

Matthew Helmke

**Ubuntu**  
**15.10**  
on DVD

# Ubuntu

**UNLEASHED**

**SAMS**

**2016 Edition**  
Covering 15.10 and 16.04

**FREE SAMPLE CHAPTER**

SHARE WITH OTHERS



Matthew Helmke  
with Andrew Hudson  
and Paul Hudson

# Ubuntu

**UNLEASHED**

2016 Edition



800 East 96th Street, Indianapolis, Indiana 46240 USA

## **Ubuntu Unleashed 2016 Edition**

Copyright © 2016 by Pearson Education, Inc.

All rights reserved. No part of this book shall be reproduced, stored in a retrieval system, or transmitted by any means, electronic, mechanical, photocopying, recording, or otherwise, without written permission from the publisher. No patent liability is assumed with respect to the use of the information contained herein. Although every precaution has been taken in the preparation of this book, the publisher and author assume no responsibility for errors or omissions. Nor is any liability assumed for damages resulting from the use of the information contained herein.

ISBN-13: 978-0-13-426811-8

ISBN-10: 0-13-426811-3

Library of Congress Cataloging-in-Publication Data: 2015913547

Printed in the United States of America

First Printing November 2015

### **Trademarks**

All terms mentioned in this book that are known to be trademarks or service marks have been appropriately capitalized. Sams Publishing cannot attest to the accuracy of this information. Use of a term in this book should not be regarded as affecting the validity of any trademark or service mark.

### **Warning and Disclaimer**

Every effort has been made to make this book as complete and as accurate as possible, but no warranty or fitness is implied. The information provided is on an “as is” basis. The author and the publisher shall have neither liability nor responsibility to any person or entity with respect to any loss or damages arising from the information contained in this book.

### **Special Sales**

For information about buying this title in bulk quantities, or for special sales opportunities (which may include electronic versions; custom cover designs; and content particular to your business, training goals, marketing focus, or branding interests), please contact our corporate sales department at [corpsales@pearsoned.com](mailto:corpsales@pearsoned.com) or (800) 382-3419.

For government sales inquiries, please contact [governmentsales@pearsoned.com](mailto:governmentsales@pearsoned.com).

For questions about sales outside the U.S., please contact [international@pearsoned.com](mailto:international@pearsoned.com).

### **Editor-in-Chief**

Mark Taub

### **Acquisitions Editor**

Debra Williams  
Cauley

### **Managing Editor**

Kristy Hart

### **Project Editor**

Andy Beaster

### **Indexer**

Cheryl Lenser

### **Proofreader**

Sarah Kearns

### **Technical Editor**

Micah Brown

### **Editorial Assistant**

Kim Boedigheimer

### **Media Producer**

Dan Scherf

### **Cover Designer**

Mark Shirar

### **Compositor**

Nonie Ratcliff

# Contents at a Glance

Introduction .....	xxxi
<b>Part I Getting Started</b>	
<b>1</b> Installing Ubuntu and Post-Installation Configuration .....	1
<b>2</b> Background Information and Resources .....	27
<b>Part II Desktop Ubuntu</b>	
<b>3</b> Working with Unity .....	41
<b>4</b> On the Internet .....	61
<b>5</b> Productivity Applications .....	75
<b>6</b> Multimedia Applications .....	85
<b>7</b> Other Ubuntu Interfaces .....	109
<b>8</b> Games .....	119
<b>Part III System Administration</b>	
<b>9</b> Managing Software .....	131
<b>10</b> Command-Line Quickstart .....	145
<b>11</b> Command-Line Master Class .....	179
<b>12</b> Managing Users .....	231
<b>13</b> Automating Tasks and Shell Scripting .....	257
<b>14</b> The Boot Process .....	303
<b>15</b> System-Monitoring Tools .....	315
<b>16</b> Backing Up .....	335
<b>17</b> Networking .....	363
<b>18</b> Remote Access with SSH, Telnet, and VNC .....	409
<b>19</b> Securing Your Machines .....	419
<b>20</b> Performance Tuning .....	435
<b>21</b> Kernel and Module Management .....	447
<b>Part IV Ubuntu as a Server</b>	
<b>22</b> Sharing Files and Printers .....	469
<b>23</b> Apache Web Server Management .....	485
<b>24</b> Nginx Web Server Management .....	517

<b>25</b>	Other Http Servers .....	529
<b>26</b>	Remote File Serving with FTP .....	533
<b>27</b>	Handling Email .....	543
<b>28</b>	Proxying, Reverse Proxying, and Virtual Private Networks (VPN) .....	563
<b>29</b>	Administering Relational Database Services .....	579
<b>30</b>	NoSQL Databases .....	605
<b>31</b>	Lightweight Directory Access Protocol (LDAP) .....	617
<b>32</b>	Linux Terminal Server Project (LTSP) .....	625
<b>33</b>	Virtualization on Ubuntu .....	633
<b>34</b>	Ubuntu in the Cloud .....	643
<b>35</b>	Managing Sets of Servers .....	657
<b>36</b>	Name Serving with the Domain Name System (DNS) .....	661

**Part V Programming Linux**

<b>37</b>	Using Programming Tools for Ubuntu .....	671
<b>38</b>	Opportunistic Development .....	683
<b>39</b>	Helping with Ubuntu Development .....	699
<b>40</b>	Helping with Ubuntu Testing and QA .....	709
<b>41</b>	Using Popular Programming Languages .....	715
<b>42</b>	Beginning Mobile Development for Android .....	729
<b>43</b>	Developing for Ubuntu Mobile/Touch .....	735
	Index .....	739

**NOTE**

Chapters 44–46 can be accessed online at [informit.com/title/9780134268118](http://informit.com/title/9780134268118).

**Part VI Bonus Chapters**

<b>44</b>	Using Perl .....	Web:1
<b>45</b>	Using Python .....	Web:23
<b>46</b>	Using PHP .....	Web:43

# Table of Contents

<b>Introduction</b>	<b>xxxi</b>
Licensing	xxxii
Who This Book Is For	xxxiii
Those Wanting to Become Intermediate or Advanced Users	xxxiii
Sysadmins, Programmers, and DevOps	xxxiv
What This Book Contains	xxv
Conventions Used in This Book	xxv
<b>Part I Getting Started</b>	
<b>1 Installing Ubuntu and Post-Installation Configuration</b>	<b>1</b>
Before You Begin the Installation	1
Researching Your Hardware Specifications	2
Installation Options	2
32-Bit vs. 64-Bit Ubuntu	4
Planning Partition Strategies	5
The Boot Loader	5
Installing from DVD or USB Drive	6
Step-by-Step Installation	6
Installing	7
First Update	12
Shutting Down	12
Finding Programs and Files	13
Software Updater	14
The <code>sudo</code> Command	17
Configuring Software Repositories	18
System Settings	20
Detecting and Configuring a Printer	21
Configuring Power Management in Ubuntu	21
Setting the Time and Date	22
Configuring Wireless Networks	24
Troubleshooting Post-Installation Configuration Problems	25
References	26
<b>2 Background Information and Resources</b>	<b>27</b>
What Is Linux?	27
Why Use Linux?	29
What Is Ubuntu?	31

Ubuntu for Business .....	32
Ubuntu in Your Home .....	33
Getting the Most from Ubuntu and Linux Documentation .....	33
Ubuntu Developers and Documentation .....	35
Websites and Search Engines .....	35
Web Search Tips .....	35
Google Is Your Friend .....	36
Ubuntu Package Listings .....	36
Commercial Support .....	36
Documentation .....	37
Linux Guides .....	37
Ubuntu .....	38
Mailing Lists .....	39
Ubuntu Project Mailing Lists .....	39
Internet Relay Chat .....	40
<b>Part II Desktop Ubuntu</b>	
<b>3 Working with Unity</b> .....	<b>41</b>
Foundations and the X Server .....	41
Basic X Concepts .....	42
Using X .....	43
Elements of the <code>xorg.conf</code> File .....	44
Starting X .....	49
Using a Display Manager .....	50
Changing Window Managers .....	50
Using Unity, a Primer .....	51
The Desktop .....	51
Customizing and Configuring Unity .....	56
Power Shortcuts .....	58
References .....	59
<b>4 On the Internet</b> .....	<b>61</b>
Getting Started with Firefox .....	61
Checking Out Google Chrome and Chromium .....	63
Choosing an Email Client .....	65
Mozilla Thunderbird .....	65
Evolution .....	66
Other Mail Clients .....	67
RSS Readers .....	67
Firefox .....	67
Liferea .....	68

Instant Messaging and Video Conferencing with Empathy .....	68
Internet Relay Chat .....	69
Usenet Newsgroups .....	72
References .....	73
<b>5 Productivity Applications</b> .....	<b>75</b>
Introducing LibreOffice .....	77
Other Office Suites for Ubuntu .....	79
Working with GNOME Office .....	79
Working with KOffice .....	80
Other Useful Productivity Software .....	81
Working with PDF .....	81
Working with XML and DocBook .....	81
Working with LaTeX .....	82
Productivity Applications Written for Microsoft Windows .....	83
References .....	84
<b>6 Multimedia Applications</b> .....	<b>85</b>
Sound and Music .....	85
Sound Cards .....	86
Adjusting Volume .....	87
Sound Formats .....	88
Listening to Music .....	89
Graphics Manipulation .....	91
The GNU Image Manipulation Program .....	93
Using Scanners in Ubuntu .....	94
Working with Graphics Formats .....	95
Capturing Screen Images .....	96
Other Graphics Manipulation Options .....	97
Using Digital Cameras with Ubuntu .....	97
Handheld Digital Cameras .....	98
Using Shotwell Photo Manager .....	98
Burning CDs and DVDs in Ubuntu .....	99
Creating CDs and DVDs with Brasero .....	99
Creating CDs from the Command Line .....	99
Creating DVDs from the Command Line .....	101
Viewing Video .....	103
TV and Video Hardware .....	104
Video Formats .....	105
Viewing Video in Linux .....	105



Personal Video Recorders .....	107
Video Editing .....	107
References .....	108
<b>7 Other Ubuntu Interfaces</b> .....	<b>109</b>
Desktop Environment .....	110
KDE and Kubuntu .....	111
Xfce and Xubuntu .....	112
LXDE and Lubuntu .....	113
GNOME3 and Ubuntu GNOME .....	114
MATE and Ubuntu MATE .....	115
Ubuntu Kylin .....	116
References .....	116
<b>8 Games</b> .....	<b>119</b>
Ubuntu Gaming .....	119
Installing Proprietary Video Drivers .....	120
Installing Games in Ubuntu .....	120
Warsow .....	121
Scorched 3D .....	121
Frozen Bubble .....	123
SuperTux .....	123
Battle for Wesnoth .....	124
Frets on Fire .....	124
FlightGear .....	126
Speed Dreams .....	126
Games for Kids .....	126
Commercial Games .....	126
Steam .....	127
Playing Windows Games .....	128
References .....	128
<b>Part III System Administration</b>	
<b>9 Managing Software</b> .....	<b>131</b>
Ubuntu Software Center .....	131
Using Synaptic for Software Management .....	132
Staying Up-to-Date .....	134
Working on the Command Line .....	135
Day-to-Day Usage .....	136
Finding Software .....	139

Compiling Software from Source .....	140
Compiling from a Tarball .....	140
Compiling from Source from the Ubuntu Repositories .....	141
Configuration Management .....	142
dotdee .....	143
OneConf .....	143
Snappy Ubuntu Core .....	143
References .....	144
<b>10 Command-Line Quickstart</b> .....	<b>145</b>
What Is the Command Line? .....	146
Accessing the Command Line .....	147
Text-Based Console Login .....	148
Logging Out .....	149
Logging In and Out from a Remote Computer .....	149
User Accounts .....	150
Reading Documentation .....	152
Using Man Pages .....	152
Using apropos .....	152
Using whereis .....	153
Understanding the Linux File System Hierarchy .....	153
Essential Commands in /bin and /sbin .....	154
Configuration Files in /etc .....	155
User Directories: /home .....	155
Using the Contents of the /proc Directory to Interact with the Kernel .....	156
Working with Shared Data in the /usr Directory .....	157
Temporary File Storage in the /tmp Directory .....	158
Accessing Variable Data Files in the /var Directory .....	158
Navigating the Linux File System .....	158
Listing the Contents of a Directory with ls .....	158
Changing Directories with cd .....	160
Finding Your Current Directory with pwd .....	161
Working with Permissions .....	161
Assigning Permissions .....	162
Directory Permissions .....	163
Altering File Permissions with chmod .....	164
File Permissions with chgrp .....	165
Changing File Permissions with chown .....	165
Understanding Set User ID and Set Group ID Permissions .....	165

Working with Files .....	167
Creating a File with <code>touch</code> .....	167
Creating a Directory with <code>mkdir</code> .....	167
Deleting a Directory with <code>rmdir</code> .....	168
Deleting a File or Directory with <code>rm</code> .....	169
Moving or Renaming a File with <code>mv</code> .....	169
Copying a File with <code>cp</code> .....	170
Displaying the Contents of a File with <code>cat</code> .....	171
Displaying the Contents of a File with <code>less</code> .....	171
Using Wildcards and Regular Expressions .....	171
Working as Root .....	172
Understanding and Fixing <code>sudo</code> .....	172
Creating Users .....	175
Deleting Users .....	176
Shutting Down the System .....	176
Rebooting the System .....	177
Commonly Used Commands and Programs .....	178
References .....	178
<b>11 Command-Line Master Class</b> .....	<b>179</b>
Why Use the Command Line? .....	180
Using Basic Commands .....	181
Printing the Contents of a File with <code>cat</code> .....	183
Changing Directories with <code>cd</code> .....	183
Changing File Access Permissions with <code>chmod</code> .....	185
Copying Files with <code>cp</code> .....	186
Printing Disk Usage with <code>du</code> .....	186
Finding Files by Searching with <code>find</code> .....	187
Searches for a String in Input with <code>grep</code> .....	189
Paging Through Output with <code>less</code> .....	190
Creating Links Between Files with <code>ln</code> .....	192
Finding Files from an Index with <code>locate</code> .....	194
Listing Files in the Current Directory with <code>ls</code> .....	194
Reading Manual Pages with <code>man</code> .....	196
Making Directories with <code>mkdir</code> .....	197
Moving Files with <code>mv</code> .....	197
Deleting Files and Directories with <code>rm</code> .....	198
Sorting the Contents of a File with <code>sort</code> .....	198
Printing the Last Lines of a File with <code>tail</code> .....	200
Using <code>echo</code> .....	201

Printing the Location of a Command with <code>which</code> .....	202
Redirecting Output and Input .....	202
<code>stdin</code> , <code>stdout</code> , <code>stderr</code> , and Redirection .....	203
Comparing Files .....	204
Finding Differences in Files with <code>diff</code> .....	204
Finding Similarities in Files with <code>comm</code> .....	205
Limiting Resource Use and Job Control .....	205
Listing Processes with <code>ps</code> .....	206
Listing Jobs with <code>jobs</code> .....	207
Running One or More Tasks in the Background .....	207
Moving Jobs to the Background or Foreground with <code>bg</code> and <code>fg</code> .....	208
Printing Resource Usage with <code>top</code> .....	209
Setting Processes Priority with <code>nice</code> .....	211
Combining Commands .....	212
Pipes .....	212
Combining Commands with Boolean Operators .....	214
Running Separate Commands in Sequence .....	214
Using Environment Variables .....	215
Using Common Text Editors .....	218
Working with <code>nano</code> .....	219
Working with <code>vi</code> .....	220
Working with <code>emacs</code> .....	221
Working with <code>sed</code> and <code>awk</code> .....	222
Working with Compressed Files .....	224
Using Multiple Terminals with <code>byobu</code> .....	225
Polite System Reset Using <code>REISUB</code> .....	226
Tips and Tricks .....	227
Running the Previous Command .....	227
Running Any Previous Command .....	228
Running a Previous Command That Started with Specific Letters .....	228
Running the Same Thing You Just Ran with a Different First Word .....	228
Viewing Your History and More .....	228
Do Two or More Things .....	229
Shortcuts .....	229
Coreutils .....	229
References .....	230

<b>12</b>	<b>Managing Users</b>	<b>231</b>
	User Accounts .....	231
	The Super User/Root User .....	232
	User IDs and Group IDs .....	234
	File Permissions .....	234
	Managing Groups .....	235
	Group Listing .....	235
	Group Management Tools .....	237
	Managing Users .....	238
	User Management Tools .....	238
	Adding New Users .....	240
	Monitoring User Activity on the System .....	242
	Managing Passwords .....	243
	System Password Policy .....	243
	The Password File .....	243
	Shadow Passwords .....	244
	Managing Password Security for Users .....	247
	Changing Passwords in a Batch .....	247
	Granting System Administrator Privileges to Regular Users .....	247
	Temporarily Changing User Identity with the <code>su</code> Command .....	248
	Granting Root Privileges on Occasion: The <code>sudo</code> Command .....	250
	Disk Quotas .....	253
	Implementing Quotas .....	253
	Manually Configuring Quotas .....	254
	Related Ubuntu Commands .....	254
	References .....	255
<b>13</b>	<b>Automating Tasks and Shell Scripting</b>	<b>257</b>
	Scheduling Tasks .....	257
	Using <code>at</code> and <code>batch</code> to Schedule Tasks for Later .....	257
	Using <code>cron</code> to Run Jobs Repeatedly .....	260
	Using <code>rtcwake</code> to Wake Your Computer from Sleep Automatically .....	262
	Basic Shell Control .....	264
	The Shell Command Line .....	265
	Shell Pattern-Matching Support .....	266
	Redirecting Input and Output .....	267
	Piping Data .....	268
	Background Processing .....	269

Writing and Executing a Shell Script .....	269
Running the New Shell Program .....	271
Storing Shell Scripts for System-Wide Access .....	272
Interpreting Shell Scripts Through Specific Shells .....	272
Using Variables in Shell Scripts .....	273
Assigning a Value to a Variable .....	274
Accessing Variable Values .....	274
Positional Parameters .....	275
A Simple Example of a Positional Parameter .....	275
Using Positional Parameters to Access and Retrieve Variables from the Command Line .....	276
Using a Simple Script to Automate Tasks .....	276
Built-In Variables .....	278
Special Characters .....	279
Using Double Quotes to Resolve Variables in Strings with Embedded Spaces .....	280
Using Single Quotes to Maintain Unexpanded Variables .....	281
Using the Backslash as an Escape Character .....	281
Using the Backtick to Replace a String with Output .....	282
Comparison of Expressions in <code>pdksh</code> and <code>bash</code> .....	282
Comparing Expressions with <code>tcsh</code> .....	287
The <code>for</code> Statement .....	291
The <code>while</code> Statement .....	293
The <code>until</code> Statement .....	295
The <code>repeat</code> Statement ( <code>tcsh</code> ) .....	295
The <code>select</code> Statement ( <code>pdksh</code> ) .....	296
The <code>shift</code> Statement .....	296
The <code>if</code> Statement .....	297
The <code>case</code> Statement .....	298
The <code>break</code> and <code>exit</code> Statements .....	300
Using Functions in Shell Scripts .....	300
References .....	301
<b>14 The Boot Process .....</b>	<b>303</b>
Running Services at Boot .....	303
Beginning the Boot Loading Process .....	304
Loading the Linux Kernel .....	306
System Services and Runlevels .....	306
Runlevel Definitions .....	306
Booting into the Default Runlevel .....	307
Understanding <code>init</code> Scripts and the Final Stage of Initialization .....	308

Controlling Services at Boot with Administrative Tools .....	309
Changing Runlevels .....	309
Troubleshooting Runlevel Problems .....	310
Starting and Stopping Services Manually .....	310
Using Upstart .....	311
systemd .....	312
Boot Repair .....	313
References .....	313
<b>15 System-Monitoring Tools</b> .....	<b>315</b>
Console-Based Monitoring .....	315
Using the <code>kill</code> Command to Control Processes .....	317
Using Priority Scheduling and Control .....	318
Displaying Free and Used Memory with <code>free</code> .....	319
Disk Space .....	320
Disk Quotas .....	321
Checking Log Files .....	321
Rotating Log Files .....	323
Graphical Process and System Management Tools .....	325
System Monitor .....	326
Conky .....	327
Other .....	332
KDE Process- and System-Monitoring Tools .....	332
Enterprise Server Monitoring .....	333
Landscape .....	333
Other .....	333
References .....	333
<b>16 Backing Up</b> .....	<b>335</b>
Choosing a Backup Strategy .....	335
Why Data Loss Occurs .....	336
Assessing Your Backup Needs and Resources .....	337
Evaluating Backup Strategies .....	339
Making the Choice .....	342
Choosing Backup Hardware and Media .....	342
Removable Storage Media .....	342
CD-RW and DVD+RW/-RW Drives .....	343
Network Storage .....	343
Tape Drive Backup .....	343
Cloud Storage .....	344

Using Backup Software .....	344
tar: The Most Basic Backup Tool .....	345
The GNOME File Roller .....	347
The KDE ark Archiving Tool .....	347
Déjà Dup .....	348
Back In Time .....	350
Unison .....	352
Using the Amanda Backup Application .....	352
Alternative Backup Software .....	353
Copying Files .....	354
Copying Files Using tar .....	354
Compressing, Encrypting, and Sending tar Streams .....	355
Copying Files Using cp .....	355
Copying Files Using mc .....	356
Using rsync .....	356
Version Control for Configuration Files .....	358
System Rescue .....	360
The Ubuntu Rescue Disc .....	361
Restoring the GRUB2 Boot Loader .....	361
Saving Files from a Nonbooting Hard Drive .....	362
References .....	362
<b>17 Networking</b> .....	<b>363</b>
Laying the Foundation: The localhost Interface .....	364
Checking for the Availability of the Loopback Interface .....	364
Configuring the Loopback Interface Manually .....	364
Checking Connections with ping, traceroute, and mtr .....	366
Networking with TCP/IP .....	368
TCP/IP Addressing .....	369
Using IP Masquerading in Ubuntu .....	371
Ports .....	372
IPv6 Basics .....	372
Network Organization .....	375
Subnetting .....	375
Subnet Masks .....	376
Broadcast, Unicast, and Multicast Addressing .....	376
Hardware Devices for Networking .....	377
Network Interface Cards .....	377
Network Cable .....	379
Hubs and Switches .....	380
Routers and Bridges .....	381
Initializing New Network Hardware .....	381



Using Network Configuration Tools .....	384
Command-Line Network Interface Configuration .....	384
Network Configuration Files .....	389
Using Graphical Configuration Tools .....	391
Dynamic Host Configuration Protocol .....	393
How DHCP Works .....	393
Activating DHCP at Installation and Boot Time .....	394
DHCP Software Installation and Configuration .....	395
Using DHCP to Configure Network Hosts .....	397
Other Uses for DHCP .....	399
Wireless Networking .....	399
Support for Wireless Networking in Ubuntu .....	399
Advantages of Wireless Networking .....	401
Choosing from Among Available Wireless Protocols .....	401
Beyond the Network and onto the Internet .....	402
Common Configuration Information .....	402
Configuring Digital Subscriber Line Access .....	404
Understanding PPP over Ethernet .....	404
Configuring a PPPoE Connection Manually .....	405
Configuring Dial-Up Internet Access .....	406
Troubleshooting Connection Problems .....	407
References .....	408
<b>18 Remote Access with SSH, Telnet, and VNC</b> .....	<b>409</b>
Setting Up a Telnet Server .....	409
Telnet Versus SSH .....	411
Setting Up an SSH Server .....	411
SSH Tools .....	411
Using <code>scp</code> to Copy Individual Files Between Machines .....	412
Using <code>sftp</code> to Copy Many Files Between Machines .....	413
Using <code>ssh-keygen</code> to Enable Key-Based Logins .....	413
Virtual Network Computing .....	415
References .....	417
<b>19 Securing Your Machines</b> .....	<b>419</b>
Understanding Computer Attacks .....	419
Assessing Your Vulnerability .....	421
Protecting Your Machine .....	422
Securing a Wireless Network .....	423
Passwords and Physical Security .....	423
Configuring and Using Tripwire .....	424
Devices .....	425

Viruses .....	425
Configuring Your Firewall .....	426
AppArmor .....	430
Forming a Disaster Recovery Plan .....	432
References .....	433
<b>20 Performance Tuning</b> .....	<b>435</b>
Hard Disk .....	435
Using the BIOS and Kernel to Tune the Disk Drives .....	436
The <code>hdparm</code> Command .....	437
File System Tuning .....	438
The <code>tune2fs</code> Command .....	438
The <code>e2fsck</code> Command .....	439
The <code>badblocks</code> Command .....	439
Disabling File Access Time .....	439
Kernel .....	440
Apache .....	441
MySQL .....	442
Measuring Key Buffer Usage .....	442
Using the Query Cache .....	444
Miscellaneous Tweaks .....	445
Query Optimization .....	446
References .....	446
<b>21 Kernel and Module Management</b> .....	<b>447</b>
The Linux Kernel .....	448
The Linux Source Tree .....	449
Types of Kernels .....	451
Managing Modules .....	452
When to Recompile .....	454
Kernel Versions .....	455
Obtaining the Kernel Sources .....	456
Patching the Kernel .....	457
Compiling the Kernel .....	458
Using <code>xconfig</code> to Configure the Kernel .....	461
Creating an Initial RAM Disk Image .....	464
When Something Goes Wrong .....	465
Errors During Compile .....	465
Runtime Errors, Boot Loader Problems, and Kernel Oops .....	466
References .....	466

**Part IV Ubuntu as a Server**

<b>22</b>	<b>Sharing Files and Printers</b>	<b>469</b>
	Using the Network File System .....	470
	Installing and Starting or Stopping NFS .....	470
	NFS Server Configuration .....	470
	NFS Client Configuration .....	472
	Putting Samba to Work .....	472
	Manually Configuring Samba with <code>/etc/samba/smb.conf</code> .....	474
	Testing Samba with the <code>testparm</code> Command .....	477
	Starting, Stopping, and Restarting the <code>smbd</code> Daemon .....	477
	Mounting Samba Shares .....	479
	Network and Remote Printing with Ubuntu .....	479
	Creating Network Printers .....	479
	Using the Common UNIX Printing System GUI .....	481
	Avoiding Printer Support Problems .....	483
	References .....	484
<b>23</b>	<b>Apache Web Server Management</b>	<b>485</b>
	About the Apache Web Server .....	485
	Installing the Apache Server .....	486
	Installing from the Ubuntu Repositories .....	486
	Building the Source Yourself .....	488
	Starting and Stopping Apache .....	490
	Starting the Apache Server Manually .....	490
	Using <code>/etc/init.d/apache2</code> .....	492
	Runtime Server Configuration Settings .....	493
	Runtime Configuration Directives .....	493
	Editing <code>apache2.conf</code> .....	494
	Apache Multiprocessing Modules .....	497
	Using <code>.htaccess</code> Configuration Files .....	497
	File System Authentication and Access Control .....	499
	Restricting Access with <code>allow</code> and <code>deny</code> .....	500
	Authentication .....	501
	Final Words on Access Control .....	503
	Apache Modules .....	504
	<code>mod_dir</code> and <code>mod_env</code> .....	506
	<code>mod_info</code> and <code>mod_log_config</code> .....	507
	<code>mod_mime</code> and <code>mod_mime_magic</code> .....	507
	Virtual Hosting .....	509
	Address-Based Virtual Hosts .....	509
	Name-Based Virtual Hosts .....	510

Logging .....	511
HTTPS .....	513
References .....	515
<b>24 Nginx Web Server Management</b>	<b>517</b>
About the Nginx Web Server .....	517
Installing the Nginx Server .....	519
Installing from the Ubuntu Repositories .....	519
Building the Source Yourself .....	519
Configuring the Nginx Server .....	520
Virtual Hosting .....	523
Setting Up PHP .....	524
Adding and Configuring Modules .....	525
HTTPS .....	526
References .....	528
<b>25 Other HTTP Servers</b>	<b>529</b>
lighttpd .....	529
Yaws .....	530
Cherokee .....	531
Jetty .....	531
thttpd .....	532
Apache Tomcat .....	532
References .....	532
<b>26 Remote File Serving with FTP</b>	<b>533</b>
Choosing an FTP Server .....	533
Choosing an Authenticated or Anonymous Server .....	534
Ubuntu FTP Server Packages .....	534
Other FTP Servers .....	534
Installing FTP Software .....	535
The FTP User .....	536
Configuring the Very Secure FTP Server .....	538
Controlling Anonymous Access .....	539
Other vsftpd Server Configuration Files .....	539
Using the <code>ftphosts</code> File to Allow or Deny FTP Server Connection .....	541
References .....	542
<b>27 Handling Email</b>	<b>543</b>
How Email Is Sent and Received .....	543
The Mail Transport Agent .....	544
Choosing an MTA .....	546

The Mail Delivery Agent .....	546
The Mail User Agent .....	547
Basic Postfix Configuration and Operation .....	548
Configuring Masquerading .....	550
Using Smart Hosts .....	551
Setting Message Delivery Intervals .....	551
Mail Relaying .....	552
Forwarding Email with Aliases .....	552
Using Fetchmail to Retrieve Mail .....	553
Installing Fetchmail .....	553
Configuring Fetchmail .....	553
Choosing a Mail Delivery Agent .....	557
Procmail .....	557
Spamassassin .....	557
Squirrelmail .....	558
Virus Scanners .....	558
Autoresponders .....	558
Alternatives to Microsoft Exchange Server .....	558
Microsoft Exchange Server/Outlook Client .....	559
CommuniGate Pro .....	559
Oracle Beehive .....	560
Bynari .....	560
Open-Xchange .....	560
Horde .....	560
References .....	560
<b>28 Proxying, Reverse Proxying, and Virtual Private Networks (VPN) .....</b>	<b>563</b>
What Is a Proxy Server? .....	563
Installing Squid .....	564
Configuring Clients .....	564
Access Control Lists .....	564
Specifying Client IP Addresses .....	569
Sample Configurations .....	570
Virtual Private Networks (VPN) .....	572
Setting Up a VPN Client .....	574
Setting Up a VPN Server .....	575
References .....	577
<b>29 Administering Relational Database Services .....</b>	<b>579</b>
A Brief Review of Database Basics .....	580
How Relational Databases Work .....	582
Understanding SQL Basics .....	584

Creating Tables .....	584
Inserting Data into Tables .....	585
Retrieving Data from a Database .....	586
Choosing a Database: MySQL Versus PostgreSQL .....	588
Speed .....	588
Data Locking .....	589
ACID Compliance in Transaction Processing to Protect Data Integrity .....	589
SQL Subqueries .....	590
Procedural Languages and Triggers .....	590
Configuring MySQL .....	591
Setting a Password for the MySQL Root User .....	592
Creating a Database in MySQL .....	592
Configuring PostgreSQL .....	594
Initializing the Data Directory in PostgreSQL .....	594
Creating a Database in PostgreSQL .....	595
Creating Database Users in PostgreSQL .....	596
Deleting Database Users in PostgreSQL .....	596
Granting and Revoking Privileges in PostgreSQL .....	597
Database Clients .....	597
SSH Access to a Database .....	598
Local GUI Client Access to a Database .....	599
Web Access to a Database .....	600
The MySQL Command-Line Client .....	601
The PostgreSQL Command-Line Client .....	602
Graphical Clients .....	602
References .....	603
<b>30 NoSQL Databases .....</b>	<b>605</b>
Key/Value Stores .....	608
Berkeley DB .....	608
Cassandra .....	609
Memcached and MemcacheDB .....	609
Redis .....	610
Riak .....	610
Document Stores .....	610
CouchDB .....	611
MongoDB .....	612
BaseX .....	612
Wide Column Stores .....	613
BigTable .....	613
HBase .....	613

Graph Stores .....	614
Neo4j .....	614
OrientDB .....	614
HyperGraphDB .....	615
FlockDB .....	615
References .....	615
<b>31 Lightweight Directory Access Protocol (LDAP)</b> .....	<b>617</b>
Configuring the Server .....	618
Creating Your Schema .....	618
Populating Your Directory .....	620
Configuring Clients .....	622
Evolution .....	622
Thunderbird .....	623
Administration .....	623
References .....	624
<b>32 Linux Terminal Server Project (LTSP)</b> .....	<b>625</b>
Requirements .....	626
Installation .....	629
Using LTSP .....	630
References .....	631
<b>33 Virtualization on Ubuntu</b> .....	<b>633</b>
KVM .....	635
VirtualBox .....	639
VMware .....	641
Xen .....	641
References .....	641
<b>34 Ubuntu in the Cloud</b> .....	<b>643</b>
Why a Cloud? .....	644
Software as a Service (SaaS) .....	645
Platform as a Service (PaaS) .....	645
Infrastructure as a Service (IaaS) .....	645
Metal as a Service (MaaS) .....	645
Before You Do Anything .....	646
Deploy/Install Basics: Public, Private, or Hybrid? .....	646
Ubuntu Cloud and OpenStack .....	647
Compute Infrastructure (Nova) .....	648
Storage Infrastructure (Swift) .....	648
Networking Service (Neutron) .....	648

Identity Service (Keystone) .....	649
Imaging Service (Glance) .....	649
Dashboard (Horizon) .....	649
Learning More .....	649
Juju .....	649
Getting Started .....	650
Charms .....	652
The Juju GUI .....	654
Juju Quickstart .....	654
Juju on Mac OS X and Windows .....	655
Mojo: Continuous Delivery for Juju .....	655
Snappy Ubuntu .....	655
Ubuntu Metal as a Service (MaaS) .....	655
Landscape .....	656
References .....	656
<b>35 Managing Sets of Servers</b> .....	<b>657</b>
Juju .....	657
Puppet .....	658
Chef .....	658
CFEngine .....	658
Ansible .....	659
Landscape .....	659
References .....	659
<b>36 Name Serving with the Domain Name System (DNS)</b> .....	<b>661</b>
Understanding Domain Names .....	663
DNS Servers .....	663
DNS Records .....	664
Setting Up a DNS Server with BIND .....	667
References .....	669
<b>Part V Programming Linux</b>	
<b>37 Using Programming Tools for Ubuntu</b> .....	<b>671</b>
Programming with Linux .....	672
Using the C Programming Project Management Tools	
Provided with Ubuntu .....	673
Building Programs with <code>make</code> .....	673
Using Makefiles .....	673
Using the <code>autoconf</code> Utility to Configure Code .....	675
Debugging Tools .....	676



Using the GNU C Compiler .....	677
Graphical Development Tools .....	678
Using the KDevelop Client .....	678
The Glade Client for Developing in GNOME .....	679
Use an IDE or SDK .....	680
References .....	682
<b>38 Opportunistic Development</b> .....	<b>683</b>
Version Control Systems .....	683
Managing Software Projects with Subversion .....	684
Managing Software Projects with Bazaar .....	685
Managing Software Projects with Mercurial .....	686
Managing Software Projects with Git .....	687
Introduction to Opportunistic Development .....	688
Launchpad .....	689
Quickly .....	691
Bikeshed and Other Tools .....	695
References .....	697
<b>39 Helping with Ubuntu Development</b> .....	<b>699</b>
Introduction to Ubuntu Development .....	700
Setting Up Your Development System .....	701
Install Basic Packages and Configure .....	701
Create a Launchpad Account .....	702
Set Up Your Environment to Work with Launchpad .....	702
Developing Apps and Scopes .....	704
Fixing Bugs and Packaging .....	704
Finding Bugs to Fix with Harvest .....	707
Masters of the Universe .....	707
References .....	708
<b>40 Helping with Ubuntu Testing and QA</b> .....	<b>709</b>
Community Teams .....	709
Ubuntu Testing Team .....	710
QA Team .....	711
Bug Squad .....	711
Test Drive .....	711
References .....	714
<b>41 Using Popular Programming Languages</b> .....	<b>715</b>
Ada .....	716
Clojure .....	717

COBOL .....	717
D .....	718
Dart .....	718
Elixir .....	719
Erlang .....	719
Forth .....	720
Fortran .....	720
Go .....	720
Groovy .....	721
Haskell .....	721
Java .....	721
JavaScript .....	722
Lisp .....	723
Lua .....	723
Mono .....	723
OCaml .....	724
Perl .....	724
PHP .....	725
Python .....	725
Ruby .....	725
Rust .....	726
Scala .....	726
Scratch .....	726
Vala .....	727
References .....	727
<b>42 Beginning Mobile Development for Android</b> .....	<b>729</b>
Introduction to Android .....	730
Hardware .....	730
Linux Kernel .....	730
Libraries .....	730
Android Runtime .....	730
Application Framework .....	731
Applications .....	731
Installing Android Studio .....	731
Install Android Studio .....	731
Install SDK Packages .....	731
Create Your First Application .....	733
References .....	734

<b>43</b>	<b>Developing for Ubuntu Mobile/Touch</b>	<b>735</b>
	Install the SDK .....	736
	Create Your First Application .....	736
	References .....	737
	<b>Index</b>	<b>739</b>

## NOTE

Chapters 44–46 can be accessed online at [informit.com/title/9780134268118](http://informit.com/title/9780134268118).

## Bonus Chapters

<b>44</b>	<b>Using Perl</b>	<b>Web:1</b>
	Using Perl with Linux .....	Web:1
	Perl Versions .....	Web:2
	A Simple Perl Program .....	Web:2
	Perl Variables and Data Structures .....	Web:4
	Perl Variable Types .....	Web:5
	Special Variables .....	Web:5
	Operators .....	Web:6
	Comparison Operators .....	Web:6
	Compound Operators .....	Web:7
	Arithmetic Operators .....	Web:7
	Other Operators .....	Web:8
	Special String Constants .....	Web:9
	Conditional Statements: <code>if/else</code> and <code>unless</code> .....	Web:9
	<code>if</code> .....	Web:9
	<code>unless</code> .....	Web:10
	Looping .....	Web:10
	<code>for</code> .....	Web:11
	<code>foreach</code> .....	Web:11
	<code>while</code> .....	Web:12
	<code>until</code> .....	Web:12
	<code>last</code> and <code>next</code> .....	Web:12
	<code>do ... while</code> and <code>do ... until</code> .....	Web:13
	Regular Expressions .....	Web:13
	Access to the Shell .....	Web:14
	Modules and CPAN .....	Web:15
	Code Examples .....	Web:16
	Sending Mail .....	Web:16
	Purging Logs .....	Web:18
	Posting to Usenet .....	Web:19
	One-Liners .....	Web:20
	Command-Line Processing .....	Web:20
	References .....	Web:21

<b>45</b>	<b>Using Python</b>	<b>Web:23</b>
	Python on Linux .....	Web:24
	The Basics of Python .....	Web:25
	Numbers .....	Web:25
	More on Strings .....	Web:27
	Lists .....	Web:30
	Dictionaries .....	Web:32
	Conditionals and Looping .....	Web:33
	Functions .....	Web:35
	Object Orientation .....	Web:36
	Class and Object Variables .....	Web:37
	Constructors and Destructors .....	Web:38
	Class Inheritance .....	Web:39
	The Standard Library and the Python Package Index .....	Web:41
	References .....	Web:41
<b>46</b>	<b>Using PHP</b>	<b>Web:43</b>
	Introduction to PHP .....	Web:44
	Entering and Exiting PHP Mode .....	Web:44
	Variables .....	Web: 44
	Arrays .....	Web: 46
	Constants .....	Web: 48
	References .....	Web: 48
	Comments .....	Web: 49
	Escape Sequences .....	Web: 49
	Variable Substitution .....	Web: 50
	Operators .....	Web: 51
	Conditional Statements .....	Web: 53
	Special Operators .....	Web: 55
	Switching .....	Web: 55
	Loops .....	Web: 57
	Including Other Files .....	Web: 59
	Basic Functions .....	Web: 60
	Strings .....	Web: 60
	Arrays .....	Web: 64
	Files .....	Web: 65
	Miscellaneous .....	Web: 68
	Handling HTML Forms .....	Web: 71
	Databases .....	Web: 72
	References .....	Web: 74

# About the Authors

**Matthew Helmke** is an active member of the Ubuntu community. He served from 2006 to 2011 on the Ubuntu Forum Council, providing leadership and oversight of the Ubuntu Forums ([www.ubuntuforums.org](http://www.ubuntuforums.org)), and spent two years on the Ubuntu regional membership approval board for Europe, the Middle East, and Africa. He has written about Ubuntu for several magazines and websites, is a lead author of *The Official Ubuntu Book*, and coauthored *The VMware Cookbook*. He works as a senior technical writer for Pearson North America's Assessment and Information division, documenting assessment software. Matthew first used Unix in 1987 while studying LISP on a Vax at the university. He has run a business using only free and open source software, has consulted, and has a master's degree in Information Resources and Library Science from the University of Arizona. You can find out more about Matthew at [matthewhelmke.com](http://matthewhelmke.com) or drop him a line with errata or suggestions at [matthew@matthewhelmke.com](mailto:matthew@matthewhelmke.com).

**Andrew Hudson** is a freelance journalist who specializes in writing about Linux. He has significant experience in Red Hat and Debian-based Linux distributions and deployments and can often be found sitting at his keyboard tweaking various settings and config files just for the hell of it. He lives in Wiltshire, which is a county of England, along with his wife, Bernice, and their son, John. Andrew does not like Emacs. He can be reached at [andy.hudson@gmail.com](mailto:andy.hudson@gmail.com).

**Paul Hudson** is a recognized expert in open-source technologies. He is also a professional developer and full-time journalist for Future Publishing. His articles have appeared in *MacFormat*, *PC Answers*, *PC Format*, *PC Plus*, and *Linux Format*. Paul is passionate about free software in all its forms and uses a mix of Linux and BSD to power his desktops and servers. Paul likes Emacs. Paul can be contacted through <http://hudzilla.org>.

# Dedication

*To Saralyn, Sedona, and Philip—the most amazing kids a guy could hope for; to Sandra and Evan, who are wonderful and welcome additions to our lives; to my grandfather for always believing in me and teaching me to believe in myself; and to my friends in the Ubuntu, developer, sysadmin, cloud computing, and DevOps communities.*

# Acknowledgments

Thank you to the many people who helped with past editions, with helpful comments and ideas, and with technical edits and both formal and informal advice. A special thanks to Curtis Gabrielson who was especially helpful with this edition. I owe a huge debt of gratitude to the Ubuntu community, Canonical, and Mark Shuttleworth for inviting me to participate in the community, including my role in the forums, a turn on the EMEA membership board, and two Ubuntu Developer Summits, back when we had to travel to be a part of them. Thanks to the Ubuntu All Stars for the chance to jam with you on guitar. Thank you to the entire Ubuntu community for your labor of love to create this wonderful operating system. Finally, thanks to my colleagues at Pearson, especially Debra Williams Cauley, for the trust placed in me and the opportunity to collaborate on projects like this one.

# We Want to Hear from You!

As the reader of this book, *you* are our most important critic and commentator. We value your opinion and want to know what we're doing right, what we could do better, what areas you'd like to see us publish in, and any other words of wisdom you're willing to pass our way.

We welcome your comments. You can email or write to let us know what you did or didn't like about this book—as well as what we can do to make our books better.

*Please note that we cannot help you with technical problems related to the topic of this book.*

When you write, please be sure to include this book's title and author as well as your name and email address. We will carefully review your comments and share them with the author and editors who worked on the book.

Email: [consumer@samspublishing.com](mailto:consumer@samspublishing.com)

Mail: Sams Publishing  
ATTN: Reader Feedback  
800 East 96th Street  
Indianapolis, IN 46240 USA

## Reader Services

Visit our website and register this book at [informit.com/register](http://informit.com/register) for convenient access to any updates, downloads, or errata that might be available for this book.

# Introduction

We are pleased to present the 2016 edition of *Ubuntu Unleashed*. Ubuntu is a Linux-based computer operating system that has taken the world by storm. From its humble beginning in 2004, Ubuntu has risen to be the vanguard of desktop Linux, as well as a popular choice for servers.

Ubuntu descends from one of the oldest and most revered Linux distributions, Debian. Debian is assembled by a team of talented volunteers, is one of the most stable and customizable distributions of Linux, and is well respected for its quality and technological prowess. It is, however, an operating system for geeks; the bar for entry into the Debian realm is set high, and its user base tends to be highly proficient and expects new users to learn the ropes before joining in. That is both appropriate and okay.

What Ubuntu has done is leverage the quality of Debian to create an operating system that ordinary people can use. That doesn't mean that Ubuntu users are not technologically proficient, just that they do not have to be. In fact, many talented and respected software developers love Ubuntu because it enables them to concentrate on their specific interests instead of the details of the operating system. This book is for these people and for those who aspire to join their ranks.

If you are new to Linux, you have made a great decision by choosing this book. Sams Publishing's *Unleashed* books offer an in-depth look at their subjects, taking in both beginner and advanced users and moving them to a new level of knowledge and expertise. Ubuntu is a fast-changing distribution that has an updated release twice a year. We have tracked the development of Ubuntu from early on to make sure that the information in this book mirrors closely

## IN THIS INTRODUCTION

- ▶ Licensing
- ▶ Who This Book Is For
- ▶ What This Book Contains
- ▶ Conventions Used in This Book



the development of the distribution. A full copy of Ubuntu is included on the enclosed disc, and it is possible for you to install Ubuntu from that disc in less than an hour!

### A QUICK WORD ABOUT MARKETING

Almost all of the content in this book applies regardless of what Ubuntu release version you are using, so long as it is reasonably current. The book has been written to try to focus on information that is useful for the longest amount of time possible. Some chapters, like those covering installation or the basics of the default Ubuntu graphical user interface, will have their information change frequently. Those chapters are the exception. The blurb on the cover of the book about which editions this book covers was added to account for these chapters and to denote clearly when the book was most recently revised.

---

Do not let the highly technical reputation of Linux discourage you, however. Many people who have heard of Linux think that it is found only on servers, looking after websites and email. Nothing could be further from the truth. Distributions like Ubuntu are making huge inroads in to the desktop market. Corporations are realizing the benefits of running a stable and powerful operating system that is easy to maintain and easy to secure. The best part is that as Linux distributions make improvements, the majority of those improvements are shared freely, allowing you to benefit from the additions and refinements made by one distribution, such as Red Hat, while continuing to use a different distribution, such as Ubuntu, which in turn shares its improvements. You can put Ubuntu to work today and be assured of a great user experience. Feel free to make as many copies of the software as you want; Ubuntu is freely and legally distributable all over the world—no copyright lawyers are going to pound on your door.

## Licensing

Software licensing is an important issue for all computer users and can entail moral, legal, and financial considerations. Many consumers think that purchasing a copy of a commercial or proprietary operating system, productivity application, utility, or game conveys ownership, but this is not true. In the majority of cases, the *end user license agreement (EULA)* included with a commercial software package states that you have paid only for the right to use the software according to specific terms. This generally means you may not examine, make copies, share, resell, or transfer ownership of the software package. More onerous software licenses enforce terms that preclude you from distributing or publishing comparative performance reviews of the software. Even more insidious licensing schemes (and supporting legislation, especially in the United States) contain provisions allowing onsite auditing of the software's use!

This is not the case with the software included with this book. You are entirely free to make copies, share copies, and install the software on as many computers as you want—we encourage you to purchase additional copies of this book to give as gifts, however. Be sure to read the README file on the disc included with this book for important information regarding the included software and disk contents. After you install Ubuntu, go to [www.gnu.org/licenses/gpl.html](http://www.gnu.org/licenses/gpl.html) to find a copy of the GNU GPL. You will see that the GPL

provides unrestricted freedom to use, duplicate, share, study, modify, improve, and even sell the software.

You can put your copy of Ubuntu to work right away in your home or at your place of business without worrying about software licensing, per-seat workstation or client licenses, software auditing, royalty payments, or any other type of payments to third parties. However, be aware that although much of the software included with Ubuntu is licensed under the GPL, some packages on this book's disc are licensed under other terms. There is a variety of related software licenses, and many software packages fall under a broad definition known as *open source*. Some of these include the Artistic License, the BSD License, the Mozilla Public License, and the Q Public License.

For additional information about the various GNU software licenses, browse to [www.gnu.org/](http://www.gnu.org/). For a definition of open-source and licensing guidelines, along with links to the terms of nearly three dozen open-source licenses, browse to [www.opensource.org/](http://www.opensource.org/).

## Who This Book Is For

This book varies its coverage from deep to shallow over its wide range of topics. This is intentional. There are some topics that are Ubuntu-specific and are not covered by any other book, and so deserve deep coverage here. There are some topics that every power user really must master. There are other topics that power users should know about, so that they understand some history, know some other options, or simply have what they need to be able to listen and participate in further discussions with other technical people without being completely confused.

Some topics, like using the Linux command line, receive deep and extensive coverage because I believe that information to be vital to anyone who wants to be a power user or become a skilled DevOps guru. That topic gets two full chapters.

Other topics, like the chapter that mentions ADA and Fortran, along with more than 15 other programming languages, only get brief coverage so that people who are interested get a few guideposts to help them continue if they are interested. In this case around 20 programming languages are covered in about a dozen pages. These are useful topics to some, but not topics I would consider vital.

Additionally, some topics are just too broad to be covered in great depth in this book, but are topics that deserve a mention because, again, an intermediate to advanced user should have at least a foundational knowledge of them. These are covered and then information is provided to help you find more resources and expand your understanding, as needed.

## Those Wanting to Become Intermediate or Advanced Users

*Ubuntu Unleashed* is intended for intermediate and advanced users or those who want to become one. Our goal is to give you a nudge in the right direction, to help you enter the higher stages by exposing you to as many different tools and ideas as possible; we want to give you some thoughts and methods to consider and spur you on to seek out more. Although the contents are aimed at intermediate to advanced users, new users who pay attention will benefit from the advice, tips, tricks, traps, and techniques presented in each

chapter. Pointers to more detailed or related information are also provided at the end of each chapter.

If you are new to Linux, you might need to learn some new computer skills, such as how to research your computer's hardware, how to partition a hard drive, and (occasionally) how to use a command line. This book helps you learn these skills and shows you how to learn more about your computer, Linux, and the software included with Ubuntu. Most important, it helps you overcome your fear of the system by telling you more about what it is and how it works.

We would like to take a moment to introduce a concept called “The Three Levels of Listening” from Alistair Cockburn’s *Agile Software Development*, published by Addison Wesley. These describe how a person learns and masters a technique. We all start at the first stage and progress from there. Few reach the last stage, but those who do are incredibly effective and efficient. People aiming for this stage are the very ones for whom we intend this book.

- ▶ **Following**—The stage where the learner looks for one very detailed process that works and sticks to it to accomplish a task.
- ▶ **Detaching**—The stage where the learner feels comfortable with one method and begins to learn other ways to accomplish the same task.
- ▶ **Fluent**—The stage where the learner has experience with or understanding of many methods and doesn’t think of any of them in particular while doing a task.

Myriad books focus on the first set of users. This is not one of them. It is our goal in *Ubuntu Unleashed* to write just enough to be sufficient to get you from where you are to where you want or need to be. This is not a book for newcomers who want or need every step outlined in detail, although we do that occasionally. This is a book for people who want help learning about what can be done and a way to get started doing it. The Internet is an amazing reference tool, so this is not a comprehensive reference book. This book is a tool to help you see the landscape; to learn enough about what you seek to get you started in the right direction with a quality foundational understanding.

## Sysadmins, Programmers, and DevOps

Systems administrators, or Sysadmins, are the people who keep servers and networks up and running. Their role is sometimes called *operations*. They deal with software installation and configuration, security, and do all the amazing things behind the scenes that let others use these systems for their work. They are often given less respect than they deserve, but the pay is good and it is a ton of fun to wield the ultimate power over a computer system. It is also a great responsibility, and these amazing guys and gals work hard to make sure they do their jobs well, striving for incredible system uptime and availability. Ubuntu is an excellent operating system for servers and networks, and in this book you can find much of the knowledge needed to get started in this role.

Programmers are the people who write software. They are sometimes called *developers*. Programmers work with others to create the applications that run on top of those systems.

Ubuntu is a great platform for writing and testing software. This is true whether you are doing web application development or writing software for desktop or server systems. It also makes a great platform for learning new programming languages and trying out new ideas. This book can help you get started.

DevOps is a portmanteau of *developer* and *operations*. It signifies a blending of the two roles already described. The information technology (IT) world is changing, and roles are becoming less clear cut and isolated from one another. In the past, it was common to witness battles between programmers excited about new technology and sysadmins in love with stability. DevOps realizes that neither goal is healthy in isolation, but that seeking a balance between the two can yield great results by removing the barriers to communication and understanding that sometimes cause conflict within a team. Because of the rise of cloud computing and virtualization, which are also covered in this book, and more agile forms of development, DevOps is a useful perspective that enables people working in IT to do an even better job of serving their ultimate clients: end users. This book is a great foundation for those wanting to learn knowledge that will help with both roles, hopefully presented in a way that balances them nicely.

## What This Book Contains

*Ubuntu Unleashed* is organized into six parts, described here. A disc containing the entire distribution is included so that you have everything you need to get started.

Part I, “Getting Started,” takes you through installing Ubuntu on your computer in the place of any other operating system you might be running, such as Windows.

Part II, “Desktop Ubuntu,” is aimed at users who want to use Ubuntu on desktop systems.

Part III, “System Administration,” covers both elementary and sophisticated details of setting up a system for specific tasks and maintaining that system.

Part IV, “Ubuntu as a Server,” gives you the information you need to start building your own file, web, and other servers for use in your home or office.

Part V, “Programming Linux,” provides a great introduction to how you can extend Ubuntu capabilities even further using the development tools supplied with it.

In addition to what has already been mentioned, after the spring release of Ubuntu, a bonus chapter will be available online at [www.informit.com/title/9780134268118](http://www.informit.com/title/9780134268118).

If you have the print copy of this book, follow the instructions on the inside back cover page to register your product and you will receive an email notification when the bonus chapter is available.

## Conventions Used in This Book

It is impossible to cover every option of every command included in Ubuntu. Besides, with the rise of the Internet and high-speed connections, reference materials are far less valuable than they used to be because most of these details are only a quick Google search away. Instead, we focus on teaching you how to find information you need while giving a

quality overview worthy of the intermediate or advanced user. Sometimes this book offers tables of various options, commands, and keystrokes to help condense, organize, and present information about a variety of subjects.

To help you better understand code listing examples and sample command lines, several formatting techniques are used to show input and ownership. For example, if the command or code listing example shows typed input, the input is formatted in boldface after the sample command prompt, as follows:

```
matthew@seymour:~$ ls
```

If typed input is required, as in response to a prompt, the sample typed input also is in boldface, like so:

```
Delete files? [Y/n] y
```

All statements, variables, and text that should appear on your display use the same boldface formatting. In addition, command lines that require root or super-user access are prefaced with the `sudo` command, as follows:

```
matthew@seymour:~$ sudo printtool &
```

The following elements provide you with useful tidbits of information that relate to the discussion of the text:

#### NOTE

A note provides additional information you might find useful as you are working. Notes augment a discussion with ancillary details or point you to an article, a whitepaper, or another online reference for more information about a specific topic.

---

#### TIP

A tip contains a special insight or a timesaving technique, as well as information about items of particular interest to you that you might not find elsewhere.

---

#### CAUTION

A caution warns you about pitfalls or problems before you run a command, edit a configuration file, or choose a setting when administering your system.

---

**SIDEBARS CAN BE GOLDMINES**

Just because it is in a sidebar does not mean that you will not find something new here. Be sure to watch for these elements that bring in outside content that is an aside to the discussion in the text. You will read about other technologies, Linux-based hardware, and special procedures to make your system more robust and efficient.

---

Other formatting techniques include the use of italic for placeholders in computer command syntax. Computer terms or concepts are also italicized upon first introduction in text.

Finally, you should know that all text, sample code, and screenshots in *Ubuntu Unleashed* were developed using Ubuntu and open-source tools.

Read on to start learning about and using the latest version of Ubuntu.

*This page intentionally left blank*

# CHAPTER 9

## Managing Software

In this chapter, we look at the options you have to manage your software in Ubuntu. If you are used to an environment where you are reliant on visiting different vendor websites to download updates, you are in for a pleasant surprise. Updating a full Ubuntu installation, including all the application software, is as simple as running the Update Manager program. You will discover just how easy it is to install and even remove various software packages.

Ubuntu provides a variety of tools for system resource management. The following sections introduce the graphical software management tools that you will use for most of your software management. This chapter also covers monitoring and managing memory and disk storage on your system.

### Ubuntu Software Center

The Ubuntu Software Center is a graphical utility for package management in Ubuntu. You can find it in the Applications menu as Ubuntu Software Center; the package and executable program is named `software-center`. The Ubuntu Software Center enables you to easily select and install a large array of applications by using the intuitive built-in search and easy one-click installation. When you open the program, you see the Package Browsing screen, as shown in Figure 9.1.

Along the left side of the screen, you have three menu options: Get Software, Installed Software, and History. At the top is a search bar that you can use to search for packages. When you click the Get Software link, you are

### IN THIS CHAPTER

- ▶ Ubuntu Software Center
- ▶ Using Synaptic for Software Management
- ▶ Staying Up-to-Date
- ▶ Working on the Command Line
- ▶ Compiling Software from Source
- ▶ Configuration Management
- ▶ Snappy Ubuntu Core
- ▶ References



presented with options to explore software Provided by Ubuntu or software For Purchase. Clicking the Installed Software link presents you with a list of all the installed applications on your Ubuntu desktop.

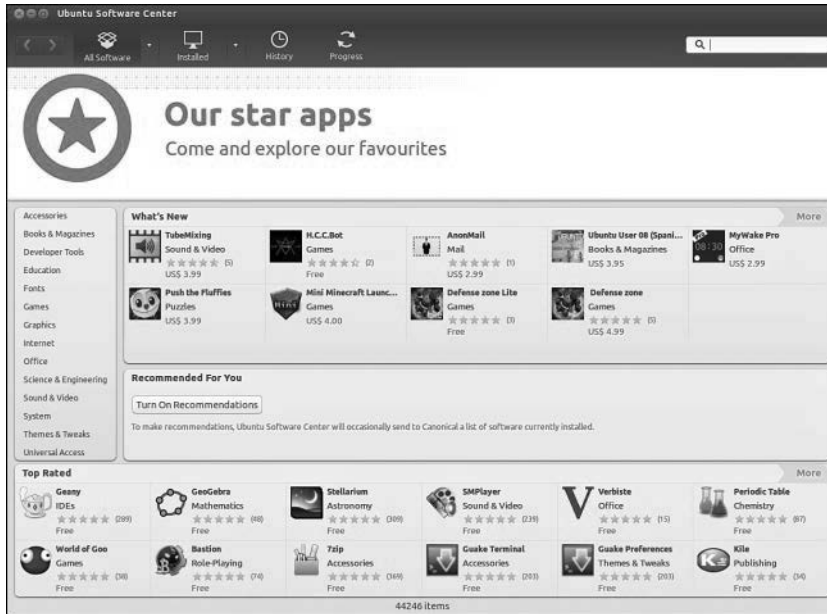


FIGURE 9.1 The initial Ubuntu Software Center screen enables you to browse through packages sorted by groups.

Installing new software via Ubuntu Software Center is as simple as finding it in the package list, double-clicking, and clicking the Install button. When you do so, you may be asked for your password; then the application is downloaded and installed. You can remove an application by finding it in Ubuntu Software Center and clicking the Remove button.

Use the Search box at the top to search for a specific application in the list. Note that this searches within the current category; so if you are in the Games category and search for “office,” you will get no results. The best place to search is within the Get Free Software category, to make sure you search all areas.

## Using Synaptic for Software Management

The Add/Remove Applications dialog works just fine for adding applications, but if you need to install something specific—such as a library—or if you want to reconfigure your installation system, you need to use Synaptic (Figure 9.2). You can install Synaptic using the Ubuntu Software Center described earlier; it is not installed by default.

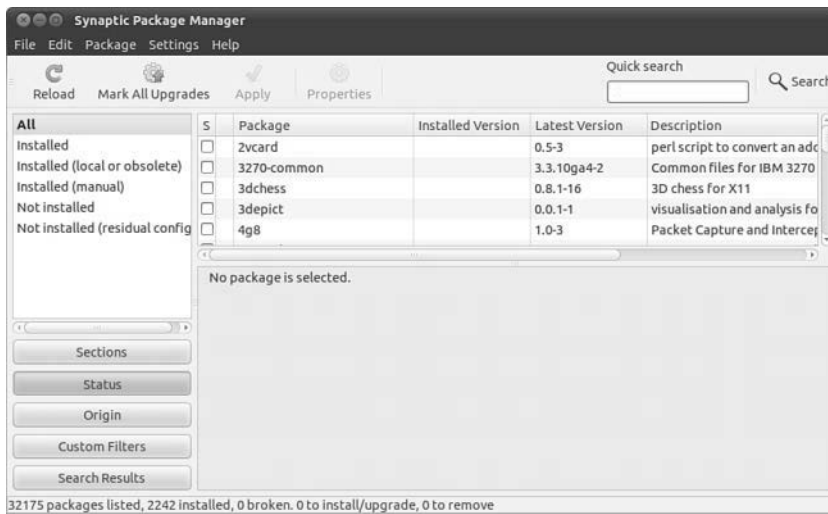


FIGURE 9.2 For more advanced software management in a GUI, Synaptic is the preferred tool.

At first glance, Synaptic looks a little like the Add/Remove Applications window. Along the left are software categories (although this time there are more of them), along the top right are the package selections for that category, and on the bottom right is the Package Information window that shows information about the currently selected package. To install or remove software, click the check box to the left of its name, and you'll see a menu that offers the following options:

- ▶ **Unmark**—If you have marked this package for installation, upgrade, or one of the other options, this option removes that mark.
- ▶ **Mark for Installation**—Add this package to the list that will be installed.
- ▶ **Mark for Re-installation**—If you have some software already installed, but for some reason it's not working, this option reinstalls it from scratch.
- ▶ **Mark for Upgrade**—If the software has updates available, this option downloads and installs them.
- ▶ **Mark for Removal**—This option deletes the selected package from your system but leaves its configuration files intact so that if you ever reinstall it you do not have to reconfigure it.
- ▶ **Mark for Complete Removal**—This option deletes the selected package from your system but also removes any configuration files, purging everything from the system.

After you have made your changes, click the Apply button to have Synaptic download, install, upgrade, and uninstall as necessary. If you close the program without clicking Apply, your changes are lost.

Beneath the categories on the left side of the screen, you see four buttons: Sections, Status, Search, and Custom, with Sections selected. These customize the left list: Sections is the Categories view; Status enables you to view packages that are installed or upgradable; Search stores results of your searches; and Custom has some esoteric groupings that are useful only to advanced users.

You can press Ctrl+F at any time to search for a particular package. By default, it is set to search by package name. You may change the Look In box setting to Description and Name. As mentioned already, your search terms are saved under the Search view (the button on the bottom left), and you can click from that list to re-search on that term.

As well as providing the method of installing and removing software, Synaptic provides the means to configure the servers you want to use for finding packages. In fact, this is where you can make one of the most important changes to your Ubuntu system: You can open it up to the Ubuntu Universe and Multiverse.

Ubuntu is based on the Debian distribution, which has thousands software packages available for installation. Ubuntu uses only a subset of that number but makes it easy for you to install the others, along with many packages that are not available in Debian. When you use Synaptic, you see small orange Ubuntu logos next to many packages; this identifies them as being officially supported by the Canonical-supported Ubuntu developers. The packages that do not have this logo are supported by the wider Ubuntu community of developers.

To enable the Universe and Multiverse repositories, go to Settings, Repositories. This list shows all the servers you have configured for software installation and updates and includes the Universe and Multiverse repositories. When you find them, check them, and then click Close.

Synaptic shows a message box warning you that the repository listings have changed and that you need to click the Reload button (near the top left of the Synaptic window) to have it refresh the package lists. Go ahead and do that, and you should see a lot more software appear for your selection. However, notice that only a small number have the official Ubuntu “seal” attached, which means you may want to be a bit more careful when installing software.

#### NOTE

Much of the software discussed in this book is available only through the Universe repository. Therefore, we highly recommend enabling it to get full use out of this book and your Ubuntu installation.

## Staying Up-to-Date

Although you can manage your software updates through Synaptic, Ubuntu provides a dedicated tool called Software Updater (shown in Figure 9.3). This tool is designed to be simple to use: When you run it, Software Updater automatically downloads the list of updates available and checks them all in the list it shows. If the update list was

downloaded automatically not too long ago, you can force Ubuntu to refresh the list of available updates by clicking the Check button. Otherwise, all you need to do is click Install Updates to bring your system up to date. If you want a little more information about the updates, click Show Details at the bottom to see what has changed in the update.

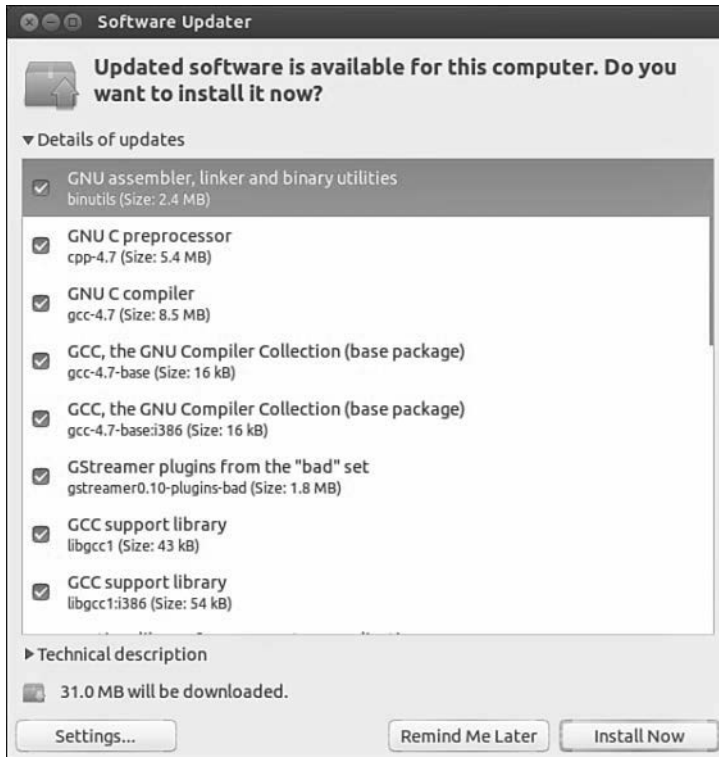


FIGURE 9.3 If you need to update your software to apply bug fixes and security upgrades, use Software Updater.

Ubuntu automatically checks for updates periodically and notifies you when critical updates are available. However, there's no harm running Software Updater yourself every so often, just to make sure; it's better to be safe than sorry.

## Working on the Command Line

With so much software available for installation, it is no surprise that Debian-based distros have many ways to manage software installation. At their root, however, they all use Debian's world-renowned *Advanced Package Tool (APT)*. A person posting on Slashdot.com once said, "Welcome to Slashdot. If you can't think of anything original, just say how

much APT rocks and you'll fit right in." You see, even though many other distros have tried to equal the power of APT, nothing else even comes close.

Why is APT so cool? Well, it was the first system to properly handle dependencies in software. Other distros, such as Red Hat, used RPM files that had dependencies. For example, an RPM for Gimp would have a dependency on Gtk, the graphical toolkit on which Gimp is based. As a result, if you tried to install your Gimp RPM without having the Gtk RPM, your install would fail. So, you grab the Gtk RPM and try again. Aha: Gtk has a dependency on three other things that you need to download. And those three other things have dependencies on 20 other things. And so on, and so on, usually until you can't find a working RPM for one of the dependencies, and you give up.

APT, on the other hand, was designed to automatically find and download dependencies for your packages. So, if you want to install Gimp, it downloads Gimp's package and any other software it needs to work. No more hunting around by hand, no more worrying about finding the right version, and certainly no more need to compile things by hand. APT also handles installation resuming, which means that if you lose your Internet connection part-way through an upgrade (or your battery runs out, or you have to quit, or whatever), APT picks up where it left off the next time you rerun it.

## Day-to-Day Usage

To enable you to search for packages both quickly and thoroughly, APT uses a local cache of the available packages. Try running this command:

```
matthew@seymour:~$ sudo apt-get update
```

The `apt-get update` command instructs APT to contact all the servers it is configured to use and download the latest list of file updates. If your lists are outdated, it takes a minute or two for APT to download the updates. Otherwise, this command executes it in a couple of seconds.

After the latest package information has been downloaded, you are returned to the command line. You can now ask APT to automatically download any software that has been updated, using this command:

```
matthew@seymour:~$ sudo apt-get upgrade
```

If you have a lot of software installed on your machine, there is a greater chance of things being updated. APT scans your software and compares it to the latest package information from the servers and produces a report something like this:

```
mmatthew@seymour:~$ sudo apt-get upgrade
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following packages will be upgraded:
  cabextract google-chrome-beta icedtea6-plugin language-pack-en
```

```

language-pack-en-base language-pack-gnome-en language-pack-gnome-en-base
libfreetype6 libfreetype6-dev libsmbclient libwbclient0 openjdk-6-jre
openjdk-6-jre-headless openjdk-6-jre-lib samba-common samba-common-bin
smbclient upstart winbind xserver-common xserver-xorg-core
21 upgraded, 0 newly installed, 0 to remove and 0 not upgraded.
Need to get 84.8MB of archives.
After this operation, 623kB of additional disk space will be used.
Do you want to continue [Y/n]?

```

Each part of that report tells you something important. Starting at the top, the line “the following packages will be upgraded” gives you the exact list of packages for which updates are available. If you’re installing new software or removing software, you see lists titled “The following packages will be installed” and “The following packages will be removed.” A summary at the end shows a total of 21 packages that APT will upgrade, with 0 new packages, 0 to remove, and 0 not upgraded. Because this is an upgrade rather than an installation of new software, all those new packages take up only 623KB of additional space. Although you have an 84.8MB download, the packages are overwriting existing files.

It’s important to understand that a basic `apt-get upgrade` never removes software or adds new software. As a result, it is safe to use to keep your system fully patched because it should never break things. However, occasionally you will see the “0 not upgraded” status change, which means some things cannot be upgraded. This happens when some software must be installed or removed to satisfy the dependencies of the updated package, which, as previously mentioned, `apt-get upgrade` will never do.

In this situation, you need to use `apt-get dist-upgrade`, so named because it’s designed to allow users to upgrade from one version of Debian/Ubuntu to a newer version—an upgrade that inevitably involves changing just about everything on the system, removing obsolete software, and installing the latest features. This is one of the most-loved features of Debian because it enables you to move from version to version without having to download and install new CDs. Keeping regular upgrades and distro upgrades separate is very useful for making sure that security updates and simple bug fixes don’t change software configurations that you may be counting on, especially on a machine that needs to be consistently available and working, such as a server.

Whereas `apt-get upgrade` and `apt-get dist-upgrade` are there for upgrading packages, `apt-get install` is responsible for adding new software. For example, if you want to install the MySQL database server, you run this:

```
matthew@seymour:~$ sudo apt-get install mysql-server
```

Internally, APT queries “mysql-server” against its list of software and find that it matches the `mysql-server-5.5` package. It then finds which dependencies it needs that you don’t already have installed and gives you a report like this one:

```

matthew@seymour:~$ sudo apt-get install mysql-server
[sudo] password for matt:

Reading package lists... Done
Building dependency tree
Reading state information... Done
The following extra packages will be installed:
  libaio1 libdbd-mysql-perl libdbi-perl libhtml-template-perl libmysqlclient18
  libnet-daemon-perl libplrpc-perl libterm-readkey-perl mysql-client-5.5
  mysql-client-core-5.5 mysql-server-5.5 mysql-server-core-5.5
Suggested packages:
  libipc-sharedcache-perl tinyca mailx
The following NEW packages will be installed:
  libaio1 libdbd-mysql-perl libdbi-perl libhtml-template-perl libmysqlclient18
  libnet-daemon-perl libplrpc-perl libterm-readkey-perl mysql-client-5.5
  mysql-client-core-5.5 mysql-server mysql-server-5.5 mysql-server-core-5.5
0 upgraded,13 newly installed, 0 to remove and 0 not upgraded.
Need to get 26.8 MB of archives.
After this operation, 96.2 MB of additional disk space will be used.
Do you want to continue [Y/n]?

```

This time, you can see that APT has picked up and selected all the dependencies required to install MySQL Server 5.5, but it has also listed one recommended package and two suggested packages that it has not selected for installation. The “recommended” package is just that: The person who made the MySQL package (or its dependencies) thinks it would be a smart idea for you to also have the mailx package. If you want to add it, press `n` to terminate `apt-get` and rerun it like this:

```

matthew@seymour:~$ sudo apt-get install mysql-server mailx

```

The “suggested” packages are merely a lower form of recommendation. They don’t add any crucial features to the software you selected for install, but it’s possible that you might need them for certain non-crucial (to the main piece of software being installed) features or tasks.

#### NOTE

APT maintains a package cache where it stores DEB files it has downloaded and installed. This usually lives in `/var/cache/apt/archives` and can sometimes take up many hundreds of megabytes on your computer. You can have APT clean out the package cache by running `apt-get clean`, which deletes all the cached DEB files. Alternatively, you can run `apt-get autoclean`, which deletes cached DEB files that are beyond a certain age, thereby keeping newer packages.

---

If you try running `apt-get install` with packages you already have installed, APT considers your command to be `apt-get update` and looks to see whether new versions are available for download.

The last day-to-day package operation is removing things you no longer want, which you do through the `apt-get remove` command, as follows:

```
matthew@seymour:~$ sudo apt-get remove firefox
```

Removing packages can be dangerous because APT also removes any software that relies on the packages you selected. For example, if you were to run `apt-get remove libgtk2.0-0` (the main graphical toolkit for Ubuntu), you would probably find that APT insists on removing more than a hundred other things. The moral of the story is this: When you remove software, read the APT report carefully before pressing `y` to continue with the uninstall.

A straight `apt-get remove` leaves behind the configuration files of your program so that if you ever reinstall it you do not also need to reconfigure it. If you want to remove the configuration files as well as the program files, run this command instead:

```
matthew@seymour:~$ sudo apt-get remove --purge firefox
```

That performs a full uninstall.

#### NOTE

You can see a more extensive list of `apt-get` parameters by running `apt-get` without any parameters. The cryptic line at the bottom, “This APT has Super Cow Powers,” is made even more cryptic if you run the command `apt-get moo`.

## Finding Software

With so many packages available, it can be hard to find the exact thing you need using command-line APT. The general search tool is called `apt-cache` and is used like this:

```
matthew@seymour:~$ apt-cache search kde
```

Depending on which repositories you have enabled, that tool returns about a thousand packages. Many of those results will not even have KDE in the package name but will be matched because the description contains the word *KDE*.

You can filter through this information in several ways. First, you can instruct `apt-cache` to search only in the package names, not in their descriptions. You do this with the `-n` parameter, like this:

```
matthew@seymour:~$ apt-cache -n search kde
```

Now the search has gone down from more than 1,000 packages to a few hundred.

Another way to limit search results is to use some basic regular expressions, such as `^`, meaning “start,” and `$`, meaning “end.” For example, you might want to search for programs that are part of the main KDE suite and not libraries (usually named something like `libkde`), additional bits (such as `xmms-kde`), and things that are actually nothing to do



with KDE yet still match our search (like `tkdesk`). Do this by searching for packages that have a name starting with `kde`, as follows:

```
matthew@seymour:~$ apt-cache -n search ^kde
```

Perhaps the easiest way to find packages is to combine `apt-cache` with `grep`, to search within search results. For example, if you want to find all games-related packages for KDE, you could run this search:

```
matthew@seymour:~$ apt-cache search games | grep kde
```

When you've found the package you want to install, run it through `apt-get install` as usual. If you first want a little more information about that package, you can use `apt-cache showpkg`, like this:

```
matthew@seymour:~$ apt-cache showpkg mysql-server-5.0
```

This shows information on “reverse depends” (which packages require, recommend, or suggest `mysql-server-5.0`), “dependencies” (which packages are required, recommended, or suggested to install `mysql-server-5.0`), and “provides” (which functions this package gives you). The “provides” list is quite powerful because it allows different packages to provide a given resource. For example, a MySQL database-based program requires MySQL to be installed, but isn't fussy whether you install MySQL 4.1 or MySQL 5.5. In this situation, the Debian packages for MySQL 4.1 and MySQL 5.0 both have “`mysql-server-4.1`” in the provides list, meaning that they offer the functionality provided by MySQL 4.1. Therefore, you can install either version to satisfy the MySQL-based application.

## Compiling Software from Source

Compiling applications from source is not that difficult. There are two ways to do this: You can use the source code available in the Ubuntu repositories, or you can use source code provided by upstream developers (most useful for those projects that are not available in the Ubuntu repositories). For either method, you need to install the package `build-essential` to ensure that you have the tools you need for compilation. You may also need to install `automake` and `checkinstall`, which are build tools.

### Compiling from a Tarball

Most source code that is not in the Ubuntu repositories is available from the original writer or from a company's website as compressed source *tarballs*—that is, `tar` files that have been compressed using `gzip` or `bzip`. The compressed files typically uncompress into a directory containing several files. It is always a good idea to compile source code as a regular user to limit any damage that broken or malicious code might inflict, so create a directory named `source` in your home directory.

From wherever you downloaded the source tarball, uncompress it into the `~/source` directory using the `-C` option to `tar`:

```
matthew@seymour:~$ tar zxvf packagename.tgz -C ~/source
matthew@seymour:~$ tar zxvf packagename.tar.gz -C ~/source
matthew@seymour:~$ tar jxvf packagename.bz -C ~/source
matthew@seymour:~$ tar jxvf packagename.tar.bz2 -C ~/source
```

If you are not certain what file compression method was used, use the `file` command to figure it out:

```
matthew@seymour:~$ file packagename
```

Now, change directories to `~/source/packagename` and look for a file named `README`, `INSTALL`, or a similar name. Print out the file if necessary because it contains specific instructions on how to compile and install the software. Typically, the procedure to compile source code is as follows:

```
matthew@seymour:~/source/packagename$ ./configure
```

This runs a script to check whether all dependencies are met and the build environment is correct. If you are missing dependencies, the `configure` script normally tells you exactly which ones it needs. If you have the Universe and Multiverse repositories enabled in Synaptic, chances are you will find the missing software (usually libraries) in there.

When your `configure` script succeeds, run the following to compile the software:

```
matthew@seymour:~/source/packagename$ make
```

And finally, run the following:

```
matthew@seymour:~/source/packagename$ sudo make install
```

If the compile fails, check the error messages for the reason and run the following before you start again:

```
matthew@seymour:~/source/packagename$ make clean
```

You can also run the following to remove the software if you do not like it:

```
matthew@seymour:~/source/packagename$ sudo make uninstall
```

## Compiling from Source from the Ubuntu Repositories

You might sometimes want to recompile a package, even though a binary package is available in the Ubuntu repositories. For example, a program might have been compiled into a binary with a specific feature disabled that you would like to use. Here is how you can do this. We will call the software package we want to compile *foo*.

First, get the source from the Ubuntu repositories:

```
matthew@seymour:~$ apt-get source foo
```

Install the build dependencies for the package:

```
matthew@seymour:~$ sudo apt-get build-dep foo
```

Change to the directory for the source code (may include the version number):

```
matthew@seymour:~$ cd foo-4.5.2
```

Make whatever changes you want to make to the package or to the compilation flags. You can do this using `./configure` and `make`, or sometimes by making manual changes to a configuration file. Each package has the potential to do this differently, so you need to see that program's documentation. Try looking for a README file in the source code to get started.

Next, create a new `debian/changelog` entry. After you enter this command, you need to enter a message that tells why a new version was made, perhaps something like *Matthew's flight of fancy with extra sauce*.

#### NOTE

Ubuntu package numbering follows a specific pattern. To help yourself later, you should stick to this pattern. Using the foo numbers shown here, a typical Ubuntu package that was inherited from Debian with no changes would then be 4.5.2-1. A package inherited from Debian, but changed for Ubuntu would be 4.5.2-1ubuntu1 (and then ubuntu2 for a second version, and so on). A package that did not have a version in Debian but which was created for Ubuntu would be 4.5.2-0ubuntu1 (and ubuntu2 and so on).

```
matthew@seymour:~$ dch -i
```

Build the source package. This creates all the files necessary for uploading a package:

```
matthew@seymour:~$ debuild -S
```

Finally, you are left with a `foo-4.5.2-1ubuntulcustom.deb` package (using whatever version number or suffix you created earlier) that you can install, and later uninstall as well, using your package manager. In some instances, multiple DEB files might be created, in which case you would replace the individual package name in the example here with `*.deb`.

```
matthew@seymour:~$ sudo dpkg -Oi foo-4.5.2-1ubuntulcustom.deb
```

## Configuration Management

This section provides a quick introduction to a couple tools that might be useful for those who want more control over system configuration management. For larger needs, see Chapter 35, “Managing Sets of Servers.”

## dotdee

If you run Linux-based systems, you will find a series of directories that end with a `.d` and that store configuration files. These are sometimes called `.d` or “dot dee” directories. If you look in `/etc/`, you find many (such as `apparmor.d` and `pam.d`). Opening these directories reveals a large number of configuration files and perhaps other directories containing even more. In Ubuntu or other Debian-based systems, it is a violation of etiquette (and Debian policy) for any software package to be allowed to directly change the configuration files of another package. This can be problematic if you want to use system configuration management software.

`dotdee` solves this problem by allowing you to take any flat file in your filesystem and replace it with a symlink pointing to a file that is generated from a `.d`-style directory. It saves the original file and then updates the generated file automatically and dynamically any time any file in the original `.d` directory is added, deleted, or modified. This way, the Debian policy and general etiquette standards are met, but configurations can be modified as needed by an external program.

`dotdee` works its magic using `inotify` to dynamically and instantly update the master file. The master file can be built three different ways: using flat files, which are concatenated; using diff/patch files, which are applied in a quiltlike manner; and using executables, which process `stdin` and dump to `stdout`. This flexibility should make any system administrator or developer guru happy.

## OneConf

OneConf is a mechanism for recording software information in Ubuntu One and synchronizing with other computers as needed. Open the Ubuntu Software Center and select File, Sync Between Computers from the menu. You’re prompted to create an Ubuntu Software Center account, if you have not already done so. Then, on any other Ubuntu computer you use, you can log in to the same account, and all the same applications will be installed, along with your copied and saved application data, to the new computer. No one else can see what you have installed or how it is configured.

## Snappy Ubuntu Core

Snappy Ubuntu Core takes the absolute minimum of files and code necessary for a usable Ubuntu server image and adds to it a new means of managing software packages. The idea is similar to how smart phones like Android-based phones provide software. In this method, the software packages include everything they need to run on the operating system, effectively making it so that a package is isolated from the operating system more completely. This is designed to protect from the possibility of a package breaking other packages or an entire operating system installation. It is also intended to make updates easier and cleaner. With the idea of convergence, where Ubuntu is aiming to use the same set of software for traditional desktops, laptops, tablets, and phones, all these devices will share the core operating system and Unity interface and packages that work on any one should also work on the others. This is intended to roll out with Ubuntu 16.04 LTS, if plans and development continue as anticipated.

While beta testing in virtual KVM containers has been available since Ubuntu 15.04, at the time of this writing it is not expected that Snappy Ubuntu Core will be available in Ubuntu 15.10, except for further testing opportunities. However, the hope and plan is to introduce it in Ubuntu 16.04 LTS. Learn more now and keep up to date as Ubuntu Snappy Core continues develop by checking out <https://developer.ubuntu.com/en/snappy/>.

## References

- ▶ [www.debian.org/doc/manuals/project-history/ch-detailed.en.html](http://www.debian.org/doc/manuals/project-history/ch-detailed.en.html)—History of the Debian Linux package system.
- ▶ [www.nongnu.org/synaptic/](http://www.nongnu.org/synaptic/)—Home of the Synaptic package manager.
- ▶ [www.ubuntu.com/usn](http://www.ubuntu.com/usn)—The official list of Ubuntu security notices.

*This page intentionally left blank*

# Index

## Symbols and Numbers

- < (redirection), 202
- > (redirection), 202
- ~ (tilde), 148
- / directory, 153
- 10BASE-T NICs, 377
- 32-bit installation, 4-5
- 64-bit installation, 4-5
- 100BASE-T NICs, 378
- 1000BASE-T NICs, 378

## A

- AbiWord (GNOME), 80
- ac command, 242, 254
- accounts. *See also* user accounts
- ACID compliance, 589-590
- address-based virtual hosting, 509-510
- addresses
  - broadcast, 376-377
  - IPv6, 372-375
  - multicast, 376
  - unicast, 376
- Adobe Flash, 106
- afio, 353
- all-in-one devices, 483
- allow directive, 500-501
- Alpine, 67
- ALSA (Advanced Linux Sound Architecture), 86
- Amanda, 352-353
- AMD drivers, 120
- AMD-V, 635

## Android, 729

- Android Studio, installation, 731
- Android Studio Project, 733
- Application Framework, 731
- applications, 731
- hardware, 730
- Java and, 730
- kernel, 730
- libraries, 730
- runtime, 730
- SDK packages, 731-733

## Ansible, 659

## Apache, 485

- access
  - allow directive, 500-501
  - authentication, 501-503
  - deny directive, 500-501
  - satisfy directive, 503-504
- apache2
  - DirectoryIndex, 496-497
  - DocumentRoot, 496
  - Group, 495
  - Listen, 495
  - ServerAdmin, 496
  - ServerName, 496
  - ServerRoot, 494
  - User, 495
  - UserDir, 496
- authentication, 501-503
- AuthUserFile directive, 502
- ./configure file and, 488-489
- file locations, 489
- file system, 499-500
- .htaccess files, 497-498
  - AllowOverrides directive, 499
  - Options directive, 498-499
- HTTPS, 513-515

## installation

- building source, 488-490
- from Ubuntu repositories, 486-488

## logging, 511-513

## modules, 504

- disabling, 504
- enabling, 504
- mod\_access, 504
- mod\_alias, 505
- mod\_asis, 505
- mod\_auth\_anon, 506
- mod\_auth\_dbm, 506
- mod\_auth\_digest, 506
- mod\_autoindex, 506
- mod\_cgi, 506
- mod\_dir, 506
- mod\_env, 506
- mod\_expires, 506
- mod\_headers, 507
- mod\_include, 507
- mod\_info, 507
- mod\_log\_config, 507
- mod\_mime, 507
- mod\_mime\_magic, 507
- mod\_negotiation, 507
- mod\_proxy, 507
- mod\_rewrite, 508
- mod\_setenvif, 508
- mod\_speling, 508
- mod\_ssl, 508
- mod\_status, 508
- mod\_unique\_id, 508
- mod\_userdir, 508
- mod\_usertrack, 509
- mod\_vhost\_alias, 509

## MPMs (multiprocessing modules), 497

## Nginx and, 519

## overview, 485-486



- performance tuning and, 441-442
- require directive, 503
- runtime configuration, 493-494
- starting
  - /etc/init.d/apache2, 492-493
  - manually, 490-492
- Tomcat, 532
- version, 486
- virtual hosting
  - address-based, 509-510
  - name-based, 510-511
- apache2, 486-487
  - DirectoryIndex, 496-497
  - DocumentRoot, 496
  - Group, 495
  - Listen, 495
  - ServerAdmin, 496
  - ServerName, 496
  - ServerRoot, 494
  - User, 495
  - UserDir, 496
- AppArmor, 430-432
- appending text files, 203
- applets, Network Manager, 24
- Application Framework, 731
- applications
  - Android, 731
  - productivity (See productivity applications)
- apropos command, 152-153
- apt, 202-203
- APT (Advanced Package Tool), 136
  - apt-cache, 140
  - apt-get dist-upgrade command, 137
  - apt-get install, 138
  - apt-get update command, 136
  - dependencies, 140
  - package removal, 139
- apt-get utility, 15-16
- arithmetic operators, Perl, Web:7-8

- ark (KDE), 347-348
- ARPANET, 61
- arrays
  - Perl, Web:5
  - PHP, Web:46-47
- at tool, 257-258
- authentication, Apache, 501-503
- autoconf utility (C/C++), 675-676
- automation, scripts and, 276-278
- autoresponders, 558
- AVI video, 105
- awk command, 222-224, 581

## B

- Back In Time, 350-351
- background tasks, 207-208
  - moving to and from, 208-209
  - shell, 269
- backslash in scripts, 281-282
- backups, 335
  - Clonezilla, 7
  - copying files
    - cp, 355-356
    - Midnight Commander, 356
    - rsync, 356-358
    - tar, 354-355
  - hardware
    - CD-RW drives, 343
    - cloud storage, 344
    - DVD+RW/-RW drives, 343
    - NAS (network-attached storage), 343
    - networks, 343
    - removable storage media, 342-343
    - tape drives, 343-344
  - levels, 340
  - needs assessment, 337-338

- partitioning and, 5
- software
  - afio, 353
  - Amanda, 352-353
  - ark, 347-348
  - Back In Time, 350-351
  - dd, 344
  - Déjà Dup, 344, 348-350
  - File Roller, 347
  - flexbackup, 353
  - tar, 344
  - tar tool, 345-347
  - Unison, 352
- strategies, 335-336
  - full backup, 341
  - home users, 339
  - large enterprise, 339
  - mirroring data, 342
  - RAID arrays, 342
  - simple, 340-341
  - small enterprise, 339
  - small office, 339
- system rescue, 360-362
  - version control, configuration files, 358-360
- badblocks, 439
- Banshee, 91-93
- Base (LibreOffice), 77
- BaseX, 612-613
- bash shell, 264
  - expression comparison, 279-282
- batch processes, password changes, 247
- batch tool, 259-260
- Battle for Wesnoth, 124-125
- Bazaar, 358, 685-686
- BBSs (Bulletin Board Systems), Usenet news
  - groups and, 72
- Behlendorf, Brian, 485
- Berkeley DB, 608-609
- Berners-Lee, Tim, 61
- beta testing, 35
- bg command, 208-209
- BigTable, 613
- /bin directory, 153
  - commands, 154-155
- BIND (Berkeley Internet Name Domain), DNS
  - server setup, 667-669
- BIOS, performance tuning and, 436-437
- Blender, 97
- BMP files, 95
- Boolean operators, 214
- boot
  - dual-boot, 5
  - runlevels, default, 307-308
  - services
    - administrative tools and, 309
    - running, 303-304
  - Upstart, 311-312
    - systemd and, 312-313
  - to X session, 49
- /boot directory, 153
- boot loading
  - BIOS and, 305
  - kernel and, 306
- Boot Repair tool, 313
- bootloader
  - dual boot, 6
  - GRUB2, 5-6, 28
  - MBR (Master Boot Record), 6
- bootmail, 697
- Brasero, 99
- break statement, 300
- Breezy Badger, 32
- bridged networking, KVM and, 636
- bridges (networks), 381
- broadcast addresses, 376-377

- browsers
  - Firefox, 61-63
  - Google Chrome, 63-64
  - Google Chromium, 63-64
- Bug Squad, 711
- built-in variables, shell scripts, 273, 278-279
- bunzip2 command, 224
- burning
  - CDs
    - Brasero and, 99
    - from command line, 99-101
  - DVDs, from command line, 101-103
- business, Ubuntu and, 32-33
- Bynari, 560
- byobu command, 225-226
- bzip2 utility, 224, 456

## C

- CAG (Carrier Advisory Group), 736
- Calc (LibreOffice), 77
- cameras, handheld, 98
- Canonical Software, 31
- case statement, 298-300
- cat command, 171, 181, 183, 203
- C/C++, 671-673
  - autoconf utility, 675-676
  - debugging tools, 676-677
  - GNU C Compiler, 677-678
  - make command, 673
  - makefiles, 673-675
- CCSM (CompizConfig Settings Manager), 57
- cd command, 160-161, 181, 183-185
- CD-RW drives, backups, 343

- CDs
  - burning
    - Brasero and, 99
    - from command line, 99-101
    - playing, 89-90
    - ripping, 90, 99
  - CERN, 61
  - CFEngine, 658-659
  - change command, 254
  - changetrack, 359
  - Chef, 658
  - Cherokee, 531
  - chfn command, 254
  - chgrp command, 165-159, 234, 254
  - Childsplay, 127
  - chmod command, 181, 185, 234, 254
  - chown command, 165, 234, 255
  - chpasswd command, 247, 255
  - Chrome, 63-64
  - Chromium, 63-64
  - chsh command, 240, 255
  - CIDR (classless interdomain routing) notation, 569
  - CinePaint, 97
  - Claws, 67
  - CLI (command-line interface), 145
  - Clojure, 717
  - Clonezilla, 7
  - cloud computing, 643
    - deploying Ubuntu, 646-647
    - DevOps, 644
    - DevStack, 647-648
    - hybrid cloud, 647
    - IaaS (Infrastructure-as-a-Service), 645
    - Juju, 649-650
      - charms, 652-654
      - GUI, 654-655
      - installation, 650-651
      - Mac OS X, 655

- Mojo, 655
- Quickstart, 654
  - service unit deployment, 651-652
  - Wordpress blog sample, 651-652
- Landscape, 656
- MaaS (Metal-as-a-Service), 645-646, 655-656
- OpenStack, 647-648
  - Glance, 649
  - Horizon, 649
  - Keystone, 649
  - Neutron, 648
  - Nova, 648
  - Swift, 648
- PaaS (Platform-as-a-Service), 645
  - preparation, 646
  - private cloud, 646
  - public cloud, 646
  - reasons for, 644-645
- SaaS (Software-as-a-Service), 645
- Snappy Ubuntu, 655
  - storage, backups, 344
- COBOL (Common Business Oriented Language), 717-718
- code names for Ubuntu, 32
- CodeWeavers, 83
- comm command, 205
- commads, less, 190-192
- command execution, 182
- command line
  - appending text files, 203
  - apropos, 152-153
  - APT and, 136
  - awk command, 222-224
  - Back In Time tool, 350-351
  - CDs, burning, 99-101
  - combining commands, 212-215
  - compressed files, 224-225
  - console text editors, 178
  - Coreutils, 229-230
  - DVDs, burning, 101-103
  - environment variables, 215-218
  - file management, 178
  - file system hierarchy, 153-158
  - gnome-terminal, 147
  - group management, 178, 237
    - gpasswd, 237
    - groupdel, 237
    - groupmod, 237
    - grpck, 237
    - useradd, 237
    - usermod, 237
  - history, 228-229
  - information, 178
  - input, redirecting, 202-203
  - Internet commands, 178
  - key combination, 147
  - log out, 149
    - remote computer, 149-150
  - login, text-based, 148-149
  - man pages, 152
  - multiple tasks, 229
  - network commands, 178
  - output, redirecting, 202
  - overview, 146-147
  - piping commands, 212-214
  - previous command, 227-228
  - processes, 316
  - redirection, streams, 203-204
  - resource usage
    - background tasks and, 207-209
    - listing jobs, 207
    - listing processes, 206
  - running programs, 178
  - sed command, 222-224
  - shell, 265-266
  - shortcuts, 229

- ssh command, 148
- super user, 150-151
- system reset, 226-227
- telnet command, 148
- Terminal and, 147
- terminals, multiple, 225-226
- text editors
  - emacs, 218, 221-222
  - nano, 218, 219-220
  - vi, 220-221
  - vim, 218
- tty1 virtual console, 147
- user management, 178
  - activity monitoring, 242
  - new users, 240-242
- whereis, 153
- commands
  - ac, 242
  - bg, 208-209
  - /bin directory, 154-155
  - Boolean operators and, 214
  - byobu, 225-226
  - cat, 171, 181, 183
  - cd, 160-161, 181, 183-185
  - chgrp, 165-159, 234
  - chmod, 181, 185, 234
  - chown, 165, 234
  - chpasswd, 247
  - chsh, 240
  - combining, 212-215
  - comm, 205
  - command execution, 182
  - cp, 170, 181, 186, 336-337
  - date, 22
  - deluser, 176
  - df, 320-321
  - diff, 204-205
  - directories
    - changing, 181
    - creating, 182
    - deleting, 182
  - dmesg, 25
  - du, 181, 186-187
  - echo, 201
  - emacs, 181
  - env, 215-216
  - fg, 208-209
  - files
    - copying, 181
    - deleting, 182
    - finding, 181
    - finding from index, 182
    - links, 181
    - listing, 182
    - moving, 182
    - print last lines, 182
    - sort order, 182
    - text, editing, 181, 182
  - filters, output and, 181
  - find, 181, 187-189
  - free, 319-320
  - grep, 181, 189-190
  - growisofs, 103
  - htpasswd, 501-502
  - hwclock, 23
  - ifconfig, 364, 369
  - jobs, 207
  - kill, 227, 317-318
  - less, 171, 181
  - ln, 181, 192-194
  - locate, 182, 194
  - logrotate utility, 325
  - ls, 158-164, 182, 194-196
  - lspci, 104
  - make, 182

- man, 182, 196-197
- man page display, 182
- mkdir, 163, 167-168, 182, 197
- mkisofs, 100
- mv, 169-170, 182, 197-198
- nano, 182
- newusers, 247
- nice, 211-212
- passwd, 239-240
- permissions, changing, 181
- pipes, 212-214
- printenv, 215-216
- printing
  - command location, 182
  - disk usage, 181
  - last file lines, 182
- printing and, 484
- printing files, 181
- program compile/install, 182
- ps, 206, 212, 315
- pwd, 161
- reasons to use, 180-181
- regular expressions, 171
- rm, 169, 182, 198
- rmdir, 168
- /sbin directory, 154-155
- screen, 225
- separate, in sequence, 214-215
- shutdown, 12, 176-177
- sort, 182, 198-200
- ssh, 148, 182
- strings, input, 181
- su, 248-250
- sudo, 17, 150, 172-175, 250-252
- sysctl, 440
- tail, 182, 200-201
- telinit, 309-310
- telnet, 148
- time, 318
- top, 209-211, 319
- touch, 161-162, 167
- uname, 28
- user accounts, 254-255
- usermod, 240
- vim, 182
- which, 182, 202
- wildcards, 171
- xargs, 182
- commercial support, 36-37
- CommuniGate Pro, 559
- community testing teams, 709-710
- comparing expressions, 282
  - file operators, 285-286
  - logical operators, 286-287
  - numbers, 284-285
  - strings, 283-284
  - tcsh, 287-291
- comparing files, 204-205
- comparison operators, Perl, Web:6-7
- compilers
  - default-jdk, 722
  - erlang, 719
  - gcc, 720
  - ghc, 721
  - gnat-gps, 716
  - open-cobol, 718
  - scalac sourceFile, 726
- compiling
  - kernel, 458-461
    - troubleshooting, 465-466
  - recompiling, 454-455
  - software, from source, 140-142
- compound operators, Perl, Web:7
- compressed files, 224-225
- conditional statements
  - Perl, Web:9-10
  - Python, Web:33-35

- configuration
  - files
    - /etc directory, 155
    - version control, 358-360
  - firewalls, 426-430
  - Internet, 402-403
  - kernel, 461-464
  - post-installation problems, 25-26
  - printers, 21
  - wireless networks, 24-25
- configuration management
  - dotdee directories, 143
  - OneConf, 143
  - Snappy Ubuntu Core, 143
- ./configure
  - Apache and, 488-489
  - Nginx and, 520
- Conky, 327-332
- console text editors commands, 178
- console-based monitoring
  - disk space, df command, 320-321
  - kill command, 317-318
  - log files, 321-323
    - rotating, 323-325
  - memory display, 319-320
  - priority scheduling, 318-319
  - ps command, 315
- consoles
  - login, text-based, 148-149
  - logout, 149
  - remote computer, 149-150
  - virtual, tty1, 147
- copying files, 181, 186
  - cp command, 355-356
  - Midnight Commander, 356
  - rsync, 356-358
  - SSH, 412-413
  - tar, 354-355

- Coreutils, 229-230
- CouchDB, 611-612
- cp command, 170, 181, 186, 336-337
  - file copy, 355-356
- CPAN (Comprehensive Perl Archive Network), Web:15-16
- crackers, 420
- cron tool, 260-262
- Crossover Games, 128
- CrossOver Office, 83
- cross-platform development, 30
- CUPS (Common UNIX Printing System), 481-483
  - LANs, 480
- current directory, 161

## D

- D, 718
- daemons, 29
- Dapper Drake, 32
- darktable, 97
- Dart, 718-719
- Dash, 13
  - Smart Scopes and, 57
  - System Monitor, 326
- data locking, databases, 589
- data loss, reasons for, 336-337
- databases, 581. *See also* LDAP (Lightweight Directory Access Protocol); RDBMS
  - awk and, 581
  - clients, 597-598
    - graphical, 602-603
    - local GUI client access, 599-600
    - MySQL command-line, 601-602
    - PostgreSQL command-line, 602
    - SSH access, 598-599
    - web access, 600

- flat file, 581
- grep and, 581
- MySQL, creating, 592-594
- NoSQL, 581, 605-608
  - document stores, 610-613
  - key/value stores, 608-610
- perl and, 581
- PHP, Web:72-74
- PostgreSQL
  - creating, 595
  - user creation, 596
  - user deletion, 596-597
- date command, 22
- DBA (database administrator), responsibilities, 580
- dd, 344
- Debian, 28
  - packages, 18
  - Synaptic and, 134
  - Ubuntu and, 31
- debugging
  - C/C++
    - gdb tool, 677
    - gprof command, 677
    - splint command, 676-677
    - symbolic, 677
  - Harvest, 707
- default-jdk compiler, 722
- Déjà Dup, 344, 348-350
- deleting
  - directories, 182
  - files, 182
- deluser command, 176
- deny directive, 500-501
- desktop, 30
  - switching, 50
  - Unity, 29
- Desktop Couch, 691
- desktop environments, 110
  - GNOME3, 114-115
  - KDE desktop, 111-112
  - Lubuntu, 113-114
  - LXDE, 113-114
  - MATE, 115
  - Ubuntu GNOME, 115
  - Ubuntu Kylin, 116
  - Xfce, 112-113
  - Xubuntu, 112-113
- /dev directory, 153
- development, 699-700
  - alpha releases, 701
  - apps, 704
  - beta releases, 701
  - bug fixing, 704-705
  - bugs, 700
  - code name, 700
  - cycle, 700
  - debugging, Harvest, 707
  - GPG key, 701-702
  - MOTU (Masters of the Universe), 707
  - pbuilder, 702
  - release day, 701
  - scopes, 704
  - source packages, 705-707
  - SSH key, 702
  - system setup
    - configuration, 701-702
    - Launchpad, 702-704
    - package installation, 701-702
- Device section (xorg.conf), 45, 48
- DevOps, 643
- DevStack, 647-648
- df command, 320-321
- DHCP (Dynamic Host Configuration Protocol), 377, 393-394
  - activating at boot, 394-395
  - host configuration, 397-399



- IP addresses, 394
  - software installation
    - dhclient, 395-396
    - server, 396
- DHCP Handbook, 399
- Dia (LibreOffice), 77, 78
- dial-up access, 406-407
- diff command, 204-205
- digikam, 97
- directories, 153
  - changing, 160-161, 181, 183-185
  - creating, 167-168, 182, 197
  - current, 161
  - deleting, 168, 169, 182, 198
  - dotdee directories, 143
  - listing contents, 158-160, 194-196
  - permissions, 163-164
  - user, 155-156
- disaster recovery, 432-433
- disk space
  - df command, 320-321
  - partitioning and, 5
  - printing, 186-187
  - quotas, 253, 321
    - implementing, 253-254
    - manual configuration, 254
- distributions, 28
- dmesg command, 25
- DNS (Domain Name Services), 403, 661-662
  - BIND (Berkeley Internet Name Domain) and, 667-669
  - labels, 663
  - records, 664
    - A, 664
    - AAAA, 664
    - CNAME, 665
    - MX, 665
    - NS, 665-666
    - SOA, 666-667
    - TXT, 667
  - servers, 663-664
    - setup, 667-669
- DocBook, 81-82
- Document Foundation, 78
- document stores (NoSQL), 610-611
  - BaseX, 612-613
  - CouchDB, 611-612
  - MongoDB, 612
- documentation, 33-35
  - guides, 37-38
  - Linux Documentation Page, 37
  - man pages, 152
  - writing, 35
- documents
  - DocBook and, 81-82
  - Gedit and, 82
  - Kile, 83
  - LateX and, 82-83
  - LyX, 83
  - PDF, 81
  - Publican and, 82
  - Texmaker and, 83
- domain names, labels, 663
- dotdee directories, 143
- do...until loop, Perl, Web:13
- do...while loop, Perl, Web:13
- Draw (LibreOffice), 77
- drivers, gaming, proprietary, 120
- DSL (digital subscriber line), 403
  - configuration, 404
  - PPPoE (Point-to-Point Protocol over Ethernet), 404-405
    - manual configuration, 405-406
- du command, 181, 186-187
- dual-boot, 6
  - partitioning and, 5
- DVD, installation, 2-3, 6
  - Desktop, 2
  - Server, 2

DVD+RW/-RW drives, backups, 343  
 dvd+rw-tools package, 102  
 DVDs, burning, from command line, 101-103

## E

e2fsck utility, 439  
 echo command, 201  
 Edgy Eft, 32  
 Elixir, 719  
 emacs command, 181  
 emacs text editor, 218, 221-222  
 email, 543
 

- Alpine, 67
- Bynari, 560
- Claws, 67
- CommuniGate Pro, 559
- elm, 65
- Evolution, 66-67
- Fetchmail
  - configuration, 553-557
  - installation, 553
- Gmail, 67
- Horde, 560
- Hotmail, 67
- Kmail, 67
- Mail, 67
- MDAs (Mail Delivery Agent), 546-547
  - autoresponders, 558
  - Procmail, 557
  - Spamassassin, 557-558
  - Squirrelmail, 558
  - virus scanners, 558
- Microsoft Exchange Server, 559
- Mozilla Thunderbird, 65-66

MTAs (mail transfer agents), 543
 

- Exim, 545-546
- Postfix, 545, 548-553
- Qmail, 545-546
- Sendmail, 544-545

 MUA (Mail User Agent), 547-548  
 Open-Xchange, 560  
 Oracle Beehive, 560  
 Outlook client, 559  
 pine, 65  
 sending, 544  
 SMTP (Simple Mail Transfer Protocol), 543  
 Yahoo!, 66-67  
 embedded spaces in strings, variables  
 and, 280  
 Empathy, chatting and, 68-69  
 emulators, games, 120  
 encryption
 

- /home, 9
- WEP encryption, 25
- WPA encryption, 25

 enterprise, system monitoring, 333  
 env command, 215-216  
 environment variables
 

- displaying, 215-216
- LANG, 215
- PATH, 215
- PWD, 215
- SHELL, 215
- shell scripts, 273
- TERM, 215
- USER, 215

 environments, Unity, 29  
 erasing, hard disk, installation and, 8  
 Erlang, 719  
 /etc directory, 153
 

- configuration files, 155

- /etc/passwd file, 243-244
  - user accounts, 232
- /etc/sudoers file, 250-251
- Ethernet cards, 369
- Evince, 81
- Evolution (GNOME), 66-67, 80, 622
- exit statement, 300
- expressions, comparison, 282
  - file operators, 285-286
  - logical operators, 286-287
  - numbers, 284-285
  - strings, 283-284
  - tcsh, 287-291

## F

- Feisty Fawn, 32
- Fetchmail
  - configuration
    - .fetchmailrc file, 553
    - global options, 554
    - mail server options, 554-555
    - user accounts, 555-557
  - installation, 553
- fg command, 208-209
- fiber optic NICs, 378
  - cable, 380
- file management commands, 178
- file operators, comparing, 285-286
- file permissions, 234-235
- File Roller, 347
- file storage, /tmp directory, 158
- file system
  - Apache, 499-500
  - directories, 153
  - performance tuning and, 438
- File Systems tab (System Monitor), 326
- filenames, wildcards, 171
- files
  - comparing, 204-205
  - compressed, 224-225
  - contents
    - displaying all, 171
    - displaying less, 171
  - copying, 170, 181, 186
    - cp command, 355-356
    - Midnight Commander, 356
    - rsync, 356-358
    - SSH, 412-413
    - tar, 354-355
  - creating, 167
  - Dash, 13
  - data loss, 336-337
  - deleting, 169, 182, 198
  - finding, 181, 187-189
    - from index, 182, 194
  - links, 181, 192-194
  - listing, 158-160, 182, 194-196
  - log files, 321-323
    - rotating, 323-325
  - moving, 169-170, 182
  - permissions, 161
    - altering, 164-165
    - assigning, 162-163
    - changing, 165, 185
    - default, 161
    - groups, 165-159
    - set group ID, 165-167
    - set user ID, 165-167
  - print last lines, 182, 200-201
  - printing, 181
    - contents, 183
  - renaming, 169-170
  - sort order, 182, 198-200
  - text, editing, 181, 182
  - version control, 358-360

- Files section (xorg.conf), 45
- filters, output and, 181
- find command, 181, 187-189, 213
- Firefox, 61-63
  - plug-ins, 62-63
  - RSS reader, 67
- firewalls
  - configuration, 426-430
  - disabling, 426
  - enabling, 426
  - IP addresses, 428
  - logging, 427
  - ports, 427
  - UFW (Ubuntu's Uncomplicated Firewall), 312, 426
    - commands, 426, 429
    - GUFW, 428
- Flash, enabling, 63
- flat file databases, 581
- flavors of Ubuntu, 3
- flexbackup, 353
- FlightGear, 126
- FlockDB, 615
- FLV video, 105
- for loops, Perl, Web:11
- for statement, 291-293
- foreach loops, Perl, Web:11
- Forth, 720
- Fortran, 720
- FQDN (fully qualified domain name), 618
- free command, 319-320
- free memory display, 319-320
- Frets on Fire, 125
- Frozen Bubble, 123
- fstab file, 155
- FTP (File Transfer Protocol), 533
  - ftphosts, connection and, 541
  - server packages
    - Bsdftpd-ssl, 535
    - NcFTPD, 534
    - ProFTPD, 535
    - vsftpd, 534
    - wu-ftp, 535
  - servers, 533-534
    - anonymous, 534
    - authenticated, 534
    - standard, 534
  - software, installation, 535-536
  - user, 536-538
  - vsftpd
    - configuration, 538-540
    - default behaviors, 540
    - FTP user, 536
- ftp.kernel.org site, 456

## G

- gaming
  - Battle for Wesnoth, 124-125
  - Childsplay, 126
  - commercial games, 127
  - Crossover Games, 128
  - emulators, 120
  - FlightGear, 126
  - Frets on Fire, 125
  - Frozen Bubble, 123
  - gCompris, 126
  - kids, 127
  - Scorched 3D, 122-123
  - Speed Dreams, 126
  - Steam, 127-128
  - SuperTux, 123-124
  - TuxPaint, 126

- video drivers, proprietary, 120
- Warsow, 121-122
- Windows games, 128
- gcc compiler, 720
- gCompris, 127
- Gedit, 82
- gedit text editor, 219, 691
- gforth, 720
- ghc compiler, 721
- GIF files, 95
- Gigabit Ethernet, 378
- GIMP (GNU Image Manipulation Program), 79, 80, 93
  - GTK (GIMP Tool Kit), widget set, 79
  - installation, 93
- Git, 358, 687-688
- Glade, 679-680, 691
- Gmail, 67
- gnat-gps, 716
- GNOME
  - File Roller, 347
  - GStreamer, 88
- GNOME Office, 79
  - AbiWord, 80
  - Evolution, 80
  - GIMP, 80
  - Gnumeric, 80
  - GTK widget set, 79
- GNOME3, 114-115
- gnome-nettool, 332
- gnome-terminal, 147
- GNU C Compiler, 677-678
- GNU GPL, 27-28
- Gnumeric (GNOME), 80
- Go, 720-721
- Google, 36
- gpasswd command, 237
- GPL (General Public License), 27
- graph stores (NoSQL), 614
  - FlockDB, 615
  - HyperGraphDB, 615
  - Neo4j, 614
  - OrientDB, 614
- graphical monitoring tools, 325-326
- graphical servers, X Server, 29
- graphics, 95
  - Blender, 97
  - BMP files, 95
  - CinePaint, 97
  - conversion programs, 95-96
  - darktable, 97
  - digiKam, 97
  - GIF files, 95
  - Hugin, 97
  - ImageMagick, 96
  - Inkscape, 97
  - JPG files, 95
  - netpbm tools, 96
  - PCX files, 95
  - PNG files, 95
  - POV-Ray, 97
  - Radiance, 97
  - SVG files, 95
  - TIF files, 95
  - Xara Xtreme, 97
- grep command, 181, 189-190, 212, 581
- Groovy, 721
- group ID (GID), 234
- group management commands, 178
- groupdel command, 237
- groupmod command, 237
- groups, 235
  - command-line tools
    - gpasswd, 237
    - groupadd, 237
    - groupdel, 237
    - groupmod, 237

- grpck, 237
- useradd, 237
- usermod, 237
- listing, 235-236
- permissions, 165-159
- groups command, 255
- growisofs command, 103
- grpck command, 237
- GRUB2 (Grand Unified Boot Loader), 6, 28
  - restore and, 361
- GStreamer, 691
- GTK (GIMP Tool Kit), 691
  - widget set, 79
- guides, 37-38
- gunzip command, 224
- gzip command, 224

## H

- hackers, 420
- handheld cameras, 98
- hard disk
  - erasing, installation and, 8
  - performance tuning, 436
    - badblocks, 439
    - BIOS and, 436-437
    - e2fsck, 439
    - file access time disable, 439
    - file system, 438
    - kernel and, 436-437
    - tune2fs, 438-439
- hardware
  - Android, 730
  - backups
    - CD-RW drives, 343
    - cloud storage, 344
    - DVD+RW/-RW drives, 343
    - networks, 343
    - removable storage media, 342-343
    - tape drives, 343-344
  - legacy, Linux and, 30
  - network
    - initialization, 381-383
    - NICs, 377-379
    - specifications, 2
    - video, 104-105
  - Harvest, 707
  - hashes, Web:5
  - Haskell, 721
  - HBase, 613-614
  - hctosys option, 23
  - hdparm utility, 437-438
  - history, viewing, 228-229
  - Hoary Hedgehog, 32
  - /home directory, 153
    - encrypting, 9
    - user directories, 155-156
  - Horde, 560
  - Hotmail, 67
  - HOWTO documents, 34
  - HPC (High Performance Computing), 672
  - .htaccess files (Apache), 497-498
    - AllowOverrides directive, 499
    - Options directive, 498-499
  - HTML (Hypertext Markup Language), 61
    - PHP, Web:71-72
  - htop command, 319
  - htpasswd command, 501-502
  - HTTP (Hypertext Transfer Protocol), 61
    - servers
      - Apache Tomcat, 532
      - Cherokee, 531
      - Jetty, 531
      - lighttpd, 529-530
      - thttpd, 532
      - Yaws, 530

hubs (networks), 380-381  
Hugin, 97  
hwclock command, 23  
HyperGraphDB, 615

## I

laaS (Infrastructure-as-a-Service), 645  
IDEs (integrated development environment), 680-681  
if statement, 297-298  
ifconfig command, 364, 369  
if/else statements (Perl), Web:9-10  
IIS (Internet Information Services), 486  
ImageMagick, 96  
images. *See also* photographs  
    graphics formats, 95-96  
    scanners, 94-95  
    screen capture, 96-97  
information gathering commands, 178  
init command, runlevels, 306  
init scripts, initialization and, 308  
init systems, 304  
Inkscape, 97  
input redirection, 202  
InputDevice section (xorg.conf), 45, 46-47  
installation  
    DVD, 2, 3, 6  
    hard disk, erasing, 8  
    hardware, specs, 2  
    initial screen, 7  
    language, 8  
    Macintosh, 3  
    operating systems installed, 8  
    post-installation configuration problems, 25-26  
    Preparing to Install Ubuntu screen, 8  
    software  
        Synaptic, 133  
        Ubuntu Software Center, 132  
    USB drive, 4, 6  
instant messaging, 68-69  
Internet  
    commands, 178  
    configuration, 402-403  
    connection troubleshooting, 407-408  
    dial-up access, 406-407  
    DNS (Domain Name Services), 403  
    DSL (digital subscriber line), 403  
        configuration, 404  
        PPPoE (Point-to-Point Protocol over Ethernet), 404-406  
    Internet Connection Wizard, 404  
    IP addresses, 403  
    ISP (Internet service provider), 402-403  
    PPP (Point-to-Point Protocol), 401  
    PPPoE (Point-to-Point Protocol over Ethernet), 404-405  
        manual configuration, 405-406  
Internet Connection Wizard, 404  
IP addresses, 403  
    firewalls, 428  
IP masquerading, 371  
IPv4, 369-371  
IPv6  
    128-bit numbering, 373  
    addresses, 373-375  
IRC (Internet Relay Chat)  
    etiquette guidelines, 71  
    irc.freenode.net, 40  
    server setup, 71  
    topics, 71  
    XChat, 70  
IS/IT (information service/information technology), 29  
ISO images DVD, 2

ISO Recorder, 4  
 ISP (Internet service provider), 402-403  
 iwconfig, 399-400  
 iwlist, 399  
 iwpriv, 399  
 iwspy, 399

## J

Java, 721-722  
     Android and, 730  
 JavaScript, 722  
 Jetty, 531  
 jobs command, 207  
 JPG files, 95  
 Juju, 649-650  
     charms, 652-654  
     GUI, 654-655  
     installation, 650-651  
     Mac OS X, 655  
     Mojo, 655  
     Quickstart, 655  
     server management, 657  
     service unit deployment, 651-652  
     Wordpress blog sample, 651-652  
 JVM (Java Virtual Machine)  
     Clojure, 717  
     Groovy, 721

## K

kate text editor, 219  
 KDE desktop, 111-112  
     ark, 347-348  
     Kmail, 67  
     KOffice, 79, 80-81  
     system monitoring, 332

kdf, 332  
 kedit text editor, 219  
 keep-one-running, 697  
 kernel, 29, 447-448  
     Android, 730  
     compiling, 458-461  
         troubleshooting, 465-466  
     configuration, 461-464  
     error messages, 454  
     first kernel, 448  
     loading, at boot, 306  
     modules, 451-452  
         depmod, 453  
         /etc/modprobe.conf file, 453  
         insmod, 452  
         lsmod, 452, 453-454  
         modinfo, 453  
         modprobe, 452-453  
         removing, 454  
         rmmod, 452  
     numbering, 28  
     patches, 457-458  
     performance tuning and, 436-437, 440-441  
     /proc directory and, 156-157  
     RAM disk image, 464-465  
     source, downloading, 456  
     source tree, 449-451  
     types, 451-452  
     /usr/src/linux-4.1.3 directory, 449-450  
     versions, 455  
         Extraversion level, 455  
         Major version, 455  
         Minor version, 455  
         Sublevel number, 455  
         uname -r, 456  
 key/value stores (NoSQL), 608  
     Berkeley DB, 608-609  
     Cassandra, 609  
     Memcached/MemcacheDB, 609-610



- Redis, 610
- Riak, 610
- Kile, 83
- kill command, 227, 269, 317-318
- Kmail, 67
- Knoppix, 114
- KOffice, 79, 80-81
  - Kontakt, 81
  - KSpread, 81
- Kontakt, 81
- ksh shell, 264
- KSpread, 81
- ksysguard, 332
- Kubuntu, 2
- kubuntu-desktop, 50
- KVM (Kernel-based Virtual Machine), 635-639
  - copying, 638
  - starting, 638
  - stopping, 639
  - vmbuilder, 637
- KWord, 80

## L

- LAMP server, 2
- Landscape, 333
  - cloud computing, 656
  - server management, 659
- LANG environment variable, 215
- languages. *See also* programming languages
  - installation, 8
  - procedural languages, 590-591
- last loops, Perl, Web:12-13
- LaTeX, 82-83
- Launchpad, 689-690
- LDAP (Lightweight Directory Access Protocol), 617-618
  - administration, 623-624
  - client configuration
    - Evolution, 622
    - Thunderbird, 623
  - server configuration, 618
    - directory population, 620-621
    - schema, 618-619
- legacy hardware, 30
- legacy printers, 483-484
- less command, 171, 181, 190-192
- /lib directory, 153
- libraries, Android, 730
- LibreOffice, 76-77
  - Base, 77
  - Calc, 77
  - Dia, 77, 78
  - Draw, 77
  - history, 78
  - Math, 77
  - OpenOffice.org and, 78
  - Planner, 78
  - Writer, 77
- Liferea RSS reader, 68
- lighttpd, 529-530
- links, files, 192-194
- Linux
  - overview, 27-29
  - reasons to use, 29-31
- Linux Documentation Page, 37
- Lisp, 723
- listing
  - installed packages, 36
  - jobs, 207
  - processes, 206
  - Python, Web:30-32
- In command, 181, 192-194
- localhost interface, 364
  - ifconfig command, 364
  - manual configuration, 364-365
  - mtr command, 368

- ping command, 366-367
- traceroute command, 367-368
- locate command, 182, 194
- log files, 321-323
  - Apache, 511-513
  - rotating, 323-325
- LogFormat statement, 511-512
- logical operators, comparing, 286-287
- login, console, text-based, 148-149
- logname command, 255
- logout, console, 149
- logrotate utility, 323-325
- looping
  - Perl
    - for, Web:11
    - do...until, Web:13
    - do...while, Web:13
    - foreach, Web:11
    - last, Web:12-13
    - next, Web:12-13
    - until, Web:12
    - while, Web:12
  - PHP, Web:57-59
  - Python, Web:33-35
- lost data, 336-337
- /lost+found directory, 153
- Lovelace, Ada, 716
- ls command, 158-160, 161-162, 163-164, 182, 194-196
- lspci command, 104
- LTS (Long Term Support) release, 20
- LTSP (Linux Terminal Server Project), 43, 625-626
  - DHCP and, 630
  - hardware
    - router, 628-629
    - server, 628
    - switch/hub, 627

- installation, 629-630
- LDM (LTSP Display Manager), 630
- requirements, 626-629
- using, 630-631

- ltsp-server-standalone, 629
- Lua, 723
- Lubuntu, 2, 113-114
- LUG (Linux Users Group), 37
- LXDE, 113-114
- LyX, 83

## M

- MaaS (Metal-as-a-Service), 645-646, 655-656
- MAC (Mandatory Access Control), AppArmor, 430-432
- Macintosh, installation, 3
- Mail, 67
- mailing lists, 39
  - project mailing lists, 39-40
- make bzImage, 459
- make clean, 459
- make command, 182, 673
- make config utility, 460
- make menuconfig, 461
- make modules, 460
- make mrproper, 459
- make xconfig, 459
- makefiles (C/C++), 673-674
  - macros and, 674-675
  - targets, 674-675
- man command, 182, 196-197
- man pages, 152
  - displaying, 182, 196-197
- manual partitioning, 10
- MATE, 115
- Math (LibreOffice), 77

- MBR (Master Boot Record), 6
- MDAs (Mail Delivery Agent), 546-547
  - autoresponders, 558
  - Procmail, 557
  - Spamassassin, 557-558
  - Squirrelmail, 558
  - virus scanners, 558
- /media directory, 153
- Memcached, 609-610
- MemcacheDB, 609-610
- memory, displaying, 319-320
- Mercurial, 358, 686-687
- Microsoft, Sound Card, 86
- Microsoft Exchange Server, 559
- Midnight Commander, copying files, 356
- Mir, 42
- mirroring data, 342
- mkdir command, 163, 167-168, 182, 197
- mkisofs command, 100
- /mnt directory, 153
- modprobe.d/ file, 155
- mod\_python, Web:24
- modular kernels, 451-452
  - depmod, 453
  - /etc/modprobe.conf file, 453
  - insmod, 452
  - lsmod, 452, 453-454
  - modinfo, 453
  - modprobe, 452-453
  - removing, 454
  - rmmod, 452
- Module section (xorg.conf), 45, 46
- MongoDB, 612
- Monitor section (xorg.conf), 45, 47-48
- Mono, 723-724
- motherboard, support, 32
- MOTU (Masters of the Universe), 707
- mount command, 479
- MOV video, 105
- Mozilla Thunderbird, 65-66
- MPEG video, 105
- MPMs (multiprocessing modules), Apache, 497
- MS-DOS, 147
- MTAs (mail transfer agents), 543
  - Exim, 545-546
  - maildir, 546
  - Postfix, 545
    - aliases, 552-553
    - forwarding mail, 552-553
    - installation, 548-550
    - masquerading, 550-551
    - message delivery intervals, 551-552
    - relaying mail, 552
    - smarthosts, 551
  - Qmail, 545-546
  - Sendmail, 544-545
- mtr command, localhost interface, 368
- MUA (Mail User Agent), 547-548
- multicast addresses, 376
- multimedia, 85
  - CDs
    - burning, 99-101
    - ripping, 99
  - DVDs, burning, 101-103
  - graphics
    - Blender, 97
    - CinePaint, 97
    - darktable, 97
    - digiKam, 97
    - Hugin, 97
    - Inkscape, 97
    - POV-Ray, 97
    - Radiance, 97
    - Xara Xtreme, 97
  - images, graphics formats, 95-96
  - listening to music, 89
    - Banshee, 91-93
    - Rhythmbox, 89-90

- Microsoft Sound Card, 86
- photographs
  - cameras, 97-98
  - Shotwell Photo Manager, 98
- sound cards
  - ALSA (Advanced Linux Sound Architecture), 86
  - Microsoft Sound Card, 86
  - OSS (Open Sound System), 86
  - support, 86
- sound formats
  - FLAC, 88
  - MP3, 88
  - MP3 codec, 88
  - MPEG, 88
  - Ogg-Vorbis, 88
  - RAW, 88
  - WAV, 88
- video
  - editing, 107-108
  - formats, 105
  - hardware, 104-105
  - personal recorders, 107
  - viewing, 105-106
- Multiverse repository, 134
- music listening, 89
  - Banshee, 91-93
  - Rhythmbox, 89-90
  - Sound Juicer, 91
- mv command, 169-170, 182, 197-198
- MySQL, 579
  - clients, command-line, 601-602
  - configuration, 591
    - root user password, 592
  - CREATE DATABASE statement, 592
  - databases, creating, 592-594
  - mysqladmin, 592

- performance tuning
  - key buffer usage, 442-444
  - query cache, 444-445
  - query optimization, 446
- versus PostgreSQL, 588-591
  - privileges, 592-593
  - subqueries, 590

## N

- Nagios, 333
- name-based virtual hosting, 510-511
- nano command, 182
- nano text editor, 218, 219-220
- NAS (network-attached storage), 343
- Neo4j, 614
- Nessus, 422
- netmasks, 376
- netpbm tools, 96
- Network Manager applet, 24
- networks. *See also* sharing; wireless networking
  - backups, 343
  - bridges, 381
  - cabling
    - fiber optic, 380
    - UTP (unshielded twisted pair), 379-380
  - commands, 178, 408
  - configuration
    - files, 389-391
    - ifconfig, 384-387
    - netstat, 388-389
    - Network Connections, 392
    - nm-connection-editor, 392
    - route, 387-388
    - static IP addresses, 392

- DHCP, 393-394
  - activating at boot, 394-395
  - host configuration, 397-399
  - software installation, 395-396
- ethereal client, 369
- Ethernet cards, 369
- hardware
  - initialization, 381-383
  - NICs (network interface cards), 377-379
- hubs, 380-381
- localhost interface, 364
  - ifconfig command, 364
  - manual configuration, 364-365
- printers, 479
  - enabling on LAN, 480
  - SMB printing, 480-481
- routers, 381
- subnet masks, 376
- subnetting, 375-376
- switches, 380-381
- TCP/IP, 368-369
  - addressing, 369-371
  - IP masquerading, 371
  - ports, 372
- newusers command, 247, 255
- next loops, Perl, Web:12-13
- NFS (Network File System), 470
  - client configuration, 472
  - installation, 470
  - server configuration, 470-471
  - start/stop, 470
- nfs-kernel-server, 470
- Nginx, 517
  - Apache and, 519
  - c10 problem, 518
  - configuration, 520-521
    - nginx.conf file, 521-522
  - ./configure, 520
  - HTTPS, 526-528
  - installation
    - building source, 519-520
    - from repositories, 519
  - modules, 525-526
  - overview, 517-519
  - PHP setup, 524-525
  - virtual hosts, 519, 523-524
- nice command, 211-212, 318
- NICs (network interface cards), 377
  - 10BASE-T, 377
  - 100BASE-T, 378
  - 1000BASE-T, 378
  - fiber optic, 378
  - Gigabit Ethernet, 378
  - initialization, 382
  - security, promiscuous mode, 425
  - Token Ring, 377
  - wireless network interfaces, 378-379
- Nmap, 422
- NNTP (Network News Transfer Protocol), 73
- nonbooting hard drive, system rescue and, 362
- NoSQL, 605-608
  - databases, 581
  - document stores, 610-611
    - BaseX, 612-613
    - CouchDB, 611-612
    - MongoDB, 612
  - graph stores, 614
    - FlockDB, 615
    - HyperGraphDB, 615
    - Neo4j, 614
    - OrientDB, 614
  - key/value stores, 608
    - Berkeley DB, 608-609
    - Memcached, 609-610
    - MemcacheDB, 609-610
    - Redis, 610
    - Riak, 610

wide column stores, 613

BigTable, 613

HBase, 613-614

numbering system

kernels, 28

Ubuntu versions, 31-32

numbers, comparing, 284-285

Nvidia drivers, 120

## O

OCaml, 724

ODSL (Open Source Development Labs), 27

office suites

GNOME Office, 79

AbiWord, 80

Evolution, 80

GIMP, 80

Gnumeric, 80

KOffice, 79, 80-81

LibreOffice, 76-77

Base, 77

Calc, 77

Dia, 77, 78

Draw, 77

history, 78

Math, 77

Planner, 78

Writer, 77

OpenDocument files, 79

OGV/OGG video, 105

OneConf, 143

OOP (object-oriented programming), Python,  
Web:36-37

class definition, Web:36-37

class inheritance, Web:39-40

class variables, Web:37-38

constructors, Web:38-39

destructors, Web:38-39

object variables, Web:37-38

open source code, 19

open-cobol compiler, 718

OpenDocument files, 79

OpenLDAP, 618

OpenOffice.org, LibreOffice and, 78

OpenSSH, 534

openssh-server, 629

OpenStack, 647-648

Glance, 649

Horizon, 649

Keystone, 649

Neutron, 648

Nova, 648

Swift, 648

OpenVPN, 575-577

Open-Xchange, 560

operating systems, additional, 8

operators

Boolean, 214

Perl

arithmetic, Web:7-8

comparison, Web:6-7

compound, Web:7

uncategorized, Web:8

PHP, Web:51-55

opportunistic development, 683, 688-689

Bikeshed, 695-696

Launchpad, 689-690

Quickly, 691-695

version control system, 683-684

Bazaar, 685-686

Git, 687-688

Mercurial, 686-687

Subversion, 684-685

/opt directory, 153

Oracle Beehive, 560

- OrientDB, 614
- OSS (Open Sound System), 86
- Outlook client, 559
- output
  - redirecting, 202
  - string replacement, 282
  - suppressing, 203
  - viewing, 190-192

## P

- PaaS (Platform-as-a-Service), 645
- packages
  - installed, listing, 36
  - listing, 36
- packaging-dev, 701
- packet writing, DVDs, 103
- PAM (Pluggable Authentication Modules), 246
- Pan newsreader client, 73
- parameters, positional, 275-276
- partitioning, 8
  - manual, 10
  - planning, 5
- passwd command, 239-240, 255
- passwd file, 155
- passwords, 11, 423-424
  - changing in batches, 247
  - /etc/passwd file, 243-244
  - MySQL root user, 592
  - policies, 243
  - security, 247
  - shadow passwords, 244-246
- pastebinit, 697
- patches, kernel, 457-458
- PATH environment variable, 215
- pattern matching, shell, 266-267
- PCMCIA cards, 369
- PCRE (Perl-Compatible Regular Expressions), Web:69-71
- PCX files, 95
- PDF Editor, 81
- PDF files, Evince, 81
- pdksh shell, 264
  - expression comparison, 282
- performance tuning
  - Apache, 441-442
  - hard disk, 436
    - badblocks, 439
    - BIOS and, 436-437
    - e2fsck, 439
    - file access time disable, 439
    - file system, 438
    - kernel and, 436-437
    - tune2fs, 438-439
  - kernel, 440-441
  - MySQL
    - key buffer usage, 442-444
    - query cache, 444-445
    - query optimization, 446
- perl, 581
- Perl (Practical Extraction and Report Language), 724-725, Web:1
  - conditional statements, Web:9-10
  - CPAN, Web:15
    - modules, Web:15-16
  - examples, Web:2-4
    - command-line processing, Web:20-21
    - one-liners, Web:20
    - posting to Usenet, Web:19
    - purging logs, Web:18-19
    - sending mail, Web:16-18
  - Linux and, Web:1-4
  - looping
    - for, Web:11
    - do...until, Web:13
    - do...while, Web:13

- foreach, 1 Web:1
- last, Web:12-13
- next, Web:12-13
- until, Web:12
- while, Web:12
- operators
  - arithmetic, Web:7-8
  - comparison, Web:6-7
  - compound, Web:7
  - uncategorized, Web:8
- regex (regular expressions), Web:13-14
- shell access, Web:14-15
- string constants, Web:9
- variables
  - arrays, Web:5
  - hashes, Web:5
  - scalar, Web:5
  - special, Web:5-6
- versions, Web:2
- perldoc command, Web:4
- permissions, 161
  - altering, 164-165
  - assigning, 162-163
  - changing, 165, 181, 185
  - default, 161
  - directories, 163-164
  - file permissions, 234-235
  - groups, 165-159
  - set group ID, 165-167
  - set user ID, 165-167
- personal video recorder, 107
- photographs
  - Shotwell, 92
  - Shotwell Photo Manager, 98
- PHP, 725, Web:43
  - array functions
    - array\_keys( ), Web:64-65
    - array\_unique( ), Web:64
    - array\_values( ), Web:64
    - extract( ), Web:65
    - arrays, Web:46-47
    - comments, Web:49
    - constants, Web:48
    - databases, Web:72-74
    - escape sequences, Web:49-50
    - file functions
      - fclose( ), Web:66
      - feof( ), Web:66
      - file\_get\_contents( ), Web:66
      - file\_put\_contents( ), Web:66
      - filesize( ), Web:67
      - fopen( ), Web:66-67
      - fread( ), Web:66-67
      - fwrite( ), Web:66-67
    - functions, miscellaneous, Web:68-71
    - HTML forms, Web:71-72
    - including files, Web:59-60
    - isset( ) function, Web:68
    - looping, Web:57-59
    - Nginx and, Web:524-525
    - operators, Web:51-53
      - conditional statements, Web:53-54
      - special, Web:55
    - PCRE functions, Web:69-71
    - PEAR repository, Web:72-73
    - PHP mode, Web:44
    - references, Web:48-49
    - string functions
      - strlen( ), Web:60
      - strpos( ), Web:62-63
      - str\_replace( ), Web:61-62
      - substr( ), Web:62
      - trim( ), Web:61
    - switching, Web:55-57
    - unset( ) function, Web:68
    - var\_dump( ) function, Web:69
    - variables, Web:44-46
      - setting/unsetting, Web:68
      - substituting, Web:50-51



- physical security, 423-424
- PID (process ID), 316-317
- ping command, localhost interface, 366-367
- pipes, 212-214
  - shell, 268
- PiTiVi, 107
- Planner (LibreOffice), 78
- plug-ins
  - Firefox, 62-63
  - Sync, 63
- PNG files, 95
- podcasts, 90
- ports, 372
  - firewalls, enabling, 427
- positional parameters, 275-276
- Postfix
  - aliases, 552-553
  - forwarding mail, 552-553
  - installation, 548-550
  - masquerading, 550-551
  - message delivery intervals, 551-552
  - relaying mail, 552
  - smarthosts, 551
- PostgreSQL, 579
  - clients, command-line, 601-602
  - configuration, 594
    - data directory initialization, 594-595
  - databases
    - creating, 595
    - user creation, 596
    - user deletion, 596-597
- versus* MySQL, 588-591
  - privileges, 597
  - subqueries, 590
- POV-Ray, 97
- PPP (Point-to-Point Protocol), 401, 406-407
- PPPoE (Point-to-Point Protocol over Ethernet), 404-405
  - manual configuration, 405-406
- preview release testing, 35
- printenv command, 215-216
- printers
  - configuring, 21
  - detecting, 21
  - network, 479
    - enabling on LAN, 480
    - SMB printing, 480-481
- printing
  - all-in-one devices, 483
  - command location, 182, 202
  - commands, 484
  - CUPS (Common UNIX Printing Systems), 481-483
  - disk usage, 181, 186-187
  - files, 181
    - contents, 183
  - last file lines, 182, 200-201
  - legacy printers, 483-484
  - resource usage, 209-211
  - USB printers, 483-484
- privileges, super user, 150-151, 233
- /proc directory, 153
  - kernel interaction and, 156-157, 316
- procedural languages, 590-591
- processes
  - kill command, 317-318
  - listing, 206
  - prioritizing, 211-212
- Processes tab (System Monitor), 326
- processors, multiple, 32
- Procmail, 557
- productivity applications, 75-77
  - DocBook, 81-82
  - Evince, 81
  - GNOME Office, 79
    - AbiWord, 80
    - Evolution, 80
    - GIMP, 80

- Gnumeric, 80
- GTK widget set, 79
- KOffice, 79-81
- LaTeX, 82-83
- LibreOffice, 76-77
  - Base, 77
  - Calc, 77
  - Dia, 77, 78
  - Draw, 77
  - history, 78
  - Math, 77
  - Planner, 78
  - Writer, 77
- PDF files, 81
- Windows, 83
- XML, 81-82
- programming languages, 671-672, 715-716
  - Ada, 716
  - Clojure, 717
  - COBOL (Common Business Oriented Language), 717-718
  - D, 718
  - Dart, 718-719
  - Elixir, 719
  - Erlang, 719
  - Forth, 720
  - Fortran, 720
  - GNOME Glade client, 679-680
  - Go, 720-721
  - Groovy, 721
  - Haskell, 721
  - IDE (integrated development environment), 680-681
  - Java, 721-722
  - JavaScript, 722
  - KDevelop client, 678
  - Lisp, 723
  - Lua, 723
  - Mono, 723-724
  - OCaml, 724
  - Perl, 724-725
  - PHP, 725
  - Python, 725
  - Ruby, 725-726
  - Rust, 726
  - Scala, 726
  - Scratch, 726-727
  - SDK (software development kits), 680-681
  - Vala, 727
- programs
  - compiling, 182
  - installing, 182
  - running, commands, 178
- project mailing lists, 39-40
- promiscuous mode, 425
- proprietary code, 19
- proprietary drivers, gaming, 120
- proxy servers, 563
  - overview, 563-564
  - reasons to deploy, 563-564
- Squid
  - access control lists, 565
  - client configuration, 564-565
  - client IP addresses, 569-570
  - configuration samples, 570-572
  - installation, 564
- ps command, 206, 212, 315
- Publican, 82
- PulseAudio, 86
- Puppet, 658
- pwd command, 161
- PWD environment variable, 215
- Python, 691, 725, Web:23-24
  - binary, Web:24
  - conditionals, Web:33-35
  - dictionaries, Web:32-33

- functions, Web:35
  - definition, Web:35-36
- lists, Web:30-32
- looping, Web:33-35
- mod\_python, Web:24
- numbers, Web:25-27
- OOP, Web:36-37
  - class definition, Web:36-37
  - class inheritance, Web:39-40
  - class variables, Web:37-38
  - constructors, Web:38-39
  - destructors, Web:38-39
  - object variables, Web:37-38
- Python 3.x, Web:23
- Python Package Index, Web:41
- shebang line, Web:24
- Standard Library, Web:41
- strings, Web:27-30

## Q

- QA testing team, 711
- QT video, 105
- Quickly, 691-695
- quota, 253, 321
- quotatool, 253
- quotes, variables in strings
  - embedded spaces, 280
  - maintaining, 281

## R

- Radiance, 97
- RAID arrays, 342
- RAM, 32
- RAM disk image, kernal configuration and, 464-465

- RARP (Reverse Address Resolution Protocol), 377
- RDBMS (relational database management system), 582-584
  - SQL and
    - creating tables, 584-585
    - inserting data, 585-586
    - retrieving data, 586-588
- recompiling, 454-455
- recording video, 107
- records, DNS, 664
  - A, 664
  - AAAA, 664
  - CNAME, 665
  - MX, 665
  - NS, 665-666
  - SOA, 666-667
  - TXT, 667
- recovery, 432-433
- redirection, 202-203
  - shell, 267-268
  - streams, 203-204
- regular expressions
  - commands, 171
  - Perl, Web:13-14
- REISUB, 226-227
- remote access
  - SSH
    - copying files, 412-413
    - key-based login, 413-415
    - scp command, 412-413
    - server setup, 411
    - sftp command, 413
    - ssh-keygen command, 413-415
  - Telnet, 409-411
    - SSH comparison, 411
  - VNC (Virtual Network Computing), 415-417
- remote computer, login/logout, 149-150
- Remote Desktop, 416-417

removable storage media, backups, 342-343

renice command, 318

repeat statement, 295

repositories

    Multiverse, 134

    software, 18-20

        compiling, 141-142

    Ubuntu-specific websites, 38-39

    Universe, 134

resources

    backups and, 337-338

    commercial support, 36-37

    guides, 37-38

    IRC (Internet Relay Chat), 40

    Linux Documentation Page, 37

    LUG (Linux Users Group), 37

    mailing lists, 39

    project mailing lists, 39-40

    usage limits

        background tasks and, 207-209

        listing jobs, 207

        listing processes, 206

    usage printing, 209-211

Resources tab (System Monitor), 326

Rhythmbox, 89-90

ripping CDs, 90, 91-93, 99

rm command, 169, 182, 198

rmdir command, 168

ROI (return on investment), 29

root account, 151, 172

    sudo command, 172-175

    system reboot, 177-178

    system shutdown, 176-177

    users

        creating, 175-176

        deleting, 176

/root directory, 153

root user, 232-234

    MySQL, 592

Rootkit Hunter, 433

rootsign, 697

rotating log files, 323-325

routers (networks), 381

rsh shell, 264

RSS readers

    Firefox, 67

    Liferea, 68

rsync, file copy, 356-358

rtcwake tool, 262-264

Ruby, 725-726

runlevels, 303-304

    changing, 309-310

    definitions, 306-307

    init command and, 306

    troubleshooting, 310

run-one, 697

run-this-one, 697

Rust, 726

## S

SaaS (Software-as-a-Service), 645

Samba

    configuration, 474-475

        global behavior, 475-476

        home directory sharing, 476

        printers, 477

    /etc/samba/smb.conf, 474-475

        global section, 475-476

        homes section, 476

        printers section, 477

    mounting shares, 479

    samba-doc, 474

    samba-doc-pdf, 474

    sharing from, 473

    SMB and, 472-473

- smbclient, 478
- smbd daemon, 477-478
  - smbclient, 478
  - smbstatus, 478
- smbstatus, 478
- testing, 477
- /sbin directory, commands, 154-155
- Scala, 726
- scalac sourceFile, 726
- scalar variables, Web:5
- scanners, 94-95
- scheduling tasks
  - at, 257-258
  - batch, 259-260
  - repeating jobs, 260-262
  - wake computer, 262-264
- schema files, LDAP, 618-619
- Scorched 3D, 122-123
- Scratch, 726-727
- screen capture, 96-97
- screen command, 225
- Screen section (xorg.conf), 45, 48-49
- scripts
  - init, 308
  - shell, 269-270
    - backslash in, 281-282
    - expression comparison, 282-287
    - functions, 300-301
    - positional parameters, 275-276
    - running, 271
    - shebang line, 272-273
    - special characters, 279-280
    - statements, 291-300
    - storage, 272
    - string replacement, 282
    - variables, 273-274, 278-279
- task automation, 276-278
- SDK (software development kits), 680-681
  - Android, 731-733
  - PPA (Personal Package Archive), 736
- searches, 187-189
  - string, 189-190
  - Web searches, 35-36
- security, 419-421
  - AppArmor, 430-432
  - disaster recovery, 432-433
  - firewall, configuration, 426-430
  - NIC promiscuous mode, 425
  - password management, 247
  - passwords, 423-424
  - physical, 423-424
  - Tripwire, 424-425
  - viruses, 425-426
  - vulnerability assessment, 421-422
  - wireless networks, 423
- sed command, 222-224
- select statement, 296
- server management
  - Ansible, 659
  - CFEngine, 658-659
  - Chef, 658
  - Juju, 657
  - Landscape, 659
  - Puppet, 658
- ServerLayout section (xorg.conf), 45-46
- servers. *See also* proxy servers
  - graphical, 29
  - Linux as platform, 30
- services
  - manual stop/start, 310
  - running, at boot, 303-304
- session writing, DVDs, 102
- set group ID permission, 165-167
- set user ID permission, 165-167

- sh shell, 264
- shadow passwords, 244-246
- shaped windows, 43
- shared data, /usr directory and, 157-158
- sharing, 469-470. *See also* networking
  - from desktop, 473
  - NFS (Network File System) and, 470-472
  - Samba, 473-474
    - manual configuration, 474-477
    - mounting shares, 479
    - SMB, 472-473
    - smbd daemon, 477-478
    - testing, 477
- shebang line, 272-273, Web:3
  - Python, Web:24
- shell, 29
  - background tasks, 269
  - bash, 264
  - command line, 265-266
  - commands, job control, 265
  - ksh, 264
  - pattern matching, 266-267
  - pdksh, 264
  - Perl access, Web:14-15
  - piping data, 268
  - redirection in, 267-268
  - rsh, 264
  - scripts, 269-270
    - backslash in, 281-282
    - expression comparison, 282-287
    - functions, 300-301
    - positional parameters, 275-276
    - running, 271
    - shebang line, 272-273
    - special characters, 279-280
    - statements, 291-300
    - storage, 272
    - string replacement, 282
    - variables, 273-274, 278-279
  - sh, 264
  - tcsh, 264
  - utilities, 29
  - zsh, 264
- SHELL environment variable, 215
- shift statement, 296
- shortcuts
  - CCSM (CompizConfig Settings Manager) and, 58
  - command line, 229
- Shotwell Photo Manager, 92, 98
- shutdown, 12-13
- shutdown command, 12, 176-177
- Shuttleworth, Mark, 31
- simple backup strategy, 340-341
- Skolnick, Cliff, 485
- slapd package, 618
- Slashdot Effect, 441
- Smart Scopes (Unity), 57
- SMB (Session Message Block), 472-473
- SMTP (Simple Mail Transfer Protocol), 543
- Snappy Ubuntu, 655
- Snappy Ubuntu Core, 143
- software. *See also* Ubuntu Software Center
  - backup
    - afio, 353
    - Amanda, 352-353
    - ark (KDE), 347-348
    - Back In Time, 350-351
    - dd, 344
    - Déjà Dup, 344, 348-350
    - File Roller, 347
    - flexbackup, 353
    - tar, 344
    - tar tool, 345-347
    - Unison, 352
  - compiling from source
    - repositories, 141-142
    - tar files, 140-141

- configuration management
  - dotdee directories, 143
  - OneConf, 143
  - Snappy Ubuntu Core, 143
- installing
  - Synaptic, 133
  - Ubuntu Software Center and, 132
- repositories, 18-20
- Synaptic and, 132-134
- updates, 134-135
  - APT and, 136-140
- Software Sources, 18
- Software Updater, 14, 134-135
  - Software Sources, 18
- SOHO (small office/home office), 29, 32, 33
  - wireless networking, 401
- sort command, 182, 198-200
- sorting, files, 198-200
- sound
  - GStreamer, 88
  - listening to music, 89
    - Banshee, 91-93
    - Rhythmbox, 89-90
  - podcasts, 90
  - PulseAudio, 86
  - volume adjustment, 87
- sound cards
  - ALSA (Advanced Linux Sound Architecture), 86
  - Microsoft Sound Card, 86
  - OSS (Open Sound System), 86
  - support, 86
- sound formats
  - converting, 89
  - FLAC, 88
  - MP3, 88
  - MP3 codec, 88
  - MPEG, 88
  - Ogg-Vorbis, 88
  - RAW, 88
  - WAV, 88
- spaces, strings, embedded, 280
- Spamassassin, 557-558
- special, Web:5-6
- special characters, 279-280
- Speed Dreams, 126
- SQL (Structured Query Language), 584
  - column types, 584
  - CONCAT( ) function, 587
  - CREATE statement, 585
  - INSERT statement, 585-586
  - queries, subqueries, 590
  - SELECT statement, 586-587
  - tables, 584-585
    - data insertion, 585-586
    - data retrieval, 586-588
- Squid
  - access control lists, 565
  - client configuration, 564-565
  - client IP addresses, 569-570
  - configuration samples, 570-572
  - installation, 564
- Squirrelmail, 558
- /srv directory, 153
- SSH (Secure Shell), 14
  - commands
    - scp, 412-413
    - sftp, 413
    - ssh-keygen, 413-415
  - copying files, 412-413
  - database access, 598-599
  - key-based login, 413-415
  - server setup, 411
  - Telnet comparison, 411
- ssh command, 148, 182
- ssh-import-id, 697
- SSID (service set identifier), 24

- Stallman, Richard M., 27
- StarOffice, 78
- Startup Disk Creator, 4
- stderr, 203, 204
- stdin, 203
- stdout, 203, 204
- Steam entertainment platform, 127-128
- storage, shell scripts, 272
- streams, redirection and, 203-204
- strings
  - comparing, 283-284
  - embedded spaces, 280
  - input, 181, 189-190
  - Python, Web:27-30
  - replacing with output, 282
  - unexpected variables, 281
- su command, 248-250, 255
- subnet masks, 376
- subnetting, 375-376
- substitute user, 248
- Subversion, 358, 684-685
- sudo command, 17, 150, 172-175, 250-252, 255
- sudoers file, 155
- super user, 150-151, 172, 232-234
- SuperTux, 123-124
- support
  - commercial, 36-37
  - guides, 37-38
  - LUG (Linux Users Group), 37
- SVG files, 95
- Swing GUI library, 722
- switches (networks), 380-381
- Synaptic, install/remove software, 133
- Sync (Xmarks), 63
- /sys directory, 153
- sysctl command, 440
- syslogd daemon, 278
- system administrator, privilege granting to regular users, 247-252
- System Monitor, 326
- system monitoring
  - Conky, 327-332
  - console-based
    - disk space, 320-321
    - kill command, 317-318
    - log file rotating, 323-325
    - log files, 321-323
    - memory display, 319-320
    - priority scheduling, 318-319
    - ps command, 315
  - enterprise, 333
  - graphical tools, 325
  - KDE, 332
  - System Monitor, 326
- system reboot, 177-178
- system rescue
  - GRUB2 boot loader restore, 361
  - nonbooting hard drive and, 362
  - Ubuntu rescue disk, 361
- system reset, 226-227
- System Settings (Unity), 56
- System Settings menu
  - power management, 21-22
  - printers, 21
  - Time & Date, 22-23
- System tab (System Monitor), 326
- system time, 22-23
- system users, 234
- systemctl command, 312
- systemd command, 304, 309, 312
- systohc option, 23
- SysVinit, 304



## T

- tail command, 182, 200-201
- tape drives, backups, 343-344
- tar command, 224
  - copying files, 354-355
- tar files, 140-141
- tar tool, 344, 345
  - full backups, 345-346
  - incremental backups, 345-346
  - restoring files, 346-347
- tasks
  - automation with scripts, 276-278
  - background, 207-208
    - moving to and from, 208-209
  - scheduling
    - at, 257-258
    - batch, 259-260
    - repeating jobs, 260-262
    - wake computer, 262-264
- TCP/IP (Transport Control Protocol/Internet Protocol), 368-369
  - addressing, 369-371
    - IPv4, 369-371
    - IPv6, 372-375
  - UDP (Universal Datagram Protocol), 368
- tcsh shell, 264
  - expression comparison, 287-291
- telinit command, 309-310
- Telnet, 409-411
- telnet command, 148
- TERM environment variable, 215
- Terminal, 147
- terminals
  - multiple, 225-226
  - Test Drive, 711
    - configuration, 712-714
    - menu, 711-712
  - testdrive-gtk, 714
  - testing, 709
    - Bug Squad, 711
  - teams
    - community teams, 709-710
    - QA Team, 711
    - Ubuntu Testing Team, 710
- Texmaker, 83
- text editors
  - emacs, 218, 221-222
  - gedit, 219
  - kate, 219
  - kedit, 219
  - nano, 218, 219-220
  - vi, 220-221
  - vim, 218
- text files, appending, 203
- text-based console login, 148-149
- thttpd, 532
- thumb drive, installation from, 4
- Thunderbird, 623
- TIF files, 95
- tilde (~), 148
- time, system time, 22-23
- Time & Date settings, 22-23
- time command, 318
- /tmp directory, 153
  - file storage, 158
- Token Ring networking, 377
- Tomcat (Apache), 532
- top command, 209-211, 319
- Torvalds, Linus, 27
  - first kernel, 448
- Totem Movie Player, 106
- touch command, 161-162, 167
- touchscreen, 736
- traceroute command, localhost interface, 367-368
- Tripwire, 424-425

## troubleshooting

- Internet connection, 407-408
- kernel compile, 465-466
- runlevels, 310

tty1 virtual console, 147

tune2fs utility, 438-439

TuxPaint, 127

## U

### Ubuntu

- 32-bit, 4-5
- 64-bit, 4-5
- code names, 32
- flavors, 3
- overview, 31-32
- version numbers, 31-32

Ubuntu GNOME, 115

Ubuntu Kylin, 116

Ubuntu Mobile, 736

- application, 736
- SDK packages, PPA, 736

Ubuntu Software Center, 82, 131-132

- installing software, 132

Ubuntu Testing Team, 710

Ubuntu Touch, 736

ubuntu-restricted-extras, 105

UDP (Universal Datagram Protocol), 368

UEFI (Unified Extensible Firmware Interface), 6

UFW (Ubuntu's Uncomplicated Firewall), 312

- commands, 426, 429
- GUFW, 428

UIDs (user IDs), 234

uname command, 28

unicast addresses, 376

Unison, 352

### Unity, 29

- CCSM (CompizConfig Settings Manager), 57
  - shortcuts and, 58
- customizing, 56-58
- Dash, 52-55
- desktop, 51
- Launcher, 51
- Panel, 55-56
- shortcuts, 58
- Smart Scopes, 57
- System Settings, 56
- Unity Tweak Tool, 58

Universal USB Installer, 4

Universe repository, 134

UNIX, 147

unless statements (Perl), Web:10

UnQL, 608

until loops, Perl, Web:12

until statement, 295

updates, 12

- software, 134-135
  - APT and, 136-140
  - Software Updater, 14

Upstart, 311-312

- systemd and, 312-313

upstart, 304

### USB

- installation from, 4, 6
- wireless networking devices, 369

USB printers, 483-484

used memory display, 319-320

Usenet newsgroups, 72

- articles, 72
- bulletin board systems, 72
- hierarchy, 72
- NNTP, 73
- Pan, 73

- user accounts, 231-232
  - commands, 254-255
  - /etc/passwd file, 232
  - GID (group ID), 234
  - root, 232-234
  - stereotypes, 235
  - super user, 232-234
  - system users, 234
  - UID (user ID), 234
- USER environment variable, 215
- user management commands, 178
- user variables, shell scripts, 274
- useradd command, 237, 239, 255
- usermod command, 237, 240, 255
- users
  - adding, 240-242
  - creating, 175-176
  - deleting, 176
  - disk quotas, 253
  - groups, 235
    - listing, 235-236
  - management tools
    - chsh, 240
    - passwd, 239-240
    - useradd, 239
    - usermod, 240
  - monitoring, 242
  - password management
    - batch changing, 247
    - /etc/passwd, 243-244
    - policies, 243
    - security, 247
    - shadow passwords, 244-246
  - privileges, super user, 150-151
  - substitute, 248
  - system administrator privileges, 247-252
  - temporary identity change, 247-250
- /usr directory, 43-44, 153
  - shared data and, 157-158

- /usr/src/linux-4.1.3 directory, 449-450
- utilities
  - apt-get, 15-16
  - bzip2, 456
  - e2fsck, 439
  - hdparm, 437-438
  - logrotate, 323-325
  - make config, 460
  - Shell, 29
  - tune2fs, 438-439
- UTP (unshielded twisted pair) cable, 377

## V

- Vala, 727
- /var directory, 153
  - variable data files, 158
- variables
  - built-in, 278-279
    - shell scripts, 273
  - environment
    - displaying, 215-216
    - LANG, 215
    - PATH, 215
    - PWD, 215
    - SHELL, 215
    - shell scripts, 273
    - TERM, 215
    - USER, 215
  - Perl
    - arrays, Web:5
    - hashes, Web:5
    - scalar, Web:5
    - special, Web:5-6
  - PHP, Web:44-46
    - substituting, Web:50-51
  - Python, Web:37-38

- shell scripts, 273-274
- strings
  - embedded spaces, 280
  - unexpected, 281
- user, shell scripts, 274
- values
  - accessing, 274
  - assigning, 274
- /var/log directory, 321-323
- version control, 683-684
  - Bazaar, 685-686
  - configuration files, 358-360
  - Git, 687-688
  - Mercurial, 686-687
  - Subversion, 684-685
- version numbers, 31-32
- vi text editor, 220-221
- video
  - Adobe Flash, 106
  - AVI, 105
  - Avidemux, 108
  - Blender, 108
  - Cinelerra, 108
  - CinePaint, 108
  - drivers, gaming, 120
  - editing, 107-108
  - FLV, 105
  - hardware, 104-105
  - Kdenlive, 108
  - MOV, 105
  - MPEG, 105
  - OGV/OGG, 105
  - OpenShot, 108
  - PiTiVi, 107
  - QT, 105
  - recording, 107
  - Totem Movie Player, 106
  - TV display, 104-105
  - ubuntu-restricted-extras, 105
  - viewing, 105-106
  - VLC, 106
  - WEBM, 105
- video conferencing, 68-69
- vim command, 182
- vim text editor, 218
- virtinst, 637
- virtual consoles, tty1, 147
- virtual hosting
  - Apache
    - address-based, 509-510
    - name-based, 510-511
  - Nginx, 523-524
- VirtualBox, 639-640
- virtualization, 633-635
  - AMD-V, 635
  - KVM (Kernel-based Virtual Machine), 635-639
  - VirtualBox, 639-640
  - VMware, 641
  - VT-x (Intel), 635
  - Xen, 641
- virus scanners, 558
- viruses, 425-426
- VLC video viewer, 106
- VM (virtual machine), 634
- vmbuilder, 637
- VMware, 641
- VNC (Virtual Network Computing), 415-417
- vncviewer, 332
- volume adjustment, 87
- VPN connections, 25
- VPNs (virtual private networks), 563, 572
  - client setup, 574-575
  - Internet router metaphor, 572
  - OpenVPN, 575-577
  - server setup, 575-577

vsftpd  
 configuration, 538  
   anonymous access, 539  
   files, 539-540  
 default behaviors, 540  
 FTP user, 536  
 VT-x (Intel), 635

## W

Warsow, 121-122  
 Warty Warthog, 32  
 Web searches, 35-36  
   Google, 36  
 WEBM video, 105  
 websites, Ubuntu-specific, 38-39  
 WEP encryption, 25  
 whereis command, 153  
 which command, 182, 202  
 while loops, Perl, Web:12  
 while statement, 293-294  
 wide column stores (NoSQL), 613  
   BigTable, 613  
   HBase, 613-614  
 wildcards, commands, 171  
 Wily Werewolf, 32  
 window managers, switching, 50  
 Windows  
   games, 128  
   Wine, 83  
 windows, 110  
   shaped, 43  
 Wine, 83  
   games, 128  
 wireless networks  
   advantages, 401  
   configuration, 24-25

  interfaces, 378-379  
   iwconfig, 399  
   iwlist, 399  
   iwpriv, 399  
   iwsby, 399  
   protocols, 401-402  
   security, 423  
 WPA encryption, 25  
 Writer (LibreOffice), 77  
 WYSIWYG (what you see is what you get), 83

## X

X protocol, 42  
 X server, 29, 42  
   clients, 42  
 X Window System, 41  
   desktop, 42  
   display manager, 50  
   Mir, 42  
   popularity, 43  
   session startup, 49  
   shaped windows, 43  
 X11, X protocol, 42  
 Xara Xtreme, 97  
 xargs command, 182  
 XChat, 70  
 xconfig, 461-464  
 Xen, 641  
 Xfce, 112-113  
 XFree86, 42  
 XML (eXtensible Markup Language), 81-82  
 X.Org, 42-44  
   /usr directory, 43-44

778 xorg.conf file

xorg.conf file, 44

- Device, 45, 48

- Files, 45

- InputDevice, 45, 46-47

- Module, 45, 46

- Monitor, 45, 47-48

- Screen, 45, 48-49

- ServerLayout, 45-46

Xubuntu, 112-113

## Y

Yahoo!, 66-67

Yaws, 530

## Z

Zenoss, 333

zsh shell, 264