

Joe Casad

Sams **Teach Yourself**
TCP/IP

in **24**
Hours

SIXTH EDITION

SAMS

800 East 96th Street, Indianapolis, Indiana 46240 USA

Sams Teach Yourself TCP/IP in 24 Hours, Sixth Edition

Copyright © 2017 by Pearson Education, Inc.

All rights reserved. Printed in the United States of America. This publication is protected by copyright, and permission must be obtained from the publisher prior to any prohibited reproduction, storage in a retrieval system, or transmission in any form or by any means, electronic, mechanical, photocopying, recording, or likewise. For information regarding permissions, request forms, and the appropriate contacts within the Pearson Education Global Rights & Permissions Department, please visit www.pearsoned.com/permissions/. No patent liability is assumed with respect to the use of the information contained herein.

Although every precaution has been taken in the preparation of this book, the publisher and author assume no responsibility for errors or omissions. Nor is any liability assumed for damages resulting from the use of the information contained herein.

ISBN-13: 978-0-672-33789-5

ISBN-10: 0-672-33789-4

Library of Congress Control Number: 2016920083

1 17

Trademarks

All terms mentioned in this book that are known to be trademarks or service marks have been appropriately capitalized. Sams Publishing 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.

Warning and Disclaimer

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 provided is on an "as is" basis. The author and the publisher 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.

Special Sales

For information about buying this title in bulk quantities, or for special sales opportunities (which may include electronic versions; custom cover designs; and content particular to your business, training goals, marketing focus, or branding interests), please contact our corporate sales department at corpsales@pearsoned.com or (800) 382-3419.

For government sales inquiries, please contact

governmentsales@pearsoned.com

For questions about sales outside the U.S., please contact

intlcs@pearson.com

Editor-in-Chief

Greg Wiegand

Executive Editor

Laura Lewin

Development Editor

Michael Thurston

Managing Editor

Sandra Schroeder

Project Editor

Mandie Frank

Copy Editor

Abigail Bass

Indexer

Tim Wright

Proofreader

Larry Sulky

Technical Editors

Ronald McFarland

Jon Snader

Eric Spielman

Editorial Assistant

Olivia Basegio

Designer

Chuti Prasertsith

Compositor

codeMantra

Contents at a Glance

Part I: TCP/IP Basics	1
HOUR 1 What Is TCP/IP?	3
2 How TCP/IP Works	19
Part II: The TCP/IP Protocol System	31
HOUR 3 The Network Access Layer	33
4 The Internet Layer	45
5 Subnetting and CIDR	69
6 The Transport Layer	85
7 The Application Layer	109
Part III: Networking with TCP/IP	119
HOUR 8 Routing	121
9 Getting Connected	143
10 Name Resolution	171
11 TCP/IP Security	197
12 Configuration	223
13 IPv6: The Next Generation	247
Part IV: Tools and Service	265
HOUR 14 Classic Tools	267
15 Classic Services	297
Part V: The Internet	315
HOUR 16 The Internet: A Closer Look	317
17 HTTP, HTML, and the World Wide Web	329
18 Web Services	359
19 Encryption, Tracking, and Privacy	379

Part VI: TCP/IP at Work	409
HOUR 20 Email	411
21 Streaming and Casting	431
22 Living in the Cloud	449
23 Internet of Things	465
24 Implementing a TCP/IP Network: 7 Days in the Life of a Sys Admin	477
Appendixes	
A Answers to Quizzes and Exercises	491
B Sources	503
Index	505

Table of Contents

Part I: TCP/IP Basics	1
HOUR 1: What Is TCP/IP?	3
Networks and Protocols	4
The Development of TCP/IP	6
TCP/IP Features	8
Standards Organizations and RFCs	13
Summary	15
Q&A	15
Workshop	15
Key Terms	16
HOUR 2: How TCP/IP Works	19
The TCP/IP Protocol System	20
TCP/IP and the OSI Model	22
Data Packages	24
A Quick Look at TCP/IP Networking	25
Summary	28
Q&A	28
Workshop	28
Key Terms	29
Part II: The TCP/IP Protocol System	31
HOUR 3: The Network Access Layer	33
Protocols and Hardware	33
The Network Access Layer and the OSI Model	34
Network Architecture	35
Physical Addressing	38

Ethernet	39
Anatomy of an Ethernet Frame	40
Summary	42
Q&A	42
Workshop	42
Key Terms	43
HOUR 4: The Internet Layer	45
IP Addresses: A Little Context	46
Addressing and Delivering	46
Internet Protocol	49
Address Resolution Protocol	61
Reverse ARP	62
Internet Control Message Protocol	63
Summary	64
Q&A	64
Workshop	65
Key Terms	66
HOUR 5: Subnetting and CIDR	69
Subnets	69
Dividing the Network	70
The Old Way: Subnet Mask	72
The New Way: CIDR	79
Summary	81
Q&A	81
Workshop	82
Key Terms	83
HOUR 6: The Transport Layer	85
Introducing the Transport Layer	86
Transport Layer Concepts	87
Understanding TCP and UDP	92
Firewalls and Ports	102
Summary	103

Q&A	103
Workshop	104
Key Terms	105
HOUR 7: The Application Layer	109
What Is the Application Layer?	109
The TCP/IP Application Layer and OSI	110
Network Services	111
APIs and the Application Layer	115
TCP/IP Utilities	115
Summary	116
Q&A	116
Workshop	117
Key Terms	118
Part III: Networking with TCP/IP	119
HOUR 8: Routing	121
Routing in TCP/IP	121
Routing on Complex Networks	133
Examining Interior Routers	134
Exterior Routers: BGP	136
Classless Routing	137
Higher in the Stack	138
Summary	139
Q&A	139
Workshop	140
Key Terms	140
HOUR 9: Getting Connected	143
Cable Broadband	144
Digital Subscriber Line	145
Wide Area Networks	146
Wireless Networking	148
Dial-Up Networking	157

Connectivity Devices	161
Switching Versus Routing	165
Summary	166
Q&A	167
Workshop	167
Key Terms	168
HOUR 10: Name Resolution	171
What Is Name Resolution?	171
Name Resolution Using Hosts Files	173
DNS Name Resolution	175
Registering a Domain	181
Name Server Types	182
Dynamic DNS	192
NetBIOS Name Resolution	193
Summary	194
Q&A	194
Workshop	195
Key Terms	196
HOUR 11: TCP/IP Security	197
What Is a Firewall?	197
Attack Techniques	205
What Do Intruders Want?	206
Summary	219
Q&A	220
Workshop	220
Key Terms	221
HOUR 12: Configuration	223
Getting on the Network	223
The Case for Server-Supplied IP Addresses	224
What Is DHCP?	225
How DHCP Works	226

DHCP Server Configuration	229
Network Address Translation	230
Zero Configuration	232
Configuring TCP/IP	235
Summary	244
Q&A	244
Workshop	244
Key Terms	246
HOUR 13: IPv6: The Next Generation	247
Why a New IP?	248
IPv6 Header Format	249
IPv6 Addressing	253
Subnetting	254
Multicasting	255
Link Local	255
Neighbor Discovery	256
Autoconfiguration	256
IPv6 and Quality of Service	257
IPv6 with IPv4	258
IPv6 Tunnels	258
Summary	261
Q&A	262
Workshop	262
Key Terms	263
Part IV: Tools and Service	265
HOUR 14: Classic Tools	267
Connectivity Problems	268
Protocol Dysfunction and Misconfiguration	268
Line Problems	274
Name Resolution Problems	274
Network Performance Problems	275

Telnet	280
Berkeley Remote Utilities	283
Secure Shell	284
Network Management	285
Summary	292
Q&A	292
Workshop	293
Key Terms	294
HOURL 15: Classic Services	297
HTTP	298
Email	298
FTP	299
Trivial File Transfer Protocol	303
File and Print Services	303
Lightweight Directory Access Protocol	306
Remote Control	309
Summary	311
Q&A	311
Workshop	311
Key Terms	312
Part V: The Internet	315
HOURL 16: The Internet: A Closer Look	317
How the Internet Looks	317
What Happens on the Internet	320
URIs and URLs	322
Summary	325
Q&A	325
Workshop	326
Key Terms	326

HOURL 17: HTTP, HTML, and the World Wide Web	329
What Is the World Wide Web?	329
Understanding HTML	332
Cascading Style Sheets	337
Understanding HTTP	338
Scripting	341
Web Browsers	344
The Semantic Web	348
XHTML	350
HTML5	351
Summary	356
Q&A	356
Workshop	356
Key Terms	357
HOURL 18: Web Services	359
Content Management Systems	360
Social Networking	361
Peer-to-Peer	364
Understanding Web Services	365
XML	368
SOAP	369
WSDL	370
Web Service Stacks	371
REST	371
E-Commerce	374
Summary	377
Q&A	377
Workshop	377
Key Terms	378

HOURL 19: Encryption, Tracking, and Privacy	379
Encryption and Secrecy	380
Tracking	395
Anonymity Networks	403
Summary	404
Q&A	405
Workshop	405
Key Terms	406
Part VI: TCP/IP at Work	409
HOURL 20: Email	411
What Is Email?	411
Email Format	412
How Email Works	413
Simple Mail Transfer Protocol	416
Retrieving the Mail	418
Email Clients	420
Webmail	422
Spam	423
Phishing	426
Summary	427
Q&A	427
Workshop	427
Key Terms	428
HOURL 21: Streaming and Casting	431
The Streaming Problem	431
A Brief Introduction to Multimedia Files	432
Real-Time Transport Protocol—Streaming Over UDP	435
RTMP—Streaming Over TCP	438
SCTP and DCCP—Replacing the Transport Layer	439
Streaming Over HTTP	440
HTML5 and Multimedia	442

Podcasting	442
Voice over IP	443
Summary	445
Q&A	445
Workshop	446
Key Terms	446
HOUR 22: Living in the Cloud	449
What Is the Cloud?	449
Private Clouds	460
Future of Computing	461
Summary	462
Q&A	462
Workshop	462
Key Terms	463
HOUR 23: Internet of Things	465
What Is the Internet of Things?	465
IoT Platforms	467
Up Close: MQTT	470
RFID	472
Summary	474
Q&A	474
Workshop	474
Key Terms	475
HOUR 24: Implementing a TCP/IP Network: 7 Days in the Life of a Sys Admin	477
A Brief History of Hypothetical, Inc.	477
7 Days in the Life of Maurice	478
Summary	487
Q&A	487
Workshop	488
Key Terms	488

APPENDIXES

APPENDIX A: Answers to Quizzes and Exercises	491
Hour 1: What Is TCP/IP?	491
Hour 2: How TCP/IP Works	491
Hour 3: The Network Access Layer	492
Hour 4: The Internet Layer	493
Hour 5: Subnetting and CIDR	494
Hour 6: The Transport Layer	494
Hour 7: The Application Layer	495
Hour 8: Routing	495
Hour 9: Getting Connected	496
Hour 10: Name Resolution	496
Hour 11: TCP/IP Security	496
Hour 12: Configuration	497
Hour 13: IPv6: The Next Generation	497
Hour 14: Classic Tools	497
Hour 15: Classic Services	498
Hour 16: The Internet: A Closer Look	498
Hour 17: HTTP, HTML, and the World Wide Web	498
Hour 18: Web Services	499
Hour 19: Encryption, Tracking, and Privacy	500
Hour 20: Email	500
Hour 21: Streaming and Casting	501
Hour 22: Living in the Cloud	501
Hour 23: Internet of Things	502
Hour 24: Implementing a TCP/IP Network: 7 Days in the Life of a Sys Admin	502
APPENDIX B: Sources	503
Index	505

About the Author

Joe Casad is an engineer, author, and editor who has written widely on computer networking and system administration. He has written or cowritten 12 books on computers and networking. He currently serves as editor-in-chief of *Linux Pro Magazine* and *ADMIN Magazine*. In a past life, he was the editor-in-chief of *C/C++ Users Journal* and the technical editor of *Sysadmin Magazine*.

Dedication

For Susan

Acknowledgments

Thanks to Laura Lewin, Olivia Basegio, Michael Thurston, Ronald McFarland, Jon Snader, Eric Spielman, Mandie Frank, Dhaya Karunanidhi, and Abby Manheim for their help with envisioning and creating this book. I also want to acknowledge the following individuals for their contributions to previous editions of *Sams Teach Yourself TCP/IP in 24 Hours*: Bob Willsey, Sudha Putnam, Walter Glenn, Art Hammond, Jane Brownlow, Jeff Koch, Mark Renfrow, Vicki Harding, Mark Cierzniak, Marc Charney, Jenny Watson, Betsy Harris, and Trina MacDonald. Thanks to Xander, Mattie, and Bridget for staying close in the storms and not wandering too far away in the sunshine. Thanks to my life partner Susan Rieger for venturing through canyons and over mountaintops with a guy who's still working on reading the map. And thanks with fond gratitude to the production department for bringing form and elegance to an inglorious collection of cryptic pencil sketches.

We Want to Hear from You!

As the reader of this book, *you* are our most important critic and commentator. We value your opinion and want to know what we're doing right, what we could do better, what areas you'd like to see us publish in, and any other words of wisdom you're willing to pass our way.

We welcome your comments. You can email or write to let us know what you did or didn't like about this book—as well as what we can do to make our books better.

Please note that we cannot help you with technical problems related to the topic of this book.

When you write, please be sure to include this book's title and author as well as your name and email address. We will carefully review your comments and share them with the author and editors who worked on the book.

Email: feedback@sampublishing.com

Mail: Sams Publishing
ATTN: Reader Feedback
800 East 96th Street
Indianapolis, IN 46240 USA

Reader Services

Register your copy of *Sams Teach Yourself TCP/IP in 24 Hours, Sixth Edition* at informit.com for convenient access to downloads, updates, and corrections as they become available. To start the registration process, go to informit.com/register and log in or create an account*. Enter the product ISBN, 9780672337895, and click Submit. Once the process is complete, you will find any available bonus content under Registered Products.

*Be sure to check the box that you would like to hear from us in order to receive exclusive discounts on future editions of this product.

HOUR 2

How TCP/IP Works

What You'll Learn in This Hour:

- ▶ TCP/IP protocol system
- ▶ The OSI model
- ▶ Data packages
- ▶ How TCP/IP protocols interact

TCP/IP is a system (or suite) of protocols, and a protocol is a system of rules and procedures. For the most part, the hardware and software of the communicating computers carry out the rules of TCP/IP communications—the user does not have to get involved with the details. Still, a working knowledge of TCP/IP is essential if you want to navigate through the configuration and troubleshoot problems you'll face with TCP/IP networks.

This hour describes the TCP/IP protocol system and shows how the components of TCP/IP work together to send and receive data across the network.

At the completion of this hour, you will be able to

- ▶ Describe the layers of the TCP/IP protocol system and the purpose of each layer
- ▶ Describe the layers of the OSI model and explain how the OSI layers relate to TCP/IP
- ▶ Explain TCP/IP protocol headers and how data is enclosed with header information at each layer of the protocol stack
- ▶ Name the data package at each layer of the TCP/IP stack
- ▶ Discuss the TCP, UDP, and IP protocols and how they work together to provide TCP/IP functionality

The TCP/IP Protocol System

Before looking at the elements of TCP/IP, it is best to begin with a brief review of the responsibilities of a protocol system.

A protocol system such as TCP/IP must be responsible for the following tasks:

- ▶ Dividing messages into manageable chunks of data that will pass efficiently through the transmission medium.
- ▶ Interfacing with the network adapter hardware.
- ▶ Addressing: The sending computer must be capable of targeting data to a receiving computer. The receiving computer must be capable of recognizing a message that it is supposed to receive.
- ▶ Routing data to the subnet of the destination computer, even if the source subnet and the destination subnet are dissimilar physical networks.
- ▶ Performing error control, flow control, and acknowledgment: For reliable communication, the sending and receiving computers must be able to identify and correct faulty transmissions and control the flow of data.
- ▶ Accepting data from an application and passing it to the network.
- ▶ Receiving data from the network and passing it to an application.

To accomplish the preceding tasks, the creators of TCP/IP settled on a modular design. The TCP/IP protocol system is divided into separate components that theoretically function independently from one another. Each component is responsible for a piece of the communication process.

The advantage of this modular design is that it lets vendors easily adapt the protocol software to specific hardware and operating systems. For instance, the Network Access layer (as you learn in Hour 3, “The Network Access Layer”) includes functions relating to the specification and design of the physical network. Because of TCP/IP’s modular design, a vendor such as Microsoft does not have to build a completely different software package for TCP/IP on an optical-fiber network (as opposed to TCP/IP on an ordinary ethernet network). The upper layers are not affected by the different physical architecture; only the Network Access layer must change.

The TCP/IP protocol system is subdivided into layered components, each of which performs specific duties (see Figure 2.1). This model, or **stack**, comes from the early days of TCP/IP, and it is sometimes called the TCP/IP model. The official TCP/IP protocol layers and their functions are described in the following list. Compare the functions in the list with the responsibilities listed earlier in this section, and you’ll see how the responsibilities of the protocol system are distributed among the layers.

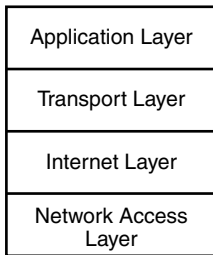
BY THE WAY

Many Models

The four-layer model shown in Figure 2.1 is a common model for describing TCP/IP networking, but it isn't the only model. The ARPANet model, for instance, as described in RFC 871, describes three layers: the Network Interface layer, the Host-to-Host layer, and the Process-Level/Applications layer. Other descriptions of TCP/IP call for a five-layer model, with Physical and Data Link layers in place of the Network Access layer (to match OSI). Still other models might exclude either the Network Access or the Application layer, which are less uniform and harder to define than the intermediate layers.

The names of the layers also vary. The ARPANet layer names still appear in some discussions of TCP/IP, and the Internet layer is sometimes called the Internetwork layer or the Network layer.

This book uses the four-layer model, with names shown in Figure 2.1.

**FIGURE 2.1**

The TCP/IP model's protocol layers.

- ▶ **Network Access layer:** Provides an interface with the physical network. Formats the data for the transmission medium and addresses data for the subnet based on physical hardware addresses. Provides error control for data delivered on the physical network.
- ▶ **Internet layer:** Provides logical, hardware-independent addressing so that data can pass among subnets with different physical architectures. Provides routing to reduce traffic and support delivery across the internetwork. (The term **internetwork** refers to an interconnected, greater network of local area networks (LANs), such as what you find in a large company or on the Internet.) Relates physical addresses (used at the Network Access layer) to logical addresses.
- ▶ **Transport layer:** Provides flow-control, error-control, and acknowledgment services for the internetwork. Serves as an interface for network applications.
- ▶ **Application layer:** Provides applications for network troubleshooting, file transfer, remote control, and Internet activities. Also supports the network application programming interfaces (APIs) that enable programs written for a particular operating environment to access the network.

Later hours provide more detailed descriptions of the activities at each of these TCP/IP protocol layers.

When the TCP/IP protocol software prepares a piece of data for transmission across the network, each layer on the sending machine adds a layer of information to the data that is relevant to the corresponding layer on the receiving machine. For instance, the Internet layer of the computer sending the data adds a header with some information that is significant to the Internet layer of the computer receiving the message. This process is sometimes referred to as encapsulation. At the receiving end these headers are removed as the data is passed up the protocol stack.

BY THE WAY

Layers

The term *layer* is used throughout the computer industry for protocol component levels such as the ones shown in Figure 2.1. Header information is applied in layers to the data as it passes through the components of the protocol stack. (You'll learn more about this later in this hour.) When it comes to the components themselves, however, the term *layer* is somewhat metaphorical.

Diagrams such as Figure 2.1 are meant to show that the data passes across a series of interfaces. As long as the interfaces are maintained, the processes within one component are not affected by the processes in other components. If you turned Figure 2.1 sideways, it would look more like an assembly line, and this is also a useful analogy for the relationship of the protocol components. The data proceeds through a series of steps in the line and, as long as it arrives at each step as specified, the components can operate independently.

TCP/IP and the OSI Model

The networking industry has a standard seven-layer model for network protocol architecture called the Open Systems Interconnection (OSI) model. The OSI model represents an effort by the International Organization for Standardization (ISO), an international standards organization, to standardize the design of network protocol systems to promote interconnectivity and open access to protocol standards for software developers.

TCP/IP was already on the path of development when the OSI standard architecture appeared and, strictly speaking, TCP/IP does not conform to the OSI model. However, the two models did have similar goals, and enough interaction occurred among the designers of these standards that they emerged with a certain compatibility. The OSI model has been very influential in the growth and development of protocol implementations, and it is quite common to see the OSI terminology applied to TCP/IP.

Figure 2.2 shows the relationship between the four-layer TCP/IP standard and the seven-layer OSI model. Note that the OSI model divides the duties of the Application layer into three layers: Application, Presentation, and Session. OSI splits the activities of the Network Access layer into a Data Link layer and a Physical layer. This increased subdivision adds some complexity, but it also adds flexibility for developers by targeting the protocol layers to more specific services. In particular, the division at the lower level into the Data Link and Physical layers separates

the functions related to organizing communication from the functions related to accessing the communication medium. The three upper OSI layers offer a greater variety of alternatives for an application to interface with the protocol stack.

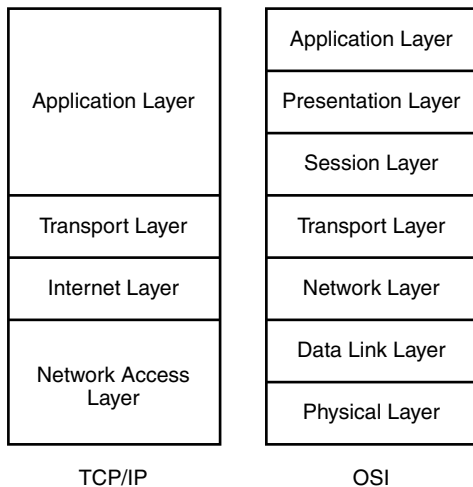


FIGURE 2.2

The seven-layer OSI model.

The seven layers of the OSI model are as follows:

- ▶ **Physical layer:** Converts the data into the stream of electrical or analog pulses that will actually cross the transmission medium and oversees the transmission of the data
- ▶ **Data Link layer:** Provides an interface with the network adapter; maintains logical links for the subnet
- ▶ **Network layer:** Supports logical addressing and routing
- ▶ **Transport layer:** Provides error control and flow control for the internetwork
- ▶ **Session layer:** Establishes sessions between communicating applications on the communicating computers
- ▶ **Presentation layer:** Translates data to a standard format; manages encryption and data compression
- ▶ **Application layer:** Provides a network interface for applications; supports network applications for file transfer, communications, and so forth

It is important to remember that the TCP/IP model and the OSI model are standards, not implementations. Real-world implementations of TCP/IP do not always map cleanly to the models shown in Figures 2.1 and 2.2, and the perfect correspondence depicted in Figure 2.2 is also a matter of some discussion within the industry.

Notice that the OSI and TCP/IP models are most similar at the important Transport and Internet (called Network in OSI) layers. These layers include the most identifiable and distinguishing components of the protocol system, and it is no coincidence that protocol systems are sometimes named for their Transport and Network layer protocols. As you will learn later in this book, the TCP/IP protocol suite is named for TCP, a Transport layer protocol, and IP, an Internet/Network layer protocol.

Data Packages

The important thing to remember about the TCP/IP protocol stack is that each layer plays a role in the overall communication process. Each layer invokes services that are necessary for that layer to perform its role. As an outgoing transmission passes down through the stack, each layer includes a bundle of relevant information called a **header** along with the actual data. The little data package containing the header and the data then becomes the data that is repackaged at the next lower level with the next lower layer's header. This process is shown in Figure 2.3. The reverse process occurs when data is received on the destination computer. As the data moves up through the stack, each layer unpacks the corresponding header and uses the information.

As the data moves down through the stack, the effect is a little like the nested Russian wooden dolls you might have seen; the innermost doll is enclosed in another doll, which is then enclosed in another doll, and so on. At the receiving end, the data packages are unpacked, one by one, as the data climbs back up the protocol stack. The Internet layer on the receiving machine uses the information in the Internet layer header. The Transport layer uses the information in the Transport layer header. At each layer, the package of data takes a form that provides the necessary information to the corresponding layer on the receiving machine. Because each layer is responsible for different functions, the form of the basic data package is very different at each layer.

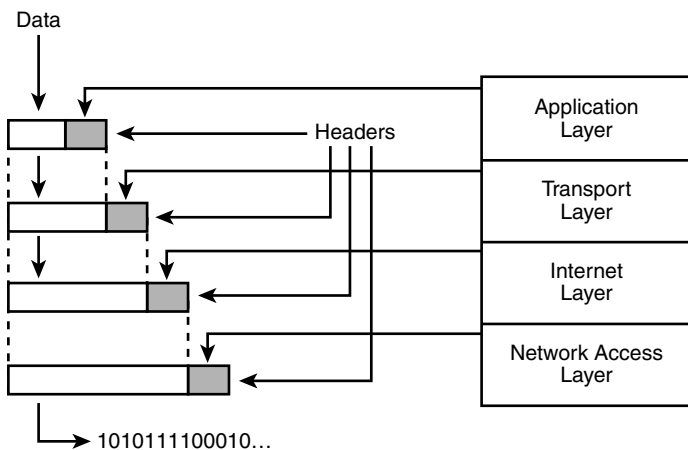


FIGURE 2.3
At each layer, the data is repackaged with that layer's header.

BY THE WAY

Transporting Dolls

The networking industry has as many analogies as it has acronyms, and the Russian doll analogy, like any of the others, illustrates a point, but must not be taken too far. It is worth noting that on a physical network such as ethernet, the data is typically broken into smaller units at the Network Access layer. A more accurate analogy would call for this lowest layer to break the concentric doll system into smaller pieces, encapsulate those pieces into tinier dolls, and then grind those tiny dolls into a pattern of 1s and 0s. The 1s and 0s are received, reconstituted into tiny dolls, and rebuilt into the concentric doll system. The complexity of this scenario causes many to eschew the otherwise-promising analogy of the dolls.

The data packet looks different at each layer, and at each layer it goes by a different name.

The names for the data packages created at each layer are as follows:

- ▶ The data package created at the Application layer is called a **message**.
- ▶ The data package created at the Transport layer, which encapsulates the Application layer message, is called a **segment** if it comes from the Transport layer's TCP protocol. If the data package comes from the Transport layer's **User Datagram Protocol (UDP)** protocol, it is called a **datagram**.
- ▶ The data package at the Internet layer, which encapsulates the Transport layer segment, is called a datagram.
- ▶ The data package at the Network Access layer, which encapsulates and may subdivide the datagram, is called a **frame**. This frame is then turned into a bitstream at the lowest sub-layer of the Network Access layer.

To be honest, people don't always use these different protocol package names anymore; the word "packet" has become a popular (if imprecise) shorthand for describing a data package at any protocol level, but it is still worthwhile to consider that the different protocol packages have different names because they are actually quite different. Each layer has a different purpose, and each header contains different information. You learn more about the data packages for each layer in later hours.

A Quick Look at TCP/IP Networking

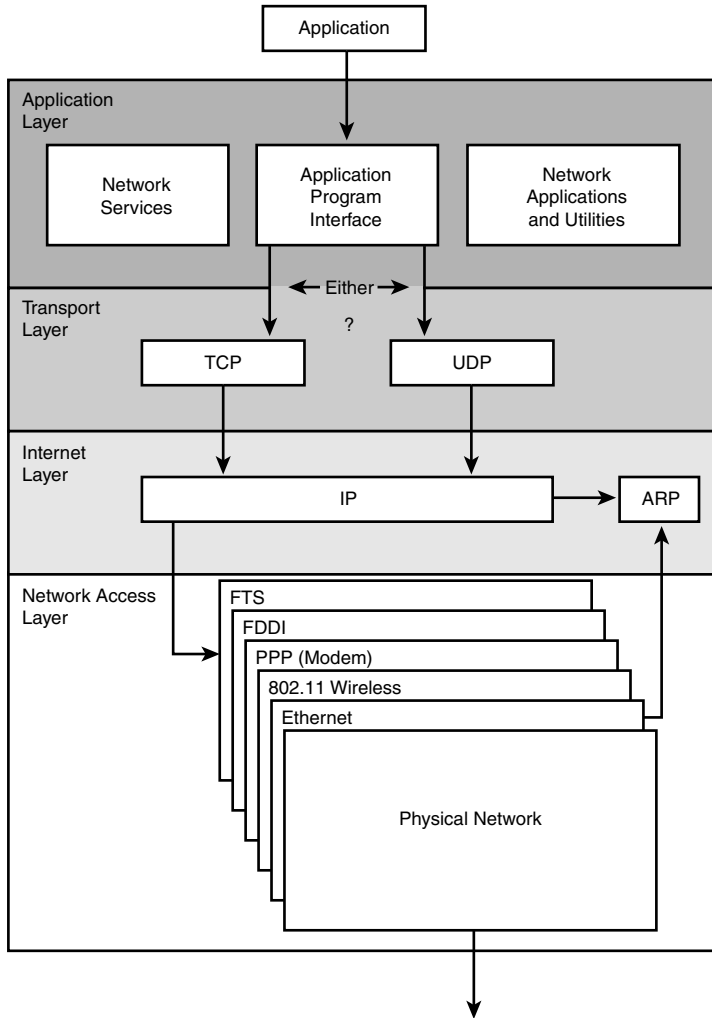
The practice of describing protocol systems in terms of their layers is widespread and nearly universal. The layering system does provide insights into the protocol system, and it's impossible to describe TCP/IP without first introducing its layered architecture. However, focusing solely on protocol layers also creates some limitations.

First, talking about protocol layers rather than protocols introduces additional abstraction to a subject that is already excruciatingly abstract. Second, itemizing the various protocols as subheads within the greater topic of a protocol layer can give the false impression that all protocols are of equal importance. In fact, though every protocol has a role to play, most of the functionality of the TCP/IP suite can be described in terms of only a few of its most important protocols. It is sometimes useful to view these important protocols in the foreground, against the backdrop of the layering system described earlier in this hour.

Figure 2.4 describes the basic TCP/IP protocol networking system. Of course, there are additional protocols and services in the complete package, but Figure 2.4 shows most of what is going on.

The basic scenario is as follows:

1. Data passes from a protocol, network service, or application programming interface (API) operating at the Application layer through a TCP or UDP port to either of the two Transport layer protocols (TCP or UDP). Programs can access the network through either TCP or UDP, depending on the program's requirements:
 - ▶ **TCP** is a connection-oriented protocol. As you learn in Hour 6, "The Transport Layer," connection-oriented protocols provide more sophisticated flow control and error control than connectionless protocols. TCP goes to great effort to guarantee the delivery of the data. TCP is more reliable than UDP, but the additional error checking and flow control mean that TCP is slower than UDP.
 - ▶ **UDP** is a connectionless protocol. It is faster than TCP, but it is not as reliable. UDP offloads more of the error control responsibilities to the application.
2. The data segment passes to the Internet level, where the **IP** protocol provides logical-addressing information and encloses the data into a datagram.
3. The IP datagram enters the Network Access layer, where it passes to software components designed to interface with the physical network. The Network Access layer creates one or more data frames designed for entry onto the physical network. In the case of a LAN system such as ethernet, the frame may contain physical address information obtained from lookup tables maintained using the Internet layer **ARP** protocol. (ARP, Address Resolution Protocol, translates IP addresses to physical addresses.)
4. The data frame is converted to a stream of bits that is transmitted over the network medium.

**FIGURE 2.4**

A quick look at the basic TCP/IP networking system.

Of course, there are endless details describing how each protocol goes about fulfilling its assigned tasks. For instance, how does TCP provide flow control, how does ARP map physical addresses to IP addresses, and how does IP know where to send a datagram addressed to a different subnet? These questions are explored later in this book.

Summary

In this hour, you learned about the layers of the TCP/IP protocol stack and how those layers interrelate. You also learned how the classic TCP/IP model relates to the seven-layer OSI networking model. At each layer in the protocol stack, data is packaged into the form that is most useful to the corresponding layer on the receiving end. This hour discussed the process of encapsulating header information at each protocol layer and outlined the different terms used at each layer to describe the data package. Finally, you got a quick look at how the TCP/IP protocol system operates from the viewpoint of some of its most important protocols: TCP, UDP, IP, and ARP.

Q&A

Q. What are the principal advantages of TCP/IP's modular design?

A. Because of TCP/IP's modular design, the TCP/IP protocol stack can adapt easily to specific hardware and operating environments. One layer can change without affecting the rest of the stack. Breaking the networking software into specific, well designed components also makes it easier to write programs that interact with the protocol system.

Q. What functions are provided at the Network Access layer?

A. The Network Access layer provides services related to the specific physical network. These services include preparing, transmitting, and receiving the frame over a particular transmission medium, such as an ethernet cable.

Q. Which OSI layer corresponds to the TCP/IP Internet layer?

A. TCP/IP's Internet layer corresponds to the OSI Network layer.

Q. Why is header information enclosed at each layer of the TCP/IP protocol stack?

A. Because each protocol layer on the receiving machine needs different information to process the incoming data, each layer on the sending machine encloses header information.

Workshop

The following workshop is composed of a series of quiz questions and practical exercises. The quiz questions are designed to test your overall understanding of the current material. The practical exercises are intended to afford you the opportunity to apply the concepts discussed during the current hour, as well as build upon the knowledge acquired in previous hours of study. Please take time to complete the quiz questions and exercises before continuing. Refer to Appendix A, "Answers to Quizzes and Exercises," for answers.

Quiz

1. What two OSI layers map into the TCP/IP Network Access layer?
2. What TCP/IP layer is responsible for routing data from one network segment to another?
3. What are the advantages and disadvantages of UDP as compared to TCP?
4. What does it mean to say that a layer encapsulates data?

Exercises

1. List the functions performed by each layer in the TCP/IP stack.
2. List the layer(s) that deal with datagrams.
3. Explain how TCP/IP would have to change to use a newly invented type of network hardware.
4. Explain what it means to say that TCP is a reliable protocol.

Key Terms

Review the following list of key terms:

- ▶ **Address Resolution Protocol (ARP):** A protocol that resolves logical IP addresses to physical addresses.
- ▶ **Application layer:** The layer of the TCP/IP stack that supports network applications and provides an interface to the local operating environment.
- ▶ **Datagram:** The data package passed between the Internet layer and the Network Access layer, or a data package passed between UDP at the Transport layer and the Internet layer.
- ▶ **Frame:** The data package created at the Network Access layer.
- ▶ **Header:** A bundle of protocol information attached to the data at each layer of the protocol stack.
- ▶ **Internet layer:** The layer of the TCP/IP stack that provides logical addressing and routing.
- ▶ **IP (Internet Protocol):** The Internet layer protocol that provides logical addressing and routing capabilities.
- ▶ **Message:** In TCP/IP networking, a message is the data package passed between the Application layer and the Transport layer. The term is also used generically to describe a message from one entity to another on the network. The term doesn't always refer to an Application layer data package.

- ▶ **Network Access layer:** The layer of the TCP/IP stack that provides an interface with the physical network.
- ▶ **Segment:** The data package passed between TCP at the Transport layer and the Internet layer.
- ▶ **TCP (Transmission Control Protocol):** A reliable, connection-oriented protocol of the Transport layer.
- ▶ **Transport layer:** The layer of the TCP/IP stack that provides error control and acknowledgment and serves as an interface for network applications.
- ▶ **UDP (User Datagram Protocol):** An unreliable, connectionless protocol of the Transport layer.

Index

Symbols

<video> element, 442

Numbers

6in4 tunneling, 260

6to4 tunneling, 260

802.11 networks, 148–154, 168

access points, association, 152

address types, 151–152

frame fields, 152–153

IBSS, 149

infrastructure BSS, 149–151

security, 153–154

WEP, 153

WPA2, 154

transmission speeds, 149

A

access methods, 35, 43

access points, association,
152, 168

ACK, 105

Acknowledgment Number (32-bit
field), 96

active open connections, 97, 105

active RFID tags, 473

AD (Active Directory), 309

Adaptive Bitrate streaming, 441

address classes, 54–55, 66

address ranges

for classful addresses, 55

for IPv6 addresses, 253–254

Adobe Flash, 439

AES (Advanced Encryption
Standard), 384

AJAX (Asynchronous JavaScript
and XML), 344

algorithms, encryption, 381–382

announcements, 96

anonymity networks, 403–404

anycast, 263

Apache, 371

APIPA (Automatic Private IP
Addressing), 233, 246

APIs (application programming
interfaces), 109–110, 115, 118
network APIs, 115

- Apple Remote Desktop utility, 310
 - Application layer firewalls, 199
 - Application layer (OSI model), 23, 110
 - Application layer (TCP/IP model), 21, 29, 109–110
 - APIs, 115
 - corresponding OSI layers, 110–111
 - file and print services, 112–113
 - messages, 25
 - name resolution services, 113
 - network services, 111–112
 - remote access, 114
 - utilities, 116
 - web services, 114
 - application-level attacks, 213–214
 - applications
 - Juggernaut, 213
 - ports, 12–13, 89
 - and firewalls, 102
 - SaaS, 451–453
 - sockets, 89–90
 - architectures, 35–36, 44
 - ethernet, 38–40
 - frames, 40–41
 - PPPoE, 146
 - in Network Access layer, 36–37
 - web services architecture, 366
 - ARP (Address Resolution Protocol), 9, 29, 37, 61–62, 66
 - connectivity problems, troubleshooting, 272–274
 - arp utility, 116
 - ARPAnet, 16
 - ARPAnet (Advanced Research Projects Agency network), 6
 - ascii command, 302
 - ASNs (autonomous system numbers), 137
 - assigning
 - IP addresses, 9
 - DHCP, 225
 - NAT, 230–232
 - Zeroconf system, 232–235
 - subnet masks, 73
 - association, 152
 - asymmetric encryption, 384–385
 - attacks
 - application-level attacks, 213–214
 - credential attacks, 207–212
 - guessing, 210
 - intercepting, 210–211
 - mitigating, 211–212
 - password protection, 208
 - Trojan horses, 208–209
 - denial-of-service attacks, 217–218
 - network-level attacks, 212–213
 - phishing, 215–217, 426–427
 - security best practices, 218–219
 - authentication
 - digital certificates, 386–388
 - digital signatures, 385–386
 - Kerberos, 393–395
 - Authentication header (IPv6), 252
 - authority field (URLs), 323–324
 - autoconfiguration, IPv6, 256–257
 - autonomous systems, 133–134, 140
 - interior gateways, 134
 - interior routing protocols, 134
 - AWS (Amazon Web Services), 459–460
- ## B
- back doors, 207
 - backbone, 318
 - backup solutions, cloud-based, 452–453
 - basic header (RTMP), 439
 - Berkeley r* utilities, 283–284
 - Berners-Lee, Tim, 329–330
 - best practices, for security, 218–219
 - BGP (Border Gateway Protocol), 136–137, 140
 - binary command, 301
 - binary IP addresses, converting to dotted-decimal format, 56–57, 74–75
 - BinHex utility, 412
 - BitTorrent, 112
 - blacklists, 424
 - blogs, 362–363
 - Bluetooth, 155–156, 168
 - BOOTP, 63, 66, 225
 - bridges, 161–162, 168
 - broadband technologies
 - cable broadband, 144–145
 - DSL, 145–146
 - broadcasts, 60, 100

browsers, 337, 344–347
 microformats, 349–350
 plug-ins, 345–346
 security, 346–347

BSD Unix, 283–284

bye command, 302

C

cable broadband, 144–145

cabling, 43

cabling rules, 35

CAs (certificate authorities), 386–387

cd command, 301

Cerberus, 395

Cerf, Vinton, 6

CGI (Common Gateway Interface), 343

chain of trust, DNSSEC, 187–188

Chatty Things, 468

Checksum field (TCP), 96

Checksum field (UDP), 101

chunk header (RTMP), 439

CIDR (Classless Inter-Domain Routing), 51, 70, 79–81, 83, 137–138, 480

CIFS (Common Internet File System), 112, 305–306

Class A addresses, 54
 delivering data to, 70–71
 subnetting, 78

Class B addresses, 54
 subnetting, 75, 78–79

Class C addresses, 54, 479
 subnetting, 75–77, 79

Class D addresses, 55

Class E addresses, 55

classful addresses, address ranges, 55

classless routing, 137–138

clients, 97

client-side scripting, 343–344

close command, 302

closing TCP connections, 99

cloud computing, 449–450
 comparing to World Wide Web, 450
 containers, 457–458
 data centers, 458–459
 elastic cloud services, 459–460
 hybrid cloud, 454
 IaaS, 453–455
 IoT platforms, 469–470
 orchestration, 458
 PaaS, 455
 private clouds, 460–461
 SaaS, 451–453
 backup solutions, 452–453
 storage, 452
 virtualization, 456–457
 drivers for adoption, 456–457
 provisioning, 458
 vendors, 456

CMS (content management system), 360–361
 blogs, 362–363
 Facebook, 361–362
 WYSIWYG editing, 360

CMTS (cable modem termination service), 144–145, 168

CNAME (canonical name) records, 183

codecs, 446

collisions, 38–40

commands
 FTP, 300–303
 SNMP, 289–290

communities (SNMP), 286

comparing
 cloud computing and World Wide Web, 450
 routing and switching, 165–166

confidentiality, 380

configuring
 IPv6, 256–257
 TCP/IP, 223–224, 235–236
 on Linux, 241–243
 on MAC OS, 240–241
 on Windows operating systems, 236–239

connecting to the Internet
 broadband technologies
 cable broadband, 144–145
 DSL, 145–146
 dial-up networking, 157–161
 point-to-point connections, 157–158
 Mobile IP, 154–155
 WANs, 146–147

connectionless protocols, 87, 105

connection-oriented protocols, 87, 88, 105

connections, TCP
 active open, 97
 closing, 99
 establishing, 98–99

- flow control, 99
- passive open, 97
- connectivity devices**
 - hubs, 162–163
 - switches, 163–164
- connectivity problems, trouble-shooting, 268–274**
 - ARP, 272–274
 - configuration information
 - utilities, 271–272
 - ping utility, 269–271
- container files, 432, 433**
- containers, 457–458**
 - provisioning, 458
- content caching, 204**
- control flags, 96, 106**
- converting**
 - binary IP addresses to
 - dotted-decimal format, 56–57, 74–75
 - decimal numbers to binary octet, 58–60
- cookies, 396–398**
 - managing, 399–400
 - persistent cookies, 398
 - session cookies, 397–398
 - third-party cookies, 398–399
- core routers, 133**
- counters, 288**
- CRC (cyclical redundancy check), 43**
- creating hosts files, 174–175**
- credential attacks, 207–212**
 - guessing, 210
 - intercepting, 210–211
 - mitigating, 211–212
 - password protection, 208
 - Trojan horses, 208–209

- cross-site scripting, 216**
- CSMA/CD (carrier sense multiple access with collision detection), 38, 43**
- CSS (Cascading Style Sheets), 337–338**
- cut-through switching, 168**

D

- data centers, 458–459**
- data frame format, 43**
- Data Link layer (OSI model), 23, 43**
 - sublayers, 35
- Data Offset field (TCP), 96**
- data packages, 24–25**
- datagrams, 25, 29**
 - DHCP, 226–227
 - IP, header fields, 51–53
- DCCP (Datagram Congestion Control Protocol), 101, 440**
- decentralized environments, 6–7**
- decimal numbers, converting to binary octet, 58–60**
- default routers, 127**
- delivery process (email), 413–416**
- demultiplexing, 86, 92, 106**
- denial-of-service attacks, 217–218**
- depletion of IPv4 addresses, 248**
- DES (Data Encryption Standard), 384**
- Destination IP Address field (IP), 53**
- Destination Options header (IPv6), 251**
- Destination Port field (UDP), 101**

- Destination Unreachable messages, 63**
- development of TCP/IP, 6–7**
- devices**
 - bridges, 161–162
 - codecs, 446
 - encoder devices, 433
 - firewalls, 102, 106
 - hubs, 162–163
 - Layer 2, 139
 - Layer 3, 121–122
 - link status lights, 274
 - modems, 144
 - NAT, 231
 - network interface devices, 46
 - routers, 9–11, 122–124
 - core routers, 133
 - exterior routers, 134
 - higher-level access, 138–139
 - Home Agent, 154
 - routing tables, 73
 - switches, 163–164
- DHCP (Dynamic Host Configuration Protocol), 112, 224, 225, 481–482**
 - leasing IP addresses, 226–227
 - relay agents, 227–228
 - server configuration, 229–230
 - time fields, 228–229
- dial-up networking, 157–161**
 - modems, 157
 - point-to-point connections, 157–158
- PPP, 158–161**
 - connection lifecycle, 161
 - frames, 160

- LCP, 159
 - NCPs, 159
 - packets, 160
 - SLIP, 158
 - Dig utility, 191–192**
 - digital certificates, 386–388**
 - digital signatures, 385–386**
 - direct routing, 128**
 - directory services, LDAP, 306–309**
 - distance-vector routing, 130–132.**
 - See also* link-state routing
 - hop count, 130–131
 - RIP, 135–136
 - updates, 131–132
 - DMZ (demilitarized zone), 200–202**
 - DN (distinguished name), 307**
 - DNS (Domain Name System)**
 - names, **11, 112, 173, 482–483**
 - Dig utility, 191–192
 - domain name registration, 181
 - dynamic DNS, 192–193
 - FQDNs, 177
 - mDNS, 234
 - name resolution, 175–180
 - name server types, 182
 - TLDs, 177
 - verifying name resolution
 - with NSlookup utility, 190–191
 - with ping, 189
 - zones, 182–186
 - resource records, 183
 - reverse lookup files, 185–186
 - SOA records, 184–185
 - DNS-SD (DNS Service Discovery), 234–235**
 - DNSSEC (DNS Security Extensions), 186–189**
 - chain of trust, 187–188
 - resource records, 187
 - Do Not Track initiative, 402**
 - DOCSIS (Data Over Cable Service Interface Specification), 145, 168**
 - domain names, 11, 16**
 - registering, 181
 - dotted-decimal format, 53–54, 66**
 - binary IP addresses,
 - converting to, 56–57, 74–75
 - downloading multimedia files, 433**
 - Dreamweaver, 360**
 - DSL (digital subscriber line), 145–146**
 - forms of, 146
 - DSLAM (digital subscriber line access multiplexer), 145**
 - DSSS (direct-sequence spread spectrum), 148**
 - Duration/ID field (802.11), 152**
 - dynamic DNS, 192–193**
 - dynamic routing, 7, 126, 129, 141**
- E**
- eBGP (Exterior Border Gateway Protocol), 136**
 - EC2 (Elastic Cloud Compute), 459–460**
 - Echo Request/Reply messages, 63**
 - e-commerce, 374–376**
 - payment gateways, 375
 - web transactions, 375
 - editing hosts files, 174**
 - elastic cloud services, 459–460**
 - email, 411–412**
 - address format, 415
 - clients, 420–422
 - delivery process, 413–416
 - format, 412–413
 - body, 412
 - header fields, 413
 - IMAP, 414
 - IMAP4, 420
 - mailbox, 414
 - MIME, 412
 - phishing, 426–427
 - POP, 415
 - POP3, 419–420
 - privacy, 427
 - retrieving mail, 418–419
 - security, 422
 - SMTP, 415, 416–418
 - client commands, 417
 - delivering email to mailbox, 417–418
 - spam, 423–426
 - blacklists, 424
 - graylists, 425–426
 - whitelists, 425
 - webmail, 422–423
 - EME (Encrypted Media Extension), 442**
 - encapsulation, 22**
 - encoder devices, 433**
 - Encrypted Security Payload header (IPv6), 253**

encryption, 210, 380

- algorithm, 381–382
- asymmetric encryption, 384–385
- digital certificates, 386–388
- digital signatures, 385–386
- keys, 380–382
- symmetric encryption, 382–384

end-node verification, 7**error control, 12****establishing TCP connections, 98–99****ethernet, 38–40, 43**

- 802.11 networks, 148–154
- collisions, 38–40
- CSMA/CD, 38
- frames, fields, 40–41
- IEEE 802.3, 36
- PPPoE, 146

extension headers, IPv6, 250–253

- Authentication header, 252
- Destination Options header, 251
- Encrypted Security Payload header, 253
- Fragment header, 252
- Hop-by-Hop options header, 251
- Routing header, 252

exterior routers, 134, 141

- BGP, 136–137

F**Facebook, 361–362****FCS (Frame Check Sequence) field, 41, 44, 152****FHSS (frequency-hopping spread spectrum), 148****fields**

- of 802.11 frame, 152–153
- of ethernet frame, 40–41
- of IP header, 51–53
- of IPv6 header, 250
- of TCP segment, 95–97
- of UDP header, 100–101

file and print services, 112–113

- CIFS, 305–306
- NFS, 304–305
- SMB, 305–306

FIN flag, 96, 106**finger protocol, 112, 116****Firestore, 468–469****firewalls, 102, 106, 197–199, 484. See also attacks**

- Application layer firewalls, 199
- DMZ, 200–202
- intruders, 205–206
 - back doors, 207
 - motivations of, 206–207
- options, 199–200
- packet filters, 198
- personal firewalls, 199
- proxy services, 203–204
- reverse proxy, 204–205
- rules, 202–203
- SOHO firewalls, 199
- stateful firewalls, 198–199
- UNIX/Linux systems, 200

Flags field (IP), 52**flow control, 12, 93**

- TCP, 99

flow levels, 249**Foreign Agent, 154****format of email, 412–413**

- body, 412
- header fields, 413

FQDNs (fully qualified domain names), 173, 177**Fragment header (IPv6), 252****Fragment Offset field (IP), 53****fragment section (URLs), 325****Fragmentation Needed messages, 64****frame control field (802.11), 152****frames, 25, 29**

- 802.11
 - address types, 151–152
 - fields, 152–153
 - ethernet, fields, 40–41
 - PPP, 160

ftp, 13**FTP (File Transfer Protocol), 112, 299–303**

- commands, 300–303
- on WWW, 299

G**gateways, 7, 16**

- default gateways, 127
- interior gateways, 134
- VoIP, 444–445

geolocation, 354**get command, 302****goals of IPv6, 249****graceful close, 93****graylists, 425–426**

H

- H.323 protocol, 444**
 - Header Checksum field (IP), 53**
 - headers, 24, 29. See also extension headers, IPv6**
 - IP, 51–53
 - IPv6, 249–253
 - pseudo-headers, 101
 - RTMP, 439
 - RTP, 436–437
 - TCP, 95–97
 - UDP, 100–101
 - Home Agent, 154**
 - HomeKit, 468**
 - hop count, 130–131**
 - Hop-by-Hop options header (IPv6), 251**
 - hops, 53**
 - host ID, 50, 66**
 - hostname utility, 116**
 - hosts, 49**
 - calculating for address classes, 54–55
 - hosts files, 172**
 - creating, 174–175
 - editing, 174
 - name resolution, 173–175
 - HR/DSSS (high-rate direct sequence spread spectrum), 148**
 - HTML (Hyper Text Markup Language), 298, 329–330, 332–337**
 - links, 337
 - tags, 332–336
 - HTML5, 351–355**
 - drawing, 353–354
 - embedded audio and video, 354
 - EME, 442
 - geolocation, 354
 - local storage, 351–353
 - MSE, 442
 - offline application support, 351–353
 - semantics, 355
 - HTTP (Hypertext Transfer Protocol), 112, 114, 297, 298, 329–330, 338–341, 366**
 - header fields, 340
 - status codes, 340
 - streaming, 434, 440–441
 - HTTP Live Streaming, 441**
 - hubs, 162–163**
 - link status lights, 274
 - hybrid cloud, 454**
- I**
- IaaS (infrastructure as a service), 453–455**
 - IAB (Internet Architecture Board), 13**
 - IANA (Internet Assigned Names Numbers Authority), 13–14**
 - iBGP (Interior Border Gateway Protocol), 136**
 - IBSS (Independent Basic Service Set), 149**
 - ICANN (Internet Corporation for Assigned Names and Numbers), 13, 79–80**
 - assignment of IP addresses, 9
 - domain name registration, 181
 - ICMP (Internet Control Message Protocol), 63–64, 66**
 - Identification field (IP), 52**
 - IEEE 802.11, 43**
 - IEEE 802.3, 36, 43**
 - IETF (Internet Engineering Task Force), 13**
 - ifconfig, 116**
 - ifconfig command, 271–272**
 - IHL (Internet Header Length) field, 52**
 - IMAP (Internet Message Access Protocol), 112, 414**
 - IMAP4 (Internet Message Access Protocol version 4), 420**
 - implementations, 5, 17**
 - indirect routing, 129, 141**
 - infrastructure BSS, 149–151**
 - intelligent hubs, 163**
 - intercepting passwords, 210–211**
 - interior gateways, 134**
 - interior routing protocols, 134**
 - OSPF, 136
 - RIP, 135–136
 - Internet**
 - autonomous systems, 133–134
 - broadband technologies
 - cable broadband, 144–145
 - DSL, 145–146
 - client applications, 321
 - decentralized environment, 6–7
 - development of, 6
 - dial-up networking, 157–161

- point-to-point connections, 157–158
- email, 298–299
- firewalls, 197–199
 - Application layer, 199
 - DMZ, 200–202
 - intruders, 205–206
 - options, 199–200
 - packet filters, 198
 - personal firewalls, 199
 - proxy services, 203–204
 - reverse proxy, 204–205
 - rules, 202–203
 - SOHO firewalls, 199
 - stateful firewalls, 198–199
 - UNIX/Linux systems, 200
- Mobile IP, 154–155
- proliferation of personal computers, 7–8
- security, 320
- server applications, 321
- services, 321–322
- structure of, 317–320
 - backbone, 318
 - IXPs, 319–320
 - Tier 1 networks, 318
 - Tier 2 networks, 318
 - Tier 3 networks, 318
- URLs, 323–325
 - authority field, 323–324
 - fragment section, 325
 - path component, 324
 - query component, 324–325
 - scheme field, 323
- WANs, 146–147
- Internet layer (TCP/IP model), 21, 29, 45, 47–49. *See also***
- IP addresses**
 - ARP, 61–62
 - datagrams, 25
 - dial-up networking, PPP, 158–161
 - IMCP, 63–64
 - IP addresses, 46
 - hosts, 49
 - RARP, 62–63
- internetworks, 21**
- intruders**
 - back doors, 207
 - motivations of, 206–207
- IoT (Internet of Things), 465–467, 484–486**
 - management systems, 469
 - MQTT, 470–472
 - platforms, 467–470
 - cloud-based, 469–470
 - Firebase, 468–469
 - HomeKit, 468
 - publishing model, 472
 - RFID, 472–474
 - active tags, 473
 - subscriptions, 472
- IoTivity, 468**
- IP (Internet Protocol), 29, 49–53**
- IP addresses, 9, 16, 46, 53–55**
 - address classes, 54–55
 - broadcasts, 60
 - CIDR, 51
 - dotted-decimal format, 53–54
 - host ID, 50
 - hosts, 49
- leasing, 226–227
 - relay agents, 227–228
 - time fields (DHCP), 228–229
- loopback addresses, 61
- multicasting, 55
- name resolution, 11, 171–173
 - DNS, 175–180, 182
 - domain name registration, 181
 - dynamic DNS, 192–193
 - hosts files, 173–175
 - NetBIOS, 193–194
- network ID, 50
- octets, 53–54
- routing tables, 126–128
 - next-hop entry, 126
- static IP addressing, 224–225
- subnetting, 50, 51, 55, 69–70, 480–481
 - CIDR, 79–81
 - Class A addresses, 78
 - Class B addresses, 75, 78–79
 - Class C addresses, 75–77, 79
 - subnet masks, 71–79
 - Zeroconf system, 232–235
- IP Data Payload field (IP), 53**
- IP forwarding, 127–128**
- IP Options field, 53**
- IP telephony, VoIP, 443–445**
 - gateways, 444–445
 - H.323 protocol, 444
 - SIP, 444
- iPodder, 443**
- IPsec, 64, 390–391**

IPv4, 48

IPv4-mapped IPv6 addresses, 258

IPv6

- address ranges, 253–254
- autoconfiguration, 256–257
- extension headers, 250–253
- headers, 249–253
- with IPv4, 258
- link-local addresses, 255–256
- multicasting, 255
- neighbor discovery, 256
- QoS, 257
- reasons for, 248–249
- subnetting, 254–255
- tunnels, 258–261
 - 6in4 tunneling, 260
 - 6to4 tunneling, 260
 - TSP, 261

IRTF (Internet Research Task Force), 13

ISN (initial sequence number), 106

ISO (International Organization for Standardization), 22

ISPs (Internet service providers), interior routers, 134

iTunes, 443

IXPs (Internet exchange points), 319–320

J

Jobs, Steve, 439

JSON (JavaScript Object Notation), 367

Juggernaut, 213

jumbo payload, 263

K

Kahn, Robert E.6

KDC (Key Distribution Center), 393

Kerberos, 393–395

key loggers, 215

keys, 380–382

- private key, 384
- public key, 384

L

LANs (local area networks), 7–8, 16

- architectures, 35–36
- ethernet, 38–40

Layer 2 devices, 139

Layer 3 devices, 121–122

layers, 22

- of ARPAnet model, 21
- encapsulation, 22
- of OSI model, 22–24
- of TCP/IP model, 21–22
 - Application layer, 21
 - headers, 24
 - Internet layer, 21
 - Network Access layer, 22
 - Transport layer, 21

LCP (Link Control Protocol), 159

LDAP (Lightweight Directory Access Protocol), 112, 306–309

- AD, 309
- schema, 307

LDIF (LDAP Data Interchange Format), 308

leasing IP addresses, 226–227.

See also name resolution

- DHCP time fields, 228–229
- relay agents, 227–228

Length field (ethernet), 41

Length field (UDP), 101

line problems, troubleshooting, 274

link status lights, 274

link-local addresses, IPv6, 255–256

links, HTML, 337

link-state routing, 132–133

- OSPF, 136

Linux, 371

- firewalls, 200
- FTP, 300
- ifconfig command, 271–272
- TCP/IP, configuring, 241–243

LLC (Logical Link Control) sub-layer, 35, 44

LLNR (Link-Local Multicast Name Resolution), 235

LMHosts files, 194

logical addressing, 8–9, 16

- physical address, 8

loopback addresses, 61

Lpr, 13

M

MAC (Media Access Control) address, 8, 37

MAC (Media Access Control) sublayer, 35, 44

MAC OS, configuring TCP/IP, 240–241

Macromedia, 438

mailbox, 414

management console (SNMP), 286

markup languages, 329–330
HTML, 332–337

mDNS (multicast DNS), 234

MediaWiki, 363

messages, 25, 29. *See also* email
ICMP, 63–64
SOAP, 370

metadata, 372

metafiles, 435

mget command, 302

MIB (Management Information Base), 287–288
counters, 288
RMON, 290–292
structure of, 288

microformats, 349–350

MIME (Multipurpose Internet Mail Extensions), 412

mitigating credential attacks, 211–212

mkdir command, 301

Mobile IP, 154–155

modem (modulator/demodulator), 144, 157

modular design of TCP/IP, 20

motivations of intruders, 206–207

MPEG-DASH (MPEG Dynamic Adaptive Streaming over HTTP), 441

mput command, 302

MQTT, 468, 470–472

MSE (Media Source Extensions), 442

multicasting, 55, 67
IPv6, 255

multihomed computers, 122–124, 479

multimedia
container files, 432
downloading files, 433
HTML5, 354, 442
podcasting, 442–443
software, 435
streaming, 431–432
container files, 433
DCCP, 440
encoder devices, 433
HTTP, 434, 440–441
metafiles, 435
QoS, 432
RTMP, 438–439
RTP, 435–438
SCTP, 440
video file formats, 434–435

multiplexing, 86, 92, 106

MySQL, 371

N

name resolution, 11, 113, 171–173, 482–483
DNS, 175–180
DNSSEC, 186–189
FQDNs, 177
name server types, 182
TLDs, 177
zones, 182–186

domain name registration, 181

dynamic DNS, 192–193

hosts files, 173–175
creating, 174–175
editing, 174

NetBIOS, 193–194

troubleshooting, 274–275

verifying
with Dig utility, 191–192
with NSlookup utility, 190–191
with ping, 189

name services, 16

NAT (Network Address Translation), 9, 230–232, 481–482

National Science Foundation, 6

NCPs (network control protocols), 159

NDP (Neighbor Discovery Protocol), 256

neighbor discovery, IPv6, 256

netstat utility, 116, 279–280

Network Access layer (TCP/IP model), 22, 30, 33–34
architectures, 36–37
ethernet, 38–40
frames, 40–41
frames, 25
and OSI model, 34–35
physical addressing, 37
responsibilities, 33

network adapters, physical address, 8

network APIs, 115

network ID, 50, 67
routing tables, 126–128

network interface devices, 46

Network layer (OSI model), 23

network services, 111–112

network-level attacks, 212–213

networks, 4. *See also* wireless networks

- architectures, 35–36
- ARPANet, 6
- connectivity devices
 - bridges, 161–162
 - hubs, 162–163
 - switches, 163–164
- LANs, 7–8
- management tools, 285–292
 - RMON, 290–292
 - SNMP, 286–290
- performance, troubleshooting, 275–280
 - netstat utility, 279–280
 - route utility, 277–278
 - traceroute utility, 276–277
- routing, 9–11
- TOR networks, 403–404

next-hop, 126

NFS (Network File System), 112, 304–305

nodes (SNMP), 286

NS (Name Server) records, 183

NSlookup, 13

- verifying name resolution, 190–191

NTIA (U.S. National Telecommunications and Information Administration), 13–14

NTP (Network Time Protocol), 112

O

octets, 53–54, 67

- decimal numbers, converting to, 58–60

OFDM (orthogonal frequency-division multiplexing), 148

open command, 302

Optional VLAN tag field (ethernet), 40

orchestration, 458

OSI (Open Systems Interconnection) model, 22–24, 34–35

- Application layer, 110
- Data Link layer, sublayers, 35
- Presentation layer, 111
- Session layer, 111

OSPF (Open Shortest Path First), 136, 141

P

PaaS (platform as a service), 455

packet filters, 198

packets, 160

Padding field (IP), 53

PAM (Pluggable Authentication Module), 309

passive open connections, 97

passive RFID tags, 473

passwords

- encryption, 210
- intercepting, 210–211
- protecting, 208

path component (URLs), 324

path MTU, 263

payment gateways, 375

peer-to-peer networking, 321, 364–365

persistent cookies, 397

personal computers

- firewalls, 199
- multihomed computers, 122–124
- proliferation of, 7–8

PGP (Pretty Good Privacy), 427

phishing, 215–217, 426–427

PHP, 343, 371

physical addressing, 16, 37, 44

Physical layer (OSI model), 23

ping utility, 13, 116, 480

- connectivity problems, troubleshooting, 269–271
- output, 271
- verifying name resolution, 189

platforms, IoT, 467–470

- cloud-based, 469–470
- Firebase, 468–469
- HomeKit, 468

plug-ins, 345–346

podcasting, 442–443

point-to-point connections, 157–158

POP (point of presence) connections, 318

POP (Post Office Protocol), 112, 415

POP (Post Office Protocol version 3), 419–420

ports, 12–13, 16, 89

and firewalls, 102

well-known ports

TCP, 90–91

UDP, 91

PowerShell, 192**PPP (Point-to-Point Protocol), 36, 158–161**

connection lifecycle, 161

frames, 160

LCP, 159

NCPs, 159

packets, 160

SLIP, 158

PPPoE (Point-to-Point Protocol over Ethernet), 146**Preamble field (ethernet), 40, 44****preconfigured static routes, 126****Presentation layer (OSI model), 23, 111****print servers, 112–113****printing**

Lpr utility, 13

print servers, 112–113

privacy, encryption, 380**private clouds, 460–461****private key, 384****privileges, root access, 214–215****proliferation of personal computers, 7–8****proprietary technologies, 6, 16****protecting passwords, 208****Protocol field (IP), 53****protocol suites, 5**

implementations, 5

protocol systems, 3

ARPAnet, 6

responsibilities of, 20

TCP/IP. *See* TCP/IP**protocols**

connectionless, 87

connection-oriented, 87, 88

network protocols, 4–5

provisioning, 458**proxy services, 203–204****pseudo-headers, 101, 106****PSH announcement, 96****public key, 384****publishing model, IoT, 472****put command, 302****pwd command, 301****Q****QoS (quality of service), 257, 432****quality assurance, 86****query component (URLs), 324–325****quit command, 302****R****r* utilities, 283–284****RARP (Reverse Address Resolution Protocol), 9, 37, 62–63, 67. *See also* ARP (Address Resolution Protocol)****RDF (Resource Description Framework), 348–349****RDN (relative distinguished name), 307****readers, RFID, 473****Recipient address field (ethernet), 40****redirectors, 114, 118, 304****registering domain names, 181****relay agents, 227–228****remote access, 114, 309–311**

BSD Unix, 283–284

SSH, 284–285

Telnet, 280–282

VPNs, 391–393

resequencing, 93, 100, 106**Reserved field (TCP), 96****resource records, 183**

DNSSEC, 187

responsibilities

of Network Access layer, 33

of protocol systems, 20

of TCP, 93

of Transport layer, 86

REST (Representational State Transfer), 371–374, 452

metadata, 372

requests, 372–373

URIs, 374

reverse lookup files, 185–186**reverse proxy, 204–205****rexec, 295****rfc-editor.org, 14****RFCs (Requests for Comments), 14, 17****RFID (Radio Frequency Identification), 472–474**
active tags, 473**RIP (Routing Information Protocol), 135–136, 141****rmdir command, 301**

RMON (Remote Monitoring), 290–292
 versions of, 291

root access, 214–215

rootkits, 215

route utility, 116, 277–278

routers, 17, 122–124. *See also* routing
 core routers, 133
 exterior routers, 134
 BGP, 136–137
 higher-level access, 138–139
 Home Agent, 154
 link status lights, 274
 routing tables, 73

routing, 9–11, 124–125
 classless routing, 137–138
 direct routing, 128
 dynamic routing, 126
 distance-vector routing, 130–132
 indirect routing, 129
 IP forwarding, 127–128
 static routing, 125
 versus switching, 165–166

Routing header (IPv6), 252

routing loops, 64

routing protocols
 BGP, 136–137
 distance-vector routing, 130–132
 hop count, 130–131
 RIP, 135–136
 updates, 131–132
 link-state routing, 132–133
 OSPF, 136

routing tables, 73, 126–128
 next-hop entry, 126

RPC (Remote Procedure Call), 92–93, 112

RST announcement, 96

RTCP (Real-Time Control Protocol), 436

RTMP (Real Time Messaging Protocol), streaming over TCP, 438–439

RTP (Real-Time Transport Protocol), 101, 435–438
 header fields, 436–437

RTSP (Real Time Streaming Protocol), 437

rules, firewall rules, 202–203

S

SaaS (software as a service), 451–453
 backup solutions, 452–453
 storage, 452

schema, 307, 368

scheme field (URLs), 323

script kiddies, 205

scripting, 341–344
 client-side scripting, 343–344
 server-side scripting, 342–343

SCTP (Stream Control Transmission Protocol), 101, 440

security
 802.11 networks, 153–154
 WEP, 153
 WPA2, 154

attackers, 205–206

attacks
 application-level attacks, 213–214
 credential attacks, 207–212
 denial-of-service attacks, 217–218
 network-level attacks, 212–213
 phishing, 215–217
 best practices, 218–219
 DNSSEC, 186–189
 resource records, 187
 email, 422
 encryption, 380
 algorithm, 381–382
 asymmetric encryption, 384–385
 keys, 380–382
 symmetric encryption, 382–384

firewalls, 102, 197–199, 484
 Application layer firewalls, 199
 DMZ, 200–202
 options, 199–200
 packet filters, 198
 personal firewalls, 199
 proxy services, 203–204
 reverse proxy, 204–205
 rules, 202–203
 SOHO firewalls, 199
 stateful firewalls, 198–199
 UNIX/Linux systems, 200

Internet, 320
 IPsec, 390–391

- Kerberos, 393–395
- root access, 214–215
- SSL, 388–390
- TLS, 388–390
- tracking, 395–404
 - anonymity networks, 403–404
 - cookies, 396–398
 - Do Not Track initiative, 402
 - trusted access, 295
 - VPNs, 391–393
 - web browsers, 346–347
- segmenting, 480–481**
- segments, 25, 30**
 - TCP, fields, 95–97
- Semantic Web, 348**
 - microformats, 349–350
 - RDF, 348–349
- Sequence Number (32-bit) field, 95**
- servers, 97, 112**
 - DNS servers, 176–177
 - virtual server systems, 454
 - WINS, 194
- server-side scripting, 342–343**
- services**
 - daemons, 299
 - email, 298–299
 - FTP, 299–303
 - HTTP, 298
 - LDAP, 306–309
 - NFS, 304–305
 - remote control, 309–311
 - SMB, 305–306
 - TFTP, 303
 - web services, 365–367
 - e-commerce, 374–376
 - HTTP, 366
 - JSON, 367
 - REST, 371–374
 - SOAP, 369–370
 - stacks, 371
 - WSDL, 370–371
 - XML, 367, 368–369
- session cookies, 397**
- session hijacking, 212**
- Session layer (OSI model), 23, 111**
- shortcomings of SNMP, 290**
- signatures, 486–487**
- SIP (Session Initiation Protocol), 444**
- Slash, 362**
- sliding windows, 99, 106**
- SLIP (Serial Line Internet Protocol), 158**
- SLP (Service Location Protocol), 235**
- smartphones, IoT solutions, 469**
- SMB (Server Message Block), 112, 305–306**
- SMTP (Simple Mail Transfer Protocol), 415, 416–418**
 - client commands, 417
 - delivering email to mailbox, 417–418
- Smurt attacks, 217–218**
- SNMP (Simple Network Management Protocol), 112, 286–290**
 - commands, 289–290
 - communities, 286
 - management console, 286
 - MIB, 287–288
 - counters, 288
 - structure of, 288
 - nodes, 286
 - shortcomings of, 290
 - traps, 295
- SOA (Start of Authority) records, 183, 184–185**
- SOAP (Simple Object Access Protocol), 367, 369–370**
- social networking, 361–362**
- sockets, 89–90, 106**
- Sockets API, 115, 118**
- software**
 - implementations, 5
 - MediaWiki, 363
 - multimedia, 435
- SOHO (small office/home office) firewalls, 199**
- Source address field (ethernet), 40**
- Source IP Address field (IP), 53**
- Source Port (16-bit) field, 95**
- Source Port field (UDP), 101**
- Source Quency messages, 63**
- spam, 423–426**
 - blacklists, 424
 - graylists, 425–426
 - whitelists, 425
- SPT (shortest path tree), 136, 141**
- SSDP (Simple Service Discovery Protocol), 235**
- SSH (Secure Shell), 102, 284–285**
- SSL (Secure Sockets Layer), 388–390**
- stacks, 20**

Stallings, William, 395
standards, 5, 23–24
 OASIS, 470–471
standards organizations
 IAB, 13
 IANA, 13–14
 ICANN, 9, 13
 IETF, 13
 IRTF, 13
stateful firewalls, 198–199
static IP addressing, 224–225
static routing, 125, 129, 141
status codes (HTTP), 340
status command, 302
storage, cloud-based, 452
store-and-forward switching, 169
streaming, 431–432
 container files, 432, 433
 DCCP, 440
 encoder devices, 433
 HTTP, 434, 440–441
 metafiles, 435
 QoS, 432
 RTMP, 438–439
 RTP, 435–438
 SCTP, 440
 software, 435
 video file formats, 434–435
stream-oriented processing,
 93, 106
structure of the Internet, 317–320
 backbone, 318
 IXPs, 319–320
 Tier 1 networks, 318
 Tier 2 networks, 318
 Tier 3 networks, 318

sublayers of Data Link layer, 35
subnet masks, 71–79, 83
 assigning, 73
 dotted notation to binary
 pattern, 78–79
subnetonline.com, 262
subnetting, 50, 51, 55, 67, 69–70,
 480–481
 CIDR, 79–81
 Class A addresses, 78
 Class B addresses, 75,
 78–79
 Class C addresses,
 75–77, 79
 IPv6, 254–255
 subnet masks, 71–79
subscriptions, IoT, 472
supernet masks, 83
switches, 163–164
 link status lights, 274
symmetric encryption, 382–384
SYN flag, 96

T

tags
 HTML, 332–336
 RFID, 473
 XML, 368
tasks of protocol systems, 20
**TCP (Transmission Control
 Protocol)**, 26–27, 30, 92–94
 announcements, 96
 connections, 97–99
 active open, 97
 closing, 99
 establishing, 98–99
 flow control, 99
 passive open, 97
 data format, 95–97
 quality assurance, 86
 responsibilities, 93
 well-known ports, 90–91
 windows, 96
**TCP/IP. See also DHCP (Dynamic
 Host Configuration Protocol)**
 Application layer, 21, 109–110
 APIs, 115
 file and print services,
 112–113
 messages, 25
 name resolution services,
 113
 network services,
 111–112
 remote access, 114
 utilities, 116
 web services, 114
 configuring, 223–224,
 235–236
 on Linux, 241–243
 on MAC OS, 240–241
 on Windows operating
 systems, 236–239
 development of, 6–7
 error control, 12
 flow control, 12
 headers, 24
 Internet layer, 21, 45
 ARP, 61–62
 datagrams, 25
 hosts, 49

- ICMP, 63–64
- IP addresses, 46
- RARP, 62–63
- layers, 21–22
- logical addressing, 8–9
- modular design of, 20
- name resolution, 11
- Network Access layer, 33–34
 - architectures, 36–37
 - ethernet, 38–40
 - frames, 25
 - and OSI model, 34–35
 - physical addressing, 37
 - responsibilities, 33
- networking, 25–27
- ports, 12–13
- RFCs, 14
- routing, 9–11, 121–122
 - IP forwarding, 127–128
- security
 - IPsec, 390–391
 - SSL, 388–390
 - TLS, 388–390
- TCP, 26–27
- Transport layer, 21, 85–87
 - ports, 89
 - responsibilities of, 86
 - segments, 25
 - TCP, 92–94
 - UDP, 99–100
- UDP, 26–27, 30
- Telnet, 280–282**
- TFTP (Trivial File Transfer Protocol), 303**
- fttp utility, 116**
- third-party cookies, 398–399**
- three-way handshake, 107**
- Tier 1 networks, 318**
- Tier 2 networks, 318**
- Tier 3 networks, 318**
- Time Exceeded messages, 63**
- time fields (DHCP), 228–229**
- TLDs (top-level domains), 177**
- TLS (Transport Layer Security), 388–390**
- topologies, . See architectures**
- TOR networks, 403–404**
- Total Length field (IP), 52**
- traceroute, 13, 116**
- traceroute utility, 276–277**
- tracking, 395–404**
 - anonymity networks, 403–404
 - cookies, 396–398
 - managing, 399–400
 - persistent cookies, 398
 - session cookies, 397–398
 - third-party cookies, 398–399
 - Do Not Track initiative, 402
- tracking pixels, 401**
- tracking scripts, 401**
- tracking tokens, 401–402**
- transmission speeds, for 802.11, 149**
- Transport layer (OSI model), 23**
- Transport layer (TCP/IP model), 21, 85–87**
 - multiplexing/demultiplexing, 92
 - ports, 89
 - well-known TCP ports, 90–91
 - well-known UDP ports, 91
- responsibilities of, 86
- segments, 25
- sockets, 89–90
- TCP, 92–94
 - announcements, 96
 - data format, 95–97
 - responsibilities, 93
 - windows, 96
- UDP, 99–100
 - broadcasts, 100
- transport mode (IPsec), 391**
- traps, 295**
- Trojan horses, 208–209**
- troubleshooting**
 - connectivity problems, 268–274
 - ARP, 272–274
 - configuration information utilities, 271–272
 - ping utility, 269–271
 - line problems, 274
 - name resolution, 274–275
 - network performance, 275–280
 - netstat utility, 279–280
 - route utility, 277–278
 - traceroute utility, 276–277
- trusted access, 295**
- TSP (Tunnel Setup Protocol), 261**
- TTL (Time To Live) field, 53**
- tunnel mode (IPsec), 391**
- tunnels, IPv6, 258–261**
 - 6in4 tunneling, 260
 - 6to4 tunneling, 260
 - TSP, 261
- type command, 302**
- Type of Service field (IP), 52**

U

UDP (User Datagram Protocol), 26–27, 30, 99–100

- broadcasts, 100
- header fields, 100–101
- quality assurance, 86–87
- RTP, 435–438
- well-known ports, 91

UNIX, 478

- BSD Unix, 283–284
- daemons, 299
- firewalls, 200
- ifconfig command, 271–272

updates, distance-vector routing, 131–132

uPnP (Universal Plug and Play), 235

URG announcement, 96

Urgent Pointer, 96

URIs (Uniform Resource Identifiers), 323, 374

URLs (Uniform Resource Locators), 308–309, 323–325, 331–332, 452

- authority field, 323–324
- fragment section, 325
- path component, 324
- query component, 324–325
- scheme field, 323

utilities, 116

- Apple Remote Desktop, 310
- BSD Unix, 283–284
- Dig, 191–192
- NSlookup, 190–191
- ping utility, 189
- PowerShell, 192
- TCP/IP, 13

Uuencode utility, 412

V

VBScript, 343

verifying name resolution

- with Dig utility, 191–192
- with NSlookup utility, 190–191
- with ping, 189

Version field (IP), 52

versions of RMON, 291

video streaming, common file formats, 434–435

virtualization, 456–457

- drivers for adoption, 456–457
- laaS, 453–455
- orchestration, 458
- provisioning, 458
- vendors, 456

viruses, 422

VMs (virtual machines), 458

VNC (Virtual Network Computing), 310

VoIP (Voice over IP), 443–445

- H.323 protocol, 444
- SIP, 444

VPNs (virtual private networks), 391–393, 486–487

W

WANs (wide-area networks), 146–147

web beacon, 401

web browsers, 344–347

- microformats, 349–350
- plug-ins, 345–346
- security, 346–347

web servers, CMS, 360–361

web services, 114, 365–367, 484–486

- e-commerce, 374–376
- HTTP, 366
- JSON, 367
- REST, 371–374
 - requests, 372–373
 - URIs, 374
- SOAP, 369–370
- stacks, 371
- WSDL, 370–371
- XML, 367, 368–369

web transactions, 375

webmail, 422–423

websites, 331

- RESTful, 374
- rfc-editor.org, 14
- Slashdot.org, 362
- subnetonline.com, 262
- wikis, 363–364

well-known ports, 107

- TCP, 90–91
- UDP, 91

WEP (Wired Equivalent Privacy), 153

WEP2, 154

whitelists, 425

Wikipedia, 363

wikis, 363–364

windows (TCP), 96

- sliding windows, 99

Windows operating systems, TCP/IP, configuring, 236–239

WINS (Windows Internet Naming Service) servers, 194

wireless networks, 148

- 802.11 networks, 148–154
 - address types, 151–152
 - IBSS, 149
 - infrastructure BSS, 149–151
 - security, 153–154
 - transmission speeds, 149
- access points, association, 152
- Bluetooth, 155–156
- IoT, 465–467
- Mobile IP, 154–155

workshops

- chapter 1, 15–16
- chapter 2, 28–29
- chapter 3, 42–43
- chapter 4, 65–66
- chapter 5, 82–83
- chapter 6, 104–105
- chapter 7, 117
- chapter 8, 140
- chapter 9, 167–168
- chapter 10, 195–196
- chapter 11, 220
- chapter 12, 244–245
- chapter 13, 262–263
- chapter 14, 293–294
- chapter 15, 311–312
- chapter 16, 326
- chapter 17, 356–357
- chapter 18, 377–378

- chapter 19, 405–406
- chapter 20, 427–428
- chapter 21, 446
- chapter 22, 462–463
- chapter 23, 474–475
- chapter 24, 488

worms, 422**WPA2 (Wi-Fi Protected Access II), 154****WSDL (Web Services Description Language), 370–371****WWW (World Wide Web)**

- browsers, 337, 344–347
 - plug-ins, 345–346
 - security, 346–347
- CSS, 337–338
- FTP, 299
- HTML, 332–337
 - links, 337
 - tags, 332–336
- HTML5, 351–355
 - drawing, 353–354
 - embedded audio and video, 354
 - geolocation, 354
 - local storage, 351–353
 - offline application support, 351–353
 - semantics, 355
- HTTP, 338–341
 - header fields, 340
 - status codes, 340

scripting, 341–344

- client-side scripting, 343–344
- server-side scripting, 342–343

Semantic Web, 348

- microformats, 349–350

RDF, 348–349

XHTML, 350

WYSIWYG editing, 360**X****X.509 standard, 388****XHTML, 350****XML (Extensible Markup****Language), 367, 368–369, 452**

AJAX, 344

schema, 368

tags, 368

Z**Zeroconf system, 61, 232–235****zones, 182–186**

DNSSEC, 186–189

resource records, 183

reverse lookup files, 185–186

SOA records, 184–185