

PEARSON
Addison
Wesley

FreeBSD

操作系统 设计与实现

The Design
and Implementation
of the FreeBSD
Operating System

[美] Marshall Kirk McKusick 著
George V. Neville-Neil 译
张辉



人民邮电出版社
POSTS & TELECOM PRESS

FreeBSD 操作系统设计与实现

The Design and Implementation of the FreeBSD Operating System

[美] Marshall Kirk McKusick 著
George V. Neville-Neil
张 辉 译

人民邮电出版社

图书在版编目 (CIP) 数据

FreeBSD 操作系统设计与实现 / (美) 麦库西克 (Mckusick, M. K.), (美) 内维尔-尼尔 (Neville-Neil, G. V.) 著; 张辉译. —北京: 人民邮电出版社, 2006.2

ISBN 7-115-13685-8

I. F... II. ①麦... ②内... ③张... III. UNIX 操作系统 IV. TP316.81

中国版本图书馆 CIP 数据核字 (2005) 第 111073 号

版 权 声 明

Simplified Chinese Edition Copyright © 2005 by PEARSON EDUCATION ASIA LIMITED and POSTS&TELECOMMUNICATIONS PRESS.

The Design and Implementation of the FreeBSD Operating System ISBN: 0-201-70245-2

By Marshall Kirk McKusick George V.Neville-Neil

Copyright © 2005.

All Rights Reserved.

Published by arrangement with Addison Wesley Longman, Pearson Education, Inc.

The edition is authorized for sale only in the People's Republic of China(excluding the Special Administrative of Hong Kong and Macau)

本书封面贴有 **Pearson Education** 出版集团激光防伪标签, 无标签者不得销售。

FreeBSD 操作系统设计与实现

◆ 著 [美] Marshall Kirk McKusick George V.Neville-Neil
译 张 辉
责任编辑 李 际

◆ 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: 33.5
字数: 816 千字 2006 年 2 月第 1 版
印数: 1-3 500 册 2006 年 2 月北京第 1 次印刷
著作权合同登记号 图字: 01-2004-6141 号

ISBN 7-115-13685-8/TP · 4805

定价: 69.00 元

读者服务热线: (010)67132692 印装质量热线: (010)67129223

内容提要

本书对 FreeBSD 操作系统的内容结构进行了最全面、最新鲜、最权威的论述。

全书共 5 部分 14 章。第一部分包括前 3 章，是对 FreeBSD 历史、设计和内核服务的综述性介绍。第二部分包括第 4 章和第 5 章，分别论述了进程管理和存储管理。第三部分第 6 章到第 10 章，详细介绍了 I/O 系统，包括概述、设备、本地文件系统、网络文件系统、终端系统等。第四部分是第 11 章到第 13 章，分别介绍了进程间通信、网络通信和网络协议。第五部分是第 14 章，介绍了启动和关机。术语表给出了本书涉及到的一些常用术语及其解释。

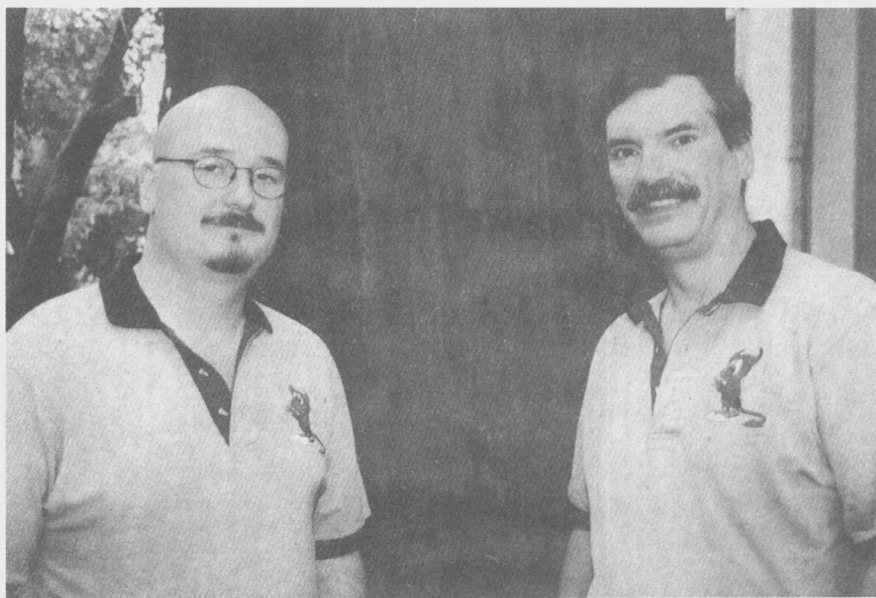
本书既可以作为一本 FreeBSD 实践指南，又可作为深入研究当代可移植的开放源代码操作系统的参考书。本书适合 FreeBSD 操作系统技术支持人员、应用程序开发人员、系统管理员和系统程序员阅读，也可供对学习 FreeBSD 感兴趣的读者参考。

献 辞

本书献给 BSD 社群。

没有该社群的成员们所做的贡献，就没有本书所写的内容。

作译者简介



George V. Neville-Neil (左) 和 Marshall Kirk McKusick (右)

Marshall Kirk McKusick 撰写了许多书籍和文章，指导和讲授 UNIX 和 BSD 方面的课程。他在加州大学伯克利分校的时候实现了 4.2BSD 的快速文件系统，作为 CSRG 的计算机研究专家 (Research Computer Scientist)，他见证了 4.3BSD 和 4.4BSD 的研发和发布历程。他特别感兴趣的研究领域是虚拟内存系统和文件系统。他从康奈尔大学 (Cornell University) 获得电子工程专业学士学位，后又进入加州大学伯克利分校开展研究生阶段的工作，并且获得计算机科学硕士学位和工商管理硕士学位，以及计算机科学博士学位。他曾两次担任 Usenix Association 的主席，目前是 ACM 的 Queue 杂志编委会成员，以及 Usenix Association、ACM 和 IEEE 的会员。他在业余时间喜欢游泳、潜水，还喜欢收藏葡萄酒。他的葡萄酒都储藏在专门建造的酒窖里 (读者朋友可以从网站 <http://www.McKusick.com/cgi-bin/readhouse> 上看到它)，酒窖位于房子的地下室里，而上面的房子属于他和他 25 年的亲密伙伴 Eric Allman¹ 共同拥有。

George V. Neville-Neil 从事编写网络和操作系统代码的工作，一方面是爱好，另一方面也是

¹ 译者注：sendmail 的作者。

张辉：清华大学副研究员，博士。主要从事网络和安全领域的科研、开发和管理，熟悉 UNIX、Linux 等操作系统。在国内外重要学术会议和刊物上发表多篇论文。主要译作有《Linux 系统管理技术手册》、《现代体系结构上的 UNIX》等。

作译者简介

为了赚钱，他还讲授有关程序设计方面多个领域的课程。他感兴趣的领域是代码深入研究、实时操作系统和网络技术。他从位于麻省波士顿的东北大学（Northeastern University）获得计算机科学学士学位，在 ACM 的 Queue 杂志编委会工作。他是 Usenix Association、ACM 和 IEEE 的会员。他热衷于骑自行车和摩托车，而且喜欢旅游。从 1990 年起，他一直生活在旧金山。

前 言

由美国加州大学伯克利分校 (University of California at Berkeley, UC Berkeley) 开发的 BSD 是 UNIX 操作系统的一个重要分支, 笔者以前主持撰写过两本书¹, 分别对 4.3BSD 和 4.4BSD 的设计和实现原理做了权威而详尽的阐述, 本书则是上述工作的自然延续。1994 年伯克利发表了最后一个 BSD 版本, 从那以后, 又有几个小组在继续对 BSD 进行开发。本书将要详细介绍的 FreeBSD 系统拥有最大的开发群体, 其多个版本也流传最广。虽然 FreeBSD 的软件发布仅基本系统就包含近千种实用程序, 而移植软件库 (port collection) 中更是有近万种可以选配的工具程序, 但是本书的讨论内容却几乎只着眼于系统的内核。

类 UNIX 的系统

类 UNIX 的系统既包括像 Solaris 和 HP-UX 这样的传统商业系统, 也包括 Red Hat、Debian、Suse 和 Slackware 等基于 Linux 的软件发布, 以及 FreeBSD、NetBSD、OpenBSD 和 Darwin 等基于 BSD 的软件发布。这些操作系统可以在小到膝上电脑, 大到巨型机的硬件平台上运行。绝大多数多处理器系统、图形系统以及向量处理系统都选择它们作为操作系统, 它们也由于其与生俱来的分时功能 (timesharing), 而得到广泛应用。类 UNIX 系统作为 Internet 上提供网络服务 (从 FTP 到 WWW) 的最通用平台, 是迄今为止已开发出的所有操作系统中可移植性最好的系统。之所以有这么好的可移植性, 一方面要归功于实现它们的开发语言 C [Kemighan & Ritchie, 1989] (C 本身就是一种被广泛移植的语言), 而另一方面则要归功于系统的良好设计。

自 1969 年 UNIX 系统[Ritchie & Thompson, 1978]诞生起, 其发展经历了一系列的分分合合。在 AT&T 的贝尔实验室, 最早的几位开发者不断地对 UNIX 进行改进, 先后开发出 UNIX 的第 9 版、第 10 版, 接着又推出了 Plan 9, 他们把它当做 UNIX 的后继系统。在这其间, AT&T 将 UNIX System V 注册为商业产品, 以后又把它卖给了 Novell。Novell 把 UNIX 这个商标转让给 X/Open, 而把源代码和发行权卖给了 SCO (Santa Cruz Operation) 公司。System V 和第 9 版的 UNIX 都受到了 BSD (Berkeley Software Distributions, 伯克利软件发布) 的很大影响, BSD 是由加州大学伯克利分校的计算机系统研究小组 (Computer Systems Research Group, CSRG) 研制的。Linux 操作系统的开发虽然相对于其他 UNIX 变体保持独立, 但因为它实现了 UNIX 的接口, 所以很容易就可以把基于 UNIX 的平台所开发的应用软件移植到 Linux 上。

伯克利软件发布

伯克利推出的众多发布版本是第一批引入以下诸多重要特性的 UNIX 系统:

¹译者注: 即 BSD 方面的经典 The Design and Implementation of 4.3BSD Operating System 和 The Design and Implementation of 4.4BSD Operating System。

- 支持具有请求调页机制的虚拟内存技术；
- 自动配置硬件和 I/O 系统；
- 具有一种速度快、易恢复的文件系统；
- 实现基于 socket 套接口的 IPC 原语；
- 提供 TCP/IP 的参考实现。

伯克利发布的版本被融入了许多厂商的 UNIX 系统中，而其他很多厂商的开发队伍也在内部用到了它们。4.2BSD 和 4.3BSD 里实现了 TCP/IP 网络协议族，这两种系统应用得非常广泛，这为 TCP/IP 网络协议族成为世界标准起到了关键性的作用。即便像微软这样的非 UNIX 厂商，也在其 Winsock IPC 接口中采用了伯克利套接口（Berkeley socket）的设计。

BSD 还对操作系统接口标准 POSIX（IEEE Std 1003.1）以及其他相关标准产生了很深远的影响。一些 BSD 的特色功能——如可靠信号（reliable signal）、作业控制（job control）、每个进程有多个访问组（access group），以及用于目录操作的例程（routine）——都被吸收到了 POSIX 标准中。

BSD 的早期版本包含有受许可证保护的 UNIX 代码，用户必须拿到 AT&T 的源代码许可证才可以得到和使用 BSD。1988 年，伯克利将其发布版本分成两部分，一部分是需要 AT&T 许可证的代码，另一部分则是可以自由地重新发布的代码。后一部分的代码独立发放许可证，任何人都能得到、使用和重新发布这些代码。在伯克利 1994 年推出的最后一个自由发布版本 4.4BSD-Lite2 中，包含了几乎整个内核以及所有重要的库和工具程序。

1993 年出现了 NetBSD 和 FreeBSD 两个组织，它们开始支持和发布由伯克利所提供的自由发布版本。NetBSD 强调可移植性，并坚持极简路线，该组织不仅把系统移植到了将近 40 种不同的平台之上，而且还努力保持系统的简约，以期达到支持嵌入式应用的目的。FreeBSD 则重视向 PC 平台提供最大化的支持，努力使系统易于安装，并面向尽可能广泛的人群。1995 年，从 NetBSD 中分化出了 OpenBSD，OpenBSD 开发的版本注重系统安全性。多年以来，BSD 的不同发布版本之间保持着一种健康向上的竞争关系，它们彼此之间在理念和代码上进行了很多交流。

本书内容覆盖的范围

本书描述了 FreeBSD 5.2 内核的内部结构，以及为实现 FreeBSD 系统功能所采用的概念、数据结构和算法。书的深度和 Bach 写的关于 UNIX System V[Bach, 1986]一书差不多；不过本书着重介绍 UNIX 操作系统变体——FreeBSD 采用的功能（facility）、数据结构和算法。本书从 FreeBSD 的系统调用层开始向下进行讲解——从接口到内核再到硬件。FreeBSD 的内核包含了诸如进程管理、虚拟内存、I/O 系统、文件系统、套接口 IPC 机制以及网络协议实现这样的系统功能。系统调用层之上的内容——例如库、shell、命令、程序设计语言以及其他用户接口——除了一些与终端界面和系统启动有关的内容之外，均不在本书的讨论之列。本书沿袭了 Organick 在其讲解 Multics 的书[Organick, 1975]里面首创的内容组织形式，对一种当代操作系统进行了深入探讨。

在涉及到具体硬件时，本书则以 Intel PC 的体系结构为参照。由于 FreeBSD 着重于 PC 平台上的开发，对该平台的支持也最完善，因而以它作为参照很方便。

计算机专业人员的使用方法

FreeBSD 被广泛用来支持许多世界级大公司的核心基础设施。因为 FreeBSD 只依托很少的资源就能构建起来，所以可以看到，它在嵌入式应用方面也用得越来越多。FreeBSD 的许可证条款不要求发布对系统所做的变更和增强。Linux 的许可证条款却要求所有对内核进行的改动和增强都必须以源代码形式公开。因此，需要掌握自身知识产权不随意扩散的公司就会采用 FreeBSD 来开发自己的产品。

本书可以直接供工作中用到 FreeBSD 系统的专业人士使用。技术和销售人员可以了解到系统的功能和限制，应用软件开发人员可以学习如何有效和高效地同系统打交道，对 FreeBSD 内核没有直接经验的系统管理员可以了解如何维护、调整和配置系统，而系统程序员则可以学会如何扩展、增强系统以及如何实现系统接口。

本书适用于包括操作系统实现人员、系统程序员、UNIX 应用程序开发人员、系统管理员和有兴趣的用户在内的读者。本书可以结合系统源代码一起阅读，它能在系统手册页和代码细节之间起到桥梁作用。不过本书既不是专门的 UNIX 编程手册，也不是专门的用户教程（想获得用户教程的读者，可以参考 Libes & Ressler [1988]）。熟悉 UNIX 系统一些版本的使用（例如 Stevens [1992]）和 C 程序设计语言（例如 Kernighan & Ritchie [1989]）对阅读本书会极有帮助。

操作系统课程的使用方法

本书适合作为高级操作系统课程的参考书，用以提供相关的背景知识。它不适合用来作为操作系统的入门教程；读者应该已经在前面的介绍中碰到了像内存管理、进程调度、I/O 系统 [Silberschatz et al., 2002] 这样的专业术语。熟悉网络协议的概念 [Comer, 2000; Stallings, 2000; Tanenbaum, 2003] 对理解后面一些章节会有帮助。

对于操作系统高级课程来说，本书可以配合 FreeBSD 系统一同使用。留给学生的作业可以是修改或者替换系统中的关键部分，如调度程序 (scheduler)、调页守护程序 (paging daemon)、文件系统、线程信号机制 (thread signalling)、网络协议中的各层，以及 I/O 管理。系统能够从正在运行的内核直接加载、替换和卸载模块，学生们不需要编译和重启系统就可以完成试验。通过实践一个真正的操作系统，学生们可以直接检验和感受他们对系统进行的修改所产生的效果。在长达 25 年的时间里，FreeBSD 的代码经过了严格的检查，并且一贯保持着良好的代码编写标准，所以比起和它同龄和同等规模的软件项目来说，FreeBSD 内核要明显整洁严谨得多，而且模块化好，更易于理解和修改。

本书每章的最后都附有习题。习题划分成 3 种难度，分别用不带星号、带有一个或者两个星号来表示。没有星号的习题的答案可以在书中的叙述里找到；标有一个星号的习题要对书中某个概念做深一层地研究或者理解才能答出来；标有两个星号的习题表示是大的设计题目或者开放性的研究问题。

本书的结构

本书不但讨论原理和设计方面的问题，还讨论实际实现的细节。讨论通常是从系统调用层开

始，然后再往下深入到内核。书里使用表格和插图来清楚地展示数据结构和控制流程，使用类似 C 语言的伪码来描述算法，用黑体字表示程序名和文件系统的路径名，用斜体字表示术语表中出现的术语以及系统调用、变量、库例程和结构的名字。库例程名（但不是系统调用）后面都跟有一对圆括号（例如，`malloc()`是例程名，而 `argv` 是变量名）。

本书分为以下 5 个部分：

- **第一部分，综述** 前 3 章介绍了 BSD 操作系统的历史、设计概述以及内核服务。第 1 章对系统的历史发展做了综述，强调了系统的研究方向。第 2 章介绍系统所提供的服务并简述内核的内部组成。这一章还讨论了开发系统时所做出的一些设计决策。其中，2.3 节到 2.14 节分别是第 2 节到第 14 章的概述。第 3 章阐述系统调用是怎样实现的，并且详细介绍了内核提供的几种基本服务；
- **第二部分，进程** 这一部分第 1 章，即本书的第 4 章是后面章节的基础。在这一章里介绍了进程的结构、构成进程的多个线程的调度算法，以及系统用来保持内核驻留数据结构访问一致性的同步机制。第 5 章详细地讨论了虚拟内存管理系统；
- **第三部分，I/O 系统** 第 6 章阐述 I/O 设备的系统接口，并且讲解支持该接口的功能结构。接下来的 4 章介绍了 I/O 系统主要部分的细节，其中第 7 章介绍 PC 的 I/O 体系结构，如何管理 I/O 子系统，内核起初如何确定设备，日后如何管理加载和卸载已连接上来的设备；第 8 章从应用程序的角度详细分析了实现文件系统的结构和算法，并且说明了如何采用第 7 章里的设备接口连接本地文件系统；第 9 章从服务端和客户端两方面介绍网络文件系统；第 10 章讨论对字符终端的支持，并且分析了伪终端的设备驱动程序；
- **第四部分，进程间通信** 第 11 章介绍了相关或不相关进程之间的通信机制。第 12 章和第 13 章的内容有密切联系，因为第 12 章所介绍的功能是通过特殊的协议来实现的，例如 TCP / IP 协议族，而协议在第 13 章介绍；
- **第五部分，系统操作** 第 14 章讨论系统的启动、关闭和配置，从进程的层次来介绍系统的初始化过程——从内核初始化一直到用户登录为止。

笔者建议按顺序阅读本书，但除第一部分外，其他部分是彼此独立成章的，可以分开来单独阅读。第 14 章应该最后看，但是专业知识丰富的读者会发现单单就这一章已经很有帮助了。

本书的最后是术语表，它给出主要术语的定义。书中的每一章都包括一份参考文献，提供了相关材料的索引。

如何得到 BSD

所有的 BSD 发行版本要么能从网上下载，要么能以 CD-ROM 或者 DVD 这类易于携带的介质形式获得。从 <http://www.FreeBSD.org> 可以知道如何获得 FreeBSD 源代码和二进制代码。市面上发行的 NetBSD 是编译好的，可以在大多数的工作站上运行。联系 NetBSD 项目组可以获得更多的信息，它们的网址为 <http://www.NetBSD.org/>。OpenBSD 的发行版本也是编译好的，可以在很多工作站上运行，而且出于可靠性和安全性的目的，这个发行版本已经经过了全面的检查。访问 OpenBSD 项目组的网站 <http://www.OpenBSD.org/> 可以获得更多的信息。

坚持看完了前言的读者或许能得到 T 恤衫作为奖励，T 恤衫上的图案照搬了 John Lasseter 专门为本书封面所画的小魔鬼形象（他就是 Pixar 公司的那个有名的 John Lasseter，电影《玩具总动员》就是由他一手策划的）。读者如果想了解怎样购买一件 T 恤衫的话，请访问“History of BSD T-shirts (BSD 历史 T 恤衫)”网站，地址是 [http://www. McKusick.com/beastie/](http://www.McKusick.com/beastie/)。这个网站销售的其他东西还有：配合本书的 32 小时视频入门教程、介绍 FreeBSD 5.2 源代码的 40 小时高级视频教程、长度为两个半小时的 BSD 历史视频讲座，以及 4CD 套装，其中包含伯克利的所有 BSD 版本和 BSD 源代码的受控史。

感谢

在此我们要特别感谢 Nate Lawson，他在 PC 体系结构的工作原理以及 FreeBSD 与 PC 的接口上提供的咨询太有价值了。

我们还要感谢下面这几位阅读整本书并提出意见的人士：Michael Schuster (Sun 公司)、Addison-Wesley 给本书请的审稿人 Chris Cooper (HP 公司) 和 Robert Kitzberger (IBM 公司)。

感谢下列人员阅读本书各章的初稿并提出意见：Samy Al Bahra (Kemeled.com)、Dorr H. Clark (Santa Clara 大学)、Matthew Dillon (DragonFly BSD 项目组)、John Dyson、Andreas Gustafsson、Poul-Henning Kamp (FreeBSD 项目组)、David G. Lawrence (BSD 项目组)、Samuel Leffler、M. Warner Losh (Timing Solutions 公司)、Andre Oppermann (Internet Business Solutions AG 公司)、David I. Parfitt (独立的 hacker)、Doug Rabson (Qube Software 公司)、Jeffrey Roberson (FreeBSD 项目组)、Soren Schmidt (FreeBSD 高级开发人员)、Ken Smith (布法罗大学计算机科学与工程系)、Gregory Sutter (Daemon News)、Charles P. Wright (纽约州立大学石溪分校) 和 Erez Zadok (Stony Brook 大学)。

感谢我们的编辑 Peter Gordon，尽管我们耽搁了几年时间，但他相信我们的能力，并且在我們完成书稿以后加快了出版的脚步。感谢 Addison-Wesley 的所有专业人士，在他们的帮助下，才使得本书得以付梓：责任编辑 John Fuller、编辑助理 Bernie Gaffney、制作监理 Elizabeth Ryan、封面设计 Chuti Prasertsith。感谢 Stratford 出版公司的同仁：责任编辑 Kathy Glidden、版面文字编辑 Debbie Prato 和校对 Hilary Farquhar。最后感谢 Jaap Akkerhuis 和 John Lasseter 对本书所做出的贡献。Jaap Akkerhuis 设计了用于 BSD 图书的 troff 宏，John Lasseter 绘制了本书封面所用的 BSD 小魔鬼。

本书在编排上使用了 James Clark 实现的 pic、tbl、eqn 以及 groff。索引由脚本语言 awk 生成，awk 是从 Jon Bentley 和 Brian Kernighan [Bentley & Kernighan, 1986] 编写的索引生成程序发展而来的。美工方面主要使用了 xfig。图片的排版以及段中短行的消除由 groff 的宏来完成，但消除段标题和段落分离的情况，以及偶数页底部的处理则是通过手工完成的。

我们鼓励读者将有关本书在排版或其他方面错误的改进建议或意见发送给我们，请发电子邮件到 FreeBSDbook-bugs@McKusick.COM。

参考文献

Bach, 1986.

- M. J. Bach, *The Design of the UNIX Operating System*, Prentice-Hall, Englewood Cliffs, NJ, 1986.
- Bentley & Kernighan, 1986.
- J. Bentley & B. Kernighan, "Tools for Printing Indexes," Computing Science Technical Report 128, AT&T Bell Laboratories, Murray Hill, NJ, 1986.
- Comer, 2000.
- D. Comer, *Internetworking with TCP/IP Volume 1*, 4th ed., Prentice-Hall, Upper Saddle River, NJ, 2000.
- Kernighan & Ritchie, 1989.
- B. W. Kernighan & D. M. Ritchie, *The C Programming Language*, 2nd ed., Prentice-Hall, Englewood Cliffs, NJ, 1989.
- Libes & Ressler, 1988.
- D. Libes & S. Ressler, *Life with UNIX*, Prentice-Hall, Englewood Cliffs, NJ, 1988.
- Organick, 1975.
- E. I. Organick, *The Multics System: An Examination of Its Structure*, MIT Press, Cambridge, MA, 1975.
- Ritchie & Thompson, 1978.
- D. M. Ritchie & K. Thompson, "The UNIX Time-Sharing System," *Bell System Technical Journal*, vol. 57, no. 6, Part 2, pp. 78-90, July-August 1978. The original version [*Comm. ACM vol. 7, no. 7, pp. 365-375 (July 1974)*] described the 6th edition; this citation describes the 7th edition.
- Silberschatz et al., 2002.
- A. Silberschatz, P. Galvin, & G. Gagne, *Operating System Concepts*, 6th ed., John Wiley and Sons, Hoboken, NJ, 2002.
- Stallings, 2000.
- R. Stallings, *Data and Computer Communications*, 6th ed., Prentice Hall, Hoboken, NJ, 2000.
- Stevens, 1992.
- W. Stevens, *Advanced Programming in the Unix Environment*, Addison-Wesley, Reading, MA, 1992.
- Tanenbaum, 2003.
- A. S. Tanenbaum, *Computer Networks*, 4th ed., Prentice-Hall, Englewood Cliffs, NJ, 2003.

目 录

第一部分 综 述

第 1 章 BSD 系统的历史和目标	3	2.6.3 设备	30
1.1 UNIX 系统的历史	3	2.6.4 套接口 IPC	30
1.1.1 UNIX 系统的起源	3	2.6.5 分散/聚集 I/O	31
1.1.2 Research 小组的 UNIX 系统	4	2.6.6 多文件系统支持	31
1.1.3 AT&T UNIX System III 和 System V	5	2.7 设备	32
1.1.4 伯克利软件发布 (BSD)	5	2.8 文件系统	32
1.1.5 UNIX 无处不在	6	2.9 网络文件系统	35
1.2 BSD 和其他系统	6	2.10 终端	36
1.3 BSD 向开放源代码的转变	8	2.11 进程间通信	37
1.3.1 Networking Release 2	8	2.12 网络通信	37
1.3.2 法律诉讼	10	2.13 网络实现	38
1.3.3 4.4BSD	11	2.14 系统运行	38
1.3.4 4.4BSD-Lite Release 2	11	2.15 复习题	38
1.4 FreeBSD 的开发模式	12	2.16 参考文献	39
1.5 参考文献	15	第 3 章 内核服务	41
第 2 章 FreeBSD 设计概述	19	3.1 内核结构	41
2.1 FreeBSD 的功能和内核	19	3.1.1 系统进程	41
2.2 内核结构	20	3.1.2 系统入口	42
2.3 内核服务	22	3.1.3 运行时刻的内核结构	42
2.4 进程管理	23	3.1.4 内核的入口	43
2.4.1 信号	24	3.1.5 从内核返回	44
2.4.2 进程组和会话	25	3.2 系统调用	44
2.5 内存管理	25	3.2.1 调用结果的处理	45
2.5.1 BSD 内存管理设计要点	26	3.2.2 从系统调用返回	45
2.5.2 内核中的内存管理	27	3.3 陷阱和中断	46
2.6 I/O 系统	28	3.3.1 陷阱	46
2.6.1 描述符与 I/O	28	3.3.2 I/O 设备中断	46
2.6.2 描述符管理	29	3.3.3 软件中断	47
		3.4 时钟中断	47
		3.4.1 统计和进程调度	48
		3.4.2 超时	49

3.5 内存管理服务	50	4.4.2 线程调度	81
3.6 时间服务	53	4.4.3 线程优先级的计算	82
3.6.1 真实时间	53	4.4.4 线程优先级例程	83
3.6.2 外部表示	54	4.4.5 线程运行队列和上下文切换	83
3.6.3 调整时间	54	4.4.6 ULE 调度程序	85
3.6.4 时间间隔	54	4.5 创建进程	87
3.7 用户、用户组和其他身份标识	55	4.6 终止进程	89
3.7.1 主机标识符	57	4.7 信号	90
3.7.2 进程组和会话	57	4.7.1 信号的历史	92
3.8 资源服务	58	4.7.2 发送信号	92
3.8.1 进程优先级	58	4.7.3 接收信号	94
3.8.2 资源利用	58	4.8 进程组和会话	95
3.8.3 资源限制	59	4.8.1 会话	96
3.8.4 文件系统配额	59	4.8.2 作业控制	98
3.9 系统运行服务	59	4.9 监管环境	99
3.10 复习题	60	4.9.1 监管环境的语义	100
3.11 参考文献	61	4.9.2 监管环境的实现	102
		4.9.3 监管环境的限制	103
		4.10 进程的调试	104
		4.11 复习题	105
		4.12 参考文献	107
		第 5 章 存储管理	109
		5.1 术语	109
		5.1.1 进程与内存	110
		5.1.2 调页机制	110
		5.1.3 替换算法	111
		5.1.4 工作集模型	112
		5.1.5 交换机制	112
		5.1.6 虚拟内存的优点	112
		5.1.7 虚拟内存的硬件要求	113
		5.2 FreeBSD 虚拟内存系统概述	113
		5.3 内核的存储管理	115
		5.3.1 内核映射和子映射	116
		5.3.2 内核地址空间的分配	117
		5.3.3 内核的存储分配程序	117
		5.3.4 内核的区域存储分配程序	119
		5.4 进程独立拥有的资源	121
第二部分 进 程			
第 4 章 进程管理	65		
4.1 进程管理概述	65		
4.1.1 多程序机制	66		
4.1.2 调度	66		
4.2 进程状态	67		
4.2.1 进程结构	69		
4.2.2 线程结构	71		
4.3 上下文切换	72		
4.3.1 线程状态	73		
4.3.2 底层上下文切换	73		
4.3.3 主动上下文切换	73		
4.3.4 同步	75		
4.3.5 互斥同步	78		
4.3.6 锁管理器的锁	79		
4.3.7 其他同步	80		
4.4 线程调度	80		
4.4.1 4.4BSD 的调度程序	81		

6.5.4	文件系统的导出服务	196	8.2.2	扩展属性	238
6.6	与文件系统无关的服务	197	8.2.3	文件系统的新功能	239
6.6.1	名字缓存	198	8.2.4	文件标志	240
6.6.2	缓冲区管理	199	8.2.5	动态的 inode	241
6.6.3	缓冲区管理的实现	201	8.2.6	管理 inode	242
6.7	可叠加的文件系统	203	8.3	命名	243
6.7.1	简单的文件系统层	204	8.3.1	目录	244
6.7.2	联合安装的文件系统	205	8.3.2	在目录中查找名字	245
6.7.3	其他文件系统	207	8.3.3	路径名转换	247
6.8	复习题	208	8.3.4	链接	248
6.9	参考文献	208	8.4	配额	249
第 7 章	设备	210	8.5	文件上锁	252
7.1	设备概述	210	8.6	软更新	256
7.1.1	PC 的 I/O 体系结构	210	8.6.1	文件系统中的更新依赖	257
7.1.2	FreeBSD 海量存储 I/O 子系统的结构	212	8.6.2	依赖关系的数据结构	260
7.1.3	设备的命名和访问	214	8.6.3	跟踪位映射表的依赖关系	262
7.2	GEOM 层	215	8.6.4	跟踪 inode 的依赖关系	262
7.2.1	术语和拓扑规则	215	8.6.5	跟踪直接块的依赖关系	263
7.2.2	改变拓扑	217	8.6.6	跟踪间接块的依赖关系	264
7.2.3	运行	218	8.6.7	跟踪新闻接块的依赖关系	265
7.2.4	拓扑的灵活性	219	8.6.8	跟踪新目录项的依赖关系	266
7.3	CAM 层	220	8.6.9	跟踪新目录的依赖关系	267
7.3.1	SCSI 子系统	220	8.6.10	跟踪删除目录项时的依赖关系	269
7.3.2	I/O 请求通过 CAM 子系统的 路径	221	8.6.11	截短文件	269
7.4	ATA 层	222	8.6.12	回收文件和目录的 inode 节点	269
7.5	配置设备	223	8.6.13	跟踪目录项重命名时的依 赖关系	270
7.5.1	识别设备	226	8.6.14	跟踪删除文件时的依 赖关系	270
7.5.2	自动配置数据结构	227	8.6.15	fsync 对软更新的要求	271
7.5.3	资源管理	231	8.6.16	删除文件时对软更新的 要求	272
7.6	复习题	232	8.6.17	fsck 对软更新的要求	273
7.7	参考文献	233	8.6.18	软更新的性能	274
第 8 章	本地文件系统	234	8.7	文件系统的快照	275
8.1	文件系统的分层管理	234			
8.2	inode 的结构	235			
8.2.1	inode 格式的变化	237			