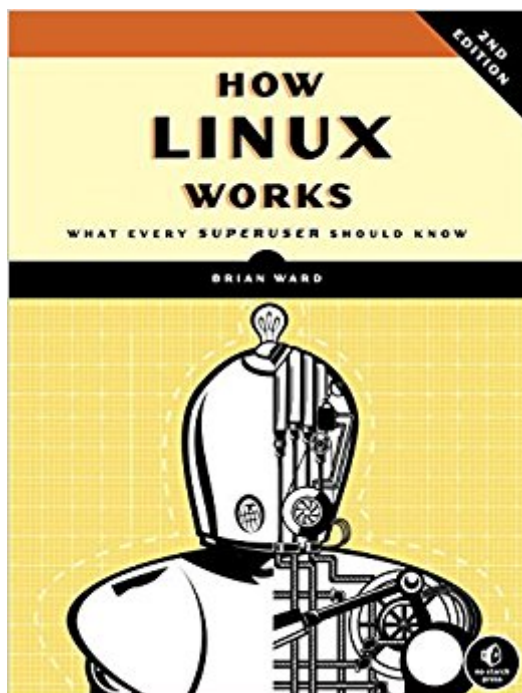


The book was found

# How Linux Works, 2nd Edition: What Every Superuser Should Know



## Synopsis

Unlike some operating systems, Linux doesn't try to hide the important bits from you—it gives you full control of your computer. But to truly master Linux, you need to understand its internals, like how the system boots, how networking works, and what the kernel actually does. In this completely revised second edition of the perennial best seller *How Linux Works*, author Brian Ward makes the concepts behind Linux internals accessible to anyone curious about the inner workings of the operating system. Inside, you'll find the kind of knowledge that normally comes from years of experience doing things the hard way. You'll learn:

- How Linux boots, from boot loaders to init implementations (systemd, Upstart, and System V)
- How the kernel manages devices, device drivers, and processes
- How networking, interfaces, firewalls, and servers work
- How development tools work and relate to shared libraries
- How to write effective shell scripts

You'll also explore the kernel and examine key system tasks inside user space, including system calls, input and output, and filesystems. With its combination of background, theory, real-world examples, and patient explanations, *How Linux Works* will teach you what you need to know to solve pesky problems and take control of your operating system.

## Book Information

Paperback: 392 pages

Publisher: No Starch Press; 2 edition (November 14, 2014)

Language: English

ISBN-10: 1593275676

ISBN-13: 978-1593275679

Product Dimensions: 7 x 1 x 9.2 inches

Shipping Weight: 1.6 pounds (View shipping rates and policies)

Average Customer Review: 4.6 out of 5 stars 173 customer reviews

Best Sellers Rank: #23,476 in Books (See Top 100 in Books) #6 in [Books > Computers & Technology > Networking & Cloud Computing > Network Administration > Linux & UNIX Administration](#) #8 in [Books > Computers & Technology > Operating Systems > Linux > Networking & System Administration](#) #27 in [Books > Textbooks > Computer Science > Operating Systems](#)

## Customer Reviews

Brian Ward has been working with Linux since 1993. He is the author of *The Linux Kernel-HOWTO*,

The Book of VMware (No Starch Press), and The Linux Problem Solver (No Starch Press).

I've read reviews all over the web of Brian's new edition, and I'm frankly confused. The range of opinions go from "This is very basic, don't bother if you're advanced" to "Don't bother if you're a beginner, way too much detail on the inner workings than you need." 3 bears? Just right for... who? Part of the problem might be the subtitle: "What every superuser should know." In Brian's usual dry sense of humor, this refers to the \$ vs. # prompt difference between user and "superuser" (kernel access) at the command prompt. It does NOT mean (goodness no) that you have to be some kind of superuser in the programming sense to understand or greatly enjoy this book. In one fell swoop Ward does a number of things with the specialized term in unix/linux (superuser is simply what Windows folk would call a sysop or administrator, or hackers would look at with rootkits/kernels as privilege escalation): 1. Letting casual users know they will be using the command line rather than graphic interface 2. Demonstrating the major update from the first edition: this book is a LOT more about the kernel than the first edition, thankfully so! This is a TOTAL update, so even though I frequently advise going back to a previous edition for unethical publishers and authors that do a cosmetic update and call it a new edition, this is NOT the case here-- Brian has totally reworked the book to bring it up to date with many features Linux was missing (especially in Ubuntu dists) when the first edition was written. This new edition is a MUST, as 10 years has been a lifetime in the Linux world-- frankly the first edition is a paperweight now. And, since this edition includes numerous detailed steps for playing along with your own IDE/SDK throughout as well as complete code on the site, it is now dist independent as much as possible. I've tried the code with Ubuntu, Debian, Fedora and RHEL so far, and found ZERO problems with only teensy adjustments. That is unusual for a newly released title, but not for No Starch, who have technical editors who actually TRY the code, unlike a lot of publishers out there in this field. If you're a beginner, there is no better introduction to operating systems than this book, period. Brian takes the time to explain in painstaking detail the INTUITIVE workings of nearly every o/s-- things you must know to go forward in any kind of programming. In my areas of robotics, embedded and security, many new candidates (some with Masters degrees in IT) flunk the interview because of lack of basic "what's really going on?" answers. These talented candidates know how to program, but when asked very basic questions like "What's the kernel doing here?" or "What does object oriented vs. functional or imperative mean at this level of system call?" their eyes glaze over. This is why, unlike some other reviewers, I strongly advise reading this book for BOTH beginners and advanced programmers. You do not have to have an objective of designing compilers either! Even basic security and hacking, when

using tools like IDA Pro, require you to be able to look at the command line and "see" what state the kernel is in, what it is doing and calling, and what certain interrupt/resume sequences really mean regardless of sequential, parallel or virtual memory paradigms. It is not enough to "let the legacy software" (or in the case of exploits, the rootkit) do the work anymore-- even basic scripters are now modifying vm's and rootkits with Python, and the typical exploit, whether pen, hack, forensics or just system admin, involves sophisticated user/kernel interactions by malware that can now not only inject sequences, but also install entire new o/s donuts around your kernel and control the entire o/s! Finally, this book is radically fun if you have any interest in computers, at any level, including high school AND grad school, as well as seasoned practicing programmers. At over 350 pages and extremely up to date, Ward packs each page with SO much information and detailed knowledge, with deeply intuitive examples, that you can either study every sentence numerous times like a text, looking up details in his many external resources, or just read casually and get a "gist" that is rare as an overview of what's really going on between kernel and user abstraction levels (or in the operational sense, interfaces). The information in this book is pertinent for both Geek Squad repair folk and MIT PhD malware designers. I guarantee, as an old professor who teaches Kali Linux to hackers and pen testers at both of these levels, and in assembly reverse engineering detail including memory forensics, that there is something delightful for everyone in this new edition... highly recommended.

Let's face it, if you are mostly a casual computer user doing email and surfing the web, then what operating system you are using is going to matter very little. Linux desktop, MS Windows, Max OS X, and Android are all going to do those tasks pretty much in the same way, and you may not really care about what's under the hood. You can go years using Linux and never see the command line. On the other hand, if you are a technoid as I am, then you are going to be curious about how your operating system works. The net result of this curiosity is gaining an order of magnitude in usability and control over your computer by understanding how it works. I have a number of books about Linux, especially Linux server administration, but none of them make the workings of Linux accessible like this book does. Mr. Ward has put together a wealth of key information about what's going on under the hood and more importantly, how to use it, and done so in easy to understand language. But let me warn you - this is not a book full of dry, passive pages; this is a lively tutorial! Mr. Ward suggests having a Linux system in front of you and trying the various explorations as you go through the book. I cannot recommend that highly enough. I also recommend having a notebook next to you and taking notes. Above all - do not fear the command line. Mr. Ward's explanations and

examples are well done. You have no more excuses for not rolling up your sleeves and getting your hands dirty. I think you are going to find that this book becomes the most used of those sitting on your computer bookshelf. Lastly, thanks, Mr. Ward, for writing this book!!

As of 07/29/2016, this book is very much well up to date and relevant to Ubuntu 16.04 from what I have read so far. The examples have all worked. It's done a good job describing the boot process, kernel space and user space. It's not too wordy so you can read it quickly and get what you need out of it. Because of this book I feel much more comfortable with Linux.

Just reading the book is educational, but after a certain point you really need to be sitting at a computer trying the examples. Worth having an actual hard copy book even though you can download the pdf version for free. Much better than a lot of the other free on line Linux guides.

Fantastic. I've gotten my toes wet in linux previously, but not very deep. This book brought me to confidence in being able to work with all the various parts, WITHOUT FEAR. I always felt sort of tentative, cautious in issuing commands before, since there were so many unknowns. This took that away, and I now feel I can get things done. There is still a lot to learn, but with these basics understood, I can proceed knowing that I at least grasp the big moving parts; whatever unknowns remain I can tackle competently.

I got the Kindle version. None of the inline commands print, so the text is effectively unreadable. I use Kindle for Android. Am I missing a font? Update: It does work on a Kindle Fire. The text that doesn't display on my Samsung tablet is a Letter Gothic like monospaced font. I'll have to investigate the problem as a Kindle for Android problem, not as a problem with the book. So I've upped rating to 4 stars.

[Download to continue reading...](#)

How Linux Works, 2nd Edition: What Every Superuser Should Know CompTIA Linux+ Powered by Linux Professional Institute Study Guide: Exam LX0-103 and Exam LX0-104 (Comptia Linux + Study Guide) What Every Student Should Know About Citing Sources with APA Documentation (What Every Student Should Know About...) Recipes Every College Student Should Know (Stuff You Should Know) Stuff Every Man Should Know (Stuff You Should Know) Insults Every Man Should Know (Stuff You Should Know) Jokes Every Man Should Know (Stuff You Should Know) Dirty Jokes Every Man Should Know (Stuff You Should Know) Stuff Every Husband Should Know

(Stuff You Should Know) Stuff Every College Student Should Know (Stuff You Should Know) Easy Linux For Beginners: A Complete Introduction To Linux Operating System & Command Line Fast! CompTIA Linux+ Guide to Linux Certification The Linux Programming Interface: A Linux and UNIX System Programming Handbook Twenty-Five Buildings Every Architect Should Understand: a revised and expanded edition of Twenty Buildings Every Architect Should Understand (Volume 2) The Road to Key West, Marathon to Key West: The guide every local should have for their guest and every visitor should have by their side (2017 Edition) 30 Things Every Woman Should Have and Should Know by the Time She's 30 The Words You Should Know to Sound Smart: 1200 Essential Words Every Sophisticated Person Should Be Able to Use The Big Book of Words You Should Know: Over 3,000 Words Every Person Should be Able to Use (And a few that you probably shouldn't) 25 Bridge Conventions You Should Know - Part 1: Learn These First (25 Bridge Conventions You Should Know - eBook Edition) 100 Things Washington Fans Should Know & Do Before They Die (100 Things...Fans Should Know)

[Contact Us](#)

[DMCA](#)

[Privacy](#)

[FAQ & Help](#)