Unix Network Programming Vol 1 Networking Apis Sockets And Xti

??????????????????????????, ??, ?????????????????????Linux??????????Linux?????, ?????Linux?????????.

“As this book shows, Linux systems are just as functional, secure, and reliable as their proprietary counterparts. Thanks to the ongoing efforts of thousands of Linux developers, Linux is more ready than ever for deployment at the frontlines of the real world. The authors of this book know that terrain well, and I am happy to leave you in their most capable hands.” –Linus Torvalds “The most successful sysadmin book of all time–because it works!” –Rik Farrow, editor of ;login: “This book clearly explains current technology with the perspective of decades of experience in large-scale system administration. Unique and highly recommended.” –Jonathan Corbet, cofounder, LWN.net “Nemeth et al. is the overall winner for Linux administration: it’s intelligent, full of insights, and looks at the implementation of concepts.” –Peter Salus, editorial director, Matrix.net Since 2001, Linux Administration Handbook has been the definitive resource for every Linux® system administrator who must efficiently solve technical problems and maximize the reliability and performance of a production environment. Now, the authors have systematically updated this classic guide to address today’s most important Linux distributions and most powerful new administrative tools. The authors spell out detailed best practices for every facet of system administration, including storage management, network design and administration, web hosting, software configuration management, performance analysis, Windows interoperability, and much more. Sysadmins will especially appreciate the thorough and up-to-date discussions of such difficult topics such as DNS, LDAP, security, and the management of IT service organizations. Linux® Administration Handbook, Second Edition, reflects the current versions of these leading distributions: Red Hat® Enterprise Linux® FedoraTM Core SUSE® Linux Enterprise Debian® GNU/Linux Ubuntu® Linux Sharing their war stories and hard-won insights, the authors capture the behavior of Linux systems in the real world, not just in ideal environments. They explain complex tasks in detail and illustrate these tasks with examples drawn from their extensive hands-on experience.

The leading book in its field, this guide focuses on the design, development and coding of network software under the UNIX operating system. Provides over 15,000 lines of C code with descriptions of how and why a given solution is achieved. For programmers seeking an indepth tutorial on sockets, transport level interface (TLI), interprocess communications (IPC) facilities under System V and BSD UNIX.

"Steve Rago offers valuable insights into the kernel-level features of SVR4 not covered elsewhere; I think readers will especially appreciate the coverage of STREAMS, TLI, and SLIP." - W. Richard Stevens, author of UNIX Network Programming, Advanced Programming in the UNIX Environment, TCP/IP Illustrated Volume 1, and TCP/IP Illustrated Volume 2 Finally, with UNIX(R) System V Network Programming, an authoritative reference is available for programmers and system architects interested in building networked and distributed applications for UNIX System V. Even if you currently use a different version of the UNIX system, such as the latest release of 4.3BSD or SunOS, this book is valuable to you because it is centered around UNIX System V Release 4, the version of the UNIX system that unified many of the divergent UNIX implementations. For those professionals new to networking and UNIX system programming, two introductory chapters are provided. The author then presents the programming interfaces most important to building communication software in System V, including STREAMS, the Transport Layer Interface library, Sockets, and Remote Procedure Calls. So that your designs are not limited to user-level, the author also explains how to write kernel-level communication software, including STREAMS drivers, modules, and multiplexors. Many examples are provided, including an Ethernet driver and a transport-level multiplexing driver. In the final chapter, the author brings the material from previous chapters together, presenting the design of a SLIP communication package. 0201563185B04062001

To build today's highly distributed, networked applications and services, you need deep mastery of sockets and other key networking APIs. One book delivers comprehensive, start-to-finish guidance for building robust, high-performance networked systems in any environment: UNIX Network Programming, Volume 1, Third Edition.

As networks, devices, and systems continue to evolve, software engineers face the unique challenge of creating reliable distributed applications within frequently changing environments. C++ Network Programming, Volume 1, provides practical solutions for developing and optimizing complex distributed systems using the ADAPTIVE Communication Environment (ACE), a revolutionary open-source framework that runs on dozens of hardware platforms and operating systems. This book guides software professionals through the traps and pitfalls of developing efficient, portable, and flexible networked applications. It explores the inherent design complexities of concurrent networked applications and the tradeoffs that must be considered when working to master them. C++ Network Programming begins with an overview of the issues and tools involved in writing distributed concurrent applications. The book then provides the essential design dimensions, patterns, and principles needed to develop flexible and efficient concurrent networked applications. The book's expert author team shows you how to enhance design skills while applying C++ and patterns effectively to develop object-oriented networked applications. Readers will find coverage of: C++ network programming, including an overview and strategies for addressing common development challenges The ACE Toolkit Connection protocols, message exchange, and message-passing versus shared memory Implementation methods for reusable networked application services Concurrency in object-oriented network programming Design principles and patterns for ACE wrapper facades With this book, C++ developers have at their disposal the most complete toolkit available for developing successful, multiplatform, concurrent networked applications with ease and efficiency.

?????Linux ?????UNIX ?????????????????????Linux C ??????????Linux ?UNIX ?????????????????????Linux ???????????DBM?MySQL???????Linux ??????X ???Linux?????????Linux

??????????????????????????????????????
???????IEEE?????????—POSIX??(????Pthreads??),????????????????????????,????????????????????????????????
????????????????????????????,???C++????????????????????????????????

Modelling of information is necessary in developing information systems. Information is acquired from many sources, by using various methods and tools. It must be recognized, conceptualized, and conceptually organized efficiently so that users can easily understand and use it. Modelling is needed to understand, explain, organize, predict, and reason on information. It also helps to master the role and functions of components of information systems. Modelling can be performed with many different purposes in mind, at different levels, and by using different notions and different background theories. It can be made by emphasizing users' conceptual understanding of information on a domain level, on an algorithmic level, or on representation levels. On each level, the objects and structures used on them are different, and different rules govern the behavior on them. Therefore the notions, rules, theories, languages, and methods for modelling on different levels are also different. It will be useful if we can develop theories and methodologies for modelling, to be used in different situations, because databases, knowledge bases, and repositories in knowledge management systems, developed on the basis of models and used to technically store information, are growing day by day. In this publication, the interest is focused on modelling of information, and one of the central topics is modelling of time. Scientific and technical papers of high quality are brought together in this book.

?????????,???????C?????,???????C?????????????.?????????,????????????????????????????,????????????????????
API.

With this comprehensive text, Solaris practitioners will find all the information they need as they face and overcome significant challenges of their everyday work. Real-world case studies, poignant examples, and illustrative diagrams are rolled into this thorough reference.

V.1 Networking APIs: sockets and XTI V.2 Interprocess communications.

????:Richard Helm,Ralph Johnson,John Vlissides ????:???,??,???

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

????:???

?????????????

Software -- Operating Systems.

The Linux Programming Interface (TLPI) is the definitive guide to the Linux and UNIX programming interface—the interface employed by nearly every application that runs on a Linux or UNIX system. In this authoritative work, Linux programming expert Michael Kerrisk provides detailed descriptions of the system calls and library functions that you need in order to master the craft of system programming, and accompanies his explanations with clear, complete example programs. You'll find descriptions of over 500 system calls and library functions, and more than 200 example programs, 88 tables, and 115 diagrams. You'll learn how to: –Read and write files efficiently –Use signals, clocks, and timers –Create processes and execute programs –Write secure programs –Write multithreaded programs using POSIX threads –Build and use shared libraries –Perform interprocess communication using pipes, message queues, shared memory, and semaphores –Write network applications with the sockets API While The Linux Programming Interface covers a wealth of Linux-specific features, including epoll, inotify, and the /proc file system, its emphasis on UNIX standards (POSIX.1-2001/SUSv3 and POSIX.1-2008/SUSv4) makes it equally valuable to programmers working on other UNIX platforms. The Linux Programming Interface is the most comprehensive single-volume work on the Linux and UNIX programming interface, and a book that's destined to become a new classic.

A text focusing on the methods and alternatives for designed TCP/IP-based client/server systems and advanced techniques for specialized applications with Perl. A guide examining a collection of the best third party modules in the Comprehensive Perl Archive Network. Topics covered: Perl function libraries and techniques that allow programs to interact with resources over a network. IO: Socket library ; Net: FTP library -- Telnet library -- SMTP library ; Chat problems ; Internet Message Access Protocol (IMAP) issues ; Markup-language parsing ; Internet Protocol (IP) broadcasting and multicasting.

Well-implemented interprocess communications (IPC) are key to the performance of virtually every non-trivial UNIX program. In UNIX Network Programming, Volume 2, Second Edition, legendary UNIX expert W. Richard Stevens presents a comprehensive guide to every form of IPC, including message passing, synchronization, shared memory, and Remote Procedure Calls (RPC). Stevens begins with a basic introduction to IPC and the problems it is intended to solve. Step-by-step you'll learn how to maximize both System V IPC and the new Posix standards, which offer dramatic improvements in convenience and performance.

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.

?????:?????

????

????????????????

???????????????????????? ???????????????????????????????? ?????????????????????? ???????? ??????????????? ??????????
????????????????????? ?????????????????? ??????????? ??????????? ???????? ????????????????PIMCO???? ???????Eric Schmidt?????Jared Cohen???
????Google???1955??

????????????