

Contents

Introduction	xxvii
Chapter 1: Introduction and Overview	1
Tasks of the Kernel	2
Implementation Strategies	3
Elements of the Kernel	3
Processes, Task Switching, and Scheduling	4
UNIX Processes	4
Address Spaces and Privilege Levels	7
Page Tables	11
Allocation of Physical Memory	13
Timing	16
System Calls	17
Device Drivers, Block and Character Devices	17
Networks	18
Filesystems	18
Modules and Hotplugging	18
Caching	20
List Handling	20
Object Management and Reference Counting	22
Data Types	25
... and Beyond the Infinite	27
Why the Kernel Is Special	28
Some Notes on Presentation	29
Summary	33
Chapter 2: Process Management and Scheduling	35
Process Priorities	36
Process Life Cycle	38
Preemptive Multitasking	40
Process Representation	41
Process Types	47
Namespaces	47

Contents

Process Identification Numbers	54
Task Relationships	62
Process Management System Calls	63
Process Duplication	63
Kernel Threads	77
Starting New Programs	79
Exiting Processes	83
Implementation of the Scheduler	83
Overview	84
Data Structures	86
Dealing with Priorities	93
Core Scheduler	99
The Completely Fair Scheduling Class	106
Data Structures	106
CFS Operations	107
Queue Manipulation	112
Selecting the Next Task	113
Handling the Periodic Tick	114
Wake-up Preemption	115
Handling New Tasks	116
The Real-Time Scheduling Class	117
Properties	118
Data Structures	118
Scheduler Operations	119
Scheduler Enhancements	121
SMP Scheduling	121
Scheduling Domains and Control Groups	126
Kernel Preemption and Low Latency Efforts	127
Summary	132
Chapter 3: Memory Management	133
Overview	133
Organization in the (N)UMA Model	136
Overview	136
Data Structures	138
Page Tables	153
Data Structures	154
Creating and Manipulating Entries	161
Initialization of Memory Management	161
Data Structure Setup	162
Architecture-Specific Setup	169
Memory Management during the Boot Process	191

Contents

Management of Physical Memory	199
Structure of the Buddy System	199
Avoiding Fragmentation	201
Initializing the Zone and Node Data Structures	209
Allocator API	215
Reserving Pages	222
Freeing Pages	240
Allocation of Discontiguous Pages in the Kernel	244
Kernel Mappings	251
The Slab Allocator	256
Alternative Allocators	258
Memory Management in the Kernel	259
The Principle of Slab Allocation	261
Implementation	265
General Caches	283
Processor Cache and TLB Control	285
Summary	287
Chapter 4: Virtual Process Memory	289
Introduction	289
Virtual Process Address Space	290
Layout of the Process Address Space	290
Creating the Layout	294
Principle of Memory Mappings	297
Data Structures	298
Trees and Lists	299
Representation of Regions	300
The Priority Search Tree	302
Operations on Regions	306
Associating Virtual Addresses with a Region	306
Merging Regions	308
Inserting Regions	309
Creating Regions	310
Address Spaces	312
Memory Mappings	314
Creating Mappings	314
Removing Mappings	317
Nonlinear Mappings	319
Reverse Mapping	322
Data Structures	323
Creating a Reverse Mapping	324
Using Reverse Mapping	325

Contents

Managing the Heap	327
Handling of Page Faults	330
Correction of Userspace Page Faults	336
Demand Allocation/Paging	337
Anonymous Pages	339
Copy on Write	340
Getting Nonlinear Mappings	341
Kernel Page Faults	341
Copying Data between Kernel and Userspace	344
Summary	345
Chapter 5: Locking and Interprocess Communication	347
Control Mechanisms	348
Race Conditions	348
Critical Sections	349
Kernel Locking Mechanisms	351
Atomic Operations on Integers	352
Spinlocks	354
Semaphores	355
The Read-Copy-Update Mechanism	357
Memory and Optimization Barriers	359
Reader/Writer Locks	361
The Big Kernel Lock	361
Mutexes	362
Approximate Per-CPU Counters	364
Lock Contention and Fine-Grained Locking	365
System V Interprocess Communication	366
System V Mechanisms	366
Semaphores	367
Message Queues	376
Shared Memory	380
Other IPC Mechanisms	381
Signals	381
Pipes and Sockets	389
Summary	390
Chapter 6: Device Drivers	391
I/O Architecture	391
Expansion Hardware	392
Access to Devices	397
Device Files	397
Character, Block, and Other Devices	397

Contents

Device Addressing Using loctls	400
Representation of Major and Minor Numbers	401
Registration	403
Association with the Filesystem	406
Device File Elements in Inodes	406
Standard File Operations	407
Standard Operations for Character Devices	407
Standard Operations for Block Devices	408
Character Device Operations	409
Representing Character Devices	409
Opening Device Files	409
Reading and Writing	412
Block Device Operations	412
Representation of Block Devices	413
Data Structures	415
Adding Disks and Partitions to the System	423
Opening Block Device Files	425
Request Structure	427
BIOs	430
Submitting Requests	432
I/O Scheduling	438
Implementation of loctls	441
Resource Reservation	442
Resource Management	442
I/O Memory	445
I/O Ports	446
Bus Systems	448
The Generic Driver Model	449
The PCI Bus	454
USB	463
Summary	471
Chapter 7: Modules	473
Overview	473
Using Modules	474
Adding and Removing	474
Dependencies	477
Querying Module Information	478
Automatic Loading	480
Inserting and Deleting Modules	483
Module Representation	483
Dependencies and References	488

Contents

Binary Structure of Modules	491
Inserting Modules	496
Removing Modules	505
Automation and Hotplugging	506
Automatic Loading with <code>kmod</code>	507
Hotplugging	508
Version Control	511
Checksum Methods	512
Version Control Functions	515
Summary	517
Chapter 8: The Virtual Filesystem	519
Filesystem Types	520
The Common File Model	521
Inodes	522
Links	522
Programming Interface	523
Files as a Universal Interface	524
Structure of the VFS	525
Structural Overview	525
Inodes	527
Process-Specific Information	532
File Operations	537
Directory Entry Cache	542
Working with VFS Objects	547
Filesystem Operations	548
File Operations	565
Standard Functions	572
Generic Read Routine	573
The <code>fault</code> Mechanism	576
Permission-Checking	578
Summary	581
Chapter 9: The Extended Filesystem Family	583
Introduction	583
Second Extended Filesystem	584
Physical Structure	585
Data Structures	592

Contents

Creating a Filesystem	608
Filesystem Actions	610
Third Extended Filesystem	637
Concepts	638
Data Structures	639
Summary	642
Chapter 10: Filesystems without Persistent Storage	643
The <code>proc</code> Filesystem	644
Contents of <code>/proc</code>	644
Data Structures	652
Initialization	655
Mounting the Filesystem	657
Managing <code>/proc</code> Entries	660
Reading and Writing Information	664
Task-Related Information	666
System Control Mechanism	671
Simple Filesystems	680
Sequential Files	680
Writing Filesystems with Libfs	684
The Debug Filesystem	687
Pseudo Filesystems	689
Sysfs	689
Overview	690
Data Structures	690
Mounting the Filesystem	695
File and Directory Operations	697
Populating Sysfs	704
Summary	706
Chapter 11: Extended Attributes and Access Control Lists	707
Extended Attributes	707
Interface to the Virtual Filesystem	708
Implementation in Ext3	714
Implementation in Ext2	721
Access Control Lists	722
Generic Implementation	722
Implementation in Ext3	726
Implementation in Ext2	732
Summary	732

Contents

Chapter 12: Networks	733
Linked Computers	734
ISO/OSI and TCP/IP Reference Model	734
Communication via Sockets	738
Creating a Socket	738
Using Sockets	740
Datagram Sockets	744
The Layer Model of Network Implementation	745
Networking Namespaces	747
Socket Buffers	749
Data Management Using Socket Buffers	750
Management Data of Socket Buffers	753
Network Access Layer	754
Representation of Network Devices	755
Receiving Packets	760
Sending Packets	768
Network Layer	768
IPv4	769
Receiving Packets	771
Local Delivery to the Transport Layer	772
Packet Forwarding	774
Sending Packets	775
Netfilter	778
IPv6	783
Transport Layer	785
UDP	785
TCP	787
Application Layer	799
Socket Data Structures	799
Sockets and Files	803
The <code>socketcall</code> System Call	804
Creating Sockets	805
Receiving Data	807
Sending Data	808
Networking from within the Kernel	808
Communication Functions	808
The Netlink Mechanism	810
Summary	817

Contents

Chapter 13: System Calls	819
Basics of System Programming	820
Tracing System Calls	820
Supported Standards	823
Restarting System Calls	824
Available System Calls	826
Implementation of System Calls	830
Structure of System Calls	830
Access to Userspace	837
System Call Tracing	838
Summary	846
Chapter 14: Kernel Activities	847
Interrupts	848
Interrupt Types	848
Hardware IRQs	849
Processing Interrupts	850
Data Structures	853
Interrupt Flow Handling	860
Initializing and Reserving IRQs	864
Servicing IRQs	866
Software Interrupts	875
Starting SoftIRQ Processing	877
The SoftIRQ Daemon	878
Tasklets	879
Generating Tasklets	880
Registering Tasklets	880
Executing Tasklets	881
Wait Queues and Completions	882
Wait Queues	882
Completions	887
Work Queues	889
Summary	891
Chapter 15: Time Management	893
Overview	893
Types of Timers	893
Configuration Options	896

Contents

Implementation of Low-Resolution Timers	897
Timer Activation and Process Accounting	897
Working with Jiffies	900
Data Structures	900
Dynamic Timers	902
Generic Time Subsystem	907
Overview	908
Configuration Options	909
Time Representation	910
Objects for Time Management	911
High-Resolution Timers	920
Data Structures	921
Setting Timers	925
Implementation	926
Periodic Tick Emulation	931
Switching to High-Resolution Timers	932
Dynamic Ticks	933
Data Structures	934
Dynamic Ticks for Low-Resolution Systems	935
Dynamic Ticks for High-Resolution Systems	938
Stopping and Starting Periodic Ticks	939
Broadcast Mode	943
Implementing Timer-Related System Calls	944
Time Bases	944
The <code>alarm</code> and <code>setitimer</code> System Calls	945
Getting the Current Time	947
Managing Process Times	947
Summary	948
Chapter 16: Page and Buffer Cache	949
Structure of the Page Cache	950
Managing and Finding Cached Pages	951
Writing Back Modified Data	952
Structure of the Buffer Cache	954
Address Spaces	955
Data Structures	956
Page Trees	958
Operations on Address Spaces	961
Implementation of the Page Cache	966
Allocating Pages	966
Finding Pages	967

Contents

Waiting on Pages	968
Operations with Whole Pages	969
Page Cache Readahead	970
Implementation of the Buffer Cache	974
Data Structures	975
Operations	976
Interaction of Page and Buffer Cache	977
Independent Buffers	982
Summary	988
Chapter 17: Data Synchronization	989
Overview	989
The <code>pdflush</code> Mechanism	991
Starting a New Thread	993
Thread Initialization	994
Performing Actual Work	995
Periodic Flushing	996
Associated Data Structures	996
Page Status	996
Writeback Control	998
Adjustable Parameters	1000
Central Control	1000
Superblock Synchronization	1002
Inode Synchronization	1003
Walking the Superblocks	1003
Examining Superblock Inodes	1003
Writing Back Single Inodes	1006
Congestion	1009
Data Structures	1009
Thresholds	1010
Setting and Clearing the Congested State	1011
Waiting on Congested Queues	1012
Forced Writeback	1013
Laptop Mode	1015
System Calls for Synchronization Control	1016
Full Synchronization	1016
Synchronization of Inodes	1018
Synchronization of Individual Files	1019
Synchronization of Memory Mappings	1021
Summary	1022

Contents

Chapter 18: Page Reclaim and Swapping	1023
Overview	1023
Swappable Pages	1024
Page Thrashing	1025
Page-Swapping Algorithms	1026
Page Reclaim and Swapping in the Linux Kernel	1027
Organization of the Swap Area	1028
Checking Memory Utilization	1029
Selecting Pages to Be Swapped Out	1029
Handling Page Faults	1029
Shrinking Kernel Caches	1030
Managing Swap Areas	1030
Data Structures	1030
Creating a Swap Area	1035
Activating a Swap Area	1036
The Swap Cache	1039
Identifying Swapped-Out Pages	1041
Structure of the Cache	1044
Adding New Pages	1045
Searching for a Page	1050
Writing Data Back	1051
Page Reclaim	1052
Overview	1053
Data Structures	1055
Determining Page Activity	1057
Shrinking Zones	1062
Isolating LRU Pages and Lumpy Reclaim	1065
Shrinking the List of Active Pages	1068
Reclaiming Inactive Pages	1072
The Swap Token	1079
Handling Swap-Page Faults	1082
Swapping Pages in	1083
Reading the Data	1084
Swap Readahead	1085
Initiating Memory Reclaim	1086
Periodic Reclaim with <code>kswapd</code>	1087
Swap-out in the Event of Acute Memory Shortage	1090
Shrinking Other Caches	1092
Data Structures	1092
Registering and Removing Shrinkers	1093
Shrinking Caches	1093
Summary	1095

Contents

Chapter 19: Auditing	1097
Overview	1097
Audit Rules	1099
Implementation	1100
Data Structures	1100
Initialization	1106
Processing Requests	1107
Logging Events	1108
System Call Auditing	1110
Summary	1116
Appendix A: Architecture Specifics	1117
Overview	1117
Data Types	1118
Alignment	1119
Memory Pages	1119
System Calls	1120
String Processing	1120
Thread Representation	1122
IA-32	1122
IA-64	1124
ARM	1126
Sparc64	1128
Alpha	1129
Mips	1131
PowerPC	1132
AMD64	1134
Bit Operations and Endianness	1135
Manipulation of Bit Chains	1135
Conversion between Byte Orders	1136
Page Tables	1137
Miscellaneous	1137
Checksum Calculation	1137
Context Switch	1137
Finding the Current Process	1138
Summary	1139
Appendix B: Working with the Source Code	1141
Organization of the Kernel Sources	1141
Configuration with Kconfig	1144
A Sample Configuration File	1144

Contents

Language Elements of Kconfig	1147
Processing Configuration Information	1152
Compiling the Kernel with Kbuild	1154
Using the Kbuild System	1154
Structure of the Makefiles	1156
Useful Tools	1160
LXR	1161
Patch and Diff	1163
Git	1165
Debugging and Analyzing the Kernel	1169
GDB and DDD	1170
Local Kernel	1171
KGDB	1172
User-Mode Linux	1173
Summary	1174
Appendix C: Notes on C	1175
How the GNU C Compiler Works	1175
From Source Code to Machine Program	1176
Assembly and Linking	1180
Procedure Calls	1180
Optimization	1185
Inline Functions	1192
Attributes	1192
Inline Assembler	1194
__builtin Functions	1198
Pointer Arithmetic	1200
Standard Data Structures and Techniques of the Kernel	1200
Reference Counters	1200
Pointer Type Conversions	1201
Alignment Issues	1202
Bit Arithmetic	1203
Pre-Processor Tricks	1206
Miscellaneous	1207
Doubly Linked Lists	1209
Hash Lists	1214
Red-Black Trees	1214
Radix Trees	1216
Summary	1221

Contents

Appendix D: System Startup	1223
Architecture-Specific Setup on IA-32 Systems	1224
High-Level Initialization	1225
Subsystem Initialization	1225
Summary	1239
Appendix E: The ELF Binary Format	1241
Layout and Structure	1241
ELF Header	1243
Program Header Table	1244
Sections	1246
Symbol Table	1248
String Tables	1249
Data Structures in the Kernel	1250
Data Types	1250
Headers	1251
String Tables	1257
Symbol Tables	1257
Relocation Entries	1259
Dynamic Linking	1263
Summary	1265
Appendix F: The Kernel Development Process	1267
Introduction	1267
Kernel Trees and the Structure of Development	1268
The Command Chain	1269
The Development Cycle	1269
Online Resources	1272
The Structure of Patches	1273
Technical Issues	1273
Submission and Review	1277
Linux and Academia	1281
Some Examples	1282
Adopting Research	1284
Summary	1287
References	1289
Index	1293

