



原理 · 应用 · 开发 · 系统 · 网络管理

操作系统教程

(第3版)

陆松年 主编

潘理 翁亮 薛质 等编著

面向教学 · 立足实用



电子工业出版社

PUBLISHING HOUSE OF ELECTRONICS INDUSTRY
<http://www.phei.com.cn>

操作系统教程 (第3版)

面向教学 立足实用

本书详细阐述了操作系统的基本原理、概念和应用,全书以传统的UNIX为主线进行讲解,而且还对 Windows 2000/XP/2008、最新推出的Windows 7、嵌入式Linux系统和操作系统安全进行了分析和介绍。本书共分3篇:第1篇介绍操作系统的基本原理及较常用的操作系统实例,如UNIX、Linux、Solaris、Windows 2000/XP/2003/Vista/2008/7等;第2篇介绍在UNIX和嵌入式Linux系统下的系统程序设计、应用开发工具及开发技术;第3篇介绍UNIX系统管理和UNIX网络系统管理。

本书既可作为高等院校计算机科学与技术专业的教材,也可以作为通信工程、电子工程、信息安全、自动化控制和信息管理类等非计算机专业的教材和教学参考书,对于计算机软件开发人员、系统和网络管理人员,也是一本很好的参考书。

- ◎ 每章后面都给出了精心设计的习题,使读者温故知新。
- ◎ 全面、详细介绍了UNIX、Linux、Windows的原理,以及应用和开发。

图书分类: 操作系统



责任编辑: 高洪霞

责任美编: 李 玲

本书贴有激光防伪标志,凡没有防伪标志者,属盗版图书。

ISBN 978-7-121-10335-3



9 787121 103353 >

定价: 45.00元

操作系统教程

(第3版)

陆松年 主编

潘理 翁亮 薛质 等编著



电子工业出版社

Publishing House of Electronics Industry

北京·BEIJING

内 容 简 介

本书详细阐述了操作系统的基本原理、概念和应用,全书以传统的 UNIX 为主线进行讲解,而且还对 Windows 2000/XP/2008、最新推出的 Windows 7、嵌入式 Linux 系统和操作系统安全进行了分析和介绍。本书共分 3 篇:第 1 篇介绍操作系统的基本原理及较常用的操作系统实例,如 UNIX、Linux、Solaris、Windows 2000/XP/2003/Vista/2008/7 等;第 2 篇介绍在 UNIX 和嵌入式 Linux 系统下的系统程序设计、应用开发工具及开发技术;第 3 篇介绍 UNIX 系统管理和 UNIX 网络系统管理。

本书可作为高等院校计算机科学与技术专业以及通信工程、电子工程、信息安全、自动化控制和信息管理类等非计算机专业的教材和教学参考书,对于计算机软件开发人员、系统和网络管理人员,也是一本很好的参考书。

未经许可,不得以任何方式复制或抄袭本书之部分或全部内容。
版权所有,侵权必究。

图书在版编目(CIP)数据

操作系统教程 / 陆松年主编;潘理等编著. —3 版. —北京:电子工业出版社,2010.2
ISBN 978-7-121-10335-3

I. 操… II. ①陆… ②潘… III. 操作系统—教材 IV. TP316

中国版本图书馆 CIP 数据核字(2010)第 021580 号

责任编辑:高洪霞

印 刷:北京天宇星印刷厂

装 订:三河市鹏成印业有限公司

出版发行:电子工业出版社

北京市海淀区万寿路 173 信箱 邮编 100036

经 销:各地新华书店

开 本:787×1092 1/16 印张:31.5 字数:820 千字

印 次:2010 年 2 月第 1 次印刷

印 数:4000 册 定价:45.00 元

凡所购买电子工业出版社图书有缺损问题,请向购买书店调换。若书店售缺,请与本社发行部联系,联系及邮购电话:(010) 88254888。

质量投诉请发邮件至 zltz@phei.com.cn, 盗版侵权举报请发邮件至 dbqq@phei.com.cn。

服务热线:(010) 88258888。

第3版前言

《操作系统教程》一书出版已经9年了,在这期间Linux和Windows操作系统都有了较大的发展,但UNIX类操作系统在商用服务器领域中的地位还没有根本动摇,经典的UNIX系统相对于Linux和Windows操作系统来说系统稳定性更好,也较适合于教学。

这次修订的第3版以操作系统的基本原理、概念和应用为框架,以传统的UNIX为主线进行讲解。第3版增加了部分操作系统的内容,删去了一些不常用的应用与命令,并反映了操作系统的最新发展。

Solaris是目前使用最为广泛、最成功的商业UNIX实现版本,也是一个最安全和最稳定的商用操作系统。自2005年Sun公司正式推出其开源项目OpenSolaris以来,到目前为止已经完全开放了Solaris最新的内核源代码,而且还开放了Solaris的库、包管理和安装程序等。Sun公司宣布,OpenSolaris的开放源代码在由开放源代码促进会批准的“通用开发和发布许可协议”下提供,Sun将与OpenSolaris相关的专利免费提供给用户,而且用户有权保护在OpenSolaris源代码基础上开发的自有技术。如此,极大地促进了OpenSolaris社团组织的发展与壮大。OpenSolaris开源项目必将大大促进操作系统的研究与发展。

为了适应操作系统开源项目的最新发展,本书第3版在存储管理、进程管理、进程通信、设备管理和文件系统各章分别对Solaris操作系统中最有特色的相关的原理进行了介绍。在第14章“UNIX系统与网络系统程序设计”中还给出了Solaris门和RPC程序设计示例。此外该章还增加了UDP套接口编程。因此本版对操作系统在通信方面的应用有所加强。

针对目前使用人数最多的Windows操作系统,本书第7章中对Windows 2000/XP进行了全面的分析和介绍,对其系统结构、存储系统、进程/线程及处理器管理、I/O系统、文件系统和网络系统等方面的核心技术进行了较为深入的分析,对Windows 2003服务器操作系统平台的特性做了介绍。之后,对微软最新的服务器操作系统产品Windows Server 2008和最新发布的下一代桌面操作系统Windows 7也分别进行了介绍,重点分析了其对以前Windows版本的功能更新部分。本章还对x86平台操作系统虚拟化技术的基本原理进行了介绍,并以开源的Xen虚拟化技术为例说明了虚拟化的基本原理。这部分内容可以和本书前面的操作系统基本原理部分进行对照学习。

目前嵌入式系统已被广泛地应用于工业控制系统、通信设备和消费电子等众多领域,而32位嵌入式操作系统则是嵌入式系统最核心的技术之一。因此,计算机、电子、通信和自动化控制专业的学生除了要掌握通用操作系统的原理和应用技术之外,还必须对嵌入式操作系统有一定的了解。先进、成熟的Linux系统作为嵌入式操作系统代表了当前嵌入式技术领域发展的潮流和方向。本书第16章全面地介绍了嵌入式Linux系统的基本内容,如系统裁剪、软件开发与调试及实时化等,可作为嵌入式操作系统学习的基础材料和应用参考资料。

随着全球计算机信息化的发展,用户对信息服务的安全性提出了越来越高的要求。计算机信息系统的安全性主要依赖于其软件系统的安全性。由于操作系统是计算机软件系统中具有基础性、战略性和唯一控制硬件的关键软件,其安全职能是其他软件安全的根基。没有操

目 录

第 1 部分 原 理 篇

第 1 章 操作系统概论..... 2	第 2 章 存储管理..... 20
1.1 什么是操作系统..... 2	2.1 存储管理基础..... 20
1.2 操作系统的发展历史..... 3	2.1.1 虚拟地址与物理地址..... 20
1.2.1 早期的计算机和人工 操作方式..... 3	2.1.2 地址定位方式..... 20
1.2.2 脱机输入 / 输出和批 处理系统..... 4	2.2 基本存储管理方法..... 22
1.2.3 缓冲、中断和 DMA 技术..... 4	2.2.1 单一连续区存储管理..... 22
1.2.4 SPOOLING..... 5	2.2.2 固定分区存储管理..... 23
1.2.5 多道程序设计..... 6	2.3 可变分区存储管理..... 24
1.3 现代操作系统类型..... 7	2.3.1 空闲存储区表..... 24
1.3.1 分时系统..... 7	2.3.2 首次适应法..... 25
1.3.2 实时操作系统..... 8	2.3.3 循环首次适应法..... 27
1.3.3 微机操作系统..... 8	2.3.4 最佳适应算法..... 27
1.3.4 多处理机、分布式和网络 操作系统..... 10	2.3.5 最差适应法..... 28
1.3.5 嵌入式操作系统和智能卡 操作系统..... 11	2.3.6 多重分区..... 28
1.4 操作系统的概念、特征和 功能..... 11	2.4 内存扩充技术..... 28
1.4.1 作业和进程..... 11	2.4.1 覆盖..... 28
1.4.2 操作系统的特征..... 12	2.4.2 交换技术..... 29
1.4.3 操作系统的功能..... 13	2.4.3 虚拟存储器..... 30
1.5 UNIX 操作系统概述..... 14	2.5 纯分页的存储管理..... 31
1.5.1 UNIX 系统历史..... 14	2.5.1 分页存储管理的基本思想..... 31
1.5.2 Solaris 操作系统..... 16	2.5.2 地址变换..... 32
1.5.3 UNIX 系统特点..... 18	2.5.3 联想存储器和快表..... 32
1.5.4 UNIX 系统基本结构..... 18	2.5.4 空闲内存页的管理..... 33
习题 1..... 19	2.6 请求分页系统..... 34
	2.6.1 请求分页的基本原理..... 34
	2.6.2 页面淘汰..... 34
	2.7 段式存储管理..... 37
	2.8 段页式存储管理..... 38
	2.9 Linux 存储管理..... 39

2.9.1 Linux 存储管理的思想及特点	39	3.7.2 多线程	71
2.9.2 Linux 中的页表	40	3.7.3 线程的状态与功能	72
2.9.3 页面的分配和回收	40	3.7.4 用户级和核心级线程	73
2.9.4 存储映射与请求分页	42	3.8 Linux 进程管理	75
2.9.5 页交换进程与页面的换入换出	43	3.8.1 Linux 进程结构	75
2.10 Solaris 存储管理	44	3.8.2 进程调度	78
2.10.1 Solaris 存储管理的体系结构	44	3.8.3 进程使用的虚拟内存	79
2.10.2 Solaris 存储管理的新特性	44	3.8.4 创建进程	79
2.10.3 Solaris 存储管理	45	3.8.5 Linux 的线程	80
2.10.4 页面淘汰算法	46	3.9 Solaris 进程和线程管理	81
习题 2	47	3.9.1 Solaris 进程和线程	81
第 3 章 进程管理	49	3.9.2 Solaris 进程和线程模型的演变	81
3.1 进程概述	49	3.9.3 进程数据结构	82
3.1.1 进程的概念	49	3.9.4 内核线程数据结构	82
3.1.2 进程的组成	49	3.9.5 Solaris 的线程调度	84
3.1.3 进程的状态及其变化	50	3.9.6 Solaris 的线程应用示例	87
3.2 进程控制块	51	习题 3	88
3.3 调度	56	第 4 章 进程通信	90
3.3.1 调度概述	56	4.1 进程的同步与互斥	90
3.3.2 进程调度策略	57	4.1.1 同步与互斥的概念	90
3.3.3 进程调度算法	57	4.1.2 临界段问题	90
3.4 UNIX 系统的进程调度	60	4.2 进程间互斥控制方法	91
3.4.1 进程的切换调度算法	60	4.2.1 锁的表示和操作	91
3.4.2 切换调度程序	62	4.2.2 锁的安全控制	91
3.4.3 进程的对换调度	63	4.2.3 用纯软件实现进程的互斥	93
3.5 进程的控制	64	4.3 信号量和 semWait、semSignal 操作	95
3.5.1 进程的阻塞	64	4.4 信号量的应用	96
3.5.2 UNIX 系统中的进程睡眠和唤醒	64	4.4.1 利用信号量实现互斥	96
3.5.3 进程的终止和等待终止	67	4.4.2 阻塞 / 唤醒协议	97
3.6 进程的创建和图像改换	68	4.4.3 两个进程间的同步	98
3.6.1 进程的创建	68	4.4.4 生产者和消费者问题	99
3.6.2 进程图像的改换	69	4.4.5 读者 / 写者问题	100
3.7 线程	70	4.4.6 管程	101
3.7.1 进程和线程	71	4.5 进程间的数据通信	101
		4.5.1 消息通信	101
		4.5.2 共享存储区	102

4.5.3	管道通信	103	5.4	操作系统对 I/O 操作的控制	129
4.6	软中断和信号机构	103	5.4.1	I/O 设备的资源分配	129
4.6.1	信号的产生与类型	103	5.4.2	I/O 通道技术	130
4.6.2	信号的处理方式及设置	105	5.4.3	I/O 缓冲技术	130
4.6.3	信号的传送	106	5.4.4	设备的驱动	131
4.7	死锁	106	5.5	设备管理的数据结构	131
4.7.1	产生死锁的原因	107	5.5.1	设备控制表	131
4.7.2	产生死锁的条件	108	5.5.2	设备开关表	132
4.7.3	死锁的预防	109	5.6	磁盘调度	133
4.7.4	死锁的避免	110	5.6.1	物理特性	133
4.7.5	死锁的检测	111	5.6.2	磁盘调度算法	133
4.7.6	死锁的解除	112	5.6.3	系统设计应考虑的几个问题	135
4.8	Linux 进程间通信	113	5.6.4	独立磁盘冗余阵列	135
4.8.1	信号	113	5.7	UNIX 系统 V 的设备管理	138
4.8.2	管道	115	5.7.1	块设备管理的主要数据结构	138
4.8.3	IPC 机制	116	5.7.2	缓冲区管理	140
4.9	Solaris 的进程通信机制	118	5.7.3	块设备管理	143
4.9.1	Solaris 的进程通信机制概述	118	5.7.4	字符设备管理	145
4.9.2	Solairs 信号机制	118	5.8	设备分配	147
4.9.3	Solairs 信号量	119	5.8.1	设备分配方式	147
4.9.4	Solairs 门	119	5.8.2	设备分配的原则	147
习题 4		120	5.9	Linux 的设备管理	148
第 5 章	设备管理	121	5.9.1	概述	148
5.1	概述	121	5.9.2	设备驱动器与内核的接口	150
5.1.1	设备的概念和分类	121	5.9.3	网络设备的管理	152
5.1.2	I/O 设备控制与驱动	122	5.10	Solaris 的 I/O 子系统	152
5.1.3	设备管理的设计要求和任务	123	5.10.1	I/O 子系统与内核的关系	152
5.2	操作系统与中断处理	123	5.10.2	设备文件与设备文件系统	153
5.2.1	中断的基本概念	123	5.10.3	设备树	153
5.2.2	中断的类型	124	习题 5		154
5.2.3	中断的响应和实现过程	126	第 6 章	文件系统	156
5.2.4	中断处理程序和驱动程序	126	6.1	概述	156
5.2.5	中断的返回与恢复	127	6.1.1	文件的术语	156
5.3	操作系统与时钟系统	128	6.1.2	文件的操作	157
5.3.1	时钟的概念	128	6.1.3	文件的组织和存取	157
5.3.2	UNIX 系统中的时钟管理	128	6.1.4	文件系统结构	160

6.2	文件目录	161
6.2.1	目录的内容	161
6.2.2	目录的结构	162
6.2.3	存取权限	163
6.2.4	并发存取控制	163
6.3	文件存储资源分配	164
6.4	文件的系统调用	165
6.4.1	文件的创建、打开、 关闭和取消	166
6.4.2	文件的读 / 写	168
6.4.3	调整文件读 / 写位置 lseek	169
6.4.4	创建任何类型文件 mknod	169
6.4.5	其他的文件系统调用	170
6.5	文件的标准子例程	171
6.5.1	标准 I/O 的概念	171
6.5.2	流文件的打开和关闭	172
6.5.3	流文件的读写	173
6.5.4	调整和获取流文件的 读写位置	174
6.5.5	格式输入与输出	174
6.5.6	流的单字符 I/O 操作	175
6.5.7	行的输入与输出	176
6.5.8	存储区中的格式转化	176
6.5.9	程序的执行	176
6.6	UNIX 文件系统的内部结构	177
6.6.1	索引节点	177
6.6.2	文件索引结构	177
6.6.3	目录结构	179
6.6.4	打开文件结构	180
6.6.5	文件系统存储资源管理	182
6.7	管道文件和管道通信	186
6.7.1	管道通信概念	186
6.7.2	管道文件	186
6.7.3	管道的读写和关闭	187
6.7.4	有名管道	188
6.8	Linux 文件系统	190
6.8.1	虚拟文件系统	190
6.8.2	Ext2 文件系统	194
6.8.3	Ext3 文件系统	197

6.9	Solaris 文件系统	197
6.9.1	Sun 虚拟文件系统模型	197
6.9.2	文件系统可恢复性	198
6.9.3	文件系统安全性	199
6.9.4	Solaris 文件系统框架	200
6.9.5	ZFS 文件系统的特点	201
	习题 6	202

第 7 章 Windows 操作系统 204

7.1	Windows 操作系统发展历程	204
7.1.1	Windows 的开发过程及 历史	204
7.1.2	Windows 主要版本的特点	204
7.2	Windows 2000/XP	207
7.2.1	系统体系结构	207
7.2.2	存储系统	210
7.2.3	进程、线程及处理器管理	216
7.2.4	I/O 系统	220
7.2.5	文件系统	223
7.2.6	网络系统	226
7.3	Windows Server 2008	230
7.3.1	系统概述	230
7.3.2	系统功能更新	230
7.4	下一代操作系统产品 Windows 7	232
7.4.1	系统概述	232
7.4.2	系统功能更新	232
7.5	操作系统虚拟化技术	234
7.5.1	虚拟化技术概述	234
7.5.2	Xen 虚拟化技术	235
7.5.3	其他虚拟化产品	238
7.6	小结	240
	习题 7	240

第 8 章 分布式和网络操作系统 241

8.1	分布式操作系统概述	241
8.2	网络通信机制	244
8.3	网络同步	246
8.4	网络操作系统	247
	习题 8	248

第9章 操作系统安全	249	9.2.4 访问控制	260
9.1 操作系统的安全性	249	9.2.5 隐蔽通道检测与控制	261
9.1.1 操作系统的安全需求	249	9.2.6 安全审计	263
9.1.2 系统安全的评估与标准	250	9.3 安全操作系统设计与实现	265
9.1.3 Linux 操作系统安全	253	9.3.1 操作系统安全设计原理	265
9.1.4 Windows 2000 操作系统安全	254	9.3.2 安全策略	266
9.2 操作系统安全机制	256	9.3.3 安全模型	267
9.2.1 标识与鉴别	256	9.3.4 安全体系结构	269
9.2.2 可信路径	258	9.3.5 典型安全操作系统	271
9.2.3 最小特权管理	259	习题9	273

第2部分 应用和开发篇

第10章 UNIX 使用基础	276	11.1.2 改变工作目录 (cd 命令)	286
10.1 登录和退出系统	276	11.1.3 列目录内容 (ls 命令)	286
10.2 Shell 基础	277	11.1.4 创建新目录 (mkdir 命令)	287
10.2.1 Shell 的家族	277	11.1.5 删除目录项 (rmdir 命令)	287
10.2.2 简单命令和命令表	277	11.2 文件操作命令	287
10.2.3 标准输入和标准输出	278	11.2.1 复制文件 (cp 命令)	287
10.2.4 输入 / 输出转向	278	11.2.2 移动或重新命名文件 (mv 命令)	288
10.2.5 后台命令与后台进程	279	11.2.3 删除文件 (rm 命令)	288
10.2.6 管道	280	11.2.4 文件的链接 (ln 命令)	288
10.2.7 特殊字符	280	11.2.5 符号链接 (ln -s 命令)	288
10.3 文件系统基本知识	281	11.2.6 报告两个文件的差别 (diff 命令)	289
10.3.1 文件的类型	281	11.2.7 推测文件的类型 (file 命令)	290
10.3.2 目录结构	282	11.2.8 查找文件 (find 命令)	290
10.3.3 文件存取控制模式	282	11.2.9 改变文件的属主 (chown 命令)	291
10.3.4 目录存取方式	283	11.2.10 改变文件的存取权限 (chmod 命令)	291
10.4 UNIX 文件系统结构	283	11.2.11 设置文件创建方式屏蔽码 (umask 命令)	292
10.4.1 根文件系统结构	284	11.3 显示和打印命令	292
10.4.2 /usr 文件系统	284		
习题10	285		
第11章 UNIX 实用程序	286		
11.1 目录操作命令	286		
11.1.1 显示工作目录 (pwd 命令)	286		

11.3.1	回应命令行上的参数 (echo 命令)	292	11.6	用户信息与进程控制	305
11.3.2	连接并显示文件 (cat 命令)	293	11.6.1	获取用户信息	305
11.3.3	分页显示 (more 命令和 pg 命令)	293	11.6.2	显示进程状态 (ps 命令)	305
11.3.4	显示文件的头部 (head 命 令) 和尾部 (tail 命令)	293	11.6.3	挂起一段时间 (sleep 命令)	306
11.3.5	打印文件 (lp 命令)	294	11.6.4	统计一个作业的执行时间 (time 命令)	306
11.3.6	查看打印作业状态 (lpstat 命令)	294	11.6.5	等待进程完成 (wait 命令)	306
11.3.7	取消一个打印作业 (cancel 命令)	294	11.6.6	杀死一个进程 (kill 命令)	306
11.3.8	格式化打印文件 (pr 命令)	295	11.6.7	忽略挂起和退出方法执行 (nohup 命令)	306
11.3.9	卸出文件 (od 命令)	295	11.6.8	设置命令的执行环境 (env 命令)	307
11.4	过滤器	296	11.7	压缩和解压缩	307
11.4.1	正则表达式	296	11.7.1	compress	307
11.4.2	排序或合并文件 (sort 命令)	298	11.7.2	zip	307
11.4.3	抽取指定模式的行 (grep 命令)	298	11.7.3	gzip	308
11.4.4	流编辑程序 (sed 命令)	299	11.7.4	pack	309
11.4.5	模式扫描和处理语言 (awk)	300	11.8	其 他 命 令	309
11.4.6	裁剪指定的列 (cut 命令)	301	11.8.1	联机手册 (man 命令)	309
11.4.7	组合数据列 (paste 命令)	302	11.8.2	检查拼写错误 (spell 命令)	310
11.4.8	建立“三通”(tee 命令)	302	11.8.3	文件加密 crypt 命令和 des 命令	310
11.5	设备操作	302	11.8.4	规范化 C 程序格式 (cb 命令)	310
11.5.1	显示或设置日期和时间 (date 命令)	302	习题 11		311
11.5.2	显示磁盘空间 (df 命令)	303	第 12 章	软件开发工具	312
11.5.3	显示磁盘使用情况 (du 命令)	303	12.1	文本编辑器 vi	312
11.5.4	显示或设置终端参数 (stty 命令)	303	12.1.1	vi 概要	312
11.5.5	磁带归档 (tar 命令)	304	12.1.2	屏幕编辑命令	313
			12.1.3	底行命令	317
			12.1.4	在 vi 程序中执行 Shell 命令	319
			12.2	C 编译系统	320

12.2.1	C 编译基本过程	320	13.2.2	显示远程用户的信息 (finger)	344
12.2.2	预处理程序	321	13.3	用户间通话	344
12.2.3	编译程序	323	13.3.1	给用户发信息 (write)	344
12.2.4	汇编程序	323	13.3.2	广播式的消息传送工具 (Wall)	344
12.2.5	连接 / 装入程序	324	13.3.3	远程用户对话 (talk)	345
12.2.6	一个使用 cc 的综合例子	325	13.4	远程登录和执行	345
12.3	库的维护	325	13.4.1	基于 UNIX 系统的远程 登录 (RLogin)	345
12.3.1	库的概念	325	13.4.2	通用的远程登录 (telnet)	346
12.3.2	库的维护	326	13.4.3	远程 Shell (RSH)	346
12.3.3	动态库	327	13.5	远程文件复制和传输	347
12.4	维护程序的程序 (Make)	327	13.5.1	远程文件复制 (RCP)	347
12.4.1	Make 使用初步	328	13.5.2	文件传输 (FTP)	347
12.4.2	Make 的内部规则	329	13.6	万维网 WWW 和浏览器	349
12.4.3	Make 中的宏	331	13.6.1	超文本和超媒体	349
12.4.4	用 Make 维护库	332	13.6.2	超文本传输协议 HTTP	350
12.4.5	一个综合例子	333	13.6.3	统一资源定位器 URL	350
12.4.6	Make 命令选项和参数	335	13.6.4	超文本标记语言 HTML	350
12.5	调试程序	335	13.6.5	Form 与 CGI	351
12.5.1	调用 sdb	336	13.6.6	网站设计语言	351
12.5.2	sdb 的调试命令	336	13.6.7	WWW 的浏览程序	353
12.5.3	删除调试信息 strip	337	习题 13		354
12.6	源代码控制系统 (SCCS)	337	第 14 章	UNIX 系统和网络程序设计	355
12.6.1	创建 SCCS 的初始版本	338	14.1	文件系统程序设计	355
12.6.2	提取一个版本	338	14.1.1	获取文件的状态	355
12.6.3	建立一个新版本	339	14.1.2	搜索目录树	356
12.6.4	显示 SCCS 文件的历史	339	14.2	用文件的系统调用实现 进程通信	358
12.6.5	SCCS 的版本号结构	339	14.2.1	利用文件的系统调用 实现信号量	358
12.6.6	ID 关键字	340	14.2.2	利用管道实现进程间通信	360
12.6.7	其他的 SCCS 的命令	341	14.3	高级进程间通信	362
12.7	其他的软件开发工具	341	14.3.1	消息通信	362
12.7.1	C 程序检查器 Lint	341	14.3.2	共享内存	365
12.7.2	词法分析器和语法分析器	341	14.3.3	信号量	366
习题 12		342	14.4	远程进程间通信 Socket	369
第 13 章	UNIX 通信与万维网 WWW	343			
13.1	检查计算机的连接情况	343			
13.2	获取远程用户信息	343			
13.2.1	显示谁在本地计算机登录 (rwho)	343			

14.4.1	Socket 通信概述	369	15.4.6	break、continue、exit 和 return 语句	414
14.4.2	Socket 系统调用	370	15.5	递归和 Shell 函数	414
14.4.3	TCP 套接字编程	372	15.5.1	递归	414
14.5	UDP 套接字编程	375	15.5.2	Shell 函数	415
14.5.1	UDP 套接字的基本概念	375	15.6	Shell 内部命令	415
14.5.2	UDP 套接字基本函数	375	15.7	Shell 环境	417
14.5.3	UDP 数据报的编程示例	376	15.8	Linux 的 BASH Shell	419
14.6	Solaris 门通信	378	15.8.1	命令编辑和历史	419
14.6.1	Solaris 门的应用编程 接口	378	15.8.2	别名和特征变量	420
14.6.2	Solaris 门的基本通信示例	380	15.8.3	BASH Shell 的环境文件	420
14.7	远程过程调用 RPC	382	习题 15		421
14.7.1	RPC 概述	382	第 16 章	嵌入式 Linux 操作系统	422
14.7.2	RPC 服务地址获取	383	16.1	Linux 与嵌入式操作系统	422
14.7.3	rpcgen 编程指南	383	16.1.1	嵌入式系统的定义	422
14.7.4	客户端身份验证	388	16.1.2	嵌入式操作系统的特点	423
14.8	一个系统程序综合设计的 例子	389	16.1.3	嵌入式 Linux 发展与应用	424
习题 14		400	16.1.4	嵌入式 Linux 操作系统 实现方法	425
第 15 章	Shell 程序设计	401	16.2	嵌入式 Linux 操作系统 构建	425
15.1	Shell 程序和参数	401	16.2.1	Linux 系统的软件层次	425
15.1.1	Shell 程序	401	16.2.2	裁剪编译内核	426
15.1.2	Shell 程序的位置参数	402	16.2.3	准备 Root 文件系统	427
15.2	Shell 变量	402	16.2.4	Root 文件系统裁剪	428
15.2.1	用户定义变量	402	16.2.5	制作 root 文件系统映像	430
15.2.2	系统定义变量	403	16.3	嵌入式 Linux 操作系统 安装	432
15.2.3	Shell 定义变量	405	16.3.1	嵌入式系统的启动	432
15.2.4	参数替换	405	16.3.2	安装嵌入式系统引导程序	433
15.2.5	引号机制	406	16.3.3	x86 系统上嵌入式 Linux 映像安装	435
15.3	测试和求值	406	16.4	嵌入式 Linux 软件开发与 调试	437
15.3.1	测试	406	16.4.1	开发系统与目标系统	437
15.3.2	求值	408	16.4.2	交叉编译环境	438
15.4	控制结构	408	16.4.3	嵌入式 Linux 软件调试 方法	440
15.4.1	顺序控制结构	408			
15.4.2	if 语句	409			
15.4.3	case 语句	411			
15.4.4	for 语句	412			
15.4.5	while 和 until 语句	413			

16.5 嵌入式 Linux 操作系统	16.5.2 Linux 内核与实时性	444
实时性	16.5.3 嵌入式 Linux 实时化	
16.5.1 嵌入式操作系统的实时性	技术	446
要求	习题 16	448

第 3 部分 系统和网络管理篇

第 17 章 UNIX 系统管理	450	第 18 章 UNIX 系统网络管理	462
17.1 系统的启动和关闭	450	18.1 配置 TCP/IP	462
17.1.1 系统自举	450	18.1.1 网络软件的安装	462
17.1.2 系统的运行级	450	18.1.2 设置 TCP/IP 文件	462
17.1.3 初始化程序执行的脚本		18.1.3 设置监听程序	464
(inittab)	451	18.2 网络服务监控进程	465
17.1.4 系统初始化过程	452	18.3 域名服务系统	466
17.1.5 系统的关闭	452	18.3.1 域名服务	466
17.2 用户管理	453	18.3.2 域名系统	466
17.2.1 用户管理涉及的文件	453	18.3.3 域名解析	467
17.2.2 加入新用户的过程	454	18.3.4 逆向域名解析	467
17.2.3 添加用户命令		18.3.5 DNS 数据库	468
(useradd)	455	18.3.6 域名服务器的配置	469
17.2.4 删除用户的注册		18.4 网络文件系统 NFS	470
(userdel)	455	18.4.1 NFS 的概念和原理	470
17.3 誊清 (更新) 系统缓冲区	456	18.4.2 NFS 服务器的设置	470
17.4 检查和修复文件系统	456	18.4.3 NFS 客户机的设置	471
17.5 构造、安装和拆卸文件		18.5 电子邮政系统	473
系统	458	18.5.1 邮件信箱	473
17.5.1 建立特别文件 (mknod)	458	18.5.2 电子邮政设置文件	473
17.5.2 构造文件系统 (mkfs)	458	18.5.3 sendmail 的设置文件	474
17.5.3 安装文件系统 (mount)	459	18.5.4 sendmail 命令的参数	476
17.5.4 拆卸一个文件系统		18.6 WWW 服务器的配置	476
(umount)	459	18.6.1 WWW 服务	477
17.6 定时运行程序	460	18.6.2 虚拟主机服务	480
17.6.1 在指定时刻运行 Shell		18.6.3 代理服务器	481
程序 (at)	460	习题 18	481
17.6.2 时钟精灵 (cron)	460	附录 A: 参考文献	483
习题 17	461		

第 1 部分 原 理 篇

CPU 是计算机系统的核心，操作系统是计算机系统的大脑。近半个世纪来，操作系统吸引了世界上一大群最热情、最有智慧的杰出人才，集中了人类现代创造性思维活动的精髓。操作系统是软件世界的万花筒、博览会，是软件王国中的一顶璀璨的皇冠。能够有机会走进操作系统这个神奇、复杂而又充满诱惑的大千世界，理解其中的基本概念、结构和奥妙的管理机制是十分幸运的。

本篇内容包括：

- 操作系统概论
- 存储管理
- 进程管理
- 进程通信
- 设备管理
- 文件系统
- Windows 操作系统
- 分布式和网络操作系统
- 操作系统安全



第 1 章 操作系统概论

计算机系统由硬件和软件两部分组成。硬件是指人所看得见、摸得着的各种计算机部件,包括存储器、处理器、输入/输出设备及电源、机箱等。软件是指存在于计算机系统中或外存储器中的程序及数据的集合。仅有硬件而没有配备软件的计算机如同一堆废物,只有在配备了软件后,计算机才能“活”起来,为用户解决各种各样的控制、处理和计算问题。

计算机软件分为系统软件和应用软件。系统软件是计算机厂商为了便于用户使用计算机而驻存在计算机硬件(如硬盘和软盘)内的系统支持程序,一般是连同计算机硬件一起出售的。主要的系统软件有操作系统、编译和解释程序、汇编程序、连接装入程序、编辑程序和设备驱动程序等。应用软件是用户为了专门的应用目的,向计算机厂商或其他软件商购买的或自己开发的解决某一类问题的软件,典型的有数据库管理软件、图像处理软件及各种 CAD、CAI、办公软件和杀毒软件等。

在所有的系统软件中,操作系统是一种首要的、最基本的、最重要的系统,也是最庞大、最复杂的系统软件。几十年来,软件界花费了大量的时间和金钱来研究、开发、扩展和完善计算机操作系统,使其获得了飞速的发展。

1.1 什么是操作系统

尽管“操作系统”(Operating Systems, OS)这个名称诞生至今已有了几十年的时间,计算机使用人员一般都知道它,但要对其下一个精确的定义并非轻而易举。很多论述操作系统的书籍从不同的角度对操作系统下了不同的定义,综合起来,通常把操作系统定义为用以控制和管理计算机系统资源,方便用户使用的程序和数据结构的集合。

1. 系统观点——计算机资源管理

在计算机系统中,CPU 是计算机硬件的核心,是计算机系统的核心;操作系统则是计算机软件的核心,是计算机系统的大脑,是整个系统的控制中心,是计算机或智能控制管理系统中首要的、最重要的、最复杂的系统软件。

计算机系统的主要硬件有 CPU、控制器、内存及磁盘、闪存、键盘、鼠标、显示器、打印机、USB 和网络接口等形形色色的外部设备。在计算机系统内运行的各个程序为了各自的“利益”,无时无刻地争夺这些有限而较为昂贵的资源。操作系统的作用类似于城市交通的决策、指挥、控制和调度中心,它组织和管理整个计算机系统的硬件和软件资源,在用户和程序之间分配系统资源,使之协调一致地、高效地完成各种复杂的任务。

现代计算机硬件设备种类越来越多,功能越来越强,控制和操作起来也越来越复杂。如果一个程序员要直接与打印机、磁盘等 I/O 设备打交道,那么就要对每一种设备编制几千、几万条机器指令,这不仅是用户力所不及的,对系统存储的信息来说,也是极其不安全的。操作系统向用户提供了高级且调用简单的服务,掩盖了绝大部分硬件设备复杂的特性和差异,使用户可以免除大量的令人乏味的杂务,而把精力集中在自己所要处理的任務上。

2. 用户观点——用户使用计算机的界面

如果只有裸机,用户几乎不可能使用计算机。为了能够使用户方便、灵活、安全、可靠地使用计算机,就要改造它,充分发挥其潜能。如果在裸机之上覆盖一层 I/O 设备管理软件,

就能使用户较方便地使用外部设备；如果在其上再覆盖一层文件管理软件，用户就很容易存取系统文件和用户文件；每覆盖一层新的软件，就构造了一台功能更强的虚拟计算机。通过 OS，计算机能提供种类更多、质量更高的服务。同样一台计算机硬件，在其中安装 MS-DOS 操作系统或安装 Windows 2003 操作系统，呈现在用户面前的是完全不同的两类“虚拟”计算机，因此，操作系统是用户与计算机硬件之间的接口。用户一般可用以下 3 种方式获得计算机系统提供的服务。

(1) 命令方式

用户可以通过键盘输入有关命令来直接使用计算机，以获得计算机所能提供的服务。用户也可将若干个命令集中存于一个文件中，以批处理的方式连续运行命令，获得多种服务。

(2) 系统调用

用户可以在应用程序中调用操作系统向用户提供的服务程序，以获得系统的服务。

(3) 图形界面

用户可以在窗口环境中通过鼠标、按键、菜单和对话框等方式操纵计算机，这是现代操作系统向多媒体转化的标志之一。

3. 软件观点——程序和数据结构的集合

操作系统是直接和硬件相邻的第一层软件，它是由大量极其复杂的系统程序和众多的数据结构集成的。在计算机中的所有软件中，它起到了核心和控制的作用，其他软件的运行都要依赖它的支持。操作系统是在系统中永久运行的超级程序。

1.2 操作系统的发展历史

1.2.1 早期的计算机和人工操作方式

在 1945 年，世界上第一台通用的数字计算机 ENIAC 诞生以后的早期真空管电子计算机时代，连像汇编语言这样简单而基本的软件也没有，更不用说操作系统了。那时的计算机尽管十分庞大，但也可称其为“个人计算机”，因为计算机每次只能为单个用户服务，系统中所有的资源在这段时间内全部分配给该用户使用。

那时，程序员为了在计算机上算一道题，先要预约登记一段机时，到时他将预先准备好的表示指令和数据的插接板带到机房，由操作员将其插入计算机，并设置好计算机上的各种控制开关，启动计算机运行。程序和数据也可通过控制板上的开关直接送入计算机。假如程序员设计的程序是正确的，并且计算机也没有发生故障，若干小时后他就能获得计算结果，否则将前功尽弃，再约定下次上机时间。

汇编语言和高级语言的问世，以及程序和数据可以通过穿孔纸带或卡片装入计算机，改善了软件的开发环境，但计算机的操作方式并没有多大的改进。程序员首先将记有程序 and 数据的纸带或卡片装到输入设备上，拨动开关，将程序和数据装入内存；接着，程序员要启动汇编或编译程序，将源程序翻译成目标代码；假如程序中不出现语法错误，下一步程序员就可通过控制台按键设定程序执行的起始地址，并启动程序的执行。

在程序的执行期间，程序员要观察控制台上的各种指示灯以监视程序的运行情况。如果发现错误，并且还未用完所预约的上机时间，就可通过指示灯检查存储器中的内容，直接在控制台上进行调试和排错。如果程序运行正常，最终将结果在电传打字机等输出设备上打印出来。

总之，在早期的计算机系统中，每一次独立的运行都需要很多的人工干预，操作过程烦