Free download Linux performance tools brendan gregg Copy

use bpf tools to optimize performance fix problems and see inside running systems bpf based performance tools give you unprecedented visibility into systems and applications so you can optimize performance troubleshoot code strengthen security and reduce costs bpf performance tools linux system and application observability is the definitive guide to using these tools for observability pioneering bpf expert brendan gregg presents more than 150 ready to run analysis and debugging tools expert guidance on applying them and step by step tutorials on developing your own you ll learn how to analyze cpus memory disks file systems networking languages applications containers hypervisors security and the kernel gregg guides you from basic to advanced tools helping you generate deeper more useful technical insights for improving virtually any linux system or application learn essential tracing concepts and both core bpf front ends bcc and bpftrace master 150 powerful bpf tools including dozens created just for this book and available for download discover practical strategies tips and tricks for more effective analysis analyze compiled jit compiled and interpreted code in multiple languages c java bash shell and more generate metrics stack traces and custom latency histograms use complementary tools when they offer quick easy wins explore advanced tools built on bpf pcp and grafana for remote monitoring ebpf exporter and kubectl trace for tracing kubernetes foreword by alexei starovoitov creator of the new bpf bpf performance tools will be an indispensable resource for all administrators developers support staff and other it professionals working with any recent linux distribution in any enterprise or cloud environment systems performance second edition covers concepts strategy tools and tuning for operating systems and applications using linux based operating systems as the primary example a deep understanding of these tools and techniques is critical for developers today implementing the strategies described in this thoroughly revised and updated edition can lead to a better end user experience and lower costs especially for cloud computing environments that charge by the os instance systems performance expert and best selling author brendan gregg summarizes relevant operating system hardware and application theory to quickly get professionals up to speed even if they have never analyzed performance before gregg then provides in depth explanations of the latest tools and techniques including extended bpf and shows how to get the most out of cloud web and large scale enterprise systems key topics covered include hardware kernel and application internals and how they perform methodologies for rapid performance analysis of complex systems optimizing cpu memory file system disk and networking usage sophisticated profiling and tracing with perf ftrace and bpf bcc and bpftrace performance challenges associated with cloud computing hypervisors benchmarking more effectively featuring up to date coverage of linux operating systems and environments systems performance second edition also addresses issues that apply to any computer system the book will be a go to reference for many years to come and like the first edition required reading at leading tech companies register your book for convenient access to downloads updates and or corrections as they become available see inside book for details the complete guide to optimizing systems performance written by the winner of the 2013 lisa award for outstanding achievement in system administration large scale enterprise cloud and virtualized computing systems have introduced serious performance challenges now internationally renowned performance expert brendan gregg has brought together proven methodologies tools and metrics for analyzing and tuning even the most complex environments systems performance enterprise and the cloud focuses on linux and unix performance while illuminating performance issues that are relevant to all operating systems you ll gain deep insight into how systems work and perform and learn methodologies for analyzing and improving system and application performance gregg presents examples from bare metal systems and virtualized cloud tenants running linux based ubuntu fedora centos and the illumos based joyent smartostm and omniti omnios he systematically covers modern systems performance including the traditional analysis of cpus memory disks and networks and new areas including

cloud computing and dynamic tracing this book also helps you identify and fix the unknown unknowns of complex performance bottlenecks that emerge from elements and interactions you were not aware of the text concludes with a detailed case study showing how a real cloud customer issue was analyzed from start to finish coverage includes modern performance analysis and tuning terminology concepts models methods and techniques dynamic tracing techniques and tools including examples of dtrace systemtap and perf kernel internals uncovering what the os is doing using system observability tools interfaces and frameworks understanding and monitoring application performance optimizing cpus processors cores hardware threads caches interconnects and kernel scheduling memory optimization virtual memory paging swapping memory architectures busses address spaces and allocators file system i o including caching storage devices controllers disk i o workloads raid and kernel i o network related performance issues protocols sockets interfaces and physical connections performance implications of os and hardware based virtualization and new issues encountered with cloud computing benchmarking getting accurate results and avoiding common mistakes this guide is indispensable for anyone who operates enterprise or cloud environments system network database and web admins developers and other professionals for students and others new to optimization it also provides exercises reflecting gregg s extensive instructional experience □□ □□□□□□□□□□□ □□□□ linux□□□ □□□□ c□□□□□ c□□□□□□□□□□□□□□□□□□□□□□□ □□□□i o □□□□□□□□□ □□□□□□□□ □□□□□ □□□□□□□ □□□□□□□ □□□□□□□□□□□□□□□□□□□□□□□□□□ □□□□ linux□□□□□□□□□□□ □□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□ □□□□2 6□□ □□□□□□□□□□□□□□□□□□□□rust□□□□□□□□□□□ □□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□ □□□□□□□□□□ □□ □□□□□□□□□□□□ □□□□□□□□ □□ □□□□□ □□□□□□□□□□□□□□□□ os□□□□□□□□□□□□□ □□□□ □□□□□□ □□□□□□□os mikanos □□□□□□□□□□ □□□□□□□□□□□□os □□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□ os□□□□□□□□□□□□ □□□□□□□□ □□□□□□□□□□□□□□□□□□□□□□□□□□□ os □□□□□□□□□□□□□□□□□□ mikanos□uefi bios□□□□□□□intel 64□□□□□□ □□□□□□□□□□□□□□□□□ usb3 0□□□□□□□ □□□□□□□□□ □□□□□□□□□□□□□□□ □□□□□□□□□□□os□□ □□□□□□□□□□□□□os□□□□□□□□□□□ mikanos□ c □□□□□□ □□□□□□ □□□□□□□□□□□□□□□□□□□□□□□□□□□□ □□□ 30□□□□□ os□□□□ □□□□□□ □□□□□□□□□□□□□ □□□□□□□□□□□□□□□□□□□□□ □□□□□□□□□□□□□□ □□□□□□□□□□□□□□□□□□ os□□□ □□□□□□□ □0□ os□□□□□□□□□ □1□ pc□□□□□□□□□□□□ □2□ edk ii□□□□□□□□□ □3□ □□□□□□□□□□□□□□□ □4□ □□□□□□□make□□ □5□ □□□□□□□□□□□□□□ □6□ □□□□□□pci □7□ □□□□ □fifo □8□ □□□□□ □9□ □□□□□□□ □10□ □□□□□ □11□ □□□□acpi □12□ □□□□ □13□ □□□□□□ 1 □14□ □□□□□□ 2 □15□ □□□□□ □16□ □□□□ □17□ □□□□□□□□ □18□ □□□□□□□□ □19□ □□□□□ □20□ □□□□□□□ □21□ □□□□□□□□□□ □22□ □□□□□□□□□□□ 1 □23□ □□□□□□□□□□□ 2 □24□ □□□□□□□□ □25□ □□□□□□□□□□□□ □26□ □□□□□□□□□□□□ □27□ □□ □□□□□□□ □28□ □□□□□□□□□□□□ □29□ □□□□□□ □30□ □□□□□□ □31□ □□□□□□ □□ □□□□□□□□□□□□□ mikanos□□□ edk ii□□□□□□□□ c □□□□□□□ ipxe ascii□□□□ □□□□□□□□□□□□□□ □□□□□□□□□□□□□□□□□□□ □□□ □□□□□□□ □□□□□□□□□□□□□□□□□□□□□□□□□□□□□ □□□□□□□□□□□□□□ □□ □□□□ □□□□□□□□□ □□□□□□□□ □□□□□□□□□ □□□□□ □□□□ □□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□ □□□□□□□□□□□□ □□□□□□□□ □□□□□□□□□□□□□□□□□□□□□□□□□□□ □□□□□□□□□ □□□□□□□□□ □□□□□□□□□□□□□□□□□ □□□□□□□□□□ □□□□□□□□□□□□□□□□□ □□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□ □□□□□□□□□□□□□□□ □□□ □□□□□□□□□□□□□□□ google□□□□□□□□□□□□□□□□□□ □□□□□□□□□□□□ □□□□□□□□□□□□□□□□□□□□□□□ □□□ rust□c□□□ □□□□□□□□□□□□□□□□□ □□□□□□□□□□□ □□□□□□□□□□□□□□□□□□□□□□□□□□ □□google□□□□□□□ □□□□□□□□□□□□□□□□□□□□□□□□□□□□ □□□□□□□□□□□□□□□□□□□□□□□□□□ chaos engineering teaches you to design and execute controlled experiments that uncover hidden problems summary auto engineers test the safety of a car by intentionally crashing it and carefully observing the results chaos engineering applies the same principles to software systems in chaos engineering site reliability through controlled disruption you ll learn to run your applications and infrastructure through a series of tests that simulate real life failures you ll maximize the benefits of chaos engineering by learning to think like a chaos engineer and how to design the proper experiments to ensure the reliability of your software with examples that cover a whole spectrum of software you ll be ready to run an intensive testing regime on anything from a simple wordpress site to a massive distributed system running on kubernetes purchase of the print book includes a free ebook in pdf kindle and epub formats from manning publications about the technology can your network survive a devastating failure could an accident bring your day to day operations to a halt chaos engineering simulates infrastructure outages component crashes and other calamities to show how systems and staff respond testing systems in distress is the best way to ensure their future

resilience which is especially important for complex large scale applications with little room for downtime about the book chaos engineering teaches you to design and execute controlled experiments that uncover hidden problems learn to inject system shaking failures that disrupt system calls networking apis and kubernetes based microservices infrastructures to help you practice the book includes a downloadable linux vm image with a suite of preconfigured tools so you can experiment quickly without risk what s inside inject failure into processes applications and virtual machines test software running on kubernetes work with both open source and legacy software simulate database connection latency test and improve your team s failure response about the reader assumes linux servers basic scripting skills required about the author mikolaj pawlikowski is a recognized authority on chaos engineering he is the creator of the kubernetes chaos engineering tool powerfulseal and the networking visibility tool goldpinger table of contents 1 into the world of chaos engineering part 1 chaos engineering fundamentals 2 first cup of chaos and blast radius 3 observability 4 database trouble and testing in production part 2 chaos engineering in action 5 poking docker 6 who you gonna call syscall busters 7 injecting failure into the jvm 8 application level fault injection 9 there s a monkey in my browser part 3 chaos engineering in kubernetes 10 chaos in kubernetes 11 automating kubernetes experiments 12 under the hood of kubernetes 13 chaos engineering for people 使ってる人も 初めての人もrailsを極めたければ この1冊 webアプリケーション開発のデファクトスタンダードであるrailsをその仕組みからrailsならではの開発手法 援用できる各種ライブラリやテストまであますことなく解説し尽くした必携かつ必読本 rails2 0完全対応 effectively debug kernel modules device drivers and the kernel itself by gaining a solid understanding of powerful open source tools and advanced kernel debugging techniques key features fully understand how to use a variety of kernel and module debugging tools and techniques using examples learn to expertly interpret a kernel oops and identify underlying defect s use easy to look up tables and clear explanations of kernel level defects to make this complex topic easy book description the linux kernel is at the very core of arguably the world s best production quality os debugging it though can be a complex endeavor linux kernel debugging is a comprehensive guide to learning all about advanced kernel debugging this book covers many areas in depth such as instrumentation based debugging techniques printk and the dynamic debug framework and shows you how to use kprobes memory related bugs tend to be a nightmare two chapters are packed with tools and techniques devoted to debugging them when the kernel gifts you an oops how exactly do you interpret it to be able to debug the underlying issue we ve got you covered concurrency tends to be an inherently complex topic so a chapter on lock debugging will help you to learn precisely what data races are including using kcsan to detect them some thorny issues both debug and performance wise require detailed kernel level tracing you ll learn to wield the impressive power of ftrace and its frontends you ll also discover how to handle kernel lockups hangs and the dreaded kernel panic as well as leverage the venerable gdb tool within the kernel kgdb along with much more by the end of this book you will have at your disposal a wide range of powerful kernel debugging tools and techniques along with a keen sense of when to use which what you will learn explore instrumentation based printk along with the powerful dynamic debug framework use static and dynamic kprobes to trap into kernel module functions catch kernel memory defects with kasan ubsan slub debug and kmemleak interpret an oops in depth and precisely identify it s source location understand data races and use kcsan to catch evasive concurrency defects leverage ftrace and trace cmd to trace the kernel flow in great detail write a custom kernel panic handler and detect kernel lockups and hangs use kgdb to single step and debug kernel module source code who this book is for this book is for linux kernel developers module driver authors and testers interested in debugging and enhancing their linux systems at the level of the kernel system administrators who want to understand and debug the internal infrastructure of their linux kernels will also find this book useful a good grasp on c programming and the linux command line is necessary some experience with kernel module development will help you follow along pythonの高速化技法を学ぶロングセラーの改訂版 python 3対応 最適化のノウハウをサンプルプログラムを使って学ぶ パフォーマンス障害を解決に導く基礎知識と実践のための方法論を徹底解説 this book on performance fundamentals covers unix openvms linux windows and mvs most of the theory and systems design principles can be applied to other operating systems as can some of the benchmarks the book equips professionals with the ability to assess performance characteristics in unfamiliar

environments it is suitable for practitioners especially those whose responsibilities include performance management tuning and capacity planning it managers with a technical outlook also benefit from the book as well as consultants and students in the world of systems for the first time in a professional capacity 2013年のアメリカ ミュータントのみならず あらゆる超能力者が迫害され 恐怖政治が人々の上に重くのしかかる暗黒の時代 今 この悲劇の連鎖に終止符を打つべく 一人のミュータントが立ち上がった 目指すは 全てが始まった1980年 現代と未来を股にかけるショッキングなストーリー展開で話題を呼んだ名作 デイズ オブ フューチャーパスト を中心に x menがコミックスシーンの頂上へと駆け上がった時期の作品を網羅 get up and running with system programming concepts in linux key featuresacquire insight on linux system architecture and its programming interfacesget to grips with core concepts such as process management signalling and pthreadspacked with industry best practices and dozens of code examplesbook description the linux os and its embedded and server applications are critical components of today s software infrastructure in a decentralized networked universe the industry s demand for proficient linux developers is only rising with time hands on system programming with linux gives you a solid theoretical base and practical industry relevant descriptions and covers the linux system programming domain it delves into the art and science of linux application programming system architecture process memory and management signaling timers pthreads and file io this book goes beyond the use api x to do y approach it explains the concepts and theories required to understand programming interfaces and design decisions the tradeoffs made by experienced developers when using them and the rationale behind them troubleshooting tips and techniques are included in the concluding chapter by the end of this book you will have gained essential conceptual design knowledge and hands on experience working with linux system programming interfaces what you will learnexplore the theoretical underpinnings of linux system architectureunderstand why modern oses use virtual memory and dynamic memory apisget to grips with dynamic memory issues and effectively debug themlearn key concepts and powerful system apis related to process managementeffectively perform file io and use signaling and timersdeeply understand multithreading concepts pthreads apis synchronization and schedulingwho this book is for hands on system programming with linux is for linux system engineers programmers or anyone who wants to go beyond using an api set to understanding the theoretical underpinnings and concepts behind powerful linux system programming apis to get the most out of this book you should be familiar with linux at the user level logging in using shell via the command line interface the ability to use tools such as find grep and sort working knowledge of the c programming language is required no prior experience with linux systems programming is assumed don t fly blind observability gives you actionable insights into your cloud native systems from pinpointing errors to increasing developer productivity to tracking compliance observability is the difference between an error message and an error explanation with a recipe how to resolve the error you know exactly which service is affected who s responsible for its repair and even how it can be optimized in the future cloud observability in action teaches you how to set up an observability system that learns from a cloud application s signals logging and monitoring all using free and open source tools in cloud observability in action you will learn how to apply observability in cloud native systems understand observability signals including their costs and benefits apply good practices around instrumentation and signal collection deliver dashboarding alerting and slos slis at scale choose the correct signal types for given roles or tasks pick the right observability tool for any given function communicate the benefits of observability to management a well designed observability system provides insight into bugs and performance issues in cloud native applications they help your development team understand the impact of code changes measure optimizations and track user experience best of all observability can even automate your error handling so that machine users apply their own fixes no more 3am calls for emergency outages about the technology cloud native systems are made up of hundreds of moving parts when something goes wrong it s not enough to know there is a problem you need to know where it is what it is and how to fix it this book takes you beyond traditional monitoring explaining observability systems that turn application telemetry into actionable insights about the book cloud observability in action gives you the background and techniques you need to successfully introduce observability into cloud based serverless and kubernetes environments in it you ll learn to use open standards and tools like opentelemetry prometheus and grafana to build your

own observability system and end reliance on proprietary software you ll discover insights from different telemetry signals including logs metrics traces and profiles plus the book s rigorous cost benefit analysis ensures you re getting a real return on your observability investment what s inside observability in and of cloud native systems dashboarding alerting and slos slis at scale signal types for any role or task state of the art open source observability tools about the reader for application developers platform owners devops and sres about the author michael hausenblas is a product owner in the aws open source observability team table of contents 1 end to end observability 2 signal types 3 sources 4 agents and instrumentation 5 backend destinations 6 frontend destinations 7 cloud operations 8 distributed tracing 9 developer observability 10 service level objectives 11 signal correlation sre□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□ google evernote□□□□□□□□ the oracle solaris dtrace feature revolutionizes the way you debug operating systems and applications using dtrace you can dynamically instrument software and quickly answer virtually any question about its behavior now for the first time there s a comprehensive authoritative guide to making the most of dtrace in any supported unix environment from oracle solaris to opensolaris mac os x and freebsd written by key contributors to the dtrace community dtrace teaches by example presenting scores of commands and easy to adapt downloadable d scripts these concise examples generate answers to real and useful questions and serve as a starting point for building more complex scripts using them you can start making practical use of dtrace immediately whether you re an administrator developer analyst architect or support professional the authors fully explain the goals techniques and output associated with each script or command drawing on their extensive experience they provide strategy suggestions checklists and functional diagrams as well as a chapter of advanced tips and tricks you ll learn how to write effective scripts using dtrace s d language use dtrace to thoroughly understand system performance expose functional areas of the operating system including i o filesystems and protocols use dtrace in the application and database development process identify and fix security problems with dtrace analyze the operating system kernel integrate dtrace into source code extend dtrace with other tools this book will help you make the most of dtrace to solve problems more quickly and efficiently and build systems that work faster and more reliably □□□□□□□□□□□□□□□□□□□□ gain both a firm practical understanding and sufficient theoretical insight into the inner workings of linux kernel internals learn to write high quality kernel module code understand the complexities of kernel synchronization purchase of the print or kindle book includes a free ebook in pdf format key features discover how to write linux kernel and module code for real world products implement industry grade techniques in real world scenarios for fast efficient memory allocation and data synchronization understand and exploit kernel architecture cpu scheduling and kernel synchronization techniques book descriptionthe 2nd edition of linux kernel programming is an updated comprehensive guide for new programmers to the linux kernel this book uses the recent 6 1 long term support lts linux kernel series which will be maintained until dec 2026 and also delves into its many new features further the civil infrastructure project has pledged to maintain and support this 6 1 super lts slts kernel right until august 2033 keeping this book valid for years to come you ll begin this exciting journey by learning how to build the kernel from source in a step by step manner you will then learn how to write your first kernel module by leveraging the kernel s powerful loadable kernel module lkm framework with this foundation you will delve into key kernel internals topics including linux kernel architecture memory management and cpu task scheduling you ll finish with understanding the deep issues of concurrency and gain insight into how they can be addressed with various synchronization locking technologies e g mutexes spinlocks atomic refcount operators rw spinlocks and even lock free technologies such as per cpu and rcu by the end of this book you ll have a much better understanding of the fundamentals of writing the linux kernel and kernel module code that can straight away be used in real world projects and products what you will learn configure and build the 6 1 lts kernel from source write high quality modular kernel code lkm framework for 6 x kernels explore modern linux kernel architecture get to grips with key internals details regarding memory management within the kernel understand and work with various dynamic kernel memory alloc dealloc apis discover key internals aspects regarding cpu scheduling within the kernel including cgroups v2

gain a deeper understanding of kernel concurrency issues learn how to work with key kernel synchronization primitives who this book is for this book is for beginner linux programmers and developers looking to get started with the linux kernel providing a knowledge base to understand required kernel internal topics and overcome frequent and common development issues a basic understanding of linux cli and c programming is assumed 本書はシステム設計の観点からunixカーネルに焦点を当てたものである 本書では多くの商用 研究用の各種のunixを述べている カーネルの構成要素に関して 主要なunixシステムがどのような実装を選択し 他の実装と比べた際のその優劣や構成と設計の観点から探求した solaristm application programming is a comprehensive guide to optimizing the performance of applications running in your solaris environment from the fundamentals of system performance to using analysis and optimization tools to their fullest this wide ranging resource shows developers and software architects how to get the most from solaris systems and applications whether you re new to performance analysis and optimization or an experienced developer searching for the most efficient ways to solve performance issues this practical guide gives you the background information tips and techniques for developing optimizing and debugging applications on solaris the text begins with a detailed overview of the components that affect system performance this is followed by explanations of the many developer tools included with solaris os and the sun studio compiler and then it takes you beyond the basics with practical real world examples in addition you will learn how to use the rich set of developer tools to identify performance problems accurately interpret output from the tools and choose the smartest most efficient approach to correcting specific problems and achieving maximum system performance coverage includes a discussion of the chip multithreading cmt processors from sun and how they change the way that developers need to think about performance a detailed introduction to the performance analysis and optimization tools included with the solaris os and sun studio compiler practical examples for using the developer tools to their fullest including informational tools compilers floating point optimizations libraries and linking performance profilers and debuggers guidelines for interpreting tool analysis output optimization including hardware performance counter metrics and source code optimizations techniques for improving application performance using multiple processes or multiple threads an overview of hardware and software components that affect system performance including coverage of sparc and x64 processors a guide to the most recent advanced features of the widely used openmp parallel programming model with coverage of major features in openmp 4 5 this book offers an up to date practical tutorial on advanced features in the widely used openmp parallel programming model building on the previous volume using openmp portable shared memory parallel programming mit press this book goes beyond the fundamentals to focus on what has been changed and added to openmp since the 2 5 specifications it emphasizes four major and advanced areas thread affinity keeping threads close to their data accelerators special hardware to speed up certain operations tasking to parallelize algorithms with a less regular execution flow and simd hardware assisted operations on vectors as in the earlier volume the focus is on practical usage with major new features primarily introduced by example examples are restricted to c and c but are straightforward enough to be understood by fortran programmers after a brief recap of openmp 2 5 the book reviews enhancements introduced since 2 5 it then discusses in detail tasking a major functionality enhancement non uniform memory access numa architectures supported by openmp simd or single instruction multiple data heterogeneous systems a new parallel programming model to offload computation to accelerators and the expected further development of openmp ever wished you could spy on your computer with a handy incantation or bewitch your programs to debug themselves now you can by becoming a linux wizard okay reading these zines won t actually make you a wizard but you ll sure feel like one after you learn some neat linux tricks with this collected edition of julia evans s wildly popular linux zines you ll view programming in a way you never have before now on fancier paper an expert guide to software performance optimization from mobile and cloud apps to video games to driverless vehicle control more and more software is time constrained it must deliver reliable results seamlessly consistently and virtually instantaneously if it doesn t customers are unhappy and sometimes lives are put at risk when complex software underperforms or fails software engineers need to identify and address the root causes this is difficult and historically few tools have been available to help in understanding

software dynamics performance expert richard l sites tackles the problem head on offering expert methods and advanced tools for understanding complex time constrained software dynamics improving reliability and troubleshooting challenging performance problems sites draws on several decades of experience pioneering software performance optimization as well as extensive experience teaching graduate level developers he introduces principles and techniques for use in any environment from embedded devices to datacenters illuminating them with examples based on x86 or arm processors running linux and linked by ethernet he also guides readers through building and applying a powerful new extremely low overhead open source software tool kutrace to precisely trace executions on every cpu core using insights gleaned from this tool readers can apply nuanced solutions not merely brute force techniques such as turning off caches or cores measure and address issues associated with cpus memory disk ssd networks and their interactions fix programs that are always too slow and those that sometimes lag for no apparent reason design useful observability logging and time stamping capabilities into your code reason more effectively about performance data to see why reality differs from expectations identify problems such as excess execution slow instruction execution waiting for resources and software locks understanding software dynamics will be valuable to experienced software professionals including application and os developers hardware and system architects real time system designers and game developers as well as advanced students register your book for convenient access to downloads updates and or corrections as they become available see inside book for details discover how to write high quality character driver code interface with userspace work with chip memory and gain an in depth understanding of working with hardware interrupts and kernel synchronization key featuresdelve into hardware interrupt handling threaded irqs tasklets softirqs and understand which to use whenexplore powerful techniques to perform user kernel interfacing peripheral i o and use kernel mechanismswork with key kernel synchronization primitives to solve kernel concurrency issuesbook description linux kernel programming part 2 char device drivers and kernel synchronization is an ideal companion guide to the linux kernel programming book this book provides a comprehensive introduction for those new to linux device driver development and will have you up and running with writing misc class character device driver code on the 5 4 lts linux kernel in next to no time you ll begin by learning how to write a simple and complete misc class character driver before interfacing your driver with user mode processes via procfs sysfs debugfs netlink sockets and ioctl you ll then find out how to work with hardware i o memory the book covers working with hardware interrupts in depth and helps you understand interrupt request irq allocation threaded irq handlers tasklets and softirqs you ll also explore the practical usage of useful kernel mechanisms setting up delays timers kernel threads and workqueues finally you ll discover how to deal with the complexity of kernel synchronization with locking technologies mutexes spinlocks and atomic refcount operators including more advanced topics such as cache effects a primer on lock free techniques deadlock avoidance with lockdep and kernel lock debugging techniques by the end of this linux kernel book you ll have learned the fundamentals of writing linux character device driver code for real world projects and products what you will learnget to grips with the basics of the modern linux device model ldm write a simple yet complete misc class character device driverperform user kernel interfacing using popular methodsunderstand and handle hardware interrupts confidentlyperform i o on peripheral hardware chip memoryexplore kernel apis to work with delays timers kthreads and workqueuesunderstand kernel concurrency issueswork with key kernel synchronization primitives and discover how to detect and avoid deadlockwho this book is for an understanding of the topics covered in the linux kernel programming book is highly recommended to make the most of this book this book is for linux programmers beginning to find their way with device driver development linux device driver developers looking to overcome frequent and common kernel driver development issues as well as perform common driver tasks such as user kernel interfaces performing peripheral i o handling hardware interrupts and dealing with concurrency will benefit from this book a basic understanding of linux kernel internals and common apis kernel module development and c programming is required opensolaris is a rapidly evolving operating system with roots in solaris 10 suitable for deployment on laptops desktop workstations storage appliances

and data center servers from the smallest single purpose systems to the largest enterprise class systems the growing opensolaris community now has hundreds of thousands of participants and users in government agencies commercial businesses and universities with more than 100 user groups around the world contributing to the use and advancement of opensolaris new releases of opensolaris become available every six months with contributions from both sun engineers and opensolaris community members this book covers the opensolaris 2008 11 release pro opensolaris was written to demonstrate that you can host your open source applications and solutions on opensolaris taking advantage of its advanced features such as containers and other forms of virtualization the zfs file system and dtrace it s assumed that you are already fairly knowledgeable about developing on linux systems so the authors give an overview of the similarities and differences between linux and opensolaris and then present details on how to use the service management facility smf zfs zones and even a bit of dtrace they also provide pointers to the many project communities associated with new opensolaris features special focus is given to web development using familiar applications such as apache tomcat and mysql along with the netbeans ide and showing you how to exploit some of opensolaris s unique technologies securing observing and troubleshooting containerized workloads on kubernetes can be daunting it requires a range of considerations from infrastructure choices and cluster configuration to deployment controls and runtime and network security with this practical book you ll learn how to adopt a holistic security and observability strategy for building and securing cloud native applications running on kubernetes whether you re already working on cloud native applications or are in the process of migrating to its architecture this guide introduces key security and observability concepts and best practices to help you unleash the power of cloud native applications authors brendan creane and amit gupta from tigera take you through the full breadth of new cloud native approaches for establishing security and observability for applications running on kubernetes learn why you need a security and observability strategy for cloud native applications and determine your scope of coverage understand key concepts behind the book s security and observability approach explore the technology choices available to support this strategy discover how to share security responsibilities across multiple teams or roles learn how to architect kubernetes security and observability for multicloud and hybrid environments シェルのワンライナー その場かぎりの1行プログラム は linuxやmacのコマンドラインインターフェースを立ち上げたら すぐにプログラムを書いて即実行できます ちょっとした作業を手っ取り早く片付けるのに向いています 大量のデータ処理 繰り返し処理ほどその効果は絶大です 本書は そんなシェル ワンライナーを身につけるためのトレーニング本です 大量の問題を繰り返し解くことで 実務でワンライナーを活用できるようになることを目指します 第1部でシェルによるデータ加工の定石を学び 第2部でとっさの閃きと柔軟な発想力を養い 第3部で実務にありそうな問題に取り組むことで応用力を鍛えます 実務でワンライナーを役立てるには os linux シェル スクリプト言語 正規表現 文字コード 画像データ 日付や時間データ 通信プロトコルなどの幅広い知識が必要となります 本書では まず 練習問題 でそのような事前知識を解説し理解を含め 問題 で力試しを行うという構成になっています go言語の実践テクニックを身につけられる pragmatic bite sized programming advice from koder with attitude kode vicious for many years i have been a fan of the regular columns by kode vicious in communications of the acm the topics are not only timely they re explained with wit and elegance from the foreword by donald e knuth writing as kode vicious kv george v neville neil has spent more than 15 years sharing incisive advice and fierce insights for everyone who codes works with code or works with coders now in the kollected kode vicious he has brought together his best essays and socratic dialogues on the topic of building more effective computer systems these columns have been among the most popular items published in acm queue magazine as well as communications of the acm and kv s entertaining and perceptive explorations are supplemented here with new material that illuminates broader themes and addresses issues relevant to every software professional neville neil cuts to the heart of the matter and offers practical takeaways for newcomers and veterans alike on the following topics the kode at hand what to do or not to do with a specific piece of code koding konundrums issues that surround code such as testing and documentation systems design overall systems design topics from abstraction and threads to security machine to machine distributed systems and computer networking human to human dealing with developers managers and other people each chapter brings together letters responses and advice that apply directly to day to day problems faced by those who work in or with computing systems while the answers to the questions posed are always written with an eye towards humor the advice given is deadly serious register your book for convenient

access to downloads updates and or corrections as they become available see inside book for details [illegible] ether [illegible]lan[illegible]lan[illegible]
lan[illegible]

BPF Performance Tools 2019-11-27 use bpf tools to optimize performance fix problems and see inside running systems bpf based performance tools give you unprecedented visibility into systems and applications so you can optimize performance troubleshoot code strengthen security and reduce costs bpf performance tools linux system and application observability is the definitive guide to using these tools for observability pioneering bpf expert brendan gregg presents more than 150 ready to run analysis and debugging tools expert guidance on applying them and step by step tutorials on developing your own you ll learn how to analyze cpus memory disks file systems networking languages applications containers hypervisors security and the kernel gregg guides you from basic to advanced tools helping you generate deeper more useful technical insights for improving virtually any linux system or application learn essential tracing concepts and both core bpf front ends bcc and bpftrace master 150 powerful bpf tools including dozens created just for this book and available for download discover practical strategies tips and tricks for more effective analysis analyze compiled jit compiled and interpreted code in multiple languages c java bash shell and more generate metrics stack traces and custom latency histograms use complementary tools when they offer quick easy wins explore advanced tools built on bpf pcp and grafana for remote monitoring ebpf exporter and kubectl trace for tracing kubernetes foreword by alexei starovoitov creator of the new bpf bpf performance tools will be an indispensable resource for all administrators developers support staff and other it professionals working with any recent linux distribution in any enterprise or cloud environment

BPF Performance Tools 2020 systems performance second edition covers concepts strategy tools and tuning for operating systems and applications using linux based operating systems as the primary example a deep understanding of these tools and techniques is critical for developers today implementing the strategies described in this thoroughly revised and updated edition can lead to a better end user experience and lower costs especially for cloud computing environments that charge by the os instance systems performance expert and best selling author brendan gregg summarizes relevant operating system hardware and application theory to quickly get professionals up to speed even if they have never analyzed performance before gregg then provides in depth explanations of the latest tools and techniques including extended bpf and shows how to get the most out of cloud web and large scale enterprise systems key topics covered include hardware kernel and application internals and how they perform methodologies for rapid performance analysis of complex systems optimizing cpu memory file system disk and networking usage sophisticated profiling and tracing with perf ftrace and bpf bcc and bpftrace performance challenges associated with cloud computing hypervisors benchmarking more effectively featuring up to date coverage of linux operating systems and environments systems performance second edition also addresses issues that apply to any computer system the book will be a go to reference for many years to come and like the first edition required reading at leading tech companies register your book for convenient access to downloads updates and or corrections as they become available see inside book for details

Systems Performance 2020-12-09 the complete guide to optimizing systems performance written by the winner of the 2013 lisa award for outstanding achievement in system administration large scale enterprise cloud and virtualized computing systems have introduced serious performance challenges now internationally renowned performance expert brendan gregg has brought together proven methodologies tools and metrics for analyzing and tuning even the most complex environments systems performance enterprise and the cloud focuses on linux and unix performance while illuminating performance issues that are relevant to all operating systems you ll gain deep insight into how systems work and perform and learn methodologies for analyzing and improving system and application performance gregg presents examples from bare metal systems and virtualized cloud tenants running linux based ubuntu fedora centos and the illumos based joyent smartostm and omniti omnios he systematically covers modern systems performance including the traditional analysis of cpus memory disks and networks and new areas including cloud computing and dynamic tracing this book also helps you identify and fix the unknown unknowns of complex performance bottlenecks that emerge from elements

and interactions you were not aware of the text concludes with a detailed case study showing how a real cloud customer issue was analyzed from start to finish coverage includes modern performance analysis and tuning terminology concepts models methods and techniques dynamic tracing techniques and tools including examples of dtrace systemtap and perf kernel internals uncovering what the os is doing using system observability tools interfaces and frameworks understanding and monitoring application performance optimizing cpus processors cores hardware threads caches interconnects and kernel scheduling memory optimization virtual memory paging swapping memory architectures busses address spaces and allocators file system i o including caching storage devices controllers disk i o workloads raid and kernel i o network related performance issues protocols sockets interfaces and physical connections performance implications of os and hardware based virtualization and new issues encountered with cloud computing benchmarking getting accurate results and avoiding common mistakes this guide is indispensable for anyone who operates enterprise or cloud environments system network database and web admins developers and other professionals for students and others new to optimization it also provides exercises reflecting gregg s extensive instructional experience

Systems Performance 2013-10-07 □□□

□□□□□□□□□□□□□□□□2□ 2023-01-24 □□□□ linux□□□ □□□□ c□□□□□ c□□□□□□□□□□□□□□□□□□□□□□□□ □□□□i o □□□□□□□□□□ □□□□□□□□ □□□□□ □□□□□□□ □□□□□□□ □□□□□□□□ □□□□□□□□□□□□□□□□□□□□□□ linux□□□□□□□□□□□ □□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□ □□□□2 6□□

Linux□□□□□□□□□□□ 2008-04 □□□□□□□□□□□□□□□□□□□□□□rust□□□□□□□□□□□□□

□□□□□□□*Rust* 2018-08 □□□ □□ □□□□□□□□□□□□□ □□□□□□□□□ □□ □□□□□ □□□□□□□□□□□□□□□□ os□□□□□□□□□□□□□□ □□□□ □□□□□□ □□□□□□□os mikanos □□□□□□□□□□ □□□□□□□□□□□os□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□ os□□□□□□□□□□□□ □□□□□□□□ □□□□□□□□□□□□□□□□□□□□□□□□□□□□□ os □□□□□□□□□□□□□□□□□ mikanos□uefi bios□□□□□□□intel 64□□□□□□ □□□□□□□□□□□□□□□□□□□ usb3 0□□□□□□□ □□□□□□□□□□ □□□□□□□□□□□□□□□□□□ □□□□□□□□□□□□□□os□□ □□ □□□□□□□□□□os□□□□□□□□□□□ mikanos□ c □□□□□□ □□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□ □□□ 30□□□□□ os□□□□ □□□□□□ □□□□□□□□□□□□ □□□□□□□□□□□□□□□□□□□ □□□□ □□□□□□□□□□□□ □□□□□□□□□□□□□□□□□□□□ os□□□ □□□□□□□ □0□ os□□□□□□□□□ □1□ pc□□□□□□□□□□□□ □2□ edk ii□□□□□□□□□ □3□ □□□□□□□□□□□□□□ □4□ □□□□□□ □make□□ □5□ □□□□□□□□□□□□□ □6□ □□□□□□pci □7□ □□□□□fifo □8□ □□□□□ □9□ □□□□□□□ □10□ □□□□□ □11□ □□□□acpi □12□ □□□□ □13□ □□□□□□ 1 □14□ □□□□□□ 2 □15□ □□□□□ □16□ □□□□ □17□ □□□□□□□□ □18□ □□□□□□□□ □19□ □□□□□ □20□ □□□□□□□ □21□ □□□□□□□□□□ □22□ □□□□□□□□□□□ 1 □23□ □□□□□□□□□□□ 2 □24□ □□□ □□□□□ □25□ □□□□□□□□□□□□□ □26□ □□□□□□□□□□□□□ □27□ □□□□□□□□□ □28□ □□□□□□□□□□□□ □29□ □□□□□□ □30□ □□□□□□ □31□ □□□□□□ □□ □□□□□□□□□□□□ mikanos□ □□ edk ii□□□□□□□ c □□□□□□□ ipxe ascii□□□□

□□□□□**OS**□□□□ 2021-03-22 □□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□ □□□ □□□□□□□□ □□□□□□□□□□□□□□□□□□□□□□□□□□□□ □□□□□□□□□□□□□□□ □□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□ □□□□□□□□□□ □□□□ □□□□□□□□□ □□□□□□□□□□□□□□□□□ □□□□□ □□□□ □□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□ □□□□□□□□□□□□ □□□□□□□ □□□□□□□□□□□□□□□□□□□□□□□□□□□ □□ □□□□□□□□ □□□□□□□□□ □□□□□□□□□□□□□□□□□□□□□□□□□□□ □□□□□□□□□□□□□□□□□□ □□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□ □□□□□□□□□□□□□□□□

□□□□□□□□□□□□□□□□□□ 2019-07-17 □□ □□□□□□□□□□□□□□□

□□□□□□□□□□□□□□□□□ 2022-03-08 google□□□□□□□□□□□□□□□□□□□□□ □□□□□□□□□□□□□□ □□□□□□□□□□□□□□□□□□□□□□□□□ □□□

SRE□□□□□□□□□□□□□□□□□□□□ 2017-08 rust□c□□□ □□□□□□□□□□□□□□□□ □□□□□□□□□□ □□□□□□□□□□□□□□□□□□□□□□□□□□□□□□

□□□□□□□□□□□□ 2021-08 □□google□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□ □□□□□□□□□□□□□□□□□□□□□□□□□□□□□

Google□□□□□□□□□□□□□□□□□ 2021-11-29 chaos engineering teaches you to design and execute controlled experiments that uncover hidden problems summary auto engineers test the safety of a car by intentionally crashing it and carefully observing the results chaos engineering applies the same principles to software systems in chaos engineering site reliability through controlled disruption you ll learn to run your applications and infrastructure through a series of tests that simulate real life failures you ll maximize the benefits of chaos engineering by learning to think like a chaos engineer and

how to design the proper experiments to ensure the reliability of your software with examples that cover a whole spectrum of software you ll be ready to run an intensive testing regime on anything from a simple wordpress site to a massive distributed system running on kubernetes purchase of the print book includes a free ebook in pdf kindle and epub formats from manning publications about the technology can your network survive a devastating failure could an accident bring your day to day operations to a halt chaos engineering simulates infrastructure outages component crashes and other calamities to show how systems and staff respond testing systems in distress is the best way to ensure their future resilience which is especially important for complex large scale applications with little room for downtime about the book chaos engineering teaches you to design and execute controlled experiments that uncover hidden problems learn to inject system shaking failures that disrupt system calls networking apis and kubernetes based microservices infrastructures to help you practice the book includes a downloadable linux vm image with a suite of preconfigured tools so you can experiment quickly without risk what s inside inject failure into processes applications and virtual machines test software running on kubernetes work with both open source and legacy software simulate database connection latency test and improve your team s failure response about the reader assumes linux servers basic scripting skills required about the author mikolaj pawlikowski is a recognized authority on chaos engineering he is the creator of the kubernetes chaos engineering tool powerfulseal and the networking visibility tool goldpinger table of contents 1 into the world of chaos engineering part 1 chaos engineering fundamentals 2 first cup of chaos and blast radius 3 observability 4 database trouble and testing in production part 2 chaos engineering in action 5 poking docker 6 who you gonna call syscall busters 7 injecting failure into the jvm 8 application level fault injection 9 there s a monkey in my browser part 3 chaos engineering in kubernetes 10 chaos in kubernetes 11 automating kubernetes experiments 12 under the hood of kubernetes 13 chaos engineering for people

Chaos Engineering 2021-02-14 使ってる人も 初めての人もrailsを極めたければ この1冊 webアプリケーション開発のデファクトスタンダードであるrailsをその仕組みからrailsならではの開発手法 援用できる各種ライブラリやテストまであますことなく解説し尽くした必携かつ必読本 rails2 0完全対応

Rails Way 2008-12 effectively debug kernel modules device drivers and the kernel itself by gaining a solid understanding of powerful open source tools and advanced kernel debugging techniques key features fully understand how to use a variety of kernel and module debugging tools and techniques using examples learn to expertly interpret a kernel oops and identify underlying defect s use easy to look up tables and clear explanations of kernel level defects to make this complex topic easy book description the linux kernel is at the very core of arguably the world s best production quality os debugging it though can be a complex endeavor linux kernel debugging is a comprehensive guide to learning all about advanced kernel debugging this book covers many areas in depth such as instrumentation based debugging techniques printk and the dynamic debug framework and shows you how to use kprobes memory related bugs tend to be a nightmare two chapters are packed with tools and techniques devoted to debugging them when the kernel gifts you an oops how exactly do you interpret it to be able to debug the underlying issue we ve got you covered concurrency tends to be an inherently complex topic so a chapter on lock debugging will help you to learn precisely what data races are including using kcsan to detect them some thorny issues both debug and performance wise require detailed kernel level tracing you ll learn to wield the impressive power of ftrace and its frontends you ll also discover how to handle kernel lockups hangs and the dreaded kernel panic as well as leverage the venerable gdb tool within the kernel kgdb along with much more by the end of this book you will have at your disposal a wide range of powerful kernel debugging tools and techniques along with a keen sense of when to use which what you will learn explore instrumentation based printk along with the powerful dynamic debug framework use static and dynamic kprobes to trap into kernel module functions catch kernel memory defects with kasan ubsan slub debug and kmemleak interpret an oops in depth and precisely identify it s source location understand data races and use kcsan to catch evasive concurrency defects leverage ftrace and trace cmd to trace the kernel flow in great detail write a custom kernel panic handler and detect kernel lockups and hangs

use kgdb to single step and debug kernel module source code who this book is for this book is for linux kernel developers module driver authors and testers interested in debugging and enhancing their linux systems at the level of the kernel system administrators who want to understand and debug the internal infrastructure of their linux kernels will also find this book useful a good grasp on c programming and the linux command line is necessary some experience with kernel module development will help you follow along

Linux Kernel Debugging 2022-08-05 pythonの高速化技法を学ぶロングセラーの改訂版 python 3対応 最適化のノウハウをサンプルプログラムを使って学ぶハイパフォーマンス*Python* 第*2*版 2023-04-14 パフォーマンス障害を解決に導く基礎知識と実践のための方法論を徹底解説詳解システム・パフォーマンス 2017-02 this book on performance fundamentals covers unix openvms linux windows and mvs most of the theory and systems design principles can be applied to other operating systems as can some of the benchmarks the book equips professionals with the ability to assess performance characteristics in unfamiliar environments it is suitable for practitioners especially those whose responsibilities include performance management tuning and capacity planning it managers with a technical outlook also benefit from the book as well as consultants and students in the world of systems for the first time in a professional capacity

High-Performance IT Services 2016-10-04 2013年のアメリカ ミュータントのみならず あらゆる超能力者が迫害され 恐怖政治が人々の上に重くのしかかる暗黒の時代 今 この悲劇の連鎖に終止符を打つべく 一人のミュータントが立ち上がった 目指すは 全てが始まった1980年 現代と未来を股にかけるショッキングなストーリー展開で話題を呼んだ名作 デイズ オブ フューチャーパスト を中心に x menがコミックスシーンの頂上へと駆け上がった時期の作品を網羅

X-MEN:デイズ・オブ・フューチャーパスト 2014-05-20 get up and running with system programming concepts in linux key featuresacquire insight on linux system architecture and its programming interfacesget to grips with core concepts such as process management signalling and pthreadspacked with industry best practices and dozens of code examplesbook description the linux os and its embedded and server applications are critical components of today s software infrastructure in a decentralized networked universe the industry s demand for proficient linux developers is only rising with time hands on system programming with linux gives you a solid theoretical base and practical industry relevant descriptions and covers the linux system programming domain it delves into the art and science of linux application programming system architecture process memory and management signaling timers pthreads and file io this book goes beyond the use api x to do y approach it explains the concepts and theories required to understand programming interfaces and design decisions the tradeoffs made by experienced developers when using them and the rationale behind them troubleshooting tips and techniques are included in the concluding chapter by the end of this book you will have gained essential conceptual design knowledge and hands on experience working with linux system programming interfaces what you will learnexplore the theoretical underpinnings of linux system architectureunderstand why modern oses use virtual memory and dynamic memory apisget to grips with dynamic memory issues and effectively debug themlearn key concepts and powerful system apis related to process managementeffectively perform file io and use signaling and timersdeeply understand multithreading concepts pthreads apis synchronization and schedulingwho this book is for hands on system programming with linux is for linux system engineers programmers or anyone who wants to go beyond using an api set to understanding the theoretical underpinnings and concepts behind powerful linux system programming apis to get the most out of this book you should be familiar with linux at the user level logging in using shell via the command line interface the ability to use tools such as find grep and sort working knowledge of the c programming language is required no prior experience with linux systems programming is assumed

Hands-On System Programming with Linux 2018-10-31 don t fly blind observability gives you actionable insights into your cloud native systems from pinpointing errors to increasing developer productivity to tracking compliance observability is the difference between an error message and an error explanation with a recipe how to resolve the error you know exactly which service is affected who s responsible for its repair and even how it can be optimized in the future cloud observability in action teaches you how to set up an observability system that learns from a cloud application s signals

logging and monitoring all using free and open source tools in cloud observability in action you will learn how to apply observability in cloud native systems understand observability signals including their costs and benefits apply good practices around instrumentation and signal collection deliver dashboarding alerting and slos slis at scale choose the correct signal types for given roles or tasks pick the right observability tool for any given function communicate the benefits of observability to management a well designed observability system provides insight into bugs and performance issues in cloud native applications they help your development team understand the impact of code changes measure optimizations and track user experience best of all observability can even automate your error handling so that machine users apply their own fixes no more 3am calls for emergency outages about the technology cloud native systems are made up of hundreds of moving parts when something goes wrong it s not enough to know there is a problem you need to know where it is what it is and how to fix it this book takes you beyond traditional monitoring explaining observability systems that turn application telemetry into actionable insights about the book cloud observability in action gives you the background and techniques you need to successfully introduce observability into cloud based serverless and kubernetes environments in it you ll learn to use open standards and tools like opentelemetry prometheus and grafana to build your own observability system and end reliance on proprietary software you ll discover insights from different telemetry signals including logs metrics traces and profiles plus the book s rigorous cost benefit analysis ensures you re getting a real return on your observability investment what s inside observability in and of cloud native systems dashboarding alerting and slos slis at scale signal types for any role or task state of the art open source observability tools about the reader for application developers platform owners devops and sres about the author michael hausenblas is a product owner in the aws open source observability team table of contents 1 end to end observability 2 signal types 3 sources 4 agents and instrumentation 5 backend destinations 6 frontend destinations 7 cloud operations 8 distributed tracing 9 developer observability 10 service level objectives 11 signal correlation

Cloud Observability in Action 2024-01-23 sre□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□ google evernote□□□□□□□□□ □□□□□□□□□□□□□□□□□□□□ 2020-06 the oracle solaris dtrace feature revolutionizes the way you debug operating systems and applications using dtrace you can dynamically instrument software and quickly answer virtually any question about its behavior now for the first time there s a comprehensive authoritative guide to making the most of dtrace in any supported unix environment from oracle solaris to opensolaris mac os x and freebsd written by key contributors to the dtrace community dtrace teaches by example presenting scores of commands and easy to adapt downloadable d scripts these concise examples generate answers to real and useful questions and serve as a starting point for building more complex scripts using them you can start making practical use of dtrace immediately whether you re an administrator developer analyst architect or support professional the authors fully explain the goals techniques and output associated with each script or command drawing on their extensive experience they provide strategy suggestions checklists and functional diagrams as well as a chapter of advanced tips and tricks you ll learn how to write effective scripts using dtrace s d language use dtrace to thoroughly understand system performance expose functional areas of the operating system including i o filesystems and protocols use dtrace in the application and database development process identify and fix security problems with dtrace analyze the operating system kernel integrate dtrace into source code extend dtrace with other tools this book will help you make the most of dtrace to solve problems more quickly and efficiently and build systems that work faster and more reliably

DTrace 2011-03-18 □□□□□□□□□□□□□□□□□□□□□□

TCP/IP□□□□□□□□□ 2003-06 gain both a firm practical understanding and sufficient theoretical insight into the inner workings of linux kernel internals learn to write high quality kernel module code understand the complexities of kernel synchronization purchase of the print or kindle book includes a free ebook in pdf format key features discover how to write linux kernel and module code for real world products implement industry grade techniques in

real world scenarios for fast efficient memory allocation and data synchronization understand and exploit kernel architecture cpu scheduling and kernel synchronization techniques book descriptionthe 2nd edition of linux kernel programming is an updated comprehensive guide for new programmers to the linux kernel this book uses the recent 6 1 long term support lts linux kernel series which will be maintained until dec 2026 and also delves into its many new features further the civil infrastructure project has pledged to maintain and support this 6 1 super lts slts kernel right until august 2033 keeping this book valid for years to come you ll begin this exciting journey by learning how to build the kernel from source in a step by step manner you will then learn how to write your first kernel module by leveraging the kernel s powerful loadable kernel module lkm framework with this foundation you will delve into key kernel internals topics including linux kernel architecture memory management and cpu task scheduling you ll finish with understanding the deep issues of concurrency and gain insight into how they can be addressed with various synchronization locking technologies e g mutexes spinlocks atomic refcount operators rw spinlocks and even lock free technologies such as per cpu and rcu by the end of this book you ll have a much better understanding of the fundamentals of writing the linux kernel and kernel module code that can straight away be used in real world projects and products what you will learn configure and build the 6 1 lts kernel from source write high quality modular kernel code lkm framework for 6 x kernels explore modern linux kernel architecture get to grips with key internals details regarding memory management within the kernel understand and work with various dynamic kernel memory alloc dealloc apis discover key internals aspects regarding cpu scheduling within the kernel including cgroups v2 gain a deeper understanding of kernel concurrency issues learn how to work with key kernel synchronization primitives who this book is for this book is for beginner linux programmers and developers looking to get started with the linux kernel providing a knowledge base to understand required kernel internal topics and overcome frequent and common development issues a basic understanding of linux cli and c programming is assumed

Linux Kernel Programming 2024-02-29 本書はシステム設計の観点からunixカーネルに焦点を当てたものである 本書では多くの商用 研究用の各種のunixを述べている カーネルの構成要素に関して 主要なunixシステムがどのような実装を選択し 他の実装と比べた際のその優劣や構成と設計の観点から探求した

Unix internals 2000-05 solaristm application programming is a comprehensive guide to optimizing the performance of applications running in your solaris environment from the fundamentals of system performance to using analysis and optimization tools to their fullest this wide ranging resource shows developers and software architects how to get the most from solaris systems and applications whether you re new to performance analysis and optimization or an experienced developer searching for the most efficient ways to solve performance issues this practical guide gives you the background information tips and techniques for developing optimizing and debugging applications on solaris the text begins with a detailed overview of the components that affect system performance this is followed by explanations of the many developer tools included with solaris os and the sun studio compiler and then it takes you beyond the basics with practical real world examples in addition you will learn how to use the rich set of developer tools to identify performance problems accurately interpret output from the tools and choose the smartest most efficient approach to correcting specific problems and achieving maximum system performance coverage includes a discussion of the chip multithreading cmt processors from sun and how they change the way that developers need to think about performance a detailed introduction to the performance analysis and optimization tools included with the solaris os and sun studio compiler practical examples for using the developer tools to their fullest including informational tools compilers floating point optimizations libraries and linking performance profilers and debuggers guidelines for interpreting tool analysis output optimization including hardware performance counter metrics and source code optimizations techniques for improving application performance using multiple processes or multiple threads an overview of hardware and software components that affect system performance including coverage of sparc and x64 processors

Solaris Application Programming 2007-12-27 a guide to the most recent advanced features of the widely used openmp parallel programming model with coverage of major features in openmp 4 5 this book offers an up to date practical tutorial on advanced features in the widely used openmp parallel programming model building on the previous volume using openmp portable shared memory parallel programming mit press this book goes beyond the fundamentals to focus on what has been changed and added to openmp since the 2 5 specifications it emphasizes four major and advanced areas thread affinity keeping threads close to their data accelerators special hardware to speed up certain operations tasking to parallelize algorithms with a less regular execution flow and simd hardware assisted operations on vectors as in the earlier volume the focus is on practical usage with major new features primarily introduced by example examples are restricted to c and c but are straightforward enough to be understood by fortran programmers after a brief recap of openmp 2 5 the book reviews enhancements introduced since 2 5 it then discusses in detail tasking a major functionality enhancement non uniform memory access numa architectures supported by openmp simd or single instruction multiple data heterogeneous systems a new parallel programming model to offload computation to accelerators and the expected further development of openmp

Using OpenMP-The Next Step 2017-10-20 ever wished you could spy on your computer with a handy incantation or bewitch your programs to debug themselves now you can by becoming a linux wizard okay reading these zines won t actually make you a wizard but you ll sure feel like one after you learn some neat linux tricks with this collected edition of julia evans s wildly popular linux zines you ll view programming in a way you never have before now on fancier paper

Your Linux Toolbox 2019-08-20 an expert guide to software performance optimization from mobile and cloud apps to video games to driverless vehicle control more and more software is time constrained it must deliver reliable results seamlessly consistently and virtually instantaneously if it doesn t customers are unhappy and sometimes lives are put at risk when complex software underperforms or fails software engineers need to identify and address the root causes this is difficult and historically few tools have been available to help in understanding software dynamics performance expert richard l sites tackles the problem head on offering expert methods and advanced tools for understanding complex time constrained software dynamics improving reliability and troubleshooting challenging performance problems sites draws on several decades of experience pioneering software performance optimization as well as extensive experience teaching graduate level developers he introduces principles and techniques for use in any environment from embedded devices to datacenters illuminating them with examples based on x86 or arm processors running linux and linked by ethernet he also guides readers through building and applying a powerful new extremely low overhead open source software tool kutrace to precisely trace executions on every cpu core using insights gleaned from this tool readers can apply nuanced solutions not merely brute force techniques such as turning off caches or cores measure and address issues associated with cpus memory disk ssd networks and their interactions fix programs that are always too slow and those that sometimes lag for no apparent reason design useful observability logging and time stamping capabilities into your code reason more effectively about performance data to see why reality differs from expectations identify problems such as excess execution slow instruction execution waiting for resources and software locks understanding software dynamics will be valuable to experienced software professionals including application and os developers hardware and system architects real time system designers and game developers as well as advanced students register your book for convenient access to downloads updates and or corrections as they become available see inside book for details

Understanding Software Dynamics 2021-11-02 discover how to write high quality character driver code interface with userspace work with chip memory and gain an in depth understanding of working with hardware interrupts and kernel synchronization key featuresdelve into hardware interrupt handling threaded irqs tasklets softirqs and understand which to use whenexplore powerful techniques to perform user kernel interfacing peripheral i o and use kernel mechanismswork with key kernel synchronization primitives to solve kernel concurrency issuesbook description linux kernel

programming part 2 char device drivers and kernel synchronization is an ideal companion guide to the linux kernel programming book this book provides a comprehensive introduction for those new to linux device driver development and will have you up and running with writing misc class character device driver code on the 5 4 lts linux kernel in next to no time you ll begin by learning how to write a simple and complete misc class character driver before interfacing your driver with user mode processes via procfs sysfs debugfs netlink sockets and ioctl you ll then find out how to work with hardware i o memory the book covers working with hardware interrupts in depth and helps you understand interrupt request irq allocation threaded irq handlers tasklets and softirqs you ll also explore the practical usage of useful kernel mechanisms setting up delays timers kernel threads and workqueues finally you ll discover how to deal with the complexity of kernel synchronization with locking technologies mutexes spinlocks and atomic refcount operators including more advanced topics such as cache effects a primer on lock free techniques deadlock avoidance with lockdep and kernel lock debugging techniques by the end of this linux kernel book you ll have learned the fundamentals of writing linux character device driver code for real world projects and products what you will learnget to grips with the basics of the modern linux device model ldm write a simple yet complete misc class character device driverperform user kernel interfacing using popular methodsunderstand and handle hardware interrupts confidentlyperform i o on peripheral hardware chip memoryexplore kernel apis to work with delays timers kthreads and workqueuesunderstand kernel concurrency issueswork with key kernel synchronization primitives and discover how to detect and avoid deadlockwho this book is for an understanding of the topics covered in the linux kernel programming book is highly recommended to make the most of this book this book is for linux programmers beginning to find their way with device driver development linux device driver developers looking to overcome frequent and common kernel driver development issues as well as perform common driver tasks such as user kernel interfaces performing peripheral i o handling hardware interrupts and dealing with concurrency will benefit from this book a basic understanding of linux kernel internals and common apis kernel module development and c programming is required

Linux Kernel Programming Part 2 - Char Device Drivers and Kernel Synchronization 2021-03-19 opensolaris is a rapidly evolving operating system with roots in solaris 10 suitable for deployment on laptops desktop workstations storage appliances and data center servers from the smallest single purpose systems to the largest enterprise class systems the growing opensolaris community now has hundreds of thousands of participants and users in government agencies commercial businesses and universities with more than 100 user groups around the world contributing to the use and advancement of opensolaris new releases of opensolaris become available every six months with contributions from both sun engineers and opensolaris community members this book covers the opensolaris 2008 11 release pro opensolaris was written to demonstrate that you can host your open source applications and solutions on opensolaris taking advantage of its advanced features such as containers and other forms of virtualization the zfs file system and dtrace it s assumed that you are already fairly knowledgeable about developing on linux systems so the authors give an overview of the similarities and differences between linux and opensolaris and then present details on how to use the service management facility smf zfs zones and even a bit of dtrace they also provide pointers to the many project communities associated with new opensolaris features special focus is given to web development using familiar applications such as apache tomcat and mysql along with the netbeans ide and showing you how to exploit some of opensolaris s unique technologies

Pro OpenSolaris 2009-05-29 securing observing and troubleshooting containerized workloads on kubernetes can be daunting it requires a range of considerations from infrastructure choices and cluster configuration to deployment controls and runtime and network security with this practical book you ll learn how to adopt a holistic security and observability strategy for building and securing cloud native applications running on kubernetes whether you re already working on cloud native applications or are in the process of migrating to its architecture this guide introduces key security and

observability concepts and best practices to help you unleash the power of cloud native applications authors brendan creane and amit gupta from tigera take you through the full breadth of new cloud native approaches for establishing security and observability for applications running on kubernetes learn why you need a security and observability strategy for cloud native applications and determine your scope of coverage understand key concepts behind the book s security and observability approach explore the technology choices available to support this strategy discover how to share security responsibilities across multiple teams or roles learn how to architect kubernetes security and observability for multicloud and hybrid environments

Kubernetes Security and Observability 2021-10-26 シェルのワンライナー その場かぎりの1行プログラム は linuxやmacのコマンドラインインターフェースを立ち上げたら すぐにプログラムを書いて即実行できます ちょっとした作業を手っ取り早く片付けるのに向いています 大量のデータ処理 繰り返し処理ほどその効果は絶大です 本書は そんなシェル ワンライナーを身につけるためのトレーニング本です 大量の問題を繰り返し解くことで 実務でワンライナーを活用できるようになることを目指します 第1部でシェルによるデータ加工の定石を学び 第2部でとっさの閃きと柔軟な発想力を養い 第3部で実務にありそうな問題に取り組むことで応用力を鍛えます 実務でワンライナーを役立てるには os linux シェル スクリプト言語 正規表現 文字コード 画像データ 日付や時間データ 通信プロトコルなどの幅広い知識が必要となります 本書では まず 練習問題 でそのような事前知識を解説し理解を含め 問題 で力試しを行うという構成になっています

1日**1**問、半年以内に習得シェル・ワンライナー**160**本ノック 2021-09-27 go言語の実践テクニックを身につけられる

Go言語による**Web**アプリケーション開発 2016-01 pragmatic bite sized programming advice from koder with attitude kode vicious for many years i have been a fan of the regular columns by kode vicious in communications of the acm the topics are not only timely they re explained with wit and elegance from the foreword by donald e knuth writing as kode vicious kv george v neville neil has spent more than 15 years sharing incisive advice and fierce insights for everyone who codes works with code or works with coders now in the kollected kode vicious he has brought together his best essays and socratic dialogues on the topic of building more effective computer systems these columns have been among the most popular items published in acm queue magazine as well as communications of the acm and kv s entertaining and perceptive explorations are supplemented here with new material that illuminates broader themes and addresses issues relevant to every software professional neville neil cuts to the heart of the matter and offers practical takeaways for newcomers and veterans alike on the following topics the kode at hand what to do or not to do with a specific piece of code koding konundrums issues that surround code such as testing and documentation systems design overall systems design topics from abstraction and threads to security machine to machine distributed systems and computer networking human to human dealing with developers managers and other people each chapter brings together letters responses and advice that apply directly to day to day problems faced by those who work in or with computing systems while the answers to the questions posed are always written with an eye towards humor the advice given is deadly serious register your book for convenient access to downloads updates and or corrections as they become available see inside book for details

The Kollected Kode Vicious 2020-10-14 最新のデータベースとストレージエンジン内部のコンセプトについて開発者に向けて解説 データベース選び 使用 保守に役立つ詳説データベース 2021-07 光を伝える媒体として存在が仮定された エーテル ether を語源とするイーサネットはlanの世界標準です また 初期のインターネットはイーサネットlan同士を接続することから始まったように インターネットの基幹技術でありながら イーサネットの体系的技術資料は多くありません 本書は イーサネットの基本仕様 動作原理から 各種ケーブルやコネクタの仕様 特性 用途 光ファイバーやギガビットイーサの情報の他 通信プロトコル 自動コンフィギュレーション 同軸ケーブルのタップの取り付け ケーブル工事 lanシステムの構築方法 運用や管理に関する情報もまとめた基礎学習から現場まで幅広く役立つ書籍です

C++ Coding Standards 2005-09

詳説イーサネット 2000-12

- thematic analysis essay example Full PDF
- tratamiento psicologico de la fobia social 4 ed nextwin .pdf
- 16 ways to love your lover (Read Only)
- order guide audi 2009 (PDF)
- the good doctor bringing healing to the hopeless (Read Only)
- kudela owaziyo zulu novels Copy
- subaru brz manual transmission .pdf
- garry kasparov on garry kasparov part 2 1985 1993 everyman chess [PDF]
- manual for 2002 ford expedition (PDF)
- english sats papers wolf (Read Only)
- the fate of ten lorien legacies 6 (PDF)
- how to heal a broken heart in 30 days [PDF]
- we love madeleines (Read Only)
- safety design in high rise construction new york city Full PDF
- speak with power and confidence patrick collins [PDF]
- the art of not being governed an anarchist history of upland southeast asia yale agrarian studies series .pdf
- le raccapriccianti avventure di una maestra mannara (PDF)
- mcdougal biology chapter 4 answer (PDF)
- il mio primo vangelo (PDF)
- fundamentals of logistics management irwinmcgraw hill series in marketing (PDF)