



图灵原版计算机科学系列



Advanced Programming in the UNIX Environment  
Second Edition

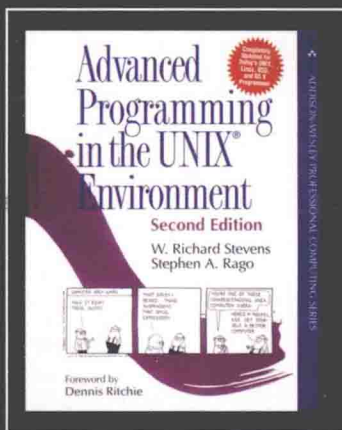
# UNIX 环境高级编程

(英文版 · 第2版)

[美]

W. Richard Stevens  
Stephen A. Rago

著



人民邮电出版社  
POSTS & TELECOM PRESS

**TURING** 图灵原版计算机科学系列

Advanced Programming in the UNIX Environment

Second Edition

# UNIX环境高级编程

(英文版·第2版)

[美] W. Richard Stevens 著  
Stephen A. Rago

 人民邮电出版社  
POSTS & TELECOM PRESS

## 图书在版编目 (CIP) 数据

UNIX 环境高级编程：第 2 版 / (美) 史蒂文斯, (美) 拉戈著.

—北京：人民邮电出版社，2006.2

(图灵原版计算机科学系列)

ISBN 7-115-14484-2

I. U... II. ①史...②拉... III. UNIX 操作系统—程序设计—教材—英文 IV. TP316.81

中国版本图书馆 CIP 数据核字 (2006) 第 005691 号

## 内 容 提 要

本书是被誉为UNIX编程“圣经”的*Advanced Programming in the UNIX Environment*一书的更新版。在本书第1版出版后的十几年中，UNIX行业已经有了巨大的变化，特别是影响UNIX编程接口的有关标准变化很大。本书在保持了前一版的风格的基础上，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多个应用示例，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。

本书内容权威，概念清晰，阐述精辟，对于所有层次UNIX程序员都是一本不可或缺的参考书。

图灵原版计算机科学系列

### UNIX 环境高级编程 (英文版·第 2 版)

- 
- ◆ 著 [美] W.Richard Stevens Stephen A.Rago  
责任编辑 杨海玲
  - ◆ 人民邮电出版社出版发行 北京市崇文区夕照寺街 14 号  
邮编 100061 电子函件 315@ptpress.com.cn  
网址 <http://www.ptpress.com.cn>  
北京顺义振华印刷厂印刷  
新华书店总店北京发行所经销
  - ◆ 开本：800×1000 1/16  
印张：59.25  
字数：1330 千字 2006 年 2 月第 1 版  
印数：1-3 000 册 2006 年 2 月北京第 1 次印刷

著作权合同登记号 图字：01-2005-5228 号

ISBN 7-115-14484-2/TP·5214

定价：99.00 元

读者服务热线：(010)88593802 印装质量热线：(010)67129223

# 版权声明

Original edition, entitled *Advanced Programming in the UNIX Environment, Second Edition*, 0201433079 by W. Richard Stevens and Stephen A. Rago, published by Pearson Education, Inc., publishing as Addison-Wesley, Copyright © 2005 by Pearson Education, Inc.

All rights reserved. No part of this book may be reproduced or transmitted in any form or by any means, electronic or mechanical, including photocopying, recording or by any information storage retrieval system, without permission from Pearson Education, Inc.

China edition published by PEARSON EDUCATION ASIA LTD and POSTS & TELECOM PRESS Copyright © 2006.

This edition is manufactured in the People's Republic of China, and is authorized for sale only in the People's Republic of China excluding Hong Kong, Macao and Taiwan.

本书英文版由Pearson Education Asia Ltd.授权人民邮电出版社独家出版。未经出版者书面许可，不得以任何方式复制或抄袭本书内容。

仅限于中华人民共和国境内（香港、澳门特别行政区和台湾地区除外）销售发行。

本书封面贴有Pearson Education（培生教育出版集团）激光防伪标签，无标签者不得销售。

版权所有，侵权必究。

# 序

我差不多每次在接受专访当中，或是做技术讲座后的提问时间里，总会被问及这样一个问题：“你想到过UNIX会生存这么长时间吗？”自然，每次的回答都是：没有，我们没想到会是这样。从某种角度说，UNIX系统已经伴随了商用计算行业历史的大半，而这也早就不是什么新闻了。

发展的历程错综复杂，充满变数。自20世纪70年代初以来，计算机技术经历了沧海桑田般的变化，尤其体现在网络技术的普遍应用、图形化的无所不在、个人计算的触手可及，然而UNIX系统却奇迹般地容纳和适应了所有这些变化。虽然商业应用环境在桌面领域目前仍然为微软和英特尔所统治，但是在某些方面已经从单一供应商向多种来源转变，近年来对公共标准和免费开放资源的信赖已经与日俱增。

UNIX作为一种现象而不单是商标品牌，有幸能与时俱进，乃至领导潮流。在上个世纪70~80年代，AT&T虽对UNIX的实际源代码进行了版权保护，但却鼓励在系统的接口和语言基础上进行标准化的工作。例如，AT&T发布了SVID (System V Interface Definition, 系统V接口定义)，这成为POSIX及其后续工作的基础。后来，UNIX可以说相当优雅地适应了网络环境，虽不那么轻巧却也充分地适应了图形环境。再往后，开源运动的技术基础中集成了UNIX的基本内核接口和许多它独特的用户级工具。

即使在UNIX软件系统本身还是专有的时候，鼓励出版UNIX系统方面的论文和书籍也是至关重要的，著名的例子就是Maurice Bach的《UNIX操作系统设计》一书。其实我要说明的是，UNIX长寿的主要原因是，它吸引了极具天分的技术作者，为大众解读它的优美和神秘所在。Brian Kernighan是其中之一，Rich Stevens自然也是。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。

然而，本书第1版毕竟出版时间太早了，那时还没有出现Linux，源自伯克利 CSRG的UNIX接口的开源版本还没有广为流行，很多人的网络还在用串行调制解调器。Steve Rago认真仔细地更新了本书，以反映所有这些技术进展，同时还考虑到各种ISO标准和IEEE标准这些年来来的变化。因此，他的例子是最新的，也是最新测试过的。

总之，这是一本弥足珍贵的经典著作的更新版。

Dennis Ritchie

2005年3月于新泽西州Murray Hill市

# 前言

## 引言

我与Rich Stevens最早是通过电子邮件开始交往的，当时我发邮件报告他的第一本书《UNIX网络编程》的一个排版错误。他回信开玩笑说我是第一个给他发这本书勘误的人。到他1999年故去之前，我们时不时地会通些邮件，一般都是在有了问题认为对方能解答的时候。我们在USENIX会议期间多次相见，并共进晚餐，Rich在会议中给大家做技术培训。

Rich Stevens真是个益友，行为举止很有绅士风度。我在1993年写《UNIX系统V网络编程》时，试图把书写成他的《UNIX网络编程》的系统V版。Rich发自内心地高兴地为 我审阅了好几章，并不把我当成竞争对手，而是当作一起写书的同事。我们曾多次谈到要合作给他的《TCP/IP详解》写个STREAMS版。天若有情，我们或许已经完成了这个心愿。然而，Rich已经驾鹤西去，修订《UNIX环境高级编程》就成为我跟他一起写书的最易实现的方式。

当Addison-Wesley公司的编辑找到我说想修订Rich的这本书时，我第一反应是这本书没有多少要改的。尽管13年过去了，Rich的书还是巍然屹立。但是，与当初本书出版的时候相比，今日的UNIX行业已经有了巨大的变化。

- 系统V的各个变种渐渐被Linux所取代。原来生产硬件配以各自的UNIX版本的几个主要厂商，要么提供了Linux的移植版本，要么宣布支持Linux。Solaris可能算是硕果仅存的占有一定市场份额的UNIX系统V版本4的后裔了。
- 加州大学伯克利分校的CSRG（计算机科学研究组）在发布了4.4BSD之后，已经决定不再开发UNIX操作系统，只有几个志愿者小组还维护着一些可公开获得的版本。
- Linux受到数以千计的志愿者的支持，它的引入使任何一个拥有计算机的人都能运行类似于UNIX系统的操作系统，并且可以免费获得源代码支持哪怕最新的硬件设备。在已经存在几种免费BSD版本的情况下，Linux的成功确实是个奇迹。
- 苹果公司作为一个富有创新精神的公司，已经放弃了老的Mac操作系统，换之以一个在Mach和FreeBSD基础上开发的新系统。

因此，我已经努力更新本书中的内容，以反映这4个平台。

Rich在1992年出版《UNIX环境高级编程》之后，我扔掉了手头几乎所有的UNIX程序员手册。这些年来，我桌上最常摆放的就是两本书，一本是字典，另一本就是《UNIX环境高级编程》。我希望读者也能认为本修订版一样有用。

## 对第1版的改动

Rich的书依然屹立，我试图不去改动他这本书原来的风格。但是13年间世事兴衰，尤其是影响UNIX编程接口的有关标准变化很大。

我依据标准化组织的标准，更新了全书相关的接口方面的内容。第2章改动较大，因为它主要是讨论标准的。本书第1版是根据POSIX.1标准的1990年版写的，本修订版依据2001年版的新标准，内容要丰富很多。1990年ISO的C标准在1999年也更新了，有些改动影响到POSIX.1标准中的接口。

目前的POSIX.1规范涵盖了更多的接口。The Open Group（原称X/Open）发布的“Single UNIX Specification”的基本规范现在已经并入POSIX.1，后者包含了几个1003.1标准和另外几个标准草案，原来这些标准是分开出版的。

我也相应地增加了些章节，讨论新主题。线程和多线程编程是相当重要的概念，因为它们为程序员处理并发和异步提供了更清楚的方式。

套接字接口现在也是POSIX.1的一部分了。它为进程间通信（IPC）提供了单一的接口，而不考虑进程的位置。它成为IPC章节的自然扩展。

我省略了POSIX.1中的大部分实时接口。这些内容最好是在一本专门讲述实时编程的书中介绍。参考文献里有一本这方面的书。

我把最后几章的案例研究也更新了，用了更接近现实的例子。例如，现在很少有系统通过串口或并口连接PostScript打印机了，多数PostScript打印机是通过网络连接的，所以我对PostScript打印机通信的例子做了修改。

有关调制解调器通信的那一章如今已经不太适用了。原始材料我们保留在本书网站上，有两种格式：PostScript (<http://www.apuebook.com/lostchapter/modem.ps>) 和PDF (<http://www.apuebook.com/lostchapter/modem.pdf>)。

书中示例的源代码也可以从[www.apuebook.com](http://www.apuebook.com)上获得。多数示例已经在下述4种平台上运行过了：

(1) FreeBSD 5.2.1，这是加州大学伯克利分校CSRG的4.4BSD的一个变种，在英特尔奔腾处理器上运行。

(2) Linux 2.4.22 (Mandrake 9.2发布)，是一个免费的类UNIX操作系统，运行于英特尔奔腾处理器上。

(3) Solaris 9，是Sun公司系统V版本4的变种，运行于64位的UltraSPARC III处理器上。

(4) Darwin 7.4.0，是基于FreeBSD和Mach的操作系统环境，也是Apple Mac OS X 10.3版本的核心，运行于PowerPC处理器上。

## 致谢

（首先要感谢）Rich Stevens独立创作了本书第1版，它立即成为一本经典著作。

没有家人的支持，我不可能修订此书。他们容忍我满屋子散落稿纸（比平常还甚），霸占了家里的好几台机器，成天埋头于电脑屏幕前。我的妻子Jeanne甚至亲自动手帮我在一台测试的机器上安装了Linux。

多名技术审校者提出了很多改进意见，确保内容准确。我非常感谢David Bausum、David Boreham、Keith Bostic、Mark Ellis、Phil Howard、Andrew Josey、Mukesh Kacker、Brian Kernighan、Bengt Kleberg、Ben Kuperman、Eric Raymond和Andy Rudoff。

我还要谢谢Andy Rudoff给我解答有关Solaris的问题，谢谢Dennis Ritchie不惜花时间从故纸堆中为我寻找有关历史方面问题的答案。再次谢谢Addison-Wesley公司的员工，与他们合作令人愉快，谢谢Tyrrell Albaugh、Mary Franz、John Fuller、Karen Gettman、Jessica Goldstein、Noreen Regina和John Wait。特别感谢Evelyn Pyle细致地编辑了本书。

就像Rich曾经做到的那样，我非常欢迎读者发来邮件，发表评论，提出建议，订正错误。

Stephen A.Rago

sar@apuebook.com

2005年4月于新泽西州Warren市



# 第1版前言

## 引言

本书描述了UNIX系统的程序设计接口——系统调用接口和标准C库提供的很多函数。本书针对的是所有的程序员。

与大多数操作系统一样，UNIX为程序运行提供了大量的服务——打开文件，读文件，启动一个新程序，分配存储区以及获得当前时间等。这些服务被称为系统调用接口（system call interface）。另外，标准C库提供了大量广泛用于C程序中的函数（格式化输出变量的值，比较两个字符串等）。

系统调用接口和库函数可参见《UNIX程序员手册》第2、3部分。本书不是这些内容的重复。手册中没有给出示例及基本原理，而这些则正是本书所要讲述的内容。

## UNIX标准

20世纪80年代出现了各种版本的UNIX，20世纪80年代后期在此基础上制定了数个国际标准，包括C程序设计语言的ANSI标准、IEEE POSIX标准系列（还在制定中）、X/Open可移植性指南。

本书也介绍了这些标准，但是并不只是说明标准本身，而是着重说明它们与应用广泛的一些实现（主要指SVR4以及即将发布的4.4BSD）之间的关系。这是一种贴近现实世界的描述，而这正是标准本身以及仅描述标准的文献所缺少的。

## 本书的组织

本书分为6个部分：

(1) 对UNIX程序设计基本概念和术语的简要描述（第1章），以及对各种UNIX标准化工作和不同UNIX实现的讨论（第2章）。

(2) I/O——不带缓存的I/O（第3章）、文件和目录（第4章）、标准I/O库（第5章）和标准系统数据文件（第6章）。

(3) 进程——UNIX进程的环境（第7章）、进程控制（第8章）、进程之间的关系（第9章）和信号（第10章）。

(4) 更多的I/O——终端I/O（第11章）、高级I/O（第12章）和守护进程（第13章）。

(5) IPC——进程间通信（第14和15章）。

(6) 示例——一个数据库的函数库 (第16章)、与PostScript 打印机的通信 (第17章)、调制解调器拨号程序 (第18章) 和使用伪终端 (第19章)。

如果对C语言较熟悉并具有某些应用UNIX的经验,对学习本书将非常有益,但是并不要求读者必须具有UNIX编程经验。本书面向的读者主要是:熟悉UNIX的程序员和熟悉其他某个操作系统且希望了解大多数UNIX系统提供的各种服务细节的程序员。

## 本书中的示例

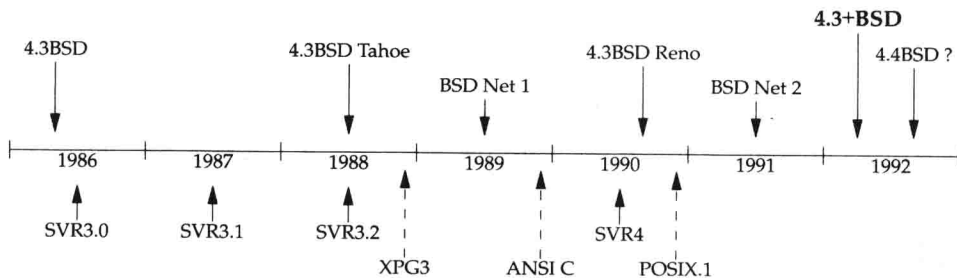
本书包含了大量示例——大约10 000行源代码。所有示例都用ANSI C语言编写。在阅读本书时,建议准备一本你所使用的UNIX系统的《UNIX程序员手册》,在细节方面有时需要参考该手册。

几乎对于每一个函数和系统调用,本书都用一个小的完整的程序进行了演示。这可以让读者清楚地了解它们的用法,包括参数和返回值等。有些小程序还不足以说明库函数和系统调用的复杂功能和应用技巧,所以书中还包含了一些较大的示例(见第16~19章)。

所有示例的源代码文件都可在因特网上用匿名ftp从因特网主机ftp.uu.net的published/books/stevens.advprog.tar.Z文件下载。读者可以在自己的机器上修改并运行这些源代码。

## 用于测试示例的系统

不幸的是,所有的操作系统都在不断变更,UNIX也不例外。下图给出了系统V和4.xBSD最近的进展情况。



4.xBSD是由加州大学伯克利分校CSRG开发的。该小组还发布了BSD Net1和BSD Net2版,其公开的源代码源自4.xBSD系统。SVR<sub>x</sub>表示AT&T的系统V第x版。XPG3指X/Open可移植性指南的第3个发行版。ANSI C是C语言的ANSI标准。POSIX.1是IEEE和ISO的类UNIX系统接口标准。2.2节和2.3节将对这些标准和不同版本之间的差别做更多的说明。

本书中用4.3+BSD表示源自伯克利的介于BSD Net2和4.4BSD之间的UNIX系统。

在本书写作时,4.4BSD尚未发布,所以不能称一个系统是4.4BSD的。为了用一个简单的名字来引用该系统,故使用4.3+BSD。

本书中的大多数示例曾在下面4种UNIX系统上运行过:

(1) U.H公司 (UHC) 的UNIX系统V/386 R4.0.2 (vanilla SVR4), 运行于Intel 80386处理器上。

(2) 加州大学伯克利分校CSRG的4.3+BSD, 运行于惠普工作站上。

(3) 伯克利软件设计公司的BSD/386 (是BSD Net2的变种), 运行于Intel 80386处理器上。该系统与4.3+BSD几乎相同。

(4) Sun 公司的SunOS 4.1.1和4.1.2 (该系统与伯克利系统有很深的渊源, 但也包含了许多系统V的特性), 运行于SPARCstation SLC上。

本书还提供了许多时间测试及用于测试的实际系统。

## 致谢

在过去的一年半中, 家人给予了我大力支持和爱, 因为写书我们失去了很多快乐的周末, 我深感歉疚。写书从许多方面影响了整个家庭。谢谢Sally、Bill、Ellen和David。

我要特别感谢Brian Kernighan对我写作此书的帮助。他审阅了全部书稿, 不但提出了大量有洞察力的技术意见, 还委婉地指出了多处修辞问题, 但愿我能够在最终成稿中已经加以体现。Steve Rago也成为了我的创作源泉, 不但审阅了全部书稿, 还为我解答了有关系统V的许多技术细节和历史问题。还要感谢Addison-Wesley公司邀请的其他技术审校者, 他们对书稿的各个部分提出了很有价值的意见, 他们是Maury Bach、Mark Ellis、Jeff Gitlin、Peter Honeyman、John Linderman、Doug McIlroy、Evi Nemeth、Craig Patridge、Dave Presotto、Gary Wilson、Gary Wright。

(感谢) 加州大学伯克利分校CSRG的Keith Bostic和Kirk McKusick给了我一个帐号, 可在最新的BSD系统上测试书中示例。(也要感谢Peter Salus) UHC的Sam Nataros和Joachim Sacksen给我提供了一份SVR4, 用来测试书中例子。Trent Hein则帮助我获得BSD/386的alpha和beta版。

其他朋友在过去这些年以各种方式提供了帮助, 看似不大, 却非常重要。他们是Paul Lucchina、Joe Godsil、Jim Hogue、Ed Tankus和Gary Wright。我的编辑是Addison-Wesley公司的John Wait, 自始至终是我的忠实朋友。我不断地延期交稿, 写作篇幅也一再超过计划, 他从不抱怨。特别还要感谢美国国家光学天文台 (NOAO), 尤其是Sidney Wolff、Richard Wolff和Steve Grandi, 为我提供准确的计算机时间。

真正的UNIX书应该用troff写成, 本书也遵循了这一优秀传统。最终清样是作者用James Clark写的groff软件包做出的。非常感谢James Clark提供了这个优异的写作软件, 并迅速地修正其中所发现的bug。也许有一天我会最终弄清楚troff软件做页脚的技巧。

我十分欢迎读者发来电子邮件, 发表评论, 提出建议, 订正错误。

W.Richard Stevens  
rstevens@kohala.com

<http://www.kohala.com/~rstevens>

1992年4月于亚利桑那州塔克森市

# Contents

<b>Chapter 1.</b>	<b>UNIX System Overview</b>	<b>1</b>
1.1	Introduction	1
1.2	UNIX Architecture	1
1.3	Logging In	2
1.4	Files and Directories	4
1.5	Input and Output	8
1.6	Programs and Processes	10
1.7	Error Handling	14
1.8	User Identification	16
1.9	Signals	18
1.10	Time Values	20
1.11	System Calls and Library Functions	21
1.12	Summary	23
<b>Chapter 2.</b>	<b>UNIX Standardization and Implementations</b>	<b>25</b>
2.1	Introduction	25
2.2	UNIX Standardization	25
2.2.1	ISO C	25
2.2.2	IEEE POSIX	26
2.2.3	The Single UNIX Specification	29
2.2.4	FIPS	33
2.3	UNIX System Implementations	33
2.3.1	UNIX System V Release 4	33
2.3.2	4.4BSD	34
2.3.3	FreeBSD	35
2.3.4	Linux	35

---

2.3.5	Mac OS X	35	
2.3.6	Solaris	35	
2.3.7	Other UNIX Systems	36	
2.4	Relationship of Standards and Implementations	36	
2.5	Limits	36	
2.5.1	ISO C Limits	38	
2.5.2	POSIX Limits	38	
2.5.3	XSI Limits	40	
2.5.4	sysconf, pathconf, and fpathconf Functions	41	
2.5.5	Indeterminate Runtime Limits	48	
2.6	Options	52	
2.7	Feature Test Macros	55	
2.8	Primitive System Data Types	56	
2.9	Conflicts Between Standards	56	
2.10	Summary	58	
<b>Chapter 3.</b>	<b>File I/O</b>		<b>59</b>
3.1	Introduction	59	
3.2	File Descriptors	59	
3.3	open Function	60	
3.4	creat Function	62	
3.5	close Function	63	
3.6	lseek Function	63	
3.7	read Function	67	
3.8	write Function	68	
3.9	I/O Efficiency	68	
3.10	File Sharing	70	
3.11	Atomic Operations	74	
3.12	dup and dup2 Functions	76	
3.13	sync, fsync, and fdatasync Functions	77	
3.14	fcntl Function	78	
3.15	ioctl Function	83	
3.16	/dev/fd	84	
3.17	Summary	85	
<b>Chapter 4.</b>	<b>Files and Directories</b>		<b>87</b>
4.1	Introduction	87	
4.2	stat, fstat, and lstat Functions	87	
4.3	File Types	88	
4.4	Set-User-ID and Set-Group-ID	91	
4.5	File Access Permissions	92	

4.6	Ownership of New Files and Directories	95
4.7	access Function	95
4.8	umask Function	97
4.9	chmod and fchmod Functions	99
4.10	Sticky Bit	101
4.11	chown, fchown, and lchown Functions	102
4.12	File Size	103
4.13	File Truncation	105
4.14	File Systems	105
4.15	link, unlink, remove, and rename Functions	108
4.16	Symbolic Links	112
4.17	symlink and readlink Functions	115
4.18	File Times	115
4.19	utime Function	116
4.20	mkdir and rmdir Functions	119
4.21	Reading Directories	120
4.22	chdir, fchdir, and getcwd Functions	125
4.23	Device Special Files	127
4.24	Summary of File Access Permission Bits	130
4.25	Summary	130
<b>Chapter 5.</b>	<b>Standard I/O Library</b>	<b>133</b>
5.1	Introduction	133
5.2	Streams and FILE Objects	133
5.3	Standard Input, Standard Output, and Standard Error	135
5.4	Buffering	135
5.5	Opening a Stream	138
5.6	Reading and Writing a Stream	140
5.7	Line-at-a-Time I/O	142
5.8	Standard I/O Efficiency	143
5.9	Binary I/O	145
5.10	Positioning a Stream	147
5.11	Formatted I/O	149
5.12	Implementation Details	153
5.13	Temporary Files	155
5.14	Alternatives to Standard I/O	159
5.15	Summary	159
<b>Chapter 6.</b>	<b>System Data Files and Information</b>	<b>161</b>
6.1	Introduction	161
6.2	Password File	161

---

6.3	Shadow Passwords	165	
6.4	Group File	166	
6.5	Supplementary Group IDs	167	
6.6	Implementation Differences	169	
6.7	Other Data Files	169	
6.8	Login Accounting	170	
6.9	System Identification	171	
6.10	Time and Date Routines	173	
6.11	Summary	177	
<b>Chapter 7.</b>	<b>Process Environment</b>		<b>179</b>
7.1	Introduction	179	
7.2	main Function	179	
7.3	Process Termination	180	
7.4	Command-Line Arguments	185	
7.5	Environment List	185	
7.6	Memory Layout of a C Program	186	
7.7	Shared Libraries	188	
7.8	Memory Allocation	189	
7.9	Environment Variables	192	
7.10	setjmp and longjmp Functions	195	
7.11	getrlimit and setrlimit Functions	202	
7.12	Summary	206	
<b>Chapter 8.</b>	<b>Process Control</b>		<b>209</b>
8.1	Introduction	209	
8.2	Process Identifiers	209	
8.3	fork Function	211	
8.4	vfork Function	216	
8.5	exit Functions	218	
8.6	wait and waitpid Functions	220	
8.7	waitid Function	226	
8.8	wait3 and wait4 Functions	227	
8.9	Race Conditions	227	
8.10	exec Functions	231	
8.11	Changing User IDs and Group IDs	237	
8.12	Interpreter Files	242	
8.13	system Function	246	
8.14	Process Accounting	250	
8.15	User Identification	256	

8.16	Process Times	257	
8.17	Summary	259	
<b>Chapter 9.</b>	<b>Process Relationships</b>		<b>261</b>
9.1	Introduction	261	
9.2	Terminal Logins	261	
9.3	Network Logins	266	
9.4	Process Groups	269	
9.5	Sessions	270	
9.6	Controlling Terminal	272	
9.7	tcgetpgrp, tcsetpgrp, and tcgetsid Functions		273
9.8	Job Control	274	
9.9	Shell Execution of Programs	278	
9.10	Orphaned Process Groups	282	
9.11	FreeBSD Implementation	285	
9.12	Summary	287	
<b>Chapter 10.</b>	<b>Signals</b>		<b>289</b>
10.1	Introduction	289	
10.2	Signal Concepts	289	
10.3	signal Function	298	
10.4	Unreliable Signals	301	
10.5	Interrupted System Calls	303	
10.6	Reentrant Functions	305	
10.7	SIGCLD Semantics	308	
10.8	Reliable-Signal Terminology and Semantics		310
10.9	kill and raise Functions	311	
10.10	alarm and pause Functions	313	
10.11	Signal Sets	318	
10.12	sigprocmask Function	320	
10.13	sigpending Function	322	
10.14	sigaction Function	324	
10.15	sigsetjmp and siglongjmp Functions		329
10.16	sigsuspend Function	333	
10.17	abort Function	340	
10.18	system Function	342	
10.19	sleep Function	347	
10.20	Job-Control Signals	349	
10.21	Additional Features	352	
10.22	Summary	353	



---

<b>Chapter 11.</b>	<b>Threads</b>	<b>355</b>
11.1	Introduction	355
11.2	Thread Concepts	355
11.3	Thread Identification	356
11.4	Thread Creation	357
11.5	Thread Termination	360
11.6	Thread Synchronization	368
11.7	Summary	385
<b>Chapter 12.</b>	<b>Thread Control</b>	<b>387</b>
12.1	Introduction	387
12.2	Thread Limits	387
12.3	Thread Attributes	388
12.4	Synchronization Attributes	393
12.5	Reentrancy	401
12.6	Thread-Specific Data	406
12.7	Cancel Options	410
12.8	Threads and Signals	413
12.9	Threads and <code>fork</code>	416
12.10	Threads and I/O	420
12.11	Summary	420
<b>Chapter 13.</b>	<b>Daemon Processes</b>	<b>423</b>
13.1	Introduction	423
13.2	Daemon Characteristics	423
13.3	Coding Rules	425
13.4	Error Logging	428
13.5	Single-Instance Daemons	432
13.6	Daemon Conventions	434
13.7	Client-Server Model	439
13.8	Summary	439
<b>Chapter 14.</b>	<b>Advanced I/O</b>	<b>441</b>
14.1	Introduction	441
14.2	Nonblocking I/O	441
14.3	Record Locking	444
14.4	STREAMS	460
14.5	I/O Multiplexing	472
14.5.1	<code>select</code> and <code>pselect</code> Functions	474