

1

Starting with Ubuntu Linux

Whether you make extensive use of Ubuntu Linux at work every day, or just putter around with it once in a while, a book that presents efficient and comprehensive ways to maintain, monitor, secure, and enhance Ubuntu can be an invaluable resource.

Ubuntu Linux Toolbox is that resource.

Ubuntu Linux Toolbox is aimed primarily at power users and systems administrators. To give you what you need, we will show you how to quickly find and install software for Ubuntu, as well as how to update, maintain, and monitor the health and security of your system. In short, we will show you the most efficient ways of using Ubuntu by working with some of the powerful tools that are at your fingertips.

The goal of this book is to pack as much useful information as possible into a small package that you can carry around with you. To that end, we describe:

- ❑ **Commands** — Tons of command line examples demonstrate clever and useful ways to navigate the often daunting command line.
- ❑ **GUI tools** — Quick tips for using graphical interface tools to administer and configure your Ubuntu system.
- ❑ **Software repositories** — Methods for downloading and installing the software which is custom made for your Ubuntu system.
- ❑ **Online resources** — Where to find useful and helpful information about Ubuntu, such as mailing lists that you can subscribe to, IRC channels, and other online resources.
- ❑ **Local documentation** — Tools for working with the man pages, the standard Linux and Unix reference volumes, as well as specific documentation for the software you install.

IN THIS CHAPTER

Introducing Ubuntu Linux

Finding Ubuntu resources

Learning quick and powerful commands

Referencing useful utilities

Working as Linux gurus do

Chapter 1: Starting with Ubuntu Linux

Because this book is for people already familiar with Linux, there won't be a lot of screenshots of icons and menus. What you get instead is the quickest path to using your Ubuntu system to its fullest extent.

What you learn in this book will help you become more adept at working with your Ubuntu or Debian system, as well as Linux in general. If this sounds useful to you, please read on.

Ubuntu, Debian, and Linux

Ubuntu is an operating system based on Debian GNU/Linux (www.debian.org/). Debian has been around since the early 1990s, and because of its maturity, is regarded as a leading Linux distribution in terms of stability and security. Debian is also known for its strict adherence to free software (www.debian.org/intro/free). It is on this foundation that Ubuntu has been formed.

If you think of Linux as a cake in a bakery, the Linux kernel and operating system files would compose the soft spongy cake material itself. Not a whole lot to look at, but extremely functional as far as cake goes. Linux distributions, in contrast, take that spongy cake and add top and bottom borders, colors, tiers, side trim, frostings, flavors, decorations, designs, candles, nuts, and sprinkles. As diverse as cakes are in a bakery window, so are Linux distributions. Ubuntu, designed by the Canonical Group (www.canonical.com/), is the icing and color of a Linux distribution built on top of a Debian cake center.

Debian has given rise to not only Ubuntu, but many other Linux distributions (www.debian.org/misc/children-distros):

- Xubuntu — An Xfce-based desktop system based on Ubuntu.
- Kubuntu — A KDE-based desktop system based on Ubuntu.
- Edubuntu — An Ubuntu derivative focused on schools.
- Knoppix — A KDE desktop-oriented live CD based on Debian.
- Kanotix — A Debian-based live CD.
- Damn Small Linux — A tiny (50MB) live CD based on Knoppix.
- Mepis — A desktop live CD based on Ubuntu and Debian.

Xubuntu, Kubuntu, and Edubuntu are the same Debian-based Ubuntu distribution under the hood. The only difference in these is the default desktop they run, or the collection of applications bundled with them. For example, Kubuntu features the KDE Desktop and Adept package manager, which are not installed on Ubuntu. Edubuntu is geared towards educational applications, many of which are not installed by default on the other *ubuntu distros.

Chapter 1: Starting with Ubuntu Linux

Because Debian and Ubuntu are open source systems with many parts built on the GNU General Public License (www.gnu.org/copyleft/gpl.html), anyone is free to take the GPL-based source code, or any part of the GPL'd system, and modify, strip down, build upon, extend, embed, reverse-engineer, and freely distribute those changes or modifications. Generally, the only requirement is that you abide by the terms of the GPL, which basically states that any changes you make to existing GPL works must be made available for others to utilize in the same way (see www.debian.org/social_contract for other licenses Debian recognizes).

In the end, you have not only a superior system with a free, online, worldwide support base, but a product which is constantly evolving and driven by people with a passion for what they do. Many other Linux distributions offer these same advantages; however, Ubuntu has certainly pulled out in front in terms of popularity.

NOTE *Ever wonder where Ubuntu (Edgy Eft) and Debian (Woody) get those odd naming conventions from? Find out at <https://wiki.ubuntu.com/DevelopmentCodeNames> or www.debian.org/doc/manuals/project-history/ch-releases.en.html.*

Ubuntu Compared to Other Linux Distributions

If you log into the command line of both an Ubuntu system and a Red Hat Enterprise Linux or Fedora system, very little will look different. There are common directories and utilities between the two, and functionality is fundamentally the same. So what makes Ubuntu different from other Linux distributions? One difference is the installer.

The complexity of booting and installing Ubuntu has been narrowed down to a handful of mouse clicks, making many of the install decisions automatic based on assumptions as to what the average user may need and want. In contrast, a Red Hat system presents the user with many install options, such as setting up a workstation or server, individually selecting packages to install, and setting administrative options.

Another major difference among Linux distributions is in software management tools. The aim of the utilities and packaging systems is the same for Debian as for other Linux distributions, however the operation and implementations are significantly different. Ubuntu and most other Debian-based systems use the APT (Advanced Package Tool) family of utilities for managing software. You use APT to install, remove, query, and update Debian (deb) packages. Red Hat uses an RPM packaging system to handle the same tasks with its rpm packages.

Another big difference is the way the systems look in regards to initialization, login screen, default desktop, wallpaper, icon set, and more. From this look-and-feel perspective, there are a lot of differences. Although Red Hat and Ubuntu both use the GNOME desktop as the default Window Manager, the GUI tools used for administering the system and their locations on the drop-down menus are entirely different.

Chapter 1: Starting with Ubuntu Linux

The login screen and autumn-colored theme of a default Ubuntu system set it apart from other distributions as well. When you drop down the menus of an Ubuntu desktop, you are not presented with a huge list of applications and utilities. What you get is a rather simple and elegant mixture of some of the best and most functional applications available for the Linux desktop. This approach is characteristic of Ubuntu and is done with the intent of keeping the user from feeling overwhelmed.

Another unique characteristic of a Ubuntu system is the intentional practice of locking the root user account, and instead implementing the use of `sudo` (www.gratisoft.us/sudo/intro.html), which allows you to run one command with root permissions, for system administration tasks (see Chapter 3 for details on the `sudo` command). The root login on a Linux system has privileges that allow unrestrained access to nearly every component of the system. It would be trivial to remove an entire file system as the root user, so Ubuntu tries to limit use of this account to only times when it is prudent. Most Linux distributions require the user to log in or `su` to root to perform administration tasks, however a user on a Ubuntu does this through `sudo` using their own login password, and not a separate one for the root user.

Ubuntu has unique features that have their advantages and disadvantages, but they are far from limiting. Ubuntu has the tools in place to allow you to customize, modify, experiment, and hack to your heart's content if that is what you want to do. Otherwise, the idea is to be an easily maintainable, secure system with a clear and concise application set which is neither limiting nor overwhelming. This makes Ubuntu a very fluid system so you can jump right in and become familiar with it very quickly.

Finding Ubuntu Resources

The Ubuntu community has a vast pool of knowledge you can draw from in the form of online resources. The following is a list of links to some of the most popular and useful venues.

- ❑ <http://ubuntuforums.org> — In this searchable web forum and moderated social network is a diverse, talented, and moderated community of Ubuntu users and support staff. Here people share their success and setbacks with each other as well as offering assistance and guidance. Chances are good that if you're having difficulty with something in Ubuntu, someone has already run into the same problem and found a solution.
- ❑ www.ubuntu.com/support — This site offers paid support from Canonical Ltd., the company behind Ubuntu. If you don't want to spend time searching through the forums, or waiting for responses, Canonical Ltd. is one avenue for telephone, e-mail, and web support costing around \$20 a month. There is also Ubuntu training available aimed at companies and corporate users.
- ❑ <https://help.ubuntu.com> — This site contains the official, up-to-date, online documentation for each Ubuntu release. As newer Ubuntu releases come out, you can come here to find out what's new.

Chapter 1: Starting with Ubuntu Linux

- ❑ <http://screencasts.ubuntu.com> — View recorded desktop sessions on how to do different things with Ubuntu, from setting up a printer, to setting up Samba file sharing, to installing updates to keep your Ubuntu system in top shape. Ubuntu users are encouraged to join the Ubuntu Screencasts Launchpad Team (<https://launchpad.net/~ubuntu-screencasts>) to contribute.
- ❑ <https://lists.ubuntu.com/mailman/listinfo/ubuntu-users> — Join the Ubuntu-users mailing list and interact with Ubuntu users over e-mail to discuss and solve problems that come up with everything from implementing mysql databases to setting up a problematic network devices. An archive of past threads can be viewed at <https://lists.ubuntu.com/archives/ubuntu-users>.
- ❑ <https://wiki.ubuntu.com/IRCResourcePage> — If you are interested in live IRC chat support, you can visit the Ubuntu IRC resource page to find guidelines, clients, and chat servers which are an available source of support, free at any time. It is advisable to visit the Ubuntu Code of Conduct page (www.ubuntulinux.org/community/conduct/) if you have not taken part in IRC chat before.

If you plan on buying hardware to use with your Ubuntu or other Linux system, these sites may be helpful in determining where to spend your money:

- ❑ www.linux-usb.org — This web site aims to maintain a working knowledge of USB devices known to be Linux-friendly. There is a search utility where you can plug in the name or model of a manufacturer and get an instant status report on the usability of that device with Linux.
- ❑ www.linux-foundation.org/en/OpenPrinting — The CUPS (<http://cups.org>) printing system is the standard printing system used on most Linux systems these days. If your printer model is not listed when you attempt to add a new printer to your Ubuntu system, you may need to search this site for an updated PPD file to add to your CUPS system. Vendors who make Linux-friendly printers can also be found at www.linux-foundation.org/en/OpenPrinting/Database/SuggestedPrinters.
- ❑ www.sane-project.org — Scanner Access Now Easy (SANE) is a site devoted to the topic of document scanning on Linux. If you are looking for a scanner or multifunction printer, check here to see how well the vendors stack up in terms of Linux support.
- ❑ <http://tldp.org> — The Linux Documentation Project is a culmination of Guides, How-To articles, and FAQs covering everything from how to make coffee with Linux to setting up QoS and Traffic Control.

Certainly this is not a complete list, but these are good places to look first. You can also try searching for Linux-related support on a hardware vendor's web site prior to making your purchase. If they intend their hardware to work with Linux, they may have drivers or instructions available. And don't forget the wealth of information you can find by searching for Linux on your favorite search engine.

Chapter 1: Starting with Ubuntu Linux

Lastly, look for a local Linux User's Group (LUG) in your area. A LUG is a local community of people keenly interested in Linux and its implementations. You will find people with a wide range of experience, from system administrators to casual Linux users, to distro maintainers, to CEOs of companies. LUGs generally meet on a regular basis for group discussions and hold presentations to demonstrate ways they've found to implement Linux and other related technology.

Some LUGs sponsor local events like install fests (http://en.wikipedia.org/wiki/Install_fest) or other Linux advocacy-type events. Chances are good that if you ask a question at a LUG meeting, someone (but more likely several) will have an answer. A search engine should help you locate a LUG in your area if you decide to pursue this. Most LUGs have web sites or mailing lists that can be easily found online.

Ubuntu Software

Most Ubuntu software can be found on the Ubuntu package web site (<http://packages.ubuntu.com>). The standard tools — Synaptic, APT, and Update Manager — are the most common ways of installing software on your Ubuntu system (Chapter 2 provides details on finding and installing software).

Someday though, you may want to experiment and look for software that is not available in the Ubuntu packages. Most packages will have an MD5sum or GPG key you can use to verify that downloaded software hasn't been tampered with (www.debian-administration.org/articles/375). You can also run into compatibility issues with non-standard software, making upgrades a difficult task. The key to experimenting with non-standard software is to test it out in ways that do not alter your system. The following list includes some web sites you can check out to see what other software is out there.

WARNING! *You should be careful about how you go about mixing the software on your Ubuntu system with software from non-Ubuntu sources. We advise you to check carefully the authenticity of anything you download.*

- ❑ www.happyenguin.org — The Linux Game Tome. A collection of games, games, and more games for Linux, varying from experimental demos to commercial game companies that have provided source code or binaries to run their games under Linux. (Thanks, commercial game companies!)
- ❑ www.freshmeat.net — Boasts the Web's largest collection of Unix and cross-platform software, themes, eye-candy, and Palm-OS software. It also sports a discussion board for each software entry to facilitate discussions and feedback. These guys have been around for a very long time.
- ❑ <http://sourceforge.net> — When open source developers get together to start a new project, many go to SourceForge to get their project hosted there. SourceForge offers web space as well as tools for managing projects, resources, communications, and code. If you are looking for software, certainly try some searching here.

Chapter 1: Starting with Ubuntu Linux

- ❑ www.linux.org/apps — Linux.org has been around since 1994; it was born out of the need for housing information about the Linux movement. This web site is a growing source of information on everything Linux. The apps section of Linux.org will direct you to a wide variety of software available for Linux systems.

Focusing on Linux Commands

These days, many important tasks in Linux can be done from both graphical interfaces and from commands. However, the command line has always been, and still remains, the interface of choice for Linux power users.

Graphical user interfaces (GUIs) are meant to be intuitive. With some computer experience, you can probably figure out, for example, how to add a user, change the time and date, and configure a sound card from a GUI. For cases such as these, we'll mention which graphical tool you could use for the job. For the following cases, however, you will probably need to rely on the command line:

- ❑ **Almost any time something goes wrong** — Ask a question at an online forum to solve some Linux problem you are having and the help you get will almost always come in the form of commands to run. Also, command line tools typically offer much more feedback if there is a problem configuring a device or accessing files and directories.
- ❑ **Remote systems administration** — If you are administering a remote server, you may not have graphical tools available. Although remote GUI access (using X applications or VNC) and web-based administration tools may be available, they usually run more slowly than what you can do from the command line.
- ❑ **Features not supported by GUI** — GUI administration tools tend to present the most basic ways of performing a task. More complex operations often require options that are only available from the command line.
- ❑ **GUI is broken or not installed** — If no graphical interface is available, or if the installed GUI isn't working properly, you may be forced to work from the command line. Broken GUIs can happen for lots of reasons, such as when you use a third-party, binary-only driver from NVIDIA or ATI and a kernel upgrade makes the driver incompatible.

The bottom line is that to unlock the full power of your Linux system, you must be able to use shell commands. Thousands of commands are available for Linux to monitor and manage every aspect of your Linux system.

But whether you are a Linux guru or novice, one challenge looms large. How do you remember the most critical commands and options you need, when a command shell might only show you this:

```
$
```

Chapter 1: Starting with Ubuntu Linux

Ubuntu Linux Toolbox is not just another command reference or rehash of man pages. Instead, this book presents commands in Ubuntu Linux by the way you use them. In other words, instead of listing commands alphabetically, we group commands for working with file systems, connecting to networks, and managing processes in their own sections, so you can access commands by what you want to do, not only by how they're named.

Likewise, we won't just give you a listing of every option available for every command. Instead, we'll show you working examples of the most important and useful options to use with each command. Then, we'll show you quick ways to find more options, if you need them, from man pages, the info facility, and help options.

Finding Commands

Some of the commands in this book may not be installed by default on your Ubuntu distro, but will certainly be available through APT or other sources. When you type a command that the bash shell cannot find, you will see an error message similar to this:

```
$ sillycommand
-bash: sillycommand: command not found
```

There are a few reasons why the `command not found` message is returned:

- You may have mistyped the command name (“fat-fingered” it).
- The command is not in any of the directories the shell has been instructed to look in (`PATH` variable).
- The command may only be available to root (through the use of `sudo` or a similar method).
- The command, or Ubuntu package containing the command, is simply not installed.

Table 1-1 shows some shell commands you can run on any Linux distribution to check whether the command you typed is on the system. Table 1-2 shows similar commands specific to Ubuntu and Debian systems.

NOTE *You may see an ellipsis (...) used in code output to note where non-essential information has been omitted for the sake of brevity.*

If you find a command listed in the output of `apt-cache search` or suspect that the command you want is not installed, you can install it from the Internet by running the command

```
$ sudo apt-get install packagename
```

where *packagename* is the name of the package you want to install.

Table 1-1: Generic Linux Commands for Finding Installed Commands

Command and Sample Output	Description
<pre>\$ echo \$PATH /usr/local/sbin:/usr/local/bin:/usr/sbin:/usr/bin:/sbin:/bin:/usr/games</pre>	Show the current PATH.
<pre>\$ which mount /bin/mount</pre>	Find the first occurrence of the mount command in the PATH.
<pre>\$ find /usr -name umount /usr/lib/klibc/bin/umount</pre>	Search the /usr file system for a filename or directory named umount.
<pre>\$ whereis mount mount: /bin/mount /usr/share/man/man8/mount.8.gz</pre>	Show where the first binary and man page are for the mount command.
<pre>\$ locate mount ... /usr/bin/fdmountd</pre>	Use the locate command to search its list of (configurable) directories for mount.
<pre>\$ apropos umount ... umount (8) - unmount file systems</pre>	Search the man page descriptions for instances of a keyword; in this example, umount.
<pre>\$ man 8 umount Reformatting umount(8), please wait...</pre>	View section 8 of the man page for umount (type q to quit).

Table 1-2: Ubuntu/Debian-Specific Commands for Finding Installed Commands

Command and Sample Output	Description
<pre>\$ apt-cache search amount gnome-mount - wrapper for (un)mounting and ejecting storage devices --</pre>	<p>Search the cached list of packages that may contain a command or description of amount.</p>
<pre>\$ dpkg-query -f='\${Package} \${Version} \${Architecture}\n' -W initscripts: /etc/init.d/umountnfs.sh --</pre>	<p>Search the list of installed packages for the filename amount, revealing the package it is in.</p>
<pre>\$ dpkg -L initscripts ... /bin/mountpoint --</pre>	<p>List all the files contained in the initscripts package</p>
<pre>\$ sudo apt-get update Password: Get:1 http://security.ubuntu.com feisty-security Release.gpg [191B] ... --</pre>	<p>Refresh the list of cached packages.</p>

Reference Information in Ubuntu

Original Linux and Unix documentation was all done on manual pages, generally referred to as *man pages*. A slightly more sophisticated documentation effort came a bit later with the GNU *info* facility. Within each command itself, help messages are almost always available.

This reference information is component oriented. There are separate man pages for nearly every command installed on the system. Man pages also document devices, file formats, system, developer info, and many other components of a Linux system. Documentation more closely aligned to whole software packages is typically stored in a subdirectory of the `/usr/share/doc` directory.

Ubuntu compresses much of this documentation, so it needs to be uncompressed before it can be read. You can use the `gzip` program to do this, but instruct `gzip` to only print the contents of the file and not decompress the files to disk. Here's the command to unzip the documentation for the `mount` command:

```
$ gzip -dc /usr/share/doc/mount/README.mount.gz
mount/umount for Linux 0.97.3 and later.
...
```

The man pages, info facility, and `/usr/share/doc` directories are all available on most Linux systems.

Using help Messages

Nearly all commands on a Linux system print some form of brief usage information if asked to. Frequently, the way to ask for this usage info is by way of the `-h` or `--help` argument to the command, and nothing more. The following command shows how to ask the `ls` command to print its usage information.

```
$ ls --help
Usage: ls [OPTION]... [FILE]...
List information about the FILEs (the current directory by default).
...
```

Since there is so much information printed by the `--help` flag, you can again use a pager to limit the output to one screen at a time:

```
$ ls --help | more
...
```

NOTE The `more` command is a popular pager command, which you will find on nearly every Unix system in use. Linux systems favor a command called `less` which is whimsically named and ironically more functional than the `more` command. It allows paging backwards in the output as well as forwards, allows the use of the arrow keys to scroll, and understands *vi* editor keystrokes for navigating and searching through text.

Chapter 1: Starting with Ubuntu Linux

The preceding examples show how to output the `ls` command help to the screen. You can also format the help output with the use of the `card` command, which will print directly to the default printer, or can be saved to a Postscript file to be viewed later with something like the `evince` utility, or converted into a PDF file with the `ps2pdf` utility.

Using man Pages

Table 1-1 briefly covered using the `apropos` command for finding the man page section for the `umount` command. You can use the `apropos` command to search the man page database for any keyword or group of characters. The output will show man page sections which contain the word you supply to `apropos`.

```
$ apropos crontab
/etc/anacrontab (5) [anacrontab] - configuration file for anacron
anacrontab (5) - configuration file for anacron
crontab (1) - maintain crontab files for individual users (V3)
crontab (5) - tables for driving cron
```

The `apropos` output here shows the section and man page where the word `crontab` was found. Sections of man pages are ways of grouping man pages by topic. Man pages in section 1 are *Executable programs or shell commands*. Man pages in section 5 fall under the topic of *File formats and conventions*. The man page sections will be the same on all Linux systems, but may vary a bit on other Unix-type systems. You should be able to view the man page for `man` to find out which sections are represented on the system you're on:

```
$ man man
Reformatting man(1), please wait...
...
```

Table 1-3 shows the section numbers of the manual followed by the types of pages they contain.

Table 1-3: man Page Sections

Number	Types of Pages
1	Executable programs or shell commands
2	System calls (functions provided by the kernel)
3	Library calls (functions within program libraries)
4	Special files (usually found in <code>/dev</code>)

Chapter 1: Starting with Ubuntu Linux

Table 1-3: man Page Sections (continued)

Number	Types of Pages
5	File formats and conventions such as <code>/etc/passwd</code>
6	Games
7	Miscellaneous (including macro packages and conventions), such as <code>man(7)</code> , <code>groff(7)</code>
8	System administration commands (usually only for root)
9	Kernel routines [Non standard]
...	...

Given this information, we can see the `crontab` word we searched for has an entry in section 1 (*Executable programs or shell commands*) as well as section 5 (*File formats and conventions*). We can view the man pages from those sections by passing the section number as an argument to the `man` command.

```
$ man 5 crontab
Reformatting crontab(5), please wait...
CRONTAB(5)                                CRONTAB(5)

NAME
    crontab - tables for driving cron

DESCRIPTION
    A crontab file contains instructions to the cron(8) daemon of
    the general form: ``run this command at this time on this date''.
    ...
```

If we omit the section number, `man` will return the man page from the first section it finds. In the next example, `man` returns section 1 of the `crontab` man pages.

```
$ man crontab
Reformatting crontab(1), please wait...
CRONTAB(1)                                CRONTAB(1)

NAME
    crontab - maintain crontab files for individual users (V3)
    ...
```

In addition to section numbers, the `man` command takes several arguments to perform different tasks. Table 1-4 shows some examples.

Chapter 1: Starting with Ubuntu Linux

Table 1-4: man Command Options

Option	Description
<code>man -a crontab</code>	Shows all man page sections, in succession, for <code>crontab</code>
<code>man 5 crontab</code>	Shows the section 5 man page for <code>crontab</code>
<code>man crontab -P more</code>	Uses the pager program <code>more</code> for paging through the <code>crontab</code> man page
<code>man -f crontab</code>	Equivalent to the <code>whatis</code> command
<code>man -k crontab</code>	Equivalent to the <code>apropos</code> command

The `whatis` command is another man page searching utility. It is different from `apropos` in that it only prints man page descriptions that match the keyword you type in. Running the `apropos` command for the `route` command returns three different man pages where a reference to the word `route` was found:

```
$ apropos route
NETLINK_ROUTE (7) - Linux IPv4 routing socket
route (8) - show / manipulate the IP routing table
traceroute6 (8) - traces path to a network host
```

In running `whatis` for the `route` command, only the section 8 man page for the `route` command is returned:

```
$ whatis route
route (8) - show / manipulate the IP routing table
```

Using info Documents

In some cases, developers have put more complete descriptions of commands, file formats, devices, or other Linux components in the `info` database, a sort of linked set of online manual pages. You can enter the `info` database by simply typing the `info` command or by opening a particular component (use the `q` key to quit the `info` utility).

```
$ info ls
```

The previous command shows information on the `ls` command. You can navigate around the `info` utility using the up, down, left, and right arrow keys, as well as the Page Up and Page Down keys. Table 1-5 shows more about navigating in `info`.

Software packages that have particularly extensive text available in the `info` database include `gimp`, `festival`, `libc`, `automake`, `zsh`, `sed`, `tar`, and `bash`. Files used by the `info` database are stored in the `/usr/share/info` directory.

Table 1-5: Moving through the info Screen

Keystroke	Movement
?	Display the basic commands to use in info screens.
Shift+l	Go back to the previous node you were viewing.
n , p, u	Go to the node that is next, previous, or up, respectively.
Enter	Go to the hyperlink that is under the cursor.
Shift+r	Follow a cross reference.
q or Shift+q	Quit and exit from info.

Summary

In one short chapter, we've covered some of the differences and similarities of Ubuntu Linux as compared to other Linux distributions and other Unix-like systems. You've learned about several online resources specifically for Ubuntu as well as those for Linux in general.

You found out where to find Ubuntu-specific software as well as other Linux software. You installed a few packages using the Debian Advanced Package Tool (APT) and worked with ways of searching for commands and man pages on the system. You also worked with the `stdin` and `stdout` I/O streams by redirecting command output (`stdout`) to temporary files as well as the input streams (`stdin`) of other commands.

While you certainly can read this book from cover-to-cover if you like, it was designed to be a reference to hundreds of features in Ubuntu and Debian Linux that are the most useful to power users and systems administrators. Because information is organized by topic, instead of alphabetically, you don't have to know the commands in advance to find what you need to get the job done.

Most of the features described in this book will work equally well in all Linux-based systems, and many will carry over to legacy Unix systems as well.

