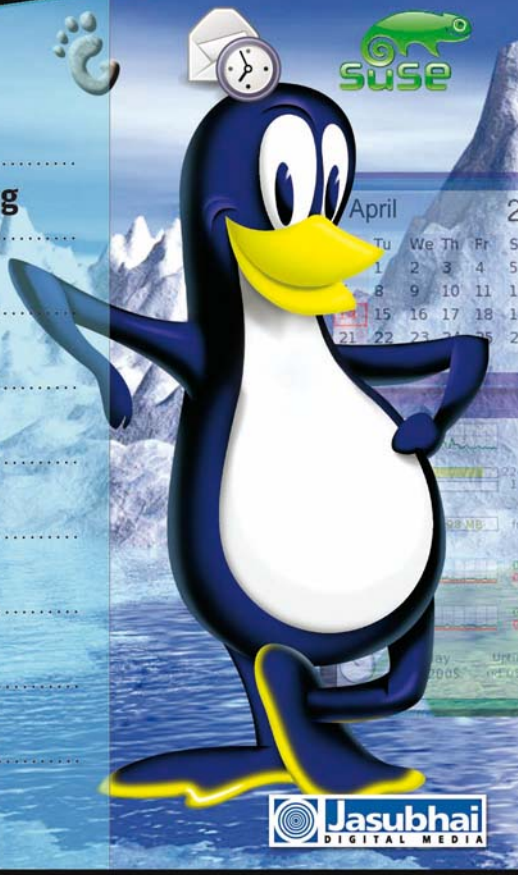
digit
April 2006

Fast Track to LINUX



YOUR HANDY GUIDE TO EVERYDAY TECHNOLOGY

Fast Track to Linux

By Team Digit

Credits

The People Behind This Book

EDITORIAL
Deepak Ajwani Editor
Robert Sovereign-Smith, Copy Editor
Ram Mohan Rao, Copy Editor
Anup Nair Writer
Kailash Shastry Writer

DESIGN AND LAYOUT
Vijay Padaya Layout Designer
Sivalal S Cover Design
Harsho Mohan Chattoraj Illustrator

April 2006
Free with Digit. Not to be sold separately. If you have paid separately for this book, please e-mail the editor at **editor@thinkdigit.com** along with details of location of purchase, for appropriate action.

The Other Operating System

Why this Fast Track on Linux? Well, mostly because we've spoken so much about it in *Digit*, and because there are many of you out there who, even then, have no clue that The Other Operating System can actually be used by human beings.

We don't aim here to give you high-flying tips—just a mild introduction, but we cover what's necessary for a first-timer to get started.

We start off with a history lesson, and with good reason: a knowledge of Linux's open source and community-driven beginnings is important to fully appreciate what it is today. Then follows a workshop-like chapter on how to get started using Linux—installation, post-installation procedures, some basic commands, and so on. File management in Linux is more "advanced" than in Windows—meaning you can play around more with the filesystem—and chapter 3 is therefore devoted entirely to files. Then follows, in chapter 4, a discussion of the X window system and the Gnome and K desktop environments.

Chapters 5 and 6 are about stuff you can do on Linux—text editing, image manipulation, multimedia—you'll find out for yourself later, but take our word for it now: Linux isn't only about kernel recompiling and C programming at a command prompt! We proceed to tell you how to use your Linux machine for gaming.

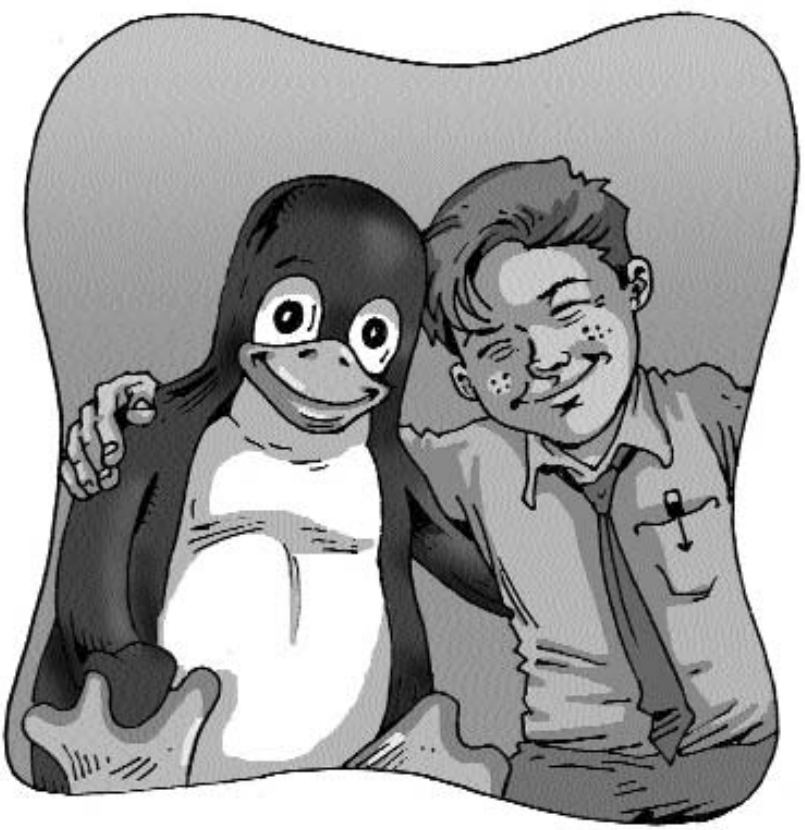
No discussion of Linux can be complete without talking about the networking aspect—because Unix and its derivatives are essentially server-oriented OSes—and that we do in chapter 7. And if you've decided by then that you want to try out a Linux distro (or flavour, if you like to call it that), chapter 8 is about the differences between the most common distros, and their salient features.

We round off the book with a discussion of system administration in Linux, and finally, we point you to Web resources.

It's our sincere hope that even if you don't actually make the switch from Windows, you will experiment a little to get an idea of the variety of choices you have!

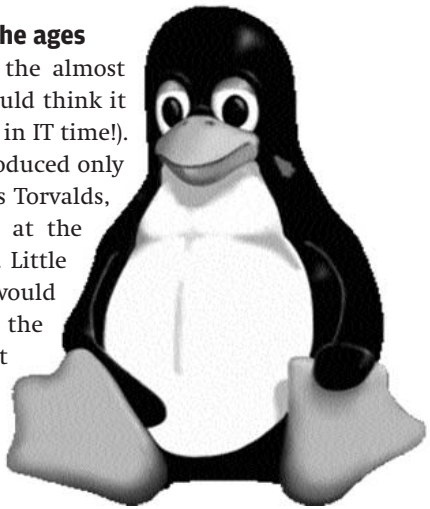
Contents

What Is Linux?



From servers to ipods, hardcore programmers to enthusiasts, all inclusive 6GM installation to floppy bootable, Linux is everywhere. Linux is standing today showing the IT world the innovation that can be achieved by sheer community work. And, Linux is free too. It appears, there is after all something like a free lunch! When the world around you is trying Linux, why would you want to be leaving behind the excitement?

1.1 A Brief History Of Linux



The penguin's growth through the ages

Considering the geek value and the almost cult status of Linux today, you would think it has been around for eons (at least in IT time!). Actually, Linux was officially introduced only in 1991 by its famed creator Linus Torvalds, who at the time was a student at the University of Helsinki in Finland. Little did he know that his creation would grow in size and popularity to the extent it has today. That he did not expect too much to come forth from his work is evident from this much-too-famous excerpt of his mail to a newsgroup:

"*Hello everybody out there using minix - I'm doing a (free) operating system (just a hobby, won't be big and professional like gnu) for 386(486) AT clones. This has been brewing since April, and is starting to get ready. I'd like any feedback on things people like/dislike in minix, as my OS resembles it somewhat(same physical layout of the file-system (due to practical reasons) among other things).*"

He ended his mail saying:

"PS. Yes — it's free of any minix code, and it has a multi-threaded fs. It is NOT portable (uses 386 task switching etc), and it probably never will support anything other than AT-harddisks, as that's all I have :-(."

Let's look at some of the interesting things that happened before and after that mail. The story of operating systems—especially that of open source OSes—is inspiring to anyone who loves technology. Linux will only be wholly appreciated when the "behind the scenes" of its development is known.

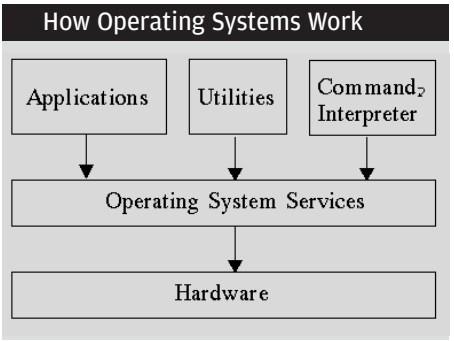
The twists and turns that have shaped Linux warrant a separate book, but here we present what we think are the key points in the evolution of Linux.

OK, so you already know Linux is an OS—that's a good starting point. Operating systems are almost as old as computers themselves. You see, one can't quite live without the other. Simply put, the OS is like the engine for a car: it is what makes everything else run.

1.1.1 The operating system

Think of any device—not just a computer—on which applications can be installed: there has to be an OS. For example, your mobile handheld that allows you to download and install applications has an OS.

The concept of the operating system came about during the 1950s and 60s. This was the time when tremendous progress happened in the field of computing (or for that matter, in most fields. Remember, this was the time when the world was rebuilding itself after the two world wars, and science and technology saw peaceable applications). Computers at that time were used only in research institutes, universities, and large banks. (But the tasks these computers did were a lot simpler than what is being thrown at our monster machines now!) As greater performance was expected off computers, better hardware and OSes came to be created and used.



A schematic representation of the way an operating system functions

In this period, there was a shift from batch processing to the time sharing method of computing. This simply meant that

instead of executing a single program continuously until it was over, the computer executed chunks of different programs sequentially so that several programs (or users) were catered to simultaneously. To do this, a sort of logic was required on how programs would be stored and given to the CPU, how the results would be stored and retrieved for the next round of processing, which user or program had to be prioritised, and so on. This task was handled by separate pieces of software that, in a way, controlled how other programs were executed in a system. Later on, as a larger variety of hardware came into use, software was needed that would be an interface between the programs and the hardware. This need for interfacing applications and hardware, and controlling how applications were serviced, led to the development of the operating system.

So, what OS was being used in those early computers?

Well, it's not quite possible to make a list of operating systems of the earlier days as we can today. This is because there was no generic OS, and operating systems would be written specifically for each machine or purpose. Some notable OSes at that time were FORTRAN Monitor System, General Motors Operating System, Admiral, B-series, and Honeywell Executive System. There were many more besides these, owing to the variety of hardware that was being developed. But for our purposes, it is enough to say that the OSes at that time greatly helped improve the performance of those primitive machines.


A Multics System at use

One of the big initiatives undertaken to develop an OS at that time was the Multics project at MIT in 1965 by Bell Labs with

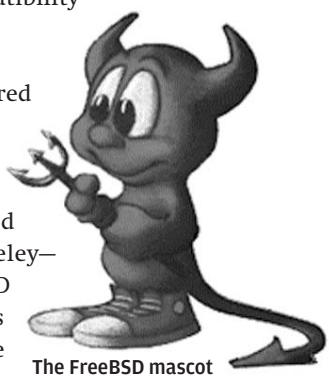
GE and MIT as partners. Multics (Multiplexed Information and Computing Service) was a time-sharing OS that ran on a special, expensive CPU. It was later marketed by Honeywell as a commercial product to the US Government, and to auto makers, universities, and commercial data processing services. But as it went on, the project sailed through rough waters and experienced delays. Eventually, Bell Labs withdrew from the project in 1969.

A lot more happened in 1969—apart from the moon landing and Bryan Adam's Summer! The scientists from Bell who worked on the Multics Project developed the first-ever UNIX, an OS meant for servers. Linus Torvalds was born the same year.

1.1.2 UNIX

When Bell Labs developed UNIX—one of the most well-known operating systems of all time—it was initially distributed for free, and gained great popularity in universities. The big boost for UNIX came when the TCP/IP protocol stack was implemented, and UNIX was adopted as the OS of choice for early workstations and servers. Quick development followed, and by 1979 the seventh version of UNIX was introduced. During this time (1972 to be precise), Dennis Ritchie developed C, the base for almost all medium- and high-level programming languages developed after it till date. C was used to write software on the UNIX kernel to make a complete OS. C, therefore, can be very easily used to write applications for UNIX, with compatibility not being an issue.

Apart from Linux, UNIX has inspired other OSes as well, notably BSD, short for Berkeley Software Distribution. BSD, a UNIX-based OS that can run on various CPU architectures, was developed in the University of California at Berkeley—hence the name. Outside of Bell Labs, BSD contributed the most to UNIX. It was intended more for experimental use



The FreeBSD mascot

than for commercial purposes. It was released in various flavours, first in 1977, and is still in vogue. FreeBSD is a flavour that is free for use, and is maintained by a large group of individuals driven by shear passion.

Many upgrades and newer versions followed as the server market continued to rely on the UNIX platform. In 1984, the Free Software foundation started the GNU (GNU is a recursive acronym for "GNU's Not UNIX") project, which aimed—out of enthusiasm—to develop a complete, free UNIX-like OS. Linux was to later use a lot of resources from GNU.

While all this was happening in the UNIX world, Microsoft introduced MS-DOS in 1981 and Windows 1.0 in 1983. However, network support came only in 1984, when a lot of systems were already running UNIX or its derivatives. Microsoft was thus a late entrant into the OS world, but owing to PC penetration into various markets—especially homes—Windows started getting popular.

In 1991, Sun Microsystems released their Solaris 2 OS. And in the same year, Linus Benedict Torvalds introduced Linux.

Torvalds had used UNIX at the University of Helsinki, and wanted to use UNIX at home too. He had a pretty powerful machine (for those days, we mean)—a 386DX 33 MHz CPU, 4 MB of RAM, but no co-processor (generally there is a math co-processor made specially to handle mathematical operations), and a 40 MB hard drive. He tried to get UNIX for his computer but could not afford it. Hence, he started out with Minix, a UNIX clone that could run on Intel processors, and which was used to teach UNIX at the University of Helsinki. Minix's source code was not entirely available, so Torvalds wanted something better, and soon started to write his own OS.

After a few revisions, it was moderately functional, and soon attracted the attention of several "hackers," who were interested

in building an OS that felt like UNIX but with its internal workings being entirely different.

We need to understand that Linux by itself is only the kernel of an OS. Several critical programs that the user needs are to be written to run on top of the kernel. Torvalds did not write all these programs himself: he used existing, free versions, thus greatly reducing the amount of coding he had to do to make a fully-functional OS. Torvalds worked in depth with the kernel—to such an extent that he often changed the kernel to make it easier to get the existing programs to run on Linux. Generally, the programs are modified to suit the kernel, but Torvalds was a kernel tweaker rather than application writer. Most of the important system software, including the C compiler, was borrowed from GNU project. (More on the GNU project later.)

Linux is what it is today because of the help of developers who worked on it not for money but for the kick of it. They were driven by passion and belief in a cause. Linux is perhaps the single largest collaborative effort in the IT world today. It is not easy to believe that something done out of sheer passion and the fun of programming is now used in several computing devices from PCs in homes to handheld devices to large servers. Torvalds, without even realising it, had written some of the most revolutionary pieces of software in that period.

Linux also carries with it a socio-cultural impact. Torvalds came to the US at the time when Microsoft and Netscape were heavily into the browser wars. People started looking at Torvalds as the man who could take on the battle against Microsoft.

Here onwards, the developments are vast and varied. Just three years later, in 1994, Red Hat Linux and Caldera were released. Red Hat is by far the most popular distribution of Linux. (Red Hat carried with it in-depth documentation and help for all kinds of users, making it a favourite with newbies). Several distributions exist today, some popular, some not so much, and some only for the geek.

1.2 Open Source—Where Everyone Can Be A Developer

OK, that sounds a little too optimistic. But then that's what open source proponents claim. You've most probably encountered the term "open source" somewhere—in a magazine, on TV, with peers, or even in a newspaper. Before we proceed to install and play with Linux, it helps to understand just what open source is. After all, Linux and open source are often uttered in the same breath. So what's open source all about?

A source code is the program code exactly as it is written by the programmer: the raw code that is not compiled and is editable is the source code. (Compilation is the process of finalising a code for execution.) For an operating system, the source code runs into several thousands of lines that appears "as written." (Needless to say, deciphering the logic behind the source code can be quite a task even for a competitive programmer.)

Since the source code is in raw, uncompiled format, anyone who has access to it can make his own additions, deletions or changes—and hence his own versions of the software with modified or added features. Obviously, with more people working on developing OSes and applications, the number of variations and innovations are huge, and open up exciting possibilities—because we can have developers all across the globe working towards creating new software. These developers need not work for a particular company or for direct monetary benefit. Enthusiasts can contribute to mainstream software. Students can learn the insides of software. All this, of course, only when the source code of the software is made freely available. And that is what the open source idea is all about.

1.2.1 Free Software

A closely-related but not similar idea is that of free software. Free software means, well, that the software is available for free. It is already compiled and ready to use. Open source only means the source code is available to whoever wishes to build around it and make fully-functional software. Free software need not be open

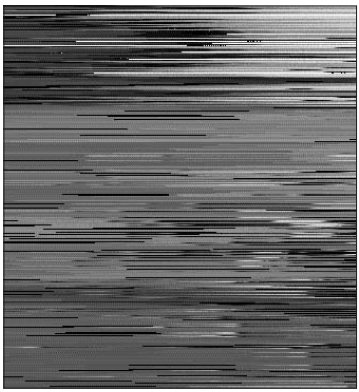
source, and similarly, open source software need not be free either. A free, but not open source, software can only be used, but cannot be modified—or for that matter dissected to see how it works. Open source software can also be *paid* software, in which the source code is made available to the buyer. Open source and free software are closely related in the Linux world, but they are not the same.

The idea of free software or open source is in fact older than that of commercial (paid) software. Computers were first used for academic and research purposes; software programmers were paid for their efforts and their programming skills, but *not* for the programs themselves. In other words, once the software was written, it was the property of the user and not of the developer.

But when computers reached banks and large establishments, the earliest software companies and programmers saw a huge potential for making money. Software, though not a physical entity, attained the status of a product. Software writers began to charge the users for every copy of the software deployed. Thus the software ownership began to rest with the programmers, and this led to the "paid software" concept. This idea seems simple. If you design a car and sell it, you charge for every car you sell. This is not only for the cost of materials but also because anyone who uses the car owes you something for your ideas and skills. (Well, in software, there is no material cost for each copy—and that's an argument that open source / free software advocates put forward.) Software that was once the property of the user is now largely the property of the maker. Some analysts believe that this will come full circle and that the trend again will be towards open source.

No discussion on open source or free software is complete without the mention of Richard Stallman, the founder of the GNU project, who has written numerous books and white papers on open source. His *Free Software as a Political Idea* is one concept and ideology that has inspired the open source world. He designed a set of rights that he feels all users are entitled to. These have been

documented in the *GNU General Public License*, or GPL. Stallman himself developed some free software such as the GNU C compiler, and GNU Emacs. (Emacs is a text editor and interpreter for Emacs Lisp, which is a version of the Lisp Programming language.)


Richard Stallman

Several discussions and debates have taken place amongst representatives of commercial vs. open-source, free software. Microsoft, too, has participated in a common platform discussion on the subject along with Linux distributors. But the two are as far apart as can be. Microsoft (we can't leave them out in any discussion, can we?) believes in commercial software as its business model. Open source fans are wary of monopolisation, and say that sharing of technologies is the right way to go about building the Internet. In the words of Michael Tiemann, Ex CTO and current Vice President, Open Source of Red Hat:

"The stakes are very high in how we construct the future of the Internet. Do we want a winner-takes-all scenario for whichever company ultimately creates that particular piece of intellectual property that maximizes interoperability (across systems) or do we want to create a commons?"

Microsoft, in trying shrug off its monopolistic and "closed" image, has taken a small step in opening up its vast share pool of source code. It has "opened" some of its software code on select products under a Shared Source License. But the License clearly states that users can experiment for non-commercial purposes only. This means if someone figures out a way of making their software better by modifying the source code, he cannot distribute it. But well, perhaps something is better than nothing!

1.3 Some Questions The World Is Asking

Will cost drive companies to go the open source and free software way? If they do, will they have to sacrifice on functionality or assured tech support?

Is the might of commercial software giants too much for open source to stand up to? Is the collective knowledge pool a formidable threat to conventional software makers?

Just how compatible will an open source system be with existing systems?

How long before open source (especially Linux) manages to shake off the hacker-geek kid image amongst corporates?

Is open source an idea that sounds all nice but may not stand firm in the gale of commercialisation blowing in?

All this, time alone will tell. But we can't help feeling that for the sake of innovation, for the sake of technology sharing (which some say is what a civilised world should adopt), and for the sake of cost cutting, open source must live.

1.4 What Is Linux?

You'd probably say, "Linux is an operating system." Yes, but remember, however, that the strictest definition of Linux is only the *kernel*. The more relaxed definition would be an overall package called a *distribution* that is ready to install and use. There are well over 300 distributions of Linux, most of them containing commonly-needed applications—and even games!

Linux was originally meant to be a UNIX clone. Here, by "clone," we mean that it would look and behave *like* UNIX. But bear in mind that Linux does not contain a single line of UNIX code! The source code of the two is entirely different.

1.4.1 Some asides...

Did you know that most servers today run on Linux? That means there's a good chance there's Linux somewhere behind all those Web pages you see.

Linux is not just an OS you use at home *instead* of Windows; a whole range of enterprise suites are now based on the Linux platform. From network servers to Web servers, several places have Linux as the backbone.

There are countless brains at work who pursue the technology—not for money, but out of sheer interest and passion. Linux boasts of some of the best online peer support today.

So how would you say "Linux"? As is usually the case with names, Linux too is not pronounced in any one particular way. While some say it as 'Li-nix' - Li as in leave and nix as in UNIX. Others mostly call it... LIH-nucks with the first syllable stressed. The "li" is said as in "limit" or "linger."

For those of you wondering who came up with the penguin called Tux, the Linux mascot, it was someone called Larry Ewing.

Why a penguin, you might wonder. It was simply because Torvalds loves penguins! There's another reason for Torvalds choosing the penguin as the mascot for Linux. In his words,

"Having a penguin as a logo also gives more freedom to people wanting to use linux-related material: instead of being firmly fixed with a specific logo (the triangle, or just "Linux 2.0" or some other abstract thing), using something like a penguin gives people the chance to make modifications that are still recognizable."

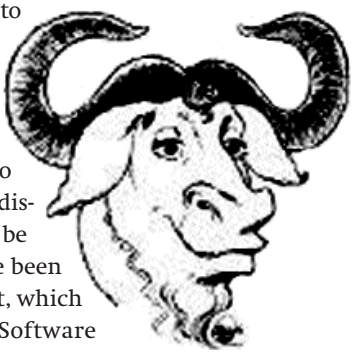
So you can have a real live penguin on a CD cover, for example, and people will get the association. Or you can have a penguin that does something specific (a Penguin writing on wordperfect for the WP Linux CD, whatever—you get the idea).

Compare that to a more abstract logo (like the Windows logo—it's not a bad logo in itself). You can't really do anything with a logo like that. It just "is".

A contest was announced to select the *best* penguin, and as you've probably guessed, Ewing's penguin was the winner.

1.5 The GPL (General Public License)

If you have installed any Microsoft OS or application, you would have seen the inevitable EULA (End User License Agreement). This is simply a set of rules you agree to (by Pressing *[F8]*!) before you install and use the software. The GPL is a set of terms and conditions that distributors of open source software have to adhere to in order to be GPL-certified. (Any distributor worth his salt would be GPL certified.) The GPL rules have been formed as part of the GNU project, which is an initiative of the Free Software Foundation.



The GPL has codified the essence of open source ideals. It ensures that anyone can modify a GPL-certified program at will and use it. If the modified program is made available for public distribution, then the modified code has to be given to the program user. This is one of the core elements of open source.

The following excerpts from the GPL should help put things in perspective:

The licenses for most software are designed to take away your freedom to share and change it. By contrast, the GNU General Public License is intended to guarantee your freedom to share and change free software—to make sure the software is free for all its users. This General Public License applies to most of the Free Software Foundation's software and to any other

program whose authors commit to using it. (Some other Free Software Foundation software is covered by the GNU Lesser General Public License instead.) You can apply it to your programs, too.

When we speak of free software, we are referring to freedom, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for this service if you wish), that you receive source code or can get it if you want it, that you can change the software or use pieces of it in new free programs; and that you know you can do these things.

To protect your rights, we need to make restrictions that forbid anyone to deny you these rights or to ask you to surrender the rights. These restrictions translate to certain responsibilities for you if you distribute copies of the software, or if you modify it.

For example, if you distribute copies of such a program, whether gratis or for a fee, you must give the recipients all the rights that you have. You must make sure that they, too, receive or can get the source code. And you must show them these terms so they know their rights.

We protect your rights with two steps: (1) copyright the software, and (2) offer you this license which gives you legal permission to copy, distribute and/or modify the software.~

The GPL is one of the most liberal, and compact rules in the computer world today. It is clear that the GPL is meant not for profit making, but for enabling wide distribution of software products that adhere to the agreement.

Some of the provisions of GPL are, however, under debate—especially the right to modify and distribute a program (with the source code) without being obligated to inform or acknowledge the original writer. While some people say that true open source means the right to modify and spread, others say this leads to confusion and can take the credit away from the original writer. But the basic philosophy seems to be "liberal is better than proprietary."

1.6 Distributions

This is where the beauty of open source lies. We found over 350 different distributions of Linux on www.linux.org! We can't say all of them are bug-free, but work is ongoing, and support is available for all major distributions. Just Google around for "Linux distributions," and you will find plenty of sites that offer downloads. Or just head to ***http://www.linux.org/dist/*** or ***www.distrowatch.com***.

Linux is available for everything—from home PCs to embedded systems, enterprise applications and Web servers to handhelds. Many distributions are hosted at ftp sites and come in the .iso (image file) format—all you need to do is download the file and burn a CD from it. Boot with the CD, and you are on your way to Linux on your desktop. It can't get any simpler!

In Linux land, more is merrier. But what is easy to install and use for a first-timer?

Of course, it is impossible to review each distribution and judge which of them is the "easiest." However, we think we're secure betting our money on some of the well known distributions—Red Hat, SUSE, Mandrake, Fedora, Gentoo and Ubuntu. All these work well for home use, and can do almost anything you can do on a Windows machine. There are some things however, such as gaming and listening to MP3s, which cannot be readily achieved on Linux. But there are work-arounds for this and we talk about it in the later chapters.

The size and functionality of the different versions varies: the distros (as distributions are often called) can be from 100 MB to 4 to 5 GB. You even have tiny distros that can be booted off a floppy! It's obvious that you have a size vs. feature trade-off. Generally, when you start the install process, you will be given an option to choose the applications you want. All the bigger distributions we've mentioned invariably have a reasonably good office suite, image editors, multimedia players, browsers (it is now mostly Mozilla Firefox, which, by the way, is an open source

project—yes, open source can exist on Windows too!), a system manager, and pretty much everything your home PC would need. Each distribution—at least the bigger ones—have their own professional support for enterprise users, besides "community" support by way of which enthusiast developers and peers lend support if you need any.

Now whatever your distribution, you'll most likely be running either the KDE (K Desktop Environment) or GNOME (GNU Network Object Modeling Environment) desktop environment. These are essentially GUIs that present the various commands available to the user. KDE was originally made for the UNIX operating system, while GNOME came as part of the GNU project.

One of the friendly (sometimes serious) debates is on how GNOME is pronounced—either "guh-nome" or "nome." Well, we aren't taking sides here, so go ahead and call it what you want! You can find more information on desktop environments in the Chapter 4, which deals with the X-Windows system.

1.7 Live CDs

Play it safe before you venture out

Installing an OS requires formatting all or part of your hard disk with a particular file system. The installation process is often time-consuming, even if it's easy. So, for someone who wants to just get a feel of an OS, Linux Live CDs come as a blessing.

Live CDs are like boot floppies. All you need to do is boot off a Live CD, and you will have Linux up and running! It's as simple as that—really! This is a great way to experiment and feel your way around the myriad of options in Linux before you sit down to install and run them from your hard disk. Earlier, Linux installation used to be a pain in the, er... brain, but thankfully, much of that has been sorted out. However if the thought of installation

and formatting partitions scares you, but you still want to try out Linux, a Live CD is the way to go. Some popular versions are DSL (Damn Small Linux), SUSE Live-Eval, and Knoppix.

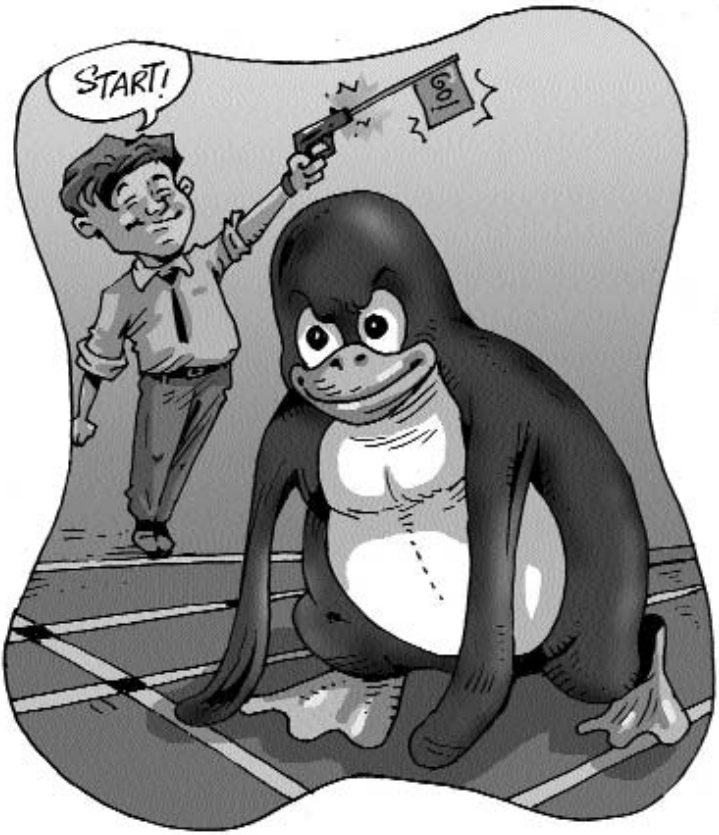
When you need to get back to Windows, just boot from the hard disk instead of the CD, and your familiar Windows comes right back. If you are impressed by the sample of Linux on the Live CD, you can venture out and download or buy a full-blown version and install it. Mind you, the Live CD cannot offer all the features of the full version, but it is still a great way to start.

Windows and Linux have always been thought of as the North Pole and the South Pole of the computing world. But guess what—they just shrunk the world! Attempts have been made to be able to run Windows programs on Linux. This, basically, is meant to enable easier migration from Windows to Linux, and cater to the principal reason many give for staying with Windows—"What about my applications?"

Linspire (earlier called Lindows until Microsoft sued them for the name) was the first to attempt this. Microsoft Office 97 and 2000 could be run from Linux via an application called Crossover Office. Later on, Xandros and SUSE, too, could open *some* Windows applications.

Armed with this background, you are now ready to proceed to take your first steps into the world of Linux. The installation procedure follows in the next chapter.

Getting GNU/Linux Running



An operating system usually has to be *installed* on a computer before it is used. In this chapter we will guide you through the installation steps in Linux. The actual steps may vary according to the distribution you are installing, but most of the concepts will remain the same.

2.1 Installing Linux

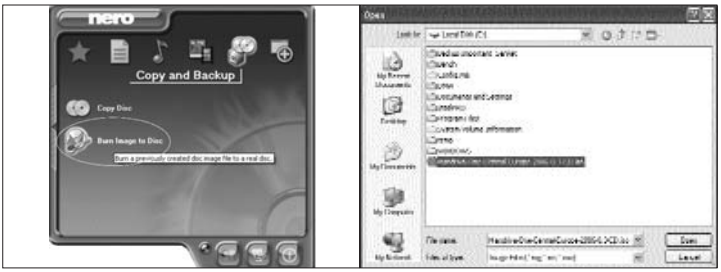
Morbid fear reigns amongst most folk on the issue of Linux installation procedure. The fact is, it's not rocket science (any more!). You simply need to follow the on-screen instructions and choose what you want. But given the vast customisation choices that Linux installations offer, sometimes even knowing what you want can be a problem.

Here we will to tell you something about a typical installation procedure. Of course, we maintain that the best way to learn is to experiment, relying on the trial and error method. Backed, undoubtedly by some logic and common sense—you don't want to be risking your precious music files or your memorable photos in case you choose a wrong format option that will wipe your disk off, do you?

Understanding the following will be easier if you've done a Windows install. In fact, the basic processes involved in setting up the two OSes are similar. And the risk of losing your data during a Linux installation is more or less the same as during a Windows install. If someone told you Linux is this untamable, fickle-minded beast that throws tantrums—well, disregard such 'expert' opinions. In fact, we recently installed a cool distribution of Linux called Ubuntu—the installation went off without a hitch—and this chapter is right now being written in OpenOffice on Linux! We don't see such an installation being a hassle for you either—so, folks, breathe easy!

2.1.1 Acquiring the OS

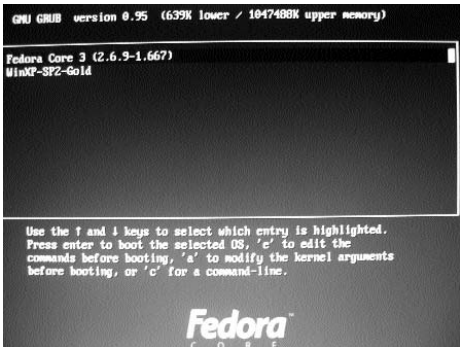
As we have said earlier, for some versions, you pay the shipping and CD costs, while some come totally free (yes—Ubuntu for example!). There are several versions that can simply be downloaded from the Internet and burnt onto a CD that can then be used to boot and install the OS. The download files are generally in the .iso format, which is nothing but an image of a CD. Use a CD burning software such as Nero, which supports burning .iso files to CD.



When you open the CD burning software, look for an option that says "Burn Image to Disc" or "Copy Image." You will be prompted to choose the image file from which the CD has to be burnt. Just point the software to the folder where you downloaded the .iso file. Make sure there's a scratch-free blank CD in the drive. It is also wise to click the option that says "verify written data," as this will be an OS install CD. Burn the CD.

2.1.2 Freeing up space for Linux

Linux can co-exist with Windows on your hard disk. It will operate on its own partition while Windows operates on another partition. When your computer boots up, you will be given an option to choose between Windows and Linux. This is called dual-booting or multi-booting.



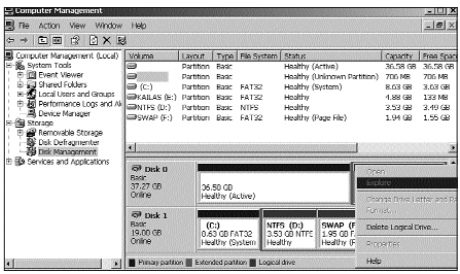
The boot loader. You have the choice to select the operating system of your choice after booting.

If you are new to Linux, it is better to keep your existing Windows intact and install Linux such that you have a dual boot. You need to allocate space for Linux to install. Linux usually requires two partitions, one for the system and user files and the other for the swap file—the idea of the swap file is the same as that of virtual memory

in Windows. So, while you are in Windows, free up two partitions by transferring all your data to other partitions. For the main partition (called *slash*, represented by the *"/"* symbol—this is explained in detail later on), you will need about 2 to 6 GB (depending on the packages you choose) if you are installing the newest version of Fedora, Mandriva or SUSE. Also, have in place about 500 MB for the swap file. If you don't have a second partition or can't free up a 500 MB partition, you can create it from the main partition you just freed up. Use a partition management tool such as Partition Magic if you are familiar with it, or alternatively, use the Windows Disk Management tool.

2.1.2.1 The Disk Management tool in Windows XP

Go to ***My Computer > Manage***, and click on Disk Management under Storage. Now select the partition you freed up, and right-click on it. Choose 'Delete Logical Partition'. Right-click on what now appears as Free or Unallocated Space, and create a 500 MB partition on it, choosing any filesystem.



The Windows Disk Management Tool

You now have one partition for the Linux installation, and another small one for the swap file. Make a note of the approximate free space, so you can identify this partition later on. With this done, you can start with your Linux installation.

Note that the process of allocating disk space can also be done during the installation, but the above method is a lot easier and straight forward for inexperienced users.

2.1.3 Booting from the CD

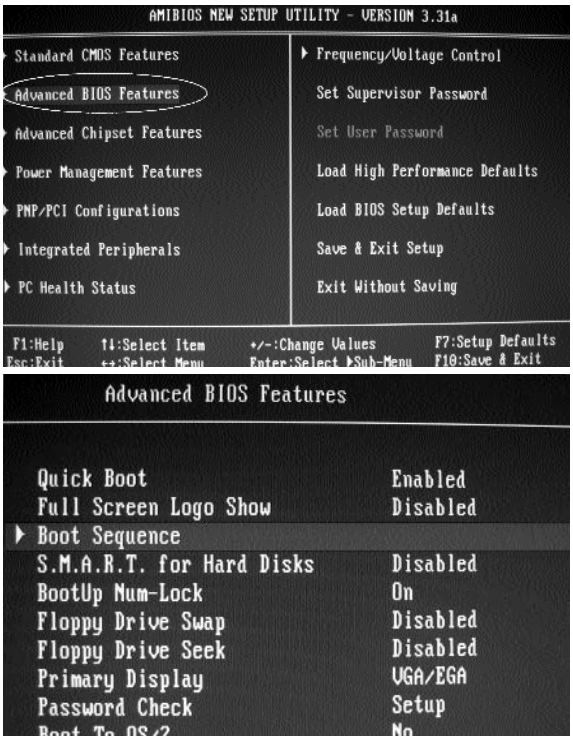
The next thing to do is restart your computer and change the first boot device to CD-ROM. To do this you need to enter the BIOS

setup by pressing *[Del]* (or *[F2]* on some machines) as the computer starts up.

In the BIOS, go to Advanced BIOS Settings, and navigate using the arrow keys to select the CD-ROM as your first boot device.

2.1.4 Disable the boot sector change warning

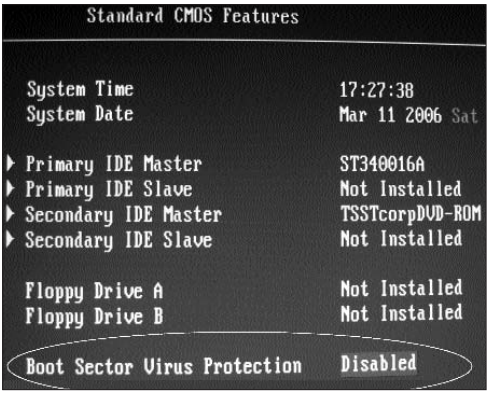
The boot sector is where information on where to find the OS is stored. When any OS is installed, an entry is made in the boot sector so that the computer knows where to find the OS. Many old



Press Enter here to see a list of available boot devices.

viruses would change the boot sector causing the OS to not start at all. To prevent this, BIOSes have an option by which no program is allowed to write to the boot sector. If such an attempt is made, the BIOS either flashes a warning message or disallows the modification outright. Therefore, while installing any OS, this option has to be disabled.

Look for this option in the Standard CMOS (or BIOS) Settings or Advanced CMOS (or BIOS) Settings and disable it. Save this setting. Remember, don't change anything else in your BIOS !

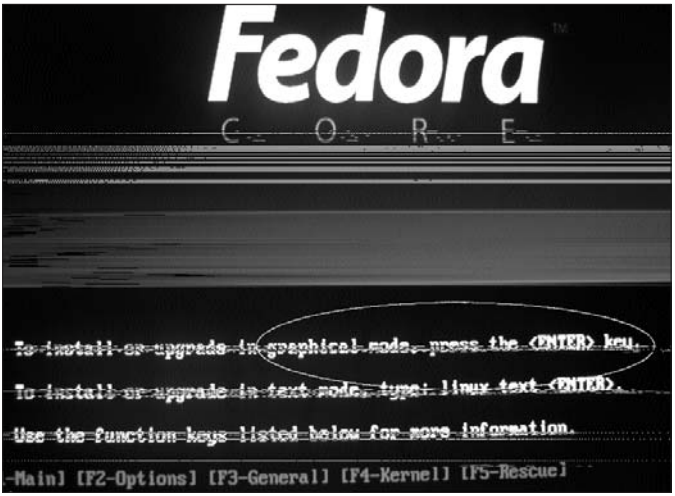


Check that this setting is Disabled

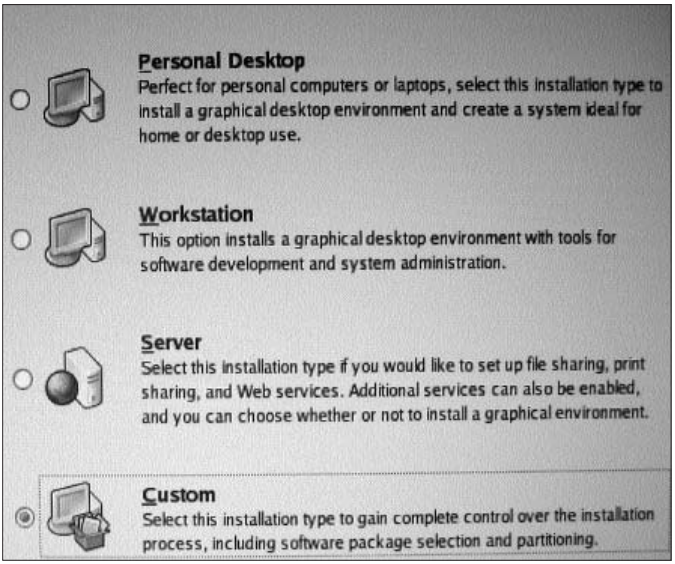
Most new distributions have an installer (Anaconda, mostly) that provide you with an easy interface during installation. You can use the mouse during installation to choose between the different options. But there are some distributions such as Ubuntu which only offer the keyboard interface. But this is no reason to get alarmed! Using *[Tab]*, the arrow keys and *[Enter]*, you can accomplish everything with ease. In fact, the number of times you need to interact with the computer during the installation is comparable to a Windows install.

2.1.5 Installation

Once you boot from the CD, you will see a welcome message and a logo of the version, which varies according to the distribution. The installer runs some pre-install checks to ensure compatibility—this takes just a few minutes. Once the preliminary checks are done with, you will typically be asked to choose from the available free



If given a choice, always choose the more intutive Graphical Mode. In the following steps choose Custom Install if you have another OS.



space in which to install Linux. This is where you have to be careful: choosing the wrong partition could mean your existing data being wiped out, or your Windows installation not working.

Recall that we'd asked you to keep some space reserved for Linux. You will now proceed to install Linux in that free space / partition. Read the on-screen instructions *carefully*—in fact, very carefully.

Important: When prompted to choose the kind of install, choose "custom" and choose to "manually edit" the partitions.

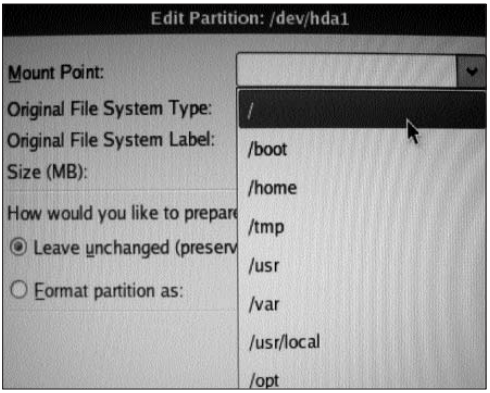
If during the course of the setup, you will find the option to install a server, just say no! If you install Linux as a server, you will not have a GUI, meaning no mouse, no graphics. This is because a server needs far more resources than a client or stand-alone system. Hence, the overhead of running a GUI is done away with in the case of a server.

Also, when server mode is chosen for installation, in most cases, the entire hard disk is wiped clean—not something you would want if you're looking at multi-booting.

The next step would be to choose where Linux gets installed. When you choose a Manual Edit of the partition table, you will see the various partitions and their sizes. The partition type ***"vfat"*** refers to the Windows partition. Here, you will need to specify which partition Linux has to install to. But before that, you need to format the freed up space (which should appear as 'Unallocated' or 'Free Space') as an ***"ext2"*** or ***"ext3"*** type of partition. Click on Edit (or an equivalent command, just follow the on screen instructions, and format it as mentioned above. Next, make it a mount point for ***"/"***.

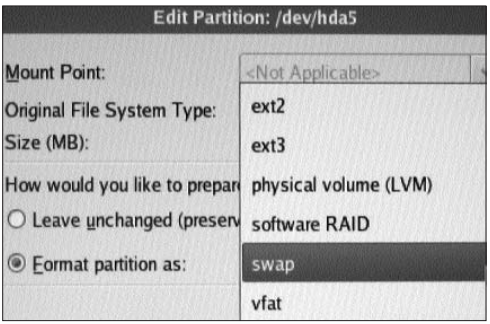
Recall now that we'd asked you to make another partition of 500 MB. This is to set Linux to use it as a swap partition. For this you will have to format it as a swap. In the partition manager / editor, choose to make it a ***swap*** type. In the mounting options, choose ***swap*** again if available. If the ***swap*** option is not present, Linux will

use it automatically. Check that the size of the partition is the same as what you created, to ensure you aren't accidentally wiping out a wrong partition.



Selecting partition for Linux to install

If you don't have a partition to be used as the swap, Linux will still make space for it within "/". But this compromises on performance, so we highly recommend you set up a swap partition.



Format a small partition (about 500MB) as Swap

In the process of setting up the partitions, just ensure you don't modify your Windows partitions, typically identified as *"vfat"* or *"NTFS"*, and your data will be fine. Again, remember, do not choose a predefined installation configuration. Manually select what partition you want to assign to Linux.

Your hard disk should at this point be ready for the Linux install. The next steps ask you for some settings, such as your default language, keyboard settings, time zone, user name with password, etc. If you are on a LAN, you will also need to provide your LAN settings—IP address, subnet mask, host address and DNS server address. These setting are the same as in Windows—you can simply write them down on a piece of paper and enter them here.

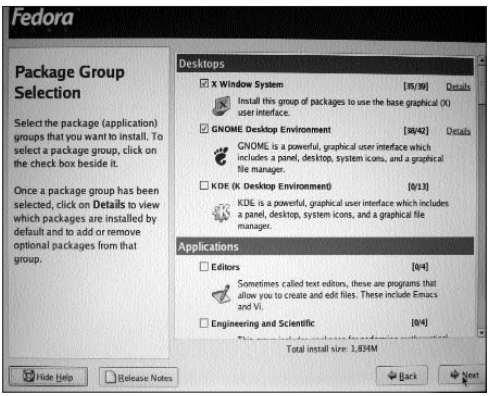
2.2 Post-installation procedures

2.2.1 Setting up the root

The root is like the Administrator in Windows 2000 or XP, except that the root user (or "account" as it is called) can do a lot more in Linux than what an Admin can do in Windows. You will be prompted to enter a root password and confirm it. Make sure you don't forget it! You will require the password to make important changes to the system later. You can add other users much like in Windows.

2.2.2 Choosing Applications

A common dig at Linux goes something like "Nah, no Linux for me... there aren't many applications available..." Those who say this need only look at the number of applications that can be installed along with Linux! All kinds of applications that you'd want in a computer come bundled with the distributions. Office suites (the equivalent of MS Office), image editors, and multimedia viewers are all bundled with all the distributions, including SUSE, Mandrake, Fedora, and Red Hat.



Select the applications you want installed

During installation, you can choose the applications you want, in order to save disk space. Often there are so many applications, you wouldn't even know they were there until you need to use them. We suggest you choose, at least, the available office application.

After you have selected the applications, you can grab some popcorn and sit in front of the screen looking at what happens. You may not understand all or some of it, but then, for a first-timer, it's an experience!

Depending on the speed of your computer and the applications chosen, the install time can vary from 30 minutes to up to three hours. After Linux and the applications are installed, you will be prompted to take the CD (or DVD) out of the drive and reboot.

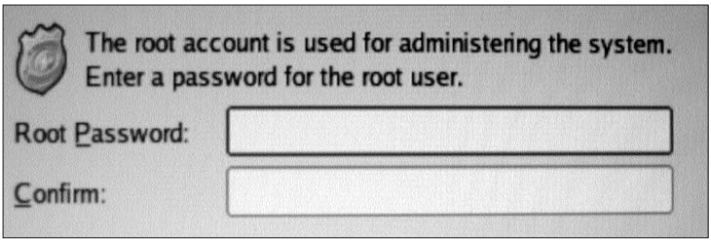
You will see a screen that allows you to choose from Windows or Linux (this is called the Boot Loader). Select Linux and voila—you have Linux up and running (unless there are some tasks that happen post-reboot, in which case your system will be ready for use by the next reboot). As Linux boots, you'll see a lot of text scrolling by as each service is started, and an "OK" or "Failed" message next to each line. In a few seconds' time you will be prompted for your user ID and password. We suggest *not* logging in as root when you are new to Linux. Log in as a user, so that you will not be allowed to perform critical tasks. Now is the time to just explore and have fun.

2.3 The Root User

Welcome aboard Linux !

First things first—we told you about the root user. So who or what exactly is the "root"?

Linux provides virtually unlimited customisation options which in Windows could be called "system hacking." Many of the changes the user can make can even result in boot failures! This is especially so because the command line (console) mode contains a large set of commands with even larger set of sub-options. A change in a command option can be the difference between the intended action and a disastrous happening. Therefore, core system modification rights are given only to the root user, who is expected to be well-versed with all he is doing.



During install you will need to supply a root password

If you are familiar with the Windows Administrator idea, you will know that tasks such as the formatting of disks or burning of CDs are accessible only to the Administrator. The root is something like that in Linux, but a lot more powerful.

Amongst others, one privilege of the root user is the ability to grant or revoke permissions to other users. A user can be given limited root privileges by the root user, for example, in a network, where the root user is on leave and an assistant has to take care of basic maintenance.

The root concept is something native to UNIX-like systems, including Linux. The term "root" springs from the file structure of UNIX, in which all directories branch off from a single directory called "root."

In Linux itself, the root user or admin must not be confused with the root directory, denoted by a forward slash (/), which is the directory that contains all other directories (system and user directories). There is also a ***/root*** directory under the (/). This is the home directory of the root user. The home directories for other users are under a directory called ***/home***.

The idea of not allowing all users to make system modifications is a derivative of the days of legacy networks, where users would be connected to a mainframe though a terminal. (The concept of PC did not exist in the early days of UNIX.) Thus, each user would need to have access to his own files, but not be able to change system set-

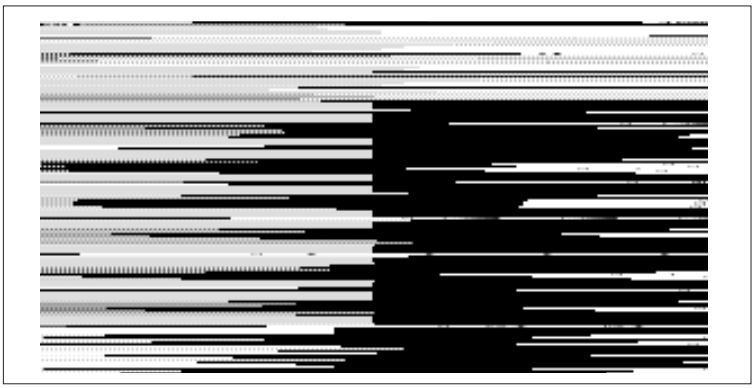
tings, which could potentially bring the whole network down. By contrast, Windows was intended for a PC, and any changes made on the PC would mostly affect the individual user rather than the network (unless it's run as a server, of course). That's why there's no strict user hierarchy, as compared to Linux.

A security feature of UNIX and hence Linux is that even programs run by a user cannot make critical changes to the system. This is one of the reasons for the fewer number of viruses affecting Linux machines. Only when a program is run from the root account does it have the permission to make any system changes. Thus, the responsibility for system security rests with the root user all the more. Thankfully, Unix and Linux system administrators are a smart lot!

2.4 The Linux Directory Structure

All operating systems have a directory structure in which system files are stored. User files will also be stored in a directory created by the OS itself unless otherwise specified. Of course, the user can create his own directories, but those do not constitute the system's directory structure. The following directories are commonly found in Linux:

The basic directory is the "/" directory—simply called the slash. This contains other directories used for various system purposes.



The directories created by Fedora during install

/bin

This is the ***binary*** directory, which contains the executable files. When a program is installed, its binary files are stored here.

/dev

dev stands for "devices." ***/dev*** contains files that point to the hardware that make up the system. All peripherals are represented as files, and when the service of a device is called for, the file is looked into for the device description and parameters.

/boot

The boot loader files are stored here. Sometimes, an image of the kernel is also maintained in this directory.

/etc

This folder contains various configuration files that can be manually edited to effect system changes. This is a particular feature of Linux by which, knowing what to alter in what files, the user can make almost any change to the system. The files are stored in the .txt format, and can be viewed and modified in any editor.

A crude analogy with Windows is the set of .ini files, and of course, the Registry.

/home

This is the directory that contains the user's files. Unless otherwise specified, work files will be saved in a user directory that is created for each user in this ***/home*** directory. If a user is created by the name of ***"digit"***, then his files will get stored in the ***"/home/digit"*** folder.

/root

This is the home directory of the root user.

/lib

"Library" files—frequently-used system files needed by programs or the operating system—are stored here, just like the library files stored in "***C:\windows\system"*** in Windows operating systems.

/mnt

All storage media other than Linux partitions are "mounted" here. (The concept of mounting is explained in the next sections). Each device has a folder created for it, for example, the CD-ROM and the floppy drive. Windows drives/partitions too are mounted here. (A FAT file system is seen as a Windows drive). To access these devices, you need to get to the /mnt directory and open the appropriate device there. There can be a shortcut to the devices on your desktop or elsewhere, depending on the Linux you have, but then these shortcuts will access the devices mounted in this folder.

/opt

The "options" directory stores add-on components such as desktop environments, databases (such as Oracle), etc. A crude analogy would be the Program Files in Windows.

/tmp

This one is simple, isn't it? As the name suggests, this is the temporary directory that the system uses for various purposes that we, as end users, need not be too worried about.

/usr

Non-critical system files are stored in this directory, which contains a copy of most of the directories in the root—you'll find here a "bin" directory containing programs, a "lib" directory containing libraries, etc. Generally, important or core Linux files are contained in the root directories, while others are put in the ***/usr*** subdirectories.

/var

The *various* folder contains just that—various files such as log files, spools, etc.

2.5 Basic Operations In Linux

All tasks can be accomplished in Linux by just using commands. A true Linux geek does not need a GUI at all! Also, the UNIX-based commands are available in all Linux distributions. But the placement of the command in the GUI menu can vary with version and distribution. So its always helpful to know certain basic commands. All you need to do is bring up the terminal (or Console, as it is called) and type in your command: *voila!,* it is executed: no need to search through menus and submenus! But this will change when you use another flavour (distributions are sometimes fondly called "flavours").

To bring up the terminal (where you can manually type in commands), try right- clicking in a blank area on the desktop and look for an option called "terminal window" or "console." Alternatively, look under the menus for something that says "term," short for "terminal." It could be kterm, xterm or just term.

Tip

In Gnome desktop, you can press [Alt] + [F2] to bring up a terminal window. In KDE, K\System\Konsole will take you to the terminal window.

2.5.1 Mounting partitions

By now you have a dual-boot system, and your Windows partitions are still alive. You will have images, documents or videos that you might want to access from Linux. However, when you try to look for the Windows partition, you will not find it! Whatever happened to it, you might ask. Well, you must first "mount" those partitions onto the Linux filesystem. In the computer world, mounting means making accessible to the system a filesystem (or any data). You might ask, while Windows automatically shows you all partitions, why doesn't Linux?

Basically, the way Windows and Linux deal with files and partitions is different. Windows recognises and clearly demarcates partitions at boot. Linux works differently: unless you mount a partition or a device, the system does not know of the existence of that parti-

tion or device. You cannot make files of one partition appear as part of another. This layout, called a *unified filesystem*, allows files from one drive to appear as part of another. Hence you need to ***mount*** the partition you wish to access, to a mount point. A mount point is the directory to which the partition is attached. After mounting, the contents of the partition appear in the mount point.

What is the use of this method of working?

Let's say that in your Windows "D:" partition you have some songs, and in partition "E:", you've copied songs from a friend. You want to know what songs are common. You can simply mount both these partitions to a single mount point in Linux and view them under it! Arrange files according to name, and your task is done. This is a trivial example, but the method can be of great use at the network administration level. Scripts that operate on files in a directory can be used on files of different partitions by simply mounting them to a common mount point.

The following codes represent hard disks connected in different configurations:

Primary Master—First IDE hard disk—hda

Primary Slave—Second IDE hard disk—hdb

Secondary Master—Third IDE hard disk—hdc

Secondary Slave—Fourth IDE hard disk—hdd

Further, the first partition on a disk is represented by 0, the second partition by 1 and so on.

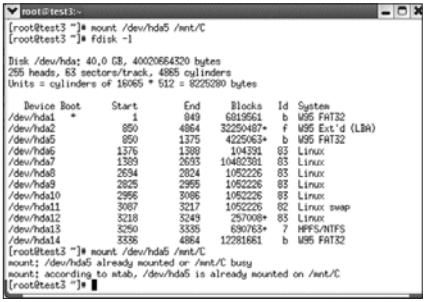
Hence, hdb2 refers to the third partition on the second IDE device (primary slave).

If you don't know which partition you wish to mount, you can get the partition information of all connected hard disks via the ***"fdisk -l"*** command and then proceed to mount.

The mount command is used to mount the partitions into Linux. The general format of the command is:

mount /dev/hd xx mount point

Here, "***xx***" stands for the characters that must be substituted, according to the hard disk connection. "Mount point" refers to the directory where you want the partition to be mounted. Linux has a separate directory for this purpose, called ***mnt***.



The contents of partition identified as hda5 will be available under /mnt/C after execution of the mount command. Once a directory is used to mount, it cannot be used as mount point of another partition.

If you want to mount the second partition (as seen by ***fdisk -l***), your command is:

mount/dev/hda2/mnt/dir1

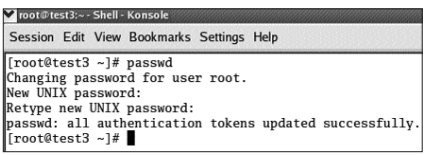
This is assuming there is a directory by the name ***dir1*** in the ***/mnt*** directory. Else you can create one using the ***mkdir*** command.

2.5.2 Changing the password

In the terminal window (obtained as explained earlier), to change your password, just type in ***"passwd newpassword"*** where ***"newpassword"*** is the password you want to use. You will then need to enter your old password.

If you are logged in as the root, type ***"passwd <username>"***, where ***"username"*** is the user name for which you wish the change the password. Upon hitting ***[Enter]***, you will be prompted for a new password. You will not be prompted to enter the old password, nor will it be echoed (displayed) to you.

To change the passwords in the GUI, you will need to do some trial and



Changing passwords in command prompt.

error and find out where this command is placed. This of course, varies from flavour to flavour, and with the desktop environment (KDE / Gnome). So, have fun exploring the GUI. But the fastest way to do it is as above.

2.5.3 Even more commands

In any OS, a command is an instruction given to the computer to do a particular task, such as moving files, renaming files, or opening a program (which itself will contain a lot of other files). In fact, everything you do boils down to commands. When you click on an icon, a command is called to open the file or application that appears as the icon. Every OS has its own set of commands. Some have only commands and not a graphical interface, such as DOS, while Windows allows both command-line and graphical interfaces Linux is available in both GUI and command-line-only versions. But when Linux is used as a server, the GUI is not installed, so as to optimise resources for performance.

We've mentioned how to bring up the terminal. Once your terminal is opened you can start experimenting with the various commands. Before that, try and get some of the basics of Linux commands so you won't get lost along your way.

One of the first things you would want to know is how to get out of the terminal. Remember the DOS ***"exit"*** command? Well, the same thing works here. Type ***"exit"*** and press ***[Enter]*** to close the terminal and return to your desktop.

If you are doing a task that will take a long time to execute—like copying a large video file—you need not wait until the process is over in order to try another command. (It makes for a funny sight when someone just sits staring at the screen doing nothing, waiting for something to happen!) After any command, just add the "&" character (for example, ***"cp x y &"***) and that process will get executed in the background while the terminal will be available for you to do something else.

Be very cautious of using unfamiliar commands when logged in as root. But go ahead and have your fun trying out new things when logged in as a user.

If you know the consequences of a command that needs root privileges, just use the ***"su -"*** command to temporarily log in as root. When you type in ***"su -"*** and press ***[Enter]***, you will be prompted for the root password. Entering your root password will allow you to execute the command. Avoid starting your session as root, though.

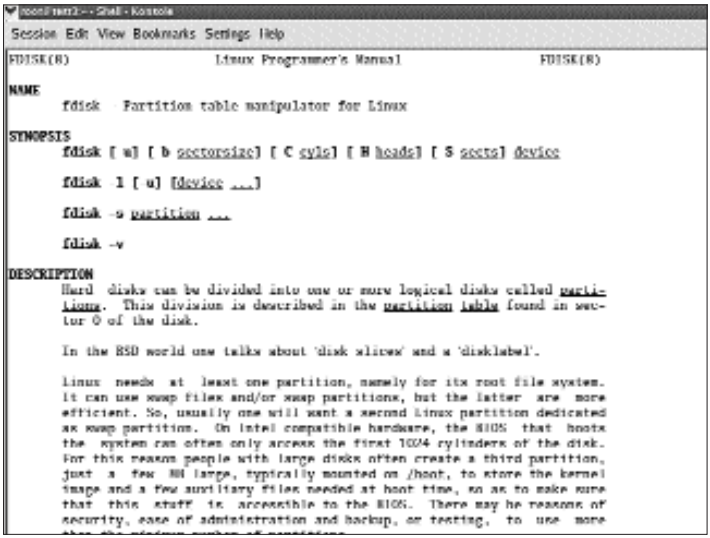
You can easily distinguish if you are operating as a user or root. A user prompt will contain the ***$*** sign, while the root will have the ***#*** sign in the prompt.

Remember that all UNIX commands will work on all distributions of Linux. There are, however, certain commands created by particular distributors which will work only on their distribution. So, keep your user manual ready if you have one.

If at any point you need to know what a command does, all you need to do is type in ***"man <command>"*** and press ***[Enter]***. Sorry ladies, this does not command your man to do the dishes—it is short for ***"Manual"***! Doing this gives you an explanation of the command specified after ***"man"***. For example, typing in ***"man fdisk"*** will display the manual information for the fdisk command.

Most commands contain options that you can use by placing a dash (-) followed by the appropriate letter. Combinations of these options can also be used. One only needs to remember what letter does what. You might find it difficult initially, but as they say, practice makes you perfect!

In the prompt, you can press the up and down arrow keys to get to previously typed commands instead of typing them again.



man is used to get information on any other command. This screen is the result of the command man fdisk

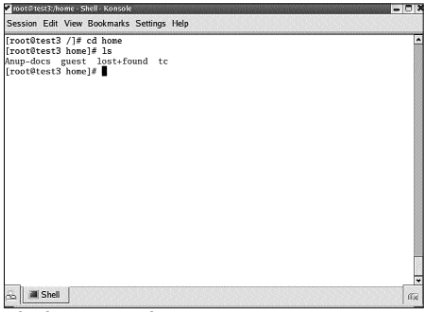
It's now time for some basic commands, just so you get a feel of the power that Linux vests you with.

2.5.4 Some Very Useful, Basic Commands

The first command you learn in any OS is to display files in a directory: in DOS, you would type ***"dir"*** to display the contents of a folder. In Linux, the equivalent is the ***"ls"*** command. This command displays the files and folders in the current directory. The ***"ls"*** command comes with some extra options; ***"ls -l"*** produces a "long list" that shows the owner, group, size, date modified and permissions. ***"ls -a"*** will list hidden files as well. In DOS, this is done by ***"dir /a"***. There are other options—which we will leave to you to explore.

Changing the current (working) folder is accomplished by ***"cd <directoryname>"***. The directory you want to change to is included after the ***cd*** and a blank space. When you want to return to the home directory, you simply need to type in ***cd*** and press ***[Enter]***. To go one directory up, just type ***"cd .."*** and hit ***[Enter]***.

"rmdir <directory-name>" will delete the directory specified.



The ls Command

These operations can be done through the GUI as well, but it pays to know these command-line alternatives! Now here is a particularly useful command not available in all GUIs: say you have a directory with some 500 songs whose listing you want to produce. All you need to do is use an output redirect command such that the result of the *"ls"* is given to a file.

ls > filename.txt

This will create a file by the name specified after the > sign in the current directory. This file will contain the list of the directories and files in the current directory.

As you try out different commands, your screen will get more and more cluttered. Want a neat look? Just type in *"clear"* and press *[Enter]*.

The command *"date"* is used to display and change the date and time.

To know about the free disk space available, type in *"df"*, and to get the disk usage (occupied) information, the command is *"du"*.

For copying files, the command used is *"cp"*. The format is the same as the copy command in DOS:

cp <source_location/file> target_location

example: *cp /songs1/spb.mp3 /songs2* will copy the file *spb.mp3* from the */songs1* folder to the */songs2* folder.

The *cp* command retains the original file and places a copy in the target location. If you wish to do a cut-paste equivalent, then replace "*cp*" with "*mv*". This will delete the file from the original location after making a copy at the target location.

2.5.3.1 Aliases

Aliases are user-created, short commands that can replace longer standard commands. If you frequently need to use a long command, you can create your own alias and use it:

alias aliasname="long command"

For example, if you type in
alias lh="ls -l -a $HOME"

you can use *"lh"* in place of the longer command.

There are several more commands that you can use in your day-to-day work on Linux. While most of your work gets done by using the GUI (text editing, Internet browsing, multimedia, etc.), you can claim to know something about Linux after playing around with some commands!

2.6 Shells and Virtual Consoles

2.6.1 Shells

A shell is a command interpreter that reads the command you type and, according to the command, performs the requested operation. A command that you type must be recognised by the command interpreter. The most commonly used shell in Linux is the "bash" shell. It is available in all versions of Linux, and is an offshoot from Unix. Like most things in Linux, "bash," too, is a contraction, and is derived from "Bourne again shell." The bash shell can distinguish between alphabet case types. Hence ***"LS"*** is not the same as ***"ls"***.

Commands that are not part of a shell are called "external commands," and relate to programs. Shell scripts can also be invoked by typing a command, which can be external—a shell script is nothing but a set of commands stored in a "script," or a file consisting of several commands that can be executed in a single go.

As a new user, you may not see an immediate need for shell programming. But at the level of system administrators or database administrators, shell programming is very useful. There are several complex commands that will be difficult to type in every time they are needed. A script can also be run at regular intervals to monitor system performance or network traffic. Another area where shell scripts can be of use is in backing up data. If data is stored in different locations, manually copying it to a backup location can be a tedious task—which can be simplified using a script. No wonder you see such a demand for shell programmers!

Apart from bash, other command interpreters include the C shell (csh), Korn shell (ksh), Bourne shell (sh), and more.

A command interpreters is a program that reads (rather, interprets) text commands typed by the user (or also from a file if it is a script file) and executes them.

2.6.2 Virtual Consoles

Strictly speaking, a console is nothing but an input and output device—a keyboard and a monitor. Simple as that. Remember we said UNIX was made to be a multi-user system where several users were connected to a single mainframe? Each user was only given a console connected to the mainframe. But with the advent of PCs, the need for consoles has diminished (actually, eliminated, except in legacy systems) and Linux only gives a feel of a console by way of a Virtual Console. It's called so because on a single system, you are allowed to log in as different users, thus simulating a console environment.

This can be very useful because you can implement tasks corresponding to different users simultaneously. Generally four to seven virtual consoles are provided, extendable up to 12. You just need to press *[Alt] + [F2]*, and you will be prompted to enter a second user name and password (you are already logged in as the first user, mind you!). You then enter the second console. Press *[Alt] + [F3]* and you get yet another, similar window for the third console. Thus you can configure to have up to 12 virtual consoles (12 is as many function keys as you have!).

Shifting between virtual consoles can be done even by using the directional arrows with *[Alt]* pressed.

To move to the next console, press *[Alt] + [>]*
To move to the previous console, press *[Alt] + [<]*

2.7 File Ownership And Permissions

We'll never tire of telling you that many of the advantages of Linux are a result of its springing from a strong multi-user operating system. In a multi-user system, it is important to ensure that the files of one user cannot be accessed by another user unless the latter is permitted to access those files. This gives the idea of file ownership. The user who creates a file is, by default, the owner of that file. He can do whatever he wishes to do to his file, such as editing and deleting. A user group can also own a file, if a member of the group has created the file. The user group will have the same permissions as that of the file creator. This is an important feature, because there can be instances where a group of users are working on a project and files created by one user have to be accessible and editable by others in the group.

Besides the owner of the file and his user group, other users can be granted privileges that will enable them to work on the files created by the owner.

There are three basic file permissions in Linux: Read Permission, Write Permission and Execute Permission. The read and write permissions behave slightly different on a file and on a directory. The read permission on a file enables any user who is granted this permission to open and view the contents of the file, but not make changes to it. On a directory, however, only the contents of the directory—the file listing—can be viewed. A file contained in the directory cannot be opened unless a read permission is given to that file.

On a file, the write permission allows to you write into the file, that is, make changes to the file. But on a directory, this permission enables you to add, remove or rename files in the directory. Hence, a write permission on a file will allow you to make changes to the file, but deleting the file or renaming it can be done only if the directory allows you to do that.

The execute permission, in case of a regular file, means you can execute the file as a program or a shell script. However, on a directory, the execute permission allows you to access files in the direc-

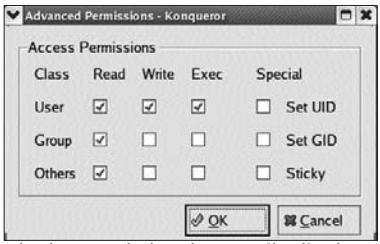
tory and enter it. Though the execute bit lets you enter the directory to add a file, you are not allowed to list its contents, unless you also have the read permissions to that directory.

With a combination of these permission, selective access can be granted to users such that their actions are strictly controlled.

Viewing File Permissions

You can right-click on a file or directory and choose ***Properties*** > ***Permissions*** tab.

There, you can see the permissions allowed to the Owner, User group, and Others for the file or directory. You can check or uncheck the read, write and execute options, and hence change the permissions. As with other things in Linux, it helps to be familiar with the command-line way of executing it too.



Viewing permissions in most distributions

In the command mode, just type in ***"ls -l"***, and you can see the permissions along with other details of the file. In the listing that follows, you will see a set of seemingly random letters that precede the file owner. From the first 10 characters that appear at the start of the line, it is possible to see the ownership and permissions of the file.

The first character indicates if it is a directory or file: ***"d"*** means directory, and ***"-"*** means file. The next three letters denote the permission that the owner has.

An ***"r"*** in the second character means read permission
A ***"w"*** in the third character means write permission
An ***"x"*** in the fourth character means execute permission
A ***"-"*** means the corresponding permission does not exist.

2.8 Changing Permissions At Command Prompt

This is accomplished by using the ***"chmod"*** command. The chmod is followed by a three-digit number used to indicate the permissions required. The first digit indicates the owner's permission, the second digit the user group's permission and the third is for all others.

The number coding is as follows:
7: full
6: read and write
5: read and execute
4: read only
3: write and execute
2: write only
1: execute only
0: none

Hence, the command ***"chmod 777 filename"*** will give full access to everybody to the file named ***"filename"***. ***"chmod 600 digit.doc"*** will give only the owner of the file ***"digit.doc"*** the permission to view and edit the file. No one else will have any rights to it.

2.9 Compressing Files And Archiving In Linux Terminal

There are strong compression utilities in Linux that can be run easily from the command line. Linux features an archiving utility called ***"tar,"*** in which you can add many files, and the tar file will be treated as a single file. You may wonder how this is different from a folder.

Linux features the zip and a gzip utilities to compress and extract (uncompress) files. There is also a bzip and bzip2, but we will talk about zip and gzip.

At the command prompt, use this format compress a file:
"gzip filename"

The file will be compressed and saved as ***filename.gz***
To extract a compressed file:

"gunzip filename.gz"

The file ***filename.gz*** will be deleted, and the extracted file, that is, ***"filename"***, is created in the current folder.

Similarly, to use the zip format so that the file can be opened even on a Windows machine, you can use the zip utility:

zip filename.zip filename (to compress)
unzip filename.zip (to extract)

2.9.1 Archiving at the command prompt

You will need to know the four basic options of the tar command generally used.

C: create a new archive
V: verbose; show the files being archived
X: extract files from an archive
F: when used with C, it means "use the filename specified for creation for the file," or when used with X, it means "use the filename specified for the extracted file."

You can add several files to the archive with the tar utility as follows:

```
tar -cvf filename.tar /dir1/dir2/dir3 /dir1/dir2/dir3
```

The files in dir1, dir2 (which is a subdirectory under dir 1) and dir3 (which is a subdirectory under dir2) will be added to the tar archive such that the directory structure is maintained in the form of dir1, dir2 (created under dir1) and dir3 (created under dir2) in the archive. You can of course, specify different folder names in the second set of directories.

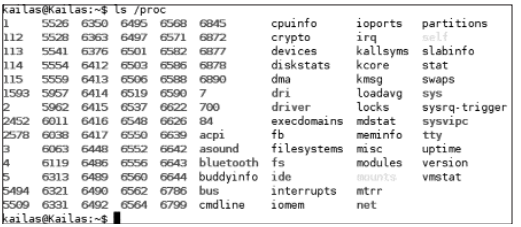
Most distributions give you a GUI interface to do this. Having explained the harder way to do this, we leave you to figure out how it is done with the GUI. Have fun!

Inside Linux

This chapter tells you about the directory structure, the proc file system, and managing users on Linux. Once you're through with this chapter, you will be a lot more familiar with Linux and on your way to making it work as you want it to.

3.1 The /proc File System

3.1.1 Virtually speaking



A typical /proc file system contents

Having come this far with Linux, have you wondered if there is a way in which an advanced user (you can be one too, with some practice and patience) will be able to take a peek at the kernel and understand what is happening at any given moment? Well, Linux gives a user more than enough information in the form of the "/proc" file system.

The /proc file system is a "virtual" file system that does not reside on any physical media. It is a file system that resides in the memory and contains real-time information about the devices that make up the PC, the processes that are running and the processes that are in a sleep state. Some of this information is readable and comprehensible, while some is not. The information needs to be interpreted, and to do this you have many utilities at your disposal.

Before we proceed, let's give you a small sample of the kind of things a user can easily access. The GUI tools that can give you system information will also read from the /proc. Let us say you want to view information about your CPU. Just get into the terminal (console) and type in "cat /proc/cpuinfo".

This will give you information about your CPU. But that's not all. You can also view running processes and what resources they are consuming, the kernel, etc. While a new user may be confused by the sheer amount of information that /proc offers, a developer or enthusiast will find a lot of useful information here. Most infor-

mation in the /proc is read-only, but there are some files that can be edited to effect immediate changes. Let's leave that to advanced users...

```
kailas@Kailas:~$ cat /proc/cpuinfo
processor       : 0
vendor_id       : GenuineIntel
cpu family      : 15
model           : 2
model name      : Intel(R) Pentium(R) 4 Family CPU 2.40GHz
stepping        : 4
cpu MHz         : 2806.696
cache size      : 512 KB
fdiv_bug        : no
hlt_bug         : no
f00f_bug        : no
coma_bug        : no
fpu             : yes
fpu_exception   : yes
cpuid level     : 2
wp              : yes
flags           : fpu vme de pse tsc msr pae mce cx8 apic sep mtrr pge mca cmov
pat pse36 clflush dts acpi mmx fxsr sse sse2 ss ht tm
bogomips        : 4751.36

kailas@Kailas:~$
```

CPU information gathered from /proc

3.1.2 More about /proc

Many files contain simple binary values—0 or 1. Generally, these files simply mean On or Off, and Enable or Disable. By simply changing the value from 0 to 1 or vice versa, the particular behaviour of a device can be altered! Something like the Windows registry? Yes, sort of, but a lot bigger. As this entire file system resides in memory, it is created every time Linux starts. Again, since it is virtual, it has a file size of 0 bytes.

```
kailas@Kailas:~$ cat /proc/meminfo
MemTotal:       248864 kB
MemFree:         17960 kB
Buffers:          1072 kB
Cached:          64052 kB
SwapCached:       3212 kB
Active:         145152 kB
Inactive:         8344 kB
HighTotal:           0 kB
HighFree:            0 kB
LowTotal:       248864 kB
LowFree:         17960 kB
SwapTotal:      722884 kB
SwapFree:       671364 kB
Dirty:             740 kB
Writeback:           0 kB
Mapped:         143488 kB
Slab:            13740 kB
CommitLimit:    847316 kB
Committed_AS:   354720 kB
PageTables:       1432 kB
VmallocTotal:   778232 kB
VmallocUsed:      4356 kB
VmallocChunk:   773748 kB
kailas@Kailas:~$
```

Memory usage statistics fetched from /proc

When you view the contents of the /proc directories (or the file system, more appropriately), you will see a lot of directories that have numerical names. These are directories containing information about the various processes that have been running at the instant we opened the /proc directory. Besides this, you will also find folders pertaining to the devices, chipsets, controllers, etc. For example, if you want to see the features of your CD-ROM drive, all you have to do is type in the following command to view the required details from the /proc: "cat /proc/sys/dev/cdrom/info".

The possibilities are endless in seeking information on your system with the /proc file system. What we have presented is only

a small preview of this tool. Along the lines of the examples we have presented, try to open other files as well. Remember, all of them won't work, but you may just end up knowing a little more about your system than you do now!

3.2 System Maintenance

Your computer, like any other machine or system, needs some maintenance, mostly preventive! You want to ensure that your system runs smoothly to keep Mr Murphy at bay—most system administrators feel that computers crash when it is most important that they run properly. In Windows, you would have used the disk defragmenter and scandisk tools; let's see what Linux offers in terms of system maintenance. As you've probably guessed, there are a lot of powerful options here as well.

With respect to operating systems, one of the most important maintenance tasks is updating it using patches released by the developer. No OS is completely secure. Hence the creators of OSes release patches in order to plug security holes, as and when they find any. Besides, the patches released may also contain certain bug fixes. Hence it is wise to keep your system up-to-date. If you are connected to the Net (we will describe configuring your computer to use the Internet later in this book), updating is easy. From within the user interface, go to the System (most flavours have this menu) and look for an option that says Update. If this doesn't work, look for a Maintenance menu and under it, look for an update option. While this takes care of inherent OS issues, there are a lot of other tasks you are advised to carry out to keep potential problems at bay.

Firstly, ensure that there is enough free space on all your partitions. If any of the partitions "overflow" (run out of space), the system is likely to show problems, even if there is lot more space available on other partitions. The command "df" shows

```
kailas@Kailas:~$ df
Filesystem           1K-blocks      Used Available Use% Mounted on
/dev/hda1             37752556   1707872  34126928   5% /
tmpfs                   124432         0    124432   0% /dev/shm
tmpfs                   124432     12588    111844  11% /lib/modules/2.6.12-9-38
6/volatile
kailas@Kailas:~$
```

The result of the df command

how much free space is available in each partition. You can delete the contents of the /tmp directory. Login as root if need be. Files that are in use will not get deleted, so don't worry about deleting anything important.

Another place you can look for dead files is the /var/tmp directory. Logs and spools will be stored here, and might not have been deleted during a sudden termination.

3.2.1 The file system and disk maintenance

As the name suggests, the file system is the method by which files are stored on storage devices (hard disk, CD-ROM, etc.). As all files are invariably stored on the media, it makes sense to ensure that there are no errors on your hard disk or in the file system itself. If your hard disk crashes, then recovery, if at all possible, will be a tough job. Sometimes hard disks give a warning before they die—unusual screeching noises or bad sectors. Here are some operations—commands that look for errors, or just change hard disk parameters:

Check For Bad Sectors (blocks): badblocks

The command "badblocks /dev/had" will search the first partition of the first IDE hard drive for badblocks. Similarly, to check for bad blocks in the second, third, fourth IDE hard drive, replace "hda" with hdb, hdc or hdd respectively.

3.2.1.1 Change partitions, format, view partition info

cfdisk is a partition table manipulator used to create or delete partitions. As you open cfdisk, you will see a bounty of options at the bottom of the screen. Browse through them, but do not make any changes unless you are absolutely sure of what you're doing! This

holds true for working with any disk management tool. You can format a drive in almost every file system that is existing today.

The cfdisk utility

fdisk is the most commonly-used disk partition viewing tool. If you want to see all your partitions, the quickest way to do so is to use the "fdisk -l" command.

3.2.1.2 View and change hard disk information

hdparm is used to get or set hard disk parameters. Some of the settings you may make here may not be compatible with the hard disk. If you see a warning that says the operation might be risky, it is best avoided.

3.2.1.3 Checking for disk errors

fsck is used to check and repair a Linux file system. It is advised that only unmouted partitions be checked by this tool.

3.2.1.4 Mounting hard disk partitions

As discussed earlier, the following codes represent hard disks connected in different configurations:

Primary Master - First IDE hard disk - hda
Primary Slave - Second IDE hard disk - hdb
Secondary Master - Third IDE hard disk - hdc
Secondary Slave - Fourth IDE hard disk - hdd

Further, the first partition on a disk is represented by 0, the second partition by 1 and so on. Hence, hdb2 refers to the third partition on the second IDE device (primary slave). If you don't know which partition you wish to mount, you can get the parition information of all connected hard disks via the "fdisk -l" command and then proceed to mount. The "mount" command is used

```
root@Kailas:/home/kailas# fdisk -l

Disk /dev/hda: 40.0 GB, 40020664320 bytes
255 heads, 63 sectors/track, 4865 cylinders
Units = cylinders of 16065 * 512 = 8225280 bytes

   Device Boot      Start         End      Blocks   Id  System
/dev/hda1   *           1        4775    38355156   83  Linux
/dev/hda2            4776        4865      722925    f  W95 Ext'd (LBA)
/dev/hda5            4776        4865      722893+  82  Linux swap / Solaris
root@Kailas:/home/kailas#
```

The result of the fdisk -l command

to mount a filesystem (or, for practical purposes, a partition).

m o u n t /dev/hda2 mnt/2 will mount hda2 partition on a directory named '2' in the "mnt" folder.

```
root@test3:~
[root@test3 ~]# mount /dev/hda5 /mnt/C
[root@test3 ~]# fdisk -l

Disk /dev/hda: 40.0 GB, 40020664320 bytes
255 heads, 63 sectors/track, 4865 cylinders
Units = cylinders of 16065 * 512 = 8225280 bytes

   Device Boot      Start         End      Blocks   Id  System
/dev/hda1   *           1         849     6819561    b  W95 FAT32
/dev/hda2             850        4864    32250487+   f  W95 Ext'd (LBA)
/dev/hda5             850        1375     4225063+   b  W95 FAT32
/dev/hda6            1376        1388      104391   83  Linux
/dev/hda7            1389        2693    10482381   83  Linux
/dev/hda8            2694        2824     1052226   83  Linux
/dev/hda9            2825        2955     1052226   83  Linux
/dev/hda10           2956        3086     1052226   83  Linux
/dev/hda11           3087        3217     1052226   82  Linux swap
/dev/hda12           3218        3249      257008+  83  Linux
/dev/hda13           3250        3335      690763+   7  HPFS/NTFS
/dev/hda14           3336        4864    12281661    b  W95 FAT32
[root@test3 ~]# mount /dev/hda5 /mnt/C
```

The mount and fdisk commands

Again, this list is in no way exhaustive. There are several more monitoring and maintenance commands that are beyond the scope of this book.

3.2.2 Backups

As every system administrator will agree, a crash is not a question of if, but when. There are just too many things that can go wrong!

Not only in offices, but also at home, backing up is a good idea. We have already explained how to make archives using the inbuilt "tar" utility. You can either do that, or back up your important files to CD or DVD. If you are not already doing this, please start right away!

A backup is meaningless if it cannot be restored. If you are backing up data on a CD, ensure that the CD is not damaged—

check for scratches, and always opt for a branded CD. When you make a tar file, check for the integrity of the file by uncompressing it into a folder and checking the contents. If the data is very important, back it up on two CDs.

3.2.2.1 Creating a Rescue Disk

During a crisis, a rescue disks can help you prevent a potential format/reinstall. Rescue disks can be got off the Net or created from within the OS itself. Floppies are now hardly being used as rescue disks owing to their limited size and unreliability. You can get .iso files online and burn the image on to a CD. Keep the CD handy. The rescue disk boots into a command prompt where you can type in "help" and press [Enter] to get the list of available commands.

3.3 Single User Mode

When things go wrong...

A robust rescue option in Unix and its derivatives, including Linux, is Single User Mode, where only the system administrator (root) account can log in, and only a few services are run (think of it as something like Windows' Safe Mode). Some administrative operations can only be done in Single User mode: for example, unmounting a user partition, since this is obviously not possible when a user is logged on. Another example is the fsck command on a user partition whose file system is corrupted. The fsck command works on raw data and not on partitions that are mounted. The root alone is mounted when in single user mode and there are no network connections available.

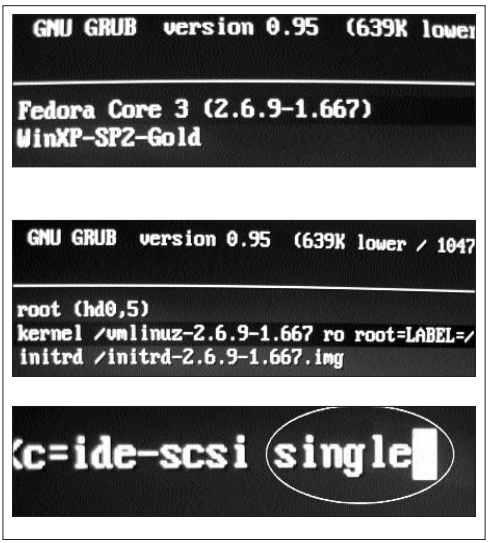
3.3.1 Entering Single User mode:

It is fairly simple to start Single User mode in Linux. All you have to do is make a small change in the boot loader. Most distributions now use Grub as a boot manager / loader. Here's how you can edit grub to start Single User mode:

1. When Grub appears with the OS choices menu, select Linux and hit [E] to edit the boot options.

2. Next, scroll to the line that starts with the word "kernel" and again hit [E] to edit it. Go to the end of that line, add a space and type in "single".
3. Press [Esc] to get to the main boot options screen, and then boot into Linux. Instead of getting the GUI, a command prompt appears where you can carry out system maintenance tasks. If you have observed, you have not been asked for the root password till this point and you are already given access to root commands! You can change the root password here as well. Yes, for those of you who are wondering, this is a security issue, but it is possible to do away with this risk by editing configuration files such that even in Single User mode, a password is asked for.

If you use LILO to boot Linux, just type in "linux single" at the LILO command prompt.



Once you're in Single User mode, you can change the root password with the "passwd root" command. Then you can reboot, go into the GUI and change all the other user passwords as well. This is how you can get back into a Linux system after you lose your root password.

Using the steps above, virtually anyone can set his own root password without knowing the original one. You have to realise that the idea of root is basically to prevent users from accidentally deleting other's data or modifying critical system files. Anyone

who logs into Single User mode is deemed to know the system well enough to not make careless mistakes. Also, earlier, the server on which the root would operate would also be physically protected, usually in a separate room. So there would be no ready access to the server. Now of course, with PCs, this vulnerability could be a concern, but there is a cure. You can set Linux to prevent Single User access. Hence emergency tasks have to be carried out using a boot disk and logging in with the root password.

3.4 Managing User Accounts

If you are using Linux, you would have created at least one other account apart from the root. If your computer is used by many people, some of whose work may be confidential, you would want to create a separate user account for each one of them. So what is a user account?

An account simply distinguishes different users of the computer, just like a bank account is unique to every account holder. On the computer, every account contains the files that the associated user has.

3.4.1 Creating a new user

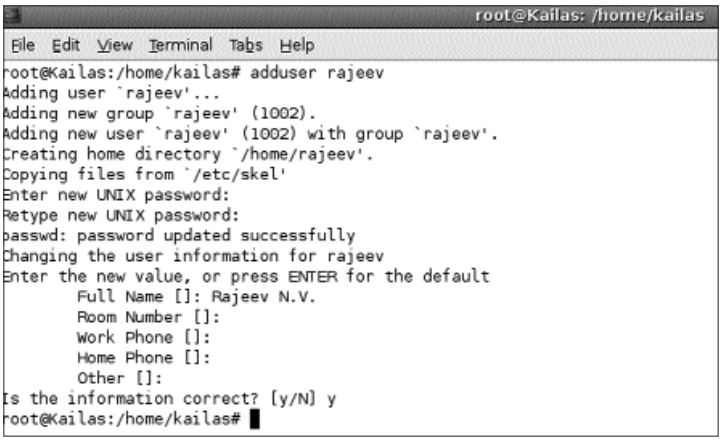
Most Linux distributions come with a program for creating accounts that can be accessed through the GUI. Alternatively, you can use commands to create users though the terminal. “adduser” and “useradd” are two commands that can be used to create a new user. A new user can only be created by the root login or user. When you execute the “adduser” command, you will be prompted to provide a username, password and other details. In less than a minute, you will have created a new user on your system!

To delete a user just, type in “deluser <username>” and press [Enter]. Be careful, however, as you will not be given any warning—the user is simply deleted!

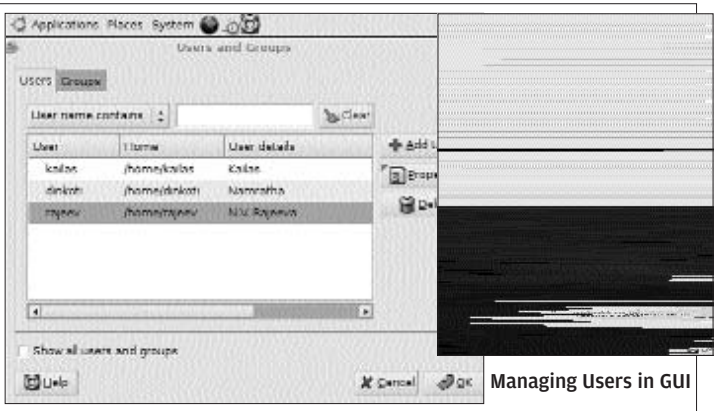
3.4.2 Changing user properties

There may be a GUI application (either bundled or add-on) which can be used to find and change user settings. But the sure-shot way of doing so is via the terminal. Here are some commands for changing various properties of an account:

chfn	Change the full name field
chsh	Change the login shell
passwd	Change the password

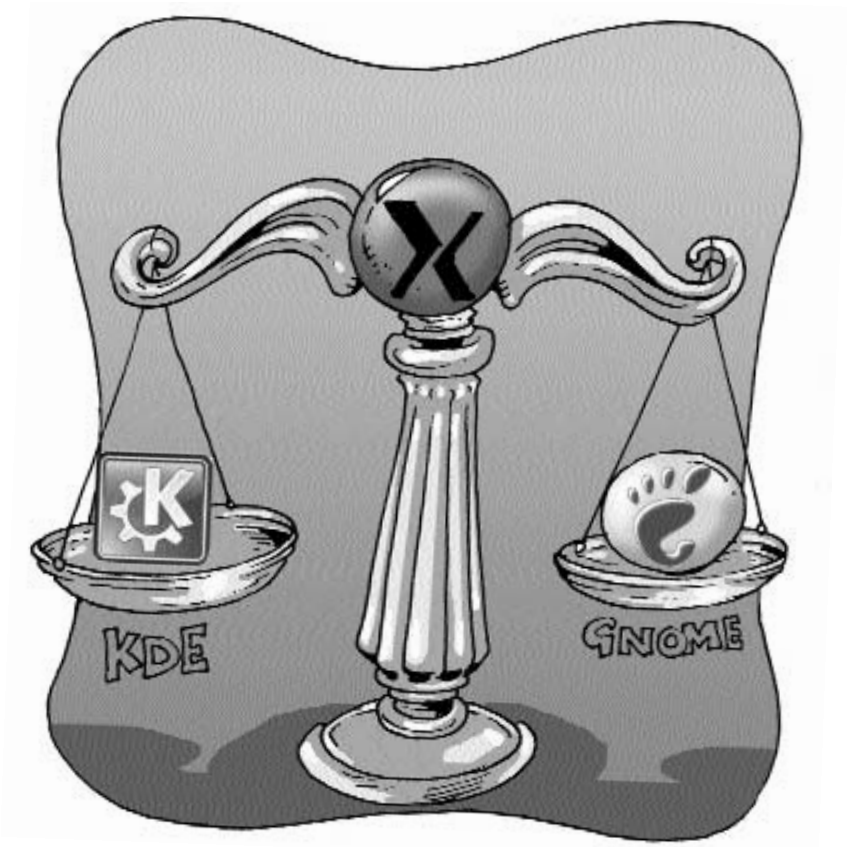


Editing user information. You may need root access to do this



Managing Users in GUI

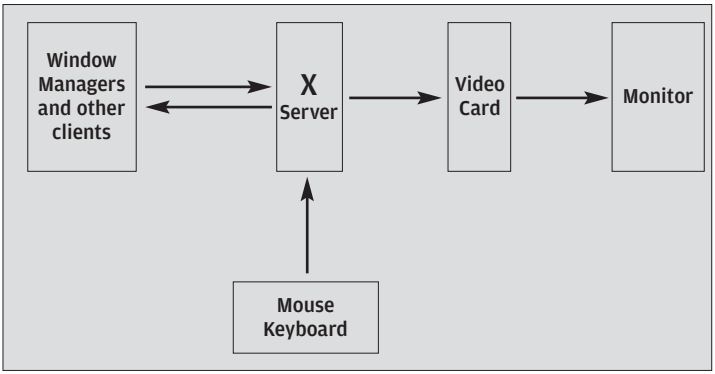
The X Window System



The biggest step in the evolution of operating systems from the naïve user perspective is perhaps the GUI (Graphical User Interface). Users need not learn and remember commands with their various options, and only need to point and click. But what makes everything appear so friendly on the screen? In Linux, it is the X Window System. In this chapter, we take a peek into its working. We proceed to tell you about the vast choices you have in making Linux look the way you fancy.

4.1 The X Window System

We will now look at what the X Window System is, and give a background of how the GUI came into Linux.



The X Server—Client Model

If you have observed different applications in Windows, you will find that all applications behave the same with respect to their window positions and the options they give the user to change the way they look (maximise, minimise, resize, drag-and-drop, etc.). This is because all applications use the same set of display functions that are built into Windows. For example, the developers simply need to "call" a particular inbuilt function in Windows for the Maximise, Minimise and Close buttons. All programs would use the same function—hence the uniformity. (If a program has to look different, the developers will have to write display functions of their own.) In Linux, however, you will see that different applications look and feel quite different from each other. Each "window" or application can have its own features. The way in which a user can modify the appearance of the window can also vary. This is due to the versatility that the X Window system offers to the different desktop environments, and the different tool kits that are available for developers. For example, if a developer so wishes to, he can write his program such that double-clicking on the title bar rolls

up the window and keeps only the title bar visible. Or, he can add the drag-and-drop feature to an application.

The X Window System is a system-level application, or rather, a process that handles the display in Linux. Basically, what you see on your screen is coming from X. X "talks" to the graphics display device (the graphics processor) and tells it what to display.

As X interacts with the hardware at a basic level, programmers need not be bothered about giving instructions to the video card for any display (mind you, the GUI taxes a system heavily). With the difficult job being taken care of, programmers only need to give instructions to X saying "move window 1 to position x" or "minimise window 2" or "refresh window 3 every 5 seconds".

The X concept came about when a GUI for UNIX was being worked out. It was released in 1984 by the *Athena Project*, an academic project undertaken at the Massachusetts Institute of Technology. It was taken over in 1998 by a the *X Consortium*, which has been maintaining the standards for X ever since. The X specifications are freely available for further development, and Linux developers adopted these specifications and called it the Xfree86.

Xfree86, like its base—the X—is very versatile, with various options that developers can use to make different windows (or apps) display differently. X follows a client-server architecture. X is the server, and the apps that need display access are clients. This is a client-server model because X can control windows on a remote PC via a LAN where the user is on a *dumb* (CPU-less) system. X can thus cater to several users simultaneously. Even on a standalone system, X acts as a server, and the desktop environments and applications act as clients.

X reads keyboard and mouse inputs and makes them available to the applications that need them. Based on the information that the applications get, they perform certain operations and if a result or a change has to be shown on screen, the applications send instructions to X to display it. This is how X services the desktops.

But X by itself does not define how content is displayed on the screen; it only provides a means to display this content. It throws open choices to the client to choose how content is displayed on the screen. When the client asks for a particular action to be done for the display, the server performs it. The X server does not place any restrictions on the client to behave in a particular fashion. In fact, different clients can behave differently using the same X server.

X maintains a list of all its functions in a library file called XLib, using which all data interpretation and visual outputs are handled at the system level. (Basically, "system level" means interacting with the hardware.)

There is a server in charge of visual output and data input, client applications, and a way for them to communicate between each other. The client simply has to tell the server, "give me an area x*y pixels large in the right bottom of the screen". So, if the client needs data from the keyboard, it would ask X for it rather than monitor the keyboard itself.

4.1.1 Xfree86 and X.Org

The battle in the Open Source community

The X Consortium was formed in 1988 and spearheaded the development and management of the X Server and the tool kits needed for application development. In 1993, the X Consortium, Inc. was formed as a successor to the original MIT X Consortium. In mid-1997, the X Consortium began handing over the responsibility of X to *The Open Group*. The Open Group consisted of vendors who were interested in implementing the X server for Linux applications

and hence had a keen interest in its growth. Trouble started in 1998 with The Open Group's release of an X Server version called X11R6.4, which had a radical change in its Licensing Policy. The X Server suddenly seemed more proprietary than open source, beating the very intent of the original X Consortium. However, The Open Group claimed it needed assured funding to keep the development of X going.

Another group, called the Xfree86, originated in 1992, which again was formed after a dispute over another proprietary X Server. An X Server, X386 1.2E was being made into a proprietary software which led purists to form the Xfree86, somewhat as a dig on the X386 name.

Now, it so happens that the Xfree86 also disagreed with The Open Group for the latter's proprietary attempts. This forced The Open Group to continue distributing the X Server under the original, more liberal licensing policy.

The Open Group founded X.Org in 1999, in an attempt at gaining some technical inputs. Most of the development that had happened since the dissolution of the original X Consortium had come from Xfree86, and so Xfree86 was invited to join the X.Org as an honorary non-paying member.

However, things were not all too well within Xfree86. As Linux's popularity grew post-2000, and as did the demand for X Servers, Xfree86 had built a reputation of not being a pure open source group—the source code between versions was not available to the public, only the final source code was. This placed almost complete control of the development of X in the hands of the Xfree86 group. After much debate, Keith Packard, who had joined XFree86 after the end of the original MIT X Consortium, was expelled from Xfree86.

Moving towards a full commitment to open source, Xfree86 and X.Org looked towards the development of X and formed the

X.Org Foundation. The Open Group gave the foundation the right to use the X.Org domain name, in effect becoming the front for X and being recognised for all the development that it did. This also brought many developers to the X.Org Foundation, which till then mostly consisted of vendors. X.Org Foundation membership was free for individuals, and corporate membership came for a sponsorship, which was necessary for X.Org to sustain itself. Several corporates such as HP, Sun Microsystems and IBM are now members of the X.Org Foundation.

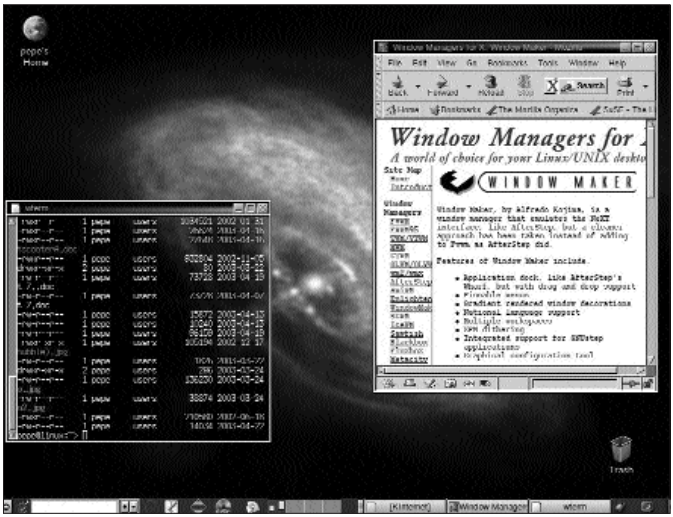
4.1.2 Window managers

A window manager is an X client responsible for positioning windows on the screen, title bars, displaying messages to the user, focusing windows, and reading customised key strokes and mouse behaviour.

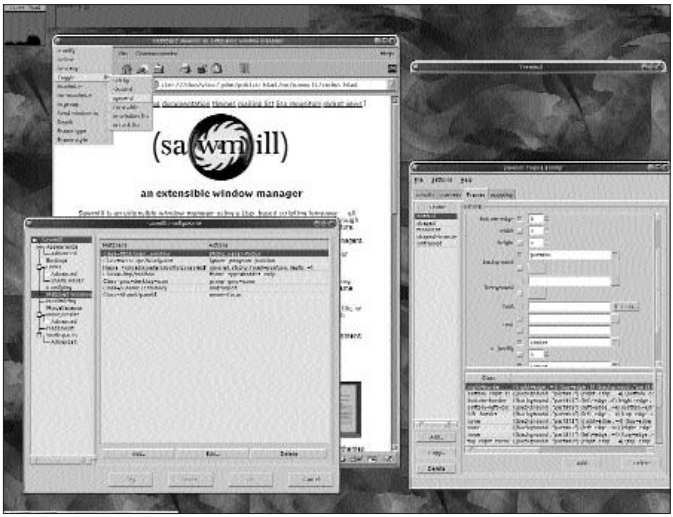
Some of the commonly-used Window Managers are KWin, Metacity, MWM, TWM, Sawfish, Enlightenment and BlackBox. KWin is the default window manager for the K Desktop Environment (more on desktop environments in a while), and Metacity is the window manager for the Gnome desktop environment. There are plenty of window managers you can download and install to change the look and feel of Linux. It is possible to give a whole new avatar to your computer with the different Window Managers and Desktop Environments.

Each window manager is created with a different focus. One can be fast and slick, another navigable, another customisable and another simply minimalist... And for the "I want Windows" people, there are also window managers that ape anything from Windows 3.11 to XP.

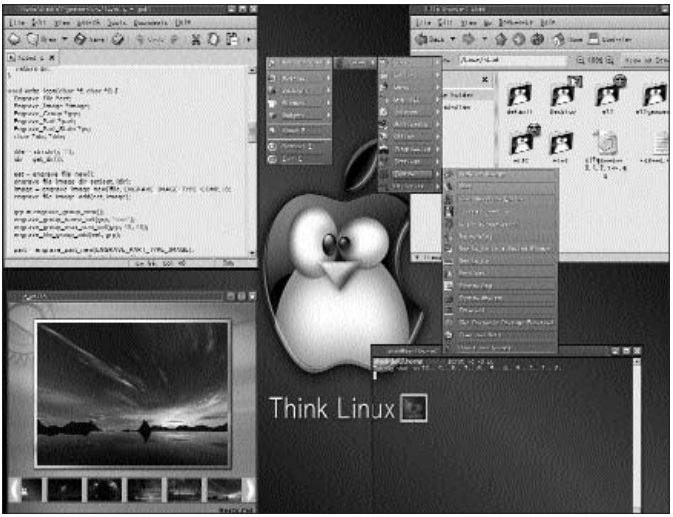
Window managers do a lot more than simply resizing and moving windows. They provide what has come to be a cool feature in Linux—multiple desktops. You can have four or more desktops and store files, shortcuts on them just like a normal desktop. You can switch from one desktop to another with a simple mouse



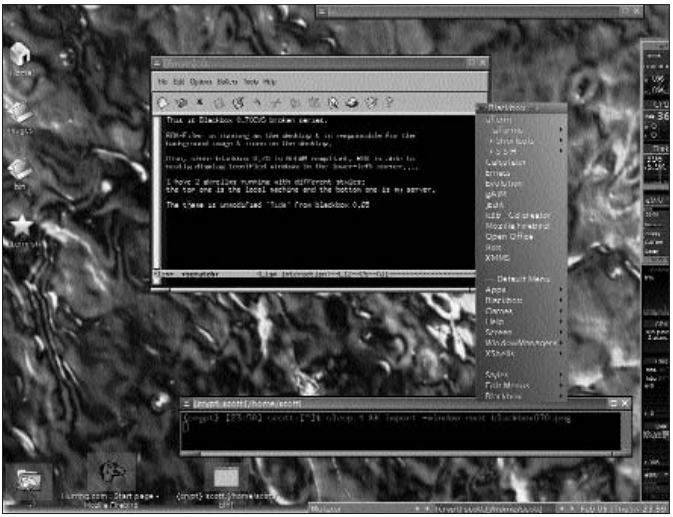
Metacity Window Manager



Enlightenment Window Manager



Sawfish Window Manager



Blackbox Window Manager

click! Window managers also provide menu(s) for launching applications. Of course, different managers place the menus in different places and organise them differently. That's why we keep saying, "learn the commands"!

4.1.3 Desktop environments

A Desktop Environment is a collection of X clients, including the Window Manager(s). It includes various applications and a collection system tools for the user. The system tools are, more often than not, a GUI for existing commands. This makes learning system management easy for new users. In a way, the Environment can be viewed as an extension of the window manager, just with applications bundled along. The two most commonly used Environments are KDE and GNOME. You may come across Linux enthusiasts arguing about which is better. Well, some people like hockey and some like football. Both KDE and GNOME come with a good set of applications such as office suites, file browsers, multi media players, etc.

If you speak to hardcore Linux fans about desktop environments, you will probably hear them bashing both KDE and GNOME! This is because, as they will tell you (rightly so), desktop environments are huge memory hogs. Besides, everything can be done from the terminal itself, so why use a desktop environment? Well, it's for the beginner—desktop environments let beginners get used to the feel of Linux, while gradually letting them move on to controlling all tasks through the command prompt.

One difference between KDE and GNOME, however, is that it is relatively easier to install different window mangers on GNOME than on KDE. This is because GNOME inherently relies on external download managers while KDE has its own download manager.

4.1.3.1 The K Desktop Environment

The KDE project, officially announced on 14th October 1996, is an open group of developers from all over the world. Today, KDE is one of the two most commonly used desktop environments in

Linux. KDE (along with GNOME) has played an important role in the penetration of Linux to many Home PCs today. KDE code is available under an open source license, meaning that the KDE source code is available free of charge to anyone who wishes to modify it.

The KDE logo

KDE claims that membership to the group 'The KDE Core Team' is solely merit-based, and the applicant must have distinguished him/herself through outstanding contributions and dedication in the KDE project over a considerable period of time. The 20-member KDE core group decides by means of democratic voting procedures on important issues.

Applications for KDE are developed using the Qt toolkit made by Troll Tech. Qt is a development framework written using C++. It has tools which can be used to develop applications for Unix and Unix-derived platforms. KDE was under attack from Linux purists for having used the Qt toolkit, which at one time was not under a free software license. If a developer wrote an application with the freely available toolkit and decided not to share the source code, he had to pay a royalty to Troll Tech, the makers of Qt. However, if the developer chose to share his source code along with his software, he was not obligated to pay anything to Troll Tech. This semi-open source arrangement was not taken to very kindly. Now, however, Qt is fully open source.

Like most open source companies, KDE, too, is backed by a team of enthusiasts who work for fun in their own chosen way, rather than work for remuneration and adhere to a well-defined rules.

From KDE's Web site (http://www.kde.org/whatiskde/devmodel.php):

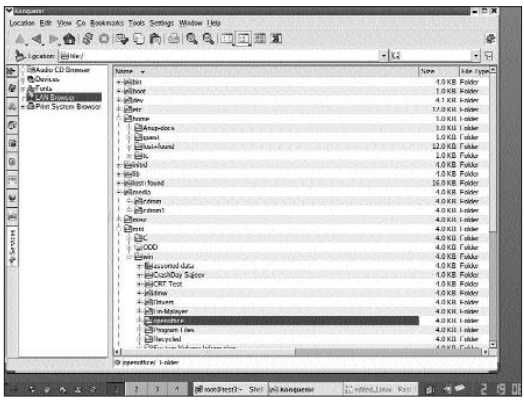

A KDE Desktop

It is better to heave 100 excited and motivated developers improve the project iteratively than having five developers working according to a blueprint and thus in the most efficient way. The group of five developers will be bored soon since there is little or no room for them to bring in their own ideas and they will fail to attract fellow developers for the same reason.

4.1.3.1.1 KDE applications

The first thing anyone needs is an Office application to make documents, presentations and maintain accounts. KDE comes with KOffice—a free, integrated office suite consisting of a word-processor, spreadsheet, presentation application, organiser, and more utilities. KOffice 1.4.2 is the current official release, while a beta of the 1.5 version is also available.

Konqueror, the Web Browser bundled with KDE (though Firefox is also included in newer distributions) supports the full gamut of current Internet technologies—JavaScript, Java, HTML 4.0, CSS-1 and -2 (Cascading Style Sheets), SSL (Secure Socket Layer), and Netscape Communicator plug-ins (for playing Flash, RealAudio, RealVideo and similar technologies).



amaroK is a versatile audio-player for Linux. It allows you to create and edit playlists with a simple drag and drop, has a 10-band equaliser, supports crossfading, can download lyrics, integrates with CD burning software, and even connects to your iPod!

CD burning is a lot easier with a powerful application called K3b. It is the best CD burning burning software for Linux, and one of the best across platforms. You can rename all mp3 files to be burnt in the artist-title format. CDs can be burnt as Data, Audio, Video or Mixed CDs. With multi-session and on-the-fly burning (without making an image file), you save both time and CDs. This is a must-have KDE application if you write many CD / DVDs.

Krusader is an advanced twin panel file manager for KDE 3. It can support the tar, zip, bzip2, gzip, rar, ace, arj and rpm archive formats. Krusader can display mounted partitions, has batch renaming, file content comparison and other advanced functions. It is also extremely customisable.

For BitTorrent fans, there is a BitTorrent client called Ktorrent, which incorporates an inbuilt browser to search various engines for torrent files. It also supports upload speed capping and IP blocking, and is expandable via plug-ins.

KIO Network Transparency is an application for network tasks such as file sharing across different types of networks.

These are just some applications that come with the KDE desktop environment. These apart, there many more applications for KDE and are included in the distributions.

4.1.3.2 The GNOME Desktop Environment

The other major desktop environment is the GNOME Desktop environment. This is also a free and open source environment developed for UNIX and its derived platforms. You can use GNOME's source code to build an in-house application for your organisation without having to share the source code or pay royalty, so long as you don't

distribute your work to the general public (in which case you'll need to share the source code as well).

The GNOME Logo

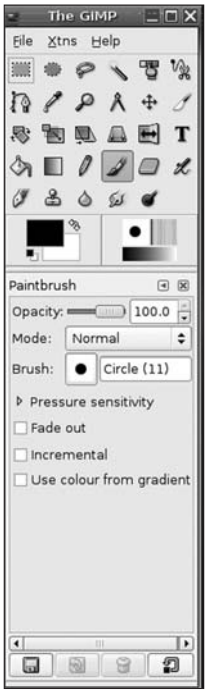
There is less software written for GNOME than for KDE, but the sheer numbers are still huge! Software is written for the GNOME environment by distributors and even enthusiasts. GNOME maintains that *usability is about creating software that is easy for everyone to use, not about piling on features*. A notable application on GNOME is The GIMP (GNU Image Manipulation Program), an image editor that claims to rival Photoshop. Applications for GNOME are written using GTK (The GIMP Tool Kit).

4.1.3.2.1 Some GNOME applications

A GNOME Desktop. Observe the menu at the top.

- GNOME bundles Nautilus, a file manager for GNOME. Nautilus can also function as a Web browser.
- For CD ripping and playing, there is Goobox or Grip.
- An image browser called F-Spot allows you to tag images for quick searching, has slideshow and full screen modes, can view and export EXIF and XMP metadata and has many other features. Editing modes include crop, resize, red eye adjustment, brightness, contrast etc.
- If you don't have OpenOffice.org, you can use AbiWord, which is a word processor with features to help in everyday work.
- Balsa is an e-mail client that supports multi threaded mails, has a spellchecker, multiple character sets, etc.

GTK+ is a platform used to write applications for the X Server. GTK+ is free and part of the GNU Project. With the licensing terms for GTK+, it is legal for all developers to include those developing proprietary software, without any license fees or royalties. GTK+ is based on three libraries called GLib, Pango and ATK, each of which is used to perform specific functions when developing X clients.



Various tools in GIMP

The current version of the GNOME Desktop is 2.12, in which work has been done to make it easier to port GNOME tools to Windows as well. This version is available as a Live CD, whose image can be downloaded using BitTorrent (http://torrent.gnome.org).

Corporate support for GNOME comes from big names such as HP, IBM, Novell, Red Hat and Sun Microsystems, among others. Among the major distributions, SuSE and one of the hot newer Linux distributions, Ubuntu, uses GNOME as the default desktop environment. Gentoo is another distribution that comes with GNOME. Fedora, Mandriva and others give an option to install GNOME, though the default is KDE.

4.1.3.3 The KDE vs. GNOME battle!

When there are two things of the same kind, there is bound to be a competition. Especially so if both are nearly as good as each other! We have told you *about* KDE and GNOME, but what really sets them apart? Is it possible to say which is actually better?

KDE and GNOME belong to the same era, with the former being founded in 1996 and the latter, a year later in '97. KDE has been coded with the object-oriented C++, while GNOME stuck to the more conservative C.

Look and feel

To start with, let us look at the first thing that a Desktop Environment is supposed to do—make it easy for the user to operate the computer. This is the primary goal of a GUI. If everyone could remember more commands than there are verses in The Mahabharata, then we would not have GUIs at all!

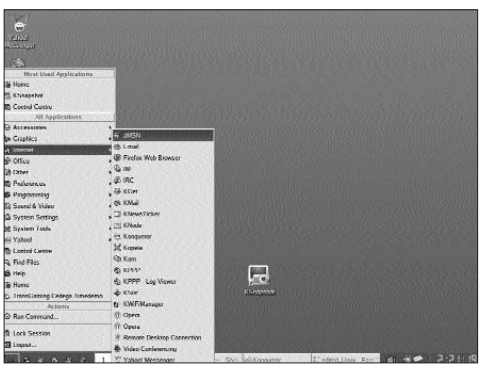
KDE does an excellent job making the Linux newbie feel at home, with its familiar Windows-like feel. GNOME, on the other hand is not so loaded with graphics, but is still easy to use. So here, choosing between the two is just a matter of preference.

The KDE Desktop has a Windows Start Menu imitation at the left bottom of the screen called 'K'. Clicking on it will open up a menu that shows the different software that are installed. For example, there is a menu that says 'Internet', which contains shortcuts to the browser, e-mail client, etc. Then there is a menu called 'Office', which contains the different applications of OpenOffice.org. The time display is to the much-familiar right, where you also find the shortcuts to the four different desktops.

GNOME, however, puts the menu on the top. At the bottom, however, there is a "show desktop" icon—a nice idea! There's also a shutdown menu right on the GNOME-panel (the equivalent of a taskbar).

Applications

Strictly speaking, an application written for KDE must work on GNOME, and vice versa, provided the libraries that the applications use are also installed. So you are not required to use only those



The K Menu

The GNOME Menu

applications that are only written for your desktop. After all, one application of your choice might be in KDE and another on GNOME. You cannot keep changing desktops, can you? In the real world, you might run into problems while trying to use applications that are not written for their native environment. If they work, they might have some problems, or just take a longer time to launch, etc. The idea is not to discourage you from trying applications from another desktop, but it's better to do some research before you do it.

KDE has a lot of applications bundled with it. The KOffice is something to take note of. It can do all that you want it to do with a document, and yes, it is fast, too! KDE has a music player called AmaroK with an exciting and easy-to-use interface. GNOME has its own music players, but not as well made as AmaroK. With a huge bundle of applications that KDE comes with, installing other desktop applications can be a problem, but GNOME on the other hand has lesser issues with installing third party applications.

Customisability

This is the best thing about Linux, isn't it? So how much do these two desktops allow you to change? Well, both of them can be made

to look and feel entirely different, but you will see a lot more options in KDE. GNOME scores lesser in this area. While there are several setting that you can change in GNOME, it is still not as much as KDE offers

Resource hogs?

A GUI by itself is a resource hog. There are so many extra calculations that the CPU is doing only to enable display on screen. This apart from the programs it has to run. KDE has a lot of eye candy inbuilt, which makes it inherently resource-hungry. More applications and complexities mean lesser stability. It isn't that KDE is unstable, but it's still not as stable and rugged as GNOME. A GNOME machine is far less prone to hanging than a KDE one.

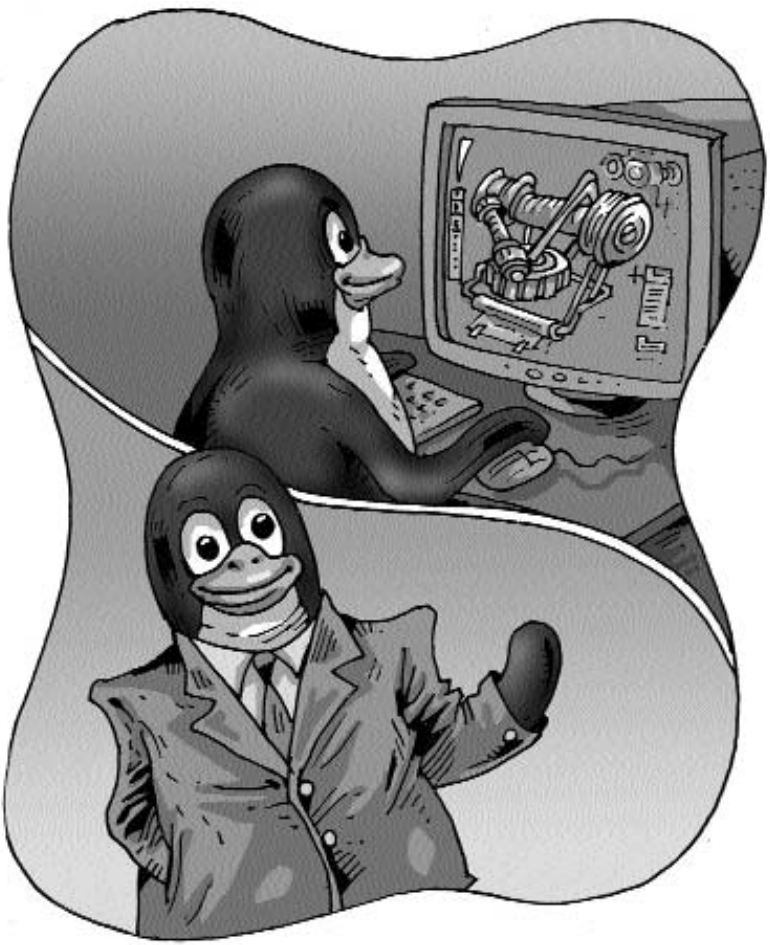
Being lighter, GNOME responds faster to mouse clicks or keystrokes than KDE does when many applications are open. Bigger isn't always better, you see!

The verdict

Sorry, no verdict. Because we think there is no absolute *best* desktop out there. It is only about the best desktop for *you*. We can only risk saying that KDE it is if you like graphics or are new to Linux and have a system loaded with RAM (at least 512MB) and GNOME if you want a more stable system that uses less resources.

When you begin using Linux, which desktop to choose is the least of your worries. You should be concentrating more on configuring Net access and setting up your applications, and only then think about your desktop environment. After all, you can change it to what you want, whenever you want!

Tools And Applications



Now comes one of the most important parts of this book! You've installed Linux, learnt a few commands, and are ready to bein using it a lot more, but are theye enough application built for it? Does Windows really beat this penguin hollow? Read on and find out...

5.1 Simple Text Editors And Utilities

A common crib about Linux is that there aren't many applications written for it. But one only needs to take a look at the myriad of applications that come with the bigger distributions—SuSe, Red Hat and Fedora. Office applications, image editors, multimedia—they're all there. The K Desktop Environment (KDE) package contents are organised under suitable headings in what appears like a Windows Start Menu.

Apart from Open Office, there are plenty of small tools that come in handy for various personal and even office purposes. In KDE, for example, you have floating sticky notes called KNotes, where you can jot down, well, notes. This being a scribble pad, you don't need to save any files—the text you typed in is retained. There is also a feature-rich alarm, which can even send out e-mails! At the scheduled time, the alarm can open up an image or a text file.

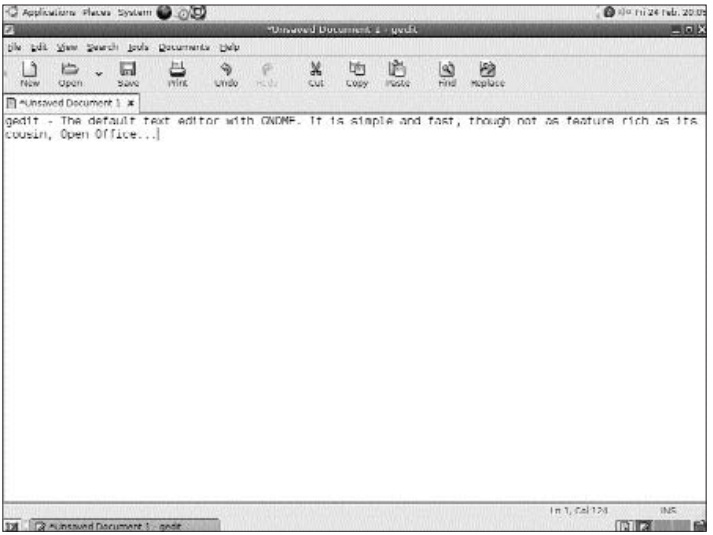
Need to make a quick calculation? You have a quick and easy-to-use calculator called KCalc. As you've guessed, applications whose names start with "K" are part of the KDE package. Gnome includes its own calculator that is called simply *Calculator*.

Came across a word you're not familiar with? There is Kdict for you, a dictionary (in KDE, of course, in the Applications Menu.). In Gnome, you can find a dictionary in the ***Applications > Accessories*** menu. Both these calculators look up an online dictionary.

5.1.1 gedit

The Gnome equivalent of Kword is gedit, the default basic word editor. We liked the look and feel of gedit better than Kword, but of course, this is subjective.

On the functionality side, there are icons at the top of the window that you can use for the basic copy and paste operations, and also a Find and Replace icon! The Edit menu throws a nice surprise



for an application this size. There is an Insert Date and Time option. In the same Edit menu, click on Preferences, and you can change your default text font and colours, besides other settings. Another nice feature is auto-recovery, which defaults to save your changes every 10 minutes, but this can be changed to a minute if you'd like.

Prone to spelling mistakes? Use the Spell Check available in the Tools Menu. This is as close as we can get to a proper office suite. gedit can be comfortably used to compose common office documents. And, it's fast.

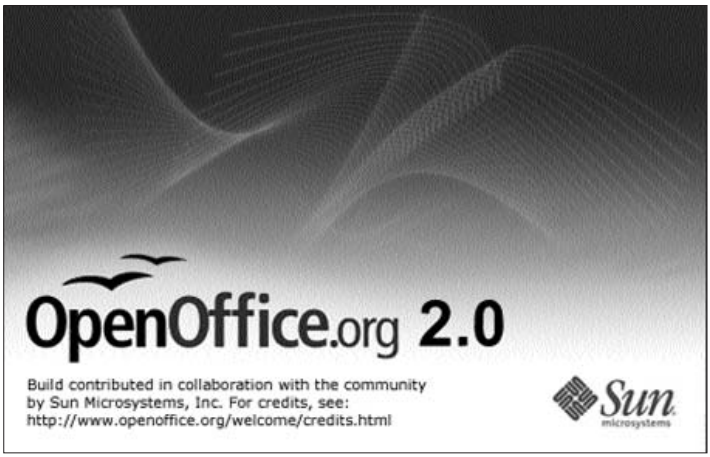
5.1.2 The gThumb image viewer

This is an easy-to-use image viewer and editor. Thumbnails of images in the selected folder appear to the right of the screen, while you can select your folder for viewing in the left pane. Double-click to open the image and find some oft-used tools. When the image is open, click on the Image menu to find some nice tools. The Enhance option adds a nice touch of extra contrast and brightness, making the image look richer. Of course, we aren't

saying gThumb matches up to The GIMP, but it sure does enable you to quickly crop a huge image, and adjust its brightness / contrast before e-mailing it! if you are not looking for serious editing, but just some basic adjustments, you can certainly use gThumb.

5.2 OpenOffice.org

OpenOffice.org is an open source package that contains Open Office Writer for creating documents; Open Office Impress for designing presentations; Open Office Draw, which can be used to make drawings and edit 2D and 3D objects—and even make some



basic page layouts; Open Office Calc, which resembles MS Excel, and which can be used for creating spreadsheets and tables; and Open Office Base, a database creation tool.

A notable feature of Open Office is that it allows you to save your documents as a PDF, which is very useful for those who wish to create professional-looking documents with ease.

A nice feature in Open Office is the Math application tool, which is extremely handy when it comes to inserting formulas

into documents. To insert a formula from Math into Writer or Impress, go to ***Insert > Object > Formula***. The Math window opens up, and you can choose from the different formulas listed.

We will take you through the features of Open Office so you can start your work the moment your Linux installation is done!

5.2.1 Open Office Writer

Typically found in the Office tab, opening it is nothing more than a two to three mouse click event. When you open Writer, the icons will seem more or less like those in MS Office. But you will see a well-defined box inside which you can type. This is nothing but the actual text area shown separately from the margins. If you're confused about a feature or option, choose the Help menu. There are detailed indexed help topics available.

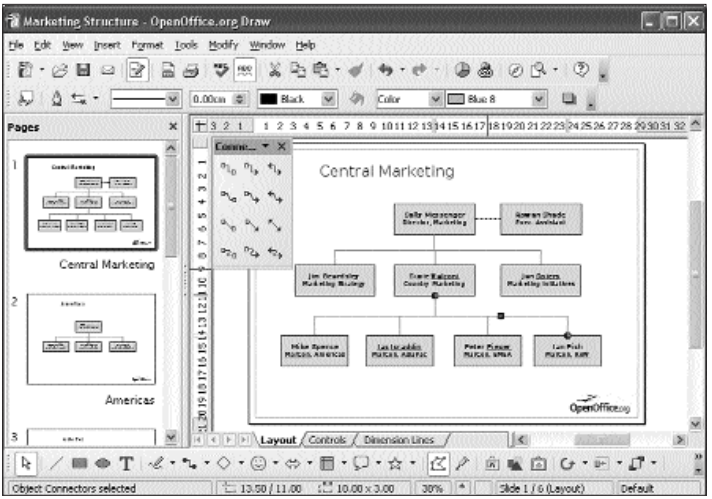
Attention has been paid to detail, and this is very apparent—no feature available on commercial, proprietary office suites is left out. Menus and sub-menus are intuitively placed in Microsoft Office style.

The number of fonts in Writer, however, is very small compared to its Windows counterpart, at least in the quick install of Fedora that we did. The all-time-favourite Times New Roman is not available, but you'll find an exotic-sounding Nimbus Roman as the default. In place of Wingdings, you'll find fonts called Open Symbol and Standard symbol.

Comments marked into documents with MS Office can also be viewed and edited with Writer. You can also store in the .doc format for Word 95 or XP—good compatibility here!

5.2.2 Open Office Draw

Imagine something as simple as a paintbrush app, but with many, many more drawing tools. And more fun too. Take a look at Open Office draw. At *Digit*, we got pretty busy playing around with this wonderful, fun-filled tool! You can, of course, use this tool in



Writer or Impress, but unfortunately, you cannot store the image you create in any of the standard formats! Somewhat kills the fun, if you ask us.

5.2.3 Impressed by Impress!

There are many templates to choose from here for your presentation. We also feel the interface is actually better than that of PowerPoint. The drawing options you find in Draw are also available here, or an already-created drawing can be imported. Impress allows you to save the presentation as a .ppt file, but if you're creating the presentation on Impress and plan to use it with PowerPoint, do check beforehand if all the slides appear as designed before you make the presentation!

5.2.4 Open Office Calc

The MS Excel equivalent in Open Office. How different could a spreadsheet application get? All we have to say here is that we found no obvious features missing.

If you don't have Open Office on your computer, you can simply download the installer from ***http://download.***

openoffice.org/index.html. It is available as a .tz archive from which you can install the suite (for details on installing applications on Linux, refer to the chapter on System Maintenance). The file size is about 100 MB for the latest 2.01 release. If you are on dial-up, find a friend who has broadband!

Open Office is available for Windows as well! So even if you are using two OSes, you can have compatibility with office documents.

5.3 The GIMP

In an earlier section, we already spoke about this cool tool. If a free software does the same thing as one of the most expensive imaging software, it merits more than a mention here! And yes, in case you don't take to Linux, The GIMP is available for Windows as well! It's not only free, but also open source. Could it get any better?

The GIMP is written and developed under X11 on UNIX platforms, but the same code also runs on Windows and Mac OS X. It is designed to be easily augmented with plug-ins and extensions to do more than what it already does. There are over a hundred plug-ins available.

The basic painting / colouring tools include all the standard ones—Brush, Air brush, Pencil, Clone and also custom brushes and patterns. There is a powerful gradient editor and blend tool. For serious graphics editors, there is a surprise in store: unlike Photoshop, The GIMP does not place any restrictions on file size. Also, any number of images can be open at the same time, so long as there is enough RAM!

Have you done a lot of changes to am image or document only to realize you need to undo all of them, but you cant go beyond a certain point? The GIMP allows unlimited Undo / Redo options!

After songs, the next-most stored type of file on home computers must be image files. And the number of image files on computers is growing, thanks to everyone becoming amateur photographers with digicams. How many of these photos are perfect? Yes, we agree that digitally touching up an image mars its sanctity, especially if it is a memorable one. But what about those wallpaper shots you took that are *almost* perfect, but which can be helped with a little bit of touch-up work? And what about that really huge 5 MB image file that you got at maximum resolution from a digicam that you want to save at a smaller size and resolution?

We will introduce The GIMP to help you with all the above. The GIMP interface is the same on Windows and Linux. You can therefore use this guide even when you are trying The GIMP on Windows.

When you open The GIMP, you will find two windows, one titled "Layers, Channels, Pixels, Undo...", and the other which simply says "The GIMP." To open an existing image, go to "The GIMP" window, and as usual, click ***File > Open***. Once the file is open, you will see the various options in the image window. At the bottom left of the screen, you can select the zoom level. There are well-thought-of zoom levels from 12% to 800%, but unfortunately, you cannot specify intermediate values. Right next to the zoom control is the unit preference, where you can select the unit shown on the scale—pixels, millimetres, centimetres, inches, and up to metres!

In theory, The GIMP is able to handle such large images. As a first-timer to the software, you can simply open a sample image, click on every menu and see what it contains. Don't be overwhelmed—just play some soft music in the background and take your time exploring the software. While this is a hard way to learn, we can make life simpler for you (as usual!) and guide you through doing simple image manipulations.

5.3.1 Changing the size of an image

To resize an image, in the Image menu on the top of the screen, select Scale Image. You can then specify the image size in centimetres, inches, or other units or as a percentage of the original, or even in pixels. (You will find a locked symbol adjacent to the height and width box. This is to preserve the aspect ratio.) For most purposes, you can leave the X-Y resolution as it is at 74 pixels/inch. For best quality, set the interpolation to cubic.

A word of caution, though—when you increase the size of the image, you will suffer some loss in quality. With a good image, you can go up to 200 per cent—anything beyond that will be noticeably hazy.

5.3.2 Cropping an image

Not all of us are photography experts. Often, the background will be more prominent than the subject itself! If you want to eliminate an unwanted area of the image, just use the crop function: keep your image window and The GIMP window open. In The GIMP options, use the select tool—the first tool in the list. Choose the ***Select*** tool, and click and drag on the part of your image that you want to keep. The selection will now be showing in the rectangle. Now go to ***Image > Crop Image***. That's it! If your selection was not proper, then just Undo it by pressing ***[Ctrl] + [Z]*** and redo the process.

5.3.3 Reducing the file size of an image

Let's say you have want to e-mail a picture taken using a digicam, and the file size is too high, some 1 MB. You can try to compress the image using a slightly lesser-quality setting. The eye may not be able to make out the difference at all. But just how much compression you can apply is something you have to decide by trial and error, as every image's quality is different. It could be an already compressed image, or it might be digitally modified, which increases distortion. So there is no rule of thumb while deciding on the amount of compression you can apply.

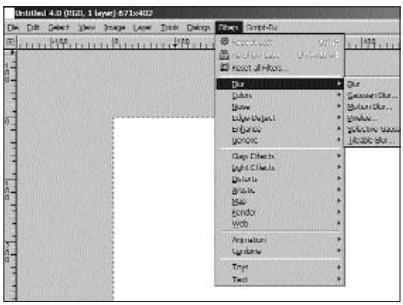
When you have a .jpg file open, choose ***Save As*** from the ***File*** menu. When you give a file name and say ***Save***, you will be given an option to choose the Quality. If you set it to 100 per cent, no compression takes place. You can set it to, say, 80 or 70 per cent (or even lower if you don't mind compromising a little on image quality) and then click ***OK***. Your file size is now reduced, for the same image size. If you want to compress it further, then open the original file and choose a lesser quality. Don't work on compressed files, as you are more likely to lose quality.

Now, resizing and rescaling options are available in other photo editors as well. So what is special about The GIMP ?

Software such as The GIMP or Photoshop use algorithms that produce less distortion while making changes to the image. So the amount of scaling or compression you can do with these software is usually more than what you can achieve with other algorithms that may not be as lossless as GIMP's.

5.3.4 Using Filters

This is where the fun starts! While photography purists loathe filters, saying it amounts to molesting a photograph, not all of us are blessed with the equipment or skill to bring out our ideas on the photograph. Filters can give a whole new feel to the image.



Using Filters to enhance images that didn't come out as perfect as you would have liked

Click on the Filters menu, and you will see a list of filters you can use to modify the feel of the image. You can also add elements like a glow or glare. The filters can be applied to the whole image or to a selection of the image.

Smoothen an image: If your image contains dots (typical of a photo taken in low light), you can try the ***Filters > Enhance > Despeckle*** filter. Set the values so you get the best results. If Despeckle does not smoothen the image, you can try the Blur filter.

There are several options in Blur, such as motion blur, which can cause the effect of the pic having been taken while moving.

Sharpen an image: This filter makes the edges of objects stand out more clearly, but at the same time, it can also cause unwanted noise. Find it in ***Filters > Enhance > Sharpen.***

Light effects: Want to add a sun? Or a star? Or create a glare effect? This is great for party shots. Try the Light Effects. There's lots to do by trial and error here.

There are many more filters, of course, and if you want even more options, there are always plug-ins you can get.

The GIMP can be downloaded from ***www.gimp.org***, where you will also find documentation and resources. The latest release is version 2.21.

5.4 Inkscape

We raved about what The GIMP is and what it can do. But there is something else in open source that can help advanced graphics designers manipulate 3D vector graphics. Inkscape works on the W3C (World Wide Web Consortium) standard Scalable Vector Graphics (SVG).

SVG is a language to describe graphics in XML, which will also make the graphics Web- compliant. Inkscape is a relatively new application that sprang up in 2003 from the ***Sodipodi*** graphics editor, a free graphics editor released under the GNU GPL.

So why Inkscape or even vector graphics?

Vector graphics are inherently very high-quality graphics. While a bitmap is defined by rows and columns of pixels, a vector image is defined by straight, and if needed curved, lines. Thus, almost any kind of image can be represented in its true shape by vector graphics: an image, after all, is composed of lines and curves. On the other hand, a pixel will cause some distortion at the edges of a shape, especially when magnified. Thus vector graphics is used when high-quality, distortion-free images are desired.

Inkscape does not quite have all the options that a professional graphics designer might want, but still has plenty of options for an enthusiast to get hooked on to image editing. It can import all common image types as .png or other vector-based formats.

You can either edit an existing image or create your own using the Pencil, Pen and Calligraphy tools. Standard shapes can be used, such as rectangles, ellipses, stars, polygons, spirals, etc. You can even introduce text and make changes to it. Moving, Scaling, and rotating can be done free-hand or by specifying values. There are a lot of options you can check out by simple trial and error. It is easy to be captivated by this program if you are the artistic kind! By the way, the maximum zoom supported is 128x—not that you'll need it, of course!

We will very quickly run you through the very basic operations of Inkscape and leave you to do the rest. The screen looks more forthcoming and friendly than does The GIMP's. You can open an existing file or create one yourself afresh. When you open Inkscape, a blank work area is available, or you can click ***File > New***, and select from the various size formats available.

You have standard ready shapes to start your work on—rectangles, stars, ellipses, and circles, from which more combinations can be developed. On the canvas, just click and drag using the appropriate shape tool to create the shapes. Each shape tool

contains numeric entries that pertain to its shape, the rounding, and so on. You can slant and rotate a shape to make it appear as if it's being seen from an angle. Combining three of more such shapes give rise to 3D model, and the extent to which you can play around with such objects are endless. Inkscape, like any designing software, requires you to invest some time and effort. But the results will be rewarding in terms of the images you'll be able to create.

If you don't already have Inkscape, you can download it from *http://www.inkscape.org/download.php*.

5.5 3D Modelling With Blend

Blend is a free, open source 3D modelling application. There are many modelling tools such as Polygons, Curves, Vector fonts, and more. These are basic shapes from which 3D models are created. On these shapes, you can choose vertices, edges or faces to edit. Once this is done, you can animate your shapes with various tools and options. In fact, there are so many things to do in Blend that you will initially be overwhelmed. It does take a few long sessions to know that what is what in Blend. But with a little patience, you can create your own works of art and animation!

Blend can also create interactive 3D games with transparencies and reflections! There is a graphical editor for defining interactive behaviour without the need for programming it. Games can be played back during editing without pre-processing or compiling, thus saving a lot of time in the design process. Files are saved in the .blend format, which supports compression, digital signatures, encryption, and forward/backward compatibility. Blend can read and write the TGA, JPG, PNG, Iris, SGI Movie, IFF, AVI and QuickTime GIF, TIFF, PSD, and MOV (Windows and Mac OS X) formats.

5.6 CAD On Linux

You've probably been hearing of CAD for a long time now—a civil or mechanical engineer friend of yours keeps mentioning it. Well, "CAD" is short for Computer Aided Design. In simple words, it is the process or method of designing a physical entity or object, using simulation, on a computer. CAD has revolutionised the design process for buildings, automobiles, machinery, etc. A civil engineer can view the effect and appeal of a building with different structures, a mechanical engineer can verify if the various parts of a machinery match, and so on. CAD is important as it helps you understand complex structures by displaying them which you otherwise had to physically see or just visualise.

There are several CAD software out there, and we will mention some of them which are available for Linux. CAD software are generally complex to handle, and need some getting used to. While not all CAD software is available for free for Linux, you can still learn and get familiar with CAD with the free applications that are available. The basic operations will be more or less similar across the software.

One of the first software you can try out to get a hang of CAD is ***BlenderCAD***, which is actually an extension to the Blender vector graphics application that we discussed earlier. You can get the source code from ***http://projects.blender.org/projects/blendercad/*** and compile it.

A simple program we can suggest is FreeCAD. The maker of FreeCAD, Askoh puts it as follows:

"FreeCAD allows users to create and manipulate assemblies of parts. The parts are simple 3D solids, which can be connected by joints, constraints, contacts, motors, actuators, springs, dampers, forces, torques or gravity. The parts and connections define the structure, mechanism or machine of interest. Both open and closed 3D loops are permitted. 'freeCAD' performs full Multibody Dynamics analysis on the assembly to predict the motion according to Newton's Laws. Animation using the simulated data produces realistic dynamic behavior of the system.

Learning geometry and understanding how trigonometry relates to angles is all easier now, thanks to an application called Dr. Geo, which is from an organisation that calls itself OFSET (Organization for Free Software in Education and Training). So all you engineering students, machine drawing or engineering drawing need not now be beyond comprehension!

QCAD is another software worth mentioning. It works on two dimensions. It has a simple interface, is fast, and works in different formats. PythonCAD is a CAD software based on the Python script. It is available under GNU and is free. With the versatility that Python scripts offer, many other CAD software are being developed using Python. At the time of writing, we found many to be still in the alpha stage. We are sure that in the near future we will have several CAD utilities out there for all you open source designers in the making!

Some software are written for the Linux as well as the Windows platforms, so you can work on the same application on both OSes. This way you need not re-learn a software.

You can find a listing of CAD software at ***http://www.tech-edv.co.at/lunix/CADlinks.html*** .

There are a lot of software that are being written, and are in the beta stage. As Linux becomes more appealing to the home-based developer and learner, applications for professional use are also being written, with CAD being one of them.

5.7 Vi Editors

When you hear "editor" in the computing world, you will immediately think of Notepad or MS Word. If you have read thus far, you will be thinking of Kwrite or gedit or Open Office Writer. But these are not of much use to the programmer: he does not need too many fonts or colours or document editing tools. Remember, a program easily stretches into hundreds and even thousand of lines, so what he needs is ease of navigation. He needs to have commands to move through words and lines. That is where a vi editor comes in.

A vi editor is like a text editor made for someone dealing with lots of text. Vi was originally made from two other UNIX-based editors called ex and ed. Ex, especially, was very complicated. This lead the creator of vi, Bill Joy, to develop an easy-to-use editing tool. Vi today is a hot favourite of programmers for its ease of use.

So, why are you being told all this?

No, we don't expect everyone to be a programmer. But we can tell you that programming is fun! Programmers are as human as you are, and it's nice to know something about how a program is usually edited. This can also come in handy as you will learn how to edit config files to change the behaviour of Linux. If the file is large, you're much better off with vi.

At first glance, the vi screen appears cold and unforthcoming. But even though vi is awkward to use for the first few times, once you get used to it, it enables fast and simple editing. You cannot use vi properly before knowing at least a handful of commands. A key concept in vi that enables efficient navigation around the text is that you can combine an *action* (such as delete, copy to buffer, capitalise, etc.) with a *movement* (go to line 63, go to the next occurrence of "digit," go to the fifth occurrence of "K" in this line, etc.). You can set the action to be performed on all lines (or characters) between two points. The points are defined by the cursor position.

Vi can easily move around or within (or even between) files. With the array of commands at your disposal, your usage of the arrow key reduces significantly.

To just start using vi, try and create a new file and type something in just to get accustomed to it. Go to the command prompt and type in ***"vi digit"*** and press ***[Enter]***, where ***"digit"*** is the name of the file you are creating, provided there is no other file by the name ***"digit."*** If there is, the vi editor will simply open that file.

The vi editor is now open, and you can type in something. To exit, press ***[Esc]*** to return to the command prompt, and then type ***"ZZ"*** and press ***[Enter]*** (you can also type ***":wq"***) to save your work and exit. Now you know the most basic of any application—open and close. Let us now see some of the things you can do in between those operations.

To use commands in vi, just press ***[Esc]***, which will get you to the command mode. If you need to get back to text enter mode, just press ***[Insert]***. We said one of the best things of vi is its navigation: in the command mode, just press ***[E]*** to go to the end of the word, ***[K]*** to go to the previous line, and ***"$"*** to move to the end of line. There are several such commands to move around the text in the file:

[H]: Move cursor one character to left
[J]: move cursor one line down
[K]: move cursor one line up
[L]: move cursor one character to right
[W]: move cursor one word to right
[B]: move cursor one word to left
[0]: move cursor to beginning of line
$: move cursor to end of line
[N] + [G]: move cursor to line n
[Ctrl] + [F]: Scroll forward one screen
[Ctrl] + [B]: Scroll backward one screen
[I]: Insert to left of current cursor position (end with [Esc])

[A]: Append to right of current cursor position (end with [Esc])
[D] + [W]: Delete current word (end with [Esc])
[C] + [W]: Change current word (end with [Esc])
[R]: Change current character
˜: Change case (upper, lower) of current character
[D] + [D]: Delete current line
[D]: Delete portion of current line to right of the cursor
[X]: Delete current character
[M] + [A]: Mark current position

A lot more can be written about vi editors and the commands for vi, but we will stop here and let you experiment a little with the commands we have already mentioned. As you begin to edit certain config files, you will appreciate the power of vi!

After trying vi, you would want it on Windows too. Vim is a vi editor available for Windows. You can download it from ***http://www.vim.org/download.php*** .

EMACS

To be mentioned along with vi editors is emacs. While vi is purely an editor, emacs is a text editor and also an interpreter for the Lisp programming language. The name "Emacs' was originally chosen as an abbreviation of Editor MACroS. But today, vi editors are by far preferred to Emacs.

Multimedia And Gaming

Music is moonlight in the gloomy night of life.

— Jean Paul Richter

But what if your computer simply refuses to play music? This can actually happen on Linux! fear not—we're here to help—you'll find all your saved MP3s playing perfectly on your Linux machine.

In this chapter, we touch upon codecs and games, and also on emulating Windows on a Linux machine, so you can run those Windows-only programs as well!

6.1 Playing Music And Videos

There are so many distributions out there that it is difficult to say which one supports what format of music (MP3, RM, OGG). But suffice to say that support for MP3 is most important.

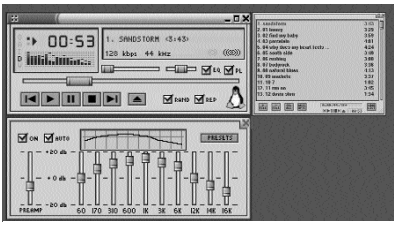
But unfortunately for us all, MP3 is a proprietary format owned by Fraunhofer IIS. Red Hat stopped MP3 support from version 8.0 onwards, as under the GPL, they could not include a proprietary software along with the distribution. Most freely distributed versions do not have MP3 support, and this can be an issue for many new users. Well, not if you have *Digit* with you! Here we will guide you towards setting up your Linux OS to play your songs and even videos. After you get acquainted with playback of songs, you can even try encoding (creating) songs or ripping audio CDs.

6.1.1 Play MP3s!

We were testing a distribution called Mandriva 2006 (Mandrake is now Mandriva), and were pleasantly surprised to see that it did not throw any tantrums when playing multimedia files. We did not even have to open the file through the player. The files were automatically recognised by the type, and merely clicking on it opened it in the default player—Kaffeine (bundled with KDE). The interface is nice and friendly; nothing seems left out in the functionality area. Impressed were we. As if that were not enough, we came across Xine player, which has some really cool features. A fast forward that looks and behaves like a VCR—fast forward without missing frames: most players fast forward in steps; Xine does it spanning across all frames. So if you can lay your hands on Mandriva 2006, multimedia is never a problem for you.

If you are using Red Hat or Fedora, life has just been kinder to you too. A versatile multimedia player called Xmms is available as a source-rpm format for download. You just need to download and compile it. Voila—your mp3s now work! Xmms can be downloaded from www.xmms.org/download.php .

The Debain packages can also be be found on You can find it at http://packages.debian.org/ or http://pdo.debian.net/stable/.


XMMS player

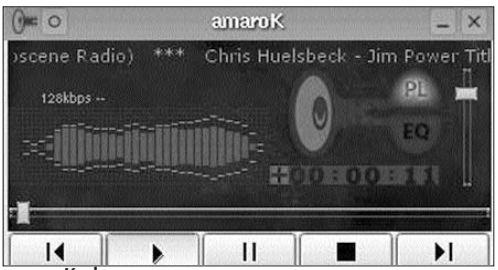
Xine player is another audio and video player that works on Red Hat or its derived distributions.


The versatile Xine player

For those of us (yes, we are also using Ubuntu for the moment) who are on a distribution other than Red Hat, SuSE or Fedora, you need to search a bit. Xmms is also available as a deb package, which you can install on your Debian or Debian-derived Linux.

There is also another player we found useful called Beep Media Player or BMP for short (don't confuse this with bitmap). Beep Media Player can be downloaded from http://bmpx.beep-media-player.org/site/Downloads. So, as you can see, it is only a matter of installing some applications in order to be able to play your multimedia files.

AmaroK is an excellent music player for KDE. It can do lots more than just play music. Using ID3 tags, AmaroK can even download the full lyrics for a



amaroK player

song. A 10-band equaliser, in-line ID3 tag editing, and many more features make it a must-have player. Though meant for KDE, AmaroK is also available for other Linux distros running GNOME as well. Visit http://amarok.kde.org/amarokwiki/index.php/Download for download instructions for your distribution.

6.1.2 VLC Media Player

VLC (earlier VideoLAN Client) is a highly portable multimedia player for various multimedia formats including MPEG-4, DivX, MP3, ogg, etc. It can also play DVDs, VCDs, and various streaming protocols. The name VideoLAN comes from the fact that it can also be used as a server to stream video in multicast (stream to many systems) IPv4 or IPv6 on a high-bandwidth network.

VLC Media Player can be downloaded from www.videolan.org/vlc/ . There are separate downloadable packages for different distributions.

6.2 Codecs

The word "codec" comes from "Coder-Decoder." This is a file or a set of files that are used to encode an audio or video data into a particular format for storage. Often, the encoding is performed to compress the data in order to save disk space. Once the audio or video data is stored in a format, it has to be decoded so that it can be played. Basically, the compression has to be interpreted in the correct way for the video or audio to be extracted. There are many different video formats and not all of them are open source. So those proprietary codecs may not be available on open source OSes.

The most popular codec today is the DivX codec, which produces excellent quality video output with smaller file sizes as compared to other formats. For a PC with a lot of videos, DivX Player is a must-have. At the time of writing, the Linux page on the DivX Web site was down. Do check http://www.divx.com/divx/linux/ sometime for download instructions. This apart, you may need

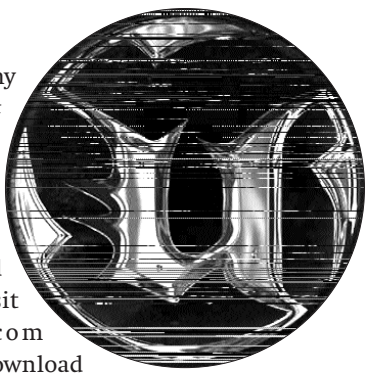
some other codec to play a video. Generally, the Web site you download a video from will tell you what codec you need, and may even have a link to download it. The best way would be to Google for it and download it.

6.3 Games On Linux

It's surprising that for a long time, there were hardly any strategy or first person shooter games for Linux, except the ones from id Software. Yes, we are talking about the almost legendary *Doom* and *Quake* series. These games are playable on Linux without using any cross-platform applications (discussed in the next section). Here we are not speaking of the small games that come bundled with the distributions. id Software choose OpenGL to implement their games instead of DirectX. The source code for *Quake* is available under the GPL. This means anyone can make changes and use it. And hence, a special place for id software in our book.

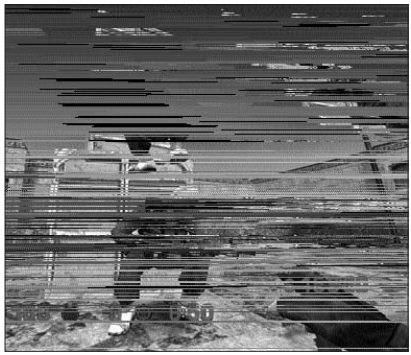
The *Doom 3* demo can be downloaded from www.doom3.com .

For the *Quake III: Arena* demo, perhaps the most famous of the *Quake* series, point your browser to ftp://ftp.idsoftware.com/idstuff/quake3/linux/linuxq3ademo-1.11-6.x86.gz.sh. For other versions of *Doom* and *Quake*, visit www.idsoftware.com.



Another game we find worthy of mention is *Unreal Tournament 2004*, which is also available for Linux. The story line is yet again a good versus bad and save-the-world idea. This game features impressive graphics and progressive gameplay. Visit www.unrealtournament.com/ut2004/ downloads.php to download and register it.

6.3.1 Counter Strike



If you who want something more realistic than aliens but still have a lot of shooting and dodging to do, *Counter Strike* might be the game for you. *Counter Strike* revolves around terrorism: you are part of an elite counter-terrorism strike force, and are engaged in various encounters. There's an almost-scary realism in the storyline and gameplay.

Go to www.counter-strike.net/download.html and find the installer for your OS. Follow the installation instructions.

6.4 Emulating A Windows Environment To Run Windows Applications

Here, emulation simply means making Linux behave like Windows to certain programs that only work on Windows. By this, the programs are more or less fooled into believing that they are working under Windows.

Most games are written to be run on Windows. For practical and economic considerations, game makers cannot rewrite every game for Linux. Hence the next available option is to make Linux act like Windows to the games and other Windows programs. This is what emulators aim to do. We will in detail look at a popular emulator for games called Cedega.

6.4.1 Cedega

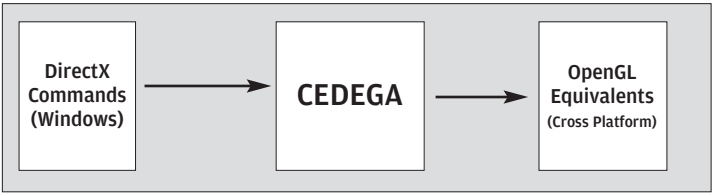
We have seen that there are numerous applications that allow you to use Linux and not miss what you are probably already used to. Won't it be nice if there was a way in which you could have your

Need for Speed or *FIFA* on Linux? Then the equation would really be complete, right? This has already been thought of, and out of several attempts, a Linux application called Cedega has emerged—a blessing indeed for all game fans using Linux.

Cedega, made by a Canadian company called TransGaming, is a Linux portability product that allows Windows games to run on Linux seamlessly and transparently. (A portability product is an application or software that acts as an interface between two mis-matching products and makes them work with each other). Cedega simply fools the game into thinking that it is running on Windows. You can simply insert your Windows game CD, install and play the game just as you would on a Windows system. TransGaming claims that Cedega can run hundreds of top-tier titles such as *BattleField 2*, *World of WarCraft*, *Dungeon Siege 2*, *Madden 2006*, *Half-life 2*, *City of Villains*, *WarCraft III* and *Star Wars Galaxies*, and more.

How Cedega Works

Cedega loads a Windows program into memory on a Linux system and links it to a Linux equivalent of the Win32 APIs. Windows games use several multimedia APIs such as Direct3D, DirectInput, and DirectSound, which are part of Microsoft's DirectX. Cedega maps them to (or converts them to the equivalent of) Linux equivalents such as OpenGL, X11, and the OSS or ALSA sound APIs.



Thus, DirectX commands are converted to the OpenGL equivalents. Remember, DirectX API sets are used by virtually every game maker.

Beating Microsoft At Their Own Game?

Cedega provides a simple interface to manage the installed games, along with options that can be selected for each title. Unlike how games are installed on a Windows system, Cedega installs each game in a separate environment. This ensures that one game can never interfere with another. To remove a game, simply delete the game environment without affecting other installed games. A feature of Cedega is that unlike Windows, it allows the user to specify that a game should run in its own window, instead of taking over the whole screen. Handy if you are playing a long-drawn battle in multiplayer game and you suddenly want to check your e-mail or even look for strategy tips—with Cedega, doing so is just a click away!

How To Get It

Through this book, you would have come to believe that some of the best things in the world are free. But then there are some things that are good and cost some money. Throughout this book we have made efforts to keep to free software as much as possible, but we thought we had to mention Cedega because it represents what can be achieved when developers simply put their minds to it. Though it is paid software, it shows that applications need not be limited to the platform they are originally built for—especially fun applications such as games.

Cedega is available on a subscription basis through TransGaming's Web site. Subscribers get access to the latest Cedega releases including regular hotfixes and updates, access to both e-mail and Web-based support, posting access to TransGaming's user forum, and the right to vote on what TransGaming's Cedega team should work on for future releases.

Thankfully, there is a fully functional time limited demo version available. The Time Demo works for 14 days, after which you need to subscribe. You can download it from www.transgaming.com/products_linux.php

After running the downloaded script, you will be prompted to install via a nice friendly GUI (for a change!). You have to register for the trial period by supplying a valid e-mail id.

Playing games through Cedega

Once you download and install Cedega, you will have to complete an online registration process in order to be able to use the demo version. Activation will be done once you enter a registration number that will be sent to the e-mail address you provided. Cedega opens once the registration is complete. The interface is minimal and can fool you into thinking less about the software's capabilities. Click on the Install button, and another window opens up. Now, guide the software to the location of the EXE installer of the game you wish to play. Mind you, Cedega is guaranteed to run only those games that are currently supported. Other games *might* also be playable, but performance is not assured. Give any description of the game in the Program Title box.

If the game to be installed is on a CD (unless you have copied it to the hard disk) you first have to mount the CD by clicking on the Mount button. Then guide it to the CD-ROM location where the setup.exe of the game is located—the name of the setup file can vary according to the game. Once that is done, click on Continue, and an installer pops up just like in Windows. Go ahead just like you would in Windows and wait while the game gets installed.

You will then see it in the pane to the left. Select the game and click Play. The game you selected will appear in a separate window, and voila—you just managed to run a Windows application on Linux!

Playing a game in a window using Cedega

Do bear in mind that the minimum hardware requirements that you must have in order to get smooth performance will be higher than on Windows, due to the extra processes involved in converting DirectX commands to OpenGL.

You can also try and install some Windows applications though Cedega. Not all of them work, however: while some of them may not even install properly, some will not work properly. So don't expect the world from Cedega. However, full marks to them for the effort they have put in. What we can safely say is that the day is not far off when programs written for one platform will be able to work on another platform with ease.

6.4.2 Wine

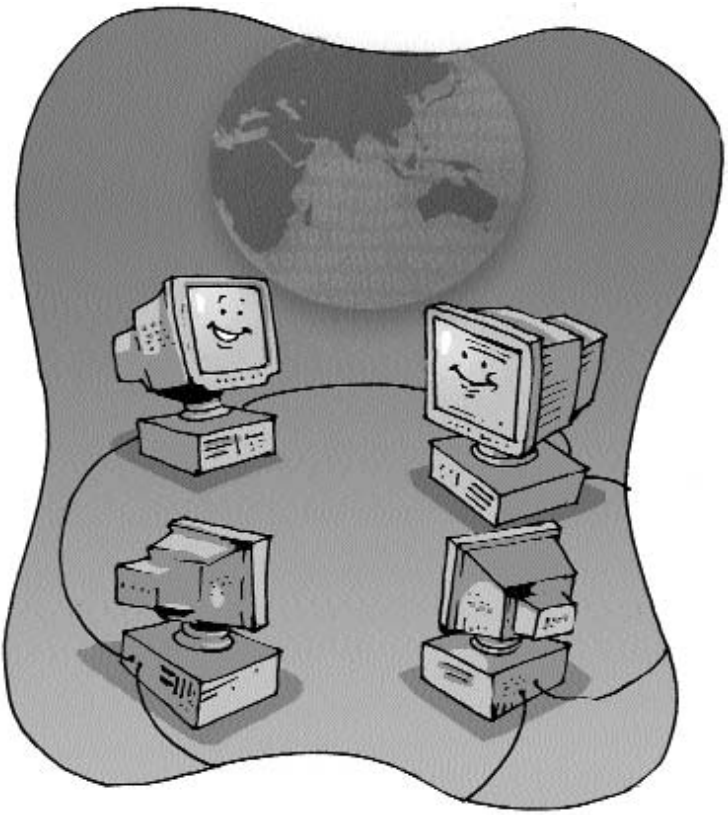
No, we're not talking about the drink! Wine is an application that creates a Windows-like environment for Windows-based software to run on. WineHQ, the makers of Wine, claim that their software can make several Windows applications run—including games! Paint Shop Pro, FrontPage and Microsoft Money are just some of the applications that can be made to work on Linux using Wine. Like Cedega, users of Wine can vote about what software that they would want Wine to support in future.

The current version of Wine is 0.9.8, and it can be downloaded from WineHQ's Web site, www.winehq.org/site/download

The good part is that there are several Linux versions supported. The bad part is that the software is not perfect as yet. As WineHQ themselves say "Wine is still under development, and it is not yet suitable for general use. Nevertheless, many people find it useful in running a growing number of Windows programs."

But don't let that deter you from experimenting with it! For all you know, you just might get your favourite Windows program running on Linux as well!

Networking



Linux is widely used as a server OS. One of the reasons for its popularity as a networking OS is its depth and strength in the basics of networking. For home users, though, installing an internal modem has always been difficult. In this chapter, we shall see how to install such devices and configure a network and Internet connection. Once you have these up and running, installing IMs, P2P and e-mail clients becomes a lot easier.

7.1 Networking Over TCP/IP

There has been a wide-scale implementation of the so-called broadband setup in India. Many users find broadband service affordable, and have switched from dial-up to Ethernet-based "broadband" connections. This kind of setup involves a LAN (Local Area Network) connection that is routed to a local gateway, which acts as your proxy to the Internet. Here, you need to know what IP addresses, subnet masks, gateways and DNS or nameservers are.

7.1.1 Fundamentals of TCP/IP

Networks are essentially of two types: public and private networks. The latter is used in local networks such as homes, offices and cyber cafes. This class of IP cannot be directly used to connect to the Internet. This is where your ISP—the Internet Service Provider—comes in. Their Domain Name Server (DNS) assigns a public IP to your machine, using which you surf the Internet. Public IPs have to be registered with the Network Information Centre (NIC) to avoid IP address conflicts. An IP is your address on the Internet, using which data packets can be sent to and from your computer.

A private IP is accessible only within a LAN. For this purpose, the NIC has reserved certain address ranges that will never be registered publicly. In short, there can be more than one identical private IP in different networks, but only within LANs. The Web sites that you regularly visit are actually public IPs. For example, the name www.yahoo.com is resolved to an IP address; this address is public—it is accessible by anyone in the world. The names are assigned for the sake of simplicity because it's not practical to remember the address of such sites.

Every IP address is represented as a "dotted quad" (in case of IPv4) as x.x.x.x. This can be split into two parts: the network portion and the host portion. The network portion is the same for all nodes (computers or printers) connected to a LAN. The host portion distinguishes each node from the other. The subnet mask

plays an important role in identifying the network and host portion of an IP address. We shall come to the subnet mask gradually as we proceed with understanding the classes of IP addresses.

7.1.2 IP Classes

IP addresses are classified under three classes, namely Class A, Class B and Class C.

Class A ranges from 0.0.0.0 to 126.255.255.(1 to 254). The subnet mask for this class is 255.0.0.0. This will mask the first number in your IP address as the network address and the remaining becomes your host address. Hence the total number of networks is 127, and the number of hosts per network is 256 x 256 x (256-2). In this entire range, 10.x.x.x is reserved as private IPs, and 127.0.0.0 is reserved as loopback localhost.

Class B IP address ranges from 128.0.0.0 to 191.255.255.(1 to 254). The subnet mask for this class is 255.255.0.0. Hence the first two bytes of an IP address is masked as the network portion. IP address ranging from 172.(16 to 32).x.x are reserved as the private range.

Class C has the minimum number of hosts amongst all the classes. Class D and E do not fall under the normal users category—they are reserved for Multicast and Military/R&D respectively. Class C continues after the Class B from 192.0.0.0 to 223.255.255.(1 to 254). The subnet mask is 255.255.255.0. The private IP range falls within 192.168.0.x.

It is not obligatory to use the prescribed subnet mask. For instance, some ISPs use Class A IPs, say 10.15.10.28, with a 255.255.255.0 subnet. This is possible, but it restricts a host to network of 10.15.10.x, whereas it could have actually belonged to a wider network of 10.x.x.x had the subnet been 255.0.0.0. In some Linux systems, you will notice an IP address given as 192.168.0.12/24. This is a combination of an IP and a subnet mask. 255.0.0.0 is denoted as /8, while 255.255.0.0 and 255.255.255.0 are denoted as /16 and /24 respectively.

7.1.3 What happens when I dial my ISP?

Internet connectivity through a dial-up service works in a different manner. First of all, you don't need an Ethernet card, so you aren't assigned a private IP. Your PC becomes an isolated node until you dial in to an ISP.

A modem and a clear (without disturbance), functioning telephone line are required to establish a connecting via dial-up. The process is very simple: you have an account with your ISP. Using a dialler as a client, you can dial in to your ISP. Modem-to-modem interaction takes place, and your username and password are verified by the server at the ISP. The connection is first authenticated, and you are allowed to enter the ISP's server. Thereafter, your account is given a random (public) IP address so you can surf the Internet. All this happens in just a few seconds. Once you are connected to the Internet via dial-up, open the Konsole or gnome-terminal window and check the IP address you've been assigned by using the command "ifconfig".

7.2 Configuring Internet Access

Setting up a Linux machine to connect to the Internet is not a difficult task. And once you do, we bet you'd like to play around with additional settings such as spoofing the MAC address. Connection to the Internet can be achieved in different ways; here we shall see how to establish a connection through Ethernet and through dial-up.

7.2.1 Through Ethernet

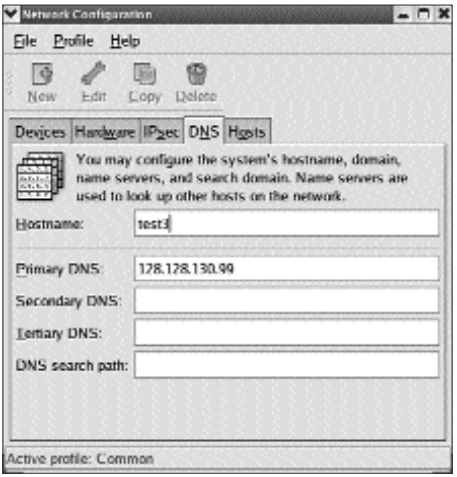
Most broadband services in India have a standard setup that involves networking of PCs to a proxy server through which they get Internet access. The ISP gives every user an IP address, Gateway address and DNS address. Once you have this information, all you need to do is set up the network card.

During installation, one of the steps involves setting up of the network adapter. It is set by default to run at boot as a DHCP client.

You can change the setting at the installation step itself, or after the installation is complete.

The setting for network adapters is most likely located under the system menu. It is difficult to pin-point the menu under which the network setting is located because this not only varies from one distribution to another, it also changes with desktop environments (KDE or GNOME). Any method used to configure the network card will edit a set of files that is common to all Linux systems. So if you are brave enough to use root privileges and edit system files, go ahead. For beginners, we shall take one example of a GNOME desktop and one for a KDE desktop. This way, you'll get a picture of how to go about looking for this setting. We shall then take a look at the files that are eventually edited. Knowing this will help you set up a network adapter on any machine. The Linux kernel supports almost all network cards, and so it's unlikely that you will have any installation troubles.

However, before you proceed with the configuration of your network card, you must disable it if it is already running. Open a terminal window and use the command "ifconfig" to know if your Ethernet card is running. It will also tell you if it is eth0, eth1, etc. Let's assume it is eth0; then, shut down the network services, use the command "ifconfig eth0 down" or "ifdown eth0". You can now proceed with the configuration of the network card.



Use the Network Configuration tool to enter the IP address and other network information

KDE

Click on the K Menu and go to System Settings and then click on Network—we're using KDE on Fedora Core 3. The network configuration dialog box opens. The configuration box will look a little different to those who are used to Windows, but only because it is a lot simpler. For better understanding, the settings are classified under tabs such as Devices, Hardware, IPsec (security), DNS and Hosts. Under the Devices tab, you will see your network card as "eth0" or "eth1" in case of multiple network cards. Choose the card (as in device) you want to configure for an Internet connection and then click on the Edit button. Another dialog box opens, displaying a variety of options such as "DHCP", "Activate on boot", "Nickname", and so on. Choose "Activate on boot" and also select "Statically set IP address"; now, the row for IP address, Subnet mask and Gateway becomes available.

Fill in the respective fields with the addresses your ISP gave you. Click on OK to get back to our previous dialog box, and choose the DNS tab. Give a Host Name, which in Windows lingo means "Computer Name." You may have already provided a Host Name during the installation process; here is your chance to change it if you didn't know what it meant. More importantly, add the DNS (also termed the Nameserver) address specified by your ISP. To complete the setting, click on File > Save, and then click on the "Activate" button to enable the network with the new settings.

GNOME

The GNOME menu normally has two buttons, namely "Applications" and "Actions". In some distributions such as Ubuntu, you will see a "System" button, under which most of the system configurations are categorised. In such cases you will find "Network" under System > Administration. On our Fedora machine, we located the Network option by going to Applications > System Settings. The same Network Configuration dialog box opens and the same settings need to be applied.

Connecting to the Internet

You can use your browser immediately to surf the Internet if your connection doesn't need any sort of authentication. Such connections are common in offices, but for home users, ISPs set a username and password for your account. In most cases, the ISP provides its users a connection client capable of running on different operating systems; it facilitates connection and authentication to their server. Under such circumstances, demand a Linux version of the client from your ISP.

7.2.2 Dial-up Internet

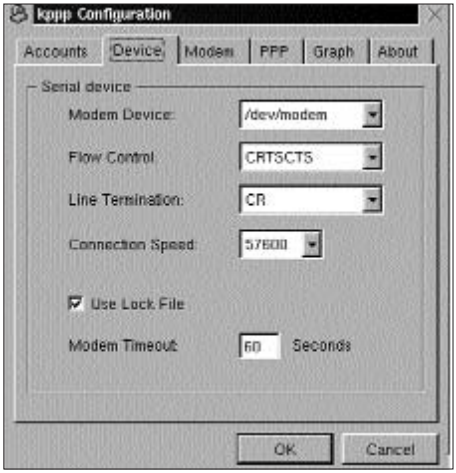
Various distributions have scripted a Wizard to help you install a modem and also configure it to dial up to your ISP. Such Wizards will normally be available under the "System Tools" or a similar application group, depending on the distribution. We can set up a dial-up connection through the command line too, which is by and large common to all Linux distribution (unless they have deliberately excluded it), but this can get very tricky. The easiest option is to locate and use the Wizard. We shall consider KPPP, a KDE application to configure a dial-up connection. This application is also available under the GNOME desktop environment.

Some distributions such as SuSE allow any user to run the application, but Redhat demands authentication for using such applications. It is better to switch to root user or log in as root when setting up the network.

You need to understand that many internal PCI modems were designed specifically for the Windows operating system, and so were termed "win modems." Certain newer ones work perfectly with Linux systems, but this problem is still not sorted out. If you happen to possess a win modem, note down your modem details such as the model number and manufacturer. The details can be accessed by using the command "lspci |grep -i modem".

The solution is dependant on your model of win modem. There are a lot of Web pages that can guide you towards overcoming the

difficulty. However, an external modem does not have such issues, and works perfectly with any OS.



KPPP is commonly used to configure a modem and modem properties—it works on Gnome too

Access KPPP either through the program menu, or open a terminal window and type "kppp &" and hit [Enter]. This will open the KPPP window. Click on the Configure button to set up the modem and account settings. Follow the steps listed below to get your dial-up connection working:

1 Click on the "configure" or "setup" button on the KPPP dialog box to check if the modem is detected and whether it is interacting with the OS you have running. Click on the Device tab, and under the pull-down menu against "Modem:", choose the COM port to which the modem is installed. If the OS has already made a symlink to your device, it may appear as "/dev/modem". After choosing the COM port, click on the Modem tab. Click on the "Query modem" button, and if you get an error message, choose another COM port. Repeat until you get a reply from the modem.

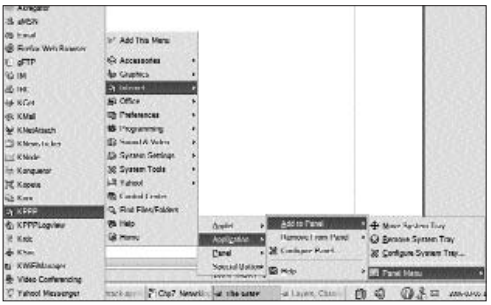
2 Once you have the modem responding to AT commands (this is a set of commands used to interact with a modem), you need to set up a user account. This is achieved by choosing the Accounts tab and then clicking on the New button. Most of the entries to be made in the New Account dialog box are pretty straightforward. Leave the "Authentication" method to "PAP", and add the following line under the field "Execute the program upon connect": "route add default ppp0".

3 Click OK to finish the setting. If your ISP has given you a DNS address, add that under the DNS tab and then click OK. You will now see the main KPPP window with a connect button and all the information you entered. Click on Connect to dial in to the ISP. If the connection is authenticated, you'll get a message verifying your connection. Syslog writes error messages to /var/log/messages. If you see a message in /var/log/messages that states:

The remote system (ppp0) is required to authenticate itself but I couldn't find any suitable secret (password) for it to use to do so

then you need to configure ppp to use the "noauth" switch. Click on Setup and then, in the Account Setup box, highlight the account name you want to edit. Now go to Edit > Arguments. Type in "noauth" in the text field under Arguments, and click Add to finish reconfiguring.

4 If you are using KDE, you can add KPPP to the Kpanel next to the clock on the desktop either by right-clicking on the panel to add the item to the menu, or just by choosing "Dock into panel on connect" by clicking on the Setup button.



Use Panel Menu properties to add KPPP to the Panel

7.2.3 Do it the command line way

As we said earlier, many Linux distros have different methods of setting up a network. Where do all the entries go? Which files are edited? If you are a Windows user, we should tell you that the process is the same in Windows. When you add TCP/IP settings in Network Connections, it actually edits a file called Hosts. Very few people know this! So let's call this a geek's way of setting up a network.

The files that control the network setup are:
/etc/hosts
/etc/sysconfig/ network
/etc/sysconfig/ network-scripts/ifcfg-eth0

The last file, ifcfg-eth0, contains all the details such as the IP address, Gateway address and even the MAC address of your ethernet card. The other two may not necessarily be edited for address changes. The "/etc/hosts" file is used for name-to-IP translation, while the "/etc/sysconfig/network" file contains the hostname and status of networking—"yes" or "no".

Before you start editing any of these files, we advise you to shut down the network service. This can be done by changing or re-logging in as root and then typing "#cd /etc/rc.d/init.d/" followed by "#network stop".

Let's take a look at the files mentioned above. We entered a certain address for our Ethernet card, and now you can see how it was all entered into these files.

In the etc/hosts file, a user can add the IP address and the name of the machine. For example, if there is a machine with IP address 10.10.124.112 and the name of that computer or host is "tux", then you can add the line:

"10.10.124.112 tux"

Now if you ping tux, /etc/hosts will resolve the name to its IP address and send data bytes to the address and not the name—the name behaves as an alias. You will receive a reply if the machine named tux is active on the network—that is, if the machine is on and it is present on the network.

Well, let's not forget that we had disabled the service. So, to restart it, go back to the directory /etc/rc.d/init.d/ and issue the command "#network start".

/etc/sysconfig/network-scripts/ifcfg-eth0

```
DEVICE=eth0                    -The primary Ethernet card
BOOTPROTO=none
BROADCAST=10.10.124.255        -Broadcast address
HWADDR=00:0B:2B:0C:49:9A       -MAC address of your Ethernet card
IPADDR=10.10.124.28            -Your IP address
NETMASK=255.255.255.0          -Subnet mask
NETWORK=10.10.124.0            -Network your machine is in
ONBOOT=no                      -Start Network on boot
TYPE=Ethernet                  -Type of card
USERCTL=no
PEERDNS=yes
GATEWAY=10.10.124.1            -Gateway address
IPV6INIT=no                    -Initialise IP version 6 (always 'no')
```

/etc/sysconfig/network

```
NETWORKING=yes    -Enable Networking (does not mean start on boot)
HOSTNAME=aryan    -Name of your machine (Computer Name)
```

/etc/hosts

```
# Do not remove the following line, or various programs
# that require network functionality will fail.
127.0.0.1       aryan   localhost.localdomain   localhost
```

7.3 Browsers

Linux comes with default browsers. Yes, we mean more than one browser. Most distributions normally have a main, default browser and a browser that is a part of the desktop environment—KDE and GNOME. Konqueror doubles up as the file explorer as well as an Internet browser on KDE, whereas Nautilus, a file explorer, doubles up as a browser for the GNOME desktop.

The main browser for any desktop environment, and almost all distributions, is Mozilla. Newer distributions use Firefox, whereas older ones running kernel 2.4.x have the older Mozilla browser. Apart from these, other browsers such as Opera, Galeon (for

GNOME), Epiphany (for GNOME), and many more free, open source browsers available to download and use. We shall take a look at Firefox and Opera.

7.3.1 Mozilla Firefox

Firefox has gained a lot of popularity, praise, fans and huge support from developers of the open source community. It is not only the primary browser for Linux users, but it has gained a lot of ground in the Windows domain as well. Its success on the Windows platform can be attributed to its focus as a challenger to IE. It is fast, secure, and compatible with either platform.

Mozilla updates Firefox frequently. To download the latest version, go to www.mozilla.com/firefox; the site detects the core of the OS—whether Linux or Windows—and provides the download link accordingly. This means that you will get the download link for the Linux version of the browser if you're using a Linux machine. Or click the link under the download button that says "other systems and languages." It will redirect you to www.mozilla.com/firefox/all.html.

Download the 8-odd MB compressed archive. Once the download is complete, extract the files to a folder in /usr. In our example we shall extract the files to /usr/more-apps. To do this, use the following commands:

```
Make a directory more-apps under /usr
$cd /usr
$sudo mkdir more-apps
Extracting the files to the new directory
$su root
#tar -C /usr/more-apps -zxvf firefox-1.5.0.1.tar.gz
```

You will not find any installer packages like RPM or DEB in the extracted folder; instead, you will find a few source object files (*.so) and some folders. Well, actually there *is* no installer. The sad part is that you won't find much assistance on how to use it from

their Web site. But at the end of our explanation you will know how easy it is! Moving on to the next step—creating symbolic links to your current plug-ins. It will better to switch to root user than using sudo all the time. Sudo will be necessary for Ubuntu users, but they can choose to open a root terminal instead of the normal GNOME-terminal.

```
Creating links to current plugins
cd /usr/more-apps/firefox/plugins
ln -s /usr/lib/mozilla-1.7.3/plugins/* . Or
ln -s /usr/lib/firefox-0.10.1/plugins/* .
```

Whenever you click on the Firefox button, or type in "firefox" at the command line, the old version of the browser pops up. So, to ensure that only the new Firefox opens and not the earlier version, you need to change the symbolic link that relates to the action of opening the browser. Use the "which" command to figure out the path to Firefox. Change your directory to the directory where "firefox" is located. Rename the old "firefox" file from "firefox" to "firefox.old".

```
Setting firefox to run the new Firefox 1.5
#which firefox
output: /usr/bin/firefox
#cd /usr/bin
#mv firefox firefox.old
#ln -s /usr/more-apps/firefox/firefox
```

Click on the button that opens Firefox, or rather, just open from the command line (as you're still in it) by typing in "firefox" and hitting [Enter]. In the browser that opens, go to Help > About Mozilla Firefox. This should show you "Firefox version 1.5.0.1".

You can add extensions to your browser from Mozilla's site (www.mozilla.com/extensions/). Browse through various extensions available on their site and choose the ones you like. Firefox

automatically downloads and installs them. The installation of extensions are finalised when you restart the browser.

7.3.2 Opera

Opera is as efficient as Firefox and very easy to install and manage. It is popular on the Windows platform for its features and ease of use. If Firefox users have to download extensions to enhance their browser, Opera has them all inbuilt! The user interface of Opera is far superior to that of Firefox, and the best feature about it is that it restores your previous surfing session in case of a crash. The drawback of Opera is that it fails to reproduce certain Web pages the way they are meant to be displayed.

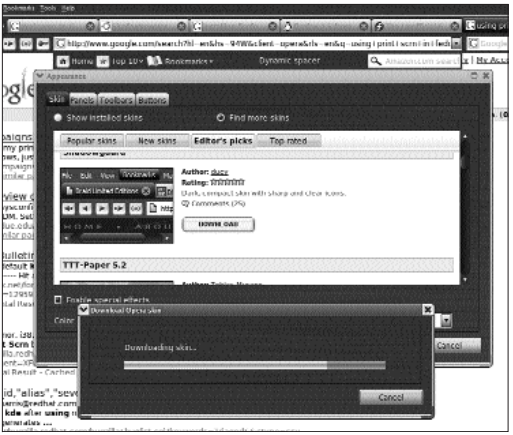
Download the latest version from http://opera.com/download/

Choose the right package from the list available in the drop-down box on their Web site, and then choose a mirror for download. Opera is compiled as an installer package for various distributions of Linux. To install, just run the command for installation, which is "dpkg -I" for debian users, and "rpm -ivh" for distros following the Redhat Package Manager. For example, type in "#rmp -ivh opera-8.51-20051114.5-shared-qt.i386-en.rpm" and hit [Enter].

Choose the Opera installer as per the distribution you use

The installation process completes in just a few seconds. Most Linux beginners get stuck at this point, not knowing how to proceed because the installer doesn't create any shortcuts on desktop or the program menu. That is because one more step remains to finalise the installation. Type in "opera &" at the shell prompt and hit [Enter]. A window will pop

up stating "detected KDE running...". After this is done, you will find shortcuts for Opera in the program menu.



Downloading skins for Opera is the same as for the Windows version, and this applies to Firefox, too

Adding extensions, add-ons, themes and skins is a piece of cake after you have the browser up and running. We shall consider an instance of Opera to demonstrate changing the skin of the browser.

Open Opera. Click on Tools > Appearance; on the window that appears, choose the Skin tab and then click on "Find more skins". Note that you have to be connected to the Internet. This will list all the skins available for Opera. Choose and download the one you like. After the download is complete, Opera changes to the new skin and asks you if you would like to keep it. Click on "yes" to have it saved, otherwise you will have to download it again.

7.4 E-mail Clients

Many e-mail clients such as Mozilla's Thunderbird are available for free download. But that is not required if your distribution has packed Evolution with it. Evolution is Novell's e-mail client. It is now under development with Novell. It integrates e-mail, calendar, tasks and contact manager in a single application. Evolution is fast, uses very little resources, and has features that are comparable to those of Microsoft Outlook. Other than IMAP, POP, SMTP and Authenticated SMTP protocols, it also supports Microsoft Exchange

2000 and 2003. But its support for Novell GroupWise is still in the beta stage.

Lightweight Directory Access Protocol (LDAP) is supported, using which any employee can have access to the employee address database and even share contact information using vCard message attachments.

A wizard guides a user through the entire process of adding a new account. The process is simple and it self-initiates when Evolution is run for the first time on your machine. The Windows version of this client is also under development. The source is open and any developer is free to join and help Novell in its roadmap to developing a comprehensive groupware client.

7.5 Communication And File-Sharing Software

Enhanced by voice and Web cam facilities to come close to video conferencing, IM is here to stay. So is P2P. This section tells you about different communications options you have on Linux.

7.5.1 The Instant Messengers

GAIM

This is an IM client that can connect you to Yahoo!, ICQ, MSN, AOL, Google and other accounts in a single window. Meaning, through a single application, you will be able to connect to the various accounts that you have. GAIM also features an exhaustive collection of emoticons, including the so-called hidden ones from Yahoo! On Windows, GAIM was extremely light on resources as compared to the memory hog Yahoo! Messenger. With this introduction, let us now tell you that GAIM was originally written for Linux systems. All newer versions of Linux flavours bundle GAIM Messenger. The current version of GAIM is 1.5. If you have an older version, you can get the latest version from GAIM's Web site (http://gaim.sourceforge.net/downloads.php) and install it. The source code is also available for the interested ones, who can see how it works from the inside and perhaps change it!

GAIM was first written in 1999 by an Auburn University student, Mark Spencer. In his own words, "I wrote GAIM as an exercise in how to write my own instant messenger," then he made the software available to other Linux users. But it has not been smooth sailing for GAIM, after AOL raised an objection to GAIM, stating that the latter had "copied" some trademarks. In the true spirit of the open source community, GAIM users donated money with which the GAIM developers sought the services of law students to arrive at a negotiation with AOL.

With GAIM around, frankly, you don't need any other messenger. But then, Linux is not about any one product. So we shall briefly tell you about other IM clients that work on Linux.

AIM

AOL Instant Messenger is one of the oldest messenger services till date. AOL does not update the Linux version of the client often. Nevertheless, if you want to try AIM on Linux, you can visit http://www.aim.com/ and click on the Download button, and choose "Linux".

aMSN

No, that's not a typo—alvaroMSN, or aMSN, is what we are talking about. It is a free, open source MSN Messenger clone featuring display images, creating custom emoticons, Webcam support, and several usable features. And if you are learning foreign languages, aMSN supports around 40 languages. It also has a tabbed window feature where every chat window appears as a tab instead of opening a separate window. You can add skins and change the look of aMSN. The current version available is 0.95 with the release date December 25, 2005. You can download it from http://amsn.sourceforge.net/download.php

Yahoo!

Yahoo! Messenger is available for Linux too. Just go to http://messenger.yahoo.com/ and near the bottom of the page, look for small-sized text that says "other versions". Click on "Unix". There

you will find installers for different flavours and versions of Linux. The source code is also available for download.

Google Talk
Unfortunately, Google Talk is not available for Linux right now, but Google officially says that other IM clients can be used to chat with the Google Talk contacts with a Google Talk id. At least Google says it officially. And that's a good thing. You can refer to http://www.google.com/talk/otherclients.html, which will guide you towards setting up other chat clients to access Google Talk accounts.

Gmail or Google Chat is a Web browser-based IM protocol. When you log in to your Gmail account, you will see, towards the left corner of your browser window, a list of your Gmail contacts with whom you can chat by simply clicking on their name. This is a really handy feature, be it in Linux or Windows! If your IM clients are not working for any reason, you can simply sign in to your e-mail account on your browser and start chatting!

7.5.2 Skype

Skype is an IM and free PC-to-PC calling application. You can also call landlines or mobile phones, for a fee though. What is special is that anyone can call you on a number that Skype will assign to you, and it acts just like a real phone number—you can "pick" the call on your computer no matter where you have logged in to your account from. Skype can be downloaded from www.skype.com/download/skype/linux/

7.5.3 Internet Relay Chat (IRC)

You've probably come across "IRC" somewhere. If we proceed to explain the concept of IRC, you may be left wondering: "So what is the difference between this IRC thingy and the IM that I use?" So, we'll just say this: IRC was a very early way of Instant Messaging, but it was not focused as a peer-to-peer (one individual to another) application, but it was aimed at group chatting. There were specific IRC chat rooms or servers, where an IRC user could just join

in and chat with an entire group. Instant Messaging is a relatively new idea wherein a sense of personalisation was created. IRC, however, was for groups of people with a common interest. There are several IRC servers running. But with the advent of e-groups (for example, Yahoo! Groups) the idea of IRC seems implemented in a more organised fashion.

Some IRC clients that can be used on Linux are ChatZilla (by Mozilla), Xchat, BitchX, etc. You can find a more detailed listing with reviews and download links on http://www.ircreviews.org/clients/platforms-unix.html

Some Linux distros have a bundled GUI IRC client that you can use. If you are new to IRC, don't just jump into any room! First find out if the group matches your interest, then watch it for a couple of days and then introduce yourself before shooting away text.

7.5.4 P2P

Peer to Peer, often called P2P applications, are software through which users can share personal files with other users on the Internet. The only requirement is that both users must have the same P2P software and, of course, an Internet connection. There are several P2P clients out there, but none can claim the success of Napster before it was pulled down. Today, one of the most popular P2P networks is BitTorrent.

BitTorrent works though user groups who track the progress of a file being shared. Rules can be set to ensure that every member shares some amount of data. There are several software that can be used to download and shares files through BitTorrent. The official site of BitTorrent is www.bittorrent.com, from where you can download the BitTorrent client. It is different from other P2P clients in the way users can look for content. You cannot simply browse another user's computer for files to share. A user who wishes to share his files has to make what is called a torrent file that contains details about the file. This torrent file has to shared with the people who want to download the file(s), either by mail

or through a tracker (something like a forum). This file has to be "opened" via a client software at the downloader's end, after which the actual download of the desired file begins.

7.5.5 DC++

Direct Connect Plus Plus is another widely-used P2P program. DC++ works by way of "hubs," which are groups you can join and share and upload files. Unfortunately, DC++ does not have a ready installer for Linux. You will have to download the source code and compile it yourself. It is available at http://dcplusplus.sourceforge.net/

Linux Distributions

There are so many distributions (or distros, as they are commonly called) out there that it can get confusing, to say the least, to choose the right one for yourself. In this chapter, we aim to give you a brief of some of the major distros available today. Once you know the basics of installation and the use of the command prompt, you can explore distributions other than your own.

8.1 A Problem Of Plenty?

Sometimes, having too much to choose from can make life difficult. You will always have someone who will come up to you and say "Hey, why did you install Fedora? You should have installed SuSE". To which someone else will probably retort, "Neither Fedora nor SUSE—you must install Mandriva!" This happens. Linux fans can get very possessive about their favourites, so you would want to take opinions with a pinch of salt and do some research yourself.

Most Linux distros give you the option of installing GNOME and KDE as desktop environments, and you can switch between them anytime you like. So no hassles there. Applications-wise, we can rate them as more or less the same. The difference, however, can come with hardware compatibility. Sound cards are always an issue for OSes, and Linux is no exception. Some distros have problems with some hardware, so it pays to know if all your hardware is compatible with the Linux distro you are about to get. All you have to do is go to the Web site of the Linux distributor (just Google around) and look for a compatibility chart or table. See if your motherboard, video card, sound card and add-on devices are supported—or worse, if they have a problem. Most of the commonly-used current hardware is supported in the newest versions, so that shouldn't be a problem. However, it pays to make sure!

8.1.1 Red Hat / Fedora

In 2003, Red Hat introduced a desktop developer version of their Linux called Fedora Core and patented the name "Fedora." Red Hat announced that their Fedora Project was aimed to be a developer as well as a newbie oriented distribution. This was targeted mainly at the home user, in other words, for non-commercial purposes. Red Hat had to do this because its Red Hat Linux (commonly shortened to RHL) was made available only for a fee, the name being changed to Red Hat Enterprise. Red Hat claims to charge not for the software itself, but for the services they provide, such as assessing an organisation's needs and setting up the systems

for them. Red Hat hence could have only stable components in the Enterprise version as it was targeted at businesses. This meant there had to be a different means of ensuring support from the developer community. The Fedora project therefore came into being, where newer components, though possibly a little unstable, would be added for the enthusiast.

Why do people say "Red Hat" and "Fedora" in the same breath? Are they not two different distributions?

Red Hat Enterprise actually borrows inputs from the Fedora community. This ensured that a new and feature enhanced version would be available for free to the home user, while at the same time, the corporate user would get a stable, tried and tested version along with assured support, but for a price. Works well, you might think, but Red Hat is at the receiving end of various criticisms by open source purists for their pricing of the software or service.

The latest Fedora is Fedora Core 4. It is available for download from http://fedora.redhat.com/Download/ . If you have a 64-bit CPU, choose the link that says X86 64, else choose i386. Fedora Core 4 is a 4-CD installer, with the total download size being about 2.6GB. So unless you are on broadband with no download restrictions, do not attempt to download it! Borrowing it from a friend is a better idea.

Fedora Core 4 offers you the choice of setting up either a KDE or GNOME Desktop or both. The kernel runs on Linux 2.6.11 and contains KDE 3.4, GNOME 2.10, and OpenOffice.org 2.0 (pre-release version). If you choose the complete installation, you will need about 7 GB only for the system files and applications. Installing typical components (such as music players, Web browsers, and OpenOffice.org) and leaving out developer tools, you can install it in about 2 to 3 GB. Installation is taken care of by Anaconda, one of the best OS installers available today. The Disk Druid tool, which allows you to edit the partition table, can be a little confusing for a newbie despite the on-screen instructions.

We installed both the KDE and GNOME environments to see how each one behaved. Applications took noticeably longer to start up with KDE than with GNOME. This was with 256 MB of RAM on a P4 2.6 GHz machine. But upping the RAM to 512 MB saw the difference in response time almost disappear.

Hardware is well-supported on Fedora Core 4. ATi and nVidia, the two largest video card makers today, have their cards supported. A variety of monitor models are also recognised, and the display is adjusted accordingly. Support for card readers is also inbuilt, with the auto-mount feature. However, it would have been great if Windows partitions, at least on the same hard disk as Fedora, would auto-mount too. Perhaps we're asking for too much!

Being a developer and enthusiast-oriented release, Fedora is scheduled for release every four to six months, while the more stable and tested Red Hat is projected for upgrades every year or longer. The biggest advantage you have with Fedora is the seemingly infinite number of applications written to run on it. The rpm (Red Hat Package Manager) format is more or less the default standard adopted in distributing applications. Even SuSE has agreed to support the rpm format. Installing applications is a matter of just one command, making it very friendly for a newbie.

In terms of features and customisability, we would say Fedora leaves little to be desired. You can spend hours just customising your machine without doing any work! The biggest plus point for Fedora must be the availability of a vast pool of applications that can be installed besides the bundled ones.

Trivia

There has been some conflict over the use of the name "Fedora." Originally, it was the name given to an academic project that was underway at Cornell University in association with the University of Virginia for the former's digital library in 1997. At that time, FEDORA stood for Flexible Extensible Digital Object and Repository Architecture. It was released for public use first as Fedora Release 1.0 in May 2003.

8.1.2 Mandriva (formerly Mandrake)

Mandriva, or Mandrake as it was earlier called, is a Red Hat derived Linux from France. It was first introduced in 1998, at a time when Linux was still associated with geekdom. Mandrake tried to fit an easy-to-use GUI to lessen the use of the command line as much as possible. Mandrake grew in popularity, and its own fan club soon followed. The latest Mandriva version is Mandriva Linux 2006, which is excellent for playing multimedia files. When we tested it, we could play almost all file types.

Mandriva 2006 incorporates KDE 3.4.2 and GNOME 2.10.2, and runs off the Kernel version 2.6.12.11. The OpenOffice.org version is the stable 1.1.5, rather than the newer pre-test 2.0 version. Application updates are very well thought out in this flavour of Linux. Mandriva now has the Smart Package Manager which can work on different types of packages, urpmi repository, rpm directory, apt-get repositories, apt-rpm repositories, Slackware repositories and deb directories.

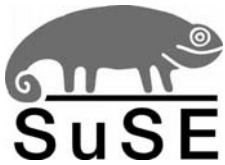
Mandriva has been known for its centralised controlling tool called Mandriva Control Center, from which you can configure the entire system, soft installation/un-installations, bootup and startup options, and hardware configuration. Mandriva creates subdirectories in the Home directory called Documents, Download, and Pictures, where the appropriate files get saved by default. Installing Mandriva is easy. If, during the install, a CD is missing, you can simply choose not to install the applications in question, and continue with the rest of it—generally a less complicated install process.

Hardware support is also not an issue for this Red Hat derivative. Like its distant cousin Red Hat, Mandriva also has an Enterprise Edition and a free Personal Edition. Corporate Desktop, Server and Security based distributions are available for a price. Mandriva comes on five CDs or a single DVD. There is a "3 CD download edition" that is available as an ISO, and which can be

burnt onto CDs. For the free edition, support is available in the form of mailing lists and forums.

You can download Mandriva from wwwnew.mandriva.com/en/downloads/mirrors

8.1.3 SuSE



One of the earliest of the Linux flavours, and seen as a competitor to Red Hat, SuSE (pronounced Soo-Zay) is almost as old as the Linux kernel itself. SuSE is actually a German acronym written as S.u.S.E., and is short for "Software und System Entwicklung," which means "Software and System Development." SuSE Linux is currently being supported by Novell. Like Red Hat's Linux, SuSE, too, is available separately for enterprise users, backed by assured support, and for personal use (enthusiast and developer) with newer features and limited support.

The latest and free version of SuSE is called *SuSE Linux*, while the enterprise versions are available as *SuSE Linux Enterprise Server*, *Novell Open Enterprise Server*, and *Novell Linux Desktop*.

OpenSUSE is a complete open source package of SuSE Linux that does not include any proprietary software such as Acrobat Reader, Macromedia Flash Player, etc. Currently, OpenSUSE 10.1 beta is available for download. For those of you who don't want to mess around with a beta release, there is always the stable release 10 available for download. Unlike SuSE Linux, OpenSUSE does not come in Live CD form.

SuSE 10 uses its YaST installer, another easy-to-use installation tool. YaST allows you to create a partition for Linux using free space from the Windows partitions. Meaning, there is no need to have in place unformatted / free space or empty partition. A typical install would consume 2.5 GB, while the minimalist install with either KDE or GNOME will take some 700 MB. The versions of the desktops are 3.4.2 for KDE and 2.12 for GNOME.

OpenOffice.org 1.9.125 is default on SuSE 10, though you can install the later version available on the Internet. Powering SuSE currently is the Linux kernel 2.6.13-15.

A single CD or 5-CD ISO version is available for the 32-bit version of SuSE 10, while for the 64-bit version, you need a DVD ISO.

8.1.4 Debian GNU/Linux

This is a Linux flavour compiled and distributed by The Debain Project, started by Ian Murdock in 1993. In the name "Debian," "DEB" comes from his wife Debra, and "IAN" from his own name. The Debian Project is an association of individuals who work towards creating and updating the Debian operating system. At present, there are no paid versions of Debian, unlike Red Hat or SuSE. Debian is purely a developer/enthusiast oriented distribution. A lot of the contributors to Debian are from academia. According to Debian, there are 15,490 free packages available for their OS! Debain uses its own format of packaging software called dpkg. This is different from the rpm (Red Hat Package Manager) used by Red Hat and SuSE.

Starting from its inception in 1993, a major chunk of the basic tools that make up Debian come from the GNU project. Hence the name Debian GNU / Linux. Currently Debian is in Release 3.1, with its last update being 20th December 2005. Work is on to provide Debian for kernels other than Linux as well.

Debain uses the apt-get package installer. What is special about apt-get is that when you execute the apt-get command with the -f switch, apt-get automatically downloads the required dependencies from the Internet and installs them. Thus, the system administrator is relieved from checking for dependencies and getting them before installating an application.

Debian finds use in a variety of businesses and academic insti-

tutes, including the Tata Institute of Fundamental Research in Mumbai. Debian is light on system resources, needing just about 500 MB of disk space for a minimal install. You can download Debian from www.debian.org/CD.

8.1.5 Ubuntu

Ubuntu is available both as a Live CD and as a full-fledged installation. Ubuntu is a derivative of Debian, and packs in some handy features. It comes with the GNOME desktop environment, and doesn't tax your system resources.

Ubuntu is an African word that means "humanity to others." The word also conveys the idea of peace and co-existence. Ubuntu has been created by Mark Shuttleworth and Canonical Ltd. Canonical Ltd has been involved with open source, and has announced that it will render support for Ubuntu Linux. Ubuntu is available in a choice of many languages which must enable users from across countries and regions to make good use of the operating system. (Incidentally, "Ubuntu" also happens to be the name of an organisation that works for the African cause.)

For all the niceties packed in Ubuntu, the lack of a GUI during installation could be its undoing, though the keyboard interface is quite easy and help is available on-screen. But for those who are just too used to a GUI everywhere, installation may be a little uncomfortable. This apart, Ubuntu syncs with the user with ease. Ubuntu packs with it OpenOffice.org, Thunderbird, Firefox, media players and system tools for all your computing needs. All this in just a one CD installer!

Ubuntu has a special feature: it doesn't allow a root login through the GUI. You have to login as a user and then switch to root in the terminal. This reduces the chances of accidentally changing anything critical.

Ubuntu is currently in version 5.10, and the next version, 6.04,

is due for release this month. Ubuntu Linux is an example of free software that can be used to spread awareness of a cause.

By default, Ubuntu comes with the GNOME desktop. There is also a KDE version available, called Kubuntu.

8.1.6 Knoppix

Knoppix is a Free and open source Live Linux CD derived from Debian. It contains commnonly-used software such as OpenOffice.org, AbiWord, The GIMP, Konqueror, Mozilla, Apache, PHP, MySQL and many other open source programs. It runs completely from a CD, without the need for anything installed onto the hard disk. Due to on-the-fly decompression, the Live CD can hold up to 2 GB of software! On-the-fly decompression means the contents are extracted as they are being read, without the whole package being copied first to the hard disk.

Knoppix can be downloaded via BitTorrent, or can be ordered from a vendor. The download site is www.knoppix.org .

Knoppix also allows you to install to the hard disk and then multi-boot, but the installer interface is not intuitive. If you wish to install Knoppix on your hard disk, read the on-screen instructions *very* carefully—a wrong move can wipe all data off your hard disk!

8.1.7 Gentoo Linux

Gentoo is literally the name of a small, fast penguin found in the Falkland Islands. The “g” in “Gentoo” is a soft “g” as in “gentle”. One of the high points of Gentoo Linux is its customisability. With Gentoo, it is possible to build an entire system from source using a choice of optimisations and customisations. The user has complete control over the packages that can be installed. Gentoo is called a meta-distribution for the same reason.

Portage is a special packaging format used by Gentoo that allows flexibility while installing and maintaining software on Gentoo. Portage features are added frequently and official releases are scheduled every six months. Portage can be made to update all the programs that you want to in a single command. Portage can build custom versions of software that will be optimised for your particular hardware. According to Gentoo, there are about 10,000 packages in the Portage tree.

The current release is Gentoo Linux 2006.0. The releases are named by the year of release followed by the release number for that year. Hence 2006.0 means the first release in 2006, 2005.1 means the second release in 2005, and so on. Gentoo is available with KDE and GNOME, so there are no issues for fans of either or both of those.

You can download Gentoo from www.gentoo.org/main/en/where.xml. Gentoo can also be run from a Live CD.

To end this brief introduction of Gentoo, here's a small excerpt from Daniel's writing published online on IBM's developerworks:

Linux is about people The next thing I learned was that Linux is about people. Isn't that refreshing? Linux isn't just a bunch of source code. It's a community. We rely on this community to get our questions answered, and we become part of the community when we start helping others by contributing our time and expertise.

8.2 Linux And 64-bit Computing

About 2 to 3 years ago, 64-bit CPUs made their foray into the desktop market, and have now become quite affordable. Linux distributors were quick to come out with 64-bit versions of Linux. Today, many flavours of Linux also come in 64-bit editions to make full use of the power of 64-bit computing. To the end user, this means faster computing, especially with graphics-heavy applications. When you visit the download sites of the distributors, you may see a download that

is available for *x86 64*. The "64" means that the installer is for a 64-bit processor. The major distributions we have covered— Fedora, SuSE, Mandriva, Ubuntu, Knoppix, Gentoo—all have 64-bit versions.

8.3 Linux On A PowerPC?

Linux can be comfortably run on an Apple Mac, too. The scalability of Linux means that it can be used on Macs that are too old for OS X but still offer most of the nice features of the Mac OS. Linux for PowerPC (the processor used by Apple Macs before they switched to Intel), including the 64-bit version called the G5, is available from all major distributors. Most of the distributors that we have covered also have a PowerPC version—look for a version of their Linux for PPC or PowerPC on their site.

Many companies have found that they are able to make better use of older Mac models such as the RS/6000 or pSeries by running Linux on them and having more current hardware. Linux and PPC make for a good combination, given Linux's stability and inherent server features. The Mac can support multiple processors and very large memory as compared to the Intel architecture.

Of particular mention here is Yellow Dog Linux, which is specifically made to be run on the PowerPC platform. Yellow Dog dual-boots with Mac OS as well. It includes The GIMP, Kontour (a vector graphics application), and OpenOffice.org; developers have C/C++, PHP, Pearl, and more. Yellow Dog uses the KDE desktop.

Tip

Downloading one CD's ISO can be a daunting task in India, where internet speeds and download limitations are problems for large downloads. Downloading four or five CDs only makes life tougher. Instead, you can order it from www.kiraninfotech.com or www.buylinuxdvd.com.

These are Indian dealers from whom you can obtain Linux CDs and DVDs at reasonable costs. The Knoppix CD, for example, costs just Rs 50, while the DVD can be bought for just Rs 170.

Freedom with Linux

The ideology of 'free' OS is not to provide an operating system free of cost but to give users the freedom to learn in depth how an OS works and allow them to change it to their liking. Therefore this freedom comes with a lot of responsibility. Being a system administrator is not an easy task, primarily because you will be handling many servers at one time. We will not deal with that level of administration. Here, we will chew over those administration areas related to home user level.

System administration is not a child's play. Be prepared to counter lots and lots of commands through this section. Most of the commands are restricted to a normal user and so you have to don the root's cap. With this power comes great responsibility!

This chapter will take you from booting your machine (also called bootstrapping) and then on to various runlevels and services of the OS. Commands are introduced mid way; these commands deal with system monitoring and certain level of tweaking. Upgrading software is discusses in section 9.4; this primarily deals with yum and apt-get. We will wind this chapter with Kernel Compilation—both Rebuilding and New Compilation.

9.1 Boot Loader And Tweaking Boot Process

The booting process of an operating system is very important because it differentiates between a functioning and a malfunctioning system. Many home users tend use Linux as the secondary operating system and dual-boot it with Windows as their primary operating system. The use of Windows as a primary OS is because of the lack of information on Linux, and also because many games and software are not available for Linux; for example: Microsoft Office.

As the administrator of your system you have the power make changes to the boot process. You can change the services that start at boot, alter the settings of the boot loader, as well as use other powerful tools such as the single-user mode.

Before we introduce you to the power of 'root', let us tell you a little about how Linux handles booting in general. We will also consider the dual-boot part, as it will also include the MBR section in detail.

Technically, one can create four primary partitions of which one is extended to a virtual area called 'Extended Partition' in order to allow creation of more partitions. Under this Extended partition, a user is allowed to create up to 64 'Logical Drives'.

How is the story of partitions linked to booting of OS? Well, almost all earlier operating systems (such as Linux 1.2) needed a primary partition to install at least its 'boot' sector. When people used the early Windows OS (such as Windows 95/98), they had to use the FDISK tool to create partitions. This disk partitioning tool by Microsoft did not allow more than one primary partition. Consequently, installing more than one operating system became difficult as installing Linux first and creating a primary partition for Windows installation will not work because Windows will overwrite the MBR. This resulted in the creation of Boot Loader. When the Boot Loader was being formulated, some changes were also made to the partition-system of Unix-like OSes. Eventually, Linux's ***/boot*** or for that matter all partitions can be a logical drive and it will boot!

This is true if it is made to dual boot with another OS. But if it is the only OS then the ***/boot*** partition has to be primary with the MBR in it of course (first cylinder of the hard disk).

Linux Loader or the popular LILO thus became a valuable asset of the Linux community. But later on Grub gained importance as it is more secure and has some features in addition to LILO. The newer distribution does not bundle LILO due to the popularity of Grub.

Assume that an OS is installed, say Windows, prior to Linux. The role of a boot loader is to append the MBR and not overwrite it. It writes its presence to the MBR. When a PC is turned on, the control of booting an OS is transferred by the MBR to the boot loader. The boot loader is set to boot a specific OS within a set time. The default OS can either be Windows or Linux. If Windows is selected to boot, then the boot loader will transfer control back to the MBR to follow up with booting Windows (boot information of Windows is still in the MBR). And, if Linux is chosen, then init takes the control from the boot loader. 'init' is a daemon (daemon is a process, explained later in the chapter) that starts various other system processes required for booting and eventually running the Linux OS.

Since we have a fair idea of the role of the boot loader let's look at the two boot loaders available for Linux systems.

9.1.1 LILO

Linux Loader is the traditional boot loader and it is perhaps still used in servers that run Linux as their Operating system. Grub or the GRand Unified Boot loader replaced it as the 'cool' and secure alternative.

LILO has a configuration script—lilo.conf—located under the ***/boot*** partition. This is the file that LILO reads during the booting of a system.

The file lilo.conf may look lsimilar to the one below.

```
boot=/dev/hda
map=/boot/map
install=/boot/boot.b
prompt
timeout=10
message=/boot/message
lba32
default=LINUX-MINE

image=/boot/vmlinuz-2.4.0-0.43.6
  label=LINUX-MINE
  initrd=/boot/initrd-2.4.0-0.43.6.img
  read-only
  root=/dev/hda5

other=/dev/hda1
  label=Windows XP
```

Among the few things that you can change here are:

'timeout=' This is the maximum time in seconds to choose an OS to boot; else it will boot default

'default=' Type the label on the OS you want to boot as default on timeout.

And ***'label='*** You can type just about anything here. Put

'Billie's OS' as the label for Windows operating system and LILO will display it that way.

After you have made changes in the configuration script file of LILO, you will have to issue a set of commands in order to finalise the changes. They are:

#lilo -v [Enter]

#lilo -t [Enter]

If you do not get any error messages then LILO is configured correctly. You may also choose to issue:

#/sbin/lilo [Enter]

The error-free output should be:

Added LINUX-MINE

Added Windows XP

If you see anything other than this then something has gone wrong. Go through the entire configuration file and also check the man file if you have any doubt. Man file can be accessed by issuing ***'man lilo'***.

Entering Single User Mode In LILO.

Turn on the system and LILO takes over to display the OS choice. Here press ***[Ctrl] + [x]*** to enter LILO's edit screen. At the prompt type 'Linux Single' and hit ***[Enter]*** to boot into single user mode. The power and use of singe user mode is explained later in this chapter.

9.1.2 Grub

Grub has eventually replaced LILO as the preferred boot loader. It not only looks pleasing in appearance, it is secure (password encryption method) and has cool features such as changing the boot screen of a boot loader. See the box on Changing the boot screen for more.\ Grub also has a configuration file and just like LILO it reads this file during the bootstrap process. The file is located under the gurb folder of /boot partition. Some distributions have the configuration file named as menu.lst. In either case the content of the file remains the same.

The grub.conf or menu.lst file will look something similar to the one below:

```
boot=/dev/hda
map=/boot/map
install=/boot/boot.b
prompt
timeout=10
message=/boot/message
lba32
default=LINUX-MINE

image=/boot/vmlinuz-2.4.0-0.43.6
  label=LINUX-MINE
  initrd=/boot/initrd-2.4.0-0.43.6.img
  read-only
  root=/dev/hda5

other=/dev/hda1
  label=Windows XP
```

A user can make plenty of changes to this menu. The options such as 'hiddenmenu' can be commented if you want to see the main Grub screen with the list of OS; else the hidden menu would count down to boot the default OS. The ***'timeout='*** specifies the time within which the ***'default='*** boots. Unlike LILO, Grub denotes its default with a number. The Number begins with 0. Assuming there are three OSs listed as 'title ' then the first one is assigned '0' the next is assigned '1' and so on. In the above example, 'default=0' indicates that Grub will boot Fedora Core by default within five seconds and a user must hit any key on the keyboard to see the main Grub screen (override hidden menu) and choose the desired OS.

Notice the line ***'splashimage=(hd0,5)/grub/splash.xpm.gz'. '(hdo,5)*** is actually the ***/boot*** partition. The whole line is indicating to a ***/boot/grub/splash.xpm.gz*** where ***splash.xpm.gz*** is the image that will be displayed on the Grub's main screen.

Changing The GRUB Splash Image

If you are bored seeing the same blue GRUB screen every time, then take a break—as you can change the GRUB 'splash' image by editing the grub's config file. Before we proceed to change it with any image let us understand the image requirements.

The image has to be 14 colours with maximum 640 x 480 resolution and in the xpm format. The first question would be "how to obtain such an image?" Well, some distributions like Redhat 9 have a command line utility called ImageMagick. Do ***#man ImageMagick*** to know more about it. Let's consider that you have an image in jpeg (*.jpg) format that you want to load as your splash screen. To change 'photo.jpg' into a splash screen image switch to root (administrator) then use the following command in konsole or an xterm:

The new look on grub after modifying the grub image

#convert -depth 8 -colors 14 -resize 640x480 /root/grub-pics/photo.jpg /root/grub-pics/photo.xpm

Assuming that your image is in folder called 'grub-pics' under ***/root***. Needless to say that convert command converts the image in the necessary format but we still have to compress it to 'gz' format; so under ***/root/grub-pics*** do ***#gzip photo.xpm***. The output of the command is the file we need ***'photo.xpm.gz'***. Now, copy the image to ***/boot/grub*** using the command

#cp /root/grub-pics/photo.xpm. gz /boot/grub

The last step is to edit the grub's config file. Open the ***grub.conf*** file that's in ***/boot/grub*** using any command-line text editor such as vi.

Look for the line ***'splashimage='*** change the path of splash screen image from ***/grub/splash.xpm.gz*** to ***/grub/photo.xpm.gz*** (as in our example).

Reboot your system to see the change in grub's screen. You can avoid the task of converting image to xpm format by using readily available splash-screen images from the net. Just save the image in ***/boot/grub*** and change the path of splash image in the ***grub.conf*** file. Flaunt the new look!

Adding Password To Grub

To secure your PC's access rite from the boot loader, you can add a password in Grub. We do not recommend this feature for home users as it is not necessary but it is an information that any Linux user should know. This feature is normally used in server-based Linux setups.

To create a password:
#grub-md5-crypt [Enter]
output:
Password: ****enter password****
Retype password: ****enter same password****
1gtBbK1$edmvHBi2uEPq4Nhbcbd7Q.

The last line which seems like garbage characters is actually the entered password in md5 encrytion format. All you have to do to enable this password protection is to copy the above encrypted output into the grub.conf or menu.lst file as:

password ? ***md5 1gtBbK1$edmvHBi2uEPq4Nhbcbd7Q.***
Just above the ***'tile'*** line.

Save the grub file and exit.

Note: You DO NOT have to run any command to effect the changes.

Entering Single User Mode In Grub

On the main screen on Grub i.e. By overriding the ***'hiddenmenu'*** you will see the list of OS. Choose Linux and then press ***[E]*** on your key-

board. On the next screen you will see three lines related to the boot entry of Linux. Choose the line stating ***'kernel..'*** and hit ***[E]*** again. A new editable screen appears, and at the end of the line type ***'single'*** and hit ***[Enter]***. This will take you back to the previous screen. Now, press ***[B]*** key of your keyboard to boot the OS in Single User Mode.

9.1.3 Changing Boot Loaders

A user is allowed the freedom to change the boot loader as required. Some basic Linux OS come with a default boot loader and an alternative one need to be downloaded from the Internet. For instance, if you have a boot loader LILO and the installation did not feature Grub then you will have to download it from the internet and install it too. Thereafter you need to make that switch between the two boot loader.

LILO to Grub

Download and install Grub from ***http://www.gnu.org/software/grub/***

To switch the boot loader issue the following command

#grub-install /dev/hda [Enter]

Grub to LILO

If LILO is not installed in your system then search on the internet for a package relevant to your distribution or refer this site: ***http://www.linuxloader.com/***

If you have a distribution that uses anaconda as its installer, then most likely you will have a file in ***/etc*** named ***'lilo.conf.anaconda'*** copy this file to ***/etc/lilo.conf*** and the execute the lilo command.

#cp /etc/lilo.conf.anaconda /etc/lilo.conf + [Enter]
#lilo -v + [Enter]
#lilo -t + [Enter]

9.2 Runlevels

Runlevel is a preset state for Linux operating system. A system can be booted into any of several runlevels, each of which is represented by a single digit integer. Each runlevel designates a different system configuration and allows access to a different combination of processes. For example, runlevel 3 doesnot start the GUI services.

Seven runlevels are supported in the standard Linux kernel. They are:

0 — System halt; no activity, the system can be safely powered down.
1 — Single user mode.
2 — Multiple users, no NFS (network filesystem); (user defined).
3 — Multiple users, command-line (i.e., all-text mode) interface; the standard runlevel for most Linux-based server machines.
4 — User-definable
5 — Multiple users, X11 (GUI); it is the standard for Desktop-level Linux machines.
6 — Reboot; used when restarting the system from Command Line Interface (CLI).

By default Linux boots either to runlevel 3 or to runlevel 5. The former permits the system to run all services except for a GUI. The latter allows all services including a GUI.

The program responsible for changing runlevel or booting a predefined runlevel is ***init***. To change from one runlevel to another the ***'telinit <runlevel>'*** command can be issued.

Eg:

#telinit 3 + [Enter] to change from runlevel 5 to 3. Assuming that current runlevel is 5.

As a system administrator, let's get down to the depth of runlevels. A file ***/etc/inittab*** has an entry for the default runlevel. In order to manually change the default runlevel edit the line:

id:5:initdefault:

Replace the number 5 with 3 to change the runlevel from GUI to Text-mode. Save the file and reboot the system. You will notice that the system now boots to a text mode (black back ground with white text).

The runlevel for system administration is 1. It is also called single user mode. It is rarely used (and should be that way). It is put into action only if there are problems encountered when booting into normal runlevels like 3 and 5. For example, if a change made in the X Window System configuration of a desktop has rendered the Graphic mode of the system unusable, then it is possible to temporarily boot into a console (either runlevel 1 or 3) and rectify the changes done to the X Window configuration files. Similarly, if a machine will not boot due to a damaged configuration file or has login problems because of a corrupted ***/etc/passwd*** file or because of a forgotten password, this problem can only be solved by booting into single-user mode (runlevel 1).

9.2.1 Select Important Services—Speed Up The Boot Process

Most of the new users install a particular distro and use it with all the default process running. Processes such as pcmcia, nfs, sshd etc. are not required on a home desktop machine; these processes are mainly used on a server and some can be used on a desktop if you have a small network environment. The normal boot process of Linux will start most of these unwanted processes resulting in a long wait. It is easy to turn these unwanted processes OFF especially on a Redhat based system.

Use the command ***#chkconfig –list*** to display the list of process and its status for levels 0, 1, 2, 3, 4, 5 and 6. Of all these levels only 3 and 5 are concerned with desktop user operation. Instead of suggesting the processes to be disabled, we would rather tell to you the processes that are necessary for ideal operation. The essential processes are anacron, crond, echo, gpm, keytable, network (enable if the PC belongs to a network), random, sendmail, sshd, syslog and xfs. You may enable some more process if you think necessary, say for instance, kudzu is a hardware detection process that takes considerable time in the boot process; most of us don't open our

Single User Mode

We have explained how to enter single user mode from LILO and Grub under section 9.1.1 and 9.1.2 respectively. Single user mode comes with lot of responsibility and anyone who uses this mode must understand the freedom that Linux OS bestows on its users. Almost all the commands of Linux system works in single user mode. This mode is normally used by system administrators at to implement certain changes that he/she cant do if users are logged into the system. At home user level this mode can be used to overwrite the password for root and/or users. Use it only when the password is forgotten.

Hardening Single User Mode:

Any local user can misuse root privileges to tamper your systems configuration through single user mode. To protect your system from such instances you can either add password to the boot loader Grub or LILO (we have explained how to add password in Grub, 9.1.2). But the advance level of security will be to enable password protection to single user mode wherein anyone who logs into this mode will be asked for root password in order to continue. But, one must now understand that enabling such a security will disable root password recovery method. Your memory power will be the only source of rescue. To harden the single user mode to password protection level you need to edit the **/etc/inittab** file and append a line **'id:S:wait:/sbin/sulogin'**

This will execute **'sulogin'** which will demand root password just before opening the shell prompt (sh-01#) of single user mode.

Remember: Never ever forget the root password!

machines and replace the PCI cards every now and then; so kudzu can be disabled. To disable a process you will have to use ***'chkconfig'*** command repeatedly for every process that needs a change in its status. The complete command is ***#chkconfig –level <levels 3 or 5> <process name> on|off***

eg: ***# chkconfig—-level 5 kudzu off***

For a comprehensive list of process and its definition visit: ***http://techrepublic.com.com/5100-3513_11-6018195.html#***

For Redhat users this task is extremely simple; thanks the nifty use of ***ncurses*** (uses colour and crachter map present on text-mode to produce GUI-like sub-window). Typing the command ***'ntsysv'*** on a konsole prompt will list out all the process with its status indicated in ***'[*]'*** box preceding the process name; use spacebar button to remove the ***'*'*** in the box, this will disable the respective process. Similarly, Suse and Mandrake also have GUI front-ends to enable disable service.

Once the necessary processes are set, reboot the system, you will see a marked difference in the booting time.

9.3 System Monitoring Methods

System monitoring involves both software level and hardware level monitoring. In this section we shall glance through processes and then take a look at certain hardware features such as disk and memory usage and also a tip on using flash drive.

9.3.1 Process: Checking And Killing Unnecessary Ones

If you have used Windows you will know about the Task Manager which is often used to terminate 'Not Responding' processes. This is quite common in Windows, but you may not see this in Linux. In Linux, an unresponsive process is allowed some time and resources and when it fails to recover it is terminated after issuing a message to the user. However, a user can kill the process using the command of the same name. This is not common and if this happens on your machine frequently then there is something wrong with the setup of the system, or you may be using an unstable version.

In this section we shall introduce you to certain commands using which you can check on the system processes.

What is a Daemon? It is a process that does not require interaction with the user to run; for example init. Some Daemons run in the background and therefore can also be called background processes. While a Daemon (or any background process) runs, the user can continue using other programs or commands.

What is a Process, Parent process and Child Process? A service that is initiated by init is moved into RAM where it becomes a process. Every process has an identification number, or PID (Process IDentification). A process resulting from a process (forking) is called the Child Process whereas the Process responsible for its initiation is termed as the Parent Process. The Child Process has a PID and the parent can be identified from its PPID (Parent Process ID).

If you issuet the ***'ps -el'*** command you will see that most processes have PPID as 1; indicating that 'init' is the mother of all processes. The 'pstree' command displays all the process in a tree format listing them under their parent process.

Issue ***'ps -aux'*** to display all the process with memory usage, cpu usage, PID, etc.

Vmstat displays memory information. There are several switches that can be used to get more information out of your system. An example of 'vmstat' is displayed below:

```
[root@test3 ~]# vmstat
procs -----------memory---------- ---swap-- -----io---- --system-- ----cpu----
 r  b swpd free   buff   cache   si  so  bi   bo in   cs   us sy id wa
 1  0 0    38516 137448 338300  0   0   30   12 1159 859 7  1  92 1
```

Listed below (see next page) are switches that can be used along with vmsat:

If any process is not responding, then open a CLI and check the PID of the non-responsive process using the above command. Terminate the process by issuing the ***'kill'*** command.

#kill <process id>

Using switch such as ***'-9'*** will try to terminate the application with immediate effect.

```
usage: vmstat [-V] [-n] [delay [count]]
        -V prints version.
        -n causes the headers not to be reprinted regularly.
        -a print inactive/active page stats.
        -d prints disk statistics
        -D prints disk table
        -p prints disk partition statistics
        -s prints vm table
        -m prints slabinfo
        -S unit size
        delay is the delay between updates in seconds.
        unit size k:1000 K:1024 m:1000000 M:1048576 (default
       is K) count is the number of updates.
```

9.3.2 Changing Priority Of A Process

Commands are designed to change the priority of processes in order to speed up the task for which a process is called. For instance, compressing a big file using bzip2 compressing tool can consume some time. A user can either initiate this command with ***'nice'*** preceding it or use ***'renice'*** to alter the priority after the bzip2 process is initiated. The syntax of the command is shown below:

#nice -n -12 tar -jcvf bigfile.tar.bz2 bigfile + [Enter]

The priority levels can be defined between 0 (lowest) and 20 (highest)

If the application is already running and you want to make it faster by increasing the priority and allowing it to consume more resources then:

#renice -15 `pidof bzip2` {Note: *[`]* is a back tick found on the key with tilde (*[~]*)}

pidof will feed the process id number of bzip2 to renice command.

Also remember that assigning highest priority to a process will make the keyboard and mouse sluggish and sometimes they may not respond. So, priority of up to 15 is considered safe.

9.3.3 Some Useful Stuff

LSPCI

It is a command line utility to display information of all the PCI devices on your machine. This is particularly useful to find drivers

for a hardware that is detected by BIOS and is installed in the Linux system but isn't functional due to the absence of proper driver or wrong driver selection.

Using the ***'lspci'*** command with ***'|grep -i <device name>'*** will display information related to the grep word. By the way, grep means "search globally for lines matching the regular expression, and print them" and the ***'-i'*** switch ignores case of the search word (case-insensitive).

Using Flash Drive

Before attaching a Pen drive/Thumb drive/Flash drive; what ever you call it, do ***'lsmod'*** and copy the output to a file, say ***'Before' [#lsmod > Before]***. The output may look something similar to the one below:

Module	Size	Used by	Not tainted
usb-storage	69332	0	
natsemi	19552	1	
ide-scsi	12208	0	
scsi_mod	107160	2	usb-storage ide-scsi
ide-cd	35708	0	
cdrom	33728	0	ide-cd
ohci1394	20168	0	
ieee1394	48780	0	ohci1394
keybdev	2944	0	
mousedev	5492	1	
hid	22148	0	
input	5856	0	keybdev mousedev hid
usb-uhci	26348	0	
ehci-hcd	19976	0	
usbcore	78784	1	usb-storage hid usb-uhci ehci-hcd
ext3	70784	7	
jbd	51892	7	[ext3]

Insert the Flash drive to the USB port and repeat the ***'lsmod'*** command and save the output to a file, call it ***'After' [#lsmod >***

After] using some text editor. Check the difference between the entries in the two files. No, we are not suggesting you to manually compare every line.

#diff Before After + [Enter]

The output of the above command is the extra module that got activated due to insertion of Flash drive. Most likely the module is ***/lib/modules/2.4.20-8/kernel/drivers/scsi/sd_mod.o***

Therefore, the Flash drive can be mounted on your system by issuing the mount command:

#mkdir /mnt/flash + [Enter]

#mount /dev/sda1 /mnt/flash + [Enter]

The content of the Flash drive is accessible from the ***/mnt/flash*** folder.

/ETC/FSTAB

The file ***'fstab'*** located under the folder ***/etc*** is the back bone of the mount command. Mount reads ***/etc/fstab*** and mounts the partitions or devices listed under it. A typical ***'fstab'*** file would read similar to the one below:

```
# This file is edited by fstab-sync—see 'man fstab-sync' for details
LABEL=/         /               ext3        defaults          1 1
LABEL=/anup     /anup           ext3        defaults          1 2
LABEL=/boot     /boot           ext3        defaults          1 2
/dev/devpts     /dev/pts        devpts      gid=5,mode=620    0 0
/dev/shm        /dev/shm        tmpfs       defaults          0 0
LABEL=/home     /home           ext3        defaults          1 2
/dev/proc       /proc           proc        defaults          0 0
/dev/sys        /sys            sysfs       defaults          0 0
LABEL=/tmp      /tmp            ext3        defaults          1 2
LABEL=/usr      /usr            ext3        defaults          1 2
LABEL=/var      /var            ext3        defaults          1 2
/dev/hdc        /media/cdrecorder           auto
pamconsole,exec ,noauto,managed 0 0
```

CDRECORD
Most Linux distributions have command line utility to burn CDs and DVDs. This command along with mkisofs completes the package of disc media writing. The newer Linux distributions supports DVD writing the eariler one dont. For more on how to use this utility refer Digit February 2006 Tips and Tricks.

Lets say, you want to mount your Windows partition at the startup. Windows partitions are located under ***/dev/hdb1***, ***/dev/hdb5***, ***/dev/hdb6***. Create a folder callled ***'win'*** under say ***/root*** or ***/mnt*** and then create folder under win for each of the three partitions. We used the drive letter as file names for the mount points. Therefore, when mounted, the Windows partition would appear under ***/mnt/win/C***, ***/mnt/win/D*** and ***/mnt/win/E***. To automate the mounting of these partitions, just open a ***'fstab'*** in a text editor and make the following entries:

```
/dev/hdb1 /mnt/win/C    vfat    defaults 0 0
/dev/hdb5 /mnt/win/D    vfat    defaults 0 0
/dev/hdb6 /mnt/win/E    vfat    defaults 0 0
```

Use ntfs instead of vfat if the partition is of that file system. Linux will warn you about mounting NTFS partitions—it mounts NTFS as read only. Having said that, there is absolutely no problem in mounting FAT32 partitions. Save the file and exit the editor. No need to restart the machine to check if it works. Use the ***'mount -a'*** command to test if the changes worked. If so, it will work at every start up. A little detail about the columns of ***'fstab'*** are provided in the table on the next page.

Using Shell Scripts

Any novice can use shell scripting effectively, provided he/she has good idea of shell commands. Taking the above mount process as an example, lets create a shell script to mount the Windows partitions only when required. All we have to do is to list the mount command with the mount device and mount point (both should be valid) in a text file using some editor (don't add extesion such as

Details of Column Entries in /etc/fstab

Column No.	Field Name	Definition
1	Filesystem	This is the device file or the device driver for the hardware such as hard disk, cdrom, floppy, or a partition in your hard disk
2	Mount point	This is the place (usually a directory) where you want the device listed in column 1 to be mounted
3	Type of filesystem	As the name indicates, it could be any filesystem like ext3, ntfs, ufs2 and so on
4	Mount Options	The fourth column in fstab lists all the mount options for the device or partition viz. auto and noauto, user and nouser, exec and noexec, sync and async, ro, rw and lastly defaults (rw, suid, dev, exec, auto, nouser, and async)
5	Dump frequency	Dump checks it and uses the number to decide if a filesystem should be backed up. If it's zero, dump will ignore that filesystem.
6	fsck	fsck looks at the number in the 6th column to determine in which order the filesystems should be checked. If it's zero, fsck won't check the filesystem. Filesystems that have the same number greater than 1 are checked in parallel. If you are mounting CDROMs or network filesystems, then this value must be 0.

.txt—that's Windows not Linux). The permission of this file has to changes in order to make it a shell script. It is that simple!

We will be using the command line text editor ***'vi'*** for the above purpose.

#vi win-mount + [Enter]

The above command will create and open the file but its unsaved. In the file type the following lines after pressing the ***[Insert]*** key.

#This script will mount my Windows partitions provided the drives exists
mount /dev/hdb1 /mnt/win/C
mount /dev/hdb5 /mnt/win/D
mount /dev/hdb6 /mnt/win/E

Now, press *[Esc]* to switch from text editing to *'vi'* command mode. To save the file and exit vi use *':wq'* [read as colon wq].

The last step is to convert this text file into a script file:

#chmod 700 win-mount + [Enter]

That's it! Whenever you want to mount the Windows partition make sure the disk is *'hdb'* (primary slave) as in our example and the files under */mnt* exists. And run the script from its parent directory.

#./win-mount + [Enter]

Few Useful Commands: df, du and tree

The *'df'* command is the simplest tool to view disk usage. Using it without any switches will display the result in block of usage which is not understandable especially for home users. Use the *'-h'* switch to make it human readable form i.e. the displayed disk usage will be in Megabytes, Kilobytes or Gigabytes. To check the inode usage (mostly necessary if quotas are introduced) use *'-i'* switch.

Filesystem	Size	Used	Avail	Use%	Mounted on
/dev/hda7	996M	525M	420M	56%	/
/dev/hda2	3.0G	1.1G	1.8G	39%	/anup
/dev/hda6	99M	9.2M	85M	10%	/boot
/dev/shm	236M	0	236M	0%	/dev/shm
/dev/hda8	487M	11M	451M	3%	/home
/dev/hda10	996M	35M	910M	4%	/tmp
/dev/hda12	7.2G	3.6G	3.3G	53%	/usr
/dev/hda11	996M	157M	787M	17%	/var

The *'du'* is much like the *'df'* command but it show file and directory usage. Again, using the command without the switch can be grasped only by a hi-fi system admin working on over five hundred

server machines. To make it understandable use ***'du -h /foldername'***. If you have tried this command along with us, then you know it shows a recursive output digging into every file and folder under the folder name your have specified. To avoid recursive output and obtain just the file size of the single directory or folder issue ***'du -hs /directory'***.

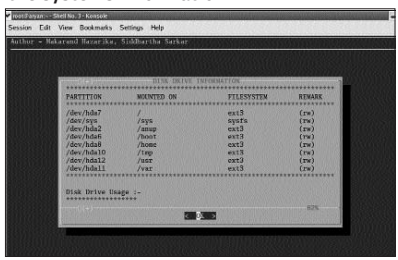
```
[root@aryan etc]# du -hs /digit/
1.1G    /digit
```

The tree command is not unique to Linux. It was there in MS-DOS too, but Microsoft discontinued it in Windows. This command gives a diagrammatic representation of the directory structure... much like the pstree command we saw earlier.

Digit Reader's Contribution To The Community

One of our readers, Makarand Hazarika and his friend Siddharta Sarkar jointly worked on a shell program that displays to a user all the information of a system. The shell script makes system administration a piece of cake as a user is not expected to remember all the commands and can work easily by use of keyboard. They have made nifty use of ***'ncurses'*** so that a GUI-like feel would be available at the Text mode (runlevel 3) too. His effort is now available for all to download at:

SysInfo2.3 gives a comprehensive details of the systems information



Disk partition information as shown by SysInfo2.3

http://forge.novell.com/modules/xfmod/project/?sysinfo

We at *Digit* congratulate him on his success.

Download the script from the site mentioned above. The download link may open as a text file in the browser itself. Save this page to you PC. Use CLI to change this Bourne Shell Script to an executable script:

#chmod 700 SYSINFOV2.3 + [Enter]

To run the script:

#./SYSINFOV2.3 + [Enter]

The script tries to run the GUI implementation and if it fails, it falls to the text mode. In either case you will get system information.

9.4 Upgrading Software ? YUM, APT-GET

Linux is an operating system that is in the constant process of updates and patches. Manually updating all machines can be a daunting task for a system administrator. Thankfully the user or administrator need not manually check for and download updates. Linux has a provision of automatically updating selective or all packages using what are called Package Managers. The Package Managers have a database of the software installed, their version, the Internet address from where an update can be obtained, etc. rpm-based Linux flavours generally use the yum package manager while the Debian-based ones use apt-get. These are basically command-line means of updating—it involves issuing a single command and configuring the config file if need be. There are GUI front ends to achieve the same. But we will just stick to the basic command line way to getting the update task done. The GUI is something you can figure out by yourself, so as usual we'll leave it for you to explore and have fun.

A group of packages are often put in a directory or a Web site called Repositories. The package manager looks for the repositories and installs the required applications from them. The repository itself can be downloaded and stored on the hard disk from where applications can be locally installed. Package managers can also be used to install applications from a package in which case the package manager will include the application in its database.

9.4.1 yum: Yellowdog Updater, Modifier

yum is used to update the rpm-based packages, meaning Red Hat / Fedora, SUSE and Mandriva, besides other distros.

How yum Works:

An rpm package consists of what is called a header along with the archive of the program itself. The header contains the list of files the make up the archive, a description of the package with the libraries it provides and requires from other packages and if it has any known conflicts with other packages. The package manager requires this information so that when a package is installed or uninstalled, none of the existing packages are affected. This process is called "resolving package dependencies". It is a challenging task for any package management tool. A situation may arise such that a packaged tool requires two or three libraries (a library is a list of commands or functions that are often used). These libraries in turn may require some other libraries. By the time the package is installed, there may be a lot more packages, and none of them must be conflicting with the other.

yum maintains a list of URL from which the repository is accessed in order to upgrade the packages when called for. This list is usually kept in a file called yum.conf located in the ***/etc*** directory. It is possible to edit this file to make changes to the way yum behaves while updating. You can also add URLs other than what is included by the distributions. Let us look at basic yum commands and then some of the changes you can make to yum.conf

To get yum to install a package the command is as follows:

yum install <package>

(replace "***package"*** with the actual name of the package).

Yum checks if ***package*** is already installed and is if, the version. If it finds the package not present, it downloads the package and installs it.

Updating the package is done using the following command :
yum update package

yum update will look for newer versions of the software and installs only if a newer version is detected.

Updating all packages listed in yum

yum update
This must be the most powerful command in yum! This command updates all the installed packages on the system to the latest version on your as specified in your repository.

Removing A Package From The System
When removing a package yum is programmed to remove all packages that depend on it. This is to ensure the system runs stably without facing dependency issues. You have to be very careful hence in removing packages. If the package that you are removing is something on which critical system processes, or say even X, depends on, you will have unknowingly removed the GUI from your Linux box! So be absolutely sure of what you are removing. The command to remove a package is
yum remove <package>

Editing The yum.conf File

You can use the ***vi*** editor to view the .conf files in Linux. When you open the yum.conf file, right in the beginning you will see options under ***[main]***. For example you will see a setting called ***"cachedir="***. This is where you can change the cache directory (like a temporary directory to store files during download or install).

Lets take another example, there is a setting used to specify the number of retries after which yum gives up acquiring the package. It is defined in the setting ***"retries="***. You can specify the number of tries here. You can insert a ***"throttle="*** setting and specify the maximum download speed for yum so that when you are downloading huge packages, your other downloads are not affected.

Using a proxy server: if you are on a network and can access the internet only through a proxy, you can simply add this line under *[main]*

proxy=proxy_url:port_no

where *proxy_url* is the address of the proxy server, you can specify the port number if required in place of *port_no*.

If a user name—password is required, all you have to do is add two lines in the following format:

proxy_username=<your username>

proxy_password=<your password>

where *<your username>* is the username and *<your yourpassword>* is the password for the proxy.

Adding URLs to download packages from repositories other than the ones given by default by the distro : When you look for download and install instructions of a particular package, you will usually be given the URL to be added to the yum.conf file.

Create a section called *[repository]* just like *[main]* (retain the boxed brackets)

Now add the following lines under this *[repository]* section: *[repositoryid] name=<repository name> baseurl=<address of the site from where the repository can be accessed>*

The repository id an unique word that you can assign to a repository. The *<repository name>* is something you want to call the repository but unlike the *[repositoryid]* it need not be unique. Two or more repositories can be called by the same name.

The address of the site from where you wish to update your packages is defined after *"baseurl="*.

9.4.2 The apt-get Package Manager

Ideally, we should not be saying apt-get package manager as apt itself means Application Package Tool. This is the default manager on Debian GNU/Linux. The commands are somewhat similar to

yum. The commands are as follows:

Installing a new package : ***apt-get install package***

Updating the package list : ***apt-get update***

Update a package : ***apt-get upgrade package***

Uninstalling a package: **apt-get remove package**

Adding Extra URLs: ***/etc/apt/sources.list*** is the file where the address of repositories where packages can be looked for is stored. It is a simple file where sources are stored in each line. The first source(line) is looked for on priority, the second source (line) next and so on. You only need to add to this list the source from where you wish to update your packages.

The format to be followed is:

type uri arguments

Here, ***type*** specifies the packing type, which is usually deb for Debain based distros.

URI! is the Universal Resource Indentifier, somewhat like a collection of URLs

Arguments are options such as stable/unstable which can be specified. In the Debian repository, only those directories specified in the arguments will be searched.

Example, the line ***deb ftp://ftp.debian.org/debian stable contrib,*** added to the sources.list file will look for the deb package in the ftp site ***ftp.debian.org/debian*** in the stable/contrib area.

Not specifying arguments will search the entire package list.

When looking for a package, you will also be find instructions on the updating you may need to do to the sources.list file.

The configurations and commands that can be used with the package managers is very vast and just as powerful. We have presented here only the basic of commands and setting which you may use to update packages on your system. For a detailed look into the other commands and options with the package managers you can use the ***manual*** command.

man yum or

man apt-get

9.5 Kernel Compilation

The title of this section may pop your eyes out! This is one of the best offers that Linux has for its users. Most newbies would wonder if this is necessary. Well, some Linux distributions have kernels specifically designed to run better on Pentium machines.

So, if you have any other processor, and you want to derive better performance out of your box — rebuild the kernel. Kernel compilation is required essentially when certain hardware support is absent; for instance, kernel 2.4.x did not support NTFS partitions by default, but it had the support to read it if enabled. Under such circumstances wherein your PC's peripheral devices are upgraded and a support from the kernel is lacking; rebuilding the same kernel with the support or compiling a new kernel becomes necessary. Compiling a new kernel imparts hardware supports available with the new flavours of Linux and you need not go about installing the entire OS. Note that the looks of the GUI and other software will not change. It is just the kernel that undergoes the changes.

Another point that we want to make here is that Linux is actually the kernel combined with certain utilities to become the OS made by Linus Torvald. The basic Linux OS plus the numerous applications from GNU and Opensource put together, constitutes the Linux Distributions such as Redhat-Fedora, SUSE, Gentoo, etc.

Kernel Compilation or Rebuilding the existing kernel is a huge process. We shall summarise it in point-to-point method and refrain from dragging the topic too far. You will get an idea of the subject and you will also be in a position to research more on the same. In our example, we have used Redhat as our reference distribution.

Necessary files

Kernel sources: Most installations maintain a copy of the source of the kernel is ***/usr/src*** folder. A new kernel will be available at ***www.kernel.org***.

Listed below are the essential utilities and libraries required for

the compilation of the kernel. The steps for installation may vary a little depending on distribution.

Compiler Utilities
gcc - It is a compiler program for C
dev86 - 8086 based processor assembler/linker
cpp - GNU C compiler procesor
make - program to create executables
binutils - Library of binary utilities
Libraries
glibc-devel - Library files for standard header files for C
ncurses-5 - Screen handling package used mainly for CLI
ncurses-devel - for developing application that use ncurses. Needed for menuconfig utility.

Rebuilding Kernel (Redhat)

1. ***cd /usr/src/linux-2.*/***
   Stock kernel—if it exists else jump to 'New Compilation'
2. ***edit Makefile***
   Modify parameter ***EXTRAVERSION=-Rebuild***
3. ***make mrproper***
   Deletes the existing .config
4. ***cp -p /usr/src/linux-2.*/configs/kernel-2.*-i686.config /usr/src /linux-2.*/.config***
5. ***make oldconfig***
   To update .config with running kernel parameters
6. ***make config / menuconfig*** (for Text mode) **/ *xconfig***
7. ***make dep***
8. ***make clean***
9. ***make bzImage***
   Actual compilation process
10. ***cp /usr/src/linux-2.*/arch/i386/boot/bzImage /boot/vmlinuz-2.*-Rebuild***
11. ***cp /usr/src/linux-2.*/System.map /boot/System.map-2.*-Rebuild***
12. ***cp /usr/src/linux-2.*/.config /boot/.config-2.*-Rebuild***
13. Edit boot loader (either lilo.conf or grub.conf) to add entry for the rebuild kernel

/sbin/lilo -v -t
/sbin/lilo -v

14. ***sync***
15. ***make modules***
16. ***make modules_install***
    Installs modules in ***/lib/modules/2.*-Rebuild/***
17. ***reboot***
19. ***uname -r***
    Check. Should give ***2.* -Rebuild*** i.e. your re-built kernel

New Compilation

Assuming that you have downloaded the new kernel follow the steps to compile it on your system. Before you proceed; just in case you have installed Linux drivers for graphic card ATI or NVIDIA. You will have to perform and additional step of editing the ***/etc/X11/xorg.conf***.

Use any editor such as ***gedit*** or ***kate*** to open the file (from terminal switch to root if you are not and open file using ***#gedit /etc/X11/xorg.conf***). Look for the line (Driver "ati"):

Section "Device"
Identifier "ATI Technologies, Inc. Radeon 330M/340M/350M (RS200 IGP)"
Driver "ati"
BusID "PCI:1:5:0"

Note: For NVIDIA the line may be ***Driver "nvidia"***

Substitute the name of the driver with ***"vesa"***
Save the file and exit editor.
Steps to new kernel compilation:
1. Untar the downloaded kernel to ***/usr/src*** directory :
 # tar -zxvf linux-2.*.tar.gz
 # tar -Ixvf linux-2.*.bz2
 # cd linux-2.*
2. ***# make mrproper***
3. ***#make config / menuconfig / xconfig (in xterm)***
4. ***#make dep***

{checking for dependencies}

5. ***#make bzImage***
   {The actual compilation of the kernel begins here. Creates a compressed image of the kernel}
6. ***#make modules&&***
   {building modules}
7. ***#make modules_install*** {installing the modules created in the previous step}
8. Check in ***/usr/src/linux-2.*/arch/i686/boot*** for a file : ***bzImage***
   This will be the gzip-compressed binary file of the new kernel.
9. ***#cp bzImage /boot/vmlinuz-2.****
10. Make necessary changes to the boot loader. For LILO it is as shown below:

image=/boot/vmlinuz-2.*
label=kernel-2.*-NEW
root=/dev/hda5
read-only

Save the above file. And issue ***#lilo -v -t***

title Linux-New-kernel
root (hd0,4)
kernel /vmlinuz-2.* ro root=LABEL=/ rhgb quiet hdc=ide-scsi

9.6 Basic Level Of Security

Why Security?

A better way of saying this will be why not security? As we make our lives more and more dependent on computers and the Internet, the more conscious we have to be on who accesses our system and does what. Someone who manages to gain access into your system may be able to impersonate you. If you have set your browser to remember your passwords to e-mail sites, just think what can happen if malicious e-mail is sent to your business or personal contacts.

If you have your business contacts or project details on your computer, all your brain work can be copied in minutes! Someone with

a sadistic mind can simply make your system crash, needing time and effort to be set it right. Or possibly cause data loss even. We know that security is something we must take seriously, especially if we are talking about offices, small ones even. So we'll leave it at that and tell you what you can do to secure your Linux box.

Firstly, we have to understand that there is no absolute security. If someone is determined to get access to your system, they can, provided it is physically accessible to them. So, the first thing to do is ensure that the computer is physically safe. If you feel like it, just lock it up in a safe. (ok, ok, we were kidding!). On the serious side, try to ensure that not everyone can come and sit on your computer. Especially if you have critical information on it.

Having said this, let us move to different stages of security, starting right from the most basic, the BIOS password setting. Most BIOS allow you to set a 'system password' by which the computer won't even fully boot without a valid password. Every time the computer is started, the user will have to key in the correct password in order to be able to even start the OS. While this sounds nice and safe is not foolproof. If someone has a simple screwdriver, he can manage to reset the BIOS by shorting particular pins on the motherboard or by taking off the BIOS battery for sometime, thereby removing the password.

This brings us to the next line of defense, the user and root password. Of course, you are expected not to keep sharing your password with everyone. A password is private, remember. Now you know that without a password, it is not possible to do any operations. Linux does not even allow you 'inside' without the password. So, this is safety enough, right? Alas no!

There is a known loophole in Linux by which anyone can make a small modification to the boot loader and start the machine in single user mode without any password and gain root privileges. So, how would you combat this loophole? This is explained in the box *'Hardening Single User Mode'*.

Another method to prevent entry to Single User Mode is by

assigning a password to the bootloader (Grub, in most cases today, or LILO, short for Linux Loader—mostly older versions)

To setup Grub such that it asks for a password to edit the boot configuration, do this:

Go to the command shell and type :

md5crypt

When asked for a password, supply your root password, single user mode gives you root privileges.

You will then get an encrypted version of the password you have supplied. Now copy this encrypted password and add it to the grub configuration file, that is ***/etc/grub.conf***

Edit the file using ***vi*** or any editor or a command such as

Cat >> /etc/grub.conf

When you use the above command the shell will wait for your input. Now type "***password –md5"*** followed by a space, just copy paste the encrypted password. Press ***[Enter]***.

When you open the file in any editor, make the ***password –md5 <encrypted password>*** line as the first line and save the file.

Now, when you reboot and try to edit the boot configuration to enter as single user mode, you will be asked for the password that you have just set! (which we have suggested you keep the same as root password, but you can have any password actually).

So far, so good. We have seen how to setup a BIOS password and prevent unauthorised single user entry. But what if someone boots your computer with some other device? Like say, an attacker using his own CD-ROM or hard disk to boot your machine... he can use a Live CD or, mount the file system and make alterations. How would you prevent this? Well, advanced system administrators would store their password configuration files on a server or in the desktop itself, but not in the obvious default location. But we are not addressing people who know computers better than the back of their hands.

As a common computer user, you can password protect your sensitive documents. OpenOffice.org provides an option to password

protect your documents, presentations and spreadsheets. This way, even if someone gains access to your computer, your personal files cannot be opened easily. If your password is strong enough, it would take at least a year for today's most powerful computer to break it.

How To Choose A Good Password

This section applies to any password(s) used anywhere, Linux or otherwise. Firstly, don't have a password that your friends or even acquaintances can guess. That leaves out your own name, your dog's name, the name of your city, etc. Use something uncommon and something only you can think of, or know how the password is arrived at. A common approach is to use a sentence that you can remember well (but not others!) and take the first letter from every word and make the password.

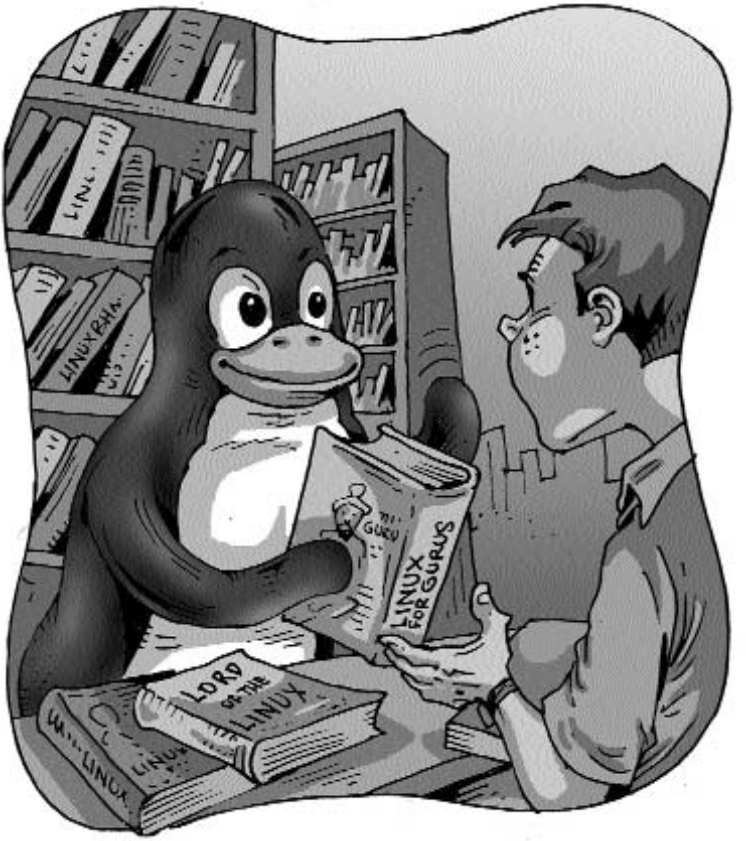
There are password cracking software for files created for Microsoft Office, Winzip, etc. They use a brute force method by which all possible combinations of characters are tested on the file to see if it matches the set password. Obviously, more the number of characters and the types of characters, the more difficult it becomes for the cracking software to crack the password.

- Avoid using English (dictionary words)
- Use many characters in your password.
- Use numbers randomly in the password.
- Capital letters (upper case) can be used in part.
- Special characters such as '@' '$' '^' may also be used if the software permits it.

This more or less covers the basics of what you need to know about securing your PC from attackers who may get physical access. Of course, the best way to secure it is to prevent physical access if at all possible.

In the next section we will see how you can prevent unauthourised access to your PC through the internet. We shall briefly look at firewalls and IP filtering.

Resources



The aim of this little book has been to introduce Linux to you if you've not had any previous experience with it, and what you've seen thus far is just the very basics. If you wish to know more about Linux, here's a guide pointing you to more resources. In addition, we list out user groups in India that you can follow and even join. Linux, after all, is a community-driven idea.

If you haven't already explored all the menus and sub-menus in your distro, do it now! Trial and Error is one of the best ways to learn a GUI. Exploring the layout of your OS is a nice way to start knowing more about what you can do with your system. Most distributions, especially the major ones, have a pretty intuitive interface—the menu system more or less leads you to the shortcut to the application you are looking for, with indicative menu names. If you are looking for OpenOffice.org's Writer, for example, the place to look for would be under the "Office" menu.

We also suggest you read the manual / help files that come with your distribution. When you go to the Help menu, take at least a cursory look at all the topics covered, so you get the basics.

10.1 Information Sources

Linux, we repeat, is community-oriented. There are a lot of resources on the Internet in the form of user groups, forums, articles, white papers and IRC networks. To check on the latest releases and news, check the Web sites of the distributor of your flavour of Linux. Usually, you will find in-depth documentation on the sites—you need to have the patience to read it, but once you do, you will get to know a lot about not only your distro, but Linux in general as well.

At the official Web site, you will also find "Release Notes," which usually contain a list of supported hardware and known issues. Minimum and recommended system requirements are contained in the Release Notes. Usually, with the minimum requirements, the software would just about run: we suggest you treat the *recommended* system requirements as the minimum, and the more you have, the better.

Here is a listing of the official Web sites of the most prominent Linux distributors:

Fedora: http://fedora.redhat.com/
SuSe: www.novell.com/linux/suse
OpenSuse: http://en.opensuse.org
Mandriva: http://wwwnew.mandriva.com
Ubuntu: www.ubuntu.com
Gentoo: www.gentoo.org
Knoppix: www.knoppix.org

There are several "community" maintained sites that offer How-tos, guides and general information on Linux. It is impossible to list all these, but here are some useful ones.

Fultus.com e-library

http://elibrary.fultus.com/technical/index.jsp

This is an online repository of a vast collection of topics relating to almost everything Linux. Right from information on distributions all the way to an all-encompassing dictionary on Linux, this site has it all. The general Linux documentation on this site is common with The Linux Documentation Project (www.tldp.org), which itself has a Web site with a large collection of articles from authors across the world. But navigation is problematic at the TLDP Web site, and that's why we recommend Fultus. It has a nice interface: documents are organised according to topics. There are How-tos, Guides, distro-specific topics, and more. The How-tos are very detailed, and are aimed at users already familiar with the basics. The Guides are somewhat simpler.

The GNU FAQ

www.gnu.org/gnu/gnu-linux-faq.html

We have spoken earlier about how the GNU project has been very important in the development of Linux. This FAQ lists Q&A pertaining to GNU and Linux, besides giving you a general idea about the GPL. If you wish to find out more about the concepts of the GNU and how it relates to Linux, this is a must-visit Web site.

Linux Command Directory from Linux Dev Center
www.linuxdevcenter.com/linux/cmd/

This is site maintained by the O'Reilly Network: it contains articles and How-tos on various Linux topics. There is an alphabetical listing of Linux commands. Clicking on a command will take you to a page that explains how the command is to be used—pretty handy!

Faqs.org
www.faqs.org/faqs/linux/

A lot of FAQs here. The layout may not be too impressive, but the content is vast.

Computing.net
http://computing.net/linux/wwwboard/wwwboard.html

One of the oldest forums on the Internet, computing.net has active forums on almost every computer-related subject. You can view threads, but you need to register to be able to post. The forum search feature is good—if what you're looking for has been discussed, you can simply view the thread rather than waiting for a response by posting.

Linuxquestions.org
www.linuxquestions.org

This is a Linux forum where you can post your queries and participate in discussions. You need to register before you can start posting. There is also a Web site on linuxquestions.org, where data from Wikipedia is collated. Well-categorised and easy to navigate, this site contains not only concepts but also specific topics such as music players. You can find the site at http://wiki.linuxquestions.org/wiki

Wikipedia
http://en.wikipedia.org/wiki/Linux

We can't leave out wikipedia, can we? The data on the Linux wiki simply seems to be growing every day.

Google's Linux Search
www.google.com/linux

That part of Google dedicated to Linux alone!

10.2 Linux User Groups

There are localised groups of Linux users all across the globe. Generally, groups have mailing lists, through which members keep in touch and exchange ideas or ask questions. They even meet at intervals, organise seminars and workshops, and more.

In India, there are Linux User Groups in almost all the major cities. Most of them are Yahoo! groups that you can join. Follow the activity on the group for some time, and then introduce yourself and start posting!

Visit www.linux-india.org and click on the "User Groups" button—you will be taken to a listing of Linux groups in India. Apart from the groups mentioned on that Web site, here are some others in different cities:

Bangalore: http://blug.in
Delhi: www.linux-delhi.org
Mumbai: www.ilug-bom.org.in
Pune: www.plug.org.in
Trivandrum: http://triglug.sarovar.org