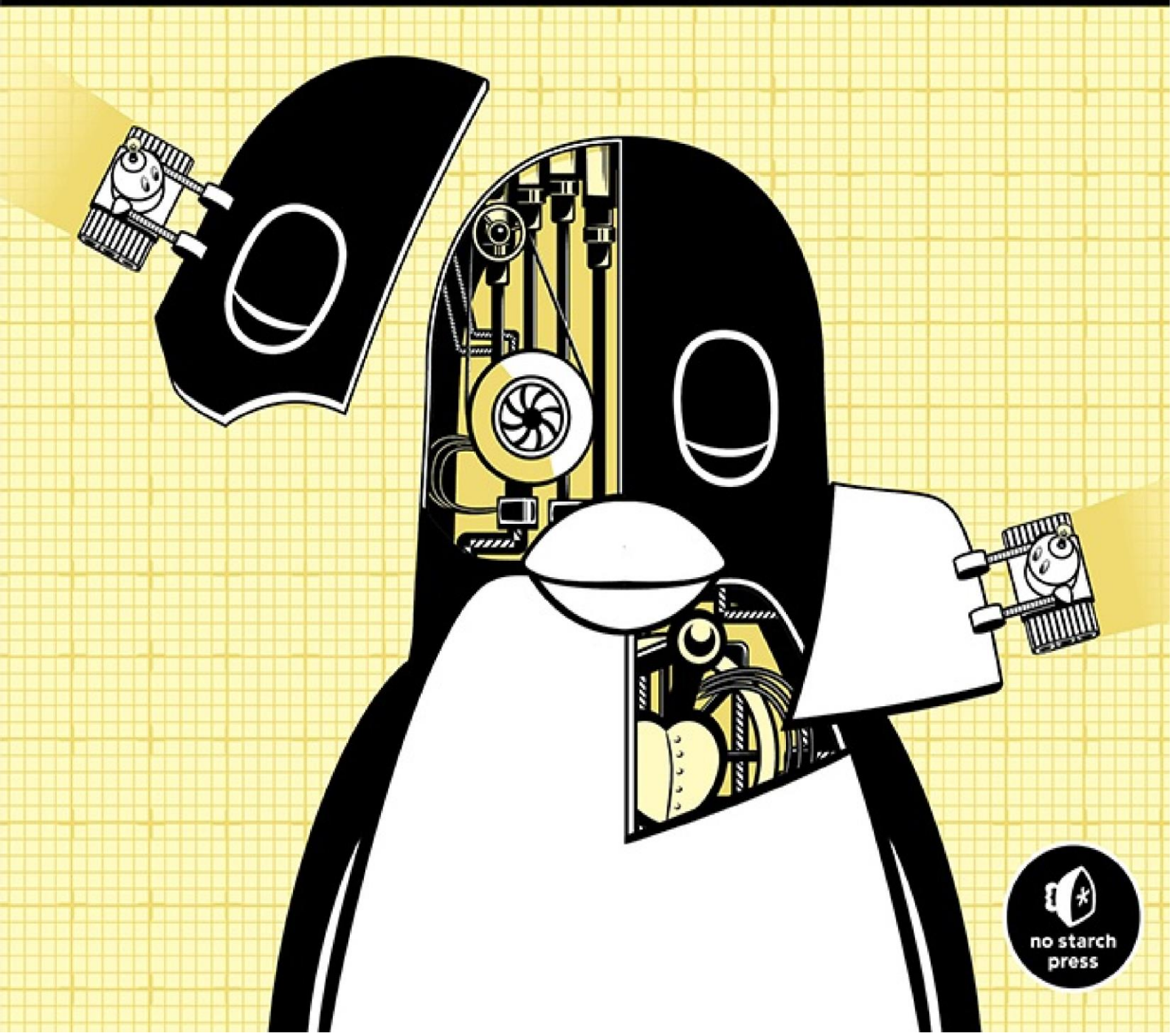


3RD EDITION

HOW LINUX WORKS

WHAT EVERY SUPERUSER SHOULD KNOW

BRIAN WARD



Contents

About the Author.....	18
About the Technical Reviewers.....	18
Prerequisites.....	20
How to Read This Book.....	21
A Hands-on Approach.....	21
How This Book Is Organized.....	22
A Note on Terminology.....	23
THE BIG PICTURE.....	24
Linux System.....	25
1.2 Hardware: Understanding Main Memory.....	28
1.3 The Kernel.....	29
1.4 User Space.....	34
1.5 Users.....	35
1.6 Looking Forward.....	37
BASIC COMMANDS AND DIRECTORY HIERARCHY.....	38
2.1 The Bourne Shell: /bin/sh.....	39
2.2 Using the Shell.....	40
2.2.1 The Shell Window.....	40
2.2.2 cat.....	41
2.2.3 Standard Input and Standard Output.....	42
2.3 Basic Commands.....	43
2.3.1 ls.....	43
2.3.2 cp.....	44
2.3.3 mv.....	44
2.3.4 touch.....	44
2.3.5 rm.....	45
2.4 Navigating Directories.....	45
2.4.1 cd.....	46
2.4.2 mkdir.....	47

2.4.3 rmdir	47
2.4.4 Shell Globbing (“Wildcards”)	47
2.5 Intermediate Commands.....	49
2.5.1 grep.....	49
2.5.2 less.....	50
2.5.3 pwd.....	51
2.5.4 diff.....	51
2.5.5 file.....	52
2.5.6 find and locate.....	52
2.5.7 head and tail.....	52
2.5.8 sort	53
2.6 Changing Your Password and Shell.....	53
2.7 Dot Files	53
2.8 Environment and Shell Variables	54
2.9 The Command Path.....	55
2.10 Special Characters.....	56
2.11 Command-Line Editing.....	57
2.12 Text Editors.....	58
2.13 Getting Online Help	59
2.14 Shell Input and Output	62
2.14.1 Standard Error.....	62
2.14.2 Standard Input Redirection	63
2.15 Understanding Error Messages	64
2.15.1 Anatomy of a Unix Error Message.....	64
2.15.2 Common Errors.....	65
2.16 Listing and Manipulating Processes	67
2.16.1 ps Command Options	68
2.16.2 Process Termination	68
2.16.3 Job Control.....	70
2.16.4 Background Processes.....	70

2.17 File Modes and Permissions	71
2.17.1 Modifying Permissions	73
2.17.2 Working with Symbolic Links	75
2.18 Archiving and Compressing Files	77
2.18.1 gzip.....	77
2.18.2 tar	77
2.18.3 Compressed Archives (.tar.gz)	79
2.18.4 zcat	79
2.18.5 Other Compression Utilities	80
2.19 Linux Directory Hierarchy Essentials.....	81
2.19.1 Other Root Subdirectories	83
2.19.2 The /usr Directory.....	83
2.19.3 Kernel Location	84
2.20 Running Commands as the Superuser.....	84
2.20.1 sudo	84
2.20.2 /etc/sudoers.....	85
2.20.3 sudo Logs	86
2.21 Looking Forward.....	86
DEVICES	87
3.1 Device Files	88
Block device	89
Character device.....	89
Pipe device	89
Socket device.....	89
3.2 The sysfs Device Path	90
3.3 dd and Devices.....	91
3.4 Device Name Summary	92
3.4.1 Hard Disks: /dev/sd*	93
3.4.2 Virtual Disks: /dev/xvd*, /dev/vd*	94
3.4.3 Non-Volatile Memory Devices: /dev/nvme*	95

3.4.4 Device Mapper: /dev/dm-*, /dev/mapper/*	95
3.4.5 CD and DVD Drives: /dev/sr*	95
3.4.6 PATA Hard Disks: /dev/hd*	95
3.4.7 Terminals: /dev/tty*, /dev/pts/*, and /dev/tty	95
3.4.8 Serial Ports: /dev/ttyS*, /dev/ttyUSB*, /dev/ttyACM* ..	97
3.4.9 Parallel Ports: /dev/lp0 and /dev/lp1	98
3.4.10 Audio Devices: /dev/snd/*, /dev/dsp, /dev/audio, and More	98
3.4.11 Device File Creation	99
3.5 udev	99
3.5.1 devtmpfs	100
3.5.3 udevadm	104
3.5.4 Device Monitoring	105
3.6 In-Depth: SCSI and the Linux Kernel	106
3.6.1 USB Storage and SCSI	111
3.6.2 SCSI and ATA	112
3.6.3 Generic SCSI Devices	113
3.6.4 Multiple Access Methods for a Single Device	114
DISKS AND FILESYSTEMS	115
4.1 Partitioning Disk Devices	118
4.1.1 Viewing a Partition Table	119
4.1.2 Modifying Partition Tables	122
4.1.3 Creating a Partition Table	124
4.1.4 Navigating Disk and Partition Geometry	126
4.1.5 Reading from Solid-State Disks	128
4.2 Filesystems	130
4.2.1 Filesystem Types	130
4.2.2 Creating a Filesystem	132
4.2.3 Mounting a Filesystem	133
4.2.4 Filesystem UUID	135
4.2.5 Disk Buffering, Caching, and Filesystems	136

4.2.6 Filesystem Mount Options.....	137
4.2.7 Remounting a Filesystem.....	139
4.2.8 The /etc/fstab Filesystem Table	139
4.2.9 Alternatives to /etc/fstab	141
4.2.10 Filesystem Capacity.....	141
4.2.11 Checking and Repairing Filesystems.....	143
4.2.12 Special-Purpose Filesystems.....	146
4.3 Swap Space	147
4.3.1 Using a Disk Partition as Swap Space	148
4.3.2 Using a File as Swap Space	148
4.3.3 Determining How Much Swap You Need.....	149
4.4 The Logical Volume Manager.....	150
4.4.2 Working with LVM	152
4.4.3 The LVM Implementation	165
4.5 Looking Forward: Disks and User Space	169
4.6 Inside a Traditional Filesystem	170
4.6.1 Inode Details and the Link Count	173
4.6.2 Block Allocation	175
4.6.3 Working with Filesystems in User Space	175
HOW THE LINUX KERNEL BOOTS	177
5.1 Startup Messages	178
5.2 Kernel Initialization and Boot Options	179
5.3 Kernel Parameters	180
5.4 Boot Loaders.....	182
5.4.1 Boot Loader Tasks.....	183
5.4.2 Boot Loader Overview.....	184
5.5 GRUB Introduction	185
5.5.1 Exploring Devices and Partitions with the GRUB Command Line	188
5.5.2 GRUB Configuration.....	191
5.5.3 GRUB Installation.....	194

5.6 UEFI Secure Boot Problems	196
5.7 Chainloading Other Operating Systems	197
5.8 Boot Loader Details	198
5.8.1 MBR Boot	198
5.8.2 UEFI Boot	199
5.8.3 How GRUB Works	200
HOW USER SPACE STARTS.....	202
6.1 Introduction to init	203
6.2 Identifying Your init	204
6.3 systemd.....	204
6.3.1 Units and Unit Types.....	205
6.3.2 Booting and Unit Dependency Graphs.....	206
6.3.3 systemd Configuration.....	207
6.3.4 systemd Operation.....	210
6.3.5 systemd Process Tracking and Synchronization	215
6.3.6 systemd Dependencies.....	216
6.3.7 systemd On-Demand and Resource-Parallelized Startup	221
6.3.8 systemd Auxiliary Components.....	227
6.4 System V Runlevels	228
6.5 System V init.....	229
6.5.1 System V init: Startup Command Sequence	231
6.5.2 The System V init Link Farm	232
6.5.3 run-parts	233
6.5.4 System V init Control.....	234
6.5.5 systemd System V Compatibility	235
6.6 Shutting Down Your System	235
6.7 The Initial RAM Filesystem.....	237
6.8 Emergency Booting and Single-User Mode	239
6.9 Looking Forward	240
LOGGING, SYSTEM TIME, BATCH JOBS, AND USERS	241

7.1 System Logging	242
7.1.1 Checking Your Log Setup	243
7.1.2 Searching and Monitoring Logs	243
7.1.3 Logfile Rotation	248
7.1.4 Journal Maintenance.....	249
7.1.5 A Closer Look at System Logging	249
7.2 The Structure of /etc.....	253
7.3 User Management Files	254
7.3.1 The /etc/passwd File	254
7.3.2 Special Users	256
7.3.3 The /etc/shadow File	256
7.3.4 Manipulating Users and Passwords.....	257
7.3.5 Working with Groups	257
7.4 getty and login	259
7.5 Setting the Time.....	260
7.5.1 Kernel Time Representation and Time Zones.....	260
7.5.2 Network Time.....	261
7.6 Scheduling Recurring Tasks with cron and Timer Units	262
7.6.1 Installing Crontab Files	264
7.6.2 System Crontab Files	265
7.6.3 Timer Units	265
7.6.4 cron vs. Timer Units.....	268
7.7 Scheduling One-Time Tasks with at	268
7.7.1 Timer Unit Equivalentents	269
7.8 Timer Units Running as Regular Users.....	270
7.9 User Access Topics.....	270
7.9.1 User IDs and User Switching	270
7.9.2 Process Ownership, Effective UID, Real UID, and Saved UID.....	271
7.9.3 User Identification, Authentication, and Authorization	273
7.9.4 Using Libraries for User Information	275

7.10 Pluggable Authentication Modules.....	276
7.10.1 PAM Configuration.....	276
7.10.2 Tips on PAM Configuration Syntax	281
7.10.3 PAM and Passwords.....	282
7.11 Looking Forward	283
A CLOSER LOOK AT PROCESSES AND RESOURCE	
UTILIZATION.....	284
8.1 Tracking Processes.....	285
8.2 Finding Open Files with lsof	286
8.2.1 Reading the lsof Output	286
8.2.2 Using lsof	287
Calls	288
8.3.1 strace	288
8.3.2 ltrace	290
8.4 Threads	291
8.5 Introduction to Resource Monitoring	293
8.6 Control Groups (cgroups).....	306
8.7 Further Topics.....	313
UNDERSTANDING YOUR NETWORK AND ITS	
CONFIGURATION.....	314
9.1 Network Basics.....	315
9.2 Packets	316
9.3 Network Layers.....	316
9.4 The Internet Layer	318
9.4.1 Viewing IP Addresses	320
9.4.2 Subnets.....	321
9.4.3 Common Subnet Masks and CIDR Notation.....	322
9.5 Routes and the Kernel Routing Table	323
9.6 The Default Gateway	324
HOW THE KERNEL CHOOSES A ROUTE.....	325
9.7 IPv6 Addresses and Networks.....	325

9.7.1 Viewing IPv6 Configuration on Your System.....	327
9.7.2 Configuring Dual-Stack Networks	328
9.8 Basic ICMP and DNS Tools	328
9.8.1 ping.....	328
9.8.2 DNS and host.....	330
9.9 The Physical Layer and Ethernet	331
9.10 Understanding Kernel Network Interfaces	332
Configuration.....	333
9.11.1 Manually Configuring Interfaces	333
9.11.2 Manually Adding and Deleting Routes.....	334
9.12 Boot-Activated Network Configuration	335
Activated Network Configuration.....	336
9.14 Network Configuration Managers.....	337
9.15 Resolving Hostnames	341
9.16 Localhost	346
Services	347
9.17.1 TCP Ports and Connections	347
9.17.2 UDP	351
9.18 Revisiting a Simple Local Network.....	353
9.19 Understanding DHCP	354
9.20 Automatic IPv6 Network Configuration.....	355
9.21 Configuring Linux as a Router	356
9.22 Private Networks (IPv4)	358
Masquerading).....	359
9.24 Routers and Linux	361
9.25 Firewalls.....	362
9.26 Ethernet, IP, ARP, and NDP.....	370
9.27 Wireless Ethernet.....	372
9.28 Summary.....	375
NETWORK APPLICATIONS AND SERVICES.....	377

10.1 The Basics of Services.....	377
10.2 A Closer Look.....	379
10.3 Network Servers.....	381
10.3.1 Secure Shell.....	382
10.3.2 The sshd Server	384
10.3.3 fail2ban.....	387
10.3.4 The SSH Client	387
Servers: inetd/xinetd.....	389
TCP WRAPPERS: TCPD, /ETC/HOSTS.ALLOW, AND /ETC/HOSTS.DENY.....	390
10.5 Diagnostic Tools	390
10.6 Remote Procedure Calls	397
10.7 Network Security	397
10.8 Looking Forward.....	401
10.9 Network Sockets.....	402
10.10 Unix Domain Sockets	403
INTRODUCTION TO SHELL SCRIPTS	405
11.1 Shell Script Basics	405
11.1.1 Limitations of Shell Scripts	407
11.2 Quoting and Literals.....	408
11.2.1 Literals.....	408
11.2.2 Single Quotes	409
11.2.3 Double Quotes	410
11.2.4 Literal Single Quotes	410
11.3 Special Variables.....	411
11.3.1 Individual Arguments: \$1, \$2, and So On.....	412
11.3.2 Number of Arguments: \$#	412
11.3.3 All Arguments: \$@	413
11.3.4 Script Name: \$0.....	413
11.3.5 Process ID: \$\$.....	414
11.4 Exit Codes.....	414

11.5	Conditionals.....	415
11.5.1	A Workaround for Empty Parameter Lists	416
11.5.2	Other Commands for Tests.....	416
11.5.3	elif.....	417
11.5.4	Logical Constructs	417
11.5.5	Testing Conditions	418
11.5.6	case.....	421
11.6	Loops	422
11.6.1	for Loops.....	422
11.6.2	while Loops	423
11.7	Command Substitution.....	423
11.8	Temporary File Management.....	424
11.9	Here Documents.....	426
11.10	Important Shell Script Utilities	426
11.10.1	basename	427
11.10.2	awk.....	427
11.10.3	sed.....	428
11.10.4	xargs	429
11.10.5	expr	430
11.10.6	exec.....	430
11.11	Subshells.....	431
11.12	Including Other Files in Scripts.....	432
11.13	Reading User Input.....	432
11.14	When (Not) to Use Shell Scripts	433
	NETWORK FILE TRANSFER AND SHARING.....	433
	We'll start by looking at some ways.....	433
12.1	Quick Copy.....	434
12.2	rsync	435
12.3	Introduction to File Sharing.....	443
12.4	Sharing Files with Samba.....	445

12.5 SSHFS.....	454
12.6 NFS.....	455
12.7 Cloud Storage	456
12.8 The State of Network File Sharing.....	457
USER ENVIRONMENTS	459
13.1 Guidelines for Creating Startup Files.....	460
13.2 When to Alter Startup Files	460
13.3 Shell Startup File Elements	461
13.3.1 The Command Path	461
13.3.2 The Manual Page Path.....	462
13.3.3 The Prompt.....	462
13.3.4 Aliases.....	463
13.3.5 The Permissions Mask.....	464
13.4 Startup File Order and Examples	465
13.4.1 The bash Shell	465
13.4.2 The tcsh Shell.....	468
13.5 Default User Settings	469
13.5.1 Shell Defaults	469
13.5.2 Editor	470
13.5.3 Pager	470
13.6 Startup File Pitfalls	470
13.7 Further Startup Topics.....	471
A BRIEF SURVEY OF THE LINUX DESKTOP AND PRINTING .	472
14.1 Desktop Components	473
14.1.1 Framebuffers	473
14.1.2 The X Window System.....	474
14.1.3 Wayland	474
14.1.4 Window Managers	475
14.1.5 Toolkits.....	475
14.1.6 Desktop Environments	476

14.1.7	Applications.....	476
14.3	A Closer Look at Wayland.....	477
14.3.1	The Compositing Window Manager	477
14.3.2	libinput	479
14.3.3	X Compatibility in Wayland	480
14.4	A Closer Look at the X Window System	481
14.4.1	Display Managers.....	482
14.4.2	Network Transparency	483
14.4.3	Ways of Exploring X Clients.....	483
14.4.4	X Events.....	484
14.4.5	X Input and Preference Settings.....	485
14.5	D-Bus.....	488
14.5.1	System and Session Instances.....	489
14.5.2	D-Bus Message Monitoring	489
14.6	Printing	490
14.6.1	CUPS	490
14.6.2	Format Conversion and Print Filters	491
14.7	Other Desktop Topics.....	491
DEVELOPMENT TOOLS		492
15.1	The C Compiler	493
15.1.1	Compiling Multiple Source Files	494
15.1.2	Linking with Libraries.....	496
15.1.3	Working with Shared Libraries	498
15.1.4	Working with Header (Include) Files and Directories	503
15.2	make.....	506
15.2.1	A Sample Makefile	507
15.2.2	Built-in Rules	508
15.2.3	Final Program Build	509
15.2.4	Dependency Updates	509
15.2.5	Command-Line Arguments and Options	510

15.2.6	Standard Macros and Variables	511
15.2.7	Conventional Targets	512
15.2.8	Makefile Organization	513
15.3	Lex and Yacc.....	514
15.4	Scripting Languages.....	515
15.4.1	Python	516
15.4.2	Perl.....	516
15.4.3	Other Scripting Languages	517
15.5	Java.....	518
15.6	Looking Forward: Compiling Packages	519
INTRODUCTION TO COMPILING SOFTWARE FROM C SOURCE		
CODE.....		
16.1	Software Build Systems	521
16.2	Unpacking C Source Packages	521
16.3	GNU Autoconf.....	523
16.3.1	An Autoconf Example	524
16.3.2	Installation Using a Packaging Tool.....	525
16.3.3	configure Script Options.....	526
16.3.4	Environment Variables.....	527
16.3.5	Autoconf Targets.....	528
16.3.6	Autoconf Logfiles.....	529
16.3.7	pkg-config	529
16.4	Installation Practice	531
16.4.1	Where to Install.....	532
16.5	Applying a Patch	533
16.6	Troubleshooting Compiles and Installations.....	534
16.6.1	Specific Errors.....	535
16.7	Looking Forward.....	538
VIRTUALIZATION		
17.1	Virtual Machines.....	540
17.1.1	Hypervisors.....	540

17.1.2	Hardware in a Virtual Machine.....	541
17.1.3	Common Uses of Virtual Machines.....	543
17.1.4	Drawbacks of Virtual Machines	543
17.2	Containers	545
17.2.1	Docker, Podman, and Privileges	546
17.2.2	A Docker Example	547
17.2.3	LXC	557
17.2.4	Kubernetes	558
17.2.5	Pitfalls of Containers.....	559
17.3	Runtime-Based Virtualization	561
	Bibliography	562
	Index.....	567

Reviews for *How Linux Works*

“If you are interested in Linux, *How Linux Works: What Every Superuser Should Know* is a must-read title.”

—LINUXINSIDER

“Lots to offer on almost every aspect of the Linux architecture.”

—EVERYDAY LINUX USER

“You’ll get an essential understanding of what’s going on under the hood without getting bogged down in minutiae—making this a very refreshing (and wholly recommended) addition to the Linux literature.”

—PHIL BULL, CO-AUTHOR OF *UBUNTU MADE EASY* AND MEMBER OF THE UBUNTU DOCUMENTATION TEAM

“Dives straight into the transparent depths of Linuxbased operating systems and shows us how all the pieces fit together.”

—DISTROWATCH

“Earns its place on the shelf as an essential reference.”

—*THE MAGPI* MAGAZINE

HOW LINUX WORKS

3rd Edition

What Every Superuser Should Know

by Brian Ward



no starch
press

San Francisco

How Linux Works, 3rd Edition. Copyright © 2021 by Brian Ward.

All rights reserved. No part of this work may be reproduced or transmitted in any form or by any means, electronic or mechanical, including photocopying, recording, or by any information storage or retrieval system, without the prior written permission of the copyright owner and the publisher.

Publisher: William Pollock

Executive Editor: Barbara Yien

Production Editor: Rachel Monaghan

Developmental Editors: Larry Wake and Jill Franklin

Cover and Interior Design: Octopod Studios

Technical Reviewers: Jordi Gutiérrez Hermoso and Petros Koutoupis

Copyeditor: Rachel Monaghan

Compositor: Cody Gates, Happenstance Type-O-Rama Proofreader:
James M. Fraleigh

For information on book distributors or translations, please contact No Starch Press, Inc. directly:

No Starch Press, Inc.

245 8th Street, San Francisco, CA 94103 phone:

1-415-863-9900;

info@nostarch.com

www.nostarch.com

The Library of Congress has catalogued the first edition as follows:

Ward, Brian.

How Linux works : what every superuser should know / Brian Ward.

p. cm. Includes index.

ISBN 1-59327-035-6

1. Linux. 2. Operating systems (Computers). I. Title.

QA76.76.O63 W3654 2004

005.4'32--dc22 2004002692

No Starch Press and the No Starch Press logo are registered trademarks of No Starch Press, Inc. Other product and company names mentioned herein may be the trademarks of their respective owners. Rather than use a trademark symbol with every occurrence of a trademarked name, we are using the names only in an editorial fashion and to the benefit of the trademark owner, with no intention of infringement of the trademark.

The information in this book is distributed on an “As Is” basis, without warranty. While every precaution has been taken in the preparation of this work, neither the author nor No Starch Press, Inc. shall have any liability to any person or entity with respect to any loss or damage caused or alleged to be caused directly or indirectly by the information contained in it.

About the Author

Brian Ward has been working with Linux since 1993. He is also the author of *The Linux Kernel HOWTO*, *The Book of VMware* (No Starch Press), and *The Linux Problem Solver* (No Starch Press).

About the Technical Reviewers

Jordi Gutiérrez Hermoso is a GNU/Linux user and developer with almost two decades of experience, with occasional contributions

in various circles of the free software communities, such as GNU Octave and Mercurial. His professional work has allowed him to collaborate with and learn about diverse topics such as digital cryptographic signatures, medical imaging, and most recently greenhouse gas accounting and ecological map data, all built entirely with Linux and other free software. When he is not near a computer he enjoys swimming, mathematics, and knitting.

Petros Koutoupis is currently a senior performance software engineer at HPE (formerly Cray Inc.) for its Lustre High Performance File System division. He is also the creator and maintainer of the RapidDisk Project (www.rapiddisk.org). Petros has worked in the data storage industry for well over a decade and has helped pioneer the many technologies unleashed in the wild today.

ACKNOWLEDGMENTS

Contributions to this book come from not just those who were involved in the development process, but also those without whom I wouldn't know anything about Linux. These include James Duncan, Douglas N. Arnold, Bill Fenner, Ken Hornstein, Scott Dickson, Dan Ehrlich, Felix Lee, and Gregory P. Smith. Previous editions included help from Karol Jurado, Laurel Chun, Serena Yang, Alison Law, Riley Hoffman, Scott Schwartz, Dan Sully, Dominique Poulain, Donald Karon, and Gina Steele.

This third edition benefits from the excellent work of Barbara Yien, Rachel Monaghan, Jill Franklin, Larry Wake, Jordi Gutiérrez Hermoso, and Petros Koutoupis. As the publisher of No Starch Press, Bill Pollock has been instrumental in this book since the first edition. And once more, Hsinju Hsieh has put up with me for another revision.

PREFACE

Your system shouldn't be a mystery. You should be able to make your software do what you want it to do without “magic” incantations or rituals. The key to attaining this power lies in understanding the fundamentals of what the software does and how it works, and that's what this book is all about. You should never have to fight with a computer.

Linux is a great platform for learning because it doesn't try to hide anything from you. In particular, you can find most system configuration details in easy-to-read plaintext files. The only tricky part is figuring out which parts are responsible for what and how they all fit together.

Who Should Read This Book?

Your interest in learning how Linux works may have come from any number of sources. In the professional realm, operations and DevOps folks need to know nearly everything that you'll find in this book. Linux software architects and developers should also know this material in order to make the best use of the operating system. Researchers and students, often running their own Linux systems, will also find that this book provides useful explanations for *why* things are set up the way they are.

Then there are the tinkerers—people who just love to play around with their computers for fun, profit, or both. Want to know why certain things work while others don't? Want to know what happens if you move something around? You're probably a tinkerer.

Prerequisites

Although Linux is beloved by programmers, you don't need to be a programmer to read this book; you need only basic computer-user knowledge. That is, you should be able to bumble around a GUI (especially the installer and settings interface for a Linux distribution) and know what files and directories (folders) are. You should also be prepared to check additional documentation on your system and on the web. The most important thing you need is to be ready and willing to play around with your computer.