Chapter 1

Duties of the System Administrator

IN THIS CHAPTER

- The Linux system administrator
- Installing and configuring servers
- Installing and configuring application software
- Creating and maintaining user accounts
- Backing up and restoring files
- Monitoring and tuning performance
- Configuring a secure system
- Using tools to monitor security

LINUX IS A MULTIUSER, multitasking operating system from the ground up, and in this regard the system administrator has flexibility — and responsibility — far beyond those of other operating systems. Now, Red Hat has employed innovations that extend these duties even for the experienced Linux user. In this chapter, we look at those requirements.

The Linux System Administrator

Linux involves much more than merely sitting down and turning on the machine. Often you hear talk of a "steep learning curve," but that discouraging phrase can be misleading. Instead, Linux is quite different from the most popular commercial operating systems in a number of ways, and while it is no more difficult to learn than other operating systems, it is likely to seem very strange even to the experienced administrator of some other system. In addition, the sophistication of a number of parts of the Red Hat Linux distribution has increased by an order of magnitude, so even an experienced Linux administrator is likely to find much that is new and unfamiliar. Fortunately, there are new tools designed to make system administration easier than it has ever been before.

Make no mistake: Every computer in the world has a system administrator. It may be — and probably is — that the majority of system administrators are probably those who decided what software and peripherals were bundled with the machine when it was shipped. That status quo remains because the majority of users who acquire computers for use as appliances probably do little to change the default values. But the minute a user decides on a different wallpaper image or adds an application that was acquired apart from the machine itself, he or she has taken on the mantle of system administration.

Such a high-falutin' title brings with it some responsibilities. No one whose computer is connected to the Internet, for instance, has been immune to the effects of poorly administered systems, as demonstrated by the Distributed Denial of Service (DDoS) and e-mail macro virus attacks that have shaken the online world in recent years. The scope of these acts of computer vandalism (and in some cases computer larceny) would have been greatly reduced if system administrators had a better understanding of their duties.

The Linux system administrator is more likely to understand the necessity of active system administration than are those who run whatever came on the computer, assuming that things came from the factory properly configured. The user or enterprise that decides on Linux has decided, too, to assume the control that Linux offers, and the responsibilities that this entails.

By its very nature as a modern, multiuser operating system, Linux requires a degree of administration greater than that of less robust home market systems. This means that even if you are using a single machine connected to the Internet by a dial-up modem — or not even connected at all — you have the benefits of the same system employed by some of the largest businesses in the world, and will do many of the things that the IT professionals employed by those companies are paid to do. Administering your system does involve a degree of learning, but it also means that in setting up and configuring your own system you gain skills and understanding that raise you above mere "computer user" status. The Linux system administrator does not achieve that mantle by having purchased a computer but instead by having taken full control of what his or her computer does and how it does it.

You may end up configuring a small home or small office network of two or more machines, perhaps including ones that are not running Linux. You may be responsible for a business network of dozens of machines. The nature of system administration in Linux is surprisingly constant, no matter how large or small your installation. It merely involves enabling and configuring features you already have available.

By definition, the Linux system administrator is the person who has "root" access, which is to say the one who is the system's "super user" (or root user). A standard Linux user is limited as to the things he or she can do with the underlying engine of the system. But the "root" user has unfettered access to everything — all user accounts, their home directories, and the files therein; all system configurations; and all files on the system. A certain body of thought says that no one should ever log in as "root," because system administration tasks can be performed more easily and safely through other, more specific means, which I discuss in due course.

The system administrator has full system privileges, so the first duty is to know what you're doing lest you break something.

NOTE

By definition, the Linux system administrator is the person who has "root" access, which is to say the one who is the system's "super user."

The word "duties" implies a degree of drudgery; in fact, they're a manifestation of the tremendous flexibility of the system measured against responsibility to run a tight installation. These duties do not so much constrain the system administrator as free him or her to match the installation to the task. But all are likely employed to some degree in every system. Let's take a brief look at them.

Installing and Configuring Servers

In the Linux world, the word "server" has a meaning that is broader than you might be used to. For instance, the standard Red Hat Linux graphical user interface (GUI) requires a graphical layer called XFree86. This is a server. It runs even on a standalone machine with one user account. It must be configured. (Fortunately, Red Hat Linux has made this a simple and painless part of installation on all but the most obscure combinations of video card and monitor; gone are the days of anguish configuring a graphical desktop.)

Likewise, printing in Linux takes place only after you have configured a print server. Again, this has become so easy as to be nearly trivial.

In certain areas the client-server nomenclature can be confusing, though. While you cannot have a graphical desktop without a server, you can have World Wide Web access without a Web server, file transfer protocol (FTP) access without running an FTP server, and Internet e-mail capabilities without ever starting a mail server. You may well want to use these servers, all of which are included in Red Hat Linux, but then again you may not. And whenever a server is connected to other machines outside your physical control, there are security implications — you want users to have easy access to the things they need, but you don't want to open up the system you're administering to the whole wide world.

TIP

Whenever a server is connected to machines outside your physical control, security issues arise. You want users to have easy access to the things they need, but you don't want to open up the system you're administering to the whole wide world.

Linux distributions used to be shipped with all imaginable servers turned on by default. This was a reflection of an earlier, more polite era in computing, when people did not consider vandalizing other people's machines to be good sport. But the realities of a modern, more dangerous world have dictated that all but essential servers are off unless specifically enabled and configured. This duty falls to the system administrator. You need to know what servers you need and how to employ them, and to be aware that it is bad practice and a potential security nightmare to enable services that the system isn't using and doesn't need. Fortunately, the following pages show you how to carry out this aspect of system administration easily and efficiently.

Installing and Configuring Application Software

This may seem redundant, but it's crucial that the new Linux system administrator understand two characteristics that set Linux apart from popular commercial operating systems: The first is the idea of the root or super user, and the second is that Linux is a multiuser operating system. Each user has (or shares) an account on the system, be it on a separate machine or on a single machine with multiple accounts.

One reason that these concepts are crucial is found in the administration of application software — productivity programs.

While it is possible for individual users to install some applications in their home directories — drive space set aside for their own files and customizations — these applications are not available to other users without the intervention of the system administrator. Besides, if an application is to be used by more than one user, it probably needs to be installed higher up in the Linux file hierarchy, which is a job that can be performed by the system administrator only. (The administrator can even decide which users may use which applications by creating a "group" for that application and enrolling individual users into that group.)

New software packages might be installed in `/opt`, if they are likely to be upgraded separately from the Red Hat Linux distribution itself; by so doing, it's simple to retain the old version until you are certain the new version works and meets expectations. Some packages may need to go in `/usr/local` or even `/usr`, if they are upgrades of packages installed as part of Red Hat Linux. (For instance, there are sometimes security upgrades of existing packages.) The location of the installation usually matters only if you compile the application from source code; if you use a Red Hat Package Manager (RPM) application package, it automatically goes where it should.

Configuration and customization of applications is to some extent at the user's discretion, but not entirely. "Skeleton" configurations — administrator-determined default configurations — set the baseline for user employment of applications. If there are particular forms, for example, that are used throughout an enterprise, the system administrator would set them up or at least make them available by adding them to the skeleton configuration. The same applies, too, in configuring user desktops and in even deciding what applications should appear on user desktop menus. Your company may not want the games that ship with modern Linux desktops to be available to users. And you may want to add menu items for newly installed or custom applications. The system administrator brings all this to pass.

Creating and Maintaining User Accounts

Not just anyone can show up and log on to a Linux machine. An account must be created for each user and — you guessed it — no one but the system administrator may do this. That's simple enough.

But there's more, and it involves decisions that either you or your company must make. You might want to let users select their own passwords, which would no doubt make them easier to remember, but which probably would be easier for a malefactor to crack. You might want to assign passwords, which is more secure in theory but which increases the likelihood that users will write them down on a conveniently located scrap of paper — a risk if many people have access to the area where the machine(s) is located. You might decide that users must change their passwords periodically, and you can configure Red Hat Linux to prompt users to do so.

And what to do about old accounts? Perhaps someone has left the company. What happens to his or her account? You probably don't want him or her to continue to have access to the company network. On the other hand, you don't want to simply delete the account, perhaps to discover later that essential data resided nowhere else.

To what may specific users have access? It might be that there are aspects of your business that make World Wide Web access desirable, but you don't want everyone spending their working hours surfing the Web. If your system is at home, you may wish to limit your children's access to the Web, which contains sites to which few if any parents would want their children exposed.

These issues and others are parts of the system administrator's duties in managing user accounts. Whether the administrator or his or her employer establishes the policies governing them, those policies should be established — if in an enterprise, preferably in writing — for the protection of all concerned.

Backing Up and Restoring Files

Until equipment becomes absolutely infallible, and until people lose their desire to harm the property of others (and, truth be known, until system administrators become perfect), there is a need to back up important files so that in the event of a failure of hardware, security, or administration, the system can be up and running again with minimal disruption. Only the system administrator may do this.

(Because of its built-in security features, Linux may not allow users to be able even to back up their own files to floppy disks.)

Again, knowing that file backup is your job is not enough. You need to formulate a strategy for making sure your system is not vulnerable to catastrophic disruption. And it's not always obvious. If you have a high-capacity tape drive and several good sets of restore diskettes, you might make a full system backup every few days. If you are managing a system with scores of users, you might find it more sensible to back up user accounts and system configuration files, figuring that reinstallation from the distribution CDs would be quicker and easier than getting the basics off a tape archive. (Don't forget the applications you've installed separate from your Red Hat Linux distribution, especially including anything heavily customized!)

Once you've decided *what* to back up, you need to decide *how frequently* you want to perform backups and whether you wish to maintain a series of incremental backups — adding only the files that have changed since the last backup — or multiple full backups, and *when* these backups are to be performed — do you trust an automated, unattended process? Or, if you have input as to the equipment used, do you want to use a redundant array of independent disks, or RAID, which is to say multiple hard drives all containing the same data as insurance against the failure of any one of them, in addition to other backup systems. (A RAID is not enough, because hard drive failure is not the only means by which a system can be brought to a halt.)

Conversely, you do not want to become complacent or to foster such an attitude among users. Part of your strategy should be the maintenance of perfect backups without ever needing to resort to them. This means encouraging users to keep multiple copies of their own important files, all in their home directories, so that you are not being asked to mount a backup so as to restore a file that a user has corrupted. (And if the system is stand-alone, you as your own system administrator might want to make a practice of backing up configuration and other important files.)

The chances are that even if you're working for a company, you'll make these decisions — all your boss wants is a system that works perfectly, all the time. Backing up is only half the story, too. You need to formulate a plan for bringing the system back up in the event of a failure. Such a plan extends to areas outside the scope of this book. Sometimes hardware failures are so severe that the only solution is replacing the hard drive, replacing everything *except* the hard drive, or even restoring from backup to a whole new machine.

TIP

Backing up is only half the story. You need to formulate a plan for bringing the system back up in the event of a failure.

Monitoring and Tuning Performance

The default installation of Red Hat Linux goes a long way toward capitalizing on existing system resources. But there is no “one size fits all” configuration, and Linux is infinitely configurable or close to it.

On a modern stand-alone system, Linux is going to be pretty quick, and if it isn’t, there’s something wrong — something that is up to the system administrator to fix. But you might want to squeeze that one last little bit of performance out of your hardware. Or you might have a number of people using the same fileserver, mail server, or other shared machine, in which case seemingly small improvements in system performance can mean a lot.

System tuning is an ongoing process aided by a variety of diagnostic and monitoring tools. Some performance decisions are made at installation time, while others are added or tweaked later. A good example is the use of the `hdparm` utility, which can increase throughput in IDE drives considerably — but for some high-speed modes a check of system logs will show that faulty or inexpensive cables can, in combination with hdparm, produce an enormity of nondestructive but system-slowing errors.

Proper monitoring allows you to detect a misbehaving application that might be consuming more resources than it should or failing to exit completely on close. Through the use of system performance tools you can determine when hardware — such as memory, added storage, or even something as elaborate as a hardware RAID — should be upgraded for more cost-effective use of a machine in the enterprise or for complicated computational tasks such as three-dimensional rendering.

Possibly most important, careful system monitoring and diagnostic practices give you an early heads-up when a system component is showing early signs of failure, so that any potential downtime can be minimized. Combined with the resources for determining which components are best supported by Red Hat Linux, performance monitoring can result in replacement components which are far more robust and efficient in some cases.

And in any case, careful system monitoring plus wise use of the built-in configurability of Linux allows you to squeeze the best possible performance from your existing equipment, from customizing video drivers to applying special kernel patches to simply turning off unneeded services to free memory and processor cycles.

TIP

To squeeze the best performance from your equipment, monitor your system carefully and use Linux’s built-in configurability wisely.

Configuring a Secure System

If there is a common thread in Linux system administration, something that is a constant presence in everything you do, it is the security of the computer and data integrity.

What does this mean? Well, just about everything. The system administrator's task, first and foremost, is to make certain that no data on the machine or network are likely to become corrupted, whether by hardware or power failure, by misconfiguration or user error (to the extent that the latter can be avoided), or by malicious or inadvertent intrusion from elsewhere. It means doing all the tasks described throughout this chapter well and with a full understanding of their implication, and it means much more.

No one involved in computing can have failed to hear of the succession of increasingly serious attacks upon machines connected to the Internet. The majority of these have not targeted Linux systems, but that doesn't mean that Linux systems have been entirely immune, either to direct attack or to the effects of attacks on machines running other operating systems. In one Distributed Denial of Service (DDoS) attack aimed at several major online companies, many of the "zombie" machines — those which had been exploited so that the vandals could employ thousands of machines instead of just a few — were running Linux that had not been patched to guard against a well-known security flaw. In the various "Code Red" attacks of the summer of 2001, Linux machines themselves were invulnerable, but the huge amount of traffic generated by this "worm" infection nevertheless prevented many Linux machines from getting much Web-based work done for several weeks, so fierce was the storm raging across the Internet. And few Internet e-mail users have gone without receiving at least some "SirCam" messages — nonsensical messages from strangers with randomly selected files from the strangers' machines attached. While this infection did not corrupt Linux machines as it did those running a different operating system, anyone on a dial-up connection who had to endure the download of several megabytes of infected mail would scarcely describe himself or herself as unaffected by the attack.

Depending on how and to what a Linux machine is connected, the sensitivity of the data it contains and the uses to which it is put, security can be as simple as turning off unneeded services, monitoring the Red Hat Linux security mailing list to make sure that all security advisories are followed, and otherwise engaging in good computing practices to make sure the system runs robustly. Or it can be an almost full-time job involving levels of security permissions within the system and systems to which it is connected, elaborate firewalling to protect not just Linux machines but machines that, through their use of non-Linux software, are far more vulnerable, and physical security — making sure no one steals the machine itself!

For any machine that is connected to any other machine, security means hardening against attack and making certain that no one is using your machine as a platform for launching attacks against others. If you are running Web, ftp, or mail servers, it means giving access to those who are entitled to it while locking out everyone else. It means making sure that passwords are not easily guessed and not made available to unauthorized persons, that disgruntled former employees no longer have access to the system, and that no unauthorized person may copy files from your machine or machines.

Security is an ongoing process — it has been said that the only really secure computer is one that contains no data and that is unplugged from networks and even power supplies, has no keyboard attached, and resides in a locked vault. While that is theoretically true, it also implies that security diminishes the usefulness of the machine, don't you think? So your job as a system administrator is to strike just the right balance between maximum utility and maximum safety, all the while bearing in mind that confidence in a secure machine today says nothing about the machine's security tomorrow.

In the pages that follow, you'll learn about the many tools that Red Hat Linux provides to help you guard against intrusion, even to help you prevent intrusion into non-Linux machines that may reside on your network. Linux is designed from the beginning with security in mind, and in all of your tasks you should maintain that same security awareness.

TIP

Your job as a system administrator is to strike the right balance between maximum utility and maximum safety, all the while bearing in mind that confidence in a secure machine today says nothing about the machine's security tomorrow.

Using Tools to Monitor Security

Crackers — people who, for purposes of larceny or to amuse themselves, like to break into other people's computers — are a clever bunch. If there is a vulnerability in a system, they will find it. Fortunately, the Linux development community is quick to find potential exploits and to find ways of slamming shut the door before crackers can enter. Fortunately, too, Red Hat is diligent in making available new, patched versions of packages in which potential exploits have been found. So your first and best security tool is making sure that whenever a security advisory is issued, you download and install the repaired package. This line of defense can be annoying, but it is nothing compared to rebuilding a compromised system.

And as good as the bug trackers are, sometimes their job is reactive. Preventing the use of your machine for nefarious purposes and guarding against intrusion are, in the end, your responsibility alone. Again, Red Hat Linux equips you with tools to detect and deal with unauthorized access of many kinds. As this book unfolds, you'll learn how to install and configure these tools and how to make sense of the warnings they provide. Pay careful attention to those sections and do what they say. If your machine is connected to the Internet, you will be amazed at the number of attempts that are made to break into your machine. And you'll be struck by how critical an issue security is.

Summary

As you, the system administrator, read this book, bear in mind that your tasks are ongoing and that there is never a machine that is completely tuned, entirely up-to-date, and utterly secure for very long. The pace of Linux development is breathtaking, so it's important that you keep current in the latest breakthroughs. This book gives you the very best information as to the Red Hat Linux distribution you're using and tells you all you need to know about getting the most from it. But more than that, you should read it with an eye toward developing a Linux system administrator's point of view, an understanding of how the system works as opposed to the mere performance of tasks. As the best system administrators will tell you, system administration is a state of mind.