Linux
技术内幕

罗秋明 著

清华大学出版社

Linux 技术内幕

罗秋明 著

清华大学出版社
北京

内 容 简 介

本书内容分成两篇，第一篇是基本框架，第二篇是文件系统及相关内容。在第一篇的各章中：第 1 章先对 Linux 进行简要介绍并给出本书分析时所用的模型；第 2 章从 C 语言程序、可执行文件到进程的虚存空间影像的全过程作为起点，给读者建立起进程用户空间管理的概念；然后第 3 章讨论物理页帧如何支撑这些虚存空间，并且讨论了与物理空间一致的内核空间的管理；接着第 4 章就是进程的概念、进程的组织、进程切换和进程的创建撤销等活动；第 5 章专门讨论进程调度和负载均衡问题；后面 4 章继续讨论进程间通信、系统调用、内核的并发活动和同步问题。第二篇开始讨论盘根错节的文件系统：先在第 10 章分析文件系统和 VFS 的基本概念；然后在第 11 章讨论页高速缓存及块缓存；第 12 章分析了 VFS 的通用文件访问操作；第 13 章讨论 ext2 文件系统的具体格式和操作细节；接着第 14 章讨论同步；第 15 章讨论内存回收问题；最后第 16 章和第 17 章讨论设备管理和块设备问题。

本书以内存模型和时空模型为主要参考来分析各章的相关内容，给出了比较直观的图示，这不仅对初学者非常有用，对希望了解 Linux 内核的读者和相关开发人员也非常有参考价值。

图书在版编目（CIP）数据

Linux 技术内幕 / 罗秋明著. —北京：清华大学出版社，2017
ISBN 978-7-302-45100-6

Ⅰ. ①L…　Ⅱ. ①罗…　Ⅲ. ①Linux 操作系统 – 高等学校 – 教材　Ⅳ. ①TP316.85

中国版本图书馆 CIP 数据核字（2016）第 227116 号

责任编辑： 龙启铭　薛　阳
封面设计： 何凤霞
责任校对： 时翠兰
责任印制： 杨　艳

出版发行： 清华大学出版社
网　　址：http://www.tup.com.cn, http://www.wqbook.com
地　　址：北京清华大学学研大厦 A 座　　邮　　编：100084
社 总 机：010-62770175　　邮　　购：010-62786544
投稿与读者服务：010-62776969，c-service@tup.tsinghua.edu.cn
质 量 反 馈：010-62772015，zhiliang@tup.tsinghua.edu.cn
课 件 下 载：http://www.tup.com.cn,010-62795954
印 刷 者： 清华大学印刷厂
装 订 者： 三河市新茂装订有限公司
经　　销： 全国新华书店
开　　本： 185mm×260mm　　**印　　张：** 49.5　　**字　　数：** 1200 千字
版　　次： 2017 年 1 月第 1 版　　**印　　次：** 2017 年 1 月第 1 次印刷
印　　数： 1～2000
定　　价： 98.00 元

产品编号：055866-01

序

此书献给所有热爱 Linux 内核的读者。我们学习它而获得的最大乐趣和收获来源于满足了我们的好奇心。作者最早接触操作系统内核是 20 世纪 90 年代初期在西安电子科技大学校园，一个偶然机会获得了一本西安电子科技大学老师写的关于揭秘 DOS 内核的书籍，如饥似渴地一口气读完，相信一些读者也有过类似的体验。不过很快 Windows 3.1、Windows NT 来了，可获得的操作系统看似要封闭在一个黑盒子里了。后来在桂林电子科技大学读研的日子里，我知道了 Linux 这个新东西，不过当时正在研究硬件——用 VHDL 写一个测试控制器，所以并没有深入了解。直到后来在华中科技大学读博士的时候，从浙江大学毛德操老师的《Linux 内核源代码情景分析》一书中又找到了当年的感觉。直到前几年遇到一个研究所的年轻人以读过内核而沾沾自喜，于是我觉得应该写一本关于 Linux 内核的书，降低 Linux 内核阅读的难度，让它变得平民化一点儿，这就是本书的由来之一。当然更重要的是在陈国良院士的研究团队中，我参与了系统软件的相关工作，这也算是交的一份作业吧。

在学习了本科操作系统课程并且对系统编程有一定了解之后，不少同学希望通过 Linux 源代码的学习来进一步掌握操作系统的细节。但是在操作系统原理和 Linux 代码之间有明显的学习上的鸿沟，两者之间侧重点的不同使得源代码的学习曲线非常陡峭。

操作系统原理性教材以 4 大管理（处理机、内存、设备和文件）为基线，主要讲述的是原理和算法性的内容，而 Linux 内核分析的书籍（包括中文和英文）则偏向于“解剖”代码本身。作者希望做到在讨论“How”的基础上，力求进一步探究“Why”，在此思路下并借助类似《Linux 内核源代码情景分析》等专著中的方式，本教材选择以主题分析的方式，从应用程序（需求）和操作系统实现两个方面同时入手，力求展现用户编程需求和内核实现的对应关系，除了“代码解剖”外，加强对各个“器官组织”之间联系的描述，书中有大量的交叉引用便用于此目的。

本教材将现有书籍通常都不在意的概念给读者澄清，从用户进程在物理机器上运行的角度，将源代码大量的细节用提出正确需求问题的形式统领在一起，结合内存模型和时空模型来分析所涉及的问题，将相关原理、过程用图示的直观方法展示给读者，降低了学习难度。

本书并不打算替代操作系统原理性教材和源代码分析书籍，读者应当先学习操作系统和系统编程的课程，最好阅读过一两本内核的书籍然后才阅读此书，其间有必要的时候，读者可以进一步研读其他各种源代码分析的专著。此书并不是开天辟地之作，书中许多内容参考了 *Understanding the Linux Kernel* 和 *Professional Linux Kernel Architecture* 的相关内容。

由于内核代码量非常大，而作者有限的精力和能力难免有理解上的偏差。因此如果读者能在阅读本书过程中快速地建立起“比较完整”和“大体上正确”的认识，那将是作者

最大的欣慰。对于书中的不足与疏漏，欢迎读者将问题反馈到 lqm@szu.edu.cn，您的反馈将可能节约其他读者宝贵的时间，因此极具价值。

罗秋明
深大荔园

致　谢

本书得以完成首先要感谢深圳大学——为我提供了一个舒适的工作环境，不必为生计而奔波。也要感谢计算机学院的陈国良院士和明仲院长，他们为本学院老师安排了合理的工作量，从而我能有空闲时间将本书完成。当然，计算机学院高性能计算所的老师们融洽地工作也是我能有精力进行编写的保证。

其次要感谢我的几个研究生，肖峰完成了 slab 机制、文件系统挂载模式等小节的主体内容，周远远、张义军、孔畅、刘国强和刘杰进行了繁重的校对工作。本科学生杨诗达对初稿进行了阅读并协助进行校对。若没有这些同学的帮助，此书将要延迟许久才能与读者见面。

最后要感谢我的家人，在我下班回家专心在计算机前写作时，他们毫无怨言。这是对我最大的支持。

目　录

第一篇　基 本 框 架

第 1 章　Linux 内核概述 3
1.1　UNIX 与 Linux 3
1.1.1　UNIX 3
1.1.2　Linux 3
1.1.3　宏内核与微内核 6
1.1.4　Linux 内核源码及版本 6
1.2　Linux 内核模型 11
1.2.1　多视角下的内核 11
1.2.2　功能模型 14
1.2.3　内存模型 15
1.2.4　时空模型 16
1.2.5　特权模型 17
1.3　本书局限性 18
小结 18
第 2 章　进程影像 20
2.1　从源代码到进程 20
2.1.1　源代码、目标文件 20
2.1.2　可执行文件与进程影像 26
2.2　proc 中的进程 32
2.2.1　进程内存空间 32
2.2.2　进程运行状态等信息 33
2.3　进程空间 35
2.3.1　进程空间描述符 36
2.3.2　虚存区域 VMA 40
2.3.3　VMA 属性 45
2.4　ELF 可执行文件装入过程 48
2.4.1　ELF 装入函数 49
2.4.2　ELF 格式 49
2.5　进程空间的动态变化 54
2.5.1　VMA 上的操作 54
2.5.2　文件映射 55
2.5.3　堆的调整 58

2.5.4 栈的变化 60
2.6 并发的进程空间 60
小结 61
第 3 章 虚拟空间的物理支撑 63
3.1 物理内存组织与管理 64
3.1.1 节点与内存域 64
3.1.2 物理页帧 79
3.1.3 buddy 系统 84
3.1.4 页帧迁移 94
3.1.5 内存热插拔 98
3.2 地址映射与页表 99
3.2.1 分页机制与页表 99
3.2.2 缺页异常 103
3.3 内核空间 104
3.3.1 一致映射与高端内存 104
3.3.2 一致内存分配 108
3.3.3 非一致内存分配 109
3.3.4 slub 分配器 115
小结 130
第 4 章 进程组织与基础行为 131
4.1 进程组织管理 131
4.1.1 PCB 进程控制块 131
4.1.2 命名空间 135
4.1.3 进程标识 141
4.1.4 进程间关系 148
4.1.5 进程资源限制 151
4.2 进程创建与撤销 152
4.2.1 进程创建 152
4.2.2 execve 系统调用 159
4.2.3 内核线程 160
4.2.4 Linux 进程树 162
4.2.5 进程的撤销 167
4.3 进程切换 167
4.3.1 切换时机 168
4.3.2 切换过程 169
4.3.3 切换示例 178
小结 181
第 5 章 进程调度与负载均衡 193
5.1 调度与均衡基本框架 193

5.2 进程状态与转换 194
5.2.1 进程调度状态 194
5.2.2 进程状态变迁 196
5.3 进程调度 198
5.3.1 调度框架 198
5.3.2 完全公平调度 215
5.3.3 实时调度 230
5.3.4 STOP 和 IDLE 调度类 234
5.3.5 调度控制与 proc 接口 236
5.4 负载均衡 241
5.4.1 处理器层次结构 242
5.4.2 调度的层次管理 243
5.4.3 CFS 任务的负载均衡 250
5.4.4 实时负载均衡 254
小结 258
第 6 章 进程间通信与同步 259
6.1 管道通信 260
6.1.1 无名管道 260
6.1.2 命名管道 261
6.1.3 管道数据结构 261
6.1.4 管道操作 266
6.2 System V IPC 269
6.2.1 IPC 标识与命名空间 269
6.2.2 IPC 公共框架 275
6.2.3 IPC 信号量 279
6.2.4 IPC 消息队列 283
6.2.5 IPC 共享内存 287
6.3 信号 290
6.3.1 信号分类 290
6.3.2 数据结构 294
6.3.3 信号产生与发送 302
6.3.4 信号的递交和处理 305
小结 312
第 7 章 内核活动 313
7.1 中断分类 313
7.1.1 x86 的中断和异常 314
7.1.2 后半部机制与软中断 315
7.1.3 中断相关概念的关系 315
7.2 中断处理 316

7.2.1 中断号 317
7.2.2 中断描述符表 320
7.2.3 公共入口 325
7.2.4 异常处理 329
7.3 高层中断处理 332
7.3.1 转向高层处理 333
7.3.2 中断的高层数据结构 336
7.3.3 中断返回处理 342
7.3.4 中断的线程化 349
7.4 中断嵌套与中断管理 350
7.4.1 中断嵌套与中断上下文 350
7.4.2 中断管理 353
7.5 软中断和 tasklet 355
7.5.1 中断的下半部 355
7.5.2 软中断执行时机 356
7.5.3 相关数据结构 360
7.5.4 软中断的执行 361
7.5.5 软中断的相关操作 363
7.5.6 tasklet 365
7.6 工作队列 369
7.6.1 工作队列机制 369
7.6.2 cmwq 数据结构 371
7.6.3 工作项 371
7.6.4 cmwq 工作队列 375
7.6.5 工作者池 worker_pool 383
7.6.6 并发度、应急处理等 386
7.7 系统调用 388
7.7.1 POSIX API、C 库和系统调用 388
7.7.2 系统调用的实现 389
小结 399
第 8 章 时间管理 400
8.1 时间管理框架 400
8.1.1 基本概念 400
8.1.2 时间中断和事件 406
8.1.3 clock_event_device 与 tick_device 407
8.1.4 TIMER_SOFTIRQ 软中断 413
8.1.5 timekeeper 414
8.2 定时器 416
8.2.1 低分辨率定时器 416

8.2.2 高精度定时器 419
8.2.3 模拟 tick 事件 422
8.2.4 通知链技术 423
小结 424
第 9 章 内核并发与同步 426
9.1 同步的需求 426
9.1.1 内核并发情形 426
9.1.2 内核抢占 429
9.2 内核共享变量的保护 432
9.2.1 被保护对象 432
9.2.2 保护原则 433
9.2.3 禁止内核并发 435
9.3 内核同步手段 437
9.3.1 原子操作 437
9.3.2 自旋锁、读写锁和顺序锁 439
9.3.3 RCU 机制 444
9.3.4 顺序和屏障 447
9.3.5 信号量与互斥量 448
9.3.6 等待队列与完成变量 452
9.3.7 每 CPU 变量 455
小结 458

第二篇 盘根错节的文件系统

第 10 章 文件系统 461
10.1 文件系统的抽象层次 461
10.1.1 进程视角下的文件 462
10.1.2 VFS 虚拟文件系统 468
10.1.3 多角度分层模型 472
10.2 VFS 核心对象 475
10.2.1 文件对象 475
10.2.2 目录项对象 479
10.2.3 索引节点对象 484
10.2.4 超级块对象 490
10.3 文件系统类型与挂载 495
10.3.1 文件系统类型与注册 495
10.3.2 挂载操作 503
10.3.3 挂载模式 513
10.3.4 特殊文件系统 519
小结 525

第 11 章 页缓存和块缓存 526
11.1 页高速缓存 527
11.1.1 address_space 528
11.1.2 页高速缓存的组织 530
11.1.3 反向映射 534
11.2 块高速缓存 540
11.2.1 块缓存 540
11.2.2 LRU 块缓存 542
11.2.3 块缓存操作 543
小结 547
第 12 章 VFS 的文件操作 548
12.1 VFS 系统调用 548
12.2 open()与 close()系统调用 549
12.2.1 open 的框架 549
12.2.2 文件定位过程 552
12.2.3 close()系统调用 558
12.3 读/写系统调用 558
12.3.1 入口代码 560
12.3.2 通用 write 写例程 568
12.3.3 通用 read 读例程 570
12.3.4 其他读写细节 583
12.3.5 向 BIO 层提交请求 587
小结 588
第 13 章 ext2 文件系统 590
13.1 ext2 磁盘数据结构 590
13.1.1 磁盘分区的组织 590
13.1.2 块组描述符和位图 591
13.1.3 盘上和内存数据结构 592
13.2 ext2 超级块 593
13.2.1 ext2 超级块数据结构 593
13.2.2 挂载与访问 597
13.3 ext2 索引节点 598
13.3.1 盘上 ext2 索引节点 599
13.3.2 内存 ext2 索引节点 601
13.3.3 inode_operations 603
13.3.4 ext2 地址空间与文件操作 604
13.4 目录及目录项 607
13.4.1 ext2_dir_entry 607
13.4.2 ext2_lookup() 609

小结 610
第 14 章 页缓存同步（回写） 611
14.1 同步/回写、交换与回收 611
14.2 脏页同步（回写） 613
14.2.1 回写机制演变 613
14.2.2 同步时机与框架 615
14.2.3 基本数据结构 617
14.3 回写接口 627
14.3.1 sync 系列系统调用 627
14.3.2 sys_sync() 628
14.3.3 sys_syncfs 633
14.3.4 单个文件的同步 635
14.3.5 被动回写 637
14.4 回写工作队列 638
14.4.1 初始化 638
14.4.2 工作队列处理函数 640
14.5 回写操作 643
14.5.1 do_writepages() 644
14.5.2 ext2_writepages() 645
14.5.3 回写等待 647
小结 649
第 15 章 内存回收与交换 650
15.1 页帧回收 650
15.1.1 直接释放 650
15.1.2 LRU 页帧组织 651
15.1.3 PFRA 回收算法 655
15.2 核心回收操作 659
15.2.1 shrink_zone() 659
15.2.2 shrink_slab() 666
15.2.3 解除页表映射 667
15.3 交换 667
15.3.1 交换功能 668
15.3.2 交换分区 668
15.3.3 交换缓存 673
小结 675
第 16 章 设备管理 676
16.1 设备管理组织 676
16.1.1 设备驱动模型 677
16.1.2 sysfs 678

16.1.3 基础组件 679
16.1.4 容器 682
16.2 设备的 VFS 接口 688
16.2.1 设备文件 689
16.2.2 从 VFS 中访问设备 691
16.3 字符设备 693
16.3.1 设备的散列组织 694
16.3.2 初始化与注册 695
16.3.3 打开字符设备 696
16.4 PCI 设备 698
16.4.1 pci_bus_type 和 pci_bus 699
16.4.2 pci_driver 700
16.4.3 pci_dev 701
16.4.4 uevent 704
小结 705
第 17 章 块设备 706
17.1 基本概念 706
17.1.1 块设备层 706
17.1.2 传送单位 708
17.2 块设备层组件 709
17.2.1 磁盘与磁盘分区 709
17.2.2 块设备 712
17.2.3 请求队列 715
17.3 提交请求及处理 728
17.3.1 plug/unplug 机制 728
17.3.2 提交请求 733
17.3.3 提交到驱动程序 742
17.3.4 硬盘的 request_fn 744
17.3.5 中断处理 746
17.4 IO 调度 749
17.4.1 IO 调度器 749
17.4.2 调度器数据结构 751
17.5 初始化及注册 754
17.5.1 块设备初始化 754
17.5.2 硬盘初始化 759
小结 764
附录 765
后记 774

第一篇

基本框架

第 1 章　Linux 内核概述
第 2 章　进程影像
第 3 章　虚拟空间的物理支撑
第 4 章　进程组织与基础行为
第 5 章　进程调度与负载均衡
第 6 章　进程间通信与同步
第 7 章　内核活动
第 8 章　时间管理
第 9 章　内核并发与同步

第一篇

操作基础

第 1 章　Linux 内核概述

Linux 内核是一个庞然大物，它涉及大量复杂的对象、机制和代码。任何尝试从单一角度去观察所得到的结论一定是类似于盲人摸象的情形——正确但又不完整。读者必须先从宏观和完整的角度去了解想要探究的 Linux 内核，弄清楚其设计目的和缘由、各部件之间的关系，而不仅仅是急于努力深究里面的细节。

下面简要地介绍 Linux 系统及内核相关的基本概念，然后引出观察和分析内核的几个不同视角，最后以贯穿本书的几个分析模型作为本章的结束。

1.1　UNIX 与 Linux

讨论 Linux 就不可避免地要涉及 UNIX，因此本书把 UNIX 作为分析讨论的起点。

1.1.1　UNIX

UNIX 诞生至今已经有四十多年，但计算机业内仍然认为它是操作系统中的典范。由 Ken Thompson 和 Dennis Ritchie 等人创始的 UNIX 已经成为传奇，它源于一个失败的多用户操作系统 Multics。在 Multics 项目失败之后，Tompson 在一台闲置 PDP-7 机器上重新实现了一个新的操作系统——UNiplexed Information and Computing System（UNICS），后来改为 UNIX。从此拉开了 UNIX 的历史序幕。特别是当 UNIX 用 C 语言改写后便于移植到不同平台，出现了 BSD、Solaris、HP-UN 和 Linux 等众多变体（见图 1-1）。

由于最初的优良设计并结合以后多年的创新与提高，UNIX 系统成为一个强大、健壮和易用的操作系统。以下几个特点和优点是 UNIX 强大的主要原因：首先，UNIX 很简洁，仅提供两百多个系统调用，从而与用户态彻底隔离（不像 Windows 那样系统调用和 API 混在一起）。其次，在 UNIX 中有“万物皆文件”的说法，使得对各种数据和设备的操作都统一在 VFS 虚拟文件系统接口下，使用统一的 open()、read()、write()和 close()等操作方法。第三，UNIX 内核和相关工具主要使用 C 语言编写，便于移植。最后，归于它的优秀性能，例如快速的进程创建、高效的调度算法和简单稳定的进程间通信机制，等等。

今天，UNIX 已经发展成一个支持抢占式多任务、多线程、虚拟内存、动态链接和 TCP/IP 协议等特性的现代操作系统。它不仅用于大型集群计算机、小型计算机、个人计算机，也用于各种嵌入式及手持设备上。

1.1.2　Linux

从图 1-1 可知，1991 年，Linus Torvalds 为当时推出的、使用 Intel 386 处理器的 PC 开发了一款全新的操作系统——Linux。当时 Linus 作为芬兰赫尔辛基大学的一名学生因无法