

# CONTENTS

---

*PREFACE xxvii*

## CHAPTER 1: WELCOME TO LINUX 1

The GNU–Linux Connection	2
The History of GNU–Linux	2
The Code Is Free	4
Have Fun!	5
The Heritage of Linux: UNIX	5
What Is So Good About Linux?	6
Why Linux Is Popular with Hardware Companies and Developers	7
Linux Is Portable	8
Standards	8
The C Programming Language	9
Overview of Linux	10
Linux Has a Kernel Programming Interface	10
Linux Can Support Many Users	10
Linux Can Run Many Tasks	11
Linux Provides a Secure Hierarchical Filesystem	11
The Shell: Command Interpreter and Programming Language	12
A Large Collection of Useful Utilities	14
Interprocess Communication	14
System Administration	14
Additional Features of Linux	15
GUIs: Graphical User Interfaces	15
(Inter)networking Utilities	16
Software Development	16
Chapter Summary	16
Exercises	17

## PART I THE LINUX OPERATING SYSTEM 19

### CHAPTER 2: GETTING STARTED 21

- Conventions Used in This Book 22
- Logging In 24
  - Logging In from a Terminal 24
  - Logging In Remotely: Terminal Emulation, ssh, and telnet 25
- Working with the Shell 25
  - Which Shell Are You Running? 26
  - Correcting Mistakes 26
- Curbing Your Power: Superuser Access 28
- Getting the Facts: Where to Find Documentation 29
  - The **--help** Option 29
  - man: Displays the System Manual 30
  - info: Displays Information About Utilities 32
  - HOWTOs: Finding Out How Things Work 34
  - Using the Internet to Get Help 34
- More About Logging In 35
  - What to Do If You Cannot Log In 36
  - Logging Out 36
  - Using Virtual Consoles 36
  - Changing Your Password 37
- Chapter Summary 38
- Exercises 39
- Advanced Exercises 39

### CHAPTER 3: COMMAND LINE UTILITIES 41

- Special Characters 42
- Basic Utilities 43
  - ls: Lists the Names of Files 43
  - cat: Displays a Text File 44
  - rm: Deletes a File 44
  - less Is more: Displaying a Text File One Screen at a Time 45
  - hostname: Displays the System Name 45
- Working with Files 45
  - cp: Copies a File 45
  - mv: Changes the Name of a File 46
  - lpr: Prints a File 47
  - grep: Finds a String 48
  - head: Displays the Beginning of a File 49
  - tail: Displays the End of a File 49
  - sort: Displays a File in Order 50
  - uniq: Removes Duplicate Lines from a File 51

---

diff: Compares Two Files	51
file: Tests the Contents of a File	52
(Pipe): Communicates Between Processes	52
Four More Utilities	53
echo: Displays Text	53
date: Displays the Time and Date	54
script: Records a Linux Session	54
unix2dos: Converts Linux Files to Windows Format	55
Compressing and Archiving Files	56
bzip2: Compresses a File	56
bunzip2 and bzip2: Decompress a File	57
gzip: Compresses a File	58
tar: Packs and Unpacks Files	58
Locating Commands	61
which and whereis: Locate a Utility	61
apropos: Searches for a Keyword	62
slocate: Searches for a File	63
Obtaining User and System Information	63
who: Lists Users on the System	64
finger: Lists Users on the System	64
w: Lists Users on the System	66
Communicating with Other Users	67
write: Sends a Message	67
mesg: Denies or Accepts Messages	68
Email	69
Chapter Summary	69
Exercises	72
Advanced Exercises	73

## CHAPTER 4: THE LINUX FILESYSTEM 75

The Hierarchical Filesystem	76
Directory and Ordinary Files	77
Filenames	78
mkdir: Creates a Directory	80
The Working Directory	81
Home Directory	82
Absolute Pathnames	83
Relative Pathnames	84
Important Standard Directories and Files	86
Working with Directories	88
rmdir: Deletes a Directory	88
Pathnames	89
mv, cp: Moves or Copies a File	90
mv: Moves a Directory	90

- Access Permissions 91
  - ls -l**: Displays Permissions 91
  - chmod: Changes Access Permissions 92
  - Setuid and Setgid Permissions 94
  - Directory Access Permissions 94
- Links 96
  - Hard Links 97
  - Symbolic Links 99
  - rm: Removes a Link 101
- Chapter Summary 102
- Exercises 103
- Advanced Exercises 105

## CHAPTER 5: THE SHELL 107

- The Command Line 108
  - Syntax 108
  - Processing the Command Line 111
  - Executing the Command Line 113
- Standard Input and Standard Output 113
  - The Screen as a File 114
  - The Keyboard and Screen as Standard Input and Standard Output 115
  - Redirection 116
  - Pipes 122
- Running a Program in the Background 125
- Filename Generation/Pathname Expansion 127
  - The ? Special Character 128
  - The \* Special Character 129
  - The [ ] Special Characters 130
- Builtins 132
- Chapter Summary 133
  - Utilities and Builtins Introduced in This Chapter 134
- Exercises 134
- Advanced Exercises 136

## PART II THE EDITORS 137

### CHAPTER 6: THE vim EDITOR 139

- History 140
- Tutorial: Creating and Editing a File with vim 141
  - Starting vim 141

---

Command and Input Modes	142
Entering Text	144
Getting Help	144
Ending the Editing Session	147
The <b>compatible</b> Parameter	148
Introduction to vim Features	148
Online Help	149
Modes of Operation	149
The Display	150
Correcting Text as You Insert It	151
Work Buffer	151
Line Length and File Size	151
Windows	151
File Locks	152
Abnormal Termination of an Editing Session	152
Recovering Text After a Crash	153
Command Mode: Moving the Cursor	154
Moving the Cursor by Characters	155
Moving the Cursor to a Specific Character	155
Moving the Cursor by Words	156
Moving the Cursor by Lines	156
Moving the Cursor by Sentences and Paragraphs	157
Moving the Cursor Within the Screen	157
Viewing Different Parts of the Work Buffer	158
Input Mode	158
Inserting Text	158
Appending Text	159
Opening a Line for Text	159
Replacing Text	159
Quoting Special Characters in Input Mode	159
Command Mode: Deleting and Changing Text	160
Undoing Changes	160
Deleting Characters	160
Deleting Text	160
Changing Text	162
Replacing Text	163
Changing Case	163
Searching and Substituting	164
Searching for a Character	164
Searching for a String	164
Substituting One String for Another	166
Miscellaneous Commands	170
Join	170
Status	171
. (Period)	171

- Yank, Put, and Delete Commands 171
  - The General-Purpose Buffer 171
  - Named Buffers 172
  - Numbered Buffers 173
- Reading and Writing Files 174
  - Reading Files 174
  - Writing Files 174
  - Identifying the Current File 175
- Setting Parameters 175
  - Setting Parameters from Within vim 175
  - Setting Parameters in a Startup File 176
  - The `.vimrc` Startup File 176
  - Parameters 177
- Advanced Editing Techniques 180
  - Using Markers 180
  - Editing Other Files 181
  - Macros and Shortcuts 182
  - Executing Shell Commands from Within vim 182
- Units of Measure 184
  - Character 184
  - Word 185
  - Blank-Delimited Word 185
  - Line 185
  - Sentence 186
  - Paragraph 186
  - Window 187
  - Repeat Factor 187
- Chapter Summary 188
- Exercises 193
- Advanced Exercises 194

## CHAPTER 7: THE `emacs` EDITOR 195

- History 196
  - Evolution 196
  - `emacs` Versus `vim` 197
- Tutorial: Getting Started with `emacs` 198
  - Starting `emacs` 198
  - Stopping `emacs` 199
  - Inserting Text 199
  - Deleting Characters 199
  - Moving the Cursor 200
  - Editing at the Cursor Position 202
  - Saving and Retrieving the Buffer 203
- Basic Editing Commands 204
  - Keys: Notation and Use 204

---

Key Sequences and Commands	205
META-x: Running a Command Without a Key Binding	205
Numeric Arguments	205
Point and the Cursor	206
Scrolling Through a Buffer	206
Erasing Text	207
Searching	207
Online Help	209
Advanced Editing	212
Undoing Changes	212
Mark and Region	213
Cut and Paste: Yanking Killed Text	215
Inserting Special Characters	216
Global Buffer Commands	217
Files	219
Buffers	220
Windows	222
Foreground Shell Commands	224
Background Shell Commands	224
Language-Sensitive Editing	225
Selecting a Major Mode	226
Human-Language Modes	226
C Mode	229
Customizing Indentation	232
Comments	233
Special-Purpose Modes	233
Customizing emacs	235
The <b>.emacs</b> Startup File	236
Remapping Keys	237
A Sample <b>.emacs</b> File	239
More Information	240
Access to emacs	240
Chapter Summary	241
Exercises	248
Advanced Exercises	250

## **PART III THE SHELLS 253**

### **CHAPTER 8: THE BOURNE AGAIN SHELL 255**

Background	256
Shell Basics	257
Startup Files	257
Commands That Are Symbols	260
Redirecting Standard Error	260

- Writing a Simple Shell Script 263
- Separating and Grouping Commands 267
- Job Control 271
- Manipulating the Directory Stack 274
- Parameters and Variables 277
  - User-Created Variables 278
  - Variable Attributes 281
  - Keyword Variables 283
  - Special Characters 291
- Processes 292
  - Process Structure 293
  - Process Identification 293
  - Executing a Command 294
- History 295
  - Variables That Control History 295
  - Reexecuting and Editing Commands 297
  - The Readline Library 305
- Aliases 312
  - Single Versus Double Quotation Marks in Aliases 312
  - Examples of Aliases 313
- Functions 315
- Controlling bash Features and Options 318
  - Command Line Options 318
  - Shell Features 319
- Processing the Command Line 322
  - History Expansion 323
  - Alias Substitution 323
  - Parsing and Scanning the Command Line 323
  - Command Line Expansion 323
- Chapter Summary 332
- Exercises 334
- Advanced Exercises 336
  
- CHAPTER 9: THE TC SHELL 339**
- Shell Scripts 340
- Entering and Leaving the TC Shell 341
  - Startup Files 342
- Features Common to the Bourne Again and TC Shells 343
  - Command Line Expansion (Substitution) 344
  - Job Control 348
  - Filename Substitution 348
  - Manipulating the Directory Stack 349
  - Command Substitution 349
- Redirecting Standard Error 349



---

Working with the Command Line	350
Word Completion	350
Editing the Command Line	353
Correcting Spelling	354
Variables	355
Variable Substitution	356
String Variables	356
Arrays of String Variables	357
Numeric Variables	358
Braces	360
Special Variable Forms	361
Shell Variables	361
Control Structures	368
if	368
goto	371
Interrupt Handling	371
if...then...else	372
foreach	373
while	375
break and continue	376
switch	376
Builtins	377
Chapter Summary	381
Exercises	382
Advanced Exercises	384

## **PART IV PROGRAMMING TOOLS 385**

### **CHAPTER 10: PROGRAMMING TOOLS 387**

Programming in C	388
Checking Your Compiler	388
A C Programming Example	389
Compiling and Linking a C Program	393
Using Shared Libraries	396
Fixing Broken Binaries	397
Creating Shared Libraries	398
make: Keeps a Set of Programs Current	399
Implied Dependencies	401
Macros	404
Debugging C Programs	407
gcc: Compiler Warning Options	408
Symbolic Debugger	411
Threads	417

System Calls	417
strace: Traces System Calls	418
Controlling Processes	418
Accessing the Filesystem	419
Source Code Management	420
CVS: Concurrent Versions System	420
Chapter Summary	430
Exercises	431
Advanced Exercises	432
<b>CHAPTER 11: PROGRAMMING THE BOURNE AGAIN SHELL</b>	<b>435</b>
Control Structures	436
if...then	437
if...then...else	440
if...then...elif	442
for...in	449
for	451
while	453
until	456
break and continue	459
case	459
select	466
Here Document	468
File Descriptors	470
Parameters and Variables	474
Array Variables	474
Locality of Variables	475
Special Parameters	478
Positional Parameters	480
Expanding Null and Unset Variables	485
Builtin Commands	487
type: Displays Information About a Command	487
read: Accepts User Input	487
exec: Executes a Command	491
trap: Catches a Signal	493
kill: Aborts a Process	497
getopts: Parses Options	497
A Partial List of Builtins	500
Expressions	501
Arithmetic Evaluation	501
Logical Evaluation (Conditional Expressions)	503
String Pattern Matching	504
Operators	505
Shell Programs	510
A Recursive Shell Script	510
The <b>quiz</b> Shell Script	513

---

Chapter Summary 520  
Exercises 522  
Advanced Exercises 524

## CHAPTER 12: THE **gawk** PATTERN PROCESSING LANGUAGE 527

Syntax 528  
Arguments 528  
Options 529  
Notes 529  
Language Basics 530  
    Patterns 530  
    Actions 531  
    Comments 531  
    Variables 531  
    Functions 532  
    Arithmetic Operators 533  
    Associative Arrays 534  
    **printf** 534  
    Control Structures 535  
Examples 537  
Advanced gawk Programming 554  
    **getline**: Controlling Input 554  
    Coprocess: Two-Way I/O 557  
    Getting Input from a Network 558  
Error Messages 559  
Chapter Summary 560  
Exercises 561  
Advanced Exercises 561

## CHAPTER 13: THE **sed** EDITOR 563

Syntax 564  
Arguments 564  
Options 564  
Editor Basics 565  
    Addresses 565  
    Instructions 566  
    Control Structures 567  
    The Pattern Space and the Hold Space 568  
Examples 568  
Chapter Summary 578  
Exercises 579

## PART V COMMAND REFERENCE 581

- Standard Multiplicative Suffixes 586
- Common Options 587
- The **sample** Utility 587
  - sample Very brief description of what the utility does 588
  - aspell Checks a file for spelling errors 589
    - at Executes commands at a specified time 593
  - bzip2 Compresses or decompresses files 596
  - cal Displays a calendar 598
  - cat Joins and displays files 599
  - cd Changes to another working directory 601
  - chgrp Changes the group associated with a file 603
  - chmod Changes the access mode (permissions) of a file 604
  - chown Changes the owner of a file and/or the group the file is associated with 608
  - cmp Compares two files 610
  - comm Compares sorted files 612
  - configure Configures source code automatically 614
  - cp Copies files 616
  - cpio Creates an archive or restores files from an archive 619
  - crontab Maintains crontab files 624
  - cut Selects characters or fields from input lines 627
  - date Displays or sets the system time and date 630
  - dd Converts and copies a file 633
  - df Displays disk space usage 636
  - diff Displays the differences between two files 638
  - du Displays information on disk usage by file 644
  - echo Displays a message 647
  - expr Evaluates an expression 649
  - file Displays the classification of a file 653
  - find Finds files based on criteria 655
  - finger Displays information about users 661
  - fmt Formats text very simply 664
  - fsck Checks and repairs a filesystem 666
  - ftp Transfers files over a network 671
  - gcc Compiles C and C++ programs 678
  - grep Searches for a pattern in files 683
  - gzip Compresses or decompresses files 688
  - head Displays the beginning of a file 691
  - kill Terminates a process by PID 693
  - killall Terminates a process by name 695
  - less Displays text files, one screen at a time 697
  - ln Makes a link to a file 702

- 
- lpr Sends files to printers 705
  - ls Displays information about one or more files 708
  - make Keeps a set of programs current 715
  - man Displays documentation for commands 721
  - mkdir Creates a directory 724
  - mkfs Creates a filesystem on a device 725
  - Mtools Uses DOS-style commands on files and directories 728
    - mv Renames or moves a file 732
    - nice Changes the priority of a command 734
  - nohup Runs a command that keeps running after you log out 736
  - od Dumps the contents of a file 737
  - paste Joins corresponding lines from files 742
    - pr Paginates files for printing 744
    - ps Displays process status 746
  - rcp Copies one or more files to or from a remote system 750
  - rlogin Logs in on a remote system 752
  - rm Removes a file (deletes a link) 753
  - rmdir Removes a directory 755
    - rsh Executes commands on a remote system 756
    - scp Securely copies one or more files to or from a remote system 758
  - sleep Creates a process that sleeps for a specified interval 760
  - sort Sorts and/or merges files 762
  - split Divides a file in into sections 771
  - ssh Securely executes commands on a remote system 773
  - strings Displays strings of printable characters 777
  - stty Displays or sets terminal parameters 778
  - tail Displays the last part (tail) of a file 783
  - tar Stores or retrieves files to/from an archive file 786
  - tee Copies standard input to standard output and one or more files 791
  - telnet Connects to a remote system over a network 792
  - test Evaluates an expression 794
  - top Dynamically displays process status 798
  - touch Changes a file's access and/or modification time 801
    - tr Replaces specified characters 804
    - tty Displays the terminal pathname 807
  - tune2fs Changes parameters on an ext2 or ext3 filesystem 808
  - umask Establishes the file-creation permissions mask 810
  - uniq Displays unique lines 812
    - w Displays information about system users 814
    - wc Displays the number of lines, words, and bytes 816
  - which Shows where in PATH a command is located 817
  - who Displays information about logged-in users 819
  - xargs Converts standard input into command lines 821

## **PART VI APPENDICES 825**

### **APPENDIX A: REGULAR EXPRESSIONS 827**

- Characters 828
- Delimiters 828
- Simple Strings 828
- Special Characters 828
  - Periods 829
  - Brackets 829
  - Asterisks 830
  - Carets and Dollar Signs 830
  - Quoting Special Characters 831
- Rules 831
  - Longest Match Possible 831
  - Empty Regular Expressions 832
- Bracketing Expressions 832
- The Replacement String 833
  - Ampersand 833
  - Quoted Digit 833
- Extended Regular Expressions 834
- Appendix Summary 835

### **APPENDIX B: HELP 837**

- Solving a Problem 838
- Finding Linux-Related Information 839
  - Documentation 839
  - Useful Linux Sites 840
  - Linux Newsgroups 841
  - Mailing Lists 841
  - Words 841
  - Software 842
  - Office Suites and Word Processors 844
- Specifying a Terminal 844

### **APPENDIX C: KEEPING THE SYSTEM UP-TO-DATE 847**

- yum: Updates and Installs Packages 848
  - Configuring yum 848
  - Using yum 849
- Apt: An Alternative to yum 850
  - Using Apt 851

BitTorrent	855
Prerequisites	856
How BitTorrent Works	856
Using BitTorrent	857

*GLOSSARY* 859

*INDEX* 913

