Getting Started With Asterisk

Version 0.1a

Written by
Andy Powell

If you would like to utilise my skills on a professional basis please visit Red Dot Systems

Feeling generous? Want to help me buy bit's n bobs to connect to Asterisk and get working? Why not

Make a Donation

Want to just have a quick play with asterisk? Try the Asterisk Live! CD which I've put together. Add a USB key or USB CF card reader and you can reload your configs automatically at boot.. NEW: CF card version What about a PXE boot version?

You can also take a look at the extended guide (work in progress) after you've done your install

Contents

Introduction

Who is your daddy and what does he do?

Oh my! A hugely funny reference to start off… well ok perhaps not, but anyway a little background on me and reasons for writing the guide. I used to work for a Swiss bank here in Amsterdam, up until December 2002 that is, when I was made redundant. Having searched for a job everywhere and finding nothing – more accurately getting no interviews - I thought I'd get myself some new skills. Asterisk caught my eye as I'd been messing about with Free World Dialup, and had even splashed out on a couple of IP phones. As you probably know by now, finding some documentation was, well, a little difficult to say the least. The official guide, although it makes sense to me now, was just a collection of gibberish. In the end I decided to do something about it, not only to stop me going insane but to try and help others on those 'first steps' to setting up and running what must be **the** Linux PBX software.

Anyway, enough about me, you didn't even read that bit anyway did you…

What is Asterisk?

Asterisk PBX, from now on just called Asterisk, is Linux based, Open Source and free PBX software. Or to quote from the Asterisk website http://www.asterisk.org (with corrected spelling ;))

Asterisk is a complete PBX in software. It runs on Linux and provides all of the features you would expect from a PBX and more. Asterisk does voice over IP in three protocols, and can interoperate with almost all standards-based telephony equipment using comparatively inexpensive hardware.

Asterisk provides Voicemail services with Directory, Call Conferencing, Interactive Voice Response, Call Queuing. It has support for three-way calling, caller ID services, ADSI, SIP and H.323 (as both client and gateway). Check the Features section for a more complete list.

Asterisk needs no additional hardware for Voice over IP. For interconnection with digital and analog telephony equipment, Asterisk supports a number of hardware devices, most notably all of the hardware manufactured by Asterisk's sponsors, Digium. Digium has single and quad span T1 and E1 interfaces for interconnection to PRI lines and channel banks. In addition, an analog FXO card is available, and more analog interfaces are in the works.

Also supported are the Internet Line Jack and Internet Phone Jack products from Quicknet.

Asterisk supports a wide range of TDM protocols for the handling and transmission of voice over traditional telephony interfaces. Asterisk supports US and European standard signaling types used in standard business phone systems, allowing it to bridge between next generation voice-data integrated networks and existing infrastructure. Asterisk not only supports traditional phone equipment, it enhances them with additional capabilities.

Using the IAX Voice over IP protocol, Asterisk merges voice and data traffic seamlessly across disparate networks. While using Packet Voice, it is possible to send data such as URL information and images in-line with voice traffic, allowing advanced integration of information.

Asterisk provides a central switching core, with four APIs for modular loading of telephony applications, hardware interfaces, file format handling, and codecs. It allows for transparent switching between all supported interfaces, allowing it to tie together a diverse mixture of telephony systems into a single switching network.

Asterisk is primarily developed on GNU/Linux for x/86. It is known to compile and run on GNU/Linux for PPC. Other platforms and standards based UNIX-like operating systems should be reasonably easy to port for anyone with the time and requisite skill to do so. Asterisk is available in the testing and unstable debian archives, maintained thanks to Mark Purcell.

Who Made Asterisk?

Asterisk was originally written by Mark Spencer of Digium dba Linux Support Services, Inc. Code has been contributed from Open Source coders around the world, and testing and bug-patches from the community have provided invaluable aid to the development of this software.

Document Conventions

Just to clear up any confusion that may occur here are the conventions used in this guide.

Text that appears in grey boxes like this contain the text that should be typed in, or output depending on the context; however you should omit the leading # character as this simply represents the shell prompt.

```
# ls /etc/asterisk
```

Installing Linux for Asterisk

In order to use Asterisk you are going to have to use Linux. If you are a Linux ‘guru’ then you might want to skip this section and just take a look at the dependencies. This section deals with installing a Linux system for use with Asterisk. I make no excuses or arguments for the chosen distribution, Red Hat 8; I have no intention of discussing the pros and cons of the many different distributions available. In other words, if you don’t like Red Hat, stop bitching and find a windows user to torment.

Getting the ISO images

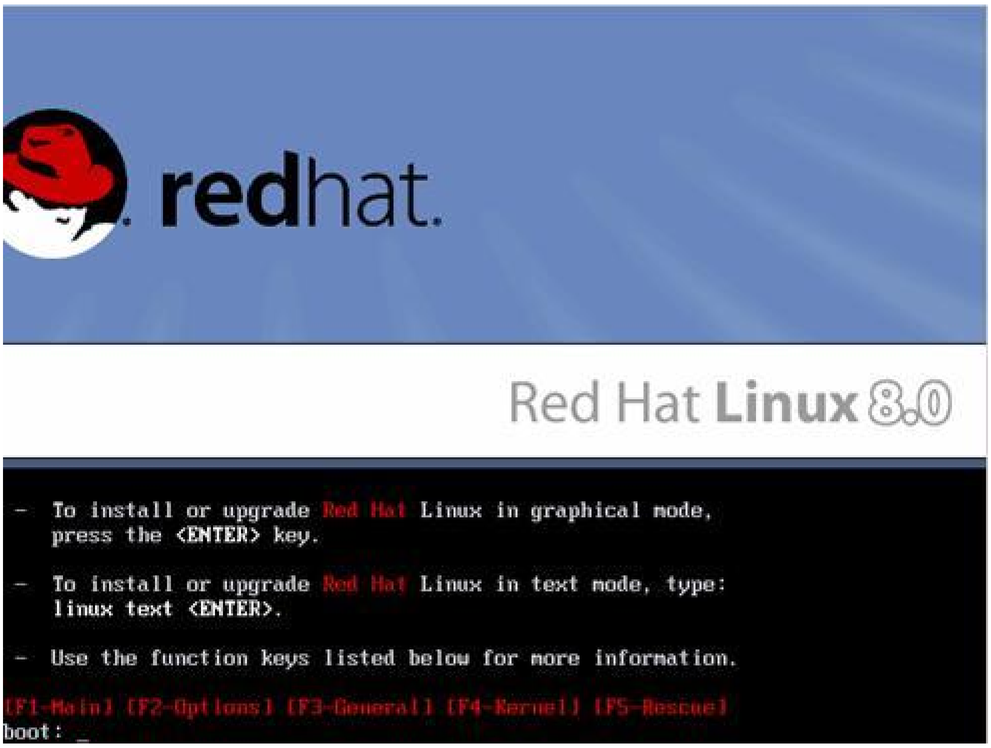
I’m not going to tell you how to create the CD’s required to install Linux, there are other sites for this. You can find the ISO images required from places like http://www.linuxiso.org. Download the ISO images and burn them. For This installation you only actually need disks 1, 2 and 3 (the assumption is you are using the default language settings)– If you are pushed for a network connection you can get away with only downloading and burning those, however, I would recommend getting the whole set for the sake of completeness.

Installing Linux

The first step is to insert CD 1 of your freshly burned ISO images into the target machine’s CDROM drive and boot. At this stage I’m going to make an assumption that you are either installing Asterisk on a new machine or overwriting the disks of whatever was there before. If you want to dual boot into Asterisk, please look elsewhere for initial configurations. You might want to consult your physician too. I make no apologies that this section is going to treat you like a moron; this is intentional. Since we want to be sure that you have no problems running Asterisk we need to make sure that your installation matches the one I used to build and run it.

The first thing you will need to do on your target machine is ensure that you can boot directly from CD. Most modern (if not all) will allow this, it is usually just a setting in your BIOS. If you don’t feel comfortable with messing in the BIOS the simplest test you can do to check and see if booting from CD is enabled is, funnily enough, to put CD 1 into the drive and reboot the machine. If the machine starts to boot and eventually shows a screen like the one on the following page, then you are ok. If you are happy fiddling with the BIOS then just enable booting from CD and make it the first device to attempt to boot from.

In either case you should have something like this appear on your screen:



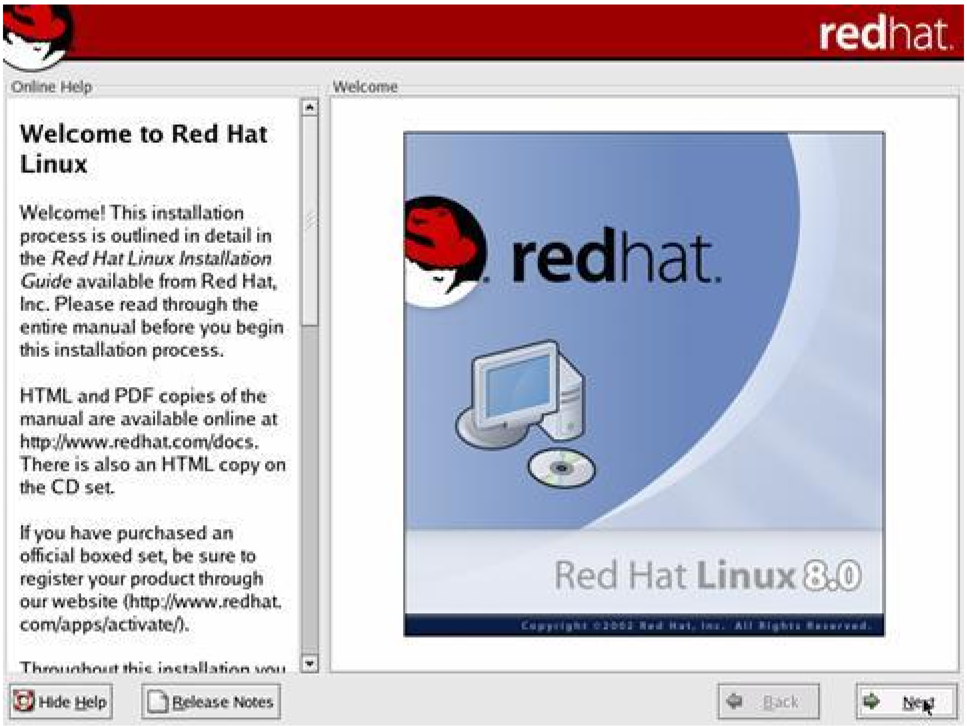
```
-  To install or upgrade Red Hat Linux in graphical mode,
   press the <ENTER> key.

-  To install or upgrade Red Hat Linux in text mode, type:
   linux text <ENTER>.

-  Use the function keys listed below for more information.

[F1-Main] [F2-Options] [F3-General] [F4-Kernel] [F5-Rescue]
boot: _
```

At this point all you need to do is press **ENTER**. After a short while, you will be presented with the following introductory screen:



Welcome to Red Hat Linux

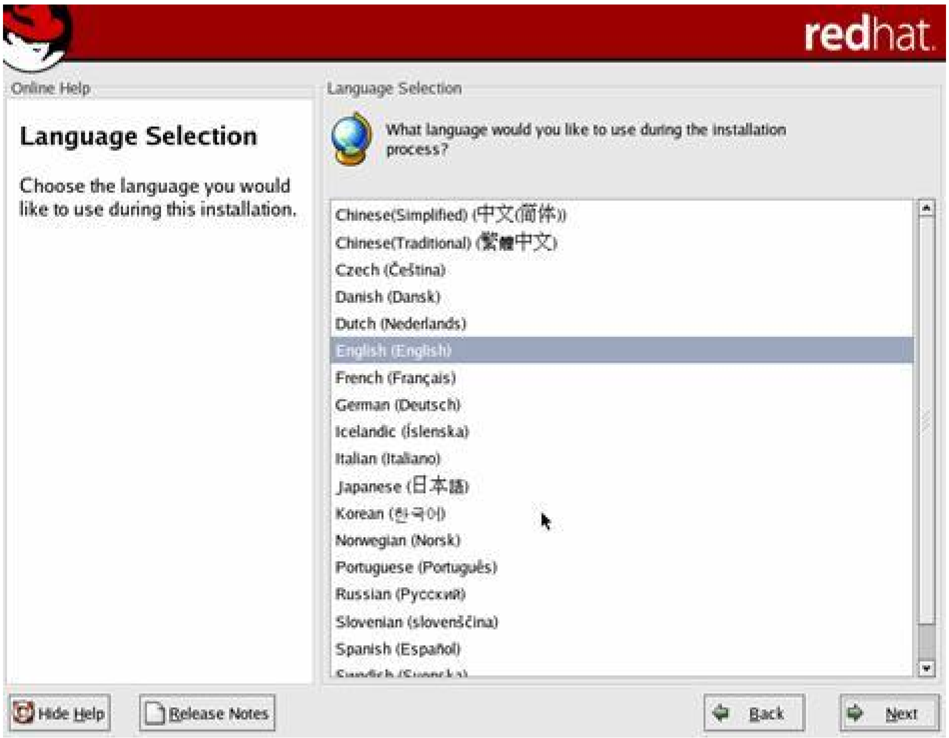
Welcome! This installation process is outlined in detail in the *Red Hat Linux Installation Guide* available from Red Hat, Inc. Please read through the entire manual before you begin this installation process.

HTML and PDF copies of the manual are available online at http://www.redhat.com/docs. There is also an HTML copy on the CD set.

If you have purchased an official boxed set, be sure to register your product through our website (http://www.redhat.com/apps/activate/).

Throughout this installation you

Click the **NEXT** button



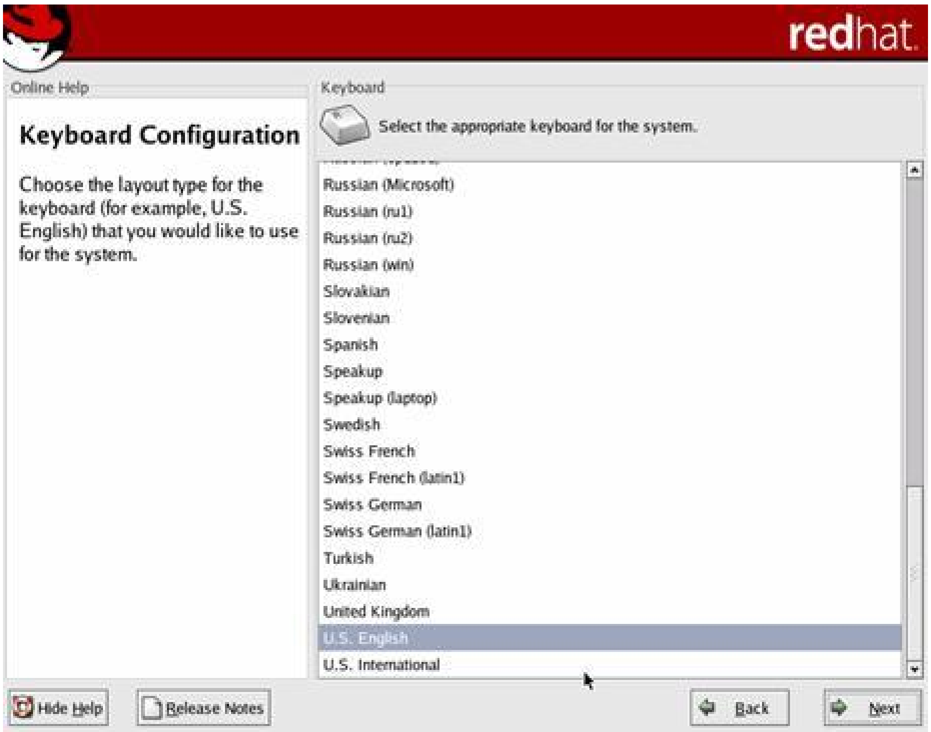
Language Selection

Choose the language you would like to use during this installation.

What language would you like to use during the installation process?

Chinese(Simplified) (中文(简体))
Chinese(Traditional) (繁體中文)
Czech (Čeština)
Danish (Dansk)
Dutch (Nederlands)
English (English)
French (Français)
German (Deutsch)
Icelandic (Íslenska)
Italian (Italiano)
Japanese (日本語)
Korean (한국어)
Norwegian (Norsk)
Portuguese (Português)
Russian (Русский)
Slovenian (slovenščina)
Spanish (Español)
Swedish (Svenska)

Select the language you want to use during the installation. Since I used the default on English, all the following screens will also be in English. Then Click **NEXT**.



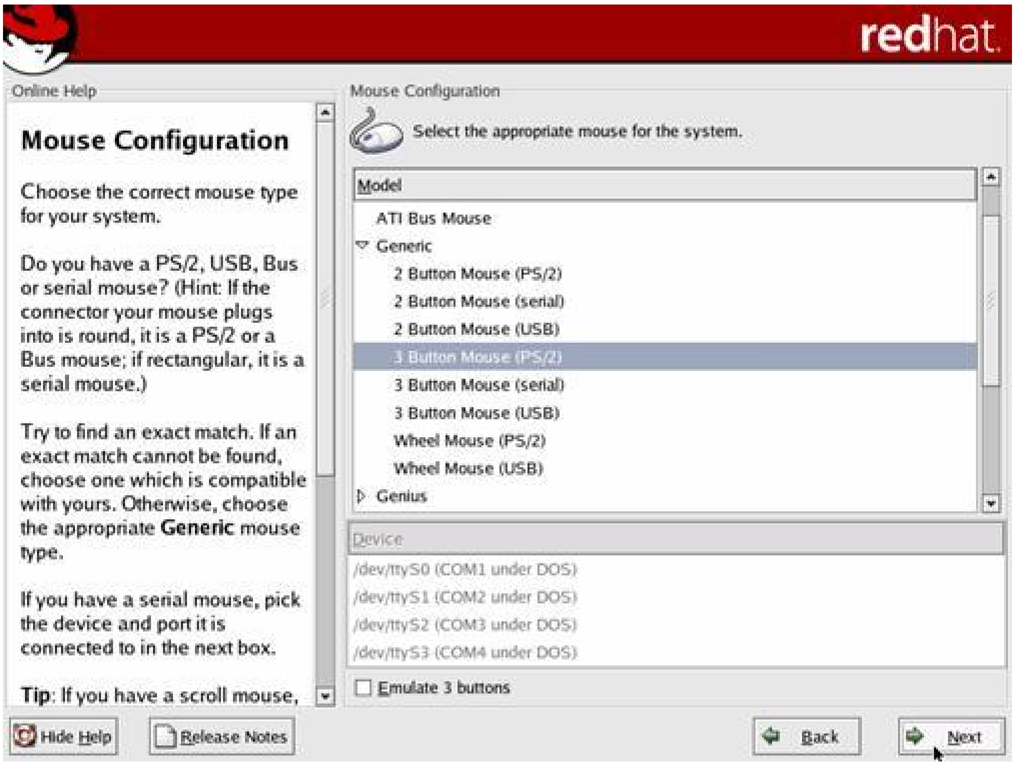
Keyboard Configuration

Choose the layout type for the keyboard (for example, U.S. English) that you would like to use for the system.

Select the appropriate keyboard for the system.

Russian (Microsoft)
Russian (ru1)
Russian (ru2)
Russian (win)
Slovakian
Slovenian
Spanish
Speakup
Speakup (laptop)
Swedish
Swiss French
Swiss French (latin1)
Swiss German
Swiss German (latin1)
Turkish
Ukrainian
United Kingdom
U.S. English
U.S. International

Select the keyboard layout for your particular PC. Most keyboards in the English speaking world tend to be U.S. English but check yours.



Mouse Configuration

Choose the correct mouse type for your system.

Do you have a PS/2, USB, Bus or serial mouse? (Hint: If the connector your mouse plugs into is round, it is a PS/2 or a Bus mouse; if rectangular, it is a serial mouse.)

Try to find an exact match. If an exact match cannot be found, choose one which is compatible with yours. Otherwise, choose the appropriate **Generic** mouse type.

If you have a serial mouse, pick the device and port it is connected to in the next box.

Tip: If you have a scroll mouse,

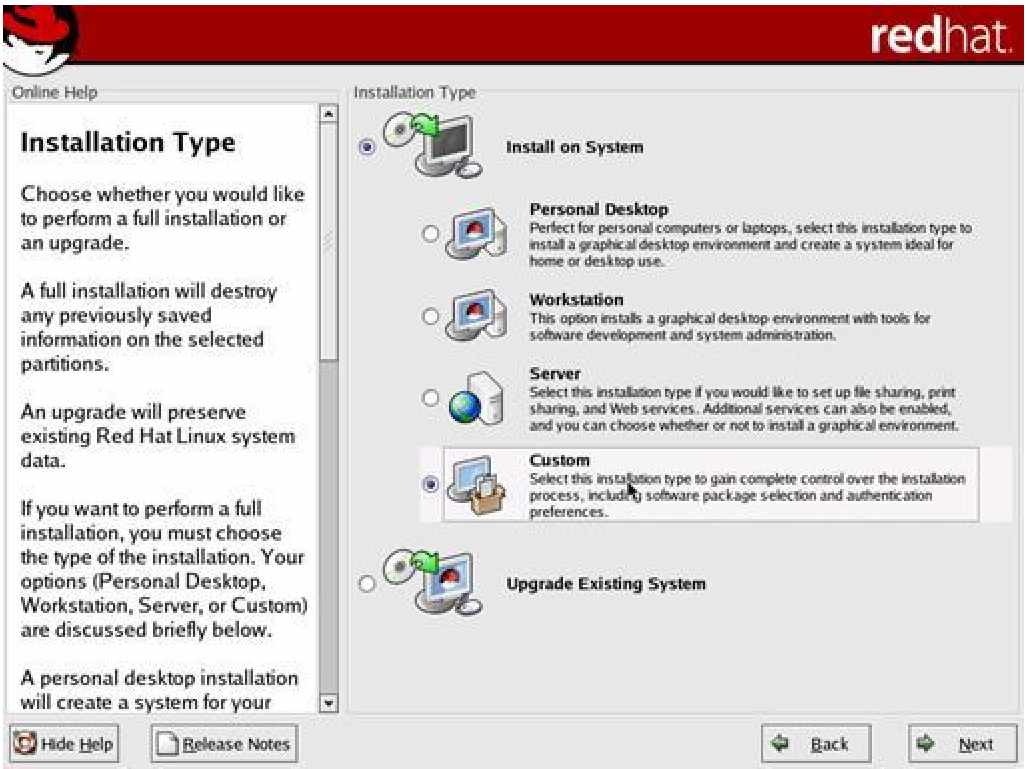
Select the appropriate mouse for the system.

Model
- ATI Bus Mouse
- Generic
  - 2 Button Mouse (PS/2)
  - 2 Button Mouse (serial)
  - 2 Button Mouse (USB)
  - 3 Button Mouse (PS/2)
  - 3 Button Mouse (serial)
  - 3 Button Mouse (USB)
  - Wheel Mouse (PS/2)
  - Wheel Mouse (USB)
- Genius

Device
- /dev/ttyS0 (COM1 under DOS)
- /dev/ttyS1 (COM2 under DOS)
- /dev/ttyS2 (COM3 under DOS)
- /dev/ttyS3 (COM4 under DOS)

□ Emulate 3 buttons

Since we are not going to be installing a windowing system on this machine, in all honesty we don't care about the mouse. Click **NEXT**.



Installation Type

Choose whether you would like to perform a full installation or an upgrade.

A full installation will destroy any previously saved information on the selected partitions.

An upgrade will preserve existing Red Hat Linux system data.

If you want to perform a full installation, you must choose the type of the installation. Your options (Personal Desktop, Workstation, Server, or Custom) are discussed briefly below.

A personal desktop installation will create a system for your

● Install on System

○ **Personal Desktop**
Perfect for personal computers or laptops, select this installation type to install a graphical desktop environment and create a system ideal for home or desktop use.

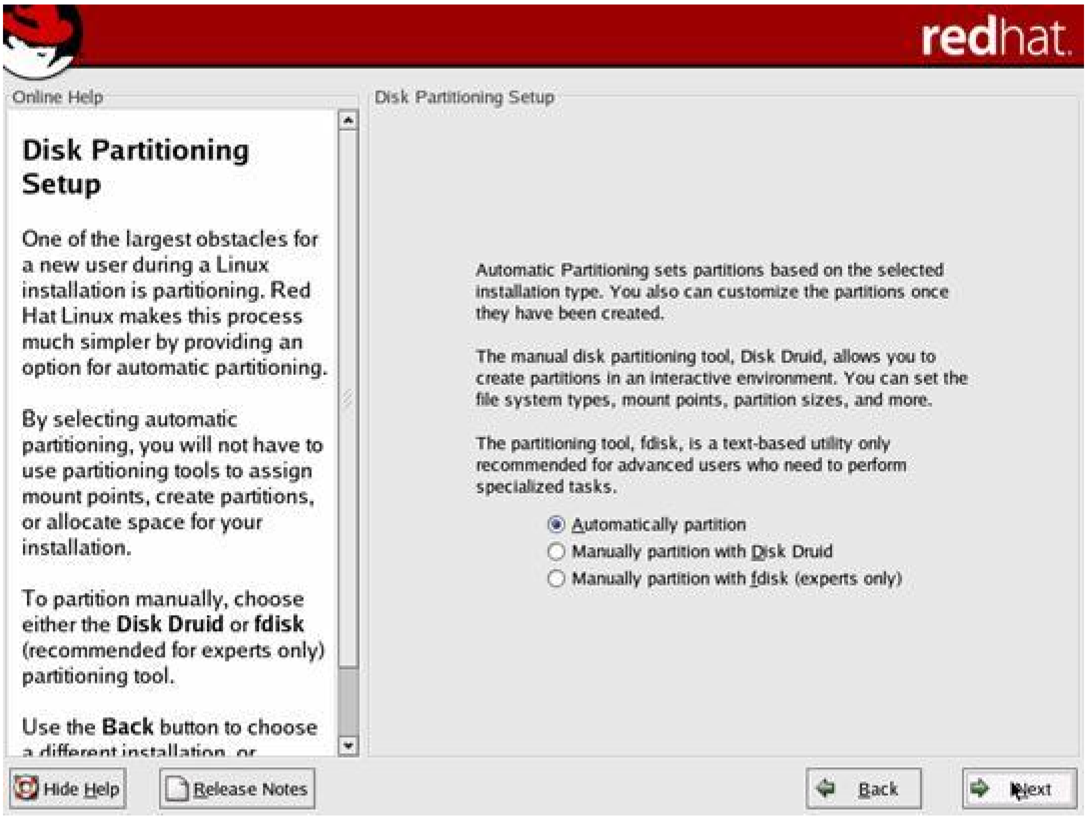
○ **Workstation**
This option installs a graphical desktop environment with tools for software development and system administration.

○ **Server**
Select this installation type if you would like to set up file sharing, print sharing, and Web services. Additional services can also be enabled, and you can choose whether or not to install a graphical environment.

● **Custom**
Select this installation type to gain complete control over the installation process, including software package selection and authentication preferences.

○ Upgrade Existing System

Since we are going to build a custom installation select **CUSTOM** and click **NEXT**.



Disk Partitioning Setup

One of the largest obstacles for a new user during a Linux installation is partitioning. Red Hat Linux makes this process much simpler by providing an option for automatic partitioning.

By selecting automatic partitioning, you will not have to use partitioning tools to assign mount points, create partitions, or allocate space for your installation.

To partition manually, choose either the **Disk Druid** or **fdisk** (recommended for experts only) partitioning tool.

Use the **Back** button to choose a different installation, or

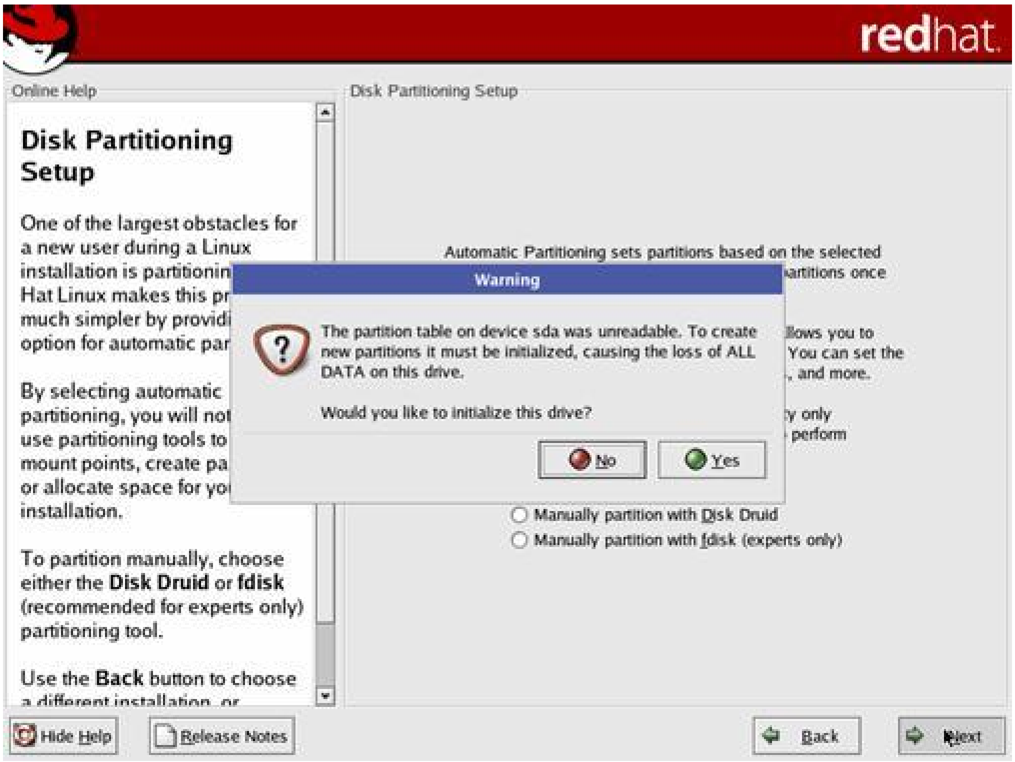
Automatic Partitioning sets partitions based on the selected installation type. You also can customize the partitions once they have been created.

The manual disk partitioning tool, Disk Druid, allows you to create partitions in an interactive environment. You can set the file system types, mount points, partition sizes, and more.

The partitioning tool, fdisk, is a text-based utility only recommended for advanced users who need to perform specialized tasks.

● Automatically partition
○ Manually partition with Disk Druid
○ Manually partition with fdisk (experts only)

This bit might seem a little scary, there's nothing to worry about, since we have already established that we are NOT going to dual boot and there is nothing that we want to keep on the disks in this machine. You did check that right?! Click **NEXT**.

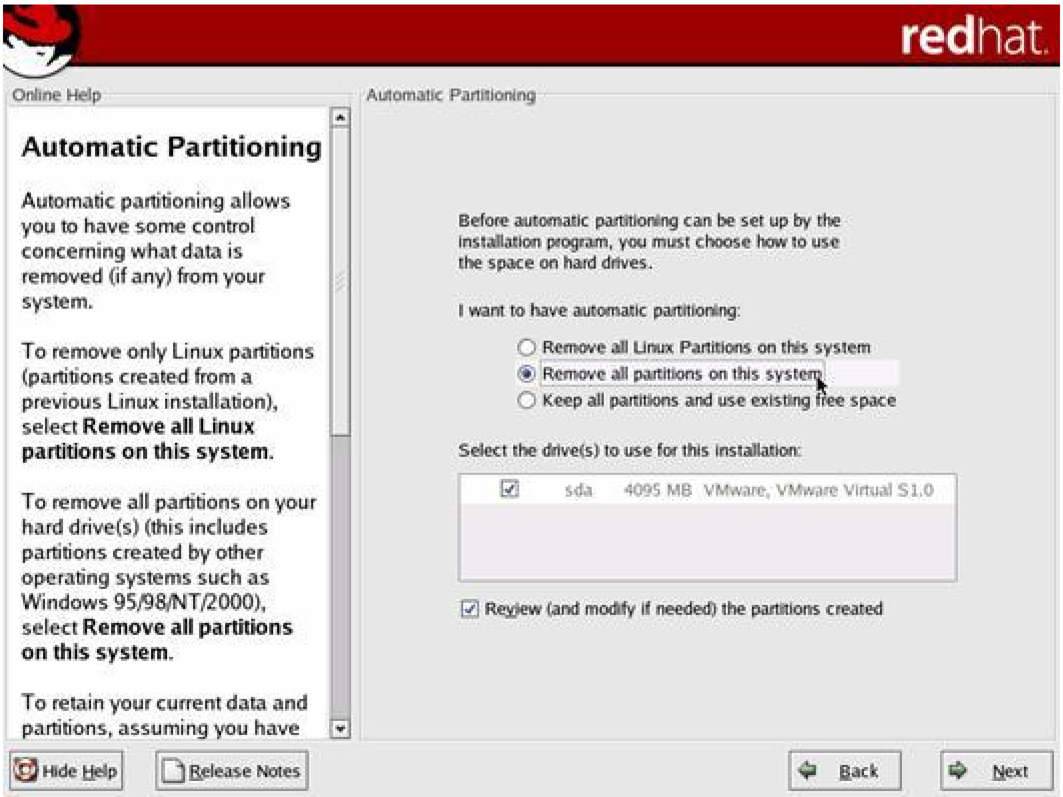


The partition table on device sda was unreadable. To create new partitions it must be initialized, causing the loss of ALL DATA on this drive.

Would you like to initialize this drive?

○ Manually partition with Disk Druid
○ Manually partition with fdisk (experts only)

If you get the previous screen, don't worry too much just click **YES**. You many note that the message box refers to 'hda' not 'sda', don't worry about this either, it's just the type of disks you are using. If you have IDE disks then this would say 'hda'.



Automatic Partitioning

Automatic partitioning allows you to have some control concerning what data is removed (if any) from your system.

To remove only Linux partitions (partitions created from a previous Linux installation), select **Remove all Linux partitions on this system**.

To remove all partitions on your hard drive(s) (this includes partitions created by other operating systems such as Windows 95/98/NT/2000), select **Remove all partitions on this system**.

To retain your current data and partitions, assuming you have

Before automatic partitioning can be set up by the installation program, you must choose how to use the space on hard drives.

I want to have automatic partitioning:

○ Remove all Linux Partitions on this system
● Remove all partitions on this system
○ Keep all partitions and use existing free space

Select the drive(s) to use for this installation:

☑ sda 4095 MB VMware, VMware Virtual S1.0

☑ Review (and modify if needed) the partitions created

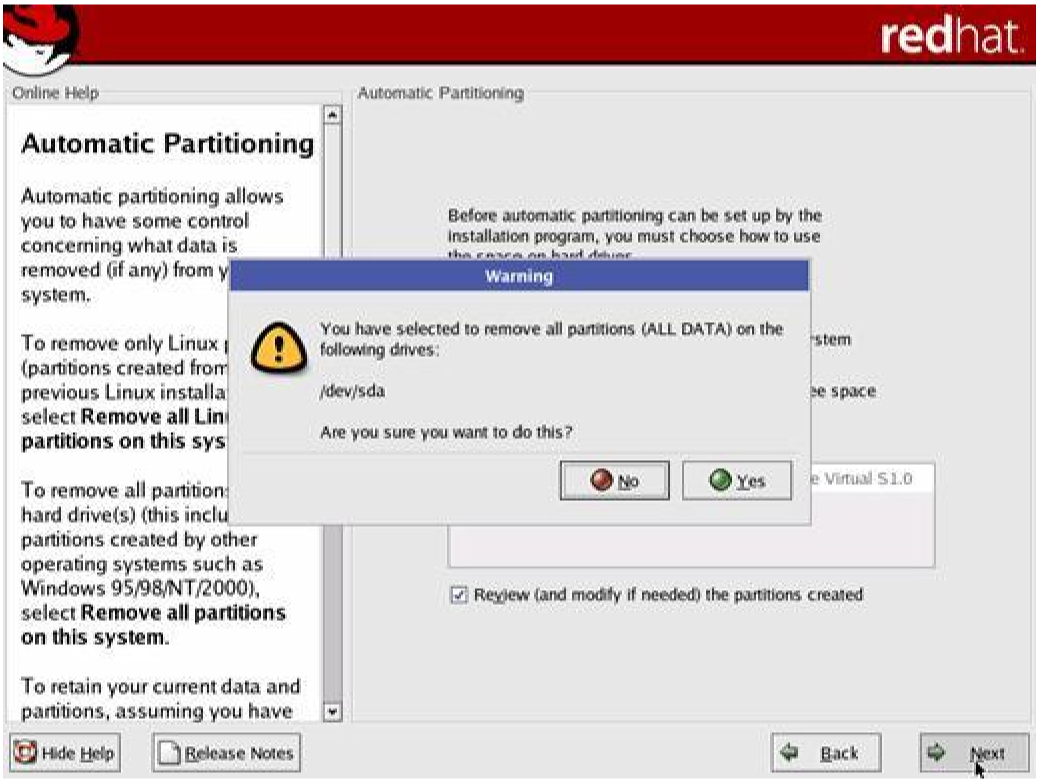
Depending on the state of this machine and the disks before you started this install you can either select 'Remove all Linux Partitions on this System' or 'Remove all Partitions on this system'

If the disks have been previously used for another Operating system then select the 'Remove all partitions on this system' option. If your disks are brand new and have not been used before then either of the first two options is fine.

The box below these options will list the hard drives available on your system, leave the selections alone.

Click the **NEXT** button. If you chose to remove all partitions on the system you will see the message box on the following page. Again the disk name /dev/sda may and most likely would be /dev/hda…

Click **YES**…



Automatic Partitioning

Automatic partitioning allows you to have some control concerning what data is removed (if any) from y system.

To remove only Linux p (partitions created from previous Linux installa select **Remove all Lin partitions on this sys**

To remove all partition hard drive(s) (this inclu partitions created by other operating systems such as Windows 95/98/NT/2000), select **Remove all partitions on this system.**

To retain your current data and partitions, assuming you have

Before automatic partitioning can be set up by the installation program, you must choose how to use the space on hard drives.

Warning

You have selected to remove all partitions (ALL DATA) on the following drives:

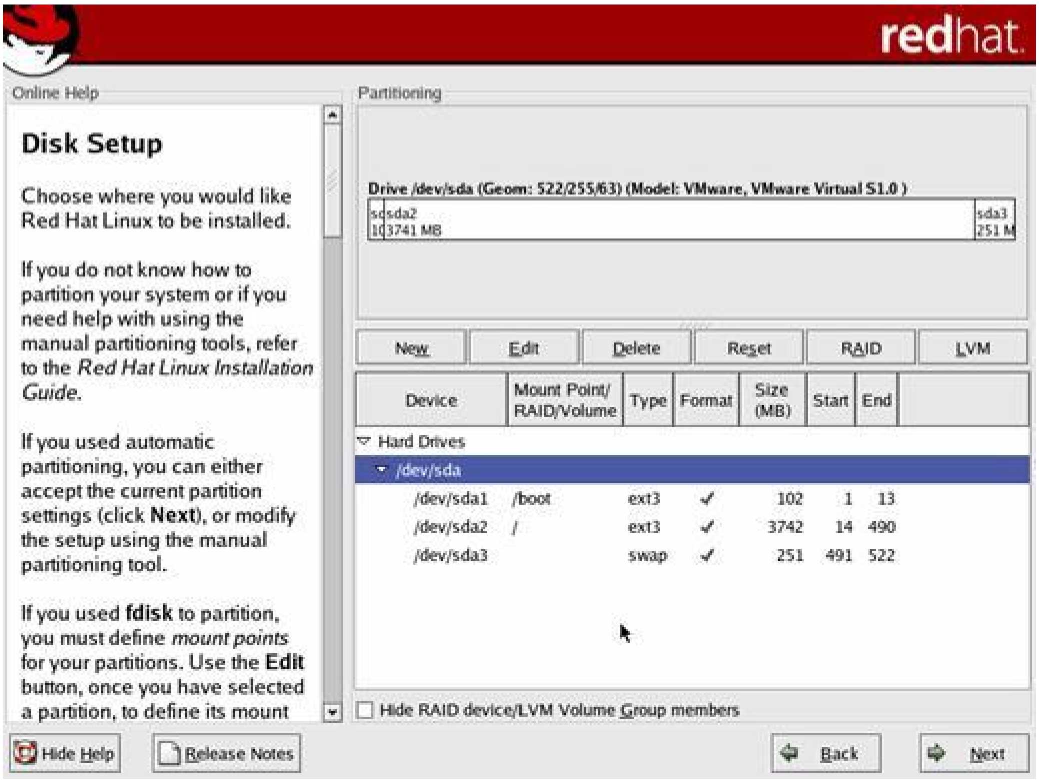
/dev/sda

Are you sure you want to do this?

No Yes

☑ Review (and modify if needed) the partitions created

You should see the following screen showing how the disks will be laid out, click the **NEXT** button.



Disk Setup

Choose where you would like Red Hat Linux to be installed.

If you do not know how to partition your system or if you need help with using the manual partitioning tools, refer to the *Red Hat Linux Installation Guide.*

If you used automatic partitioning, you can either accept the current partition settings (click **Next**), or modify the setup using the manual partitioning tool.

If you used **fdisk** to partition, you must define *mount points* for your partitions. Use the **Edit** button, once you have selected a partition, to define its mount

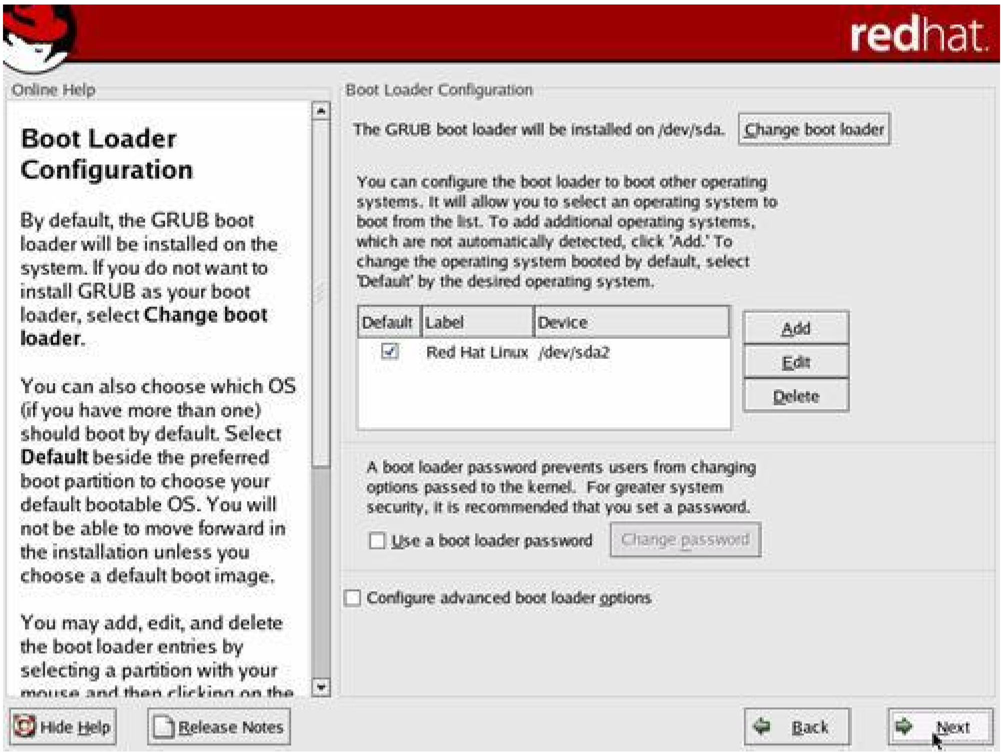
Drive /dev/sda (Geom: 522/255/63) (Model: VMware, VMware Virtual S1.0)

sda2 3741 MB | sda3 251 M

Device	Mount Point/ RAID/Volume	Type	Format	Size (MB)	Start	End
▽ Hard Drives						
▽ /dev/sda						
/dev/sda1	/boot	ext3	✓	102	1	13
/dev/sda2	/	ext3	✓	3742	14	490
/dev/sda3		swap	✓	251	491	522

□ Hide RAID device/LVM Volume Group members

Click **NEXT**



Boot Loader Configuration

By default, the GRUB boot loader will be installed on the system. If you do not want to install GRUB as your boot loader, select **Change boot loader**.

You can also choose which OS (if you have more than one) should boot by default. Select **Default** beside the preferred boot partition to choose your default bootable OS. You will not be able to move forward in the installation unless you choose a default boot image.

You may add, edit, and delete the boot loader entries by selecting a partition with your mouse and then clicking on the

The GRUB boot loader will be installed on /dev/sda.

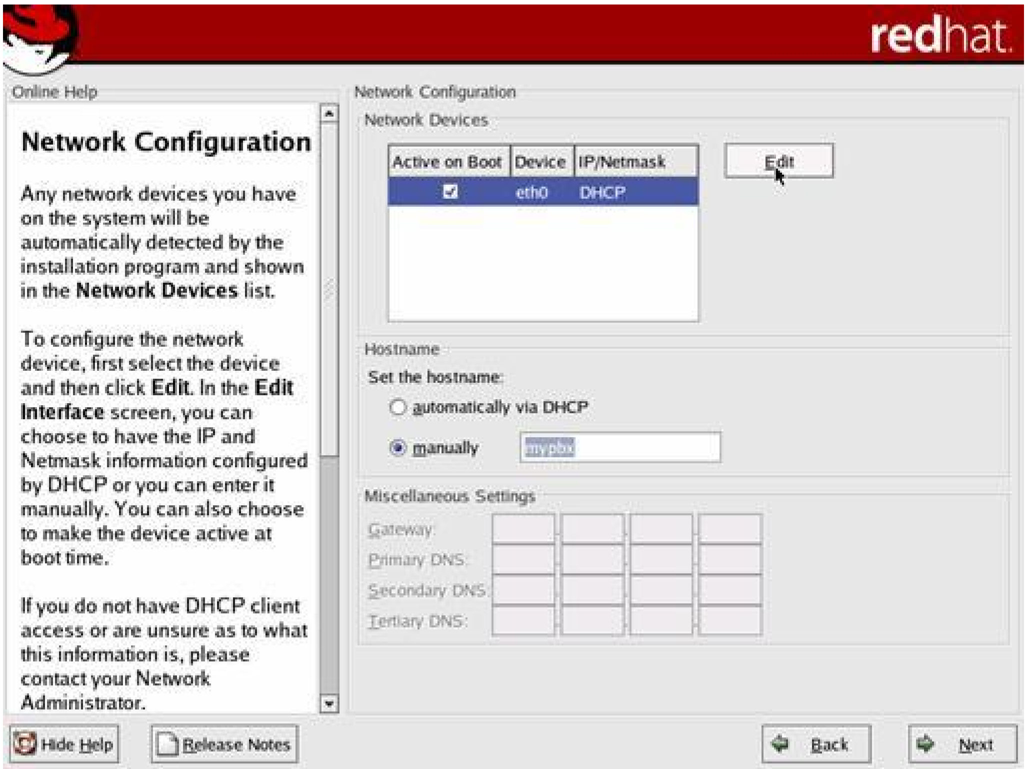
You can configure the boot loader to boot other operating systems. It will allow you to select an operating system to boot from the list. To add additional operating systems, which are not automatically detected, click 'Add.' To change the operating system booted by default, select 'Default' by the desired operating system.

Default	Label	Device
☑	Red Hat Linux	/dev/sda2

A boot loader password prevents users from changing options passed to the kernel. For greater system security, it is recommended that you set a password.

□ Use a boot loader password

□ Configure advanced boot loader options



Network Configuration

Any network devices you have on the system will be automatically detected by the installation program and shown in the **Network Devices** list.

To configure the network device, first select the device and then click **Edit**. In the **Edit Interface** screen, you can choose to have the IP and Netmask information configured by DHCP or you can enter it manually. You can also choose to make the device active at boot time.

If you do not have DHCP client access or are unsure as to what this information is, please contact your Network Administrator.

Active on Boot	Device	IP/Netmask
☑	eth0	DHCP

Set the hostname:

○ automatically via DHCP

● manually mypbx

Gateway:

Primary DNS:

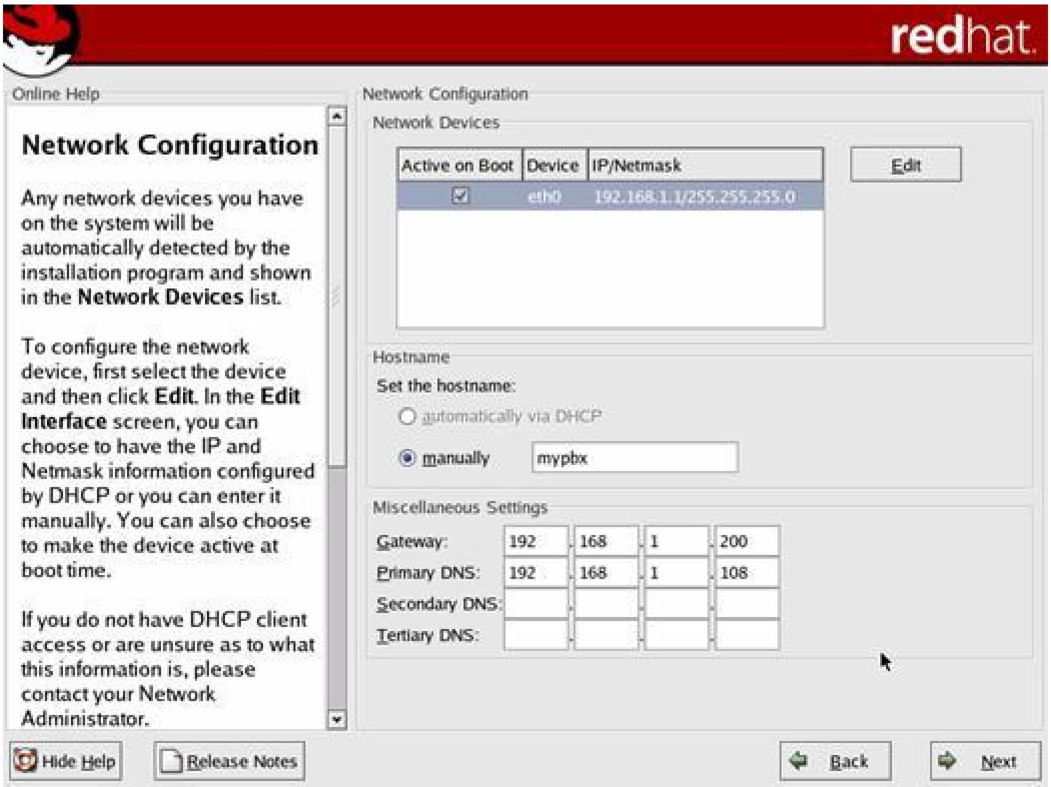
Secondary DNS:

Tertiary DNS:

Now comes the hard part. If you know nothing about your network then we are going to be a little stuck. This is the part that is specific to your installation. I am not going to attempt to explain how networks function, if you have one they you should at least understand the terms in the next couple of paragraphs.

As shown on the screenshot on the previous page, the default for a Red Hat installation is to use DHCP. If you are using a DHCP server on your network then you can click **NEXT**. If you are not then you will need to click the 'edit' button, then deselect DHCP and enter an IP address and subnet mask for this machine. As a side note most home run networks tend to use the address range 192.168.0.x and a subnet mask of 255.255.255.0 – The x being replaced with the individual IP address of a particular machine. You can establish this by looking at one of your other machine's, and its configuration settings. Again, generally, the Gateway and DNS are usually going to be something like 192.168.0.1 for both. This is however VERY SPECIFIC to your network. You much make the setting match your existing network.

If you are using DHCP then for the hostname you can let DHCP handle it. If not then select 'manually' and type a name for this host.



Network Configuration

Any network devices you have on the system will be automatically detected by the installation program and shown in the **Network Devices** list.

To configure the network device, first select the device and then click **Edit**. In the **Edit Interface** screen, you can choose to have the IP and Netmask information configured by DHCP or you can enter it manually. You can also choose to make the device active at boot time.

If you do not have DHCP client access or are unsure as to what this information is, please contact your Network Administrator.

Network Configuration

Network Devices

Active on Boot	Device	IP/Netmask
☑	eth0	192.168.1.1/255.255.255.0

Hostname

Set the hostname:

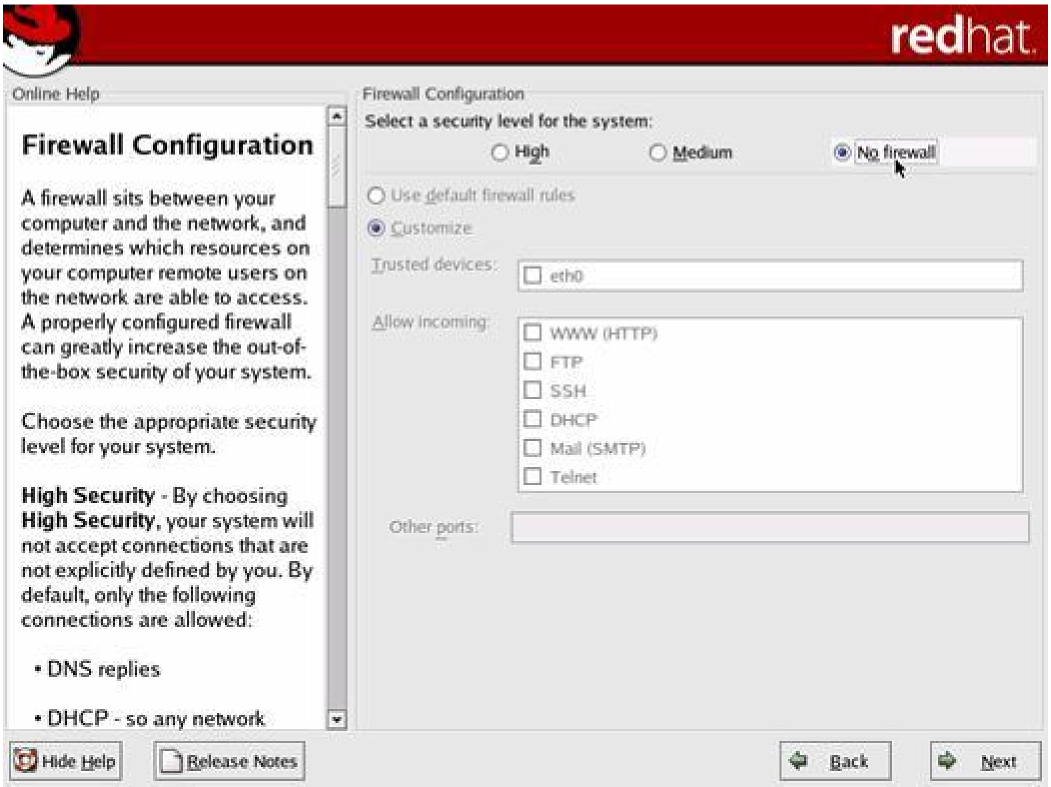
○ automatically via DHCP

● manually mypbx

Miscellaneous Settings

Gateway:	192	168	1	200
Primary DNS:	192	168	1	108
Secondary DNS:				
Tertiary DNS:				

Again, for a non-DHCP environment fill in the Gateway and Primary DNS fields. Do NOT use the settings in the above screenshot unless they match your network. Once you have done this click **NEXT**



Online Help

Firewall Configuration

A firewall sits between your computer and the network, and determines which resources on your computer remote users on the network are able to access. A properly configured firewall can greatly increase the out-of-the-box security of your system.

Choose the appropriate security level for your system.

High Security - By choosing **High Security**, your system will not accept connections that are not explicitly defined by you. By default, only the following connections are allowed:

- DNS replies
- DHCP - so any network

Firewall Configuration

Select a security level for the system: ○ High ○ Medium ● No firewall

○ Use default firewall rules

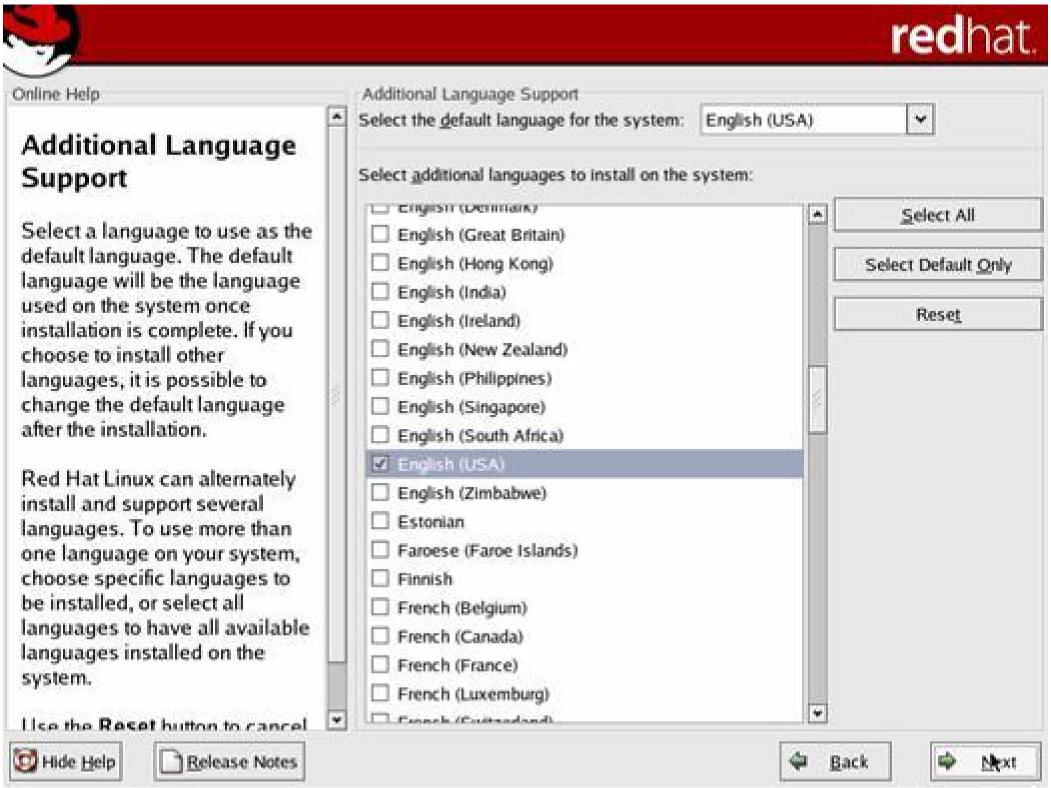
● Customize

Trusted devices: □ eth0

Allow incoming: □ WWW (HTTP) □ FTP □ SSH □ DHCP □ Mail (SMTP) □ Telnet

Other ports:

Select ‘no firewall’ and click **NEXT**.



Online Help

Additional Language Support

Select a language to use as the default language. The default language will be the language used on the system once installation is complete. If you choose to install other languages, it is possible to change the default language after the installation.

Red Hat Linux can alternately install and support several languages. To use more than one language on your system, choose specific languages to be installed, or select all languages to have all available languages installed on the system.

Use the **Reset** button to cancel

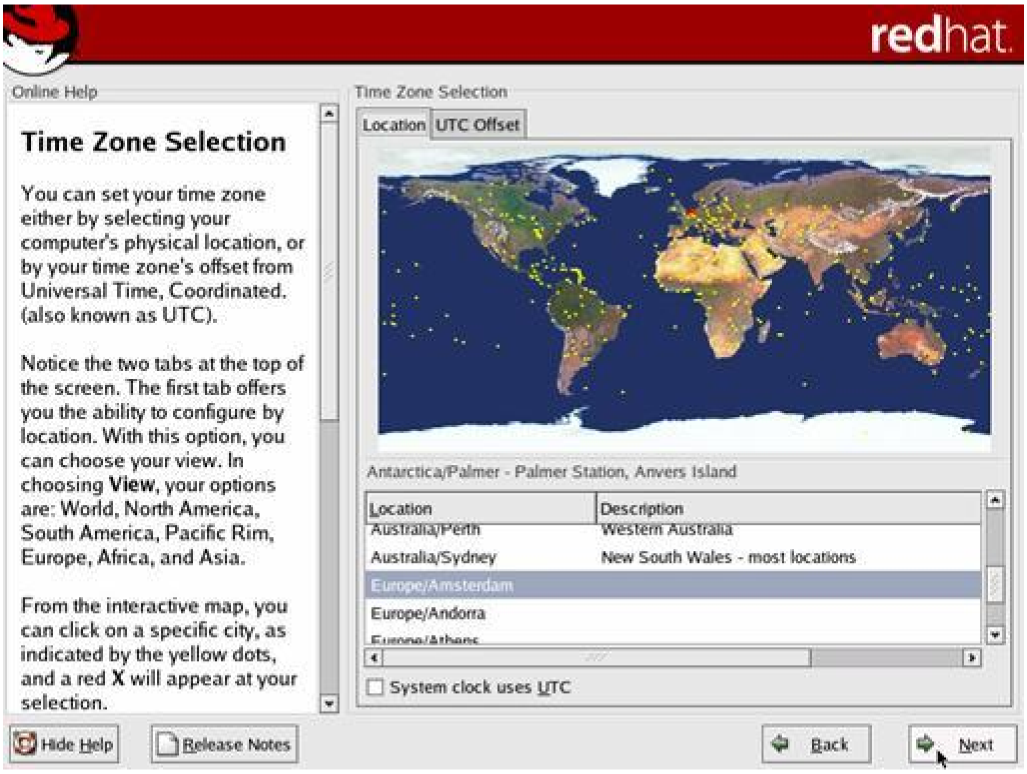
Additional Language Support

Select the default language for the system: English (USA)

Select additional languages to install on the system:

- □ English (Denmark)
- □ English (Great Britain)
- □ English (Hong Kong)
- □ English (India)
- □ English (Ireland)
- □ English (New Zealand)
- □ English (Philippines)
- □ English (Singapore)
- □ English (South Africa)
- ☑ English (USA)
- □ English (Zimbabwe)
- □ Estonian
- □ Faroese (Faroe Islands)
- □ Finnish
- □ French (Belgium)
- □ French (Canada)
- □ French (France)
- □ French (Luxemburg)
- □ French (Switzerland)

Click **NEXT**. If you want to use additional languages, select the ones you want first.



Time Zone Selection

You can set your time zone either by selecting your computer's physical location, or by your time zone's offset from Universal Time, Coordinated. (also known as UTC).

Notice the two tabs at the top of the screen. The first tab offers you the ability to configure by location. With this option, you can choose your view. In choosing **View**, your options are: World, North America, South America, Pacific Rim, Europe, Africa, and Asia.

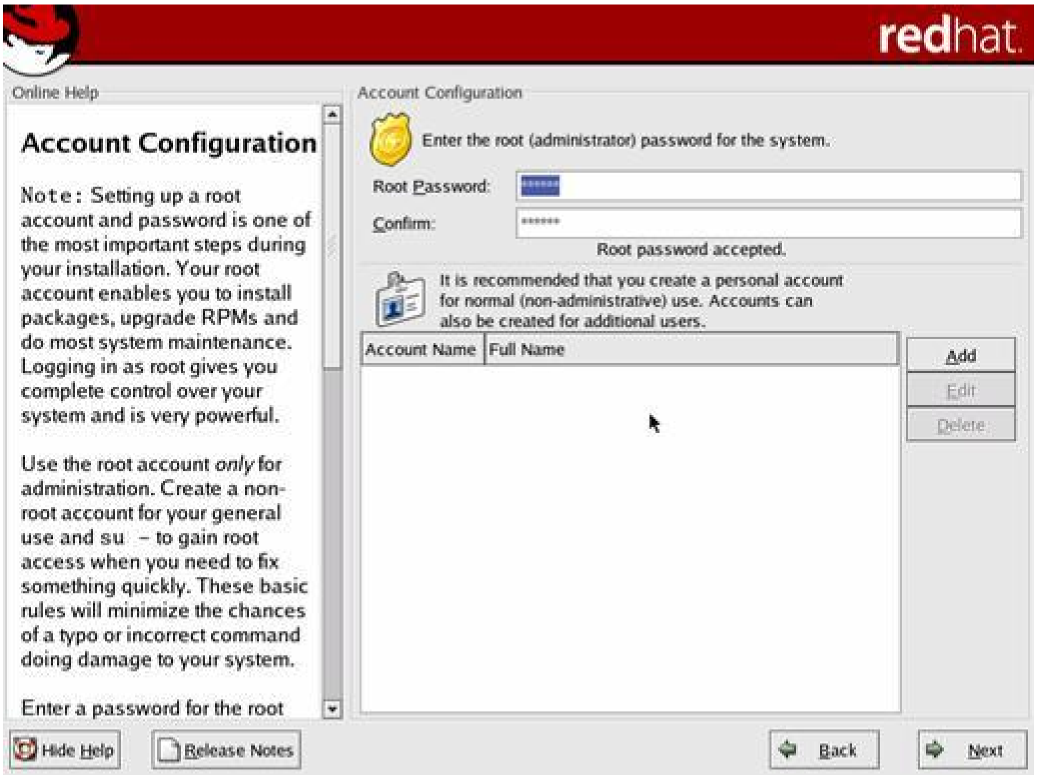
From the interactive map, you can click on a specific city, as indicated by the yellow dots, and a red **X** will appear at your selection.

Antarctica/Palmer - Palmer Station, Anvers Island

Location	Description
Australia/Perth	Western Australia
Australia/Sydney	New South Wales - most locations
Europe/Amsterdam	
Europe/Andorra	
Europe/Athens	

□ System clock uses UTC

Select your location or at least the nearest place to you that is listed and click **NEXT**.



Account Configuration

Note: Setting up a root account and password is one of the most important steps during your installation. Your root account enables you to install packages, upgrade RPMs and do most system maintenance. Logging in as root gives you complete control over your system and is very powerful.

Use the root account *only* for administration. Create a non-root account for your general use and su – to gain root access when you need to fix something quickly. These basic rules will minimize the chances of a typo or incorrect command doing damage to your system.

Enter a password for the root

Enter the root (administrator) password for the system.

Root Password: ******

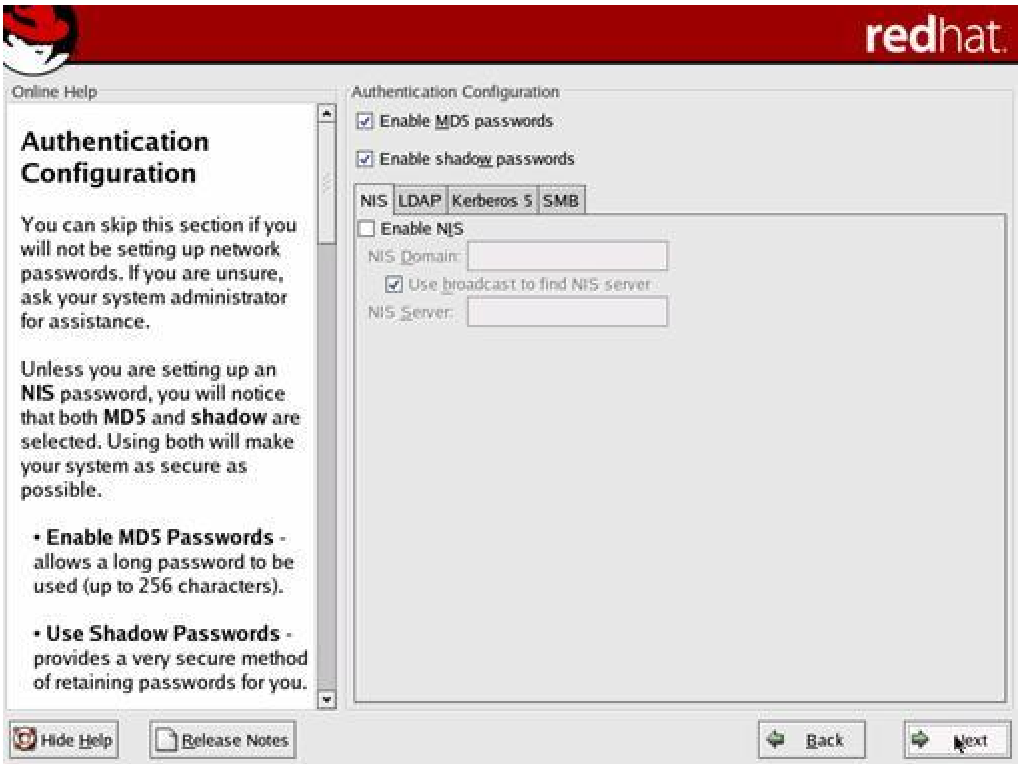
Confirm: ******

Root password accepted.

It is recommended that you create a personal account for normal (non-administrative) use. Accounts can also be created for additional users.

Account Name	Full Name

Enter a password for the 'root' user. If you were unaware, the root user is like the administrator account on a Windows machine, the big boss account.. Use TAB or the mouse to move between the fields. Once you've done that (and remembered the password!) click **NEXT**.



Authentication Configuration

You can skip this section if you will not be setting up network passwords. If you are unsure, ask your system administrator for assistance.

Unless you are setting up an **NIS** password, you will notice that both **MD5** and **shadow** are selected. Using both will make your system as secure as possible.

- **Enable MD5 Passwords** - allows a long password to be used (up to 256 characters).
- **Use Shadow Passwords** - provides a very secure method of retaining passwords for you.

☑ Enable MD5 passwords

☑ Enable shadow passwords

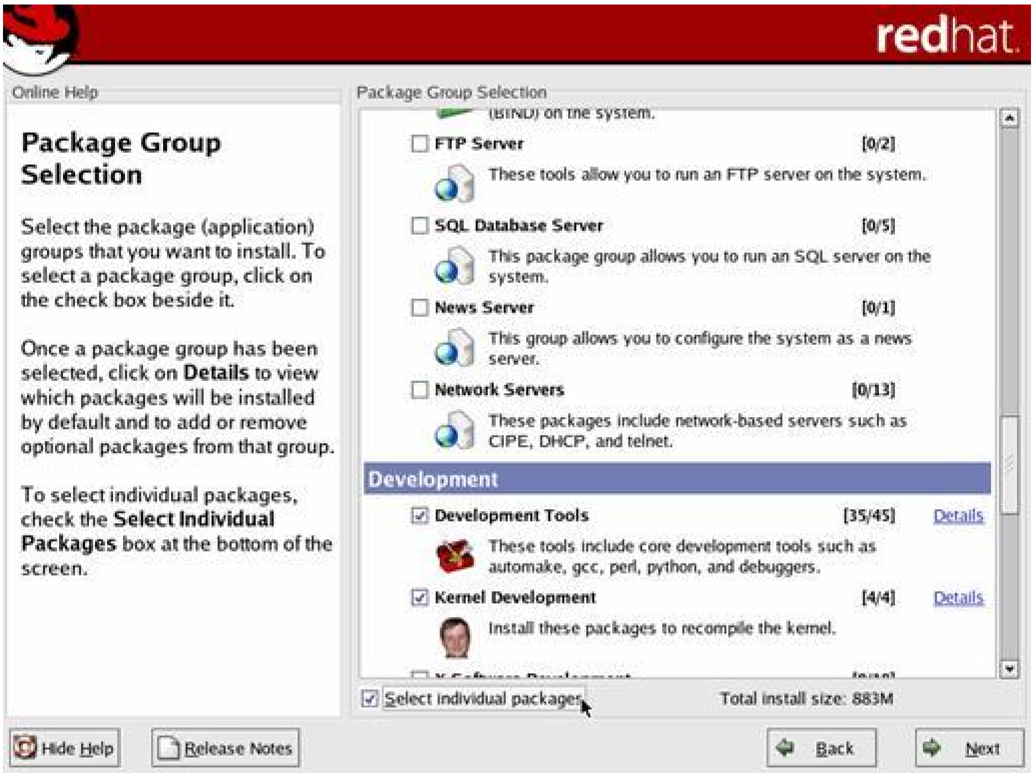
□ Enable NIS

NIS Domain:

☑ Use broadcast to find NIS server

NIS Server:

Click **NEXT**



Package Group Selection

Select the package (application) groups that you want to install. To select a package group, click on the check box beside it.

Once a package group has been selected, click on **Details** to view which packages will be installed by default and to add or remove optional packages from that group.

To select individual packages, check the **Select Individual Packages** box at the bottom of the screen.

(BIND) on the system.

□ **FTP Server** [0/2]
These tools allow you to run an FTP server on the system.

□ **SQL Database Server** [0/5]
This package group allows you to run an SQL server on the system.

□ **News Server** [0/1]
This group allows you to configure the system as a news server.

□ **Network Servers** [0/13]
These packages include network-based servers such as CIPE, DHCP, and telnet.

Development

☑ **Development Tools** [35/45] Details
These tools include core development tools such as automake, gcc, perl, python, and debuggers.

☑ **Kernel Development** [4/4] Details
Install these packages to recompile the kernel.

□ X Software Development [0/10]

☑ Select individual packages

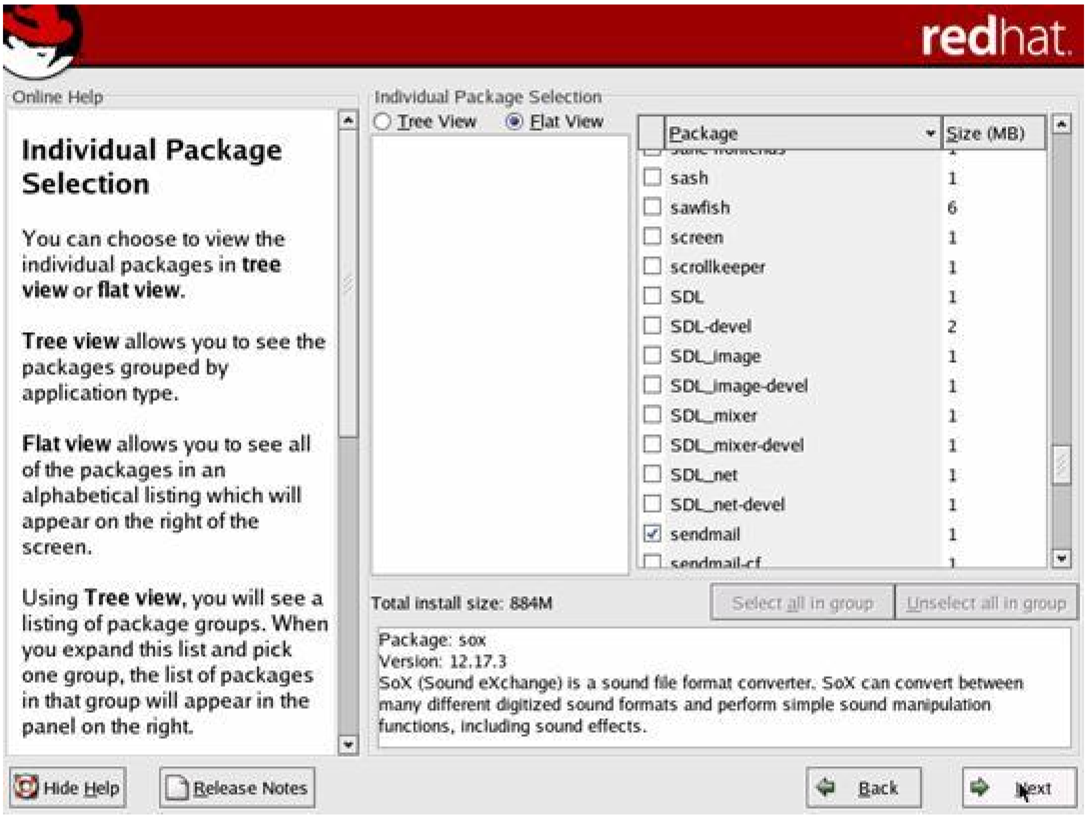
Total install size: 883M

Now we have to select the applications that we want to install. The only things you need to select are:

- **Development Tools**
- **Kernel Development**
- **Text Based Internet (optional but advised)**

Make sure you also have ‘Select individual packages’ selected (see cursor on previous page screenshot).

Now click **NEXT**



Online Help

Individual Package Selection

You can choose to view the individual packages in **tree view** or **flat view**.

Tree view allows you to see the packages grouped by application type.

Flat view allows you to see all of the packages in an alphabetical listing which will appear on the right of the screen.

Using **Tree view**, you will see a listing of package groups. When you expand this list and pick one group, the list of packages in that group will appear in the panel on the right.

Individual Package Selection

○ Tree View ● Flat View

	Package	Size (MB)
□	sash	1
□	sawfish	6
□	screen	1
□	scrollkeeper	1
□	SDL	1
□	SDL-devel	2
□	SDL_image	1
□	SDL_image-devel	1
□	SDL_mixer	1
□	SDL_mixer-devel	1
□	SDL_net	1
□	SDL_net-devel	1
☑	sendmail	1
□	sendmail-cf	1

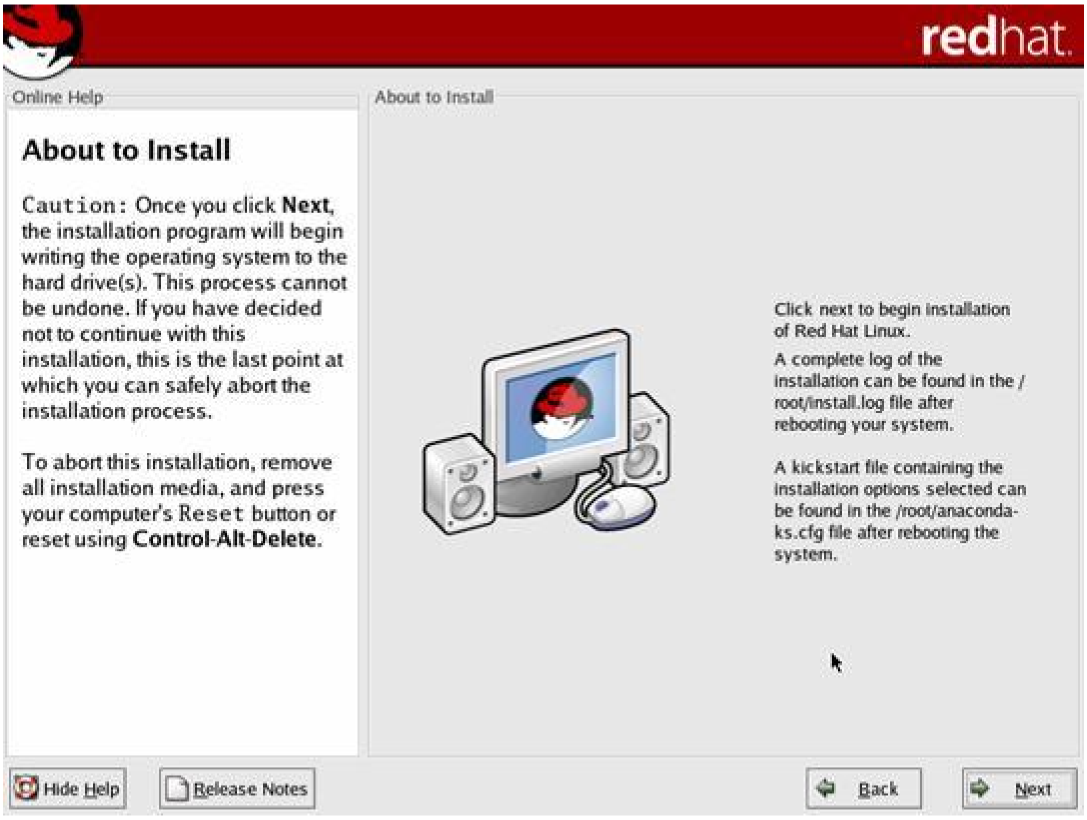
Total install size: 884M

Package: sox
Version: 12.17.3
SoX (Sound eXchange) is a sound file format converter. SoX can convert between many different digitized sound formats and perform simple sound manipulation functions, including sound effects.

Select 'Flat View' and ensure that the following ***ADDITIONAL*** items are selected:

- OpenSSL-Devel
- Readline41
- Ncurses4
- Ncurses C++ Devel
- SOX

Now click **NEXT**



About to Install

Caution: Once you click **Next**, the installation program will begin writing the operating system to the hard drive(s). This process cannot be undone. If you have decided not to continue with this installation, this is the last point at which you can safely abort the installation process.

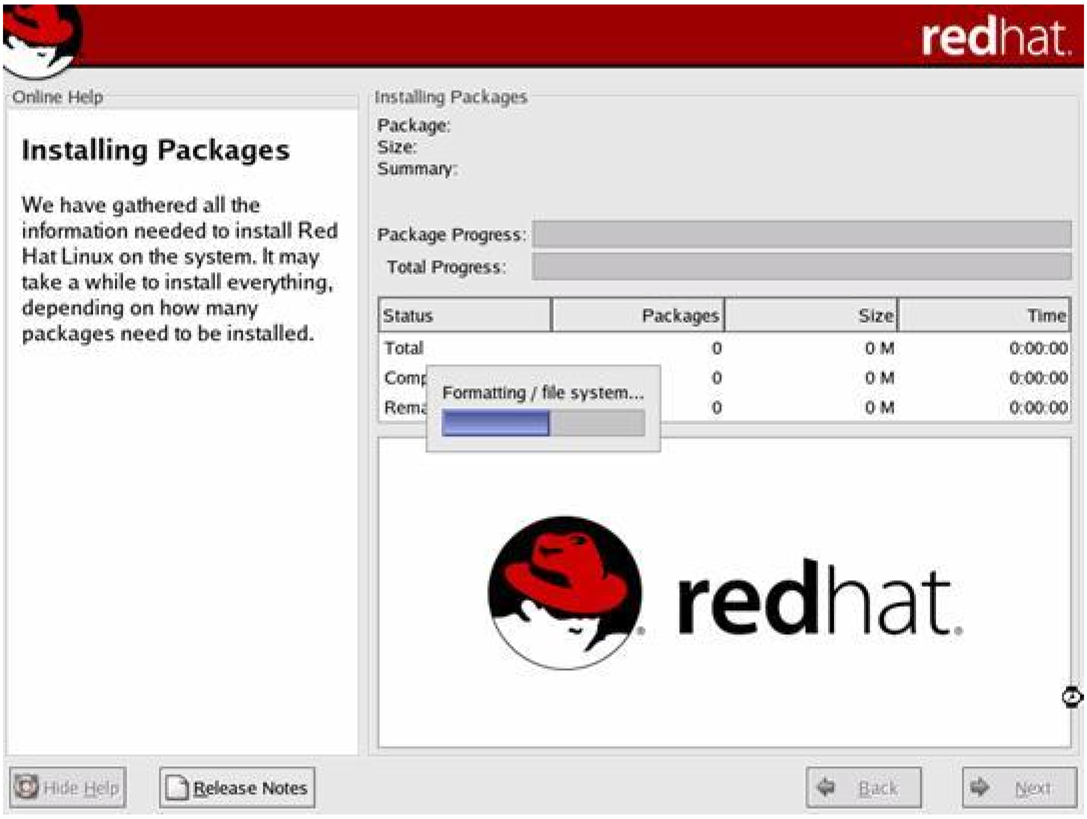
To abort this installation, remove all installation media, and press your computer's Reset button or reset using **Control-Alt-Delete**.

Click next to begin installation of Red Hat Linux.

A complete log of the installation can be found in the /root/install.log file after rebooting your system.

A kickstart file containing the installation options selected can be found in the /root/anaconda-ks.cfg file after rebooting the system.

Ok, this is it, we are about to do the install, Click **NEXT**



Installing Packages

We have gathered all the information needed to install Red Hat Linux on the system. It may take a while to install everything, depending on how many packages need to be installed.

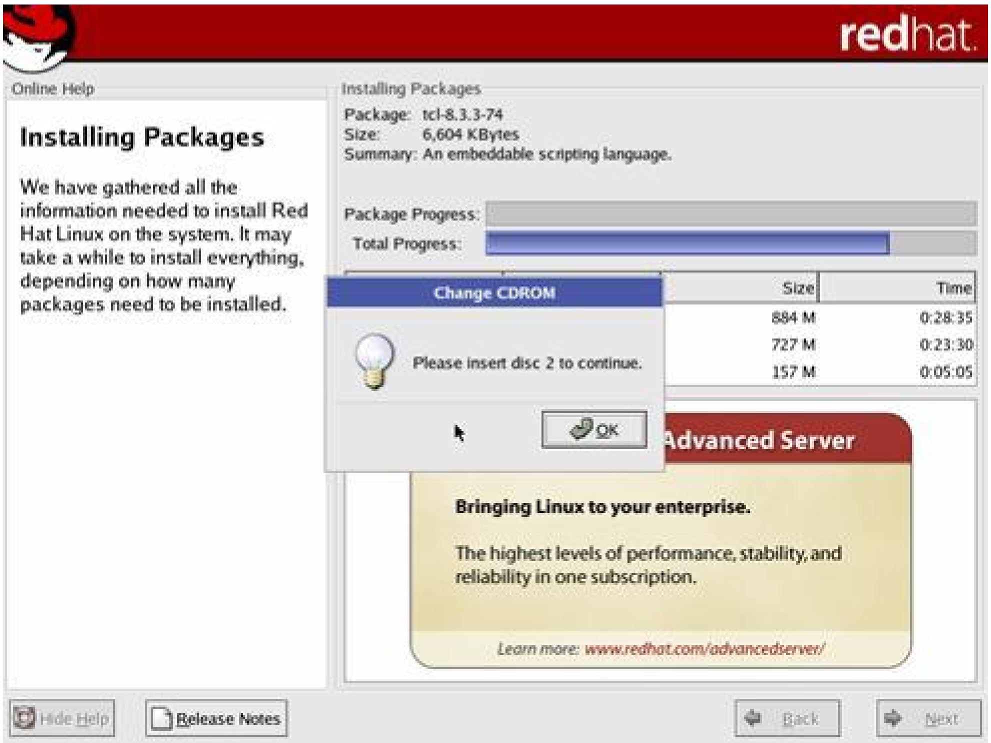
Package:
Size:
Summary:

Package Progress:
Total Progress:

Status	Packages	Size	Time
Total	0	0 M	0:00:00
Comp	0	0 M	0:00:00
Rema	0	0 M	0:00:00

Formatting / file system...

The first ting you'll see is the disks being formatted and partitioned, once complete the various packages will be installed. You will be prompted to insert disks, change the CD, and then click OK.



Installing Packages

We have gathered all the information needed to install Red Hat Linux on the system. It may take a while to install everything, depending on how many packages need to be installed.

Package: tcl-8.3.3-74
Size: 6,604 KBytes
Summary: An embeddable scripting language.

Package Progress:
Total Progress:

	Size	Time
	884 M	0:28:35
	727 M	0:23:30
	157 M	0:05:05

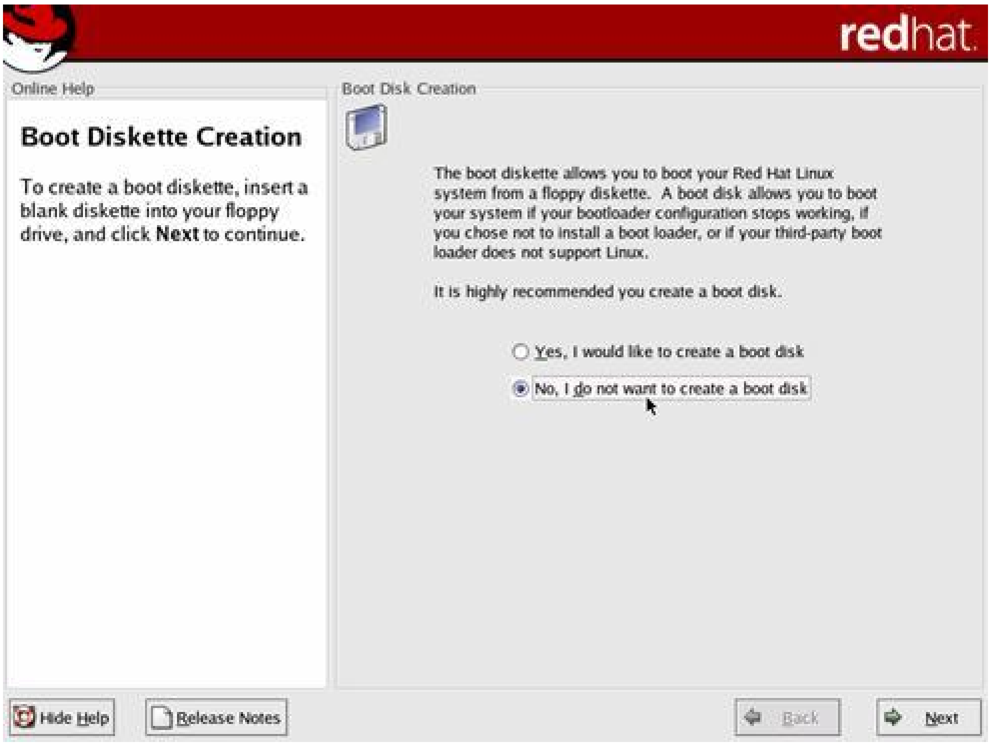
Please insert disc 2 to continue.

Advanced Server

Bringing Linux to your enterprise.

The highest levels of performance, stability, and reliability in one subscription.

Learn more: www.redhat.com/advancedserver/



Boot Diskette Creation

To create a boot diskette, insert a blank diskette into your floppy drive, and click **Next** to continue.

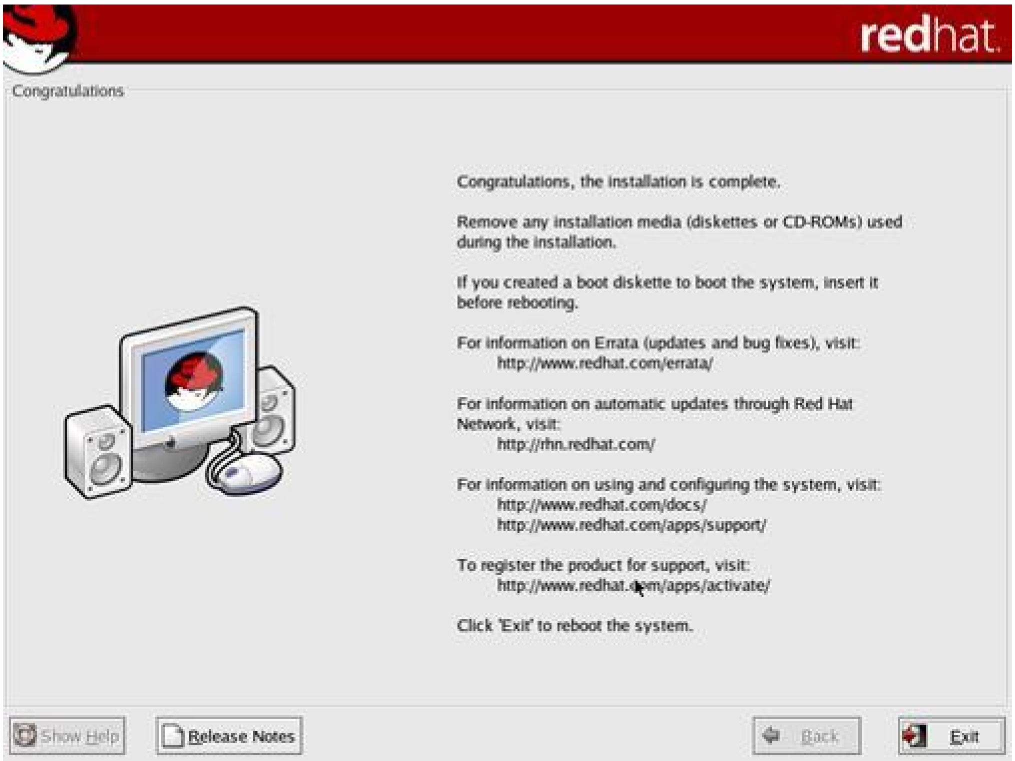
The boot diskette allows you to boot your Red Hat Linux system from a floppy diskette. A boot disk allows you to boot your system if your bootloader configuration stops working, if you chose not to install a boot loader, or if your third-party boot loader does not support Linux.

It is highly recommended you create a boot disk.

○ Yes, I would like to create a boot disk

● No, I do not want to create a boot disk

You don't really need to create a boot disk. Select no and click **NEXT**.



Congratulations, the installation is complete.

Remove any installation media (diskettes or CD-ROMs) used during the installation.

If you created a boot diskette to boot the system, insert it before rebooting.

For information on Errata (updates and bug fixes), visit:
    http://www.redhat.com/errata/

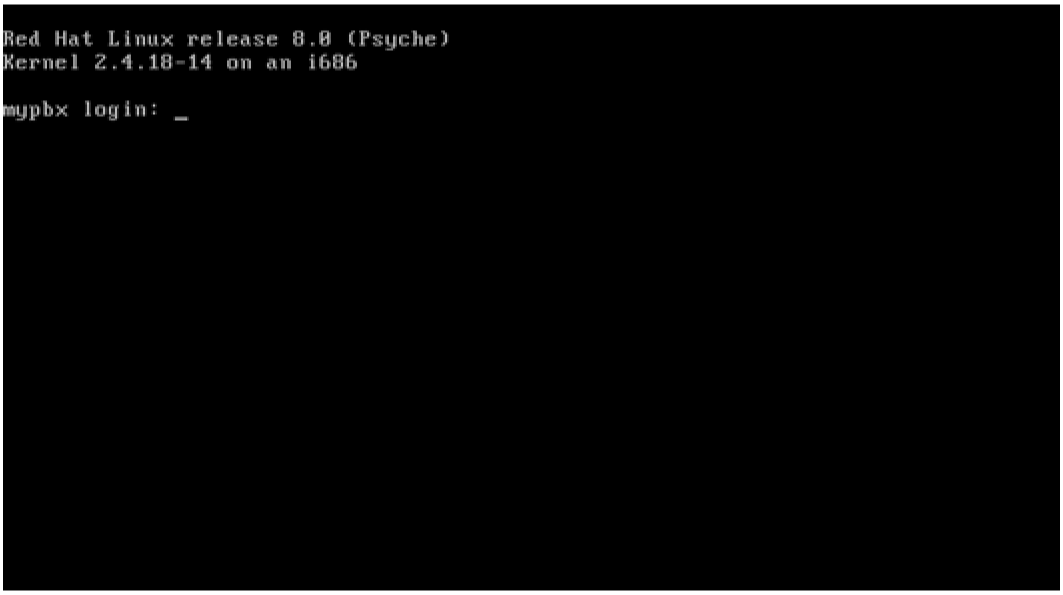
For information on automatic updates through Red Hat Network, visit:
    http://rhn.redhat.com/

For information on using and configuring the system, visit:
    http://www.redhat.com/docs/
    http://www.redhat.com/apps/support/

To register the product for support, visit:
    http://www.redhat.com/apps/activate/

Click 'Exit' to reboot the system.

Do NOT remove the CD, the install will eject it for you, Click **EXIT** and your machine will reboot. Once complete you will be presented with something like:



```
Red Hat Linux release 8.0 (Psyche)
Kernel 2.4.18-14 on an i686

mypbx login: _
```

You may now dance around the room a little; you have completed the Linux install part of Asterisk…

Get a coffee, tea, cola or drink of your choice and bask in this glorious moment. Then read the next section on installing Asterisk.

Additional Installs

You may also need (depending on your desire to have music on hold) mpg123 which can be found at http://www.mpg123.de/mpg123/mpg123-0.59r.tar.gz

From your Linux box you can also type

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

Extract it and compile

```
# tar -zxvf mpg123-0.59r.tar.gz
# cd  mpg123-0.59r
# make linux
# make install
```

Once compiled make sure there is a copy in

```
/usr/bin/mpg123
```

Note that the copy of mpg123 that comes with Red Hat 8 is of no use to you, it will not work with Asterisk.

Obtaining Asterisk PBX

You can run Asterisk without any of Digium's hardware but unless you have some sort of PSTN gateway/card you'll only be making network calls.

From the Asterisk website download page:

The best way to obtain Asterisk is to check out a fresh copy from the CVS Server.
To check out code from our CVS repository:

```
# cd /usr/src
# export CVSROOT=:pserver:anoncvs@cvs.digium.com:/usr/cvsroot
# cvs login            - the password is anoncvs.
# cvs checkout zaptel libpri asterisk
```

This will create four directories, zaptel, libpri, and asterisk. Compiling them is generally quite straightforward. Just change to each directory and type make install, in this order. Compile zaptel, then libpri, and then asterisk.

```
# cd zaptel
```

note if you do not have any Digium hardware and you want to use music on hold etc you may find

that you need to modify the Makefile in this directory. Near the end of line 82, change # ztdummy to ztdummy
(i.e. remove the #) then continue

```
# make clean ; make install
# cd ../libpri
# make clean ; make install
# cd ../asterisk
# make clean ; make install
```

Remember to do a

```
# make samples
```

to generate the basic configuration files. Hopefully everything compiled ok, you can test that Asterisk will run by typing

```
# asterisk –vvvvc
```

And hitting return, you should see screens and screens of information that means nothing to you at all. This doesn't matter at this point, at least Asterisk works! Type

```
STOP NOW
```

And hit enter in the Asterisk command line, Asterisk will exit.

Adding Digium Cards

Digium sell a number of cards for use with Asterisk, these range from single FXO in the guise of the X100P through to fully fledged quad E/T1 cards. If you are setting up Asterisk at home then an X100P (or two if you have 2 lines) should be enough for your needs. You might also want to get an FXS card so that you can get standard analog phones working with Asterisk.

My own configuration at home consists of a single X100P and a dual port TDM20B card – TDM400P is the generic name for the card, specific names exist for the number of ports on the card i.e.:

TDMx0B where x is either 1, 2, 3 or 4 – denoting the number of ports.

There are a couple of things to note about the X100P. Although the card should work in your country things like caller id detection may not. In the USA you should have no problems at all, but in Europe/rest of the world you should consider searching the archives for information.

At the time of writing the following countries appear to have either full or rudimentary support for caller id:

USA	Full
UK	Not on BT, *SOME* cable companies may work.
FR/DE	Just being implemented (patched)
RO	Yes (patched)

Those that don’t work include:

NL/DK (although some preliminary work has been done on these locations and hopefully they will be supported soon)

If anyone has any other locations that are known to work/not work please let me know. If you are not going to be adding cards to your system you may want to skip this section to get started, although it is recommended that you read the entire document for completeness.

In the machine I use for Asterisk; a Dell Optiplex GX115 I have disabled the soundcard since I do not need it. You may wish to keep it enabled but bear this in mind if you get IRQ conflicts.

X100P Installation

I hope I can make the assumption that you are capable of adding a new card to your system, if not get someone who knows what they are doing to help out. Oh and please wear a wrist strap.

The X100P card is what we call an FXO (Foreign Exchange Office) card. Unless you know about telecoms then that probably didn’t mean much. What it means to you and me is that you can plug a cable into the phone socket on your wall and into the back of the X100P; this is your incoming line from your telecoms provider (KPN/BT/Bell etc).

You may have noticed that there are 2 sockets on the back of the X100P; one marked as the line interface and the other as a phone interface. Plug in an appropriate cable between the wall socket and the line interface on the card. If you have a spare phone then plug this into the phone interface on the card too. It is always good to have a phone plugged into this interface because in the event of asterisk failing, or a power cut the card actually still allows access to the PSTN line. Obviously if you decide to use a phone that is not powered from the phone line, if you have a power cut, it will not work.

TDM400P Installation

The TDM400P card is an FXS (Foreign Exchange Station) card; it’s what you plug standard analog phones into to use them with asterisk.

For those using phones from the UK you *may* need an adaptor that provides a ring capacitor to actually get the phone to physically ring. I have both a BT Easicom 1000 and a Panasonic KXTCD-777 (this is a DECT base station and 2 handsets sold in one box). The Panasonic phones do not require the adaptor with ring capacitor but the BT Easicom does.

If you are using phones from the USA (aside from any power requirements they may have) you should just be able to plug them in.

Configuration

Once the cards have been physically plugged into your Asterisk machine, power it up again. Once you've logged in you'll need to configure the cards. It's worth pointing out that you do **not** have to have one of each card type but this section will make the assumption that you have got one of each so that we can cover a much fuller configuration.

The first thing you'll need to do is modprobe for the cards, basically this makes your linux system 'poke' the card via its driver

```
modprobe zaptel
modprobe wcfxo
modprobe wcfxs
```

The order in which you do the modprobe's IS important. If you modprobe the FXO (modprobe wcfxo) card first then it will be channel 1, if you modprobe the FXS (modprobe wcfxs) card first then its first port will be channel 1, the second channel 2 and so on…

Now you need to tell asterisk which ports are for what… edit /etc/zaptel.conf

```
# vi /etc/zaptel.conf
```

Note to exit vi hit the escape key then type **: wq** to write the file and quit. If you want to quit without saving use **q!** instead of **wq**

My zaptel.conf consists of the following

```
fxsks=1
fxoks=2
fxoks=3
loadzone=nl
defaultzone=nl
```

The loadzone and defaultzone are fairly obvious and are simply your ISO country code – Not all country codes are available, but to be honest I was running with 'us' for a couple of months here in Holland with no issues..

The first 3 lines are very important so pay attention.

Just to confuse you more FXO ports use FXS signaling and FXS ports use FXO signaling. So from the above configuration you can see that port 1 (fxsks=1) is actually an FXO card (X100P) and ports 2 and 3 are on the FXS card (TDM20B). Note that you do not have to have the cards in this order, but they must match the order you modprobed them. Since I have a single port FXO card and a 2 port FXS (with the ability to add 2 more ports) it makes sense to put the FXO card as the first, otherwise I'd have to change my configuration files a bit when (if) I added more FXS ports.

Save your file and then edit /etc/asterisk/zapata.conf

```
# vi /etc/asterisk/zapata.conf
```

This is where you set the configuration for each of the channels on the cards (you can think of the channels as lines if you like). It's important to note that the configuration in zapata.conf is what I would call 'backwards' – i.e. you set the features for a channel, then
you assign the channel to them. There is actually a very good reason for this. We are only dealing with a couple of channels so it's not a big deal, but imagine if you were dealing with 128, 256 or more… Doing it 'backwards' allows you to set the configuration for all the channels and assign them all at once (making slight modifications if required)

When you edit the file you may see things like

```
[channels]

relaxdtmf=yes
callwaiting=yes
callwaitingcallerid=yes
threewaycalling=yes
transfer=yes
cancallforward=yes
usecallerid=yes
```

these are all pretty specific to your environment, they take either yes or no as the argument. Hopefully what they do is obvious, for example:

```
callwaiting=yes
```

gives you a call waiting tone if you are on the phone and someone else rings. Many of these features need to be supported by your local telco (Bell/BT/KPN..etc)

At the bottom of this section are example zaptel and zapata config files that should work immediately if your configuration MATCHES mine – i.e. you have 1 X100P and 1 TDM20B.

Some of the options available in this configuration file are as follows:

More than likely you will want to turn on echo cancellation

```
echocancel=yes
echocancelwhenbridged=yes
```

The gain options allow us to increase or decrease the ‘volume’ if the devices we have appear to quite or loud – to be honest you should not really need to change the rx/tx gain settings under normal circumstances.

```
rxgain=0.0
txgain=0.0
```

Groups allow us to bundle devices together to act more or less as one, got example in Dial strings you can use g<device-group-number>. A dial string would look something like this:

```
Dial(ZAP/g1/1234567890,60)
```

There is a good reason for this, if for example, you had 10 outgoing lines, you would normally want to dial out on any line that was available, not a specific one. Using the group option will allow asterisk to search for a free line within that group.

```
group=1
```

Pickup groups allow you to group devices, phones or lines together so that you can pick them up using *8. For example if a phone connected to FXS port 1 rings you may want the phone on FXS port 2 to be able to pick it up.

```
pickupgroup=1-4
```

The immediate feature is what I like to call theBatphone option. If this is set to yes, when you pick up the line it will automatically dial the ‘s’ extension for the context this line is in. Think back to the TV show Batman, when Commissioner Gordon was in his office and needed to talk to Batman, he simply picked up the receiver on the red Batphone and it rang in the Batcave. So you see, even super heroes can use asterisk!

In this instance it’s of no real use to us, as we are defining the PSTN line..

```
immediate=no
```

The context is the place in extensions.conf that calls on this line will jump to. It’s a good idea to identify the 2 types of card in using this. Setting the FXO care to jump to the Bell context when a call comes in…

```
context=bell
```

As I said before, FXO devices use FXS signaling so we need to set that here too, it needs to match what you had in your zaptel.conf file BUT, and the reson is unknown to me, instead of fxsks we need to use fxs_ks. There are of course other signaling methods (see the draft manual for all the options)

```
signalling=fxs_ks
```

If we have callerid we can pass it on to asterisk's extension logic

```
callerid=asreceived
```

All the settings above have not been assigned to a channel yet, so we do that next

```
channel=1
```

It's worth noting that all the values (above the channel=1) now become the default values for all other channels, this means you only need to change those you want and all others will be taken from above.

```
context=home
group=2
signalling=fxo_ks
mailbox=2468                   ; you will need to create this mailbox!
callerid="Phone 1" <2468>
channel=2


signalling=fxo_ks
mailbox=3579                   ; you will need to create this mailbox!
callerid="Phone 2" <3579>
channel=3
```

So, finally we have the zapata and zaptel config files complete, they should look similar to this:

```
/etc/zaptel.conf

fxsks=1
fxoks=2
fxoks=3
loadzone=nl
defaultzone=nl


/etc/asterisk/zapata.conf

[channels]

busydetect=1
busycount=7

relaxdtmf=yes
callwaiting=yes
callwaitingcallerid=yes
threewaycalling=yes
transfer=yes
cancallforward=yes
```

```
usecallerid=yes

echocancel=yes
echocancelwhenbridged=yes

rxgain=0.0
txgain=0.0

group=1
pickupgroup=1-4

immediate=no

context=bell

signalling=fxs_ks
callerid=asreceived
channel=1

context=home

group=2
signalling=fxo_ks
mailbox=2468
callerid="Phone 1" <2468>
channel=2

signalling=fxo_ks
mailbox=3579
callerid="Phone 2" <3579>
channel=3
```

You will need to restart asterisk completely if you make changes to these files at a later date. At this point we can also run ztcfg with –vv (v v not w)

```
[root@ASTERISK asterisk]# ztcfg -vv

Zaptel Configuration
======================


Channel map:

Channel 01: FXS Kewlstart (Default) (Slaves: 01)
Channel 02: FXO Kewlstart (Default) (Slaves: 02)
Channel 03: FXO Kewlstart (Default) (Slaves: 03)

3 channels configured.
```

The next stage is to modify our extensions.conf to configure the numbers for the cards…

[coming soon…]

Rush Hour III – I need something quick

Setting up SIP

The configuration files for Asterisk are stored in /etc/asterisk so go there:

```
# cd /etc/asterisk
```

the files we are particularly interested in are sip.conf and extensions.conf. The first thing we will need to do is edit sip.conf - Use whichever editor you are familiar with under Linux - I use vi. – Note to exit vi hit the escape key then type **: wq** to write the file and quit. If you want to quit without saving use **q** instead of **wq**

```
# vi sip.conf
```

If you've ever looked at a windows .ini file you'll notice the format is very similar. In the [general] section you need to add a line to register asterisk with your sip proxy (eg ix66), you should end up with something like the example below.

```
[general]
port = 5060 ; Port to bind to
bindaddr = 0.0.0.0 ; Address to bind to
context = sip ;default Default for incoming calls
```

An important point here, if you do not have a sip aware firewall and are just using port forwarding then ensure that your context points to somewhere like ‘invalidcalls’. If you do not do this then someone could call one of your extensions direct from the Internet. If you had an FXO card in the machine, this could lead to them being able to make PSTN calls!!

```
register => me@mysipproxy.com/1000
```

A couple of things to note here, obviously the sip address "me@mysipproxy.com" is one that someone would normally call you on. The /1000 at the end of the 'register =>' line is actually the extension that asterisk will use for this address, ie when someone calls me@mysipproxy.com extension 1000 will ring.

If you re registering with FreeworldDialup then you will need to add your password to the register line too: e.g.

```
register => 21103:mypassword@fwd.pulver.com/1000
```

You now need a section to define the proxy information:

```
[mysipproxy.com]
type=peer
host=192.168.72.2
fromuser=andy
secret=mypassword
fromdomain=mysipproxy.com
```

That part is fairly straight forward using these guidelines:

[section name] - the section name is the bit after the @ in your 'register =>' line

type = - this is a peer.

host= - the ip address of the ASTERISK server.

fromuser - username used for registration

secret - password for this account if required, if no password is required comment it out using a ; in front

fromdomain - the domain that the username is from.

For FreeworldDialup use

```
[fwd.pulver.com]
type=friend
secret=mypassword
username=my fwd number
host=fwd.pulver.com
```

We now need to define a couple of phones. I use Snom 100's, but you may just want to use sjphone (http://www.sjlabs.com) or something like that.

```
[phone1]
type=friend
host=dynamic
defaultip=192.168.1.4
;username=blah
;secret=blah
dtmfmode=rfc2833 ; Choices are inband, rfc2833, or info
mailbox=1000 ; Mailbox for message waiting indicator
context=sip
callerid="Me" <2124>

[phone2]
type=friend
;secret=blah
```

```
host=dynamic
defaultip=192.168.1.3
dtmfmode=rfc2833 ; Choices are inband, rfc2833, or info
mailbox=1000 ; Mailbox for message waiting indicator
context=sip
callerid="Mini Me" <2123>
```

Since the configuration for both phones will be similar I'll only deal with 'phone1'. First of all the [phone1] is important, it's what you will use in your extensions file to identify this physical phone when issuing Dial commands. (e.g. Dial(SIP/phone1,20,tr) - we'll get to this later).

The **type=** is going to be a friend in this instance, peer is used when Asterisk is contacting a proxy, user is used for phones that can only make calls, and friend acts as both a peer and a user.

Set your **host=**dynamic and the IP address in the **defaultip**= entry too. If your host has an entry in your DNS then you just enter the machines name in the **host=** field. Unless you have a good reason to leaving it as dynamic and using the IP given to the phone (fixed or DHCP allocated) is perfectly ok.

The username and secret fields are only used when the username is not the same as the client. You should be able to safely ignore this for the moment.

dtmfmode= is somewhat trial and error, it depends on your physical phone or software, I've found that my Snom 100's need rfc2833 for DTMF tones to be heard, and SJphone needs inband. Essentially it's trial and error, but once you establish which is needed for what type of phone, it applies to all. This should not be installation specific, ie I've just told you that Snom 100's use rfc2833, so you can use those too..(for the Snoms also change the phone config.

mailbox= is the voicemail mailbox that is checked for messages, this information is then passed to the phone, on a Snom 100 phone it results in an envelope icon. Usually you'll have more than one mailbox, one for each user or extension.

context= is extremely important you should, in the first instance make this the same for all your sip clients. If a phone is not in a a valid context you will not be able to use it. I've used 'sip' you can use whatever you like, but make sure they are the same, you will need to make an entry in your extensions.conf file (which we will get to later)

callerid= is pretty self explanatory, however, note that the callerid must be in the format above, ie "text" <number> it's the whole line that's the callerid, not just the bit in quotes. Essentially the name goes in the quotes and the number between the < and >.

Once you've set up these phones you'll need to add some information in your extensions.conf file:

```
# vi extensions.conf
```

You'll see there is a lot of information here, for the moment ignore this. Go to the bottom of the file and add [sip] and press return. This needs to be the same as the context= section for your phones defined in sip.conf surrounded by square brackets. If I had set the context= to wibble (context=wibble) then this section would be called [wibble].

Next we'll add a couple of extensions for the phones defined in sip.conf

```
exten => 1,1,Dial(SIP/phone1,20,tr)
exten => 2,1,Dial(SIP/phone2,20,tr)
exten => 1000,1,Dial(SIP/phone1&SIP/phone2,20,tr)
```

Add the above lines in the [sip] section. The format of the exten lines is fairly simple:

```
exten => extension number, command priority, command
```

Since an extension may have a number of commands there may be multiple entries for an exten. We'll do some more complex stuff later but for the moment save this file.

When you start asterisk without any parameters it runs as a daemon. Enter the following command:

```
# asterisk
```

If you get the error

```
ERROR[8192]: File asterisk.c, Line 1249 (main): Asterisk already running on /var
```

then asterisk is already running and you should use the command

```
# asterisk –r
```

to connect to the daemon, adding vvvvvvvvvgc to this will give us debug information too ie:

```
# asterisk –vvvvvvvgrc
```

If the daemon was running and you got the error type

```
reload
```

and hit enter at the prompt. If you did not get the error you do not need to do anything, however you might want to attach to the daemon anyway using.

```
# asterisk –vvvvvvvgrc
```

You will need to configure your phones to connect to the asterisk machine. This varies from device to device here is an example for **Snom** phones:

Under Home > Settings > SIP > Lines

Add the Name as phone1, the Account as phone1 and the Registrar as the IP of your asterisk pbx and click save. Do the same for your second phone, replacing 'phone1' with 'phone2'.You should now have 2 phones registered to asterisk (it may take a few seconds to actually happen). If you like you can check by typing:

```
sip show peers
```

from the asterisk console, this should result is a display a little like the one below

```
Name/username Host        Mask               Port Status
phone1/phone1 192.168.1.4 (D) 255.255.255.255 5060 Unmonitored
phone2/phone2 192.168.1.3 (D) 255.255.255.255 5060 Unmonitored
```

Go to 'phone1' and dial '2', phone2 should start ringing. If not double check everything. If it does, hang up and then go to phone2 and dial '1'. phone1 should start ringing. When someone calls the address in your sip.conf (eg me@mysipproxy.com) both phones should ring. Of course all this is wonderful if you only want to talk internally, but how can you talk to other people over the internet? If you've registered with a proxy for example FreeworldDialup add an entry like this:

```
[fwd]
exten => _8.,1,Dial,SIP/${EXTEN-1}@fwd.pulver.com,tr
```

Next add

```
include => fwd
```

in the [sip] section of your extensions.conf. This will allow you to dial 8 then the users FreeworldDialup number e.g. 8100001 would dial FWD's voicemail. On hard phones, this is much easier than having to type a full sip address. Got some friends you call often over the internet? just set up an exten line for them:

```
exten => 4002,1,Dial,SIP/keith@his-sip-address.com
```

Now you just dial 4002 and it will call them...

Voicemail - Please leave a message after the tone...

Ok, so you've got the basics going, and it's great - if you happen to sit by you phone all the time. What happens if you are out/away from your desk/sleeping you'll miss those vital calls. We need to set up voicemail to capture all those messages if we miss them.

The first thing we need to do is create the mailbox for Asterisk to use, thankfully there is a little utility to do this:

```
#  /usr/src/asterisk/addmailbox
```

You'll be prompted for a mailbox number,

```
Enter mailbox number:
```

for now enter 9999 and hit return. This will copy a number of files required for the mailbox. Voicemail files can be found in

```
/var/spool/asterisk/vm/
```

each mailbox has is a subdirectory from this point, so mailbox 9999 is

```
/var/spool/asterisk/vm/9999/
```

You don't really need to worry about these files, but at least you now know where they are on your system. The next thing we need to do is modify a few files so that the mailbox can be used. We need to edit the config files, so go to their location:

```
# cd /etc/asterisk
```

The first of the files we are interested in is voicemail.conf, edit the file

```
# vi voicemail.conf
```

The section we are interested in is [default]. If, which is likely, you have the sample configs installed then you will see a number of entries in this file. You can comment them out if you like by placing a semi-colon ';' in front of the line.

```
[default]
;1234 => 4242,Example Mailbox,root@localhost
```

Add the line

```
9999 => 1234,<your name in here>,<your email address here>
```

The format of the file is pretty simple:

```
mailbox number => mailbox password, mailbox description, mailbox user
email address
```

Once you have added the line for the 9999 mailbox, save the file. Next we'll need to edit out sip.conf to make sure that the phones we set up earlier are going to use the correct mailbox for notification.

```
# vi sip.conf
```

find the entries for [phone1] and [phone2] and change the lines that read:

```
mailbox=1000 ; Mailbox for message waiting indicator
```

to read

```
mailbox=9999 ; Mailbox for message waiting indicator
```

Note that this will be of most use if your phones support either a message waiting indicator of some sort.

Save the file. For the moment we are going to use the same mailbox for both phones, but you can set up one for each if you like. Now edit your extensions.conf file

```
# vi extensions.conf
```

locate the [sip] section where you added the entries for your extensions earlier in this document, the lines should look similar to these:

```
exten => 1,1,Dial(SIP/phone1,20,tr)
exten => 2,1,Dial(SIP/phone2,20,tr)
exten => 1000,1,Dial(SIP/phone1&SIP/phone2,20,tr)
```

we are going to add a few more lines to the exten => 1000 entry to add voicemail:

```
exten => 1000,2,VoiceMail,u9999
exten => 1000,102,VoiceMail,b9999
```

add these lines after the entry for exten => 1000, it should look like this

```
exten => 1000,1,Dial(SIP/phone1&SIP/phone2,20,tr)
exten => 1000,2,VoiceMail,u9999
exten => 1000,102,VoiceMail,b9999
```

What have we done? Well, the exten entries are like a list of things to do when we get a call. In this case we have said:

```
exten => 1000,1,Dial(SIP/phone1&SIP/phone2,20,tr)
```

When extension 1000 rings (exten =>1000) , the first thing we do (exten =>1000,1) is dial phone1 and phone 2 (exten =>1000,1,Dial(SIP/phone1&SIP/phone2,20,tr)) and make it ring for 20 seconds.

```
exten => 1000,2,VoiceMail,u9999
```

If the extension is not answered in the 20 seconds, the second entry will be executed, which is voicemail. The mailbox is specified by the u9999 at the end of the line. Now, this is all 'fine and dandy' but what if you are actually on the phone when the call comes in? Well, this is what the next line is for

```
exten => 1000,102,VoiceMail,b9999
```

There are 2 things to note here, the first is that the priority number has jumped way up to 102. This is a feature of Asterisk, and a useful one at that. When the call comes in and Asterisk tries to dial the extension 1000, if you are on the phone, Asterisk will jump to the current priority + 101 (n + 101). This gives us a priority of 102. The second thing to note is that the mailbox number at the end of the line is preceded by a b (b9999) this indicates that the busy message should be played to the user, and then the user should be allowed to leave a message.

This is all very well, people can leave you messages, but at the moment you can't listed to them! Not really useful is it. We need to set up an extension so that you can actually get to the messages. there are 2 ways to set this up. The first is useful in home environments where you are only really going to have one mailbox or where you trust people completely: I say this because the first way to do it is to effectively not use the password.(I'll show you the other was in a moment). Under the exten => entries you've just added add the following lines.

```
exten => 1001,1,Ringing
exten => 1001,2,Wait(2)
exten => 1001,3,VoicemailMain,s9999
```

I've 'padded' this out so that you can follow the chain of events when extension 1001 is dialed. Follow this sequence in this document (don;t actually try it on your phones yet)

When extension 1001 is dialed the following things happen in the following order,

1. The phone gets a ringing tone
2. There is a 2 second wait (the phone is still getting the ringing tone)
3. The call is answered and goes straight to the voicemail menu for mailbox 9999

Note the user doesn't get asked for a password.

The second way to do this is to omit the ',s9999' from the line giving:

```
exten => 1001,3,VoicemailMain
```

When extension 1001 is dialed this time the following things happen in the following order,

1. The phone gets a ringing tone
2. There is a 2 second wait (the phone is still getting the ringing tone)
3. The call is answered and asked the user to enter a mailbox number, then a password. In our examples above you would enter 9999 as the mailbox and 1234 as the password (remember, we added the entry in voicemail.conf)

Save your extensions.conf file ... Sorry but we've been doing quite a bit here and I want to make sure you remember to save the file.

None of the changes you have made are functional yet, if you went to one of your phones and tried to dial extension 1001, you'd more than likely get a message telling you that the number was unavailable. We need to tell Asterisk to reload our .conf files.. So as before (and yes, I used cut and paste :P):

When you start asterisk without any parameters it runs as a daemon. Enter the following command:

```
# asterisk
```

If you get the error

```
ERROR[8192]: File asterisk.c, Line 1249 (main): Asterisk already running on /var
```

then asterisk is already running and you should use the command

```
# asterisk -r
```

to connect to the daemon, adding vvvvvvvvvgc to this will give us debug information too ie:

```
# asterisk -vvvvvvvgrc
```

If the daemon was running and you got the error type

```
reload
```

and hit enter at the prompt. If you did not get the error you do not need to do anything, however you might want to attach to the daemon anyway using.

```
# asterisk -vvvvvvvgrc
```

Now go to phone 1 and dial extension 1000, let phone 2 ring, don't answer it... You should hear a voice announce that

```
"The person at extension 1000 is not available..."
```

After the beep, leave yourself a message and then hang-up. Then dial extension 1001 and listen to your own message, make sure it all works.. Remember, when we first started, we set up a registration with FWD and allocated extension 1000 to it? Well, you've now added voicemail to your FWD number!

If you are using Snom phones and the voicemail sounds 'funny', like the person is underwater add the following lines to your sip.conf under the general section

```
disallow=all
allow=ulaw
allow=alaw
```

Save the file, go to the Asterisk console and type

```
reload
```

and try calling the voicemail again...it should sound a lot better. Sadly this uses more bandwidth, but Snom has assured me that there will be a fix for this in their next firmware release.. (I have no date on when that will be)

You're probably getting quite caught up in Asterisk by now, so take a look at the documentation, see if the sections we've covered make more sense to you now...Well, that's all for now, next time we'll look at Call parking, and meetme extensions.

Parking Zone....

Call parking is a wonderful invention, the best way to explain how useful it can be is to use a real world example.

Imagine the scene, you're on your own in the office. Your desk is on the 2nd floor and the computer room is on the 4th. You are working late to fix a problem on one of the servers, but you need some help. You call another support team - they say they are busy but will call you back in 5 minutes.

When they call you back they ask you to fiddle with the server.

So, now you have a number of options

1. Say "hang on", run up stairs and do what they ask then run back and say "done that" only to hear them ask you to do something else.

2. Say "hang on, I'll transfer the call upstairs", you transfer the call to the computer room, run up stairs, pick up the phone in the computer room, run downstairs hang up, run upstairs and talk to the support team.

3. Say "hang on, I'll transfer the call upstairs", you park the call, hang up, run upstairs to the computer room and pick up the parked call.

Now, it's clear that option 3 is the best for us. Parking the call, basically involves transferring the call to a 'fictional' extension that will hold the call until we either pick it up, the caller hangs up, or the timeout period expires.

Call parking is a wonderful thing... embrace it with open arms

So How do we do this with Asterisk? Well, it's simple. In your Asterisk config directory is a file called parking.conf. Let's take a look at it:

```
# cd /etc/asterisk
```

Edit the file

```
# vi parking.conf
```

You should see something similar to below. Let's take a closer look at it:

```
[general]
```

```
parkext => 701                ; What ext. to dial to park
parkpos => 702-720            ; What extensions to park calls on
context => parkedcalls               ; Which context parked calls are in
parkingtime => 1355           ; Number of seconds a call can be parked for (defa
```

It should be self explanatory, but we'll go through it just to be sure you understand.

parkext => is the extension you will dial to park a call. You will probably note that your

parking.conf file has this number set at 700. I've changed mine to 701 because I was having an issue with Asterisk - although it would 'see' (looking at the console) I had tried to transfer to 700 it appeared not to believe that I had dialed it. This was essentially due to the 00 in the 700, changing it to 701 eliminates the problem completely.

parkpos=> is the range of numbers used to park calls in the extract above I get 19 parking extensions, enough for me but you might like to expand it if you have more users.

context=> is the context that you need to include to allow types of call to use call parking, this prevents external users parking YOU on your own system!

parkingtime => is the length of time in seconds to park the call. Once this time expires the original extension will be called again.

So how do you actually park a call? Well first of all, if we look back at the extensions we first created:

```
exten => 1,1,Dial(SIP/phone1,20,tr)
exten => 2,1,Dial(SIP/phone2,20,tr)
```

you'll notice a couple of things that I haven't yet talked about. After the '20' (the number of rings) we have a 't' and an 'r'. The 't' means that the called user can transfer the call, the 'r' just means tell the calling party that the extension is ringing, but only send audio when the call is answered. We can actually modify these 2 lines, changing the 'tr' to 'Ttr'

```
exten => 2,1,Dial(SIP/phone2,20,Ttr)
```

The 'T' allows the calling user to transfer the call as well. Be careful. you don't want an external user to be able to transfer your call around the system.

Note: This is only 'SAFE' here because the 2 extensions we set up are not accessible from the outside world. In our current setup, all calls are sent to extension 1000 - just because it's physically the same device doesn't mean anything. So, do NOT add a 'T' to your 1000 extension line unless you REALLY REALLY want someone from the outside to transfer you on your system.

I can't stress this enough, you don't want external callers messing with your setup.

Edit your extensions.conf file and include the parking context in your [sip] section, so add

```
include => parkedcalls
```

just under the include => fwd line, it should look something like this...

```
[sip]
include => fwd
include => parkedcalls
```

Make these changes, save the file and go to the Asterisk console and type:

```
reload
```

Now dial from extension 'phone1' to extension 'phone2'. Pick up both ends of the call.. Now on phone1 press the # key. You should hear a voice saying 'Transfer'. Dial the number you have in your parking.conf file that parkext =>points to. In my configuration that would be 701. If you don't dial the extension quickly enough the voice will say that it didn't recognize the extension number. If you dialed the extension number quickly enough then the voice will come back and give you an extension number, in the range specified in parkpos=> . Go back to phone2, if everything is working you should be hearing some music*, or a voice talking.

Since we've only set up 2 phones, go back to phone1 and dial the extension that you were given when you parked the call. Once you dial that extension you should be connected back to phone 2. If you dial the wrong extension, the voice will tell you there is no parked call at that extension an hang up. In all honesty unless you have multiple parked calls, it's more than likely going to be the first extension specified in your parkpos=> range.

* we haven't added any music to the system, so unless you have done so yourself you'll get the default music on hold file, a mildly comical sentence.

Meetme behind the bike sheds...

Meetme is even easier to set up, there's very little to do here. There are however some potential pitfalls, everything tells us that a Zaptel card is required to use the Meetme functions, I have such a card so I cannot confirm that the use of ztdummy will be enough (remember we modified the Makefile at the top of this document to use ztdummy - effectively faking the presence of a Zaptel card) - I would however be interested in results from people who have tried only using ztdummy..

The Meetme function of Asterisk is like a conference call, multiple users can call in and either listed to a conference (one way) or actively take part (two way). One was conferencing can be useful if you don't want other participants to be heard, perhaps when someone is giving a presentation.

We'll just set up a simple conference, a bit like a coffee room where everyone can talk and listen too. You should have got the hang of where the config files are, and how to get there and edit a file:

```
# cd /etc/asterisk
```

Edit the file

```
# vi meetme.conf
```

The file actually explains all you need to know, again we'll go through it to double check your understanding

```
[rooms]
;
; Usage is conf => confno
;
conf => 1234
```

Ok, it's simple really, for each conference room you want to set up add a line

```
conf => <room number>
```

where <room number> is a unique number add the following line to the file:

```
conf => 54321
```

Save the file, now edit your extensions.conf file again,

```
# vi extensions.conf
```

At the bottom of your [sip] section add a new extension:

```
exten => 5557,1,Meetme,54321
```

The format of the Meetme line is:

```
exten => <extension to dial>, Meetme,<room to enter>
```

So when <extension to dial> is called, the Meetme application takes you to room <room to enter>. Save the file, and go to the Asterisk console and type

```
reload
```

Now from phone1 dial, 5557. You should be told that you are the only one in the conference. Go to phone2 and also dial 5557. On phone1 you should hear a tone, signifying that someone has entered the room, phone1 and phone 2 should now be in a conference. If you hang up phone1 phone 2 will hear an exit tone signifying that someone has left the conference.

All these tones are fine for two or three people, but if you had 20 people coming and going it would get quite annoying. Thankfully we can turn this off. Edit your extensions.conf file again (you should still be in the /etc/asterisk directory)

```
# vi extensions.conf
```

find the line you just added,

```
exten => 5557,1,Meetme,54321
```

and change it adding |q - it should look like this:

```
exten => 5557,1,Meetme,54321|q
```

Save the file, and go to the Asterisk console and type

```
reload
```

Now call the conference number from both phones, note that the first phone gets no announcement that they are the only caller in the conference, and when the second phone enters no tone is played.

Configs! Configs? We don’t need no stinkin’ configs

Well of course you know you do. Asterisk comes with a large number of configuration files, enough to daunt all but the most experienced. This next section will deal in detail with the main configurations that you’ll need to know about.

The files covered in this section are:

musiconhold.conf	Music on hold configurations
cdr_mysql.conf	Call record logging using mySQL
manager.conf	Remote management configuration
adsi.conf	ADSI phone support

meetme.conf	Conference configurations
oss.conf	o
enum.conf	
mgcp.conf	
parking.conf	Call parking configurations
voicemail.conf	Voicemail configurations
agents.conf	Call agents
extensions.conf	Extension definitions
modem.conf	Modem configurations
phone.conf	phone config
vpb.conf	
alsa.conf	
festival.conf	
modules.conf	
privacy.conf	
iax.conf	
musiconhold.conf	
zapata.conf	
asterisk.conf	
indications.conf	
rpt.conf	
logger.conf	
sample.call	