



FIRST-STEP SERIES

TCP/IP first-step

Your first step into the world
of TCP/IP

- No TCP/IP experience required
- Includes clear and easily understood explanations
- Makes learning easy



ciscopress.com

Mark A. Sportack

Best-selling author and adjunct professor

FREE SAMPLE CHAPTER



SHARE WITH OTHERS

TCP/IP First-Step

Mark A. Sportack



Cisco Press
800 East 96th Street
Indianapolis, IN 46240

TCP/IP First-Step

Mark A. Sportack

Copyright© 2005 Cisco Systems, Inc.

Published by:

Cisco Press

800 East 96th Street

Indianapolis, IN 46240 USA

All rights reserved. No part of this book may be reproduced or transmitted in any form or by any means, electronic or mechanical, including photocopying, recording, or by any information storage and retrieval system, without written permission from the publisher, except for the inclusion of brief quotations in a review.

Printed in the United States of America 6 7 8 9 10

Sixth Printing August 2009

Library of Congress Cataloging-in-Publication Number:

2003109693

ISBN: 1-58720-108-9

Warning and Disclaimer

This book is designed to provide information about the basics of TCP/IP. Every effort has been made to make this book as complete and as accurate as possible, but no warranty or fitness is implied.

The information is provided on an “as is” basis. The authors, Cisco Press, and Cisco Systems, Inc. shall have neither liability nor responsibility to any person or entity with respect to any loss or damages arising from the information contained in this book or from the use of the discs or programs that may accompany it.

The opinions expressed in this book belong to the author and are not necessarily those of Cisco Systems, Inc.

Trademark Acknowledgments

All terms mentioned in this book that are known to be trademarks or service marks have been appropriately capitalized. Cisco Press or Cisco Systems, Inc. cannot attest to the accuracy of this information. Use of a term in this book should not be regarded as affecting the validity of any trademark or service mark.

Publisher

John Wait

Editor-in-Chief

John Kane

Cisco Representative

Anthony Wolfenden

Cisco Press

Program Manager

Jeff Brady

Executive Editor

Jim Schachterle

Production Manager

Patrick Kanouse

Development Editor

Grant Munroe

Senior Project Editor

San Dee Phillips

Copy Editor

Tonya Cupp

Technical Editors

Henry Benjamin

Ron Kovac

David Kurtiak

Editorial Assistant

Tammi Barnett

Cover and Book Designer

Louisa Adair

Composer

Mark Shirar

Indexer

Tim Wright

Proofreader

Katherin Bidwell

Corporate and Government Sales

Cisco Press offers excellent discounts on this book when ordered in quantity for bulk purchases or special sales.

For more information please contact: U.S. Corporate and Government Sales 1-800-382-3419
corpsales@pearsontechgroup.com

For sales outside the U.S. please contact: International Sales international@pearsoned.com

Feedback Information

At Cisco Press, our goal is to create in-depth technical books of the highest quality and value. Each book is crafted with care and precision, undergoing rigorous development that involves the unique expertise of members from the professional technical community.

Readers' feedback is a natural continuation of this process. If you have any comments regarding how we could improve the quality of this book, or otherwise alter it to better suit your needs, you can contact us through email at feedback@ciscopress.com. Please make sure to include the book title and ISBN in your message.

We greatly appreciate your assistance.



Corporate Headquarters
Cisco Systems, Inc.
170 West Tasman Drive
San Jose, CA 95134-1706
USA
www.cisco.com
Tel: 408 526-4000
800 553-NETS (6387)
Fax: 408 526-4100

European Headquarters
Cisco Systems International BV
Haarlerbergpark
Haarlerbergweg 13-19
1101 CH Amsterdam
The Netherlands
www-europe.cisco.com
Tel: 31 0 20 357 1000
Tel: 31 0 20 357 1100

Americas Headquarters
Cisco Systems, Inc.
170 West Tasman Drive
San Jose, CA 95134-1706
USA
www.cisco.com
Tel: 408 526-7660
Fax: 408 527-0883

Asia Pacific Headquarters
Cisco Systems, Inc.
Capital Tower
168 Robinson Road
#22-01 to #29-01
Singapore 068912
www.cisco.com
Tel: +65 6317 7777
Fax: +65 6317 7799

Cisco Systems has more than 200 offices in the following countries and regions. Addresses, phone numbers, and fax numbers are listed on the
Cisco.com Web site at www.cisco.com/go/offices.

Argentina • Australia • Austria • Belgium • Brazil • Bulgaria • Canada • Chile • China PRC • Colombia • Costa Rica • Croatia • Czech Republic
Denmark • Dubai, UAE • Finland • France • Germany • Greece • Hong Kong SAR • Hungary • India • Indonesia • Ireland • Israel • Italy
Japan • Korea • Luxembourg • Malaysia • Mexico • The Netherlands • New Zealand • Norway • Peru • Philippines • Poland • Portugal
Puerto Rico • Romania • Russia • Saudi Arabia • Scotland • Singapore • Slovakia • Slovenia • South Africa • Spain • Sweden
Switzerland • Taiwan • Thailand • Turkey • Ukraine • United Kingdom • United States • Venezuela • Vietnam • Zimbabwe

Copyright © 2003 Cisco Systems, Inc. All rights reserved. CCIP, CCSP, the Cisco Arrow logo, the Cisco *Powered* Network mark, the Cisco Systems Verified logo, Cisco Unity, Follow Me Browsing, FormShare, iQ Net Readiness Scorecard, Networking Academy, and ScriptShare are trademarks of Cisco Systems, Inc.; Changing the Way We Work, Live, Play, and Learn, The Fastest Way to Increase Your Internet Quotient, and iQuick Study are service marks of Cisco Systems, Inc.; and Aironet, ASIST, BPN, Catalyst, CCDA, CCDP, CCIE, CCNA, CCNP, Cisco, the Cisco Certified Internetwork Expert logo, Cisco IOS, the Cisco IOS logo, Cisco Press, Cisco Systems, Cisco Systems Capital, MGX, MICA, the Networkers logo, Network Registrar, Packet, PDX, Post-Routing, Pre-Routing, RateMUX, Registrar, SlideCast, SMARTnet, StrataView Plus, Stratum, SwitchProbe, TeleRouter, TransPath, and VCO are registered trademarks of Cisco Systems, Inc. and/or its affiliates in the U.S. and certain other countries.

All other trademarks mentioned in this document or Web site are the property of their respective owners. The use of the word partner does not imply a partnership relationship between Cisco and any other company. (0303R)

Printed in the USA

About the Author

Mark A. Sportack has worked in the information technology industry for more than 20 years. Mark's experience includes everything from computer programming and systems analysis, to running a national ISP network and hosting centers, to managing the daily technical operations of small, medium, and large enterprises. During his career, Mark has had the privilege to lead large teams of technical personnel, set technical direction and long-range plans, and manage multimillion-dollar capital and expense budgets. Mark is chief information officer for Pitney Hardin, LLP.

Mark has authored numerous books including *IP Addressing Fundamentals and IP Routing Fundamentals* for Cisco Press. He has also written *High Performance Networking Unleashed*, *Networking Essentials Unleashed*, and *Windows NT Clustering Blueprints*.

Mark holds an undergraduate degree from Montclair State University and an M.B.A. degree from Lehigh University.

About the Technical Reviewers

Henry Benjamin, CCIE No. 4695, holds three CCIE certifications (Routing and Switching, ISP Dial, and Communications and Services). He has more than 10 years experience with Cisco networks and recently worked for Cisco in the internal IT department helping to design and implement networks throughout Australia and Asia. Henry was a key member of the CCIE global team, where he was responsible for writing new laboratory examinations and questions for the CCIE exams. Henry is an independent consultant with a large security firm in Australia.

Dr. Ron Kovac is currently employed with the Center for Information and Communication Sciences at Ball State University in Muncie, Indiana as a full professor. The Center prepares graduate students in the field of telecommunications. Previous to this, Dr. Kovac was the telecommunications manager for the State of New York and an executive director for a large computing center located on the east coast. Dr. Kovac's previous studies included electrical engineering and education. Dr. Kovac has numerous publications and has consulted in both the education and telecommunications field, speaks worldwide on telecommunications issues, holds numerous certifications including the CCNA, CCAI, and almost complete CCNP.

David Kurtiak is experienced in many telecommunications technologies and is recognized as the resident expert in TCP/IP networking. He specializes in end-to-end network analysis, planning, troubleshooting, and performance tuning. David has a master's degree (MS) in telecommunications from the University of Colorado at Boulder and a bachelor's degree (BS) in information systems from the University of North Carolina at Greensboro.



Dedications

I dedicate this book in memory of Veronika J. (Sportack) Swisher. Thank you for being my sister.

I would also like to dedicate this book to my wife Karen, my son Adam, and my daughters Jennifer and Jacqueline. I don't know how I would have made it through the last year without you. If ever there was an ill-timed project, this was it. Yet, your love, support, and patience helped me to persevere.

Acknowledgments

There are many people to whom I am indebted for their assistance with this book.

First, thanks to Jim Schachterle for his patience as I missed deadline after deadline due to the events in my personal life. I know I sorely tested your patience, but thank you for sticking with me.

I'd also like to thank my technical editors for keeping me honest and not laughing too loudly at my inevitable mistakes and typos. Special thanks to my long-suffering friend and favorite technical editor, David Kurtiak. Knob Creek beckons! Thanks also to Ron Kovac and Henry Benjamin for their contributions as technical editors.

I'd also like to thank John Kane for continuing to watch over my authoring career with Cisco Press, Christopher Cleveland, Grant Munroe, and everyone else at Cisco Press that helped to make this book possible.

Lastly, I'd like to thank Michael Simmons: my friend, fellow long-distance commuter, and surrogate for this book's target audience.

Contents at a Glance

	Introduction	xviii
Part I	Introduction and Architecture	3
Chapter 1	TCP/IP: The Official Protocol of the Internet!	5
Chapter 2	What Has TCP/IP Done for Me Lately?	19
Chapter 3	The Quest for Freedom of Choice	33
Chapter 4	TCP/IP: The Networking Protocol That Changed the World	61
Part II	Protocols: The Building Blocks of TCP/IP	83
Chapter 5	Peeking Under the Covers	85
Chapter 6	Pushing the Envelope	109
Chapter 7	More Fun with IP Addresses	139
Chapter 8	Guaranteed Delivery: Your Package Will Get Delivered...Eventually!	163
Chapter 9	Best-Effort Delivery: It's Now or Never!	189
Part III	Network Services: Making Your Network Easy to Use	207
Chapter 10	Special Delivery for Special Messages	209
Chapter 11	How Do I Get There from Here?	227
Part IV	User Services: Making the Most Use of Your Network	251
Chapter 12	Connecting to TCP/IP Networks	253
Chapter 13	Smiling and Dialing	281

Chapter 14	Taking the Next Step	313
Part V	Appendix	345
Appendix A	Chapter Review Answers	347
Glossary		375
Index		387

Contents

	Introduction	xviii
Part I	Introduction and Architecture	3
Chapter 1	TCP/IP: The Official Protocol of the Internet!	5
	What Is TCP/IP?	6
	Finding TCP/IP's Roots	7
	The Birth of TCP/IP	8
	The Early Days of Networking	8
	Enabling the Internet's Success	9
	The Abilities and Limitations of TCP/IP	11
	TCP/IP's Top Five Critical Functions	13
	What It Can't Do	15
	Chapter Summary	16
	Chapter Review Questions	17
Chapter 2	What Has TCP/IP Done for Me Lately?	19
	Business and Pleasure?	19
	At Play	20
	At Work	21
	What's the Secret?	21
	Open Standards	22
	Generalizing to See the Patterns	23
	Generalizing from TCP/IP's Perspective	24
	A Closer Look at the Example	25
	The Need for Reliable Communications	26
	The Need for Timely Communications	28
	Chapter Summary	30
	Chapter Review Questions	31
Chapter 3	The Quest for Freedom of Choice	33
	Open or Closed?	34
	The Case for Closed Technologies	34
	The Consumer's Benefits	35
	The Manufacturer's Benefits	36
	The Case for Open Technologies	36
	The Benefits of Open Computing	37
	TCP/IP: The Ultimate Open Technology	37

How Do You Not Keep a Secret?	38
Open Architectures Equals Open Standards	38
Reference Models: Keeping Things Organized	39
Speaking of Political Committees...	42
EIA/TIA	43
IEEE	44
IETF	46
Layers of Layered Standards	48
OSI Reference Model	48
Using the OSI Reference Model	51
What It Looks Like	53
What It Really Does	54
Gotcha!	55
Getting a Jump on Emerging Standards	56
Using Open Standards to Create Proprietary Products	57
Making Sense of the Chaos	57
Chapter Summary	58
Chapter Review Questions	59

Chapter 4	TCP/IP: The Networking Protocol That Changed the World	61
	The Official Protocol of the Internet	61
	Groups of Functions	63
	Inside the Tool Box	64
	Tools for Users	64
	Tools for Applications	65
	TCP	67
	UDP	71
	Tools for the Network	72
	The Big Picture	75
	Dissecting TCP/IP	76
	Process/Application Layer	77
	Host-to-Host Layer	78
	Internet Layer	79
	Network Access Layer	79
	Chapter Summary	80
	Chapter Review Questions	80

Part II	Protocols: The Building Blocks of TCP/IP	83
Chapter 5	Peeking Under the Covers	85
	Tools of the Trade	85
	TCP's Functional Requirements	86
	Revisiting TCP's Functions	86
	TCP's Header	88
	UDP	92
	Revisiting UDP's Functions	93
	UDP's Header	94
	Comparing TCP and UDP	95
	IP	96
	Revisiting IP's Functions	96
	IP's Header	97
	A Closer Look at Port Numbers	100
	Well-Known Port Numbers	102
	Registered Port Numbers	104
	Private and Dynamic Port Numbers	104
	Chapter Summary	105
	Chapter Review Questions	105
Chapter 6	Pushing the Envelope	109
	The Life of a Packet	109
	Stage 1: Wrap It Up	111
	Stage 2: Pass It On	112
	Stage 3: Take It Away	114
	The Numbers Game	116
	Finding Your IP Address	117
	Decimal Numbers	119
	Binary Numbers	120
	The Architecture of an IP Address	121
	The Real Deal	122
	User-Friendly IP Addresses	123
	Decimal Numbers Are Easy to Remember	123
	Smaller Numbers Are Easier to Remember	124
	Small Decimal Numbers Are the Easiest to Remember	125
	Learning to Count All Over Again	125
	Using IP Addresses	128

	How Can You Tell a Host Address from a Network Address? 129
	One Address, Two Parts 130
	Classical IP: The Way It Was 132
	Classless IP: The Way It Is 134
	Chapter Summary 135
	Chapter Review Questions 135
Chapter 7	More Fun with IP Addresses 139
	Subdivision: It's Not Just for Real Estate 139
	The Need to Subdivide 140
	Cut It Up! 143
	Subnetting a Network 145
	Where's the Subnet Address? 147
	Checking the Math 148
	Staring at the Bits 149
	Leaving Room for the Network 151
	The Benefits and Drawbacks of Subnetting 151
	The Benefits 152
	Drawbacks 152
	Mystery Behind the Mask 153
	Decimal Masks 153
	Network Masks 154
	Subnet Masks 155
	Two Types of Subnets 158
	Chapter Summary 160
	Chapter Review Questions 160
Chapter 8	Guaranteed Delivery: Your Package Will Get Delivered...Eventually! 163
	Reliability and Networking 163
	Communication Is the Key 164
	Six Little Flags, One Great Adventure 165
	Communicating in a Crowd 166
	Start It Up! 167
	A Little Acknowledgment? 168
	Let's Talk 168
	What Did You Say? 169

Deal With This: Now!	169	
Where Was I?	170	
Later!	170	
Recapping the Conversation	171	
Communicating in a Network	172	
Getting Started	172	
Three-Way Handshake	173	
Keeping Track of Things	175	
Finding Common Ground	177	
Push It Along	179	
The Do-Over	181	
Stick a Fork in It: It's Done	182	
Multitasking with TCP	184	
Chapter Summary	186	
Chapter Review Questions	186	
Chapter 9	Best-Effort Delivery: It's Now or Never!	189
User Datagram Protocol	190	
What Do You Mean, Best Effort?	191	
Built for Speed	191	
The Header's Shape	193	
Yeah, but What's It Good For?	194	
Video Varieties	194	
Real-Time Video	195	
Not-So-Real-Time Video	196	
Sound Options	197	
It Gets a Little More Complicated	199	
UDP—In Action!	200	
The View from the Source	200	
The View from the Destination	201	
UDP Port Numbers	203	
Chapter Summary	204	
Chapter Review Questions	205	

Part III	Network Services: Making Your Network Easy to Use	207
Chapter 10	Special Delivery for Special Messages	209
	ICMP: A Protocol for System Messages	210
	The Architecture	211
	Fields and Functions	212
	Types of Messages	213
	Echo and Echo Reply	214
	Destination Unreachable	215
	Source Quench	216
	Redirect	217
	Time Exceeded	219
	Parameter Problem	219
	Timestamp and Timestamp Reply	220
	Using ICMP	221
	PING	221
	TRACEROUTE	222
	Chapter Summary	223
	Chapter Review Questions	224
Chapter 11	How Do I Get There from Here?	227
	Routers Route!	227
	What Is a Router?	228
	What Does a Router Do?	229
	How Does That Work?	231
	Learning About New Destinations	233
	Remembering Best Paths	234
	Forwarding Packets	236
	Calculating Routes	239
	Measuring Distances	241
	How It Works	241
	The Downside	243
	Measuring Quality of the Connections	244
	How It Works	245
	The Downside	246
	Applying Rules	247
	Chapter Summary	248
	Chapter Review Questions	249

Part IV	User Services: Making the Most Use of Your Network	251
Chapter 12	Connecting to TCP/IP Networks	253
	Anatomy of a Local-Area Network	253
	Wire-Based Networks	255
	Coaxial Cables	256
	Twisted-Pair Wire	256
	The Best of Both Worlds	257
	Benefits of Wire-Based Networks	258
	Drawbacks of Wire-Based Networks	259
	Wireless Networks	260
	Benefits of Wireless Networks	263
	Drawbacks of Wireless Networks	263
	Configuring a Network Connection	265
	Laptop Versus Desktop Computers	266
	Laptops	266
	Desktops	267
	Connecting to a Wire-Based Network	268
	Connecting to a Wireless Network	271
	Wireless B and G	272
	Wireless A	272
	Connecting Without Wires	273
	Chapter Summary	278
	Chapter Review Questions	278
Chapter 13	Smiling and Dialing	281
	What, Exactly, Is the Internet?	281
	All ISPs Are Not Created Equal	282
	Picking an ISP	284
	Picking an Access Technology	286
	More Power!	286
	What's the Catch?	287
	What's the Story with Dial-Up Connections?	288
	How Does It Work?	289
	Dial-Up Today	292
	When All Else Fails...	293
	Even if Cable Is Available...	293

Configuring a Dial-Up Connection	295
Tracking Down the Wizard	300
General Tab	302
Options Tab	303
Security Tab	305
Networking Tab	306
Advanced Tab	308
Chapter Summary	310
Chapter Review Questions	310
Chapter 14	Taking the Next Step 313
Information, Please!	314
Checking Your Configuration	314
Internet's Yellow Pages	317
InterNIC's whois	318
Packet Internet Groper	321
ARIN's whois	323
TCP/IP's Bugshooting Tools	325
Map It Out	326
Check the Browser	326
Check TCP/IP	329
Show Me the Numbers	330
Check TCP/IP's Performance	332
Check the LAN	336
Tracking Down Problems Beyond Your Control	338
Back to ping	339
Tracing the Route	340
Chapter Summary	342
Chapter Review Questions	343
Part V	Appendix 345
Appendix A	Chapter Review Answers 347
Glossary	375
Index	387

Introduction

TCP/IP is the language of the Internet. As such, it has become the single-most widely used communications protocol ever. It has literally revolutionized our world by enabling the Internet to become what it is today. Yet, for all its success, it remains a shadowy mystery to all but a handful of computer geeks and network nerds.

I'll be the first to admit that TCP/IP is as complex and mysterious a protocol suite as you could hope to find. That's a shame because it doesn't need to be. TCP/IP has a rich history and offers many capabilities that the vast majority of today's Internet users don't know about. Although you could live a lifetime without ever knowing about or using these features, you would also be missing out on quite a bit!

To my twisted way of thinking, the only way to truly understand the Internet and learn how to use it to maximum advantage requires that you first understand TCP/IP. Armed with the knowledge you gain from this book, you can take the first step into your future and become more than just another point-and-click Internet user.

Goals of This Book

My goal in writing this book was to help make TCP/IP as understandable as possible for all people, not just computer science majors or network engineers. The Internet has become everybody's most powerful tool yet so much of it is hidden behind a graphical user interface. It seemed only fair and fitting that somebody should write a book for the nontechnical Internet user community to help them harness more of the power of the Internet.

Who Should Read This Book

If you use the Internet, or any other network that uses TCP/IP, you should read this book. I intentionally stayed away from technical jargon and used everyday analogies to help you understand otherwise technical material. However, TCP/IP is sufficiently complex that many a fine IT engineer can benefit from the plain English explanations and illustrations in this book.

How This Book Is Organized

This book is organized into a logical, step-by-step approach to building a comprehensive understanding of TCP/IP and the Internet. That approach is demonstrated for you in the following outline of the chapters:

Part I, “Introduction and Architecture”

- **Chapter 1, “TCP/IP: The Official Protocol of the Internet”**—In this chapter, you learn what TCP/IP is, what it isn’t, and where it came from. This provides you with a foundation that the rest of the book builds on.
- **Chapter 2, “What Has TCP/IP Done for Me Lately?”**—TCP/IP literally supports everything that you do on the Internet and any other private network that uses it. When you think about it, that’s a Herculean task that implies TCP/IP can support virtually any type of application anywhere around the world. This chapter shows you just how TCP/IP manages to successfully meet that challenge.
- **Chapter 3, “The Quest for Freedom of Choice”**—TCP/IP is perhaps the best example of a nonsecret technology. Unlike many companies that jealously guard their trade secrets, TCP/IP was made, and continues to be developed, in an open and public forum. This chapter shows you how TCP/IP became the best-unkept secret in the world.
- **Chapter 4, “TCP/IP: The Networking Protocol That Changed the World”**—One of the hallmarks of success is consistency over time. TCP/IP achieves consistency by having a carefully sequenced set of steps that it must perform time after time. This chapter shows you how TCP/IP manages to achieve consistency and introduces you to those sequenced steps that it needs to perform when sending and receiving your data.

Part II, “Protocols: The Building Blocks of TCP/IP”

- **Chapter 5, “Peeking Under the Covers”**—Whenever you want to see how something was made, or really works, you need to look beyond what’s superficial and obvious. In this chapter, you see what’s under the hood of TCP/IP so that you can see the actual mechanisms that enable it to work.

- 
- **Chapter 6, “Pushing the Envelope”**—All data sent across the Internet or any other TCP/IP network is wrapped up in an envelope (known more properly as a packet) and pushed through the network; however, to get to where it needs to go, that envelope *must* have an address. This chapter introduces you to the seemingly arcane mathematics of the Internet’s addressing system. Don’t worry: It’s not that bad!
 - **Chapter 7, “More Fun with IP Addresses”**—One important trick that can be performed with IP addresses is taking a big block of them and breaking them down into many smaller blocks. That lets you take a block of addresses designed for use on one network and use it to provide unique addresses for many networks. That concept, known as subnetting, is demystified for you in this chapter.
 - **Chapter 8, “Guaranteed Delivery: Your Package Will Get Delivered...Eventually!”**—The notion of guaranteeing the delivery of your data to a remote computer is paramount in a TCP/IP network. In this chapter, you learn how TCP/IP actually does that. In the process, you might be surprised to learn of a “gotcha” or two that lies hiding in wait.
 - **Chapter 9, “Best-Effort Delivery: It’s Now or Never”** —This chapter is all about a stripped-down, built-for-speed transport mechanism built into TCP/IP. You learn about this mechanism, UDP, how it is used, why it’s important, and which of your favorite applications use it without you even knowing it!

Part III, “Network Services: Making Your Network Easy to Use”

- **Chapter 10, “Special Delivery for Special Messages”**—Not all messages sent across a TCP/IP network are sent by users and their computers. Sometimes, machines need to communicate and coordinate activities with each other. For such instances, there is a special delivery mechanism, ICMP. This chapter introduces you to ICMP and shows you how it works and the role it plays in a TCP/IP network.
- **Chapter 11, “How Do I Get There from Here?”**—Ahh, the universal question. You will be pleased to know that knowing how to get from Point A to Point B is quite the mathematical science in a TCP/IP network. That science is known as routing, and in this chapter, you learn more about how your data actually finds its way to where you want it to go.

Part IV, “User Services: Making the Most Use of Your Network”

- **Chapter 12, “Connecting to TCP/IP Networks”**—In this chapter, things become a bit more real; you learn how to connect to a TCP/IP network.
- **Chapter 13, “Smiling and Dialing”**—This chapter builds upon the previous chapter by showing you that there is still a place in a modern TCP/IP network for a modem and good, old-fashioned dial-up access to the Internet.
- **Chapter 14, “Taking the Next Step”**—In this final chapter, you advance beyond the first step toward understanding TCP/IP and take the next step: learning how to diagnose network problems from your computer.

Part V, “Appendix”

- **Appendix A, “Chapter Review Answers”**—This appendix contains answers and explanations to the “Chapter Review Questions” that appear at the end of each chapter.
- **Glossary**—The glossary is a tool that you can easily reference as you come across key terms throughout the book.

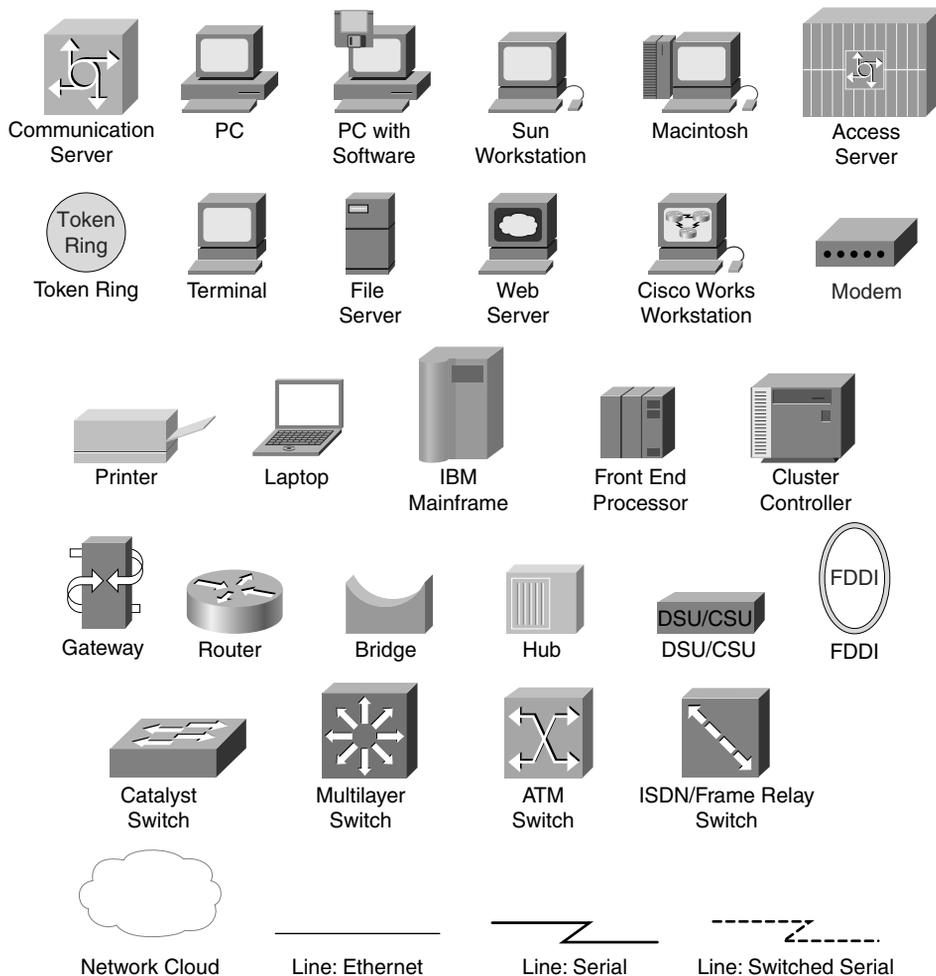
Stuff You’ll Find in This Book

This book includes several features to help you master wireless topics. Here’s a summary of the elements that you’ll find:

- **What You Will Learn**—Every chapter begins with a list of objectives that are addressed in the chapter. The objectives summarize what you learn in the chapter.
- **Key terms and glossary**—Throughout this book, you will see key terms formatted with bold and italics. These terms are particularly significant in wireless and networking. So, if you find you aren’t familiar with the term or at any point need a refresher, simply look up the term in the glossary toward the end of the book to find a full definition.

- 
- **Chapter Summaries**—Every chapter concludes with a comprehensive “Chapter Summary” that reviews chapter objectives, ensuring complete coverage and discussing the chapter’s relationship to future content.
 - **Chapter Review Questions**—Every chapter concludes with “Chapter Review Questions” that test the basic ideas and concepts covered in each chapter. You can find the answers and explanations to the questions in Appendix A.
 - **Nontechnic headings and explanations**—The headings and text used throughout this book avoid the use of technical terms when possible, focusing instead on words that connote something about the underlying concepts.

The illustrations in this book use the following icons for networking devices and connections:



What You Will Learn

After reading this chapter, you should be able to answer the following questions:

- ✓ What is the IETF?
- ✓ What is the difference between open standards and closed, or proprietary, technologies?
- ✓ What is interoperability?
- ✓ What is the benefit of interoperability?
- ✓ How do you create and maintain the technical standards that enable the Internet and its technologies to be interoperable?
- ✓ What is a reference model and why is it needed?
- ✓ What is logical adjacency?



CHAPTER 3

The Quest for Freedom of Choice

The freedom of choice: It seems like such a basic right that people almost take it for granted. That's especially true when it comes to choosing whose hardware and software products to buy for your home or office networks. You're obviously free to spend your money on technology products in any way you please. Customers have grown accustomed to buying computers from one company, printers from another, and network equipment from yet another. More to the point, buyers expect all these different brands to work together perfectly. That capability is known as *interoperability*. For example, customers expect to buy a PC from one manufacturer, a network interface card (NIC) from another manufacturer, and local-area network (LAN) equipment from a third company and still expect all to work together seamlessly.

Things didn't always work that way. In fact, it wasn't so many years ago when technology manufacturers couldn't imagine sharing their technical secrets with anyone and took great pains to ensure that if you purchased their brand, that you also had to purchase their peripherals. They made certain nothing else would work with their computer!

This chapter introduces the concept of vendor interoperability, including how it became possible, how new technologies are designed for interoperability right from the start, and how the standards permitting interoperability are created and maintained. These important concepts, not coincidentally, also tell TCP/IP's story.

Open or Closed?

Which is better, open or closed? The question is simple to ask, but not necessarily easy to answer. For one thing, open and closed aren't physical states, such as an open door versus a closed door. In the context of information technologies, *open* or *closed* refers to the way a product or technology was developed. You can't determine that development just by looking at the product in question.

A *closed*, or *proprietary*, technology is developed in secret. Details about how it is made or how it works are kept a secret. Conversely, an *open* technology is developed publicly. Its technical details are shared freely with anyone who would like to know them. These two polar opposites represent the before and after perspectives of the information technology industry.

Emotionally, the notion of working hard, developing something special, and then keeping it a secret so nobody else can profit is appealing. In some industries, collaborating with a competitor is self-defeating. Consequently, the benefits of collaborating publicly with competitors on a new technology are not readily apparent.

Take a closer look at the benefits and limitations of both open and closed approaches to technology development. Then you can better understand them and appreciate why TCP/IP might be the ultimate open technology.

The Case for Closed Technologies

The time-honored approach to creating an advantage over competitors was to secretly develop something and then jealously guard that secret. You created something special—your competitors don't know how and buyers can only buy it from you. This approach is logical and intuitive. You invested your time and effort to develop something, so why give it away? Why let someone else profit from your efforts? When you look at closed or proprietary technology from that perspective, it seems quite logical and even fair. The business world is full of trade secrets, from McDonald's secret sauce to the Colonel's blend of secret seasonings.

It's an accepted business practice, and keeping trade secrets provides real and tangible benefits.

In fact, the first few generations of information technologies were created and sold under this mindset. One great distinction lies between secret recipes and proprietary computer technologies. Those proprietary technologies won't work with other products and other products won't work with them: They were designed to not interoperate with other vendors' products!

The Consumer's Benefits

The benefits of proprietary computing are few and easy to enumerate. This approach greatly simplifies the following things for consumers:

- **Purchasing decisions.** If you pick Brand X, you pick Brand X for everything! Only Brand X has the cables that work with its computers. In a proprietary computer market, deciding to purchase a particular brand is a de facto decision to buy all peripheral parts, products, and supplies from the same company for as long as you continue to own and operate that computer system.
- **Vendor management.** In an attempt to find something positive to say about such an approach to buying technology, it's that vendor-management duties are greatly simplified. You only have one vendor to worry about! Instead of trying to keep track of multiple vendors with different products and prices, you can deal with only one vendor that handles all your needs. Make one single call to a single vendor when you need new items.
- **Trouble escalation.** Troubleshooting is greatly simplified in this instance. You still need to identify potential causes, isolate them, and test them one at a time until you found the problem's source. However, you never have to worry about your network vendor pointing the finger at your computer vendor and vice versa: All parts come from the same vendor! You can rest assured it all works together because it was all designed to work together.

Aside from this paltry list of consumer benefits, the benefits of a closed approach to information technologies accrue to the manufacturers.

The Manufacturer's Benefits

From the manufacturer's perspective, the consumer benefits listed in the previous section are nothing more than reflections of their own significant benefits.

The total investment is so great that customers find it nearly impossible to change vendors. They have to scrap their entire investment to do so! Consequently, the customer becomes more a hostage than a customer.

Think about it: A proprietary approach to a computer system is an all-or-nothing proposition. The customer loses all leverage over the vendor as soon as she commits to purchasing that vendor's products. Needless to say, this works in the vendor's favor with respect to negotiating prices. Thus, manufacturer efforts to create and maintain proprietary products are rewarded with healthy profit margins.

Of course, anyone can buy one a product (a cable, for instance) and take it apart to figure out how it was made. That person could then manufacture his own version of those proprietary cables and sell them at a discount relative to the manufacturer's prices. Such practices, though not necessarily illegal, run contrary to manufacturers' goals of keeping their products tightly integrated and proprietary. Manufacturers fought this practice by declaring their warranties and maintenance agreements null and void if anything other than their own products were used.

A proprietary approach to computer systems afforded manufacturers tremendous profit potential and long-term customers. They stubbornly fought and resisted efforts to circumvent their secretive approach. Yet, we know that this approach has faded into oblivion...at least within the world of information technologies! In its place is a radically different paradigm—an industry in which customers expect the products of different companies to work together perfectly.

The Case for Open Technologies

A technology intentionally developed to permit multivendor interoperability is known as an *open* technology. In the vast majority of industries, giving away your secrets can be fatal to your business. In addition, you just looked at the how the computer industry started out following that business philosophy.

However, information technologies are a different story! You almost have to give away your secrets if you want any chance at all of making money from your newly developed product. That might sound like a paradox, and it is in some ways. It's an approach that works so well, however, that it became an almost subconscious expectation amongst the world's computer users.

The Benefits of Open Computing

The benefits of open computing platforms should be fairly obvious:

- **Multivendor interoperability**—The most obvious benefit of open computing has been this chapter's focus thus far: the ability to build a networked computing infrastructure from pieces mixed and matched from any brand you like. This allows you to pick the best of breed in each technology category, such as servers, printers, network switches, and so on.
- **Price-based competition between manufacturers and vendors**—The single biggest benefit of choosing technology suppliers is getting to shop around for the best product and the best prices. Unlike the days when you picked one manufacturer and became its hostage, today the consumer is in the driver's seat.
- **Communications between different end systems**—A subtle implication of multivendor interoperability is that with the right set of open standards, you can share data between computers that aren't connected to the same network. All you need is for some connection (however slight or indirect) to exist between the two networks and to agree on which protocol to use.

TCP/IP: The Ultimate Open Technology

By now you're probably wondering why so much ink was spent talking about open versus closed technical architectures. Rest assured this reason is good! You see, TCP/IP is an open technology. If it weren't an open technology, you might have an awful time trying to access your favorite websites.

For example, when you access www.cisco.com, you don't care about the computing platform or operating system, nor do you care who developed the TCP/IP protocol

stack that runs that website. Instead, you trust that you can access that website and all its contents thanks to the open standards that underlie the Internet's technologies—including TCP/IP.

These days, you might tend not to think much about buying a TCP/IP protocol stack; it's just not a piece of software that gets purchased. There was a time, back at the dawn of personal computing's time, when you had to shop around, purchase TCP/IP, and then figure out how to install it. Today, TCP/IP is bundled into every computer operating system that gets sold. Thus, deciding where to buy your copy of TCP/IP is moot. Instead, you have to decide only which operating system and computing platform you like.

How Do You Not Keep a Secret?

Having explored the ins and outs of open versus proprietary architectures, it should be fairly obvious why the benefits of openness have proven so compelling.

The next logical question is this: How do you develop a technology without keeping it secret? That sounds funny—usually keeping a secret is difficult. As you see in this section, keeping a secret can be child's play in comparison to the complex process by which new open technologies are hammered out in political committees.

Open Architectures Equals Open Standards

Every open technology is based on an open standard. An *open standard* is simply one not kept secret. That open standard might have been developed privately and then introduced publicly, but the result is the same: It is a public technology.

Alternately, it might have started out as a committee initiative to explore a vexing problem. Regardless of its actual origin, every open technology conforms to a set of technical specifications that are laid out in a public document. Typically, a neutral organization maintains the open standard. Such organizations are known as *standards bodies*.

Today, everything has open standards, from the size and shape of credit cards to what dictates how the Internet works. The more complicated the function, the more likely it is to have multiple open standards working together to perform that function. That last point is worth further investigation. Though it's quite a neat trick to get a political committee to agree on the technical specifications for one technology, creating a framework within which multiple open technologies can interoperate can quickly get out of control.

Fortunately, a rather handy mechanism keeps things organized in both the short and long term. That mechanism is known as a *reference model*.

Reference Models: Keeping Things Organized

One of the simplest, yet most confusing, concepts to understand is a reference model. A *reference model* is a simple tool. In essence, this framework enables you to dissect a complicated function into a series of individual steps and keep those tasks organized relative to each other. A common framework enables different people or companies to collaborate even if they aren't working directly with each other. That's a powerful and valuable capability!

When you start dissecting a function into its individual components, one thing becomes remarkably clear: Those tasks usually have to occur in the same sequence for things to work right. Using a rather mundane and not necessarily technical example to illustrate this point, consider driving a car. The first step is gaining entry to the inside of the vehicle. Usually that's done by unlocking the door, but not always. Some people leave their doors unlocked, or it might be a top-down convertible whose driver can hop inside. (Alternatively, an entire profession is dedicated to gaining entrance to cars without having to rely on keys, but that's another issue altogether!) The point is that entering a vehicle—regardless of how you achieve that task—is necessarily the first step in the function of driving.

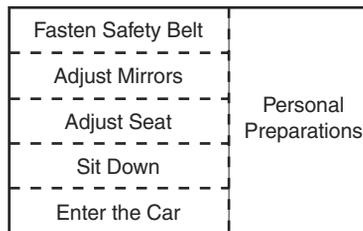
The next steps are sitting in the driver's seat, buckling up, and making sure your seat and mirrors are properly adjusted. The sequence of these events doesn't leave a lot of room for debate. You could, for example, adjust your mirrors before adjusting

your seat, but then you'd probably have to readjust the mirrors after getting the seat in the right position.

Thus, these tasks have a natural or logical sequence. A more subtle point is that they can be lumped together and generally regarded as a set of related functions. Their relationship is that people do these things to prepare to drive. For the sake of example, call this set of related functions Personal Preparations. The next set of functions comes immediately after completing Personal Preparations.

Start visualizing your reference model for driving a car. Figure 3-1 shows the Personal Preparations in their logical order.

Figure 3-1 Organizing the Personal Preparations into a Simple Reference Model

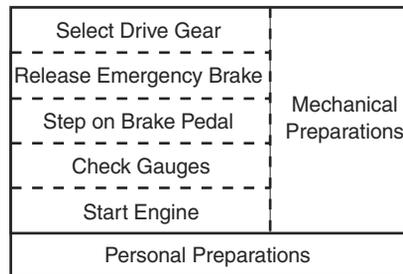


Despite completing Personal Preparations, you're still not quite ready to drive. Take your preparations a bit further and actually engage the mechanical systems that let you drive the car. Again, letting logic guide you, the next thing to do is start the car. Thus, inserting the key in the ignition and turning it should result in the engine starting. Release the parking brake and put the car into gear!

These steps should be separate and distinct from Physical Preparations because they represent a higher level of readiness: You are preparing the car's mechanical systems for operation. Call this group of functions Mechanical Preparations.

Figure 3-2 shows the reference model with Personal Preparations and Mechanical Preparations. At a glance, you should see how they relate to each other and why you want to complete all Personal Preparations before starting Mechanical Preparations.

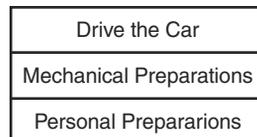
Figure 3-2 Adding a Higher Layer of Functionality to the Reference Model



After making yourself comfortable behind the steering wheel and engaging (or disengaging) all the necessary mechanical systems, you are finally ready to drive! Rather than try to deconstruct the act of driving into discrete tasks, take one last look at your fictitious reference model. It shows how driving is yet another higher level of functionality and how that functionality relates to other functions.

Driving can be thought of as the application and the reason you're bothering with both Personal and Mechanical Preparations. The relationship between these tiers of functions is depicted in Figure 3-3.

Figure 3-3 The Act of Driving Becomes the Highest Layer of Functionality in the Reference Model



Note that the logic works in two directions. This model is based on the logical steps needed to start and drive a car. It works in reverse, too. The model shows the logical sequence of events in which you would stop driving a car. You might not need all the same steps, but the sequencing of steps you do need remains valid, as does the grouping around layers of functions.

For example, you stop driving (which means stop the car), disengage the mechanical systems (first the transmission, then the engine), and deal with Personal Preparations (unbuckle your seatbelt and get out of the car). This important concept takes on greater meaning when you look at reference models for data communications later in this chapter.

Although this rather simple example demonstrates the concept of a dividing a complex function into a logically arranged sequence of individual tasks, it's important to recognize that no standards body governs this function! The layered sequence of tasks might be a de facto standard, but that's logic and common sense at work, not a political committee.

The next section looks at some real reference models that are indispensable when it comes to understanding open standard networking technologies and the standards bodies that developed them.

Speaking of Political Committees...

Interoperable products requires companies to agree on the technical specifications for those products. In other words, a consensus must be reached on important design criteria, such as what the product will do, how it will do it, what it won't do, and so on. Developing such a consensus amongst competing companies isn't easy and, more often than not, requires a neutral third-party to mediate or judge during the consensus-building process.

The world, it seems, is full of neutral third-party organizations whose business is to standardize all kinds of things. Rather than bore you to tears with an exhaustive review of all the standards bodies, the discussion is limited to just three that set

standards for data networking and communications technologies. You look at three organizations:

- Electronics Industry Association and the Telecommunications Industry Association (EIA/TIA)
- Institute of Electrical and Electronic Engineers (IEEE)
- Internet Engineering Task Force (IETF)

Many other organizations are chartered to set and maintain technical standards, some of which focus on networking technologies. This chapter focuses on these three.

EIA/TIA

One common set of technical standards is attributed to the EIA/TIA. These important standards focus on the physical wires that build networks and carry data between machines connected to networks. The acronym EIA/TIA actually stands for two different standards organizations: the *Electronics Industry Alliance (EIA)* and *Telecommunications Industry Association (TIA)*. Thus, standards from the EIA/TIA are endorsed by both organizations.

This is not uncommon. Quite frequently, two or more standards organizations see a technology as being their responsibility. Such overlap usually results in cooperation, rather than competition, for the development of that standard. Sometimes that cooperation means working together to actively develop a standard; other times it means one group does the work and the other endorses its output.

The EIA/TIA standards for network cabling are interesting in that they do *not* specify physical standards! Instead, they standardize a set of performance levels that cables must meet or exceed. Most technology standards spell out, in minute detail, everything about how a technology should be built, how it will work, and all its specific attributes and capabilities. However, this is just a piece of wire; there's not much to it! You shouldn't care about what it looks like, or what kind of metal it is, or even how thick that metal might be. All you need to know is that it

meets your needs. Consequently, the EIA/TIA settled on standardizing performance levels as opposed to more complex physical standards that define the wires' physical properties.

The EIA/TIA performance levels, known more properly as *categories of performance*, are numbered 1 through 6. The higher the category number, the higher the performance capability. Performance capabilities are defined in terms of how fast can data be transmitted and for what distance. The most commonly used cable these days is known as **Cat-5**, which is an abbreviation of Category 5. Cables that comply with the EIA/TIA performance Category 5 support transmissions of up to 100 megabits per second for up to 100 meters. Cables must meet this minimum capability set before they can be advertised and sold as Cat-5 cables.



For more information on the EIA, please visit its website at <http://www.eia.org>. Additional information about the TIA and its standards activities can be found at <http://www.tiaonline.org>.

IEEE

The *Institute of Electrical and Electronic Engineers* (IEEE or *I Triple E*, as it is more commonly pronounced) is an independent, nonprofit organization that sponsors research and establishes the technical standards for a wide variety of electrical and electronic technologies. The IEEE's contributions to the data networking industry are numerous, but the single most significant was the research project it launched in February of 1980. That project, known formally as *Project 802* in deference to the year and month it launched, defined the technical standards for **local-area networks (LANs)**. That research initiative resulted in the standardization of numerous LAN technologies, including Ethernet and Token Ring (among other, less recognizable ones).

Ethernet was a particular standardization priority. This LAN technology was developed by scientists at Xerox Corporation's Palo Alto Research Center (PARC) as a way of sharing an expensive new laser printer. Over time, the value

of the LAN they created became apparent, and steps were taken to market it commercially. These steps resulted in a couple of Ethernet variations.

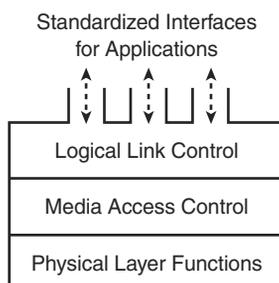
Though each variation could legitimately claim to be Ethernet, they were not necessarily compatible. Consequently, the IEEE set out to create an open standard version of Ethernet, as well as an open standard version of Token Ring. The bonus of using a layered model to organize their efforts: Interoperability between Ethernet and Token Ring also became possible.

To create technical standards for all the LANs popular at that time, the IEEE launched several engineer teams, each asked to develop the standards for a specific task. Recognizing that certain functions were common to any LAN, the IEEE sought to avoid duplicating the wheel. The first thing they did was identify all those logical steps—and their sequence—necessary in any LAN. This became the foundation for the IEEE's 802 LAN reference model.

Some of these basic steps and functions included a system for end-point addressing and tools for monitoring and managing the LAN. With the foundation built, all that was left was chartering different teams to develop the rest of the LANs that needed standardization.

To help correlate activities between the various engineering teams, the IEEE developed a reference model. This model helped everyone understand the boundaries of certain functions and kept everything organized. Figure 3.4 shows the IEEE's reference model.

Figure 3-4 The IEEE's Reference Model for Local-Area Networks



The Project 802 reference model and the steps it organized became the foundation for all the IEEE's LAN standards within Project 802. Each of the LANs standardized within that initiative, including Ethernet (known as IEEE 802.3) and Token Ring (IEEE 802.5), shared this common foundation.

It is important to note that although the IEEE developed protocol standards for sending data out over several types of physical wires, it did not develop standards for those wires. That distinction is subtle, but important. Ethernet and Token Ring needed to support transmission over wires, so they relied on the work of other standards bodies, such as the EIA and TIA.



For more information on the IEEE, you can visit its website: <http://www.ieee.org>.

IETF

The *Internet Engineering Task Force (IETF)* is responsible for developing and maintaining the Internet's technologies, as well as for guiding its growth and development. A short list of their achievements include the definition of TCP/IP, IP version 6, *Simple Mail Transfer Protocol (SMTP)*—which is the foundation for literally exchanging e-mail between open systems—and many other equally significant technologies.

The IETF is composed of engineers who volunteer their time to the various development efforts constantly underway at the IETF. Typically, a working group is chartered to perform a specific task. Numerous working groups might be launched to tackle related tasks. In that manner, a large and complex project can be completed more quickly, because of parallel work on smaller pieces.

Literally everything the IETF does is in the open; every participant in each working group has an equal vote in the proceedings and recommendations. Consensus is the mechanism by which standards are developed and accepted. Plus, consensus

and request for input aren't limited to select members of any given working group. You see, all work is documented in a publicly accessible document. These documents, known collectively as *Requests For Comments (RFCs)*, are available without charge to anyone via the Internet.

The phrase *Request For Comments* is a bit misleading. Not all RFCs are requests for comments in the literal sense. Some are little more than wild ideas an individual created and then posted for the world to see. In the spirit of openness, anyone can write and publish an RFC for the Internet community's consideration. Most RFCs are created by a working group that the IETF sponsored to solve a specific problem. Such RFCs probably carry the weight of a technical standard someday, although it might take a while for them to be accepted.



Some RFCs are jokes. It has become tradition for an outrageous technical proposal to be published on April 1 each year. To the uninitiated, these might at first appear serious, but the date should be a clear indication that someone is trying to have fun on April Fool's Day.

To learn more about the IETF, please visit its website at <http://www.ietf.org>. To see the many RFCs that spell out the rules for virtually every aspect of TCP/IP and the Internet, please visit the RFC page on the IETF website.

Each RFC is serially numbered in the order it was published. E-mail, for example, came to life through RFC 821 in 1980. That document has long since been made obsolete by much more complicated and feature-rich versions of e-mail. Yet, you can see it at <http://www.ietf.org/rfc/rfc0821.txt>. Substituting any four-digit number for the 0821 in the preceding URL lets you browse that RFC.

RFCs are, quite literally, the source documents in which all the Internet's technical details are published. The vast majority aren't fun to read, but they are available. That's just how open the IETF is when it comes to enabling interoperability via the Internet!

Layers of Layered Standards

Having just seen a small sample of the standards bodies that organize data networking and communications technologies, you might have noticed that they focus on different aspects of network-based communications. The EIA and TIA standards are limited to the wires carrying electronic signals. The IEEE is best known for its work in standardizing Ethernet and other LAN technologies. Lastly, the IETF focuses on the Internet and all the technologies that let you communicate between LANs. Communicating from one LAN to another LAN is known as *internetworking*, and TCP/IP has become the world's preferred internetworking protocol.

The work of these three different standards bodies do not overlap; each focuses on a distinct area. However, their work is highly complementary! All three sets of standards are necessary to all network functions, including Internet functions. Assume you use at least three sets of open standards when you are online and that those standards always work well. The standards' creators logically, then, can keep things coordinated between themselves.

You saw the IEEE reference model for Ethernet and took a look at the IETF reference model for TCP/IP. The two models bear little, if any, resemblance to each other. Yet the two work so well together that IP over Ethernet has become the de facto standard for LANs around the world, including large enterprise networks and home networks.

How do you make sure two technologies, developed separately, continue to interoperate well over time? The answer is simple: by using a neutral, third reference model! The reference model most frequently used in this capacity is the OSI reference model.

OSI Reference Model

Sometimes, the alphabet soup-like atmosphere of the information technology industry gets carried to a ridiculous extreme. This section's reference model and the standards body that developed it form a marvelous example of that extreme.

The *International Organization for Standardization (ISO)* was chartered by the United Nations and founded in 1946. Its mission is to set global standards for virtually everything. Everything, that is, except for anything electrical or electronic. ISO developed a generic model for the interconnection of open systems. That model is known as the *Open Systems Interconnection (OSI) reference model*. This model contains seven layers that encompass every aspect of communication between networked computers.

Although it doesn't describe a particular product or technology, the OSI reference model has become the de facto standard means of correlating other, technology-specific reference models and for ensuring interoperability between open computer systems. For this reason, it has become the single most frequently encountered reference model. The problem is that though it is overused, it is seldom adequately explained. Consequently, most people who encounter it for the first time usually walk away, scratching their heads and wondering what it is all about!

ISO's OSI (how's that for alphabet soup?) reference model is shown in Figure 3-5.



ISO's official name is the *International Organization for Standardization*. Conventional wisdom holds that it should be identified by its abbreviation: IOS. That Anglocentric abbreviation is for a global organization. To transcend language barriers, ISO was chosen. In Greek, which forms the foundation of many languages, *iso* means equal or standard. That seems much more fitting for an international organization.

You might be wondering how a reference model for connecting computer systems isn't a violation of the ISO charter. Remember: Things electrical or electronic are not within their domain. (Yet connecting computer systems seems to require electricity!) The answer is that the model describes a standardized *process*, but not any particular electronic or electrical *product*. Table 3-1 describes the layered functions from the top up.

Figure 3-5 OSI Reference Model

OSI Reference Model Layer Description	Layer Number
Application	7
Presentation	6
Session	5
Transport	4
Network	3
Data Link	2
Physical	1

Table 3-1 OSI Reference Model

Layer #	Layer Name	Description of Functions
7	Application layer	This layer is the top of the stack, or the highest level of functionality. In the driving example, the application drove the car. In the OSI model, the application layer is your computer program. It could be your browser, e-mail, or instant messaging software.
6	Presentation layer	The presentation layer is responsible for directly accepting data from the application, as well as preparing data that was received from another computer for the application. This could include things like encrypting or decrypting data and a lot of other, less familiar, functions.
5	Session layer	The flow of communications between two computer systems is known as a <i>session</i> . The session layer describes all the mechanisms and processes needed to manage a session.

Table 3-1 OSI Reference Model (continued)

Layer #	Layer Name	Description of Functions
4	Transport layer	The next set of necessary functions manages the data sent and received in a session. When sending data, this includes breaking received data into smaller pieces for transmission and uniquely numbering each piece so the recipient knows how reassemble them. When receiving data, this set of steps includes making sure the data arrives intact (not damaged) and then putting everything together in its original order before handing the data off to the session layer.
3	Network layer	A communications session doesn't necessarily always occur between two computers on the same network. Sometimes, those computers are literally half a world away from each other. In such cases, the network layer contains the mechanisms that map out the best route for that session.
2	Data link layer	The data link layer is where the rules, processes, and mechanisms for sending and receiving data over a LAN are defined.
1	Physical layer	This layer includes all the procedures and mechanism you need to both place data onto the network's wire for transmission and to receive data sent to you on that same wire.

Note that, like your model for driving a car, these function layers work in both directions. In this case, those two directions are sending and receiving data.

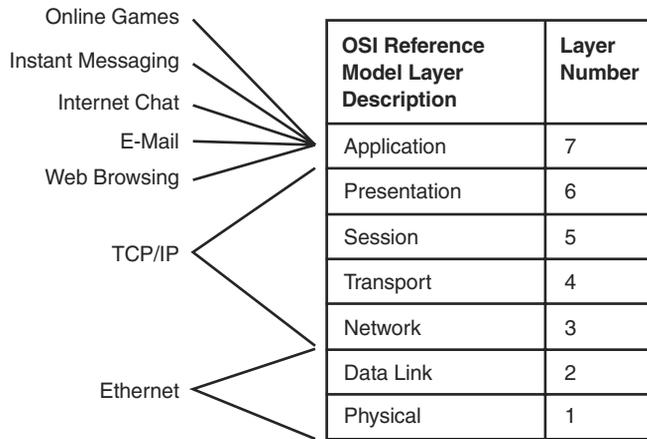
Using the OSI Reference Model

The best way to understand the OSI reference model is to remember that it is a framework that establishes the sequence of events that interconnect two open computer systems. No single product or technology fulfills the requirements of all seven layers! Instead, you tend to find that communications between open systems is more a patchwork between at least four different sets of open-standard technologies.

Consequently, the OSI reference model serves marvelously as a neutral frame of reference for correlating two or more dissimilar technologies.

Take a look at Figure 3-6, which shows the OSI model correlating the functions of the EIA/TIA, IEEE, and IETF open standards.

Figure 3-6 Using the OSI Reference Model to Correlate Functions on Other, Technology-Specific Reference Models



In Figure 3-6, you can see that any good piece of software, such as e-mail, takes care of the functions laid out in Layers 5, 6, and 7. The IETF standards for TCP/IP typically take care of Layers 3 and 4. Layer 4—the transport layer—provides the logic for accepting data from applications (a higher level of functionality) and preparing that data for transport across a network. Layer 3, the network layer, provides the mechanisms for finding and accessing a remote computer system across a network.

TCP goes beyond just being a Layer 4 technology: It encompasses many small utilities that span up through Layer 6. A visual representation of this can be misleading—truthful, but misleading. TCP/IP focuses on Layers 3 and 4 of the OSI model. You use it to support other products that span Layers 5–7. Such products are known generically as application software, and they can include anything from e-mail and browsers to chat or instant messaging. All need TCP/IP for their Layer 3 and 4 mechanisms.

Figure 3-6 also shows how the IEEE's Ethernet standard occupies Layers 1 and 2 (physical and data link layers, respectively). This illustration does not make clear why the EIA/TIA standards are not considered part of the physical layer. The answer lies hidden in the fact that this is a logical model only—it does not describe a physical product or technology. Thus, what is described in the physical layer isn't the actual wire. Instead, the physical layer is limited to the processes and mechanisms required to place data on that wire and to receive data on that wire. This includes the wire's required performance category, the physical connector at its ends (such as the modular jack at the end of your telephone cord), but does *not* include a physical description of the wire.

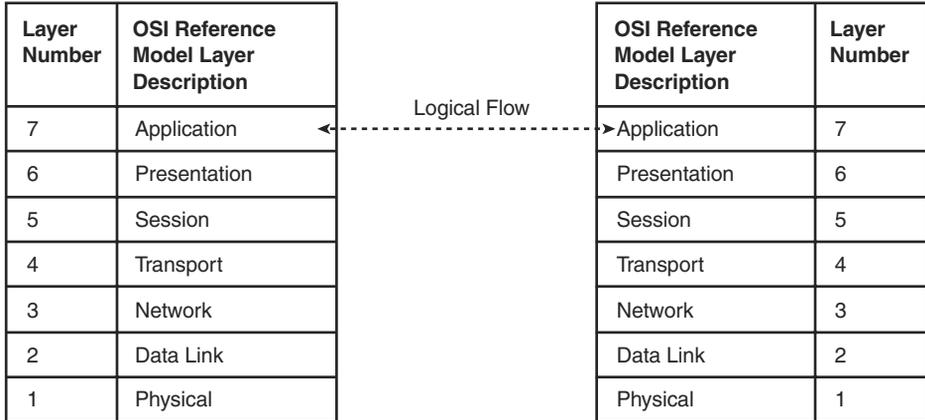
Put Layers 1 and 2 together with the physical wire and you have a complete system capable of reliably providing communications between open computer systems. This is possible only because the clearly defined steps create a logical and well-understood sequence.

What It Looks Like

One of the neat things about having a reference model so well defined is that hardware and software manufacturers can focus on what they are good at, entrusting the remaining functions to someone else. The net effect is that a software developer that makes e-mail software for PC users can focus on e-mail features and functions, without worrying much about the other steps necessary for two users to communicate using e-mail.

From that software developer's perspective, as well as from the user's perspective, the two e-mail software packages appear to communicate directly with each other. This concept is known as *logical adjacency*. Figure 3-7 shows how logical adjacency works. For the sake of example, assume your application software (Layer 7 on the model) is a PC e-mail software package such as Microsoft Outlook.

Figure 3-7 Using the OSI Reference Model to Show Logically Adjacent E-mail



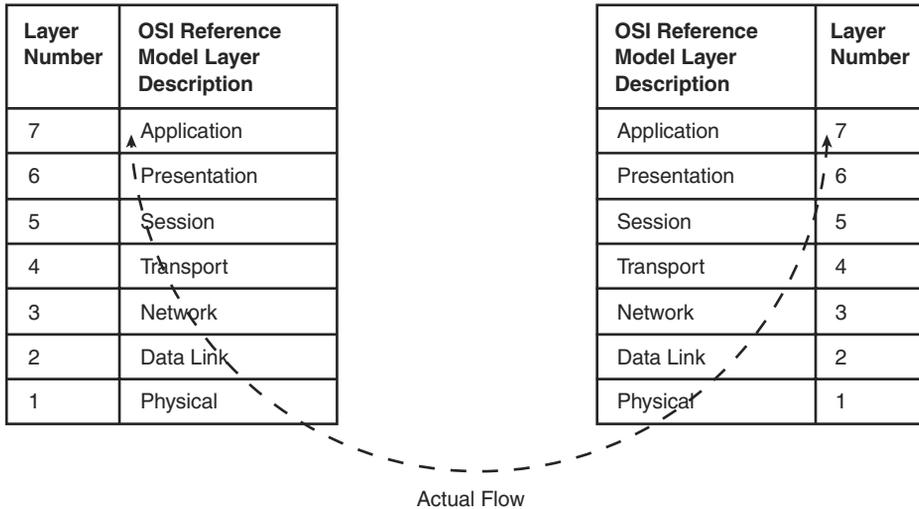
In this illustration, John is sending an e-mail to Jane. Each enjoys the perception that the e-mail is sent directly from the e-mail software on his PC to the e-mail software on her PC. You know that many other steps lie between sending and receiving the e-mail. However, that perception feels real. The two e-mail packages appear to be communicating directly with each other.

What It Really Does

What happens when John sends Jane an e-mail? A whole series of lower-level mechanisms in TCP/IP and Ethernet actually send the e-mail. Jane's PC performs all the same steps as John's PC, only in reverse. This is shown for you in Figure 3-8.

Each layer of functions occurs in the order necessary for everything to work. Anyone who has ever sent an e-mail over the Internet knows just how smoothly it goes! You quite literally take the underlying mechanisms for granted because they work so reliably and consistently.

Figure 3-8 Using the OSI Reference Model to Show What Happens When John Sends Jane an E-mail



Gotcha!

Now that the case for interoperable technologies is built and you have looked at how committees develop technologies standards, you might be wondering if this is real. The standards bodies do create and maintain open technical standards and they do rely on volunteers from for-profit companies in the industry to do so. That much is true. However, those volunteers—and their employers—are not necessarily as altruistic as they might appear.

Interoperable products are also, by their nature, interchangeable. Thus, a LAN *switch*, a device similar to a router that performs packet filtering before forwarding, from Nortel Networks can be replaced by a switch from Cisco Systems without any functionality change impacting users. The switch is literally a commodity—it is functionality in a box that doesn't vary appreciably, regardless of who made it or where you bought it.

What would compel you to buy one manufacturer's product instead of someone else's? Marketing and brand loyalty are two easy answers, but the open-standard technology development process is more subtly exploited to create an advantage.

Two common exploits follow:

- Releasing products in advance of a completed open standard
- Creating products that embrace an open standard, but also include proprietary features outside that standard

These hidden “gotchas” apply to both hardware and software, and can even be found amongst advances made within the TCP/IP protocol suite.

Getting a Jump on Emerging Standards

One way companies can use an open standard to their own advantage is by jump-starting the development of a new open-standard technology. Think about it: You don't want a system that stifles creativity or innovation, but you do want interoperability. The information technology industry has evolved so that most people who volunteer for work within the various standards-settings bodies are contributing on direct behalf of their employer. Thus, they tend to lobby and argue in favor of things that benefit their employer. Often, they argue in favor of things their employer has already developed.

This technique rewards proprietary forays into new technologies, or advances of existing technologies, by creating a small window of opportunity in which that company is the only game in town. Everyone else who agrees to the proposal has to play catch-up!

People who buy products before that product's standard is completed must realize they are buying a nonstandard product. That product anticipates what the standard will be and might be close to the standard when it gets finished—but you can't be sure!

Using Open Standards to Create Proprietary Products

Another approach is to create proprietary technologies through open-standards-setting processes. That sounds like a contradiction, but is a fairly common practice. Companies embrace an open standard, but then add special features that weren't a part of the original standard. Consequently, those features are only a part of that particular manufacturer's product set.

The net effect is that interoperability is okay within the confines of the open standard. Interoperability breaks down when you attempt to use a proprietary feature. Because most consumers don't know where the standard ends and the proprietary extensions begin, symptoms of the mismatch are often viewed as a problem with standards-compliant products, as opposed to a problem with products that go beyond the standard.

Making Sense of the Chaos

The world of open technology standards is far from perfect. In fact, its strength—collaborative development—also creates the potential for some serious traps that can ensnare the unwary buyer! Despite the seemingly altruistic goal of openly collaborating for the greater good, companies are tempted by the profits had by offering proprietary technologies. The net result: Let the buyer beware!

Take time to research when you need to buy any piece of network or computing gear. The newer the technology, the more critical it is to research each manufacturer's products. More importantly, you need to figure out the current state of any open standard that guides the development of those products and see how closely they conform to that standard.

You can tell if what you are buying meets your expectation of interoperability by understanding the standards and asking questions before you buy.

Chapter Summary

Interoperability across vendor platforms is something users take for granted. It's been a long, hard struggle to get to this point! You could argue that it's still not over. The information technology industry has well-defined public standards for virtually everything. These standards, and their ready availability (openness), are what enable products from different companies to interoperate with each other.

These standards, in turn, are built upon a well-understood and necessary sequence of functions. Those functions are kept organized through the use of layered reference models. Different reference models might draw the boundaries between the layers a little differently, but the sequence of steps within those layers remains constant.

Traps await the unwary; some companies do try to recapture some of the lost bounty of proprietary hardware by embellishing open standards. Yet, the era of open computing has begun and shows no signs of ending.

Having seen how open communications works, as well as where TCP/IP fits in the grander scheme of things, focus a little more on this book's subject. TCP/IP, like driving a car, can be thought of as a complicated function that consists of a tremendous array of simple, individual tasks. These tasks are embodied in individual, single-purpose software programs known as *protocols*. Together, protocols enable open communications between dissimilar computers across networks that span both large and small distances. The next chapter takes a much closer look at TCP/IP—the networking protocol that changed the world!

Chapter Review Questions

The following questions reinforce the key concepts in this chapter.

1. What is the IETF?
2. What is the difference between open standards and closed, or proprietary, technologies?
3. What is interoperability?
4. What is the benefit of interoperability?
5. How do you create and maintain the technical standards that enable the Internet and its technologies to be interoperable?
6. What is a reference model and why is it needed?
7. Explain the concept of logical adjacency.



Numerics

32-bit addresses, 123

A

access technologies

broadband, 287

dial-up, 288

configuring, 295, 299–309

modems, 292–293

trunk lines, 289

Acknowledgment Number field (TCP), 90

acknowledgments, 175

active/passive clusters, 140

Advanced tab (Network Setup Wizard), 308–309

always-on connections, 286–287

applications

developing

closed technologies, 34–36

open standards, 22–25, 37

for TCP/IP, 13

functions, categorizing, 25

interoperability, 33–34

port numbers, 100–102, 184

dynamic, 104

private, 104

registered, 104

well-known, 102–103

UDP port numbers, 203–204

web browsers, troubleshooting

TCP/IP, 326–328

ARIN (American Registry for Internet Numbers), whois database, 323

ARPAnet, 8

asymmetric cable services, 294–295

ATM (Asynchronous Transfer Mode), 27

audio, transmitting with UDP, 197

B

bandwidth, 244

of modems, 292–293

base address, 149

Base2 number system, 116, 120–121

counting, 126–128

Base10 number system, 119–120

benefits of subnetting, 152

best path determination, 232–234, 239–240

- cost, 234–235
- hop counting, 241–243
- link-state routing, 245–247

best-effort delivery, 28–30, 189

- latency, 191
- of UDP, 71
- propagation delay, 192

BGP4 (Border Gateway Protocol, version 4), 248**binary number system, 120–121**

- 32-bit addresses, 123
- bits, 122
- converting to decimal, 128
- representing subnetting addresses, 149–150

bits, 88, 122**broadband connections, 287****browsers, troubleshooting TCP/IP, 326–328****bytes, 88, 122****C****calculating best paths, 239–240****carrier-class networks, 259****categorizing**

- network performance requirements, 257
 - reliable delivery*, 26–27
 - timely communications*, 28–30
- software functions, 25

checksum algorithm, 115**checksum field**

- ICMP messages, 213
- IP, 99
- TCP, 91
- UDP, 94

choosing an ISP, 284–285**CIDR (classless inter-domain routing), 134****classful IP addressing, 132–133****classless IP addressing, 132–135****CLI (command-line interface), 179****closed architectures, 8****closed technologies, 34**

- consumer benefits of, 35
- manufacturer's benefits of, 36

coaxial cables, 256**code field (ICMP messages), 213****commands**

- ipconfig, 118–119, 314–316
- netstat
 - e option*, 336–338
 - s option*, 332–335
 - troubleshooting TCP/IP*, 330–331

ping, 321–322, 339–340

tracert, 340–342

whois, 315–320

ARIN database, 323

InterNIC database, 318–320

commercialization of Internet, 9

communication

reliability of, 168

TCP

sessions, 168–171, 177–178

three-way handshakes, 172–174

computers

desktop, 267

laptop, 266

configuring

dial-up connections, 295, 299

Network Setup Wizard, 300–309

NICs, 265–266

*for wire-based network connection,
268–271*

*for wireless network connection,
271–277*

connectionless protocols, 203

consensus-based standards building, 7, 42

consumer benefits of closed technologies, 35

converting

binary to decimal number system, 128

decimal to binary number system, 121

cost metric, 234–235

counting in Base2, 126–128

D

data

receiving with UDP, 201–203

transmitting with UDP, 200–201

data integrity, 14

Data Offset field (TCP), 90

datagrams, 95, 176

decimal masks, 153–155

decimal number system, 119–120

converting to binary, 121

dotted quad format, 122

expressing in binary, 125

dedicated connections, 286–287

default gateways, 119

delimiters, 124

desktop computers, 267

destination applications, 89

destination machines, 89

destination port field

TCP, 89

UDP, 94

**Destination Unreachable errors
(ICMP), 215–216, 335**

developing applications

- for TCP, 13
- closed technologies, 34
 - consumer benefits of, 35*
 - manufacturer's benefits of, 36*
- open standards, 22–23
 - functional abstraction, 23–25*
- open technologies, 37

development of TCP/IP, 8

- ARPAnet, 8
- commercialization of Internet, 9
- NCP, 63

devices

- hubs, 254
- modems, 292–293
- NICs, configuring, 265–277
- routers, 112, 143
 - and IP addresses, 129*
 - functions performed, 227, 230*
 - packet forwarding, 236–238*
 - software-based, 228*
 - subnetting, 151*
- switches, 254

dial-up connections

- asymmetric cable solutions, 294–295
- configuring, 295, 299
 - Network Setup Wizard, 300–309*
- modems, 292–293
- POTS, 288–289

displaying ICMP statistics, 334–335

distance-vector routing, 241–243

DMZ (demilitarized zone), 146

DNS (domain naming system), 128

dotted quad format, 122

dynamic port numbers, 104

E

Echo messages (ICMP), 214, 334

Echo Reply messages (ICMP), 334

EIA/TIA, 43–44

categories of performance, 44

encapsulating IP packets, 111

checksum algorithm, 115

frames, 112

encryption, enabling on wireless networks, 277

error rate, 245

extended network prefix, 147

F

fields

of ICMP messages, 212–213

of TCP header, 89–92

checksum, 91

PSH, 180

of TCP headers, 165–166

of UDP header, 94, 193

checksum, 94

destination port, 94

message length, 94

payload, 95

source port, 94

FIN packets, 183

FIN-ACK packets, 183

firewalls, 143, 308

enabling on wireless networks,
275–277

**FLSM (fixed-length subnet mask-
ing), 158**

forwarding packets, 236–238

fragment offset field (IP), 98

frames, 112

Ethernet, 238

**FTP (File Transfer Protocol),
11, 287**

functional abstraction, 23–25

functional layers of TCP/IP, 77

host-to-host, 78

Internet, 79

network access, 79

process application, 77

functions of TCP/IP, 13–14, 63–64

application tools, 65–66

TCP, 67–70

UDP, 71–72

network tools, IP, 72–75

user tools, 64–65

G-H

**General tab (Network Setup
Wizard), 302**

globally unique addresses, 14

hardware-based routers, 228

header information, 85

IP, 97–99

TCP, 88

fields, 89–92, 165–166, 180

UDP, 94–95

fields, 193

history of TCP/IP, 8, 61

ARPAnet, 8

commercialization of Internet, 9

NCP, 63

hop counting, 241–243

host addresses, 129–131, 141

hostmasters, 159

host-to-host layer, 77–78

hubs, 254

versus switches, 255

I**IANA, 100–102****ICMP (Internet Control Message Protocol), 209–211, 322**

fields, 212–213

message types, 213–214

destination unreachable,
215–216

Echo, 214

Parameter Problem, 219–220

redirect, 217–219

source quench, 216–217

Time Exceeded, 219

Timestamp, 221

ping, 221–222

ping command, 339–340

statistics, viewing with netstat -s
command, 334–335

tracert, 222–223, 340–342

Identifier field (IP), 98**identifying IP addresses on computers, 117–119****IEEE (Institute of Electrical and Electronic Engineers), 44–45****IETF (Internet Engineering Task Force), 46**

RFCs, 47

Internet, 281

access technologies

broadband, 287

dial-up, 288–289

selecting, 286

backbone providers, 282

Internet Header Length field (IP), 97**Internet layer, 77–79****InterNIC (Internet Network Information****Center), 318**

whois database, 318–320

interoperability, 33–34

logical adjacencies, 53

IP (Internet Protocol), 96

functions of, 96

header information, 97–99

packets, 109

checksum algorithm, 115

encapsulation, 111

frames, 112

IP addressing, 116

and routers, 129

bits, 122

classful, 132–133

classless, 134–135

default gateway, 119

delimiters, 124

DNS, 128

- host address, 131
- identifying on computers, 117–119
- network address, 130
- subnetting, 139, 142, 145–146
 - benefits of*, 152
 - binary representation*, 149–150
 - decimal masks*, 153–155
 - FLSM*, 158
 - limitations of*, 152
 - masks*, 131, 153
 - network address*, 147
 - network masks*, 154
 - subnet masks*, 155–157
 - subnetwork address*, 147
 - two-level hierarchy*, 130
 - VLSM*, 158–159

IP suite, 5

ipconfig command,
118–119, 314–316

**ISO (International Organization
for Standards), 49**

**ISPs (Internet service providers),
282**

- access technologies, selecting, 286
- Internet backbone providers, 282
- on-ramp providers, 282
- selecting, 284–285
- Tier 3, 283

L

LANs, 112, 253

- connecting with routers, 230
- Ethernet, 229
- NICs, configuring, 265–277
- routers, functions of, 231–232
- wire-based, 255–256
 - benefits of*, 258–259
 - coaxial cable*, 256
 - disadvantages of*, 259
 - twisted-pair wire*, 256–258
- wireless, 260–261
 - benefits of*, 263
 - disadvantages of*, 263–264

laptop computers, 266

latency, 191

Layer 4 (transport layer)

- best-effort delivery, 189
- TCP, 86–87
 - header information*, 88–92
- UDP, 92, 190–191
 - best-effort delivery*, 191–192
 - header fields*, 193
 - header information*, 94–95
 - port numbers*, 203–204
 - real-time video*, 195
 - receiving data*, 201–203
 - transmitting audio*, 197
 - transmitting data*, 200–201
 - transmitting video*, 194–195

layers of OSI reference model, 49–52**layers of TCP/IP**

functions of, 64

application tools, 65–72

network tools, IP, 72–75

user tools, 64–65

host-to-host, 78

internet, 79

network access, 79

process/application, 77

lightweight protocols, ICMP, 211

destination unreachable messages,
215–216

Echo messages, 214

fields, 212–213

message types, 213–214

Parameter Problem messages,
219–220

ping, 221–222

redirect messages, 217–219

source quench messages, 216–217

Time Exceeded messages, 219

Timestamp messages, 221

tracert, 222–223

limitations

of subnetting, 152

of TCP/IP, 11

link-state routing, best path determination, 244–247**logical adjacencies, 53****logical models, 11**

M

manufacturer's benefits

of closed technologies, 36

of open standards, 56

maximum segment size, 179**message length field (UDP), 94****messages, ICMP, 213–214**

Destination Unreachable, 215–216

Echo, 214

fields, 212–213

Parameter Problem, 219–220

Redirect, 217–219

Source Quench, 216–217

Time Exceeded, 219

Timestamp, 221

milliseconds, 191**modems, 292**

bandwidth, 292–293

multitasking with TCP, 184**multivendor interoperability, 37**

N

NCP (Network Control Protocol), 63

netstat command, troubleshooting

TCP/IP, 330–331

-e option, 336–338

-s option, 332–335

network access layer, 79

network addresses,

129–130, 141, 147

network masks, 154

Network Setup Wizard, 300–302

Advanced tab, 308–309

General tab, 302

Networking tab, 306

Options tab, 303–305

Security tab, 305–306

network tools, IP, 72–75

Networking tab (Network Setup

Wizard), 306

NICs (network interface cards), 253

configuring, 265–266

for wire-based network

connection, 268–271

for wireless network connection,

271–277

NOS (Network Operating System), 15

NTP (Network Time Protocol), 221

number systems, 116

binary, 120–121

bits, 122

counting in, 126–128

decimal, 119–120

expressing in binary, 125

O

on-ramp providers, 282

open standards, 22–23, 37–38

creating proprietary technologies, 57

functional abstraction, 23–25

manufacturer's benefits, 56

reference models, 40–42

Options tab (Network Setup
Wizard), 303–305

origins of TCP/IP, 8

ARPAnet, 8

commercialization of Internet, 9

OS (operating system), 15

OSI reference model, 48–49

layers, 49–52

logical adjacencies, 53

transport layer

TCP, 86–92

UDP, 92–95, 190–197, 200–204

P

packets, 10, 176

FIN, 183

forwarding, 236–238

IP, 109

checksum algorithm, 115

encapsulation, 111

frames, 112

SYN-ACK, 174

TTL, 73

unicast, 337

padding field (IP), 91, 99

Parameter Problem messages

(ICMP), 219–220

PARC (Palo Alto Research Center), 44

payload field

IP, 92, 99

UDP, 95

PBXs, 302

ping, 221–222

ping command, 321–322, 339–340

policy-based routing, 248

port numbers, 69, 100–102, 184

dynamic, 104

private, 104

registered, 104

UDP, 203–204

well-known, 102–103

ports, 89

POTS, 288

asymmetric cable solutions,
294–295

dial-up connections, configuring,
295, 299–309

modems, 292

bandwidth, 292–293

trunk lines, 289

priority handling, 180

private port numbers, 104

process/application layer, 77

Project 802 reference model, 44–46

propagation delay, 192

proprietary technologies, 8, 34

creating through open standards, 57

Protocol field (IP), 98

protocols, 5, 75

PSH field, 180

R

real-time video, 195

receiving data with UDP, 201–203

redirect messages (ICMP), 217–219

reference models, 39–42

consensus-building process, 42

IEEE, 45

OSI reference model, 48–49

- layers, 49–52*
- logical adjacencies, 53*

Project 802 reference model, 44–46

registered port numbers, 104

reliability

- of communications, 168
- of data delivery, 26–27, 86, 163
 - TCP, 72*

resequencing, 14

Reserved field (TCP), 90

Reset Connections errors (ICMP), 335

RFCs (Requests for Comments), 47

routers, 112, 143

- and IP addresses, 129
- functions of, 227, 230
 - forwarding packets, 236–238*
 - in single-router networks, 231–232*
- software-based, 228
- subnetting, 151

routing, policy-based, 248

routing protocols, best path determination, 232–234, 239

- cost, 234–235
- distance-vector protocols, 241–243
- link-state protocols, 244–247

routing tables, 234

RTT (round trip time), 181

S

Security tab (Network Setup Wizard), 305–306

segmentation and reassembly, 14

segments, 10, 175–178

- maximum size of, 179

selecting

- access technologies, 286

- ISPs, 284–285

sequence number field (TCP), 89

sequence numbers, 169–170, 174–175

sessions, 164

- TCP, 168–171, 177–178

- timed out packets, 182*

shared networks, 143

single-router networks, 231–232

sliding windows, 91, 178

sockets, 101, 185

software

- closed technologies, 34

- consumer benefits of, 35*

- manufacturer's benefits of, 36*

- developing, open standards, 22–23

- functional abstraction, 23–25*

- functions, categorizing, 25
- interoperability, 33–34
- open technologies, 37
- software-based routers, 228**
- source machines, 89**
- source port field**
 - TCP, 89
 - UDP, 94
- Source Quench errors (ICMP), 216–217, 334**
- sourceIP address field (IP), 99**
- standards, 38**
 - consensus-building process, 42
 - EIA/TIA, 43–44
 - categories of performance, 44*
 - IEEE, 44–45
 - IETF, 46–47
 - open standards, manufacturer’s benefits of, 56
 - reference models, 39–42
 - IEEE reference model, 45*
 - OSI reference model, 48–53*
- subnetting, 139, 142, 145–146**
 - benefits of, 152
 - binary representation, 149–150
 - decimal masks, 153–155
 - extended network prefix, 147
 - limitations of, 152
 - masks, 153
 - network address, 147
 - network masks, 154
 - subnet masks, 131, 155–157
 - FLSM, 158*
 - VLSM, 158–159*
 - subnetwork address, 147
- subnetworks, 132, 147**
- switches, 254**
 - versus hubs, 255
- SYN-ACK packets, 174**
- system messages, ICMP, 209–211**
 - Destination Unreachable, 215–216
 - Echo, 214
 - fields, 212–213
 - message types, 213–214
 - Parameter Problem, 219–220
 - Redirect, 217–219
 - Source Quench, 216–217
 - Time Exceeded, 219
 - Timestamp, 221
- system-level processes, 103**

T

TCP (Transmission Control Protocol), 67–70, 86

- header fields, 165–166
- multitasking, 184
- port numbers, 69, 100–102, 184
 - dynamic, 104*
 - private, 104*

- registered, 104*
 - well-known, 102–103*
 - reliability of, 168
 - segments, 175–178
 - maximum size of, 179*
 - sequence numbers,
 - 169–170, 174–175
 - sessions, 165, 168–171, 177–178
 - timed out packets, 182*
 - sliding windows, 91, 178
 - three-way handshakes, 172–174
 - versus UDP, 95
 - window size, 178
- three-way handshakes, 172–174**
- Tier 3 ISPs, 283**
- Time Exceeded messages (ICMP), 219**
- timed out packets, 181**
- timely communications, 28–30**
- Timestamp messages (ICMP), 221**
- tools. *See* utilities**
- ToS field (IP), 97**
- Total Length (IP), 97**
- tracert, 222–223**
- tracert command, 340–342**
- transmission media, 255**
- coaxial cable, 256
 - twisted-pair wire, 256–258
 - wireless, 260–261
 - benefits of, 263*
 - disadvantages of, 263–264*
- transmitting data with UDP, 200–201**
- transport layer**
- best-effort delivery, 189
 - TCP, 86–87
 - header information, 88–92*
 - UDP, 92, 190–192
 - header fields, 193*
 - header information, 94–95*
 - port numbers, 203–204*
 - real-time video, 195*
 - receiving data, 201–203*
 - transmitting audio, 197*
 - transmitting data, 200–201*
 - transmitting video, 194–195*
- troubleshooting**
- IP addresses, ping command, 339–340
 - ipconfig command, 314–316
 - netstat utility, 330–331
 - e option, 336–338*
 - s option, 332–333, 335*
 - ping command, 321–322
 - tracert command, 340–342
 - web browsers, 326–328
 - whois command, 317–320
 - ARIN database, 323*
 - InterNIC database, 318–320*

trunk lines, 289

TTL (Time-To-Live), 73

TTL field (IP), 98

twisted-pair wire, 256–258

**type indicator field
(ICMP messages), 212**

U

ubiquitous nature of TCP/IP, 21

**UDP (User Datagram Protocol),
71–72, 92, 190**

best-effort delivery, 191

latency, 191

propagation delay, 192

header fields, 193

header information, 94–95

port numbers, 203–204

receiving data, 201–203

transmitting audio, 197

transmitting data, 200–201

transmitting video, 194–195

real-time, 195

versus TCP, 95

unicast packets, 337

universality of TCP/IP, 21

urgent packets, 170

priority handling, 180

USB (Universal Serial Bus) cable, 262

utilities, 64

application tools, 65–66

TCP, 67–70

UDP, 71–72

ipconfig, 314–316

netstat

-e option, 336–338

-s option, 332–335

troubleshooting TCP/IP, 330–331

network tools, IP, 72–75

ping, 321–322

ping command, 339–340

tracert command, 340–342

whois, 317–320

ARIN database, 323

InterNIC database, 318–320

V

versatility of TCP/IP, 20

Version field (IP), 97

video

real-time, 195

transmitting with UDP, 194–195

viewing ICMP statistics, 334–335

**VLSM (variable-length subnet
masking), 158–159**

VoIP (Voice over IP), 28

W-Z

WANs (wide-area networks), 229

routers, 112

web browsers, troubleshooting

TCP/IP, 326–328

well-known port numbers,

102–103, 185

whois command, 315–320

ARIN database, 323

InterNIC database, 318–320

Window size field (TCP), 91, 178

wire-based networks, 255–256

benefits of, 258–259

coaxial cable, 256

connecting to, 268–271

disadvantages of, 259

twisted-pair wire, 256–258

wireless networks, 260–261

benefits of, 263

connecting to, 271–273

disadvantages of, 263–264

firewalls, enabling, 275–277

wizards, 269

zero suppression, 121