Unix For Programmers And Users 3rd Edition

Shell Programming in Unix, Linux and OS XUNIX for the MS-DOS UserMlti PackUNIX Shell ProgrammingUNIX System SecurityLinux for DevelopersUNIX for Programmers and UsersThe Art of UNIX ProgrammingUNIX for OpenVMS UsersLearning the Korn ShellUNIX Systems Programming for SVR4Learning the Bash ShellLinux and the Unix PhilosophyThe UNIX Programming EnvironmentPython for Unix and Linux System AdministrationA Practical Guide to UNIX for Mac OS X UsersLinux for Programmers and UsersSed & AwkMastering Unix Shell ScriptingLearning the Unix Operating SystemAdvanced Linux ProgrammingUNIX for Programmers and UsersAdvanced UNIX ProgrammingUNIX Systems ProgrammingSams Teach Yourself Shell Programming in 24 HoursUNIX for Programmers and UsersLinux System ProgrammingThe C Programming LanguageThe Berkeley UNIX EnvironmentUnix in a NutshellUnderstanding Unix/Linux ProgrammingUnix ProgrammingLinux and UNIX Shell ProgrammingUnix Awk and Sed Programmer's Interactive WorkbookUnderstanding the Linux KernelUNIX System V Network ProgrammingThe Linux Programmer's ToolboxUNIX Power ToolsThe UNIX-haters HandbookThe Art of UNIX Programming

Shell Programming in Unix, Linux and OS X

Linux for Developers shows you how to start writing great code for Linux, whether you’re a Linux user with little or no coding experience, or an experienced Windows programmer. Leading IT trainer/author William “Bo” Rothwell begins with a clear and up-to-date review of modern open source software, including the licensing arrangements and tradeoffs all developers need to understand. He presents essential skills for both Linux command line and GUI environments, introducing text editors and other tools for efficient coding. Building on this knowledge, Rothwell introduces scripting tools such as Bash, Python, and Perl, as well as traditional object-oriented programming languages such as Java, C++, and C. Finally, he presents a full section on the powerful Git version control system, teaching skills you can use in Linux and many other environments. Access Linux systems, use GUIs, and work at the command line Learn how Linux organizes files and navigate its filesystem Use basic developer commands such as gzip and grep Edit programs with vi and vim, and explore alternative editors Perform basic sysadmin tasks that developers often need to handle Compare Linux languages to choose the best one for each task Write Bash scripts that interact with users or other shell features Program with Python and Perl: flow control, variables, and more Understand Linux features related to building C, C++, and Java programs Stay on top of complex projects with GIT revision control Work in GIT: staging, committing, branches, diffs, merges, and patches Manage local and remote GIT repositories This guide’s modular coverage helps you quickly access whatever information you need right now.

UNIX for the MS-DOS User

To thoroughly understand what makes Linux tick and why it's so efficient, you need to delve deep into the heart of the operating system--into the Linux kernel itself. The kernel is Linux--in the case of the Linux operating system, it's the only bit of

software to which the term "Linux" applies. The kernel handles all the requests or completed I/O operations and determines which programs will share its processing time, and in what order. Responsible for the sophisticated memory management of the whole system, the Linux kernel is the force behind the legendary Linux efficiency. The new edition of Understanding the Linux Kernel takes you on a guided tour through the most significant data structures, many algorithms, and programming tricks used in the kernel. Probing beyond the superficial features, the authors offer valuable insights to people who want to know how things really work inside their machine. Relevant segments of code are dissected and discussed line by line. The book covers more than just the functioning of the code, it explains the theoretical underpinnings for why Linux does things the way it does. The new edition of the book has been updated to cover version 2.4 of the kernel, which is quite different from version 2.2: the virtual memory system is entirely new, support for multiprocessor systems is improved, and whole new classes of hardware devices have been added. The authors explore each new feature in detail. Other topics in the book include: Memory management including file buffering, process swapping, and Direct memory Access (DMA) The Virtual Filesystem and the Second Extended Filesystem Process creation and scheduling Signals, interrupts, and the essential interfaces to device drivers Timing Synchronization in the kernel Interprocess Communication (IPC) Program execution Understanding the Linux Kernel, Second Edition will acquaint you with all the inner workings of Linux, but is more than just an academic exercise. You'll learn what conditions bring out Linux's best performance, and you'll see how it meets the challenge of providing good system response during process scheduling, file access, and memory management in a wide variety of environments. If knowledge is power, then this book will help you make the most of your Linux system.

Mlti Pack

Learn how to develop powerful and robust shell scripts in order to get the most out of your Unix/Linux system.

UNIX Shell Programming

O'Reilly's bestselling book on Linux's bash shell is at it again. Now that Linux is an established player both as a server and on the desktop Learning the bash Shell has been updated and refreshed to account for all the latest changes. Indeed, this third edition serves as the most valuable guide yet to the bash shell. As any good programmer knows, the first thing users of the Linux operating system come face to face with is the shell the UNIX term for a user interface to the system. In other words, it's what lets you communicate with the computer via the keyboard and display. Mastering the bash shell might sound fairly simple but it isn't. In truth, there are many complexities that need careful explanation, which is just what Learning the bash Shell provides. If you are new to shell programming, the book provides an excellent introduction, covering everything from the most basic to the most advanced features. And if you've been writing shell scripts for years, it offers a great way to find out what the new shell offers. Learning the bash Shell is also full of practical examples of shell commands and programs that will make everyday use of Linux that much easier. With this book, programmers will learn: How to install bash as your login shell The basics of interactive shell use, including

UNIX file and directory structures, standard I/O, and background jobs Command line editing, history substitution, and key bindings How to customize your shell environment without programming The nuts and bolts of basic shell programming, flow control structures, command-line options and typed variables Process handling, from job control to processes, coroutines and subshells Debugging techniques, such as trace and verbose modes Techniques for implementing system-wide shell customization and features related to system security

UNIX System Security

This is the eBook of the printed book and may not include any media, website access codes, or print supplements that may come packaged with the bound book. For an introductory course on UNIX. UNIX for Programmers and Users, Third Edition follows in the tradition of previous editions to provide students with complete, up-to-date coverage of UNIX. In this new edition they will find information on basic concepts, popular utilities, shells, networking, systems programming, internals, system administration, and much more.

Linux for Developers

Explains how to develop programs in the UNIX operating system, discussing how to perform tasks including building, debugging, and understanding how shell scripts work.

UNIX for Programmers and Users

Unlike so many books that focus on how to use Linux, Linux and the Unix Philosophy explores the "way of thinking that is Linux" and why Linux is a superior implementation of this highly capable operating system. This book is a revision and expansion of a computer science classic. Every chapter has been thoroughly updated with Linux coverage. Linux and the Unix Philosophy falls squarely between the "softer" texts on iterative software design and project management and the "how-to" technical texts. Thus far, no one has come out with a book that addresses this topic, either in the Unix space or the Linux space. Linux and the Unix Philosophy covers the same ground as the first edition, while it also presents bold new ideas about Linux and Open Source. · Concise list of philosophy tenets makes it a handy quick reference · Anecdotal examples personalize the book for the reader · Conversational style makes it easy and joyful to read

The Art of UNIX Programming

Provides the nitty gritty details on how UNIX interacts with applications. Inlcudes many extended examples on topics ranging from string manipulation to network programming

UNIX for OpenVMS Users

Multi pack contains: 0130465534 - UNIX for Programmers and Users 0131103628 - C Programming Language

Learning the Korn Shell

UNIX for OpenVMS Users, 3E, makes it easy to see what OpenVMS and UNIX have in common, and to transfer your knowledge and experience in OpenVMS over to the world of UNIX. Since most shops rely on more than one operating system, it is critical for system administrators and managers to understand the similarities and differences between platforms, so they can easily work in both environments while taking full advantage of the tools and applications available on each. This book offers OpenVMS professionals a concise source of information, so that they can quickly bring their expertise to bear on UNIX file management, e-mail, networking, and security. This new edition of the book is enhanced with updated references to VMS, incorporates suggestions made by readers of previous editions, and particularly, recognizes other UNIX implementations in addition to HP's Tru64. Includes extensive additions to the sections on VMS logical names, on the emacs editor, and on shell programming and Perl Describes the interfaces common to both operating systems, with appendices covering command and editor summaries Adds emphasis on Linux

UNIX Systems Programming for SVR4

This book is a discussion about the Unix operating system. The first part of the book helps you get started with Unix by understanding how it works and the various parts of the Unix operating system. Unix files are also discussed in detail, and you will learn how to use the various commands to interact with files and data in a Unix system. In Unix, files are associated with permissions which determine the actions which one can perform on a particular file. This book discusses such permissions and guides you on how to add and remove the various permissions for various users from a Unix file or directory. The basic utilities provided by Unix are also explored, and you are guided on how to use them. Examples of such utilities include the print and mail utilities, which are very useful to a Unix user. The concepts of piping and filtering are examined in this book. You will learn how to combine various Unix commands into a single command by the use of pipes. Communication over the network is very essential in Unix. This book helps you to understand the various commands you can use to communicate with other Unix hosts via a network. You will also learn how to work with signals and traps in Unix. User management in Unix is also explained. The following topics are discussed in this book: - Getting Started with UNIX - Unix Files - File Permissions - Basic Unix Utilities - Pipes and Filters - Network Communication - Signals and Traps - User Administration in Unix

Learning the Bash Shell

Linux and the Unix Philosophy

Master the Linux Tools That Will Make You a More Productive, Effective Programmer The Linux Programmer's Toolbox helps you tap into the vast collection of open source tools available for GNU/Linux. Author John Fusco systematically describes the most useful tools available on most GNU/Linux distributions using

concise examples that you can easily modify to meet your needs. You'll start by learning the basics of downloading, building, and installing open source projects. You'll then learn how open source tools are distributed, and what to look for to avoid wasting time on projects that aren't ready for you. Next, you'll learn the ins and outs of building your own projects. Fusco also demonstrates what to look for in a text editor, and may even show you a few new tricks in your favorite text editor. You'll enhance your knowledge of the Linux kernel by learning how it interacts with your software. Fusco walks you through the fundamentals of the Linux kernel with simple, thought-provoking examples that illustrate the principles behind the operating system. Then he shows you how to put this knowledge to use with more advanced tools. He focuses on how to interpret output from tools like sar, vmstat, valgrind, strace, and apply it to your application; how to take advantage of various programming APIs to develop your own tools; and how to write code that monitors itself. Next, Fusco covers tools that help you enhance the performance of your software. He explains the principles behind today's multicore CPUs and demonstrates how to squeeze the most performance from these systems. Finally, you'll learn tools and techniques to debug your code under any circumstances. Coverage includes Maximizing productivity with editors, revision control tools, source code browsers, and "beautifiers" Interpreting the kernel: what your tools are telling you Understanding processes–and the tools available for managing them Tracing and resolving application bottlenecks with gprof and valgrind Streamlining and automating the documentation process Rapidly finding help, solutions, and workarounds when you need them Optimizing program code with sar, vmstat, iostat, and other tools Debugging IPC with shell commands: signals, pipes, sockets, files, and IPC objects Using printf, gdb, and other essential debugging tools Foreword Preface Acknowledgments About the Author Chapter 1 Downloading and Installing Open Source Tools Chapter 2 Building from Source Chapter 3 Finding Help Chapter 4 Editing and Maintaining Source Files Chapter 5 What Every Developer Should Know about the Kernel Chapter 6 Understanding Processes Chapter 7 Communication between Processes Chapter 8 Debugging IPC with Shell Commands Chapter 9 Performance Tuning Chapter 10 Debugging Index

The UNIX Programming Environment

Don't miss this guide to building networked and distributed applications for UNIX® System V. Using many helpful examples, the author provides a solid introduction to networking and UNIX programming, plus information on the programming interfaces most important to building communication software in System V, such as STREAMS, the Transport Layer Interface library, Sockets, and Remote Procedure Calls. The book also explains how to write kernel-level communication software, including STREAMS drivers, modules, and multiplexors. A final chapter on SLIP is essential reading.

Python for Unix and Linux System Administration

Appropriate for an introductory course on UNIX. This new edition provides complete up-to-date coverage of UNIX, including basic concepts, popular utilities, shells, networking, systems programming, internals, and system administration.

A Practical Guide to UNIX for Mac OS X Users

Shell Programming in Unix, Linux and OS X is a thoroughly updated revision of Kochan and Wood's classic Unix Shell Programming tutorial. Following the methodology of the original text, the book focuses on the POSIX standard shell, and teaches you how to develop programs in this useful programming environment, taking full advantage of the underlying power of Unix and Unix-like operating systems. After a quick review of Unix utilities, the book's authors take you step-by-step through the process of building shell scripts, debugging them, and understanding how they work within the shell's environment. All major features of the shell are covered, and the large number of practical examples make it easy for you to build shell scripts for your particular applications. The book also describes the major features of the Korn and Bash shells. Learn how to… Take advantage of the many utilities provided in the Unix system Write powerful shell scripts Use the shell's built-in decision-making and looping constructs Use the shell's powerful quoting mechanisms Make the most of the shell's built-in history and command editing capabilities Use regular expressions with Unix commands Take advantage of the special features of the Korn and Bash shells Identify the major differences between versions of the shell language Customize the way your Unix system responds to you Set up your shell environment Make use of functions Debug scripts Contents at a Glance 1 A Quick Review of the Basics 2 What Is the Shell? 3 Tools of the Trade 4 And Away We Go 5 Can I Quote You on That? 6 Passing Arguments 7 Decisions, Decisions 8 'Round and 'Round She Goes 9 Reading and Printing Data 10 Your Environment 11 More on Parameters 12 Loose Ends 13 Rolo Revisited 14 Interactive and Nonstandard Shell Features A Shell Summary B For More Information

Linux for Programmers and Users

This volume is designed to help MS-DOS programmers become rapidly proficient in the UNIX environment. It focuses on the similarities and differences between the two operating systems, enabling programmers to perform all the operations they did in MS-DOS plus those available only on UNIX systems. First considers the operations that most MS-DOS users perform and the user interface to the operating system (the Shell); then explains the features unique to UNIX--multi-user, multi-tasking; and examines in detail the UNIX shell script files (Bourne shell, Korn shell, C shell)--which are comparable to MS-DOS batch files--showing how they produce the same result, but whose constructs are different. Concludes with an examination of the administration features of UNIX, and its text processing utilities. For MS-DOS users who want to become rapidly proficient in UNIX systems.

Sed & Awk

Because the UNIX system was originally designed by programmers for use by other programmers, it was used in an environment of open cooperation where security was of minimal concern. Now that its use has spread to universities, businesses, and government, the confidential and sensitive nature of the data stored on UNIX systems has made the security of these systems of paramount importance. Despite all the technical papers and workshops on UNIX security, this book is

unique. "UNIX System Security" is the first up-to-date source to provide the UNIX system user or administrator with the information needed to protect the data and system from unauthorized use. By following the procedures described in this book and making use of the C programs and shell scripts provided as examples, you can protect your UNIX system from most attackers. The author begins by examining four high-profile breaches of UNIX security as illustrations of how a UNIX system can be attacked. He then provides the information necessary to protect against these forms of attack, and offers the tools that can be used to do so. Focusing on the most recent release of Berkeley and System V UNIX, and such vendor derivatives as SunOS and ULTRIX, the book gives information that can be applied to any version of UNIX since Seventh Edition. Issues discussed include account and password security, securing the file system, encryption and authentication systems, TCP/IP network security, the Network Information Service (NIS), NFS, RFS, workstation security, terminals and modems, and UUCP. Other chapters describe how to respond if your system is attacked and how to develop a comprehensive security policy for your organization. The book also gives comprehensive lists of freely available security software, and publications and mailing lists dealing with UNIX security.

Mastering Unix Shell Scripting

Advanced UNIX Programming goes beyond the fundamentals of UNIX programming and presents information and techniques the readers needs to expand their knowledge base. Designed for professional UNIX programmers, this book builds on the skills and knowledge the reader already possesses. It includes coverage of internet processes, interprocess control, file system manipulation, synchronization, and much more.

Learning the Unix Operating System

Advanced Linux Programming

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.

UNIX for Programmers and Users

Offering full coverage of Linux in one source, this book documents the most

commonly needed topics for new and experienced Linux users and programmers - including over 100 utilities and their common options. Provides a good foundation of understanding for the most often-used Linux utilities. Devotes a chapter to helpful installation information for those who must install their own systems. Includes hundreds of command and code examples throughout. Provides approximately 50 diagrams throughout. Features FTP-able files; code used in the book will be made available on a website hosted by the publisher. A useful reference for anyone using a Linux platform, including programmers, system administrators, and any user who must understand the operating system outside of a specific application.

Advanced UNIX Programming

The Most Useful UNIX Guide for Mac OS X Users Ever, with Hundreds of High-Quality Examples! Beneath Mac OS® X's stunning graphical user interface (GUI) is the most powerful operating system ever created: UNIX®. With unmatched clarity and insight, this book explains UNIX for the Mac OS X user–giving you total control over your system, so you can get more done, faster. Building on Mark Sobell's highly praised A Practical Guide to the UNIX System, it delivers comprehensive guidance on the UNIX command line tools every user, administrator, and developer needs to master—together with the world's best day-to-day UNIX reference. This book is packed with hundreds of high-quality examples. From networking and system utilities to shells and programming, this is UNIX from the ground up–both the "whys" and the "hows"–for every Mac user. You'll understand the relationships between GUI tools and their command line counterparts. Need instant answers? Don't bother with confusing online "manual pages": rely on this book's example-rich, quick-access, 236-page command reference! Don't settle for just any UNIX guidebook. Get one focused on your specific needs as a Mac user! A Practical Guide to UNIX® for Mac OS® X Users is the most useful, comprehensive UNIX tutorial and reference for Mac OS X and is the only book that delivers Better, more realistic examples covering tasks you'll actually need to perform Deeper insight, based on the authors' immense knowledge of every UNIX and OS X nook and cranny Practical guidance for experienced UNIX users moving to Mac OS X Exclusive discussions of Mac-only utilities, including plutil, ditto, nidump, otool, launchctl, diskutil, GetFileInfo, and SetFile Techniques for implementing secure communications with ssh and scp–plus dozens of tips for making your OS X system more secure Expert guidance on basic and advanced shell programming with bash and tcsh Tips and tricks for using the shell interactively from the command line Thorough guides to vi and emacs designed to help you get productive fast, and maximize your editing efficiency In-depth coverage of the Mac OS X filesystem and access permissions, including extended attributes and Access Control Lists (ACLs) A comprehensive UNIX glossary Dozens of exercises to help you practice and gain confidence And much more, including a superior introduction to UNIX programming tools such as awk, sed, otool, make, gcc, gdb, and CVS

UNIX Systems Programming

The Art of UNIX Programming poses the belief that understanding the unwritten UNIX engineering tradition and mastering its design patterns will help programmers of all stripes to become better programmers. This book attempts to

capture the engineering wisdom and design philosophy of the UNIX, Linux, and Open Source software development community as it has evolved over the past three decades, and as it is applied today by the most experienced programmers. Eric Raymond offers the next generation of "hackers" the unique opportunity to learn the connection between UNIX philosophy and practice through careful case studies of the very best UNIX/Linux programs.

Sams Teach Yourself Shell Programming in 24 Hours

UNIX expert Randal K. Michael guides you through every detail of writing shell scripts to automate specific tasks. Each chapter begins with a typical, everyday UNIX challenge, then shows you how to take basic syntax and turn it into a shell scripting solution. Covering Bash, Bourne, and Korn shell scripting, this updated edition provides complete shell scripts plus detailed descriptions of each part. UNIX programmers and system administrators can tailor these to build tools that monitor for specific system events and situations, building solid UNIX shell scripting skills to solve real-world system administration problems.

UNIX for Programmers and Users

This Nutshell Handbook® is a thorough introduction to the Korn shell, both as a user interface and as a programming language.The Korn shell, like the C and Bourne shells, is a program that interprets UNIX commands. It has many features that aren't found in other shells, including command history (the ability to recall and edit previous commands). The Korn shell is also faster; several of its features allow you to write programs that execute more quickly than their Bourne or C shell equivalents.This book provides a clear and concise explanation of the Korn shell's features. It explains ksh string operations, co-processes, signals and signal handling, and one of the worst "dark corners" of shell programming: command-line interpretation. It does this by introducing simple real-life examples and then adding options and complexity in later chapters, illustrating the way real-world script development generally proceeds. An additional (and unique) programming aid, a Korn shell debugger (kshdb), is also included.Learning the Korn Shell is an ideal resource for many UNIX users and programmers, including software developers who want to "prototype" their designs, system administrators who want to write tools for their own use, and even novices who just want to use some of ksh's more advanced interactive features.

Linux System Programming

The Art of UNIX Programming poses the belief that understanding the unwritten UNIX engineering tradition and mastering its design patterns will help programmers of all stripes to become better programmers. This book attempts to capture the engineering wisdom and design philosophy of the UNIX, Linux, and Open Source software development community as it has evolved over the past three decades, and as it is applied today by the most experienced programmers. Eric Raymond offers the next generation of "hackers" the unique opportunity to learn the connection between UNIX philosophy and practice through careful case studies of the very best UNIX/Linux programs.

The C Programming Language

As an open operating system, Unix can be improved on by anyone and everyone: individuals, companies, universities, and more. As a result, the very nature of Unix has been altered over the years by numerous extensions formulated in an assortment of versions. Today, Unix encompasses everything from Sun's Solaris to Apple's Mac OS X and more varieties of Linux than you can easily name. The latest edition of this bestselling reference brings Unix into the 21st century. It's been reworked to keep current with the broader state of Unix in today's world and highlight the strengths of this operating system in all its various flavors. Detailing all Unix commands and options, the informative guide provides generous descriptions and examples that put those commands in context. Here are some of the new features you'll find in Unix in a Nutshell, Fourth Edition: Solaris 10, the latest version of the SVR4-based operating system, GNU/Linux, and Mac OS X Bash shell (along with the 1988 and 1993 versions of ksh) tsch shell (instead of the original Berkeley csh) Package management programs, used for program installation on popular GNU/Linux systems, Solaris and Mac OS X GNU Emacs Version 21 Introduction to source code management systems Concurrent versions system Subversion version control system GDB debugger As Unix has progressed, certain commands that were once critical have fallen into disuse. To that end, the book has also dropped material that is no longer relevant, keeping it taut and current. If you're a Unix user or programmer, you'll recognize the value of this complete, up-to-date Unix reference. With chapter overviews, specific examples, and detailed command.

The Berkeley UNIX Environment

Unix in a Nutshell

Learn how to create and develop shell scripts in a step-by-step manner increasing your knowledge as you progress through the book. Learn how to work the shell commands so you can be more productive and save you time.

Understanding Unix/Linux Programming

Unix Programming

UNIX, UNIX LINUX & UNIX TCL/TK. Write software that makes the most effective use of the Linux system, including the kernel and core system libraries. The majority of both Unix and Linux code is still written at the system level, and this book helps you focus on everything above the kernel, where applications such as Apache, bash, cp, vim, Emacs, gcc, gdb, glibc, ls, mv, and X exist. Written primarily for engineers looking to program at the low level, this updated edition of Linux System Programming gives you an understanding of core internals that makes for better code, no matter where it appears in the stack. -- Provided by publisher.

Linux and UNIX Shell Programming

bull; Learn UNIX essentials with a concentration on communication, concurrency, and multithreading techniques bull; Full of ideas on how to design and implement good software along with unique projects throughout bull; Excellent companion to Stevens' Advanced UNIX System Programming

Unix Awk and Sed Programmer's Interactive Workbook

Explains the progression in Unix from grep to sed and awk, describes how to write sed scripts, covers common programming constructs, and details awk's built-in functions

Understanding the Linux Kernel

Getting started with Unix; C programming oveview; Using the vi editor; The C shel, csh; Networking programs; Compiler rools - LEX; Compiler tools - YACC; Library functions for input - output; Additional library functions; Processes and signals; Terminal and window handling; Communicating between processes; Developing large C programs; Project management and version control; Debugging & profiling C code; The emacs editor; Converting ansi C to K&R C; Index; Function index.

UNIX System V Network Programming

A handy book for someone just starting with Unix or Linux, and an ideal primer for Mac and PC users of the Internet who need to know a little about Unix on the systems they visit. The most effective introduction to Unix in print, covering Internet usage for email, file transfers, web browsing, and many major and minor updates to help the reader navigate the ever-expanding capabilities of the operating system.

The Linux Programmer's Toolbox

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.

UNIX Power Tools

An accessible, yet comprehensive text that clearly explains Unix programming and structuring by addressing the fundamentals of Unix and providing alternative solutions to problems in concrete terms.

The UNIX-haters Handbook

Python is an ideal language for solving problems, especially in Linux and Unix networks. With this pragmatic book, administrators can review various tasks that often occur in the management of these systems, and learn how Python can provide a more efficient and less painful way to handle them. Each chapter in

Python for Unix and Linux System Administration presents a particular administrative issue, such as concurrency or data backup, and presents Python solutions through hands-on examples. Once you finish this book, you'll be able to develop your own set of command-line utilities with Python to tackle a wide range of problems. Discover how this language can help you: Read text files and extract information Run tasks concurrently using the threading and forking options Get information from one process to another using network facilities Create clickable GUIs to handle large and complex utilities Monitor large clusters of machines by interacting with SNMP programmatically Master the IPython Interactive Python shell to replace or augment Bash, Korn, or Z-Shell Integrate Cloud Computing into your infrastructure, and learn to write a Google App Engine Application Solve unique data backup challenges with customized scripts Interact with MySQL, SQLite, Oracle, Postgres, Django ORM, and SQLAlchemy With this book, you'll learn how to package and deploy your Python applications and libraries, and write code that runs equally well on multiple Unix platforms. You'll also learn about several Python-related technologies that will make your life much easier.

The Art of UNIX Programming

PLEASE PROVIDE COURSE INFORMATION PLEASE PROVIDE