# Chapter 1

# An Overview of CentOS



The goal of the CentOS project (`www.centos.org`) is to produce an enterprise-class Linux operating system distribution. Like most Linux distributions, CentOS leverages the work done by thousands of software developers around the world that release their software under free and Open Source (FOSS) licenses.

But unlike other Linux distributions, CentOS distinguishes itself not by how it is different, but by how it strives to be the same.

The CentOS Linux distribution includes software from many of the most respected and mature projects in the Open Source world. If you want to use CentOS as a server, you can take advantage of server software in CentOS that includes:

- **Apache Web Server** (`http://httpd.apache.org`) — The most popular HTTP server in the world
- **Samba** (`www.samba.org`) — A suite of applications used for sharing files, printers, and related information using protocols that are native to Windows, OS/2, and other PC-based systems
- **Sendmail** (`www.sendmail.org`) — An e-mail server that lets you send and store e-mail that can be accessed using a variety of e-mail clients
- **CUPS** (`www.cups.org`) — The Common UNIX Printing System includes software for configuring print servers.
- **vsFTPd** (`http://vsftpd.beasts.org`) — A File Transfer Protocol (FTP) server that lets users upload and download files over a network
- **MySQL** (`www.mysql.com`) — A multiuser SQL database server

## IN THIS CHAPTER

- **BIND** (`www.isc.org/products/BIND`) — The Berkeley Internet Name Domain (BIND) server that implements the Domain Name System (DNS) protocols to resolve hostnames to IP addresses on the Internet (or similar networks)

As for desktop client configurations, CentOS includes features that are appropriate for corporate desktops, but could also be useful for personal or small-office desktops. Those features include:

- **GNOME Desktop Environment —** GNOME is the standard desktop environment included with CentOS.
- **K Desktop Environment (KDE) —** KDE is another popular desktop environment that comes with CentOS.
- **X Window System —** Provides the framework on which the graphical user interface is supported.
- **Firefox —** The most popular Open Source Web browser (produced by the Mozilla project)
- **Thunderbird —** A client program for sending, receiving, composing and otherwise managing e-mail that, like Firefox, comes from the Mozilla project
- **OpenOffice.org Office Suite —** A full suite of office applications for working with documents, spreadsheets, presentations, and other personal productivity tasks
- **GNU Image Manipulation Program —** A powerful graphics editing application
- **Rhythmbox —** Audio application for playing and managing your music and other audio files

The server and desktop applications just mentioned are only a few of the software projects available with CentOS. While those are powerful and useful software projects, most other popular Linux distributions also include those features.

What you find in CentOS that you don't find in many Linux systems, particularly those geared toward personal or small-office use, are software packages for cluster computing and virtualization.

By configuring computers to work together in clusters, a group of servers can share a common file system, offer high-availability applications, load-balance to make best use of computing resources, and greatly improve performance. Clustering and other advanced features in CentOS include:

- **Global File System Suite (GFS) —** With GFS, multiple computers in a cluster can share a single, consistent file system namespace.
- **Management of High-Availability Services —** If a node in a cluster becomes inoperative, tools for managing high-availability services can move those services to another node.
- **Linux Virtual Server Routing —** Provides Internet Protocol load-balancing to distribute client requests evenly among server systems.

- **Linux Virtual Server Administration** — Xen virtualization tools are available to create, run, and manage virtual machines (guest operating systems) that run natively in CentOS.

While descriptions of some of these Enterprise-class features are beyond the scope of this book, detailed documentation on using these projects is included with the CentOS system itself. But finding out how those features work is only part of the trick.

Integrating enterprise-class software so all the pieces can work seamlessly together in financial institutions and government agencies for their mission-critical applications is a daunting task. That is why the CentOS project's own mission is to not only take advantage of the value produced from individual Open Source software projects, but also to leverage the work that was done integrating these many projects.

In short, the source code that CentOS uses to build its operating system doesn't come directly from individual software projects. Instead, it comes from rebuilding source code made available from a "prominent North American Enterprise Linux vendor," as the CentOS web site notes. That company is Red Hat, Inc.

## What You Get with CentOS

Some people think that Red Hat, Inc. is somehow associated with the CentOS project. It is not.

Projects such as the Apache project, Samba project, and GNU project create code that is licensed as Open Source and publicly released. Just as that code can be built into binaries and included in a Linux distribution (more on how that works later), so too can the source code subsequently released by each Linux distribution be rebuilt and reused.

CentOS takes the freely available Red Hat Enterprise Linux source code (not the binaries), removes trademark and branding information that identifies it as a Red Hat product, then rebuilds and rebrands that source code as CentOS. The results are:

- A Linux distribution that includes most of the same Open Source software projects contained in Red Hat Enterprise Linux
- A set of packages and ISO images, for installing CentOS or running it as a live CD, that can be freely distributed without paying subscription fees
- A dedicated, if small, band of developers and an active community of supporters that are accessible through forums, IRC chat, and mailing lists through the `CentOS.org` site

Although CentOS doesn't achieve 100 percent compatibility with Red Hat Enterprise Linux, it does strive to do so. For some people who just want to use CentOS to try out Linux or are willing to take full responsibility for the quality of that system in their business, CentOS can suit their needs.

## Stepping Stone to Red Hat Enterprise Linux

Some people use CentOS as a learning tool to become familiar with features that are in Red Hat Enterprise Linux. Once they are ready to make a serious commitment to a commercial use of Linux, however, they often buy subscriptions to Red Hat Enterprise Linux. In professional settings, there is a lot you get with Red Hat Enterprise Linux that you don't get with CentOS:

- **The Backing of a Major Linux Vendor —** If something goes wrong with a Red Hat Enterprise Linux system, your boss can contact Red Hat. If something goes wrong with CentOS, your boss calls you. (Of course, you might get called in either case, but with Red Hat, there's a commercial vendor behind you.)
- **Technical Support —** While CentOS offers forums and mailing lists to support its users, Red Hat offers full technical support programs that range from timely software updates and Web-based support through Red Hat Network to full 24 × 7 onsite support. By adding Satellite Servers to their locations, customers can centrally manage and monitor all their Red Hat Enterprise Linux systems on site.
- **Certified Hardware and Software —** With each release of Red Hat Enterprise Linux, dozens of hardware manufacturers (OEMs) and independent software vendors (ISVs) certify that their products work with that release. There are no such guarantees with CentOS. In fact, at least some of these vendors don't respond to bug reports of their products on CentOS systems.
- **Timely Security Patches and Updates —** The CentOS project strives to keep up with updates and security patches as they are released in Open Source from Red Hat, Inc. However, there will always be some delay (even if a short one) between when a patch comes out from Red Hat and when it can be rebuilt to be used for CentOS. And, of course, there will be some lag time after a new Red Hat release before a CentOS version can be built and made available.
- **Certification and Training —** If you are looking to become a Linux professional, there is no official training and certification with CentOS. Red Hat, Inc. offers professional certification for systems administrators, such as the Red Hat Certified Engineer and Red Hat Certified Technician certifications (more on that later in this chapter).

In short, using the CentOS operating system and learning about it using the documentation in this book or other places can provide a good value if your goal is to become a Linux professional or try out enterprise-class software. However, before committing your business or organization to CentOS, you should weigh your needs for stability and accountability for those systems against any cost savings it might bring.

So far in describing what CentOS is and is not, we have neglected a few critical issues. For example, we haven't talked about where Linux itself came from and how the free and Open Source software model works. The next sections address those issues.

# What Is Linux?

Linux is a free operating system that was created by Linus Torvalds when he was a student at the University of Helsinki in 1991. Torvalds started Linux by writing a *kernel* — the heart of the operating system — partly from scratch and partly by using publicly available software. (For the definition of an operating system and a kernel, see the sidebar "What Is an Operating System?" later in this chapter.) Torvalds then released the system to his friends and to a community of "hackers" on the Internet and asked them to work with it, fix it, and enhance it. It took off.

NOTE **I make the distinction here between *hackers* (who just like to play with computers) and *crackers* (who break into computer systems and cause damage).**

Today, there are thousands of software developers around the world contributing software to the free and Open Source software (FOSS) community that feeds the Linux initiative. Because the source code for the software is freely available, anyone can work on it, change it, or enhance it. Developers are encouraged to pass their fixes and improvements back into the community so that Linux can continue to grow and improve.

On top of the Linux kernel effort, the creators of Linux also drew on a great deal of system software and applications that are now bundled with Linux distributions from the GNU project (*GNU* stands for "GNU is not UNIX"), which is directed by the Free Software Foundation (`www.gnu.org`). There is a vast amount of software that can be used with Linux, making it an operating system that can compete with or surpass features available in any other operating system in the world.

If you have heard Linux described as a free version of UNIX, there is good reason for it. Although much of the code for Linux started from scratch, the blueprint for what the code would do was created to follow POSIX (Portable Operating System Interface for UNIX) standards. POSIX is a computer industry operating system standard that every major version of UNIX complied with. In other words, if your operating system was POSIX-compliant, it was UNIX. Today, Linux has formed its own standards and services organizations to help interoperability among Linux systems, including the Linux Foundation, which supports such efforts as the Linux Standard Base (`www.linux-foundation.org/en/LSB`).

# Linux's Roots in UNIX

Linux grew within a culture of free exchange of ideas and software. Like UNIX — the operating system on which Linux is based — the focus was on keeping communications open among software developers. Getting the code to work was the goal, and the Internet was the primary communications medium. Keeping the software free and re-distributable was a means to that goal. What, then, were the conditions that made the world ripe for a computer system like Linux?

In the 1980s and 1990s, while Microsoft flooded the world with personal computers running DOS (Disk Operating System) and Windows operating systems, power users demanded more from an operating system. They ached for systems that could run on networks, support many users at once (multiuser), and run many programs at once (multitasking). DOS and Windows didn't cut it.

UNIX, on the other hand, grew out of a culture in which technology was king and marketing people were, well, hard to find. Bell Laboratories in Murray Hill, New Jersey, was a think tank where ideas came first and profits were somebody else's problem. A quote from Dennis Ritchie, co-creator of UNIX and designer of the C programming language, in a 1980 lecture on the evolution of UNIX, sums up the spirit that started UNIX. He was commenting on both his hopes and those of his colleagues for the UNIX project after a similar project called *Multics* had just failed:

> *What we wanted to preserve was not just a good environment in which to do programming, but a system around which a fellowship could form. We knew from experience that the essence of communal computing as supplied by remote-access, time-shared machines, is not just to type programs into a terminal instead of a keypunch, but to encourage close communication.*

In that spirit, the first source code of UNIX was distributed free to universities. Like Linux, the availability of UNIX source code made it possible for a diverse population of software developers to make their own enhancements to UNIX and share them with others.

## What Is an Operating System?

An operating system is made up of software instructions that lie between the computer hardware (disks, memory, ports, etc.) and the application programs (word processors, Web browsers, spreadsheets, etc.). At the center is the *kernel*, which provides the most basic computing functions (managing system memory, sharing the processor, opening and closing devices, etc.). Associated with the kernel are a variety of basic services needed to operate the computer, including:

- **File Systems** — The file system provides the structure in which information is stored on the computer. Information is stored in files, primarily on hard disks inside the computer, but also on removable media such as CDs and DVDs. Files are organized within a hierarchy of directories. The Linux file system holds the data files that you save, the programs you run, and the configuration files that set up the system.
- **Device Drivers** — These provide the interfaces to each of the hardware devices connected to your computer. A device driver enables a program to write to a device without needing to know details about how each piece of hardware is implemented. The program opens a device, sends and receives data, and closes a device.

*continued*

*continued*

- **User Interfaces** — An operating system needs to provide a way for users to run programs and access the file system. Linux has both graphical and text-based user interfaces. GNOME and KDE provide graphical user interfaces, whereas shell command interpreters (such as `bash`) run programs by typing commands and options.
- **System Services** — An operating system provides system services, many of which can be started automatically when the computer boots. In Linux, system services can include processes that mount file systems, start your network, and run scheduled tasks. In Linux, many services run continuously, enabling users to access printers, web pages, files, databases, and other computing assets over a network.

Without an operating system, an application program would have to know the details of each piece of hardware, instead of just being able to say, ''open that device and write a file there.''

By the early 1980s, UNIX development moved from the organization in Murray Hill to a more commercially oriented development laboratory in Summit, New Jersey (a few miles down the road). During that time, UNIX began to find commercial success as the computing system of choice for applications such as AT&T's telephone switching equipment, for supercomputer applications such as modeling weather patterns, and for controlling NASA space projects.

Major computer hardware vendors licensed the UNIX source code to run on their computers. To try to create an environment of fairness and community to its OEMs (original equipment manufacturers), AT&T began standardizing what these different ports of UNIX had to be able to do to still be called *UNIX*. To that end, POSIX standards and the AT&T UNIX System V Interface Definition (SVID) were specifications that UNIX vendors could use to create compliant UNIX systems. Those same documents also served as road maps for the creation of Linux.

Elsewhere, the UNIX source code that had been distributed to universities had taken on a life of its own. The Berkeley Software Distribution (BSD) began life in the late 1970s as patches to the AT&T UNIX source code from students and staff at the University of California at Berkeley. Over the years, the AT&T code was re-written and BSD became freely distributed, with offshoot projects such as FreeBSD, OpenBSD, and NetBSD still available.

Linux has been described as a UNIX-like operating system that reflects a combination of SVID, POSIX, and BSD compliance. Linux continues to aim toward POSIX compliance, as well as compliance with standards set by the new owner of the UNIX trademark, The Open Group (`www.unix.org`). Much of the direction of Linux today comes from the Linux Foundation (`www.linux-foundation.org`), which was founded in 2007 by a merger of the Free Standards Group and the Open Source Development Labs.

# Common Linux Features

No matter what version of Linux you use, the piece of code common to all is the Linux kernel. Although the kernel can be modified to include support for the features you want, every Linux kernel can offer the following features:

- **Multiuser** — Not only can you have many user accounts available on a Linux system, you can also have multiple users logged in and working on the system at the same time. Users can have their own environments arranged the way they want: their own home directory for storing files and their own desktop interface (with icons, menus, and applications arranged to suit them). User accounts can be password-protected, so that users can control who has access to their applications and data.
- **Multitasking** — In Linux, it is possible to have many programs running at the same time, which means that not only can you have many programs going at once, but that the Linux operating system can itself have programs running in the background. Many of these system processes make it possible for Linux to work as a server, with these background processes listening to the network for requests to log in to your system, view a web page, print a document, or copy a file. These background processes are referred to as *daemons*.
- **Hardware Support** — You can configure support for almost every type of hardware that can be connected to a computer. There is support for floppy disk drives, CD-ROMs, removable disks (such as DVDs and USB flash drives), sound cards, tape devices, video cards, and most anything else you can think of. As device interfaces, such as USB and FireWire, have been added to computers, support for those devices has been added to Linux as well.

  For Linux to support a hardware device, Linux needs a *driver*, a piece of software that interfaces between the Linux kernel and the device. Drivers are available in the Linux kernel to support hundreds of computer hardware components that can be added or removed as needed.

NOTE **Most hardware manufacturers don't provide Linux drivers with their peripheral devices and adapter cards. Although most popular hardware will be supported eventually in Linux, it can sometimes take a while for a member of the Linux community to write a driver. Also, some outdated hardware may not be updated to work with the latest Linux kernels.**

- **Networking Connectivity** — To connect your Linux system to a network, Linux offers support for a variety of local area network (LAN) cards, modems, and serial devices. In addition to LAN protocols, such as Ethernet (both wired and wireless), all the most popular upper-level networking protocols can be built in. The most popular of these protocols is TCP/IP (used to connect to the Internet). Other protocols, such as IPX (for Novell networks) and X.25 (a packet-switching network type that is popular in Europe), are also available.
- **Network Servers** — Providing networking services to the client computers on the LAN or to the entire Internet is what Linux does best. A variety of software packages are

available that enable you to use Linux as a print server, file server, FTP server, mail server, Web Server, news server, or workgroup (DHCP or NIS) server.

To make a Linux distribution useful, components need to be added on top of the Linux kernel. For humans to access a Linux system, they can enter commands to a shell or use graphical interfaces to open menus, windows, and icons. Then you need actual applications to run. In particular, a useful Linux desktop system includes the following:

- **Graphical User Interface (X Window System) —** The powerful framework for working with graphical applications in Linux is referred to as the *X Window System* (or simply X). X handles the functions of opening X-based graphical user interface (GUI) applications and displaying them on an X server process (the process that manages your screen, mouse, and keyboard).

  On top of X, you use an X-based desktop environment to provide a desktop metaphor and window manager to provide the look-and-feel of your GUI (icons, window frames, menus, and colors, or a combination of those items called *themes*). There are a few desktop environments and even more desktop managers to choose from. (CentOS focuses on the GNOME and KDE desktop environments, but also has several other desktop environments and window managers available.)

- **Application Support —** Because of compatibility with POSIX and several different Application Programming Interfaces (APIs), a wide range of free and Open Source software is available for Linux systems. Compatibility with the GNU C libraries is a major reason for the wide-ranging application support. Often, making an Open Source application available to a particular version of Linux can be done by simply recompiling the source code to run on that Linux version.

# Primary Advantages of Linux

When compared to different commercially available operating systems, Linux's best assets are its price, its reliability, and the freedom it gives you. Scalability is one of its greatest assets. The reliability of Linux includes its built-in security features and architecture, both of which make Linux much more secure than Windows.

Most people know that its initial price is free (or at least under $50 when it comes in a box or with a book). However, when people talk about Linux's affordability, they are usually thinking of its total cost, which includes no (or low) licensing fees, the ability to reuse any of the code as you choose, and the capability of using inexpensive hardware and compatible free, add-on applications. Although commercial operating systems tend to encourage upgrading to more powerful hardware, Linux doesn't require that (although faster hardware and larger disks are nice to have).

In terms of reliability, the general consensus is that Linux is comparable to many commercial UNIX systems but more reliable than most desktop-oriented operating systems. This is especially

true if you rely on your computer system to stay up because it is a Web Server or a file server. (You don't have to reboot every time you change something, unless you've replaced the kernel itself.)

This reliability also extends into the realm of safety. While there have been exploits aimed at Linux software, Linux users are for the most part safe from the same type of malware and viruses that plague Windows users.

With so many people peering at the Linux source code (a benefit of its freedom), mistakes are often fixed in record time. Of course, like a house that has doors and windows, if you leave them open, a burglar can come in. But by properly configuring, monitoring, and updating software on Linux systems that are created with security in mind, such as CentOS, your Linux system can be as secure as any operating system in the world.

Because you can get the source code, you are free to change any part of the Linux system, along with any Open Source software that comes with it, in any way that you choose. Unlike many self-contained commercial products, Open Source software tends to be built in pieces that are meant to interact with other pieces, so you are free to mix and match components to suit your tastes. As I mentioned earlier, Linux is a culture that encourages interoperability. For example, if you don't like a window manager, you can plug in a different one because so many were built to operate within the same framework.

Another advantage of using Linux is that help is always available on the Internet. There is probably someone out there in a Linux newsgroup or mailing list willing to help you get around your problem. Because the source code is available, if you need something fixed, you can even patch the code yourself! On the other hand, I've seen proprietary software vendors sit on reported problems for months without fixing them. Remember that the culture of Linux is one that thrives on people helping other people.

# Going Forward with CentOS

If you find that you need additional support for your CentOS systems or want to increase your skills or become more involved with Linux, there are many directions in which you can go.

## Help from the CentOS Project

If you enjoy CentOS and want to become more connected to the project, there are many ways to do that. To contribute to the CentOS project, start by introducing yourself on a CentOS mailing list (`http://centos.org/modules/tinycontent/ index.php?id=16`). For information on the status of the CentOS project, refer to the News page of the CentOS web site, at `http://centos.org/modules/news/`.

If you want to expand beyond the bounds of what CentOS offers, there are third-party repositories for CentOS containing software packages that are not part of the CentOS distribution. (See the descriptions of software repositories in Chapter 5.)

As the end-user forum of choice for CentOS users, visit the CentOS forums at `http://centos.org/modules/newbb/`. That site already has more than 18,350 topics and more than 67,000 posts you can search for answers to your questions.

## Training and Certification

If you are looking for a career in Linux, Red Hat offers some well-respected programs for becoming a certified expert in Red Hat Linux software. Those skills should translate easily to your work with CentOS. Learn about available programs from Red Hat's training web site (`www.redhat.com/training`).

If you plan to pursue any of the Red Hat certification programs, don't be surprised if after some theoretical training you are given a misconfigured computer and asked to fix it. Those who get Red Hat certifications are expected to be able to clean up and repair Linux systems in the real world, as well as install and secure Red Hat Enterprise Linux systems.

Here is a list of some available certifications from Red Hat:

- **Red Hat Certified Technician (RHCT)** — An RHCT is the most basic Red Hat certification. It focuses on core skills needed by a Red Hat system administrator. Besides being able to install and configure an RHEL system to come up on a corporate network, you are also expected to understand basic troubleshooting techniques.
- **Red Hat Certified Engineer (RHCE)** — The RHCE program builds on the skills developed in the RHCT program. For an RHCE, however, additional capabilities in security and deploying network services are expected.
- **Red Hat Certified Security Specialist (RHCSS)** — As the name implies, an RHCSS becomes proficient in security-related aspects of managing Red Hat Enterprise Linux systems. Courses with this certification include Enterprise Network Services Security, Enterprise Directory Services and Authentication, and SELinux Policy and Administration.
- **Red Hat Certified Architect (RHCA)** — An RHCA's skills are expected to go beyond those of an RHCT or RHCE. The RHCA program focuses on deploying and managing multiple Linux systems across an enterprise, with special attention given to systems management, storage management, performance tuning, and directory services.

For courses on RHCT, RHCE, RHCSS, and RHCA certifications, visit the Red Hat Certified Engineer Program page (`www.redhat.com/training/`). Before taking the RHCT or RHCT certification exams, check out the *RHCE and RHCT Exam Preparation Guide* (`www.redhat.com/certification/rhce/prep_guide/`). Red Hat offers many courses online for Red Hat Linux training, as well as courses in networking, programming (Java, Object, web, and general programming), IT management, and e-business.

## Documentation

In addition to the documentation provided by the CentOS team, there are a lot of avenues to find help using RHEL-based products offered by Red Hat. A good place to start is the Red Hat

Enterprise Linux documentation (`www.redhat.com/ docs/manuals/enterprise`), which includes manuals for installation, system administration, security, SELinux, reference materials, and release notes for different products. You can use keyword searches to find answers to your questions from the Red Hat Knowledgebase.

## Summary

CentOS is an Open Source software project that aims at providing an enterprise-quality Linux operating system that can be distributed without cost to its supporters. The project builds its Linux system from source code that has been tested and deployed in mission critical settings.

This book specifically describes CentOS 5, a complete version of which is included on the DVD that comes with this book. Because CentOS aims at compatibility with Red Hat Enterprise Linux systems, CentOS can be used to learn the skills needed to grow your enterprise-class computing skills.