

The Daemon The Gnu And The Penguin

The Struggle with the Daemon is a brilliant analysis of the European psyche by the great novelist and biographer Stefan Zweig. Zweig studies three giants of German literature and thought: Friedrich Holderlin, Heinrich von Kleist and Friedrich Nietzsche – powerful minds whose ideas were at odds with the scientific positivism of their age; troubled spirits whose intoxicating passions drove them mad but inspired them to great works. In their struggle with their inner creative force, Zweig reflects the conflict at the heart of the European soul – between science and art, reason and inspiration. Both highly personal and philosophically wide-ranging, this is one of the most fascinating of Zweig's renowned biographical studies. From the Trade Paperback edition.

This book is the comprehensive guide to Samba administration, officially adopted by the Samba Team. Wondering how to integrate Samba's authentication with that of a Windows domain? How to get Samba to serve Microsoft Dfs shares? How to share files on Mac OS X? These and a dozen other issues of interest to system administrators are covered. A whole chapter is dedicated to troubleshooting! The range of this book knows few bounds. Using Samba takes you from basic installation and configuration -- on both the client and server side, for a wide range of systems -- to subtle details of security, cross-platform compatibility, and resource discovery that make the difference between whether users see the folder they expect or a cryptic error message. The current edition covers such advanced 3.x features as: Integration with Active Directory and OpenLDAP Migrating from Windows NT 4.0 domains to Samba Delegating administrative tasks to non-root users Central printer management Advanced file serving features, such as making use of Virtual File System (VFS) plugins. Samba is a cross-platform triumph: robust, flexible and fast, it turns a Unix or Linux system into a file and print server for Microsoft Windows network clients. This book will help you make your file and print sharing as powerful and efficient as possible. The authors delve into the internals of the Windows activities and protocols to an unprecedented degree, explaining the strengths and weaknesses of each feature in Windows domains and in Samba itself. Whether you're playing on your personal computer or an enterprise network, on one note or a full three-octave range, Using Samba will give you an efficient and secure server.

This practical guidebook explains not only how to get a computer up and running with the FreeBSD operating system, but how to turn it into a highly functional and secure server that can host large numbers of users and disks, support remote access and provide key parts of the Inter

Life is not always easy or fair for the follower of Christ. Yet, Sheryl Giesbrecht has learned, from the depths of past pain, it is possible to be raised up. It is possible to heal. Most importantly, it is possible to exchange hurt for hope. In her

inspirational new book, *Get Back Up: Trusting God When Life Knocks You Down*, Sheryl tells her own story. She uses scripture to tell the stories of so many other Biblical figures who stumbled before they could be helped up. Sheryl's message is one of hope. Trusting God in the midst of a life turned upside down is essential to survival. Without Him, life would not be worth living. Climbing out of the pits of despair would be impossible. Yes, with God to lean on, finding hope is possible. Healing is possible. God is capable of taking our losses and mistakes and turning them into something remarkably beautiful. Won't you let Him?

Focusing on the design decisions and standards which have made internetworking possible, this new book charts the intriguing history of this communications/computing phenomenon. From its beginnings as a Department of Defense project to its current position as the global network for computing communications, the full Internet story is told here.

Find solutions to all your problems related to Linux system programming using practical recipes for developing your own system programs

Key Features

- Develop a deeper understanding of how Linux system programming works
- Gain hands-on experience of working with different Linux projects with the help of practical examples
- Learn how to develop your own programs for Linux

Book Description

Linux is the world's most popular open source operating system (OS). *Linux System Programming Techniques* will enable you to extend the Linux OS with your own system programs and communicate with other programs on the system. The book begins by exploring the Linux filesystem, its basic commands, built-in manual pages, the GNU compiler collection (GCC), and Linux system calls. You'll then discover how to handle errors in your programs and will learn to catch errors and print relevant information about them. The book takes you through multiple recipes on how to read and write files on the system, using both streams and file descriptors. As you advance, you'll delve into forking, creating zombie processes, and daemons, along with recipes on how to handle daemons using `systemd`. After this, you'll find out how to create shared libraries and start exploring different types of interprocess communication (IPC). In the later chapters, recipes on how to write programs using POSIX threads and how to debug your programs using the GNU debugger (GDB) and Valgrind will also be covered. By the end of this Linux book, you will be able to develop your own system programs for Linux, including daemons, tools, clients, and filters. What you will learn

- Discover how to write programs for the Linux system using a wide variety of system calls
- Delve into the working of POSIX functions
- Understand and use key concepts such as signals, pipes, IPC, and process management
- Find out how to integrate programs with a Linux system
- Explore advanced topics such as filesystem operations, creating shared libraries, and debugging your programs
- Gain an overall understanding of how to debug your programs using Valgrind

Who this book is for

This book is for anyone who wants to develop system programs for Linux and gain a deeper understanding of the Linux system. The book is beneficial for anyone who is facing issues related to a particular part of

Linux system programming and is looking for specific recipes or solutions.

Based on the popular Artech House classic, Digital Communication Systems Engineering with Software-Defined Radio, this book provides a practical approach to quickly learning the software-defined radio (SDR) concepts needed for work in the field. This up-to-date volume guides readers on how to quickly prototype wireless designs using SDR for real-world testing and experimentation. This book explores advanced wireless communication techniques such as OFDM, LTE, WLA, and hardware targeting. Readers will gain an understanding of the core concepts behind wireless hardware, such as the radio frequency front-end, analog-to-digital and digital-to-analog converters, as well as various processing technologies. Moreover, this volume includes chapters on timing estimation, matched filtering, frame synchronization message decoding, and source coding. The orthogonal frequency division multiplexing is explained and details about HDL code generation and deployment are provided. The book concludes with coverage of the WLAN toolbox with OFDM beacon reception and the LTE toolbox with downlink reception. Multiple case studies are provided throughout the book. Both MATLAB and Simulink source code are included to assist readers with their projects in the field.

Once upon a time Linus Torvalds was a skinny unknown, just another nerdy Helsinki techie who had been fooling around with computers since childhood. Then he wrote a groundbreaking operating system and distributed it via the Internet -- for free. Today Torvalds is an international folk hero. And his creation LINUX is used by over 12 million people as well as by companies such as IBM. Now, in a narrative that zips along with the speed of e-mail, Torvalds gives a history of his renegade software while candidly revealing the quirky mind of a genius. The result is an engrossing portrayal of a man with a revolutionary vision, who challenges our values and may change our world.

This book is for all people who are forced to use UNIX. It is a humorous book--pure entertainment--that maintains that UNIX is a computer virus with a user interface. It features letters from the thousands posted on the Internet's "UNIX-Haters" mailing list. It is not a computer handbook, tutorial, or reference. It is a self-help book that will let readers know they are not alone.

This is the eBook version of the printed book. If the print book includes a CD-ROM, this content is not included within the eBook version. Advanced Linux Programming is divided into two parts. The first covers generic UNIX system services, but with a particular eye towards Linux specific information. This portion of the book will be of use even to advanced programmers who have worked with other Linux systems since it will cover Linux specific details and differences. For programmers without UNIX experience, it will be even more valuable. The second section covers material that is entirely Linux specific. These are truly advanced topics, and are the techniques that the gurus use to build great applications. While this book will focus mostly on the Application Programming Interface (API) provided by the Linux kernel and the C library, a preliminary introduction to the development tools available will allow all who purchase the book to make immediate use of Linux. A guide to Ubuntu covers such topics as installation, configuration, the filesystem, the command line, system maintenance and security, networking, using OpenOffice.org, Web browsing, and playing games.

"Covers GNU Make basics through advanced topics, including: user-defined functions, macros, and path handling; creating makefile

assertions and debugging makefiles; parallelization; automatic dependency generation, rebuilding targets, and non-recursive Make; and using the GNU Make Standard Library"--

Even small applications have dozens of components. Large applications may have thousands, which makes them challenging to install, maintain, and remove. Docker bundles all application components into a package called a container that keeps things tidy and helps manage any dependencies on other applications or infrastructure. Docker in Action, Second Edition teaches you the skills and knowledge you need to create, deploy, and manage applications hosted in Docker containers. This bestseller has been fully updated with new examples, best practices, and entirely new chapters. You'll start with a clear explanation of the Docker model and learn how to package applications in containers, including techniques for testing and distributing applications. Purchase of the print book includes a free eBook in PDF, Kindle, and ePub formats from Manning Publications.

An annotated guide to program and develop GNU/Linux Embedded systems quickly About This Book Rapidly design and build powerful prototypes for GNU/Linux Embedded systems Become familiar with the workings of GNU/Linux Embedded systems and how to manage its peripherals Write, monitor, and configure applications quickly and effectively, manage an external micro-controller, and use it as co-processor for real-time tasks Who This Book Is For This book targets Embedded System developers and GNU/Linux programmers who would like to program Embedded Systems and perform Embedded development. The book focuses on quick and efficient prototype building. Some experience with hardware and Embedded Systems is assumed, as is having done some previous work on GNU/Linux systems. Knowledge of scripting on GNU/Linux is expected as well. What You Will Learn Use embedded systems to implement your projects Access and manage peripherals for embedded systems Program embedded systems using languages such as C, Python, Bash, and PHP Use a complete distribution, such as Debian or Ubuntu, or an embedded one, such as OpenWrt or Yocto Harness device driver capabilities to optimize device communications Access data through several kinds of devices such as GPIO's, serial ports, PWM, ADC, Ethernet, WiFi, audio, video, I2C, SPI, One Wire, USB and CAN Practical example usage of several devices such as RFID readers, Smart card readers, barcode readers, z-Wave devices, GSM/GPRS modems Usage of several sensors such as light, pressure, moisture, temperature, infrared, power, motion In Detail Embedded computers have become very complex in the last few years and developers need to easily manage them by focusing on how to solve a problem without wasting time in finding supported peripherals or learning how to manage them. The main challenge with experienced embedded programmers and engineers is really how long it takes to turn an idea into reality, and we show you exactly how to do it. This book shows how to interact with external environments through specific peripherals used in the industry. We will use the latest Linux kernel release 4.4.x and Debian/Ubuntu distributions (with embedded distributions like OpenWrt and Yocto). The book will present popular boards in the industry that are user-friendly to base the rest of the projects on - BeagleBone Black, SAMA5D3 Xplained, Wandboard and system-on-chip manufacturers. Readers will be able to take their first steps in programming the embedded platforms, using C, Bash, and Python/PHP languages in order to get access to the external peripherals. More about using and programming device driver and accessing the peripherals will be covered to lay a strong foundation. The readers will learn how to read/write data from/to the external environment by using both C programs or a scripting language (Bash/PHP/Python) and how to configure a device driver for a specific hardware. After finishing this book, the readers will be able to gain a good knowledge level and understanding of writing, configuring, and managing drivers, controlling and monitoring applications with the help of efficient/quick programming and will be able to apply these skills into real-world projects. Style and approach This practical tutorial will get you quickly prototyping embedded systems on GNU/Linux. This book uses a

variety of hardware to program the peripherals and build simple prototypes.

This book provides an introduction to Bluetooth programming, with a specific focus on developing real code. The authors discuss the major concepts and techniques involved in Bluetooth programming, with special emphasis on how they relate to other networking technologies. They provide specific descriptions and examples for creating applications in a number of programming languages and environments including Python, C, Java, GNU/Linux, Windows XP, Symbian Series 60, and Mac OS X. No previous experience with Bluetooth is assumed, and the material is suitable for anyone with some programming background. The authors place special emphasis on the essential concepts and techniques of Bluetooth programming, starting simply and allowing the reader to quickly master the basic concepts before addressing advanced features.

In the early days of computer networking IBM mainframes could only connect to other IBM mainframes, Burroughs only to other Burroughs, etc. Beginning in 1967 the US Defense Department's Advanced Research Projects Agency (ARPA) office sponsored development of a "heterogeneous" network compatible with computers from any manufacturer. That R&D effort, one of the most successful in history, resulted in the on-time, on-budget construction of the revolutionary ARPANET, the immediate predecessor of today's Internet. The ARPANET Sourcebook: The Unpublished Foundations of the Internet reproduces the seminal papers, reports, and RFCs that led to the birth of modern network computing. Most appear here in book form for the first time. Part A, Imagining the ARPANET, covers the initial studies of network feasibility and includes: the introductory and concluding chapters of Paul Baran's seminal but little-known RAND research report On Distributed Communications in which packet switching was first conceptualized. the classic 1968 paper The Computer as a Communication Device by J.C.R. Licklider and Robert Taylor, respectively the ARPANET's earliest proponent and the ARPA administrator who pushed the development project. Part B, Planning the ARPANET includes: scans of the earliest RFCs ("Requests for Comments"), some publicly available here for the first time. RFCs were in effect the design documents for the ARPANET and later the Internet. the 1968 ARPA-commissioned SRI study that modeled a heterogeneous network and concluded that it was indeed feasible. forewords by Steve Crocker (author of RFC #1) and Leonard Kleinrock (noted author and head of the UCLA computing lab that hosted the first ARPANET node). Part C, Building the ARPANET, reproduces the quarterly technical reports from the government's contractor Bolt Beranek and Newman contemporaneously describing the development group's progress, difficulties encountered, and final success. Dave Walden, former BBN VP and a key member of the ARPANET team, has contributed a retrospective Foreword. Other noteworthy material: historical perspectives from Peter Salus, Robert Taylor, Willis Ware, Michael Padlipsky, and Les Earnest, and a long-forgotten RFC which anticipated JAVA by more than 20 years.

In addition to covering a history of free and open source, The Daemon, the Gnu, and the Penguin explores how free and open software is changing the world. It is authored by Peter H. Salus, a noted UNIX, open source, and Internet historian and author of A Quarter Century of UNIX and Casting The Net and other books. Salus has interviewed well over a hundred key figures to document the history and background of free and open source software. In his book, Salus reaches back into the early days of computing, showing that even in "pre-UNIX" days there was freely available software, and rapidly moves forward to the Free Software movement of today and what it means for the future, drawing analogies

and linkages from various aspects of economics and life.

With Gradle, you can efficiently build automation framework along with some flexible alternatives to customized build logic. This book starts with sound basics about build automation and how Gradle fits into this automation. It then moves on to give you a good exposure on Groovy—a scripting language used to write Gradle—and helps you understand a key elements of Groovy programming language. In the following chapters, you will deal with task management and learn how to integrate Ant tasks into build scripts. Furthermore, you will learn dependency management, plugin management, and its configuration techniques in Gradle. You will also get hands-on with building and testing projects using Gradle. You will then begin to cover diverse topics, such as Continuous Integration with Jenkins and TeamCity, Migration strategies, and Deployment, which enables you to learn concepts useful for Agile software development. Finally, you will also learn how to create a simple mobile application using Android and explore how Gradle can help you to build and test the application.

Linux® is being adopted by an increasing number of embedded systems developers, who have been won over by its sophisticated scheduling and networking, its cost-free license, its open development model, and the support offered by rich and powerful programming tools. While there is a great deal of hype surrounding the use of Linux in embedded systems, there is not a lot of practical information. Building Embedded Linux Systems is the first in-depth, hard-core guide to putting together an embedded system based on the Linux kernel. This indispensable book features arcane and previously undocumented procedures for: Building your own GNU development toolchain Using an efficient embedded development framework Selecting, configuring, building, and installing a target-specific kernel Creating a complete target root filesystem Setting up, manipulating, and using solid-state storage devices Installing and configuring a bootloader for the target Cross-compiling a slew of utilities and packages Debugging your embedded system using a plethora of tools and techniques Details are provided for various target architectures and hardware configurations, including a thorough review of Linux's support for embedded hardware. All explanations rely on the use of open source and free software packages. By presenting how to build the operating system components from pristine sources and how to find more documentation or help, this book greatly simplifies the task of keeping complete control over one's embedded operating system, whether it be for technical or sound financial reasons. Author Karim Yaghmour, a well-known designer and speaker who is responsible for the Linux Trace Toolkit, starts by discussing the strengths and weaknesses of Linux as an embedded operating system. Licensing issues are included, followed by a discussion of the basics of building embedded Linux systems. The configuration, setup, and use of over forty different open source and free software packages commonly used in embedded Linux systems are also covered. uClibc, BusyBox, U-Boot, OpenSSH, tftpd, tftp, strace,

and gdb are among the packages discussed.

The Family Handyman's 100 Things Every Homeowner Must Know gives you the knowledge you need to be a better homeowner. You'll make informed decisions, avoid frustrations and save thousands over the life of your home. The Family Handyman's 100 Things Every Homeowner Must Know gives you the knowledge you need to be a better homeowner. You'll make informed decisions, avoid frustrations and save thousands over the life of your home. Here's just a small sample of what's inside: The smartest thing you can do before going on vacation Push a button and save \$100 Be ready for any natural disaster Make burglars bypass your house The most effective way to cut heating and cooling costs Clever shortcuts for home maintenance—indoors and out Keep mice, ants and other invaders out of your house Boost curb appeal without remodeling What to do when the power goes out The most common—and costly—homeowner mistakes Eliminate noises, odors and other household irritations Grow the best lawn on the block Secret weapons pros use for easy, instant repairs Make appliances last years longer The key to drip-free faucets Make your mower start right up—every time Cures for damp basements Exterminate mold and mildew Pick a paint color you'll love Get faster downloads and better TV quality Stop peeling paint Plus, incredibly easy repairs for appliances, plumbing, flooring, walls, furniture and more!

Freely available source code, with contributions from thousands of programmers around the world: this is the spirit of the software revolution known as Open Source. Open Source has grabbed the computer industry's attention. Netscape has opened the source code to Mozilla; IBM supports Apache; major database vendors have ported their products to Linux. As enterprises realize the power of the open-source development model, Open Source is becoming a viable mainstream alternative to commercial software. Now in Open Sources, leaders of Open Source come together for the first time to discuss the new vision of the software industry they have created. The essays in this volume offer insight into how the Open Source movement works, why it succeeds, and where it is going. For programmers who have labored on open-source projects, Open Sources is the new gospel: a powerful vision from the movement's spiritual leaders. For businesses integrating open-source software into their enterprise, Open Sources reveals the mysteries of how open development builds better software, and how businesses can leverage freely available software for a competitive business advantage. The contributors here have been the leaders in the open-source arena: Brian Behlendorf (Apache) Kirk McKusick (Berkeley Unix) Tim O'Reilly (Publisher, O'Reilly & Associates) Bruce Perens (Debian Project, Open Source Initiative) Tom Paquin and Jim Hamerly (mozilla.org, Netscape) Eric Raymond (Open Source Initiative) Richard Stallman (GNU, Free Software Foundation, Emacs) Michael Tiemann (Cygnus Solutions) Linus Torvalds (Linux) Paul Vixie (Bind) Larry Wall (Perl) This book explains why the majority of the Internet's servers use open-source technologies

for everything from the operating system to Web serving and email. Key technology products developed with open-source software have overtaken and surpassed the commercial efforts of billion dollar companies like Microsoft and IBM to dominate software markets. Learn the inside story of what led Netscape to decide to release its source code using the open-source mode. Learn how Cygnus Solutions builds the world's best compilers by sharing the source code. Learn why venture capitalists are eagerly watching Red Hat Software, a company that gives its key product -- Linux -- away. For the first time in print, this book presents the story of the open-source phenomenon told by the people who created this movement. Open Sources will bring you into the world of free software and show you the revolution.

The bash shell is a complete programming language, not merely a glue to combine external Linux commands. By taking full advantage of shell internals, shell programs can perform as snappily as utilities written in C or other compiled languages. And you will see how, without assuming Unix lore, you can write professional bash 4.0 programs through standard programming techniques. Complete bash coverage Teaches bash as a programming language Helps you master bash 4.0 features

The Daemon, the Gnu, and the Penguin

This introduction to networking on Linux now covers firewalls, including the use of ipchains and Netfilter, masquerading, and accounting. Other new topics in this second edition include Novell (NCP/IPX) support and INN (news administration). Provides information on writing a driver in Linux, covering such topics as character devices, network interfaces, driver debugging, concurrency, and interrupts.

Casanova, Stendhal, Tolstoy: Adepts in Self-Portraiture, the final volume of Stefan Zweig's masterful Master Builders of the Spirit trilogy, discloses the smaller version of a writer's own ego. Unconscious though it is, no reality is as important to the writer as the reality of their own life. Giacomo Casanova, Stendhal (Marie-Henri Beyle), and Leo Tolstoy have different approaches to self-portraiture, but Zweig shows that together they symbolize three levels which represent successively ascending gradations of the same creative function. Casanova is depicted as having a primitive gradation; he simply records deeds and happenings, without any attempt to appraise them or to study the deeper working of the self. Stendhal's self-portraiture is depicted as psychological; he observes himself and investigates his own feelings. Tolstoy has the highest level; he describes his own life, records what led him to his own actions, and focuses on self-reflection in a completely unexaggerated manner. At first glance it might seem as if self-portraiture is an artist's easiest task. With no further trouble than a probing of memory and a description of the facts of life, "the truth" is revealed. The history of literature shows that ordinary autobiographers are no more than commonplace witnesses testifying to facts that chance has brought to their knowledge. A practiced artist is needed to discern the innermost happenings of the soul; few

who have attempted autobiography have been successful in this difficult task. The present volume expounds the characteristics of these subjectively minded artists, and of autobiography as their typical method of personal expression. In 1948, a group of conservative white southerners formed the States' Rights Democratic Party, soon nicknamed the "Dixiecrats," and chose Strom Thurmond as their presidential candidate. Thrown on the defensive by federal civil rights initiatives and unprecedented grassroots political activity by African Americans, the Dixiecrats aimed to reclaim conservatives' former preeminent position within the national Democratic Party and upset President Harry Truman's bid for reelection. The Dixiecrats lost the battle in 1948, but, as Kari Frederickson reveals, the political repercussions of their revolt were significant. Frederickson situates the Dixiecrat movement within the tumultuous social and economic milieu of the 1930s and 1940s South, tracing the struggles between conservative and liberal Democrats over the future direction of the region. Enriching her sweeping political narrative with detailed coverage of local activity in Alabama, Louisiana, Mississippi, and South Carolina--the flashpoints of the Dixiecrat campaign--she shows that, even without upsetting Truman in 1948, the Dixiecrats forever altered politics in the South. By severing the traditional southern allegiance to the national Democratic Party in presidential elections, the Dixiecrats helped forge the way for the rise of the Republican Party in the region.

The authors meet the growing demands of de-centralized companies that need a secure and functional network using Linux. The only book available that extensively covers the combination of VPN technology and Linux, this volume teaches first hand how to build various VPN solutions with individual setup guides.

It's a plain fact: regardless of how smart, creative, and innovative your organization is, there are more smart, creative, and innovative people outside your organization than inside. Open source offers the possibility of bringing more innovation into your business by building a creative community that reaches beyond the barriers of the business. The key is developing a web-driven community where new types of collaboration and creativity can flourish. Since 1998 Ron Goldman and Richard Gabriel have been helping groups at Sun Microsystems understand open source and advising them on how to build successful communities around open source projects. In this book the authors present lessons learned from their own experiences with open source, as well as those from other well-known projects such as Linux, Apache, and Mozilla. * Winner of 2006 Jolt Productivity Award for General Books * Describes how open source development works and offers persuasive reasons for using it to help achieve business goals. * Shows how to use open source in day-to-day work, discusses the various licenses in use, and describes what makes for a successful project. * Written in an engaging style for executives, managers, and engineers that addresses the human and business issues involved in open source development as well as its history, philosophy, and future

CD-ROM contains: all source code and datafiles from the book.

The open source saga has many fascinating chapters. It is partly the story of Linus Torvalds, the master hacker who would become chief architect of the Linux operating system. It is also the story of thousands of devoted programmers around the world who spontaneously worked in tandem to complete the race to shape Linux into the ultimate killer app. Rebel Code traces the remarkable roots of this unplanned revolution. It echoes the twists and turns of Linux's improbable development, as it grew through an almost biological process of accretion and finally took its place at the heart of a jigsaw puzzle that would become the centerpiece of open source. With unprecedented access to the principal players, Moody has written a powerful tale of individual innovation versus big business. Rebel Code provides a from-the-trenches perspective and looks ahead to how open source is challenging long-held conceptions of technology, commerce, and culture.

Computers are an advancement whose importance is comparable to the invention of the wheel or movable type. While computers and the Internet have already changed many aspects of our lives, we still live in the dark ages of computing because proprietary software is still the dominant model. One might say that the richest alchemist who ever lived is my former boss, Bill Gates. (Oracle founder Larry Ellison, and Google co-founders Sergey Brin and Larry Page are close behind.) Human knowledge increasingly exists in digital form, so building new and better models requires the software to be improved. People can only share ideas when they also share the software to display and modify them. It is the expanded use of free software that will allow a greater ability for people to work together and increase the pace of progress. This book will demonstrate that a system where anyone can edit, share, and review the body of work will lead not just to something that works, but eventually to the best that the world can achieve! With better cooperation among our scientists, robot-driven cars is just one of the many inventions that will arrive -- pervasive robotics, artificial intelligence, and much faster progress in biology, all of which rely heavily on software. - Publisher.

You've experienced the shiny, point-and-click surface of your Linux computer—now dive below and explore its depths with the power of the command line. The Linux Command Line takes you from your very first terminal keystrokes to writing full programs in Bash, the most popular Linux shell. Along the way you'll learn the timeless skills handed down by generations of gray-bearded, mouse-shunning gurus: file navigation, environment configuration, command chaining, pattern matching with regular expressions, and more. In addition to that practical knowledge, author William Shotts reveals the philosophy behind these tools and the rich heritage that your desktop Linux machine has inherited from Unix supercomputers of yore. As you make your way through the book's short, easily-digestible chapters, you'll learn how to: *

- Create and delete files, directories, and symlinks
- * Administer your system, including networking, package installation, and process management
- * Use standard input and output, redirection, and pipelines
- * Edit files with Vi, the world's most popular text editor
- * Write shell scripts to automate common or boring tasks
- * Slice and dice text files with cut, paste, grep, patch, and sed

Once you overcome your initial

Download Ebook The Daemon The Gnu And The Penguin

"shell shock," you'll find that the command line is a natural and expressive way to communicate with your computer. Just don't be surprised if your mouse starts to gather dust. A featured resource in the Linux Foundation's "Evolution of a SysAdmin"

This updated reference offers a clear description of make, a central engine in many programming projects that simplifies the process of re-linking a program after re-compiling source files. Original. (Intermediate)

Based on interviews with the key software engineers who invented and built the powerful UNIX operating system, this book provides unique insight into the operating system that dominates the modern computing environment. Originating from a small project in a backroom at AT & T Bell Labs, UNIX has grown to be a dominant operating system in the commercial computing world -the operating system responsible for the development of the C programming language and the modern networked environment. Peter Salus is a longtime and well-recognized promoter and spokesman for UNIX and the UNIX community.

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.

For the past 20 years, UNIX insiders have cherished and zealously guarded pirated photocopies of this manuscript, a "hacker trophy" of sorts. Now legal (and legible) copies are available. An international "who's who" of UNIX wizards, including Dennis Ritchie, have contributed essays extolling the merits and importance of this underground classic.

With a comprehensive overview of Linux hardware and architecture, this book provides the latest information on the variety of Linux certification exams, covers crucial topics found on the exams, and previews upcoming certifications. Lessons feature real-world examples, interactive activities, and more than 100 hands-on projects that reinforce key concepts.

By its very nature, Unix is a "power tools" environment. Even beginning Unix users quickly grasp that immense power exists in shell programming, aliases and history mechanisms, and various editing tools. Nonetheless, few users ever really master the power available to them with Unix. There is just too much to learn! Unix Power Tools, Third Edition, literally contains thousands of tips, scripts, and techniques that make using Unix easier, more effective, and even more fun. This book is organized into hundreds of short articles with plenty of references to other sections that keep you flipping from new article to new article. You'll find the book hard to put down as you uncover one interesting tip after another. With the growing popularity of Linux and the advent of Mac OS X, Unix has metamorphosed into something new and exciting. With Unix no longer perceived as a difficult operating system, more and more users are discovering its advantages for the first time. The latest edition of this best-selling favorite is loaded with advice about almost every aspect of Unix, covering all the new technologies that users need to know. In addition to vital information on Linux, Mac OS X, and BSD, Unix Power Tools, Third Edition, now offers more

Download Ebook The Daemon The Gnu And The Penguin

coverage of bcash, zsh, and new shells, along with discussions about modern utilities and applications. Several sections focus on security and Internet access, and there is a new chapter on access to Unix from Windows, addressing the heterogeneous nature of systems today. You'll also find expanded coverage of software installation and packaging, as well as basic information on Perl and Python. The book's accompanying web site provides some of the best software available to Unix users, which you can download and add to your own set of power tools. Whether you are a newcomer or a Unix power user, you'll find yourself thumbing through the gold mine of information in this new edition of Unix Power Tools to add to your store of knowledge. Want to try something new? Check this book first, and you're sure to find a tip or trick that will prevent you from learning things the hard way.

[Copyright: 61b42fe4c6de1f3024e13361e47e74e7](#)