



www.vskills.com

Certified Linux and Windows
Integration Professional
VS-1146

Vskills Certifications

Vskills Reading Material



Skills for a secure future

Certified Linux and Windows Integration Professional Administrator

VS-1146

V skills

www.vskills.in

Proficiency Testing Programme

of

Intelligent Communication Systems India

Limited

JV of DSIIDC (Govt of NCT Delhi)

& TCIL (Govt of India)

This document describes strategies, processes and techniques for Linux and Windows integration which includes Linux deployment strategies, Samba, file and printer sharing, CUPS, Using NetBIOS Network, LDAP, NT Domains, Kerberos configuration, remote GUI programs, Linux thin client configurations, mail servers and network backups. This document is for beginners and intermediaries.

**Certified - Linux and Windows
Integration Professional
Training Material**

TABLE OF CONTENTS

- 1. Linux Features
 - 1.1. Where Linux Fits in a Network
 - 1.2. Linux as a Server
 - 1.3. Linux on the Desktop
 - 1.4. Comparing Linux and Windows Features

www.vskills.in

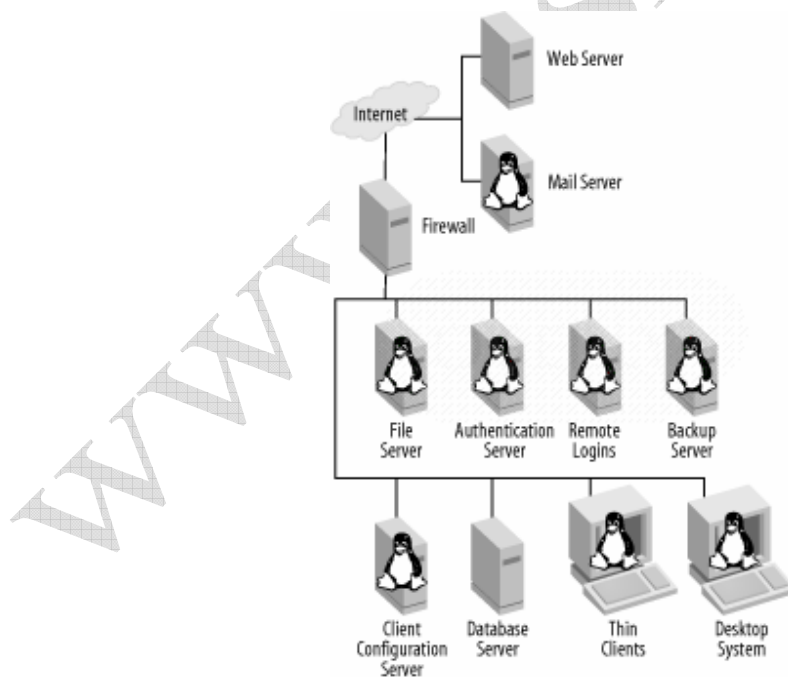
1. LINUX FEATURES

Linux can be an effective addition to a Windows network for several reasons, most of which boil down to cost. Windows has achieved dominance, in part, by being less expensive than competitors from the 1990s, but today Linux can be less expensive to own and operate. This is particularly true if you're running Windows NT 4.0, which has reached end-of-life and is no longer supported. (Windows 2000 will soon fall into this category, as well.) For these old versions of Windows, you're faced with the prospect of paying to upgrade to a newer version of Windows or switch to another operating system. Linux can be that other OS, but you should know something about Linux's features and capabilities before you deploy it.

Effectively deploying Linux requires understanding the OS's capabilities and where it makes the most sense to use.

1.1. Where Linux Fits in a Network

Most operating systems—and Linux is no exception to this rule—can be used in a variety of ways. You can run Linux (or Windows, or Mac OS, or most other common general-purposes OSs) on personal productivity desktop systems, on mail server computers, on routers, and so on. This book doesn't cover every possible use of Linux; instead, it focuses on how Linux interacts with Windows systems on a local area network (LAN) or how Linux can take over traditional Windows duties. The below figure, depicts a typical office network. Linux's mascot is a penguin (known as Tux), so the figure uses penguin images to mark the areas of Linux deployment.



Of course, Linux can be used in roles not shown in the figure. In fact, Linux can be an excellent choice for an OS for such roles as a web server; however, because such uses aren't LAN-centric or don't tie closely to Windows, this book doesn't cover them. You might want to begin with just one

or two functions for Linux on your network, such as a file server or a Dynamic Host Configuration Protocol (DHCP) server. Some systems, such as backend database servers, may be so vital and data-intensive that replacing them with Linux systems, although possible, is a major undertaking that can't be adequately covered here.

1.2. Linux as a Server

Traditionally, Linux's strength has been as a server OS. Many businesses rely upon Linux to handle email, share files and printers, assign IP addresses, and so on. Linux provides a plethora of open source programs to handle each of these server tasks, and many more. Before you attempt to deploy a Linux server, though, you should understand Linux's strengths and weaknesses in this role, what type of hardware you're likely to need, and what types of software you'll need.

Linux Server Capabilities

As seen in figure above, Linux can be deployed in many different ways. Indeed, the figure presents an incomplete picture because it focuses on only those roles described in this book. Linux firewalls, web servers, databases, and more are all available. Still, Linux has certain strengths and weaknesses as a server that you should understand as you plan where to use it. Linux's greatest strengths as a server include the following:

- ✓ Reliability - Linux has earned a reputation as a very reliable OS, which, of course, is a critically important characteristic for servers.
- ✓ Cost - You can download Linux from the Internet at no cost (aside from connect charges), which can be important in keeping costs down. Of course, the up-front purchase price (or lack of it) is only part of the equation; support costs, hardware costs, and other factors can be much more important. Linux's total cost of ownership (TCO) is a matter of some debate, but most studies give Linux high marks in this area.
- ✓ License issues - The Linux kernel is licensed under the GNU General Public License (GPL), and much of the rest of Linux uses the same license. Most other Linux programs use other open source licenses. The result is that you're not bound by restrictive commercial license terms; as a user or administrator, you can do anything with Linux that you can do with a commercial OS, and then some. If you want to redistribute changes to a program, though, some open source licenses impose restrictions, so you should check the license to see how you're permitted to distribute the changes. (Of course, most commercial OSs don't even let you see the source code!)
- ✓ Security issues - Linux isn't vulnerable to the worms and viruses that plague the Internet today; almost all of these pests target Windows systems. Of course, a Linux server can still be inconvenienced by worms and viruses because it may need to process them in some way; a Linux mail server may still need to accept email with worms and perhaps then identify and delete the worm. Linux won't be infected by the worm, though—at least, not by any worm that's known to be spreading as I write.
- ✓ Server software selection - As a Unix-like OS, Linux has inherited many popular Unix servers, such as sendmail and Samba. In fact, some of these, including Samba, were written using Linux as a primary target OS.
- ✓ Remote administration - Linux provides several remote administration methods, ranging from remote logins using text-mode tools such as Secure Shell (SSH) or Telnet to tools designed for remote administration via web browsers, such as Webmin (<http://www.webmin.com>). Of

course, remote administration isn't unique to Linux, but Linux presents more options than do most non-Unix OSs.

- ✓ **Resource use** - With Linux, you have fine control over what programs you run, which enables you to trim a system of unnecessary items to help get the most out of your hardware. For instance, most servers don't need to run a local GUI, so Linux enables you to run a system without one, and even to omit the X files and programs from the hard disk.
- ✓ **Customization** - In addition to customizing the system to minimize resource use, you can modify Linux to achieve other ends. For instance, you can recompile the kernel to add or omit features that help the system operate as a router, or you can alter the startup sequence to accommodate special needs. Taken to the extreme, these features help those who run Linux on specialized embedded devices, but such uses are well beyond the scope of this book.
- ✓ **Hardware flexibility** - Linux is available on a variety of hardware, ranging from specialized embedded versions of Linux to supercomputers. This book is designed to help those running Linux on fairly traditional small- to mid-sized servers and desktop systems using conventional Intel Architecture 32 (IA-32; a.k.a. x86) hardware or other hardware of comparable power. Even in this realm, Linux is very flexible; you can run it on AMD64, PowerPC (PPC), Alpha, and other CPUs, which lets you standardize your OS even if you happen to have different hardware platforms. Most of these advantages are advantages of Unix-like OSs generally, and so apply to other OSs, such as Solaris and FreeBSD. Compared to such OSs, Linux's greatest strengths are its hardware flexibility, open source licensing, and low cost (although several other low-cost and open-source Unix-like OSs exist).

Of course, no good thing is without its problems, and Linux is no exception to this rule. Fortunately, Linux's problems are minor, particularly when the OS is used on a server:

- ✓ **Administrative expertise requirements** - Linux requires more in the way of administrative expertise than do some alternatives. For most organizations, this factor ultimately boils down to one of the variables in TCO calculations: Linux administrators are likely to demand higher salaries than Windows administrators do. On the other hand, Linux's reliability, scalability, and other factors frequently more than compensate for this problem in the TCO equation.
- ✓ **Security issues** - Although immunity to infection by common Windows worms and viruses is a Linux advantage, Linux has its own security drawbacks. Crackers frequently attempt to break into Linux servers, and they sometimes succeed. In theory, this should be difficult to do with a well-administered system, but neglecting a single package upgrade or making some other minor mistake can leave you vulnerable. Of course, Linux isn't alone in this drawback, but it's one to which you should be alert at all times.

The term hacker is used by the popular media to refer to computer miscreants—those who break into computers and otherwise wreak havoc. This term has an older and honorable meaning as referring to skilled and enthusiastic computer experts, and particularly programmers. Many of the people who wrote the Linux kernel and the software that runs on it consider themselves hackers in this positive sense. For this reason, I use an alternative term, cracker, to refer to computer criminals.

Overall, Linux's strengths as a server far outweigh its weaknesses. The OS's robustness and the number of server programs it runs are powerful arguments in its favor. Indeed, those are the reasons commercial Unix variants have traditionally run many important network services. Linux

has been slowly eroding the commercial Unix market share, and its advantages can help you fill the gaps in a Windows-dominated network or even replace existing Windows servers.

Typical Linux Server Hardware

As noted earlier, one of Linux's strengths is that it runs on a very wide range of hardware. Of course, this isn't to say that you can use any hardware for any particular role; Linux won't turn a 10-year-old 80486 system with a 1-GB hard disk into a powerhouse capable of delivering files to thousands of users.

Linux most commonly runs on IA-32 hardware, and much Linux documentation, including this book, frequently presents IA-32 examples. IA-32 hardware is inexpensive, and it's the original and best-supported hardware platform for Linux. Still, other options are available, and some of these are well worth considering for a Linux server.

One of the problems with IA-32 is that it's a 32-bit platform. Among other things, this means that IA-32 CPUs are limited to addressing 2³² bytes, or 4 GB, of RAM. (Intel Xeon processors provide a workaround that involves page swapping, or hiding parts of memory to keep the total available to the CPU at just 4 GB.) Although a 4-GB memory limit isn't a serious problem for many purposes, some high-powered servers—particularly those that support many user logins—need more RAM. For them, using a 64-bit CPU is desirable. Such CPUs can address 2⁶⁴, or 1.8x10¹⁹, bytes of RAM, at least in theory. (In practice, many impose lower limits at the moment, but those limits are still usually in the terabyte range.)

Several 64-bit CPUs are available, including the DEC (now Compaq) Alpha, several AMD and Intel CPUs that use the AMD64 architecture, Intel's IA-64 Itanium, the IBM Power64 (the first of which is the PowerPC G5), and the SPARC-64. Of these, the Power64 and AMD64 platforms are likely to become more common in the next few years. With AMD and Intel both producing AMD64 CPUs, they are likely to take over the market dominated by IA-32 CPUs through most of the 1990s and early 2000s. Apple is rapidly shifting its Macintosh line to the Power64, and IBM and a few others are producing Power64-based servers. Of course, if you already have another type of 64-bit system, or you have an opportunity to get one at a good price, you can run Linux on it quite well. Linux support for the AMD64 and Power64 platforms is likely to be more mature than for other 64-bit platforms, though.

Of course, not all servers need 64-bit CPUs. For them, IA-32 CPUs, such as Intel's Pentium 4 or AMD's Athlon, are perfectly adequate. In fact, many systems can make do with much weaker CPUs. A DHCP server can run quite well on an old 80386, for instance. Just how much CPU power you need depends on the function of the server. Functions such as handling thin-client or other remote logins, converting PostScript into non-PostScript printer formats (particularly for multiple heavily used printers), and handling hundreds or thousands of clients, are likely to require lots of CPU power. Lighter duties, such as running a DHCP server, a local Domain Name System (DNS) server, or even a remote login server for a network with a dozen or so computers, requires much less in the way of CPU power. For such purposes, you can probably run Linux on a spare or retired computer. Even a system that's too weak to run a modern version of Windows can make a good small server.

Disk space requirements also vary with the server's intended role. Most obviously, a file server is likely to require lots of disk space. Precisely what "lots" is, though, depends on how many users you have and what types of files they store. Disk-intensive servers frequently use Small Computer Systems Interface (SCSI) hard disks rather than Advanced Technology Attachment (ATA) disks, because SCSI disks scale better in multidisk setups and because disk manufacturers often offer higher-performance disks in SCSI form only. SCSI disks cost more than do ATA disks, though. You'll have to judge for yourself whether your budget permits the use of SCSI disks. Recently, Serial ATA (SATA) disks have started to emerge as an alternative to traditional parallel ATA disks and SCSI. Depending on the drivers, SATA disks may appear to be SCSI disks in Linux, but they aren't.

Although servers can vary greatly in their major hardware components and needs, network connectivity is a common factor. All servers require good network links. On a LAN, this most commonly means 100- or 1000-Mbps (1-Gbps, or gigabit) Ethernet. Linux ships with excellent Ethernet support; chances are any Ethernet adapter will work. Modern motherboards frequently come with built-in Ethernet, too. Of course, not all Ethernet adapters are created equal: some are more stable, produce better throughput, or consume less CPU time. As a general rule, Ethernet adapters from major manufacturers, such as Intel, 3Com, and Linksys, are likely to perform best. No-name bargain-basement Ethernet cards will almost certainly work, but they may give more problems or perform less well under heavy network loads.

Most servers have similar video display capabilities. In this case, though, servers' needs are unusually light; because a server's primary duty is to deliver data over a network, a high-end graphics card is not a requirement. You might want something that's at least minimally supported by Linux (or by a Linux X server, to be precise) so you can administer the computer at the console using GUI administration tools; however, this isn't a requirement.

Typical Linux Server Software

When deciding how to deploy Linux on a LAN, you must consider what hardware and software to use. Linux isn't a monolithic beast you can decide to install and be done with; you must make choices about your Linux installation. These choices begin with your decision about a Linux distribution—a collection of software and configuration files that's bundled together with an installation program. Some distributions are better suited than others to use on a server, although with enough extra effort, you can use just about any Linux distribution on a server computer. Beyond the distribution, you must pick individual server programs. These choices are very specific for the purpose of the computer. For instance, if you run a mail server computer, you need to decide which mail server program to run, and this decision can have important consequences for everything else you do on the computer. Such a decision is likely to be relatively unimportant on other types of server computers.

The term server can have multiple meanings; it can refer to either an individual program that delivers network services or to the computer on which that program runs. (A similar dual meaning applies to the word client on the other end of the connection.) In most cases, the meaning is obvious from the context, but when necessary, I clarify by explicitly specifying a server computer or program. Some people use the term service to refer to server programs or to the features that they provide.

Picking a distribution for server use - Your choice of distribution depends partly on your choice of hardware platform. Some distributions, such as Debian GNU/Linux, are available on a wide range of CPU architectures, whereas others, such as Slackware Linux, are available for just one CPU. If you're already familiar with a distribution, and you want to use it for your server, you may want to plan your hardware purchases around this fact. If you already have the hardware, though, or if you're constrained to use a particular platform for policy or budget reasons, you may need to narrow the range of your hardware choices. Broadly speaking, the IA-32 platform has the most choices for distributions, although a few distributions run only on other platforms. The most popular Linux distributions used on servers include the following:

- ✓ CentOS - This distribution, headquartered at <http://freshmeat.net/projects/centos/>, is a community-based fork of Red Hat's Enterprise Linux. As such, it's technically very similar to Red Hat, but support details are quite different.
- ✓ Debian - This distribution is one of the few completely noncommercial distributions; it's maintained entirely by volunteers. It uses the Debian package format and is well-respected for its stable main ("stable") branch. This branch is on a very long release cycle, though, so it sometimes lags when major new versions of component packages are released. (Bug-fix and security updates are prompt, however.) Debian's "unstable" branch is much more up to date, but it's not as well-tested as the reliable main "stable" branch. Keeping up to date is fairly simple because of Debian's Advanced Package Tools (APT) package, which enables software updates over the Internet by typing a couple of commands. Because Debian doesn't sell official packages with support, obtaining outside support requires you to hire an independent consultant. To configure Debian, you normally edit text-mode configuration files in a text editor rather than use a GUI configuration tool. Overall, Debian is a good choice for servers, which usually must be stable above all else. Debian is available for an unusually wide range of CPUs, including IA-32, SPARC, PowerPC, Alpha, IA-64, and several other platforms. To learn more, check Debian's web site, <http://www.debian.org>.
- ✓ Fedora Core - This distribution is the freely redistributable version of Red Hat. Its development cycle is faster than that of the official Red Hat releases, and part of Fedora's purpose is to serve as a test bed for new packages that will eventually work their way into Red Hat. Fedora can be a good choice if you like Red Hat, don't have a lot of money to spend on a commercial distribution, and don't mind doing without the official Red Hat support. Fedora Core is available for IA-32 and AMD64 CPUs; you can find it at <http://fedora.redhat.com>.
- ✓ Gentoo - Like Debian, Gentoo is maintained by volunteers. This distribution emphasizes building packages from source code; its package manager, known as portage, enables you to type a one-line command that downloads the source code and patches, compiles the software, and installs it. (This system is similar to the ports system of FreeBSD.) Portage can be a good way to tweak compiler settings for your CPU, installed libraries, and so on, but the time spent compiling packages can be a drawback. Also, if you maintain many systems with differing hardware, you may have trouble cloning a system, because the optimizations used on your original system may not work on other systems. One advantage of Gentoo is that it's easy to keep up to date with the latest packages, but you can make your system unstable if the latest version of an important package breaks other programs. Like Debian, Gentoo eschews GUI configuration tools in favor of raw text-mode configuration file editing. Gentoo is available for IA-32, AMD64, SPARC, and PowerPC CPUs. You can learn more at <http://www.gentoo.org>.
- ✓ Red Hat - Red Hat is probably the most popular distribution in North America, particularly if you include its Fedora variant. This distribution originated the RPM Package Manager (RPM)

package format. Even many programs that don't ship with Red Hat are available in IA-32 and source RPM packages, which makes software installation easy. Many third-party programs are released with Red Hat in mind, making Red Hat a safe bet for running such programs. With the release of Fedora 1, Red Hat has been focused its main product line (Red Hat Enterprise) on the business market, especially servers, although it can certainly be used on desktops. The main selling point of Red Hat Enterprise is its support, including system updates and maintenance via the Red Hat Network. These include GUI update tools that can grab updates over the Internet. Red Hat ships with a number of Red Hat-specific GUI configuration tools, but they're often limited enough that you'll need to bypass them and edit configuration files by hand. IA-32 is the primary Red Hat platform, although some versions are available for AMD64, Itanium, and certain IBM servers. Some older versions ran on SPARC and Alpha CPUs, but these versions are very outdated. The official Red Hat web site is <http://www.redhat.com>.

- ✓ Slackware - Slackware is the oldest of the common Linux distributions. It uses tarballs as its package distribution format and eschews GUI configuration tools. Slackware ships with fewer packages than most Linux distributions; to install more exotic programs, you may need to compile them from source. Slackware may be a good choice if you're used to "old-style" Unix system administration, but if you're relatively new to Linux, you may want to pick something else. Slackware is available only on the IA-32 platform. You can learn more at <http://www.slackware.com>.
- ✓ SuSE - This distribution is RPM-based but isn't derived from Red Hat directly. It's a good choice for both server and desktop use, and it includes a GUI administration tool called YaST. You can update packages over the Internet with this tool, as well as administer the local system. SuSE is primarily an IA-32 and AMD64 distribution, although an older PowerPC release is also available. Check <http://www.suse.com> for more information. SuSE was originally an independent German company, but it was purchased by Novell in early 2004.

Ultimately, any of these distributions (or various other less popular ones) can be configured to work equally well, assuming you're using IA-32 hardware. You may be better able to get along with a particular distribution depending on your system administration style and particular needs, though. If you like to tweak your system to get the very best possible performance, Gentoo's local-build approach may be appealing. If you like GUI configuration tools for handling the simple tasks, Red Hat or SuSE should work well. For an extremely stable system, Debian is hard to beat. If your hardware is old, consider Debian or Slackware, which tend to install less extraneous software than the others. If you want to use the most popular distribution, look at Red Hat. If the package management tool is important, pick a distribution that uses the tool you like.

Although the Linux kernel and most Linux tools are open source, some distributions include small amounts of proprietary software. This inclusion makes redistribution of the OS illegal. For this reason, you should check license terms before doing so or before installing it on many systems. Most distributions are freely redistributable, although most of them are available for sale. Buying a full package helps provide financial support to the distributions, which helps to advance future development. (Debian and Gentoo are exceptions; no commercial packages of these distributions are available, although you can buy them on cut-rate CDs that provide little or no profit to the actual developers. CentOS and Fedora have no for-sale versions, either, unless you count Red Hat in that role.)

Distribution choice is a highly personal matter. What works well for one administrator may be a poor choice for another. You may also run into policy issues at your workplace; for instance, you may need to buy a package from a vendor that offers certain support terms, which might rule out some distributions. If you haven't used Linux extensively in the past, you should study the options, focusing on distributions that provide GUI configuration tools you can use to get the basics up and running quickly. For more experienced administrators, I recommend whatever you've used in the past, provided you found it satisfactory. If you've used other Unix-like OSs but not Linux, try to find a Linux distribution with an administrative style similar to what you've used before. You should also try to minimize the number of Linux distributions you use, ideally to just one; this helps simplify system administration because you'll have just one set of distribution-specific tools and packages to learn, and you can retrieve package updates from just one source.

Picking individual server programs

Once you've picked and installed a Linux distribution, you'll need to decide what server programs to use. For most major server classes, most distributions provide a single default choice, such as the sendmail mail server. It's usually easiest to stick with the default, but if the server in question is the primary function of the computer, and if the default choice isn't what you'd like to use, you can change it.

Sometimes the alternative programs work with the same protocol as the default; for instance, the Postfix, Exim, and qmail servers are all popular alternatives to sendmail. All are implementations of the Simple Mail Transfer Protocol (SMTP), and you can replace one package with another without changing the server computer's interactions with other computers. (You usually need to implement minor or major changes to the server system's local configuration, though, and perhaps replace some support programs.) Other protocols for which multiple server program implementations are available in Linux include pull email using the Post Office Protocol (POP) or Internet Message Access Protocol (IMAP), the File Transfer Protocol (FTP), Hypertext Transfer Protocol (HTTP; a.k.a. web server protocol), the Kerberos authentication protocol, the Remote Frame Buffer (RFB) remote GUI login protocol, and the DNS protocol. This book covers many, but not all, of these protocols. In some cases, alternative servers are configured in similar ways, but sometimes configuration is very different for the various server programs.

Other times, you may need to choose between two incompatible protocols that accomplish similar tasks. For instance, POP and IMAP are two different ways to deliver email received by an SMTP server to client systems (that is, users running mail programs such as Outlook Express or KMail). Other examples include RFB and X11R6 for remote GUI access; Telnet and SSH for remote text-mode access; the Server Message Block/Common Internet File System (SMB/CIFS), Network File System (NFS), and AppleShare for remote file-sharing protocols; SMB/CIFS, AppleShare, Line Printer Daemon (LPD), and Common Unix Printing System (CUPS) for printer sharing; SSH and FTP for remote file transfers; and NetBIOS domain logins, Lightweight Directory Access Protocol (LDAP), Kerberos, and Network Information Services (NIS) for remote authentication. In all these cases, the competing protocols have their own advantages and disadvantages. For instance, SSH provides encryption whereas FTP and Telnet do not. Some protocols are closely associated with particular OSs; for instance, SMB/CIFS and NetBIOS most commonly are used on Windows-dominated networks, whereas NFS is used in Unix-to-Unix file sharing and AppleShare is used most often with Mac OS systems. This book covers many of these protocols, but it focuses

on those that are used most commonly on Windows-dominated networks, or at least that have potential to enhance such networks.

1.3. Linux on the Desktop

Although it's only one system in Figure 1-1 (or two, if you count the thin client), Linux use as a desktop OS is different enough from Linux server use that it requires its own description. Several classes of differences are particularly noteworthy.

- ✓ User interfaces - Generally speaking, desktop systems require better user interface devices (video cards, monitors, keyboard, and mice) than do servers. Linux usually works well with the same hardware as Windows systems, but with one caveat: the very latest video cards sometimes aren't well supported in Linux. Staying a generation or two behind the leading edge is therefore desirable in Linux.
- ✓ Disk and network hardware - Many classes of servers require the very best in disk and network hardware, but this is less often the case for desktop uses. You can often get by with average ATA devices and typical Ethernet (or other network) hardware. Some desktop systems, though, do need excellent disk or network hardware. These are typically high-performance systems that run scientific simulations, specialized engineering software, and so on.
- ✓ RAM - Desktop systems' needs for powerful CPUs and lots of RAM vary with the application. Generally speaking, modern GUI environments are RAM-hungry, so you should equip a modern desktop system with at least 256 MB of RAM, and probably 512 MB or even 1 GB if possible. Linux does support slimmer environments that can work well in 128 MB or less if necessary, though. Most desktop applications don't really need powerful 64-bit CPUs, but some programs are written inefficiently enough that a fast CPU is desirable. Also, certain applications are CPU-intensive.
- ✓ Peripheral hardware - One of Linux's weakest hardware points as a desktop system is its degree of support for peripheral hardware that's common on desktop systems but less common on servers, such as scanners, digital cameras, video input cards, external hard drives, and so on. Drivers for all major classes of hardware exist, but many specific devices are unsupported. If you're buying or building a new system, including such peripherals, you can easily work around this problem by doing a bit of research and buying only compatible devices. If you want to convert existing systems to Linux, though, existing incompatible hardware can drive up the conversion cost.
- ✓ Linux distributions - The distributions outlined earlier, in Section 1.2.3.1, can all function as desktop distributions. Others, such as Mandrake and Xandros, are geared more toward desktop use.
- ✓ Configuration and administration - Configuring and administering a desktop Linux system is much like handling a server system, but certain details do differ, mostly related to the specific software used to support each role. You might not even install an SMTP mail server on a desktop system, for instance; instead, you might install the OpenOffice.org office suite. The kernel, the basic startup procedures, and so on are likely to be similar for both types of system.

The terms desktop and workstation have similar meanings in the computer world; both refer to systems that are used by end users to accomplish real-world use. Typically, workstation refers to slightly more powerful computers, to those used for scientific or engineering functions as opposed to office productivity, to systems running Unix or Unix-like OSs as opposed to Windows, or to

those with better network connections. The exact word use differs from one author to another, though. I use the two words interchangeably, but I use desktop most frequently.

Traditionally, Linux hasn't been a major player in the workstation arena; however, it does have all the basic features needed to be used in this way. Over the past few years, Linux's user interface has been improving rapidly, in large part because of the K Desktop Environment (KDE; <http://www.kde.org>) and the GNU Network Object Model Environment (GNOME; <http://www.gnome.org>). These are two desktop environments for Linux that provide a GUI desktop metaphor familiar to users of Windows, Mac OS, OS/2, and other GUI-oriented OSs. These environments rest atop the X Window System (or X for short) that provides low-level GUI tools such as support for opening windows and displaying text. Finally, tools such as office suites (OpenOffice.org, KOffice, GNOME Office, and so on), GUI mail readers, and web browsers make Linux a productive desktop OS. All these tools, but particularly desktop environments and office suites, have advanced substantially over the past few years, and today Linux is roughly as easy to use as Windows, although Linux is less familiar to the average office worker.

Many people think of Linux as a way to save money over using a commercial OS. Although Linux can indeed help you save money in the long term, you shouldn't blindly believe that Linux will do so, particularly in the short term. Costs in the switch, such as staff time installing Linux on dozens or hundreds of computers, retraining, replacing hardware for which no Linux drivers exist, and converting existing documents to new file formats, can create a net short-term cost to switching to Linux. In the long term, Linux may save money in license fees and easier long-term administration, but sometimes Linux's limitations can put a drag on these advantages. You'll need to evaluate Linux with an eye to how you intend to use it on your network.

1.4. Comparing Linux and Windows Features

When deploying Linux, you must consider the overall feature sets of both Linux and its potential competitors. In an environment that's dominated by Windows, the most relevant comparison is often to Windows, so that comparison will be described in the rest of this chapter.

Linux shares many of its strengths with other Unix-like OSs, and particularly with other open source Unix-like OSs, such as FreeBSD. Linux is probably the most popular and fastest-growing of these OSs because of its dynamic community and large number of distributions. If you prefer to run, say, FreeBSD, you certainly may, and much of this book is applicable to such environments; however, this book does focus on Linux, and it doesn't always point out where FreeBSD or other Unix-like OSs fit into the picture.

Linux is a powerful operating system, but Microsoft's latest offerings (Windows 2003 and Windows XP) are also powerful. Important differences between the two OS families include the following:

- ✓ Cost - Linux itself is low-cost, and this fact can be a big plus; however, the cost of the software is likely to be a small factor in the overall cost of running a computer. The TCO of Linux versus Windows is a matter of some debate, but it's likely to be lower for Linux if experienced Linux or Unix administrators are already available to deal with the system.

- ✓ GUI orientation - All versions of Windows are largely tied to their GUIs; administering a Windows box without its GUI is virtually impossible. This linkage can make picking up Windows administration a bit easier for those unfamiliar with text-mode configuration, but it imposes some overhead on the computer itself, and it restricts the ways in which the system can be administered. These limitations are particularly severe for servers, which may not need a flashy GUI to handle mail or deliver IP addresses, except insofar as the OS itself requires these features. Linux, by contrast, is not nearly so GUI-oriented. Many distributions do provide GUI tools, but bypassing those tools to deal with the underlying text-mode configuration files and tools is usually a simple matter, provided you know where those files and tools are located and how to handle them.
- ✓ Hardware requirements - In part because of Windows' reliance on its GUI, it requires slightly more powerful hardware than does an equivalent Linux server. This factor isn't extremely dramatic, though; chances are you won't be able to replace a 3-GHz Pentium 4 Windows system with a 200-MHz Pentium Linux system and achieve similar performance. Linux also runs on an extremely broad range of hardware platforms—IA-32, AMD64, PowerPC, Sparc, and so on. On the other hand, in the IA-32 world, the vast majority of hardware comes with Windows drivers, whereas Linux driver support isn't quite as complete. Linux drivers are available for most, but not all, IA-32 hardware.
- ✓ Software choices - Both Linux and Windows provide multiple choices for many server software categories, such as mail servers or FTP servers; however, those choices are different. The best choices depend on the server type and your specific needs. Much of this book focuses on servers that work very well for Linux and for which the Windows equivalents have problems of one sort or another—cost, reliability, flexibility, or something else.
- ✓ Windows client integration - This issue is really one of server features. Many Windows server programs are designed around proprietary or semiproprietary Microsoft protocols, or provide extended features that can be accessed from Microsoft clients. For these functions, Linux servers must necessarily either play catch-up or use alternative protocols. For instance, the Samba server on Linux does not provide the full features of a Windows 2000 or 2003 Active Directory (AD) domain controller. Thus, if you want such features, you must run either the Windows server or find some other way to implement the features you want.
- ✓ File compatibility - Because Linux doesn't run the popular Windows programs except under emulators, file format compatibility may be an issue. This can be a factor when you read your own existing files or exchange files with other sites (with clients, say). In the office field, OpenOffice.org provides very good, but not absolutely perfect, Microsoft Office document compatibility.

On the whole, Linux makes an excellent choice for many small, mid-sized, and even large servers that use open protocols. When the server uses proprietary protocols or Microsoft extensions, the situation may change. Linux can also be a good choice as a desktop OS, particularly if your organization isn't tied to proprietary Microsoft file formats.

Certifications

► Accounting, Banking & Finance

- Certified AML- KYC Compliance Officer
- Certified Business Accountant
- Certified Commercial Banker
- Certified Equity Research Analyst
- Certified Foreign Exchange Professional
- Certified Hedge Fund Manager
- Certified Merger and Acquisition Analyst
- Certified Tally 9.0 Professional
- Certified Treasury Markets Professional
- Certified Wealth Manager

► Foreign Trade

- Certified Export Import (Foreign Trade) Professional

► Hospitality

- Certified Restaurant Team Member (Hospitality)

► Human Resources

- Certified HR Compensation Manager
- Certified HR Staffing Manager
- Certified Human Resources Manager
- Certified Performance Appraisal Manager

► Logistics & Supply Chain Management

- Certified International Logistics Professional
- Certified Logistics & SCM Professional
- Certified Purchase Manager

► Law

- Certified IPR & Legal Manager

► Life Skills

- Certified Business Communication Specialist
- Certified Public Relations Officer

► Media

- Certified Advertising Manager
- Certified Advertising Sales Professional

► Office Skills

- Certified Data Entry Operator
- Certified Office Administrator

► Project Management

- Certified Project Management Professional

► Real Estate

- Certified Real Estate Consultant

► Information Technology

- Certified Android Apps Developer
- Certified ASP.NET Programmer
- Certified Basic Network Support Professional
- Certified Business Intelligence Professional
- Certified C# Professional
- Certified CAD Professional
- Certified Cloud Computing Professional
- Certified Computer Fundamentals (MS Office) Professional
- Certified Core Java Developer
- Certified CSS Designer
- Certified Data Mining and Warehousing Professional
- Certified DHTML & Javascript Developer
- Certified Django Developer
- Certified DTP operator
- Certified E-commerce Professional
- Certified E-Governance Professional
- Certified Enterprise Applications Integration Specialist (Biztalk)
- Certified Ethical Hacking and Security Professional
- Certified Facebook Apps Developer
- Certified Grid Computing Professional
- Certified Hadoop and Mapreduce Professional
- Certified HTML Designer
- Certified HTML5 Developer
- Certified iPhone Apps Developer
- Certified IT Support Professional
- Certified J2ME Programmer
- Certified Joomla Developer
- Certified Linux Administrator
- Certified Magento Professional
- Certified MySQL DB Administrator
- Certified Network Security Professional
- Certified Open Source CMS (Drupal) Professional
- Certified PHP Professional
- Certified PL/SQL Developer
- Certified Python Professional
- Certified Router Support Professional
- Certified Selenium Professional
- Certified SEO Professional
- Certified Software Quality Assurance Professional
- Certified Software Security Professional
- Certified Software Testing Professional
- Certified SQL Server 2008 Programmer
- Certified WiMax(4G) Professional
- Certified Wordpress Developer
- Certified XML Developer

► Sales, BPO

- Certified Telesales Executive

Contact us at:

V-Skills

011-473 44 723 or info@vskills.in

Intelligent Communication Systems India Limited

DSIIDC Administrative Building,

Okhla Industrial Estate-III, New Delhi-110020

www.vskills.in