

<<Linux高级程序设计>>

图书基本信息

书名：<<Linux高级程序设计>>

13位ISBN编号：9787115213907

10位ISBN编号：7115213909

出版时间：2009-10

出版单位：人民邮电出版社

作者：杨宗德，邓玉春 编著

页数：432

字数：660000

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：<http://www.tushu007.com>

<<Linux高级程序设计>>

前言

《Linux高级程序设计》一书主要介绍Linux应用层程序开发中所涉及的系统调用，主要包括文件管理。

进程管理。

进程间通信与同步。

线程管理。

线程间同步以及网络编程等内容。

要求读者有较好的C语言及数据结构基础。

网络技术基础。

。

Linux应用开发是目前最为广泛的软件开发内容之一，同时也是从事Linux内核及驱动开发的基础。

。

《Linux高级程序设计》一书出版以来，收到了大量的读者来信，对本书提出了各种意见和建议，综合各方面的考虑，笔者做了大量的改进，推出了它的第2版。

这次修订的原则是：(1)保持和加强原书优点，如理论观点鲜明，注重实际与应用，并添加新的案例，(2)在内容上去旧更新，不仅修正了第1版中的错误和疏漏之处，更对原版的章节。

内容进行了更新和补充，力求删繁就简。

本书修订内容如下。

第1章：精简文字，修订部分笔误，更新编程错误处理办法。

第2章：精简文字，修订部分笔误，删除部分冗余内容，包括对GCC、GDB、Makefile介绍的内容。

。

增加部分工具说明。

第3章：精简文字，修订部分笔误，并对命令行参数匹配。

进程基本环境等进行介绍。

第4章：修订格式化输入输出节内容，增加scanf / sprintf函数应用内容，并增加部分应用案例，主要包括流类型的判断。

简单shell命令(例如cp)实现等。

第5章：对各函数说明及应用案例进行更新，并增加目录流编程内容。

第6章：对链接文件操作函数进行分类归纳整理，并以ls-1应用案例总结第4、5、6章内容。

第7章：对进程创建。

进程属性等内容进行修订，添加守候进程以及日志管理的内容，并对孤儿进程。

僵死进程进行比较。

第8章：精简文字，修订部分笔误，添加了重定向案例，并对信号处理一节进行了总结，更新了各函数的应用案例。

第9章：增加双向队列传递应用案例。

信号量生产消费问题案例以及信号量的非阻塞应用案例。

第10章：精简文字，修订部分笔误，增加线程私有数据介绍以及各函数应用代码。

第11章：精简文字，修订部分笔误。

增加线程在信号处理方面的内容。

第12章：精简文字，修订部分笔误。

增加UDP。

TCP同步及异步通信案例。

新增第13章：对网络编程部分专用函数，包括大小端与字节顺序。

socket属性。

地址解析。

网络调试工具等相关内容进行介绍。

<<Linux高级程序设计>>

新增第14章：增加对TCP及UDP高级编程的介绍，包括TCP多路选择。
TCP非阻塞。
信号驱动。
UDP广播。
UDP组播等内容。

新增第15章：增加网络服务器构建案例，并以构建HTTP服务器为例，介绍了包括文件及目录管理，进行线程管理及通信。
网络编程内容，从而使读者学以致用。

在第2版中还增加了大量应用案例，特别是增加了大量网络编程内容，以求为读者进行Linux应用程序开发提供一条扎实的进阶之路。本书还提供了完整的代码和教案，方便广大师生使用。

<<Linux高级程序设计>>

内容概要

本书以Linux操作系统（内核为2.6版本）为开发平台、GCC 4.0/GDB 6.3为开发调试环境，详细介绍了Linux系统下编程环境及编程工具、文件管理（文件类型、ANSI以及POSIX标准下文件读写操作）、进程管理（创建、退出、执行、等待、属性控制）、进程间通信（管道、消息队列、共享内存）、进程间同步机制（信号量）、进程间异步机制（信号）、线程管理（创建、退出、取消等以及属性控制）、线程间同步（互斥锁、读写锁、条件变量）以及网络基本编程、高级应用等内容。

本书内容丰富、紧扣应用，适合从事Linux下C应用编程的人员阅读，也适合从事嵌入式Linux开发的人员阅读。

<<Linux高级程序设计>>

书籍目录

第1章 Linux下C语言开发环境	1.1 Linux操作系统简介	1.1.1 Linux操作系统简介
1.1.2 GNU/Linux简介	1.1.3 相关术语介绍	1.2 Linux开发初步
1.2.1 Linux下C程序标准	1.2.2 库函数和系统调用	1.2.3 在线文档介绍
1.2.4 获取错误信息	1.3 部分常用工具简介	1.3.1 tar打包器
1.3.2 Linux常用命令及工具	1.4 Linux下编码风格	1.4.1 GNU编码规范
1.4.2 Linux内核编码规范	第2章 Linux下C语言开发工具	2.1 常用编辑工具
2.1.1 VIM编辑器	2.1.2 Emacs编辑器	2.1.3 Source Insight工具
2.2 GCC编译工具	2.2.1 GCC/G++简介	2.2.2 头文件及预处理结果分析
2.3 Make工具与Makefile文件	2.3.1 Make工具简介	2.3.2 Makefile常用规则
2.4 常用调试工具	2.4.1 GDB调试工具简介	2.4.2 GDB演示示例
2.4.3 GDB调试器常用语法	2.4.4 strace	2.4.5 GCC程序开发过程实例
2.5 Linux库文件使用与创建	2.5.1 Linux系统库文件管理的基本策略	2.5.2 Linux下静态库的创建与使用
2.5.3 Linux下共享库创建及使用	2.5.4 静态库与共享库的区别	2.6 Autoconf/Automake自动化工具
2.6.1 Autoconf/Automake工具介绍	2.6.2 Autoconf/Automake工具使用示例	第3章 Linux进程存储管理
3.1 Linux程序存储结构与进程结构	3.1.1 Linux可执行文件结构	3.1.2 Linux进程结构
3.1.3 C变量及函数存储类型	3.1.4 栈和堆的区别	3.1.5 示例：查看代码中各数据存储位置
3.1.6 常见内存错误示例分析	3.2 ANSI C内存管理API函数	3.2.1 内存分配的基本方式
3.2.2 示例：为程序申请动态内存空间	3.2.3 内存数据管理函数	3.3 常用Linux内存管理及调试工具
3.3.1 mcheck函数	3.3.2 Valgrind内存检测工具	3.4 Linux进程环境及系统限制
3.4.1 进程与命令参数	3.4.2 进程与环境变量	3.4.3 Linux系统限制
3.4.4 Linux时间管理	第4章 ANSI C文件管理	第5章 POSIX文件及目录管理
第6章 普通文件、连接文件及目录文件属性管理	第7章 Linux进程管理与程序开发	第8章 进程间通信——管道和信号
第9章 System V进程间通信	第10章 Linux多线程编程	第11章 线程间同步机制
第12章 Linux socket网络编程	第13章 网络编程工具介绍	第14章 网络编程高级应用
第15章 构建网络服务器		

章节摘录

第1章 Linux下C语言开发环境 1.1 Linux操作系统简介 1.1.1 Linux操作系统简介

UNIX操作系统于1969年由Ken Thompson在AT&T贝尔实验室的一台DEC PDP-7计算机上实现。后来Ken Thompson和Dennis Ritchie使用C语言对整个系统进行了再加工和编写，使得UNIX能够很容易地移植到其他硬件的计算机上。

由于此时AT&T还没有把UNIX作为它的正式商品，因此研究人员只是在实验室内部使用并完善它。正是由于UNIX是被作为研究项目，其他科研机构 and 大学的计算机研究人员也希望能得到这个系统，以便进行自己的研究。

AT&T采用分发许可证的方法，大学和科研机构仅仅需要很少的费用就能获得UNIX的源代码以进行研究。

UNIX的源代码被散发到各个大学，一方面使得科研人员能够根据需要改进系统，或者将其移植到其他硬件环境中去，另一方面培养了大量懂得UNIX使用和编程的学生，这使UNIX的使用更为普及。

到了20世纪70年代末，在UNIX发展到版本6之后，AT&T认识到了UNIX的价值，并成立了UNIX系统实验室（UNIX System Lab，USL）来继续发展UNIX。

因此一方面AT&T继续发展内部使用的UNIX版本7，一方面由USL开发对外正式发行的UNIX版本，同时AT&T也宣布对UNIX产品拥有所有权。

几乎在同时，加州大学伯克利分校计算机系统研究小组（CSRG）借助UNIX对操作系统进行了研究，他们对UNIX进行的改进相当多，增加了很多当时非常先进的特性，包括更好的内存管理、快速且健壮的文件系统等，大部分原有的源代码都被重写，很多其他的UNIX使用者，包括其他大学和商业机构，都希望能得到经CSRG改进的UNIX系统。

因此CSRG的研究人员把他们的UNIX组成一个完整的UNIX系统——BSD UNIX（Berkeley Software Distribution）向外发行。

<<Linux高级程序设计>>

编辑推荐

《Linux高级程序设计(第2版)》主要涵养了Linux下C应用层程序开发的大部分内容。编程基本工具GCC / GDB的使用方法, 库文件的管理。

ANSI标准I / O管理、POSIX标准I / O管理、普通文件和目录管理以及内存管理。

进程管理及进程间通信机制, 包括同一台主机通信机制(无名管道、命名管道、信号、信号量、消息队列以及共享内存), 网络主机间通信机制(socket)。

线程管理及线程间通信机制, 包括互斥锁、条件变量、信号灯、信号和读写锁。

由浅入深, 按理论分析 实际操作 案例分析的顺序组织内容 作者多年教学及工程实践的总结, 整合了Linux程序设计的绝大多数知识点, 涵盖Linux操作系统下C语言应用程序设计的所有关键内容

内容丰富 总结作者多年教学及工程经验, 整合Linux程序设计的绝大多数知识点, 涵盖Linux操作系统下C语言应用程序设计的所有关键内容。

循序渐进避开大量理论, 按知识体系介绍 应用函数分析 应用案例开发的顺序写作, 读者可以针对每个知识点进行编程演练。

案例指导 《Linux高级程序设计(第2版)》遵循案例教学思想, 每一个知识点都对应一个应用程序, 且所有代码都在教学实践过程中使用过, 读者可以直接编译调试。

紧扣应用 《Linux高级程序设计(第2版)》所采用的开发平台为2.6内核, 开发工具为GCC 4.0。

<<Linux高级程序设计>>

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:<http://www.tushu007.com>