



8th Edition

OPERATING

SYSTEM

CONCEPTS



Silberschatz, Galvin, Gagne

ASSOCIATE PUBLISHER	Dan Sayre
EDITORIAL ASSISTANT	Carolyn Weisman
SENIOR PRODUCTION EDITOR	Ken Santor
COVER DESIGNER	Howard Grossman
COVER ILLUSTRATIONS	Susan Cyr
TEXT DESIGNER	Judy Allan

This book was set in Palatino by the author using LaTeX and printed and bound by R.R. Donnelley/Jefferson City. The cover was printed by R. R. Donnelley/Jefferson City

This book is printed on acid free paper. ∞

Copyright © 2009 John Wiley & Sons, Inc. All rights reserved.

No part of this publication may be reproduced, stored in a retrieval system or transmitted in any form or by any means, electronic, mechanical, photocopying, recording, scanning or otherwise, except as permitted under Sections 107 or 108 of the 1976 United States Copyright Act, without either the prior written permission of the Publisher, or authorization through payment of the appropriate per-copy fee to the Copyright Clearance Center, Inc. 222 Rosewood Drive, Danvers, MA 01923, (978)750-8400, fax (978)750-4470. Requests to the Publisher for permission should be addressed to the Permissions Department, John Wiley & Sons, Inc., 111 River Street, Hoboken, NJ 07030 (201)748-6011, fax (201)748-6008, E-Mail: PERMREQ@WILEY.COM.
To order books or for customer service call 1-800-CALL WILEY (225-5945).

ISBN: 978-0-470-12872-5

Printed in the United States of America

10 9 8 7 6 5 4 3

Contents

PART ONE ■ OVERVIEW

Chapter 1 Introduction

- 1.1 What Operating Systems Do 3
- 1.2 Computer-System Organization 6
- 1.3 Computer-System Architecture 12
- 1.4 Operating-System Structure 18
- 1.5 Operating-System Operations 20
- 1.6 Process Management 23
- 1.7 Memory Management 24
- 1.8 Storage Management 25
- 1.9 Protection and Security 29
- 1.10 Distributed Systems 30
- 1.11 Special-Purpose Systems 32
- 1.12 Computing Environments 34
- 1.13 Open-Source Operating Systems 37
- 1.14 Summary 40
- Exercises 42
- Bibliographical Notes 46

Chapter 2 System Structures

- 2.1 Operating-System Services 49
- 2.2 User Operating-System Interface 52
- 2.3 System Calls 55
- 2.4 Types of System Calls 58
- 2.5 System Programs 66
- 2.6 Operating-System Design and Implementation 68
- 2.7 Operating-System Structure 70
- 2.8 Virtual Machines 76
- 2.9 Operating-System Debugging 84
- 2.10 Operating-System Generation 88
- 2.11 System Boot 89
- 2.12 Summary 90
- Exercises 91
- Bibliographical Notes 97

PART TWO ■ PROCESS MANAGEMENT

Chapter 3 Process Concept

- 3.1 Process Concept 101
- 3.2 Process Scheduling 105
- 3.3 Operations on Processes 110
- 3.4 Interprocess Communication 116
- 3.5 Examples of IPC Systems 123
- 3.6 Communication in Client-Server Systems 128
- 3.7 Summary 140
- Exercises 141
- Bibliographical Notes 152

Chapter 4 Multithreaded Programming

- | | | | |
|---------------------------|-----|-------------------------------|-----|
| 4.1 Overview | 153 | 4.5 Operating-System Examples | 171 |
| 4.2 Multithreading Models | 157 | 4.6 Summary | 174 |
| 4.3 Thread Libraries | 159 | Exercises | 174 |
| 4.4 Threading Issues | 165 | Bibliographical Notes | 181 |

Chapter 5 Process Scheduling

- | | | | |
|-----------------------------------|-----|-------------------------------|-----|
| 5.1 Basic Concepts | 183 | 5.6 Operating System Examples | 206 |
| 5.2 Scheduling Criteria | 187 | 5.7 Algorithm Evaluation | 213 |
| 5.3 Scheduling Algorithms | 188 | 5.8 Summary | 217 |
| 5.4 Thread Scheduling | 199 | Exercises | 218 |
| 5.5 Multiple-Processor Scheduling | 200 | Bibliographical Notes | 222 |

PART THREE ■ PROCESS COORDINATION**Chapter 6 Synchronization**

- | | | | |
|-----------------------------------------|-----|------------------------------|-----|
| 6.1 Background | 225 | 6.7 Monitors | 244 |
| 6.2 The Critical-Section Problem | 227 | 6.8 Synchronization Examples | 252 |
| 6.3 Peterson's Solution | 229 | 6.9 Atomic Transactions | 257 |
| 6.4 Synchronization Hardware | 231 | 6.10 Summary | 267 |
| 6.5 Semaphores | 234 | Exercises | 267 |
| 6.6 Classic Problems of Synchronization | 239 | Bibliographical Notes | 280 |

Chapter 7 Deadlocks

- | | | | |
|------------------------------------|-----|----------------------------|-----|
| 7.1 System Model | 283 | 7.6 Deadlock Detection | 301 |
| 7.2 Deadlock Characterization | 285 | 7.7 Recovery from Deadlock | 304 |
| 7.3 Methods for Handling Deadlocks | 290 | 7.8 Summary | 306 |
| 7.4 Deadlock Prevention | 291 | Exercises | 307 |
| 7.5 Deadlock Avoidance | 294 | Bibliographical Notes | 310 |

PART FOUR ■ MEMORY MANAGEMENT**Chapter 8 Memory-Management Strategies**

- | | | | |
|----------------------------------|-----|--------------------------------|-----|
| 8.1 Background | 315 | 8.6 Segmentation | 342 |
| 8.2 Swapping | 322 | 8.7 Example: The Intel Pentium | 345 |
| 8.3 Contiguous Memory Allocation | 324 | 8.8 Summary | 349 |
| 8.4 Paging | 328 | Exercises | 350 |
| 8.5 Structure of the Page Table | 337 | Bibliographical Notes | 354 |

Chapter 9 Virtual-Memory Management

- | | | | |
|--------------------------|-----|--------------------------------|-----|
| 9.1 Background | 357 | 9.8 Allocating Kernel Memory | 396 |
| 9.2 Demand Paging | 361 | 9.9 Other Considerations | 399 |
| 9.3 Copy-on-Write | 367 | 9.10 Operating-System Examples | 405 |
| 9.4 Page Replacement | 369 | 9.11 Summary | 407 |
| 9.5 Allocation of Frames | 382 | Exercises | 409 |
| 9.6 Thrashing | 386 | Bibliographical Notes | 416 |
| 9.7 Memory-Mapped Files | 390 | | |

PART FIVE ■ STORAGE MANAGEMENT**Chapter 10 File System**

- | | | | |
|-----------------------------------|-----|-----------------------|-----|
| 10.1 File Concept | 421 | 10.6 Protection | 451 |
| 10.2 Access Methods | 430 | 10.7 Summary | 456 |
| 10.3 Directory and Disk Structure | 433 | Exercises | 457 |
| 10.4 File-System Mounting | 444 | Bibliographical Notes | 458 |
| 10.5 File Sharing | 446 | | |

Chapter 11 Implementing File Systems

- | | | | |
|---------------------------------|-----|------------------------------------|-----|
| 11.1 File-System Structure | 461 | 11.7 Recovery | 486 |
| 11.2 File-System Implementation | 464 | 11.8 NFS | 490 |
| 11.3 Directory Implementation | 470 | 11.9 Example: The WAFL File System | 496 |
| 11.4 Allocation Methods | 471 | 11.10 Summary | 498 |
| 11.5 Free-Space Management | 479 | Exercises | 499 |
| 11.6 Efficiency and Performance | 482 | Bibliographical Notes | 502 |

Chapter 12 Secondary-Storage Structure

- | | | | |
|-----------------------------------------|-----|------------------------------------|-----|
| 12.1 Overview of Mass-Storage Structure | 505 | 12.7 RAID Structure | 522 |
| 12.2 Disk Structure | 508 | 12.8 Stable-Storage Implementation | 533 |
| 12.3 Disk Attachment | 509 | 12.9 Tertiary-Storage Structure | 534 |
| 12.4 Disk Scheduling | 510 | 12.10 Summary | 543 |
| 12.5 Disk Management | 516 | Exercises | 545 |
| 12.6 Swap-Space Management | 520 | Bibliographical Notes | 552 |

Chapter 13 I/O Systems

- | | | | |
|-------------------------------------------------------|-----|-----------------------|-----|
| 13.1 Overview | 555 | 13.6 STREAMS | 580 |
| 13.2 I/O Hardware | 556 | 13.7 Performance | 582 |
| 13.3 Application I/O Interface | 565 | 13.8 Summary | 585 |
| 13.4 Kernel I/O Subsystem | 571 | Exercises | 586 |
| 13.5 Transforming I/O Requests to Hardware Operations | 578 | Bibliographical Notes | 588 |

PART SIX ■ PROTECTION AND SECURITY**Chapter 14 System Protection**

- | | | | |
|--------------------------------------|-----|----------------------------------|-----|
| 14.1 Goals of Protection | 591 | 14.7 Revocation of Access Rights | 606 |
| 14.2 Principles of Protection | 592 | 14.8 Capability-Based Systems | 607 |
| 14.3 Domain of Protection | 593 | 14.9 Language-Based Protection | 610 |
| 14.4 Access Matrix | 598 | 14.10 Summary | 615 |
| 14.5 Implementation of Access Matrix | 602 | Exercises | 616 |
| 14.6 Access Control | 605 | Bibliographical Notes | 618 |

Chapter 15 System Security

- | | | | |
|-----------------------------------------------------|-----|-------------------------------------------|-----|
| 15.1 The Security Problem | 621 | 15.8 Computer-Security
Classifications | 662 |
| 15.2 Program Threats | 625 | 15.9 An Example: Windows XP | 664 |
| 15.3 System and Network Threats | 633 | 15.10 Summary | 665 |
| 15.4 Cryptography as a Security Tool | 638 | Exercises | 666 |
| 15.5 User Authentication | 649 | Bibliographical Notes | 667 |
| 15.6 Implementing Security Defenses | 654 | | |
| 15.7 Firewalling to Protect Systems and
Networks | 661 | | |

PART SEVEN ■ DISTRIBUTED SYSTEMS**Chapter 16 Distributed Operating Systems**

- | | | | |
|---------------------------------------------------|-----|-----------------------------|-----|
| 16.1 Motivation | 673 | 16.7 Robustness | 694 |
| 16.2 Types of Network-
based Operating Systems | 675 | 16.8 Design Issues | 697 |
| 16.3 Network Structure | 679 | 16.9 An Example: Networking | 699 |
| 16.4 Network Topology | 683 | 16.10 Summary | 701 |
| 16.5 Communication Structure | 684 | Exercises | 701 |
| 16.6 Communication Protocols | 690 | Bibliographical Notes | 703 |

Chapter 17 Distributed File Systems

- | | | | |
|----------------------------------------|-----|-----------------------|-----|
| 17.1 Background | 705 | 17.6 An Example: AFS | 718 |
| 17.2 Naming and Transparency | 707 | 17.7 Summary | 723 |
| 17.3 Remote File Access | 710 | Exercises | 724 |
| 17.4 Stateful versus Stateless Service | 715 | Bibliographical Notes | 725 |
| 17.5 File Replication | 716 | | |

Chapter 18 Distributed Synchronization

- | | | | |
|--------------------------|-----|--------------------------|-----|
| 18.1 Event Ordering | 727 | 18.6 Election Algorithms | 747 |
| 18.2 Mutual Exclusion | 730 | 18.7 Reaching Agreement | 750 |
| 18.3 Atomicity | 733 | 18.8 Summary | 752 |
| 18.4 Concurrency Control | 736 | Exercises | 753 |
| 18.5 Deadlock Handling | 740 | Bibliographical Notes | 754 |

PART EIGHT ■ SPECIAL PURPOSE SYSTEMS

Chapter 19 Real-Time Systems

- | | | | |
|-----------------------------------------------|-----|-------------------------------|-----|
| 19.1 Overview | 759 | 19.5 Real-Time CPU Scheduling | 768 |
| 19.2 System Characteristics | 760 | 19.6 An Example: VxWorks 5.x | 774 |
| 19.3 Features of Real-Time Kernels | 762 | 19.7 Summary | 776 |
| 19.4 Implementing Real-Time Operating Systems | 764 | Exercises | 777 |
| | | Bibliographical Notes | 777 |

Chapter 20 Multimedia Systems

- | | | | |
|-----------------------------------------|-----|----------------------------|-----|
| 20.1 What Is Multimedia? | 779 | 20.6 Network Management | 789 |
| 20.2 Compression | 782 | 20.7 An Example: CineBlitz | 792 |
| 20.3 Requirements of Multimedia Kernels | 784 | 20.8 Summary | 795 |
| 20.4 CPU Scheduling | 786 | Exercises | 795 |
| 20.5 Disk Scheduling | 787 | Bibliographical Notes | 797 |

PART NINE ■ CASE STUDIES

Chapter 21 The Linux System

- | | | | |
|-------------------------|-----|---------------------------------|-----|
| 21.1 Linux History | 801 | 21.8 Input and Output | 834 |
| 21.2 Design Principles | 806 | 21.9 Interprocess Communication | 837 |
| 21.3 Kernel Modules | 809 | 21.10 Network Structure | 838 |
| 21.4 Process Management | 812 | 21.11 Security | 840 |
| 21.5 Scheduling | 815 | 21.12 Summary | 843 |
| 21.6 Memory Management | 820 | Exercises | 844 |
| 21.7 File Systems | 828 | Bibliographical Notes | 845 |

Chapter 22 Windows XP

- | | | | |
|-------------------------------|-----|---------------------------|-----|
| 22.1 History | 847 | 22.6 Networking | 886 |
| 22.2 Design Principles | 849 | 22.7 Programmer Interface | 892 |
| 22.3 System Components | 851 | 22.8 Summary | 900 |
| 22.4 Environmental Subsystems | 874 | Exercises | 900 |
| 22.5 File System | 878 | Bibliographical Notes | 901 |

Chapter 23 Influential Operating Systems

- | | | | |
|------------------------|-----|----------------------------------------------|-----|
| 23.1 Feature Migration | 903 | 23.9 IBM OS/360 | 915 |
| 23.2 Early Systems | 904 | 23.10 TOPS-20 | 917 |
| 23.3 Atlas | 911 | 23.11 CP/M and MS/DOS | 917 |
| 23.4 XDS-940 | 912 | 23.12 Macintosh Operating System and Windows | 918 |
| 23.5 THE | 913 | 23.13 Mach | 919 |
| 23.6 RC 4000 | 913 | 23.14 Other Systems | 920 |
| 23.7 CTSS | 914 | Exercises | 921 |
| 23.8 MULTICS | 915 | | |

Chapter A BSD UNIX

- A.1 UNIX History 1
- A.2 Design Principles 6
- A.3 Programmer Interface 8
- A.4 User Interface 15
- A.5 Process Management 18
- A.6 Memory Management 22
- A.7 File System 25
- A.8 I/O System 32
- A.9 Interprocess Communication 35
- A.10 Summary 40
 - Exercises 41
 - Bibliographical Notes 42

Appendix B The Mach System

- B.1 History of the Mach System 1
- B.2 Design Principles 3
- B.3 System Components 4
- B.4 Process Management 7
- B.5 Interprocess Communication 13
- B.6 Memory Management 18
- B.7 Programmer Interface 23
- B.8 Summary 24
 - Exercises 25
 - Bibliographical Notes 26
 - Credits 27

Appendix C Windows 2000

- C.1 History 1
- C.2 Design Principles 2
- C.3 System Components 3
- C.4 Environmental Subsystems 19
- C.5 File System 22
- C.6 Networking 28
- C.7 Programmer Interface 33
- C.8 Summary 40
 - Exercises 40
 - Bibliographical Notes 41

Bibliography 923**Credits 941****Index 943**

System Protection

The processes in an operating system must be protected from one another's activities. To provide such protection, we can use various mechanisms to ensure that only processes that have gained proper authorization from the operating system can operate on the files, memory segments, CPU, and other resources of a system.

Protection refers to a mechanism for controlling the access of programs, processes, or users to the resources defined by a computer system. This mechanism must provide a means for specifying the controls to be imposed, together with a means of enforcement. We distinguish between protection and security, which is a measure of confidence that the integrity of a system and its data will be preserved. In this chapter, we focus on protection. Security assurance is a much broader topic, and we address it in Chapter 15.

CHAPTER OBJECTIVES

- To discuss the goals and principles of protection in a modern computer system.
- To explain how protection domains, combined with an access matrix, are used to specify the resources a process may access.
- To examine capability- and language-based protection systems.

14.1 Goals of Protection

As computer systems have become more sophisticated and pervasive in their applications, the need to protect their integrity has also grown. Protection was originally conceived as an adjunct to multiprogramming operating systems, so that untrustworthy users might safely share a common logical name space, such as a directory of files, or share a common physical name space, such as memory. Modern protection concepts have evolved to increase the reliability of any complex system that makes use of shared resources.

We need to provide protection for several reasons. The most obvious is the need to prevent the mischievous, intentional violation of an access restriction