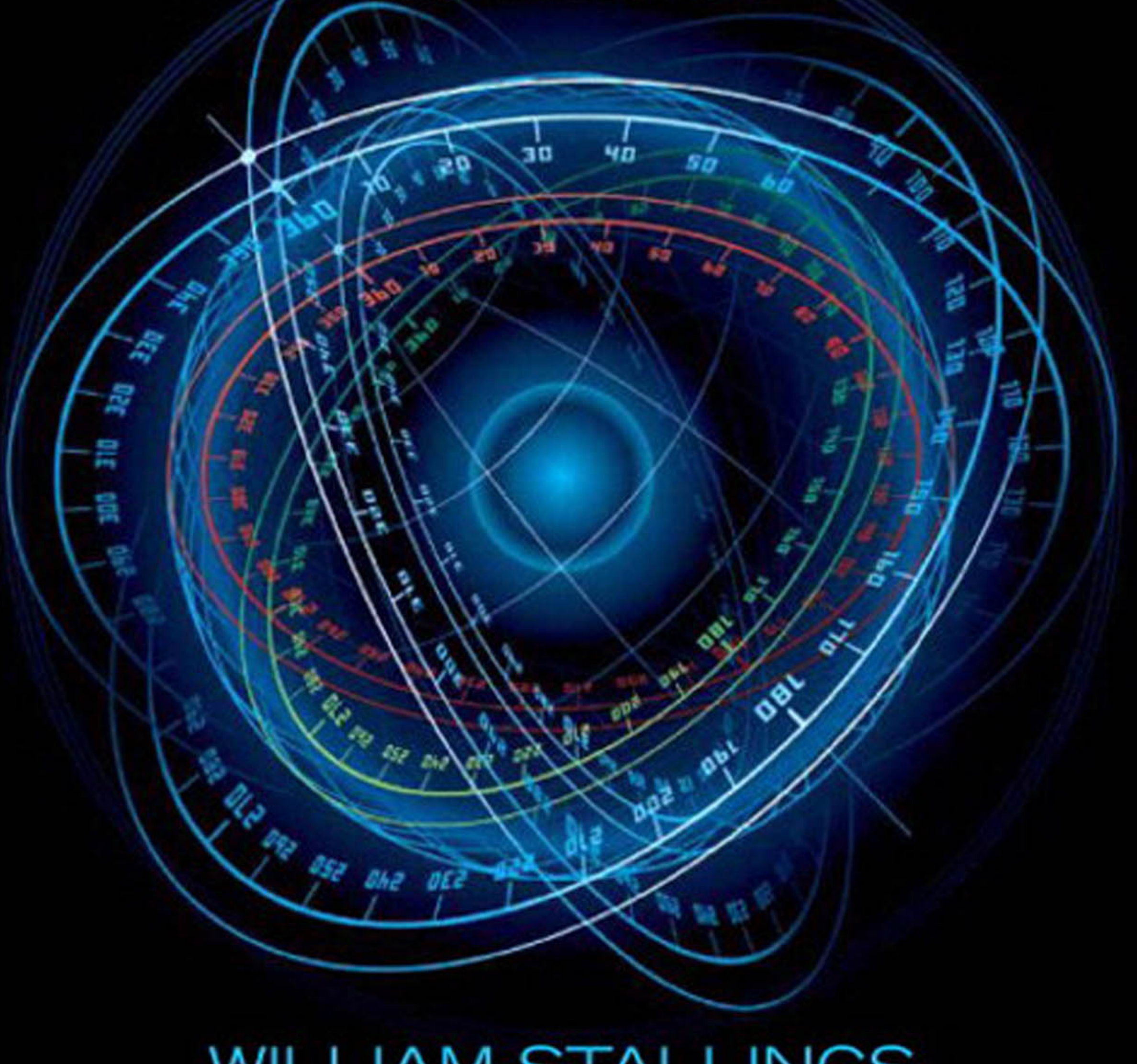
# OPERATING SYSTEMS

## Internals and Design Principles

EIGHTH EDITION



WILLIAM STALLINGS

# ONLINE ACCESS

Thank you for purchasing a new copy of ***Operating Systems: Internals and Design Principles,*** **Eighth Edition**. Your textbook includes six months of prepaid access to the book's Premium Content. This prepaid subscription provides you with full access to the following student support areas:

- Animations and Videonotes
- Supplemental Chapters
- Supplemental Appendices
- Supplemental homework problems with solutions

*Use a coin to scratch off the coating and reveal your student access code.*
*Do not use a knife or other sharp object as it may damage the code.*

To access the ***Operating Systems: Internals and Design Principles,*** **Eighth Edition,** Premium Content for the first time, you will need to register online using a computer with an Internet connection and a web browser. The process takes just a couple of minutes and only needs to be completed once.

1. Go to **http://www.pearsonhighered.com/stallings/**
2. Click on the **Premium Content**.
3. Click on the **Register** button.
4. On the registration page, enter your student access code* found beneath the scratch-off panel. Do not type the dashes. You can use lower- or uppercase.
5. Follow the on-screen instructions. If you need help at any time during the online registration process, simply click the **Need Help?** icon.
6. Once your personal Login Name and Password are confirmed, you can begin using the *Operating Systems: Internals and Design Principles* Premium Content Website!

**To log in after you have registered:**

You only need to register for this Premium Content once. After that, you can log in any time at **http://www.pearsonhighered.com/stallings/** by providing your Login Name and Password when prompted.

*Important: The access code can only be used once. This subscription is valid for six months upon activation and is not transferable. If this access code has already been revealed, it may no longer be valid.

If this is the case, you can purchase a subscription by going to **http://www.pearsonhighered.com/stallings/** and following the on-screen instructions.

# Operating Systems

## *Internals and Design Principles*

*This page intentionally left blank*

# Operating Systems

## *Internals and Design Principles*

## Eighth Edition

**William Stallings**

PEARSON

Boston Columbus Indianapolis New York San Francisco Upper Saddle River
Amsterdam CapeTown Dubai London Madrid Milan Munich Paris Montréal Toronto
Delhi Mexico City São Paulo Sydney Hong Kong Seoul Singapore Taipei Tokyo

Editorial Director, ECS: Marcia Horton
Executive Editor: Tracy Johnson (Dunkelberger)
Editorial Assistant: Jenah Blitz-Stoehr
Director of Marketing: Christy Lesko
Marketing Manager: Yez Alayan
Marketing Assistant: Jon Bryant
Director of Program Management: Erin Gregg
Program Management-Team Lead: Scott Disanno
Program Manager: Carole Snyder
Project Management-Team Lead: Laura Burgess
Project Manager: Robert Engelhardt
Procurement Specialist: Linda Sager
Cover Designer: Bruce Kenselaar
Permissions Supervisor: Michael Joyce
Permissions Administrator: William Opaluch
Director, Image Asset Services: Annie Atherton
Manager, Visual Research: Karen Sanatar
Cover Art: © Fixer00/Shutterstock
Media Project Manager: Renata Butera
Full-Service Project Management: Integra Software Services Pvt. Ltd. /Allan Rayer

Credits and acknowledgments borrowed from other sources and reproduced, with permission, in this textbook appear on the appropriate page within text.

**Library of Congress Cataloging-in-Publication Data on File**

10 9 8 7 6 5 4 3 2 1

PEARSON

ISBN 10: 0-13-380591-3
ISBN 13: 978-0-13-380591-8

*For Tricia*

*This page intentionally left blank*

# Contents

**APPENDICES**

*This page intentionally left blank*

# Online Chapters and Appendices[1]

[1] Online chapters, appendices, and other documents are Premium Content, available via the access card at the front of this book.

VideoNote

# VideoNotes

**Locations of VideoNotes**

**http://www.pearsonhighered.com/stallings**

# PREFACE

## WHAT'S NEW IN THE EIGHTH EDITION

Since the seventh edition of this book was published, the field has seen continued innovations and improvements. In this new edition, I try to capture these changes while maintaining a broad and comprehensive coverage of the entire field. To begin the process of revision, the seventh edition of this book was extensively reviewed by a number of professors who teach the subject and by professionals working in the field. The result is that, in many places, the narrative has been clarified and tightened, and illustrations have been improved.

Beyond these refinements to improve pedagogy and user friendliness, the technical content of the book has been updated throughout, to reflect the ongoing changes in this exciting field, and the instructor and student support has been expanded. The most noteworthy changes are as follows:

- **Windows 8:** Windows 8 is Microsoft's latest OS offering for PCs, workstations, and servers, which includes a number of changes to the internal architecture. The new edition provides details on Windows 8 internals in all of the key technology areas covered in this book, including process/thread management, scheduling, memory management, security, file systems, and I/O.
- **Android operating system:** Android is the fastest growing mobile platform. The real-world constraints and operating environment of mobile devices are quite different from traditional desktop or server computers. It is important for students to learn this new environment.
- **Embedded Linux:** The use of a minimal version of Linux for embedded systems has grown in popularity. This new edition provides an overview of the key elements of the embedded Linux approach.
- **Virtual machines:** Server virtualization and other forms of virtual machines are becoming increasingly widespread. A new chapter deals with the operating system design issues for virtual machines.
- **Multicore design issues:** The dominant computer architecture is now multicore. This raises new OS design issues that are addressed in this new edition.
- **I/O standards:** The book has been updated to reflect the latest developments, including Thunderbolt.
- **Storage hardware:** The discussion of storage hardware has been updated and now includes discussion of solid-state drives.
- **Fault tolerance:** The ACM/IEEE Computer Science Curricula 2013 lists fault tolerance as one of the core topics for an OS course. A new section provides an overview of fault tolerance.

## OBJECTIVES

This book is about the concepts, structure, and mechanisms of operating systems. Its purpose is to present, as clearly and completely as possible, the nature and characteristics of modern-day operating systems.

This task is challenging for several reasons. First, there is a tremendous range and variety of computer systems for which operating systems are designed. These include embedded systems, smart phones, single-user workstations and personal computers, medium-sized shared systems, large mainframe and supercomputers, and specialized machines such as real-time systems. The variety is not just in the capacity and speed of machines, but in applications and system support requirements. Second, the rapid pace of change that has always characterized computer systems continues with no letup. A number of key areas in operating system design are of recent origin, and research into these and other new areas continues.

In spite of this variety and pace of change, certain fundamental concepts apply consistently throughout. To be sure, the application of these concepts depends on the current state of technology and the particular application requirements. The intent of this book is to provide a thorough discussion of the fundamentals of operating system design and to relate these to contemporary design issues and to current directions in the development of operating systems.

## EXAMPLE SYSTEMS

This text is intended to acquaint the reader with the design principles and implementation issues of contemporary operating systems. Accordingly, a purely conceptual or theoretical treatment would be inadequate. To illustrate the concepts and to tie them to real-world design choices that must be made, four operating systems have been chosen as running examples:

- **Windows 8:** A multitasking operating system for personal computers, workstations, servers, and mobile devices. This operating system incorporates many of the latest developments in operating system technology. In addition, Windows is one of the first important commercial operating systems to rely heavily on object-oriented design principles. This book covers the technology used in the most recent versions of Windows, known as Windows 8.
- **Android:** Android tailored for embedded devices, especially mobile phones. Focusing on the unique requirements of the embedded environment, the book provides details of Android internals.
- **UNIX:** A multiuser operating system, originally intended for minicomputers, but implemented on a wide range of machines from powerful microcomputers to supercomputers. Several flavors of UNIX are included as examples. FreeBSD is a widely used system that incorporates many state-of-the-art features. Solaris is a widely used commercial version of UNIX.
- **Linux:** An open-source version of UNIX that is now widely used.

These systems were chosen because of their relevance and representativeness. The discussion of the example systems is distributed throughout the text rather than assembled as a single chapter or appendix. Thus, during the discussion of concurrency, the concurrency mechanisms of each example system are described, and the motivation for the individual design choices is discussed. With this approach, the design concepts discussed in a given chapter are immediately reinforced with real-world examples. For convenience all of the material for each of the example systems is also available as an online document.

## SUPPORT OF ACM/IEEE COMPUTER SCIENCE CURRICULA 2013

The book is intended for both an academic and a professional audience. As a textbook, it is intended as a one-semester or two-semester undergraduate course in operating systems for computer science, computer engineering, and electrical engineering majors. This edition is designed to support the recommendations of the current (February 2013) draft version of the ACM/IEEE Computer Science Curricula 2013 (CS2013). The CS2013 curriculum recommendation includes Operating Systems (OSs) as one of the Knowledge Areas in the Computer Science Body of Knowledge. CS2013 divides all course work into three categories: Core-Tier 1 (all topics should be included in the curriculum), Core-Tier-2 (all or almost all topics should be included), and Elective (desirable to provide breadth and depth). In the OS area, CS2013 includes two Tier 1 topics, four Tier 2 topics, and 6 Elective topics, each of which has a number of subtopics. This text covers all of the topics and subtopics listed by CS2013 in these three categories.

Table P.1 shows the support for the OS Knowledge Area provided in this textbook. A detailed list of subtopics for each topic is available as the file CS2013-OS.pdf at box.com/OS8e.

## PLAN OF THE TEXT

The book is divided into five parts (see Chapter 0 for an overview):

- Background
- Processes
- Memory
- Scheduling
- Advanced topics (embedded OSs, virtual machines, OS security, and distributed systems)

The book includes a number of pedagogic features, including the use of animations and video notes and numerous figures and tables to clarify the discussion. Each chapter includes a list of key words, review questions, homework problems, and suggestions for further reading. The book also includes an extensive glossary, a list of frequently used acronyms, and a bibliography. In addition, a test bank is available to instructors.

**Table P.1** Coverage of CS2013 Operating Systems (OSs) Knowledge Area

| Topic | Coverage in Book |
|---|---|
| **Overview of Operating Systems (Tier 1)** | Chapter 2: Operating System Overview |
| **Operating System Principles (Tier 1)** | Chapter 1: Computer System Overview<br>Chapter 2: Operating System Overview |
| **Concurrency (Tier 2)** | Chapter 5: Mutual Exclusion and Synchronization<br>Chapter 6: Deadlock and Starvation<br>Appendix A: Topics in Concurrency<br>Chapter 18: Distributed Process Management |
| **Scheduling and Dispatch (Tier 2)** | Chapter 9: Uniprocessor Scheduling<br>Chapter 10: Multiprocessor and Real-Time Scheduling |
| **Memory Management (Tier 2)** | Chapter 7: Memory Management<br>Chapter 8: Virtual Memory |
| **Security and Protection (Tier 2)** | Chapter 15: Operating System Security |
| **Virtual Machines (Elective)** | Chapter 14: Virtual Machines |
| **Device Management (Elective)** | Chapter 11: I/O Management and Disk Scheduling |
| **File System (Elective)** | Chapter 12: File Management |
| **Real Time and Embedded Systems (Elective)** | Chapter 10: Multiprocessor and Real-Time Scheduling<br>Chapter 13: Embedded Operating Systems<br>Material on Android throughout the text |
| **Fault Tolerance (Elective)** | Section 2.5: Fault Tolerance |
| **System Performance Evaluation (Elective)** | Performance issues related to memory management, scheduling, and other areas addressed throughout the text |

## INSTRUCTOR SUPPORT MATERIALS

The major goal of this text is to make it as effective a teaching tool for this fundamental yet evolving subject as possible. This goal is reflected both in the structure of the book and in the supporting material. The text is accompanied by the following supplementary material to aid the instructor:

- **Solutions manual:** Solutions to end-of-chapter Review Questions and Problems.
- **Projects manual:** Suggested project assignments for all of the project categories listed in this Preface.
- **PowerPoint slides:** A set of slides covering all chapters, suitable for use in lecturing.
- **PDF files:** Reproductions of all figures and tables from the book.
- **Test bank:** A chapter-by-chapter set of questions with a separate file of answers.

- **VideoNotes on concurrency**: Professors perennially cite concurrency as perhaps the most difficult concept in the field of operating systems for students to grasp. The edition is accompanied by a number of VideoNotes lectures discussing the various concurrency algorithms defined in the book.
- **Sample syllabuses:** The text contains more material than can be conveniently covered in one semester. Accordingly, instructors are provided with several sample syllabuses that guide the use of the text within limited time. These samples are based on real-world experience by professors with the seventh edition.

All of these support materials are available at the **Instructor Resource Center (IRC)** for this textbook, which can be reached through the publisher's Web site www.pearsonhighered.com/stallings or by clicking on the link labeled *Pearson Resources for Instructors* at this book's Companion Web site at WilliamStallings.com/OperatingSystems. To gain access to the IRC, please contact your local Pearson sales representative via pearsonhighered.com/educator/replocator/requestSalesRep.page or call Pearson Faculty Services at 1-800-526-0485.

The **Companion Web site**, at WilliamStallings.com/OperatingSystems (click on *Instructor Resources* link), includes the following:

- Links to Web sites for other courses being taught using this book.
- Sign-up information for an Internet mailing list for instructors using this book to exchange information, suggestions, and questions with each other and with the author.

## PROJECTS AND OTHER STUDENT EXERCISES

For many instructors, an important component of an OS course is a project or set of projects by which the student gets hands-on experience to reinforce concepts from the text. This book provides an unparalleled degree of support for including a projects component in the course. In the online portion of the text, two major programming projects are defined. In addition, the instructor's support materials available through Prentice Hall not only includes guidance on how to assign and structure the various projects but also includes a set of user's manuals for various project types plus specific assignments, all written especially for this book. Instructors can assign work in the following areas:

- **OS/161 projects:** Described below.
- **Simulation projects:** Described below.
- **Programming projects:** Described below.
- **Research projects:** A series of research assignments that instruct the student to research a particular topic on the Internet and write a report.
- **Reading/report assignments:** A list of papers that can be assigned for reading and writing a report, plus suggested assignment wording.

- **Writing assignments:** A list of writing assignments to facilitate learning the material.
- **Discussion topics:** These topics can be used in a classroom, chat room, or message board environment to explore certain areas in greater depth and to foster student collaboration.

In addition, information is provided on a software package known as BACI that serves as a framework for studying concurrency mechanisms.

This diverse set of projects and other student exercises enables the instructor to use the book as one component in a rich and varied learning experience and to tailor a course plan to meet the specific needs of the instructor and students. See Appendix B in this book for details.

## OS/161

This edition provides support for an active learning component based on OS/161. OS/161 is an educational operating system that is becoming increasingly recognized as the preferred teaching platform for OS internals. It aims to strike a balance between giving students experience in working on a real operating system and potentially overwhelming students with the complexity that exists in a fully-fledged operating system, such as Linux. Compared to most deployed operating systems, OS/161 is quite small (approximately 20,000 lines of code and comments), and therefore it is much easier to develop an understanding of the entire code base.

The IRC includes:

1. A packaged set of html files that the instructor can upload to a course server for student access.
2. A getting-started manual to be handed out to students to help them begin using OS/161.
3. A set of exercises using OS/161, to be handed out to students.
4. Model solutions to each exercise for the instructor's use.
5. All of this will be cross-referenced with appropriate sections in the book, so that the student can read the textbook material and then do the corresponding OS/161 project.

## SIMULATIONS

The IRC provides support for assigning projects based on a set of seven **simulations** that cover key areas of OS design. The student can use a set of simulation packages to analyze OS design features. The simulators are all written in Java and can be run either locally as a Java application or online through a browser. The IRC includes specific assignments to give to students, telling them specifically what they are to do and what results are expected.

## ANIMATIONS

This edition also incorporates animations. Animations provide a powerful tool for understanding the complex mechanisms of a modern OS. A total of 53 animations are used to illustrate key functions and algorithms in OS design. The animations are used for Chapters 3, 5, 6, 7, 8, 9, and 11. The animations are available at the Premium Web site for this book.

## PROGRAMMING PROJECTS

This edition provides support for programming projects. Two major programming projects, one to build a shell, or command line interpreter, and one to build a process dispatcher are described in the online portion of this textbook. The IRC provides further information and step-by-step exercises for developing the programs.

As an alternative, the instructor can assign a more extensive series of projects that cover many of the principles in the book. The student is provided with detailed instructions for doing each of the projects. In addition, there is a set of homework problems, which involve questions related to each project for the student to answer.

Finally, the project manual provided at the IRC includes a series of programming projects that cover a broad range of topics and that can be implemented in any suitable language on any platform.

## ONLINE DOCUMENTS, ANIMATIONS, AND VIDEONOTES FOR STUDENTS

For this new edition, a substantial amount of original supporting material for students has been made available online, at two Web locations. The **Companion Web site**, at WilliamStallings.com/OperatingSystems (click on *Student Resources* link), includes a list of relevant links organized by chapter and an errata sheet for the book.

Purchasing this textbook new also grants the reader six months of access to the **Premium Content site**, which includes the following materials:

- **Online chapters:** To limit the size and cost of the book, two chapters of the book, covering security, are provided in PDF format. The chapters are listed in this book's table of contents.
- **Online appendices:** There are numerous interesting topics that support material found in the text but whose inclusion is not warranted in the printed text. A total of 18 online appendices cover these topics for the interested student. The appendices are listed in this book's table of contents.

- **Homework problems and solutions:** To aid the student in understanding the material, a separate set of homework problems with solutions is available.
- **Animations:** Animations provide a powerful tool for understanding the complex mechanisms of a modern OS. A total of 53 animations are used to illustrate key functions and algorithms in OS design. The animations are used for Chapters 3, 5, 6, 7, 8, 9, and 11.
- **Videonotes:** VideoNotes are step-by-step video tutorials specifically designed to enhance the programming concepts presented in this textbook. The book is accompanied by a number of VideoNotes lectures discussing the various concurrency algorithms defined in the book.

To access the Premium Content site, click on the *Premium Content* link at the Companion Web site or at pearsonhighered.com/stallings and enter the student access code found on the card in the front of the book.

## ACKNOWLEDGMENTS

Through its multiple editions, this book has benefited from review by hundreds of instructors and professionals, who gave generously of their time and expertise. Here I acknowledge those whose help contributed to this latest edition.

The following instructors reviewed all or a large part of the manuscript: Bernard Ku (Austin Community College, Northridge), Hemant Pendharker (Worcester State University), Nancy Birkenheuer (Regis University, Denver), Greg Ozbirn (University of Texas, Dallas).

Thanks also to the people who provided detailed reviews of the example systems. Reviews of the Android material were provided by Kristopher Micinski, Ron Munitz, Atte Peltomaki, Durgadoss Ramanathan, Manish Shakya, Samuel Simon, Wei Wang, and Chen Yang. The Linux reviewers were Tigran Aivazian, Kaiwan Billimoria, Peter Huewe, Manmohan Manoharan, Rami Rosen, Neha Naik, and Hualing Yu. The Windows material was reviewed by Francisco Cotrina, Sam Haidar, Christopher Kuleci, Benny Olsson, and Dave Probert. Nick Garnett of eCosCentric, for the review of the material on eCos, and Philip Levis, one of the developers of TinyOS, for the review of the material on TinyOS.

Andrew Peterson of the University of Toronto prepared the OS/161 supplements for the IRC. James Craig Burley authored and recorded the VideoNotes.

Adam Critchley (University of Texas at San Antonio) developed the simulation exercises. Matt Sparks (University of Illinois at Urbana-Champaign) adapted a set of programming problems for use with this textbook.

Lawrie Brown of the Australian Defence Force Academy produced the material on buffer overflow attacks. Ching-Kuang Shene (Michigan Tech University) provided the examples used in the section on race conditions and reviewed the section. Tracy Camp and Keith Hellman, both at the Colorado School of Mines, developed a new set of homework problems. In addition, Fernando Ariel Gont

contributed a number of homework problems; he also provided detailed reviews of all of the chapters.

I would also like to thank Bill Bynum (College of William and Mary) and Tracy Camp (Colorado School of Mines) for contributing Appendix O; Steve Taylor (Worcester Polytechnic Institute) for contributing the programming projects and reading/report assignments in the instructor's manual; and Professor Tan N. Nguyen (George Mason University) for contributing the research projects in the instruction manual. Ian G. Graham (Griffith University) contributed the two programming projects in the textbook. Oskars Rieksts (Kutztown University) generously allowed me to make use of his lecture notes, quizzes, and projects.

Finally, I thank the many people responsible for the publication of this book, all of whom did their usual excellent job. This includes the staff at Pearson, particularly my editor Tracy Johnson, her assistant Jenah Blitz-Stoehr, program manager Carole Snyder, and project manager Bob Engelhardt. I also thank Shiny Rajesh and the production staff at Integra for another excellent and rapid job. Thanks also to the marketing and sales staffs at Pearson, without whose efforts this book would not be in front of you.

*This page intentionally left blank*

# About the Author

**Dr. William Stallings** has authored 17 titles, and counting revised editions, over 40 books on computer security, computer networking, and computer architecture. His writings have appeared in numerous publications, including the *Proceedings of the IEEE, ACM Computing Reviews* and *Cryptologia*.

He has 12 times received the award for the best Computer Science textbook of the year from the Text and Academic Authors Association.

In over 30 years in the field, he has been a technical contributor, technical manager, and an executive with several high-technology firms. He has designed and implemented both TCP/IP-based and OSI-based protocol suites on a variety of computers and operating systems, ranging from microcomputers to mainframes. As a consultant, he has advised government agencies, computer and software vendors, and major users on the design, selection, and use of networking software and products.

He created and maintains the *Computer Science Student Resource Site* at ComputerScienceStudent.com. This site provides documents and links on a variety of subjects of general interest to computer science students (and professionals). He is a member of the editorial board of *Cryptologia*, a scholarly journal devoted to all aspects of cryptology. He is Networking Category Expert Writer at networking.answers.com.

Dr. Stallings holds a Ph.D. from M.I.T. in Computer Science and a B.S. from Notre Dame in electrical engineering.

*This page intentionally left blank*