Asterisk - The Basics

PacNOG 3 VoIP Workshop
June 2007, Cook Islands

Jonny Martin
jonny@jonnynet.net

What is Asterisk

- Asterisk, *The* Open Source PBX. www.asterisk.org
- A complete PBX in software
- Runs on virtually any OS
- Support for most VoIP protocols
- Most full-featured PBX features already built in
    - MOH, conferencing, queues, voicemail, IVR...
- Supports many different hardware telephony cards

Asterisk Documentation

- There's lots of info all over the place, some of it contrary though
- www.voip-info.org
  - Lots of really good information, lots of plain wrong information too!
  - Defacto documentation store at this stage
- www.asterisk.org
- www.digium.org - hardware cards
- Asterisk CLI !

Useful Reading

- Asterisk, The Future of Telephony. By Jared Smith, Jim Van Meggelen, Leif Madsen. ISBN: 0-596-00962-3
  - Published under Creative Commons license
  - Can download, or buy a real book from O'Reilly
  - http://www.asteriskdocs.org/modules/tinycontent/index.php?id=11

Asterisk Versions

- Three versions currently in popular use:
  - 1.0 - becoming obsolete rapidly, but it’s good and stable
  - 1.2 - the current release of choice for most, stable
    - We’ll be dealing with v1.2
  - 1.4 - all the new features in here, still a few bugs

Installing Asterisk

- Asterisk uses three main packages:
  - asterisk
  - zaptel
  - libpri
- Compile Requirements:
  - GCC (version 3.x or later)
  - Kernel source
  - Kernel headers
  - bison
  - openssl, openssl-dev, libssl-dev
  - libnewt

Download Source

```
# cd /usr/src/
# wget --passive-ftp ftp.digium.com/pub/asterisk/asterisk-1.*.tar.gz
# wget --passive-ftp ftp.digium.com/pub/asterisk/asterisk-sounds-*.tar.gz
# wget --passive-ftp ftp.digium.com/pub/zaptel/zaptel-*.tar.gz
# wget --passive-ftp ftp.digium.com/pub/libpri/libpri-*.tar.gz
```

```
# tar zxvf zaptel-*.tar.gz
# tar zxvf libpri-*.tar.gz
# tar zxvf asterisk-*.tar.gz
# tar zxvf asterisk-sounds*.tar.gz
```

*** If using Linux kernel 2.4 a symbolic link named linux-2.4 is required pointing to your kernel source:**

```
#ln –s /usr/src/`uname –r` /usr/src/linux-2.4
```

Compile Zaptel

- Several features in Asterisk require an accurate timing source, e.g. conferencing
- Digium PCI hardware provides this 1kHz timing clock
- If you aren’t using PCI hardware the *ztdummy* driver can be used
  - Kernels 2.4.5 and greater use the UHCI USB controller for this (so you need the *usb-uhci* module loaded)
  - The 2.6 kernel provides a 1kHz so a USB controller is not needed
- Need to uncomment out ‘ztdummy’ in Makefile

```
MODULES=zaptel tor2 torisa wcusb wcfxo wctdm \
        ztdynamic ztd-eth wct1xxp wct4xxp wcte11xp # ztdummy
```

Compile Zaptel

```
# cd /usr/src/zaptel-version
# make clean
# make
# make install
# make config
```

- Also installs some tools:
  - *ztcfg* - reads config in */etc/zaptel.conf* to configure hardware
  - *zttool* - for monitoring installed hardware
  - *ztmonitor* - for monitoring active channels
- zconfig.h contains many zaptel compile-time options - echo cancellation options, RAS options, etc.

Compile Libpri

```
# cd /usr/src/libpri-version
# make clean
# make
# make install
```

- Used by many manufacturers of PCI TDM cards
    - Safe to compile even if a card is not installed/used

Compile Asterisk

```
# cd /usr/src/asterisk-version
# make clean
# make
# make install
# make samples
```

Package Install

- Much easier to use pre-compiled binary packages!
  - RPM packages for redhat
  - DEB packages for Debian
  - Asterisk.pkg for MacOSX http://www.astmasters.net
- We'll be using Debian .deb packages
  - Debain testing
  - Asterisk version 1.2

Debian Install

```
apt-get install asterisk
apt-get install asterisk-sounds-extra
apt-get install zaptel
apt-get install zaptel-source
apt-get build-dep asterisk
   * if you need ztdummy:
   m-a prepare
   m-a build zaptel

dpkg -i zaptel-modules-xxxxxx.deb
depmod
modprobe zaptel
modprobe wcte11xp   # if using TE110P single span T1/E1 card
modprobe wcfxo      # if using single port FXO card
modprobe ztdummy    # if using ztdummy

ztcfg
zttool

nano /etc/default/asterisk
* To get ztdummy, modify Makefile to uncomment ‘ztdummy’
* On Debian, add ‘ztdummy’ to /etc/module to get ztdummy to load at boot
* set RUNASTERISK=yes in /etc/default/asterisk
```

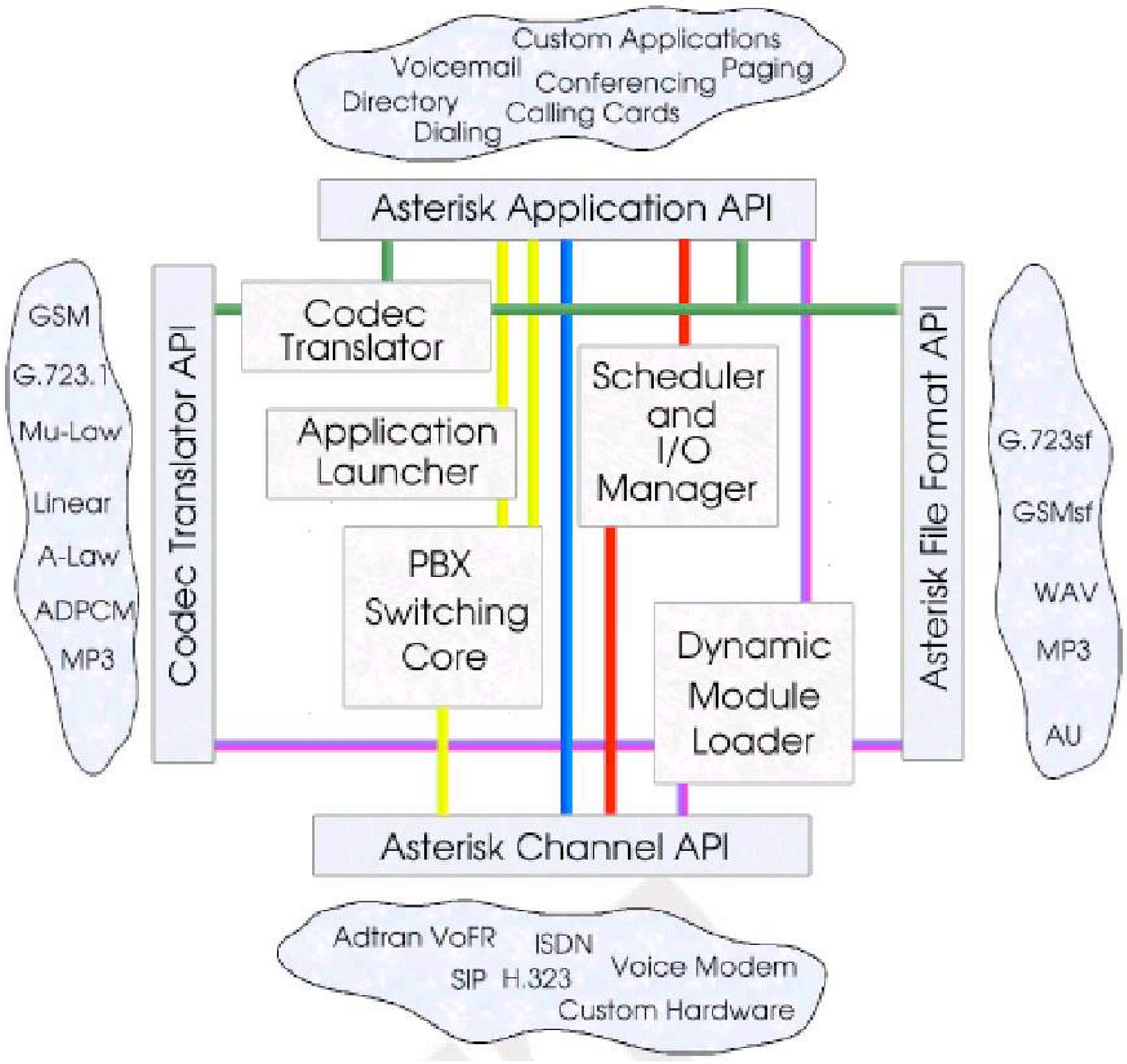
Asterisk File Locations (debian)

- /etc/asterisk/ - Asterisk configuration files
- /var/lib/asterisk/ - contains the astdb, firmware and keys
- /usr/share/asterisk/sounds - in built asterisk sound prompts
- /var/spool/asterisk/ - temporary files and voicemail files
- /var/log/asterisk/ - Asterisk log files
- /var/log/asterisk/cdr-csv/ - Asterisk call detail records

How Asterisk Works, in one slide or less!

- Asterisk is a hybrid TDM and packet voice PBX
- Interfaces any piece of telephony hardware or software to any application
- Prime components: channels and extensions.conf - the Asterisk dial plan
- Channels can be many different technologies - SIP, IAX, H323, skinny, Zaptel, and others as they are created
- extensions.conf is basically a programming language controlling the flow of calls
- Applications do the work - answer a channel, ring a channel, voicemail, etc.

Asterisk Architecture



TrixBox

- www.trixbox.org
- Asterisk PBX up and running in one hour
- The PBX formally known as Asterisk@Home
- Latest version = 2.0, based on Asterisk 1.2
- Full featured PBX system including all the regulars:
  - Voicemail, conferencing, call forwarding, extensions
- Provides web based interface, which in turn drives Asterisk configuration files
- We’ll be looking at this later in the workshop

Asterisk Configuration Details

- Text based configuration files
    - sip.conf
    - extensions.conf
    - voicemail.conf
    - agents.conf
    - queues.conf

sip.conf

```
/etc/asterisk/sip.conf

[general]
context=default                ; Default context for incoming calls
port=5060                      ; UDP Port to bind to (SIP standard port is 5060)
bindaddr=0.0.0.0               ; IP address to bind to (0.0.0.0 binds to all)
srvlookup=yes                  ; Enable DNS SRV lookups on outbound calls


[2000]
type=friend                    ; both send and receive calls from this peer
host=dynamic                   ; this peer will register with us
username=2000
secret=j3nny
canreinvite=no                 ; don't send SIP re-invites (ie. terminate rtp
stream)
nat=yes                        ; always assume peer is behind a NAT
context=phones                 ; send calls to 'phones' context
dtmfmode=rfc2833               ; set dtmf relay mode
allow=all                      ; allow all codecs
```

sip.conf ...ctd

```
[pstn-gateway]
type=friend
disallow=all
allow=alaw
context=from-pstn-gateway
host=pstn-gateway.jonnynet.net
canreinvite=no
dtmfmode=rfc2833
allow=all
```

extensions.conf

```
/etc/asterisk/extensions.conf

[general]
static=yes                ; default values for changes to this file
writeprotect=no           ; by the Asterisk CLI

[globals]
; variables go here

[default]
; default context

[from-pstn-gateway]
; context for calls coming from wlg-gateway
exten => 4989560,1,GoTo(phones,2000,1)
exten => _.,1,Congestion()                   ; everyone else gets congestion
```

extensions.conf ...ctd

```
[phones]
; context for our phones
exten => 2000,1,Dial(SIP/2000)
exten => 2000,2,Voicemail(u2000)

exten => 500,1,Answer()
exten => 500,2,Playback(demo-echotest)        ; Let them know what's going on
exten => 500,3,Echo                           ; Do the echo test
exten => 500,4,Playback(demo-echodone)        ; Let them know it's over
exten => 500,5,Hangup

exten => _1.,1,Dial(SIP/${EXTEN:1}@pstn-gateway)  ; match all and send to wlg-gateway
exten => _1.,2,Hangup
```

Dial Plan - Contexts

- extensions.conf split into sections called contexts
- [context-name]
- contexts isolated from one another - can have the same extension in multiple contexts
- Calls from a channel land in the context specified by that channel,
- Calls land in default context if nothing is specified
- Be careful with what is in the default context - it is easy to give access to more than is intended

Dial Plan - Extensions

- One or more extensions in each context
- An extension is followed by an incoming call or digits dialled on a channel
  - exten => name,priority,application()
  - exten => 2000,1,Dial(SIP/2000)
- Priorities are numbered and followed sequentially from ‘1’
  - Asterisk will stop processing an extension if you skip a priority
- Each priority executes one specific application

Dial Plan - Applications

- Applications ‘do things’ in the Asterisk dial plan
  - play a sound
  - answer a call
  - interact with a database
- Can take zero or more arguments
  - Answer()
  - Dial(SIP/2001)
  - AnApplicationWithThreeArguments(arg1,arg2,arg3)
- Arguments can be separated with a pipe (|) or a comma (,).

Dial Plan - ‘n’ priority

- Asterisk 1.2 onwards understands the ‘n’ priority

```
exten => 2000,1,FirstApplication()
exten => 2000,n,NextApplication()
exten => 2000,n(priority_label),AnotherApplication()
```

- Saves renumbering your extensions if you add or remove a priority
- Labels can make dial plan more readable, particularly when branching using gotos.

Dial Plan - Variables

- Three types of variables available in the dial plan
- Global
  - Set in the [globals] section of extensions.conf
- Channel
  - Variables set automatically, and using the set command on a per channel basis
- A number of pre-defined channel variables - e.g. ${EXTEN}

Dial Plan - Variables

- Some of the common pre-defined channel variables:

  ${CALLERID}
  ${CALLERIDNAME}
  ${CALLERIDNUM}
  ${CHANNEL}
  ${CONTEXT}
  ${EXTEN}
  ${SIPUSERAGENT}

Dial Plan - Extension Matching

- exten => _04NXXXXXX,1,SomeApplication()
- exten => _.,1,SomeApplication()
  - _ denotes a pattern matching extension
  - N matches any number from 2 through 9
  - X matches any single digit
  - . matches one or more of any digit
  - [2-6] matches any of 2,3,4,5,6

Dial Plan - Extension Matching

- Examples
  - _02[1579].
    - matches NZ mobiles, i.e. numbers starting in 021, 025, 027, or 029
  - _027NXXXXXX
    - matches numbers starting in 027 and exactly 10 digits long, where the fourth digit is from 2 - 9

Starting Asterisk

- On Debian systems:
    - /etc/init.d/asterisk start
- Or, /usr/sbin/asterisk
    - asterisk -c if you want asterisk to load straight into a console
- To connect to a running instance of Asterisk:
    - asterisk -r

Running Asterisk

```
jonny@collins:~# asterisk -h
Asterisk 1.0.7-BRIstuffed-0.2.0-RC7k, Copyright (C) 2000-2004, Digium.
Usage: asterisk [OPTIONS]
Valid Options:
   -V              Display version number and exit
   -C <configfile> Use an alternate configuration file
   -G <group>      Run as a group other than the caller
   -U <user>       Run as a user other than the caller
   -c              Provide console CLI
   -d              Enable extra debugging
   -f              Do not fork
   -g              Dump core in case of a crash
   -h              This help screen
   -i              Initialize crypto keys at startup
   -n              Disable console colorization
   -p              Run as pseudo-realtime thread
   -q              Quiet mode (suppress output)
   -r              Connect to Asterisk on this machine
   -R              Connect to Asterisk, and attempt to reconnect if disconnected
   -t              Record soundfiles in /var/tmp and move them where they belong
after they are done.
   -v              Increase verbosity (multiple v's = more verbose)
   -x <cmd>        Execute command <cmd> (only valid with -r)
```

Running Asterisk

```
jonny@collins:~# asterisk -r
Asterisk 1.0.7-BRIstuffed-0.2.0-RC7k, Copyright (C) 1999-2004 Digium.
Written by Mark Spencer <markster@digium.com>
=========================================================================
Connected to Asterisk 1.0.7-BRIstuffed-0.2.0-RC7k currently running on collins (pid
= 10763)
collins*CLI>
```

Asterisk CLI

- Similar to IOS:
  - sip show peers
  - reload
  - ? for help, tab for command autocomplete
  - sip show ?
- Restart commands
  - restart gracefully: Restart Asterisk gracefully
  - restart now: Restart Asterisk immediately
  - restart when convenient: Restart Asterisk at empty call volume
  - reload: Reload configuration
- stop gracefully: Gracefully shut down Asterisk
- stop now: Shut down Asterisk imediately
- stop when convenient: Shut down Asterisk at empty call volume

Asterisk CLI

- sip debug: Enable SIP debugging
- sip no debug: Disable SIP debugging
- sip reload: Reload sip.conf

- SIP Show commands
  - sip show channels: Show active SIP channels
  - sip show channel: Show detailed SIP channel info
  - sip show inuse: List all inuse/limit
  - sip show peers: Show defined SIP peers (clients that register to your Asterisk server)
  - sip show registry: Show SIP registration status (when Asterisk registers as a client to a SIP Proxy)
  - sip show users: Show defined SIP users

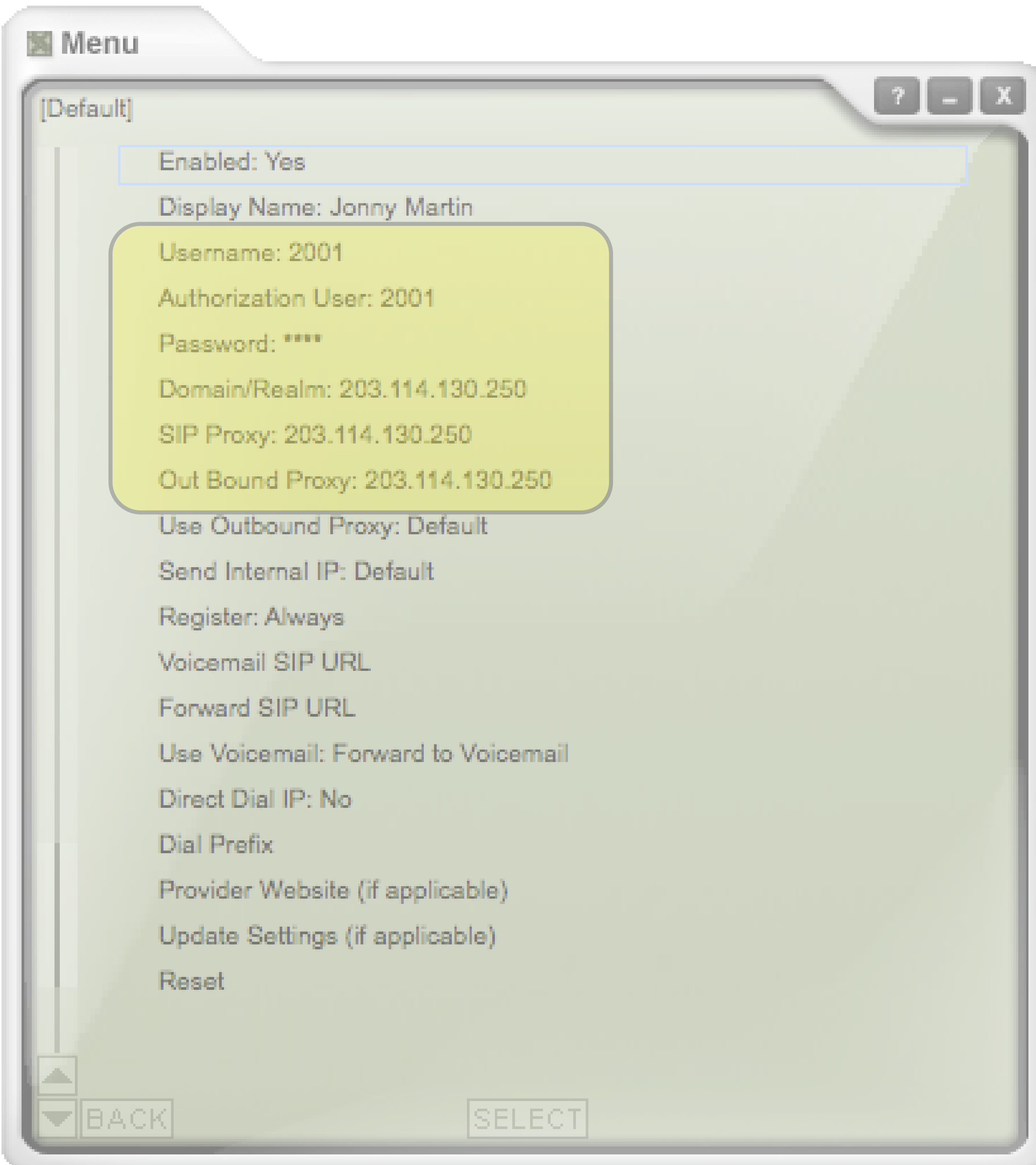
Soft Phone Client

- Any SIP client can be used for the lab
- We’ll use the Xten Xlite client
  - Works on Win, Mac, Linux
  - http://www.xten.com/index.php?menu=download
- You can use a Wifi phone or similar if you have one with you

Xlite Softphone Setup

- Only need to set a few basic paramaters
  - SIP username
  - SIP password
- This is done in
  - Main Menu > System Settings > SIP Proxy > Default

Xlite Softphone Setup



Lab 1: Initial Asterisk Install

Asterisk Variables

- Why use variables?
  - Pattern match - how do we know what extensions was dialled?

    ```
    exten => 2000,1,Dial(SIP/2000)
    exten => 2001,1,Dial(SIP/2001)
    exten => 2002,1,Dial(SIP/2002)
    ```

  - OR

    ```
    exten => _200X,1,Dial(SIP/${EXTEN})
    ```

  - ${some_variable} = the value of some_variable.
  - some_variable = the variable itself

Asterisk Variables

- Set default variables

  ```
  [globals]
  default_ring_time=10

  [context]
  exten => 2000,1,Dial(SIP/2000,${default_ring_time})
  ```

- Now only one place in dial plan to update if it is changed

- Setting variables:

  - exten => s,1,Set(a_variable=2000)

Asterisk Variables

- Complete list of Asterisk variables

The ‘s’ Start Extension

- The standard extension a call starts in without needed to specifically match an extension

- Often used with FXS/FXO cards due to lack of end to end signalling with analogue channels

```
[incoming]
exten => s,1,Answer()
exten => s,2,Background(enter-ext-of-person)

exten => 1,1,Playback(digits/1)
exten => 1,2,Goto(incoming,s,1)

exten => 2,1,Playback(digits/2)
exten => 2,2,Goto(incoming,s,1)

exten => 3,1,Hangup
```

The Standard Extensions

- i : Invalid
- s : Start
- h : Hangup
- t : Timeout
- T : AbsoluteTimeout
- o : Operator

Dial Command

- Dial(*tech/username:password@hostname/extension,ring-timeout,flag*)
- Can include complete information in the dial string, or reference a peer in sip.conf
  - exten => 2000,1,Dial(SIP/passwd:sipdevice@host.tld)
    - or
  - exten => 2000,1,Dial(SIP/sipdevice)

    where there is a channel [sipdevice] defined in sip.conf containing at least definitions for username, password and host.

Voicemail

- Comedian Mail - a fully functional voicemail system included with Asterisk
  - Supports busy and unavailable messages
  - exten => 2001,1,Voicemail(b2001)
  - exten => 2001,1,Voicemail(u2001)
- Voicemail can be emailed out a .wav attachment to users
- Standard IVR voicemail access
  - exten => 510,1,VoicemailMain

voicemail.conf

```
[general]
format=wav49|gsm|wav
serveremail=voicemail@jonnynet.net
mailcmd=/usr/sbin/sendmail -t
attach=yes
maxmsg=100
maxmessage=180
skipms=3000
maxsilence=10
silencethreshold=128
maxlogins=3

emailbody=Dear ${VM_NAME}:\n\n\tjust wanted to let you know you were just left a
 ${VM_DUR} long message (number ${VM_MSGNUM})\nin mailbox ${VM_MAILBOX} from ${V
M_CALLERID}, on ${VM_DATE}, so you might\nwant to check it when you get a chance
.  Thanks!\n\n\t\t\t\t--Asterisk\n
emaildateformat=%A, %B %d, %Y at %r

[default]
; all our mailboxes here
; mailbox number => pin,name,email
2000 => 1234,Jonny,jonny@jonnynet.net
```

Music on Hold

- Music on hold (MOH) played automatically when a channel is is placed on hold
  - Multiple classes of MOH defined

    ```
    exten => 100,1,Answer()
    exten => 100,2,MusicOnHold(default)  ; class = default, could be any other
    ```

- Default file directory, Debian:
  - /usr/share/asterisk/mohmp3
- RedHat, or if compiling from source
  - /var/lib/asterisk/mohmp3

musiconhold.conf

```
[default]
;mode=quietmp3
mode=files
directory=/var/lib/asterisk/mohmp3

; valid mode options:
; quietmp3      -- default
; mp3           -- loud
; mp3nb         -- unbuffered
; quietmp3nb    -- quiet unbuffered
; custom        -- run a custom application
; files         -- read files from a directory in any Asterisk supported format
```

MeetMe Conferencing

- Powerful application built in to Asterisk
- Some use Asterisk purely for it's conferencing abilities
- Ad Hoc MeetMe conferencing, or individual conference rooms with PIN

```
/etc/asterisk/meetme.conf
; Configuration file for MeetMe simple conference rooms
;
[rooms]
; Usage is conf => confno[,pin]
;
conf => 101,1234
conf => 102,2345

/etc/asterisk/extensions.conf
exten => 5101,1,Meetme(101|M)
exten => 5102,2,Meetme(102|M)
```

Interactive Voice Response

- Interactive Voice Response (IVR) is inherent to the Asterisk dialplan
- Simply a matter of playing prompts, waiting, accepting input in a channel, and moving around the dial plan
  - Useful applications:
    - Background(prompt-to-play-whilst-waiting-for-intput)
    - Playback(prompt-to-play-whilst-NOT-accepting-input)
    - Goto(context,extension,priority)
    - Dial(SIP/2000)
    - Wait(seconds)

Sample IVR

```
[test-ivr]
exten => s,1,Answer()
exten => s,2,Background(enter-ext-of-person)

exten => 1,1,Playback(digits/1)
exten => 1,2,Goto(incoming,s,1)

exten => 2,1,Playback(digits/2)
exten => 2,2,Goto(incoming,s,1)

exten => i,1,Playback(pbx-invalid)
exten => i,2,Goto(incoming,s,1)

exten => t,1,Playback(vm-goodbye)
exten => t,2,Hangup()

[phones]
; allow our phones to dial into the IVR
exten => 2010,1,Goto(test-ivr,s,1)
```

Lab 2: Basic Asterisk Configuration