs.

What features do the users require?

How much voicemail will there be?

How many users are there now?

How many users will there be in the future?

How many phones are needed?

How many IP phones, how many analog phones?

How many fax machines are there?

Are there existing telephone numbers that must be kept?

What will the connection to the telephone system be? Analog lines or T1?

Will there be multiple providers for the PSTN or long distance?

How many simultaneous calls will there be, on average and at maximum?

What are the requirements for long distance service or toll free numbers?

Is the telephone wiring you are going to need already installed? If not you will have to design and install phone wire. There are other resources than this book that describe telephone and networ wiring.

What will the connection to the Internet be? How much bandwidth is needed for the Asterisk system? Is a separate Internet connection required for

Asterisk? What kind of Internet connection is available

Is the local area network already installed? If not you will have to design and install it. Is it sufficient, or will you need more network connections or even a new network? Network design and installation is beyond the scope of this book

Here are some questions designed to help you collect requirements. This will help get you started, it is not a complete list. There are useful pre-installation checklists in the appendix.

Services

How many incoming lines do you have/need?

How many incoming and outgoing calls per day do you average?

Do you need Emergency, 911 dialing.

Do you need video conferencing.

Do you want Voice Encryption.

Do you need direct inward dialing (DID,) that is telephone company service?

How many modem and FAX lines do you need?

If you need DID, for how many employees?

What is the expected growth over the next 5 years?

Do you need phones in public areas?

Do you need phones in conference rooms?

How many conference rooms do you have?

How many people will need a telephone?

How many people will need voicemail?

How many people will need caller ID?

How many people will need speaker phone capabilities?

Do you need dial-in capabilities for mobile users?

Do you want/need an automated attendant?

Will you have a receptionist who will answer and route calls?

Do you need voicemail?

What features do you want in voicemail if it is needed?

Do you need an overhead paging system?

Do you need door entry systems with an intercom?

Do you need to be able to turn phones on and off (hotel, hospital, and so on)?

Telephone Wiring

Do you have telephone wiring in place for analog phones or fax machines?

If there is existing wiring, is it adequate?

How will the phones be powered, transformers or inline on the ethernet?

Do you have wire and phone jacks in the desired locations?

Network

Do you have room for a phone server and the associated cable plant?

Do you have several buildings that will be served by this phone system?

If you have room, is it climate controlled?

If you need to wire for the phone system, will this be done in-house or contracted?

How difficult will it be to pull cables in your facility?

Do you know the local and state codes for wiring in your facility?

Do you have existing data lines like T1 or DSL?

Will these lines be shared or will new lines be needed?

Do you have an on-site programmer?

Do you have an on-site system administrator?

What is your existing network infrastructure?

Do you have routers, hubs, firewalls or switches?

Is there an installed ethernet?

Does the ethernet run to every workstation including fax machines or conference rooms?

What is the quality of the existing network? CAT5 or CAT 3? 10baseT or 100baseT or 1000baseT?

How heavily loaded is the existing network?

Legal Issues

You should have a contract with your buyer. What are you responsible for? What are they responsible for?

What happens if the telephone system fails? Are you financially responsible for any business losses?

What happens if a user needs to call for emergency services and the call doesn't go through? Are you responsible or liable?

Do you have a written service agreement?

Service Issues

Who will support the Asterisk users? What support hours will be required? Business hours? Evenings? Weekends? 24 by 7?

How many support staff will be needed? In how many locations? Who wills service hardware, for example, servers, telecom equipment or network equipment? What service level agreements ar required?

Quality of Service

What is the interaction between the Asterisk server and the existing network? Will Asterisk share an existingInternet connection? Will Asterisk users share an existing data network? How heavily loaded is the network? What will happen if the network is attacked, for example a denial of service attack What will happen if a backup is started across the network? What will happen if a user drags and drop 1,000 files across the network?

Reliability

What is the electricity supply like? Is there backup power? How long will backup power last? How long will the Asterisk server and all the related equipment run during a power outage?

Is there backup equipment? Is there a backup Asterisk server? Is there automatic failover? Is spare equipment easily available? Are spare communications boards readily available? Is there automatic cal forwarding to alternate telephone numbers in case of an Asterisk or communications failure

Change Management

Aterisk is rapidly evolving. New versions are available on an almost daily basis. New features and facilities are being added. How and when do you move to a newer version?

Maintain a copy of any installed systems. Have backups available in case the move to a newer version fails.

Moving to a newer version will require testing outside of the production environment. Test any new system completely in a test environment before deploying it.

Deploying a new system may require changing documentation or operating procedures and more user training.

Server Hardware

You need a server running Linux. If you install Linux yourself, it's much easier to install all the distribution, all the packages and all the source code. This will waste some disk space, but disk space is cheap. The Mepis release of Linux athttp://www.mepis.org comes pre-configured for Asterisk.

If your installation is a business PBX, you need redundant hardware to approach the "five nines" reliability of a traditional PBX. Get a server with ECC memory, RAID-1, dual power supplies and hot swappable disks. Keep a spare hard drive and spare interface boards on hand

In addition to the computer, you should have a power backup system. **If your users expect to be able to call for emergency services through the Asterisk server, backup power is critically important**

An uninterruptable power supply (UPS) will isolate your asterisk computer from power problems. It will keep you Asterisk server running for some time when the power is out. The UPS can communicate with the Asterisk server to provide for a graceful shutdown after a power failure. Note that othe network equipment, for example switches or routers, and telephones will need to be serviced with UPS. Newer IP phones use power over ethernet. This makes providing emergency power easier.

Make sure you have a current service agreement with an appropriate response time commitment. Consider installing a redundant system, or having a spare system, or at least spare parts on hand.

Sizing Your Server

An inexpensive server with a 2GHz processor, 512Mb of memory and 60GB of disk space can run Asterisk for a small to medium size office.

The size server you will require depends heavily on the architecture of your system. The type and mixture of phones--analog, SIP, Skinny, or H.323 or soft--makes a difference. The number of phones makes a difference. The mixture of internal and external calls makes a difference. The network bandwidth and quality make a difference. Transcoding is very CPU intensive as is echo cancellatio

As an example, a single machine with a 2.6 GHz Pentium 4, 1 GB of RAM and 3 T1 connections can manager 40 concurrentSIP to Zap conversations and over 5000 total phone calls per day. The load on a server like this can in a matter of moments vary from 0.00 to 6.25

Interface Hardware

To connect between your Asterisk server and the phone network, you will need an interface board. For example a T1, E-1 orFXO analog interface card from **http://www.digium.com**.

For guaranteed access to emergency calling services like 911 consider having at least one landline available from the telephone compan.

An FXS analog interface card from Digium will allow you to connect analog phones and fax machines directly to your Aterisk server. These phones can use your existing telephone wires. If you wish t switch facsimile traffic through an Asterisk server you must use a lossless codec and you must have a high quality network connection. Significant packet loss or high latency will prevent facsimile transmission or reception.

Network Hardware

An ethernet interface connects your Asterisk server to your local area network. You can connect IP telephones to this network. You can use IP adaptors, for example the Cisco ATA-188, to connect analog phones to the local area network

IP telephones and IP adaptors require power. Some IP phones and adaptors can draw their power from a remote source over the ethernet cable. Powering the phones over the ethernet makes it easier t provide backup power. You can provide a single UPS for the switch instead of trying to provide a UP for each phone. The UPS will keep the switch, and the IP phones, running during any power outage It will be more expensive and more difficult to maintain backup power for individual phones.

Telephones

SIP phones are available from a number of vendors including Cisco, Snom, Polycom, IP Dialog, ATelNet, Swiss Voice and Grandstream.SIP adaptors for analog phones are available from several vendors including Cisco, Motorola and Sipura

A number of software phones are available for use with Asterisk including

```
XTENSIP phone: http://www.xten.com/
ESTAR SIP phone: http://www.estara.com
SJPhoneSIP phone: http://www.sjlabs.com/
eye SIP phone: http://www.eyepmedia.com
GnoPhone LinuxSIP phone: http://www.gnophone.com/#
Asterisk IAX Phone by Steven Sokol:http://www.sokol-associates.com
Asterisk IAC Phone:http://iaxclient.sourceforge.net/iaxcomm/
```

There must be a sound card on the machine where the soft phone runs.

Sizing Your Network Connections

If you are using a T1 connection to the PSTN for telephone service you should determine the percentage of time your users are on telephone calls. Count the number of telephones in the office including conference rooms and fax machines. Try and find out the usage patterns for the phones. Is there ever time when everyone has to be on the phone? If not, fewer than the 23 channels may be enough for you office and you can rent a partial T1.

Asterisk uses a CODEC (Compessor Decompressor) to change an analog voice signal into a digital data stream and back. Several different CODECs are supported. You can select the CODEC yo want to use. This process is described later.

For calling over the Internet or LAN, you must have network connectivity and sufficient bandwidth. Each telephone conversation will consume from 45 to 150 Kilo-bits per second of bandwidth depending on sound quality. At 50Kbs call quality is comparable with a cell phone. At 75 Kbs call quality can rival a land line call.

The CODEC selection determines how many calls can be sent over your Internet connection. John Todd, an accomplished Asterisk consultant, has tested various CODECs. John has graciously permitted the inclusion of his results here.

TABLE: 04-1 CODEC Bandwidth Requirements		
CODEC	**Estimated Calls per Mbs**	**John's Comment**
G.711 (ulaw	15	Â
ILBC	47	Â
G.729	103	Â
GSM	68	Â
LPC10	164	Users may not be pleased with the voice quality.
SPEEX	57	Â

Buy Configuration Services

You may find that after you have purchased your hardware, purchasing installation for your Aterisk system from a vendor like Signate is an advantage. This can dramatically reduce the number of problems you will encounter and the time it will take to solve problems. A Signate installation can includ a support agreement.

Software and Configuration

Download and compile the Asterisk software. Again, the details are in later chapters.

Add any interface boards. Add the drivers for the interface boards to your Linux system. Note that the Asterisk software download is always the most recent development branch. You may have do download again at a later time to get a working version of Asterisk.

You must configure your network. This may include making TFTP (Trivial FTP) available. You will most likely need to configure DHCP (Dynamic Host Configuration Protocol.) For more informatio about DHCP, refer to RC 2131, 3396, and 3397

You must configure your Asterisk server for your environment. This is covered at length later in this book.

Configure any IP phones and IP adaptors. Install any analog telephone equipment.

Testing and Documentation

Test your system thoroughly before letting your users try it. You must deliver a reliable, complete working system or you will alienate your users and your project will fail.

Test the full system including all the connections. Make sure any SIP, H.323 or PSTN connections operate correctly. Test all thePBX functions. There are different ways to transfer calls. Do they all work with all the protocols and phones you are using? Does the transfer button on yourSIP phone transfer calls to other nonSIP phones or a different manufacturer's phones? Can non-SIP phones transfer calls toSIP phones? Create a grid of choices to assist your testing.

Test echo cancellation and change it as needed. If you don't test echo cancellation in advance, you are sure to get complaints from your users.

Document what you have done. Document your system hardware and software architecture.

Rollout

Test the system in the IT department before rolling it out to your company. Consider bringing a few users on line first. Don't try to bring the whole business up at once. Get some buy-in from early users A few happy test users will be very helpful in converting everyone else to happy users.

Train your users well. If your users aren't trained, they will fail and you will fail.

Provide at least some simple documentation for your users. User's rarely read documentation, but they may look at a short guide that gives them vital information quickly.

Upgrades or Changes

Install new systems or additions in off hours. Test thoroughly in a test environment before deploying. Test thoroughly in the production environment in off hours before deploying to users.

Maintaining

Keep clear records about hardware and software vendors, maintenance agreements and contact information.

If parts are critical, purchase spares. For example, at the fastest it could take a day or two to get a new or replacement interface card from Digium. Stock a spare so that you can quickly respond when something goes wrong.

Share Your Experience

Asterisk is an open source project. Don't just go to the user forums for help. Share your experiences there and give others a helping hand.

What's left?

The telephone system is the life's blood of any enterprise. Nothing you can do will upset your users more than interrupting their telephone service. To survive, you must plan ahead and execute well. Yo must be responsive to the continuing needs and desires of your users.

If you implement your system correctly, you can have happy users. There are many happy users of Asterisk systems. If you do your job right, your users will be happy.

ChapterÂ 5 - Install Linux and Asterisk

Asterisk will run well with any stable Linux distribution. The bootable CD downloadable from-***www.mepis.or***g contains the Mepis distribution of Debian Linux. The Mepis distribution is pre-configured to make it easy to install Linux and Asterisk.

You may choose to use a different distribution of Linux than Mepis. There are many excellent references available if you need to learn how to install or manage any Linux distribution including Debian versions.

Booting from the Mepis CD on a PC provides immediate access to a working Linux system. Linux will boot and run from the CD without installing anything. You can run Linux from the included CD or you can permanently install Mepis Linux to a hard drive

This book, and this chapter, assume that you are familiar with Linux administration and network administration. If you have never used Linux before, becoming proficient with Linux before installin and running Asterisk is a large undertaking. While it's possible, it could take a great deal of time. Network administration has a substantial learning curve

Asterisk was built for the Linux operating system. Some work has been done to port Asterisk to other operating systems like BSD. The path of least resistance, and greatest reliability, is to install Mepis an Asterisk.

This chapter shows you how to install Linux and Asterisk on your PC. The required steps are

Install and configure Linux.

Install and configure telephony related hardware.

Download and compile asterisk.

Configure asterisk.

After you have installed Asterisk, you will have to configure any adaptors, for example T1 adaptors, that you have installed. This is described in separate chapters.

After you have installed Asterisk and configured any adaptors, you will need to configure Asterisk for your environment. A later chapter describes Asterisk configuration

This chapter assumes that your Asterisk server is connected to the Internet, at least while you are installing and configuring Asterisk. AnInternet connection is required for downloading Asterisk.

Information in this chapter concerning DHCP, TFTP and NTP configuration should be noted when installing any version of Linux or Unix

PC Hardware Selection

Linux and Asterisk are both efficient consumers of computing resources. Simple hardware will usually run Asterisk well. For example, an Asterisk system for a small office with ten seats can run comfortabl on a PC with a 2 GHz processor, 256MB of memory and an ethernet adaptor. A 40GB drive wil allow you to install Linux and Asterisk and have a considerable amount let over for voicemail. Make sure there are enough open slots for any communications boards you will be running, for exampl Digium T1,FXO or FXS adaptors.

A minimum configuration might be a 1GHz processor with 128MB of memory and a 20GB disk.

Telephony Hardware Selection

Asterisk will run as a VoIP server with no telephony interface boards. This can make for a very useful system. An Asterisk server can use Inter-Asterisk Exchange (IAX) to connect to a remote Asteris server. If the remote server has the required Digium boards and an interface to thePSTN, the first server can access thePSTN through the remote server with IAX.

Even if you don't have any interface boards installed, you must install the Zaptel drivers to use confer-encing.

Telephone interface boards that work particularly well with Asterisk are available at an attractive price from**http://www.digium.com**. Digium provides boards to interface to T-Carrier, POTS and local Analog devices.

Linux Installation Issues

The Mepis Linux distribution includes all the Linux software you need to run Asterisk. Mepis is a Debian Linux distribution. Other distributions may require more work to install and configure. Asterisk should install easily and run well on a recent Linux distribution.

The easiest way to guarantee the operating system packages Asterisk requires are available is to install all Linux source packages and utilities when you first install Linux. This may waste some disk spac but make your installation much simpler. The Mepis distribution includes all necessary sources an libraries.

You should be running Linux 2.4.x. You must have installed the runtime packages for bison, cvs, gcc, and libtermcap-devel. Before building and then installing Asterisk, you must install the full source fo the Linux kernel, the source for openSSL including headers, NCurses4, Ncurses C++ Devel, SOX and the source for the readline library including headers. Everything you need is included with th Mepis distribution.

This book assumes that you are working as the root user. Mepis includes *dhcp3*. If you are installing the *dhcp3* package for another distribution, you should be logged in as a different user.

Wait until after you have installed and configured Linux to install any telephony hardware. Don't install any telephone related hardware yet. Consult the later chapters for assistance with hardwar installation and configuration.

Getting Help

The Asterisk mailing list is always a good place to start when seeking help. To find the mailing lists, consult the support page atwww.asterisk.org. Support for Digium hardware is available from Digium, www.digium.com. Commercial Asterisk support is available from Signate, www.signate.com or info@signate.com.

You can register your Mepis distribution. This will provide you with access to support resources and including updates. If you need assistance installing or configuring Mepis Linux, commercial paid support is available. Please contact Mepis at*http://www.mepis.com*. Tell them Signate sent you!

Installing Mepis Linux

Boot from the Mepis CD after successfully booting from the CD, you will see the prompt

```
boot
```

Don't press any keys! Just wait, and mepis will continue the boot process. Wait until you see the mepis login screen. Mepis will run linux entirely from the CD. After the boot process is complete, you wil see a login titled*Welcome to MEPIS linux*. Logon as *root* with the password *root*.

Mepis will start KDE and initialize itself. This will take a few minutes. Booting from the CD is slower than booting from a hard drive. Next you will see the Mepis Linux desktop.

Click on the icon labeled *MEPIS Installation Center.* Click on *Install MEPIS on Hard Drive.* Read the notice and then click on*Next*.

Look for choice *1c* and select *Auto-install using entire disk*. More complex installations are beyond the scope of this book.

Click on *Next* and answer *Yes* to the question *OK to format and use the entire disk (dev/hda) for Mepis.* This will partition and format the hard drive. Mepis is then copied to the hard drive

For the next dialog select *Next* to install lilo in the system boot disk master boot record. On the next dialog select*Yes*, and then on the next *OK*.

In the next dialog select a password for the default account *username* and for the Root account. Select *Next* to continue.

For the next dialog enter a computer name and computer domain. If you want this server to participate in an Microsoft Networking workgoup, enter the name of the group. Select*Next* to move to the next dialog.

Turn off the Guarddog firewall service for now. You can start it later after you have Asterisk successfully running. Select *Next*.

Turn on the Apache web server if you want to access Asterisk through via the Web. Start the SSH server, the dchp3 server and the tftp server. SSH will allow you to access the machine from remot locations or from other machines on your local area network. VariousSIP telephones require dhcp and tftp. Select*Next* and then *Finish*.

Type ctrl-alt-del to bring up the shutdown screen and stop your computer. Remove the CD from the drive. Start the computer again.

As one or your installation options, turn on the tftp server. To access the machine remotely, turn on SSH. If you want to use your Asterisk server for DHCP, turn this on during installation as well. If yo are placing the server behind a firewall, and you would like to access it from outside the firewall, forward the ports for tftp and ssh

Mepis Network Configuration

As you will be running Asterisk as a server, you should configure the network interface for your Server with a permanent IP address. The*Mepis System Center* will allow you to easily change your network settings. Open the System Center. Select*Network Interfaces* You will need to have an IP address for the Asterisk server, a subnet mask, and the addresses of twoDNS servers. Use the *Interface* tabs to set the adaptors and the*Status* tab to start and stop the interfaces. Detailed Linux network administration is beyond the scope of this book.

Network Time Server

You may wish to configure your Linux server to periodically set the system clock by accessing an Internet time server. This is a good idea. Mepis by default enables network time resolution.

The Mepis directory */etc/cron.daily* contains a file named *ntpdate*. The file permissions must be *-rwxrx-rx*. In this file the command *rdate* sets the system clock. You can can use a time server of your choice as long as you are within the server's usage policies. Replace*your.server.com* shown below with the IP address of a time server. A list of public time servers is available at*http://www.eecis.udel.edu/~mills/ntp/ clock2a.htm*.

```
#!/bin.s
rdate -ssome.server.com
```

Sound Card and MPG Installation

A sound card is not required for Asterisk operation. The copy of the mpg audio software shipped with some Linux distributions including Red Hat will not work with Asterisk. If you are going to use musi on hold you will need mpg123. The mpg software on the Mepis CD works. If you need mpg123. I can be found at

`http://www.mpg123.de/mpg123/mpg123-0.59r.tar.gz`

Alternatively, from the command prompt you can type

```
# cd /usr/src Â

# wget http://www.mpg123.de/mpg123/mpg123-0.59r.tar.gz
```

Extract the archive and compile it.

```
# tar -zxvf mpg123-0.59r.tar.gz
# cd mpg123-0.59
# make linu
# make instal
```

Make sure the compiled package is in */usr/bin/mpg123*.

Firewall

If you install the Guarddog firewall, and you want to access the machine remotely, you will have to enable access to your machine for SSH or whatever access utilities you may prefer. It is better to leav the firewall off, at least during the initial steps of configuring and connecting your Asterisk server.

DHCP Server

You may require a DHCP server, for example for configuring SIP phones dynamically.

The Mepis distribution comes with an installed and operational DHCP server. This server has been configured to be the authoritativeDNS server on its network. The DHCP configuration file is found in */etc/dhcp3* in the Mepis distribution. Here is a sample dhpcd. conf file.

```
# Sample configuration file for ISC dhcpd for Debian and Asteris
# Signate, LLC 12/15/0
# $Id: dhcpd.conf,v1.1.1.1 2002/05/21 00:07:44 peloy Exp $


# The ddns-updates-style parameter controls whether or not the server
 will
# attempt to dDNS update when a lease is confirmed. We default to the
# behavior of the version 2 packages ('none', since DHCP v2 didn'
# have support for DDNS.
ddns-update-style none

# Gateway
option route192.168.1.1;

# Change this to the domain name where youDNS servers live
option domain-name"yoururl.com";

# IP addresses for your domain name servers
option domain-name-serve206.16.128.12, 209.16.31.12;

# URL of a network time protocol server
option ntp-servetick.usno.navy.mil;

option tftp-server-nam"192.168.1.10";

default-lease-time 600;
max-lease-time 7200;

# If this DHCP server is the official DHCP server for the local
# network, the authoritative directive should be uncommented.
authoritative;

#192.168.1.0 netmask 255.255.255.0 {
  range192.168.1.100 192.168.1.150;
```

After configuring DHCP, you can restart the DHCP daemon with the commands

```
cd /etc/init.
./dhcp3-server restar
```

TFTP Server

Some phones, for example Cisco phones, require access to a TFTP sever. They download their firmware and configuration settings from TFTP. TFTP is installed and enabled on the Mepis CD. In the Mepis distribution,*/var/tftp* is the default TFTP directory.

If you would rather run TFTP from a Windows server, you will have to find and install a TFTP server. NoTFTP server is included with Windows.

In other distributions, make sure the TFTP sever directory named in the configuration file exists. Make sure this directory has universal read and write permission. Make sure all files in the*TFTPboot* directory are readable

Be sure to test TFTP by requesting a file from a machine separate from you server. Many operating systems, including Windows, include aTFTP client. The Mepis TFTP installation writes log messages to */var/log/syslog*. TFTP for Red Hat 8 leaves its message in the file */var/log/messages.*

Download Asterisk

There is no option on the Mepis CD to install Asterisk from the CD. You can order an install CD for Asterisk from Signate, or use cvs to copy the most recent version of Asterisk to your computer.Use cv to copy the most recent version of Asterisk to your computer. Your Asterisk server must be connecte to theInternet to download the source code. CVS must be installed on your computer. CVS is automatically installed with Mepis. You must have root permission to perform these operations. From a shell at the command prompt, execute the following commands.

```
# cd /usr/sr
# export CVSROOT":pserver:anoncvs@cvs.digium.com:/usr/cvsroot"
```

After issuing the following command, you will be prompted for a password, use *anoncvs*.

```
# cvs logi
```

The following commands will create three directories within */usr/src* named *zaptel*, *libpri* and *asterisk*.You must, of course, have Internet connectivity for this command to work. This command will checkout the Asterisk sources to your server.

```
# cvs checkout zaptel libpri
```

To check out the stable release instead of the development release, use the command. For an Asterisk server you plan to put in production, you should use this version.

```
# cvs checkout -r v1-0_stable asterisk
```

To check out the development branch, use the command

```
# cvs checkout asteris
```

The cvs command will display many lines as the various sources are checked out of cvs and copied to your Asterisk server.

Install any Digium Telephony Boards

Next, install any Digium cards. Reboot the computer. In some Linux environments, for example Red Hat, Kudzu may inform you of the new hardware. Allow Linux to detect and install any new hardware Use the Kudzu dialog to configure the computer for the new PnP boards.

Be sure to have any hardware, for example T1 cards, installed in your server before you compile Asterisk. Any boards will need to be configured later. This is covered in later chapters.

Timing Sources

The music on hold application and conferencing rely on access to a timing source. Three sources are available, the Zaptel drivers used with Digium's Wildcard boards, ztdummy, or zaprtc which uses th system clock.

If you install any Digium Zaptel card, loading the driver for the card with the *modprobe* command automatically sets up the Zaptel interface. Timing is then automatically available with no further configuration

The ztdummy zaptel driver provides timing information when no Wildcard board is installed. *Ztdummy* is a kernel module that you load with the Linux command modprobe. The *ztdummy* driver can provide timing information. It is available in the zaptel directory from the Asterisk CVS repository.

The ztdummy module uses USB-UHCI timers found in linux USB drivers. You must load UHCIUSB as a module before loading ztdummy. Ztdummy won't work if you try and compile uusb-uhci it into the kernel

The ztdummy driver is included with the MEPIS Asterisk source. It is not compiled by default. To include ztdummy in your Asterisk installation, edit the makefile in*/usr/src/zaptel*. Remove the # in front of*ztummy.o* from the following line

```
MODULES=zaptel.o tor2.o torisa.o wcusb.o wcfxo.o wcfxs.o
ztdynamic.o ztd-eth.o wct1xxp.o wct4xxp.o # ztdummy.
```

When you make zaptel as described in the following section, ztdummy will compile.

You will have to load the ztdummy kernel driver before starting Asterisk.

```
modprobe ztdummy
```

To make the change permanent, edit the file */etc/modules* and insert the line

```
ztdumm
```

When you reboot the machine ztdummy will now load. To see a list of loaded drivers run the command

```
lsmo
```

A third timing source is available from [*http://www.junghanns.net/asterisk*](http://www.junghanns.net/asterisk)*/*. *Zaprtc* uses the system clock to provide timing information. To use this module, you will need to recompile the kernel withou real time clock support. You will need to change to the kernel source directory and disable enhance real time clock support in menuconfig Note that this utility will not work on a multi-processor system The module zaptrtc will replace the standard real time clock module and includes extra facilities fo Zaptel.

Compile the Asterisk Packages

Any telephony boards, for example a Digium T1 card, should already be installed in your computer.

Various drivers are needed to operate Asterisk. These drivers are derived from the open source Zapata project. These drivers are found in the zaptel directory.

Even if you don't have any interface boards installed, at least one ZAPTEL interface has to be installed to enable applications that require timing, for example voicemail and meetme conferencing.

As the super user, from the command prompt issue the following commands. Please note that the order of these commands is important. The commands should be executed in the order shown.

```
cd zaptel
make clean ; make instal
cd ../libpr
make clean ; make instal
cd ../asteris
make clean ; make instal
make sample
```

The Asterisk compilation can take ten minutes or more depending on your computer. The other compilation steps should finish in a few minutes or less.

A later chapter shows how to run Asterisk. You will need to configure Asterisk before you run Asterisk. You are now ready to configure Asterisk. Asterisk configuration is described in a later chapter.

Use *make update* to update Asterisk to a more current version. After an update, restart Asterisk for the changes to take effect.

Compiling builds any drivers required for the installed telephony hardware. You do not need to restart your server after these compilation steps

The last step, the make of the samples, creates a variety of sample configurations. Configuration is described in a later chapter.

Common Build Errors and Warnings

You may be using a Via motherboard with a C3 processor. If you are, you may get the error message

```
Via C3 is not an i68
```

Resolving Zaptel Compilation Issues

Compiling the Zaptel package requires a version of the kernel sources that matches the kernel version running on your system. Check the version with following commands

```
cat /proc/versio
uname -
```

The output from this command will be similar to

```
Linux version 2.4.28 (rootlocalhost) (gcc version 3.2 20020903 (Red Hat
  Linux 8.0 3.2-7)) #1 Tue Jan 28 11:01:02 CST 200
```

In this example, the kernel source of 2.4.28 version in */usr/src.*

```
ls -ld /usr/src/linux
```

should be

```
lrwxrwxrwx 1 root root 12 Feb 10 2003 /usr/src/linux > linux-2.4.28
drwxr-xr-x 17 root root 4096 Jan 27 2003 /usr/src/linux-2.4.1
```

Make sure that the config file for the running kernel is available. The *.config* file is often in the */usr/src/ linux/*directory. You may also find it in the /boot directory. The version number should be the same as the version number of the kernel sources.

```
ls /boot/config*
/boot/config-2.4.2
```

In the kernel source directory, create a kernel config file.

```
"cd /usr/src/linux; make menuconfig"
```

Load the current kernel config file and exit saving a new config. Execute this make command to create the modversion.h kernel header file. Zaptel requires this file be present.

```
"make dep"
```

The zaptel sources should compile now.

Reporting Bugs

If you find a bug with Asterisk, you should report it by going to bugs.digium.com. This is a great service to the Asterisk community.

A Custom Debian Kernel

If you have installed a custom Debian kernel, the kernel Makefile in */usr/src/linux/ Makefile* may not have the correct*EXTRAVERSION* variable.

If matching the Kernel, as described in the section directly above, doesn't work. examine the Makefile. Make sure the version information in the Makefile matches the information returned by the comman

```
#uname -
```

If needed, edit the Makefile and try compiling again.

Installing Red Hat 9

At the time of writing, the complete guide to Red Hat 9 installation could be found at *https://www.redhat.com/docs/manuals/linux/*

You should have the Red Hat Linux version 9 Installation media. Boot the PC with the *Red Hat Linux 9Installation* CD. At the selections during installation choose the language, keyboard and mouse settings. If there is an existing operating system installed on the computer, you will be given an opportunity to *Perform a new Red Hat Linux Installation*. Next choose a *custom* installation. Configure your disk partitions, boot loader and network setting. At the dialog for firewall configuration select*No Firewall*. Select the language, time zone, root password and authentication settings for your system.

In the package group selection screen scroll to the bottom and select *Minimal Installation* and *Select Individual Package*. At this screen select *Flat View*. From the displayed list, select the following packages.

```
bison Â
cvs Â
gcc Â
kernel-source Â
libtermcap-devel Â
newt-devel Â
ncurses-devel Â
openssl096b Â
openssl-devel Â
readline42 Â
realine-devel
```

The next screen shows the required dependency packages. Select *next* to install the required packages and*next* again to start the installation.

When the installation has finished, you will be given the choice to create a boot disk. After this step installation will be complete. The CD will*eject*. Click on the exit button. This will restart the server.

You will now have to configure the various packages like DCHP and TFTP.

Installing Red Hat Fedora

Here are some tips for installing Red Hat Fedora.

1) Install Fedora Core 1 with all the development environments. *This is available at http:// fedora.redhat.com/download/*. Be sure to install the kernel development source. You will not be able to build Asterisk without the kernel development package.

Here are some suggested choices for choices you will have to make while installing Fedora.

```
1. Upgrade: new installation Â
2. Install Type: completeÂ
3. Partitioning: automatic, remove all partitions Â
4. Firewall: Only install if you know how to configure firewalls.
```

```
Â

5. Package Group Selection: Â

Desktops (none), Applications (Editors, Text-based,Internet, Sound
and Video, Servers (all), Development (Development Tools, Kernel
Development), System (all) Â

5b. Static vs. Dynamic IP Address: Static Address
```

You can configure the system to boot to run level three. You may want to turn off any non-essential services. For remote access, enable SSH.

ChapterÂ 6 - Asterisk Configuration

Before configuring Asterisk, you must configure any hardware you are using. This includes SIP phones, soft phones, channel banks or communications boards. The following chapters show configuration for these various channel types.

After any hardware and channels have been configured, you can configure Asterisk.

Getting Help

Much of the information in the book came from the Asterisk Wiki pages at *http://www.voip-info.org/ tiki-index.ph*. This is a gold mine of Asterisk information. While I have mined some of the gold, there is still a lot left for you to find. The Asterisk community done a tremendous service to the community in creating this resource.

You can get help from the Asterisk mailing list. Consult *www.asterisk.org* for more information on support and the mailing lists.

Digium, of course, offers Asterisk support and free support for issues related to their hardware. Sig-nate, *www.signate.com*, is in the business of supporting Asterisk.

Configuration Files

Configuration files control Asterisk operation. Samples are provided to help you get started more quickly. Sample configuration files are also provided with the Asterisk distribution. You should b familiar with Asterisk architecture as explained in the earlier chapter before attempting to configur Asterisk.

After installing Asterisk and making the samples, the following configuration files are present in */etc/ asterisk* You will have to modify many of these files to adapt Asterisk to your needs. The following chapters will assist you.

```
asterisk.conf - Configuration directories for components
agents.conf -  agents
enum.conf -  ENUM lookups
extensions.conf - The dial plan
festival.conf -  interface to Festival Speech Synthesis software
H.323.conf -  H.323 channels
iax.conf -  IAX channels
indications.conf -  various indications (busy tones etc)
manager.conf -  the Asterisk manager API
meetme.conf -  conferences, MeetMe
mgcp.conf -  MGCP channels
modem.conf -  ISDN and Modem connections
modules.conf -  Asterisk module loading
musiconhold.conf -  the MusicOnHold command
parking.conf -  call parking
queues.conf -  call queues
rtp.conf -  RTP ports for media
sip.conf -  SIP channels
voicemail.conf -  voicemailboxes
zapata.conf -  digium interface cards
```

Figure: 06-1 Configuration Files

Configuration File Syntax

Asterisk configuration files are flat ASCII files.

Comments

A semicolon starts a comment. Anything from the semicolon to the end of the line is treated as a comment and not acted upon. For example,

```
CONSOLE=Console/dsp    ;  This is a comment
;CONSOLE=Zap/
```

The # sign is used to indicate extensions and is thus not used for comments.

Lines

A configuration file includes multiple lines. There is no continuation between lines.

Sections

Configuration files are divided into Sections. Sections group lines of similar purpose.

Sections are named with a string inside square brackets. The string can contain letters, numerals. and the underscore character. For example,

```
[general]
```

Variables

Variables are assigned values with the equals sign.

```
myvar = myvalu
```

Variables set within a *[globals]* section are available from anywhere within the configuration file. Here are some examples

```
[globals
CONSOLE=Console/dsp ;Console interface for dem
CONSOLE=Zap/
CONSOLE=Phone/phone
IAXINFO=guest ;AXtel username/passwor
IAXINFO=myuser:mypas
TRUNK=Zap/g2  Trunk interface
TRUNKMSD=1  ;Most Signficant Digits to strip (usually 1 or 0
TRUNK=IAX2/user:pass@provide
```

Variables may be set with *SetGlobalVar* in an extension definition. Here is an example,

```
exten => s,2,SetVar(counter=0)
```

Variables are referenced with a dollar sign and curly braces, for example,

```
${MYVAR}
```

Options

Options are set using the equals sign. Spaces are ignored. For example

```
myoption = valu
myoption=valu
```

An option can take multiple values. Multiple values are listed within square brackets and are separated by the pipe symbol "|".) For example,

```
myoption = [value1|value2|value3]
```

In this example, *myoption* can be assigned a value of *value1*, *value2* or *value3*.

Objects

Objects are instantiated with the "=>" construct. For example,

```
myobject => some_parameter
```

creates an object named *myobject* with the value of *some_parameter*.

Commands

Configuration commands are keywords and value pairs separated with equals or equals greater than. The asterisk command parser treats the equals and equals greater than the same

```
keyword = value1
keyword2 > value2
```

The Configuration Process

Asterisk switches communications sessions between channels, for example a SIP channel or an IAX channel. You must be familiar with the channels Asterisk supports before attempting any configuration. Refer to the*Asterisk Architecture* chapter or the individual chapters on channel configuration for information about channels.

To configure Asterisk, you must alter the contents of the configuration files listed above. For example, to receive calls, you must first configure the channel that the call will come in on. You must then modify*extensions.conf* to process the incoming calls. You might then wish to modify *voicemail.conf* to provide voicemail for unanswered incoming calls.

If you want to receive calls from an IAX channel, you must change *iax.conf.* Here is a sample IAX configuration. The following entry in *iax.conf* will register your Asterisk server with the IAX server that will be sending the calls. Changing*iax.conf* to include the following entry will register with the remote server found at*iax.url.com.*

```
register > user:passwd@iax.url.com
```

The registration informs the remote server of the location of your Asterisk server. This is how the remote server knows how to send calls to your Asterisk server.

Next, you must configure *extensions.conf* so that the dial plan will correctly process incoming calls.

That is, you must modify *extensions.conf* to process calls that arrive on the IAX channel. The following entry in*extensions.conf* could process calls coming from *iax.url.com*. Don't worry about what this example does exactly, that will be covered in the later chapter on dial plan configuration.

```
[iax-incoming
; This context tells Asterisk what to do wit
; incoming calls from the IAX channe

; You should hear a"congratulations" recording
; on incoming calls
exten > _NXXNXXXXXX,1,Playback(demo-congrats)
exten > h,1,Hangup
exten > i,1,Hangup
exten > t,1,Hangup
```

The registration statement in *iax.conf* informed the remote server of the location of your Asterisk server. You must modify*iax.conf* to indicate what context in the dial plan will process the call. In this example, the context named *ax-incoming* is named. This specifies that a call coming in on the IAX channel will be processed by the*extensions.conf* instructions shown above, that is the iax-incoming context. Note that the remote server must be correctly configured to send calls to the *iaxserver* context specified here in *ax. conf.*

```
[iaxserver
context = iax-incomin
secret=iJKLmNo
auth=md
type=frien
hostgw5.voicepulse.com
```

You could additionally modify *extensions.conf* and *voicemail.conf* to pass any unanswered calls to voice-mail.

Dial Plans

For any enterprise telephony system, a dial plan determines call routing and processing. For example, if a call comes in on aPOTS line, where should that call be directed? If someone doesn't answer their phone, what should be done with the call? Should phones be answered after 5pm?

The file *extensions.conf* is the main Asterisk configuration file. This file contains the Asterisk dial plan. The dial plan controls all Asterisk call switching. The Asterisk dial plan controls the behavior of al connections through Asterisk. The dial plan determines the route a call takes through the interfaces o an Asterisk system. Commands in the*extensions.conf* file route calls based on either the called or caller number.

Sections of extensions.conf

Two section names in *extensions.conf* are reserved, *general* and *globals*. A section with any name other than*general* or *globals* defines an extension context. An extension context is a group of extensions.

[general]

This should always be the first section of *extensions.conf*. This section contains two variables used by Asterisk to control protection of the extensions file.

```
static=yes
writeprotect=n
```

If static is set to no, or doesn't appear in the *extensions.conf* file, the configuration file can be overwritten by the running Asterisk system. If static is set to yes and *writeprotect* is set to no you can use the comman

```
save dialpla
```

from the Asterisk command line interface to save the dial plan in use.

[globals]

This should always be the second section of *extensions.conf*. The globals section of the extensions configuration file contains variables that are available from anywhere within the extensions file. For example,

```
[globals]
CONSOLE = Console/dsp  ; Console Interfac
CONSOLE=Console/dsp    ;  sole interface for dem
;CONSOLE=Zap/
;CONSOLE=Phone/phone
IAXINFO=guest ; IAXtel username/passwor
;IAXINFO=myuser:mypas
TRUNK=Zap/g2 ;Trunk interface
TRUNKMSD=1 ; MSD digits to strip (usually 1 or 0
;TRUNK=IAX2/user:pass@provide
```

Globals are referenced in the dial plan with a dollar sign and then within curly braces

```
${VARIABLE}
```

References to globals can be nested, for example

```
${text${VARIABLE}}
```

Accessing Environment Variables

Operating system environment variables are accessed with this syntax

```
${ENV{VARIABLE}}
```

Extensions

An extension is identified by an alpha-numeric identifier. Extension identifiers can contain numbers, letters, and the special character *****, and **#**. For example, the following entry is for extension 1000.

```
exten > 1000,1,Goto(default,s,1)
```

Some extension names are reserved as shown in the following table.

TABLE: 06-1 Reserved Extension Names		
Character	**Name**	**Usage**
s	Start	A call that does not have digits associated with it, for example a loopstart analog line, begins at the "s" extension.
t	Timeout	When a caller in a voice menu doesn't enter the correct number of digits, the timeout extension is executed. If there is no timeout extension, the caller is disconnected.
T	Absolute Timeout	When a call exceeds the value held in an Absolute Timeout variable.
i	Invalid	Executed when a caller enters an invalid extension.
0 (zero)o	Operator	Executed when a caller presses 0.
h	Hangup	Executed at the end of a call when the caller hangs up or is hung up. Applications executed in this extension cannot access the closed channel. Useful for logging or executing commands.

Patterns

An extension prefixed with the underscore character indicates a pattern match. For example,

```
_NXXXXXX
```

A pattern matching expression can include the following special pattern matching characters.

TABLE: 06-2 Characters Used in Extension Pattern Matches	
Character	**Matches**
N	any digit from two to nine
X	**any digit** from zero to nine
[1235-9]	any digit within the brackets, in this case 1, 2, 3, 5, 6, 7, 8, and 9
. (period)	any one or more characters (positive cloture)

For example, the extension

```
_NXXXXXXX
```

matches a regular seven digit dialed number and the extension

```
_1NXXNXXXXXX
```

matches the character one followed by an area code and then a seven digit phone number.

Ignore Pattern

An ignorepat prevents dialtone from being cancelled when a specified pattern is encountered. A common ignore pattern allows dialtone to be continued after the number nine is dialed.

```
ignorepat > 9
```

Note that SIP phones generate their own dialtone. For a SIP phone, continue dialtone after dialing by reprogramming the phone. Consult the manufacturer's documentation for this.

Applications

Applications, with optional priorities or optional arguments, can be associated with an extension. Each of the available applications is detailed in a later section.

Each extension is defined with one or more lines like

```
exten > ,,,()
```

The components of an extension definition are

```
 - an alphanumeric extension identifier
 - used to determine the execution order
 - the name of an application, e.g. Dial
 - optional arguments for the named application.
```

The dial plan associates one or more applications with an extension. Multiple applications are associated with a single extension by adding additional *exten* lines to the configuration file.

The following example associates two applications, dial and voicemail, with extension 600. Here is an example. In this example, the u specifies the unavailable message and the*b* specifies the busy message.

```
exten => 600,1,Dial,Zap/9|15
exten > 600,2,Voicemail,u600
exten > 600,102,Voicemail,b600
```

Priorities

The priority field specifies the execution order of applications. When a call starts, applications for an extension are executed starting with the lowest priority. Each higher priority application is executed i turn. Applications are run in order of priority until a call ends.

In the example above, the dial application would be executed first before the Voicemail application because the priority of 1 for the dial application is the lowest priority listed for extension 600. When call is made to extension 600 the dial application is run, then the voicemail application is run.

Changing the Execution Order of Applications

Applications can add values to priorities that change the order of execution. These values can cause some lines associated with an extension to be skipped, or change the order of execution.

In the example below, after the Dial application executes either 2 or 102 is executed. That is, after the Dial application runs, one of the two voicemail commands will be selected. The addition of 100 on the third line to the priority of two on the second line determines which of the two commands is executed TheDial command executes one of the two commands, but either command is available after the Dial command executes.

```
exten => 600,1,Dial,Zap/9|15
exten > 600,2,Voicemail,u600
exten > 600,102,Voicemail,b600
```

A *goto* argument can change the order of execution.

```
exten => s,3,Goto(${ARG2},1) ; If they press #, return to start
```

Asterisk applications or AGI scripts can modify priorities and thus the call flow.

Extension Contexts

Contexts are the central building blocks of an Asterisk dial plan. An extension context is a special named section holding commands for a collection of extensions. Any section that is not named*general* or*globals* in *extensions.conf* is a named context.

Asterisk contexts divide dialing plans into logical units. Each context interprets numbers differently and has its own security model.

Most users are given access to the default context. Trusted users could be given access to a context with more capabilities

Contexts can contain multiple commands for each extension, one command for each processing step for the extension. Commands are executed in order starting with the lowest listed priority. For example,

```
exten => _9011.,1,Dial(${TRUNK}/${EXTEN:${TRUNKMSD}})
exten > _9011.,2,Congestion
```

runs two commands, Dial and Congestion.

Two formats are provided for arguments.

```
exten => someexten,priority,application(arg1,arg2,...)
exten > someexten,priority,application,arg1|arg2...
```

The first format is preferred as it is the most commonly used.

One context can be included within another context. The following example includes the daytime context.

```
include => daytime|9:00-17:00|mon-fri|*|*
```

The following figure shows two contexts named Sales and Support.

Context	Sales
Extension	Description
100	Jim
101	Janet
102	Jose
103	Corrinda

Context	Support
Extension	Description
201	Steve
202	Miriam
203	Sally
204	Jacques

Figure: 06-2 Contexts - Sales and Support

One context can include another context. In the next figure, the Dial Out context includes the Sales context. This permits the extensions in the Sales context to dial out. This prevents the extensions i the Support context from dialing out. The inclusion of one context in another can be restricted by dat and time. For example, the Sales context could be included in theDial Out context only during business hours. This would prevent anyone with an extension in the Sales context from dialing out after hours or on weekends.

Extension contexts can help manage the security of an Asterisk installation. Access to services or interfaces can be restricted to an extension context or by date and time. This is described further in the later sections on Asterisk security.

Context	Dial Out
Extension	Description
9	Outside Line

Context	Sales
Extension	Description
100	Jim
101	Janet
102	Jose
103	Corrinda

Context	Support
Extension	Description
201	Steve
202	Miriam
203	Sally
204	Jacques

Figure: 06-3 Contexts - Including

An extension can link to a context. In the following figure, a new context named Main is added to the last example. Extension 100 in the Main context links to the Sales context. In this example, incomin calls would be directed to the Main context. This would allow someone dialing in to press 100 and b redirected to the sales department or 201 to be redirected to the Support department. By directin incoming calls to the Main context, incoming callers would be prevented from pressing 9 and reachin an outside line.

The Interactive Voice Response (IVR) facilities of Asterisk can provide voice prompts for each of the contexts. An outside caller reaching the Main context could be presented with a message saying,"Press 100 for sales or 201 for support"

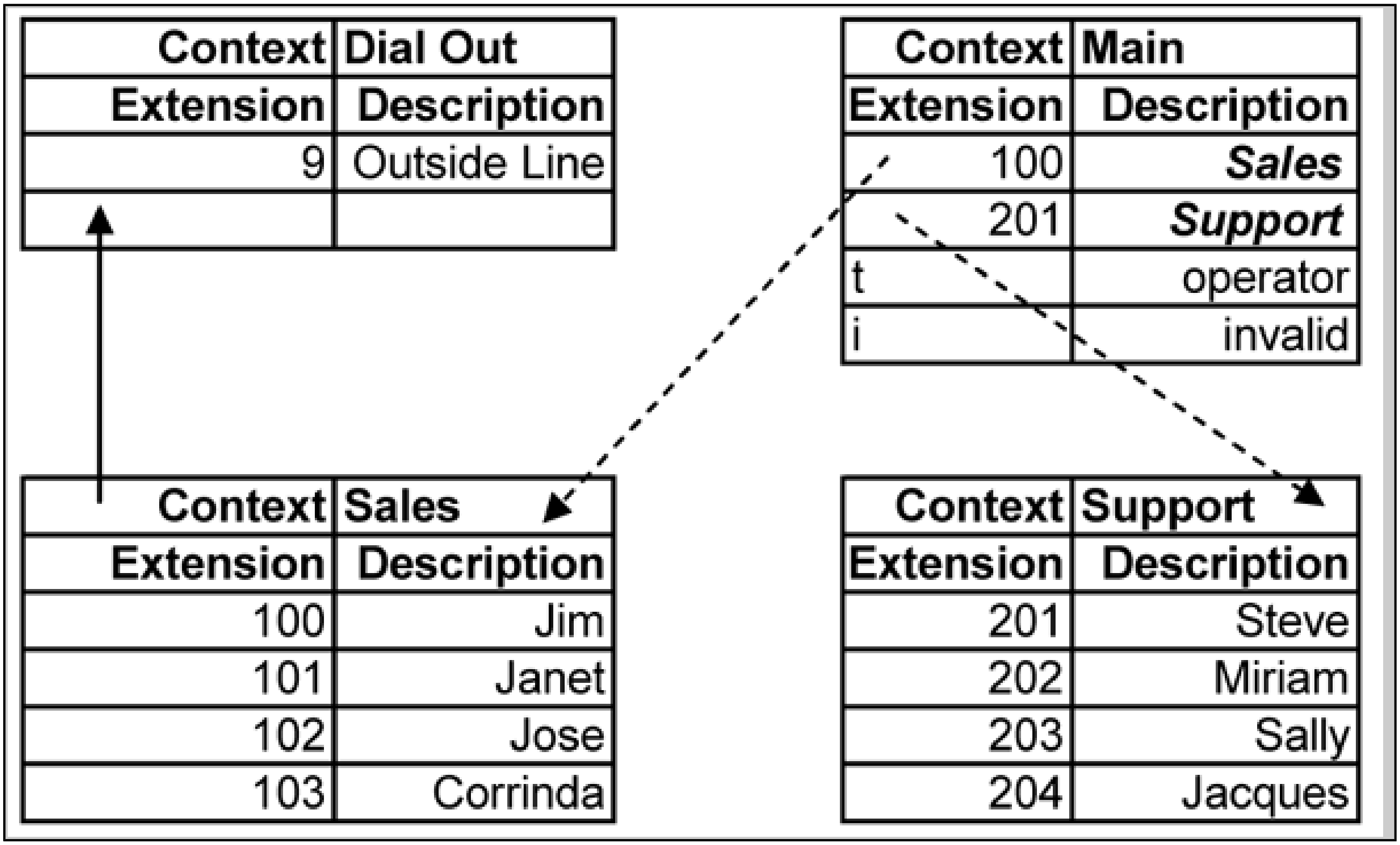


Figure: **06-4** Contexts - Linking

Extensions can be of any length and can be included in any other context. In the example above, the extension 201 has been reused in two different contexts, the Main context and the Support context Including the Sales context in the Main context would allow callers to select the extension of someon in Sales from the main menu. This is shown below.

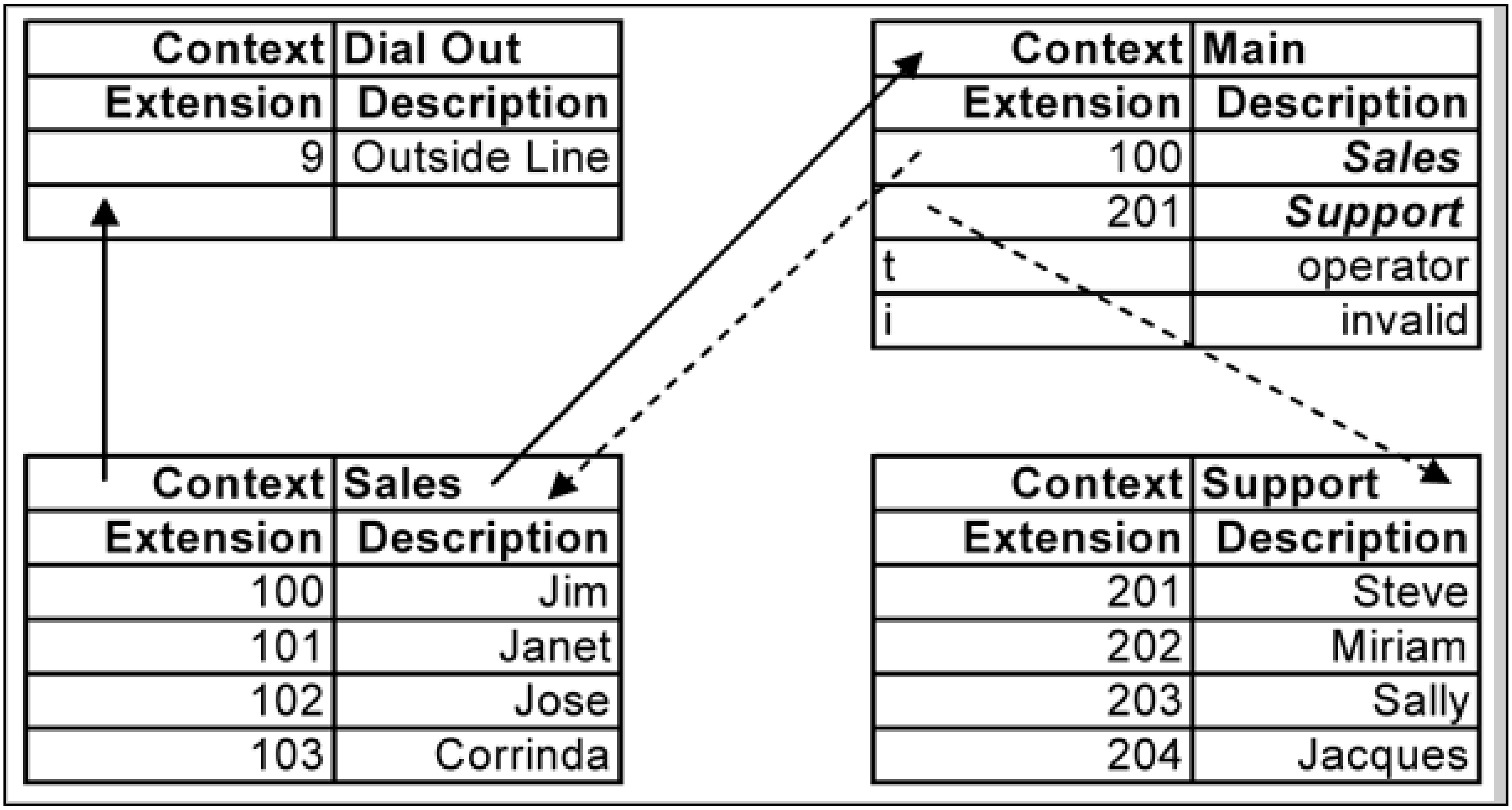


Figure: 06-5 Contexts - Including

A context can include the contents of another context with an *include* statement. Here is an example.

```
[trunklocal]
exten > 4035,1,Dial(SIP/CAB,20)
include > trunktollfree
```

Ordering in Contexts

There is no implied order for the extensions in a context. Here is an example from a *sip.conf* configuration file with two extensions in a context named general

```
[general]
port = 5060 ; TheTCP/IP port for SIP communications

[4035]
type=friend           ; This device takes and makes calls
username=4035
secret=cisco
context=from-sip
ca"Bill" <415551212>
qualify=100
host=dynamic          ; This host is not on the same IP addr every tim
canreinvite=n
mailbox=4035           ; Activate the message waiting lightfor message
defaultip192.160.0.12

[4009]
type=friend           ; This device takes and makes calls
username=4009
secret=cisco
context=from-sip
ca"Paul" <4155551212>
qualify=100
```

```
host=dynamic  ; This host is not on the same IP addr every tim
canreinvite=n
mailbox=4009  ; Activate the message waiting light for message
defaultip192.168.0.12
```

There is no particular order to these two extensions within the context. When Asterisk starts, the extensions are loaded into memory in numeric order. In this example, extension 4009 will appear i memory ahead of extension 4035

Changing the Execution Order Within Contexts

Asterisk parses a context before it parses any includes within the context. Because of this, the include statement can change the execution order of extensions within contexts. Here is an example with fou contexts. This example assures that the contexts are executed in the order shown,*ex*, *ex1*, *ex2* and then *ex3*.

```
[ex
include > ex1
include > ex2
include > ex3
exten > h,1,Hangup

[ex1]
exte> 1234,1,Dial(SIP/1234)
exten > 9992,1,Dial(SIP/9992)

[ex2]
exte> _9.,1,Dial(Zap/1/${EXTEN})

[ex3]
exte> _.,1,Playback(sorry-no-match)
exten > _.,2,Hangup
```

Authentication, Multi-hosting, Callback and External References

Contexts can provide authentication services. For example, a user could be required to have a passcode to move from one context to another.

Contexts can easily support PBX multi-hosting. For example, if two companies were sharing a single Asterisk server, incoming calls could be routed to the dial plan for the correct company based on th incoming DID number. Here is an example from Eric Wieling.

```
[zap-incoming
; DIDs for Microsof
exten > _2126661XXX,1,GoTo(microsoft,${EXTEN},1)
exten > 5046662000,1,GoTo(microsoft,${EXTEN},1)
exten > 5046662500,1,GoTo(microsoft,${EXTEN},1)

; DIDs for Sun Microsystem
exten > _6165551XXX,1,GoTo(sun,${EXTEN},1)
exten > 2285552000,1,GoTo(sun,${EXTEN},1) exten =>
  2285552500,1,GoTo(sun,${EXTEN},1
[microsoft
exten > 2126661000,1,Dial(SIP/1000)
exten > 2126661001,1,Dial(SIP/1002)
exten > 2126661002,1,Dial(SIP/1002)
exten > 5046662000,1,Dial(SIP/2000)
exten > 5046662500,1,Dial(SIP/2500)
[sun
```

Extension contexts can be combined with external scripts and the Asterisk application *app_qcall* to implement callback services. Asterisk could prompt an incoming caller for a number and then initiate call back to the supplied number.

Since a context can reference an external Asterisk system, the external system can add to the functions of the local system. Using IAX, the dial plan of a remote server can be accessed. The local switch ca reference the remote dial plan. This allows a complex dial plan for multiple servers to be centralized o a single server.

Referencing Interfaces in extensions.conf

As described in the earlier chapter on Asterisk architecture, an Asterisk interface is specified as:

```
/
```

Here is an example in *extensions.conf* that uses the *Dial* application to associate extension 4035 with SIP line F8. If this entry is included in the dial plan, calls directed to extension 4035 will be switched toSIP line F8.

```
exten => 4035,1,Dial(SIP/F8,20)
```

In this example, Asterisk extension 1010 dials SIP the client SIP/OEJ. SIP/OEJ is on the local asterisk server.

```
exten=> 1010,1, Dial(SIP/oej,20,tr)
```

Next, extension 1015 dials extension 10000 on the remote SIP server *fwd.pulver.com*. Pulver could be a SIP server or a SIP Proxy.

```
exten=> 1015,1, Dial(SIP/10000@fwd.pulver.com:5060)
```

Macros

Groups of commands can be reused by combining them into a macro. A macro accepts arguments. A macro is named with the prefix*macro* in the context name. The macro shown here rings an extension for some number of seconds before forwarding the call to a different extension. Note the use of th variable*s* instead of an extension number. Arguments are specified with the syntax *$(ARG)*

```
[macro-stdexten]

;"standard extension" macro for single-stage ringing.

;  Calls an extension for ${ARG2} seconds. If that fails
;  goes to voicemail for extension ${ARG1}. Rings th
;  devices listed in ${ARG3}

;   ${ARG1} - voicemail contex
;   ${ARG2} - Extension for voicemail and other use
;   ${ARG3} - Time to rin
;   ${ARG4} - Device(s) to rin

exten> s,1,Dial(${ARG4},${ARG3}) ; Ring the interface
exten > s,2,Voicemail2(u${ARG2}@${ARG1}); If unavailable, send to vm as unavail
exten > s,3,Goto(${ARG2},1) ; If they press #, return to start
exten > s,102,Voicemail2(b${ARG2}@${ARG1}) ; If busy, send to vm w/busy announce
exten > s,103,Goto(${ARG2},1) ; If they press #, return to start
```

Here is an example of this macro in use. The first argument is the name of the macro to run, the remaining are arguments to the macro.

```
exten > *19355,1,Macro(stdexten,default,355,12,355)
```

This example uses a macro to create extensions. The *u* and the *b* choose between unavailable and busy voicemail messages.

```
[globals
PHONE1=Zap/
PHONE2SIP/6002

[macro-oneline]
exte> s,1,Dial(${ARG1},20,t)
exten > s,2,Voicemail(u${MACRO_EXTEN})
exten > s,3,Hangup
exten > s,102,Voicemail(b${MACRO_EXTEN})
exten > s,103,Hangup

[local]
exte> 6601,1,Macro(oneline,${PHONE1})
exten > 6602,1,Macro(oneline,${PHONE2})
```

Applications

The following applications are available for use in *extensions.conf*. To see a list of applications, from the Asterisk command prompt typ

```
show applications
```

As of the time of writing of this book, the available applications are, in alphabetical order

```
AbsoluteTimeout: Set absolute maximum time of call
AddQueueMember: Dynamically adds queue member
ADSIProg: Load Asterisk ADSI Scripts into phon
AgentCallbackLogin: Call agent callback logi
AgentLogin: Call agent logi
AgentMonitorOutgoing: Monitor Outgoing Agent Calls (0.7.3
AGI: Executes an AGI compliant applicatio
Answer: Answer a channel if ringin
AppendCDRUserField: Append data to the CDR user fiel
Authenticate: Authenticate a use
BackGround: Play a file while awaiting extensio
Busy: Indicate busy condition and sto
CallingPres: Change the presentation for the calleri
ChangeMonitor: Change monitoring filename of a channe
ChanIsAvail: Check if channel is availabl
Congestion: Indicate congestion and sto
Cut: String handling functio
DateTime: Say the date and tim
DBdel: Delete a key from the databas
DBdeltree: Delete a family or keytree from the databas
DBget: Retrieve a value from the databas
DBput: Store a value in the databas
Dial: Place an call and connect to the current channel
DigitTimeout: Set maximum timeout between digit
Directory: Provide directory of voicemail extension
DISA: DISA (Direct Inward SystemAccess)
EAGI: Executes an AGI compliant applicatio
Echo: Echo audio read back to the use
EnumLookup: Lookup number in ENU
Eval: Evaluate arguments before calling applicatio
Festival: Say text to the use
Flash: Flashes a ZapTrunk
GetCPEID: Get ADSI CPE I
Goto: Goto a particular priority, extension, or contex
```

```
GotoIf: Conditional got
GotoIfTime: Conditional goto on current tim
Hangup: Unconditional hangu
HasNewVoicemail: Conditionally branches to priority + 10
ICES: Streaming calls to theInternet
LookupBlacklist: Look up Caller*ID name/number from blacklist databas
LookupCIDName: Look up CallerID Name from local databas
Macro: Macro Implementatio
MeetMe: Simple MeetMe conference bridg
MeetMeCount: MeetMe participant coun
Milliwatt: Generate a Constant 1000Hz tone at 0dbm (mu-law
Monitor: Monitor a channe
MP3Player: Play an MP3? file or strea
MusicOnHold: Play Music On Hold indefinitel
NBScat?: Play an NBS local strea
NoCDR: Make sure asterisk doesn't save CDR for a certain cal
NoOp: No operatio
ParkAndAnnounce: Park and Announc
ParkedCall: Answer a parked cal
Playback: Play a fil
Playtones: Play a tone lis
Prefix: Prepend leading digit
PrivacyManager: Require phone number to be entered, if no CallerID? sen
Queue: Queue a call for a call queu
Random: Make a random jump in your dial pla
Read: Read a variabl
Record: Record to a fil
RemoveQueueMember: Dynamically removes queue member
ResetCDR: Reset CDR dat
ResponseTimeout: Set maximum timeout awaiting respons
Ringing: Indicate ringing ton
SayDigits: Say Digit
SayNumber: Say Numbe
SayUnixTime: Say Time in a number of format
SendDTMF: Sends arbitrary DTMF digit
SendImage: Send an image fil
SendURL: Send a URL
SetAccount: Sets account cod
SetCallerID: Set CallerID
SetCDRUserField: Set CDR User Field. See Billing
SetCIDName: Set CallerID Name
SetGlobalVar: Set variable to valu
SetLanguage: Sets user languag
SetMusicOnHold: Set default Music On Hold clas
SetVar: Set variable to valu
SIPdtmfMode: Change DTMF mode duringSIP call
SMS: Send and receive SMS (short messaging service) - not yet in CVS!
SoftHangup: Soft Hangup Applicatio
StopMonitor: Stop monitoring a channe
StopPlaytones: Stop playing a tone lis
StripLSD: Strip Least Significant Digit
StripMSD: Strip leading digit
SubString: Save substring digits in a given variabl
Suffix: Append trailing digit
System: Execute a system comman
Transfer: Transfer caller to remote extensio
VoiceMail: Leave a voicemail messag
VoiceMail2: (deprecated) Leave a voicemail messag
VoiceMailMain: Enter voicemail syste
VoiceMailMain2: (deprecated) Enter voicemail syste
Wait: Waits for some tim
WaitForRing: Wait for Ring Applicatio
WaitMusicOnHold: Wait, playing Music On Hol
Zapateller: Block telemarketers with SI
ZapBarge: Barge in (monitor) Zap channe
ZapRAS: Executes ZaptelISDN RAS application
```

Here are the the same applications listed by group.

General commands

```
ADSIProg: Load Asterisk ADSI Scripts into phon
Authenticate: Authenticate a use
ChangeMonitor: Change monitoring filename of a channe
GetCPEID: Get ADSI CPE I
SendDTMF: Sends arbitrary DTMF digit
SendImage: Send an image fil
SendURL: Send a URL
System: Execute a system comman
Transfer: Transfercaller to remote extension
Wait: Waits for some tim
WaitForRing: Wait for Ring Applicatio
WaitMusicOnHold: Wait, playing Music On Hol
Billin
NoCDR: Make sure asterisk doesn't save CDR for a certain cal
ResetCDR: Reset CDR dat
SetAccount: Sets account cod
Asterisk cmd SetCDRUserField: Set CDR User fiel
Asterisk cmd AppendCDRUserField: Append data to CDR User fiel
```

Call management (hangup, answer, dial, etc)

```
Answer: Answer a channel if ringin
Busy: Indicate busy condition and sto
Congestion: Indicate congestion and sto
Dial: Place an call and connect to the current channel
DISA: DISA (Direct Inward SystemAccess)
Hangup: Unconditional hangu
Caller presentation (ID, Name etc
CallingPres: Change the presentation for the calleri
LookupBlacklist: Look up Caller*ID name/number from blacklist databas
LookupCIDName: Look up CallerID Name from local databas
PrivacyManager: Require phone number to be entered, if no CallerID? sen
Ringing: Indicate ringing ton
SetCallerID: Set CallerID
SetCIDName: Set CallerID Name
SoftHangup: Request hangup on another channe
Zapateller: Block telemarketers with SI
```

Database handling

```
DBdel: Delete a key from the databas
DBdeltree: Delete a family or keytree from the databas
DBget: Retrieve a value from the databas
DBput: Store a value in the databas
Extension logic - strings, application integratio
AbsoluteTimeout: Set absolute maximum time of cal
AGI: Executes an AGI compliant applicatio
Cut: String handling functio
DigitTimeout: Set maximum timeout between digit
EAGI: Executes an AGI compliant applicatio
EnumLookup: Lookup number in ENU
Goto: Goto a particular priority, extension, or contex
GotoIf: Conditional got
GotoIfTime: Conditional goto on current tim
Macro: Macro Implementatio
NoOp: No operatio
Prefix: Prepend leading digits (Obsolete
Random: Make a random jump in your dial pla
Read: Read a variable with DTM
ResponseTimeout: Set maximum timeout awaiting respons
SetGlobalVar: Set variable to valu
SetVar: Set variable to valu
StripLSD: Strip trailing digit
StripMSD: Strip leading digits (Obsolete
SubString: Save substring digits in a given variable (Obsolete
Suffix: Append trailing digits (Obsolete
Sounds - background, musiconhold et
BackGround: Play a file while awaiting extensio
DateTime: Say the date and tim
```

```
Echo: Echo audio read back to the use
Festival: Say text to the use
Milliwatt: Generate a Constant 1000Hz tone at 0dbm (mu-law
Monitor: Monitor a channe
MP3Player: Play an MP3? file or strea
MusicOnHold: Play Music On Hold indefinitel
Playback: Play a fil
Playtones: Play a tone lis
Record: Record to a fil
SayDigits: Say Digit
SayNumber: Say Numbe
SayUnixTime: Say Time in a number of format
SetLanguage: Sets user languag
SetMusicOnHold: Set default Music On Hold clas
StopMonitor: Stop monitoring a channe
StopPlaytones: Stop playing a tone lis
SIP commands
SIPdtmfMode: Change DTMF mode duringSIP call
```

ZAP commands

```
ChanIsAvail: Check if channel is availabl
Flash: Flashes a ZapTrunk
ZapBarge: Barge in (monitor) Zap channe
ZapRAS: Executes ZaptelISDN RAS application
```

Voicemail and conferencing

```
Directory: Provide directory of voicemail extension
HasNewVoicemail: Conditionally branches to priority + 10
MeetMe: Simple MeetMe conference bridg
MeetMeCount: MeetMe participant coun
VoiceMail: Leave a voicemail messag
VoiceMailMain: Enter voicemail syste
deprecated: VoiceMail2: Leave a voicemail messag
deprecated: VoiceMailMain2: Enter voicemail syste
VoiceMail, version 1 is now replaced with VoiceMail version 2 so al
  voicemail commands leads to voicemail version 2
```

Queue and ACD management

```
AddQueueMember: Dynamically adds queue member
AgentCallbackLogin: Call agent callback logi
AgentLogin: Call agent logi
ParkAndAnnounce: Park and Announc
ParkedCall: Answer a parked cal
Queue: Queue a call for a call queu
RemoveQueueMember: Dynamically removes queue member
```

External applications (not in the CVS)

```
Asterisk app_dbodbc: dial plan modifiers using unixODB
Asterisk cmd DynExtenDB: Store extensions in databas
app Prepaid: Designed for Postgre
```

Enhancements to Extension Logic

The following enhancements are provided for extensions within *extensions.conf*.

QUOTING

```
exten > s,5,BackGround,blabla
```

The parameter *blabla* can be quoted, for example *"blabla"*. A comma does not terminate a quoted parameter.

Characters special to variable substitution and expression evaluation can be escaped. For example, to use a literal*$* in the string *$1231*, escape it with a preceding \. The special characters *[ ] $ "* \ must be escaped. To escape\ , use a double back-slash \\.

VARIABLES

Variable names are arbitrary strings. To set a variable to a particular value,

```
exten => 1,2,SetVar,varname=value
```

To substitute the value of a variable use *${variablename}*. For example, to stringwise append *$lala* to *$blabla* and store result in *$koko*,

```
exten > 1,2,SetVar,koko=${blabla}${lala}
```

The following are special reserved identifiers

```
${CALLERID} Caller ID
${CALLERIDNAME}Caller ID Name only
${CALLERIDNUMCaller ID Number only
${EXTEN} Current extensio
${CONTEXT} Current contex
${PRIORITY} Current priorit
${CHANNEL} Current channel nam
${ENV(VAR)} Environmental variable VA
${LEN(VAR)} String length of VAR (integer
${EPOCH} Current unix style epoc
${DATETIME} Current date time in the format: YYYY-MM-DD_HH:MM:S
${TIMESTAMP} Current date time in the format: YYYYMMDD-HHMMS
${UNIQUEID} Current call unique identifie
${DNID}  Dialed Number Identifie
${RDNIS}  RedirectedDial Number ID Service
${HANGUPCAUSE}  Hangup cause on lastPRI hangup
${ACCOUNTCODE}  Account code (if specified
${SIPDOMAIN} SIP destination domain of an inbound call (if appropriate)
```

References can be by value or by name. To refer to a variable by its name, for example as an argument to a function that requires a variable, just write the name. To refer to a variable value, enclose it insid *${ }*. For example, *SetVar* takes a variable name as the first argument before the equals sign.

```
exten => 1,2,SetVar,koko=lala
exten > 1,3,SetVar,${koko}=blabla
```

The first example above stores in *koko* the value *lala*. The second example stores in *lala* the value *blabla*. The variable*${koko}* is replaced with the value of the variable *koko*.

EXPRESSIONS

Everything inside brackets and prefixed by a *$* is considered as an expression and is evaluated.

```
$[this]
```

Evaluation is similar to variable substitution. The expression, including the square brackets, is replaced by the result of the expression evaluation. The arguments and operands of the expression**must be** separated with spaces. Don't leave any spaces between opening and closing square brackets and the first and last arguments). Parentheses are used for grouping.

For example, after the sequence

```
exten > 1,1,SetVar,"lala=$[1 + 2]";
exten > 1,2,SetVar,"koko=$[2 * ${lala}]";
```

the value of variable koko is six.

Operators are listed below in order of increasing precedence. Operators with equal precedence are grouped withi *{ }* symbols.

```
expr1 | expr2
```

Return the evaluation of expr1 if it is not an empty string or zero, otherwise, returns the evaluation of expr2

```
expr1 & expr2
```

Return the evaluation of expr1 if neither expression evaluates to an empty string or zero; otherwise, returns zero

```
expr1 {=, >, >=, <, <=, !=} expr2
```

Return the results of integer comparison if both arguments are integers; otherwise, returns the results of string comparison using the locale-specific collation sequence. The result of each comparison is 1 i the specified relation is true, or 0 if the relation is false

```
expr1 {+, -} expr2
```

Return the results of addition or subtraction of integer-valued arguments.

```
expr1 {*, /, %} expr2
```

Return the results of multiplication, integer division, or remainder of integer-valued arguments.

```
expr1 : expr2
```

The *:* operator matches expr1 against expr2, which must be a regular expression. The regular expression is anchored to the beginning of the string with an implicit ^.

If the match succeeds and the pattern contains at least one regular expression sub expression *\(...\)*, the string corresponding to*\1* is returned; otherwise the matching operator returns the number of characters matched. If the match fails and the pattern contains a regular expression sub expression the null string is returned; otherwise 0

GOTO

The order of execution can be changed with a *goto* statement. The *goto* can change execution to any context, extension or priority. The return from the*goto* is always zero, even if the *goto* fails. The syntax of the goto statement is

```
goto([[context|]extension|]priority
```

You can specify a priority, an extension and a priority, or a context, extension and priority.

```
Goto(context,extension,priority)
Goto(extension,priority
```

```
Goto(priority
```

Here is an example.

```
exten => 1,1,Goto,sales
```

The special extension *BYEXTENSION* allows a transfer to a different context without having to specify the extension. That is, the current extension will be used in the new context.

Conditionals

There is one conditional operator - the conditional gotoif,

```
exten => 1,2,gotoif,condition?label1:label2
```

If *condition* is true go to *label1*, else go to *label2*. Labels are interpreted the same as in the normal goto command. The*condition* is just a string. If the string is empty or zero, the condition is considered to be false, if it's anything else, the condition is true. This is used with the expression syntax described above for example,

```
exten => 1,2,gotoif,$[${CALLERID} = 123456]?2|1:3|1
```

Examples

```
exten > s,2,SetVar,"vara=1"
exten > s,3,SetVar,"varb=$[${vara} + 2]"
exten > s,4,SetVar,"varc=$[${varb} * 2]"
exten > s,5,GotoIf,"$[${varc} = 6]?99|1:s|6";
```

IGNOREPAT

Pressing a dial pad key at a telephone often stops the dialtone. Use the *ignorepat* command to continue dialtone after a key is pressed. Note that the*ignorepat* command does not apply to SIP phones as a SIP phone generates its own dialtone. You should be able to program mostSIP phones to continue dial-tone during dialing.

```
ignorepat => 9
```

Commands

Here are some examples of commands that are available for use in *extensions.conf*.

Answer

```
exten > s,2,Answer ; Answer the line
```

BackGround

```
exten > s,5,BackGround(demo-congrats); Play a congratulatory message
```

Congestion

```
congestion = tonelis
```

The set of tones played when there is congestion on the network

Dial

The dial command sends a call out on one or more channels. When one of the dialed channels picks up the call, the dial command will bridge the two channels. The dial command can answer a call fro an originating channel

If there is no answer, and the calling party does not hang-up, only a time-out will top the dial command. If a time-out is not specified, the dial application will wait indefinitely until either one of the called channels answers, the user hangs up, or all channels return busy or error

They syntax for the dial command is

```
Dial(Technology/resource&Technology2/resource2...[|timeout]
  [|options][|URL]
```

The option string for the dial command may contain zero or more of the following characters:

```
't' â€” allow the called user to transfer the calling user
'T'â€” allow the calling user to transfer the call.
'r'â€” sound ringing to the calling party,
     pass no audio until answered
'm'â€” provides hold music to the calling party until answered.
'H'â€” allow caller to hang up by hitting *.
'C'â€” reset the call detail record for this call.
'P(x)'â€” privacy mode, using 'x' as database if provided.
'g'â€” continues in context if the destination channel hangs up
'A(x)'â€” play an announcement to the called party, using the
       sound file named
'S(x)'â€” hang-up the call x seconds AFTER the called party
       answer
'D([digits])'â€” allow post connect dtmf stream. Sends the DTMF digit
            string after called party has answered but before th
            w=500ms bridge paus
```

A dialed call may be transferred. A dialed call may be parked for later pickup.

The optional url argument is only sent on channels that will support the transmission of a URL.

The most common use of *dial* connects a call from an extension to an interface. Here is an example that switches a call from extension 100 to Zap channel one and dials for twenty seconds

```
[dial
exten > 100,1,Dial(Zap/1,20)
```

Here is another example for dialing out,

This example allows the user to dial nine before dialing an outside number. The call is sent out over

```
${TRUNK2
```

The *exten* variable contains the extension number. The following in *extensions.conf* will say "ninety-seven" when a caller dials extension 97.

```
exten => _9NXXXXXX,1,Dial(${TRUNK2}/${EXTEN:1})
```

Here is another example that repeats a number.

```
exten => 97,1,SayNumber(${EXTEN})
```

The *exten* variable serves a different purpose with the *dial* command than with other commands. When dialing, the*exten* variable holds the digits the user has selected on the keypad. Here is an example.

```
exten => _9NXXXXXXX,1,Dial(Zap/g2/$(EXTEN))
```

You can strip leading digits off the number to be dialed. The number after the colon specifies how many leading digits are stripped from the number before it is dialed. Note that the nine in the example above is strippe off of the number before it is dialed by specifyin

```
exten > _9NXXXXXXX,1,Dial(Zap/g2/$(EXTEN):1)
```

Here is example configuration for outbound dialing. First, outbound dialing is defined for local calls. Any call started by dialing 9 is defined as a local call. Emergency 911 calling is supported. The dia command routes these calls out over the Zap group two interface.

```
[directdial]
ignorepat > 9
exten > 9,1,Dial(Zap/g2/)
exten > 9,2,Congestion

[local]
ignorepa> 9
exten > _9NXXXXXXX,1,Dial(Zap/g2/$(EXTEN:1))
exten > _9NXXXXXXX,2,Congestion
include > default

[longdistance]
ignorepa> 9
exten > _91NXXNXXXXXX,1,Dial(Zap/g2/$(EXTEN:1))
exten > _91NXXNXXXXXX,2,Congestion
include > local

[international]
ignorepa> 9
exten > _9011.,1,Dial(Zap/g2/$EXTEN:1))
exten > _9011.,2,Congestion
include > longdistance
```

The local context uses pattern matching. The ignorepat command causes the number nine to be ignored when dialed. The underscore character in the dial string indicates a pattern is to be matched This pattern matches the user dialing a nine followed by a one. The N matches any number from on to nine. An X matches any number from zero to nine. This can now be easily seen to match a loca dialed number. The dialed number will be tried by dialing out on any Zap g2 (group two) channel. I the call cannot be dialed out on the Zap interface, the caller is directed to the congestion tone.

Note the local context includes the default context, the long distance context includes the local context, and the international context includes the long distance context.

This example creates four contexts. Each context has a different access level to the PSTN. First, dialing nine connects the caller to a channel for an outside line.

The ignorepat command instructs Asterisk not to stop dialtone after the nine is dialed. This makes sure the user will still hear dialtone after dialing nine.

The local context can only dial a seven digit number. The long distance context permits

1+ dialing. The international contexts provides for dialing an international access number, which starts with 011.

The following example dials out to Voicepulse, the SIP and IAX provider.

```
exten => _1NXXNXXXXXX,1,Dial,IAX2/baV36QYm51@voicepulse/${EXTEN}
```

Phones may be missing, they can be turned off or disconnected from the network. Asterisk treats a missing phone as*busy*, not as *unavailable*. Asterisk uses the status *unavailable* when a phone remains unanswered.

When interacting with a remote system, the remote system may prompt to press the # key to continue. To keep the local Asterisk system from capturing the # and executing a transfer, don't use a T or t in the option for an outbound dial string

```
exten => 91xxxxxxxxx,1,Dial(H.323/${Exten:[EMAIL PROTECTED])
exten > 1236,1,Dial(Console/dsp); Ring forever
```

ZAP dialing

Zaptel dialing uses the Zapata chan_zap analog card channel driver. The syntax for a Zaptel dialing string is

The syntax is

```
Zap/group|port|span-port/extensio
```

Here are some examples of dialing with Zap.

```
Zap/g1/12394      : dial 12394 on first available channel on group1
Zap/g1/WW12394    : Wait 1 second before dialing 1239
                  ; on first available channel on group
Zap/1-1/12394     : dial 12394 on span 1, port
Zap/1/12394       : dial 12394 on port
```

Note that the special dial modifier *c* allows for clear channel connections between PRI ports Adding W to the number adds a 0.5 second pause. This causes a wait for dial tone before sending digits

You could keep your user list in an SQL database. Look at the code in the chan_iax2.c source file for further information.

You can change the ringing on zap channels. Here is an example.

```
Dial(Zap/3r2,,r)
```

The first *r2* is an option to the Zaptel channel driver, telling it that you want distinctive ring 2, while the second r indicates to dial that you want ringing to be immediately indicated to the caller.

The available distinctive ringing choices are

```
1: Quick chirp followed by normal rin
2: British style rin
3: Three short burst
4: Long rin
```

Simultaneous Calling on Multiple Interfaces

When using a dial group, the dial command finds one of the group that is not busy and dials it. To ring multiple phones (extensions) simultaneously, each extension must be included in the dial comman and separated with an ampersand,*&.* This example will dial the SIP phone at 192.168.50.188 and the ZAP phone at the same time.

```
exten => 353,1,Dial(SIP/192.168.50.188&Zap/10,18)
```

This example uses two Asterisk features, Caller*ID matching and simultaneous calling on multiple interfaces

```
exten => 100/2565551212,1,Congestion
exten > 100,1,Dial,Zap/9&IAX/paul/s|15
exten > 100,2,Voicemail,u600
exten > 100,102,Voicemail,b600
```

If the incoming caller has the Caller*ID of 256-555-1212, they are immediately routed to a congestion tone. This makes it sound to the caller that the number they called is wrong or inoperative. Otherwise theDial application calls both Zap/9 and another remote IAX host "marko" at the same time. If there is no answer, the call is switched to voicemail where they get the"unavailable" message. If both interfaces are busy, the call is switched to voicemail where they get the "busy" message.

Here is another example that rings several extensions at the same time as suggested by Chris Hariga.

```
exten =>s,2,Dial(SIP/paul&SIP/pauloffice&SIP/jerry&SIP/jerryhome&SIP/
 sa&SIP/xten)
```

Automated Call Distribution

Call distribution can be automated. For example, take a sales department where the manager wants all the sales people to participate equally in incoming calls. Automated call distribution can randoml assign the next incoming call to a sales extension.

DigitTimeout

```
exten > s,3,DigitTimeout,5 ; Set Digit Timeout to 5 seconds
```

Echo

```
exten > 600,2,Echo ; Do the echo test
```

Hangup

```
exten > #,2,Hangup  ; Hang Up
```

Macro

```
exten > 1234,2,Macro(stdexten,1234,${CONSOLE})
```

MeetMe

```
exten > 8600,1,Meetme,1234
```

Playback

```
exten > 1234,1,Playback(transfer,skip) ; "Please hold while..."
```

ResponseTimeout

```
exten > s,4,ResponseTimeout,10; Set Response Timeout to 10 seconds
```

Ringing

Plays a ringing signal for the calling party.

```
exten > s,1,Ringing
```

Here is an example.

```
exten > _5551212,1,Answer
exten > _5551212,2,Ringing
exten > _5551212,3,Dial(SIP/6710,12,tr)
```

SetLanguage

```
exten > 3,1,SetLanguage(fr); Set language to french
```

Voicemail

```
exten > 1235,1,Voicemail(u1234)  ; Right to voicemail extension 1234
```

Voicemail is covered in greater detail in a following chapter.

In the next example, if there is no answer within 20 seconds, the call is sent to voicemail.

The following dial plan implements a simple extension with voicemail. The extension is numbered 600. Three commands are shown. The commands are executed in order of priority. The arguments 1 2 and 102 prioritize the commands.

```
exten => 600,1,Dial,Zap/9|15
exten > 600,2,Voicemail,u600
exten > 600,102,Voicemail,b600
```

Note there are two priorities for the voicemail transfer. If the call is unanswered, the second command for message u600 is executed. The u600 message is the unanswered message. If the line is busy, th third line for message b600 is executed. The b600 message is the busy message.

When an incoming call is directed to extension 600, Asterisk switches the call to the Zap/9 interface (channel 9 of the Zaptel interface) for up to fifteen seconds. If the call is unanswered, it is forwarded t voicemail.

The Dial application provides a special capability. It provides separate operations for busy or unanswered extensions.

The Dial command can determine which command should execute next. Adding 100 to the priority of the secondVoicemail command indicates a busy referral instead of an unanswered referral. Different voicemail recordings can be played for a busy and unanswered calls. In this example the priority of and 102 are equivalent priorities, but theDial application recognizes the difference between the two commands.

If there is no answer, the Dial application redirects the call to voicemail. The "u" in the u600 argument indicates a referral to"unavailable" voicemail. The "b" in the b600 argument indicates a referral to the busy voicemail message.

Wait

```
exten > s,1,Wait,1  ; Wait one second
```

A Simple Call Queue

This example demonstrates a simple call queue.

```
exten => 600,1,Dial,Zap/9|15
exten > 600,2,Voicemail,u600
exten > 600,102,WaitMusicOnHold,5
exten > 600,103,Goto,1
```

This dial plan tries to switch the incoming call to the Zap/9 interface for up to 15 seconds. If the extension remains unanswered, the calling party hears music on hold for five seconds. They are the returned to the first extension. This puts the calling party on hold until the called party becomes available. The caller hears music on hold as they are waiting.

Operator Extension

The following dial plan creates an operator extension.

```
exten => 0,1,Dial,Zap/9|15
exten > 0,2,Dial,Zap/10&Zap/11&Zap/12|15
exten > 0,3,Playback,companymailbox
exten > 0,4,Voicemail,0
exten > 0,5,Hangup
```

As the "0" extension is first executed, Asterisk switches the call to Zap/9. If there is no answer, or if the phone is busy, Asterisk attempts to switch the call to three other extensions, Zap/10, Zap/11, an Zap/12. If none of these extensions answer, the call is switched to the operator's (extension zero) voice-mail. In this case, no announcement is played.

Least Cost Routing

Here is an example of least cost routing on outgoing lines. If a ZAP channel isn't available, the call will go out over an IAX channel.

```
exten => _9NXXXXXX,1,Dial,Zap/g2/BYEXTENSION
exten > _9NXXXXXX,2,Dial,IAX/oh/BYEXTENSION
exten > _9NXXXXXX,3,Congestion
```

This example demonstrates pattern matching. This shows that everything in an Asterisk dial plan is treated as an extension, even if it's an outgoing line.

Asterisk first tries to switch the outgoing call to any interface in "group 2." If that interface is unavailable, Asterisk tries to switch the call to a different IAX host named "oh." If this connection fails, the congestion tone is played.

Main Menu

Here is a simple Main Menu dial plan.

```
exten => s,1,Wait,1
exten > s,2,Answer
exten > s,3,DigitTimeout,5
exten > s,4,ResponseTimeout,10
exten > s,5,Background,intro
exten > s,6,Background,instructions
exten > 1,1,Goto,sales
exten > 2,1,Goto,support
exten > i,1,Playback,pbx-invalid
exten > i,2,Goto,s|6
exten > t,1,Goto,0|1
```

An incoming call is held for one second to let the calling party hear a ring. The call is answered. The digit and response time-outs are set to five and ten seconds. Asterisk then plays the"intro" message. This message could provide the calling party with a greeting, for example,"Thank you for calling our company" This is played in the background. This means the calling party can interrupt the message by pressing a key on the telephone keypad.

After the introduction, another message, the "instructions," is played. This could be a message like, "If you know your parties extension, dial it now.Dial 1 for sales or 2 for support."

If they calling party does not provide an extension, Asterisk switches the call to the operator. The dial plan for the operator is not shown in this example.

If the calling party enters an invalid extension, the pbx-invalid message is played to them. They are then played the instructions again.

Recording Sound Files

This configuration, suggested by Robert C, when added to *extensions.conf* will enable you to record messages. Whatever you say into a telephone is saved into a file. This is useful for recording Asteris responses.

Dialing extension 100 will record whatever you say and leave it in */tmp/asterisk-recording.gsm*. Press the # key or hang up to stop recording. Remember to rename the file *asterisk-recording* before recording another message. Note that Asterisk expects sound files to be held in the directory*/var/lib/asterisk/ sound*.

```
; Record a temp.GSM file
exten > 100,1,Wait(2)
exten > 100,2,Record(/tmp/asterisk-recording:gsm)
exten > 100,3,Wait(2)
exten > 100,4,Playback(/tmp/asterisk-recording)
exten > 100,5,Wait(2)
exten > 100,6,Hangup
```

Interactive Voice Response (IVR)

The following example shows how to create an interactive menu for incoming calls.

```
 [main
# lower case letter
# after an extension is reached, pressing the letter
```

```
# starts voicemai
exten > o,1,voicemailmain

exten> 2800,1,Dial(ZAP/${RECEPTIONIST},25,r)
exten > 2800,2,DigitTimeout,5
exten > 2800,3,ResponseTimeout,12
exten > 2800,4,Background,heartland

exten> i,1,Playback,pbx-invalid
exten > i,2,Goto,2800|1

# Time Out
exte> t,1,Goto,2800|1

exten> 0,1,Macro(zapdial,${RECEPTIONIST},20)
exten > 1,1,Macro(zapdial,2800,20)
exten > 7,1,Directory(inside)
```

Routing by Caller ID

Asterisk can route a call based on the caller ID of the incoming call.

```
exten > 100/6505551212,1,Congestion
exten > 100,1,Dial(Zap/1,20)
exten > 100,2,Voicemail(u100)
exten > 100,102,Voicemail(b100)
```

If the incoming call is from (650) 555-1212 a busy signal is played. Other calls are forwarded to the extension. If there is no answer, the call is forwarded to voicemail.

Music on Hold

An entry like this in extensions.conf will provide callers with music on hold.

```
exten > 2091,1,Answer
exten > 2091,2,Wait,1
exten > 2091,3,MusicOnHold,default
```

Note that *musiconhold.conf* must be configured properly as well. Consult the later section on musiconhold.conf for an example.

Using Globals

This example will ring two extensions simultaneously. Globals are used to make the configuration more easily readable.

```
[globals
PHONE1SIP/101
PHONE2SIP/102

TWOPHONES=${PHONE&${PHONE2}
..
[Sample
exten > 101,1,Dial(${TWOPHONES},30,t)
```

Goto and GotoIf

This is an example of using *goto* and *gotoif.* In the following example, the *GotoIfTime* executes every weekday from 9am to 5pm, in every month

```
exten > 4035,1,GotoIfTime(9:00-17:00|*|*|1-12?4:2)
exten > 4035,2,Dial(${N1})
exten > 4035,3,Dial(Hangup)
```

```
exten > 4035,4,Goto(default,4009,1)

exten> 4009,1,Dial(${N2})
exten > 4009,2,Dial(Hangup)
```

GotoIf expect two labels. If you only provide one label, a warning is written to */var/log/asterisk/messages*.

911 Support

Here is a sample configuration for including emergency 911 and 411 dialing support in your dial plan.

```
;-------------------------------------
; 911 Emergency and Directory Assistanc
;-------------------------------------

[emergency]
ignorepa> 9

; :1 - strip off the first digit dialed
exte> _9[49]11,1,Dial(${PRITRUNK1}/${EXTEN:1})
exten > _9[49]11,2,Congestion
exten > _9[49]11,102,Busy

exten> _[49]11,1,Dial(${PRITRUNK1}/${EXTEN:0})
exten > _[49]11,2,Congestion
exten > _[49]11,102,Busy
```

Local Calling

This is an example of local calling support.

```
;-----------
; Local call
;-----------

[trunklocal]

ignorep> 9
exten > _9NXXXXXX,1,Dial(${PRITRUNK1}/650${EXTEN:1})
exten > _9NXXXXXX,2,Congestion
exten > _9NXXXXXX,102,Busy

exten> _NXXXXXX,1,Dial(${PRITRUNK1}/650${EXTEN:0})
exten > _NXXXXXX,2,Congestion
exten > _NXXXXXX,102,Busy
```

Long Distance Dialing

Here is a sample dial plan for long distance calling.

```
;----------------------
; Domestic long distanc
;----------------------

[trunkld]

ignorep> 9

;exten> _91NXXNXXXXXX,1,Dial(${PRITRUNK1}/${EXTEN:1})
;exten > _91NXXNXXXXXX,2,Congestion
;exten > _91NXXNXXXXXX,102,Busy

exten> _1NXXNXXXXXX,1,Dial(${PRITRUNK1}/${EXTEN:0})
exten > _1NXXNXXXXXX,2,Congestion
```

```
exten > _1NXXNXXXXXX,102,Busy
```

Toll Free Calls

Here is a sample for toll free calling.

```
;------------------
; Domestic toll fre
;------------------

[trunktollfree]

ext> _91800NXXXXXX,1,Dial(${PRITRUNK1}/${EXTEN:2}})
exten > _91800NXXXXXX,2,Congestion
exten > _91800NXXXXXX,102,Busy
exten > _91888NXXXXXX,1,Dial(${PRITRUNK1}/${EXTEN:2})
exten > _91888NXXXXXX,2,Congestion
exten > _91888NXXXXXX,102,Busy
exten > _91877NXXXXXX,1,Dial(${PRITRUNK1}/${EXTEN:2})
exten > _91877NXXXXXX,2,Congestion
exten > _91877NXXXXXX,102,Busy
exten > _91866NXXXXXX,1,Dial(${PRITRUNK1}/${EXTEN:2})
exten > _91866NXXXXXX,2,Congestion

exten> _1800NXXXXXX,1,Dial(${PRITRUNK1}/${EXTEN:1}})
exten > _1800NXXXXXX,2,Congestion
exten > _1800NXXXXXX,102,Busy
exten > _1888NXXXXXX,1,Dial(${PRITRUNK1}/${EXTEN:1})
exten > _1888NXXXXXX,2,Congestion
exten > _1888NXXXXXX,102,Busy
exten > _1877NXXXXXX,1,Dial(${PRITRUNK1}/${EXTEN:1})
exten > _1877NXXXXXX,2,Congestion
exten > _1877NXXXXXX,102,Busy
exten > _1866NXXXXXX,1,Dial(${PRITRUNK1}/${EXTEN:1})
exten > _1866NXXXXXX,2,Congestion
exten > _1877NXXXXXX,102,Busy
```

Detecting an Incoming Fax

The following entry will detect and transfer an incoming fax.

```
exten => fax,1,Dial(SIP/ata1-2,20)
```

IAXtel

Iaxtel.com allows Asterisk users and IAX clients to connect with each other over the Inter-Asterisk eXchange protocol and theInternet instead of the PSTN. Once registered with IAXtel, each user gets a 1.700 telephone number that rings their IAX compatible client from anywhere on theInternet. You can register for an IAXtel number at*http://www.iaxtel.com*.

Here is a sample dial plan for making outgoing calls over IAXtel.

```
;----------------------------------------------------
; Calls to IAXTEL  (1700NXXXXXX
;---------------------------------------------------
[iaxtel
exten > _1700NXXXXXX,1,Dial(IAX2/username:password@iaxtel.com/
  ${EXTEN}@iaxtel
```

PBX functions with Asterisk

Various PBX functions are implemented as applications or a combination of applications.

General support (for all channels)

```
Music on Hold: Standard in Asterisk
Call Parking: Standard in Asteris
Call Pickup: Standard in Asteris
  note that *8 is defined in res_parking.
Call Recording: Using the 'Monitor' applicatio
Conferencing: Using the 'MeetMe' applicatio
IVR: Standard in Asterisk with applications
  note you can employ AGI or EAGI if even more control is neede
```

For SIP Phones

```
Call Hold: Normally implemented by the phone
Unattended Transfer (or"blind transfer"): Implemented in Asterisk (#),
  or optionally in the phon
Consultation Hold: Normally implemented by the phon
Attended Transfer (or"consultative transfer")
No Answer Call Forwarding: Implemented in the dial plan
Busy Call Forwarding:Implemented in the dial plan
Single-Line Extension
3-way Calling: usually implemented by the phon
Incoming Call Screening: Implemented in the dial pla
Find-Me
Call Pickup: Standard in Asteris
Outgoing Call Screening: Implemented in the dial pla
Automatic Redial: Implemented in the dial plan with some AGI suppor
Manual Redia
Do-not-disturb (DND
Message waiting (MWI): Standard in Asterisk, requires support on th
  phon
Call waiting indication: Standard in Asterisk, requires support on th
  phon
```

Analog Phones on a Zaptel channel

```
Call Hold: Implemented by the phone
Unattended Transfer (or"blind transfer")
Consultation Hold: Implemented by the phon
Attended Transfer or"consultative transfer"
No Answer Call Forwarding: Implemented in the dial plan
Busy Call Forwarding:Implemented in the dial plan
Single-Line Extension
3-way Calling: Iimplemented by the phon
Incoming Call Screening: Implemented in the dial pla
Find-Me
Call Pickup: Standard in Asteris
Outgoing Call Screening: Implemented in the dial pla
Automatic Redial: Can be implemented in the dial plan with AGI suppor
Manual Redia
Do-not-disturb (DND
Message waiting (MWI): Implemented in Asterisk, requires support fro
  the phon
```

for MGCP Phones

```
Manual Redial: Normally implemented by your phone
Unattended transfer (or"blind transfer"): Implemented in Asterisk (#)
Attended transfer: Implemented in Asterisk (FLASH
Call Forwarding: Implemented in Asterisk (*72 and *73); optionall
  implemented in the phon
Call Pickup: Implemented in Asterisk (*8
Call Waiting Indication: Implemented in Asterisk; disable with *7
Call Number Delivery Blocking: Implemented in Asterisk (*67
Do-not-disturb (DND): Normally implemented by your phone; also implemented in Asterisk (*78 and *79
```

```
Message waiting (MWI): Implemented in Asterisk, but must be support o
  the phon
```

on the CAPI channel

```
Call Deflection (CD) (redirect without answering): Implemented by
  chan_cap
CLIP& CLIR (display caller ID & hide my caller ID): Implemented by
  chan_cap
CID& DNID: Implemented by chan_capi
HOLD& RETRIEVE: Hold a call using ISDN (not the PBX): Implemented by
  chan_cap
Early B3 Connects (always,success,never): Implemented by chan_cap
DID (for Point to Point mode): Implemented by chan_cap
ECT (explicit call transfer): Preserve the orginial CID - Implemented b
  chan_cap
```

ChapterÂ 7 - SIP Configuration

SIP is a description protocol similar to HTTP and SMTP that allows two systems to initiate and control a media stream between endpoints. SIP supports authentication, caller ID, and media stream control.

SIP is rapidly gaining acceptance for VoIP. There are many commercial SIP providers, for example Voicepulse.

Sip Configuration Overview

Here is an overview, the details are covered at greater length below. SIP channels are configured in *sip.conf*. SIP calls, like any other call, are managed by the dial plan found in *extensions.conf.*

All calls arrive on a channel, for example a SIP channel. An incoming SIP call starts with a connection to aSIP channel. There is a configuration file for every type of channel, for example *sip.conf* for SIP channels. Here is an example of*sip.conf*. This example has a single context named *general*. Note this is not the same as a context in*extensions.conf.*

```
[general
port = 5060 ; TheTCP/IP port for SIP communications
bindaddr =0.0.0.0 ; Address to bind to.
context = from-sip ; Default for incoming call
```

The *context* in this example links this *sip.conf* context to a context in *extensions.conf*. In this example, the context comman names *from-sip.* Any call on the SIP channel will be by default processed by the context *from-sip* in *extensions.conf*.

Here is a sample from *extensions.conf* that supports outgoing SIP calls.

```
[from-sip]
exten > _*26.,1,Dial(SIP/${EXTEN:3}@inoc-dba.pch.net)
exten > _*26.,2,Congestion
exten > _*26.,102,Busy
```

The SIP dialstring depends on the channel. A SIP dialstring is specified as

```
/
```

The format of a SIP dialstring in *extensions.conf* is

```
SIP/@:
```

or

```
SIP/peer/exten
```

Peer is either a service defined in *sip.conf*, or a domain name, or the hostname of a SIP Proxy server.

Asterisk must register with an external SIP server to accept incoming calls from that server. The registration notifies the foreign server where the SIP calls should be sent. here are two examples of SIP registration with a foreign server. that could appear in *sip.conf*. In the first example, the user id is *1835* and the secret is*12345*.

```
register > 1835:12345@inoc-dba.pch.net/*1835

regist => 8776:ka6vep@iptel.org/*8776
```

The first registration provides inoc-dba.pch.net with the destination for calls to extension 1835. The second registration registers wit *iptel.org*. incoming calls will be referred to extension 8776.

Configuring Asterisk with SIP Phones

If you are using SIP phones, you must first configure the SIP phones, then you must configure Asterisk to operate with those phones. Configuring asterisk requires configuringSIP and then configuring the dial plan in extensions.conf.

The SIP configuration file for a phone is often a configuration file that is downloaded to the telephone, often with tftp. This configuration of the phone is done outside of Asterisk. Asterisk itself does not send aSIP configuration file to a telephone. Typically a server like TFTP is used to send the configuration file to the SIP phone.

Several configuration files must be modified to use a SIP telephone with Asterisk. As shown in the following figure, the information in each of the configuration files must be in agreement.

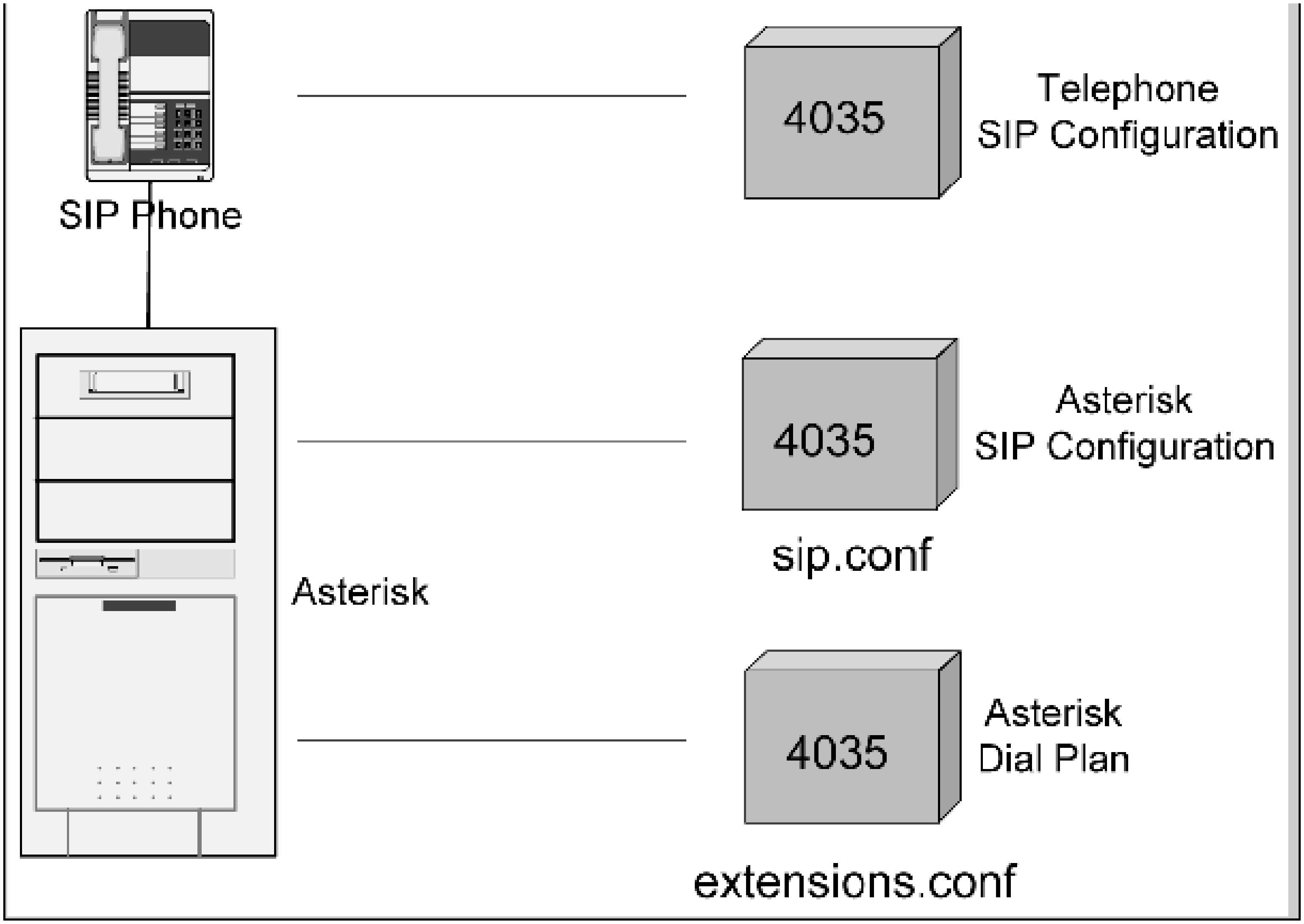


Figure: 07-1 SIP Phone Configuration

The SIP configuration for a phone must assign a numeric extension identifier for each line of the telephone. Here is a fragment from a configuration file for a Cisco 7960 that assigns extension 4035 to line one.

```
line1_authname: "4035"
# Line 1 Registration Passwor
line1_password:"cisco"
```

Every extension identifier must be unique across all telephones. Two different phones, or two different lines on a single phone, should never have the same extension identifier.

Every telephone extension must be configured in */etc/asterisk/sip.conf.* Here is fragment from *sip.conf* that configures extension 4035.

```
[4035]
type=friend           ; This device takes and makes call
username=403
secret=cisc
context=from-si
callerid"AUser" <4155551212>
qualify=100
host=dynamic  ; This host is not on the same IP addr every tim
canreinvite=n
mailbox=4035 ; Activate the message waiting light for waiting message
```

```
defaultip192.160.0.12
```

Any extension must be configured in */etc/asterisk/extensions.conf* to associate an extension identifier with one or more of theSIP device identities in *sip.conf.* Here is a fragment from *extensions.conf* that sends incoming calls to extension 403

```
[from-sip]
; If the number dialed by the calling party was"4035", then
;Dial the user "4035" via the SIP channel driver. Let the number
; ring for 20 seconds, and if no answer, proceed to priority 2
; If the number gives a"busy" result, then jump to priority 102

exten > 4035,1,Dial(SIP/4035,20)
```

A simple configuration using two SIP phones is shown later in this chapter. The simple configuration is the sample configuration on the Mepis distribution CD

Session Initiation Protocol (SIP) Channels

Outgoing SIP channels use the following format.

```
SIP/[@[:]
 - the name of the peer, or hostname or IP address of a remote
 server
 - an optional port number. Defaults to 5060, the standard SIP
 port
 - an optional extention.
```

Note the full length of the SIP string may not exceed 256 characters

Examples

```
SIP/ipphone - SIP peer "ipphone."
SIP/8500@sip.com:5060 - Extension 8500 at sip.com port 5060.
SIP/1010 - The SIP client '1010' on the local Asterisk server
SIP/OEJ - SIP client "OEJ" on the local asterisk server
SIP/10000@fwd.pulver.com:5060 - SIP client 10000 at fwd.pulver.com
```

Incoming SIP channels use the following format.

```
SIP/-
 - the identified peer.
 - a random identifier used to uniquely identify a call from a single
 peer
```

Examples

```
SIP/192.168.0.1-01fb34db a SIP call from 192.168.0.1.
SIP/sipphone-45ed721c a SIP call from the peer named "sipphone."
```

Defining SIP Channels

Any SIP client or server is identified in sip.conf. The syntax for defining a SIP channel is

```
[xxx
parameter1=valu
parameter2=valu
```

In this configuration, *xxx* is a username associated with a SIP client. Other configuration files use the section namexxx to refer to this SIP device. For example, if a SIP phone has been assigned a phone number of 123 in*extensions.conf*, then the corresponding section in *sip.conf* should be named [123].

A statement like

```
Host209.234.23.3
```

will allow incoming calls to be accepted from a remote server without a register entry in *sip.conf* for registration to the remote host. If the host is dynamic, then the SIP client must register to accept incoming calls from the remote host.

Sip.conf

The file *sip.conf* contains the definitions of SIP channels. All SIP channels must be defined here. This file is divided into contexts. The*[general]* context of sip.conf can reference the following variables.

```
port = : Port to bind to

bindaddr 0.0.0.0 :IP Address to bind to (listen on)

externip 200.201.202.203 :The SIP Address put in SIP messages when sent
  from behindNAT

context  :Default context for incoming calls in extensions.conf

srvlookup = yes|no :EnablDNS SRV lookups on outbound calls

pedantic = yes|no :Enable slow, pedantic checking of Call-ID:s for Pingtel

tos=lowdelay : SetQoS? parameters for outgoing media streams (numeric
  values are  accepted, like tos=184

maxexpirey=3600 :Max length of incoming registration we allow

defaultexpirey=120 :Default length of incoming/outoing registration

notifymimetype=text/plain :Allow overriding of mime type in NOTIFY used
  in voicemail online messages

videosupport=yes|no : Turn on supSIP video

disallow=all :Disallow all codecs

al :Allow codecs in order of preference

register> @/
  :Register with aSIP provider
```

There is currently no alternative to showing passwords in clear text in *sip.conf*.

SIP Configurations for Peers and Clients

SIP peer definitions are configured with the following variables in *sip.conf.*

```
accountcode: Used by Asterisk billing. Users may be associated with a
  accountcode

amaflags: Categorization for CDR records. Choices are default, omit,
  billing, documentation. See Asterisk billing
```

```
canreinvite: If the client is able to supSIP re-invites

context: Context in the dial plan for outbound calls from this client

defaultip: Default IP address of client host= is specified as DYNAMIC.
  Used if client have not been registred at any other IP adress.

dtmfmode: How the client handles DMTF signalling

fromuser: Specify user t"from" instead of callerid

host: How to find the client - IP # or host name. In case of DHCP networks, use the keyword dynamic

nat: This variable changes the behaviour of Asterisk for clients behind a
  firewall. This does not solve the problem if Asterisk is behind the
  firewall and the client on the outside.

mVoicemail extension (for message waiting indications)

qualify: Check if client is reachable

secret: PasswordSIP client (A shared secret)

md5secret: MD5-Hash o":asterisk:" (can be used instead
  of secret

type: Relationship to client - outbound provider or full client

username: Login nameSIP client

restrictid: (yes/no) To have the callerid restricted> sent as ANI

language: A language code defined in indications.conf - defines language
  for prompts and specific local phone signals

incominglimit and outgoinglimit: Limits for number of simultaneous
  active calls SIP client
```

Register Asterisk as a SIP client

Asterisk can function as a SIP client. In this case, SIP calls can be directed from some outside SIP server to your Asterisk server. Asterisk working as a client can recieve calls from a remoteSIP server.

A client must register with a server if the client is to accept calls from the server and the client appears on a dynamic IP address. The following entry in*sip.conf* at the server specifies that different calls from a client may arrive on different IP addresses.

```
host=dynami
```

To use Asterisk as a of SIP client when the IP address is dynamic, add a register definition to *sip.conf* in the section*[general]* of the client. This registration informs the remote server of the location of your Asterisk client. This is how the remote serve knows how to forward calls to your Asterisk client.

```
register > user:secret:authuser@host:port/extension
```

Example

This registers the extension 2345 at the SIP provider *asipprovider* as the local extension 1234.

```
register => 2345:password@asipprovider.com/1234
use - the user id for this SIP server (ex 2345)
authuser - an optional authorization user for theSIP server
secret - is the user passwor
host - is the domain or host name for theSIP server. This SIP server
  must have a corresponding definition in a separate section of sip.con
  titledmysipprovider.com.
/1234 - the Asterisk extension used for incoming calls. This must appea
  in extensions.con
```

The configuration at the SIP server accepting this registration would be

```
[mysipprovider.com]
type=pee
secret=passwor
username=234
hostsipserver.mysipprovider.com
fromuser=234
fromdomainfwd.pulver.com
nat=o
```

Asterisk as a SIP Server

SIP clients connecting to Asterisk must be defined in *sip.conf*.

Examples

```
[snomsip
type=frien
secret=bla
host=dynami
dtmfmode=inband ; Choices are inband, rfc2833, or inf
defaultip192.168.0.59
mailbox=1234,2345 ; Mailbox for message waiting indicato

[pingtel]
type=friend
username=pingtel
secret=blah
host=dynamic
qualify=1000 ; Consider it down if it's 1 second to reply
callgroup=1,3-4
pickupgroup=1,3-4
192.168.0.60

[cisco]
type=friend
username=cisco
secret=blah
nat=yes ; This phone may be natted
host=dynamic
canreinvite=no ; Cisco poops on reinvite sometimes
qualify=200 ; Qualify peer is no more than 200ms away
192.168.0.4

[cisco1]
type=friend
username=cisco1
fromuser=markster ; Specify user to pu"from" instead of callerid
secret=bla
host=dynami
defaultip192.168.0.4
amaflags=default ; Choices are default, omit, billing, documentatio
accountcode=markster ; Users may be associated with an accountcode t
  ease billin
```

A definition for any of these SIP clients in *sip.conf* enables logins and calls to the asterisk server from clients.

Example

```
exten > 1010,1, Dial(SIP/cisco1,10,t)
```

A call to extension 1010 is connected to the sip client logged in as cisco1.

Voicemail Waiting Indicator

Some phones have an indicator, for example a light, for waiting voicemail. To enable this light put an entry in*sip.conf* like

```
mailbox=7188@ContextInVoicemailCon
```

The context is the context for the mailbox specified in *voicemail.conf*.

Call Pickup

A call group allows any phone in the group to answer an incoming call directed to any of the phones in the group. If you include aSIP channel as part of a call group, you can use **8* to pick up an extension when it rings from any extension in the call group. You must specify the callgroup and pickupgroup i *sip.conf*

```
[3000
type=frien
username=300
secret=mypasswor
host=dynami
context=from-si
callgroup=
callgroup=
```

Other SIP Issues

As of the time of writing this book, Asterisk does not yet support SIP over TCP. Asterisk only supports SIP over UDP.

ChapterÂ 8 - Zaptel Configuration

Digium cards provide connectivity to the PSTN or to local telephony devices like analog telephones or fax machines. Digium makes a variety of telephony interface cards for Asterisk. They range from th singleFXO line X100P to quad span T1 and quad span FXO/FXS cards. You can have one or more of these cards installed in your Asterisk server. Other manufacturers make channel banks that supplemen the connectivity available with Digium cards.

The following quote is from *zapatatelephony.org* and explains why the interface is named *Zapata*.

> *When you buy standard commercially-available computer telephony hardware these days, after having your wallet absolutely raped, you find that the product i*

broken, or at least has funny quirks that even the manufacturer doesn't seem t know about (or care) about, and isn't interested in (or for that matter capable of giving you any reasonable level of support. This is completely consistent (withou exception) among all of the major manufacturers.

There is now finally hope after 15 years of this type of severe dysfunction.

The Zapata project, named after the famous Mexican Revolutionary, is an attempt to address these issues in a practical and livable manner.

As with SIP or IAX, Zaptel provides communications channels. Calls can arrive or leave over Zaptel channels. The dial plan determines how these calls are processed.

Digium Wildcard boards are Zaptel hardware devices. They share a common driver suite, the Zapata Telephony Driver Suite, Zaptel for short, and a common interface library.

The immediately following sections describe various Zaptel boards that are available from Digium. Configuration of the cards and Asterisk is then described in the following sections.

Wildcard X100P

The Wildcard X100P provides a single-port FXO PCI interface card for interfacing with a standard analog phone line. This board allows Asterisk to answer calls from a service provider's standard analog line or to receive calls from anotherPBX over TDM without the use of T1 hardware. The X100P is ideal for Interactive Voice Response andVoicemail applications.

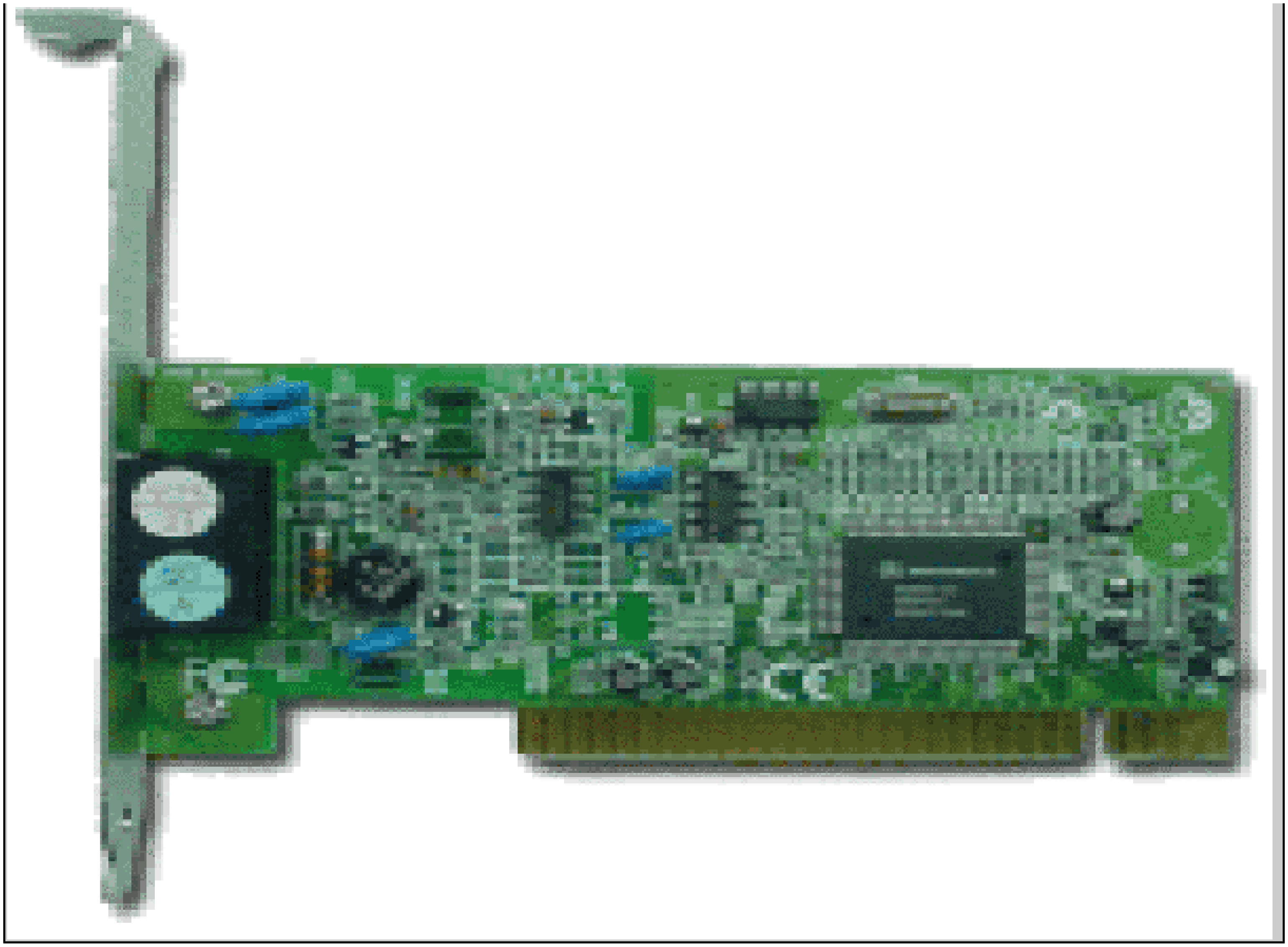


Figure: 08-1 X100P

The X100P supports all standard enhanced call features including CallerID, Call Conferencing and Call Waiting CallerID.

The X100P supports FXS Loopstart and "Kewlstart" (Loopstart with far end disconnection supervision). It can detect ringing and remote hangup and fully supports Pseudo-TDM bridging through Zaptel. The device is fully supported by AsteriskPBX for both incoming and outgoing calls.The two sockets on the back of the X100P; one labele *line interface* and *phone interface*. Connect the wall socket to the line interface. You can then optionally connect an analog telephone to the phone interface. Thi phone will operate if Asterisk fails or if there is a power failure

Wildcard TDM400P

The Wildcard TDM400P is a half-length PCI 2.2 compliant card that supports from one to four telephone interfaces for connecting analog telephones or analog lines to a PC. This quad-station FXS or FXO half-length PCI card supports standard analog and ADSI telephones for SOHO (Small Office Home Office) applications. This card accepts any combination of up to fourFXO and FXS modules.

Using Digium's Asterisk PBX software and standard PC hardware, one can create a SOHO (Small Office Home Office) telephony environment that includes all the sophisticated features of a high-en business telephone system

The TDM400P takes the place of an expensive channel bank and brings the system price point to a low level. By usingFXO and FXS modules with the TDM400P, one can create a solution with support for a range of telephones. To scale this solution, simply add additional TDM400P cards populate with modules.

In the UK you may need an adaptor that provides a ring capacitor or the phone may not ring. If you are using phones from the USA (aside from any power requirements they may have) you should just b able to plug them in

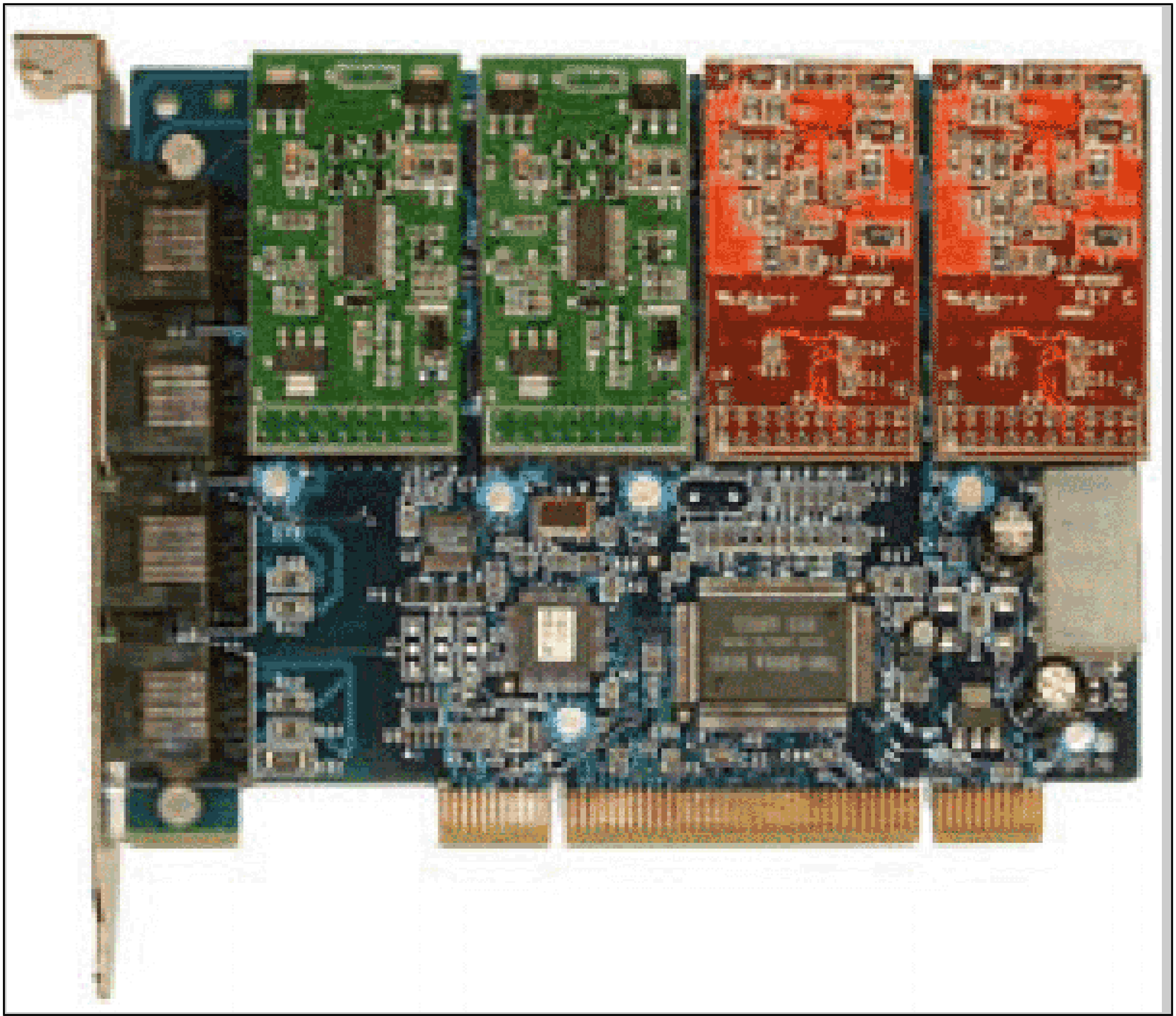


Figure: 08-2 TDM400P

Wildcard T100P

The T100P is a compact and powerful interface card supporting voice an data transmission over T1 andPRI connections. The single-span T1 half-length (available with 2U bracket) PCI card has the same features as the T400P. The low profile, half-length PCI form factor allows this device to fi within a 2U rack mount case or equivalent chassis. This provides excellent density for call center, service provider and other space-sensitive applications.

Used with Asterisk, the T100P offers the power to create a seamless network interconnecting traditional telephony systems with the emerging VoIP technologies. The T100P can be used to deliver a wide range ofPBX and IVR services to the network or handset including Voicemail, Call Conferencing, Three-way calling and VoIP Gateways. The European equivalent is the E100P.

This card supports both voice and data modes on its single T-span. For example, the card can support 12 channels dedicated to voice and 12 to data while passing all traffic through to the AsteriskPBX, which reliably routes the channels to their designated locations. This eliminates the need for an external router.

The T100P supports industry standard telephony and data protocols including Robbed Bit Signalling (RBS) and Primary RateISDN (PRI) protocols for voice, Cisco HDLS, PPP and Frame Relay for data transmission.

```
Switch Compatibility
  AT&T 4ESS
  DMS 100
  Lucent 5E
  National ISDN2
  Network or CPE

RBS Voice Modes
  A-Law, Mu-Law and Linnear Modes
  E&M
  E&M Wink
  Feature Group D
  Groundstart (FXO and FXS)
  Loopstart (FXO and FXS) with Optional Disconnect Supervision

Data Modes
  SyncPPP (both fixed and dialup)
  Frame Relay
  Cisco HDLC

Services and Features
  Caller ID
  Transmission /Reception Pseudo-TDM conferencing with Zaptel channels
  Digital gain control, transmit and recieve
  Dynamic span interaction (TDM over Ethernet)
Echo canceller
ISDN RAS capability
ISDN RAS capability
Local and remote loop backs
Pseudo-TDM bus architecture provides low latency
Supports same span voice and data
Tone internaltionalization (tone zones)
```

Figure: 08-3 T100P Features

By utilizing Digium TDMoE (TDM over Ethernet) technology, an exclusive Digium process, one can easily connect multiple PCs equipped with the T100P and achieve voice quality on par with singl PBX implementations. Scalability for this product is derived from adding multiple T100Ps to each individual PC. Add addition cards as you need them for your expanding applications

The T100P supports industry standard telephony and data protocols, including both RBS and Primary RateISDN (PRI) protocol families for voice and PPP, Cisco HDLC, and Frame Relay data modes. The board drives both line-side and trunk-side interfaces, including call features. The T100P is no FCC approved for Part 68.

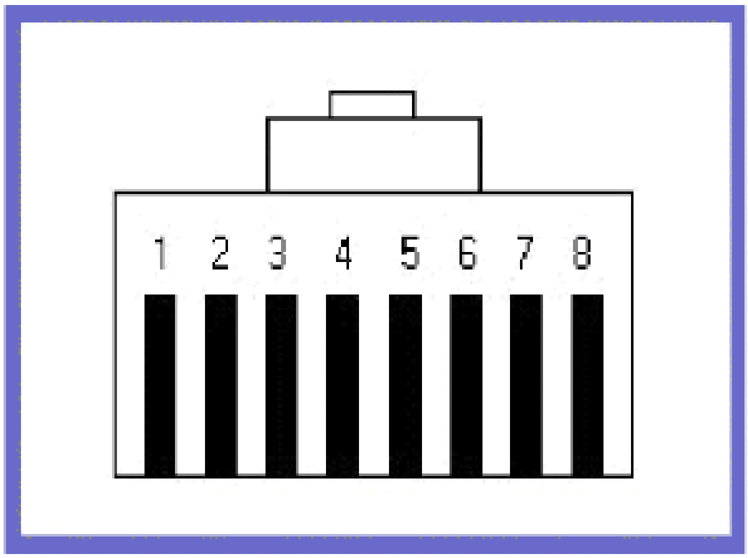
The E100P is the European equivalent of the T100P, providing a single E1 (32-channel) interface.

T1 Cables

First, note that a real T1 cable is not the same as a CAT5 cable. You are much better served by using a real T1 cable.

Second, note that the T100P is manufactured in such a manner that you may very likely need a T1 crossover cable to connect between the T100P and an incoming T1 line. This means that you wil most likely not need a crossover cable to connect between a T100P and a channel bank.

Here is the wiring for a T1 crossover cable.



Pin 1	⟷	Pin 5
Pin 2	⟷	Pin 4
Pin 3	⟷	Pin 3
Pin 4	⟷	Pin 2
Pin 5	⟷	Pin 1
Pin 6	⟷	Pin 6
Pin 7	⟷	Pin 7
Pin 8	⟷	Pin 8

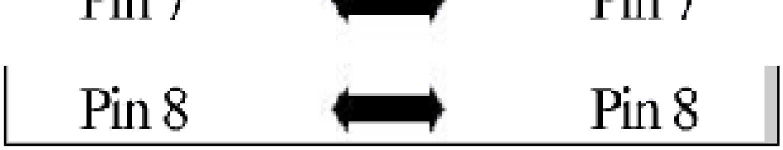


Figure: 08-4 T1 Crossover Cable

Wildcard E100P

A single-span E1 half-length (available with 2U bracket) PCI card sporting the same features as the T400P, the quad port version. The E100P is a single span E-1 (30-channel) card that supports all th functionality of our quad E1 card. This card supports both voice and data modes on its single-T span For example, the card can support 16 channels dedicated to voice and 16 to data while passing all traffic through to the AsteriskPBX, which reliably routes the channels to their designated locations. This eliminates the need for an external router

```
PRI Switch Compatibility
  EuroISDN
  network or CPE

CAS Voice Modes
  A-Law, Mu-Law and linear Modes Supported
  E&M
  E&M Wink
  Feature Group D
  Groundstart (FXO & FXS)

Data Modes
  SyncPPP (both Fixed and Dialup)
  Frame Relay
  Cisco HDLC

Services and Features
  Caller ID
  Transmission / Reception Psuedo-TDM Conferencing with Zaptel
  Channels
  Digital Gain Control (Transmit & Recieve)
  Dynamic Span Interaction (TDM over Ethernet)
  Echo Canceller
  ISDN RAS Capability
  Local and Remote Loop Backs
  Pseudo-TDM Bus Architecture keeps Latency Low
  Support Same Span Voice and Data
  Tone Internaltionalization (Tone Zone)
```

```
Support Same Span Voice and Data
Tone Internaltionalization (Tone Zone)
```

Figure: 08-5 E100P Features

By utilizing our TDMoE (TDM over Ethernet) technology, an exclusive Digium process, one can easily connect multiple PCs equipped with the E100P and achieve voice quality on par with single PBX implementations. Scalability for this product is derived from adding multiple E100Ps to each individual PC. Add addition cards as you need them for your expanding applications

The E100P supports industry standard telephony and data protocols, including both RBS and Primary RateISDN (PRI) protocol families for voice and PPP, Cisco HDLC, and Frame Relay data modes. The board drives both line-side and trunk-side interfaces, including call features

The T100P is the US equivalent of the E100P, providing a single T1 (24 channel) interface.

Wildcard TE410P/TE405P

A quad-span togglable E1/T1 card enables per card or per port selection of either T1 or E1 signaling formats. The TE410P is a 3.3 volt PCI card, the TE405P is a 5 volt card. This card provides four separate connections, or spans. Each span can provide for T1 or E1 signalling. The TE405P can also be quad E1 or T1 selectable per card or per port. You can do both signaling formats in a single card. Thi card improves performance and scalability with a bus mastering design. The TE405P has beenFCC, CE, and UL approved

The TE405P supports a 5.0v PCI slot only. The TE410P supports a 3.3v PCI slot only - typically available on newer motherboards and in 64-bit PCI bus architectures. These cards are not interchangeable between 3.3v and 5.0v PCI slots. Customers ordering a card not matching their availabl PCI slots will be held accountable for all freight charges and incur a $30 handling fee to rectify the situation. If you are unsure about the PCI slots on your motherboard, please click the following link

We do not recommend use of the TE405P in dual processor Athlon systems.

FXO and FXS Devices

If you are not using T1 or E1 connections, if you are using FXO or FXS adaptors, you don't need span definitions. WithFXO or FSX adaptors channels appear in the order the drivers are loaded. For example, if you have a single port FXO card and a USB single port FXS interface, you would load the FXO driver and then the USB driver. TheFXO driver would appear as channel one and the USB FXS would be channel two.

FXO and FXS signalling is the reverse of the type of signalling for the interface itself. FXS interfaces are signalled withFXO. FXO interfaces are signalled with FXS.

Only a single line is required to configure each interface. For example.

```
fxsks=1
fxoks=
```

loads the FXO device as channel one and the FXS device as channel two.

PCI Slots

Today's PC motherboards feature a variety of PCI slot types. Here, for example, is a picture of a typical dual processor motherboard with varying types of slots

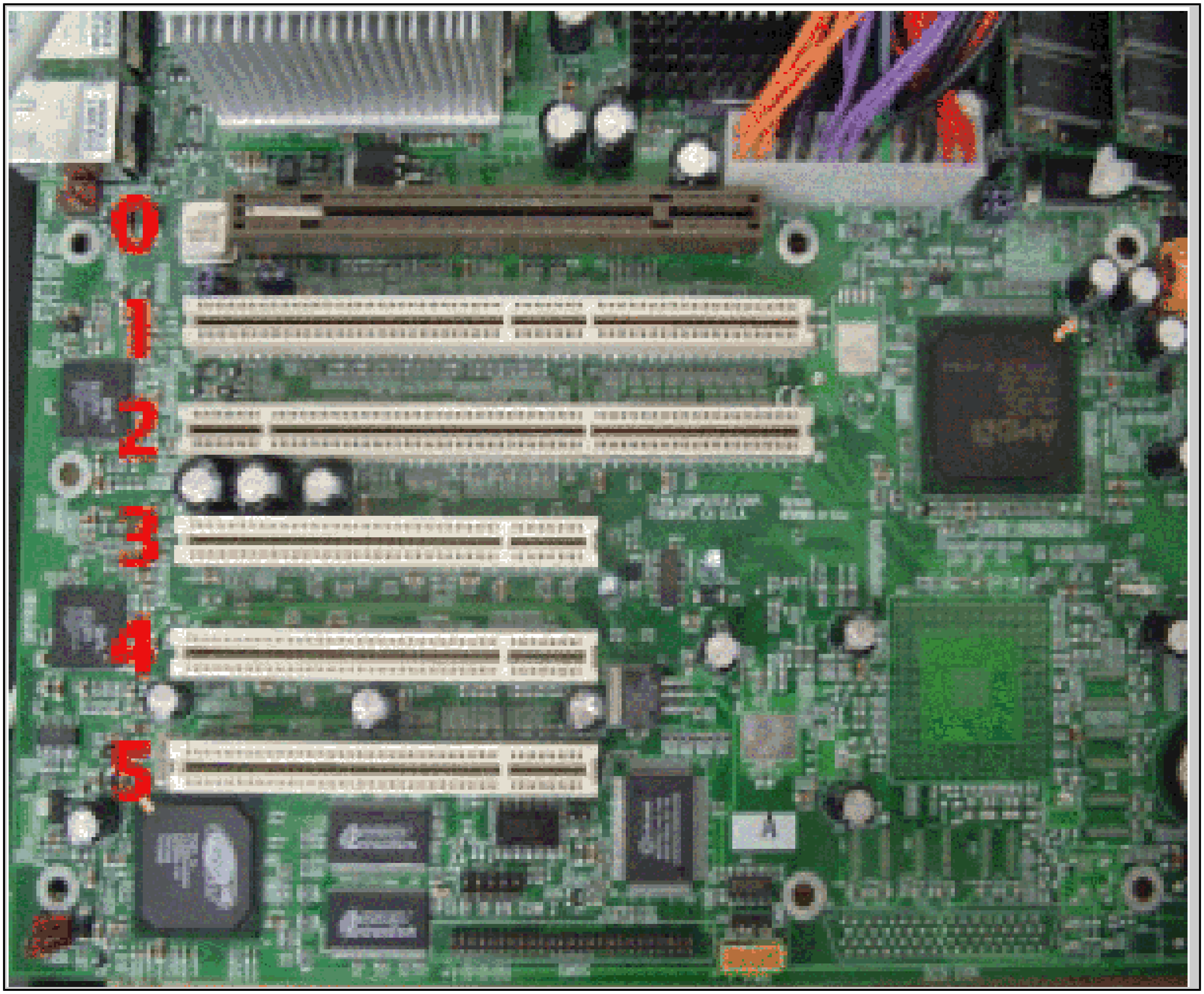


Figure: 08-6 Sample Motherboard

The following table calls out the PCI and AGP slots shown on the motherboard above. Each of the slots provides different interfaces. The top slot shown in the illustration is the AGP Pro slot, slot number zero.

```
0: AGP Pro Slo
1: 64-bit 5.0 volt PCI Slo
2: 64-bit 3.3 volt PCI Slo
3: 32-bit 5.0 volt PCI Slo
4: 32-bit 5.0 volt PCI Slo
```

```
5: 32-bit 5.0 volt PCI Slo
```

Note that the different types of slots have a different physical configuration. Boards are keyed to fit into the correct type of slot.

The TE410P is a 32-bit 33MHz card keyed for 3.3 volt operation. This means that in the mother-board pictured here, the TE410P will only fit into Slot #2. The TE410P will not fit into Slots 1, 3, 4, or 5.

The TE405P is a 32-bit 33MHz card keyed for 5.0 volt operation. This means that in the mother-board pictured here, the TE405P will fit into Slots 1, 3, 4, and 5. The TE405P will not fit into Slot #2.

International Use and Caller ID

Note that Digium cards will operate well in most countries, but not all countries telephone networks supply caller ID.

Channel Banks

A channel bank is a multiplexer. A channel bank has one or more high-speed T1 connections on one side and multipleFXS or FXO ports on the other side. A channel bank manages multiple telephone connections. For example, a channel bank can provide 24FXS ports or 24 FXO ports. The channel bank can connect to a T1 Zaptel card.

If you use a channel bank, you will need to configure it for use with Asterisk. Consult the manufacturer's documentation for assistance with configuration.

There are several manufacturers of channel banks including Adit, Adtran and Rhino. Features you want in a channel bank include 2-wire support, disconnect supervision, and support for fx lines. Th channel bank must be able to function as a ring generator, that is it must be able to supply 100 va ringing voltage.

Modern channel banks can translate analog signaling features into a T-1 format. For example, a modern channel bank should be able to interpret the 1200 baud FSK caller ID stream that is inserted between the first and second ring and translate that into digital caller ID delivery.

You should look for the following features in a channel bank.

```
Caller ID
Caller ID call waiting
distinctive ring
call waitin
analog 3-way calling (flash hook
analog call transfer (3-way call w/hang up
stutter dial-tone (message waiting
far end disconnect supervision onFXO cards
```

Some channel banks like the ADIT 600 provide dynamic impedance. This is very helpful for eliminating echo at the source

The channel bank and the Asterisk server talk T1 to each other. You supply a T1 connection between the channel bank and the Asterisk server. That is, you put a T1 card in the Asterisk box and then connect it to the channel bank, usually with a T1 crossover cable. Please remember to use a real T1 cabl and not a cat-5 cable. The channel bank can then"break out" the individual channels from the T1 card into separate ports.

For example, take an installation with a T1 line from a phone company and a channel bank. A Digium T1 card in the Asterisk server provide for a connection to the channel bank. A crossover cable connec the two devices. The channel bank ports are set up for any combination of fxs or fxo. The channel ban expects T1 signalling, for example B8ZS/ESF with wink start or some other T1 protocol.

With the availability of quad span Digium cards, there is less occasion to use a channel bank. For example, with six open slots you could run six quad span Wildcard TDM400 cards. This would provide 24 channels in any combination ofFXO or FXS channels. However with a Digium quad span T1 card, you could run 96 channels with a channel bank. If you need to access a large number of analo lines, a channel bank may be just what you need.

Hardware Installation

First install any cards into the computer. Be sure to be well grounded, preferably with a wrist strap, before installing any cards. Note that some Digium cards require a modern motherboard that supplies Â

5.0 volts. Some cards require a connection to the computer power supply.

Configuration Files

There are two configuration files you must change when you use Zaptel cards. The two files are *zaptel.conf* and *zapata.conf*.

The file *zapata.conf*, often found in the directory */etc,* contains configuration information for Zaptel boards. This file contains information used to configure the hardware for the corresponding hardwar drivers.

The file *zapata.conf,* often found in the directory */etc/asterisk,* contains configuration information that describes how Asterisk interacts with the Zaptel cards

Kernel Drivers

Before starting Asterisk, you must have loaded the drivers for any Digium boards you have installed. Asterisk may not start or operate correctly if the drivers for the boards are not loaded. You can ru *modprobe* manually from the command line for each driver

```
modprobe wct1xx
```

or automatically load the drivers with the Linux boot files. For example, Debin lists drivers to load in the file*/etc/modules.*

The modprobe command loads the appropriate driver while resolving any known dependencies on other modules. For example, the following command loads the drivers for the four portFXS board

```
modprobe wcfxs
```

At the time of writing, the following boards were available. Column two shows the argument for the modprobe command.

TABLE: 08-1 Digium Interface Cards		
Card	modprobe	description
TE410P	wct4xxp	Quad-span togglable E1/T1. 3.3 volt PCI onl
TE405P	wct4xxp	Quad-span togglable E1/T1. 5.0 volt PCI onl
TDM400P	wcfxs	Quad-Station FXS or FXO
T100P	wct1xxp	Single-Span T1
E100P	wct1xxp	Single-Span E1
X100P	wcfxo	Single-port FXO

Note that the order that the drivers are loaded will determine the channel assignments of the drivers. You must load the drivers in the appropriate order. For example, if you have a T100P board and a X100P board and you could load the drivers wit

```
modprobe wct1xxp
modprobe wcfx
```

To see errors produced by the *modprobe* command, use the command *dmesg*. Other helpful error related information is avalable in any of the files created in the directory /proc/zaptel. This command, an these files, can help you diagnose errors in the zaptel configuration process, for example boards tha have not been provided with power or drivers that are loading in the wrong order.

With FXO or FXS adaptors channels appear in the order the drivers are loaded. For example, if you have a single portFXO card and a USB single port FXS interface, you would load the FXO driver and then the USB driver. TheFXO driver would be channel one and the USB FXS would be channel two.

The T100P board has twenty-four channels, the X100P board has one channel. Loading the driver for the T100P driver first causes the first twenty-four channels to be assigned to the T100P board an channel twenty-five to be assigned to the X100P board.

Note that zaptel.conf must configure all the channels for all the boards, even if they are not all in use. Here is an example with three Digium boards.

```
# zaptel.con
# T100p - T1 Lin
span=1,0,0,esf,b8z
&m=1-24
# TDM400p - fxs lin
fxoks=2
# X100P - fxo lin
fxsks=2
loadzone=u
defaultzone=u
```

In the example above, X100P is an FXO card. This card is designed to accept a connection from the PSTN. Note that the configuration for the card shown above lists the configuration as *fxsks* not *fxoks*. In this example, the TDM400p board only has one fxs module installed and the other three position are empty. Even so, the X100p card appears on channel 29.

FXO and FXS signalling is the reverse of the type of signalling for the interface itself. FXS interfaces are signalled withFXO and FXO interfaces are signalled with FXS.

Only a single line is required to configure each interface. For example,

```
fxsks=1
fxoks=
```

loads the FXO device as channel one and the FXS device as channel two.

Zaptel drivers may conflict with other drivers. For example, Digium drivers will often require the same interrupt as the USB interface. You may have to unload drivers that conflict with the Digium drivers.

To see a list of loaded drivers run the command

```
lsmo
```

To unload a driver use the command

```
rmmo
```

ztcfg

The program *ztcfg* reads the configuration information in *zaptel.conf* and configures the drivers. You must run*ztcfg* each time zaptel driver are loaded, for example after booting the machine. You can run *ztcfg* after you have made any changes to *zaptel.conf* to reconfigure the drivers.

zttool

The *zttool* program displays the status of installed Zaptel boards. The drivers for the cards must be loaded with*modprobe* as described above for *zttool* to work. *Zttool* will show if the installed Zaptel cards are running correctly. If they are not, you will need to alter the configuration information in*zaptel.conf*. Remember that

To install *stool* with Mepis, use the following commands.

```
cd /usr/src/zaptel
apt-get updat
apt-get install libnewt-de
make zttoo
```

Redhat users should install the newt-devel package. You should now be able to run */sbin/zttool*. This will display the status of each of the running interfaces.

The options for the command are

```
-c      use  instead of /etc/zaptel.conf
-h            show the available argument
-v            verbos
-t            test mode--don't use
-s            shutdown spans onl
```

Zttool shows the current channels and their states. Use the *tab* key to select between the two buttons when you wish to exit the program.

The states shown by zttool correspond to the states for the boards.

IRQ Settings

It is better to provide Zaptel cards with exclusive access to an IRQ. The file /proc/interrupts lists interrupt assignments. You may be able to change interrupt assignments through the BIOS utility for your motherboard. Disable any USB drivers like sound drivers or USB drivers that you don't need.

Zaptel Configuration

You must compile and install the zaptel, zapata and Asterisk software before configuring any Zaptel cards. You must configure*/etc/zaptel.conf* to configure the hardware interface for any Digium cards and */etc/asterisk/zapata.conf* to configure Asterisk for use with any Digium cards. While the configuration files may look intimidating, setting up zaptel cards is actually pretty easy.

The zaptel channels are configured in the file */etc/zaptel.conf*. The file *zaptel.conf* contains configuration lines of the forma

```
parameter=valu
```

Comment lines begin with the pound sign, #.

Here is an example configuration for a T100P and a TDM400P with four FXS modules taken from a working installation. In this installation, the T100P is connected to aPRI from SBC. Four analog phones in the office are connected to the TMD400P. Here is what appears in zaptel.con

```
# zaptel.conf
span=1,1,0,esf,b8z
bchan=1-2
dchan=2
loadzone = u
defaultzone=u
fxoks=25-2
```

The following is part of the corresponding zapata.conf file and configures the T1 line for twenty-three voice channels and the one data channel that is reserved for thePRI signalling.

```
; zapata.con
[channels
context=defaul
switchtype=nationa



signalling = pri_cp
switchtype=dms10
group=
context=mai
channel > 1-23

signalling=fxo_ks
context=inside
chan>25-28
```

Note that only 23 channels are available as the T1 is set up as a PRI. Any calls coming in on a PRI channel will be managed by the main context in*extensions.conf.*

The four FXO ports are set up as channels 25 to 28. An calls made from one of these phones is managed by the *inside* context.

This example is used in an installation that connects the first two spans of a TE400P to a channel bank. The channel bank makes the forty-eight T1 channels available asFXS ports. The third and fourth spans in this example connect to two T1 lines, another forty-eight channels. These channel connect to thePSTN over the two T1 lines. These T1 lines are not PRI lines.

Here is the configuration in zaptel.conf

```
# zaptel.conf
span=1,0,0,esf,b8z
span=2,0,0,esf,b8z
span=3,0,0,esf,b8z
span=4,0,0,esf,b8z

fxoks=1-48&m=49-96

loadzone = us
defaultzone
```

This sets the channel configuration for each of the four spans. Channels 1-48 will be used to connect to the channel bank, channels 49-96 will connect to the two T1 lines from XO.

If you are not using T1 or E1 boards, but you are using FXO or FXS adaptors, yo don't need span definitions.

Here is part of the corresponding configuration in *zapata.conf*. This example is drawn from the same working installation. This sample configures the access to theFXO channels.

```
; zapata.con

; 4/17/2004 - Paul Mahler www.signate.com

[channels]
language=en

;switchtype=national
signalling=fxo_ks
rxwink=300

usecallerid=yes
hidecallerid=no
callwaiting=yes
callwaitingcallerid=yes
threewaycalling=yes
transfer=yes
cancallforward=yes
callreturn=yes

signalling=fxo_ks> 1-48

signalling=em_w
group=2
chann> 49-96
```

Outgoing Zap channel names use the following format

```
Zap/[g[c][r/
 - numberical indentifier for the physical channel number of
 the selected channel
[g] - the identifier is a group number instead of a channel. Seezapata.conf.
[c] - request answer confirmation. A number is not consdered answere
 until the called party presses#.
[r] - distinctive rin
[cadence] - an integer between one and four
```

Examples

```
Zap/1 - TDM Channel
Zap/g1 - First available channel in group
Zap/3r2 - TDM Channel 3 with 2nd distinctive rin
Zap/g2c - First available channel in group 2 with confirmatio
```

Incoming Zap channels are labeled

```
Zap
 - the channel number
 - a number from 1 to 3. Indicates the logical channel associated with a single physical channel.
```

Examples

```
Zap/1-1 - First call appearance on TDM channel
Zap/3-2 - Second call appearance on TDM channel
```

Zaptel.conf

In *zaptel.conf*, T1/E1 interfaces take several values and have the format

```
span=(spannum),(timing),(LBO),(framing),(coding
```

The values for each of these arguments depends on the configuration of the equipment at the far end of the T1 or E1 line. Timing defines how timing is synchronized between the devices.

```
0 - don't use this span as a sync sourc
1 - primary sync sourc
2 - secondary sync source, etc
```

The line build-out (or LBO) is an integer, from the following table:

```
# 0: 0 db (CSU) / 0-133 feet (DSX-1)
# 1: 133-266 feet (DSX-1
# 2: 266-399 feet (DSX-1
# 3: 399-533 feet (DSX-1
# 4: 533-655 feet (DSX-1
# 5: -7.5db (CSU
# 6: -15db (CSU
# 7: -22.5db (CSU
```

The choices for framing are one of *d4* or *esf* for T1 or *cas* or *ccs* for E1

The coding is one of *ami* or *b8zs* for T1 or *ami* or *hdb3* for E1. E1 lines may have the additional keyword *crc4* to enable CRC4 checking

If the keyword *yellow* follows, yellow alarm is transmitted when no channels are open. Here are some examples.

```
span=1,1,0,esf,b8z
span=2,0,0,esf,b8z
```

```
span=3,0,0,esf,b8z
span=4,0,0,esf,b8z
```

or

```
span=3,0,0,ccs,hdb3,crc4
```

Dynamic span definitions have the form

```
dynamic,
,,
 - the name of the driver (e.g. eth),
 - the driver specific address (like a MAC for ethernet)
 - the number of channels
 - a timing priority, like for a normal span.
```

Use a value of zero to not use this as a timing source. You can prioritize them as primary, secondary, etc. Note that you MUST have a REAL zaptel device if you are not using external timing

The definitions for using the channels are next. The format is:

```
=
```

Valid devices are:

```
e&m     : Channel(s) are signalled using E&M signalling (specific
         implementation, such as Immediate, Wink, or Feature Group
         are handled by the userspace library)
fxsls   : Channel(s) are signalled using FXS Loopstart protocol.
fxsgs   : Channel(s) are signalled using FXS Groundstart protocol.
fxsks   : Channel(s) are signalled using FXS Koolstart protocol.
fxols   : Channel(s) are signalled using FXO Loopstart protocol.
fxogs   : Channel(s) are signalled using FXO Groundstart protocol.
fxoks   : Channel(s) are signalled using FXO Koolstart protocol.
sf    : Channel(s) are signalled using in-band single freq tone.
```

The syntax is

```
channel# => sf:,,,,,
xfreq is rx tone freq in hz, rxbw is rx notch (and decode)
bandwith in hz (typically 10.0), rxflag is either 'normal' or
'inverted' txfreq is tx tone freq in hz, txlevel is tx ton
level in dbm, txflag is either 'normal' or 'inverted'. Se
rxfreq or txfreq to 0.0 if that tone is not desired
unused  : No signalling is performed, each channel in the list remains
 idl
clear   : Channel(s) are bundled into a single span.  No conversion or
 signalling is performed, and raw data is available on the master
indclear: Like clear except all channels are treated individually and are
 not bundled.  bchan is an alias for this
rawhdlc : The zaptel driver performs HDLC encoding and decoding on the
 bundle, and the resulting data is communicated via the masterdevice
fcshdlc : The zapdel driver performs HDLC encoding and decoding on the
 bundle and performs incoming and outgoing FCS insertion and verification.  dchan is an alias for this
nethdlc : The zaptel driver bundles the channels together into an
hdlc network device, which in turn can be configured with sethdl
 (available separately)
dacs    : The zaptel driver cross connects the channels starting at
the channel number listed at the end, after a colo
```

The channel list is a comma-separated list of channels or ranges, for example:

```
1,3,5 (channels one, three, and five)
16-23, 29 (channels 16 through 23, as well as channel
```

Here are some complete examples.

```
&m=1-12
nethdlc=13-2
fxsls=25,26,27,2
fxols=29-3

fxoks=1-24
bchan=25-47
dchan=48
fxols=1-12
fxols=1&m=25-29
nethdlc=30-3
clear=4
clear=4
clear=4
clear=4
fcshdlc=4
dacs=1-24:4
```

You can preload some tone zones to prevent them from getting overwritten by other users (if you allow non-root users to open /dev/tor* # interfaces anyway. This means they won't have to be loaded at runtime. The format is

```
loadzone=
```

where the *zone* is a two letter country code.

You can specify a default zone with

```
defaultzone=
```

where zone # is a two letter country code.

```
loadzone = u
#loadzone=f
#loadzone=d
#loadzone=u
#loadzone=f
#loadzone=j
#loadzone=s
#loadzone=n
defaultzone=u
```

zapata.conf

The file */etc/asterisk/zapta.conf* contains the configuration information Asterisk needs for its use of any Zaptel hardware. Following is the sample configuration file shipped with Asterisk for*/etc./asterisk/zapata.conf.*

```
; Zapata telephony interfac

; Configuration fil

[channels]
;
; Default language
;
;language=en
;
; Default context
;
context=default
;
; Switchtype:  OnPRI.

; national:  NationalISDN 2 (default)
; dms100:  Nortel DMS10
; 4ess:          A&T 4ESS
```

```
; 5ess:          Lucent 5ES
; euroisdn:       EuroISD
; ni1:            Old NationalISDN 1

switchtype=nationa

;PRI Dialplan:  Only RARELY used for PRI.

; unknown:        Unknow
; private:        PrivateISDN
; local:          LocalISDN
; national:  NationalISDN
; international:  InternationalISDN

;pridialplan=nationa

; Overlap dialing mode (sending overlap digits

;overlapdial=ye

; Signalling method (default is fxs).  Valid values
; em:      E& M
; em_w:    E& M Wink
; featd:   Feature Group D (The fake, Adtran style, DTMF
; featdmf: Feature Group D (The real thing, MF (domestic, US)
; featb:   Feature Group B (MF (domestic, US)
; fxs_ls: FXS (Loop Start)
; fxs_gs: FXS (Ground Start)
; fxs_ks: FXS (Kewl Start)
; fxo_ls: FXO (Loop Start)
; fxo_gs: FXO (Ground Start)
; fxo_ks: FXO (Kewl Start)
; pri_cpe:PRI signalling, CPE side
; pri_net:PRI signalling, Network side
; sf:      SF (Inband Tone) Signallin
; sf_w:      SF Win
; sf_featd:   SF Feature Group D (The fake, Adtran style, DTMF
; sf_featdmf: SF Feature Group D (The real thing, MF (domestic, US)
; sf_featb:   SF Feature Group B (MF (domestic, US)
; The following are used for Radio interfaces
; fxs_rx:  Receive audio/COR on anFXS kewlstart interface (FXO at the
 channel bank
; fxs_tx:  Transmit audio/PTT on anFXS loopstart interface (FXO at the
 channel bank
; fxo_rx:  Receive audio/COR on anFXO loopstart interface (FXS at the
 channel bank
; fxo_tx:  Transmit audio/PTT on anFXO groundstart interface (FXS at the
 channel bank
; em_rx:   Receive audio/COR on an &M interface (1-way)
; em_tx:   Transmit audio/PTT on an &M interface (1-way)
; em_txrx: Receive audio/COR AND Transmit audio/PTT on an &M interface
 (2-way
; em_rxtx: same as em_txrx (for our dyslexic friends
; sf_rx:   Receive audio/COR on an SF interface (1-way
; sf_tx:   Transmit audio/PTT on an SF interface (1-way
; sf_txrx: Receive audio/COR AND Transmit audio/PTT on an SF interfac
 (2-way
; sf_rxtx: same as sf_txrx (for our dyslexic friends

;signalling=fxo_l

; A variety of timing parameters can be specified as wel
; Including
;    prewink:     Pre-wink tim
;    preflash:    Pre-flash tim
;    wink:        Wink tim
;    flash:       Flash tim
;    start:       Start tim
```

```
;    rxwink:      Receiver wink tim
;    rxflash:     Receiver flashtim
;    debounce:    Debounce timin

rxwink=300; Atlas seems to use long (250ms) wink

; Whether or not to use caller I

usecallerid=ye

; Whether or not to hide outgoing caller ID (Override with *67 or *82

hidecallerid=n

; Whether or not to enable call waiting onFXO lines

callwaiting=ye

; Whether or not restrict outgoing caller ID (will be sent asANI only,
 not available for the user
; Mostly use withFXS ports

;restrictcid=n

; Whether or not use the caller ID presentation for the outgoing cal
 that the calling switch is sendin

usecallingpres=ye

; Support Caller*ID on Call Waitin

callwaitingcallerid=ye

; Support three-way callin

threewaycalling=ye

; Support flash-hook call transfer (requires three way calling

transfer=ye

; Support call forward variabl

cancallforward=ye

; Whether or not to support Call Return (*69

callreturn=ye

; Stutter dialtone support: If a mailbox is specified, then when voicemai
; is received in that mailbox, taking the phone off hook will caus
; a stutter dialtone instead of a normal on

;mailbox=123

; Enable echo cancellatio
; Use either"yes", "no", or a power of two from 32 to 256 if you wish
; to actually set the number of taps of cancellation

echocancel=ye

; Generally, it is not necessary (and in fact undesirable) to echo cance
; when the circuit path is entirely TDM.  You may, however, reverse thi
; behavior by enabling the echo cancel during pure TDM bridging below

echocancelwhenbridged=ye

; In some cases, the echo canceller doesn't train quickly enough an
 ther
; is echo at the beginning of the call.  Enabling echo training wil
 caus
```

```
; asterisk to briefly mute the channel, send an impulse, and use th
 impuls
; response to pre-train the echo canceller so it can start out with
 muc
; closer idea of the actual echo

;echotraining=ye

; If you are having trouble with DTMF detection, you can relax th
; DTMF detection parameters.  Relaxing them may make the DTMF detecto
; more likely to have"talkoff" where DTMF is detected when it
; shouldn't be

;relaxdtmf=ye

; You may set the default receive and transmit gains (in dB

rxgain=0.
txgain=0.

; Logical groups can be assigned to allow outgoing rollover.  Group
; range from 0 to 31, and multiple groups can be specified

group=

; Ring groups (a.k.a. call groups) and pickup groups.  If a phone i
 ringin
; and it is a member of a group which is one of your pickup groups, the
; you can answer it by picking up and dialing *8#.  For simple offices
 jus
; make these both the sam

callgroup=
pickupgroup=

;
; Specify whether the channel should be answered immediately or
; if the simplitch should provide dialtone, read digits, etc.

immediate=n

; CallerID can be set to"asreceived" or a specific number
; if you want to override it.  Note that"asreceived" only
; applies to trunk interfaces

;callerid=256428600

; AMA flags affects the recording of Call Detail Records.  If specifie
; it may be 'default', 'omit', 'billing', or 'documentation'

;amaflags=defaul

; Channels may be associated with an account code to eas
; billin

;accountcode=lss010

; ADSI (Analog Display Services Interface) can be enabled on a per-channe
; basis if you have (or may have) ADSI compatible CPE equipmen

;adsi=ye

; On trunk interfaces FXS) and E&M interfaces (E&M, Wink, Feature Group

; etc, it can be useful to perform busy detection either in an effort t
; detect hangup or for detecting busie

;busydetect=ye

; On trunk interfaces FXS) it can be useful to attempt to follow the
 progres
```

```
; of a call through RINGING, BUSY, and ANSWERING.   If turned on, cal
; progress attempts to determine answer, busy, and ringing on phon
 lines
; This feature is HIGHLY EXPERIMENTAL and can easily detect fals
 answers
; so don't count on it being very accurate.  Also, it is ONLY configure
 fo
; standard U.S. tones.  This feature can also easily detect fals
 hangups
; The symptoms of this is being disconnected in the middle of a call fo
 n
; reason

;callprogress=ye

; Select which class of music to use for music on hold.  If not specifie
; then the default will be used

;musiconhold=defaul

;PRI channels can have an idle extension and a minunused number.  So
 lon
; as at least"minunused" channels are idle, chan_zap will try to call
;"idledial" on them, and then dump them into the PBX in the "idleext"
; extension (which is of the form exten@context).  When channels ar
 neede
; the"idle" calls are disconnected (so long as there are at least "minidle"
; calls still running, of course) to make more channels available.  Th
; primary use of this is to create a dynamic service, where idle channel
; are bundled through multilink PPP, thus more efficiently utilizin
; combined voice/data services than conventional fixed mappings/muxings

;idledial=699
;idleext=6999@dialou
;minunused=
;minidle=

; Configure jitter buffers in zapata (each one is 20ms, default is 4

;jitterbuffers=

; Each channel consists of the channel number or range.  I
; inherits the parameters that were specified above its declaratio

;callerid"Green Phone"<(256) 428-6121>
;channel > 1
;callerid"Black Phone"<(256) 428-6122>
;channel > 2
;callerid"CallerID Phone" <(256) 428-6123>
;callerid"CallerID Phone" <(630) 372-1564>
;callerid"CallerID Phone" <(256) 704-4666>
;channel > 3
;callerid"Pac Tel Phone" <(256) 428-6124>
;channel > 4
;callerid"Uniden Dead" <(256) 428-6125>
;channel > 5
;callerid"Cortelco 2500" <(256) 428-6126>
;channel > 6
;callerid"Main TA 750" <(256) 428-6127>
;channel > 44

; For example, maybe we have some other channel
; which start out in a different context and us
; E& M signalling instead.

;context=remot
;sigalling=e
;channel > 15
;channel > 16

;signalling=em_w
```

```
;
; All those in group 0 I'll use for outgoing calls
;
; Strip most significant digit (9) before sending
;
;stripmsd=1
;callerid=asreceived
;group=0
;signalling=fxs_ls> 45

;signalling=fxo_ls
;group=1
;calle"Joe Schmoe" <(256) 428-6131>
;channel > 25
;callerid"Megan May" <(256) 428-6132>
;channel > 26
;callerid"Suzy Queue" <(256) 428-6233>
;channel > 27
;callerid"Larry Moe" <(256) 428-6234>
;channel > 28

; SamplePRI (CPE) config:  Specify the switchtype, the signalling as
; either pri_cpe or pri_net for CPE or Network termination, and generall
; you will want to create a single"group" for all channels of the PRI.

; switchtype = nationa
; signalling = pri_cp
; group =
; channel > 1-23

signalling = pri_cpe
switchtype=dms100
group=1
context=main
cha> 1-23

signalling=fxo_ks
context=inside
chan>25-28
```

Example

The following example sets up four zaptel channels with user names and caller id information.

```
signalling=fxo_l
group=
callerid"Joe Schmoe" <(256) 428-6131>
channel > 25
callerid"Megan May" <(256) 428-6132>
channel > 26
callerid"Suzy Queue" <(256) 428-6233>
channel > 27
callerid"Larry Moe" <(256) 428-6234>
channel > 28
```

Vertical Service Activation Codes

The following activation codes are available with analog telphones operating on Zaptel interfaces.

```
*0 Flash external trunk on bridged channel
*67 DisableCaller ID for next outgoing call (per call blocking).
*69 Call return. Dials number of last caller if caller ID was present
*70 Disable call waiting for the next call or until hangup
*72 Cancel call forwarding
*73 Enable call forwarding
*78 Enable do not disturb
```

```
*79 Disable do not disturb
*80 Blacklist the caller who called previously (IfCaller ID was
 present)
*82 Enable caller ID on a line with per-line blocking
```

Transferring a Call and 3-Way Calling

To transfer a call from an analog phone on a ZAP channel,

```
hook flash (On some phones, press the R button), this puts call 1 on hold
dial tone is playe
dial another end poin
talk to that extensio
hook flash agai
```

This creates a 3-way call. You can stay on the 3-way call. If the line is enabled in the dial plan, hanging up will leave the other two parties on the call. If call transfer isn't enabled for the line, hanging up wil disconnect all parties.

ChapterÂ 9 - IAX Configuration

Asterisk servers or Asterisk devices like IAX telephones can connect to remote Asterisk systems with Inter Asterisk Exchange (IAX.) IAX allows calls to be switched between Asterisk systems or devices In addition, IAX allows dial plans to be shared, combined or centralized.

IAX is a community effort, not a standardization effort. Why was a new proprietary protocol developed? IAX supports the following functions that are not available with SIP or H.323.

Interoperability with NAT/PAT/Masquerade firewalls: IAX seamlessly interoperates through all sorts ofNAT and PAT and other firewalls, including the ability to place and receive calls, and transfer calls to other stations.

IAX uses a single UDP port. IAX uses port 5036 and IAX2 uses port 4569. This assures that IAX works well withNAT.

High performance, low overhead protocol: When running on low-bandwidth connections, or when running large numbers of calls, optimized bandwidth utilization is imperative. IAX uses only 4 bytes of overhead

Internationalization support: IAX transmits language information, so that remote PBX content can be delivered in the native language of the calling party.

Remote dial plan polling: IAX allows a PBX or IP phone to poll the availability of a number from a remote server. This allowsPBX dial plans to be centralized.

Flexible authentication: IAX supports cleartext, md5, and RSA authentication, providing flexible security models for outgoing calls and registration services.

Multimedia protocol: IAX supports the transmission of voice, video, images, text, HTML, DTMF, and URL's.

Call statistic gathering: IAX gathers statistics about network performance (including latency and jitter, as well as providing end-to-end latency measurement.

Call parameter communication: Caller*ID, requested extension, requested context, etc. are all communicated through the call.

Single socket design: IAX's single socket design allows up to 32768 calls to be multiplexed.

Outgoing Calls to a Remote Server with IAX

One Asterisk machine functions as an IAX server, the other Asterisk device functions as an IAX client. In this example, an IAX user on the client wishes to make an outbound call through the IAX server The call is sent from the IAX client to the IAX server. The call can then be dialed out from the IA server to theInternet or the PSTN. The dial plan of the server manages the call.

The server must have an appropriate entry in *iax.conf* that accepts and switches the incoming calls. This configuration uses the*trusted* context in the server dial plan to process the incoming IAX call.

```
[cpc]
type=frien
username=cp
secret=mysecre
context=truste
host=dynami
```

There are three client types

TABLE: 09-1 IAX client types	
type	**purpose**
user	incoming calls
peer	outgoing calls
friend	incoming and outgoing calls

The following entry in the *extensions.conf* file of the IAX client switches the call to the IAX server at *sip.iaxserver.com*. The variable *$(EXTEN)* holds the outgoing number the user dialed. The URL *sip.iaxserver.com* is resolved to the IP address of the IAX server the call will be sent to.

```
exten=> _1NXXNXXXXXX,1,Dial(IAX2/cpc:mysecret@sip.iaxserver.com/
 ${EXTEN}
```

Free calls can be made over the Internet between Asterisk machines with *iaxtel*. *Iaxtel* information is available at*www.iaxtel.com*. A registration at iaxtel.com provides a 700 area code telephone number usable within the*iaxtel* network. With this registration calls can be made to or from other iaxtel users. There are a few publicly available bridges from*iaxtel* to the PSTN.

The next example shows the configuration for outgoing calls with Voicepulse. Voicepulse can be found at*http://www.voicepulse.com*. Voicepluse is an IAX service provider in the eastern US. You can purchase an IAX connection from Voicepulse for incoming and outgoing calls. Voicepulse service can include a DID with a telephone number in many areas. Voicepulse provides long distance services at attractive rates.

```
exten > _1NXXNXXXXXX,1,DIAL(IAX2/loginID@voicepulse/${EXTEN})
```

IAX and a Mobile Client

If the client moves and appears on different ip addresses, the IAX client must register with the IAX server. The IAX registration informs the IAX server of the ip address of the IAX client. The IAX client registration statement is in the*[general]* section of the client *iax.conf* file.

Here are some registration examples.

```
; register withiaxserver.com
register > cpc:mysecret@sip.iaxserver.com

; Register witvoicepulse.com
register > vpuser:vpsecret@voicepulse.com

; Register with another IAX server
; server named tormenta, username marko and password secretpass
regist> marko:secretpass@tormenta.linux-support.net

; Register joe at remote host with no passwor

register > joe@remotehost:5656

; Register marko attormenta.linux-support.net using RSA key "torkey"

register > marko:[torkey]@tormenta.linux-support.net
```

In this example, the dial plan of the client has an entry that switches the incoming calls to the server named*cpc* and the context named tcom.

```
exten => 1833,1,Dial(IAX2/tcom:mysecret@cpc/${EXTEN})
```

As shown below, for the server to accept the incoming call, the server *iax.conf* file must include a context named *tcom*.

```
[tcom
type=frien
username=tco
secret=mysecre
context=defaul
host=dynami
```

Because the host is listed as dynamic, an IAX connection is opened whenever it is used. This connection will stay open across any NAT devices for the duration of a call.

Note that the IAX configurations at IAX client and the IAX host should correspond. For example, the following entry in the*[general]* context of both *iax.conf* files supports a low speed connection.

```
disallow=all
allow=gs
```

IAX Channels

Outgoing IAX channel names use the following format

```
IAX/[:]]@][:][/[@][/
```

```
]]
 - user name
 - authorization password
 - host to connect to
 - port at host
 - extension to dial
 - optional context at peer
 - a for autoanswer
```

Examples

```
IAX/mark:asdf@myserver/6275@defaul
IAX/iaxphone/s/
IAXguest@misery.digium.com
```

Incoming IAX channels use the following format

```
IAX[@]]/
 - username if known
 - apparent host making incoming connection
 - the local call number
```

Examples

```
IAX[mark192.168.0.1]/14 - call number 14 from mark at 192.168.0.1
IAX192.168.10.1]/13 - call 13 from 192.168.10.1
```

The [general] section of iax.conf

A section begins with the identifier in square brackets. The identifier should be an alphanumeric string

```
identifier
```

The section name of the first section of iax.conf must always be *general*.

The following commands are allowed in the general section of *iax.conf*.

```
port =
```

This sets the port that IAX binds to. The default IAX port number is 5036. Don't change this port number.

```
bindaddr =
```

This binds IAX to a specific local IP address instead of binding to all addresses. This can enhance security. For example, you might only wanted IAX to be available to users on your LAN.

```
bandwidth = [low|medium|high]
```

This selects codecs appropriate for the given bandwidth. The value high enables all codecs and is recommended only for 10Mbps or higher connections. A value of medium eliminates signed linear, Mulaw and A-law codecs, leaving only the codecs which are 32kbps and smaller with MP3 as a special case. A value of medium is useful with broadband connections. A value of low eliminates ADPC and MP3 formats and uses only the G.723.1,GSM, and LPC10.

```
allow = [gsm|lpc10|g723.1|adpcm|ulaw|alaw|mp3|slinear|all]
disallow = [gsm|lpc10|g723.1|adpcm|ulaw|alaw|mp3|slinear|all
```

The allow and disallow commands override the initial bandwidth selection on a codec-by-codec basis.

The recommended configuration is to select a low bandwidth and disallow the LPC10 codec. The LPC10 codec doesn't sound very good.

```
jitterbuffer = [yes|no]
dropcount =
maxjitterbuffer =
maxexcessbuffer =
```

These parameters control the operation of the jitter buffer. The jitter buffer should always be enabled unless you all your connections are over a LAN. The drop count is the maximum number of voic packets to allow to drop (out of 100). Useful values are 3-10. The maxjitterbuffer is the maximu amount of jitter buffer to permit. The maxexcessbuffer is the maximum amount of excess jitter buffe that is permitted before the jitter buffer is automatically shrunk to eliminate latency.

```
accountcode =
amaflags = [default|omit|billing|documentation
```

These affect call detail record generation. *Accountcode* sets the account code for records received with IAX. The account code can be overridden on a per-user basis for incoming calls *Amaflags* controls how a record is labeled and*omit* prevents a record from being written. *Billing* and *documentation* label the records as billing or documentation records.*Default* selects the system default.

```
tos = [lowdelay|throughput|reliability|mincost|none]
```

IAX can optionally set the TOS (Type of Service) bits to improve routing performance. The recommended value is *lowdelay*. Many routers, including any Linux routers with 2.4 kernels that have not been altered with ip tables, will give priority to these packets. This improves voice quality.

```
register => [:]@[:port]
```

Multiple *register* entries are allowed in the *general* section. Registration sends a remote Asterisk server the ip address of the IAX client. The remote Asterisk server must have a peer entry with the sam name and secret.

The <*name*> is a required field. It is the remote peer name that an IAX client uses to identify itself. A optional secret may be provided. The secret is a shared password between the IAX server and the IA client.

If the secret is in square brackets it is interpreted as the name of a key. The IAX client must have the private key*/var/lib/asterisk/keys/.key* and the IAX server must have the corresponding public key.

The *host* is a required field. It is the hostname or IP address of the IAX server. The port specification is optional and is by default 5036 if not specified. This should not be changed

User Sections of iax.conf

Users can be one of three types, user, peer or friend. A user type of *user* defines a connection for incoming calls. A user type of *peer* defines a connection for outgoing calls. A user type of *friend* defines a connection for both incoming and outgoing calls.

```
type = [user|peer|friend]
```

One or more context lines may be specified for a user. The context links the IAX configuration to the dial plan. A call coming in on this channel will be directed to the named context in*extensions.conf*

```
context =
```

Permit and deny rules may be applied to users, allowing them to connect from certain IP addresses and not others. The permit and deny rules are interpreted in sequence and all are evaluated on a given I address, with the final result being the decision.

```
permit = /
deny =/
```

For example

```
permit =0.0.0.0/0.0.0.0
den = 192.168.0.0/255.255.255.0
```

would deny anyone in 192.168.0.0 with a netmask of 24 bits (class C.) The following example denies no one because of the*permit* mask.

```
deny = 192.168.0.0/255.255.255.0
permi = 0.0.0.0/0.0.0.0
```

If no permit/deny rules are listed, it is assumed that someone may connect from anywhere.

```
callerid =
```

The callerid command overrides the Caller*ID information received from a user.

```
auth = [md5|plaintext|rsa]
```

Different authentication methods may be specified, and are separated by commas. If md5 or plaintext authentication is selected, a secret must be provided. If RSA authentication is specified, then one o more key names must be specified with*inkeys*. If no secret is specified and no authentication method is specified, then no authentication is required

```
secret =
```

The secret is the shared secret for md5 and plaintext authentication methods. Never use plaintext except when debugging

```
inkeys = key1[:key2...
```

Inkeys specifies the keys used to authenticate a remote peer. The key file is */var/lib/asterisk/keys/ .pub*. Public keys are not DES3 encrypted and do not need initialization.

IAX Connection Syntax in extensions.conf

At the time of writing, an IAX client can directly connect to an IAX server. No further redirection is allowed. That is, an IAX client cannot connect to an IAX server through another IAX server.

The IAX client calls the IAX server with a dial command in extensions.conf. This syntax is used for an IAX connection within a dial command in the client dial plan

```
IAX/[:]@][:][/[@][/
]]

user: UserID on remote peer or name of client configured in iax.conf
secret: Password
peer: Name of server to connect to
portno: Port number for connection on server
exten: Extension in the remote Asterisk server
context: Context to use in the remote Asteriskserver
options: Only 'a' is def 'request autoanswer'
```

Examples

```
    IAX/iaxphone/s/
This example above callsiaxphone and requests an immediate answer. The next example calls Digium.
    IAXguest@misery.digium.com
```

This next example makes a call to *myserver* using *mark* as username and *asdf* as password. This example connects to extension*6275* in the *default* context

```
IAX/mark:asdf@myserver/6275@defaul
```

If you are going to reference an IAX connection in multiple places, you may wish to create a global for the connection string. Please see the*iax.conf* example file for further information about IAX usage.

IAX Trunking

Inter-Asterisk eXchange trunk mode eliminates the IP overhead of individual VoIP IP streams by pipeliningRTP data from multiple calls into single (larger) packets.This removes the redundancy of IP overhead for eachRTP stream. This supports better bandwidth scaling. This mode is only useful for all the calls are between two specific Asterisk servers. This is frequently the case, for example betwee two branch offices or with a connection to a service like Voicepulse.

IAX2 supports PKI-style security and trunking. TDMoIP protocols other than Asterisk allocate bandwidth to keep all channels open. IAX trunking only uses the bandwidth needed for calls i progress.

Trunking requires that both sides are valid peers. Use a *register* statement to register with the systems you want to trunk with. Note that trunking requires that a timing source be available.

Sharing a Dial Plan

The switch command in *extensions.conf* connects dial plans between an IAX client and an IAX server. When a switch command is used, the connection between the IAX client and the IAX server is hel permanently open.

The switch statement in *extensions.conf* allows two Asterisk servers to share a dial plan. Here are several examples from the Wiki page.

Example 1

```
[default
exten > _801XXX,1,Goto,left|${EXTEN}|1
exten > _802XXX,1,Goto,right|${EXTEN}|1

[left]
exte> _801XXX,1,StripMSD,3
exten > _XXX,2,Goto,1
switch > IAX/left

[right]
exte> _802XXX,1,StripMSD,3
exten > _XXX,2,Goto,1
switch > IAX/left
```

and the same for right.

Example 2

In *extensions.conf*

```
[outbound
switch > IAX2/master:secret@iax-gw1.company.net/outbound

[slave]
type=user
auth=plaintext
context=outbound
context=outbound2 ; (can have multiple if you want)
secret=secret
host=dynamic
"slave"
trunk=ye
notransfer=ye
```

```
[slave]
type=peer
auth=plaintext
context=outbound-nuphone
secret=secret
host=dynamic
trunk=yes
notra
```

in extensions.conf:

```
[assigned-dids]
; uncomment a dial mechanism, first one goes to specific extensio
; other one goes to dial parameter s

;exten> 7046446999,1,Dial,IAX2/master@slave/${EXTEN}
;exten > 7046446999,1,Dial,IAX2/master@slave

machine slave iax.conf:

regist> slave:secret@iax-gw1.company.net

[master]
type=peer
hiax-gw1.company.net
secret=secre
context=outboun
trunk=ye
canreinvite=n

[master]
type=user
secret=secret
context=acontext
trunk=yes
canrein
```

This example in *iax.conf* forwards calls to another Asterisk server. The user and key must be specified in the iax.conf file of the called machine. A context named*servers* must appear at the calling machine in*extensions.conf*.

```
[iaxprovider
switch > IAX2/user:[key]@server/context
```

ChapterÂ 10 - Application Configuration

Voicemail

Asterisk voicemail provides many features including

- Password protection
- Separate away and unavailable greetings
- Default or custom greetings
- Multiple mail folders
- Web interface for checking of voicemail
- E-mail notification of voicemail with audio file attachment

Voicemail forwarding

Visual message waiting indicator

Message waiting stutter dialtone

Optionally play the CID of the caller heard before the voicemail

Optionally reach an operator after leaving a voicemail

Optionally review, rerecord, or save voicemails after leaving them

Optionally review, rerecord, or save busy, unavailable, and name prompts.

Optionally allow dialing out from within voicemail

Optionally allow calling back of the person who left voicemail

Several compression types are supported for storing voicemail. For voicemail messages forwarded to email, the first type named is used to compress the message.

```
[general]
; Default formats for writingVoicemail
;format=g723sf|wav49|wa
format=wav49|gsm|wa
```

The total number of voicemails that can be saved at your Asterisk system depends on your hardware and especially available disk space. It depends on the codec you select for compressing voice mail There can be additional overhead in voicemail from translations between the codec for the incomin call and the codec used to record the call.

Configuring Voicemail

The file */etc/asterisk.conf* holds voicemail related configuration settings. Consult the *voicemail.conf* sample file shown below for additional information. The permissions of *voicemail.conf* must allow Asterisk to write to this file.

The directory */var/spool/asterisk/vm* holds voicemail related files, for example messages. This can be changed in*/etc/asterisk.conf*.

Two applications are used in *extensions.conf, voicemailmain* and *voicemail*. The voicemail application returns a -1 if a mailbox cannot be located, or if the caller hangs up. Otherwise, it returns a zero.

Calls are placed to a user. A user must have an extension. The user's extension is specified in *extensions.conf*. The extensions are specified within a context. Here, extension *1265* is included in the *main* context.

```
[main]
exten > 1265,1,Dial(ZAP/1,15)
```

Each user mailbox is configured in *voicemail.conf.* A user extension must be included within a context in*voicemail.conf*. In this example, extension 1265 is included in the voicemail context named *main*. Note that the context names must be the same, in this example*main*, in *extensions.conf* and *voice-mail.conf* for voicemail to work correctly.

```
[main]
4008 > 2624,Joe User
```

You must create an empty voicemail box for each user. Edit the file *voicemail.cnf* to create a new mailbox. Entries for users appearing in *voicemail.conf* have the syntax

```
=,,,,

password - the numeric password for accessing the mailbox, for example
 1234

name - a user name, for example Bill

email - if email is specified a copy of the message will be sent to this
  address via email. Not that this means email must be configured properly for the Linux server runniinstance
of Asterisk.

pager_email - a second e-mail address to which a pager notification may
  be forwarded

options - not yet i
```

Make sure you do not have any spaces around the extension and password. Here is an example voice-mail configuration with one voicemail box specified at the end of the example.

```

;Voicemail Configuration

[general
; Default formats for writingVoicemail
;format=g723sf|wav49|wa
format=wav49|gsm|wa
; Who the e-mail notification should appear to come fro
serveremail=asteris
;serveremailasterisk@linux-support.net
; Should the email contain the voicemail as an attachmen
attach=ye
; Maximum length of a voicemail messag
maxmessage=18
; Maximum length of greeting
;maxgreet=6
; How many miliseconds to skip forward/back when rew/ff in message playbac
skipms=300
; How many seconds of silence before we end the recordin
maxsilence=1
```

```
; Silence threshold (what we consider silence, the lower, the more sensitive
silencethreshold=12
; Max number of failed login attempt
maxlogins=

; Skip th"[PBX]:" string from the message title
;pbxskip=ye
; Change the From: strin
;fromstring=The AsteriskPBX
; Change the email body, variables: VM_NAME, VM_DUR, VM_MSGNUM, VM_MAILBOX
  VM_CALLERID, VM_DAT
;emailbody=Dear ${VM_NAME}:\n\n\tjust wanted to let you know you were just left
 ${VM_DUR} long message (number ${VM_MSGNUM})\nin mailbox ${VM_MAILBOX} fro
 ${VM_CALLERID}, on ${VM_DATE} so you might\nwant to check it when you get
 chance.  Thanks!\n\n\t\t\t\t--Asterisk\

;
; Users may be located in different timezones, or may have different
; message announcements for their introductory message when they enter
; the voicemail system. Set the message and the timezone each user
; hears here. Set the user into one of these zones with the tz= attribute
; in thes field of the mailbox. Of course, language substitution
; still applies here so you may have several directory trees that have
; alternate language choices.
;
; Look in /usr/share/zoneinfo/ for names of timezones.
; Look at the manual page for strftime for a quick tutorial on how the
; variabstitution is done on the values below.
;
; Supported values:
; 'filename'    filename of a soundfile (single ticks around the filename
 required)
; ${VAR}        variable substitution
; A or a        Day of week (Saturday, Sunday, ...)
; B or b or h   Month name (January, February, ...)
; d     numeric day of month (first, second, ..., thirty-first)
; Y             Year
; I or l        Hour, 12 hour clock
; H             Hour, 24 hour clock (single digit hours preceded "oh")
; k             Hour, 24 hour clock (single digit hours NOT preceded by"oh")
; M             Minut
; P or p        AM or P
; Q            "today", "yesterday" or ABdY (*note: not standard strftime value)
; q            "" (for today), "yesterday", weekday, or ABdY (*note: not standard
 strftime value
; R             24 hour time, including minut


[zonemessages
eastern=America/New_York|'vm-received' Q 'digits/at' IM
central=America/Chicago|'vm-received' Q 'digits/at' IM
central24=America/Chicago|'vm-received' q 'digits/at' H 'digits/hundred'
 'hours

; Mailboxes may be organized into multiple contexts for
; voicemail virtualhosting
;
; Each mailbox is listed in the =,,,,
; if the e-mail is specified, a message will be sent when a message i
; received, to the given mailbox. If pager is specified, a message will be sen
  there as well

4200 > 9855,Mark Spencer,markster@linux-support.net,mypager@digium.com,attach=no|serveremail=myaddy@digium.
com|tz=central

[other]
400> 4008,Firstname Lastname
```

Note that the location of saved messages depends on the voicemail context. The base directory for voicemail is specified in*asterisk.*

conf.

```
/var/spool/asterisk/voicemail/YourVoicemailContext/210/INB
```

Voicemail Tree

Here is an outline of the commands available with *VoicemailMain.*

```
1 Read voicemail messages
3 Advanced options
1 Reply
2 Call back(1)
3 Envelope
4 Outgoing call(1)
5 Repeat current message
6 Play next message
7 Delete current message
8 Forward message to another mailbox
9 Save message in a folder
* Help; during msg playback: Rewind
# Exit; during mkip forward
2 Change folders
0 Mailbox options
1 Record your unavailable message
2 Record your busy message
3 Record your name
4 Change your password
* Return to the main menu
* Hel
```

After an incoming message, busy message, unavailable message, greeting, or name has been recorded, the following commands are available.

```
1 - Accept
2 - Revie
3 - Re-recor
0 - Reach operator(1) (not available when recording greetings/name
```

During the playback of a voicemail message, press # to fast forward or * to rewind. The setting of *skipms* determines the length of the skip in milliseconds. This is set in *voicemail.conf* and defaults to 3000 ms.

Calling in for Voicemail

The following commands in the dial plan will allow a user to type * and an extension to connect to a mailbox. This example assumes that extensions are three digits from 100 to 199.

```
exten => _*1XX,1,Voicemail(u${EXTEN:1})
exten > _*1XX,2,Hangup
```

If voicemail mailbox IDs and extension numbers are the same, the following commands in extensions.con will allow users to access their mailbox directly.

```
exten => 199,1,VoicemailMain(s${CALLERIDNUM})
exten > 199,2,Hangup
```

The following entry in *extensions.conf* will send a caller to voicemail when the zero key is pressed. Note this uses a lower case letter o.

```
# lower case letter o
# after an extension is reached, pressing zer
# starts voicemai
exten > o,1,voicemailmain
```

Resetting the Password

The following commands change the user's voicemail password.

- * dial VoiceMailMain
- * enter 0
- * enter 4
- * change the password
- * confirm the new password

The Directory Command

Including a directory command in *extensions.conf* provides a directory for callers. When a caller presses the correct key, they will hear instructions for searching a directory of users. With the following command in your dial plan, when the user presses seven they will hear the directory instructions

```
exten => 7,1,Directory(main)
```

The directory command looks in *voicemail.conf* for a list of extensions. The directory command does not by itself read any names to the caller. The argument given here,*main*, names the context in *voice-mail.conf* where the directory command looks for a list of extensions. Note that when the user selects an extension found in*voicemail.conf* their call will forward to that extension found the same context, in this case*main*, in *extensions. conf*. The context name must be the same in s*ip.conf, extensions. conf and voicemail.conf*for voicemail and directory services to work properly.

Web Interface to Voicemail

A perl script */usr/src/asterisk/vmail.cgi* is included in the source distribution. The command *make install* does not install the interface. Run*make webvmail* to create the interface. This is a perl script and requires that perl and the perl-suidperl packages are installed. You will need a web server running o the Aterisk server.

You may have to modify the script to get it working for your installation.

Don't forget to make the script executable.

```
chmod +x vmail.cgi
```

Sending Voicemail as Email

You can forward voicemail to an email account by adding an email address to voicemail.conf. Here is an example,

```
[other]
4008 > 4008,Firstname Lastname,yourname@company.com
```

Linux must be configured to forward mail. If you are using *smail*, make sure that it is turned on at boot time. For example, with the Mepis Debian release you will need symbolic links that cause*smail* to start

```
ln -s /etc/init.d/smail /etc/rc3.d/S85smai
ln -s /etc/init.d/smail /etc/rc5.d/S85smai
```

Edit */etc/smail/config* to reference the proper SMTP server where mail is to be sent, in this example *yourdomain.com*.

```
visible_nameyourdomain.com
```

Lastly you can start *smail* with the command

```
/etc/init.d/smail star
```

Configuring musiconhold.conf

The mp3 player that ships with your distribution may not work with Asterisk. you may have to replace it with another mp3 utility. Note that you will need a timing source for music on hold to work.

Music on hold, as any other application, is accessed from the dial plan and configured in extensions.conf. Here is an example,

```
exten => 6789,1,Answer()
exten > 6789,2,MusicOnHold(mymusic)
```

You must modify *musiconhold.conf.* Here is an example.

```
; Music on hold class definitions
[classes
default > quietmp3:/var/lib/asterisk/mohmp3
mymusic > quietmp3:/usr/share/mp3/mymusic
random-music > quietmp3:/usr/share/mp3/mymusic,-z
loud-music > mp3:/usr/share/mp3/mymusic
```

The quietmp3 directive automatically levels music to listenable levels. The *-z* option plays songs randomly rather than sequentially.

US Copyright laws may not allow you to play unlicensed music on hold. You can get an inexpensive license to play copyrighted music from the BMI library of over 4.4 million songs. More information i available athttp://www.bmi.com.

Recording Sound Files

Asterisk sounds are found in */var/lib/asterisk/sounds*. The format of these files is gsm. The Asterisk record command can be used to record sound files as described in the dial plan configuration chapter.

When recording new files in a studio for later use with Asterisk, try recording 8Khz, 16 bit wav files which will are likely to work better than 8 bit files. Then convert the wav files to gsm files. The Linu sox utility can convert files. Here is an example.

```
sox inputfile.wav -r 8000 -c 1 outputfile.gsm resample -ql
```

Quicktime for Windows will play back gsm files.

Configuring meetme.conf

It is very easy to configure meetme conferencing. With a meetme conference, any incoming calls are added to a conference. Note you will need a timing source for meetme conferencing to work.

First, add a conference id to *meetme.conf*

```
[rooms
conf > 123
```

The *MeetMe* command in *extensions.conf* provides access to a conference call.

```
MeetMe(confno[|options])
```

Add the MeetMe application to your dial plan. With the following lines in *extensions.conf,* callers to extension 18 are prompted for a conference number. If they enter*123* on the dial pad, they will be added to conference 123.

```
; Conferencin
exten > 18,1,Answer
exten > 18,2,Wait(1)
exten > 18,3,Meetme
```

Note that meetme conferencing requires trunking which implies an incoming T1 or E1. Trunking from the phone company allows successive incoming calls to be forwarded to the Asterisk server Without trunking, the second caller to the incoming number will receive a busy signal. You coul potentially work around this by providing incoming callers different telephone numbers.

The available options are

```
'm' -- set monitor only mode (user can only hear the audio, not participate)
'p' -- allow user to exit the conference by pressing '#
```

```
't' -- set talk only mode, user won't be able to hea
'v' -- video mod
'q' -- quiet mode (don't play enter/leave sounds
'd' -- dynamically add conferenc
'M' -- enable music on hold when the conference has a single calle
'b' -- run AGI script specified in ${MEETME_AGI_BACKGROUND}. Default i
 conf-background.agi (Zap channels only, does not work with non-Za
 channels in the same conference
```

You can configure your system to allow a user to join a conference but not speak with the *m* option. The*t* option allows a caller to speak but not listen!

The example below includes the conference number in the dial plan. In this case, Callers will not be prompted for a conference number, they will be automatically directed to conference 18. The argument*p* allows users to exit the conference by pressing # on the telephone keypad.

```
exten => 999,1,MeetMe(123|p)
```

Here are some additional examples.

```
exten > 998,1,MeetMe(999|mp)  ;caller dials 998 and can only hear audio,
 not spea
exten > 997,1,MeetMe(999|tp)  ;caller dials 997 and can only speak to
  conf. but can't hear i
```

Password protect a meeting by adding a password in *meetme.conf*.

```
conf > ROOMNO,PASSWRD
```

For example,

```
conf => 100,54321
```

Note the MeetMe application must be able to access a Zaptel timer. No timer is installed by default if there is no Digium Zaptel hardware interface card installed.

The return value of this application is always -1.

You can play an announcement to those joining a conference by adding the following to the dial command.

```
'A(x)' -- play an announcement to the called party, using x as file
```

Further information is available from the sample configuration files and from *http://www.voipinfo.org/wiki-Asterisk+cmd+MeetMe*.

Fax

Facsimile requires a lossless codec like G.711 ULAW. Fax will not work with lossy codecs like GSM. Compression removes portions of the audio spectrum that people can't hear but that fax transmissio relies upon.

Asterisk can, in the dial plan, accept an incoming fax. When a call is answered with the answer() command, Asterisk will listen for beeping. You will need to add additional third party software to process the incoming fax transmission.

SoftFax software is available at *ftp://ftp.opencall.org/pub/spandsp/*. The instructions for SoftFax are available at*http://www.opencall.org/instructions.html*. Hylafax software may be of use.

Scott Laird has posted an excellent example to the Asterisk mailing list. This can be found at *http:// lists.digium.com/pipermail/asterisk-users/2004-March/041408.htm*l and is shown here. This example will receive a fax and identify it with a unique ID.

```
In case anyone's interested, I spent a bit of time on incoming faxes
yesterday, prototyping a DID FAX-type setup. Here are a few snippets
in case anyone's interested

[macro-faxreceive]
  exte> s,1,SetVar(FAXFILE=/var/spool/asterisk-fax/${UNIQUEID}.tif)
  exten > s,2,DBGet(EMAILADDR=extensionemail/${MACRO_EXTEN})
  exten > s,3,rxfax(${FAXFILE})
  exten > s,103,SetVar(EMAILADDR=defaultuser@example.com)
  exten > s,104,Goto(3)

[fax]
  exte> 2201,1,Macro(faxreceive)
  exten > 2202,1,Macro(faxreceive)
  exten > 2203,1,Macro(faxreceive)

  exten> h,1,system(/usr/local/sbin/mailfax ${FAXFILE} ${EMAILADDR} \
    "${CALLERIDNUM} ${CALLERIDNAME}")

; I'm using a shared analog line for testing this, so I'm using the fax
; autodetection code to yank faxes out of IVR and into the 'fax'
; pseudo-extensio
[outside
  ..
  exten > fax,1,Goto(fax,2201,1)


Finally, here's /usr/local/sbin/mailfax:

#!/bin/sh

FAXFILE=$1
RECIPIENT=$2
FAXSENDER=$3

tiff2ps -2eaz -w 8.5 -h 11 $FAXFILE |
  ps2pdf - |
  mime-construct --to $RECIPIE"Fax from $FAXSENDER"
--attachment fax.pdf --type application/pdf --file In Debian, tiff2ps comes in libtiff-tools, ps2pdf is part o
Ghostscript, and mime-construct is its own package
To set the email address associated with each extension, do 'databas
put extensionemail EXTENuser@example.com'
```

Call Parking

If you are in your office on a support call. You want to transfer the call from your office to the computer room phone. You can park the call, hang up your office phone, go to the computer room and then pic up the call there. You can transfer a call to a special extension where the call is parked. The call is hel at that extension until you pick it up again, the caller hangs up, or the call times out.

Call parking is configured with the file parking.conf. Here is an example.

```
[general]
parkext > 701 ; dial this extension to park the call
parkpos > 702-720 ; extensions to park calls
context > parkedcalls ; context for parked
parkingtime > 1355 ; Time limit for parked calls (default is 45 seconds)
```

Be very careful with the parking context. Only allow authorized users to use parking. You don't want an outside caller to be able to park calls.

Here is a sample to create a parking context that includes two extensions in *extensions.conf*.

```
[parking
exten > 1,1,Dial(SIP/phone1,20,tr)
exten > 2,1,Dial(SIP/phone2,20,tr)
```

In this example, the *t* after the time of twenty seconds allows calls to be transferred. The r lets the calling party know the extension is ringing. Using a capital *T* will allow the calling party to park calls.

During a call press # to park that call. You will hear a voice say "transfer." Dial the number of the parking extension, in the example above extension 701. If you dial the parking extension quickly enough, you will hear a voice prompt of the extension the call is parked on.Dial that extension from any phone in the parking group to retrieve the call.

ChapterÂ 11 - Run and Manage Asterisk

While running, Asterisk provides a command line interface. Commands may be given to examine or control a running Asterisk system.

If an application is already using your sound device, Asterisk may not process sounds properly. For example, if xmms, mplayer, xine, esd or other similar applications are already running, Asterisk ma have problems with sound playback.

To start Asterisk manually from the command line, open a command prompt and enter the command

```
asterisk -vvvv
```

The string of *v* characters specifies verbose messages. The option *c* opens a console, the lower case *p* options specifies a real-time priority.*Asterisk -vvvvc*p displays all possible debugging information. After this command, Asterisk will start and the console will start.

After Asterisk successfully starts, you will be left at the Asterisk command prompt. To stop Asterisk, enter the comman

```
stop no
```

The option *-f* will prevent Asterisk from forking a separate process. This is useful when starting Asterisk with an entry in inittab. For example,

```
ax:2345:respawn:/usr/sbin/asterisk -vvvcf
```

Starting Asterisk from *init* will cause Asterisk to automatically restart if it stops for any reason.

Running the Simple Configuration

When Asterisk is successfully running, from extension 4035 dial extension 4009. Many messages should display at the Asterisk console. The phone at extension 4009 should ring.

Confirm at the console that the client phones are registering with the Asterisk server. If the call is immediately directed to the"busy" message, a phone client has most likely not registered with the Asterisk server. If the registry interval is set to more than fifteen seconds, it will take at least fifteen seconds after Asterisk starts before calls can be placed to a client telephone.

Connecting to a Running Asterisk Instance

If Asterisk is already running, the r command will attach to that running instance. Any other commands you may wish to use must be included. For example, if you want to attach to a running Asterisk server with verbose output from the command prompt use the comman

```
asterisk -vvvv
```

To end the session without stopping Asterisk, use the *exit* command.

Reattaching to Asterisk

To reattach to a running Asterisk server, from a command prompt use the command

```
/usr/sbin/asterisk -
```

Exit the Console

Sending a SIGINT, typically by typing control-c, will stop the Asterisk console. If Asterisk is not running as a background process, this will stop Asterisk.

If you start Asterisk as a background process, either from a startup script or from the command prompt, you can reattach to asterisk with the command*asterisk -r*. After you have reattached to Asterisk when it is running as a background process, the *exit* command will exit the console without stopping Asterisk.

Asterisk Command Arguments

The following arguments are available when starting Asterisk with the Asterisk command.

```
-c - Enables console mode. If console mode is enabled, Asterisk will provide
     a command line that can issue commands and view the state o
     the system. Implies -f as wel
-C  - Executes Asterisk with a different configuration file.
-d - Enables extra debugging across all modules.
-f - Prevents Asterisk from daemonizing into the background.
-g - Forces Asterisk to dump core after a segmentation violation.
-h - Displays basic command line help.
-i - Forces Asterisk to prompt for cryptographic initialization passcodes at startup.
-n - Disables ANSI color support.
-p - Run with a real-time priority.
-q - Run in quiet mode.
-v - Runs Asterisk in verbose mode. More v's produce more verbose output.
-x  - Executes a command in Asterisk (when combined with -r)
```

Connecting to a Running Instance

```
-r:  Connect to Asterisk running in the background and present a command
 line interfac
-x:  In combination with -r, execute an Asterisk CLI comman
-n:  Disable ANSI colour suppor
```

Asterisk Commands

The following commands are available from the asterisk command line

! Â	Execute a shell command
abort halt Â	Cancel a running halt
add extension Â	Add new extension into context
add ignorepat Â	Add new ignore pattern
add indication Â	Add the given indication to the country
answer Â	Answer an incoming console call
autoanswer Â	Sets/displays autoanswer
database del Â	Removes database key/value
database deltree Â	Removes database keytree/values
database get Â	Gets database value
database put Â	Adds/updates database value
database show Â	Shows database contents
debug channel Â	Enable debugging on a channel
dial Â	Dial an extension on the console

dont include Â	Remove a specified include from context
dump agihtml Â	Dumps a list of agi command in html format
exit Â	Exit Asterisk
extensions reload Â	Reload extensions and *only* extensions
hangup Â	Hangup a call on the console
help Â	Display help list, or specific help on a command
iax2 debug Â	Enable IAX debugging
iax2 no debug Â	Disable IAX debugging
iax2 set jitter Â	Sets IAX jitter buffer
iax2 show cache Â	Display IAX cached dialplan
iax2 show channels Â	Show active IAX channels
iax2 show peers Â	Show defined IAX peers
iax2 show registry Â	Show IAX registration status
iax2 show stats Â	Display IAX statistics
iax2 show users Â	Show defined IAX users
iax2 trunk debug Â	Request IAX trunk debug
iax debug Â	Enable IAX debugging
iax no debug Â	Disable IAX debugging
iax set jitter Â	Sets IAX jitter buffer
iax show cache Â	Display IAX cached dialplan
iax show channels Â	Show active IAX channels
iax show peers Â	Show defined IAX peers
iax show registry Â	Show IAX registration status
iax show stats Â	Display IAX statistics
iax show users Â	Show defined IAX users
include context Â	Include context in other context
init keys Â	Initialize RSA key passcodes
load Â	Load a dynamic module by name
logger reload Â	Reopens the log files
logger rotate Â	Reopens the log files
mgcp audit endpoint Â	Audit specified MGCP endpoint
mgcp debug Â	Enable MGCP debugging
mgcp no debug Â	Disable MGCP debugging
mgcp show endpoints Â	Show defined MGCP endpoints
no debug channel Â	Disable debugging on a channel

pri debug span Â	Enables PRI debugging on a span
pri intense debug sp Â	Enables REALLY INTENSE PRI debugging
pri no debug span Â	Disables PRI debugging on a span
quit Â	Exit Asterisk
reload Â	Reload configuration
remove extension Â	Remove a specified extension
remove ignorepat Â	Remove ignore pattern from context
remove indication Â	Remove the given indication from the country
restart gracefully Â	Restart Asterisk gracefully
restart now Â	Restart Asterisk immediately
restart when Â convenien	Restart Asterisk at empty call volume
send text Â	Send text to the remote device
set verbose Â	Set level of verboseness
show agents Â	Show status of agents
show agi Â	Show AGI commands or specific help
show applications Â	Shows registered applications
show application Â	Describe a specific application
show audio codecs Â	Shows audio codecs
show channel Â	Display information on a specific channel
show channels Â	Display information on channels
show codecs Â	Shows codecs
show codec Â	Shows a specific codec
show conferences Â	Show status of conferences
show dialplan Â	Show dialplan
show image codecs Â	Shows image codecs
show image formats Â	Displays image formats
show indications Â	Show a list of all country/indications
show keys Â	Displays RSA key information
show locals Â	Show status of local channels
show manager command Â	Show manager commands
show manager connect Â	Show connected manager users
show modules Â	List modules and info
show parkedcalls Â	Lists parked calls

show queues Â	Show status of queues
show switches Â	Show alternative switches
show translation Â	Display translation matrix
show uptime Â	Show uptime information
show version Â	Display version info
show video codecs Â	Shows video codecs
sip debug Â	Enable SIP debugging
sip no debug Â	Disable SIP debugging
sip show channels Â	Show active SIP channels
sip show channel Â	Show detailed SIP channel info
sip show inuse Â	List all inuse/limit
sip show peers Â	Show defined SIP peers
sip show registry Â	Show SIP registration status
sip show users Â	Show defined SIP users
skinny debug Â	Enable Skinny debugging
skinny no debug Â	Disable Skinny debugging
skinny show lines Â	Show defined Skinny lines per device
soft hangup Â	Request a hangup on a given channel
stop gracefully Â	Gracefully shut down Asterisk
stop now Â	Shut down Asterisk imediately
stop when convenient Â	Shut down Asterisk at empty call volume
transfer Â	Transfer a call to a different extension
unload Â	Unload a dynamic module by name
zap destroy channel Â	Destroy a channel
zap show channels Â	Show active zapata channels
zap show channel Â	Show information on a channel

Starting and Stopping Asterisk Automatically

With Redhat Linux, copy the script */usr/src/redhat/asterisk/redhat* to */etc/init.d*. Then run the command

```
chkconfig asterisk on
```

Asterisk will now start automatically when you reboot Linux.

Don't install Asterisk to start automatically until you are comfortable with your Asterisk configuration. While you are learning, you will want to start and stop Asterisk many times manually from the command line.

There are open source tools available at *http://cr.yp.to/daemontools.html* that help manage Unix processes. You can use these tools to automatically start Asterisk if it fails.

Starting Asterisk using safe_asterisk

Another script is available for starting Asterisk. This script attempts to keep Asterisk running. Start Asterisk as a daemon with the safe_asterisk script located at

```
/usr/sbin/safe_asterisk
```

Echo Suppression

Echo can ruin a telephone conversation. A caller expects to hear their own voice as they are talking. It is annoying if they hear their voice with a delay of more than about 25 ms. Long or loud echo can b intensely annoying.

Start by finding the source of the echo. Echo is best eliminated at the source.

In the PSTN, echo is commonly caused by impedance mismatches between the four-wire network and the two-wire local loop. A hybrid is the interface where a two wirePOTS line divides into four wires with two lines for transmit and two lines for recieve. The hybrid circuit makes it possible to transmi two channels of information in opposite directions on a single pair of wires. Echo is often created by a unbalanced hybrid at thePSTN to TDM interface. When installed properly, the hybrid should subtract some of the transmitted signal from the received signal. This will remove any echo from the signal that is caused by a local loopback of the transmitted signal. The PSTN tightly controls impedance matching and uses echo cancellers.

Echo commonly occurs when the hybrid is installed wrong or damaged in some way. This can make the impedance of thePOTS line unpredictable. If echo is caused by such an impedance problem, only the near-end user will hear it. Such near-end echo can be easily removed by repairing the physical circuit.

The far end user has no such repair available to them. The echo can still be removed with signal processing. Echo suppression algorithms can remove echo.

Echo can be caused by IRQ problems with installed Zaptel board. If this is the case, turn off the automatic BIOS detection and IRQ assignments, turn off any unneeded hardware and assign the IRQs manually. This information was found the hard way by a major

California VoIP company, Race Technologies, Inc. at*www.race.com*.

Echo suppression algorithms typically sample the actual signal and then sample again after a small time delay. One or more delayed samples can be weighted and then subtracted from the incoming signal. Different echo cancellation algorithms are available that use different sampling and weighting criteria.

Echo suppression algorithms will never be as effective as eliminating echo at the source by balancing the hybrid. Asterisk includes several echo cancellers.

Acoustic echo can be caused by feedback between a headset or handset microphone and speaker. Replacing the handset with better equipment can cure this problem.

Echo cancellation can be built into hardware or software. Echo cancellation done by a hardware Digital Signal Processor (DSP) in the telephone is more effective than software echo cancellation.

Managing Asterisk

Managing Asterisk of course means managing Linux. This book assumes that you are already familiar with Linux administration. You may want to use a GUI client like gastman or astman monitor you Linux installation. You should regularly monitor the size of the log files in**/var/log/asterisk**.

For quality of service, you should separate your PC network from your VoIP network. At least separate them logically at layer three. You may want to isolate them physically.

Ensure a stable Asterisk installation by using a staging server. Test any new release on the release server before placing it into production. Changes to*extension.conf* can easily break your Asterisk server. Be careful to keep backups of your configuration files. This will allow you to revert to a working state.

Use the latest 1.0 release version rather than the latest development version.

Regularly, perhaps once a week, stop and start your Asterisk server. A restart is not as effective. If you have configured Asterisk for automatic startup, a cronjob can stop and start the machine and Asterisk.

Add a provision to your startup scripts to detect and restart a hung Asterisk server. Daemontools can help you accomplish this.

Regularly telnet into your Asterisk server to make sure it is still running. Tools like mon (cache), big monit, brother, big sister and nagios can help you monitor your Asterisk server.

Don't use mpg123 for music-on-hold, or be prepared to kill hung mpg123 threads. Often mpg123 won't terminate after Asterisk is stopped. This will prevent Asterisk from restarting.

There should be, but currently is not, a quota on voice mailbox sizes. The alternative is to use the script described below that deletes all voicemail after a predetermined time.

Use a network sniffer to analyze your network traffic. Ethereal is an superb free product and has an available IAX plug-in.

Configure an *AbsoluteTimeout* value for calls that are charged. This will prevent a call of unlimited length if either Asterisk or a phone fails. Note thatSIP has limited facilities for detecting a disconnected client which can result in calls that do not hang up.

Carefully consider your hardware environment. Asterisk lends itself well to shared servers. Think about redundancy, load balancing or clustering. Stock any needed spare parts. Provide in advance fo timely hardware maintenance.

A T1 monitoring-switching device will let two Asterisk servers share a single T1 line. Should one server fail, the backup server will immediately take over although any calls in progress will be lost.

Backup, backup, and more backup. Backup your complete installation with a tool like Mondo Resuce. Backup all your Asterisk specific configuration files. If your installation calls for it, backup any voice-mail

Remote Management with SSH

It is very easy to manage an Asterisk system remotely. Use the utility of your choice to get to a command prompt on the remote system. Your most secure option is to communicate with SSH, which runs onTCP port 22.

To enable an SSH connection with Mepis, you will have to modify the file */etc/hosts.deny.* Comment out the denial line as shown below.

```
#ALL: PARANOID
```

Sharing a Remote Session

The Linux *screen* command will allow you to share what you are seeing with another user. The second user can connect to the server and arrive at a command prompt. The second user can then issue th screen command. The screen command will allow the second user to see in their command windo whatever is in your command window.

With Debian Linux, se the command

```
apt-get install screen
```

Use the Web site wwww.rpmfind.com to locate the rpm for Redhat Linux. Go to the Web site and search for*screen*. For example, screen can be found at

```
screen-3.9.13-5.i386.rp
```

To download this package right click on the link and copy the shortcut to get the address. Download the file and install it as follows.

```
cd /tmp
wgetftp://195.220.108.108/linux/redhat/9/en/os/i386/RedHat/RPMS/screen3.9.13-5.i386.rpm
  rpm -Uvh screen-3.9.13-5.i386.rp
```

Consult the manual page for the screen command for usage instructions.

Automatically Removing Old Voice Mail Messages

The expire-messages facility finds messages more than X days old and deletes them. expire-messages reorganizes every mailbox folder

Older messages have lower numbers. For example, msg0000 is older than msg0005. The expire-message routine deletes and then renumbers messages. File deletion is done with the find command.

If someone checks their voice mail during expire-message processing they may have a problem accessing messages. They may need to wait until the reorganization is finished before they will be able to access their voice mail. This is a good reason to expire messages in off hours.

When Should You Update Asterisk?

At the time of writing, version 1.0 of Asterisk is available. For a production system, you should use the most recent version of the 1.0 release, not a development branch. The Asterisk sources are rapidl changing. This includes bug fixes. You should get a newer version of the source if there is somethin broken in your system that the new release fixes.

Always thoroughly test any new release in a separate test environment before putting it into production. Infrequently, the most recent version of Asterisk may be broken. If you put a broken version into production you will have a broken production server and upset users.

Asterisk Security

Asterisk is a complex product that works in a complex environment. Security issues and securing your Asterisk server are very important. Some of these issues are addressed here

First, the network physical and network environments that the Asterisk server is in must be secure. The server must be physically secure and protected from intrusion or disaster including fire or flood The network that the Asterisk server is attached to must be secure. If the network becomes unavailable, the Asterisk server is unusable, even if it's not because of the Asterisk server itself.

As described earlier in this book, TCP ports may have to be opened for SSH and TFTP.

Firewall Setup

It is safer to run Asterisk behind a firewall. Here is a sample configuration for a Linux IPtables firewall.

```
#SIP on UDP port 5060. Other SIP servers may need TCP port 5060 as well
-A INPUT -p udp -m udp --dport 5060 -j ACCEP
# IAX2- the IAX protoco
-A INPUT -p udp -m udp --dport 4569 -j ACCEP
# IA
-A INPUT -p udp -m udp --dport 5036 -j ACCEP
#RTP:the media stream
-A INPUT -p udp -m udp --dport 10000:20000 -j ACCEP
# SSH?:Secure shell session
-A INPUT -p tcp -m tcp --dport 22 -j ACCEP
```

SIP Security

Asterisk implements SIP MD5 digest authentication. The MD5 algorithm produces 128-bit "fingerprint" or "message digest" of an input. The MD5 spec states, "It is conjectured that it is computationally infeasible to produce two messages having the same message digest, or to produce any message having a given prespecified target message digest. The MD5 algorithm is intended for digital signature applications, where a large file must be"compressed" in a secure manner before being encrypted with a private (secret) key under a public-key cryptosystem such as RSA"

Asterisk Configuration Security

Remove all unneeded modules from your Asterisk server. For example if you are only doing ZAP and SIP then specify noload= for MGCP and Skinny in *modules.conf*. This will streamline your system and reduce the risk of exploits

Don't allow users to login to your Asterisk server. Most recent kernel exploits required local user access. Don't allow file sharing or other user-services on your Asterisk server.

Extension contexts should isolate outgoing or toll services from any incoming extensions. Don't allow access to outgoing or toll services in contexts that are accessible from incoming channels. Configur your dial plan to isolate outgoing and toll service calls from any incoming connections.

Never allow outgoing toll services in the default context. Remove the *demo* context from the *default* context. Always include the"default" context within other private contexts with the command

```
include > default
```

Any channel that can enter an extension context that it has the capability of accessing any extension within that context is a potential problem. A channel or incoming line that is allowed to access a extension context where that extension context can in turn access any other context, can access an extension. This allows incoming calls to connect to outgoing services. This allows incoming callers t make free toll calls.

Here is an example secure configuration.

```
[longdistance]
exten > _91NXXNXXXXXX,1,Dial,Tor/g2/BYEXTENSION
include > local

[local]
exte> _9NXXNXXX,1,Dial,Tor/g2/BYEXTENSION
include > default

[default]
exte> 6123,Dial,Tor/1
```

Logging

The amount of logging is controlled by the file *logger.conf*. Here is an example.

```
debug => debug
;console > debug,notice,warning,error
console > notice,warning,error
;messages > notice,warning,error
messages > warning,error
```

Note that if you turn on full loggin int the messages or debug files, the log files will get very large very fast. When the log files exceed 2 GB, Asterisk will stop running. This can take just a few days on busy Asterisk server.

ChapterÂ 12 - Your First Configuration

You should start learning Asterisk with a very simple configuration. Getting a simple configuration running with your Asterisk server and your telephones will be a major step towards learning and usin Asterisk.

This chapter demonstrates a simple configuration for two SIP phones connected to an Asterisk server. This example assumes that the phones and the Asterisk server are on the same subnet and that there i no firewall between the phones and the server. Four files must be configured for this example,*sip.conf*, *zapata.conf*, *extensions.conf* and *voicemail.conf*.

Get this simple configuration working before attempting more complex configurations. Your goal for this configuration should be making a call from one phone to the other phone

The SIP phones could be hardware or soft phones. The example shows the configuration with Cisco 7960 telephones. You will have to learn how to configure your phones to work with this simple configuration.SIP phone configuration is not shown in this chapter.

This simple configuration will allow two phones networked to the Asterisk server to call each other. The example configuration supports the Digium four-portFXS board. The previous section on configuring voicemail shows how voice mail should be configured for this simple example.

Remember that a more complex set of sample Asterisk configurations are created by running the make command,

```
make samples
```

from a Linux command prompt while in the */usr/src/asterisk* directory. This simple configuration is less complex than the examples provided in*/usr/src/asterisk/configs*. You should read these Asterisk supplied samples to learn more about Asterisk configuration

The Network Environment

Running the Asterisk server on a separate subnet, or even better a separate physical network, will make your first configuration**much** easier. Consider starting with the Asterisk server, a hub or switch and twoSIP telephones.

Connect Asterisk and two IP phones to the network. Make sure the two IP phones are properly configured for SIP and Asterisk. Configuration for several manufacturer's phones and other SIP devices are described in other chapters. You can find help for telephone configuration through he Asteris mailing lists and archives, or from the telephone manufacturers support facilities. Make sure you ca ping the phones from the Asterisk server.

Go to the directory */etc/asterisk*. Save copies of the files *sip. conf*, *extensions.conf* and *voicemail.conf*. Replace the contents of these files with the configuration files found in the directory / simple-config o the CD. Be sure that the ownership and permissions for the configuration files remain unchanged.

This configuration allows two SIP phones to call each other. Unanswered calls will be connected to voicemail.Voicemail can be directly dialed, too.

Telephone Configuration

In this example, one SIP phone is going to call another SIP phone. You must configure any SIP phones before attempting to use them with Asterisk. This may require reloading a different firmwar image to the phone. Here is a simple configuration file for a Cisco 7960 that will work with the sampl Asterisk configuration. This file is sent to the telephone withTFTP.

```
# 7960 SIP Configuration File


image_version: P0S3-06-0-00

preferred_codec: g711ulaw

# Line 1  appearance
line1"4035"
# Line 1 display name, used for caller i
line1_displayname:"415-555-1212"
# Line 1 Registration Authenticatio
line1_authname:"4035"
# Line 1 Registration Passwor
line1_password:"cisco"
# Line 1 Short Nam
line1_shortname:"4035"
# Phone Label (Text desired to be displayed in upper right corner
phone_label:"CPC "  ; Has no effect on SIP messaging
# Line 1 Display Name (Display name to use forSIP messaging)
line1_displayname:"4035"


# Phone Prompt (The prompt that will be displayed on console and telnet)
# phone_prompt"SIP> " ; Limited to 15 characters
# Phone Password (Password used for console or telnet login
# phone_password:"cisco" ; Limited to 31 characters (Default - cisco)
# User classifcation used when Registerin
# [ none(default), phone, ip
user_info: non
```

sip.conf

Once the telephones are running the correct version of SIP and are configured correctly, configure Asterisk for these phones. You must modify*sip.conf* for use with the two phones. Here is a sample configuration file for two SIP telephones at extensions 4009 and 4035. This configuration directs calls on the incomingSIP channel to the *from-sip* context in the dial plan.

```
[general]
port = 5060 ; TheTCP/IP port for SIP communications

[4035]
type=friend           ; This device takes and makes calls
username=4035
secret=cisco
context=from-sip
ca"Bill" <415551212>
qualify=100
host=dynamic          ; This host is not on the same IP addr every tim
canreinvite=n
mailbox=4035           ; Activate the message waiting lightfor message
defaultip192.160.0.12
[4009
type=friend           ; This device takes and makes call
username=400
secret=cisc
context=from-si
callerid"Paul" <4155551212>
qualify=100
host=dynamic  ; This host is not on the same IP addr every tim
canreinvite=n
mailbox=4009  ; Activate the message waiting light for message
defaultip192.168.0.12
```

extensions.conf

The SIP call comes in over a SIP channel. The entry in sip.conf names a context in the dial plan. The call is processed by the instructions in the named context in the dial plan.

Here is the complete *extensions.conf* file for your simple configuration. This dial plan has two contexts, *default* and *from-sip*.

The context *from-sip* in the dial plan supports the two SIP telephones at extensions 4009 and 4035. There are two sets of entries, on set for each extension.

```
[general
static=yes       ; These two lines prevent the command-line interfac
writeprotect=yes ; from overwriting the config file. Leave them her

[default]
exte> 4035,1,VoicemailMain2

[from-sip]
; If the number dialed by the calling party w"4035", then
;Dial the user "4035" via the SIP channel driver. Let the number
; ring for 20 seconds, and if no answer, proceed to priority 2
; If the number gives a"busy" result, then jump to priority 102

exten > 4035,1,Dial(SIP/4035,20)
```

```
; Priority 2 send the caller to voicemail, and gives th"u"navailable
; message for user 4035, as recorded previously. The only way ou
; of voicemail in this instance is to hang up, so we have reache
; the end of our priority list

exten > 4035,2,Voicemail2(u4035)

; If the Dialed number in priority 1 above results in
;"busy" code, then Dial will jump to 101 + (current priority)
; which in our case will be 101+1=102. This +101 jump is buil
; into Asterisk and does not need to be defined

exten > 4035,102,Voicemail2(b4035)
exten > 4035,103,Hangup

;
; Now, what if the number dialed w"4009"?


exten> 4009,1,Dial(SIP/4009,20)
exten > 4009,2,Voicemail2(u4009)
exten > 4009,102,Voicemail2(b4009)
exten > 4009,103,Hangup

; Define a way so that users can dial a number to reach
; voicemail. Call the VoicemailMain application with the
; number of the caller already passed as a variable, so
; all the user needs to do is type in the password.
;

> 4040,1,VoicemailMain(${CALLERIDNUM})

; Tech Support at Digium
exte> 500,1,Dial(IAX2/guest@misery.digium.com/6161@default)     ;
 Call the Asterisk dem
exten > 500,2,Playback(demo-nogo)      ; Couldn't connect to the demo
 sit
exten > 500,3,Goto(s,6)                ; Return to the start over message.

; Four Lines foFXS board
exten > 6000,1,Dial(ZAP/25,20)
exten > 6000,2,VoiceMail2(u6000)
exten > 6000,3,Hangup
exten > 6000,102,VoiceMail2(b6000)
exten > 6000,103,Hangup

exten> 6001,1,Dial(ZAP/26,20)
exten > 6001,2,VoiceMail2(u6001)
exten > 6001,3,Hangup
exten > 6001,102,VoiceMail2(b6001)
exten > 6001,103,Hangup

exten> 6002,1,Dial(ZAP/27,20)
exten > 6002,2,VoiceMail2(u6002)
exten > 6002,3,Hangup
exten > 6002,102,VoiceMail2(b6002)
exten > 6002,103,Hangup

exten> 6003,1,Dial(ZAP/28,20)
exten > 6003,2,VoiceMail2(u6003)
exten > 6003,3,Hangup
exten > 6003,102,VoiceMail2(b6003)
exten > 6003,103,Hangup

exten> 8500,1,VoiceMailMain2
exten > 8500,2,Hangup

[local]
```

```
inclu> from-sip
```

This dial plan sets up extension 500 in the *from-sip* context to dial Digium technical support over IAX. These calls to Digium would require anInternet connection.

zapata.conf

The last entries in the *from-sip* context provide support for a Digium four-port FXS card. This card would be configured in*zapata.conf* with an entry similar to

```
signalling=fxo_k
context=from-si
channel>1-4
```

Note that the *zapata.conf* entry indicates the context *from-sip* for calls from this interface. This now makes*from-sip* a poor choice for the name of the context. A name like *main* would be better.

Voicemail.conf

This configuration assumes that you will provide voice mail for each of the two telephones. Here is an example of*voicemail.conf* for the two users.

```
[general]
; Default formats for writingVoicemail
;format=g723sf|wav49|wa
format=wav49|gsm|wa
; Who the e-mail notification should appear to come fro
serveremail=asteris
;serveremailasterisk@linux-support.net
; Should the email contain the voicemail as an attachmen
attach=ye
; Maximum length of a voicemail messag
maxmessage=18
; Maximum length of greeting
;maxgreet=6
; How many miliseconds to skip forward/back when rew/ff in message playbac
skipms=300
; How many seconds of silence before we end the recordin
maxsilence=1
; Silence threshold (what we consider silence, the lower, the more sensitive
silencethreshold=12
; Max number of failed login attempt
maxlogins=

[zonemessages]
eastern=America/New_York|'vm-received' Q 'digits/at' IMp
central=America/Chicago|'vm-received' Q 'digits/at' IMp
central24=America/Chicago|'vm-received' q 'digits/at' H 'digits/hundred'
 M 'hours'

[from-sip> 4009,Paul
4035 > 4035,Daryl
```

Before using the voicemail system, create an empty voicemail box for each user. The shell script */usr/ src/asterisk/addmailbo* creates a directory each user. It installs default greetings. Before starting Asterisk, run the addmailbox script twice to create mail folders for extensions 4035 and 4009.

Running the Sample Configuration

Start asterisk from the command prompt. Extension 4009 should be able to dial extension 4035 and extension 4035 should ring. Watch the console for the messages during dialing and after you hang up If the busy message immediately appears, a phone probably hasn't registered with the Asterisk server Make sure the phone is sending register statements. Asterisk relies upon the register statements t ensure that a remote client is available for inbound calls.

Next, try leaving voicemail. Dial one extension from the other extension. Dial into voicemail and set your preferences.Dial into voicemail and check your messages.

Getting your first Asterisk system up and running can be difficult. It can very much be a process of trial-and-error. Check the Asterisk users mailing list archives for help. You can refer any remainin questions you have to the asterisk-users mailing.

When you have your simple configuration running, congratulations and welcome to VoIP telephony with Asterisk.

ChapterÂ 13 - Cisco 7960

This chapter describes how to configure the Cisco 7960 IP telephone for SIP. SIP is described in a separate chapter. Cisco phones and adaptors can act as aSIP client and communicate with a SIP server.

The 7960

The 7960 is a very high quality phone and a highly capable SIP device. It is expensive. It is poorly documented. Cisco support for the phone has often not been good. Often the tech support staff are not familiar with the 7960 runningSIP. Once you overcome these barriers and the phone is operational it is very reliable. Users like this phone a lot.

For operation with Asterisk, the 7960 should be configured to run with SIP instead of the proprietary SKINNY protocol. There are several versions ofSIP available from Cisco for the 7960. This chapter shows how to convert a 7960 toSIP if it is not already running SIP.

This chapter gives detailed instructions for upgrading the 7960 to each of the available versions of SIP. Note that all the 7960 telephones on a subnet must run the same version ofSIP. At the time of writing, the latest version of SIP for the 7960 was version 6.0. You should upgrade your phones to at least this version. CiscoSIP version 6.0 is known to work well with Asterisk.

Configuring a 7960 is difficult and error prone. The steps documented here have been tested and verified. If you differ from these steps you will likely encounter problems that will be time consuming to solve.

You may find that you need help in addition to what is in this chapter and the Cisco provided documentation. Additional technical help for the 7960 is available from the Cisco Web site and Cisco support.

You must have a maintenance contract for a Cisco product to get a login to access the Cisco Web site. Contact Cisco or your authorized reseller for information about a maintenance contract and access t the Cisco Web site.

If you have a new 7960 IP phone, you can get a maintenance contract for that phone. If you have an older phone that is out of warranty, you may be able to get the phone re-certified and then get a maintenance contract. Cisco resellers can get your phone re-certified and sell you a maintenance contract At the time of writing a maintenance agreement for a new phone was a few dollars and an agreemen to put a phone back in warranty was less than $100.

Once you have a login, you can access any information about any Cisco product at the Cisco Technical Assistance Center (TAC.) Access the TAC Web site at **http://www.cisco.com/tac**.

A documentation CD-Rom ships with each 7960 phone. You can order a current documentation CDROM from Cisco. Two documents available from Cisco can help you configure your phone, the Cisco *SIP IP Phone Administrator's Guide* and the document *How to Convert a Cisco 7960 Call Manager Phone toSIP and Reverse the Process.*

You should have both these documents available before proceeding. This chapter does not include all the information found in these two documents. For example, consult the*Administrator's Guide* for instructions on physical installation, connecting to the network, or accessing a phone remotely over th network.The following sections assume that you are familiar with the 7960 controls including th scroll key and soft keys. Figure One shows the 7960 controls

Figure: 13-1 7960 Controls

The 7960 can draw power from an external 48 volt transformer. The 7960 can draw power from the ethernet cable. Power over ethernet requires a powered switch or a power patch panel.

Push the button on the side of the telephone to adjust the foot stand to the desired height.

Phone Lines

he Cisco 7960 provides up to six different lines. An inbound call flashes a line icon on the LCD screen. for the line called. Pressing a button for line button before dialing causes the outboundSIP call to appear to originate from that line. Each line has a message waiting indicator, a flashing envelope.

Overview of the 7960 Initialization Process

The 7960 connects to an ethernet with a CAT5 cable. It provides all the functions of a standard desk telephone. The 7960**must** be attached to an ethernet network to operate. There is no connection to thePSTN.

The 7960 can run SIP or the proprietary Cisco Skinny protocol. For use with Asterisk you must run SIP. You can easily switch the phone between the two protocols.

The 7960 contains flash memory. The flash memory saves SIP firmware and saves phone configuration information. Information written to flash memory is saved when the phone power is off. Flash memory stores hardware configuration information, user configuration information, and local configuration information.

A 7960 can be configured from the keyboard or from files downloaded from a TFTP server. This chapter demonstrates 7960 configuration via downloading.

To configure the phone from a TFTP server, a TFTP server address must be manually programmed into the phone network settings or be sent to the phone from a DHCP server. Network settings manually entered into the phone may be lost when the phone is rebooted.

A 7960 phone **must** be able to communicate with a TFTP server to change to a different version of SIP. The phone loads configuration information from the TFTP server including SIP images or SIP settings.

Note that Windows TFTP servers can be difficult to use. They can be insensitive to the case of names, or names with special characters or spaces, for example. This chapter assumes you are using a version o TFTP supplied with Linux.

Turning on a 7960 phone starts a complex initialization process. When power is applied to the phone, a bootstrap program runs.

If a 7960 phone is running SIP, simultaneously pressing the * key, the **6** key and the **settings** key reboots the phone. This does not work if the phone is configured forSkinny.

Flash memory holds a bootstrap. When the phone boots, the bootstrap runs. The bootstrap loads and executes the phone firmware image from flash memory.

The phone next requests VLAN settings from a Cisco Catalyst switch. The LCD panel shows a message for this request. The phone can operate without a VLAN. The configuration of VLANs is beyond the scope of this book. You may need assistance from your system administrator if your environment uses a VLAN

Next, the phone contacts the TFTP server. Note that you must have a TFTP server to configure 760 phones.

The dual boot image, OS79XX.TXT contains the name of the SIP version the phone should use. The phone will download the correctSIP software from the TFTP server.

SIP firmware is only downloaded to the phone and stored in flash memory when a SIP version named in the configuration files is different than the version already stored in the phone.

The phone will next obtain SIP parameters from the TFTP server. If these steps all completed correctly, the phone is ready for use.

Converting a 7960 to SIP from Skinny

The dual boot file OS79XX.TXT contains the name of a firmware image. The 7960 will attempt to download the firmware version named in this file. This download can be aSkinny image or a SIP image. Loading a new image changes the configuration of the phone betweenSIP and Skinny.

Skinny is the proprietary Cisco Call Manager protocol. If your phone is configured for Skinny, you will need to convert it toSIP for use with Asterisk. Note that you can always convert a phone back to Skinny.

If you have an older phone, or a phone configured for Skinny instead of SIP, you may not be able to load one of the newerSIP images.

Older phones must be upgraded in-turn with each of the SIP releases two, two point two, three, four, five and six. You are more likely to be successful in converting a 7960 fromSkinny to SIP if you upgrade through each of the available versions ofSIP starting with version 2.0. That is, change the phone toSIP version 2.0 then 2.2. Then upgrade the phone to SIP version three, then four and then five and six. Note that a phone that has been upgraded to version five cannot be downgraded.

Installation Steps

The following steps show how to configure a 7960 telephone for SIP. There is a following section for each version ofSIP. In general, you will need to perform each of these seven steps for each SIP release.

1. Download the files you will need from the Cisco Web site. Copy them to your TFTP data directory. In the Mepis distribution, theTFTP data directory is /boot.
2. Rename and modify the configuration files held in the

TFTP directory as needed.

3. Configure the DHCP server.

3. Connect the phone to the network.

4. Apply power to the phone. Phone power can be supplied over the ethernet cable, or directly to the phone by a separate wall transformer

5. Unlock the phone.

6. Configure your phone for your network or configure your DHCP server with the setting required by the phone.

8. Re-boot the phone.

7. Check the phone settings and status messages.

Network Settings With DHCP

Each phone must be configured for your network. If you use a DHCP server, the following DHCP options must be set. Explaining the meaning or use of each of these options is beyond the scope of this book. Note, though, that your DHCP server must be capable of setting values for each of these option including theTFTP server address.

IP Address (DHCP Option 50) *

Subnet Mask (DHCP Option 1)

Routers (Default IP Gateway) (DHCP Option 3) *

DNS Server Address (DHCP Option 6)

TFTP Server (DHCP Option 66)

Domain Name (DHCP Option 15)

Note that with DHCP3, the version of DHCP shipped with Mepis, the TFTP server address is set with the option*next-server.* Here is an example

```
option tftp-server-name"192.168.1.12"
```

You can get the TFTP server ip address from a DNS host. here is an example

```
option domain-name-servers 192.168.100.20, 192.168.8.100;
option domain-name"dname.com";
option tftp-server-name"sip.dname.com";
```

Here is an example DHCP configuration that will correctly configure the 7960.

```
# Sample DHCP configuration file for Asteris

# The ddns-updates-style parameter controls whether or no
# the server will attempt to do aDNS update when a lease is confirmed.
# We default to the behavior of the version 2 package
#('none', since DHCP v2 didn't have support for DDNS.
ddns-update-style none

option routers192.168.0.1 ; default gateway
option domain-name"dnsdomain.net";
option domain-name-servers206.13.28.12, 206.13.31.12;
option ntp-serverstime.windows.com;
option tftp-server-name"192.168.0.12";
default-lease-time 600
max-lease-time 7200
# If this DHCP server is the official DHCP server for the loca
# network, the authoritative directive should be uncommented
authoritative

subnet192.168.0.0 netmask 255.255.255.0 {
range192.168.0.50 192.168.0.150;
```

Setting Network Parameters Manually

Consult the Cisco supplied documentation for more information about manual network configuration.

Briefly, to configure the network settings for the phone, unlock the phone as shown in the following section. If the phone is runningSIP version 4.2 or newer, you will need a password. The default password is "cisco."

Press the **Settings** button.

Press the down arrow to select **Network Configuration** and then press the **Select** soft key. Look at the upper-right portion of your LCD, there should be a unlocked padlock icon

Modify parameters with the toggle button and the arrow keys. When entering IP addresses, the on the keypad will include a period in the IP address.

Press the **Save** soft button to save your changes.

Locking and Unlocking the Phone

For phones up to SIP version 4.1 pressing the three keypad buttons * * # will lock or unlock the phone. To see if the phone is locked or unlocked, press the"settings" key, use the arrow key under the display to select"Network Configuration," and press the button at the bottom of the display labeled "Select." The padlock shown at the right end of the top display line shows as locked or unlocked.

For phones running later versions of SIP, versions 4.2 and later, select the menu item "Lock Config" to lock or unlock the phone. You will require a password to access this item. The default password i "cisco."

Through version 4.1, exiting the settings menu will lock the phone. Rebooting locks the phone.

Recovering From a Lost Password

You may have a 7960 locked with an unknown password. You may be able to change the unknown password. From the keypad, try to change the AlternateTFTP address in the phone Network Configuration DHCP settings to "yes." Enter the IP address of your TFTP server. The configuration file *SIP[MAC_Address].cnf* has a *phone_password* entry. Changing this entry to the password of your choice may change the password for the phone.

Downloading Files from Cisco

At the time of writing, SIP files for the Cisco 7960 were stored at **http://www.cisco.com/cgi-bin/ tablebuild.p/sip-ip-phone7960**. You will need an authorized login and password to access these files. Some of the available files are shown in the following table. The images shown are for the latest mino revision shown in each of he major releases. The next table shows many of the available files. The file you must download for eachSIP version are listed in the following sections.

Copy the downloaded files to the TFTP server directory. In the Mepis distribution this is /boot. Make sure all the files in theTFTP directory are readable by everyone. Note that the names are case sensitive. For example, if the file OS799XX.**TXT** is renamed OS79XX.**txt** the 7960 won't find it.

TABLE: 13-1 SIP Download Files

File Name	Required	Description
OS79XX.TXT	REQUIRED	The contents of this file indicate which software the phone should load. You must edit this file as described below.
SIPDefault.cnf	OPTIONAL	Contains SIP parameters that are to be applied to all phones.
SIPmacaddress.cnf	REQUIRED	Contains SIP parameters for an individual phone. Must be copied and renamed for each phone as described below.
RINGLIST.DAT	OPTIONAL	Lists custom ring options. The audio files named in RINGLIST.DAT must be available in the TFTP data directory.
ringer1.pcm	REQUIRED	Ringer tone
ringer2.pcm	REQUIRED	Ringer tone

POS3xxyy.bin	REQUIRED	SIP IP phone image. The phone must have a SIP image available. The earliest release, version 2, uses this naming convention. xx-major version, yy- minor version
POS-xx-yy-zz.bin	REQUIRED	SIP IP phone image.The phone must have a SIP image available. From version 3 forward, this naming convetion is used. xx-major version, yy-minor version, zz-sub version.
POS-xx-yy-zz.sbn	REQUIRED	SIP IP phone image.The phone must have a SIP image available. Release 5.0 and 5.1 secured phone image. xx-major version, yy-minor version, zz-sub version.
dialplan.xml	OPTIONAL	This dialplan may be downloaded to the phone.
syncinfo.xml	OPTIONAL	Used for remotely booting the phone. Contains an image version and an associated synchronization value.

In addition to the software files, release notes are available for each firmware release.

TABLE: 13-2 Some SIP Image Versions for the 7960

Version	File Name	Release Notes
2.3	P0S30203.bin	SipPhoneReleaseNotes.2.3.txt
3.2	P0S3-03-2-00.bin	SipPhoneReleaseNotes.3.2.txt
4.4	P0S3-04-4-00.bin	SipPhoneReleaseNotes.4.4.txt
5.3	P0S3-05-3-00.zip	SipPhoneReleaseNotes.5.3.txt
6.0	P0S3-06-0-00.zip	SipPhoneReleaseNotes.6.0.pdf

Failure to Upgrade

Here is an example of what the TFTP log entries can look like after a failure to upgrade to SIP, in this example to version 3.0

```
Wed Nov 06 11:58:51 2002: Sending 'OS79XX.TXT' file to10.1.1.1 in binary
  mod
Wed Nov 06 11:58::51 2002: Successful
Wed Nov 06 11:58:51 2002: Sending 'P0S30300.bin' file to10.1.1.1 in
  binary mod
Wed Nov 06 11:58:52 2002: Failed ( State Error )
Wed Nov 06 11:59:00 2002: Sending 'P0S30300.bin' file to10.1.1.1 in
  binary mod
Wed Nov 06 11:59:02 2002: Failed ( State Error )
Wed Nov 06 11:59:10 2002: Sending 'P0S30300.bin' file to10.1.1.1 in
  binary mod
Wed Nov 06 11:59:13 2002: Failed ( State Error
```

SIP Version 2.0

To convert or program a 7960 for version 2.0 SIP download the following files from the Cisco Web site. Copies of these file must be in theTFTP server directory with read and write permission for everyone.

TABLE: 13-3
OS79XX.TXT

SIPDefault.cnf
SIPSIPmacaddress.cnf
RINGLIST.DAT
ringer1.pcm
ringer2.pcm
P0S30200.bin

Connect the phone to the network, but don't power it on yet. The first conversion from Skinny to SIP should be toSIP version 2.0. Trying any later version may cause problems. These are the instructions for modifying the downloaded files when usingSIP version 2.0. Upgrading from version 2.0 to more recent versions is described below.

Edit the file *OS79XX.txt*. The contents of this file determine if the phones will operate as SIP phones or use the Cisco call managerSkinny protocol. This file must contain the name of the version of the SIP operating software you want to install on the phone. In this example, the contents of *OS79XX*.txt references the file*POS30200.bin*. For running SIP version 2.0, the file must contain the text

```
POS30200
```

The name is case sensitive. Note the image version, *P0S30200*, does not need surrounding quotes. Note the name of the image in*OS79X*. txt does not have a*.bin* or other extension.

Here is the encoding of the file name in *OS79XX.txt*.

```
P - the device is a phon
O - indicates a combined image containing the application andDSP
S - protocol, S forSIP, O for Skinny.
3 - the fourth digit indicates the ARM processo
0200 - the name of theSIP image, in this case version 02.00
```

There are two types of configuration files that are available to a 7960. The configuration file that is named*SIPDefault.cnf* contains configuration information that is applied to all SIP phones. Open the file with the editor of your choice. Near the top, theSIP image version is listed.

```
# Image Version
image_version: POS3020
```

You may encounter problems configuring a 7960 for SIP even if you follow the directions below exactly. If you do encounter problems, and your phone doesn't accept the configuration files, edit th *SIPDefault.cnf* file and remove all the comments. Lines with comments start with a # character.

Make sure the image *POS30200* is specified in *OS79XX.txt* as shown above. If the SIP image named in this file is different than theSIP image already in the phone's flash memory, the phone will attempt to download the image named in*OS79XX.txt* from the TFTP server and save it in flash memory. Any SIP image to be downloaded to the phone must be in the TFTP directory.

The SIP parameters in this file are applied to every phone. If you change these files, all the phones on your network will be affected.

You can use phone specific files as shown in the next section, section 7, without a *SIPDefault.cnf*. In this case, you will have to provide parameters found in*SIPDefault.cnf* in the phone specific files. If you want to upgrade all the phones on your system to a different version ofSIP, change the image_version parameter shown in*SipDefault.cnf*. Change the SIP image named in *OS79XX.txt* too.

Save a copy of *SIPDefault.cnf* under a different name. This is because you are about to remove part of the file that you will need when upgrading to later versions.

Edit *SIPDefault.cnf*. Remove all the lines that apply to versions later than version 2.0. If you don't do this, the phone will produce an error as it initializes. Press the**settings** button to view the menu choices. The error message can be viewed in the status messages selection reached through the statu menu item. The**status** button is a soft-key.

In addition, each phone must have a corresponding, unique, individual configuration file. Parameters in the phone specific file will override parameters in the generic configuration file.

Save a copy of the original file SIPmacaddress.cnf. Make a copy of the file *SIPmacaddress.cnf* for each 7960 phone. Every telephone must have a file with the format*SIP[MAC_Address].cnf* available in the TFTP data directory, for example, *SIP002094D245CB.cnf.* Note this file name is all in upper case letters. The MAC address in the file name **must** be capitalized. The file **must** have read permissions for all users, that is chmod 666

The MAC address of the phone is on a sticker on the phone bottom. You can display the MAC address by pressing buttons on the phone. Press**Settings**. Use the **scroll key** below the screen and the **select** soft button below the LCD screen to select *Network Configuration* and then *MAC Address*.

Edit the file. Remove all the lines at the end of the file that are for SIP versions past version 2.0. If you don't remove these lines, after booting the phone status will show an error condition, either a buffe overflow condition or a failure to parse the file.

Booting the Phone

Connect the telephone with the MAC address named by the *SIP [MAC_Address].cnf* file to the network. Power on or reboot the phone.

When the phone boots, it will first request the file named *OS79XX. TXT* from the TFTP server. This file contains the name of the image the phone should access

The image file POS30203.bin must be available in the TFTP directory. The other configuration files must be available in theTFTP directory.

The headset, mute and speaker lights light together on for a moment and then turn off. The Green headset lamp lights for about fifteen seconds. Then the mute light comes on for a moment followed b the speaker light.

The phone display will show

```
Configuring vla
Configuring I
```

During this step, configuring IP, the phone contacts the DHCP server. You can monitor this process by looking at the log file where DHCP writes its log entries. In Mepis, this is*/var/log/ syslog*. Use the Linux comman

```
tail -f syslo
```

to monitor the file as it is written to. You should see log messages as the phone requests the dual boot file, the generic configuration file and the phone specific configuration file.

The phone will lastly display at the bottom,

```
Phone Unprovisioned
```

This message is displayed because no SIP proxy server was selected. The SIP proxy server is found in the default configuration fil

```
# Proxy Serve
proxy1_address:192.168.1.1
# Proxy Server Port (default:5060
proxy1_port: 506
```

Check the phone status for any error messages. Any error messages can be corrected by configuring the SIP parameters.

Note that the phone may fail to convert to the new SIP firmware. If this happens, check the network settings for the phone. The tftp server listed may not be right. Unlock the phone and change the tft server address manually to the correct address and try booting the phone again.

SIP Version 2.2

To convert or program a 7960 for version 2.0 SIP to version 2.2, download the SIP image *POS30202.bin* from the Cisco to your TFTP data directory. Edit the file *OS79XX.TXT* and change the contents to*POS30202*. Edit the file SIPDefault.cnf. Change the image name to

```
# Image Versio
image_version: P0S3020
```

These files should be in the TFTP data directory.

TABLE: 13-4
OS79XX.TXT
SIPDefault.cnf
SIPSIPmacaddress.cnf
RINGLIST.DAT
ringer1.pcm
ringer2.pcm
P0S30202.bin

Note that you may also have to have a phone specific file with the os version to convert to SIP version Â

2.2. Copy the original, unchanged file *SIPmacaddress.cnf* to a file named for the individual phone, for example

```
SIP0007505A3E8B.cnf
```

Change the SIP image named at the top of this file to POS30202. Reboot the phone. This should cause the 2.2SIP image to download into the flash memory of the 7960. At the phone, check the settings. Look under Status-Settings-Firmware. The application load ID should now be POS30202

SIP Version Three

To upgrade from release 2.1 or earlier to release 3.0 requires upgrading to release 2.2 first.

To move from version two to version 2.2 to version 3.02, edit the file *OS79XX.txt* to contain

```
P0S30302
```

Copy the original, unchanged file *SIPmacaddress.cnf* to *SIPDefault.cnf*. Edit this file and change the image name to match the sip release as shown belo

```
# Image Version
image_version: P0S3-03-2-0
```

Edit *SIPDefault.cnf* and remove the lines for versions four and five.

Copy the original, unchanged file *SIPmacaddress.cnf* to a file named for the individual phone, for example

```
SIP0007505A3E8B.cnf
```

Copy the SIP image file *POS3-02-00.bin* to the TFTP directory. Make sure all the files have read permission.

These files should be in the TFTP data directory

TABLE: 13-5
OS79XX.TXT
SIPDefault.cnf
SIPSIPmacaddress.cnf
RINGLIST.DAT
ringer1.pcm
ringer2.pcm
P0S3-03-2-00.bin

Boot the phone. When loading a new SIP image, the boot process is longer. The new SIP image has to be copied to the phone's flash memory. It may take as much as several minutes for the phone to finish the conversion process. The sequence of headphone, mute and speakerphone lights flashing i longer. A new message, BootingDSP, appears at the bottom of the 7960 screen at the end of the download process.

When the phone has finished the boot process, check under the firmware version to insure that the newSIP image has been loaded.

SIP Version Four

To upgrade to version four, edit *OS79XX.txt* to contain P0S30404. Change *SIPDefault.cnf* to include

```
# Image Version
image_version: P0S3-04-4-0
```

Edit *SIPDefault.cnf* and remove any lines for version 4 (yes four!) and five.

Make sure a copy of the corresponding SIP image is in the TFTP directory, *P0S3-04-4-00.bin*

TABLE: 13-6
OS79XX.TXT
SIPDefault.cnf
SIPSIPmacaddress.cnf
RINGLIST.DAT
ringer1.pcm
ringer2.pcm
P0S3-04-4-00.bin

When you boot the phone, a new message will appear after the network configuration message, *Upgrading Software*, to indicate the SIP image is being replaced in flash memory.

The settings options are now different as well, with more choices. From settings-status-status messages, there should be only two messages,

```
Invalid proxy_emergency
Invalid proxy-backu
```

From settings-status-firmware versions the application load ID should be POS-04-4-00.

To change the password, edit the file *SIP[MAC_address].cnf* and change the password in the line phone_password:"cisco"

Edit the file SIPDefault.cnf and replace the lines for version four that you deleted earlier. Reboot the phone.

When the phone has started, the bottom of the LCD display should show "unprovisioned" and 1234567Sip should appear in the upper right hand corner.

Note that you can access the phone with telnet. This will, of course, require the password.

SIP Version Five

The version 5 SIP image is signed. Because it is signed, it is not possible to downgrade to earlier SIP versions after you have upgraded a 7960 phone to version five.

The download file for version five is a ZIP file, not a configuration file. Unzip the files in the zip file to a convenient location. Read the text files containing the release notes. Copy the POS images to th TFTP data directory. Note an additional file is present in this release, POS3-05-3-00.**sbin**.

Edit OSX79XX.txt to contain P0S30503. Edit the file SIPDefault.cnf to contain the image name POS3-05-3-00. Edit the SIPDefault.cnf file to name this image. Reset the telephone.

TABLE: 13-7

OS79XX.TXT
SIPDefault.cnf
SIPSIPmacaddress.cnf
RINGLIST.DAT
ringer1.pcm
ringer2.pcm Â
P0S3-05-3-00.**bin**
P0S3-05-3-00.**sbin**

After the reset has completed, check the settings of the phone to make sure the firmware image has been updated.

SIP Version Six

You should upgrade any 7960 phones to version six. The download file for version six is a ZIP file, not a configuration file.

Copy the POS3-06 images to the TFTP data directory. Edit OSX79XX. txt to contain P0S30600. Edit SIPDefault.cnf to reflect POS3-06-0-00. Reset the phone. The files the phone will request fro theTFTP data directory are shown below.

TABLE: 13-8
OS79XX.TXT
SIPDefault.cnf
SIPSIPmacaddress.cnf
RINGLIST.DAT
ringer1.pcm
ringer2.pcm
P0S3-06-0-00.**bin**
P0S3-06-0-00.**sbin**

Configuring the Phone from the Keypad

The Cisco SIP IP Phone Menu Interface settings are changed through the menu interface.

Use the down arrow to scroll to and highlight a parameter, or press the number for the parameter on the keypad. The number is shown to the left of the parameter on the LCD display. Use* for dots (periods) or press the"." soft key when available on the LCD.

Cancel cancels all changes and exits the current menu.

To configure a SIP IP address or ID parameter press the *Number* soft key to enter a number or press the Alpha soft key to enter a name. Then, use the buttons on the dial pad to enter the desired value.

The 2 key has the letters A, B, and C. For a lowercase a, press the 2 key once. To select different letters or numbers, keep pressing the same key. Press the<< soft key to backup. After changing a parameter, press the Validate soft key to save the value and exit the Edit panel

The Dial Plans

The xml file *dialplan.xml* in the tftp directory specifies the dial plan for all installed 7960 phones. A dial plan changes how the phone operates while the user is dialing. For example, without a dial pla the user must press the numbers to be dialed and then press the dial soft button to start a call. With dial plan, dialing numbers can start a call immediately. A dial plan can support automatic dialing an automatic generation of a secondary dial tone.

The same dial plan can be specified for all phones by changing the file *dialplan.xml*. You can change the dial plan for phones individually by changing the*dial_template* parameter in the phone-specific configuration file.

The xml file must start with and must end with dial plan rules are matched from start to finish. The longest matching rule is always used. Matches against a period are not counted for the length to be the longest.

Use any ASCII editor to change *dialplan.xml*. Each rule is specified on a separate line. The syntax for rules i

```
TEMPLATE MATCH="pattern" Timeout="sec" User="type" Rewrite="altstrng"
 Route"route"
MATCH= "pattern" is the dial pattern to match.

A period (.) matches any character. An an asterisk (*) to matches one or
 more characters. A comma causes the phone to generate a secondary dial
 tone after part of a template matcheTimeout= "sec" The number of seconds before a timeout. Specify zero to
 dail immediately
User= "type" Either IP or Phone. Add User=phone or User=IP to automatically add the tag to the dialed number.
This parameter is not case sensitive.
Rewrite= "xxx" An alternate string to be dialed instead of what is
 entered by the user

Rewrite rules are matched from start to finish. The longest matching rule
 is used. A complete rule is only matched when it has more nonwildcard
 matches than an incomplete r
```

Comments tart with .

Here is an example. Without a dial plan, the user has to press the *Dial* soft button to start a call. This entry in*dialplan.xml* will start a call without pressing the dial button.

Here is a sample North American dial plan.

Custom Ring Tones

Two ring tones Chirp1 and Chirp2 are supplied with the 7960 configuration files. By changing the file *RINGLIST.DAT*, you can add new ring tones.

Ring tones must be a PCM file stored in the TFTP directory. The PCM files must not contain any header information and must be in the following format

```
8000 Hz sampling rat
8 bits per sampl
u-law compressio
```

Use any ASCII editor to change the file *RINGLIST.DAT*. Add the name of each new ring tone to this file, press Tab, and then enter the filename of the ring type. Here is an example. The first entry is th name that displays on the phone, the second entry is the name of the pcm encoded file. The sound file must be located in the tftp directory.

```
oldstyle  oldstyle.pc
what whatwhatwhat.pc
synthlow synthlow.pc
```

Note, the label and the file name must be separated with a TAB character or the download will fail.

Enabling the Messages Button

Here is how to configure Asterisk so that the *messages* button on the 7960 will dial voice mail. Thanks are due to John Baker, Adam Low, and Brian Pollack for figuring this out.

First, in the configuration file for the 7960, *SIPDefault.cnf,* add an entry that specifies the uniform resource indicator for messages, in this example extension 8500.

```
messages_uri:"8500"
```

Make sure that *sip.conf* has the caller ID specified for each user.

```
callerid=Brian 300
callerid=John 310
```

The following entry in *extensions.conf* will enable voice mail. The argument *yourcontext* refers to the voice mail context in*voicemail.conf*.

```
exten > 8500,1,VoicemailMain2(${EXTEN}@yourcontext)
exten > 8500,2,Hangup
```

Any user dialing extension *8500* will be directed to voice mail.

Enabling the Waiting Messages Light

Specify the messages uniform resource indicator in the configuration file for the individual phone.

```
SIPXXXXXXXXXXXX.cnf
messages_uri:"4008"
```

Modify *sip.cnf* to include a mailbox entry as shown below. This specifies a mailbox number and a context found within *voicemail.cnf*, in this case *4008@inside.*

```
[4008
type=friend          ; This device takes and makes call
username=400
secret=yoursecre
context=inside   ; The context in voicemail.cn
callerid"TUser" <8005551212>
qualify=100
host=dynamic         ; This host is not on the same IP addr every tim
canreinvite=n
mailbox=4008@inside  ; Activate the message waiting light
defaultip192.168.0.12
```

SIP Parameters

Please consult *Cisco SIP IP Phone Administrator's Guide* for an explanation of all the SIP parameters for the 7960. There is a separate edition of this document for eachSIP firmware version. You should do this to familiarize yourself with the capabilities of the phone

ChapterÂ 14 - SNOM Telephones

This chapter describes the SNOM IP telephone. SNOM phones are a SIP client and communicate with aSIP server. SIP is described in a separate chapter.

Figure: <u>14-1</u> Snom 200 Telephone

Configuration and Setup

The Snom phone is easy to configure for Asterisk. The Snom has a built-in Web server. Because it has a built-in Web server, you can configure the phone with a browser. Just supply your browser with th ip address of the telephone you wish to configure.

There are several useful documents on the snom Web site in the FAQ section. You can download these FAQs from*http://www.snom.com/faq_en.php*. This is a very useful link for technical information. You should check it periodically or when you encounter a technical issue. Some of the issues covered in the FAQs include

- * using snom phones with Asterisk.
- * Configuring snom phones for Mass Deployment
- * Dial plan on snom phone.
- * How to update the firmware for a Snom phone.
- * Operating SNOM phones behind NAT
- * Power Over Ethernet.

- * Setting up DHCP for snom 100/200
- * Configuring Cisco Call Manager for snom Phones

Several useful documents are available on the ABP Tech website under Support FAQs. You can download these documents from *http://www.abptech.com/mainpages/support/faq_index.html*. Available documents includ

- * How to update snom phone firmware with TFTP
- * Setting up a snom phone behind LinkSys UPnP router
- * Using the programmable keys on the snom 200

If your snom 200 telephones are operating on the same sub-net behind a SIP-enabled firewall, you should turnNAT Detection to OFF (Settings/SIP/Stack) to avoid possible conflicts. If you are installing snom 200 telephones behind a NAT router at a remote location, you can activate Automatic or STUN settings.

Documentation

One of the choices within the configuration Web pages from the telephone Web server will show you the manual for the telepohne. Two documents are additionally available from the Snom Web site.

> *Snom 200 User Manual: Operations Manual for End Users*. Provides instructions for web interface and phone operations.
>
> *Snom 200 Administrators Manual*. A technical reference manual for configuration and setup of snom 200 VoIP Phones.

You can download these manuals from *http://www.snom.com/snom200_en.php*. Note the links for the manuals are at the bottom of this Web page

Administrator Password

If you want to turn Administrator Mode ON or OFF to restrict the menu options available to users, the default password is"0000" (four zeros).

Firmware

You can access all the firmware versions for snom phones from http://www.snom.com/ support_dl_en.ph

Technical Support

If you should run into a technical issue, you should immediately open a trouble ticket on the ABP website athttp://www.abptech.com/ mainpages/support/supportcase.html ABP Tech Support uses this database to respond to issues and we track every open ticket.

ChapterÂ 15 - T-Carrier and SONET

The most common business connection to the PSTN (Public Switched Telephone Network,) or Internet is a T1 line, or in areas outside the US an E1 line. A T1 line is often called a DS-1. The following sections describe T1 and other"T" type lines.

This chapter is not a complete reference to T-Carrier or ISDN. For more in-depth information, consult one of the excellent references listed in the appendix.

A T1 line, provides a point-to-point connection. For example, you can use a T1 line to connect your office to the telephone company central office switch for dial telephone service. You can use a T1 lin to connect your local computer network to an ISP to establish a connection to theInternet. You, the user, determine the end points. You, the user, determine what the T1 line is used for, voice or data o both.

T-Carrier is a series of digital communications systems used by telephone companies around the world.T-Carrier is a digital protocol developed by AT&T by 1957 and first implemented in the early 1960's. TheT-Carrier was developed to support the transmission of digitized voice. T-Carrier provides telephone companies the technology to move voice or data digitally over what had been before an analog system

T-Carrier uses two pairs of wire. It is full-duplex, that is data can be sent and received at the same time. Signals are digitized and then sent over the T1 connection. Voice is sampled 8,000 times a second and converted into eight bit words. An frame is built that contains a word for each of the 24 channels. A frame is transmitted 8,000 times a second.

Digital T-Carrier circuits provide much greater bandwidth than analog circuits. A set of copper wires used to transmit an analog signal can instead transmit data digitally. Sending data digitally allow much more data, even much more digitized voice, to be sent over the same copper wires.

T-Carrier is used to build the ISDN, Integrated Services Data Network. ISDN is a set of integrated standards used to build a digital telephone network. WithISDN the same switches and digital transmission paths are used to establish connections for different services, for example data and voice.

The ISDN standard was first published as one of the 1984 ITU-T Red Book recommendations and expanded in the 1988 Blue Book.ISDN uses Public Switched Telephone Network (PSTN) switches and wiring. This wiring is upgraded to support the basic"telephone call" on a digital network.

Different types of T-Carrier circuits are available. When you order T-Carrier line, for example a T1 line, you order a circuit with a specified amount of bandwidth. For example, a 24 channel T1 line wil provide 1.544mbps of bandwidth or a T-3 line will provide 44.736 mbps of bandwidth.

T-Carrier costs are continually dropping. T-Carrier lines are extremely popular for business users who wish to connect to theInternet or the PSTN.

T-Carrier and DS0

The "T" designation specifies the physical interface for services obtained from a local carrier. That is, T-Carrier specifies a physical set of wires, repeaters, connectors, plugs, jacks, etc.

In terms of the OSI standard network model (briefly described in the appendix,) T-Carrier is the standard for layers one and two. T-Carrier specifies the physical connection and the carrier signal sent over that physical connection.

Data is carried on top of the T-Carrier. Data is carried on a T-Carrier channel at a digital data rate that is called*Digital Signal Level Zero* or DS0. DS0 is described below.

T-Carrier describes the physical layer interface to a provider network. A T-Carrier circuit is typically provided as two pairs of wire. These are bare wires that run directly from the central office to the customer premises without any conditioning[1]

The maximum T-Carrier signal distance is 3000 wire feet measured from the egress at the cnetral office. Repeaters are used to extend aT-Carrier signal further than 3000 wire feet. The first repeater is placed within 3000 wire feet of the CO. Successive repeaters are placed every 5000 wire feet. The las repeater is installed within 3000 wire feet of the customer's termination point

1. Conditioning devices like bridge taps and load coils are used on analog telephone lines to help maintain or improve signal quality. Splices, which are common, tend to degrade signal quality

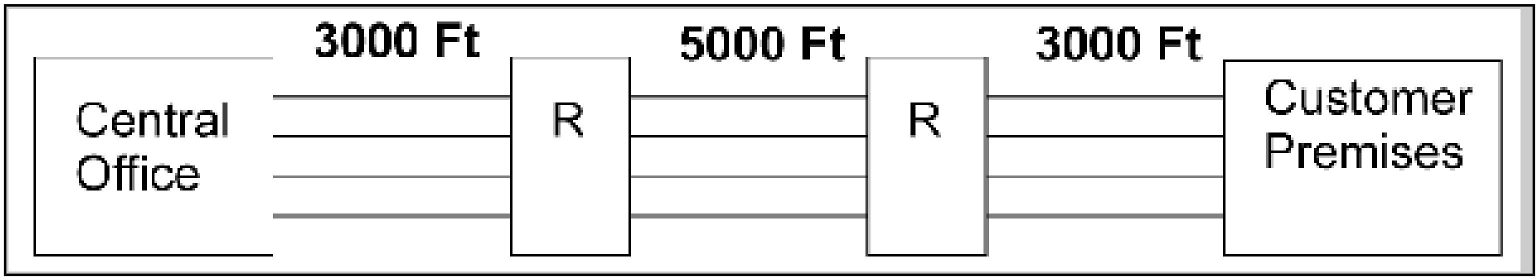


Figure: **15-1** T1 Repeaters

Once the physical T-Carrier line is installed, you can use it to send and receive data. Customer data including voice (for telephone calls,) data or video can be sent over theT-Carrier line.

Note that this type of circuit is rapidly becoming obsolete. Many new DS-1 circuits are being delivered on one pair of copper wires using HDSL technology.

Digital Signal Zero

T-Carrier is a channelized system. In North America, the basic data channel is called a Digital Signal Zero (DS0) channel.

Digital Signal Zero was standardized by the ANSI T1.107 guidelines. The international ITU-T guidelines are slightly different.

DS0 is a dedicated, point-to-point line service. DS0 service can send voice and digital data including video.Each DS0 channel provides 64 kbs of bandwidth, enough bandwidth to transmit a digitize voice signal. Each DS0 provides a 64 kilobits per secondPCM end-to-end channel transmitted over theT-Carrier. Voice signals are sampled 8,000 times a second. Each of the samples is digitized into an 8-bit word which supports a 64Kbs signal. Each of the 8-bit words is sent over the DS0 channel.

The multiple T-Carrier channels in a single T-Carrier connection can transmit voice or to transmit data. The separate channels in aT-Carrier circuit can be assigned to different uses. Some channels can be dedicated to telephone usage while others are simultaneously dedicated to data

As described in the following section, DS0 channels can be combined to create high bandwidth connections.

The T-Carrier-Ds Hierarchy

T-Carrier systems combine channels to provide greater bandwidth. For example, in North America a T1 line provides 24 channels for a total bandwidth of 1.544mbps and in Europe an E-1 line provides Â

2.048 mbps of bandwidth and 30 channels. T-Carrier bandwidth is aggregated by combining DS0 channels.

There is a hierarchy of T-Carrier circuits. Each step provides more bandwidth. The hierarchy of combinations for T-Carrier circuits are shown in Table 1. It is possible to purchase a "fractional" T1 line where fewer than 24 channels are provided

TABLE: 15-1 T-Carrier Hierarchy

T-Carrier Systems	North America	Japan	International
channel data rate	64 kbs (DS0)note one	64 kbs	64 kbs (DS0)
T1 - first level	1.544 mbps (DS1) (24 user channels	1.544 mbps (24 user channels)	2.048 mbps (E1) (30 user channels
intermediate level	3.152 mbps (DS1C) (48 Ch.	Not Available	Not Available
second level	6.312 mbps (DS2) (96 Ch.	6.312 mbps (96 Ch.), or 7.786 mbps (120 Ch.)	8.448 mbps (E2) (120 Ch.
T3 - third level	44.736 mbps (DS3) (672 Ch.	32.064 mbps (480 Ch.)	34.368 mbps (E3) (480 Ch.
fourth level	274.176 Mb/s (DS4) (4032 Ch.	97.728 Mb/s (1440 Ch.)	139.268 mbps (E4) (1920 Ch.
fifth level	400.352 mbps (5760 Ch.)	565.148 mbps (7680 Ch.)	565.148 mbps (7680 Ch.)

Note 1: The DS designations other than DS0 are used in connection with the North American hierarchy only.

Note 2: Other data rates are in use. The military has systems that operate at six and eight times the DS1 rate. At least one commercial system operates at 90 Mb/s, twice the DS3 rate

Note 3: T1c, T-2 and T-4 are rarely used.

T1 lines are in common use today in for connections to the Internet. The T-3 line, providing 44.736 mbps, is commonly used between Internet service providers.

ISDN

Integrated Services Data Network (ISDN) was standardized in the 1980s. ISDN is an international standard interface protocol from The International Telecommunications Union (ITU-T formerly th CCITT,) providing single access to multiple services.ISDN signaling is SS7 compatible. ISDN subscribers can access SS7 network services and intelligence through ISDN.

ISDN provides a variety of communications services in circuit switched networks. These include bearer services for speech, 3.1 kHz audio for modems and 64 kbps digital data. Teleservice support fa and telex. Supplementary services include calling line identification (caller ID,) user-to-user signalling call waiting, and call hold among others.

ISDN provides D and B channels. Bearer (B) channels are bi-directional 64 kbs channels that carry user information. B channels do not carry signalling information. Bi-directional 9.6 kbs Data (D channels carry signalling information.

BRI

When someone talks about an ISDN connection, they are usually referring to a Basic Rate Interface. A (BRI) provides two 64K"bearer" DS0 channels and a single "delta" DS0 channel ("2B +D"). The bearer channels are used for data transmission. The delta channel is used for out of band signalling, fo example call setup. Because of tariffs, BRIISDN is typically an expensive service to operate. BRI ISDN lines are typically charged by the minute which causes the cost to quickly rise. While ISDN has had some success in video conferencing, because of the cost it has never become very popular in Nort America. More modernDSL technology has replaced ISDN for anyone who has access to DSL. BRI is still popular and cost effective in many European locals.

PRI

A Primary Rate Interface (PRI) in North America and Japan consists of 24 channels, usually 23 B + 1 D channel with the same physical interface as a T1 where all the channels operate at 64 kbps.Th combinedPRI channels results in a digital signal 1 (T1) interface at the network boundary.

In some areas outside the US, the PRI usually has 30 B + 1 D channel and an E1 interface. As with the BRI, the D channel is used for out of band signalling. While aPRI is an ISDN connection, it is rarely referred to as such.

How T-Carrier Channels Are Combined

T-Carrier sends data over the line in bytes. Each byte is sent in order, one after the other in "frames." A single frame contains one Byte (8 bits) of data for each channel. An extra bit is then sent to synchronize the data stream. This extra bit is called a Frame Bit.

193 bits, 192 data bits and one framing bit, are sent for each frame. This increases the total bandwidth to 1.544mbps. 24 64 Kb DS0 channels taken together provide 1.536 mbps. A T1 provides 1.544 mbps of bandwidth. The extra bandwidth comes from the Frame bits. T1C frames differ as they are made u of 1272 bits.

T-Carrier uses pulse code modulation and time-division multiplexing. The time division multiplexing is illustrated in the following figure. A frame is sent in 125 micro seconds.T-Carrier uses four wires and provides duplex capability. Two wires are used for receiving and two for simultaneous sendin

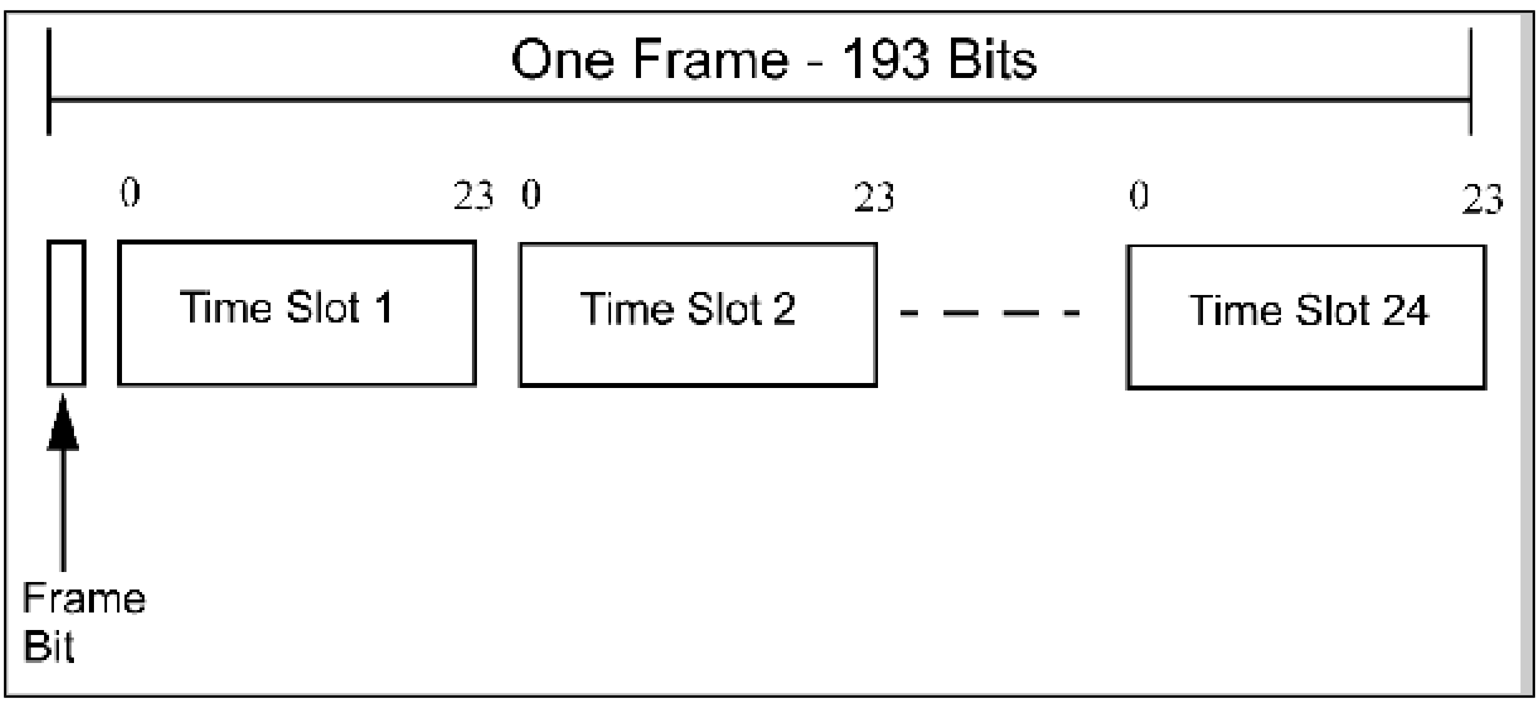


Figure: 15-2 Data Frame

T1 Framing Formats and signalling

In North America, Canada, Hong Kong, and Taiwan two framing formats are in use, Superframe and Extended Superframe.

A Superframe consists of twelve 193-bit frames. A framing bit can support different functions, depending upon which of the twelve frames it is in. There are two types of framing bits; Termina framing (Ft) and Signaling framing (Fs) bits.

With Superframe, the standard frame is 193 bits long and includes 1 Framing bit plus 24 8-bit time slots. Each Superframe time slots is scanned at a rate of 8000 times per second. Therefore, in one second, there are: (8000 * 8 bits)/TS * 24 TS = 1,536,000 Bits of payload data transmitted. There are 800 1 = 8,000 bits of synchronization bits transmitted within a one second interval. Therefore, the tota aggregate rate of the T1 signal is 1,544,000 bps (1.544 Mbps)

The standard frame is 193 bits long, 1 framing bit + 24 8-bit time slots. Each time slot is scanned at the rate of 8000 times per second, as in D4/SF. The line rate is 1.544 Mbps and supports a data paylo of 1.536 Mbp

Signalling states are transported within a Superframe. This is required to support Switched voice or data service. Signals are sent with a"Robbed Bit" bit:8 of each channel's time slot is "robbed" to indicate a signaling state in the 6th and 12th frames. Effective throughput for the A signaling bit (Frame Â

6) is 666.66 BPS. Effective throughput for the B signaling bit (Frame 12) is the same (666.66 BPS).

An Extended Superframe consists of twenty-four 193-bit frames. There are three types of framing bits; Frame Pattern Sync (FPS), Datalink (DL), and CyclicRedundancy Check (CRC) bits. Of the 8 kbs framing bit bandwidth 4 kbs is allocated to the Datalink, 2 kbs is allocated to the CRC-6 characte and 2 kbs is used for synchronization purpose

ESF (Extended Superframe Signaling) uses a "Robbed Bit" Each channel's timeslot is "robbed" to create a signaling in the 6th, 12th, 18th, and 24th frames. Effective throughput for the A signaling bit (Frame 6) is 333.33 BPS. Effective throughput for the B, C and D bits is the same (333.33 BPS)

Using T Carrier Channels for Telephone Calls

After your T1 provider drops the T1 into your premises, they may then hand you a CSU/DSU or a router. This router will have a T1 connection on the back. The router contains circuitry that communicate withT-Carrier. A connection between the telephone company T1 drop and the router establishes the connection.

You can connect from the T1 drop to the router. It is advisable to use a real RJ45 cable instead of a CAT5 cable. This is described in the section on cables and connectors below. If you are using the T1 line only for data, your configuration may be complete when you configure your router and connect i to your LAN. This will provide a path for data from your company to the other end of the T1 line

A channel that is used to place telephone calls to the PSTN must be connected to the PSTN, for example a CO (central office.)

If you are using the T1 for telephone calls to the PSTN, you will need some piece of equipment that provides a connection between your analog telephones or fax machines and the T1 line. If you are using the T1 for making telephone calls, your router may have a connector on the back that accepts T1 cable. This means the router is smart enough to take telephone traffic off the T1 channels an route them to the telephone connectors on the back of the router. You may have a separate piece o equipment called a channel bank that accepts the T1 line

IP Phones will, of course, connect to your local area network, not the analog connectors. A VoIP call can be sent over a T1 DS0 channel as data. This data channel could be connected to your ISP. Th telephone call would then be routed over theInternet instead of the PSTN. Such a call might eventually be connected back to the PSTN through a gateway elsewhere.

The Confusion Surrounding T-Carrier and DS0

When you hear someone say T1 you will probably have a hard time figuring out exactly what they mean.T-Carrier discussions are very confusing because of the interchangeability of words and the confusing requirements for connecting to the PSTN.

A T1 line can refer to a connection that has 1.544 mbps of bandwidth. It might be referring to a network that uses the T carrier electrical interface specification (DSX-1.) Or, it might mean that the network uses one of several framing formats, D4, ESF, etc.

T1 Cables

A T1 cable is different from a CAT5 ethernet cable. Use a real T1 cable when a T1 cable is called for. When extending T1 lines from the phone company drop to your customer equipment, use a T1 cable not aCAT5 cable.

T1 cables use Individually Shielded Twisted Pair (ISTP.) ISTP is used because of the susceptibility of T1 signals to Near End Cross Talk NEXT.)

Unshielded Twisted Pair (UTP) cable characteristics are similar to ISTP. However, due to the unshielded characteristics ofUTP, the proximity of the unshielded transmit and receive cable pairs can causeNEXT. This can result in link errors if you use a CAT5 cable.

T1 Optional Services

Various vendors may have optional T1 services that you may want. Here is an example.

The AT&T Digital Carrier System is referred to as ACCUNET T1.5. It is described as a "two-point, dedicated, high capacity, digital service provided on terrestrial digital facilities capable of transmitting 1.544 Mbs" The interface to the customer can be either a T1 carrier or a higher order multiplexed facility such as those used to provide access from fiber optic and radio systems

AT&T offers services in addition to point-to-point data service. For example, four "transfer arrangements" can be purchased:

1. The customer can change the terminating location of a T1 link with AT&T assistance.
2. M24 Multiplexing allows the user to subscribe to any of the 24 T1 channels individually to switched and non-switched services offered by A&T.
3. M44 Multiplexing combines 2 T1 lines, each carrying up to 22 channels, onto one T1 line using Bit Compression Multiplexing (BCM)
4. Customer Controlled Reconfiguration (CCR) allows the customer to dynamically allocate circuits without A&T assistance.

AT&T states that their performance objective is 95% Error Free Seconds (EFS) on a daily basis and Â

99.7% availability on a yearly basis.

Be sure to check what features your service provider that might be helpful in your application.

Where did the T in T1 come from?

In 1917 AT&T deployed the first carrier system, called the "A" system. Seven A-systems, with four voice channels over pair of wires, were ever deployed. Over time, newer analog frequency division multiplex systems named B, C and D, were developed. Few of these saw commercial service. The L syste was very successful and provided 600 (L1) and later 1800 (L3) voice channels over a pair of coaxia cables.

The telephone companies refer to long distance service as "long haul" or "long lines." This system stayed in long haul service from 1944 to 1984 when the breakup of the Bell System forced A&T to move to optical fiber. The last analog carrier system was the N system. This system provided 12 voic channels and was used for intracity short haul. O, P, and U systems were never put into service, th emergence of T killed them

In 1957 digital systems were first proposed and developed. A manager at AT&T, then the only telephone company, decided to skip Q, R, S, and to use T, for Time Division. This was to be the world's first time division system. Except for"U", another system that was never deployed, this naming system ended

The variants of T1 called T1C, T2, and T4, all vanished. They are survived by signals that would have been carried on all these systems. These are called DS1, DS2, DS3, and DS4. The successor to theT-Carrier protocols are various protocols running on optical fiber, for example SONET, but they don't have a letter designation.

SONET

The next step up from T-Carrier is SONET, Synchronized Optical Network. SONET is a very high speed physical layer network protocol. It is designed to transmit large volumes of traffic over long distances on fiber optic cables. ANSI developedSONET for the public telephone network in the mid-1980s. You would be able to make a very large number of telephone calls over a SONET connection.

SONET specifies interoperability standards between products from different vendors. SONET can carry different data protocols including IP.SONET includes management and maintenance support. SONET is cost competitive with alternatives like ATM and Gigabit Ethernet

SONET specifies OC (optical carrier) signal levels. The OC signal levels place STS (synchronous transport signal) frames onto a multimode fiber-optic line at a variety of speeds. The base signal rate is Â

51.84 Mbps (OC-1); each signal level thereafter operates at a speed divisible by that number (thus, OC-3 runs at 155.52 Mbps)

This system is built with multiplies of the OC-1 rate of 51.840 Mbps. This is called STS-1 (Synchronous Transport Signal, Level 1). T

TABLE: 15-2 SONET Speeds	
Name	**Data Rate**
STS-1	51.849 Mbs
STS-3	155.520 Mbs
STS-9	466.560 Mbs
STS-12	622.080 Mbs
STS-48	2488.320

International SDH (Synchronous Digital Hierarchy)

This system uses a fundamental rate of 155.520 mbps, three times the speed of SONET. The fundamental signal is STM-1 (Synchronous Transport Module, Level 1). The transmission media fiber, butis the BroadbandISDN specifies a User-Network Interface STM-1 (155.520 mbps) operating over coaxial cables. Some typical rates within this hierarchy

TABLE: 15-3 SDH Speeds	
Name	**Data Rate**
STM-1	155.520 Mbs
STM-3	466.560 Mbs
STM-4	622.080 Mbs
STM-16	2488.320 Mbs

ChapterÂ 16 - Networks and Signaling

The Public Switched Telephone Network started in 1876 when Alexander Grahm Bell made the first telephone call. The first call was from Mr. Bell to his assistant Mr. Watson where he said,"Mr. Watson, come here. I need you." The second call was from a telemarketer.

This first call was made over a ring-down circuit. Two wires connected the two telephones. The first phone was always connected to the second phone and there was no ringing. This was a half-duplex circuit where only one person could talk at once. As shown in the following diagram, every phone wa connected to every other phone

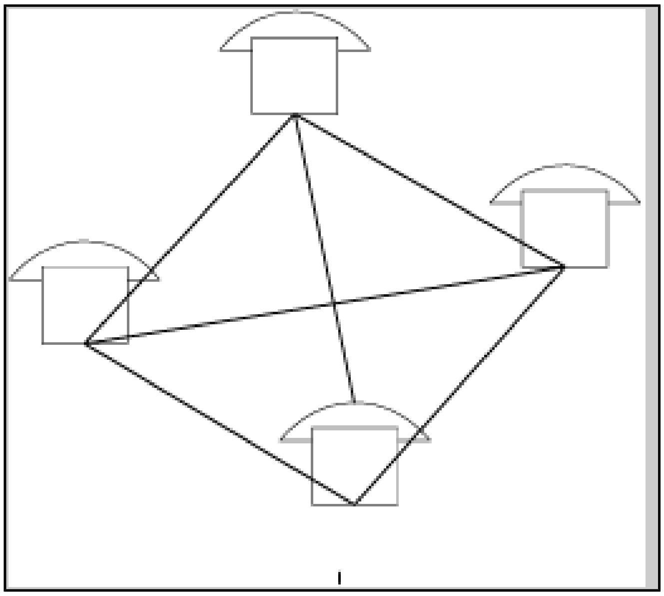


Figure: **16-1** Fully Connected or Full-Mesh

It would be too expensive, and too difficult, to build a large telephone network with this topology. The solution to this topology problem is a switch. A switch only requires a wire pair from each phone to central office. At the central office, a switch is used to connect one call to another call. The origina switch was a person, the operator.

The PSTN quickly evolved to a full duplex system where both parties could talk at the same time. The person was soon replaced by a mechanical switch. Years later, the mechanical switch was replaced wit the electronic switch. Now, Asterisk running on a PC with Digium interface boards can switch calls.

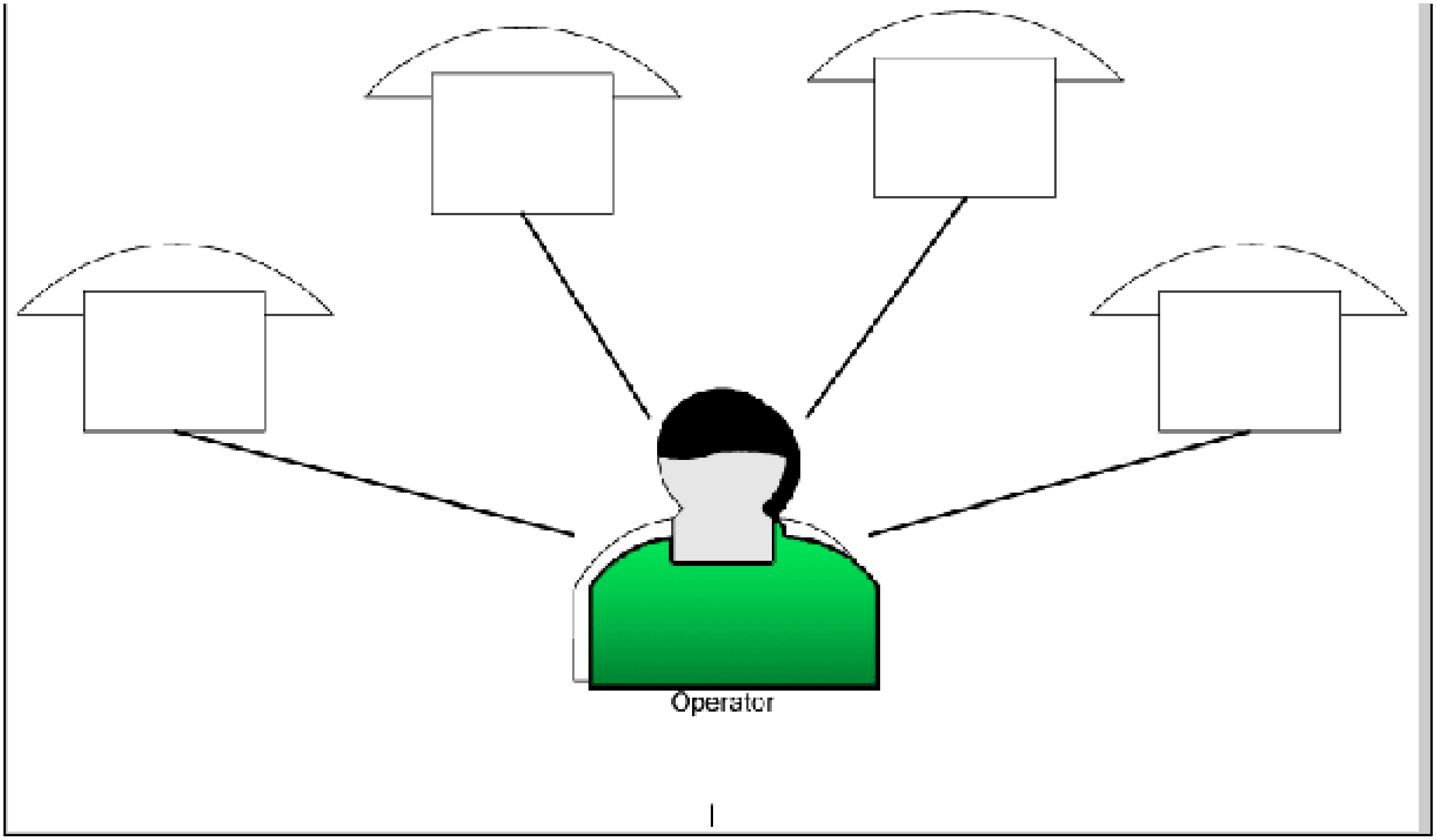


Figure: 16-2 Fully Connected

PSTN Basics

Sounds are analog. They are continuous wave forms that vary in frequency and amplitude. The PSTN originally sent analog signals from one phone to another. Over longer distances, the signals nee amplification. Unfortunately, amplification makes the noise louder as it makes the signal louder. Eac additional amplifier adds more noise and degrades the signal further as it traveled over longer distances.

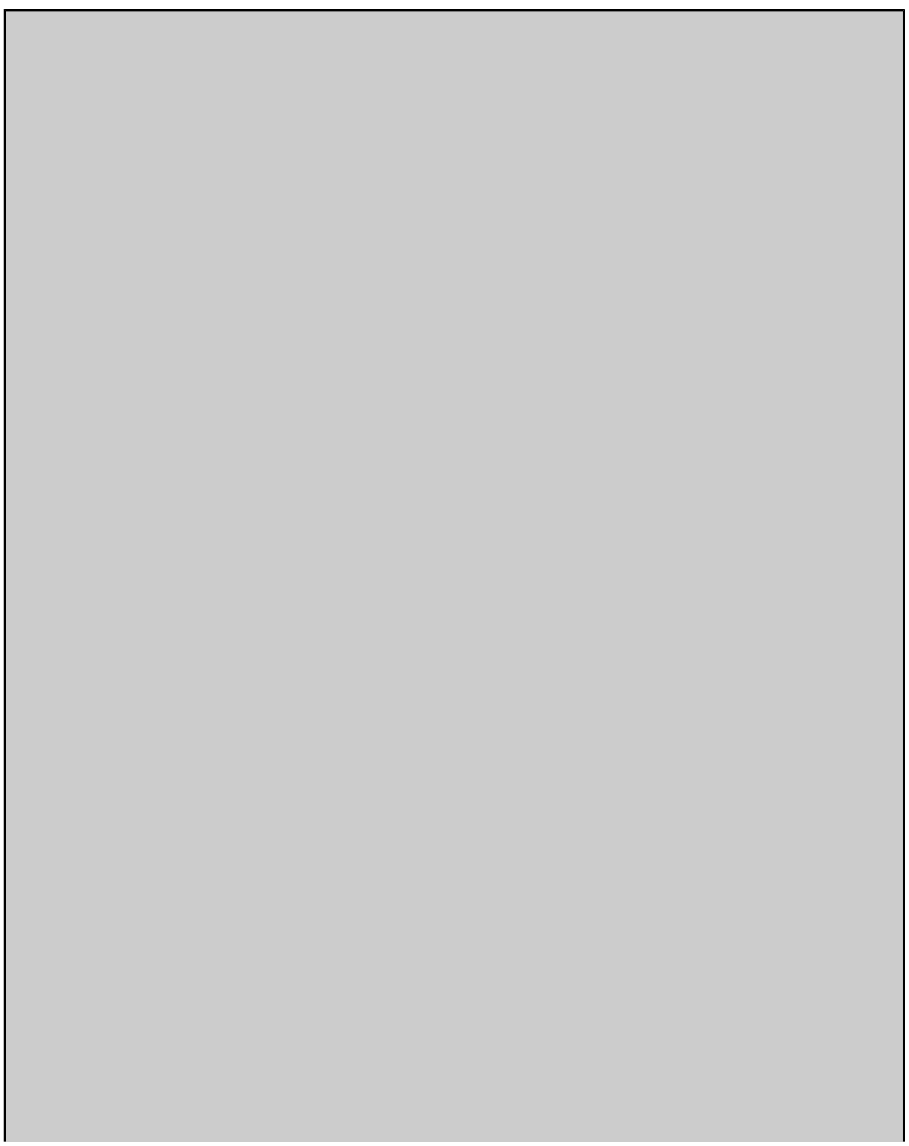
More recent technology allows analog signals to be digitized. The original analog waveform can be represented as a stream of numbers. Digitization relies on the Nyquist theorem. A high quality digita representation of an analog wave form can be created by sampling the waveform twice as fast as th highest frequency found in the analog waveform.

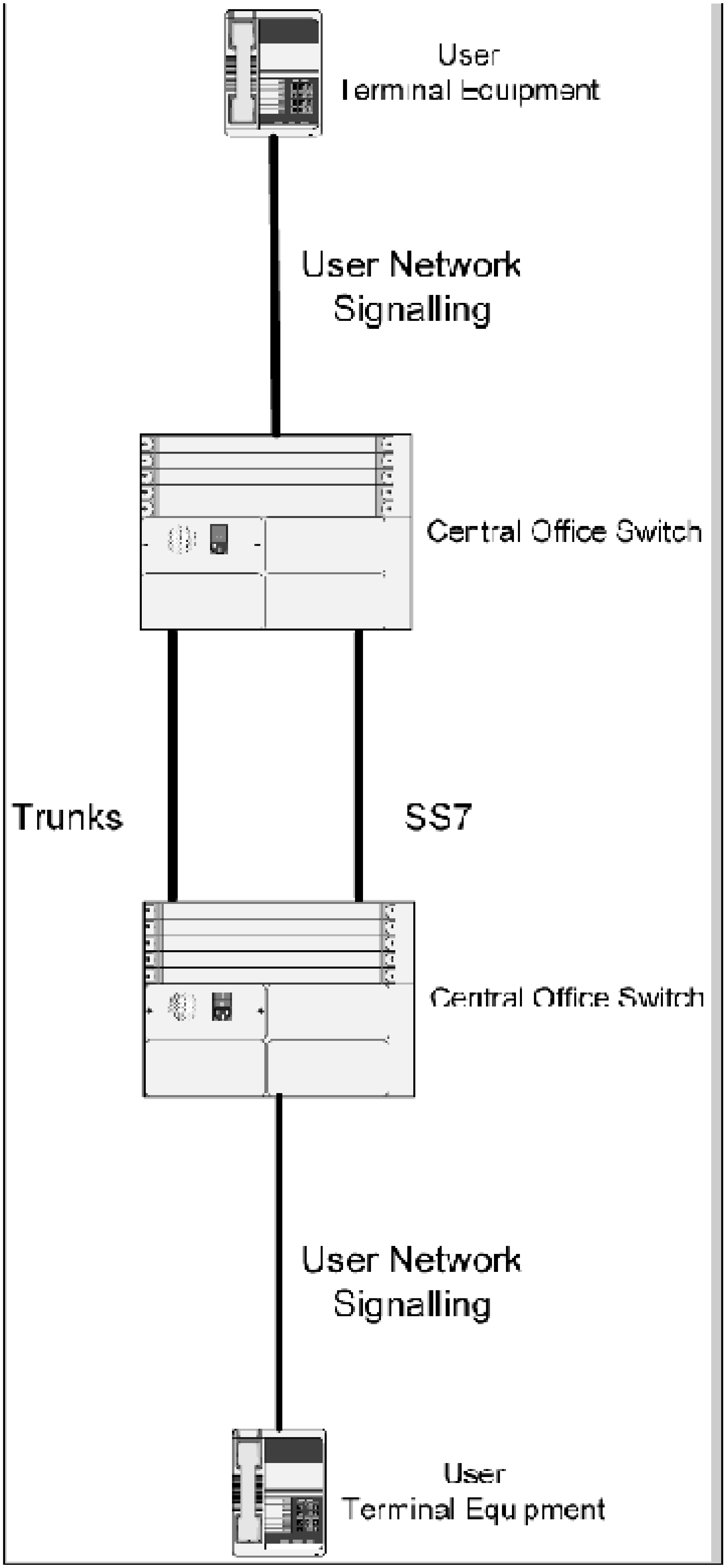
The most common method of digitizing analog signals is Pulse Code Modulation. With PCM, the analog signal is first filtered, for example to remove any frequencies above 4kHz or below 100Hz. Thi signal is then sampled 8,000 times per second, twice the highest frequency.

The samples create a digital data stream. Each data element in the data stream represents the amplitude of the original analog waveform at the moment the sample was taken. PCM uses an eight bit coding scheme coupled with a logarithmic compression algorithm. Sampling eight bit values at 8,000 times a second produces a 64 kbs data stream.

A pair of wires running from a central office to a telephone is called a local loop. The local loop connects the telephone to a switch in the central office. The communications link between one central office and another is called a trunk. Central offices are connected hierarchically. Central office switche connect through trunks to tandem switches. Tandem switches are referred to as Class 4 switches.

Class 5 switches often connect directly to each other. These connections are put in place after analyzing calling patterns between switches. If there are enough calls between two class 5 switches a dedicated circuit is installed.





User
Terminal Equipment
User Network
Signalling
Central Office Switch
Trunks
SS7
Central Office Switch
User Network
Signalling
User
Terminal Equipment

Figure: 16-3 PSTN

PSTN Signalling

A local loop, that is a pair of copper wires, can transmit analog or digital data to a central office. There are two signalling paths in thePSTN. End users signal the PSTN with user-to-network signalling. Switches in thePSTN signal each other with network-to-network signalling.

Signals can be analog or digital. Dual Tone Multi-Frequency (DTMF) signalling sends two simultaneous tones over the voice path.

Signaling can be in-band or out-of-band. For example, DTMF is in-band signalling. Dialing a number sends analog DTMF signals to the central office switch over the voice circuit.

Out of band signalling sends signalling information on a separate channel from the transmitted voice. For example, a Basic Rate Interface provides two 64kbps bearer (B) channels used to send and receiv voice and a third 16 kbs D (data) channel used for out of band signalling.

Out of band signalling has several benefits including reduced dialing delay, higher signal bandwidth and the ability to multiplex multiple signals over single channel. Out-of-band signalling greatl improves call service including call completion.

PSTN Network-to-Network Signalling

Network-to-network signalling includes in-band signalling methods like Multi-Frequency (MF) and Robbed Bit Signalling (RBS.) MF is like DTMF but uses different frequencies.

SS7 (C7 in Europe) is the common out-of-band signalling protocol used between switches. SS7 is used to send messages between switches for basic call control. SS7 allows signalling to control th Intelligent Network. The Intelligent Network implements Custom Local Area Calling Services lik three way calling or call waitin

CLASS services include

- Call Forwarding
- Call Waiting
- Three-way Calling
- Speed Calling

Anonymous Call Rejection

Automatic Callback

Automatic Recall

Call Forwarding Busy

Call Forwarding No Answer

Call Name and Number Delivery

Call Name and Number Delivery w/Call Waiting

Call Number Delivery

Call Number Delivery w/Call Waiting

Call Number Delivery Blocking

Customer Originated Trace

Distinctive Ringing / Call Waiting

Selective Call Acceptance

Selective Call Forwarding

Selective Call Rejection

Voice Mail

SS7 to database connections support network-based services including 800-number service and Local Number Portability.

The following sequence diagram shows a typical SS7 call flow. In this example, picking up the phone sends an off-hook signal to the SS7 switch at the local office. The switch sends dial tone to the phone The caller presses buttons on the phone. This sends a message to the switch containing a telephon number. TheSwitch responds to the dialed number with a setup or Initial Address Message (IAM.) The local switch sends a new IAM across the SS7 network to the second switch. The second switc sends an Address Complete Message (ACM) back over the SS7 network. The called phone rings. Th calling party hears a ringing sound. The called user picks up the phone. This action sends an off hoo message back to the switch. The switch send an alerting message back over the SS7 network. Hangin up a phone disconnects the call.

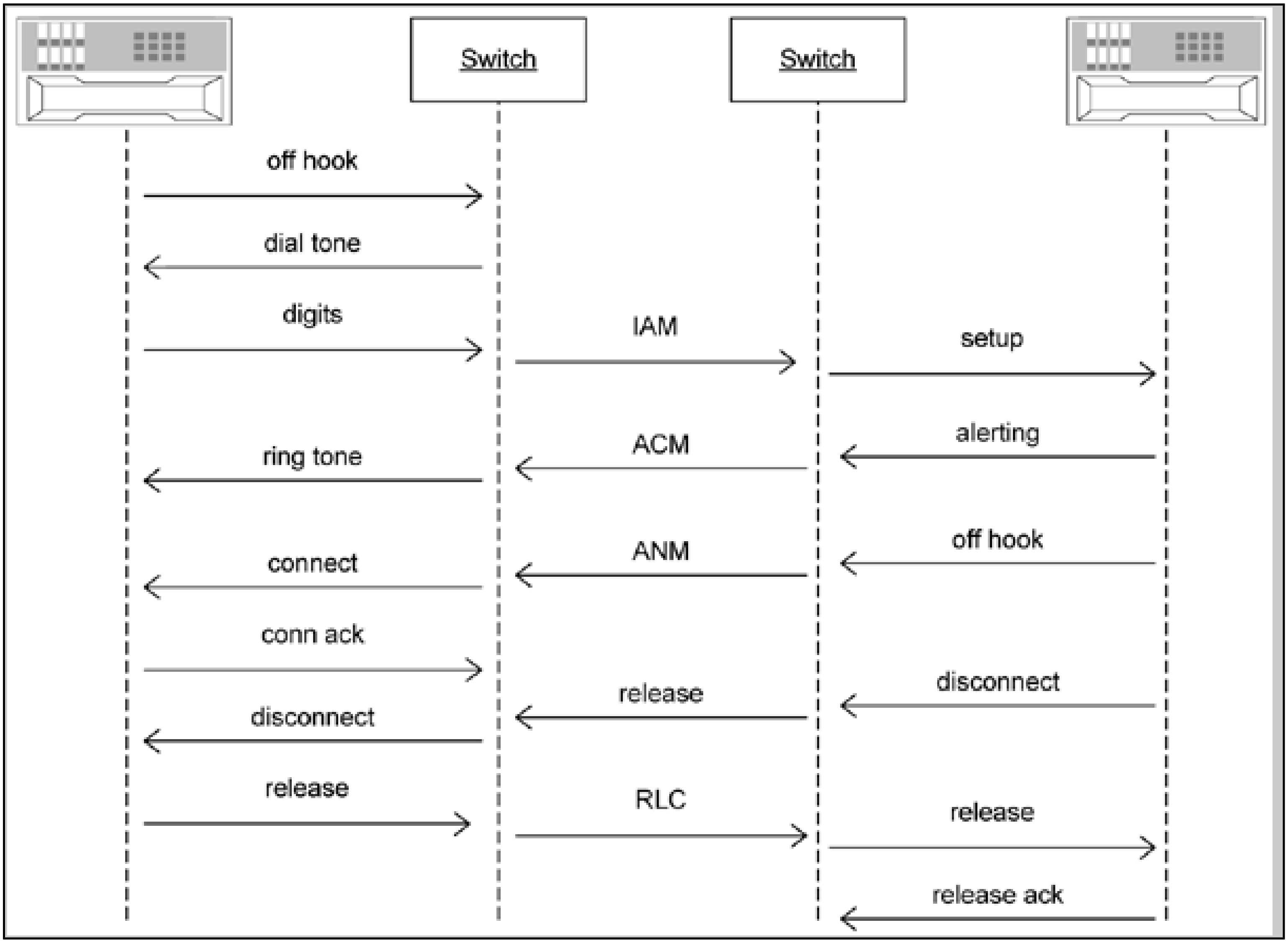


Figure: 16-4 SS7 Call Flow

PSTN Dial Plan

A local call can usually be dialed with seven digits. Dialing a long distance call requires dialing 1, and then an area code, and then the three digit exchange number, and then the last four digits of the telephone number. This scheme is the dialing plan for thePSTN.

The number of telephone numbers that are needed has dramatically grown over the years. Because of this, the current dialing plan may have to be changed to demand eleven digit dialing for all numbers.

Dial around is now available for a user to specify a long distance carrier. Dialing some number like 10+XX+XXX can switch a call to the desired long distance carrier.

The ITU-T Recommendation E.164 International Numbering Plan uses a Country Code (CC), national Destination Code (NDC) andSubscriber Number (SN) to switch a call to a user. The CC can be one, two or three digits. The NDC and SN can vary in length from country to country. Neither can have more than 15 digits.

The Future of the PSTN

The PSTN has held up well over the years for switching telephone calls from one user to another. On many networks built for voice, there is more data being sent than voice. This data is being sent over network that was optimized for voice. ThePSTN was never designed for data traffic and suffers for it.

In the near future, most voice will be carried as data over networks that were designed to carry data. In the future, more and more voice traffic will be sent over IP or ATM telephone company networks.

VoIP Standards

This chapter briefly addresses VoIP standards, especially H.323 and SIP. SIP is obsoleting H.323 so the emphasis is onSIP. For a more comprehensive discussion of SIP, consult the SIP standard or the book*Internet Communications Using SIP* by Henry Sinnreich.

You do not need an in-depth understanding of VoIP standards to build Asterisk systems or to use Asterisk. Asterisk hides most of the complexity of VoIP protocols for you. A more detailed understanding of these protocols could be necessary if you decide to become an Asterisk developer.

Open VoIP separates calling into bearer (IP, RTP) streams, services and call control. Standards define each of these three protocol stacks.

Packet Networks

This book assumes you are already familiar with networks and TCP/IP. There is no attempt here to describe basic networking. There are many excellent references for this.

Data networks, both IP and ATM, are packet based. Packet networks are obsoleting circuit switched networks.

IP is particularly attractive for data transport. IP is a transparent transport layer. It is a widely adopted standard and provides the most common application interface. IP transparently transports data endto-end regardless of the application.

Packet loss is common in IP networks. IP networks are self-healing. Dynamic routing protocols allows a network to re-converge to overcome packet loss or to find the best possible route. Dynamic routin means the packets in a data stream can travel separate paths. This means that packet transit and arriva times can vary from packet to packet.

Packet loss is a normal occurrence in an IP network. TCP/IP uses packet loss to control packet flow. If a packet is lost,TCP re-sends the packet. TCP uses packet loss to tune packet transmission.

ITU-T recommends a one way packet delay of no more than 150 ms. This is why TCP suffers over a satellite link.TCP does not deal well with the extremely long propagation delays of a satellite link.

IP does not directly support real-time traffic sessions. Real-Time Transport Protocol (RTP) is the emergent protocol for real-time traffic sessions over IP networks. The packets for a particularRTP session are referred to as an*RTP stream* or a *media stream*. RTP is commonly used to transport voice traffic. Many applications, for example Microsoft Net Meeting, useRTP.

In a real-time environment like voice, re-sending a lost packet is too time consuming. Small numbers of lost packets in a voice stream are not noticeable to a listener. It's better to ignore the lost packet than re-transmit them. UnlikeTCP, UDP is an unreliable protocol. That is, there is no guaranteed delivery of a packet withUDP. This is one of the reasons why RTP runs over UDP instead of TCP.

Packets that are part of a real-time session can arrive out of order. RTP packets each contain a timestamp. The timestamp allows the receiving application to reassemble incoming packets in the correct order.RTP uses the packet timestamps to tune its settings. RTP can use the timing information to adjust for network problems like delay and jitter as well as packet loss

Open Call Control

Call control is the process of managing and routing a call. For the PSTN, management and routing are both managed by SS7. VoIP IP bearer streams are separate from call control.

An enterprise class switch is circuit switched. Like the PSTN, channels are usually 64 kbps. The PSTN and enterprise switches can both offer services like call waiting, call hold and call transfer. While a Class 5 switch can handle hundreds of thousands of simultaneous calls, enterprise switches ar typically much smaller.

Class 5 is an telephone industry call control standard. Central office switches use Class 5. Most enterprise switches use proprietary manufacturer protocols. Most proprietary enterprise switches provide advanced features that are not available on Class 5 switches. Class 5 switches were developed to support residential telephony, not complex business services. Enterprise switches typically provide much, much richer feature set. The high-use feature-rich services available on proprietary enterprise switche are available on Aterisk.

There are a variety of IP routing protocols including Router Information Protocol (RIP,) Interior Gateway Routing Protocol IGRP,) Enhanced Interior Gateway Routing Protocol (EIGRP,) Intermediary System to intermediary System (IS-IS,) Open Shortest Path First (OSPF,) and Border gateway protocol BGP.) Each of these protocols provides a different solution to the problem of routing updates that solves a different problem. Each of these accomplishes the same thing, routing a packet from th source to the destination.

Similarly, there are several Internet open call control protocols. They all resolve traffic to IP addresses. They currently include H.323, SGCP, MGCP, andSIP. There are proprietary protocols like the Cisco Skinny protocol. More protocols will appear in the future to address new needs.

There is no need to standardize on a single call control protocol. These protocols enable standards for applications at the call-control layer. With the open protocols, applications from different vendors ar interoperatble. Asterisk operates with many of these protocols includingSkinny.

H.323 is currently the most widely deployed VoIP call-control protocol. H.323 is not robust enough to use in a system that can compete with the SS7PSTN. SIP is the most likely packet based competitor to SS7.

H.323

H.323 is an International Telecommunications Union Telecommunications Standardization Sector (ITU-T) specification for transmitting multimedia traffic including video and voice over an IP network. H.323 works with other existing standards like Q.931. Compliant vendor products an applications can communicate with each other via this protocol.

H323 is complex. It's not easy to create H.323 applications. H.323 applications do not scale well.

H.323 comprises the following components and protocols

TABLE: 16-1 H.323 Protocols	
Feature	**Protocol**
Call Signalling	H.225
Media Control	H.245
Audio Codecs	G.711, G.722, G.723, G.728, G.729
Video Codecs	H.261, H.263
Data Sharing	T.120
Media Transport	RTP/RTCP

H.323 elements include terminals, gateways, gatekeepers and multipoint control units (MCU.)

Terminals, often called endpoints, provide point-to-point and multipoint conferencing for audio, video and data. Gateways can interconnect to thePSTN or ISDN networks.

Gateways are used to connect between a Switched Circuit Network (SCN) endpoints and H.323 endpoints. Gateways are only needed when an H.323 endpoint needs to interconnect to a different network.

Gateways provide address translation services and admission control. Gateways translate between audio, video and data transmission formats. Gateways interconnect communication systems and protocols

A gatekeeper provides pre-call and call-level control services to H.323 endpoints. H.323 gatekeepers are separated logically from the other network elements. Inter-gateway communications isn't currentl specified by H.323. A gatekeeper can provide call control signalling, call authorization, bandwidt management and call management functions.

A multipoint controller (MC) supports conferencing between three or more endpoints. A multipoint processor (MP) receives audio, video and data streams and then redistributes those streams to the endpoints in a multipoint conference.

An MCU is an endpoint that supports multipoint conferences. An MCU must include at least an MC and one or more MPs. A typical MCU for centralized multipoint conferences includes an MC, a audio MP, a video MP and a data MP.

An H.323 proxy server operates at the application layer. It examines packets sent between to communicating applications. The proxy supports reservations, H.323 traffic routing and Network Address Translation NAT.)

The following figure shows a sequence diagram for the call flows between two IP addresses. This example assumes that the two endpoints have already resolved each other's address.

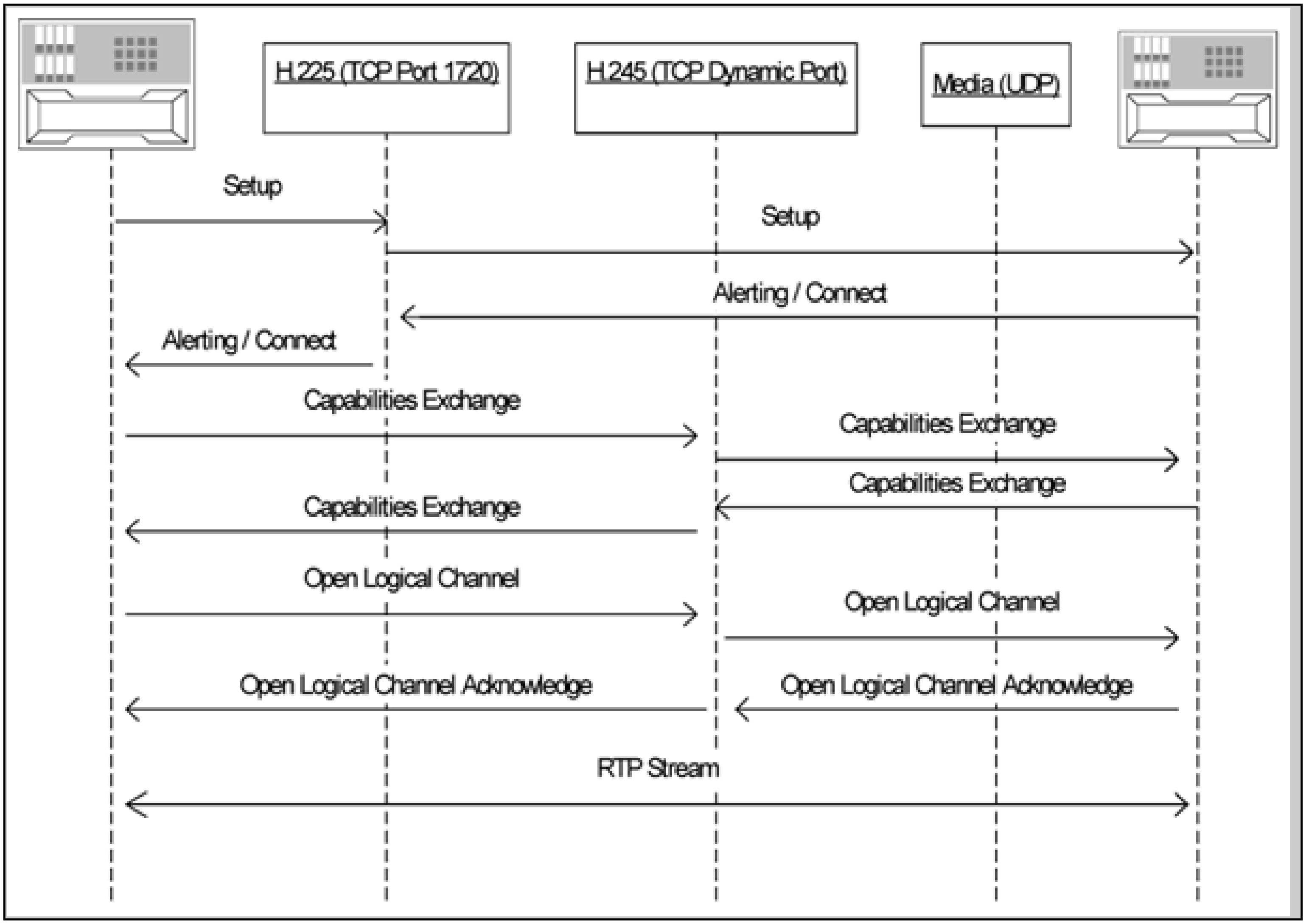


Figure: 16-5 H.323

In the example, endpoint one sends a setup message to endpoint B. This message is sent to TCP port 1720. Endpoint B replies with an alerting message that includes a port number. This message initiates Â

H.245 negotiations.

The H.245 negotiations setup the codec types and port numbers for the RTP streams. The Codec types are specified by G.729 and G.723.1. Any other capabilities the endpoints share are negotiated.

Logical channels for the UDP streams are negotiated, opened and acknowledged. The two endpoints can now send and receive the media stream containing the voice traffic.

Real Time Control Protocol can transmit information about the RTP stream to the two endpoints during the session

This call-flow shows an example of H.323 version one. H.323 version two allow H.245 to be negotiated through a tunnel in the H.225 setup message. This is called *fast-start*. A fast-start reduces the number of messages needed to initiate a call.

SIP

SIP is described in RFC.2543. SIP is an application-layer control protocol used to create, modify and terminate a communications session. ASIP invitation can establish sessions and describe sessions. SIP features of user location, user capability, user availability, call setup and call handling can initiate or en communications sessions.

Henning Schulzrinne, one of the original architects of SIP, said that the objective of SIP is the "re engineering of the telephone system from the ground up" He said this is an "opportunity that appears only once after 100 years"

A SIP session can have one or more participants. Sessions can include audio, video and data streams. SIP is flexible enough to support ad-hoc conferencing. Multi-media SIP sessions can be multicast, unicast, point-to-point, or combine broadcast methods.

While SIP is not yet as widespread as H.323, it is catching up fast. Most modern application implementations are relying on SIP rather than H.323. SIP is extensible and will easily support additional functionality as it is needed.SIP will outmode any proprietary protocols.

A sip user agent is a client end application continuing a user-agent client (UAC) and user-agent server (UAS.) These are know as aSIP client and SIP server. The client initiates SIP requests as a user's agent. A server gets requests. ASIP server acts as a user's agent.

There are two types of SIP network servers: proxy servers and redirect servers. Proxy servers contain client and server functions. A proxy server acts on the behalf of other clients. It can rewrite headers t identify the proxy as the request initiator. The proxy server makes sure that traffic is sent back to th correct client.

A redirect server accepts SIP requests and responds to the client with the address of the next server. A redirect server doesn't manage calls. A redirect server doesn't process or forwardSIP requests.

A SIP client must be able to locate a SIP server. A SIP client must determine the IP address and port number of a target server. The defaultSIP port is 5060. The SIP client can query a Domain Name Server DNS) for a sever IP address.

After SIP address resolution, the SIP client sends one or more SIP requests and gets back one or more SIP responses. All the requests and responses are part of a SIP transaction.

Signalling sets up, mantains and terminates calls. SIP provides a rich set of signaling facilities for VoIP. SIP can

- * Register IP phones.
- * Register other SIP devices.
- * Register end-user preferences.
- * Authentication, authorization and accounting.
- * Address resolution, name mapping, and call redirection.
- * Find the media capabilities of a target endpoint using Session Description Protocol.
- * Determine the availability of a target endpoint.
- * Establish a session between an originating and target endpoint.
- * Allow mid-call changes like the addition of another endpoint to a conference.
- * Report call progress including call success and failure.
- * Transfer and terminate.

SIP supports a variety of intelligent network services. These include:

- * Call Hold

- * Consultation Hold
- * Unattended Transfer
- * Unconditional Call Forward
- * Call Forward on Busy
- * Call Forward on No Answer
- * Three-Way Conferencing
- * Single Line Extension
- * Find-Me
- * Incoming Call Screening
- * Outgoing Call Screening
- * Secondary Number In
- * Secondary Number Out
- * Do Not Disturb
- * Call Waiting

SIP was designed to support multimedia conferencing. SIP also supports multimedia conferencing, multipoint conferencing and call control for conferencing.SIP enables instant messaging and instant communications.

What SIP Doesn't Do

SIP is a powerful, general protocol for establishing interactive communications sessions. SIP provides facilities for initiating, modifying and terminating interactive communications sessions.SIP is not a resource reservation or prioritization protocol. There is no Quality of Service (QOS) support inSIP. SIP is not a data transport protocol. SIP is not designed for managing interactive sessions after the sessions have been established. SIP is not designed to replace all the features and services provided by the PSTN. Many of the Class 5 features are not needed in the context of the Internet. Some features are provided by other protocols besidesSIP.

SIP Elements

SIP elements are User Agents, Servers and Location servers. User Agents are the endpoints of a SIP network. User Agents originateSIP requests to start and stop sessions and to send and receive data. A User Agent can be a hardware phone, a software phone running on a PC, or a gateway to another network like thePSTN.

Every SIP User Agent includes a User Agent Client and a User Agent Server. A User Agent Client (UAC) is the component of the User Agent that initiates requests. The User Agent server (UAS) is th component of the User Agent that responds to requests. Both are typically used during aSIP session.

Servers are intermediaries. They help User Agents establish and manage a SIP session. There are three types ofSIP server. SIP proxies forward SIP requests. Redirect servers get a request from a user agent, they return an indication of where the request should be resent to. Registrar servers update location o other database information.

Location servers maintain databases of information like URLs, IP addresses, scripts, features and preferences. User agents usually interact with Location Servers through a SIP proxy.

Addressing

SIP Uniform Resource Locators (URLs) provide addressing similar to e-mail addressing. A SIP URL can have various forms and can include a telephone number, for example,

```
sip:someone@somewhere.com
sip:1-415-555-1212@somewhere.com; user=phone
sip:1-415-555-1212@somewhere.com; user=phone; phone-context=VNET
```

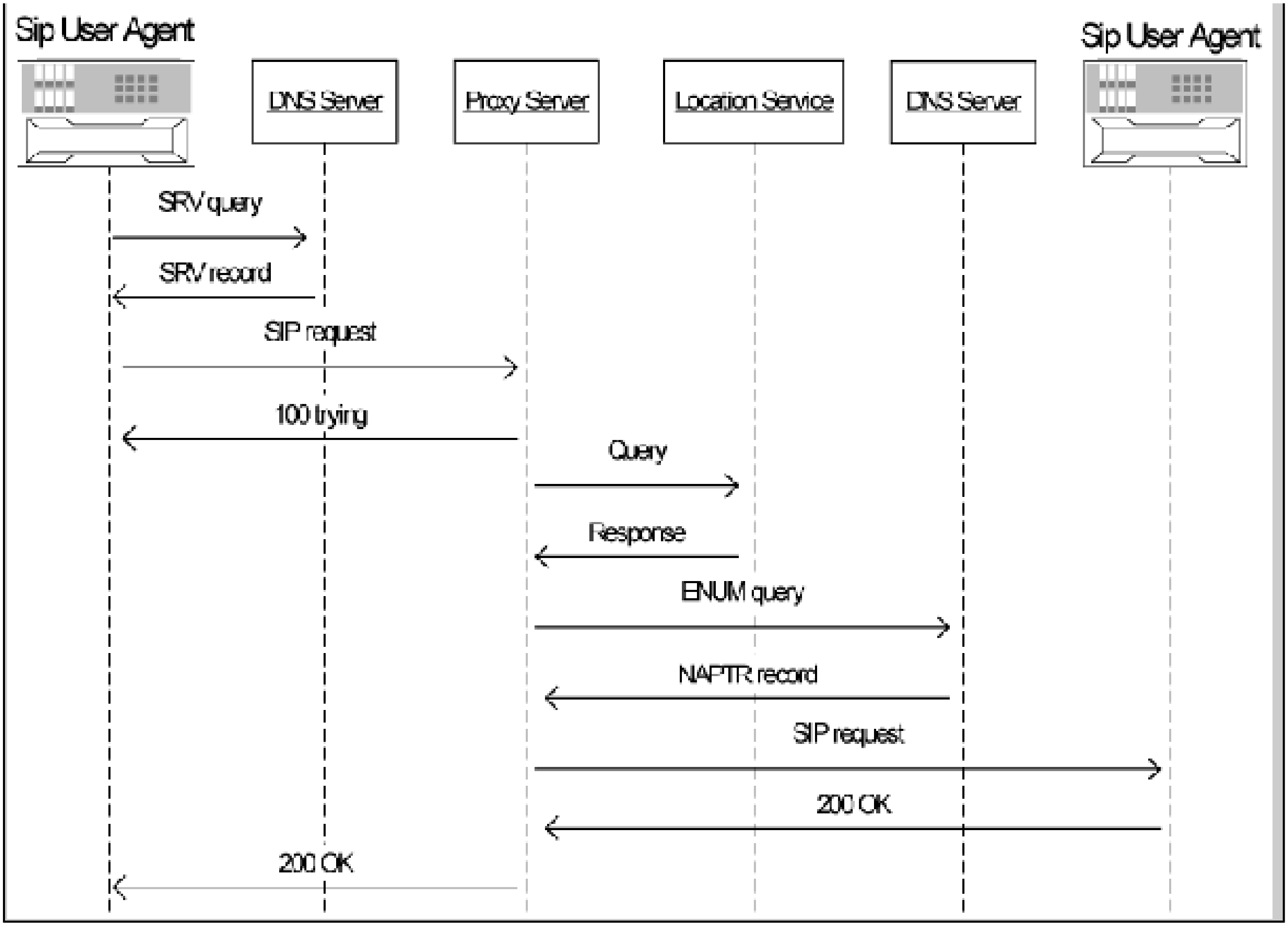


Figure: 16-6 SIP Address Resolution

SIP support of telephone number addressing and Web addressing supports bridging between the two networks. If aSIP endpoint knows the URL of another SIP endpoint, direct communications is possible.

SIP address resolution starts with a URI that resolves to a username at an IP address. The figure above shows a sequence diagram for a typical address resolution sequence where a URI is resolved to a user a an IP address.

Session Setup

Session Setup is the primary function of SIP. SIP sends an invite request. The invite request can contain a message describing the desired session type. The following sequence diagram shows a typical session setup.

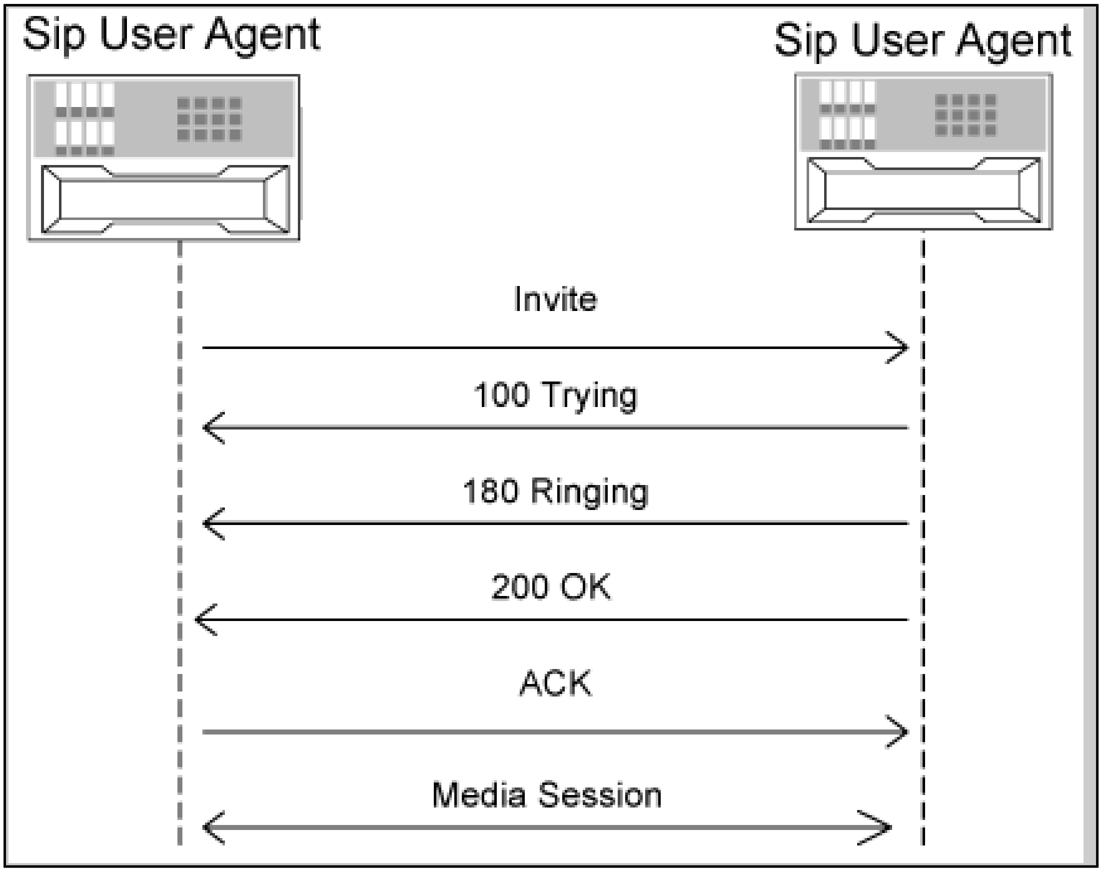


Figure: 16-7 SIP Session Setup

This has been a fast introduction to a very complex topic. For more information please consult one of the excellent references.

Glossary

Note - see the excellent and more comprehensive references at

http - //www.its.bldrdoc.gov/fs-1037/ Â

http - /isp.webopedia.com/

Abandoned Call

A call that is disconnected after a connection has been made to the called telephone but before the call is established

---

Abbreviated Dialing

A method of allowing a user to dial a call with fewer than the usual number of required numbers

---

Access

A means by which Company service is provided to a Customer. Access may be "Dedicated," in which case it is available to theCustomer on a full-time, unshared, basis, or it may be "Switched," in which case it is available to theCustomer and others on a usage, shared, basis.

Access Service Request

An order placed with a Local Access provider for Local Access.

Add On Conference

A call where additional users are added to a conversation without operator intervention.

ANI

See automatic number identification.

Alternate Access

Access to the PSTN provided by a vendor who is not a LEC but is authorized or permitted to provide services

---

Alternate Access Carrier

Provides access in competition with local exchange carriers or RBOCs.

Area Code

See Numbering Plan Area.

Automatic Number Identification

Provides the telephone number of the calling party.

Answer Supervision

When a called station answers, an off-hook signal is sent to the call originator.

Ballot

A release form a customer competes to switch between long distance carriers or resellers.

BAN

See Billing Account Number

Bearer Channel

A communications channel used for transmitting an aggregated signal generated by multi-channel transmitting equipment. Also the designation of a 64 kbs channel provided to anISDN user

---

BGP

Border Gateway Protocol. Border Gateway Protocol (BGP) is an inter-autonomous system routing protocol. An autonomous system is a network or group of networks under a common administration and with common routing policies.BGP is used to exchange routing information for the Internet and is the protocol used between Internet service providers (ISP).

Billing Account Number

A designated billing account, a customer or customer location where the bill is sent. A single customer can have multiple BANs

---

Banded Rates

Tarriffed Rates which a carrier can change at their discretion within a certain range.

Bell Customer Code

A three digit number appended to the end of a billing account number to assist in the unique identification of a customer

---

Bell Operating Company

A local or regional telephone company that operates local exchanges.

BOC

See Bell Operating Company Â

BGP

Border Gateway Protocol Â

Bong

An sound used to prompt a user to enter additional information. For example, after typing 1010555 a bong might sound to indicate that the user should enter an billing code

---

Billing Telephone Number

The phone number calls are billed to. The calling number can differ from the billing number

---

Bypass

Access to an alternate IEC by dialing an access code. For example, dialing 1010222 at the beginning of a call might access Sprint long distance

---

Call Data Record

A record of a call including the time the call was placed and the length of the call.

Called Station

The station called, or the terminating point of a call.

Calling Station

The station at which a call is originates.

Caller ID

The transmission of the telephone number of the calling party.

Calling Card

A credit card accepted by a telecommunications carrier. Typically used for charging telephone calls when the user is away from their home or office

---

Carrier Identification Code

A three digit number used with Group B and D feature groups to access a IECs switched services from a local exchange.

Casual Customer

Any person that dials a CIC code without necessarily being presubscribed to the carrier

---

CAT5

Category 5. An ethernet standard describing the physical characteristics of a cable and connector.

Centrex

Services typically provided to a user by a PBX that are instead hosted at a central office.

Channel or Circuit

A communications path between two or more points.

Channel Associated Signaling (CAS)

See Robbed Bit Signaling

Channel Termination

The point at which the Company's channel originates, terminates, or drops for the insertion or removal of aCustomer's signal.

CIC

See Carrier Identification Code.

Class of Service

The limits on what numbers can or cannot be called, for example local, statewide, international, etc

---

CDMA

Code Division Multiple Access - an American standard for encoding cellular telephone calls

---

CLEC

See Competitive Local Exchange Carrier

Collect

A call paid for by the party receiving the call.

Commercial Service

A switched network service involving dial station originations for which the Customer pays a rate that is described as a business or commercial rate in the applicable local exchange service tariff for switched service

---

Competitive Local Exchange Carrier

Companies that compete locally for telecommunications services, for example telephone, Internet access, cable TV, etc.

Common carrier

A telecommunications company that provides communications transmission services.

Computer Telephony Integration

The extension of computing over the telephone network to a telephone, or access to telephony from a computer.

Contract Tariffs

Rates and services contracted with an individual customer, but available to all customers of the operating company.

Country Code

Two or three digits used to identify the foreign destination country of a telephone call.

Customer

The person, firm, corporation or other entity which orders service and is responsible for the payment of all charges for service and for compliance with Company contract and tariff requirements. The term"customer" includes a person, firm, corporation or other entity that either knowingly or unknowingly accesses service and completes a communication over the Company's network. Fo Resp Org Service, theCustomer is the person, firm, corporation or other entity that selects or is directed to select the Company as the Responsible Organization (Resp Org) for a toll-free telephon number. For purposes ofSMS Resp Org Changes, the customer is the person, firm, corporation, or other entity that submits the change request

---

Customer Premises

A Customer or Authorized User location at which service is provided.

Cutover

The time and date that a change is to be made between services or implementations.

CTT

See Computer Telephony Integration.

DAL

See Dedicated Access Line.

DDD

See Direct Distance Dialing.

DDR

See Dial on Demand Routing

Dedicated Access Line

A non-switched circuit between a carrier and a customer.

Dedicated Access/Termination

An access line service consisting of a continuously connected circuit between aCustomer Premises or serving telephone company central office and a Company terminal, available to theCustomer on a full-time, unshared, basis, which is used for the origination or termination of services.

Dedicated Line

A private line leased from a telecommunications carrier.

Dial

Place a call on a switched telephone network. This term springs for a time when telephones had dials instead of buttons

---

Dial on Demand Routing

A data connection established via dial up service

Dial

Place a call on a switched telephone network. This term springs for a time when telephones had dials instead of buttons

---

Dial Plan

The organization that determines how calls are routed through an Asterisk system.

Dial Tone

An audible tone used to indicate a call can be dialed.

Dialer

Equipment that sends standard dialing signals.

Digital Signal

A signal where data is transmitted in discrete steps

Digital Signal One

A digital signaling rate of 1.544 Mbs corresponding and North American T1 designation.

Digital Signal One C

A digital signaling rate of 3.152 Mbs corresponding to a North American T1c designation

---

Digital Signal Two

A digital signaling rate of 6.312 Mbs corresponding to a North American T2 designation

---

Digital Signal Three

A digital signaling rate of 44.736 Mbs corresponding to a North American T3 designation

---

Digital Signal Four

A digital signaling rate of 274.176 Mbs corresponding to a North American T4 designation

Digital Signal Zero

A 64 kbs signal corresponding to the data rate of a single voice-frequency equivalent channel.

Digital Subscriber Line

A method of sending high speed digital data over a telephone circuit.

DNS

Domain Name Server

DS1 to DS4

See Digital Signal One to Digital Signal Four

DSL

See Digital Subscriber Line Â

DSP

Digital Signal Processor Â

Due Date

The date on which payment for service by the Customer is due.

End-to-End

Customer Premise to Customer Premise

EIGRP

Enhanced Interior Gateway Routing Protocol

Equal Access

The provision for reaching an inerLATA carrier with an access code. The right of a user to select the long distance provider or local provider of their own choice

---

Exemption Certificate

A written notification provided by a Customer certifying that its dedicated facility should be exempted from the monthlySpecial Access Surcharge because - (a) the facility terminates in a device not capable of interconnecting service with the local exchange network; or (b) the facility is associated with aSwitched Access Service that is subject to Carrier Common Line Charges.

Expedite

A Service Order that is processed at the request of the Customer in a time period shorter than the Company standard Service interval

---

Extension context

A group of extensions.

FBC

See Facilities Based Carrier.

Facilities Based Carrier

A carrier with their own facilities as opposed to a reseller of another companies services that has no equipment of their own.

FCC

Federal Communications Commission.

File Transfer Protocol

An internet protocol used for transferring files. FTP uses TCP/IP.

Foreign Exchange

An exchange that is not a user's local exchange. (see local office)

Foreign Exchange Office

Synonym for foreign exchange.

Foreign Exchange Service

A service provided by a foreign exchange. A network-provided service where a telephone in a local exchange area is connected, via a private line, to a central office in another "foreign" exchange instead of the local exchange area's central office. Note - To call originators, the subscriber having the FX service appears to be located in the foreign exchange area

---

FTP

See File Transfer Protocol **FX** - see Foreign Exchange.

FXO

See Foreign Exchange Office.

FXO port

A port used to connect to a DID line.

FXS

See Foreign Exchange Service

FXS Port

A port used to connect to a local analog telephone device.

GSM

Global System for Mobile Communications. A European protocol used for encoding cellular telephone calls

---

Hang Up

End the telephone connection.

IC

See Interexchange Carrier

ILEC

See Incumbent Local Exchange Carrier Â

Incumbent Local Exchange Carrier

The dominant phone carrier providing exchange service within a geographic area as determined by theFCC.

InterExchange carrier

A company that provides long distance services between LECs and LATAs.

In Band

Signals sent over the same bandwidth as the data.

Installation

The provision of connections for new or additional service.

IGRP

Interioe Gateway Routing Protocoll

Institutional Phones

Telephones, other than payphones, located in public institutions such as universities, prisons, or public offices, or in hotels or motels, or in other premises where the Customer may not be able to control access to the phones

---

Integrated Services Digital Network

A set of communications standards providing digital network services

---

Interactive Voice Response system

An automated voice response system used to guide users through a series of choices

---

Interexchange

Communications between different LATAs.

Interexchange Carrier

A company that provides long-distance telephone services between LECs and LATAs

---

Interexchange (IXC) Service

The portion of a Channel or Circuit between a Company designated Point-of-Presence in one exchange and a Company designated Point-of-Presence in another exchange

---

InterLata

Communications between Local Access Transport Areas.

Internet

With a small i as in internet, a network connecting differing subnets. With a capital I as in Internet, the global Internet connecting all publicly accessible internets.

Internet Service Provider

A company that provides Internet access to its customers.

Internet Telephony Service Provider

A company that provides customers with the ability to place telephone calls over theInternet.

Interstate

Between states.

IntrasInterruption

A condition that arises when service or a portion thereof is inoperativetate - within a single state

---

ISDN

See Integrated Services Digital Network.

ISTP

Individually Sheilded Twisted Pair

Kb

With a small b, kilo-bits. With a large B, kilo-Bytes.

Kbs

Kilo bits per second.

IVR

See Interactive Voice Response system.

IXC

See Interexchange Carrier.

Kewlstart

Loop Start with far end disconnection supervision. This allows the local device to detect when the remote device hangs up

---

LATA

See Local Access Transport Area.

Latency

The time between the transmission and arrival of a signal transmitted through a network.

Letter of Agency

See Ballot.

LEC

See Local Exchange Carrier.

LLP

See Local Loop Provider.

Local Access

The connection from a customer to their local office. The portion of service between a Customer Premises and a Company designated Point-of-Presence.

Local Access Channel

The connection between a Customer Premises and a Company Point-of-Presence.

Local Access Transport Area

By government regulation a geographical area within which a Bell Operating Company is permitted to offer Exchange Telecommunications and ExchangeAccess Services. A geographic area established by law and regulation for the provision and administration of telecommunications services.

Local Exchange

Synonym for a local office.

Local Exchange Carrier -A company which furnishes exchange telephone service. The local or regional telephone company that owns and operates local exchanges. . LECs have connections to other LECs or IECs Local Exchange Service

The service that provides a customer the ability to place local calls.

Local Loop

The connection from a user to a local office. The circuit connecting a customer's premise equipment to the local office

---

Local Loop Provider

The company that provides access to a local loop.

Local Office

A place where loops and trunks are terminated. Also the central office supplying users in a specified geographical area with telephone services

---

Loop Start

A signal sent by a telephone or PBX that indicates the loop path has been completed.

Message Toll Service

Switched long distance phone services between LECs and LATAs. Typically charged for by the minue.

Mb, mB

With a capital B, Mega Bytes. With a lower case m Mega bits.

---

mbps

Mega-bits per second

---

mbps

Mega-bytes per second Â

Modem

Modulator De-Modulator. A device used to send data over POTS lines by converting the data into sound

---

Multiline Terminating Device

Switching equipment, key telephone type systems or other similar customer premises terminating equipment which is capable of terminating more than one access line

---

MTS

See Message Toll Service.

NASC Number Search

An application used to find available numbers in the 800 area code and reserve them for up to sixty days

---

NAT

Network Address Translation

NEXT

Near End Cross Talk.

NPA

See Numbering Plan Area.

Numbering Plan Area

The North American three digit codes used to identify a specific calling area.

Numbering Plan Area Split

Division of an NPA by the addition of a new three digit code.

NUS

See NASC Number Search **OC** - see Optical Carrier **OCC** - See Other Common Carrier.

OSPF

Open Shortest Path First

One Plus Dialing

Access to long distance services by prefixing the dialed number with the digit 1.

Operator

Theperson who assists people in placing telephone calls.

Operator Service Call

A call placed with the assistance of an operator.

Operator Station

Service that requires the assistance of an operator to complete a call.

Optical Carrier

Series of physical protocols including defined for SONET optical signal transmissions. OC signal levels put STS frames onto multimode fiber-optic line at a variety of speeds. The base rate is 51.84mbps (OC-1); each signal level thereafter operates at a speed divisible by that number (thus, OC-3 runs at 155.52mbps).

Other Common Carrier

A common carrier that was not part of the original AT&T system.

Out of Band

Signals sent on a channel separate from the data.

PABX

Private Automatic Branch Exchange - see Public Branch Exchange.

PAX

Private Automatic Exchange - see Public Branch Exchange.

PBX

See Public Branch Exchange.

PCM

See Pulse Code Modulation

Personal Identification Number

A number used as a security code in order to restrict unauthorized access to an account or service

---

Person-to-Person

An operator assisted call only completed to a named individual.

PIC

See Primary Interexchange Carrier.

POTS

Plain Old Telephone Service.

PIC Freeze

Prevents long distance services from being changed to a new vendor.

PIC Request

A request sent to a LEC that contains a response code indicating if the requested service was performed.

PIN

See Personal Identification Number.

Point-of-Presence

A location where a Company maintains a Terminal Location for purposes of providing service.

POP

See point of presence

Primary Interexchange Carrier

The IEC that One Plus Dialing calls are routed through.

PRI

See Primary Rate Interface.

Primary Rate Interface

A type of ISDN interface providing 23 bearer channels and 1 data channel.

Private Line

A dedicated circuit connecting customer equipment at both ends of the circuit. The private line does not include any switching services.

Provisioning

The process of designing, implementing and tracking the fulfillment of a service order.

Promotion

Periodic financial inducement offered by the Company to new and/or existing Customers of service to subscribe to and use new or additional service(s).

PSTN

Public Switched Telephone Network.

Public Branch Exchange

A telephone system within an enterprise that switches calls between enterprise users on local lines and allows all users to share external phone lines. APBX saves the cost of every user having a line to the telephone company

---

In older usage, a private telephone switchboard that provided on-premises dial services.

Public Utilities Commission

An agency that regulates intrastate telecommunications services.

PUC

See Public Utilities Commission.

Pulse Code Modulation

A signal is sampled, then the magnitude (with respect to a fixed reference) of each sample is quantized and digitized

---

QoS

Quality of Service

Rate Center

A specified geographical location used for determining mileage measurements

---

Rate Element

A low level component of a recurring fixed charge for IEC or LEC services.

Rates and Tariffs

Published standards that define what services are available, how much they cost, and how they are provisioned

---

RBOC

See Regional Bell Operating Company.

Real Time Transport Protocol

A protocol for transmitting and re-assembling IP data packets.

Redundancy

An offering of alternate service through the use of one or more different routings, circuits, and/or additional equipment Â

Regional Bell Operating Company

One of the seven "Baby Bell" operating companies. One of the seven LECs established in the U.S. Department of Justice 1984 Consent Decree with A&T. The RBOC carriers are Ameritech, Verizon (NYNEX) or Verizon North, Verizon (Bell Atlantic) or Verizon South, Bell South, Pacific Bell (PacBell), Southwestern Bell and US West (Qwest).

Regulators

FCC, PUC, Federal Courts, ETC.

Requested Service Date

The date requested by the Customer for the commencement of service and agreed to by the Company

---

Reseller

An IEC that leases bulk capacity and then resells some of it at a higher rate.

Residential Customer

An individual, non-business telephone customer.

Restoration

The re-establishment of service.

RIP

Router Information Protocol

Robbed Bit Signaling

The same as Channel Associated Signaling (CAS). A method of signaling each traffic channel instead of having a dedicated signaling channel (likeISDN). The signaling for a circuit is permanently associated with that circuit. The common forms are loopstart, groundstart Equal Access North American (EANA), and E&M. The disadvantage of CAS signaling is its use of user bandwidth for signaling. As well as call reception, CAS signaling can processes Dialed Numbe Identification Service (DNIS) and automatic number identification ANI) information.

Route Diversity

Two channels furnished partially or entirely over two physically separate routes.

RTP

See Real Time Transport Protocol.

Service Management System

A system used to manage services.

Simple Network Management Protocol

A protocol that provides for the remote management of network connected equipment.

SIP

Session Initiation Protocol.

Skinny

Cisco proprietary VoIP protocol.

Slam

Changing a customers long distance provider without their permission.

SMS

See Service Management System.

SNMP

See Simple Network Management Protocol.

SONET

See Synchronous Optical Network

Special Access Surcharge

A charge imposed by a Local Exchange Carrier in accordance with Section 69.115 of the FCC Rules and Regulations.

Speed Dialing

A service to dial numbers by dialing fewer than the usual number of digits.

State Tax

The taxes that each state is allowed to charge. States are allowed to charge taxes on a call if two out of the three following conditions are met -the call originates in the state, the call terminates i the state or the call is billed within the state

---

Station

Telephone equipment from or to which calls are placed.

Station-to-Station

A directly dialed call where no operator is used.

Subscriber

The ultimate user of the PSTN.

Surcharge

A charge that is in addition to the normal base charge.

Switch

A telecommunications product that connects incoming data to the correct destination.

Switched Access

Non-dedicated access between a user and their local carrier.

Switched Access Service

A class of LEC services providing switched services from a customer's premises to the IEC. An service consisting of an occasionally connected circuit between a Customer Premises or serving telephone company central office and a Company terminal, available to the Customer on a usage, shared, basis, which is used for the origination or termination of service

---

Switched Reseller

Resellers selling services with their own hardware.

Switching Fee

A per-line fee imposed by a LEC to reprogram their switch when a user changes to a new carrier. This fee is usually paid when a user changes to a reseller

---

Switchless Reseller

A reseller of long distance services that does not own or operate its own switches or lines

---

Synchronous Optical Network

A standard for optical telecommunications data transport developed by the Exchange Carriers Standards Association (ECSA) for the American National Standards Institute (ANSI.) ANSI sets industry standards in the U.S. for telecommunications and other industries

---

T1 or DS-1

A high speed telephone connection providing 1.544 mb of bandwidth.

T2 or Ds-2

The equivalent of four T1 lines providing 6.312 mb of bandwidth.

T3 or Ds-3

The equivalent of 28 T1 lines providing 44.736 mb of bandwidth.

T4 of Ds-4

The equivalent of six T3 channels providing 274.176 mb of bandwidth.

T-Carrier

The generic designation of several different digitally multiplexed telecommunications carrier systems.

TCP

See Transmission Control Protocol.

TDD

Telecommunications Device for the Deaf.

Tariffs

See Rates and Tariffs.

Telco

See Telephone Company.

Telephone

User equipment used for sending and receiving voice frequency signals including voice and touch tones

---

Telephone call

A connection maintained over time used to send and receive voice frequency signals.

Telephone Company

A company that owns and operates lines to customer locations and central offices

---

Terminal Equipment

Devices, apparatus and their associated wiring, such as teleprinters, telephone handsets or data sets, interconnected to service

---

Telephone Switch

A switch that switches telephone calls.

Termination Gateway

Computer equipment that provides an interface between an IP network and thePSTN.

Terms of Service

The body of prescribed rules governing the offering and furnishing of service, including"general" and "service-specific" terms contained in this tariff, as supplemented by any additional or alternative terms in a contract.

TFTP

See Trivial FTP

Third Party Billing

Use of an outside provider for bill processing.

Time of Day Routing

Call routing based on the time of day. Used to reduce the cost of calls.

Toll

A charge for a telephone call.

Toll Call

A call that has an incremental charge.

Toll Fraud

The illicit access to long distance services.

Transmission Control Protocol

A reliable protocol for moving packets of data, often over an IP network.

Trivial FTP

Trivial File Transfer Protocol -a simple implementation of FTP. TFTP uses UDP and has no security features.TFTP is used to transfer a boot image from a server to peripheral equpment like diskless workstations, routers, x-terminals and ip telephones

---

Trunk

One of several phone lines that originate and terminate in the same location.

Trunk Group

Telephone lines that originate and terminate in the same location.

UDP

See User Datagram Protocol.

UTP

Unshielded Twisted Pair.

U.S. Mainland

The District of Columbia and the 48 conterminous states.

U.S. Territories

Puerto Rico, the U.S. Virgin Islands, Guam, the Commonwealth of the Northern Mariana Islands, and American Samoa

---

User Datagram Protocol

An unreliable protocol used for transmitting data packets, typically over an IP network

---

Voicemail

A system that receives, stores, plays and manages voice messages.

---

voicemail Box

The storage area for voice messages.

WATS

See Wide Area Telephone Service.

Wide Area Telephone Service

A special tariff for a specified calling area.

Wide Area Network

A network over several locations that are widely separated.

Wire Center

The service area where a Customer Premises would normally obtain exchange service or dial tone from anILEC.

Wireless

Transmission without a wire, typically by radio or light waves.

Wireless Number Portability

The service allowing a customer to retain their phone number when moving to a new provider

---

WNP

See Wireless Number Portability.

Working Telephone Number

A telephone number with established operational telephone service.

WTN

See Working Telephone Number.

Checklist

Pre-Installation

TABLE: checklist-1 Site Installation Information
Company Name
Site Street Address
City
State
Zip
Site Contact Name
Telephone Number
E-Mail Address
Cell Number
Pager Number

TABLE: checklist-2 Pre-Installation Requirement
Network diagram displaying all devices
Electrical power outlets available
Outlets close enough to equipment to meet local codes
Air conditioning required
Air conditioning capacity
Air conditioning outlet close enough to equipment
Lan connections next to system location
110 or 66 blocks clearly marked
Cell Number
Pager Number

TABLE: checklist-3 T1
Provider company name
Provider comapny contact
Contact Phone number
Contact email
Contact cell phone number
Circuit ID
Circuit completed and tested?
Framing
CSU/DSU Data Port Number
Telephone numbers

TABLE: checklist-4 SIP Provider
Provider company name
Provider comapny contact
Contact Phone number
Contact email
Contact cell phone number
Circuit ID
Circuit completed and tested?
Telephone numbers

TABLE: checklist-5 IP
IP address for Asterisk server Subnet Mask?
Router address (default gateway)
Primary DNS Server
Secondary DNS Server

TABLE: checklist-6 Frane Rekat
Provider company name
Provider comapny contact
Contact Phone number
Contact email
Contact cell phone number
Port Speed
Circuit completed and tested?
PVC CIR
Circuit Number
LMI Type
Carrying voice and data on the same PVC?

TABLE: checklist-7 Asterisk Server
Provider company name
Provider comapny contact
Contact Phone number
Contact address
Contact city
Contact state
Contact zip
Contact phone number
Contact cell phone number
Computer Model
Processor Speed
Memory
Controller Type (SCSII/IDE)
RAID (YES/NO)
Disk 1 Size

Disk 2 Size
Disk 3 Size
Disk 4 Size
Removeable media 1 (CD-ROM/DVD-ROM/CD-RW/DVD-RW)
NIC 1 - 10 or 100 or gigabig
NIC 2 - 10 or 100 or gigabit
Removeable media 2 (CD-ROM/DVD-ROM/CD-RW/DVD-RW)
USB Ports (USB-1/USB-2)
Number of USB Ports
Monitor Type
Monitor Size
Keyboard
Mouse
Maintenance Contract ID
Maintenance contract expires
Maintance Contact Name
Maintance Contact Telephone Number
Maintance Contact Hours
Maintance Contract agreeed response time
Linux Version
Linux Provider

TABLE: checklist-8 Network Equipment
Provider company name
Provider comapny contact
Contact Phone number
Contact email
Contact cell phone number
Equipment Type (router, switch)
Model
Power over Ethernet?

TABLE: checklist-9 Electrical
Provider company name
Provider comapny contact
Contact Phone number
Contact email
Contact cell phone number
Required service size
Circuit completed and tested?
Outlet within five feet of equipment?
UPS Required
UPS Model
Available standby time

TABLE: checklist-10 Telephones
Provider company name

Provider comapny contact
Contact Phone number
Contact email
Contact cell phone number
Telehpone Model
Desciption (e.g. for speaker phone)
Analog or IP
SIP Version Installed
SIP Version Availalbe
Service contract number
Service contract end date
Service contact name
Service contact hours