- -4pc - -4pc

The Working Centre Linux Project: User Guide

- -4pc - -4pc

The Working Centre Linux Project: User Guide

- -4pc - -4pc

Table of Contents

- -4pc - -4pc

- -4pc - -4pc

Chapter 1. Basic Tasks

This chapter documents the most important information you will need to know in order to start using your computer: turning the computer on, running applications, accessing your floppy drive, and safely shutting down the computer. [WHAT ABOUT UNIX ESSENTIALS?]

If you have some computer experience, much of the information in this chapter will likely be familiar to you. You will probably want to skim through the chapter (or the table of contents!) and then read sections in more depth as the need arises. If you have some computer experience but have never used Linux before, you may find the sections on saving files to floppy [WHERE?] and shutting down your computer [WHERE?] relevant. [AND DIFFERENCES BETWEEN UNIX AND WINDOWS?]

Even if you are new to computers, you need not spend hours poring over this chapter. You may want to skim through the contents once to get an idea of how to use your computer, but do not worry too much about topics you do not understand fully. The most important topics to learn about are how to set up your computer [WHERE?], how to run applications [WHERE?], how to save files to floppies [WHERE?], and how to shut down your computer [WHERE?]. You will also need to know how to boot your computer [WHERE?] but you need not worry too much about the step by step explanations of the boot process.

Most of all, do not be afraid to explore. Although it is possible to mess up your computer by fiddling with the wrong things, it is very difficult. Many people are too afraid of their computer to play around, and that is a shame -- playing around is one of the best (and one of the only) ways that people learn.

Having said that, there are a few situations in which you do need to exercise caution. You need to be cautious when deleting files. You need to be cautious when changing the configuration of your computer. Finally, you need to be cautious when using the *root account* of your computer (if you don't know what the root account is, you are not using it). For more information about safe computing, see Chapter [WHERE?]

Before You Start

Before you start your machine make sure that you have connected the hardware together properly. Here is a checklist of the usual connections. For more information, see [WHERE?]

- Is a power cable plugged into your case?
- Is a power cable plugged into your monitor?
- Is your monitor cable plugged into to your case?
- Is your keyboard cable plugged into your case?
- Is your mouse plugged into your case?
- If you have a printer, is your printer cable connected to the case?
- If you have a modem (and an ISP) is a telephone cable plugged into your modem?
- If you have a network card and high speed Internet access, is a network cable plugged into your network card? (The other end of the cable might go into a cable modem or a router box.)

- -4pc - -4pc

Of these tasks, it is most important to make sure that your monitor and case have power, that your monitor cable is plugged into your case, and that your keyboard and mouse are plugged into the proper port.

Be careful: it is easy to mix up the PS/2 keyboard and mouse connections, and it is easy to plug in serial mice into the wrong place. Fortunately, it is easy to tell if you get things wrong: often your computer will beep and complain at you if you don't plug your keyboard in properly, and if your mouse is plugged into the wrong port, your mouse will just fail to work. [POINT TO REFERENCE]

Once you have your cables set up, you are ready to boot up the computer.

Booting Your Computer

Booting your computer is the process of powering it on and starting the operating system. This section will identify the steps in the boot process. Usually, you will not need to know too much about this process; you will power on your machine, and in a few minutes your computer will be ready to use. However, it is useful to know a little about the boot process in case your computer stops booting.

Powering Up Your Computer

To start your machine, you need to turn on the monitor and computer. I usually turn on the monitor first so that I don't forget. If you have a printer you can turn that on as well.

If all goes well, you will see an image on your monitor, and your computer will start the boot process. If you see no image, make sure the monitor cable is plugged in, and make sure that your computer and monitor are both plugged in and are turned on.

Usually your computer will beep once or twice on bootup. This is normal, and nothing to worry about. Repeated beeping combined with no display and/or error messages indicate that something might be wrong. Otherwise, your computer will proceed to perform a power-on self test.

POST: Power On Self Test

NOTE: these steps vary from computer to computer. Do not worry if your computer does not follow these steps exactly when it starts booting. So long as your computer boots into Linux, all is well.

When your computer first starts it will start counting its memory in the top left corner of the screen. [SCREENSHOT] You may see the keyboard lights flash, see the floppy and/or CDROM lights turn on, and hear the floppy and hard drives spin. Then your computer will beep (possibly a few times). This process is called the *POST*: power on self test.

On most machines, the POST is followed by one or two screens of information about your hardware: the sizes of your hard drives, the extra cards that are on your system, and so on. In a few seconds the GRUB menu should appear. [SCREENSHOT OF POST]

It is safe to power off your machine during this stage.

GRUB Screen

GRUB stands for "GRand Unified Bootloader". Its job is to start up the operating system on a computer. The GRUB screen looks something like this: [SCREENSHOT]

- -4pc - -4pc
After the POST screens you should see the GRUB screen. If you had multiple operating systems on your computer, you would be able to choose which operating system to boot at this step. Probably you will want to leave the GRUB screen alone. In a few seconds it will start booting Linux. [WHICH ENTRY?]

Your GRUB screen may include a *memtest86* entry. Usually you will not need this. It is used to test the integrity of your system memory.

It is safe to power off your computer at the GRUB screen.

The Linux Boot Process

After the GRUB screen completes Linux will load up. You will some messages like the following:

[SOMETHING SOMETHING SOMETHING] Uncompressing Linux.....

The "Uncompressing Linux....." line is your last opportunity to power down the computer safely. Once the Linux boot process starts, you should not power down the machine directly. Rather, you should wait for Linux to boot, and then shut down the computer from the desktop. [HOW? PROVIDE A REFERENCE]

After Linux has uncompressed itself a lot of mysterious text messages will scroll down your screen. These are called *boot messages*. You do not need to understand these messages, but they are very useful to have in diagnosing problems with your machine. For example, if someone's computer stops booting, that person will often be asked to copy down the last boot messages that appeared on screen.

In addition to boot messages, you may also see a picture of Tux, the cute Linux mascot, in the top left corner of your screen.

After Linux has finished booting, one of three things can happen:

1. The computer might proceed to log in a user and bring up the *IceWM desktop*. (This is the default.)
2. The computer might display a graphical login screen. [SCREENSHOT]
3. The computer might display a textual login prompt. [SCREENSHOT]

What happens after Linux finishes booting depends on how your machine was configured. In most cases, your machine comes with one user account, which is started automatically, and you don't need to log in explicitly.

If you requested several user accounts to be created on your machine (for example, if several different people are using your computer) then you should have been provided with a list of *user names* and *passwords* for these accounts. In this case you will use the user name and password for your account to log in and start the IceWM desktop, as described in the next section.

If the IceWM desktop does not start up and you have never received a user name and password for your machine, then something is wrong. (Note that the *root password* is not a user account password.)

Logging in at a prompt

NOTE: If your computer starts the IceWM desktop for you automatically (which should be the case for most people) then this section does not apply to you.

If your computer has been configured to use several user accounts, each user needs to login to the machine. Along with your computer, you should have received a list of user accounts created on your machine.

- -4pc - -4pc

Your login prompt might be graphical [SCREENSHOT] or text-based [SCREENSHOT]. In either case the login procedure is the same:

1. At the `login:` prompt, type your user name. (If using a graphical login you may need to move your mouse to the "login" text entry window.) Then press Enter
2. At the `password:` prompt, type your password. (You will not see your password on the screen -- this is a security feature.) Then press Enter

Note that user names and passwords in Linux are case sensitive. The password "MYPASSWORD" is different than the password "Mypassword", for instance.

At this point, the IceWM desktop should start. If your login prompt is text-based, then you might see a regular command prompt instead: [SCREENSHOT]

In this case you will have to enter one additional command to start the IceWM desktop. At the prompt, type:

```
startx
```

After entering this command and pressing Enter, graphical mode should start up and you should see the IceWM desktop.

If you go through these steps and you do not see the IceWM desktop, then something may be wrong. Otherwise you are done, and you can start using your computer.

Using the IceWM Desktop

Once Linux has booted and you have logged in, an environment similar to [SCREENSHOT] should appear. We call this environment the *IceWM desktop*. You will use the IceWM desktop to start applications, move windows around, and shut down your system.

IceWM is a program known as a *window manager*. The primary job of a window manager is to allow you to create, move, resize, minimize, maximize, and close program windows on your system. In addition to these basic functions, IceWM offers features including program menus, workspaces, network and CPU monitors, and a handy shutdown menu. Many of these features are labelled in [SCREENSHOT]. In the following sections we will explore each of these features in detail.

If you have used Windows, KDE, GNOME or another desktop environment before, you may notice that the IceWM desktop lacks some features you are used to. For example, IceWM does not allow you to drag and drop files from one application to another, and IceWM does not support desktop icons. On the other hand, IceWM offers features -- such as workspaces and easy network monitoring -- that are not readily available in other desktop environments. Although you might find using the IceWM desktop strange at first, give it a chance. I grew to appreciate the IceWM interface after using it for a while, and you might as well.

Tooltips

Tooltips are information boxes that are associated with certain objects on screen. A tooltip box pops up when you hover your mouse pointer over the associated object. *Hovering* your mouse pointer means letting the mouse pointer sit on top of a screen object for a few seconds, without clicking any mouse buttons or moving the mouse.

- -4pc - -4pc

For example, if you hover your mouse over the WCLP program menu at the bottom left corner of the IceWM desktop, a small yellow box displays the text "Favorite Applications". Clicking this box reveals the program menu, which (hopefully) contains your favorite applications. Similarly, when you hover your mouse over the clock in the right bottom corner of the screen, a tooltip pops up displaying the date and time in more detail. [SCREENSHOT]

Many screen objects in the IceWM desktop have tooltips associated with them. Exploring the tooltips in the IceWM desktop is one of the easiest and fastest ways to become familiar with what IceWM has to offer. Let your mouse roam, hovering over different parts of the screen and reading the tooltips that pop up. Similarly, when you are not sure what a button does, hover your mouse over the button and see whether a tooltip pops up.

Many application programs (such as Abiword and XFig) have tooltips as well, so you can explore these applications in the same way.

The Toolbar

The grey strip along the bottom of your screen is called the *toolbar*. The toolbar contains the program menus, the clock, and several other tools offered by the IceWM desktop.

Recent versions of IceWM have introduced a button which allows you to hide the toolbar. This button is on the leftmost edge of the toolbar, and it is marked with an angle-bracket arrow. [SCREENSHOT] clicking this button will slide the toolbar out of the way. Clicking it again will restore the toolbar.

Why might you want to hide the toolbar? By default, the toolbar appears in front of any other windows. Usually this is useful, but sometimes (for example, when playing games or using very large windows) this can obscure useful information. In this case, hiding the toolbar can allow you to access information or buttons at the bottom of the screen. [EXAMPLE? MOVING OR RESIZING IS BETTER.]

The Program Menu

When you click on the WCLP button at the bottom left corner of the screen, the IceWM program menu pops up. [SCREENSHOT] This menu is similar to the Apple Menu in Mac OS or the Start Menu in Windows. It provides a menu of programs and options you can activate by moving your mouse over the relevant item and clicking.

The program menu pops up when you click the WCLP button. It will stay open until you do one of the following things:

1. You select an application to run.
2. You click on the desktop background.
3. You click on the WCLP button again.

- -4pc - -4pc

The last two options provide a way for you to close the program menu if you open it accidentally.

The program menu is hierarchical, which means that menus can contain submenus. IceWM indicates that a menu entry contains a submenu by placing an arrow to the right of the menu entry. Sometimes clicking on the menu entry opens the submenu; the Programs entry works like this. Otherwise, clicking on the arrow itself will open the submenu. This is the case with the Logout submenu. [SCREENSHOT] IceWM separates the menu entry from the submenu arrow using a faint vertical line when you need to click on the arrow.

The program menu is made of several components, which we will describe in the next few sections.

Featured Programs

We have designated some of the applications on your computer as "featured" -- these are the applications we expect most people will use regularly. For easy access, we have plopped menu items for these applications right at the top of the program menu. These applications include:

- A word processor, which is probably Abiword, but may be the Writer application from OpenOffice.org.
- A web browser, which may be Dillo, Links, Opera or Firefox, depending on how your computer was configured.
- A *file manager*, which allows you to organize files and directories. Our file manager is named XFE.
- A text editor, which is probably Nedit. [DIFF BETWEEN TEXT EDITOR AND WORD PROCESSOR?]
- A printer tool, which allows you to monitor the status of any printers connected to your machine. Our printer tool is called Printtop

[WHAT ELSE? HELP? GAMES?]

Depending on how your computer was configured, you may have other featured applications in your program menu. To start these applications, click on their menu entries.

WCLP Apps

[TO BE WRITTEN]

The Debian Menu Hierarchy

Below the featured applications is a menu item labelled Programs. Clicking on this menu item opens a submenu with entries including Applications, Games, Help, WindowManagers, and XShells. Clicking any of these menu items opens further submenus containing application programs and yet more submenus. [SCREENSHOT]

The collection of menus in the Programs menu is called the *Debian menu hierarchy*. It contains links to many applications and programs installed on your computer. You should feel free to explore the Programs menu and its contents; the programs installed on your computer are there for your education, enjoyment and productivity. Be aware, however, that certain programs on the menu may not be very user-friendly, and others may not be relevant to your needs. [DISCUSS WHY THIS IS CALLED THE DEBIAN MENU HIERARCHY?]

There are a few entries in the Programs menu that you might find particularly interesting. These include:

- The Games menu, which contains many of the games installed on the system. You should stay away from this menu if you have work to get done on your computer; some of these games are fairly addictive.

- -4pc - -4pc

- The Help menu, which contains links to documents and applications to help you access the documentation installed on your system. For more information about using these programs, see [WHERE].

Themes

The Themes menu item allows you to change the look of the IceWM desktop to your tastes. A *theme* is a way to configure the colours, fonts, and positions of IceWM elements on the screen. If you wish, you may ignore this menu item entirely. On the other hand, selecting a new theme is an easy way to customize the look and feel of your desktop to your preferences. [SCREENSHOT]

To select a theme, select any of the entries in the Themes submenu. Some of the theme entries have submenus themeselves; these submenus indicate variations of the theme you can select. [SCREENSHOT]

The default theme shipped with your computer is called "WCLP". If you ever find yourself stuck with a theme you do not like, you can always select the WCLP theme to get back to where you started.

One note of caution: themes are allowed to relocate your window controls about each window. [JARGON? WHICH SECTION?] For example, a theme might move the close button on windows from the top right corner to the top left corner. [SCREENSHOT] If the meaning of a theme's buttons are not obvious at first glance, hover over the buttons in question to expose the tooltips, or experiment by opening "test windows" (that is, windows where you have saved any important data before playing) and then pressing the buttons.

Logout...

The Logout... menu has two parts. On the left is an entry labelled "Logout". In general you do not want to press this button. If you do (and you proceed to log out) then you may end up with a textual login prompt. [SCREENSHOT] If you find yourself in this screen you have three options. If your account has a password [SEE WHICH SECTION?] you can log in again. If you do not have a password but want to work on your computer some more, you can press ctrl-alt-delete to reboot your computer, then wait until the computer reboots and restarts the IceWM desktop. If you want to shut down your computer, press ctrl-alt-delete to start the reboot process, and then turn off your computer during the memory count [SECTION?], POST screen [SECTION], or GRUB screen [SECTION].

The arrow on the right of the Logout button is slightly more useful. It opens a submenu which contains the same menu options contained in the shutdown [REF?] menu. The [WHICH?] section documents the meanings of these options.

Show Desktop Button

To the right of the program menu is a mysterious, unlabelled button. Hovering your mouse over it reveals the text "Show Desktop". Clicking this button will *minimize* all of your open windows. Clicking the button again will restore your open windows. [SCREENSHOT]

As the IceWM desktop does not currently support *desktop icons*, this button is not all that useful. However, it can help you get organized if you have cluttered your desktop with too many open applications.

Quick Launch Buttons

To the right of the show desktop button are some *quick launch* buttons. These buttons are intended to be shortcuts to launch certain applications with one click of the mouse. Often included are buttons launching Abiword and a web browser. [SCREENSHOT]
[HOW DO YOU ADD YOUR OWN?]

Workspaces

- -4pc - -4pc

Workspaces help reduce screen clutter by providing four areas for you to place application windows. If you regularly find your screen cluttered up with a lot of overlapping windows, you might find that workspaces help you reduce your frustration.

Four buttons labelled 1, 2, 3, and 4 allow you to switch between the four workspaces. The IceWM desktop starts in workspace 1, and you are free to do all of your work in that workspace if you wish.

[SCREENSHOTS SCREENSHOTS SCREENSHOTS] Here is a case where workspaces might be useful. Say you are working on an essay in Abiword, and you decide that a diagram would help you explain your ideas better. You would like to start Xpaint to create the diagram, but you want to leave your essay open as a reference in creating the diagram.

One possibility is to open Xpaint directly. Then the Xpaint and Abiword windows will overlap, as shown in [SCREENSHOT]. Since Xpaint opens multiple windows when running, this can get frustrating.

Another possibility is to switch to workspace 2 and open XPaint there. Then you can create the diagram without the Abiword window getting in the way. When you want to refer to your essay, you simply switch workspaces again.
[ESSAY: "Mullets in Contemporary Society"]

Status Indicators

To the left of the clock you will see one, two, or three small black boxes. These are called *status indicators* and they are very useful for keeping an eye on what your computer is doing. If your computer is responding very slowly to your commands, the status indicators can give you an idea of what is happening. Hovering over the indicators reveals additional information, but often the information is fairly cryptic.

The rightmost box is the *CPU indicator*. It shows the activity of your CPU; roughly speaking, it indicates how hard your computer is "thinking". If the coloured bars on the CPU indicator are low or missing, your computer is mostly sitting idle. If the CPU indicator is solid with colour, the CPU is working very hard, and you can expect your computer to be slower and less responsive than usual. [SCREENSHOT]

The other two boxes indicate *Network status*. Depending on whether you have Internet access at home, they may or may not be present. The red [IS IT ACTUALLY RED?] portion on the bottom indicates download speed -- how much information is going into your computer. The yellow portion on the top indicates upload speed -- how much information is going out of your computer. When you download large files, you can expect this indicator to be mostly red. [SCREENSHOT] If the box in question is black then you are neither uploading nor downloading anything.

If you have high-speed Internet access, the middle box will indicate the activity of your Internet connection. The middle box is called the *ethernet indicator*. If you have dial-up Internet access, the rightmost box will indicate Internet activity. The rightmost box is called the *PPP indicator*. If you have no Internet access at home, the leftmost two indicators may not be present.

Unfortunately, there is no indicator to measure hard drive usage. However, if you hear a lot of hard drive activity then you should expect your computer’s performance to be sluggish.

One thing that might be happening is that your computer is *swapping*, which means it is copying a lot of information from your computer’s memory (the *RAM*) to the hard drive or vice versa. Computers will often swap a lot when you have many applications open, and switch between the applications frequently. To reduce swapping, you might be able to run fewer applications at a time.

Another cause of high hard drive activity and sluggish performance is when your computer performs certain regularly-scheduled maintenance operations. If your hard drive churns for a while even when it is idle or you are only running one or two applications, this may be the reason (and it is nothing to worry about). [DO WE SCHEDULE HEAVY I/O JOBS INFREQUENTLY?]
[THIS INFORMATION ABOUT SWAPPING DOES NOT BELONG HERE]

- -4pc - -4pc

The Clock

Near the bottom right corner of your screen you will see a clock. Hovering over the clock will reveal the date. [SCREENSHOT]

Unlike in Windows, clicking on the clock will not bring up a screen that allows you to modify the time and date. In fact, fixing an incorrect computer clock is surprisingly involved. If your system clock is incorrect and you would like to fix it, please see [WHERE?]

Working With Windows

When you start Abiword or any other graphical program, one or more *windows* appears on the screen. When you do so, IceWM automatically draws borders and buttons around the window. These controls allow you to close windows, resize them, hide them, maximize them, move them between workspaces, and more.

[SCREENSHOT] shows a typical program window and its components. In the next few sections we will describe how to perform basic window operations.

Window Focus

Say that you have two windows open on your screen: one Abiword window and one web browser window. [SCREENSHOT] Now you type at your keyboard. Which application receives your keystrokes?

The answer depends on which window has *focus* -- which window is "active" at the time. Only one window can be active at any one time. IceWM indicates that a window is active by colouring in the *titlebar*. [SCREENSHOT] To make a window active, click its titlebar.

Closing, Minimizing, and Maximizing Windows

Reading from left to right, the three buttons at top righthand side of a window's titlebar respectively allow you to *minimize*, *maximize*, and *close* that window. Each of these buttons is activated by a single mouse click.

Minimizing a window means hiding it from view. You can restore the window by clicking on its entry on the toolbar. [SCREENSHOT]

Maximizing a window will increase its size so that it fills the entire screen. When you maximize a window the "maximize" button changes to a *restore* button; clicking this button will restore the window to its original size. [2 SCREENSHOTS]

Closing a window will eliminate that window. If you are closing an application window, be sure to save your work before clicking the close button. (Many applications will prompt you to save your work, but not all will.) Closing a windows often ends the application that was running in the window. [SCREENSHOT]

Moving and Resizing Windows

Resizing windows or moving them on the screen involves first clicking on a particular part of the window, then *dragging* the mouse while holding down the mouse button. When you have clicked on the window for a move or resize operation, the mouse cursor will change. [SCREENSHOT?]

The operation you perform on the window depends on what part of the window you click and drag. If you click on the window's titlebar, you can move the window. If you click on one of the bottom corners of the window, you can stretch or shrink the window horizontally or vertically. If you click on the left or right border of the window, you

- -4pc - -4pc
can resize the window horizontally, and if you click on the bottom border of the window you can resize it vertically. [SCREENSHOT WITH LABELS]

Sometimes naughty applications create windows that are bigger than your screen size. [SCREENSHOT] These windows can hide window controls that you need offscreen. IceWM provides a way to force window movement: hold down the Alt key, then click and drag the window to reposition it. [SCREENSHOT]
[SECTION ON THE MENU BUTTON?]

Shutting Down Your Computer Safely

Just as Linux has a fairly involved startup process [LINK] it needs to go through several steps before you turn off the power to your computer. During these shutdown steps Linux stops programs and saves data. For this reason, you should never power down your computer directly (for example, while logged into the IceWM desktop). Instead, you should let Linux go through its shutdown sequence before turning off power to your computer.

Linux is a fairly robust operating system. It can usually withstand sudden losses of power -- the next time you boot up your computer, Linux will notice that it has been shut down unexepectedly and try to recover its data. However, it is possible that you could lose data or corrupt files, and in the worst case your computer might refuse to boot again without human intervention. [REF TO TROUBLESHOOTING]

The first step in shutting down your computer is to save your work and close the applications you have open. Don’t count on your application programs to save your data automatically for you; some applications will, and others will not.

After saving your work press ctrl-alt-delete . That is, press the ctrl key, then while holding that key down press alt, then while holding those two keys down press delete. Pressing this sequence should bring up the following IceWM menu: [SCREENSHOT][REF TO "How to use this manual"?]

To shut down your machine, click on Shutdown. The following (not very helpful) dialog box appears: [SCREENSHOT]

Although the dialog box talks about "Logout" and not "Shutdown", your computer will start the shutdown process if you click Ok. Not surprisingly, clicking Cancel will cancel the process.

After clicking Ok, the computer may beep. Then you will leave the graphical desktop and go back into text-only mode. Messages will scroll by on the screen for a while. Eventually you will see the words "Power Down" at the bottom of your screen. [SCREENSHOTS] When you see this message it is safe to turn off your computer.

Other Ways to Shut Down Your Computer

Under normal circumstances, the easiest way to shutdown your computer is from the IceWM desktop. However, you might click the wrong buttons or hit the wrong keystrokes and find yourself in some other environment. For example, if you accidentally click the
<mi>Logout</mi>
entry in the program menu, you will find yourself on a screen that looks like this: [SCREENSHOT]

If you find yourself on this screen and you want to turn off your computer, press ctrl-alt-delete . Linux will go through the shutdown process, and then reboot your computer. When your computer starts rebooting (during the memory test, the POST screen, or the GRUB menu) you may safely power off your computer. [REF TO BOOTUP CHAPTER].

Another way to shutdown your machine is to use the command line. To do this, open an X-Terminal-Emulator or Xterm window. At this window, type

- -4pc - -4pc

```
/sbin/shutdown -h now
```

The computer will print the following message:

```
The system is going down for system halt NOW!
```

and then run through the shutdown process. Again, you should wait until you see the `Power Down` message before turning off your computer.

A Few Things You Need to Know About Linux

Linux is a large, complicated collection of software; people spend years learning about its features and nuances. Thankfully, you do not need to know a lot about Linux in order to put your computer to good use. In this section, we provide you with the bare essentials: the information you need when getting started. As you need to accomplish particular tasks you may need more information, some of which we provide in [WHERE].

If you have used other operating systems in the past but are new to Linux, you may find this section particularly useful. It outlines some aspects of Linux that people find particularly confusing when migrating to Linux from another operating system.

User and Administrative Accounts

Linux is designed to be a multiuser system. In theory, dozens (or hundreds!) of people can be logged into a Linux system at the same time, all running programs simultaneously. To support the many possible user, Linux uses the ideas of *accounts*. Accounts are identities: they identify who is on the system and what they can do.

Even though only one person can use your computer at a time, your computer contains many different accounts. A few of these accounts (or maybe just one) are *user accounts* (also known as *regular accounts*). User accounts are used by people to log into the computer. If you do not need to type in a password to get into the IceWM desktop, then your computer likely has one user account (probably called `linuxuser`). If you have to type a login name and password to get into the IceWM desktop, then your user account is identified by your login name.

If you are logged into the IceWM desktop, you can use the File Manager to discover the login name you are using:

1. Start the file manager (click on the program menu, then click on File Manager item).
2. Read the entry in the location bar. [SCREENSHOT] It should be of the form `/home/`*`something`* . The *`something`* is your login name.

- -4pc - -4pc

It is possible for all the users of a Linux system to have separate user accounts. However, most WCLP computers come with a single user account, and all the people using the computer share this account. There are advantages and disadvantages to each approach. Having a single user account means that you do not need to memorize the login name and password for your account. Every user account gets some private storage space, so having separate user accounts mean that users can customize their accounts and store data files independently of each other.

If you have a single account but want every user to have a separate account then it is possible to add users to your system. See [WHERE] for instructions on how to do this.

The Root Account

When reading about Linux, you will likely see the terms "root user" or "superuser" used frequently. These refer to the *administrative account* in Linux. The *root user* in Linux is a special account that can be used to administrate the system. The root account has total control over the system; a person using the root account can see or change any file on the system, and can run any program. However, with great power comes great responsibility: unlike regular user accounts, the root user is capable of deleting every file on your computer, configuring the computer so that it is unusable, or other bad things.

The root account is too powerful for daily use. You don’t want to use it to compose Abiword documents or to surf the Internet. Use your regular user account for these tasks instead.

You should have received a *root password* along with your computer. If you find yourself needing to do administrative tasks on your computer, you will need this password, so be sure to keep it in a safe place. In general, you want to keep this password (and all passwords) private.

User Tasks and Administrative Tasks

Linux makes a fairly sharp distinction between *user tasks* and *administrative tasks*. A user task can be done by a user without special privileges. Most administrative tasks can only be done by the root user. Here are some examples of user tasks:

- Writing a letter in Abiword
- Surfing the Internet.
- Spending hours playing gtan.
- Changing the IceWM theme you use.

Here are some administrative tasks in Linux:

- Adding new user accounts to the system.
- Installing or removing software on the system.
- Changing the date and time.
- Shutting down or rebooting the system.
- Configuring a new hard drive for use on the system.

- -4pc - -4pc

Some users find this breakdown of tasks confusing. A good rule of thumb is to pretend that a hundred users are all logged into your system. User tasks affect only the user in question. For example, if you change your IceWM theme, the themes chosen by the other 99 users will not be affected. On the other hand, administrative tasks have the potential to affect many users. If you change the time on your computer, the time would change for the other 99 users as well.

Note that even though shutting down the system is technically an administrative task (and for good reason!), your computer is configured so that you can shut down the system without needing to use the root account.

All of the other administrative tasks require the use of the root account. You can learn about how to do some of these administrative tasks in section [WHERE?]. Note that you probably do not want to play around with administrative tasks until you have become comfortable with using your computer as a regular user.

Files and Directories

Files, directories, and symbolic links are three of the primary ways Linux organizes information on your computer. Files contain most of the data you care about -- the data you (and the computer) uses on a daily basis. Directories are used to organize files into a coherent structure. As always, the situation is more complicated than this: Linux offers other tools to represent and organize information, such as *symbolic links*, *block files*, *hard links*, and *sockets*. Fortunately, you are not likely to run into any of these concepts when starting out.

Files

Roughly speaking, a *file* is some data grouped into a single unit. Here are some examples of files:

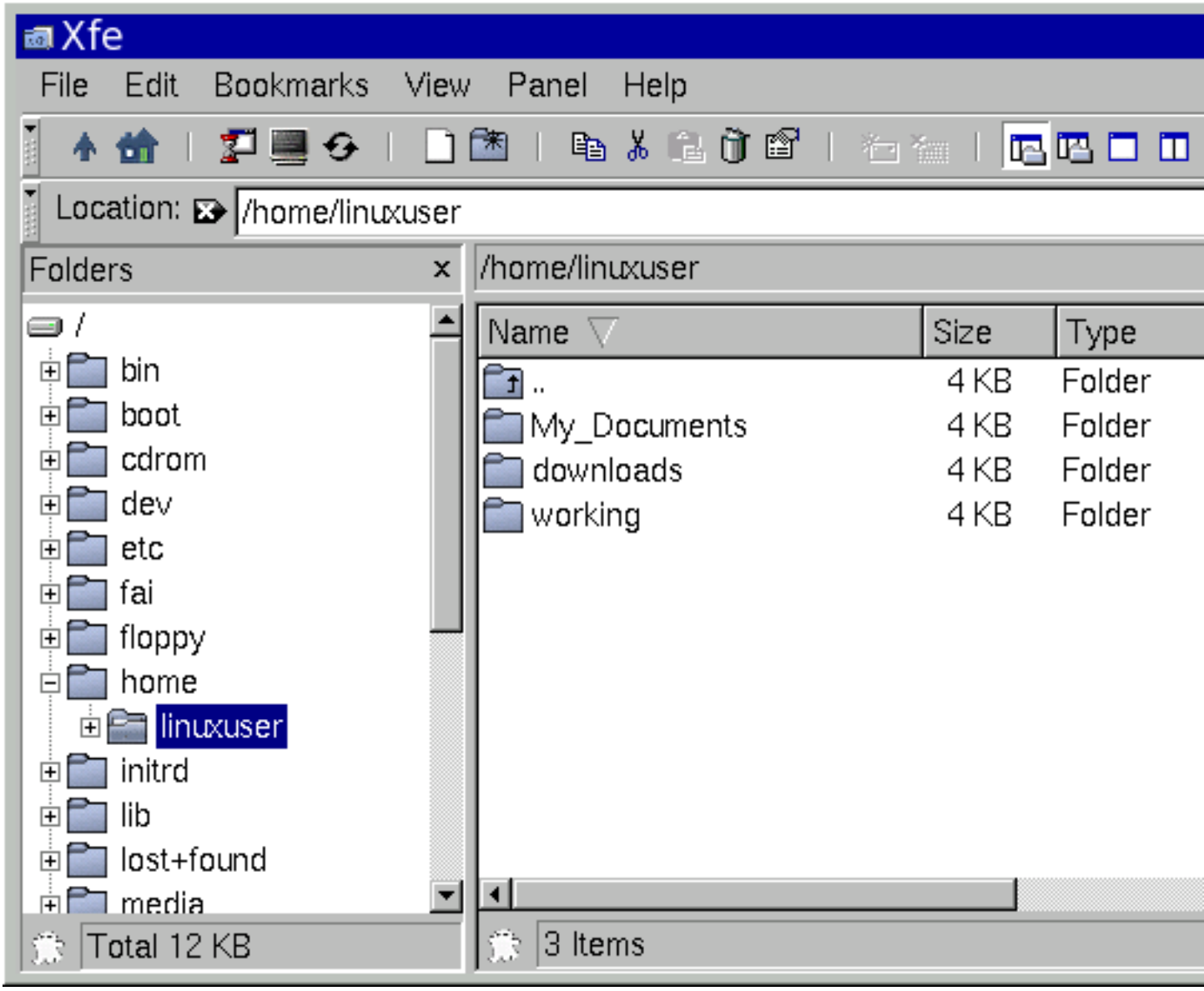
- A resume stored as an Abiword document.
- An HTML file you retrieve from the World Wide Web.
- An OGG music file by your favourite band.
- A JPEG picture of a llama.
- The `.bashrc` configuration file.
- The `xclock` executable program.

Some of these files are data files that you create or download. Other files come pre-installed on your system. Furthermore, Linux creates other files (such as temporary files and log files) as you use your computer. People often think of files as similar to the files you might store in a filing cabinet (in fact, that is where the word "file" comes from.) If you can store some data in a filing cabinet, you can store it in your computer as a computer file.

Directories

The files in your machine are organized into *directories*. A directory contains files and possibly other directories. Many people (and operating systems) think of directores as being similar to file folders: you can store documents in a file folder, and you can nest file folders into each other. The file manager XFE follows this convention: it identifies directories with a folder icon. [SCREENSHOT]

- -4pc - -4pc



In the `/home/linuxuser` directory, you can see the subdirectories `My_Documents`, `working` and `downloads` represented as folders.

Linux organizes directories in the following way: there is a topmost directory called the *root directory*, which is indicated by `/` . The root directory contains other directories, which can contain further directories, and so on. For example, the directory: `/usr/share/doc/debian/FAQ/` refers to a directory named "FAQ", which is contained in a directory named "debian", which is contained in a directory named "doc", which is contained in a directory named "share", which is contained in a directory named "usr", which is contained in the root directory.

Similarly, the file: `/usr/share/doc/debian/FAQ/index.html` refers to a file called "index.html" in the "FAQ" directory. This particular file is an HTML file that you could read with your web browser.

- -4pc - -4pc
The collection of directories on your computer is called the *directory tree*. Every file on your computer occurs somewhere in this tree.

Linux likes to treat everything in terms of files and directories. Unlike Windows or Mac OS, Linux does not really have a concept of "drives" (for example, "c:" or "a:"). Rather, it thinks of hard drives, floppy drives, CDROM drives, and partitions as directories. This is relevant when saving to and retrieving information from floppy disks: instead of looking in "a:" for your files, you look in the directory `/media/floppy` .

You should know about the following directories:

- `/home` which contains *home directories* for users on the system. Each user account gets a directory in the `/home` directory in which the account's owner can store his or her personal files. Your home directory's name is the same as your username: if your username is `linuxuser`, then your home directory is `/home/linuxuser`
- `/media` where you can find peripherals. In particular, you access your floppy drive by visiting `/media/floppy`, and access the CDROM by visiting `/media/cdrom`.
- `/tmp` which is used to store temporary files. The `/tmp` directory is emptied every time your reboot your computer.
- `/` which is the topmost directory.

There are many other standard directories on your Linux system. You can find out more by reading [WHERE?] or searching for "Filesystem Hierarchy Standard" on the Internet.

Moving Files To and From Floppy Disks

Although by today's standards floppy disks do not store a lot of information, they come in handy when you want to transfer files to other computers (for example, to print the files) or to make temporary backups of your files. You can use the XFE File Manager to copy files to and from floppy disks. Here's how:

1. Insert your floppy disk into the floppy disk drive of the computer.
2. Start the file manager by clicking on the program menu, then selecting File Manager [SCREENSHOT]
3. The file manager supports different *views*. Probably the easiest view to use when transferring files is the "Tree and two panels view." To activate this view, click the button shown in [SCREENSHOT], or press Ctrl-F2.
4. Your file manager window should now be divided into three parts. The leftmost part shows the directories and files on the computer. It is called the *directory tree*. The middle and right parts of the file manager are called *panel*. Clicking inside one of the panels will make it active. The top bar of the other panel will be greyed out. [SCREENSHOT]

   We will navigate to the floppy drive in the middle panel, and navigate to your home directory in the right panel. [REF: home directory]
5. To navigate to your home directory, click inside the right panel and click the "Home" button. [SCREENSHOT]

- -4pc - -4pc

6. To navigate to the floppy drive, click the middle panel to activate it, then use the tree view to navigate to the folder `media`. You should see the floppy drive light up when you click this folder. Then click the `floppy` "directory", which should have a hard drive icon beside it. [SCREENSHOT] The middle panel should show the contents of your floppy.

7. Next, select the files you want to move. You can move one file at a time, or you can select multiple files by holding down the ctrl key and clicking your files. Once you have selected your files, let go of the ctrl key.

8. Now comes the tricky part: *Right-click* the mouse, and select the Copy option. (You can also use the drop down menus for this: select Edit and then Copy [CHECK]

   Do not click on the Move or Cut options, and do not drag the files to the other panel. Any of these actions will cause the file manager to happily *move* your files instead of *copying* them -- the file manager is too dumb to understand that you are trying to transfer files between your hard drive and floppy disk. This is almost certainly not what you want: if you move a file from your hard drive to your floppy, you will no longer have that file on your hard drive. If you lose that floppy -- or if the floppy gets damaged -- then you will lose your file.

9. Next, click on the other panel to activate it. Right-click again, and select the Paste option at the very bottom of the pop-up menu. A dialog box will pop up asking whether you really want to copy the file. Click Accept to begin the transfer. [SCREENSHOT]

10. After copying your files, activate the middle panel and click on it. Then click the home button to move back to your home directory. Finally, go to the tree view and click on the `media` folder to close it. These actions are not necessary, but they will stop the floppy light from going on every few seconds.

11. Exit the file manager. Another dialog box will ask you whether you really want to exit. Click Okay [CHECK]

Command-line Linux gurus will shudder at this long list of instructions. The reality is that you can also copy files from the command line, using the **cp** command. You are welcome to use this command if you learn about it; the only thing to remember is that the floppy directory is `/media/floppy` .
[NEED: how do you format a floppy?]

- -4pc - -4pc

Chapter 2. Running Applications

Your computer comes with several *applications* installed. The following sections will document some basic information about some of these applications, so that you can start using the applications quickly. We will focus on the basics: what the application is for, how to start and exit it, the basic layout of the program, how to save and retrieve data, and confusing program quirks.

In general, we will not discuss advanced usage or configuration of the application. Where possible, we will point you to further resources so that you can learn more. In addition to reading documentation, don't be afraid to play: exploring the program menus and trying stuff out is one of the best ways to learn about your software.

The set of applications installed on your computer might be different than the applications that we discuss below. The number and selection of programs installed on your computer depends on the size of your hard drive, and how your computer was configured when the software was installed. If you have a smaller (500MB - 1GB) hard drive on your computer, you can expect to have fewer applications, and if your computer's hard drive is large (over 3 GB) then you may find extra programs installed.

Productivity Software

Productivity software refers to applications that you might use in an office or at school. It includes word processors and file managers, but does not include Internet applications (which have their own section).

The AbiWord Word Processor

AbiWord is a *word processor*. With it, you can enter, edit, and typeset documents such as letters, essays, and resumes. Although it is not the most sophisticated word processor in the world, it supports many of the features people expect in a modern word processor -- spell checking, tables, image insertion, and some fancy formatting. If you are starting out with computers you will probably find AbiWord a good way to develop your word processing skills, practice your typing, and produce professional-looking documents.

You can start AbiWord in three ways: by clicking on the AbiWord button on the toolbar [SCREENSHOT], by selecting the AbiWord Word Processor menu item from the program menu [SCREENSHOT], or by typing **abiword** in a command-line window. A splash screen should come up, followed by the main AbiWord screen. [SCREENSHOT]

To exit AbiWord, select the File->Quit menu item. If you have unsaved work, AbiWord will prompt you to save your work or explicitly discard it. [SCREENSHOT]

Saving Files

AbiWord provides two different ways to save your work. Choosing File->Save As will allow you to give a name to your document. You can use this to save different versions of your document with different names. If your document already has a name, File->Save will save your progress. Both of these options have buttons on the main menu bar for quick access. [SCREENSHOT]

When you select the Save As option, a dialog box will pop up and you will be prompted for a name and format for your file. You can use the Folders window to find a directory for your document. The Files window shows files in the current directory. Enter a name for the file in the Selection box. [SCREENSHOT] shows an example of this dialog box. In this example, we are saving the document in `/home/linuxuser/My_Documents` and naming the document `demonstration.rtf` .

- -4pc - -4pc
The "rtf" at the end of `demonstration.rtf` stands for *Rich Text Format*. By default, we have configured AbiWord to save files in this format. If we named the document `demo` instead, then AbiWord will add the ".rtf" at the end, so the file will have the name `demo.rtf` .

Rich text format is well-supported by AbiWord and many other word processors. This makes it easy for you to open your documents in other word processors for printing or viewing by others. Although AbiWord is capable of saving to other formats, we recommend using Rich Text Format unless you have a good reason not to. In particular, the "Microsoft Word" save format does not work very well, and the "AbiWord" save format is not recognised by many other word processors.

Recovering From Crashes

Your best defence against computer crashes is to save your work frequently. The more frequently you save the less work you will lose in the event of a crash. Having said that, if AbiWord crashes before you have saved your work there may be some hope. Every five minutes AbiWord creates backup files of your working documents. With a bit of luck, you can use these backups to retrieve some of your lost work.

AbiWord appends a `.bak~` to the names of the files it backs up. For example, the backup file for `demo.rtf` will be named `demo.rtf.bak~`. These backup files are placed in the same directory as the original file.

If your document does not have a name, AbiWord makes backup files called `Untitled1.bak~`, `Untitled2.bak~` and so on. It leaves these files in your home directory.

The next time you start AbiWord, you can open these backup files as you would any other document. [SCREENSHOT] If you find that this file contains lost work, use the Save As button to save the document with a new name. You can then copy and paste the changes to your original document.

You can configure AbiWord to autosave your work more frequently by selecting the Tools->Preferences menu, then selecting the Preference Schemes tab. [SCREENSHOT] Just change the Auto save current file every 5 minutes entry to some other number.

AbiWord Help Resources

AbiWord comes with informative tooltips. Hover your mouse over the different buttons to see what they do. This is one of the best ways to explore AbiWord's features.

AbiWord also comes with help files that describe its functions in some detail. You can access this help from the Help->Help Contents menu entry. The Help contains other useful entries as well. However, note that the Search for Help, Check Version Report a Bug, and About GNU Free Software entries of the Help menu require Internet access to work correctly.

Depending on the size of your hard drive, you may also have some AbiWord tutorials installed. Some of these tutorials are in HTML format, and some in AbiWord's native file format. You can access the HTML tutorials from the WCLP Help entry of the IceWM program menu. You can access the AbiWord formatted tutorials in the `/usr/share/doc/abiword-doc/` directory. [SCREENSHOT] These tutorials vary in quality; some of them describe older versions of the program, and so some of the information may be out of date.

The XFE File Manager

XFE is a graphical file manager. It allows you to organize files into directory, rename files, copy, move, and delete files, and navigate through the directory tree.

- -4pc - -4pc
Before reading this section, you may want to familiarize yourself with the "Files and Directories" and "Moving Files to and from Floppy Disks" sections of the "Basics" chapter. [LINK] These sections introduce concepts we will use in this section.

To start XFE, select the File Manager entry from the IceWM program menu. [SCREENSHOT?] You can exit XFE by selecting File->Quit

XFE starts by displaying the contents of your *home directory*. [LABELLED SCREENSHOT] The Folders panel on the left shows the Linux *directory tree*, which you can navigate by clicking folders. The right panel shows the contents of the current selected directory, and the Location bar allows you to change the current directory by typing a new location manually. Above the location bar is a toolbar which provides one-click access to some common functions. You can discover the purposes of these buttons by hovering your mouse over them. [SCREENSHOTS]

Selecting Files

Most operations in XFE deal with sets of files. You select a file by first navigating to its directory, and then single-clicking the file. You can also select multiple files by holding the Ctrl key and clicking each file you want to include. Selecting multiple files comes in handy when moving files, copying files and creating archives. [SCREENSHOT]

Once you have selected the files you wish to manipulate, let go of the Ctrl key and right-click the mouse. This brings up a menu that lists the operations you can perform. [SCREENSHOT]

Creating Directories

Directories help keep your data files organized by project or purpose. For example, the `linuxuser` account in [SCREENSHOT] has three subdirectories in its home directory: one called `My_Documents` for documents and personal data files, one called `downloads` for information downloaded from the Internet, and one called `working` for no good reason.

To create a directory, first navigate to the location where the directory is to be created. Then right-click and select New Folder. [SCREENSHOT] A dialog box pops up with your current location (which is known as the *path*). [SCREENSHOT] Add the name of your directory to the end of the path. In the example, `/home/linuxuser/My_Documents` is the path and `exampledir` is the name of our new directory. [SCREENSHOT] Click Accept to finish the operation.

Copying and Moving Files

The difference between *copying* and *moving* a file is small but important. When you copy a file in Linux, you two distinct instances of that file -- one in the original location, and one in the new location. When you move a file, the file exists only in the new location. Often, you copy files when making backups or when transferring your data to a different medium such as a floppy disk. You move files to reorganize them.

Copying and moving files is best done using the "tree and two panels" view of XFE. To activate this view, press Ctrl-F2 or click the following toolbar button [SCREENSHOT].

To carry out the file copy or move, use one panel (we'll say the right panel) to navigate to the original file location, and the left panel to navigate to the destination. Select the file (or files) you wish to copy or move, then right-click and select the operation you want. Next, click in the destination panel and select Paste. [SCREENSHOT] A dialog box will pop up asking for confirmation; click Accept to do the operation.

If there are already files in the destination directory whose names match the files you are copying or moving, you will get a warning dialog. [SCREENSHOT] If you choose Yes or Yes to All then you will *overwrite* the files in the destination directory. Sometimes this is safe and sometimes this is not -- if you are unsure about whether you should overwrite a particular file, you should select Skip and verify that it is safe for you to overwrite the file in question.

- -4pc - -4pc

Renaming a File

XFE provides several ways to rename a file. You can right-click a file and select the Rename option, but the associated dialog box is confusing. Instead, right-click the file and select the the Properties option. [SCREENSHOT] A dialog box with three tabs pops up. You can change the name of the file by altering the Name field of the General tab. [SCREENSHOT]

Deleting Files

The steps in deleting a file are similar to those of moving or copying a file: select the file to delete, right-click, and select Delete.

Beware: the "recycle bin" icon in the deletion dialog window is misleading. [SCREENSHOT] There is no "recycle bin". For all practical purposes, once you delete your files they are gone for good.

Using Archives

Archive files serve two purposes: they collect related data files together and they compress data so it takes up less disk space. Archive files have many uses. Here are some examples:

- Archive files are often used to transfer data. Say you had to transfer 37 little files by floppy disk to another computer. It is a good idea to collect these 37 small files into one bigger archive file, and then transfer the archive than to transfer the files individually. So long as you made the archive file correctly, you would not risk forgetting a file. In most cases, you would also save space by using the compressed archive file.
- True to their name, archive files are used to archive data. If you had a lot of e-mail you wanted to keep for reference, you could store the e-mail in an archive file, and then store the archive file on your hard drive. Because the archive file is compressed, you save space over storing the e-mail in its original format.

There are three archive formats that are widely used in the Linux world. Each format has advantages and disadvantages, and tends to be used in different situations:

1. The *zip* format has been around since the DOS days, and it is widely supported. Zip archives are often called *zipfiles*, and they usually end with a `.zip` extension.

   Windows XP now supports zip archives natively, as do most unzip programs. For this reason, zip archives may be your best choice if you are transferring archives to an unknown computer running an unknown operating system. In particular, the zip format is one of the most widely-used formats in the Windows world.

   Unfortunately not every computer has zip archive support, and if you want to guarantee compatibility you may need to avoid archives entirely.

   The big disadvantage of zip files is that they do not compress data as well as the other formats. In the Linux and UNIX worlds, zipfiles are less widely supported than tarfiles, so if you know you are transferring files to a machine running Linux or UNIX, you probably want to use a tar format.

- -4pc - -4pc

2. The *gzipped tar* format is made up of two parts. The "tar" part refers to the archiving program, which is called **tar** and is an acronym for "tape archive". The "gzip" part refers to the compression scheme. These archives are often called *tarballs*, and the files themselves either end with `.tar.gz` or `.tgz` extensions.

   Gzipped tar files are the standard archive format for distributing files in the UNIX and Linux worlds. In the Windows world, the popular archive program WinZip supports gzipped tar files, and many other archivers do as well. Windows XP does not support the format natively, however.

   Gzipped tar files are quick to create, and they offer significantly better compression over zipfiles. Although **bzip2** offers better compression, gzipped tarballs are a useful format both for distributing files and for long-term storage.

3. The *bzipped tar* format also uses the **tar** format to collect files together, but the data is compressed using the "bzip2" file compression scheme. These archives generally end with the `.tar.bz2` extension, although other extensions (such as `.tbz2`) are also supported.

   The bzip2 compression scheme has been around for several years now, but it is much newer than either the gzip or the zip formats, and as such it is as not widely supported as either of these two formats. [DOES WINZIP SUPPORT bz2?] For this reason bzipped tarballs are not most people's first choice when transferring files (although it is becoming more popular).

   Bzipped tar files are good for long-term data storage. The compression tends to be better than gzipped tar files. This can result in significant disk savings for large archives that are not accessed frequently. The biggest disadvantage of the bzip2 scheme is that the compression process is fairly slow. (Uncompression is fast, however.) For casual, short-term storage, the gzip standard is faster and more widely supported. However, on your own machine bzipped tar archives make a lot of sense as an archival format.

The amount of compression you get with an archive depends on the type of files you are archiving. Some file types -- especially multimedia files such as PNGs, JPEGs, MP3s and MPEGs -- are already compressed, and thus will not compress any further when put into a compressed archive. Most other data files will compress a fair amount.

XFE is capable of compressing and uncompressing archives in any of these formats. It determines the type of archive to create or uncompress based on the file extension. [SCREENSHOT] shows the directory `lxsplit-0.1.1` directory archived in the three formats, and an archive called `axxrom-default-1.4.tar.gz` in `.tar.gz` format.

Extracting Archives

To extract an archive in XFE first select the archive file, then right click and select Extract To [SCREENSHOT]. A directory tree will pop up. Select the directory to which you want to extract the archive, and click OK. [SCREENSHOT]

If you are unsure of the contents of the archive, you might want to extract the archive to `/tmp`, or to create a temporary directory into which you extract the archive. Zipfiles in particular have a nasty habit of not creating their own subdirectories, which leads to a mess of files dumped in your destination directory. This can be dangerous: if you have a file in your destination directory whose name matches some file in the archive, your file will be overwritten.

After you have selected an appropriate destination directory, XFE will list the files it extracted [SCREENSHOT] You can then traverse to your destination directory and use the files in the archive. [SCREENSHOT]

Creating Archives

Before creating an archive, you should first organize the files that will go into the archive. Copy or move the files so they all sit in a single directory. The idea is to archive the directory, which will make your life easier when you choose to open the archive again.

- -4pc - -4pc

Once you have created the archival directory, right-click the directory and select Add to Archive. [SCREENSHOT] A (rather confusing) dialog box will pop up, asking you for an archive name. This dialog box wants you to type both the full location of the archive file -- both the *path* and the filename. If you don't have a particular location in mind, you can create the archive in `/tmp` and move it from there once it has been created. (Note that you cannot use `/tmp` for long-term storage -- it is cleared every time you reboot your computer.)

The name of the file is important -- XFE chooses the archive based on the file extension.

- To create a zip archive, append `.zip` as the extension.
- To create a gzipped tar archive, append `.tar.gz` as the extension.
- To create a bzipped tar archive, append `.tar.bz2` as the extension.

In [SCREENSHOT] we create an archive of the `/tmp/axxrom-1.4` directory in the `/home/linuxuser/downloads` directory named `axxrom.tar.bz2` . Since the extension is `.tar.bz2`, XFE will create a bzipped tar archive. [SCREENSHOT]

After selecting the filename, XFE lists the files it has archived. The archive should then live in your destination directory.

You may want to test the success of the archiving process by extracting your new archive to some new directory (perhaps a subdirectory in `/tmp` . Verify that the files you wanted archived were actually archived correctly. If you were archiving your files to save disk space, you can then delete the original directory.

Launching Programs

XFE is capable of associating files with applications. For example, double-clicking a document in Rich Text Format will launch AbiWord. [SCREENSHOT]

In some cases (such as with picture and music files) you don't want to edit the file -- you just want to view it. Sometimes right-clicking and selecting View will open an application that will allow you to see the files without being able to change them. [SCREENSHOT AND EXAMPLE]

You can view the file associations for a particular file by right-clicking the file, selecting Properties and clicking the File Associations tab. [SCREENSHOT] The entries for "Open", "View" and "Edit" correspond to the names of programs on your system as you would type them on the command line. Although it is possible to edit these entries, most beginning users will not want to. It often takes some fiddling to get the entries correct.

Internet Software

Internet software includes applications you typically would use for Internet-related tasks -- reading e-mail, chatting with friends, and surfing the web.

There are likely many Internet tools installed on your computer. Programs such as **ping** and **netstat** are primarily diagnostic tools. Other software such as Mutt and w3m exist for reading e-mail or websites, but are text-based.

Web Browsers

A web browser is a useful application to have even if you do not have Internet access at home. In addition to visiting websites on the Internet, you can also use a web browser to view much of the documentation installed on your computer.

- -4pc - -4pc

Depending on the capabilities of your system, you might have a number of different web browsers on your system. Dillo and Links are somewhat limited in functionality, but they run well on most computers. Opera is a more powerful browser, but due to licencing restrictions we may not distribute it freely. Firefox is an excellent web browser, but it requires a lot of memory to run well.

One of these web browsers is set to be the default on your system. The default web browser is activated when you click the Web Browser icon in the IceWM program menu or toolbar. In addition, programs that make use of a web browser will generally call the default browser. It is possible to change the default browser if you have an alternative installed. For instructions, see [WHERE?]

In the following sections we will briefly describe some of these web browsers and their quirks.

Links

When first started, Links presents a rather blank screen, offering nothing but a "back button" in the top left corner of the screen. [SCREENSHOT] However, Links does support menu options. You can activate them by clicking on the topmost grey bar, or by pressing the Esc key.

Links was originally developed as a text-only browser, and so it has good keyboard support and not a lot of graphical features. However, it is fast and light and displays many websites reasonably well. [SCREENSHOT OF AMAZON.COM? A BANK?]

To visit a website in Links, click File->Go to URL or press the g. A box will open up where you can type the website you wish to visit. [SCREENSHOT]

You can also visit local directories on your computer, but there is no nice graphical selection screen to do so. Instead, you must type in the location, prefixed by `file:// `. For example, to browse the documentation in `/usr/doc/debian/` you would type **`file:///usr/share/doc/debian`** in the Enter URL field. [SCREENSHOT] Note that you type in three slashes after `file:`.

Although it is a reasonable browser, Links has some quirks and deficiencies you should know about:

- Links does not (yet?) support *Cascading Style Sheets* (CSS), which is a language used by many websites to format colours, layout, and text. Links is capable of displaying such sites, but they will appear very plain.
- Links will collect and use *cookies*, but it gives you no way to edit the cookies you use. (For more information about cookies, see [WHERE?]) [VERIFY THIS]
- Links does not support *plugins* very well. For example, it is hard to get *Macromedia Flash* or *Quicktime* files to play in the browser.
- [SSL SUPPORT?]

Dillo

Dillo is a lightweight web browser designed with security in mind. It is a fairly simple browser, but it is capable of rendering most Internet sites. Like Links, it is often a good browser to use when reading documentation. [SCREENSHOT]

Here are some quirks and deficiencies you should know about Dillo:

- -4pc - -4pc

- Like Links, Dillo has no support for CSS or plugins. Unlike Links, Dillo does not support *Javascript*, a programming language used for animations, popup windows and some fill-in forms on the Internet.

- Dillo uses the file `~/.dillo/cookiesrc` to control the websites that can use *cookies*. For security reasons, by default Dillo disables all cookies. [WILL WE CHANGE THIS?] If you require the use of cookies (for example, to log into websites) you may need to edit this configuration file using a text editor such as Nedit. Instructions about the cookie format can be found in the file `/usr/share/doc/dillo/Cookies.txt`

Opera

Opera is a commercial web browser. It is fairly complete, and runs reasonably well on any computer with more than 32MB of RAM. [IS THIS TRUE?] Because it is a proprietary commercial product, we cannot distribute it freely. However, if you wish you may download the browser from http://www.opera.com. If you choose to do so, you should get the Debian package, which ends in `.deb`. You probably want to get the version that has "QT statically compiled", and you will want the version for Debian Sarge. [THIS WILL GO STALE]

After you have downloaded the package, see [WHERE] for instructions on how to install the package.

The Gaim Instant Message Client

An *instant message client* is an program you use to chat with others online, using *chat protocols* such as AIM, Yahoo! chat, or IRC. Gaim is a multi-purpose client that supports the most popular chat protocols used today.
[WHAT INFO YOU NEED] [HOW TO SIGN UP FOR ACCOUNTS]

Games

You can find several games in the Programs->Games area of the IceWM program menu. Instead of documenting all the games on your computer, we will mention quirks of certain games.

The Ace of Penguins Suite

The Ace of Penguins is a suite of (rather addictive) solitare games, including Freecell, Minesweeper and Taipei. Most of the games offer online help that you can access by pressing F1. Each helpfile documents the rules of the game in question, how to win, and the keys used to start new games.

XDigger

XDigger is a puzzle/arcade game. [SCREENSHOT] The game does not have menus; it is controlled by the keyboard. Use the arrow keys to navigate your player, and press Esc to restart a level.

- -4pc - -4pc

Chapter 3. Safe Computing

Computers are powerful tools, and like many other tools they can be misused. When you buy a table saw or a sewing machine, you want to be aware of how to use your tool safely; not doing so could cost you limbs you would rather keep. Although your computer will not cut off your hand or sew your fingers together, computers can be used to cause you harm in other ways. In this chapter, we will discuss some of the dangers, and share some tips you can follow to protect yourself.

In the interests of education, we'll be describing some scary scenarios in this chapter. Please don't let these scenarios frighten you away from using your computer. Although there are always risks to using computers (just as there are always risks when using table saws) computers also have a lot to offer. Being aware of these dangers can help you minimize the risks, and help you maximize the enjoyment and productivity you get from your machine.

The term "computer safety" can refer to many different things. In this chapter we will touch upon the following areas:

- Keeping your files safe in case anything happens to your computer.
- Keeping your personal information private, and out of the hands of people who would abuse it.
- Using the Internet wisely, without making a fool of yourself.
- Using your computer responsibly, so that it does not take over your life.

We won't be able to cover any of these topics very deeply; no doubt entire books have been written on each of these topics. However, the introduction in this chapter will go a long way towards making your computer experiences safer and more pleasant. [REF TO FURTHER INFORMATION?]

Protecting Your Files From Data Loss

Computers are excellent at archiving data and keeping records. Some of the important files I keep on my computer include:

- Information about my monthly earnings and expenses.
- My resume and employment history, my medical information, and information about where I have lived over the past five years.
- Lists of books to read, movies to watch, music to listen to, and corresponding lists of books I have read, movies I have watched, and music I have listened to.
- All the writing, notes, and projects I have worked on ever since I bought my first computer.
- All of the e-mail I have received since I started using e-mail.

- -4pc - -4pc

Whenever I am fortunate enough to buy a new computer, I transfer all of the data from my old computer to the new one. I plan to keep doing this indefinitely. Keeping my information in electronic format makes it easy to search, easy to retrieve and easy to transport when I have to move. Although I do keep paper copies of some of my most important information, I have far too much data to keep it all in paper format.

The convenience of electronic recordkeeping has its costs, however. If my house caught on fire and my computer burned up, I would be in some trouble. Losing my data would not kill me, but it would make certain aspects of my life (such as future employment and finding employment references) difficult. For this reason, I want to keep my data safe.

There are two aspects of data safety to worry about. The first is safe file handling: how to work with your files so that you do not accidentally damage, lose, or delete them. The second concerns long-term storage and disaster recovery, which usually involves some kind of backup system. In the following sections we will discuss both of these in some detail.

Handling files safely

Most people don't intend to do stupid things with their data. Nobody says to themselves, "I think I will delete my work report today, thus making my life miserable for the next few weeks." Rather, bad things happen to good files because people aren't aware of what they are doing. In this section I will talk about some of the stupid things I have done so that you can learn from my mistakes.

Accidentally deleting files

Here's the scenario: I look through my directories and find some directory that is full of junk. It's taking up hard drive space on my account for no good reason. So I open up the file manager, select that directory and delete it.

Then 10 minutes later I realize that the directory contained some important configuration file I now need. At this point I am stuck.

One thing you should know about Linux is that typically files cannot be undeleted. (This is because of the *filesystems* we use on your computer.) Once you delete a file, it is gone forever, and for the most part cannot be retrieved.

I know that deleted files are irretrievable in Linux, but I still make this dumb mistake? Why? My error is not to check inside the directory I am about to delete before I delete it. I sometimes make this dumb error even when I check my directories, because I will not realize that some file (or group of files) is important until it is gone.

How do you avoid this? Here are some tips.

1. Whenever you are about to delete a file, pause for a couple of seconds. Ask yourself whether you really want to delete the file in question. Ask yourself whether you can recreate the file (by typing it in again, or whatever) if you need it in the future. If you have some doubt about the file, you may want to keep it around.

2. Instead of deleting files, make a directory called `garbage` in your home directory. Every time you want to get rid of that file, move it to the garbage directory. Every week (or month), delete the contents of the garbage directory.

   How does this help? It moves your files to a place you don't expect. If you find that you need the file, you will notice that it is not in its usual location, and realize that it can be retrieved.

- -4pc - -4pc

3. Use a temporary directory to store temporary files. I avoid cluttering up my home directory with files I know I will only need for a few minutes. Instead of saving those files in my home directory, I save them to a special directory named `/tmp` . This directory is special because it is automatically emptied every time the computer boots up. This means that my temporary files get wiped out, the files I want to keep are stored in my home directory, and I am less tempted to clean out the clutter, possibly deleting important files in the process.

4. Organize your directories well. Directories are an excellent tool to keep the files in your home directory tidy. If you have a tidy home directory, then you will have to clean out files less often, which means that you are less likely to delete important files.

   For example, I keep my writing in a folder called `writing`, my personal record-keeping files in a directory called `personal`, and things I download from the Internet in a directory called `downloads`. Some of these directories contain other directories. Although this structure seems complicated at first, it helps me keep track of where my files belong. It also gives me a sense of where my important data lies: I could probably lose my `downloads` directory without too much damage, since I could just download those items from the Internet again. However, I really want to preserve my `personal` and `writing` directories, because those directories are hard to recreate.

   [THIS DOES NOT BELONG HERE]

5. Archive files instead of deleting them. The truth is that hard drive space is cheap and it gets cheaper all the time. If you have enough space to store your files now, you will have even more space if you ever upgrade your hard drive or computer. Thus, it usually does not cost a lot to keep extra files around. (On the other hand, I am a packrat, and I understand that this approach has its disadvantages.)

   One technique I use is to archive files I know I will not need in the near future. For example, I use a command called **tar** to archive ("zip up") the directories that contain my old e-mail. Since I have the space to store them, these files do no harm. But when I need them, I can simply unarchive them and use them again. [HOW CAN USERS CREATE ARCHIVES?]

6. Be careful about wildcards. Some programs allow you to select groups of files using wildcards. A common example is the command-line utility **rm**. The following invocation of **rm** will delete all files that end in `.bak` (which is a common ending for backup files):

   ```
   rm *.bak
   ```

   The "*" is a *wildcard*. It means "match any pattern." If an unfortunate user types the following instead:

   ```
   rm * .bak
   ```

   then the **rm** will delete all files in the current directory. The difference is one space.

   Since you can delete files using the file manager, you never need to use **rm** directly, but you may run into other programs that support wildcards. This is fine: wildcards are a powerful tool, and you should feel free to use them. However, be especially careful that your wildcards match the patterns you are thinking of.
   [MAYBE DON'T MENTION WILDCARDS?]

- -4pc - -4pc

Save Early, Save Often

Sometimes bad things happen to good computers, and you lose the files you are working on as a result. The reality of computers is that sometimes they break or turn off unexpectedly. Your house may lose power. Your computer might suddenly lock up. A malevolent child may disconnect your keyboard plug. You might stretch out and flip the switch on your power bar.

Is there anything you can do to prevent data loss? Your best defence is to save your work frequently. If you save your work every five minutes, you will lose at most five minutes of work if the power goes out. Losing five minutes of work is still not fun, but it is a lot better than losing five hours of work. I save my work every time I take a pause to think of what to write or program next. That way, I incrementally save every thought I enter into the computer.

Some programs have "auto-save" features. They will automatically back up your work every few minutes. Such features are good, but it is still wiser to save your work explicitly. Not every program you use will automatically save your work, and retrieving autosave files is not always easy.

One tool that can make it easier for you to get into the habit of saving frequently is the *keyboard shortcut*. A keyboard shortcut is a sequence of keystrokes that will perform some command in your program. For example, in Abiword the keyboard shortcut to save your document is [WHAT?]. Pressing this shortcut will save your document without you having to use your mouse to navigate through the menus. Alternatively, many programs have a single toolbar button that you can click to save your work. [SCREENSHOTS -- ABIWORD, ???]

Recovering From Crashes

[LIST CRASH FILES HERE] [MAYBE PUT THIS IN TROUBLESHOOTING]

Backups

A *backup* is a copy of your files, preferably stored in a safe place. If anything should happen to your original files (for example, if your hard drive dies), you can restore the lost files from backup. Backups are also used to *archive* your data, so that you can access it at a later date.

People are funny when it comes to making backups. Everybody knows that they *should* back up their data regularly -- but many people (maybe most people) do not make regular backups. Then when something bad happens, those people lose their data and feel sad. You do not want to be one of those people.

Why don’t people do what they know is good for them? One big reason is convenience: if it takes a nontrivial amount of time, energy or money to do backups, then the backups don’t get done. Ideally, the backup process should be completely automated: the less humans are involved with their backups, the more likely backups are to get done.

How often should you make backups? It depends on how often your data changes, how much you care about your data, and how much hassle it is to carry out the backup process. The more frequently you make backups, the less data you will lose in the event of a crash.

There are many sophisticated backup schemes available. Some require complicated software; others require expensive hardware devices that likely did not come with your computer. We will focus on the simplest, cheapest alternatives, and then mention some other possibilities at the end of this section.

Each backup alternative has advantages and disadvantages. Some factors to consider when deciding on a backup solution include:

- -4pc - -4pc

- Convenience. If it is inconvenient to use a backup method, the backup will not get done. If a backup method can be fully automated, you can set it up and then forget about it.
- Cost. What additional charges are associated with this method? You may have to buy extra hardware, or consumable resources (such as blank CD-ROMs or floppy disks), or services (such as Internet connectivity).
- Backup capacity. How much data can you backup with this method? You have to match this to the amount of data you want to backup.
- The nature of your data. How much data do you need to back up? What data can you not afford to lose? If you only have a few files that are irreplacable, you may be able to back up just those few files. Most people focus on backing up personal files; program files can always be reinstalled.
- Configuration difficulty. How long does it take to set up this backup system? Will you be able to do this yourself? Will you need to recruit a local computer genius to set up the backup method for you?
- Privacy. What data of yours contains sensitive information? How easy is it for others to see or steal your backed-up data?
- Disaster protection. Say that your computer catches on fire, or is stolen. Will you still be able to access your backups?

Floppy disks

Unless you buy additional hardware for your computer, your most readily available backup device is likely to be your floppy drive. To back up files to floppy disks, you copy the files you want to keep to the floppy from time to time.

Here are some advantages of this method:

- You probably have a floppy drive already, and you know how to copy files. [REF] It is easy to get started with this option immediately.
- Floppies can be transported and stored in a safe place. You can keep floppies with very important information in a safety deposit box.
- Many computers have floppy drives, so you can usually find another computer on which to restore your files.

And here are some disadvantages:

- This is a very inconvenient method. You have to remember to do the backups, and you have to remember to have backup floppies on hand.

  You can get around this disadvantage a bit by backing up files as you create them. Immediately after finishing with an important file for the day, you can back it up to a floppy.
- Floppy disks typically store 1.44MB (megabytes) of information. By todays standards, this is not much information. The amount of data you can back up is very limited. In fact, some data files are bigger than 1.44MB in size. Getting these files onto floppies is tricky. (Two commands named **split** and **join** exist to do this, but they both require the use of the commandline.)

- -4pc - -4pc
  - Floppies are notoriously unreliable. They break down from time to time. I get around this problem by backing up files to two floppy disks at a time. That way, if one floppy breaks down, I have my file on the other floppy.

  - You need to obtain and store stacks of floppies. Compared to other storage media, floppies are quite expensive. Sometimes you can find used floppies for free or cheap, but otherwise you will have to spend some money to get some floppy disks.

Backing data to floppy disks is not a great method, but they are a good way to get into the habit of backing up your data.

Internet Storage

If you get Internet access at home, you may be able to backup your files to servers on the Internet. Some Internet Service Providers provide you with storage space for your web pages or other data. It may be possible for you to store copies of important files to this space. You might be able to e-mail yourself some files you wish to back up, and store them in your Web-based e-mail account. You might be able to purchase storage space from some other Internet provider as well.

Here are some advantages of internet backups:

- The process can often be completely automated. If you can get the backup system set up (which often will require the services of a computer guru) then your computer can back up its data while you are asleep.

- Since your Internet server is far away, your data will be safe even if your computer catches on fire.

Because of their convenience, Internet backups are an excellent option for many people. This method does have some disadvantages, however:

- You need to purchase Internet access at home. You may also need to purchase some Internet storage, if your Internet service provider does not provide it.

- This method works best if you have a reasonably fast, reasonably stable internet connection.

- Currently, we know of no easy-to-use software to configure Internet backups. That means you will likely have to recruit a computer guru to help you out.

- Security is a big issue. As soon as you transmit your data on the Internet, you run the risk of naughty and/or curious people looking at it. If the data you want to keep is confidential (that is, you don't want to share it with the entire world) then you probably need to *encrypt* it, which means scrambling it so that you can recover your data and (hopefully) nobody else can. This process takes some care and knowledge, and it is not foolproof. (You may want to investigate **gpg** if you are interested in this.)

- The amount of storage you will have on the Internet is probably small. You will probably have to pick and choose which files to back up.

- -4pc - -4pc

[MORE INFO -- ssh, gpg, ftp, tar, gz, rsync]

CD/DVD backups

If you have (or get) a *CD-RW* or *DVD-RW* drive installed on your computer, you may want to back up your important files to CDs or DVDs. To back up your files, you would insert a blank CD or DVD, use a program (such as xcdrost) to select the files you want to back up, and "burn" the data to the blank CD. This is a very popular backup method.

Some advantages of CD or DVD backups include:

- Blank CDs are cheap. They are probably the cheapest backup media you can buy these days. Blank DVDs are more expensive, but they store even more than a blank CD.
- CDs offer a lot of storage space at a reasonable cost. You may not be able to back up your entire hard drive to CD, but you will be able to back up more files than with almost any other cheap backup system. This is a big advantage.
- CDs and DVDs can be stored far away from your computer. You can likely store them in a safety-deposit box.
- Once you have learned how to burn a CD, the process is fairly easy. Furthermore, the process can be mostly automated through the use of scripts (but you may require a computer expert to set this up for you).
- CD/DVD drives are useful for many things besides backing up data.

This backup method does have some notable disadvantages, however:

- Although CDs are cheap, they are consumable. Regular writable CD-ROMs can generally be written to once, and only once. That means you might waste a lot CD-Rs if you make a lot of backups.

  It is worth noting that rewriteable blank CDs and DVDs exist. They can be blanked and rewritten to hundreds of times. Rewritable CDs are getting hard to find, but rewritable DVDs are readily available.
- The CD-based backup system is inconvenient: you need to remember to put blank CDs into your computer. It is possible to set up reminders so that you remember to do your backups, but the process cannot be completely automated.
- Usually, you have to buy (and install) extra hardware to make use of this option. CD-RW drives are not that expensive, but as of this writing (late 2004) we do not include them with the computer systems we sell. DVD-RW drives are expensive. In either case you have to install this hardware in order to use it.

- -4pc - -4pc

[NEED: Better CD-burning software] [PICTURES OF CD/DVD DRIVES. HOW DO YOU IDENTIFY THEM?]

USB Keychain Drives

USB keychain drives are small storage devices that plug into the *USB* port of your computer. They are called "keychain drives" because they are small enough to attach to a keychain. The capacities of these devices is growing rapidly. At some point we may see very cheap, low capacity keychain drives for sale, but currently drives with smaller capacities fall off the market and are not sold at significantly cheaper prices than the expensive ones.

Once a keychain drive is plugged in Linux treats it as a hard drive. It is first *mounted*, and then it appears as a directory in the *directory tree* [REF]. You can then copy files to it just as you would copy files onto a floppy.

Keychain drives have replaced floppy disks as a standard way to transport files from computer to computer. However, they have advantages as a backup solution as well:

- Keychain drives offer a fair amount of storage capacity. You will not be able to back up your entire hard drive, but you will be able to back up many files.
- Keychain drives are easier to set up than most other hardware. You can plug them in while your computer is running, and Linux will autodetect the drives. It is fairly straightforward to set up Linux so that it mounts the drives as well.
- Since keychain drives are so small, they can easily be transported and stored in a safe place.
- Most keychain drives are durable. I have heard of a keychain drive going through a washing machine undamaged.
- Much of the backup process can be automated once the drive is plugged into the computer.
- Occasionally it is possible to get older models (with smaller capacities) on sale.

As always, there are some disadvantages to consider:

- Keychain drives are usually not that cheap. In Canada, they will cost you $50-100 each.
- In theory, any USB keychain drive will work with any USB port. In practice this is not always the case: some drives have problems with the ports on older computers, and some drives do not work under Linux as well as they should.
- Either you have to remember to plug in the drive and start the backup process manually, or you have to leave the drive plugged in all the time. Leaving the drive plugged in all the time is more convenient, but less safe if your computer is stolen or damaged.
- Because keychain drives are so light and small they are easy to lose or have stolen. You have to be careful to keep your drive in a safe place, especially if you back up important data to it. (On the other hand, some drives are sold with security features so that strangers cannot look at your data.)
- If your computer only has USB ports at the back of the machine, it can be awkward to plug in the drives. Some drives come with extension cables to assist you with this, however, and some computers have USB ports at the front of the computer for easy access.

- -4pc - -4pc

Second hard drives

An interesting solution to the backup problem is to install (or get somebody to install) a second hard drive on your machine. This drive will be a backup drive -- every so often Linux will back up your data to the backup drive. Although this has some drawbacks, the convenience is a big win, and I use this backup solution myself. Here are some of the advantages:

- The convenience is advantage number one. Every aspect of the backup process can be automated. My computer backs up my data once a day.
- You can use any kind of hard drive you have available. It is sometimes possible to get cheap used drives on sale. Doing so will limit the data that you can backup, but can be very cost effective.
- Your backed-up data is readily available on your computer. All you have to do is go to the directory containing the backup data. This is convenient if you accidentally delete a file and want to retrieve the previous day's version.
- You can lose your data only by having both of your hard drives fail. You can restore all of your data if either your backup drive or your main hard drive break. (This has already saved my data once.)

As with other solutions, however, this backup system has drawbacks:

- Your computer is not protected in the event of a disaster. If a power surge wipes out one drive, it will likely wipe out the other as well.
- If you install a used hard drive as the backup drive, the backup drive has a greater chance of failure.
- You will probably want somebody's help in setting up the backup system so that it runs automatically.
- From time to time you have to verify that your backup system is still working. Linux does some of this for you -- if a drive fails, usually Linux will report errors when you start your computer.
- You need to obtain a hard drive, and you have to get it installed. Depending on your computer you may not have the physical capacity to store the drive.

- -4pc - -4pc

Other Backup Possibilities

Many, many backup solutions exist. In the following section we'll list some of the alternatives that are available, and briefly cover some advantages and disadvantages. Most of these alternatives are probably out of reach for the home user, however.

ZIP Drives

ZIP drives used to be fairly popular, but their niche has been taken over by USB keychain drives. ZIP drives take cartridges that act as portable hard drives. They come in internal versions that are installed into your computer like a floppy drive, and external versions that plug into a *parallel port* [REF] or USB port. You might find other makes and models of cartridge drives, such as Jaz drives, or Shark drives.

The cartridges do not hold a lot of information by today's standards (100MB and 250MB are the most popular sizes) and they are fairly expensive to purchase new. However, if you can purchase cartridges and a drive used, this can be a worthwhile solution.

The backup procedures are similar to USB keychain drives, and many of the same advantages and disadvantages hold.

External Hard Drives

You can purchase an external hard drive which plugs into a USB or *Firewire* port of your computer. (Be careful: most computers we sell do not have Firewire support.)

These drives are portable and have lots of storage capacity. Unfortunately they are quite expensive -- more expensive than internal hard drives of the same capacity. Other than this, they have many of the same advantages and disadvantages as installing a second internal hard drive, except that external hard drives are portable.

Network backups

Some people own multiple computers, and designate one of those computers to be a backup machine. They configure their other computers to back up data to this computer via a *local network*.

This solution is similar to having Internet backups, except that the backed up data can be more secure (since you never send your data out to the Internet for storage). On the other hand, setting up these solutions means running multiple computers, setting up a home network, and configuring each computer to send its data to the backup server. For these reasons, network backups are probably too much work for the average home user.

RAID

A *RAID array* is not really a backup mechanism, but it is similar, and very popular with system administrators. The "R" in RAID stands for "redundant", because these solutions write data multiple times to one or many hard drives. If any one copy of the data becomes corrupt, then the corrupted data can be detected and recovered from the other copies. Some setups use nine hard drives set up in parallel -- if any one hard drive fails, it can be replaced with another drive without losing data and without turning off the computer.

Professional-level RAID can get very expensive, and takes some work to configure. Even simple RAID setups require a lot of storage space. You probably do not want to consider this solution unless you are genuinely interested in the topic, or (for some reason) your computer is a server that contains very important data that must be accessible all the time.
Tape Drives

- -4pc - -4pc
Tape drives take cartridges that use a similar technology (and look much like) audio cassettes. Tape-based backups used to be very popular with system administrators. They are still in wide use.

Tape drives offer great deals of storage per tape. There is a lot of software written which supports tape drives as a backup storage medium. They are often used to archive data for a few months, so that the administrators can restore data that was deleted a few months ago.

On the other hand, tape drives (and tapes) are not that cheap and are increasingly hard to find. It also takes a fair amount of work to configure the backup system, and -- unless you are given a robotic jukebox which switches tapes for you -- you have to remember to insert tapes into the drives when it is time to backup the system. If you are given a tape drive and a set of tapes for free, you might want to consider this option. Otherwise it is likely too much work for too little benefit -- you can get many of these same advantages by using CDs or DVDs instead.

Printouts

If you have very important documents you cannot afford to lose, it is worth considering printing those documents out and storing them in a safe place, far away from your computer.

Some people go overboard with printouts. I have seen people who print out every e-mail they receive. Although reading paper printouts is easier than looking at text on a screen, printing out all of your data is not a smart idea. In addition to wasting paper and ink, paper is very heavy, and filing away paper copies of your documents becomes very cumbersome very quickly. Having said that, in certain situations archiving printouts is worth the cost.

How To Choose a Backup System

As we have seen, there are a large number of backup systems to choose from. Deciding which backup system to invest in can be difficult. Here are some pointers that might help you in your decision:

- If you honestly don't care about your data, and you would not miss your data at all if it were to disappear, then maybe you don't need a backup system at all. In my opinion few people are in this category, however.
- If you have no other backup system, start by backing up your most important files to pairs of floppy disks. If you only have a few files you desperately want to keep, and you can remember to do your backups regularly, this may be sufficient for your needs.
- If you have a hard time remembering to do your backups, then using the Internet or installing a second hard drive in your computer may be the most reliable. Another possibility is to use a USB key that is always plugged into your computer.

  Another alternative is to use a calendar or other reminder device to prompt you to get your backups done. One application that can pop up regular reminders is called rclock. [HOW CAN PEOPLE LEARN MORE? RCLOCK IS A BAD APP..]
- If cost is a big concern, then it may be worth saving your money to buy a USB key or second hard drive. The inital costs for these options is high, but there are no follow-up costs to pay on a regular basis.
- If you do not have easy access to computer experts who are willing to help you set up your backup system, then CD/DVD or USB key backups may be the easiest.

- -4pc - -4pc

[NEED: BACKUP APPLICATIONS]

Using Your Computer Responsibly

Computers are incredibly useful devices. They are capable of increasing human productivity and potential immensely: we can use them to store and organize vast quantities of data, to carry out complicated calculations, to model all sorts of real-world situations, and to automate all kinds of boring, repetitive tasks. There are certainly aspects of computing we do not like, but it is hard to argue that they haven't improved our lives considerably.

Ironically, many people find that their productivity decreases when they start using computers. This is because computers are excellent ways to waste time. Thanks to computers, we can play video games, we can spend hours reading e-mails and websites each day, we can chat in real-time with people on the other side of the world, we can polish presentations and documents to a degree that would have been inconceivable a few decades ago, we can learn and explore and play and invent and program. In moderation, any of these capabilities can enrich our lives. In excess, they can ruin our lives. We spend so much time on our computers that we ignore our loved ones, our jobs, and our health. We sit in front of our computers all day, growing fat and sedentary and developing all sorts of repetitive stress injuries. We blow our budgets -- on new hardware, on service fees, on shopping online. These are not nice things to say, but they are realities. I jeopardized my schooling because I preferred playing NetHack to completing assignments and studying for tests. Even today, I have to carefully monitor my Internet usage, and I dare not subscribe to Internet access at home. Other people lose jobs and/or get divorced because they cannot control their computer use. You do not want to be one of those people.

I suspect many people who first start using computers go through an obsessive phase, where they spend more time than they should with their new toys. Some people grow out of this phase quickly, and avoid letting their computers take over their lives. The rest of us have to be vigilant and take precautions to prevent our digital toys from ruining our productivity and our lives. Learning to use our computers responsibly is one of the most important skills we can develop. It is at least as important as learning how to use programs and applications proficiently.

In this section, I will outline a few of the traps people fall into, and describe some techniques you can practice to keep your computer usage under control.

Scheduling Your Computer Time

If you find your computer time cutting into your other activities, you will want to schedule your computer activities and time so that they do not interfere with the rest of your life. This is much easier said than done, but there are some strategies that can help:

- Be aware of your computer usage. Keep a log of the time you spend on the computer, and what activities you use the computer for. Categorize the time you spend on the computer in terms of work and entertainment. This exercise will give you a clear idea of where your time is going, and possibly motivate you to improve your habits. [PROVIDE TABLE]
- Make conscious decisions about how much time you are comfortable spending at the computer. Then compare your computer usage against those limits, and plan adjustments to your computer usage accordingly. If you find that you would otherwise be spending your computer time on less productive and enriching entertainment, then you may be happy with your computer usage. On the other hand, if your computer usage interferes with your work, your family, your volunteer work, your exercise, or other worthwhile causes and activities, then you will want to modify your habits.

- -4pc - -4pc

- Schedule the computer activities that suck up your time just before appointments and other activities that get you away from the computer. Provided that you have a lot of incentive to keep those appointments, you will have to end your computer sessions to get on with your next scheduled activity.

  Be aware that this strategy can backfire disastrously. If you are not careful you can miss important appointments, and it is easy to be late for appointments because you are tempted to spend "just a few more minutes" at the computer. To get around this, I keep a clock (with an alarm!) that goes off when I need to get off the computer. I know that I will be late if I ignore the alarm, so I reluctantly shut down the computer and leave.

  If you find that you cannot control your computer use just before appointments, you may need to employ the opposite strategy: *never* schedule time-sucking activities just before important appointments.

- Be especially careful of computer activities that occur in real-time: chat rooms, online multiplayer games, online auctions, and so on. Such activities create a sense of urgency that is difficult to resist. I personally avoid all forms of real-time computer activity, because I do not have the self-control to resist the urgency.

- A common problem people face is that they check their e-mail too many times a day. You can get around this problem by scheduling one or two e-mail checking sessions per day, at specific times. I get around this problem by avoiding Internet access at home; if I want to check my e-mail, I have to walk for twenty minutes to do so. Even though e-mail is my primary means of communicating with others, I find that checking my e-mail once a day is usually sufficient for me to keep in touch.

- Keep track of the activities you need to accomplish each day. Make sure that you get those activities done before starting your computer.

  I find long "to-do" lists overwhelming, so I write short lists of tasks (of four or five items) that need to be done before I can play on the computer.

- If you start your computer to do productive work only to find that you are wasting time, it can help to write down a list of items you had meant to accomplish on the computer. Write this list before you power on the computer, and keep this list where you can easily see it while you are on the computer. This list will serve as a reminder of the work you need to get done.

- If you live with roommates or family members, keep the computer in a public place, as opposed to a private bedroom. The presence of other people can help you stay on task (unless they tempt you into playing video games with them!).

- You may have to refrain from certain activities you cannot control. I cannot be trusted not to play NetHack instead of doing useful work, so I refuse to install the program on my computer. I cannot be trusted not to waste my time on the Internet, so I refuse to subscribe to Internet services. I cannot be trusted to control my time when visiting certain web communities, so I refuse to visit them. It is never fun and often difficult to keep these promises, but in my life it is necessary. You may find that there are certain activities you must avoid in your life as well.

- Enlist the help of a friend or a loved one to help you keep your computer use under control. Choose a friend or loved one with a thick skin; when that person reminds you of your commitments you will surely be cranky and resistant.

- Get into a regular weekly schedule of computer use you are comfortable with. Finding healthy routines goes a long way towards keeping your productivity up.

- -4pc - -4pc

Controlling Your Computer Expenses

Even though computer prices have dropped astoundingly over the years, computers remain an effective way to separate you from your money. If you do not have much disposable income, this can be a big issue. In this section we will describe some of the expenses associated with computing, and a few tips to keep those expenses managable.

The most useful tip applies to all of the computer expenses below (and, for that matter, to any expense): be conscious. Be aware of how much money you can be afford to spend, and be aware of how much money you are spending, and always be sure that you never spend more money than you have.

A closely-related tip is to never spend money you do not have. Computers can be expensive, but they are not good investments; the value of hardware depreciates too rapidly. If you are faced with a decision between going in debt to finance your computer habit and doing without, you are often better off doing the latter.

With these basic guidelines in mind, here are some specific expenses to watch out for.

Hardware

If you are reading this guide you are likely using an older computer. You have already resisted the temptation to buy the latest, greatest hardware. Keep up the good work! If this is your first computer, you may be tempted to purchase a newer computer sometime in the future. If you choose to do so, there are some things you should know about the computer market.

The reality of the computer industry is that products grow obsolete very quickly. As a result there is a lot of marketing pressure for consumers to keep up with the newest (and most expensive) hardware. An interesting sign of this marketing phenomenon is the "word processor" sell-story. Computers that were marketed as high-end workstations three years ago are now marketed as machines suitable for "word processing and web surfing", and little else. The computers did not change. Their capabilities did not change. Only people’s expectations changed.

If you understand what you need your computer for, you can save a lot of money by buying computer peripherals suited to your needs. Finding those needs requires research, however: you don’t want to buy an underpowered computer that is not suitable for the work you need to get done. There is always some tradeoff between functionality and price. An excellent way to research your needs is to ask others how they use their computers. Another way is to find out about what software products are available, and then to buy a computer capable to run that hardware. A third way is to intentionally buy an underpowered computer cheaply, and then find out what functionality you are missing.

An important aspect to keep in mind is quality. The cheapest computers on the market may have impressive specifications, but they are made with cheap components. As with so many other consumer items, you can often get better value by purchasing higher-quality hardware that is less powerful. The used market can be a good place to find such items.

On average, it takes three years for computers to fall through the market. In other words, the latest, newest machines sold today will be sold as "bottom of the line" computers three years down the road, and four years down the road you will only be able to find them used. This used to matter a lot, but it is far less important now, as much software available today will run on computers five years old or older.

Games Machines

Video games are notorious for being demanding on hardware, so they deserve special mention. The latest, greatest commercial games will always demand the latest, greatest hardware for best performance, and often these games will not run at all on older computers. If you (or your kids!) have your heart set on playing the newest games, be prepared to spend a lot of money.

- -4pc - -4pc

Having said that, Linux is less prone to the games phenomenon, because not as many commercial games are available for Linux. Your computer also comes with a number of non-violent games preinstalled. These games do not have the fanciest graphics or most elaborate soundtracks, but many people will enjoy them. For the most part, the computers we sell at computer recycling are not capable of playing video games that you will find at the store. [EXCEPTIONS?]

Upgrades and Peripherals

Instead of purchasing brand new computers, many people attempt to prolong the lives of their existing computers by *upgrading* them -- installing newer components in the computer to make it faster and more capable. This can be a smart strategy, but as always you need to keep your budget in mind. You should also balance the cost of your upgrade components against the cost of purchasing a newer computer. Finally, you should consider the value of the upgrade. If an upgrade is not going to improve the quality of your computing experience much, you may be better off saving your money for something more worthwhile.

In addition to upgrading the computer's existing components, people purchase *peripherals* for the computer to expand its functionality. Some common peripherals include printers, scanners, and webcams.

We describe some of the technical considerations in buying computer components and peripherals in [WHERE?].

Consumables

Some devices on your computer may require consumable goods in order to function properly. For examples, printers are fairly useless without ink or toner and paper. If you wish to burn CDs or DVDs with a CD-RW or DVD-RW drive, you need to purchase blank CDs or DVDs. The cost of these consumables can really add up over time. In this section we will discuss some common consumables and some ways to reduce their cost.

Blank CDs/DVDs

If you get a CD-RW or DVD-RW drive installed on your computer, you will want to buy some blank CDs/DVDs (which I will refer to as *blank media*) so that you can record information. Individually-packaged blank media are the most expensive. Blank media are also available in *spindles*, which reduces the cost per CD/DVD, but requires a bigger expense upfront. You also do not get individual envelopes or cases for your media when you purchase a spindle.

A good alternative to purchasing many blank write-once media is to buy *rewritable* media. This will cost you more money per CD/DVD, but this option makes a lot of sense if you do not need to keep your data around in the long term. For example, I use rewritable media for storing programs and Linux distributions because software goes out of date quickly -- by the time I would reuse the CD, a new version is available. I also use rewritable media for making backups.

The big disadvantage of rewritable CDs is that the writing speeds are slow. Instead of being able to record data at 12x or higher, the blank rewritable CDs I have found have all been limited to 4x speeds. This means it takes 15 minutes or longer to record a CD's worth of information.

Printer Cartridges and Paper

I'm probably not allowed to say this, but printer manufacturers remind me of drug dealers. Most printer manufacturers sell their printers very cheaply to get you hooked on their product. Once they have locked you in to using their printer model, they can charge high prices for ink or toner cartridges. Since the cartridges are not interchangeable between manufacturers, you are stuck buying printer refills for the specific make and model of your printer. For this reason, it makes a lot of sense to find out how much printer refills will cost *before* deciding on a printer model. Prices vary widely -- some older printers have cartridges that cost $30 or more, while newer printers have cartridges costing six or seven dollars.

- -4pc - -4pc
Fortunately, the situation has improved over the past few years. Large office suppliers (such as Business Depot and Staples) now sell house-branded printer cartridges for a wide variety of printers. The cost of these printer cartridges is usually cheaper than the versions sold by the manufacturers, but usually the price difference is only a few dollars.

There also exists a thriving market for refilled printer cartridges. Some companies sell refill kits for ink cartridges, and others sell refurbished ink and toner cartridges. These services can be useful, but only up to a point -- the failure rates for these refurbished products is much higher than purchasing new cartridges. A failed cartridge can smear ink all over your pages, or can leave a sticky mess inside your printer. The other problem with refurbished cartridges is that several printer models contain some important printer machinery in the printer cartridge itself. This machinery has a limited lifespan -- it is designed to be replaced every time you install a fresh printer cartridge. For these printer models, cartridges can only be reused a few times before they break. Nonetheless, if you find a printer cartridge refurbishing service that you trust, using refilled cartridges can save you some money.

There are also some important things to keep in mind about the type of printer you obtain. Most printers available today fall into one of two broad categories: *laser printers* and *inkjet printers*. Laser printers take *toner* cartridges and produce very high-quality output. Inkjet printers take *ink* cartridges; their quality tends to be lower than laser printers, but they are more easily capable of producing colour output. [PICTURES OF INK CARTRIDGES AND TONER CARTRIDGES]

Toner cartridges for laser printers tend to be very very expensive -- sometimes in the hundreds of dollars. Inkjet cartridges tend to be cheaper, so many people purchase inkjet printers. This is not always a wise decision: although toner cartridges are far more expensive than inkjet cartridges, often the cost per page for laser printers is lower than for inkjet printers. This is because a typical toner cartridge is good for thousands of pages of printing, while a typical inkjet cartridge will print no more than a couple of hundred of pages before running dry. This means you will have to refill your inkjet printer more often, which drives the cost up. When deciding how much printing will cost you, be sure to factor in both the cost of replacement cartridges and the expected lifespan of each cartridge.

In addition to printer cartridges, printers consume paper. Many printers take standardized fine paper, but some printers require specially-treated paper. As always, you can expect specially-treated paper to be more expensive than standard paper, so it is worth determining the types of paper your printer can handle before making a purchase. [IS THERE MORE TO BE SAID?]
[REF TO HOW TO BUY A PRINTER?]

Purchasing Internet Access

In many ways, your computer is fully usable without the purchase of any additional services. The software already installed on your computer allows you to compose documents, create art of all kinds, listen to music, play games, and more. However, some people choose to subscribe to additional services. The most popular service is home Internet access, but other services exist as well. In this section we will describe some of the options and costs associated with home Internet service.

The first thing to remember about Internet access is that speed costs money. The faster your internet access, the more money you will spend. The second thing to remember is that people are addicted to speed: once you have played with high-speed access for a while, reverting to more-affordable access is painful. For this reason, you may want to be wary of a common trick access providers play: they offer cheap rates for three or four months, and then raise their subscription fees. If you choose to try such offers, you should either have the willpower to revert back to cheap access if the more expensive fees are out of your budget, or you should be budget for the higher fees from the start.

Currently, there are roughly three categories of Internet access you can purchase:

- -4pc - -4pc

1. *Dial-up* access is the oldest, slowest, and often the cheapest access. You connect to the Internet using your phone line, using a *modem* installed on your computer. This ties up your phone line, so that you cannot use your phone while accessing the Internet.

2. *DSL* access refers to a number of related technologies, all of which give you access to the Internet by sending special signals through your phone line. Unlike dial-up access, DSL access allows people to use their phones and Internet access simultaneously.

3. *Cable* access gives you Internet access through the same cables used for cable television. For this reason, cable television providers usually offer this service.

There is a fourth type of access known as *wireless* access, but as of this writing (late 2004) I know of no commercial providers of wireless Internet access, and if they do exist they are not yet widespread.

Each of the three commercially-available Internet alternatives has hidden costs. In addition to being slow, dial-up access requires that you have a modem compatible with Linux in your computer. The fact that dial-up access ties up phone service is a problem for many people; some people go so far as to have a second phone line installed so that they can receive calls while their loved ones surf the Internet. All dial-up solutions require that you purchase basic phone service at home as well.

DSL solutions can be much more expensive than dial-up access, although some companies offer cheaper, slower Internet access via DSL. In addition to purchasing a service you will need a network card and a DSL modem. You may already have a network card installed on your computer (and network cards are cheap in any case). Your DSL service provider usually supplies you with a DSL modem, although there may be set up charges and/or monthly rental fees for the modem. In addition, DSL service provided by phone companies will usually add a surcharge to your monthly service if you do not purchase regular phone service from them as well.

Cable modem services have similar costs to DSL solutions. You will need a network card and a cable modem (which is again provided by your cable provider, perhaps at a cost). Moreover, every cable Internet provider will charge you extra if you do not purchase their television cable service along with your Internet access. Sometimes this charge is small, but it is always buried in the fine print.

Cable and DSL access to the Internet is often referred to as "high-speed" access. In fact, both of these services are sometimes offered at slower speeds, or with additional limits (such as limits on the amount of information you can transmit per month). They will never be slower than conventional dial-up access, however.

One sad aspect of Internet access providers is that they are often biased against people who use Linux at home. Some providers will provide support and installation to people using Linux to connect to the Internet; many will not. Some will go so far as to say that their services will not work under Linux. This is untrue. What may be true is that you have to seek outside help to get your Internet access set up, unless you find a service provider who is reasonably-priced and Linux-friendly.

Internet service providers often try to entice you with extra features such as antivirus protection, pop-up protection, firewalls, web space for personal homepages, and e-mail accounts. Some of these add-ons will not apply to you: you will likely not be able to run the antivirus or firewalling software that the service provider provides. Your computer already has a simple firewall, and your web browser can be configured to block pop-ups.

E-mail accounts are often worse than useless: they lock you into using a single provider -- if you ever switch your provider you have to switch your e-mail address, which means you have to contact everybody you correspond with and tell them about the change. One way to get around this practice is to *forward* your mail from some other permanent e-mail address to the one supplied by your service provider.

- -4pc - -4pc
When purchasing high-speed Internet access security becomes a big concern. You do not want people breaking into your computer over the Internet. To some degree, the software firewall we provide addresses this problem. Another solution is to purchase a *router* box. These boxes come with firewalls pre-configured, and they provide several *network connection ports* so that several computers in your home can share one Internet connection. At one point simple router boxes (those without wireless capabilities) were fairly cheap, but by the time you read this they may no longer be available on the market.

Online Purchases

One grand promise of the "Information Superhighway" was online shopping -- that you would be able to purchase a wide array of goods and services from the comfort of your own home. The information superhighway has arrived, and so has online shopping. You can purchase books, housewares, computer components, and even groceries over the Internet. Many of these services require the use of a credit card, which makes them particularly insidious.

Credit cards (and debit cards in general) are notorious for separating people from their money. As with any other credit card purchase, keep your budget in mind when making online purchases. Purchasing items may be as easy as clicking a mouse, but the money you pay (and the interest rates if you don't pay on time!) are very real.

There are some other factors to consider when making online purchases. The first is quality: often you cannot inspect the goods you buy before making your payment. It helps if your seller has a money-back guarantee on his or her goods. At the very least, you should have some way to report fraudaulent business practices when they occur.

The other expense of online purchasing is shipping. Shipping costs can be astronomical, especially when shipments are made across international borders. One unfortunate fellow living in Canada purchased a dashing (if pricy) red fedora from an American seller, and he ended up paying more in shipping for than he did for the hat! After factoring in shipping costs, many "great deals" on the Internet end up being more expensive than shopping at a local store.

Online auction sites such as E-Bay and Yahoo! Auctions [URL?] deserve special mention. Many people reap much benefit from buying and selling goods through online auctions. In particular, online auction sites are wonderful for purchasing items that are rare and not available locally. At the same time, many many people end up spending far more than they intend making purchases on auction sites. Online auctions are dangerous for the same reason regular in-person auctions are dangerous -- it is easy to get caught up in the spirit of competition, making bids far above what you intended to pay. Furthermore, online auctions run every minute of every day, so you can face temptation around the clock.

There are two key factors to controlling yourself when participating in online auctions: procrastination and stinginess. Like multiplayer video games, online auctions foster a sense of urgency, that feeling that if you do not win the auction for Very Important Item X, then it will disappear forever! Procrastination is your friend here. When you stumble across some item that tempts you, wait a day or two before deciding to put in a bid. Decide whether the item will really benefit your life. Decide how much you are willing to spend on the item and write that number down on a sheet of paper. Determine the shipping and taxes for the item, and subtract that number from the amount you are willing to pay. This will tell you how much you can afford to bid for the item. Write this maximum bid price down. Then (and only) then, put in the bid for your item if you are still interested.

When you are bidding for the item, keep your maximum bid written down in front of you. Do not go over this amount, even if you are "just about" to win the item. Know your limits, and respect them. If you cannot withstand the temptation, then you cannot afford to participate in online auctions.

If you find yourself losing a lot of auctions, then rejoice: you and that auction were not meant to be. If you had been meant to own the item in question, you would have had the money on hand to pay for it. In any case, sooner or later a similar item will come up for auction again. This is the nature of Internet commerce.

The final key to controlling your online expenses is very basic: don't spend money you don't have. Debt may seem enticing at first, but it quickly gets to be a drag, sucking up your cash, constraining your life choices, and adding to

- -4pc - -4pc

your stress levels. Budgets are not fun either, but making a realistic budget and sticking to it far more satisfying than paying 18% interest to pay off some trinket.

Safe Computing On the Internet

It has become a cliche to talk about the "dangers of the Internet". Ever since its inception the Internet has been branded a safe haven for criminals, perverts, and nutcases of all kinds. Government agencies all over the world talk about protecting their citizens from the dangers of the Internet, about implementing censors in public schools and libraries, about controlling what information and access points their citizens have a right to see. [MENTION CHINA AND OTHER COUNTRIES THAT FIREWALL THE INTERNET.] Certainly, dangers do exist, and in some cases legislation may be necessary to prevent some abuses that do occur. Despite this, we do not need government or corporate intervention to protect us from the vast majority of the dangers. What we do need is to educate ourselves and each other, to be aware of the kinds of the dangers that are out there, to develop good habits when using the Internet, and to develop a sense of when a situation might be an Internet scam. People develop these skills over time -- through trial and error, by being exposed to the scams that are out there, and by being lectured by their Internet-using friends when they fall for a scam. In this section, we will give you a head start on your education. We will outline a few of the dangers that occur when using the Internet, and offer advice about how to protect yourself.

The main thing to remember about the Internet is that it makes communication cheap for all. This means that people without a lot of money or power can publish information and tell their stories to a worldwide audience. It also means that people can spread misinformation, and that a few dishonest people can use lies and tricks to take advantage of the gullible and the inexperienced more easily than ever before. Indeed, most Internet scams can be traced to the spread of misinformation: people try to convince you that they have secret information, that they represent an a bank or a Nigerian prince, that they have specially selected you out of all the people in the Internet, that they are sending you a really neat screensaver (that actually contains a virus) out of the goodness of their hearts.

Given all of the wonderful information out there on the Internet, it is not hard to understand why people fall for such lies. How can one distinguish false claims from true ones? There are some basic principles to keep in mind:

1. Don’t believe everything you read. Be skeptical.
2. Do your research before taking any action. Verify sources and stories.
3. If something sounds too good to be true it probably is.
4. Protect your personal information.

These principles will pop up again and again in the following sections. They are your first defence against getting ripped off.

E-mail Safety

Many lies and scams spread via e-mail. E-mail is almost free to send, and e-mail is one of the first applications new users of the Internet try out. E-mail is also an open protocol: anybody who knows your e-mail address can send you a letter, and it is not that uncommon to receive unexpected e-mails from strangers or people with whom you have lost touch. All of these features make e-mail a powerful, accessible form of communication, but they also make it easy for a few malevolent people to spread all sorts of lies, scams, and advertising.

In the following sections we will describe some of the threats that can end up in your INBOX, and how you can recognise and protect yourself against them.

Chain letters

- -4pc - -4pc

Chain letters are messages designed to be spread to as many people as possible. They are usually innocuous -- in most cases the worst they will do is tarnish your reputation when you pass them on. They sometimes contain information about rituals to give you good luck, and sometimes contain hysterical warnings about imminent threats to your computer, your family, or the Internet. Some more sinister chain letters advise you to remove files on your computer to "remove viruses".

Unlike many other e-mail threats, chain letters are usually spread by friends and family members. The distinguishing characteristic of any chain letter is an instruction (or a threat) to forward the chain letter to everybody you know. As soon as you read an e-mail that encourages you to forward that mail to everybody else, alarm bells should go off: this mail is probably a chain letter.

In general, passing on chain letters is a great way to annoy your friends and loved ones. One way to avoid aggravating others is to keep chain letters you receive to yourself. Another way is to verify the authenticity of the letter's message before passing it on. This does take some time and effort, but you will be wasting the time and energy of your friends and loved ones if you forward a hoax on.

Here is a concrete example of a chain letter I received in 2002. It was sent by a well-meaning individual who was subscribed to one of the mailing lists I read. To protect the guilty I have cleaned up the e-mail and removed identifying information:

```
Subject: (fwd) virus

I just found out that we caught a virus through email. It's a
worm, that automatically sends to everyone in your address
book. It is not detected by Norton or McAfee anti-virus
programs. It sits quietly for fourteen days, before damaging
your computer system.

The directions are below, the bad part is that you MUST send
this email to all of the people in your address book because
if you found it in your hard drive, they get it automatically.

1. Go to Start, then Find or Search.
2. In Files/Folder, write the name jdbgmgr.exe
3. Be sure to search in your "C" drive
4. Click Find or Search
5. The virus has a teddy bear logo with the name
jdbgmgr.exe - DO NOT OPEN
6. Right click and delete it
7. Go to the recycle bin and delete it there also.

NOTE: IF YOU FIND THE VIRUS, YOU MUST CONTACT EVERYONE IN YOUR
ADDRESS BOOK WITH THESE DIRECTIONS.
```

As soon as I read this e-mail, I was suspicious. A number of warning signs jumped out at me:

1. The e-mail told us that we "MUST CONTACT EVERYONE" about this virus, because we passed it on to them. This is the classic chain-letter warning.

- -4pc - -4pc

2. The letter claimed that the worm was not detected by the MacAfee or Norton antivirus checkers. I found this hard to believe, because all these checkers would have to do is look for the file `jdbgmgr.exe` on your hard drive to find the virus.

3. The e-mail wanted us to delete some file in our hard drives. As it turns out I would not have had this file on my hard drive -- this e-mail was written for people using Windows, not Linux. Nonetheless, any instruction to delete some arbitrary file is dangerous.

4. The letter did not give the "worm" a name. Antivirus companies usually name the infections that they discover on the Internet, even if they do not know how to stop them.

5. The letter claimed that the virus waited fourteen days before damaging the computer system, but it did not specify what the damage was. Would the virus wipe out my hard drive? Would it cause the computer to crash? Before sending on this warning I wanted to know what the danger was.

As it turns out the chain letter was indeed a hoax. I learned this by visiting the Norton Antivirus [http://www.symantec.com/avcenter/] website, which lists all of the viruses Norton knows about and had a page [http://www.symantec.com/avcenter/venc/data/jdbgmgr.exe.file.hoax.html] devoted to the hoax.

I also found information about the hoax by searching for the term "jdbgmgr.exe" on the Internet. This term (which is the name of the file we were supposed to delete) was distinctive enough that I quickly found websites devoted to this hoax. [MORE INFO. WHAT WEBSITES?] Since I trusted some of these websites already [MORE ABOUT THIS] I convinced myself that the warning was fake.

The person who originally forwarded this e-mail (to over a hundred people!) meant well -- he wanted to protect us from harm and warn us about possible damage to our computers. However, by spending a few minutes verifying the story before sending out the e-mail, he would have saved hundreds of people grief. In this case deleting the file in question caused little harm for most people. However, if the file had been some important system file, many people could have corrupted their Windows installations by following the advice he forwarded. By doing your research you gain credibility, so that if you ever receive *real* warnings you should pass on, your friends and family will believe you when they receive them from you.

One final word about this case study: after I discovered that the e-mail was a hoax, I wrote the sender of this e-mail a response. I told the sender that the warning was false, and I listed some websites documenting the chain letter. I also encouraged the sender to do his research in the future, and listed some websites to consult. Sometimes it is also worth sending this information to the entire list, but in general the original sender of the e-mail should send an apology and retraction if appropriate.

As you develop a critical eye and learn to identify chain letters better, you will be tempted to send scathing angry e-mails to the poor person who passed the chain letter to you. It is wiser to be polite, and to educate the sender so that he or she does not make the same mistake again. All of us start out as new users, and all of us make mistakes from time to time.
[WEBSITES AND RESOURCES FOR EXPOSING SCAMS?]

Spam

Spam refers to unwanted, unsolicited e-mail -- e-mail you do not want and that you did not ask to get. Often, spam is sent by strangers trying either to sell you something or to steal your personal information. Spam is the junk mail of the Internet, and it is a huge problem that threatens the usefulness of e-mail as a means of communication. A huge proportion of e-mail transmitted on the Internet is spam -- at some ISPs [CITE?] over two-thirds of all mail received is spam. On a personal level, a large amount of spam can be overwhelming: some people receive a hundred spams a day. This is what makes spam dangerous: hidden in all the junk mail are messages (sometimes from strangers) that you want to read. It is sometimes tricky to distinguish spams from legitimate mail, so sometimes the real e-mail goes

- -4pc - -4pc
unnoticed and unread. Worse, sometimes you might mistake legitimate e-mail for spam, and delete it or run it through a spam filter (which makes it harder for the same e-mail to get to you next time).

Why is there so much spam on the Internet? There are two reasons spam exists on the Internet: e-mail is cheap to send and e-mail addresses are available on the Internet. Spammers depend on volume to make their profits: a spammer can send a hundred thousand e-mails to the world for a few hundred dollars [CHECK]. If a dozen people respond to the spam and purchase the spammer's product, then the spammer can make a profit. If a hundred people respond, then the profits can be large even though the response rate is tiny.

Being able to send thousands of spams cheaply is of little use unless you have a list of addresses to send the e-mails to, however. This is where the Internet comes in. Spammers and their friends *harvest* addresses that are posted on web pages, to newsgroups, on bulletin boards, and on many other locations on the Internet. Sometimes spammers can buy lists of e-mail addresses from shady businesses. All of these e-mail addresses can be organized into lists, which are then sold on CD-ROM. A spammer buys such a list, and sends the spams of the day to all the addresses on the CD-ROM.

Some of the addresses in a collected list are fake or no longer in use. For this reason, spammers value addresses that are verified as being active e-mail accounts. For this reason some spams are designed to verify e-mail address information: by responding to the e-mail (or in certain circumstances, just by opening it!) you verify that the e-mail reached you, and that your e-mail address is legitimate.

Here is an example spam I have received. There are many, many more: [EXAMPLE SPAM HERE]

Your goals in dealing with spam are threefold:

1. You don't want to let your e-mail address be harvested in the first place.
2. You do not want to verify your e-mail address for spammers (because then you can expect more spam).
3. You want to keep the spam you receive from overwhelming your real e-mail.

Unfortunately, you probably will receive some spam no matter how hard you try to conceal your e-mail address. Furthermore, the longer you keep your e-mail address, the more spam you can expect. How do you deal with it? Here are some simple tips that you can follow to reduce the amount of spam you get, and to deal with spam when you do get it.

- The first and most important rule is to *never respond to spam*. The worst thing you can do is be tempted by the offers for cheap Rolex watches or prescription drugs. Not only does your response support the spammers' business model (thus encouraging them to sends millions of more spams to people all over the world), but many spammers are shady operators. Some of the products they advertise are fake -- they are frauds to get your credit card or banking information. Even if the businesses are legitimate, by responding to a spam you have demonstrated your willingness to support the business of spammers, and you can expect to receive many hundreds of spams in return.

  Some forms of e-mail advertising are acceptable, but spamming is not. Don't support it!

- Some spams contain links to click to "remove you" from their mailing lists. Do not click these links, and do not send response e-mails to the spammers asking them to get you off their lists. By doing so, you just verify that your e-mail address is legitimate, and invite more spams.

- -4pc - -4pc

- Be careful of who you release your e-mail address to, and how your e-mail address shows up on the Internet. In general, you do not want to hand out your e-mail address to anybody you do not trust, and if you do give your e-mail address to an organization, you should expect to receive e-mails from that organization.

  Unfortunately, many services on the Internet require the use of a valid e-mail address, so it is hard to avoid giving your e-mail address away to anybody. For this reason, some people use multiple e-mail addresses. They give one address only to close friends and family. They use a second e-mail address for legitimate bulk communication, such as mailing lists and web forum memberships. Some have more e-mail addresses which they give to organizations they do not trust, or which they use in situations where addresses are easily harvested.

  The advantage of this approach is stability. Your private e-mail address can stay stable -- people will be able to contact you at that address five years from now, and still be able to get to you. The disadvantage is that you have to check multiple e-mail accounts for legitimate mail.

- Some forms of Internet communication -- such as Usenet newsgroups -- are particularly prone to e-mail harvesting. For this reason people who communicate using such services often obscure their e-mail addresses. For example, if a person's e-mail address was: `<mohammad@somewhere.org>` that person might display their address as

  ```
  mohammad at somewhere dot org
  ```

  or

  ```
  mohammadnospam@somewhere.deleteme.org
  ```

  The goal of such tricks is to make it easy for humans to decode the real e-mail address, but to make it difficult for e-mail harvesters (which are usually computer programs) to do so. If you are sufficiently clever, such tricks can be useful. However, people usually provide instructions on how to decode the address in their newsgroup posts, and some techniques (such as adding the word "nospam" somewhere in the e-mail) are common enough that harvesting programs can recognise these e-mails as legitimate addresses. Obscuring your e-mail address can be a fun and useful trick, but it does not guarantee that your e-mail address will not be harvested.

- Spammers have gotten sneakier about verifying e-mail addresses. A favourite trick is to send mails in *HTML*. HTML is a language used to format web pages, so the e-mail you receive looks like a web page, with pretty colours, graphics, and other doodads.

  Many people reject HTML-formatted e-mail on moral grounds: they believe that e-mail is best left as a text-only medium. That aside, HTML e-mail provides a convenient way to check the integrity of your e-mail address. Spammers embed links to images (which are sometimes visible, sometimes not) into the e-mails they send. Each image has a unique identifier associated with the victim's e-mail address. As soon as you open the message, your e-mail client displays the HTML-formatted e-mail by fetching the image from the spammer's *web server*, which gives the spammer two useful pieces of information: that your e-mail account is valid and that you opened the spam e-mail.

  Your best defence against such trickery is to refuse HTML-formatted e-mails. By default, the mail client Sylpheed will not display HTML-formatted e-mail. In general, you want to avoid opening spam e-mails at all.

  A related trick does not require the use of HTML-formatted e-mails. Some e-mail programs support the use of [WHAT?] *send verification*. When you receive the e-mail and open it, your e-mail client sends a response to the

- -4pc - -4pc
  sender that you received the mail. [HOW DO YOU DEAL WITH THIS? DOES SYLPHEED SUPPORT IT? WRITE MORE HERE]

- Maybe the most popular solution to dealing with spam is to filter it out. A *spam filter* is a program that looks through your e-mail and tries to separate the e-mail you want from the e-mail you don't. Many spam filters are able to learn about the kinds of e-mail you receive to make their filtering better.

  Good spam filters can eliminate almost all the spam sent to your INBOX. If you use web-based e-mail, you probably have a spam filter installed already. If you have the option of marking messages as spam to train the filter, you should use it. The e-mail program Sylpheed [DOES NOT HAVE SPAM FILTERING YET! YIKES!]

  The problem with spam filters is that they can be too aggressive, and sometimes they label e-mail you want to keep as spam. Most spam filters sort the mail they think is spam into another folder. In theory you are supposed to go through that folder periodically to look for mail that is not spam. However, this is easier said than done; the best approach I have found to this problem is to look through the spam folder every day.

Eventually, people will find a workable solution to the spam problem. If we are lucky, this solution will keep e-mail cheap and accessible while preventing its abuse. In the meantime, spam is prevalent; your jobs are to prevent it from ruining your e-mail experience and to avoid getting hurt from the truly malicious frauds.
[EXAMPLE OF SPAM] [NEED: SOMETHING ABOUT SPAM FILTERING]

Attachments

In addition to text and HTML, it is possible to send arbitrary files to others via e-mail. These files are called *attachments*. In office environments, people sometimes send each other copies of documents via attachments. At home people sometimes send photographs via attachments. These uses can be useful, but attachments have two big disadvantages:

1. Data files tend to be much larger than regular e-mail messages. This is a problem since many e-mail accounts have limited capacities (known as *account quotas*). Receiving large attachments can quickly use up that quota, overflowing your account. Generally, once your e-mail account has overflowed you will be able to receive no further mail until you clean out your account and make some space. In the meantime anybody who sends you an e-mail message will have that message bounce back.

2. Attachments can be arbitrary files, and that is dangerous. Some malicious people disguise *viruses*, *worms* and other bad programs as screensavers, photographs, and other seemingly-harmless files. Most of these bad programs are targeted at Windows users; most of them will not run in Linux, and thus cannot hurt your data or your computer. However, there is no good reason why Linux will not be targeted by similar threats in the future.

- -4pc - -4pc

Both of these situations result in aggravation and lost data, and Windows users are particularly vulnerable to the threats attachments pose. For these reasons, you need to treat attachments with caution. If you are thinking of sending attachments to others, a good strategy is to send two e-mails. In the first e-mail, you tell your recipients the attachment you plan to send and how large it is. This will tell them to prepare for your attachment (or to send you a response asking you not to overflow their accounts by sending the mail). The second e-mail you send should contain the attached data. [PICTURE?]

Similarly, you should be cautious of opening any unexpected attachments. If you receive an unexpected attachment from a stranger, you should be very suspicious. If you cannot figure out the reason somebody would send you the attachment, you are better off not opening it. If you trust that the person who sent you the attachment is legitimate (as opposed to a spammer using a fake e-mail address) then you might ask the sender what the attachment is and why it was sent to you. If you don't trust that the person is legitimate then you should discard both the e-mail and the attachment without opening it.

Unfortunately, you cannot even trust attachments sent by family or friends. Some *worms* are spread through e-mail. Once a victim e-mail account becomes infected the worm sends copies of itself to everybody on the victim's addressbook. If you are on the addressbook of somebody who has been infected, you will receive an e-mail containing an attachment, which in turn contains the worm. This is yet another reason to ask whether the attachments you receive are legitimate when you get something unexpected. [ARE THESE VIRUSES OR WORMS?] [EXAMPLE OF MALICIOUS ATTACHMENT]

In addition to notifying senders of unexpected attachments, there are a few other ways people protect themselves. One is by running attachments through virus scanners. Currently we do not ship our computers with virus scanners, largely because there are currently (in 2004) almost no viruses targetted at Linux. [SO WHERE CAN PEOPLE TURN?]

The other step that people take is to verify that the attachment's name matches its type: that JPEG picture files are actually pictures, that ZIP files are actually archives, and so on. The **file** command is very useful for determining the true nature of a file. [WHERE? PUT THIS IN SYLPHEED]

Although attachments can be useful, there are generally better ways to transmit data files to others. If you have a website and the data is not confidential, you can post the data to your website, and then send people the location of the file via e-mail. If your data is more confidential then you could *encrypt* the data, or use other file transfer methods. If your data is confidential then you may not want to transfer it via e-mail anyways: e-mail is notorious for being less private than many people think.

The rule of thumb when dealing with attachments is that nobody should ever receive an attachment unexpectedly. If you practice this rule both when sending and receiving attachments, you reduce the risk considerably.

Phishing

Phishing is a form of fraud where unscrupulous people try to obtain your personal and financial information by posing as businesses or other authorities. The term is derived from "fishing" -- the scammers cast their nets of convincing e-mails hoping to catch unwary e-mail users.

Phishing scams are scary because they can be fairly convincing. Consider the following example:

```
From: "eBay" <Billing@eBay.com>
Subject: eBay Account Verification
Reply-To: Billing@eBay.com

Dear valued eBay member,
```

```
- -4pc - -4pc

    It has come to our attention that your eBay Billing Information
    records are out of date. That requires you to update the Billing
    Information If you could please take 5-10 minutes out of your
    online experience and update your billing records, you will not
    run into any future problems with eBay's online service.
    However, failure to update your records will result in account
    termination.  Please update your records in maximum 24 hours.
    Once you have updated your account records, your eBay session
    will not be interrupted and will continue as normal. Failure to
    update will result in cancellation of service, Terms of Service
    (TOS) violations or future billing problems.

    Please click here to update your billing records.
    http://www.eBay.com/verification/%?6488820019

    Thank you for your time!
    Marry Kimmel,
    eBay Billing Department team.

    As outlined in our User Agreement, eBay will periodically send
    you information about site changes and enhancements. Visit our
    Privacy Policy and User Agreement if you have any questions.

    Copyright 2004 eBay Inc. All Rights Reserved.  Designated
    trademarks and brands are the property of their respective
    owners.  eBay and the eBay logo are trademarks of eBay Inc
```

At first glance, this e-mail looks convincing, even professional. If opened in a graphical HTML-capable mail reader, the e-mail looks very much like the real eBay site, because the scammers stole graphics and text from that website to include in their e-mail.

This mail becomes even more convincing when you consider that eBay (a popular online auction site) communicates with its members via e-mail.

Nonetheless, this e-mail is not legitimate. It is a scam. The link labelled http://www.eBay.com/verification/%?6488820019 actually directs you to http://64.156.26.4/*goldtraders.com/httpdocs/eBay/. This website (which also looks legitimate) contains a form that allows you to enter billing information such as your credit card information. Once you have submitted your information it goes straight to the scammers, who can then make purchases and bill them to you.

Other common phishing scams include banks and online payment services such as PayPal.

How can you protect yourself against such scams? Your first defence is skepticism. If you are not a member of the service in question, then you clearly should not be receiving it. If you are a member of the service, then your job becomes harder. A good first response is to ignore the e-mail. If it turns out that your account is really in danger, you should receive followup e-mails.

Your second defence is to understand that e-mail is an insecure medium. It is easy to hide your identity and pretend to be somebody else. For that reason, no reputable online business will ask you to *submit* sensitive information by e-mail. If any e-mail asks you for your password, or credit card information then that e-mail is almost certainly a scam. In fact, alarm bells should go off if you receive any e-mail that asks you to submit personal information of any kind.

- -4pc - -4pc
Your third defence is to avoid the use of HTML e-mail. The scam above only works if the user clicks on the link in the webpage. If a user copy and pastes that link into a web browser, the scam website will not come up.

Before taking any action, you need to do your homework. In this case, this would include the folliowing:

- Visiting the eBay site and looking for their e-mail policies. This should tell you what kind of e-mails you can expect to receive from them.
- Searching the Internet for phrases unique to this e-mail. If you searched for "Marry Kimmel" or "your eBay Billing Information records are out of date" or "Subject: eBay Account Verification" then you might find some information about the e-mail you received.
- You might consider contacting eBay by e-mail to ask whether the e-mail is legitimate. When doing this, do not respond to the address contained in the e-mail -- that address could be fake. Instead, visit the eBay site and look for their support contact, and send an e-mail to that address.

The e-mail itself contains some clues that should alert you that it may be a scam:

1. The e-mail never identifies you personally. If the business knew your billing information was out of date then it certainly should know who you are, and they should address you by your name or userid, not "valued eBay member". If the e-mail does not identify you using the information you provided when you signed up for the service, it is almost certainly fake.
2. The name "Marry" is kind of odd. It is possible that a person named "Marry Kimmel" works at eBay, but it is more likely that the scammers were trying to mislead you into thinking the e-mail was sent by "Mary Kimmel". Also note that Marry does not provide any contact info for him/herself.
3. That the sentence "That requires you to update the Billing Information" does not end with a period. This may seem like an overly-picky point, but it is not: grammar and spelling errors are an excellent way to spot illegitimate e-mails. Any reasonably-sized corporation is going to make sure that its public relations staff are capable of basic grammar and spelling, or at the very least that they run a spell-checker through their e-mails before they send them. It used to be the case that almost every scam e-mail I received gave itself away because the scammer couldn’t spell. This e-mail is much more convincing, but the missing punctuation should arouse your suspicions.

- -4pc - -4pc

4. The following sentences are very strange:

```
If you could please take 5-10 minutes out of your
online experience and update your billing
records, you will not run into any future problems with eBay's
online service. However, failure to update your records
will result in account termination.
```

   The phrase "If you could please take 5-10 minutes out of your online experience" rings false. It does not sound like corporate language. It sounds as if it is asking you for a favour.

   Contrast this with the next sentence, which wields the big stick: if you do not do this favour, your account will be terminated! This dire consequence immediately exposes the scam. For the crime of not updating your information, eBay will cancel your service, thus losing your business and putting you into a bad mood? Furthermore, they will not send you further warnings, and provide no deadline for updating this information? Given that eBay wants to keep your business, this makes no sense.

   In general, you should be suspicious of e-mails that threaten you with suspension of your accounts -- especially if they don't identify you by name.

5. The biggest hint in the e-mail is that the link advertised in the e-mail does not match the actual link, and the actual link does not go to a website on ebay.com. Any link that redirects you to a website location that does not match the website of the service in question is an instant tip-off.

   Note, however, that some scammers are becoming clever about this as well. For example, a common Citibank scam points you to a website on citibank.com, which was run by the scammers. The actual Citibank site is citi.com, but you might not recognise that immediately.

The key to spotting fraud in these e-mails is to wait and use your head. Pause, take a deep breath, and read the e-mail carefully and critically. Take some time to think about the mail before responding to it. Check with computer-savvy friends to get their opinions as to whether the e-mail is fake or not. Look on the Internet to find information about whether the e-mails are scams. If you are still unsure, contact a real person (preferably by phone) working at the company and verify that there really is some problem with your account. Scammers depend upon your ignorance to make their money. With a bit of vigilance you can do a pretty good job of protecting yourself.

Web Safety

The World Wide Web offers many of the same advantages as e-mail, and many of the same threats. The Web is easily accessible to anybody with Internet access, and it is fast and cheap for individuals to publish information on Internet servers. These are great strengths: for the first time in history, we can gather information and listen to stories published by ordinary people from all over the world. At the same time, the Internet makes it easy for people to publish lies and misinformation. As always, protecting yourself against such threats involves being aware of the threats, being skeptical, and using good judgement.

In the following sections we will outline a few dangers of the Internet. Much of the advice outlined in the section on e-mail [REF] applies when surfing the Web as well.

Protecting Your Personal Information

Many, many websites ask for personal information in order for you to access their services. Pretty much any online service will require you to enter a valid e-mail address. Some services go further, demanding that you enter phone

- -4pc - -4pc
numbers, names, home addresses, or financial information. Sometimes giving up this information makes sense, and sometimes it does not. For example, it may be reasonable to give credit card information to a business when making a purchase, but giving the same information to an e-mail update service is just asking for trouble.

When you have submitted your personal information, it may be put to many uses. Usually it will be filed in a database somewhere and stored in perpetuity. Some businesses collect information about their customers and use it to gather marketing information. Often organizations will use e-mail addresses to shower you with updates and product offers, and some organizations will sell lists of contact information to others.

Once you have released your personal information, it can be very difficult to regain your privacy. I once made the mistake of releasing my e-mail address to an online music service. I am still receiving spam e-mails from that service and the organizations to which it sold my e-mail address.

It is well worth carefully considering the information you are willing to share with others. Here are some questions to ask when somebody is asking you for some information:

- What information do they require me to submit?
- Why are they asking me for this information? Is this information necessary to provide me with the advertised service?
- Is there a documented way for me to get the service to forget my personal information at some later date?
- How will the service use my personal information? Have they promised to respect my privacy rights in any way? How can I hold them to their promises?
- How badly do I require the use of this service? Can I get by without using it? Are there better alternatives available?
- What is the reputation of this service provider? Do they have a history of abusing personal information? Do they belong to organizations (such as the Better Business Bureau) which offer some assurance that they behave well?

Many websites publish *privacy policies* which provide some of the answers to these questions. A privacy policy should tell you what information the website collects, whether they identify you personally (and if so, how), and how they use the information they collect about you. Keep in mind that a privacy policy alone does not protect you; without some way to enforce the policy it is nothing more than a set of promises.

Cookies

Many websites use *cookies* to keep track of information about you. A cookie is some information issued by a website and stored on your computer for retrieval later. Some cookies keep track of how many times you have visited the site. Others keep track of your status on the website -- if you ever log into a website, a cookie is almost always used to keep track of your identity and your login status. When making online purchases, cookies can keep track of who you are and what purchases you have made. The web search engine Google keeps track of all the searches you ever made using cookies. In general, cookies transmit information to website about you and your surfing habits.

Sometimes, cookies are necessary to use a website properly. For example, it is difficult to log into online mail services without cookies. In some ways, however, cookies are a privacy threat. Even when they do not identify you by name or e-mail address, they can collect a wide range of information that reveals a lot about your online habits.

Unfortunately, it is hard to restrict the use of cookies when using the Internet. Most web browsers provide ways to manage cookies. They allow you to see what cookies are stored on your computer, and give you the opportunity to delete them. Some browsers allow you to confirm each cookie you receive, or to reject cookies completely. Both of these approaches have problems. Confirming each individual cookie quickly becomes aggravating because so many websites use cookies. (It is an excellent exercise to carry out if you want to get a sense of how widely cookies are

- -4pc - -4pc
used, however.) Rejecting all cookies is better for your privacy, but then you will be unable to use certain websites. A good middle ground is to disable cookies by default, but then enable cookies coming from the websites you need to login to. [WHAT DO OUR WEB BROWSERS DO? HOW DO THEY DO IT? PROVIDE INSTRUCTIONS AND A TUTORIAL.]

Security Certificates

[LEARN ABOUT THESE. WHAT ARE THE DANGER SIGNS?]

Downloads

The Internet makes it easy to download all sorts of goodies to your hard drive. In many cases these goodies are harmless, but you should treat Internet downloads with the same caution you treat e-mail attachments.

Some web browsers ask you whether you wish to download or save strange files off the Internet. [SCREENSHOT] You almost always want to save these files to your hard drive instead of running them from your web browser.

One point to remember from a Linux perspective is that many executable programs you can download will not run in Linux. Sometimes this is a pain but often it is a blessing in disguise. Executable programs are more likely to contain viruses and other nasties than other things you download from the Internet.

As always, use your judgement. If you are not sure where an Internet download came from or what it is for, you may want to think twice about running or viewing it on your computer.

Protecting Your Reputation

Many people overestimate their anonymity on the Internet. They think that -- so long as they never reveal their real names -- the things they say cannot be traced back to them. In many cases, the opposite is true: when you use the Internet you leave traces everywhere you go. The identity of your computer is logged at each Internet site you visit. Cookies keep track of your surfing habits. E-mails contain header information that can be traced back to your *ISP* and ultimately back to you.

The other aspect of the Internet that catches people by surprise is that information posted to the Internet stays there permanently. Search engines make copies of websites (called *cached copies*) to help them index information. People run archives to keep track of old websites. Newsgroup posts are archived in perpetuity. A single embarrassing photograph might be downloaded by thousands of Internet users, and reside on their hard drives for years and years. In general, information just does not disappear.

For these reasons, when using the Internet I make the following two assumptions:

1. Nothing you say or do is truly anonymous. Everything and anything can be traced back to you.
2. Everything you say and do on the Internet is part of the permanent public record.

- -4pc - -4pc

These assumptions are less paranoid than they sound. Although it is possible to make your communications mostly secure, doing so is not a trivial task. Although it is possible to prohibit search engines from accessing the data you post online, you have no guarantees that somebody has not made a copy of the data and is distributing it to others. Furthermore, in the future it will become easier -- not harder -- to associate people with their data. It is totally conceivable that employers will carry out online background checks before making hiring decisions. This could affect your employability either way; an online history that demonstrates you are a good worker and a team player could make it easier for you to get hired, while an online history that gives employers a negative impression of you could make your life difficult. For this reason alone it is worth being careful about your online activities.

Even ignoring the actions of hypothetical employers, it makes sense to behave well on the Internet. Internet communities are like schoolyard playgrounds; if you earn a reputation for being a bully, nobody will want to be your friend.

The Internet provides many opportunities for learning and sharing knowledge with others, and for socializing with other people who share your values and interests. It would be a shame to let the fear of your online record prohibit you from taking advantage of what the Internet has to offer. The thing to remember is that you are as accountable for your actions on the Internet as you are anywhere else; you should be prepared to stand behind what you do and say to the same degree.

Here are some tips and pointers to help you maintain your online reputation.

- Be careful how you interact with people. If you would not insult a person face-to-face, do not do so online. Be especially careful when you are angry; if you make some rash or hurtful statement online, it is not easy to retract your words. The best policy to take when you are angry is to step away from your keyboard for a while and calm down.

  Civility is always the best policy when conducting online communications. People enjoy calling each other names online, but such behaviour ruins the atmosphere for everybody except the combatants.

- Respect the rules and conventions of the communities in which you participate. If people tend to use proper English grammar in their communications, you should as well. If people avoid profanity or explicitly sexual phrasing, you should as well. If the group is focussed on a particular topic, you should be wary of posting off-topic material. If you post something that others find offensive or inappropriate, be humble and respect the dictates of the community.

  This advice is particularly important when you are first joining an online community. After you have participated for some time you get a feel for what conventions you can break and which ones you can’t.

  An excellent rule of thumb is to "lurk before you post". *Lurking* is the practice of paying attention to an online community, but not posting anything. Lurking is a good way to learn about the rules and social conventions of the community before contributing to it.

- -4pc - -4pc

- Some people take great delight in *trolling* -- visiting online communities and deliberately aggravating the members there. For example, some goofballs enjoy visiting vegatarianism sites and harassing the members there about their hypocritical eating habits. They insult the members and ask inflammatory questions, then hide behind their rights to free speech when confronted about their rude behaviour.

  These goofballs don't really care about the answers to their questions. They are not driven by curiosity or a desire to help their fellows. Rather, they are interested in making people upset for ideological reasons. Don't be one of those goofballs. Disagreeing with the views of others is no crime, and dissenting points of view can add a lot of quality to online discussions. However, confrontational behaviour rarely changes people's minds. If you wish to express your dissent, there are psychologically more effective ways to do so than trolling.

- Disagreement and miscommunication happen frequently online. Sarcasm and humour are often misinterpreted as insults. When people misinterpret your words, be gracious. Make amends and don't be afraid to apologise.

- Accept that sometimes people will disagree on fundamental principles, and refuse to let those disagreements get in the way of your interactions together. It is not always easy to get along with people who do not share your values, but in the end it can be very rewarding.

- If you feel the need to chastise or correct other people who are misbehaving, it is worthwhile doing so using more private forms of communication. For example, if you participate on a web forum and somebody is behaving in appropriate ways, you might consider sending that member a private e-mail or instant message to (civilly!) let the person know that their behaviour is inappropriate. Chastising the person on the board will sour the atmosphere of the community and will often make the offending member hostile. Nobody likes getting criticised in public.

- Keep confidential information confidential. If you spill secrets others have entrusted to you, then you will gain a reputation as one who cannot be trusted with secrets. This is particularly important when dealing with sensitive corporate information.

  On the other hand, sometimes you need to reveal sensitive information. For example, keeping certain secrets might endanger the well-being of others. In such cases you should be very careful of how and to whom you reveal this information.

- Stand behind what you say and do. If you would be ashamed of admitting your online actions to others, then you should reconsider your actions (and maybe your shame).

- Be careful about spreading misinformation. If you are unsure about whether some fact or figure is true, admit it. Better yet, check your facts before you spread them. The Internet makes it so easy to find information that it is rapidly becoming expected that people back up the claims they make.

- -4pc - -4pc

You would be surprised at how frequently people forget some of these common sense rules. It is especially easy to get into trouble when one is involved in a heated, emotional debate. At these times, be particularly careful of your actions.

Avoiding Online Breakins

The Internet is just a network of computers connected together. When you go online to surf the Internet or read your e-mail, your computer becomes part of that network. Your computer is intended to be a *client* -- a computer that retrieves data from other computers. For the most part it is not intended to act as a *server* -- a computer that provides information to other computers. In fact, it is possible to reconfigure the software on your computer to act as a server.

One danger of being online is that people use your computer as a server when you don't want them to. These people gain access to your computer, and then take over its resources. They might run strange programs on your CPU. They might store illegal data on your hard drive. Or they might use your computer to commit all sorts of shady activities, such as relaying spam to other computers. People who break into other people's computers are popularly known as *hackers*, but many computer geeks discourage the use of "hacker" to mean "computer criminal". They prefer the term *cracker*, because crackers try to "crack" the security on your computer system. Cracking has always been a problem on the Internet, but with the increased use of high-speed Internet access it is becoming even more widespread. Unfortunately, computers running Linux are generally more attractive targets than computers running Windows.

The field of computer security is huge, and securing computers from crackers is difficult work that requires vast knowledge and focussed paranoia. No computer on the Internet is 100 percent secure from cracking; the best we can hope for is to keep the risk of cracking low.

We have preconfigured your computer to eliminate some risks. We have included a firewall and limited the ways others can access your machine. [LINK TO MORE DETAILED INFORMATION?] In addition, there are some good habits you can develop to keep the risk low:

- Disconnect your computer from the Internet when you are not using it. [HOW?] An easy way to do this is to unplug the network cable from your machine when you are not using the Internet. Another method is to physically power down your machine when you are finished using it. A third way requires the use of the superuser account: for details see [WHERE?]

  The less your computer is connected to the Internet, the fewer opportunities others will have to find flaws in your machine's security.

  Some people advocate leaving your computer on at all times to increase the computer's lifespan. I disagree with this position. In addition to wasting power it leaves your computer vulnerable to blackouts and power surges.

- A common source of vulnerabilities are software flaws. When a security-related flaw is discovered in software, the Debian organization (which provides the software on your computer) provides an updated version of the software that fixes the flaw. Installing these updates is a straightforward process, but it requires use of the *superuser* account. Some people check for updates every day; checking for updates every few weeks is probably acceptable. For instructions on how to upgrade the software on your computer, see [WHERE?]

- -4pc - -4pc

Ergonomic Considerations

In recent years people have become concerned with the long-term effects of computer use on people's health. It is well known that computers can contribute to sedentary lifestyles. Sitting in front of a computer for hours each day is an excellent way to avoid exercise, which in turn promotes obesity and encourages serious health problems such as strokes and heart attacks. More recently, computers have been blamed for an increase in *repetitive stress injuries* (RSIa). Perhaps the best-known RSI is *carpal tunnel syndrome*, but computers have been associated with other RSIs as well.

Repetitive stress injuries can be very serious. They can lead to a loss of strength in the hands and wrists, making it difficult and painful to carry out routine daily tasks. In addition, RSIs often make it impossible to work at a computer for any length of time, which is a disaster to programmers and others who depend on using computers to make their livings. Some RSIs can be treated, but recovery times can be long and frustrating -- one young, healthy fellow I know suffered an RSI in his wrists, and it took him well over two years to recover.

People have turned to the field of *ergonomics* to try and understand repetitive stress injuries and their causes better. Ergonomics researchers study the relationship between people and their working environments; and a good deal of research has gone into the ways people injure themselves by using computers. Currently, the consensus seems to be that many repetitive stress injuries can be avoided by avoiding positions and situations that stress the body. [CAN I BACK THIS STATEMENT UP?]

There are a number of steps you can take to keep your body healthy when using computers. You can buy devices such as contoured chairs [PICTURE] and ergonomic keyboards [PICTURE] suitable for touch-typists. You can position your existing computer equipment to reduce the strain on your arms, wrists, neck, and back. You can also develop good habits in your computer use, being conscious of your body and the amount of uninterrupted time you spend on your computer.

Here are some basic guidelines. For more information, see [WHERE?].

- Maybe the most important tip is to pay attention to your body. If you notice that certain body parts ache after using your computer for a while, you are putting your body at risk, and you need to change the way you use your computer.

  For example, after years of two-finger typing I decided to finally learn touch-typing. I used the gtypist program to develop my skills, and sure enough after some practice my speed and accuracy began to improve. However, I noticed that after an hour of touch-typing my finger joints started to ache. I never diagnosed the problem completely, but I did notice that I positioned my hands differently when touch-typing than when two-finger typing, and that my hands did not ache when two-finger typing. Fearing for my wrists, I abandoned my touch-typing adventure and went back to two-finger typing. Other good solutions might have been to invest in an ergonomic keyboard or to figure out how to improve my touch-typing position. The worst solution I could have taken would have been to ignore the problem. The increased speed and accuracy I would have gained by learning to touch-type efficiently was not worth the loss of my hand strength.

  The take-home lesson of this story is simple: if something hurts, pay attention.

- -4pc - -4pc

- It is a bad idea to sit at your computer continuously for several hours without taking a break. Sitting in the same position for hours on end is not good for your body. Some posters at my workplace recommend taking a five-minute break to stretch after every hour of sitting at the computer. After sitting at the computer for a while, I often get out of my chair, stretch my back and neck, and walk around before sitting down again.

- If you start noticing recurring pains when using your computer, you should see a doctor. As with any other illness, it is much better to get RSIs diagnosed and treated earlier in their development than waiting for too long and dealing with even more severe consequences.

- -4pc - -4pc

Chapter 4. Finding Documentation and Help

We like to fool ourselves into believing that computers are intuitive devices that are easy to use and learn. The truth is that computers are complicated tools, and it takes practice, patience and information to use these tools proficiently.

One of the most frustrating tasks new users face is finding out how to find information. A lot of documentation comes pre-installed on your computer. You can also find documentation on the Internet if you know how to search for it effectively. When written documentation is insufficient, electronic and real-life communities exist where you can find other people willing to help answer your questions. This chapter surveys these options, letting you know what documentation is available and how to access it.

Local Documentation

The documentation on your system does not come in a single format. Some documentation forms you will run into include manual pages, GNU info files, PDF documents, HTML files, and in-application help. For the most part, you use different tools to access each of these formats. In this section we will describe the tools used to access helpfiles on your system, and the types (and quality) of documentation you can expect to find.

For the most part, we will list resources that are friendlier to new users first.

A word of caution: computer technology changes quickly, and one of the weaknesses of open source software (and of software in general) is that documentation is not always kept up to date. Some of the documentation you run across will refer to versions of software older than what is installed on your system. Some documentation will not apply to your situation at all. Having said that, even older documentation is often a good resource in learning the basics of a new program.

In-Application Help

In-application help [UGH. YUCKY TERM.] refers to the help that you can access from the applications you run. In-application help is sometimes referred to as *online help*, but you usually do not need to be online (that is, connected to the Internet) to access this help.

Tooltips are your first source of information about a running program. Tooltips are small information boxes that pop up when you hover your mouse over different icons on the screen. You can often learn a lot about what a program does just by exploring the screen and reading tooltips. Similarly, you can learn about the different options available in your application of choice by exploring the drop-down menus. It is often possible to figure out the basics of a new and unfamiliar program by exploring the program menus alone.

Some applications come with manuals you can access while running the program. If your application has drop-down menus, look in the Help menu. A few sophisticated applications offer interactive help, where you can type in a question and search for information about the tasks you wish to accomplish.

WCLP Help

The WCLP Help option of the IceWM menu opens an HTML document that lists different help resources available on your computer. Included are links to manuals for some of the major applications installed on your computer, links to the HOWTOs, and resources on the Internet that you might find helpful. Many of the documents listed in this chapter are accessible via the WCLP help document.

- -4pc - -4pc

HTML Documentation

Many of the Debian packages on your system come with documentation in HTML format. If the dhelp package is installed (which is usually the case), you can access this documentation by using your web browser to view the contents of `/usr/share/doc/HTML/index.html` If for some reason the dhelp application is not installed, you can find some documentation by browsing the subdirectories in `/usr/share/doc/`.

The Debian Reference Manual

Of special interest is the *Debian Reference Manual*. Unless you have a very small hard drive, you can access this document at Programs->Help->Debian Reference of the IceWM menu.

The Debian Reference Manual contains a wealth of information about configuring and administrating Debian-based computers (which includes WCLP systems such as your computer).

This manual is aimed at non-programmers, but it assumes that you are comfortable using the command-line shell. If you are brand new to computers or brand new to Linux, you may find the entries too brief and technical to be useful. Nonetheless, it is a good place to start when learning about a topic.

Manual Pages and TkMan

The *manual page* (or *manpage*) is one of the oldest and best-supported documentation formats for Linux (and UNIX) computers. Manual pages have an unfortunate reputation for being hard to work with, but the TkMan application makes them easier to work with.

In the Debian distribution, all programs must have a manual page, but in most cases the manual page is oriented towards command-line usage. This is useful if you are programming shell scripts, customizing your startup environment or administrating your computer. Some manual pages also document keystrokes or menu options, and several contain references to related documentation and programs. However, manual pages for graphical applications such as Abiword or Dillo tend to be sparse. In these cases it is more useful to look for other documentation specific to the application you are interested in.

The command line tools to access manual pages are **man** and **apropos**. The **man** command displays a manual page given a name, and the **apropos** command searches for keywords so that you can find the name of a command to accomplish some task. For example, to find all the programs and applications that contain the word "icewm" in their title or short description, you would type

```
apropos icewm
```

in a command-line shell.

The TkMan application is a powerful graphical application for viewing and searching manual pages. You can find TkMan in Programs->Help of the IceWM menu.

Upon starting up, TkMan displays its user manual. You will likely see some warnings about "whatis" files at the top of the screen, but these are harmless. [SCREENSHOT]

- -4pc - -4pc

Above the main display are a number of buttons and an information line, which describes the functions of each button when you click it. The TkMan manual describes the buttons and other program controls in detail, so we will briefly describe the main features.

Page Outlines

The top left button has a picture of a page on it. When you click and hold this button it will display the sections available in the current page. You can jump to any of these sections by clicking it.

Page Highlighting

The button labelled Hi is used to control highlighting in the manual page. If you find yourself referring to a specific passage of a manual page frequently, you might highlight the beginning of that section: select the text you want with the mouse, and press the Hi button: the text should turn yellow. [SCREENSHOT] The highlighted section appears in the highlighting menu, which you access by clicking and holding the Hi button.

By accessing the highlighting menu, you can quickly access frequently-accessed passages in your manual pages.

Displaying and searching pages

The man button and the text entry box to the right are used to display and search for manual pages. If you know the name of the manual page to display, type the name in the text box and click the Man button. If you want to search for keywords in manual pages, enter the search term in the text box, then click and hold the Man button to display the Apropos entry. This will carry out an **apropos** keyword search and display the results in the main window. [SCREENSHOT]

The button to the right of the text entry box shows the history of pages you have visited; it reveals a menu of recently-accessed pages when you click and hold the button.

The button marked x adds and removes bookmarks of pages. If you frequently access a certain page, you can visit the page and click on this button (which will become a +). This will add the manual page to the list of your bookmarks.

You can also search for text within a page. To do this, type the text to search in the textbox at the bottom of the screen, and then press the Search button. Use the up and down arrow buttons to visit the search results.

Displaying volumes

The Volumes button shows different categories of manual pages. Traditionally manual pages are organized into sections, where each section is reserved for different purposes. For example, section 1 documents general commands, and section 5 documents file formats.

In reading Linux documentation, you might run into commands written with a number after them: for example, somebody may reference `aptitude(1)` . The number in parentheses refers to the volume where the manual page can be found. Occasionally two manual pages will share a name, but be in different sections. In this case TkMan will display a button labelled ALSO, which will allow you to choose which version of the manual page to read.

By clicking and holding the Volumes button, you can visit manual pages by section. This can be useful if you would like to browse the manual pages available, or if an **apropos** search reveals nothing.

From the Volumes menu, you can also access the most recently accessed pages, pages with highlights, and all the pages you have visited in the past. [SCREENSHOT]

Info Pages and TkInfo

- -4pc - -4pc

In addition to HTML and the manual page format, some program manuals are written in a format called *Info*, which organizes informations into a hierarchy. You can browse the info documentation installed on your computer with the program TkInfo, which can be found in the Programs->Help section of the IceWM menu.

The first screen TkInfo displays is a list of all the info manuals installed on your computer. [SCREENSHOT] Following links takes you one level deeper in the hierarchy, and clicking the Up button takes you one level higher. The Last button behaves like a "Back" button in a web browser: it takes you back to the page you visited previously.

The Next and Previous buttons allow you to read info files section by section. Sometimes this form of navigation gets confusing; if you get lost in the hierarchy your best action is to use the Up until you get to a section you recognise.

Info files are generally used for in-depth manuals than manual pages, but most of the manuals are oriented towards command-line usage.

Documentation directories

Documentation for the Debian packages installed on your system are stored in the directory `/usr/share/doc`. Inside this directory you will find a subdirectory for each installed package. For example, the icewm application comes from the packages `icewm` and `icewm-common`, so you might look at the contents of `/usr/share/doc/icewm-common` for documentation.

Things to look for in the `/usr/share/doc/` subdirectories include:

- The file `README.Debian`, which contains notes specific to this Debian version of the software. Sometimes this file is used to document program oddities or problems.
- The directory `examples`, which often contains sample configuration files (or usage files) for the application.
- A file named `FAQ`, which stands for "Frequently Asked Questions". FAQ files will often contain information about problems users often run into, so they can be good for troubleshooting.
- Documentation in PDF format which can be read using xpdf, or Postscript (.ps) format which can be read using gv.
- A directory named `html`, which contains documentation in HTML format. Note that the dhelp will put links to these documentation directories in `/usr/share/doc/HTML/index.html` automatically.

- -4pc - -4pc

Documents in /usr/share/doc/ are often compressed using **gzip**. If your file manager does not open these files correctly, you may have to use the **zless** command to read them. [REF zless]

HOWTOs

A HOWTO is a tutorial written about some particular topic, such as setting command-line prompts or setting up a firewall. HOWTO documents are one of the best ways to learn about your computer and how it works. They are usually written by individual users who want to share what they have learned about some task. Although many of the topics covered by HOWTOs are aimed at intermediate or advanced users, even beginning users can learn about the kinds of things it is possible to do with Linux and open source software.

Unless you have a very small hard drive, you will find a collection of HOWTO documents in the /usr/share/doc/HOWTO/en-txt/ directory. These HOWTOs are text files compressed using **gzip**. Your web browser or XFE may uncompress and display these files when you double-click on them. If they don't, then you can also read these documents from the command line, using the **zless** command. For example, to read the Reading-List-HOWTO, open a command-line window and type the following:

```
cd /usr/share/doc/HOWTO/en-txt/
zless Reading-List-HOWTO.gz
```

For more information about using **zless** and the command line, see [WHERE?]

For some reason, HOWTOs are more prone to being out-of-date than other documentation. Check the publication date of the HOWTO carefully before trusting its contents. Also note that some aspects of HOWTOs (such as compiling code from source) usually do not apply to you, since you can install Debian packages.

Other Help Resources

One of the great strengths of the open source movement is its grassroots tradition. Historically, few corporate or institutional supports have been offered for Linux software. Hardware manufacturers did not provide Linux drivers, Internet Service Providers did not support Internet access to computers running Linux, and few businesses sold support services to users wishing to run Linux. As a result, Linux users (and open source users in general) took on the responsibility of helping their movement grow. Although the popularity (and business support) of Linux has grown, this tradition continues. Linux users organize user groups, run online forums, and continue to write and share software.

The grassroots nature of open source software influences the way you can most effectively access help for your computer. Instead of calling a technical support line, you might post a message to a web forum or mailing list. Instead of looking to centralized knowledge bases for troubleshooting information, you might conduct a web search to see whether other users have run into the same problem you have.

Although it is certainly possible to pay others to support and maintain your computer, many people find that freely-available resources are sufficient to keep their systems running. This "free" support has a cost, however: in order for grassroots support to work regular users have to spend time and energy to support each other. As your Linux experience and knowledge grows, you may find opportunities to support other Linux users. Embracing these opportunities is an excellent way to "pay forward" the support and knowledge other Linux users have donated to your benefit.

- -4pc - -4pc

In this section we will describe some of the help and documentation resources available. Many of these resources require Internet access, but a few are available offline.

Linux User Groups

A *Linux User Group* (LUG) is made of users who discuss, provide support for, and discuss Linux. LUGs are usually locally-based. Often a LUG will hold regular in-person meetings where people have discussions, make presentations, and support fellow users. Many LUGs also run online discussion and support forums, such as mailing lists or web forums.

Joining a LUG is a great way to meet other people who use Linux. You can make new friends and contacts, share information with people, and get personalized help with your problems. Participating in a LUG can also help you learn more about Linux in particular and computers in general.

There may already be a LUG operating in your area. To find it, you might ask the person who installed the software on your computer. You might also search on the World Wide Web, using the search term "LUG" and then your city, town, or geographic area. Finally, you might consult an Internet directory of worldwide LUGs. Two websites that maintain LUG lists are http://www.linux.org/groups and http://lugww.counter.li.org .

If no LUG exists in your area, you might consider organizing your own. The *User-Group-HOWTO* is good guide to organizing and running a LUG. You can access it on the Internet at http://en.tdlp.org/HOWTO/User-Group-HOWTO . [WHY IS THIS NOT IN THE HOWTO DIRECTORY?]

Web Searches

Just as the Internet has made it easy for people to distribute software across the globe, the Internet has made it easy for people to search for information across the globe. In the open source world, the humble web search is by far the most common form of technical support. In fact, in many circles it is considered rude to ask a question before spending time searching for the answer on the Internet first. It is well worth learning how to conduct effective web searches for this reason alone.

Conducting good web searches is not hard, but it takes some practice. The key is to find search terms that are as specific to your problem as possible. To that end, here are some tips:

- The best search terms are error messages. For example, once my kernel crashed with the following message:

```
Jul  5 11:19:23 headache kernel: Unable to handle kernel NULL
pointer dereference at virtual address 00000005
```

  To turn this message into a set of search terms, I would strip out the parts of the error message specific to my machine (in this case the date and "headache", the name of my computer) and use the following as my search term:

```
"kernel: unable to handle kernel NULL pointer dereference at
virtual address"
```

  If other users have run into a similar problem, they might have reproduced the same error message in their cries for help.

- -4pc - -4pc

- If an application is crashing, include the name of the application. If some hardware is failing, include as much identifying information as you can. [REF IDENTIFYING INFO]

- Often it is a good idea to append the term "linux" or "debian" to the end of your search terms. For example, to find information about how to get my Zip 100 drive working under Linux, I might use the following search term:

```
zip 100 linux
```

Internet Forums

If you have Internet access, you can take advantage of the many Linux support forums that exist. In this section, we will discuss asynchronous support forums, where users are not required to be logged in simultaneously to communicate with each other. Forums that fall under this category include:

- *Web bulletin boards*, which users access using a web browser.

- *Mailing lists*, which users access via e-mail. Each user sends e-mail to a central address, which distributes the e-mail to every user on the list.

- *Newsgroups*, which users access via a *newsreader*. Newsgroups are similar to mailing lists: users post to a central location, and everybody reads the posted messages. Traditionally, special applications were needed in order to access newsgroups, but these days some e-mail clients (including Sylpheed support newsgroup access; websites such as Google Groups [http://groups.google.com] provide access to newsgroups as well.

All of these forms of communication have advantages and disadvantages. Web forums are probably the most straightforward for many users, but people have to remember to visit them, which sometimes reduces the quality of help available. Web forums generally require that you have good Web access, which makes them hard to use if your Internet access is very slow or very expensive.

Mailing lists are very popular, but large mailing lists can overwhelm your e-mail account with hundreds of messages a day.

Newsgroups have declined in popularity as a means of support. One of the biggest problem of newsgroups is that they are prone to *e-mail address harvesting* -- spammers collect lists of e-mail addresses used on newsgroups and send those people *spam*. [REF Spam]

In some ways there are too many support forums out there. Finding one or two effective forums you feel comfortable using can be difficult. Here are some tips to get you started: [FINISH ME]

- Forums run by your local LUG are often ideal. You get to know the people on the forum, and you open the possibility of getting face-to-face help if you really get stuck. The number of people participating in a local LUG forum tend to be small, which is both a good and a bad thing -- it is good because the tone tends to be friendlier, but bad because none of the participants may be able to help you with difficult questions.

- -4pc - -4pc

- If you choose to subscribe to mailing lists, you should find out the number of messages you can expect each day. Make sure your e-mail account has enough capacity to deal with the volume of messages. You may also want to filter messages coming from the mailing list, either by subscribing to the *digest* version of the mailing list (which will send you one message with all of the day's messages) or by sending all messages from the mailing list to a special folder. [HOW DO YOU DO THIS IN SYLPHEED?]

- In general, you are better off choosing web forums with lots of subscribers. This increases the chance that somebody will happen to see your question and know the answer.

- If you choose to use newsgroups (or any forum where your true e-mail address will be viewable by non-users) you should be aware that this address will be harvested by spammers. You might consider mangling your e-mail address [REF] or using an address that you don't care about.

- If you have a question about a specific product (for example, a particular web browser) you might look for a help forum devoted to that product. People on such specialized forums usually have in-depth knowledge of the product.

- Before asking questions on any forum you should be aware of the posting rules. See whether the group in question has posted a FAQ -- a list of frequently-asked questions. Search through the group's archives to see whether other people have asked the same question that you have. It is often good to *lurk* by reading the forum for a few days before making your posting. This will give you a better sense of the norms and culture of the forum, and help keep your foot out of your mouth.

- Be polite, even if others are rude to you. For the most part other people are supporting you for free, and it is always a good idea to express your gratitude for their help. Consider all responses fairly. If you find a solution to your problem, post the solution so that others can learn from your experience. Whatever you do, don't insult others on the forum.

- On a similar note, consider the tone of the forum you are using before you participate. If the forum is plagued by rudeness and mean-spirited *flamewars* (pointless arguments and character attacks) you may not want to participate. Similarly, if people are friendly and helpful, and offer good answers to people's questions, you might want to devote more time to supporting that forum.

A good starting point for your questions might be the WCLP mailing list. You can subscribe to that list at http://lists.sourceforge.net/lists/listinfo/wclp-user .

The web forums at LinuxQuestions [http://www.linuxquestions.org] are another good place to ask general Linux questions.

Asking Questions Effectively

Sometimes a new user will post a question to a help forum and get no response. This can be frustrating, especially when other questions posted later generate lots of replies. It's easy to conclude that the forum members are snooty or that they just don't appreciate questions from newbies, but this is not always the case. Sometimes nobody on the forum has an answer to the question, and sometimes nobody on the forum understands what the question is asking.

Although you cannot do much if people do not have an answer to your question, there is a lot you can do to ensure that your questions are clear and understandable. It is especially important to provide good information when you participate in asynchronous forums such as web boards or mailing lists, because the communication is rather slow; your helper cannot log into your machine directly and look around for information, and giving you directions on what to look for is time-consuming and error prone. Here are some tips to asking better questions:

- -4pc - -4pc

- Describe what you are trying to do and what is going wrong. The more detailed you can be about the problem, the better. If you can reproduce the problem consistently, you should provide a list of steps that produces the problem. If the problem only occurs once in a while, say that as well.
- List the error messages you see. Copy them down exactly. The error messages may not mean much to you, but they often contain clues that your helpers may be able to decipher.
- Describe the environment you are working with as best you can. Often people will be interested in the following things:
  - Your kernel version.
  - The distribution you are using, and the version of the distribution. For example, "Debian Sarge" or "Fedora Core 3".
  - The versions of the software which are causing the problem.
  - When some hardware is failing or not working, the make and model of the hardware.

  [HOW DO PEOPLE GET THIS INFORMATION?]
- Use the best grammar and spelling you can. Try not to use slang. Your goal is to be as clear as you can.
- State your question clearly. Simply describing a situation may leave people wondering exactly what help you are seeking.
- List the steps you have taken to try to solve the problem already. This will help people understand the problem better.

Before asking your question, be sure to do some research on your own. Look for FAQs and other postings on the same topic. Perform web searches using the error messages you find as search terms. Read the manuals and documentation associated with the hardware or software in question. Others will be more willing to help you if you have demonstrated that you have put some effort into finding a solution yourself.

- -4pc - -4pc

Further Reading

To learn more about the art of asking effective questions, you might consult the following documents:

- *How to Ask Good Questions*, by Eric S. Raymond and Rick Moen, available from http://www.catb.org/~esr/faqs/smart-questions.html . The tone of this document is somewhat arrogant, but it is loaded with good advice, and it is frequently cited.

  As requested by the authors, we remind you that this document is a reference; please do not contact Raymond or Moen for troubleshooting help on the basis of this document.

[REF howto, Debian list of good questions, debian bug report form]
[THIS IS NOT REALLY THE RIGHT TITLE. IRC IS AN INTERNET FORUM AS WELL.]

Realtime forums

Realtime internet communications such as instant messaging services and chatrooms can be effective ways to get support. Since messages are exchanged more quickly than e-mail, those providing support can troubleshoot problems in realtime, which can make fixing much faster than e-mail, mailing list or web forum communication.

The most popular form of realtime support in the open source world is *IRC*, which stands for "internet relay chat". In order to participate in IRC you need to know a *server* and a *channel*. Your local LUG may run IRC support channels. In addition, the Debian project runs a (rather busy) support channel called `#debian` on the server irc://irc.debian.org .

In order to use IRC you need to use a client program that suppports the protocol. Gaim is one such application; for instructions on how to configure it, see [WHERE?]

- -4pc - -4pc

Chapter 5. Administrative Tasks

Sooner or later, you will find yourself in a situation that requires a change in your computer's configuration. You may find yourself installing a printer, scanner, or CD-burner. You may need to configure Internet access when you start or change Internet Service Providers. You may want to install or remove software from your computer, and you will certainly want to keep your software up to date with security fixes. In Linux, all these tasks are *administrative*: they require the use of the *root account*. Because they permanently change the configuration of your system, all of these tasks must be carried out with caution.

One way to deal with administrative tasks is to get somebody else to do them for you. Friends or family members who are familiar with Linux may be willing and able to do your system administration. The organization or individual who installed Linux on your computer may offer follow up support for free or a fee. You may be able to find companies who offer home administration services. The members of your local LUG (Linux Users Group) are another good resource for administration help. Getting other people to do your administration for you is a legitimate strategy -- it is the usual practice in corporate and academic environments -- but it can be expensive (if you hire professionals), unreliable (if your friends and family know less about Linux than they think), or both.

Even if you find others willing to administrate your computer for you, it is worthwhile to learn a little bit about system administration. A strong "Do it Yourself" ethic runs through Linux culture. People will generally be more receptive to helping you administrate your machine if you demonstrate that you are willing to learn and do some of the work yourself. Furthermore, understanding something about the situation will help you follow what your administrator is doing to your machine.

In many cases people are willing to share their experience and knowledge, but will be unable or unwilling to come to your computer physically and help you out. For example, a person living in a foreign company might help you via e-mail, but will not be able to log into your computer and fix things for you. [1] You will have to serve as an intermediary between your computer and the person helping you by typing in commands and relating the error messages and prompts that your computer displays. Knowing some basic system administration skills will help you do a better job of helping your helper.

If you are new to Linux administration, it is well worth finding a human mentor (or group of mentors) who will answer your questions and offer their advice. The reality of system administration is that it rarely goes smoothly. When you get stuck, a fresh brain and experienced set of eyes can save you days of frustration.

In addition to human help and this chapter, a large amount of written documentation exists about Linux system administration using the command line. A good place to start is the [DEBIAN USER REFERENCE], which may be installed on your computer already. [HOW DO YOU ACCESS THIS?] Additional help resources are listed [WHERE].

Fpr better or for worse, many administrative tools assume a familiarity with the command line, so the first section introduces the command line and some useful utilities from a user's point of view. We then discuss how to become the root user, and then move into a long list of administrative tasks, including setting file permissions, installing software, and changing the date and time. [FINISH ME]

Your Friend the Command Line

Although graphical tools exist for some administrative tasks in Linux, the interface of choice remains the *command line shell*. There are some good reasons for this. The command line has thirty years of history behind it. In some ways -- such as manual pages -- it certainly shows its age. However, over the decades people have refined the command line into an efficient, powerful interface. A lot of good tools exist for the command line that do not exist in GUI format.

[1] This is technically not true. You could reconfigure your system to allow this person access to your computer over the Internet, but you probably do not want to do this as a beginning Linux user.

- -4pc - -4pc
This situation is unlikely to change anytime soon. For this reason alone it is worth becoming comfortable with using the command line.

One of the biggest advantages of the command line interface is that it is *scriptable*. The command line shell supports an entire programming language which makes it possible to automate repetitive or tedious administration tasks. Programs that are run in the command line can often be chained together using *pipes* to solve difficult problems. Although you may not need to write scripts or complicated pipelines yourself, the scripts others have written will make your administration life easier.

Starting the Command Line

To get a command line, select WCLP Apps->x-terminal-emulator [DID THIS CHANGE TO "command line shell"?] from the IceWM program menu. A command line window with a *shell prompt* will appear. [MAKE SCREENSHOT]

By default, the command line prompt looks like this:

```
linuxuser@wclp-box:~$
```

You interact with the command line shell by typing in commands, then pressing Enter. The command line shell will interpret what you have typed and attempt to follow your orders. If it gets confused then it will display an error message. [EXAMPLE]

Commands and files in Linux are case-sensitive. For example, `my_documents` and `My_Documents` are different files. Most Linux commands and many files and directories use only lower-case letters.

Shell Prompt Information

Let's look at the format for the command line prompt. By default, it looks something like this:

```
linuxuser@wclp-box:~$
```

This gives you some information. It tells you that this is the prompt for a regular user called "linuxuser", who is logged into a computer named "wclp-box". The ~ indicates that we are in linuxuser's home directory, and the "$" tells us that this account is not using root privileges.

If we changed directories, the prompt also changes:

```
linuxuser@wclp-box:~$ cd My_Documents/
linuxuser@wclp-box:~/My_Documents$
```

The **cd** command stands for "change directory". We have changed the directory to the `My_Documents` directory, and this is reflected in the prompt.

To become the *root user*, we use the **su** command:

- -4pc - -4pc

```
linuxuser@wclp-box:~/My_Documents$ su
```

At this point we are asked for the *root password*. After typing this in, we get the following prompt:

```
wclp-box:/home/linuxuser/My_Documents#
```

The "#" at the end oft the prompt tells us that we are the root user. `/home/linuxuser/My_Documents` is the *full path* of the `My_Documents` directory.

The shell prompt can be configured to show all kinds of information. See the HOWTO in `/usr/share/doc/HOWTO/en-txt/Bash-Prompt-HOWTO.gz` for more information. Customizing your prompt can be fun, but the default prompt shows enough information to be useful.

Command Line Tips and Tricks

Some people feel that the command line interface is inefficient because you have to type in all your commands. In fact, the opposite is true. Thanks to shortcuts offered by the shell, most commands and filenames can be entered with only a few keystrokes, without needing to reach for a mouse. The following sections will document some of the most useful shell shortcuts.

Autocompletion using Tab

Autocompletion may be the most useful feature the shell offers. When you type a few characters and hit the Tab key, the shell attempts to guess the rest of the command or file you want. If there is no ambiguity, the shell will complete the command or file fully. Otherwise, the shell will complete as much of the line as it can. Pressing Tab twice will show the possible completions it has found.

For example, to get to my working directory for this manual, I need to enter the following line:

```
cd volunteer/wclp/manuals/userguide/
```

To enter this line, I type the following keystrokes:

```
cd vTabwTabmaTabuTab
```

Instead of typing 39 keystrokes, I end up typing 9, where four keystrokes are Tab.

It turns out that I have two directories in my `volunteer/wclp/` directory. One is called `manuals` and the other `morphix`. If I press

- -4pc - -4pc

```
cd vTabwTabmTab
```

then the shell will not be able to tell which of the directories I want to visit, and it will display

```
cd volunteer/wclp/m
```

Pressing Tab again will show my options:

```
manuals  morphix
```

I can then type **oTab** to go to the `morphix` directory, or **aTab** to go to the `manuals` directory.

The shell knows that I am looking for a filename or directory because I already typed in a command -- namely, the **cd** command. If you press Tab when typing the first word of a command line, the shell will know that you are trying to type a command, and it will try to complete the name of the command for you.

For example, say I type:

```
xscTab
```

The shell completes this to the following:

```
xscreensaver
```

If I press Tab twice more I can see all the commands starting with the word `xscreensaver`:

```
xscreensaver            xscreensaver-getimage-file
xscreensaver-command      xscreensaver-getimage-video
xscreensaver-demo        xscreensaver.kss
xscreensaver-getimage
```

I could then press Enter (to run the **xscreensaver** command), or I could type the next few characters of the command I wanted, and then press Tab to let the shell autocomplete the command.

The shell knows what commands to look through by examing its *$PATH environment variable* . This is a variable that lists directories in which the shell should look for commands. You can display the contents of your `$PATH` by typing

```
echo $PATH
```

- -4pc - -4pc

The values of this variable are often set in the `.bash_profile` configuration file. Usually you will not need to change your `PATH` variable, but knowing the list of directories can be useful if there are two commands with the same name; in this case the command in the directory that appears first in the `PATH` gets executed by default.
[SHOULD I GET RID OF THIS LAST SECTION?]

Repeating Commands with the Arrow Keys

The shell keeps track of the commands you type in. You can visit this *command line history* by pressing the up and down arrow keys. You can then edit the previous line and run it again. This is very useful if you mistyped a command and need to re-edit it, or if you need to repeat some complicated command on a different file or directory.

The shell provides other keystrokes (such as ctrl-A) to speed up command-line editing. To learn about these, see [WHERE?]

Wildcards

Command-line *wildcards* allow you to specify many files at once. There are several wildcards, but probably the most useful one is the asterisk (`*`) wildcard. For example, to look at all the text files in a directory, one at a time, you might type

```
less *.txt
```

This will match all files that end with `.txt`, and ignore all other files. You can surround the asterisk wildcard with other characters. For example,

```
less a*.txt
```

would match `aardvark.txt` and `acorn.txt`, but not `bunnyrabbit.txt`.

Wildcards are useful whenever you need to perform some action on a group of files, but you must treat them with caution. A famous example is the **rm** command, which is used to delete files. Typing

```
rm * .bak
```

instead of

```
rm *.bak
```

- -4pc - -4pc
is almost always disastrous. In the first case, you will delete all files in the current directory, and also any files named `.bak` . In the second case, you will delete all files that have an extension of `.bak`, which is probably what you wanted.

Common Commands

This section will document some of the most frequently used shell commands. For the most part, these commands are not administrative; they are everyday commands used by regular users. Although this list is not comprehensive, it should get you started. For a more thorough introduction, see [WHERE? THERE IS NO HOWTO, IS THERE?].

Listing files

The command to list the files in a directory is **ls**. This command shows different information depending on the options you specify. Without any arguments, **ls** shows files and directories that do not begin with a period. Here is some sample output:

```
linuxuser@migraine:~/My_Documents$ ls
Blah.rtf  demo.rtf.bak~    exampledir    pix
demo.rtf  demonstration.rtf  joeresume.rtf  rename_this.txt
```

Depending on your configuration, **ls** will show different colours depending on file type. For example, directories will be displayed in blue.

To display slightly more information, you might use **ls -sF**:

```
linuxuser@migraine:~/My_Documents$ ls -sF
total 36
4 Blah.rtf     4 demonstration.rtf  4 pix/
4 demo.rtf     4 exampledir/      4 rename_this.txt
4 demo.rtf.bak~  8 joeresume.rtf
```

The `s` option displays a file size between each file, and the `F` displays the file type. For example, directories are displayed followed by a slash. Note that the `F` option must be in uppercase.

By default, **ls** will not display so-called *hidden files* -- files or directories that begin with a period. The `a` option displays these files:

```
linuxuser@migraine:~/My_Documents$ ls -a
.  .hidden  demo.rtf     demonstration.rtf  joeresume.rtf
.. Blah.rtf  demo.rtf.bak~  exampledir      pix
```

The `.` and `..` directories have special meaning. The first refers to the current directory, and the second refers to the directory above this one. We also see a regular directory called `.hidden` .

A very common option to **ls** is `l`:

```
- -4pc - -4pc
```

```
linuxuser@migraine:~/My_Documents$ ls -l
total 36
-rw-r--r--  1 linuxuser linuxuser 3160 Jan  2 17:17 Blah.rtf
-rw-r--r--  1 linuxuser linuxuser 2008 Dec 28 00:01 demo.rtf
-rw-r--r--  1 linuxuser linuxuser 2832 Dec 28 11:08 demo.rtf.bak~
-rw-r--r--  1 linuxuser linuxuser 2008 Dec 27 23:59 demonstration.rtf
drwxr-xr-x  2 linuxuser linuxuser 4096 Dec 31 00:55 exampledir
-rw-r--r--  1 linuxuser linuxuser 5051 Dec 30 23:34 joeresume.rtf
drwxr-xr-x  4 linuxuser linuxuser 4096 Dec 29 23:12 pix
-rw-r--r--  1 linuxuser linuxuser  106 Jan  2 17:20 rename_this.txt
```

Each row of this listing corresponds to the information associated with a single file. To understand the meanings of these rows, it helps to know something about file ownership and permissions in Linux.

File Ownership and Permissions

For security reasons, every file in Linux has permissions associated with it. File permissions determine how the file is allowed to be accessed. Here are the meanings for regular files:

- *Read permissions* determine who can access the contents of the files. A file with read permissions can be viewed but not necessarily edited.
- *Write permissions* determine who can modify the files.
- *Execute permissions* determine who can treat a file as a program. In order to run a program in Linux, you must have execute permissions for the program file -- if you do not have such permissions, the program file is treated as a regular data file by Linux. Execute permissions become important when dealing with *shell scripts*, which are specially-formatted regular text files with execute permissions set.

In Linux, directories are treated as files. Like files, every directory has read, write, and execute permissions. However, the meanings of these permission types is different.

To illustrate these concepts, we will use the example of a directory called `My_Documents` which contains a file called `resume.rtf`, and we'll say that both `My_Documents` and `resume.rtf` are owned by a user `linuxuser`.

- -4pc - -4pc

- Read permissions on directories determine who may list the contents of the directories to find out what files are inside. This is different than being able to *use* files inside the directory.

  Read permissions are almost always granted along with execute permissions. If a directory has read permissions but not execute permissions, then users can see the names of the files in the directory, but cannot use them in any way. Say that `My_Documents` had read but not execute permissions for `linuxuser`, and that `resume.rtf` had both read and write permissions. Contrast the following situations:

  ```
  $ ls My_Documents
  resume.rtf
  ```

  ```
  $ ls -l My_Documents
  ls: My_Documents/resume.rtf: Permission denied
  ```

  In the second case, **ls** complains because it cannot access information about `resume.rtf`. It cannot even look up the file size.

- Write permissions on directories determine who may create and delete files within that directory. In our example, assume that `linuxuser` had read and write permissions `resume.rtf`, and read and execute (but not execute) permissions on on `My_Documents`. Then `linuxuser` could read or edit `resume.rtf` -- and could even erase all the data inside the file -- but would not be able to delete the file totally. Similarly, `linuxuser` would not be able to create any new files in `My_Documents`.

- Execute permissions on directories determine who may use the files in the directory. Say that `linuxuser` had execute (but not read or write) permissions on `My_Documents` and had read and write permissions on `resume.rtf`. Then a person using the `linuxuser` account could view or edit `resume.rtf` if he or she knew the filename. For example, the user might be able to type

  ```
  $ cp My_Documents/resume.rtf /media/floppy
  ```

  but would not be able to successfully execute the following:

  ```
  $ cp My_Documents/*.rtf /media/floppy
  cp: cannot stat ‘My_Documents/*.rtf’: No such file or directory
  ```

  In the second example, "cannot stat" means that the **cp** command cannot read the directory’s contents.

  Execute permissions are often used for webpages, so that people visiting the webpage can access particular HTML files without being allowed to snoop around.

- -4pc - -4pc

Read, write, and execute permissions for files and directories can be given independently to three different sets of users: the *owner* of a file, members of the *group* associated with that file, and everybody else. Each file has an owner and a group. Owners are individual users defined in `/etc/passwd`. Groups are defined in `/etc/group`. A user may belong to many groups, and many users may belong to a group.

Traditionally, groups are used to give multiple users access to a file. For example, in a class project people might work in pairs, and each pair might get a group. That way, both members of the pair can modify the class project files, but outsiders will be locked out. Another use of groups is to give particular users access to resources such as floppy drives, CD-ROMs and the sound card.

In practice, you will rarely have to worry about group ownership or permissions of files.

In the Debian distribution (and therefore on your computer) each user account gets a group with its own name. This is used to keep files private -- by default, only the user should be able to access his or her files.

Reading ls -l output

We can now understand the output produced by **ls -l**:

```
linuxuser@migraine:~/My_Documents$ ls -l
total 36
-rw-r--r--  1 linuxuser linuxuser 3160 Jan  2 17:17 Blah.rtf
-rw-r--r--  1 linuxuser linuxuser 2008 Dec 28 00:01 demo.rtf
-rw-r--r--  1 linuxuser linuxuser 2832 Dec 28 11:08 demo.rtf.bak~
-rw-r--r--  1 linuxuser linuxuser 2008 Dec 27 23:59 demonstration.rtf
drwxr-xr-x  2 linuxuser linuxuser 4096 Dec 31 00:55 exampledir
-rw-r--r--  1 linuxuser linuxuser 5051 Dec 30 23:34 joeresume.rtf
drwxr-xr-x  4 linuxuser linuxuser 4096 Dec 29 23:12 pix
-rw-r--r--  1 linuxuser linuxuser  106 Jan  2 17:20 rename_this.txt
```

The output of the command is organized into columns. The first column consists of ten characters. The leftmost character describes the file type, and the other nine characters describe the permissions of the file.

The leftmost character can take on several different values. An entry of `-` indicates a regular file. An entry of `d` indicates a directory. There are a number of other possibilities as well, including `l` (symbolic links), `b` (block devices), and `c` (character devices).

From this character alone, we can tell that `exampledir` and `pix` are directories, and the other files are regular files.

The other nine characters are grouped into three groups of three characters each. The first set (characters 2-5) list the owner permissions for the file. The second set lists group permissions, and the third set lists *world permissions* -- permissions for accounts that are not the owner and not in the group. Entries of `-` indicate that the permission is *not* granted, and `r`, `w`, or `x` indicate read, write, and execute permissions respectively.

From the output above, we can see that all the regular files give read and write permissions to the owner, and read permissions to the group and rest of the world. The directories give read, write, and execute permissions to the owner, and read and execute permissions to the group and world.

This configuration is unusual: usually regular files should deny all access to accounts other than the owner or group.

The second column is a single number. It is mostly unimportant. It records the number of references to the file in question. All the regular files have a single reference. The directories have multiple references because they refer to

- -4pc - -4pc
themselves and their parent directories also refer to them. The `pix` directory has two subdirectories, both of which point to it and increase the reference count. Sometimes the reference count becomes important when dealing with security issues or *hard links*, but for the most part you can ignore it.

The third column lists the owner of the file, and the fourth column lists the group associated with the file. We can see that all of these files are owned by *linuxuser* and the *linuxuser* group. [MAKE THIS EXAMPLE MORE INTERESTING?]

The fourth column lists the size of the file in bytes. The next three columns list the date and time of the last modification to the file, and the final column lists the filename.

Changing directories

The shell keeps track of a *working directory* -- a location in the directory tree that the shell is currently "active". Unless specified otherwise, the shell operates on files in the working directory.

By default, the shell initializes its working directory to your *home directory*. You can change your working directory by using the **cd** command.

Typing **cd** with no arguments will return you to your home directory. You can also specify the home directory using ~. For example, by typing

```
cd ~/My_Documents
```

you can navigate to the `My_Documents` directory inside your home directory. You can also specify the home directories of other users using ~. For example, typing

```
cd ~charm
```

would navigate to the home directory of a user named `charm` .

If you are currently in a directory, you can navigate one level "upward" by typing

```
cd ..
```

This navigates to the current directory's *parent directory* -- the directory that contains this one. For example, say that your current directory was `/home/linuxuser` . Then typing **`cd ..`** would change the current working directory to `/home`. The topmost directory `/` is called the *root directory*, and its parent directory is itself.

Directories can be specified as their locations in the directory tree (called the *absolute path*) or as directories relative to the current directory (called a *relative path*). Entering either the absolute or a relative path of a directory to the **cd** command will change to that directory.

- -4pc - -4pc
For example, say we have a home directory called `/home/linuxuser` which contains two subtrees called `My_Documents` and `downloads`, and that the working directory is currently `/home/linuxuser/downloads`. Any of the following commands will change to the `My_Documents` directory:

```
cd ../My_Documents
```

or

```
cd /home/linuxuser/My_Documents
```

or

```
cd ~/My_Documents
```

You can use whichever option is most convenient for you.

If you ever get lost and want to know your current working directory, type

```
pwd
```

This will print out your current working directory. Usually, the current working directory is also displayed in the shell prompt.

Manipulating Files and Directories

Although graphical file managers such as XFE allow you to manipulate files and directories easily, it is not hard to perform these operations on the command line. Here is a quick summary of the most common commands:

- The **mkdir** command creates directories.

- -4pc - -4pc

- The **rmdir** command removes empty directories. This command will fail if the directory is non-empty. To delete non-empty directories you need to use **rm -Rf**:

```
rm -Rf tempfiles
```

  Beware! The **rm** command is notorious for causing accidental data loss. See [WHERE?] for more information.

- The **cp** command copies files. To copy directories and their contents, use **cp -r**:

```
cp -r document document-copy
```

- The **mv** command moves files and directories. It will overwrite files if you are not careful. You can also use this command to rename files:

```
mv oldname newname
```

Examining text files: less and script

One of the most useful command line utilities is **less**, which displays the contents of text files one page at a time. For example, typing

```
less /etc/passwd
```

would show you the password file, which shows all the user accounts on the system. The **less** command is keyboard driven. Here are some of the most common keystrokes:

- The q quits the command.
- The space bar advances a page.
- The Page Up and Page Down work as advertised, and the up and arrow keys navigate a line at a time.
- The h key activates the help.
- The / key searches forward in a file. Type /, and then the term you want to search for, then press Enter. To repeat your last search, press n.
- The ? key searches backwards in a file. You can repeat searches with the n key.

- -4pc - -4pc

A variation of **less** called **zless** will display text files compressed using **gzip**, and **bzless** will display text files compressed using **bzip2**. For example, to display the Security Quickstart HOWTO, type:

```
zless /usr/share/doc/HOWTO/en-txt/Security-Quickstart-HOWTO.gz
```

The **less** command also comes in handy to read output which would otherwise scroll off the screen. For example, on most systems the following command will produce too much text to fit on one screen:

```
ls -l /usr/share/doc/
```

You can use **less** to display this information a page at a time by typing:

```
ls -l /usr/share/doc/ | less
```

The vertical bar | is known as a *pipe*. It takes the output of **ls** and feeds it as the input to **less**.

If you have many commands to type and you want a textual log of the input and output, use the **script** command. This command is also useful when **less** does not capture the output you want. To start **script**, type:

```
script /tmp/output
```

This tells **script** to record the following input and output to a file called `/tmp/output`. If you do not specify a file, **script** will write its output to a file called `typescript` in the current directory.

After you have started script, run the commands you need to run. **script** will record everything you type on the command line, and the output of every command. When you are done, type

```
exit
```

to end the **script** session. You can then read the contents of the session using **less**:

- -4pc - -4pc

```
less /tmp/output
```

Both **less** and **script** should be used for text-only commands. Do not use them for programs that offer an interactive text-user interface such as aptitude. If you do so, you will just see a lot of garbage characters. [THIS IS NOT EXPLAINED CLEARLY. HOW DO I DESCRIBE CURSES PROGRAMS AND DISTINGUISH THEM FROM OTHER PROGRAMS?]

Working With Processes

A *process* is a running program. Every time you run a command or application, a new Linux process starts (and depending on the application, sometimes many processes start). When a process is started, it is given a unique identifier called a *process ID* (PID).

From time to time, you may need to look at what processes are running on your computer. Occasionally a program might crash or lock up, and you will need to use brute force to end the process. You might notice a large amount of hard drive activity and wonder what is going on. The **ps**, **pstree** and **top** commands allow you to check what processes are running and what they are up to.

Processes also have an owner, which is the name of the account that started the process. Administrative processes (usually started by the root account) are called *system processes* and processes started by user accounts are called *user processes*. Some processes provide services (such as web access, printing, and font display). The programs that provide these services are called *daemons*. Most daemons are started when your computer boots up, by running scripts in the `/etc/init.d/` directory.

Processes are started by other processes. For example, when you run **ls** in a command line shell, the shell starts a new process to run the command. The starting process (in this example, the shell) is called the *parent process*, and the process that is created (in this case the process for **ls**) is called the *child process*.

The **ps** command lists running processes and their PIDs. For example, if your user account is `linuxuser` then the following command will list the processes you are running:

```
ps -u linuxuser
```

Here is some sample output of this command:

```
PID TTY          TIME CMD
214 tty1     00:00:00 bash
309 tty1     00:00:00 startx
316 tty1     00:00:00 xinit
321 tty1     00:00:00 wm2
349 ?        00:00:00 ssh-agent
350 tty1     00:00:00 x-terminal-emul
351 tty1     00:00:01 x-terminal-emul
354 tty1     00:00:21 coral
355 pts/0    00:00:00 bash
356 pts/1    00:00:00 bash
```

```
- -4pc - -4pc
  397 pts/1   00:00:00 screen
  398 ?      00:00:01 screen
  399 pts/2   00:00:00 bash
  422 pts/3   00:00:00 bash
  537 pts/3   00:00:02 vi
  860 pts/2   00:00:00 ps
```

The first and last columns of this output are the most important. The "PID" column lists the process identifier of the process, and the "CMD" column lists the name of the command.

To list all of the processes running on the machine type:

```
ps aux
```

This command produces a lot of output, so you may need to pipe it through **less**.

The **pstree** command provides another way to look at the processes running on your computer. This command shows the parent-child relationships between processes. Here is some sample output:

```
    pnijjar@wclp-box:~$ pstree | less
init-+-atd
    |-bash---startx---xinit-+-XF86_SVGA
    |                 ‘-wm2-+-ssh-agent
    |                       |-x-terminal-emul---bash
    |                       ‘-x-terminal-emul---bash---screen---screen-+-
bash-+-less
    |
    |
    ‘-pstree
    |
    |-bash---vi
    |
    |-bash---woody
    |
    ‘-bash---less
    |-coral
    |-cron
    |-dictd
    |-5*[getty]
    |-inetd
    |-keventd
    |-khubd
    |-2*[kjournald]
    |-klogd
```

The process names that end with the letter "d" (`dictd`, `klogd`, `inetd`, `kswapd` and so on) are daemons. The processes that begin with "k" are *kernel processes*, which are important system processes.

- -4pc - -4pc

The **pstree** command has some useful options. The `l` option wraps long lines, and the `p` option prints the PIDs of processes along with their relationships. You can get quite a lot of information about running processes by typing

```
pstree -lu | less
```

A third tool to observe processes is top, which is a text-only interactive application that shows the processes that are currently running on your system. To start top, type

```
top
```

The output looks something like this:

```
10:45:09 up 51 min,  5 users,  load average: 0.33, 0.41, 0.47
45 processes: 41 sleeping, 4 running, 0 zombie, 0 stopped
CPU states:  15.5% user,   4.4% system,   4.0% nice,  76.1% idle
Mem:    29592K total,   27640K used,    1952K free,    3564K buffers
Swap:  124736K total,    3900K used,  120836K free,    9916K cached

  PID USER    PRI  NI  SIZE  RSS SHARE STAT %CPU %MEM   TIME COMMAND
  530 root     12 -10  4316 1888   980 S <  28.3  6.3   7:47 XF86_SVGA
  711 pnijjar  14   0   932  932   740 R    4.7  3.1   0:00 top
  567 pnijjar  19  17  1524 1436   880 R N  1.8  4.8   1:37 petri
    6 root      9   0    0   0     0 SW   0.9  0.0   0:00 kupdated
    1 root      8   0  116   72    72 S    0.0  0.2   0:03 init
    2 root      9   0    0   0     0 RW   0.0  0.0   0:00 keventd
    3 root     19  19    0   0     0 SWN  0.0  0.0   0:00 ksoftirqd_CPU0
    4 root      9   0    0   0     0 SW   0.0  0.0   0:05 kswapd
    5 root      9   0    0   0     0 SW   0.0  0.0   0:00 bdflush
    8 root      9   0    0   0     0 SW   0.0  0.0   0:00 kreiserfsd
   38 root      9   0    0   0     0 SW   0.0  0.0   0:00 khubd
   75 root      9   0    0   0     0 SW   0.0  0.0   0:00 kjournald
  647 pnijjar   9   0 1300 1300  1036 S    0.0  4.3   0:00 bash
  675 pnijjar   9   0 2228 2228  1136 S    0.0  7.5   0:05 woody
  679 pnijjar   9   0 2076 2076  1276 S    0.0  7.0   0:01 vi
  680 pnijjar   9   0 1280 1280  1028 S    0.0  4.3   0:00 bash
```

The first five lines show useful information about the overall performance of the computer. In particular, the third line shows how much of your CPU is being used, and by what processes. If your "idle" percentage is low, then you can expect your applications to be sluggish. The fourth line tells you about your RAM. If very little RAM is free, your computer will start *swapping* -- it will move some memory contents to the *swap space* on your hard drive. This will

- -4pc - -4pc
cause a lot of hard drive activity, and will typically make applications run very slowly. The fifth line tells you about your swap space. For good performance you do not want too much of your swap in use. If all of your swap gets used up then your computer is in essence out of memory, and unpredictable things will happen. Fortunately, running out of swap is rare.

Note that these numbers are approximations. Having a low amount of free memory does not necessarily mean that your computer will be slow, because the memory usage numbers can inflate over time or frequent use.

The rest of the screen contains information about the processes that are running. Many of the columns are self-explanatory, but a few deserve attention.

The "STAT" column tells you the *state* of the process. An "R" indicates that the process is currently running -- that it is using CPU cycles. An "S" indicates that the process is sleeping -- that it is currently in memory, but is not using any CPU cycles. A "W" indicates that the process currently has some of its memory swapped to the hard drive. There can also be "N", "D", and "Z" flags as well, which you can learn more about by reading the top manual page.

The "TIME" column tells you the cumulative *CPU Time* the process has used so far. A process can have a large number here for two reasons: it may require a lot of CPU, or it may have been running for a long time. This column provides a history of this process's activity.

The "%CPU" entry shows the CPU cycles *currently* being used by processes. You can discover what processes hog the CPU by looking in this column.

If your computer is very slow and you notice a lot of hard drive activity, watch the activity of the **kswapd** daemon. Its job is to swap memory to the hard drive and back. If it shows a lot of activity, your computer might be *thrashing* -- one or more processes cannot fully fit in RAM, and the **kswapd** daemon repeatedly moves memory contents between RAM and the hard drive to accomodate everybody. Thrashing is usually bad -- it will slow your computer down to the point where it becomes aggravating to use. If you find that your computer is thrashing, you may have to stop some processes that are hogging memory. This may mean that you do not run hoggy applications together (for example, you don't run OpenOffice and Mozilla at the same time) or that you avoid certain memory-intensive applications entirely. A third option is to install more memory in your computer, but that can get expensive.

The top application will continue to run until it is stopped. It will refresh the display every five seconds. To exit top, press the q key.

Note that top itself is a process, and that it can consume a fair number of CPU cycles as well. It is an excellent program to run to get a sense of what is going on with your computer, but you may not want to run it continually.

Another command called **watch** can also help you monitor processes. The **watch** periodically runs a command and displays the output. For example, if you wanted to look at the current running processes, you might type:

```
watch -n 10 -d pstree
```

The `n` option tells **watch** to refresh the display every 10 seconds, and the `d` option tells **watch** to highlight differences between screen updates. To quit the command, type ctrl-c .

The nano text editor

Some applications (such as w3m, **cvs** and mutt) will start a text editor so that you can edit files. For example, mutt will start a text editor so that you can compose a mail message. Unless otherwise specified by the `EDITOR` environment variable, command-line applications will start the nano editor. You can identify that this editor is running by the "GNU

- -4pc - -4pc
nano" in the top left hand corner. [SCREENSHOT?] This editor is fairly easy to use, but it can be confusing if you have never used it before.

The first thing to understand about nano is that it is keyboard driven by default. The second thing to understand is that most of the commands offered by the editor are listed at the bottom of the screen. Many commands consist of the ctrl key followed by a letter. (To save screen real estate, the instructions at the bottom of the screen write the ctrl key as "^".) Here are the most common commands:

- ctrl-g displays the help. Pressing F1 will also display this help. The help lists all the commands available in the editor.
- ctrl-x quits the editor. You will be prompted whether you want to save any changes you made to your file.
- ctrl-o saves the file you are working on without quitting the editor.

One annoyance about nano is that it wraps long lines to the screen width by default. While this is useful when composing e-mails, it sometimes causes errors when editing configuration files. To turn off word wrap, type Esc-w.

Searching files with grep

The **grep** command is used to search through text files for patterns. In its simplest form, the command has the following form:

```
grep pattern file
```

For example, to find all occurences of the word "HOSTNAME" in `/etc`, you would type

```
grep "HOSTNAME" /etc/*
```

The `/etc/*` means that every file in the `/etc` directory should be searched. However, files inside subdirectories of `/etc` will not be searched. To do this, you can use the `r` flag:

```
grep -r "HOSTNAME" /etc
```

If you do not care about whether your search pattern uses upper or lower case letters, use the `i` flag:

```
grep -i "HOSTNAME" /etc
```

- -4pc - -4pc

The **grep** command has many other options. In particular, it is possible to specify complicated patterns by specifying *regular expressions*. To learn about regular expressions, see [WHERE?] Although regular expressions are a powerful tool, you do not need them to benefit from **grep**.

Working with tar files

[REF: discussion of compression]

In addition to graphical applications like XFE, command line tools exist to work with archives. The most common archive format in Linux are *tar* files, which are manipulated using the **tar** command.

For example, consider a gzipped tar file named `example.tar.gz` which is in the current directory. To list the contents of the archive, you can type:

```
tar tvzf example.tar.gz
```

The `t` flag tells **tar** to list the contents of the archives. The `vzf` flags are often used together, but they have distinct meanings. The `v` stands for "verbose"; it prints out detailed information about the files in the archive. The `z` is used to indicate that the archive is compressed using **gzip**. Finally, the `f` indicates that we should use a file for input: in this case the file is `example.tar.gz`.

We can perform other operations to this archive. To unpack the archive into the current directory, type:

```
tar xvzf example.tar.gz
```

The `x` option stands for "extract".

Another common use of **tar** is to create archives. By convention, people first place all the files they wish to archive in a directory, and then archive the directory. For example, if you wanted to create a tarball of a big project, you could create a directory named `project`, place copies of the files to archive in that directory, then traverse to the parent of the `project` directory and type:

```
tar cvzf project.tar.gz project
```

The `c` option stands for "create". This will create a tarball in the current directory called `project.tar.gz`.

Note that `x`, `t` and `c` are "special" options -- they must appear first in the list of options, and only one of them may be present when invoking **tar**.

In addition to archives compressed with **gzip**, you may run into tar files compressed with **bzip2**. To work with these files, replace the `z` option with `j`. For example, to unpack `example.tar.bz2`, you would type

- -4pc - -4pc

```
tar xvjf example.tar.bz2
```

Becoming the root user

In order to perform administrative tasks you will have to use the *superuser account*, which is also known as the *root account*. This account is the most powerful one on your computer -- it can manipulate any file and access any device, regardless of the file ownership or permissions. In fact, in some ways it is too powerful -- many security exploits work by obtaining access to the root account. (In fact, there exist initiatives such as *SELinux* to reduce the power of the root account.) Because the root account is so powerful you do not want to use it for your day-to-day work, and you should take precautions to keep your *root password* safe and secure.

On the one hand, you do not want to log in using the root account. On the other, in order to make administrative changes you will have to operate as the root user. Some programs (such as aptitude) prompt you for the root password when it is time to do adminstrative tasks. Other programs and commands must be run from the root account. This section explains how to do this.

There are more mechanisms for running commands as root than are explained here. One of the most popular is a command called **sudo**. If you want to keep logs of the actions perfomed using the root account, or if many people handle the system administration of your computer, you may want to investigate this command.

Using su to change your identity

The **su** command is used to change users. To use it, type:

```
su -
```

and enter the root password when prompted. (You should have received this password when you got your computer.)

This will change the working directory to `/root` (which is the home directory of the superuser account). If you wish to stay in the current directory after changing identities, type

```
su
```

without the dash. There are some subtle differences between these two commands, but usually either will work for carrying out administrative tasks.

After becoming the root user you should find that your command prompt ends with a hash:

```
wclp-box:~#
```

- -4pc - -4pc

Any commands you type will be run as the root user. To revert your identity out of root, type **exit**.

In fact, you can change your identity to any other user using **su**. For example, to become user `bill`, you could type

```
su - bill
```

and then enter in the password for `bill` .

Running graphical utilities as root

Although most administrative commands are text-based, a few (such as printtool) are graphical. You also may want to run nedit as the root user to edit configuration files. If you try running these applications after changing your identity with **su**, you will get the following message:

```
Xlib: connection to ":0.0" refused by server
Xlib: Client is not authorized to connect to Server
rxvt: can’t open display :0
```

In order to launch graphical apps, you need to set an *environment variable* called `XAUTHORITY`. This gives the root account “permission” to show graphical applications on your screen, even though you did not log in as the root user. Say that your usual account is called `linuxuser`. Then you would type the following at the root prompt to enable graphical applications:

```
export XAUTHORITY=/home/linuxuser/.Xauthority
```

After running this command, you should be able to launch graphical applications. For example, to launch printtool, type:

```
printtool &
```

Some Administrative Tasks

Unless otherwise specified, all of these applications must be run as the root user.

- -4pc - -4pc

Changing the Date and Time

The **date** can be used to change the date and time on your computer. The format of the command is odd, but straightforward:

```
date MMDDhhmmYYYY
```

where the letters have the following meanings:

- `MM` is the month (01 to 12)
- `DD` is the day (01 to 31)
- `hh` is the hour in 24 hour format (00 to 23)
- `mm` is the minute (00 to 59)
- `YYYY` is the year (0000 to 9999)

For example, to change the date and time to 4:45pm on June 15, 2007, type

```
date 061516452007
```

If you have Internet access, you can also set the date using NTP, the Network Time Protocol. To use this you need the **ntpdate** package installed on your computer. This command will connect to servers on the Internet and use them to set your clock.

Changing file permissions and ownership

You may want to read the section file ownership and permissions[REF] before running the commands in this section.

To change file permissions, use the **chmod** command. The form of the command is:

```
chmod (who)(operation)(what) filename
```

where the arguments take on the following values:

- *`who`* indicates whose permissions get changed. Values can be "u" for user, "g" for group, and "o" for world (other) permissions. There is also "a", which will change permissions for all users. These letters can be combined in one command.
- *`operation`* is "+" to add permissions, and "-" to remove permissions.

- -4pc - -4pc
  - *what* indicates what permissions get changed. Values can be "r" for read, "w" for write, and "x" for execute permissions. These letters can also be combined in one command.

For example, to give the owner of the file `myscript.sh` execute permissions, type

```
chmod u+x myscript.sh
```

To take away read, write, and execute permissions on `resume.rtf` for everybody except the file's owner, type

```
chmod go-rwx resume.rtf
```

To give everybody access to a `public_html` directory, type

```
chmod ugo+x public_html
```

or equivalently

```
chmod a+x public_html
```

A file's owner can change that file's permissions, so this command does not necessarily need to be run as root.

The root user can also change the ownership of files -- that is, the group and user associated with the file -- using the **chown** command:

```
chown user:group filename
```

For example, to give user `linuxuser` and group `staff` ownership of the file `resume.rtf`, type

```
chown linuxuser:staff resume.rtf
```

- -4pc - -4pc

Adding users

At some point, you may want to create additional user accounts on your computer. Separate user accounts allow each user to keep his or her own set of files, preferences and customizations. Note if your computer is set to login a user automatically, you may want to turn off autologins so that different users can log in to your machine. [REF turning off autologins]

The **adduser** command allows you to add user accounts. Here is the format:

```
adduser accountname
```

where *accountname* is the username of the account to create (for example, `linuxuser`). After running the command you will be prompted for the following information:

1. A password for the user. If you are having problems creating a good password, you can run the **apg** command to generate some possibilities. The user will need this password in order to log into the computer, so don't forget it. You will be prompted for this password twice.
2. A full name, such as "Charles McColm". This is optional, but it can be useful.
3. A room number, work phone, home phone, and "other" entry. It is acceptable to leave these blank.

Note that by default the new user will need to login using a password. If the user will never use this password (for example, if the user will always be automatically logged in) then it is safer to create an account that does not have a password. To do this, type

```
adduser --disabled-password username
```

If you use multiple user accounts on your systems, you probably want users to log in using passwords.

Managing autologins

Your computer may be configured to automatically log in a user when it boots. Two files are used to configure this behaviour. The file `/etc/wclp/autologin_user` contains the name of the user account to log in automatically. The file `/etc/wclp/autologin_enable` controls whether autologin is activated. If you wish to turn off autologins, you can delete this file. If for some reasons autologins have been disabled and you wish to re-enable autologins, type

```
touch /etc/wclp/autologin_enable
```

- -4pc - -4pc

to reactivate autologins.

In order to log in a different user account automatically, you need to change the contents of `/etc/wclp/autologin_user` .

These two files do not guarantee that graphical mode will start on bootup. If you change the `autologin_user` file and the user is logged into a command-line window (called the console), you need to change the autologin user's configuration. Look for a file called `.bash_profile` in the user's home directory. It should contain the following lines:

```
# WCLP addition: Try to start X if the terminal is
# linux (and this is tty1?)
ttytype=‘tty‘
autologin_enable=/etc/wclp/autologin_enable
is_xserver=/etc/X11/X
if [[ -f $autologin_enable && $TERM == "linux" \
      && $ttytype == "/dev/tty1" && -f $is_xserver ]]; then
        green=’\[\033[32m\]’
        nocolour=’\[\033[0m\]’
        echo "${green}Starting graphical mode... please wait ${nocolour}"
        exec startx
fi
```

The most important line is the "exec startx" line. If that is not present, then graphical mode will not start. By adding the above code the user you chose should be able to be automatically logged in.

Note that if your system was not set up to autologin users by default, adding the `/etc/wclp/autologin_enable` and `/etc/wclp/autologin_user` files will not enable autologins. In this case, you may want to install wdm or gdm to configure automated logins. [MORE INSTRUCTIONS?]

Managing software

One of the strengths of the open source movement is that a lot of software is available that you can try out for little or no cost. At some point you may find yourself needing software that is not available on your computer. If you use your computer to access the Internet, you will certainly want to keep your computer's software updated for security reasons. This section explains how the package update system on your computer works, and some of the tools you can use to install and maintain software packages on your system.

As usual, we will cover enough information to get you started. If you wish to learn more, a good document to read is the APT-HOWTO, available in the `apt-howto` package or on the Internet at [WHERE?].

How package management works

The software installed on your computer is packaged by a Linux distribution called Debian, which is well-known for its large selection of software and powerful software management tools. Debian organizes the software it releases into *packages*. Each package ends with the file extension `.deb`, and contains some software which provides some specific functionality. Packages usually have *dependencies*, which means that they require other packages to work properly. Complex software packages such as Abiword have dozens of dependencies, all of which must be installed

- -4pc - -4pc
on a computer before Abiword will run. Furthermore, the packages on which Abiword depends will themselves have dependencies, which in turn will have other dependencies.

There are several relationships packages can have with each other. A package can depend on, suggest, recommend, or conflict with another package. All of these relationships are called dependencies.

In addition to dependencies, packages also have *versions*. Newer versions of packages generally fix bugs and add features. Often packages will depend on specific versions of other packages, which further increases the headache of dealing with package management. Downloading and installing packages manually quickly becomes painful, which is why package management systems exist.

A package management system keeps track of which packages are installed on your computer, which packages are available, and the relationships between packages, and version information about all these packages. The package management system in Debian is called *APT*, which stands for "A Package Tool".

Fundamental to APT is the idea of a *repository*. A repository stores a collection of packages which could potentially be installed on your computer. Most Debian package repositories are on the Internet, but APT supports repositories on CD-ROM, local hard drives and local networks as well.

APT keeps track of the packages offered by all the repositories listed in the `/etc/apt/sources.list` configuration file. It organizes information about these packages, their versions, and their dependencies. When you select a package for installation, APT checks whether all of the dependencies for that package are satisfied. If not, it will automatically select other packages to satisfy the missing dependencies until all dependencies are satisfied. You can then install the resulting set of packages with a single command, which makes trying out new software very easy.

Debian Releases

In the open-source world, the word *distribution* has two meanings. A distribution refers to an organization whose job is to collect and package software that works together. Some popular Linux distributions include Debian, Red Hat/Fedora, Mandrake, Gentoo, SuSE, Arch Linux, and Slackware. A distribution also refers to the set of software packaged by a distribution.

Although in some sense WCLP is a Linux distribution, in reality the project is closely tied to the Debian distribution. We install Debian packages onto your computer, and you can install and upgrade software from Debian repositories. Learning a little bit about how the Debian project organizes its distribution will help you when you want to install new software on your system and when you want to get help.

Debian simultaneously maintains three collections of software: a "stable" distribution, a "testing" distribution, and an "unstable" distribution. Each collection has a codename, and the stable release has a version number. As of early 2005, the stable release was codenamed "woody" and versioned Debian 3.0, the testing release was codenamed "sarge", and the unstable release was (and always will be) codenamed "sid".

The Debian project works hard to ensure that the stable release has a high degree of quality: that the packages do not contain too many severe bugs, that the packages all work with each other, that security updates are available for the packages, and that the Debian installers work correctly to install the distribution. Software from the stable distribution tends to run well, but because the Debian project does not release stable distributions frequently, the software is often out of date.

The testing distribution is used to prepare the next stable release. The testing distribution offers a few guarantees about the software -- the packages should all be installable -- but the testing distribution typically contains some software with bad problems, and software in the testing distribution can change versions unexpectedly and quickly. Every so often, the Debian organization "freezes" the testing distribution, makes it stable as possible, and releases it as the new stable distribution. As of the current writing (early 2005) the Debian project is preparing to make sarge its stable release. Once it does so, woody will become unsupported, and the testing distribution will be renamed to "etch".

- -4pc - -4pc
When the Debian project decides to prepare another stable release, etch will be frozen and eventually be released as stable, and the testing distribution will get a new name.

The other distribution is the "unstable" distribution, which is where package developers release the latest versions of their packages. The unstable distribution is well-named; package versions change quickly, and sometimes installing the packages from the unstable distribution will break dependencies on your computer. The purpose of the unstable distribution is to find bugs and get the software ready to migrate to the testing distribution, although some people use the unstable distribution because it has the latest package versions.

The version of Debian used on your computer is specified by the entries in the file `/etc/apt/sources.list`. For example, the following lines come from the stable distribution, woody:

```
deb ftp://ftp.debian.org/debian/ woody main non-free contrib
deb-src ftp://ftp.debian.org/debian/ stable main non-free contrib
deb http://non-us.debian.org/debian-non-US stable/non-US main contrib non-free
```

The first line specifies the codename of the distribution ("woody") and the second and third indicate the type of distribution ("stable"). Either format is acceptable, but we use codenames in the `sources.list` files that we set up.

Depending on the age of the stable distribution, you will either be running the stable or the testing distribution of Debian. When starting out with your computer, you will not want to change these lines. However, as you gain experience (or as newer versions of Debian are released), you may want to change your distribution. You will do this by changing the entries in your `sources.list` . For instructions on upgrading your distribution, see [WHERE?]

Package Status

A package can be in one of several states on your system. The vast majority of packages are *uninstalled*. Some packages are *installed* and *configured* -- these are the packages that are usable on your system. A package can be installed but not configured -- in this case the package is known as *half-installed*, and it may or may not be usable on your system.

The APT database keeps track about packages you have installed, then removed. There are two senses in which a package can be removed. When a package has been *removed*, the program files are deleted but the configuration information is kept. This is useful if you later decide to reinstall the package -- since the configuration is preserved, the package will behave as if it had never been removed. A *purged* is one that has had both its program files and its configuration files removed from your system; it is as if you never installed the package.

Sometimes packages are *broken* on your system. This can happen when a package dependency is not satisfied, or when a package installation run failed for some reason. Often broken packages must be fixed before you can carry out further APT operations.

Using Aptitude

Aptitude is a *package manager* -- a program that allows you to browse, search through, install, upgrade and remove packages in your computer's APT database. It is interactive, but not graphical.

In this chapter we will describe how to use aptitude to manage packages on your system. Several other package managers exist for Debian. **apt-get** is a popular command-line package manager, and synaptic is a fully-graphical offering. You are welcome to experiment with these package managers if you wish, but we will focus on aptitude for the following reasons:

- -4pc - -4pc

- Unlike synaptic, it is installed on all WCLP machines.
- It is fairly easy to use, and fairly powerful.
- It offers an important feature (marking dependencies as automatically-installed) not supported by **apt-get**.

Unlike most administrative programs, you need not be the root user to use aptitude. You can start aptitude as a regular user, which is useful when you are simply browsing through the package lists. If you attempt some function that requires root privileges, you will be prompted for the root password.

To start aptitude in interactive mode, type

```
aptitude
```

The program will initialize with a `Loading Cache` message. Then you should see a screen similar to the following appear: [SCREENSHOT]

aptitude is a keyboard-driven program. There are a few basic keystrokes that you will want to learn:

- The q key quits a view. aptitude works with the idea of *views* -- every time you display a new information screen (for example, when looking at a package's dependency information) aptitude opens a new view. The q key quits the latest opened view.

  The moral of the story is that in order to fully-quit the program you may have to press the q key several times.

- The arrow keys are used for navigation. The enter is used to select a description (but not to choose a package for installation).

- The F10 key accesses the drop down menus. From these menus you can access most of the main functions in aptitude.

- To undo your last action, press ctrl-u . This is very useful when you accidentally select packages for removal.

- Selecting packages for installation and removal uses a funny set of keys that takes some time getting used to. The + key selects a package (and all its dependencies) for installation. The - key unselects a selected package, or selects a package (and maybe its dependencies) for removal. The = puts a package on *hold* -- it will stay installed or uninstalled until you specify otherwise. (This is useful when aptitude insists on removing a package you would rather keep. You can put the package on hold to tell aptitude to leave the package alone.)

- -4pc - -4pc

- The g key carries out a package installation/removal request. (The g stands for "get"). When pressing this key, a view will appear that summarizes the action you are about to take. Review this summary carefully before proceeding, and don't be afraid to opt out of the update by pressing q.

  If the summary seems reasonable, press the g key a second time to initiate the action. aptitude will then do one of two things:

  1. If you are running aptitude as the root user (or you previously typed in the root password to aptitude) then the action should proceed, installing, upgrading, and removing packages as indicated by the summary.

  2. If you are running aptitude as a normal user, aptitude will prompt you to type the root password. If you do this successfully, you will see the summary screen again. At this point you will have to type g a third time to actually initiate the action.

- The u updates package information in the local APT database. You want to do this if you have added an entry to your `/etc/apt/sources.list` file, or if you have APT sources on the Internet.

  Updating package lists requires root privileges, so you will be prompted for the password if you have not authenticated your root privileges with aptitude yet.

  You will usually want to update package information before installing and removing packages, in case package versions have changed.

- The U key *upgrades* packages -- it tells aptitude to upgrade all the packages it can to their latest versions. Often, you will want to upgrade your packages after updating package information.

  The U key does not actually upgrade the packages. It simply selects applicable packages for upgrade. To complete the upgrade you must press g (and possibly enter your root password).

- Packages in aptitude are grouped into categories. Categories are nested in each other. When you have navigated to a section title, you can press Enter to open the category, revealing subcategories. Pressing [will open the category and all of its subcategories, and pressing] will close the category and all of its subcategories.

  For example, using the arrow keys to highlight the `Installed Packages` category and pressing [will show all the packages that are installed on your computer.

  Package selection keys (+, -, =) also work when category titles are highlighted. Pressing a package selection key will apply that action to all packages in the category. This is usually not what you want to do, so use this feature with caution.

- Highlighting a package name and pressing Enter will reveal further information about that package, including its dependencies, its size, and a long description.

- One of the most useful keys is the search key, which is / . Pressing this key and then entering a word will navigate to the nearest package that contains that word. You can repeat searches by pressing the n.

- -4pc - -4pc

Perhaps the best way to illustrate aptitude is to use it to explore some packages:

1. As a regular user, start aptitude by typing

   aptitude

2. If you have access to the Internet and you have Internet sources in your `sources.list`, press u to update the APT package information. You will be prompted for the root password, so enter it.

3. Next, let's search for the jester package. Press / and type "jester". Note that as you press each key the display is updated, so you may not need to type all the characters.

4. Once you have highlighted the jester package, press Enter to see the package details. You should see that the package depends on `libc6` and `xlibs`, which are very common dependencies for graphical programs.

5. Press q to leave the package details view, and then press + to select the package. At the top right corner of the screen you will see two numbers appear: one that shows the amount of additional space this package will take up, and one which shows the amount of information that must be installed.

6. To install the package, press g. You will see the packages that will be installed along with jester. To continue the process press g again, enter the root password when prompted, and press g a third time. (If you do not have access to the Internet and you are using Internet repositories, you probably want to skip this step.)

7. The package (and its dependencies, if they are not installed already) will be downloaded (from CD or the Internet, depending on your `sources.list`), then installed. Some packages prompt you for configuration information. [SCREENSHOT -- MAYBE OF XFREE86?] These questions are designed to have sensible defaults, so if you are unsure of how you should answer the questions posed, just select the defaults.

8. After the package has been configured and installed, you will be returned to a aptitude package browser. If you selected the jester package again and pressed -, then the package would be selected for removal. (You do not need to remove the package if you do not wish.)

9. Next, we can examine the `abiword` package. Type /, and then type "abiword". Since several packages (`abiword-doc`, `abiword-common`, `abiword-gnome` and others) begin with the word "abiword", you may have to repeat the search several times by pressing the n key.

10. Once you have highlighted the `abiword` package, press Enter to examine the package details. You can see the description, the download and install sizes for the package, and lists of dependencies. Scroll through the dependencies until you have highlighted `abiword-common`, and press Enter again. This will display the available versions for `abiword-common`. Highlight a package version and press Enter to see the package details for `abiword-common`.

11. In the `abiword-common` package, you will see different dependency categories displayed. Here is an explanation of what they mean: [THIS SHOULD NOT BE HERE]

    • The packages in the *Depends* category are needed by `abiword-common` in order to run.

- -4pc - -4pc

- Packages in the *Recommends* category are not strictly necessary, but they provide important and/or frequently-used functionality for the package. aptitude will install Recommends dependencies by default (although this can be turned off).

- Packages in the *Suggests* category add enhancements to the package that some users may find useful. A "suggests" dependency is less important than a "recommends" dependency, and aptitude will not install these dependencies by default. If you wish to install "suggests" dependencies for a package, you will have to select them with the + key.

- Packages in the *Conflicts* category cannot exist on the system at the same time as the `abiword-common` package. This sometimes happens when two packages provide the same program, or when a package was renamed between versions. If any of the conflicts dependencies are installed when you select the `abiword-common` package, the packages that conflict will be marked as "broken". [REF BROKEN PACKAGES]

- The *Replaces* category lists old versions of the package with different names. These packages will be removed when the new package is installed.

- The section labelled *Packages which depend on abiword-common* displays packages that list *abiword-common* as one of the dependencies described above. If you were to remove `abiword-common` and discovered that some other package that depended on `abiword-common` broke, you could look through the packages in this category to see if you could find the culprit.

- The *Versions* category lists all the versions of this package the APT database knows about. Sometimes you can select from multiple available versions of a package (but you probably do not want to do this unless you know what you are doing).

Feel free to explore the `abiword-common` package and its dependencies for a while. Then press q to get out of this view and back into the view for `abiword`, and press q again to get back into the main view.

12. Highlight the "Installed Packages" category and press [. This will display all the installed packages in all categories. You can then explore your installed packages. Pressing] on an open category will close it up.

13. If you have Internet sources in your `sources.list`, then you may want to upgrade the packages on your system. Press U and then g to install the packages. Note that you are not prompted for the root password a second time; aptitude remembers that you have entered the root password once already.

14. After upgrading the packages on your system, you might want to quit the program. Press q until you get the `Really quit aptitude?` prompt. Press Enter to exit the program.

This walk-through should give you a sense of how to use aptitude and some of its basic functions. For more information, you will want to read the manual, which can be accessed in the `Help` menu.

Finding Broken Packages

When selecting packages for installation or removal, you might inadvertantly break the package dependencies. In aptitude you will see a count of broken packages appear in the top left of the display. Package dependencies break for a number of reasons:

- Some package has a dependency that is no longer met. This can happen because of version conflicts, or because APT cannot find the dependency package in its database.

- Two or more packages are marked as conflicting with each other.

- -4pc - -4pc

You want to fix broken dependencies before attempting to install or remove additional software on your system. Often aptitude will refuse to complete package installations when packages are broken.

The first step in fixing broken dependencies is to identify the broken packages. The easiest way to find broken dependencies is to go to the main view and search for them. Press the / and type

```
~b
```

to find broken packages. Looking at the package details should tell you why the package is broken. Depending on the reason, you have the following options:

- You may decide to remove the broken packages. This will work if other packages do not depend on the packages you are removing, or if a package you are installing provides the same functionality as the package you are removing.
- You may decide to undo the changes that caused the broken dependency. For example, if you selected a conflicting package for installation, you might consider removing it.

[ARE THERE OTHER THINGS HERE? WHAT ABOUT PUTTING PACKAGES ON HOLD?] [SCREENSHOT]

Running aptitude from the command line

You can run many aptitude operations from the command line, without fussing with full-screen mode. This is useful if you know exactly the names of the packages you wish to install/remove, or if you are just upgrading the packages on your machine.

Unlike interactive mode, you should be root before using aptitude to do package management from the command line.

Here are some examples that illustrate the use of aptitude on the command line. To install package `jester`, type

```
aptitude install jester
```

To remove package `jester`, type

```
aptitude remove jester
```

To update your package lists, type

- -4pc - -4pc

```
aptitude update
```

To upgrade the packages on your system, type

```
aptitude upgrade
```

You can also search for packages in aptitude from the command line, but unless you are writing shell scripts using interactive mode is easier.

Finding software

Often you will find yourself looking for software that accomplishes a particular task, but you will not know the names of software that accomplishes that task. In this section we will discuss some strategies and tools you can use to discover candidate packages.

For example, say you were looking for a spreadsheet application, but you did not know the names of any spreadsheets that run under Linux. How could you discover the names of spreadsheet packages?

- A good starting point is to use the command **apt-cache**. This command can search through all the descriptions of all packages your APT database knows about. A good starting query to **apt-cache** might be:

  ```
  apt-cache search spreadsheet
  ```

  The "search" parameter tells **apt-cache** to return a listing of all the packages whose names or descriptions contain your search term. The command searches all the packages in your computer's APT package database.

  Once you have found some promising package names, you can read the long descriptions by typing

  ```
  apt-cache show packagename
  ```

  For example, you could type

  ```
  apt-cache show gnumeric
  ```

  to display information about the Gnumeric package.

- -4pc - -4pc
You can also perform these operations in aptitude. Press / to start a search, and then type a "~d" before your search term to search through the package descriptions. For example, the following search term finds a package containing the search term "spreadsheet":

```
~dspreadsheet
```

As usual, you can repeat searches by using the n key.

- In addition to generic software types, you might search for the names of common data formats and competing software. For example, the following search terms can all be effective in finding spreadsheets:
  -
```
apt-cache search xls
```
  -
```
apt-cache search csv
```
  -
```
apt-cache search excel
```
- In aptitude, you can often explore package dependencies to turn up lists of packages that provide the same functionality. For example, the package `www-browser` is a *virtual package*. Other packages *provide* this virtual package if they provide a web browser. Thus, the `www-browser` package will reveal browser alternatives.
- The packages in Debian are organized into categories. For example, the "web" section collects all programs that are Internet related. You can browse through these programs to find web browsers. Similarly, gnumeric is in the "math" section. Browsing through this section can reveal other spreadsheets.

- -4pc - -4pc

- The Internet is good way to search for package names as well. Search for the software you are looking for, and the words "linux" or "debian". For example, you might search for the terms "spreadsheet linux" or "spreadsheet debian" or "excel xls linux" to find spreadsheets.

  If you find the names of spreadsheet programs that are available you can then search for their package names using **apt-cache** or aptitude

  Sometimes people publish lists of spreadsheet alternatives. To find these lists, you might search for all the alternatives you know. For example, the search terms "linux gnumeric oleo siag" might turn up a list of Linux spreadsheet alternatives, which may include alternatives you have not listed.

Security Updates

One of the most important reasons to learn about package management is to install *security updates* -- packages which fix important security holes that others could exploit to break into your computer or worse. [REF safe_computing] The Debian project releases security updates for its distribution at
<url>http://security.debian.org</url>
. If you have a line similar to the following in your `/etc/apt/sources.list`, you are able to get security updates for your packages:

```
deb http://security.debian.org/ stable/updates main contrib non-free
```

You can fetch and install security updates by using the upgrade function of aptitude: first update the package lists, and then upgrade your packages.

If your computer has access to the Internet, you should be installing security updates frequently. Once a week is not too often, and some people check for security updates every day. Once security holes become known, it does not take long before they are used to break into computers.

Using dpkg

The **dpkg** command is used to manage individual packages. Usually you will not deal with **dpkg** directly. Instead, you will use APT, which will fetch all the package dependencies and then use **dpkg** to install the individual packages on your machine. However, some programs (notably Opera) do not have associated APT repositories. In these cases you may have to download the `.deb` package manually and use **dpkg** to install it.

In order to use this method, you first need to download the package and all of its uninstalled dependencies into a directory on your computer. Then you would become the root user and type

```
dpkg --install *.deb
```

This will attempt to install all the packages at once. The **dpkg** command will select the packages for installation, unpack them on the hard disk, and attempt to configure them. If dependencies are missing then **dpkg** will fail at the

- -4pc - -4pc
configuration step. You will then need to download the dependencies, install them using **dpkg** and possibly execute the following command:

```
dpkg --configure --pending
```

to fix the broken packages.

This process is error-prone, and you should avoid it when you can. The case of Opera is not that bad, however, because in most cases you only need to install one package.

Listing packages

The **dpkg** command can also be used to generate a textfile listing all the installed packages on your system. To do this, type

```
dpkg -l
```

As this command generates a lot of output, you may want to pipe the command through **less** or redirect it to a file:

```
dpkg -l > /tmp/package-listing
```

This will generate a list that looks something like the following:

```
Desired=Unknown/Install/Remove/Purge/Hold
| Status=Not/Installed/Config-files/Unpacked/Failed-config/Half-installed
|/ Err?=(none)/Hold/Reinst-required/X=both-problems (Status,Err:
uppercase=bad)
||/ Name        Version      Description
+++-==============-==============-============================================
ii  aalib1       1.4p5-13     ascii art library
ii  acroread     4.05-3       Adobe Acrobat Reader: Portable Document Form
ii  adduser      3.47        Add and remove users and groups
ii  anacron      2.3-6        a cron-like program that doesn't go by time
ii  apt         0.5.4       Advanced front-end for dpkg
ii  apt-utils     0.5.4        APT utility programs
rc  apt-zip      0.13.2       Update a non-networked computer using apt an
```

The first two letters indicate the install status of the package. In the above example, all of the listed packages are installed except for `apt-zip` which has been removed. As indicated, the remaining columns indicate the package name, the version, and the short description of the package.

```
- -4pc - -4pc
```
You can also get a list of package selections by typing

```
dpkg --get-selections
```

This will generate simpler output:

```
aalib1 install
acroread install
adduser install
anacron install
apt install
apt-utils install
apt-zip deinstall
aptitude install
```

This just lists the name of the package, followed by either "install" or "deinstall". The output of this command can be used to tell **dpkg** which packages to install on another computer using the `--set-selections` flag of **dpkg**.

Associating files with packages

The **dpkg** can also be used to search for the associations between files and packages. For example, typing

```
dpkg --search sensible-browser
```

will tell you that the `debianutils` package installed the **sensible-browser** command.

Upgrading the entire distribution

If you keep your computer for long enough, you may find that your version of Debian becomes obsolete and another version is released as the stable distribution. In this case you can upgrade your system to the new distribution without having to reinstall your computer from scratch. You simply need to change your `sources.list` entries to reflect the new distribution, and then use aptitude to upgrade all of your packages.

Some people mix distributions on one system, but unless you learn more about APT and its tools, you should not attempt this.

Important files and directories

Linux hides useful information about your computer all over your system. These resources are one of the reasons Linux is so useful, but learning about these resources can be tough. This section documents some useful files and directories that can make your administration life easier.

`/etc`

- -4pc - -4pc

System-wide configuration files are kept in `/etc`. If you need to configure some service or program that affects all users, you want to look here. Generally, the configuration files for a package will bear that package's name.

Configuration files in Linux are usually plain text files, which you can edit with an editor. Before editing any configuration file, you should make a backup so that you can restore settings if you mess up.

In addition to system-wide configuration, individual users can control the behaviour of applications using *dotfiles* in their home directories. A dotfile is a file or directory that begins with a period (and so is hidden when typing **`ls`** but revealed when typing **`ls -a`**). Unless you want to change an application's configuration for every user on the system, you often want to change the appropriate dotfiles in your home directory instead.

Runlevels and `/etc/init.d`

Functionality such as printing, mail delivery, and the firewall are called *services*. Services on your system are controlled by *daemons* -- special programs started on bootup. These programs wait for some event to happen (for example, somebody trying to print to your printer) and then take action.

Daemons are started and stopped by scripts in the directory `/etc/init.d/` . These scripts are called in a standard way; they take the options `start`, `stop`, `restart`, and `reload`. For example, to stop the firewall script, you could type

```
/etc/init.d/firewall stop
```

Runlevels

A *runlevel* determines the set of services/daemons that are allowed to run at any given point. When Linux enters a runlevel it will start and stop services so that the set of running services matches the services that are allowed to run at that time. Runlevels are important primarily when booting and shutting down the system.

There are eight runlevels that exist on your system, although in practice only five of them are distinct:

- Runlevel 0 is used to shut down the system. As no services should be running when the system halts, the purpose of this runlevel is to stop all the daemons.
- Runlevel 1 is used for *single-user mode*. This mode is used for maintenance: only the root user is allowed to log in, and most services are not started. Administrators typically use this mode when something serious has gone wrong (such as a hard drive's data becoming corrupt) and they need to make sure nobody else is using the system to fix things.
- Runlevels 2-5 are used for *multiple-user* mode. By default these runlevels are identical. When your system starts, it will go into runlevel 2. Runlevels 3-5 will probably not be of much use to you unless for some reason you need to run different sets of services at different times.
- Runlevel 6 is used when rebooting the system. It is fairly similar to runlevel 0 in that most services are stopped.
- There is also a runlevel called runlevel S. Its purpose is to help make the transition to runlevel 1.

```
- -4pc - -4pc
```

The **init** command is used to enter these runlevels. For example, if you wanted to enter single-user mode, you would type (as root)

```
init 1
```

You will usually not use this command directly.

Your system identifies the services to start and stop at each runlevel using *symbolic links*. Each runlevel gets its own directory: runlevel 0 uses `/etc/rc0.d`, runlevel 1 uses `/etc/rc1.d`, and so on.

Inside each of these directories is a set of symbolic links. The symbolic links point to the scripts in `/etc/init.d`. Each link is named in the following way:

- The first character is ‘S’ if the service should be started on that runlevel, and ‘K’ if the service should be stopped ("killed") on that runlevel.
- The next two characters are digits which specify the order that the scripts will be executed. Scripts with earlier numbers are executed first.
- The rest of the name is the name of the script in `/etc/init.d/`

If for some reason you need to modify the services that run at some runlevel, you have two options: you can modify the symlinks directly, or you can use the **update-rc.d** command. For example, to disable the firewall from all runlevels, you would type

```
update-rc.d -f firewall remove
```

This would remove all the symbolic links in the `rc.d` directories, but it will not touch the original script in `/etc/init.d` . Note that this command alters the services you run permananently; if you just wanted to temporarily stop the firewall, you would type

```
/etc/init.d/firewall stop
```

and the firewall would be disabled until you rebooted or turned it on again.

Networking control files

Files to configure your Internet connectivity are scattered throughout `/etc`. Each file controls a different aspect of networking:

- -4pc - -4pc

- If you have a network card in your machine, the file `/etc/network/interfaces` determines how the network card is configured. The network card is named `eth0`. A line in the `interfaces` file reading

```
auto eth0
```

  determines whether you connect on to the Internet upon booting up. If the line is present and not commented out, then your network card attempts to connect to the Internet (or whichever network you are on) upon booting. If you are not always connected to the Internet, you may want to comment this line out:

```
#auto eth0
```

  Then you would type the following command (as root) to start networking manually:

```
ifup eth0
```

- The file `/etc/resolv.conf` configures *name servers* -- computers that turn numerical Internet addresses (such as 142.168.1.1) to domain names (such as "debian.org"). Often this file is configured for you automatically.

/var/log

Many daemons and applications write information to *logfiles*, recording status messages and errors. When things go wrong, it is often worth looking at the log files for errors and other unusual events.

Programs are supposed to keep their logfiles in the `/var/log` directory. Some files and directories of note include:

- The `XFree86.0.log` file records information about graphical mode startup. If you find that graphical mode does not start, this file may tell you the error. The file is regenerated every time the computer attempts to start graphical mode.
- The `syslog` is a general log. Many different programs record information to this file, so it is a good place to look if you cannot find a more specific file.
- The `aptitude` file logs operations you have performed using aptitude. Use this when you are wondering why you installed a particular package.
- The `debug`, `messages`, `kern.log`, and `messages` files store messages from the kernel.

- -4pc - -4pc

Often you will not want to look through these files manually. Rather, you will want to search through the files for expressions using **grep**.

Finding Information About Your Computer

When troubleshooting problems or trying to configure hardware to work with your computer, it often becomes important to identify your hardware components as precisely as you can. This section documents some software tools that you can use to identify components of your computer.

Physical Identification

One technique for identifying hardware components on your computer is to look for labels and model numbers. This is one of the best ways to identify the makes and models of external components such as monitors and printers.

Often it is possible to read marking on the internal components of your computer, such as your video card, network card, or sound card. In such cases you want to look for stickers or model numbers on the chips or circuit boards. One handy identifier is the *FCC identifier*, which is [WHAT? -- FINISH ME] However, in order to read these stickers you will usually have to open your case and pull out the hardware components, which is a bad idea unless you know what you are doing.

Opening computer cases and looking for identifying marks used to be a common strategy for identifying hardware. These days, software tools make identification easier and more accurate for most hardware, the exception being very old expansion cards (specifically, some modems and cards that use the *ISA* bus).

Software tools

[DISCOVER, DETECT, MDETECT, LSPCI, ISA-PNP]

The `/proc` filesystem

The *kernel* is a special program that is central to Linux. It controls your hardware, allocating resources such as CPU and memory to running programs so that your computer operates smoothly. A special directory named `/proc` lets you peek at the kernel's operations. The files in this directory reflect the state of the hardware, and information about all the running programs. To view these files, you can use **less**. This comes in very handy when you are troubleshooting problems, or when you and Linux disagree about your hardware. Here are a few useful files in `/proc`:

- `/proc/cpuinfo` provides information about the CPU installed on your computer, including its make, model, and speed.
- `/proc/interrupts` displays *IRQ* information. An IRQ is a channel that hardware devices (such as sound cards, network cards, and the keyboard) use to communicate with the CPU. This file is useful if you ever have to deal with *IRQ conflicts*, which occur when two or more devices want the same IRQ. Hopefully you will never have to deal with an IRQ conflict, but this file is indispensible if you do.
- The files `/proc/pci` and `/proc/isapnp` display information about devices (in particular expansion cards such as graphics cards and network cards) installed on your computer.