

《UNIX环境高级编程（第3版）》

图书基本信息

书名：《UNIX环境高级编程（第3版）》

13位ISBN编号：9787115352119

出版时间：2014-6-1

作者：史蒂文斯 (W.Richard Stevens),拉戈 (Stephen A.Rago)

页数：812

译者：戚正伟,张亚英,尤晋元

版权说明：本站所提供下载的PDF图书仅提供预览和简介以及在线试读，请支持正版图书。

更多资源请访问：www.tushu000.com

内容概要

《UNIX环境高级编程（第3版）》是被誉为UNIX编程“圣经”的Advanced Programming in the UNIX Environment一书的第3版。在本书第2版出版后的8年中，UNIX行业发生了巨大的变化，特别是影响UNIX编程接口的有关标准变化很大。本书在保持前一版风格的基础上，根据最新的标准对内容进行了修订和增补，反映了最新的技术发展。书中除了介绍UNIX文件和目录、标准I/O库、系统数据文件和信息、进程环境、进程控制、进程关系、信号、线程、线程控制、守护进程、各种I/O、进程间通信、网络IPC、伪终端等方面的内容，还在此基础上介绍了众多应用实例，包括如何创建数据库函数库以及如何与网络打印机通信等。此外，还在附录中给出了函数原型和部分习题的答案。

《UNIX环境高级编程（第3版）》内容权威，概念清晰，阐述精辟，对于所有层次UNIX/Linux程序员都是一本不可或缺的参考书。

适读人群：所有层次UNIX/Linux程序员

20多年来，严谨的C程序员都是依靠一本书来深入了解驱动UNIX和Linux内核的编程接口的实用知识的，这本书就是W. Richard Stevens所著的《UNIX高级环境编程》。现在，Stevens的同事Steve Rago彻底更新了这本经典著作。新的第3版支持当今领先的系统平台，反映了最新技术进展和最佳实践，并且符合最新的Single UNIX Specification第4版（SUSv4）。

Rago保留了使本书前版成为经典之作的精髓和方法。他在Stevens原著的基础上，从基础的文件、目录和进程讲起，并给诸如信号处理和终端I/O之类的先进技术保留较大的篇幅。他还深入讨论了线程和多线程编程、使用套接字接口驱动进程间通信（IPC）等方面的内容。

这一版涵盖了70多个最新版POSIX.1标准的新增接口，包括POSIX异步I/O、旋转锁、屏障（barrier）和POSIX信号量。此外，这一版删除了许多过时的接口，保留了一些广泛使用的接口。书中几乎所有实例都已经在目前最主流的4个平台上测试过，包括Solaris 10、Mac OS X 10.6.8（Darwin 10.8.0）、FreeBSD 8.0、Ubuntu 12.04（基于Linux 3.2内核）。

与前两版一样，读者仍可以通过实例学习，这些实例包括了1万多行可下载的ISO C源代码，书中通过简明但完整的程序阐述了400多个系统调用和函数，清楚地说明它们的用法、参数和返回值。为了使读者能融会贯通，书中还提供了几个贯穿整章的案例，每个案例都根据现在的技术环境进行了全面更新。

《UNIX环境高级编程（第3版）》帮助了几代程序员写出强大、高性能、可靠的代码。第3版根据当今主流系统进行更新，更具实用价值。

精彩书评：

本书第1版连同Stevens所著的系列网络技术书籍，被公认为优秀的、匠心独具的名著，成为极其畅销的作品……总之，这是一本弥足珍贵的经典著作的更新版。

——Dennis Ritchie，图灵奖得主，UNIX操作系统和C语言之父

对任何一个严谨的、专业的UNIX系统程序员而言，本书都是不可或缺的权威参考书。Rago更新和扩展了Stevens的经典著作，并保持了原书的风貌。书中利用清晰的实例演示了API的使用过程，还提到了许多在不同UNIX系统实现上编程时需要注意的陷阱，并指出如何使用相关的标准（如POSIX 1003.1 2004版和Single UNIX Specification第3版）来避免这些错误。

——Andrew Josey，The Open Group标准部门主管，POSIX 1003.1标准工作组主席

绝对的UNIX编程经典之一。

——Eric S. Raymond，《UNIX编程艺术》作者

Stephen Rago的更新版本对于使用众多UNIX及相关操作系统环境的广大专业用户来说是一个迟来的喜讯。这一版不仅删除了过时的接口，吸纳了较新的开发接口，还根据UNIX及类UNIX操作系统环境的几种主流实现发布的新版本全面更新了所有主题、实例和应用的背景。难能可贵的是，这一版本还保持了经典的第1版的风格和品位。

——Mukesh Kacker，Pronto Networks公司联合创始人和前任CTO

本书对于任何在UNIX系统上编写程序的开发人员来说都是非常重要的参考书。当我想要了解或者重新回顾各种系统接口时，这本书是首选的求助工具。Stephen Rago成功地修订了本书，使其与新的操作系统（如GNU/Linux和苹果的OS X）相容，并保持了第1版易读和实用的特质。它将永远摆放在我桌上随手可及的位置。

《UNIX环境高级编程（第3版）》

——Benjamin Kuperman博士，斯沃斯莫尔学院

这是每一位严谨的UNIX C程序员必备的书籍。它深入、全面、清晰的解释是无可匹敌的。

——UniForum Monthly

从W. Richard Stevens的这本书中可以找到更多易于理解的、详尽的UNIX系统内部细节。这本书包含了大量实际的例子，对系统编程工作非常有益。

——RS/Magazine

《UNIX环境高级编程（第3版）》

作者简介

作者介绍

W. Richard Stevens，国际知名的UNIX和网络专家，备受赞誉的技术作家。生前著有多部经典的传世之作，包括《UNIX网络编程》（两卷本）、《TCP/IP详解》（三卷本）和本书第1版。

Stephen A. Rago，资深UNIX程序员，目前任NEC美国实验室存储系统集团研究员。之前是贝尔实验室的UNIX系统V版本4的开发人员之一。著有《UNIX系统V网络编程》，并曾担任本书第1版的技术审校和第2版的共同作者。

译者介绍

戚正伟，博士，上海交通大学软件学院副教授，微软亚洲研究院（2008）和美国CMU大学（2011-2012）访问学者。研究方向为系统软件和程序分析，著有《New Blue Pill深入理解硬件虚拟机》和《嵌入式GIS开发及应用》等书。

张亚英，博士，同济大学电子与信息工程学院计算机系副教授，研究方向为分布与移动计算、嵌入式系统以及系统软件等。

尤晋元，上海交通大学计算机科学及工程系教授、博士生导师。在科研方面，主要从事操作系统和分布对象计算技术方面的研究。在教学方面，长期承担操作系统及分布计算等课程的教学工作。主编和翻译了多本操作系统教材和参考书，包括《UNIX操作系统教程》、《UNIX高级编程技术》、《UNIX环境高级编程》和《操作系统：设计与实现》等。

书籍目录

第1章 UNIX基础知识

- 1.1 引言
- 1.2 UNIX体系结构
- 1.3 登录
- 1.4 文件和目录
- 1.5 输入和输出
- 1.6 程序和进程
- 1.7 出错处理
- 1.8 用户标识
- 1.9 信号
- 1.10 时间值
- 1.11 系统调用和库函数
- 1.12 小结

习题

第2章 UNIX标准及实现

- 2.1 引言
- 2.2 UNIX标准化
 - 2.2.1 ISO C
 - 2.2.2 IEEE POSIX
 - 2.2.3 Single UNIX Specification
 - 2.2.4 FIPS
- 2.3 UNIX系统实现
 - 2.3.1 SVR4
 - 2.3.2 4BSD
 - 2.3.3 FreeBSD
 - 2.3.4 Linux
 - 2.3.5 Mac OS X
 - 2.3.6 Solaris
 - 2.3.7 其他UNIX系统
- 2.4 标准和实现的关系
- 2.5 限制
 - 2.5.1 ISO C限制
 - 2.5.2 POSIX限制
 - 2.5.3 XSI限制
 - 2.5.4 函数sysconf、pathconf和fpathconf
 - 2.5.5 不确定的运行时限制
- 2.6 选项
- 2.7 功能测试宏
- 2.8 基本系统数据类型
- 2.9 标准之间的冲突
- 2.10 小结

习题

第3章 文件I/O

- 3.1 引言
- 3.2 文件描述符
- 3.3 函数open和openat
- 3.4 函数creat

- 3.5 函数close
- 3.6 函数lseek
- 3.7 函数read
- 3.8 函数write
- 3.9 I/O的效率
- 3.10 文件共享
- 3.11 原子操作
- 3.12 函数dup和dup2
- 3.13 函数sync、fsync和fdatasync
- 3.14 函数fcntl
- 3.15 函数ioctl
- 3.16 /dev/fd
- 3.17 小结
- 习题
- 第4章 文件和目录
- 4.1 引言
- 4.2 函数stat、fstat、fstatat和lstat
- 4.3 文件类型
- 4.4 设置用户ID和设置组ID
- 4.5 文件访问权限
- 4.6 新文件和目录的所有权
- 4.7 函数access和faccessat
- 4.8 函数umask
- 4.9 函数chmod、fchmod和fchmodat
- 4.10 粘着位
- 4.11 函数chown、fchown、fchownat和lchown
- 4.12 文件长度
- 4.13 文件截断
- 4.14 文件系统
- 4.15 函数link、linkat、unlink、unlinkat和remove
- 4.16 函数rename和renameat
- 4.17 符号链接
- 4.18 创建和读取符号链接
- 4.19 文件的时间
- 4.20 函数futimens、utimensat和utimes
- 4.21 函数mkdir、mkdirat和rmdir
- 4.22 读目录
- 4.23 函数chdir、fchdir和getcwd
- 4.24 设备特殊文件
- 4.25 文件访问权限位小结
- 4.26 小结
- 习题
- 第5章 标准I/O库
- 5.1 引言
- 5.2 流和FILE对象
- 5.3 标准输入、标准输出和标准错误
- 5.4 缓冲
- 5.5 打开流
- 5.6 读和写流

- 5.7 每次一行I/O
- 5.8 标准I/O的效率
- 5.9 二进制I/O
- 5.10 定位流
- 5.11 格式化I/O
- 5.12 实现细节
- 5.13 临时文件
- 5.14 内存流
- 5.15 标准I/O的替代软件
- 5.16 小结
- 习题
- 第6章 系统数据文件和信息
- 6.1 引言
- 6.2 口令文件
- 6.3 阴影口令
- 6.4 组文件
- 6.5 附属组ID
- 6.6 实现区别
- 6.7 其他数据文件
- 6.8 登录账户记录
- 6.9 系统标识
- 6.10 时间和日期例程
- 6.11 小结
- 习题
- 第7章 进程环境
- 7.1 引言
- 7.2 main函数
- 7.3 进程终止
- 7.4 命令行参数
- 7.5 环境表
- 7.6 C程序的存储空间布局
- 7.7 共享库
- 7.8 存储空间分配
- 7.9 环境变量
- 7.10 函数setjmp和longjmp
- 7.11 函数getrlimit和setrlimit
- 7.12 小结
- 习题
- 第8章 进程控制
- 8.1 引言
- 8.2 进程标识
- 8.3 函数fork
- 8.4 函数vfork
- 8.5 函数exit
- 8.6 函数wait和waitpid
- 8.7 函数waitid
- 8.8 函数wait3和wait4
- 8.9 竞争条件
- 8.10 函数exec

8.11 更改用户ID和更改组ID

8.12 解释器文件

8.13 函数system

8.14 进程会计

8.15 用户标识

8.16 进程调度

8.17 进程时间

8.18 小结

习题

第9章 进程关系

9.1 引言

9.2 终端登录

9.3 网络登录

9.4 进程组

9.5 会话

9.6 控制终端

9.7 函数tcgetpgrp、tcsetpgrp和tcgetsid

9.8 作业控制

9.9 shell执行程序

9.10 孤儿进程组

9.11 FreeBSD实现

9.12 小结

习题

第10章 信号

10.1 引言

10.2 信号概念

10.3 函数signal

10.4 不可靠的信号

10.5 中断的系统调用

10.6 可重入函数

10.7 SIGCLD语义

10.8 可靠信号术语和语义

10.9 函数kill和raise

10.10 函数alarm和pause

10.11 信号集

10.12 函数sigprocmask

10.13 函数sigpending

10.14 函数sigaction

10.15 函数sigsetjmp和siglongjmp

10.16 函数sigsuspend

10.17 函数abort

10.18 函数system

10.19 函数sleep、nanosleep和clock_nanosleep

10.20 函数sigqueue

10.21 作业控制信号

10.22 信号名和编号

10.23 小结

习题

第11章 线程

- 11.1 引言
- 11.2 线程概念
- 11.3 线程标识
- 11.4 线程创建
- 11.5 线程终止
- 11.6 线程同步
 - 11.6.1 互斥量
 - 11.6.2 避免死锁
 - 11.6.3 函数pthread_mutex_timedlock
 - 11.6.4 读写锁
 - 11.6.5 带有超时的读写锁
 - 11.6.6 条件变量
 - 11.6.7 自旋锁
 - 11.6.8 屏障
- 11.7 小结
- 习题
- 第12章 线程控制
 - 12.1 引言
 - 12.2 线程限制
 - 12.3 线程属性
 - 12.4 同步属性
 - 12.4.1 互斥量属性
 - 12.4.2 读写锁属性
 - 12.4.3 条件变量属性
 - 12.4.4 屏障属性
 - 12.5 重入
 - 12.6 线程特定数据
 - 12.7 取消选项
 - 12.8 线程和信号
 - 12.9 线程和fork
 - 12.10 线程和I/O
 - 12.11 小结
- 习题
- 第13章 守护进程
 - 13.1 引言
 - 13.2 守护进程的特征
 - 13.3 编程规则
 - 13.4 出错记录
 - 13.5 单实例守护进程
 - 13.6 守护进程的惯例
 - 13.7 客户进程-服务器进程模型
 - 13.8 小结
- 习题
- 第14章 高级I/O
 - 14.1 引言
 - 14.2 非阻塞I/O
 - 14.3 记录锁
 - 14.4 I/O多路转接
 - 14.4.1 函数select和pselect

14.4.2 函数poll

14.5 异步I/O

14.5.1 System V异步I/O

14.5.2 BSD异步I/O

14.5.3 POSIX异步I/O

14.6 函数readv和writev

14.7 函数readn和writen

14.8 存储映射I/O

14.9 小结

精彩短评

- 1、备用参考书
- 2、很权威，但有些章节讲的也是让人昏昏欲睡，总体来说是一本很不错的书，跟着书的节奏，再找一部视频，不断练习，理解起来也没什么难的，只是不知道为啥这本书评分这么高？
- 3、略读。常翻常新
- 4、高山仰止。
- 5、这书的每一句都是极为内涵，不要随便扫过，得不断拷问自己为什么是这样，那么我在开发的时候要怎么做？
- 6、超级棒的一本书，送了室友一本，希望他不要整天想着靠脸吃饭。
- 7、看了一半
- 8、看完1、3、5、7-17章，暂时先告一段落
- 9、整书看下来感觉整个c库讲解得不是很到位 针对性得API实例给得很少 这也是一大漏伤 过多得偏重参数解释 没有跟进实用 整体实用性一般般
- 10、可以挑选一些章节重点阅读，大部分内容作为查阅的资料
- 11、对于了解linux/UNIX的环境编程很有用，是一本不错的工具参考的书，另外书中还有很多对于系统调用的自己实现的源代码（C语言版本），不错
- 12、还要多翻几遍。。
- 13、好书
- 14、unix系 系统编程权威书籍。值得韦编三绝！
- 15、api手册
- 16、赶上双十二。
- 17、专业书籍。
- 18、每个程序员必读，别的书教你语法、特性，这本书会教你怎么写程序
- 19、ipc apue
- 20、五星给原作，翻译水平略水，有的句子读几遍才明白。
- 21、花了将近一个月，只有最后两章未读，代码全打了一边。书很厚，但是很好。竟然有人说不用完全读，当工具书即可，我想说，如果想学linux这本书是必读，因为就算这本书里面很多细节的东西也没有很完整的呈现，用到的时候需要更进一步的深究。所以没有勇气读完就说只需要当工具书的人是不负责的，或者你们不屑于读此书，愿意自己一点点去研究linux手册。必读！
- 22、不错的书

精彩书评

1、4.5节 p80 “ 进程每次打开、创建或删除一个文件时，内核就进行文件访问权限测试，而这种测试可能涉及文件的所有者（st_uid和st_gid）、进程的有效ID(有效用户ID和有效组ID)以及进程的附属组ID（若支持的话）。两个所有者ID是文件的性质，而两个有效ID和附属组ID则是进程的性质。” 英语原文：“ The file access tests that the kernel performs each time a process opens, creates, or deletes a file depend on the owners of the file (st_uid and st_gid), the effective IDs of the process (effective user ID and effective group ID), and the supplementary group IDs of the process, if supported. The two owner IDs are properties of the file, whereas the two effective IDs and the supplementary group IDs are properties of the process.” 这里properties这个词直译为“性质”我觉得是偷懒，没表达透实际的意思，我也是看了英文版才理解到含义。4.8节 p83“umask函数为进程设置文件模式创建屏蔽字”这是说的是什么鬼?英语原文：The umask function sets the file mode creation mask for the process. 8.11节 p205 “ 仅当对程序文件设置了设置用户ID位时，exec函数才设置有效用户ID。如果设置用户ID位没有设置，exec函数不会改变有效用户ID，而将维持其现有值。任何时候都可以调用setuid，将有效用户ID设置为实际用户ID或保存的设置用户ID。自然地，不能将有效用户ID设置为任一随机值。” 我第一次看这句话的时候完全被绕晕了，来看英语原文：The effective user ID is set by the exec functions only if the set-user-ID bit is set for the program file. If the set-user-ID bit is not set, the exec functions leave the effective user ID as its current value. We can call setuid at any time to set the effective user ID to either the real user ID or the saved set-user-ID. Naturally, we can't set the effective user ID to any random value. 有没有觉得中文版翻译就是在逗你玩儿~~effective user ID、saved set-user-ID这几个词如果不翻译的话其实读起来反而还流畅一点。8.17节把wall clock time翻译成了“墙上时钟时间”，比起“壁钟时间”差远了看wiki的解释：https://en.wikipedia.org/wiki/Wall-clock_time，这根本无法跟“墙上时钟时间”联系起来好吧

《UNIX环境高级编程（第3版）》

版权说明

本站所提供下载的PDF图书仅提供预览和简介，请支持正版图书。

更多资源请访问：www.tushu000.com