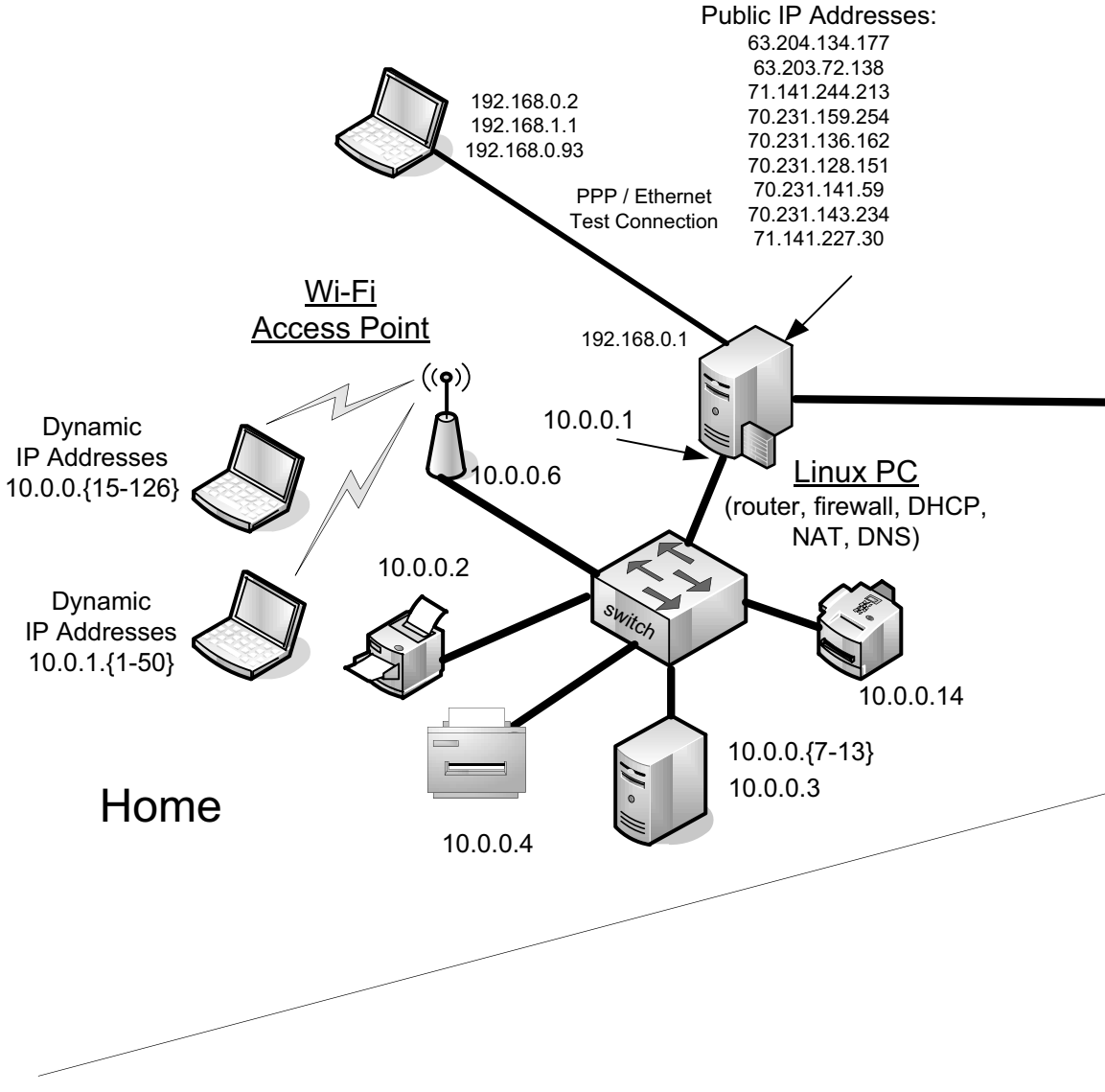
TCP/IP Illustrated, Volume 1

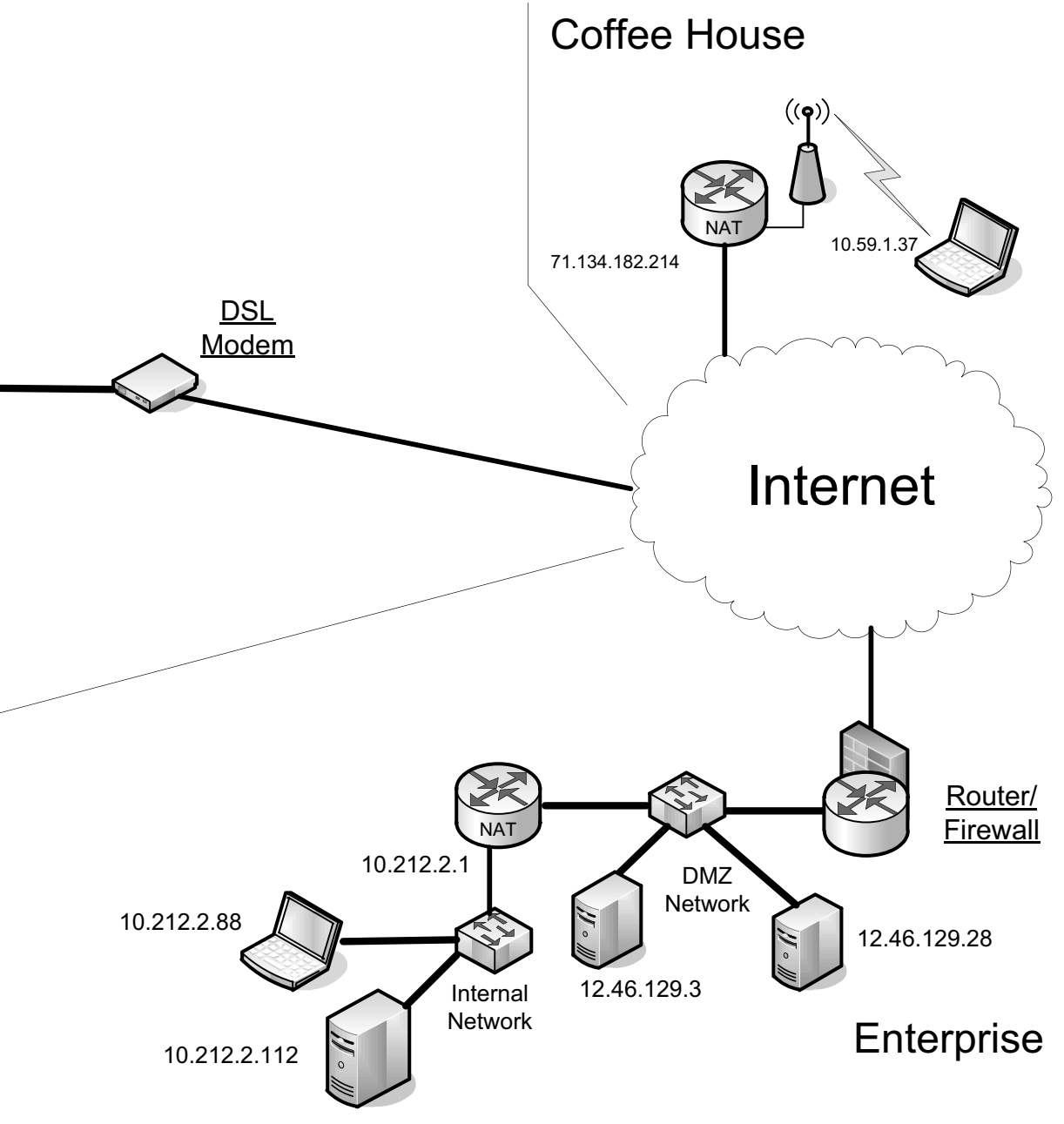
The Protocols

SECOND EDITION

Kevin R. Fall
W. Richard Stevens

ADDISON-WESLEY PROFESSIONAL COMPUTING SERIES


Public IP Addresses:
63.204.134.177
63.203.72.138
71.141.244.213
70.231.159.254
70.231.136.162
70.231.128.151
70.231.141.59
70.231.143.234
71.141.227.30
192.168.0.2
192.168.1.1
192.168.0.93
PPP / Ethernet
Test Connection
Wi-Fi
Access Point
192.168.0.1
10.0.0.1
Linux PC
(router, firewall, DHCP,
NAT, DNS)
Dynamic
IP Addresses
10.0.0.{15-126}
10.0.0.6
10.0.0.2
Dynamic
IP Addresses
10.0.1.{1-50}
switch
10.0.0.14
10.0.0.{7-13}
10.0.0.3
10.0.0.4
Home


Coffee House
NAT
71.134.182.214
10.59.1.37
DSL
Modem
Internet
NAT
10.212.2.1
Router/
Firewall
DMZ
Network
10.212.2.88
Internal
Network
12.46.129.3
12.46.129.28
10.212.2.112
Enterprise

Praise for the First Edition of *TCP/IP Illustrated, Volume 1: The Protocols*

"This is sure to be the bible for TCP/IP developers and users. Within minutes of picking up the text, I encountered several scenarios that had tripped up both my colleagues and myself in the past. Stevens reveals many of the mysteries once held tightly by the ever-elusive networking gurus. Having been involved in the implementation of TCP/IP for some years now, I consider this by far the finest text to date."

—Robert A. Ciampa, network engineer, Synernetics, division of 3COM

"While all of Stevens' books are readable and technically excellent, this new opus is awesome. Although many books describe the TCP/IP protocols, Stevens provides a level of depth and real-world detail lacking from the competition. He puts the reader inside TCP/IP using a visual approach and shows the protocols in action."

—Steven Baker, networking columnist, *Unix Review*

"*TCP/IP Illustrated, Volume 1*, is an excellent reference for developers, network administrators, or anyone who needs to understand TCP/IP technology. *TCP/IP Illustrated* is comprehensive in its coverage of TCP/IP topics, providing enough details to satisfy the experts while giving enough background and commentary for the novice."

—Bob Williams, vice president, Marketing, NetManage, Inc.

". . . [T]he difference is that Stevens wants to show as well as tell about the protocols. His principal teaching tools are straightforward explanations, exercises at the ends of chapters, byte-by-byte diagrams of headers and the like, and listings of actual traffic as examples."

—Walter Zintz, *UnixWorld*

"Much better than theory only. . . . W. Richard Stevens takes a multihost-based configuration and uses it as a travelogue of TCP/IP examples with illustrations. *TCP/IP Illustrated, Volume 1*, is based on practical examples that reinforce the theory—distinguishing this book from others on the subject, and making it both readable and informative."

—Peter M. Haverlock, consultant, IBM TCP/IP Development

"The diagrams he uses are excellent and his writing style is clear and readable. In sum, Stevens has made a complex topic easy to understand. This book merits everyone's attention. Please read it and keep it on your bookshelf."

—Elizabeth Zinkann, sys admin

"W. Richard Stevens has produced a fine text and reference work. It is well organized and very clearly written with, as the title suggests, many excellent illustrations exposing the intimate details of the logic and operation of IP, TCP, and the supporting cast of protocols and applications."

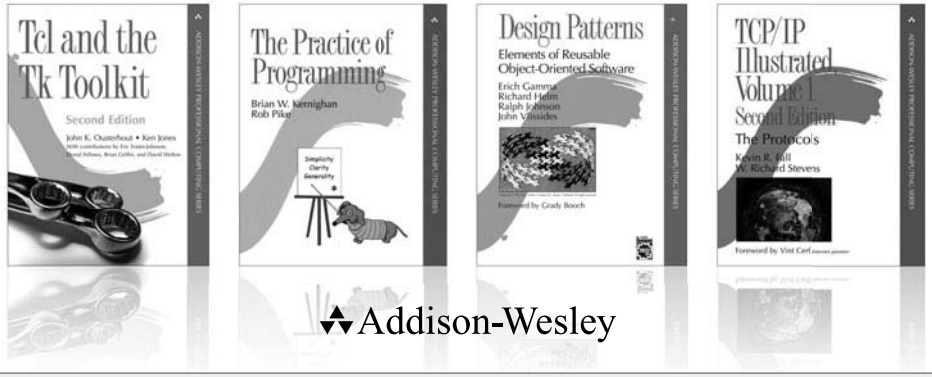
—Scott Bradner, consultant, Harvard University OIT/NSD

This page intentionally left blank

TCP/IP Illustrated, Volume 1

Second Edition

The Addison-Wesley Professional Computing Series



Visit informit.com/series/professionalcomputing
for a complete list of available publications.

The Addison-Wesley Professional Computing Series was created in 1990 to provide serious programmers and networking professionals with well-written and practical reference books. There are few places to turn for accurate and authoritative books on current and cutting-edge technology. We hope that our books will help you understand the state of the art in programming languages, operating systems, and networks.

Consulting Editor Brian W. Kernighan

Make sure to connect with us!
informit.com/socialconnect

Addison Wesley | informIT.com THE TRUSTED TECHNOLOGY LEARNING SOURCE | Safari Books Online

ALWAYS LEARNING PEARSON

TCP/IP Illustrated, Volume 1

The Protocols

Second Edition

Kevin R. Fall
W. Richard Stevens

Originally written by Dr. W. Richard Stevens.
Revised by Dr. Kevin R. Fall.

Addison-Wesley

Upper Saddle River, NJ • Boston • Indianapolis • San Francisco
New York • Toronto • Montreal • London • Munich • Paris • Madrid
Capetown • Sydney • Tokyo • Singapore • Mexico City

Many of the designations used by manufacturers and sellers to distinguish their products are claimed as trademarks. Where those designations appear in this book, and the publisher was aware of a trademark claim, the designations have been printed with initial capital letters or in all capitals.

The authors and publisher have taken care in the preparation of this book, but make no expressed or implied warranty of any kind and assume no responsibility for errors or omissions. No liability is assumed for incidental or consequential damages in connection with or arising out of the use of the information or programs contained herein.

The publisher offers excellent discounts on this book when ordered in quantity for bulk purchases or special sales, which may include electronic versions and/or custom covers and content particular to your business, training goals, marketing focus, and branding interests. For more information, please contact:

U.S. Corporate and Government Sales
(800) 382-3419
corpsales@pearsontechgroup.com

For sales outside the United States, please contact:

International Sales
international@pearson.com

Visit us on the Web: informit.com/aw

Library of Congress Cataloging-in-Publication Data
Fall, Kevin R.
TCP/IP illustrated.—2nd ed. / Kevin R. Fall, W. Richard Stevens.
p. cm.
Stevens' name appears first on the earlier edition.
Includes bibliographical references and index.
ISBN-13: 978-0-321-33631-6 (v. 1 : hardcover : alk. paper)
ISBN-10: 0-321-33631-3 (v. 1 : hardcover : alk. paper) 1. TCP/IP (Computer network protocol)
I. Stevens, W. Richard. II. Title.
TK5105.55.S74 2012
004.6'2—dc23

2011029411

Copyright © 2012 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. To obtain permission to use material from this work, please submit a written request to Pearson Education, Inc., Permissions Department, One Lake Street, Upper Saddle River, New Jersey 07458, or you may fax your request to (201) 236-3290.

ISBN-13: 978-0-321-33631-6
ISBN-10: 0-321-33631-3
Text printed in the United States on recycled paper at Edwards Brothers in Ann Arbor, Michigan.
Second printing, May 2012

To Vicki, George, Audrey, Maya, Dylan, and Jan,
for their insight, tolerance, and support
through the long nights and weekends.

—Kevin

This page intentionally left blank

Contents

Foreword xxv

Preface to the Second Edition xxvii

Adapted Preface to the First Edition xxxiii

Chapter 1 Introduction 1

1.1 Architectural Principles 2
1.1.1 Packets, Connections, and Datagrams 3
1.1.2 The End-to-End Argument and Fate Sharing 6
1.1.3 Error Control and Flow Control 7
1.2 Design and Implementation 8
1.2.1 Layering 8
1.2.2 Multiplexing, Demultiplexing, and Encapsulation in Layered Implementations 10
1.3 The Architecture and Protocols of the TCP/IP Suite 13
1.3.1 The ARPANET Reference Model 13
1.3.2 Multiplexing, Demultiplexing, and Encapsulation in TCP/IP 16
1.3.3 Port Numbers 17
1.3.4 Names, Addresses, and the DNS 19
1.4 Internets, Intranets, and Extranets 19
1.5 Designing Applications 20
1.5.1 Client/Server 20
1.5.2 Peer-to-Peer 21
1.5.3 Application Programming Interfaces (APIs) 22

1.6 Standardization Process 22
1.6.1 Request for Comments (RFC) 23
1.6.2 Other Standards 24
1.7 Implementations and Software Distributions 24
1.8 Attacks Involving the Internet Architecture 25
1.9 Summary 26
1.10 References 28

Chapter 2 The Internet Address Architecture 31

2.1 Introduction 31
2.2 Expressing IP Addresses 32
2.3 Basic IP Address Structure 34
2.3.1 Classful Addressing 34
2.3.2 Subnet Addressing 36
2.3.3 Subnet Masks 39
2.3.4 Variable-Length Subnet Masks (VLSM) 41
2.3.5 Broadcast Addresses 42
2.3.6 IPv6 Addresses and Interface Identifiers 43
2.4 CIDR and Aggregation 46
2.4.1 Prefixes 47
2.4.2 Aggregation 48
2.5 Special-Use Addresses 50
2.5.1 Addressing IPv4/IPv6 Translators 52
2.5.2 Multicast Addresses 53
2.5.3 IPv4 Multicast Addresses 54
2.5.4 IPv6 Multicast Addresses 57
2.5.5 Anycast Addresses 62
2.6 Allocation 62
2.6.1 Unicast 62
2.6.2 Multicast 65
2.7 Unicast Address Assignment 65
2.7.1 Single Provider/No Network/Single Address 66
2.7.2 Single Provider/Single Network/Single Address 67
2.7.3 Single Provider/Multiple Networks/Multiple Addresses 67
2.7.4 Multiple Providers/Multiple Networks/Multiple Addresses (Multihoming) 68

2.8 Attacks Involving IP Addresses 70
2.9 Summary 71
2.10 References 72

Chapter 3 Link Layer 79

3.1 Introduction 79
3.2 Ethernet and the IEEE 802 LAN/MAN Standards 80
3.2.1 The IEEE 802 LAN/MAN Standards 82
3.2.2 The Ethernet Frame Format 84
3.2.3 802.1p/q: Virtual LANs and QoS Tagging 89
3.2.4 802.1AX: Link Aggregation (Formerly 802.3ad) 92
3.3 Full Duplex, Power Save, Autonegotiation, and 802.1X Flow Control 94
3.3.1 Duplex Mismatch 96
3.3.2 Wake-on LAN (WoL), Power Saving, and Magic Packets 96
3.3.3 Link-Layer Flow Control 98
3.4 Bridges and Switches 98
3.4.1 Spanning Tree Protocol (STP) 102
3.4.2 802.1ak: Multiple Registration Protocol (MRP) 111
3.5 Wireless LANs—IEEE 802.11(Wi-Fi) 111
3.5.1 802.11 Frames 113
3.5.2 Power Save Mode and the Time Sync Function (TSF) 119
3.5.3 802.11 Media Access Control 120
3.5.4 Physical-Layer Details: Rates, Channels, and Frequencies 123
3.5.5 Wi-Fi Security 129
3.5.6 Wi-Fi Mesh (802.11s) 130
3.6 Point-to-Point Protocol (PPP) 130
3.6.1 Link Control Protocol (LCP) 131
3.6.2 Multilink PPP (MP) 137
3.6.3 Compression Control Protocol (CCP) 139
3.6.4 PPP Authentication 140
3.6.5 Network Control Protocols (NCPs) 141
3.6.6 Header Compression 142
3.6.7 Example 143
3.7 Loopback 145
3.8 MTU and Path MTU 148
3.9 Tunneling Basics 149
3.9.1 Unidirectional Links 153

3.10 Attacks on the Link Layer 154
3.11 Summary 156
3.12 References 157

Chapter 4 ARP: Address Resolution Protocol 165

4.1 Introduction 165
4.2 An Example 166
4.2.1 Direct Delivery and ARP 167
4.3 ARP Cache 169
4.4 ARP Frame Format 170
4.5 ARP Examples 171
4.5.1 Normal Example 171
4.5.2 ARP Request to a Nonexistent Host 173
4.6 ARP Cache Timeout 174
4.7 Proxy ARP 174
4.8 Gratuitous ARP and Address Conflict Detection (ACD) 175
4.9 The **`arp`** Command 177
4.10 Using ARP to Set an Embedded Device's IPv4 Address 178
4.11 Attacks Involving ARP 178
4.12 Summary 179
4.13 References 179

Chapter 5 The Internet Protocol (IP) 181

5.1 Introduction 181
5.2 IPv4 and IPv6 Headers 183
5.2.1 IP Header Fields 183
5.2.2 The Internet Checksum 186
5.2.3 *DS Field* and *ECN* (Formerly Called the *ToS Byte* or IPv6 *Traffic Class*) 188
5.2.4 IP Options 192
5.3 IPv6 Extension Headers 194
5.3.1 IPv6 Options 196
5.3.2 Routing Header 200
5.3.3 Fragment Header 203
5.4 IP Forwarding 208
5.4.1 Forwarding Table 208
5.4.2 IP Forwarding Actions 209

5.4.3 Examples 210
5.4.4 Discussion 215
5.5 Mobile IP 215
5.5.1 The Basic Model: Bidirectional Tunneling 216
5.5.2 Route Optimization (RO) 217
5.5.3 Discussion 220
5.6 Host Processing of IP Datagrams 220
5.6.1 Host Models 220
5.6.2 Address Selection 222
5.7 Attacks Involving IP 226
5.8 Summary 226
5.9 References 228

Chapter 6 System Configuration: DHCP and Autoconfiguration 233

6.1 Introduction 233
6.2 Dynamic Host Configuration Protocol (DHCP) 234
6.2.1 Address Pools and Leases 235
6.2.2 DHCP and BOOTP Message Format 236
6.2.3 DHCP and BOOTP Options 238
6.2.4 DHCP Protocol Operation 239
6.2.5 DHCPv6 252
6.2.6 Using DHCP with Relays 267
6.2.7 DHCP Authentication 271
6.2.8 Reconfigure Extension 273
6.2.9 Rapid Commit 273
6.2.10 Location Information (LCI and LoST) 274
6.2.11 Mobility and Handoff Information (MoS and ANDSF) 275
6.2.12 DHCP Snooping 276
6.3 Stateless Address Autoconfiguration (SLAAC) 276
6.3.1 Dynamic Configuration of IPv4 Link-Local Addresses 276
6.3.2 IPv6 SLAAC for Link-Local Addresses 276
6.4 DHCP and DNS Interaction 285
6.5 PPP over Ethernet (PPPoE) 286
6.6 Attacks Involving System Configuration 292
6.7 Summary 292
6.8 References 293

Chapter 7 Firewalls and Network Address Translation (NAT) 299

7.1 Introduction 299
7.2 Firewalls 300
7.2.1 Packet-Filtering Firewalls 300
7.2.2 Proxy Firewalls 301
7.3 Network Address Translation (NAT) 303
7.3.1 Traditional NAT: Basic NAT and NAPT 305
7.3.2 Address and Port Translation Behavior 311
7.3.3 Filtering Behavior 313
7.3.4 Servers behind NATs 314
7.3.5 Hairpinning and NAT Loopback 314
7.3.6 NAT Editors 315
7.3.7 Service Provider NAT (SPNAT) and Service Provider IPv6 Transition 315
7.4 NAT Traversal 316
7.4.1 Pinholes and Hole Punching 317
7.4.2 UNilateral Self-Address Fixing (UNSAF) 317
7.4.3 Session Traversal Utilities for NAT (STUN) 319
7.4.4 Traversal Using Relays around NAT (TURN) 326
7.4.5 Interactive Connectivity Establishment (ICE) 332
7.5 Configuring Packet-Filtering Firewalls and NATs 334
7.5.1 Firewall Rules 335
7.5.2 NAT Rules 337
7.5.3 Direct Interaction with NATs and Firewalls: UPnP, NAT-PMP, and PCP 338
7.6 NAT for IPv4/IPv6 Coexistence and Transition 339
7.6.1 Dual-Stack Lite (DS-Lite) 339
7.6.2 IPv4/IPv6 Translation Using NATs and ALGs 340
7.7 Attacks Involving Firewalls and NATs 345
7.8 Summary 346
7.9 References 347

Chapter 8 ICMPv4 and ICMPv6: Internet Control Message Protocol 353

8.1 Introduction 353
8.1.1 Encapsulation in IPv4 and IPv6 354
8.2 ICMP Messages 355
8.2.1 ICMPv4 Messages 356

8.2.2 ICMPv6 Messages 358
8.2.3 Processing of ICMP Messages 360
8.3 ICMP Error Messages 361
8.3.1 Extended ICMP and Multipart Messages 363
8.3.2 Destination Unreachable (ICMPv4 Type 3, ICMPv6 Type 1) and Packet Too Big (ICMPv6 Type 2) 364
8.3.3 Redirect (ICMPv4 Type 5, ICMPv6 Type 137) 372
8.3.4 ICMP Time Exceeded (ICMPv4 Type 11, ICMPv6 Type 3) 375
8.3.5 Parameter Problem (ICMPv4 Type 12, ICMPv6 Type 4) 379
8.4 ICMP Query/Informational Messages 380
8.4.1 Echo Request/Reply (`ping`) (ICMPv4 Types 0/8, ICMPv6 Types 129/128) 380
8.4.2 Router Discovery: Router Solicitation and Advertisement (ICMPv4 Types 9, 10) 383
8.4.3 Home Agent Address Discovery Request/Reply (ICMPv6 Types 144/145) 386
8.4.4 Mobile Prefix Solicitation/Advertisement (ICMPv6 Types 146/147) 387
8.4.5 Mobile IPv6 Fast Handover Messages (ICMPv6 Type 154) 388
8.4.6 Multicast Listener Query/Report/Done (ICMPv6 Types 130/131/132) 388
8.4.7 Version 2 Multicast Listener Discovery (MLDv2) (ICMPv6 Type 143) 390
8.4.8 Multicast Router Discovery (MRD) (IGMP Types 48/49/50, ICMPv6 Types 151/152/153) 394
8.5 Neighbor Discovery in IPv6 395
8.5.1 ICMPv6 Router Solicitation and Advertisement (ICMPv6 Types 133, 134) 396
8.5.2 ICMPv6 Neighbor Solicitation and Advertisement (IMCPv6 Types 135, 136) 398
8.5.3 ICMPv6 Inverse Neighbor Discovery Solicitation/Advertisement (ICMPv6 Types 141/142) 401
8.5.4 Neighbor Unreachability Detection (NUD) 402
8.5.5 Secure Neighbor Discovery (SEND) 403
8.5.6 ICMPv6 Neighbor Discovery (ND) Options 407
8.6 Translating ICMPv4 and ICMPv6 424
8.6.1 Translating ICMPv4 to ICMPv6 424
8.6.2 Translating ICMPv6 to ICMPv4 426
8.7 Attacks Involving ICMP 428

8.8 Summary 430
8.9 References 430

Chapter 9 Broadcasting and Local Multicasting (IGMP and MLD) 435

9.1 Introduction 435
9.2 Broadcasting 436
9.2.1 Using Broadcast Addresses 437
9.2.2 Sending Broadcast Datagrams 439
9.3 Multicasting 441
9.3.1 Converting IP Multicast Addresses to 802 MAC/Ethernet Addresses 442
9.3.2 Examples 444
9.3.3 Sending Multicast Datagrams 446
9.3.4 Receiving Multicast Datagrams 447
9.3.5 Host Address Filtering 449
9.4 The Internet Group Management Protocol (IGMP) and Multicast Listener Discovery Protocol (MLD) 451
9.4.1 IGMP and MLD Processing by Group Members ("Group Member Part") 454
9.4.2 IGMP and MLD Processing by Multicast Routers ("Multicast Router Part") 457
9.4.3 Examples 459
9.4.4 Lightweight IGMPv3 and MLDv2 464
9.4.5 IGMP and MLD Robustness 465
9.4.6 IGMP and MLD Counters and Variables 467
9.4.7 IGMP and MLD Snooping 468
9.5 Attacks Involving IGMP and MLD 469
9.6 Summary 470
9.7 References 471

Chapter 10 User Datagram Protocol (UDP) and IP Fragmentation 473

10.1 Introduction 473
10.2 UDP Header 474
10.3 UDP Checksum 475
10.4 Examples 478
10.5 UDP and IPv6 481
10.5.1 Teredo: Tunneling IPv6 through IPv4 Networks 482

10.6 UDP-Lite 487
10.7 IP Fragmentation 488
10.7.1 Example: UDP/IPv4 Fragmentation 488
10.7.2 Reassembly Timeout 492
10.8 Path MTU Discovery with UDP 493
10.8.1 Example 493
10.9 Interaction between IP Fragmentation and ARP/ND 496
10.10 Maximum UDP Datagram Size 497
10.10.1 Implementation Limitations 497
10.10.2 Datagram Truncation 498
10.11 UDP Server Design 498
10.11.1 IP Addresses and UDP Port Numbers 499
10.11.2 Restricting Local IP Addresses 500
10.11.3 Using Multiple Addresses 501
10.11.4 Restricting Foreign IP Address 502
10.11.5 Using Multiple Servers per Port 503
10.11.6 Spanning Address Families: IPv4 and IPv6 504
10.11.7 Lack of Flow and Congestion Control 505
10.12 Translating UDP/IPv4 and UDP/IPv6 Datagrams 505
10.13 UDP in the Internet 506
10.14 Attacks Involving UDP and IP Fragmentation 507
10.15 Summary 508
10.16 References 508

Chapter 11 Name Resolution and the Domain Name System (DNS) 511

11.1 Introduction 511
11.2 The DNS Name Space 512
11.2.1 DNS Naming Syntax 514
11.3 Name Servers and Zones 516
11.4 Caching 517
11.5 The DNS Protocol 518
11.5.1 DNS Message Format 520
11.5.2 The DNS Extension Format (EDNS0) 524
11.5.3 UDP or TCP 525
11.5.4 Question (Query) and Zone Section Format 526
11.5.5 Answer, Authority, and Additional Information Section Formats 526
11.5.6 Resource Record Types 527

11.5.7 Dynamic Updates (DNS UPDATE) 555
11.5.8 Zone Transfers and DNS NOTIFY 558
11.6 Sort Lists, Round-Robin, and Split DNS 565
11.7 Open DNS Servers and DynDNS 567
11.8 Transparency and Extensibility 567
11.9 Translating DNS from IPv4 to IPv6 (DNS64) 568
11.10 LLMNR and mDNS 569
11.11 LDAP 570
11.12 Attacks on the DNS 571
11.13 Summary 572
11.14 References 573

Chapter 12 TCP: The Transmission Control Protocol (Preliminaries) 579

12.1 Introduction 579
12.1.1 ARQ and Retransmission 580
12.1.2 Windows of Packets and Sliding Windows 581
12.1.3 Variable Windows: Flow Control and Congestion Control 583
12.1.4 Setting the Retransmission Timeout 584
12.2 Introduction to TCP 584
12.2.1 The TCP Service Model 585
12.2.2 Reliability in TCP 586
12.3 TCP Header and Encapsulation 587
12.4 Summary 591
12.5 References 591

Chapter 13 TCP Connection Management 595

13.1 Introduction 595
13.2 TCP Connection Establishment and Termination 595
13.2.1 TCP Half-Close 598
13.2.2 Simultaneous Open and Close 599
13.2.3 Initial Sequence Number (ISN) 601
13.2.4 Example 602
13.2.5 Timeout of Connection Establishment 604
13.2.6 Connections and Translators 605
13.3 TCP Options 605
13.3.1 Maximum Segment Size (MSS) Option 606

13.3.2 Selective Acknowledgment (SACK) Options 607
13.3.3 Window Scale (WSCALE or WSOPT) Option 608
13.3.4 Timestamps Option and Protection against Wrapped Sequence Numbers (PAWS) 608
13.3.5 User Timeout (UTO) Option 611
13.3.6 Authentication Option (TCP-AO) 612
13.4 Path MTU Discovery with TCP 612
13.4.1 Example 613
13.5 TCP State Transitions 616
13.5.1 TCP State Transition Diagram 617
13.5.2 TIME_WAIT (2MSL Wait) State 618
13.5.3 Quiet Time Concept 624
13.5.4 FIN_WAIT_2 State 625
13.5.5 Simultaneous Open and Close Transitions 625
13.6 Reset Segments 625
13.6.1 Connection Request to Nonexistent Port 626
13.6.2 Aborting a Connection 627
13.6.3 Half-Open Connections 628
13.6.4 TIME-WAIT Assassination (TWA) 630
13.7 TCP Server Operation 631
13.7.1 TCP Port Numbers 632
13.7.2 Restricting Local IP Addresses 634
13.7.3 Restricting Foreign Endpoints 635
13.7.4 Incoming Connection Queue 636
13.8 Attacks Involving TCP Connection Management 640
13.9 Summary 642
13.10 References 643

Chapter 14 TCP Timeout and Retransmission 647

14.1 Introduction 647
14.2 Simple Timeout and Retransmission Example 648
14.3 Setting the Retransmission Timeout (RTO) 651
14.3.1 The Classic Method 651
14.3.2 The Standard Method 652
14.3.3 The Linux Method 657
14.3.4 RTT Estimator Behaviors 661
14.3.5 RTTM Robustness to Loss and Reordering 662

14.4 Timer-Based Retransmission 664
14.4.1 Example 665
14.5 Fast Retransmit 667
14.5.1 Example 668
14.6 Retransmission with Selective Acknowledgments 671
14.6.1 SACK Receiver Behavior 672
14.6.2 SACK Sender Behavior 673
14.6.3 Example 673
14.7 Spurious Timeouts and Retransmissions 677
14.7.1 Duplicate SACK (DSACK) Extension 677
14.7.2 The Eifel Detection Algorithm 679
14.7.3 Forward-RTO Recovery (F-RTO) 680
14.7.4 The Eifel Response Algorithm 680
14.8 Packet Reordering and Duplication 682
14.8.1 Reordering 682
14.8.2 Duplication 684
14.9 Destination Metrics 685
14.10 Repacketization 686
14.11 Attacks Involving TCP Retransmission 687
14.12 Summary 688
14.13 References 689

Chapter 15 TCP Data Flow and Window Management 691

15.1Introduction 691

15.2 Interactive Communication 692
15.3 Delayed Acknowledgments 695
15.4 Nagle Algorithm 696
15.4.1 Delayed ACK and Nagle Algorithm Interaction 699
15.4.2 Disabling the Nagle Algorithm 699
15.5 Flow Control and Window Management 700
15.5.1 Sliding Windows 701
15.5.2 Zero Windows and the TCP Persist Timer 704
15.5.3 Silly Window Syndrome (SWS) 708
15.5.4 Large Buffers and Auto-Tuning 715
15.6 Urgent Mechanism 719
15.6.1 Example 720
15.7 Attacks Involving Window Management 723

15.8 Summary 723
15.9 References 724

Chapter 16 TCP Congestion Control 727

16.1 Introduction 727
16.1.1 Detection of Congestion in TCP 728
16.1.2 Slowing Down a TCP Sender 729
16.2 The Classic Algorithms 730
16.2.1 Slow Start 732
16.2.2 Congestion Avoidance 734
16.2.3 Selecting between Slow Start and Congestion Avoidance 736
16.2.4 Tahoe, Reno, and Fast Recovery 737
16.2.5 Standard TCP 738
16.3 Evolution of the Standard Algorithms 739
16.3.1 NewReno 739
16.3.2 TCP Congestion Control with SACK 740
16.3.3 Forward Acknowledgment (FACK) and Rate Halving 741
16.3.4 Limited Transmit 742
16.3.5 Congestion Window Validation (CWV) 742
16.4 Handling Spurious RTOs—the Eifel Response Algorithm 744
16.5 An Extended Example 745
16.5.1 Slow Start Behavior 749
16.5.2 Sender Pause and Local Congestion (Event 1) 750
16.5.3 Stretch ACKs and Recovery from Local Congestion 754
16.5.4 Fast Retransmission and SACK Recovery (Event 2) 757
16.5.5 Additional Local Congestion and Fast Retransmit Events 759
16.5.6 Timeouts, Retransmissions, and Undoing *cwnd* Changes 762
16.5.7 Connection Completion 766
16.6 Sharing Congestion State 767
16.7 TCP Friendliness 768
16.8 TCP in High-Speed Environments 770
16.8.1 HighSpeed TCP (HSTCP) and Limited Slow Start 770
16.8.2 Binary Increase Congestion Control (BIC and CUBIC) 772
16.9 Delay-Based Congestion Control 777
16.9.1 Vegas 777
16.9.2 FAST 778

16.9.3 TCP Westwood and Westwood+ 779
16.9.4 Compound TCP 779
16.10 Buffer Bloat 781
16.11 Active Queue Management and ECN 782
16.12 Attacks Involving TCP Congestion Control 785
16.13 Summary 786
16.14 References 788

Chapter 17 TCP Keepalive 793

17.1 Introduction 793
17.2 Description 795
17.2.1 Keepalive Examples 797
17.3 Attacks Involving TCP Keepalives 802
17.4 Summary 802
17.5 References 803

Chapter 18 Security: EAP, IPsec, TLS, DNSSEC, and DKIM 805

18.1 Introduction 805
18.2 Basic Principles of Information Security 806
18.3 Threats to Network Communication 807
18.4 Basic Cryptography and Security Mechanisms 809
18.4.1 Cryptosystems 809
18.4.2 Rivest, Shamir, and Adleman (RSA) Public Key Cryptography 812
18.4.3 Diffie-Hellman-Merkle Key Agreement (aka Diffie-Hellman or DH) 813
18.4.4 Signcryption and Elliptic Curve Cryptography (ECC) 814
18.4.5 Key Derivation and Perfect Forward Secrecy (PFS) 815
18.4.6 Pseudorandom Numbers, Generators, and Function Families 815
18.4.7 Nonces and Salt 816
18.4.8 Cryptographic Hash Functions and Message Digests 817
18.4.9 Message Authentication Codes (MACs, HMAC, CMAC, and GMAC) 818
18.4.10 Cryptographic Suites and Cipher Suites 819
18.5 Certificates, Certificate Authorities (CAs), and PKIs 821
18.5.1 Public Key Certificates, Certificate Authorities, and X.509 822
18.5.2 Validating and Revoking Certificates 828
18.5.3 Attribute Certificates 831

18.6 TCP/IP Security Protocols and Layering 832
18.7 Network Access Control: 802.1X, 802.1AE, EAP, and PANA 833
18.7.1 EAP Methods and Key Derivation 837
18.7.2 The EAP Re-authentication Protocol (ERP) 839
18.7.3 Protocol for Carrying Authentication for Network Access (PANA) 839
18.8 Layer 3 IP Security (IPsec) 840
18.8.1 Internet Key Exchange (IKEv2) Protocol 842
18.8.2 Authentication Header (AH) 854
18.8.3 Encapsulating Security Payload (ESP) 858
18.8.4 Multicast 864
18.8.5 L2TP/IPsec 865
18.8.6 IPsec NAT Traversal 865
18.8.7 Example 867
18.9 Transport Layer Security (TLS and DTLS) 876
18.9.1 TLS 1.2 877
18.9.2 TLS with Datagrams (DTLS) 891
18.10 DNS Security (DNSSEC) 894
18.10.1 DNSSEC Resource Records 896
18.10.2 DNSSEC Operation 902
18.10.3 Transaction Authentication (TSIG, TKEY, and SIG(0)) 911
18.10.4 DNSSEC with DNS64 915
18.11 DomainKeys Identified Mail (DKIM) 915
18.11.1 DKIM Signatures 916
18.11.2 Example 916
18.12 Attacks on Security Protocols 918
18.13 Summary 919
18.14 References 922

Glossary of Acronyms **933**

Index **963**

This page intentionally left blank

Foreword

Rarely does one find a book on a well-known topic that is both historically and technically comprehensive and remarkably accurate. One of the things I admire about this work is the "warts and all" approach that gives it such credibility. The TCP/IP architecture is a product of the time in which it was conceived. That it has been able to adapt to growing requirements in many dimensions by factors of a million or more, to say nothing of a plethora of applications, is quite remarkable. Understanding the scope and limitations of the architecture and its protocols is a sound basis from which to think about future evolution and even revolution.

During the early formulation of the Internet architecture, the notion of "enterprise" was not really recognized. In consequence, most networks had their own IP address space and "announced" their addresses in the routing system directly. After the introduction of commercial service, Internet Service Providers emerged as intermediaries who "announced" Internet address blocks on behalf of their customers. Thus, most of the address space was assigned in a "provider dependent" fashion. "Provider independent" addressing was unusual. The net result (no pun intended) led to route aggregation and containment of the size of the global routing table. While this tactic had benefits, it also created the "multi-homing" problem since users of provider-dependent addresses did not have their own entries in the global routing table. The IP address "crunch" also led to Network Address Translation, which also did not solve provider dependence and multi-homing problems.

Reading through this book evokes a sense of wonder at the complexity that has evolved from a set of relatively simple concepts that worked with a small number of networks and application circumstances. As the chapters unfold, one can see the level of complexity that has evolved to accommodate an increasing number of requirements, dictated in part by new deployment conditions and challenges, to say nothing of sheer growth in the scale of the system.

The issues associated with securing "enterprise" users of the Internet also led to firewalls that are intended to supply perimeter security. While useful, it has become clear that attacks against local Internet infrastructure can come through

internal compromises (e.g., an infected computer is put onto an internal network or an infected thumb-drive is used to infect an internal computer through its USB port).

It has become apparent that, in addition to a need to expand the Internet address space through the introduction of IP version 6, with its 340 trillion trillion trillion addresses, there is also a strong need to introduce various security-enhancing mechanisms such as the Domain Name System Security Extension (DNSSEC) among many others.

What makes this book unique, in my estimation, is the level of detail and attention to history. It provides background and a sense for the ways in which solutions to networking problems have evolved. It is relentless in its effort to achieve precision and to expose remaining problem areas. For an engineer determined to refine and secure Internet operation or to explore alternative solutions to persistent problems, the insights provided by this book will be invaluable. The authors deserve credit for a thorough rendering of the technology of today's Internet.

Woodhurst Vint Cerf
June 2011

Preface to the Second Edition

Welcome to the second edition of *TCP/IP Illustrated, Volume 1*. This book aims to provide a detailed, current look at the TCP/IP protocol suite. Instead of just describing how the protocols operate, we show the protocols in operation using a variety of analysis tools. This helps you better understand the design decisions behind the protocols and how they interact with each other, and it simultaneously exposes you to implementation details without your having to read through the implementation's software source code or set up an experimental laboratory. Of course, reading source code or setting up a laboratory will only help to increase your understanding.

Networking has changed dramatically in the past three decades. Originally a research project and object of curiosity, the Internet has become a global communication fabric upon which governments, businesses, and individuals depend. The TCP/IP suite defines the underlying methods used to exchange information by every device on the Internet. After more than a decade of delay, the Internet and TCP/IP itself are now undergoing an evolution, to incorporate IPv6. Throughout the text we will discuss both IPv6 and the current IPv4 together, but we highlight the differences where they are important. Unfortunately, they do not directly interoperate, so some care and attention are required to appreciate the impact of the evolution.

The book is intended for anyone wishing to better understand the current set of TCP/IP protocols and how they operate: network operators and administrators, network software developers, students, and users who deal with TCP/IP. We have included material that should be of interest to both new readers as well as those familiar with the material from the first edition. We hope you will find the coverage of the new and older material useful and interesting.

Comments on the First Edition

Nearly two decades have passed since the publication of the first edition of *TCP/IP Illustrated, Volume 1*. It continues to be a valuable resource for both students and professionals in understanding the TCP/IP protocols at a level of detail difficult to

obtain in competing texts. Today it remains among the best references for detailed information regarding the operation of the TCP/IP protocols. However, even the best books concerned with information and communications technology become dated after a time, and the TCP/IP Illustrated series is no exception. In this edition, I hope to thoroughly update the pioneering work of Dr. Stevens with coverage of new material while maintaining the exceptionally high standard of presentation and detail common to his numerous books.

The first edition covers a broad set of protocols and their operation, ranging from the link layer all the way to applications and network management. Today, covering this breadth of material comprehensively in a single volume would produce a very lengthy text indeed. For this reason, the second edition focuses specifically on the *core* protocols: those relatively low-level protocols used most frequently in providing the basic services of configuration, naming, data delivery, and security for the Internet. Detailed discussions of applications, routing, Web services, and other important topics are postponed to subsequent volumes.

Considerable progress has been made in improving the robustness and compliance of TCP/IP implementations to their corresponding specifications since the publication of the first edition. While many of the examples in the first edition highlight implementation bugs or noncompliant behaviors, these problems have largely been addressed in currently available systems, at least for IPv4. This fact is not terribly surprising, given the greatly expanded use of the TCP/IP protocols in the last 18 years. Misbehaving implementations are a comparative rarity, which attests to a certain maturity of the protocol suite as a whole. The problems encountered in the operation of the core protocols nowadays often relate to intentional exploitation of infrequently used protocol features, a form of security concern that was not a primary focus in the first edition but one that we spend considerable effort to address in the second edition.

The Internet Milieu of the Twenty-first Century

The usage patterns and importance of the Internet have changed considerably since the publication of the first edition. The most obvious watershed event was the creation and subsequent intense commercialization of the World Wide Web starting in the early 1990s. This event greatly accelerated the availability of the Internet to large numbers of people with various (sometimes conflicting) motivations. As such, the protocols and systems originally implemented in a small-scale environment of academic cooperation have been stressed by limited availability of addresses and an increase of security concerns.

In response to the security threats, network and security administrators have introduced special control elements into the network. It is now common practice to place a *firewall* at the point of attachment to the Internet, for both large enterprises as well as small businesses and homes. As the demand for IP addresses and security has increased over the last decade, *Network Address Translation* (NAT) is now supported in virtually all current-generation routers and is in widespread use. It

has eased the pressure on Internet address availability by allowing sites to obtain a comparatively small number of routable Internet addresses from their service providers (one for each simultaneously online user), yet assign a very large number of addresses to local computers without further coordination. A consequence of NAT deployment has been a slowing of the migration to IPv6 (which provides for an almost incomprehensibly large number of addresses) and interoperability problems with some older protocols.

As the users of personal computers began to demand Internet connectivity by the mid-1990s, the largest supplier of PC software, Microsoft, abandoned its original policy of offering only proprietary alternatives to the Internet and instead undertook an effort to embrace TCP/IP compatibility in most of its products. Since then, personal computers running their Windows operating system have come to dominate the mix of PCs presently connected to the Internet. Over time, a significant rise in the number of Linux-based systems means that such systems now threaten to displace Microsoft as the frontrunner. Other operating systems, including Oracle Solaris and Berkeley's BSD-based systems, which once represented the majority of Internet-connected systems, are now a comparatively small component of the mix. Apple's OS X (Mach-based) operating system has risen as a new contender and is gaining in popularity, especially among portable computer users. In 2003, portable computer (laptop) sales exceeded desktop sales as the majority of personal computer types sold, and their proliferation has sparked a demand for widely deployed, high-speed Internet access supported by wireless infrastructure. It is projected that the most common method for accessing the Internet from 2012 and beyond will be smartphones. Tablet computers also represent an important growing contender.

Wireless networks are now available at a large number of locations such as restaurants, airports, coffeehouses, and other public places. They typically provide short-range free or pay-for-use (flat-rate) high-speed wireless Internet connections using hardware compatible with commonly used office or home local area network installations. A set of alternative "wireless broadband" technologies based on cellular telephone standards (e.g., LTE, HSPA, UMTS, EV-DO) are becoming widely available in developed regions of the world (and some developing regions of the words that are "leapfrogging" to newer wireless technology), offering longer-range operation, often at somewhat reduced bandwidths and with volume-based pricing. Both types of infrastructure address the desire of users to be mobile while accessing the Internet, using either portable computers or smaller devices. In either case, mobile end users accessing the Internet over wireless networks pose two significant technical challenges to the TCP/IP protocol architecture. First, mobility affects the Internet's routing and addressing structure by breaking the assumption that hosts have addresses assigned to them based upon the identity of their nearby router. Second, wireless links may experience outages and therefore cause data to be lost for reasons other than those typical of wired links (which generally do not lose data unless too much traffic is being injected into the network).

Finally, the Internet has fostered the rise of so-called peer-to-peer applications forming "overlay" networks. Peer-to-peer applications do not rely on a central server to accomplish a task but instead determine a set of *peer* computers with which they can communicate and interact to accomplish a task. The peer computers are operated by other end users and may come and go rapidly compared to a fixed server infrastructure. The "overlay" concept captures the fact that such interacting peers themselves form a network, overlaid atop the conventional TCP/IP-based network (which, one may observe, is itself an overlay above the underlying physical links). The development of peer-to-peer applications, while of intense interest to those who study traffic flows and electronic commerce, has not had a profound impact on the core protocols described in *Volume 1* per se, but the concept of overlay networks has become an important consideration for networking technology more generally.

Content Changes for the Second Edition

Regarding content in the text, the most important changes from the first edition are a restructuring of the scope of the overall text and the addition of significant material on security. Instead of attempting to cover nearly all common protocols in use at every layer in the Internet, the present text focuses in detail first on the non-security core protocols in widespread use, or that are expected to be in widespread use in the near future: Ethernet (802.3), Wi-Fi (802.11), PPP, ARP, IPv4, IPv6, UDP, TCP, DHCP, and DNS. These protocols are likely to be encountered by system administrators and users alike.

In the second edition, security is covered in two ways. First, in each appropriate chapter, a section devoted to describing known attacks and their countermeasures relating to the protocol described in the chapter is included. These descriptions are not presented as a recipe for constructing attacks but rather as a practical indication of the kinds of problems that may arise when protocol implementations (or specifications, in some cases) are insufficiently robust. In today's Internet, incomplete specification or lax implementation practice can lead to mission-critical systems being compromised by even relatively unsophisticated attacks.

The second important discussion of security occurs in Chapter 18, where security and cryptography are studied in some detail, including protocols such as IPsec, TLS, DNSSEC, and DKIM. These protocols are now understood to be important for implementing any service or application expected to maintain integrity or secure operation. As the Internet has increased in commercial importance, the need for security (and the number of threats to it) has grown proportionally.

Although IPv6 was not included in the first edition, there is now reason to believe that the use of IPv6 may increase significantly with the exhaustion of unallocated IPv4 address groups in February 2011. IPv6 was conceived largely to address the problems of IPv4 address depletion and, and while not nearly as common as IPv4 today, is becoming more important as a growing number of small devices (such as cellular telephones, household devices, and environmental

sensors) become attached to the Internet. Events such as the World IPv6 Day (June 8, 2011) helped to demonstrate that the Internet can continue to work even as the underlying protocols are modified and augmented in a significant way.

A second consideration for the structure of the second edition is a deemphasis of the protocols that are no longer commonly used and an update of the descriptions of those that have been revised substantially since the publication of the first edition. The chapters covering RARP, BOOTP, NFS, SMTP, and SNMP have been removed from the book, and the discussion of the SLIP protocol has been abandoned in favor of expanded coverage of DHCP and PPP (including PPPoE). The function of IP forwarding (described in Chapter 9 in the first edition) has been integrated with the overall description of the IPv4 and IPv6 protocols in Chapter 5 of this edition. The discussion of dynamic routing protocols (RIP, OSPF, and BGP) has been removed, as the latter two protocols alone could each conceivably merit a book-long discussion. Starting with ICMP, and continuing through IP, TCP, and UDP, the impact of operation using IPv4 versus IPv6 is discussed in any cases where the difference in operation is significant. There is no specific chapter devoted solely to IPv6; instead, its impact relative to each existing core protocol is described where appropriate. Chapters 15 and 25–30 of the first edition, which are devoted to Internet applications and their supporting protocols, have been largely removed; what remains only illustrates the operation of the underlying core protocols where necessary.

Several chapters covering new material have been added. The first chapter begins with a general introduction to networking issues and architecture, followed by a more Internet-specific orientation. The Internet's addressing architecture is covered in Chapter 2. A new chapter on host configuration and how a system "gets on" the network appears as Chapter 6. Chapter 7 describes firewalls and Network Address Translation (NAT), including how NATs are used in partitioning address space between routable and nonroutable portions. The set of tools used in the first edition has been expanded to include Wireshark (a free network traffic monitor application with a graphical user interface).

The target readership for the second edition remains identical to that of the first edition. No prior knowledge of networking concepts is required for approaching it, although the advanced reader should benefit from the level of detail and references. A rich collection of references is included in each chapter for the interested reader to pursue.

Editorial Changes for the Second Edition

The general flow of material in the second edition remains similar to that of the first edition. After the introductory material (Chapters 1 and 2), the protocols are presented in a bottom-up fashion to illustrate how the goal of network communication presented in the introduction is realized in the Internet architecture. As in the first edition, actual packet traces are used to illustrate the operational details of the protocols, where appropriate. Since the publication of the first edition, freely

available packet capture and analysis tools with graphical interfaces have become available, extending the capabilities of the `tcpdump` program used in the first edition. In the present text, `tcpdump` is used when the points to be illustrated are easily conveyed by examining the output of a text-based packet capture tool. In most other cases, however, screen shots of the Wireshark tool are used. Please be aware that some output listings, including snapshots of `tcpdump` output, are wrapped or simplified for clarity.

The packet traces shown typically illustrate the behavior of one or more parts of the network depicted on the inside of the front book cover. It represents a broadband-connected "home" environment (typically used for client access or peer-to-peer networking), a "public" environment (e.g., coffee shop), and an enterprise environment. The operating systems used for examples include Linux, Windows, FreeBSD, and Mac OS X. Various versions are used, as many different OS versions are in use on the Internet today.

The structure of each chapter has been slightly modified from the first edition. Each chapter begins with an introduction to the chapter topic, followed in some cases by historical notes, the details of the chapter, a summary, and a set of references. A section near the end of most chapters describes security concerns and attacks. The per-chapter references represent a change for the second edition. They should make each chapter more self-contained and require the reader to perform fewer "long-distance page jumps" to find a reference. Some of the references are now enhanced with WWW URLs for easier access online. In addition, the reference format for papers and books has been changed to a somewhat more compact form that includes the first initial of each author's last name followed by the last two digits of the year (e.g., the former [Cerf and Kahn 1974] is now shortened to [CK74]). For the numerous RFC references used, the RFC number is used instead of the author names. This follows typical RFC conventions and has the side benefit of grouping all the RFC references together in the reference lists.

On a final note, the typographical conventions of the TCP/IP Illustrated series have been maintained faithfully. However, the present author elected to use an editor and typesetting package other than the Troff system used by Dr. Stevens and some other authors of the Addison-Wesley Professional Computing Series collection. Thus, the particular task of final copyediting could take advantage of the significant expertise of Barbara Wood, the copy editor generously made available to me by the publisher. We hope you will be pleased with the results.

Berkeley, California Kevin R. Fall
September 2011

Adapted Preface to the First Edition

Introduction

This book describes the TCP/IP protocol suite, but from a different perspective than other texts on TCP/IP. Instead of just describing the protocols and what they do, we'll use a popular diagnostic tool to watch the protocols in action. Seeing how the protocols operate in varying circumstances provides a greater understanding of how they work and why certain design decisions were made. It also provides a look into the implementation of the protocols, without having to wade through thousands of lines of source code.

When networking protocols were being developed in the 1960s through the 1980s, expensive, dedicated hardware was required to see the packets going "across the wire." Extreme familiarity with the protocols was also required to comprehend the packets displayed by the hardware. Functionality of the hardware analyzers was limited to that built in by the hardware designers.

Today this has changed dramatically with the ability of the ubiquitous workstation to monitor a local area network [Mogul 1990]. Just attach a workstation to your network, run some publicly available software, and watch what goes by on the wire. While many people consider this a tool to be used for *diagnosing* network problems, it is also a powerful tool for *understanding* how the network protocols operate, which is the goal of this book.

This book is intended for anyone wishing to understand how the TCP/IP protocols operate: programmers writing network applications, system administrators responsible for maintaining computer systems and networks utilizing TCP/IP, and users who deal with TCP/IP applications on a daily basis.

Typographical Conventions

When we display interactive input and output we'll show our typed input in a **bold font**, and the computer output like this. *Comments are added in italics.*

```
bsdi % telnet svr4 discard          connect to the discard server
Trying 140.252.13.34...             this line and next output by Telnet client
Connected to svr4.
```

Also, we always include the name of the system as part of the shell prompt (bsdi in this example) to show on which host the command was run.

Note

Throughout the text we'll use indented, parenthetical notes such as this to describe historical points or implementation details.

We sometimes refer to the complete description of a command on the Unix manual as in ifconfig(8). This notation, the name of the command followed by a number in parentheses, is the normal way of referring to Unix commands. The number in parentheses is the section number in the Unix manual of the "manual page" for the command, where additional information can be located. Unfortunately not all Unix systems organize their manuals the same, with regard to the section numbers used for various groupings of commands. We'll use the BSD-style section numbers (which is the same for BSD-derived systems such as SunOS 4.1.3), but your manuals may be organized differently.

Acknowledgments

Although the author's name is the only one to appear on the cover, the combined effort of many people is required to produce a quality text book. First and foremost is the author's family, who put up with the long and weird hours that go into writing a book. Thank you once again, Sally, Bill, Ellen, and David.

The consulting editor, Brian Kernighan, is undoubtedly the best in the business. He was the first one to read various drafts of the manuscript and mark it up with his infinite supply of red pens. His attention to detail, his continual prodding for readable prose, and his thorough reviews of the manuscript are an immense resource to a writer.

Technical reviewers provide a different point of view and keep the author honest by catching technical mistakes. Their comments, suggestions, and (most importantly) criticisms add greatly to the final product. My thanks to Steve Bellovin, Jon Crowcroft, Pete Haverlock, and Doug Schmidt for comments on the entire manuscript. Equally valuable comments were provided on portions of the manuscript by Dave Borman for his thorough review of all the TCP chapters, and to Bob Gilligan who should be listed as a coauthor for Appendix E.

An author cannot work in isolation, so I would like to thank the following persons for lots of small favors, especially by answering my numerous e-mail questions: Joe Godsil, Jim Hogue, Mike Karels, Paul Lucchina, Craig Partridge, Thomas Skibo, and Jerry Toporek.

This book is the result of my being asked lots of questions on TCP/IP for which I could find no quick, immediate answer. It was then that I realized that the easiest way to obtain the answers was to run small tests, forcing certain conditions to occur, and just watch what happens. I thank Peter Haverlock for asking the probing questions and Van Jacobson for providing so much of the publicly available software that is used in this book to answer the questions.

A book on networking needs a real network to work with along with access to the Internet. My thanks to the National Optical Astronomy Observatories (NOAO), especially Sidney Wolff, Richard Wolff, and Steve Grandi, for providing access to their networks and hosts. A special thanks to Steve Grandi for answering lots of questions and providing accounts on various hosts. My thanks also to Keith Bostic and Kirk McKusick at the U.C. Berkeley CSRG for access to the latest 4.4BSD system.

Finally, it is the publisher that pulls everything together and does whatever is required to deliver the final product to the readers. This all revolves around the editor, and John Wait is simply the best there is. Working with John and the rest of the professionals at Addison-Wesley is a pleasure. Their professionalism and attention to detail show in the end result.

Camera-ready copy of the book was produced by the author, a Troff die-hard, using the Groff package written by James Clark.

Tucson, Arizona W. Richard Stevens
October 1993

This page intentionally left blank

12

TCP: The Transmission Control Protocol (Preliminaries)

12.1 Introduction

So far we have been discussing protocols that do not include their own mechanisms for delivering data reliably. They may *detect* that erroneous data has been received, using a mathematical function such as a checksum or CRC, but they do not try very hard to repair errors. With IP and UDP, no error repair is done at all. With Ethernet and other protocols based on it, the protocol provides some number of retries and then gives up if it cannot succeed.

The problem of communicating in environments where the communication medium may lose or alter the messages being delivered has been studied for years. Some of the most important theoretical work on the topic was developed by Claude Shannon in 1948 [S48]. This work, which popularized the term bit and became the foundation of the field of *information theory*, helps us understand the fundamental limits on the amount of information that can be moved across an information channel that is *lossy* (that may delete or alter bits). Information theory is closely related to the field of *coding theory*, which provides ways of encoding information so that it is as resilient as possible to errors in the communications channel. Using *error-correcting codes* (basically, adding redundant bits so that the real information can be retrieved even if some bits are damaged) to correct communications problems is one very important method for handling errors. Another is to simply "try sending again" until the information is finally received. This approach, called *Automatic Repeat Request* (ARQ), forms the basis for many communications protocols, including TCP.

12.1.1 ARQ and Retransmission

If we consider not only a single communication channel but the multihop cascade of several, we realize that not only may we have the types of errors mentioned so far (packet bit errors), but there may be others. These problems might arise at an intermediate router and are the types of problems we brought up when discussing IP: packet reordering, packet duplication, and packet erasures (drops). An error-correcting protocol designed for use over a multihop communications channel (such as IP) must cope with all of these problems. Let us now explore the protocol mechanisms that can be brought to bear on them. After we discuss these in the abstract, we shall explore how they are used by TCP in the Internet.

A straightforward method of dealing with packet drops (and bit errors) is to resend the packet until it is received properly. This requires a way to determine (1) whether the receiver has received the packet and (2) whether the packet it received was the same one the sender sent. The method for a receiver to signal to a sender that it has received a packet is called an *acknowledgment*, or ACK. In its most basic form, the sender sends a packet and awaits an ACK. When the receiver receives the packet, it sends the ACK. When the sender receives the ACK, it sends another packet, and the process continues. Interesting questions to ask here are (1) How long should the sender wait for an ACK? (2) What if the ACK is lost? (3) What if the packet was received but had errors in it?

As we shall see, the first question turns out to be deep. Deciding how long to wait relates to how long the sender should *expect* to wait for an ACK. Determining this may be difficult; we postpone the discussion of techniques for it until we discuss TCP in detail later (see Chapter 14). The answer to question 2 is easier: if an ACK is dropped, the sender cannot readily distinguish this case from the case in which the original packet is dropped, so it simply sends the packet again. Of course, the receiver may receive two or more copies in that case, so it must be prepared to handle that situation (see the next paragraph). As for the third question, we can appeal to the codes mentioned in Section 12.1. It is generally much easier to use codes to *detect* errors in a large packet (with high probability) using only a few bits than it is to correct them. Simpler codes are typically not capable of correcting errors but are capable of detecting them. That is why checksums and CRCs are so popular. In order to detect errors in a packet, then, we use a form of checksum. When a receiver receives a packet containing an error, it refrains from sending an ACK. Eventually, the sender resends the packet, which ideally arrives undamaged.

Even with the simple scenario presented so far, there is the possibility that the receiver might receive *duplicate* copies of the packet being transferred. This problem is addressed using a *sequence number*. Basically, every unique packet gets a new sequence number when it is sent at the source, and this sequence number is carried along in the packet itself. The receiver can use this number to determine whether it has already seen the packet and if so, discard it.

The protocol described so far is reliable but not very efficient. Consider what happens when the time to deliver even a small packet from sender to receiver (the

delay or latency) is large (e.g., a second or two, which is not unusual for satellite links) and there are several packets to send. The sender is able to inject a single packet into the communications path but then must stop until it hears the ACK. This protocol is therefore called "stop and wait." Its throughput performance (data sent on the network per unit time) is proportional to M/R where M is the packet size and R is the round-trip time (RTT), assuming no packets are lost or irreparably damaged in transit. For a fixed-size packet, as R goes up, the throughput goes down. If packets are lost or damaged, the situation is even worse: the "goodput" (useful amount of data transferred per unit time) can be considerably less than the throughput.

For a network that doesn't damage or drop many packets, the cause for low throughput is usually that the network is not being kept busy. The situation is similar to using an assembly line where new work cannot enter the line until a complete product emerges. Most of the line goes idle. If we take this comparison one step further, it seems obvious that we would do better if we could have more than one work unit in the line at a time. It is the same for network communication—if we could have more than one packet in the network, we would keep it "more busy," leading to higher throughput.

Allowing more than one packet to be in the network at a time complicates matters considerably. Now the sender must decide not only when to inject a packet into the network, but also how many. It also must figure out how to keep the timers when waiting for ACKs, and it must keep a copy of each packet not yet acknowledged in case retransmissions are necessary. The receiver needs to have a more sophisticated ACK mechanism: one that can distinguish which packets have been received and which have not. The receiver may need a more sophisticated buffering (packet storage) mechanism—one that allows it to hold "out-of-sequence" packets (those packets that have arrived earlier than those expected because of loss or reordering), unless it simply wants to throw away such packets, which is very inefficient. There are other issues that may not be so obvious. What if the receiver is slower than the sender? If the sender simply injects many packets at a very high rate, the receiver might just drop them because of processing or memory limitations. The same question can be asked about the routers in the middle. What if the network infrastructure cannot handle the rate of data the sender and receiver wish to use?

12.1.2 Windows of Packets and Sliding Windows

To handle all of these problems, we begin with the assumption that each unique packet has a sequence number, as described earlier. We define a *window* of packets as the collection of packets (or their sequence numbers) that have been injected by the sender but not yet completely acknowledged (i.e., the sender has not received an ACK for them). We refer to the *window size* as the number of packets in the window. The term *window* comes from the idea that if you lined up all the packets sent during a communication session in a long row but had only a small aperture

through which to view them, you would see only a subset of them—like peering through a window. The sender's window (and the line of other packets) can be graphically depicted as shown in Figure 12-1.

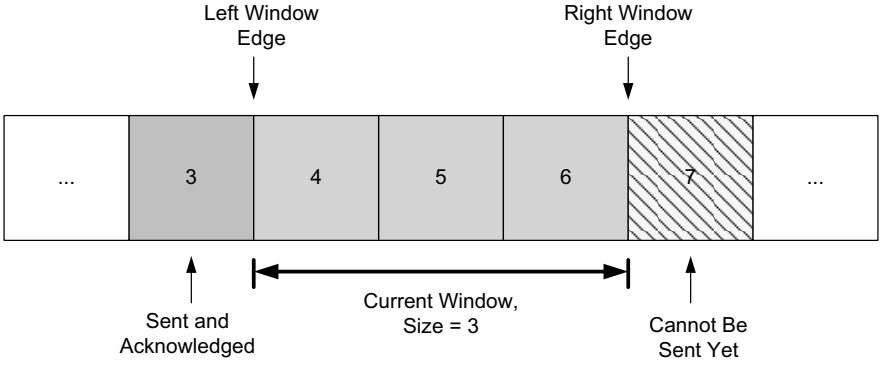


Figure 12-1 The sender's window, showing which packets are eligible to be sent (or have already been sent), which are not yet eligible, and which have already been sent and acknowledged. In this example, the window size is fixed at three packets.

This figure shows the current window of three packets, for a total window size of 3. Packet number 3 has already been sent and acknowledged, so the copy of it that the sender was keeping can now be released. Packet 7 is ready at the sender but not yet able to be sent because it is not yet "in" the window. If we now imagine that data starts to flow from the sender to the receiver and ACKs start to flow in the reverse direction, the sender might next receive an ACK for packet 4. When this happens, the window "slides" to the right by one packet, meaning that the copy of packet 4 can be released and packet 7 can be sent. This movement of the window gives rise to another name for this type of protocol, a *sliding window* protocol.

The sliding window approach can be used to combat many of the problems described so far. Typically, this window structure is kept at both the sender and the receiver. At the sender, it keeps track of what packets can be released, what packets are awaiting ACKs, and what packets cannot yet be sent. At the receiver, it keeps track of what packets have already been received and acknowledged, what packets are expected (and how much memory has been allocated to hold them), and which packets, even if received, will not be kept because of limited memory. Although the window structure is convenient for keeping track of data as it flows between sender and receiver, it does not provide guidance as to how large the window should be, or what happens if the receiver or network cannot handle the sender's data rate. We shall now see how these are related.

12.1.3 Variable Windows: Flow Control and Congestion Control

To handle the problem that arises when a receiver is too slow relative to a sender, we introduce a way to force the sender to slow down when the receiver cannot keep up. This is called *flow control* and is usually handled in one of two ways. One way, called *rate-based* flow control, gives the sender a certain data rate allocation and ensures that data is never allowed to be sent at a rate that exceeds the allocation. This type of flow control is most appropriate for streaming applications and can be used with broadcast and multicast delivery (see Chapter 9).

The other predominant form of flow control is called *window-based* flow control and is the most popular approach when sliding windows are being used. In this approach, the window size is not fixed but is instead allowed to vary over time. To achieve flow control using this technique, there must be a method for the receiver to signal the sender how large a window to use. This is typically called a *window advertisement*, or simply a *window update*. This value is used by the sender (i.e., the receiver of the window advertisement) to adjust its window size. Logically, a window update is separate from the ACKs we discussed previously, but in practice the window update and ACK are carried in a single packet, meaning that the sender tends to adjust the size of its window at the same time it slides it to the right.

If we consider the effect of changing the window size at the sender, it becomes clear how this achieves flow control. The sender is allowed to inject W packets into the network before it hears an ACK for any of them. If the sender and receiver are sufficiently fast, and the network loses no packets and has an infinite capacity, this means that the transfer rate is proportional to (SW/R) bits/s, where W is the window size, S is the packet size in bits, and R is the RTT. When the window advertisement from the receiver clamps the value of W at the sender, the sender's overall rate can be limited so as to not overwhelm the receiver. This approach works fine for protecting the receiver, but what about the network in between? We may have routers with limited memory between the sender and the receiver that have to contend with slow network links. When this happens, it is possible for the sender's rate to exceed a router's ability to keep up, leading to packet loss. This is addressed with a special form of flow control called *congestion control*.

Congestion control involves the sender slowing down so as to not overwhelm the network between itself and the receiver. Recall that in our discussion of flow control, we used a window advertisement to signal the sender to slow down for the receiver. This is called *explicit* signaling, because there is a protocol field specifically used to inform the sender about what is happening. Another option might be for the sender to *guess* that it needs to slow down. Such an approach would involve *implicit* signaling—that is, it would involve deciding to slow down based on some other evidence.

The problem of congestion control in datagram-style networks, and more generally *queuing theory* to which it is closely related, has remained a major research topic for years, and it is unlikely to ever be solved completely for all circumstances. It is also not practical to discuss all the options and methods of performing flow

control here. The interested reader is referred to [J90], [K97], and [K75]. In Chapter 16 we will explore the particular congestion control technique used with TCP in more detail, along with a number of variants that have arisen over the years.

12.1.4 Setting the Retransmission Timeout

One of the most important performance issues the designer of a retransmission-based reliable protocol faces is how long to wait before concluding that a packet has been lost and should be resent. Stated another way, What should the retransmission timeout be? Intuitively, the amount of time the sender should wait before resending a packet is about the sum of the following times: the time to send the packet, the time for the receiver to process it and send an ACK, the time for the ACK to travel back to the sender, and the time for the sender to process the ACK. Unfortunately, in practice, none of these times are known with certainty. To make matters worse, any or all of them vary over time as additional load is added to or removed from the end hosts or routers.

Because it is not practical for the user to tell the protocol implementation what the values of all the times are (or to keep them up-to-date) for all circumstances, a better strategy is to have the protocol implementation try to estimate them. This is called *round-trip-time estimation* and is a statistical process. Basically, the true RTT is likely to be close to the sample mean of a collection of samples of RTTs. Note that this average naturally changes over time (it is not stationary), as the paths taken through the network may change.

Once some estimate of the RTT is made, the question of setting the actual timeout value, used to trigger retransmissions, remains. If we recall the definition of a mean, it can never be the extreme value of a set of samples (unless they are all the same). So, it would not be sensible to set the retransmission timer to be exactly equal to the mean estimator, as it is likely that many actual RTTs will be larger, thereby inducing unwanted retransmissions. Clearly, the timeout should be set to something larger than the mean, but exactly what this relationship is (or even if the mean should be directly used) is not yet clear. Setting the timeout too large is also undesirable, as this leads back to letting the network go idle, reducing throughput. We shall defer further exploration of this topic to Chapter 14, where we explore how TCP, in particular, approaches this problem.

12.2 Introduction to TCP

Given the background we now have regarding the issues affecting reliable delivery in general, let us see how they play out in TCP and what type of service it provides to Internet applications. We also look at the fields in the TCP header, noticing how many of the concepts we have seen so far (e.g., ACKs, window advertisements) are captured in the header description. In the chapters that follow, we examine all of these header fields in more detail.

Our description of TCP starts in this chapter and continues in the next five chapters. Chapter 13 describes how a TCP connection is established and terminated. Chapter 14 details how TCP estimates the per-connection RTT and how the retransmission timeout is set based on this estimate. Chapter 15 looks at the normal transfer of data, starting with "interactive" applications (such as chat). It then covers window management and flow control, which apply to both interactive and "bulk" data flow applications (such as file transfer), along with TCP's *urgent mechanism*, which allows a sender to mark certain data in the data stream as special. Chapter 16 takes a look at congestion control algorithms in TCP that help to reduce packet loss when the network is very busy. It also discusses some modifications that have been proposed to increase throughput on fast networks or improve resiliency on lossy (e.g., wireless) networks. Finally, Chapter 17 shows how TCP keeps connections active even when no data is flowing.

The original specification for TCP is [RFC0793], although some errors in that RFC are corrected in the Host Requirements RFC, [RFC1122]. Since then, specifications for TCP have been revised and extended to include clarified and improved congestion control behavior [RFC5681][RFC3782][RFC3517][RFC3390][RFC3168], retransmission timeouts [RFC6298][RFC5682][RFC4015], operation with NATs [RFC5382], acknowledgment behavior [RFC2883], security [RFC6056][RFC5927][RFC5926], connection management [RFC5482], and urgent mechanism implementation guidelines [RFC6093]. There have also been a rich variety of experimental modifications covering retransmission behaviors [RFC5827][RFC3708], congestion detection and control [RFC5690][RFC5562][RFC4782][RFC3649][RFC2861], and other features. Finally, there is an effort to explore how TCP might take advantage of multiple simultaneous network-layer paths [RFC6182].

12.2.1 The TCP Service Model

Even though TCP and UDP use the same network layer (IPv4 or IPv6), TCP provides a totally different service to the application layer from what UDP does. TCP provides a *connection-oriented*, reliable, byte stream service. The term *connection-oriented* means that the two applications using TCP must establish a TCP connection by contacting each other before they can exchange data. The typical analogy is dialing a telephone number, waiting for the other party to answer the phone and saying "Hello," and then saying "Who's calling?" There are exactly two endpoints communicating with each other on a TCP connection; concepts such as broadcasting and multicasting (see Chapter 9) are not applicable to TCP.

TCP provides a byte stream abstraction to applications that use it. The consequence of this design decision is that no record markers or message boundaries are automatically inserted by TCP (see Chapter 1). A record marker corresponds to an indication of an application's write extent. If the application on one end writes 10 bytes, followed by a write of 20 bytes, followed by a write of 50 bytes, the application at the other end of the connection cannot tell what size the individual writes were. For example, the other end may read the 80 bytes in four reads of 20 bytes at a time or in some other way. One end puts a stream of bytes into TCP, and

the identical stream of bytes appears at the other end. Each endpoint individually chooses its read and write sizes.

TCP does not interpret the contents of the bytes in the byte stream at all. It has no idea if the data bytes being exchanged are binary data, ASCII characters, EBCDIC characters, or something else. The interpretation of this byte stream is up to the applications on each end of the connection. TCP does, however, support the urgent mechanism mentioned before, although it is no longer recommended for use.

12.2.2 Reliability in TCP

TCP provides reliability using specific variations on the techniques just described. Because it provides a byte stream interface, TCP must convert a sending application's stream of bytes into a set of packets that IP can carry. This is called *packetization*. These packets contain sequence numbers, which in TCP actually represent the byte offsets of the first byte in each packet in the overall data stream rather than packet numbers. This allows packets to be of variable size during a transfer and may also allow them to be combined, called *repacketization*. The application data is broken into what TCP considers the best-size chunks to send, typically fitting each segment into a single IP-layer datagram that will not be fragmented. This is different from UDP, where each write by the application usually generates a UDP datagram of that size (plus headers). The chunk passed by TCP to IP is called a *segment* (see Figure 12-2). In Chapter 15 we shall see how TCP decides what size a segment should be.

TCP maintains a mandatory checksum on its header, any associated application data, and fields from the IP header. This is an end-to-end pseudo-header checksum whose purpose is to detect any bit errors introduced in transit. If a segment arrives with an invalid checksum, TCP discards it without sending any acknowledgment for the discarded packet. The receiving TCP might acknowledge a *previous* (already acknowledged) segment, however, to help the sender with its congestion control computations (see Chapter 16). The TCP checksum uses the same mathematical function as is used by other Internet protocols (UDP, ICMP, etc.). For large data transfers, there is some concern that this checksum is not really strong enough [SP00], so careful applications should apply their own error protection methods (e.g., stronger checksums or CRCs) or use a middleware layer to achieve the same result (e.g., see [RFC5044]).

When TCP sends a group of segments, it normally sets a single retransmission timer, waiting for the other end to acknowledge reception. TCP does not set a different retransmission timer for every segment. Rather, it sets a timer when it sends a window of data and updates the timeout as ACKs arrive. If an acknowledgment is not received in time, a segment is retransmitted. In Chapter 14 we will look at TCP's adaptive timeout and retransmission strategy in more detail.

When TCP receives data from the other end of the connection, it sends an acknowledgment. This acknowledgment may not be sent immediately but is normally delayed a fraction of a second. The ACKs used by TCP are *cumulative* in the sense that an ACK indicating byte number N implies that all bytes up to number N

(but not including it) have already been received successfully. This provides some robustness against ACK loss—if an ACK is lost, it is very likely that a subsequent ACK is sufficient to ACK the previous segments.

TCP provides a *full-duplex* service to the application layer. This means that data can be flowing in each direction, independent of the other direction. Therefore, each end of a connection must maintain a sequence number of the data flowing in each direction. Once a connection is established, every TCP segment that contains data flowing in one direction of the connection also includes an ACK for segments flowing in the opposite direction. Each segment also contains a window advertisement for implementing flow control in the opposite direction. Thus, when a TCP segment arrives on a connection, the window may slide forward, the window size may change, and new data may have arrived. As we shall see in Chapter 13, a fully active TCP connection is bidirectional and symmetric; data can flow equally well in either direction.

Using sequence numbers, a receiving TCP discards duplicate segments and reorders segments that arrive out of order. Recall that any of these anomalies can happen because TCP uses IP to deliver its segments, and IP does not provide duplicate elimination or guarantee correct ordering. Because it is a byte stream protocol, however, TCP never delivers data to the receiving *application* out of order. Thus, the receiving TCP may be forced to hold on to data with larger sequence numbers before giving it to an application until a missing lower-sequence-numbered segment (a "hole") is filled in.

We will now begin to look at some of the details of TCP. In this chapter we will only introduce the encapsulation and header structure for TCP. Other details appear in the next five chapters. TCP can be used with IPv4 or IPv6, and the pseudo-header checksum it uses (similar to UDP's) is mandatory for use with either IPv4 or IPv6.

12.3 TCP Header and Encapsulation

TCP is encapsulated in IP datagrams as shown in Figure 12-2.

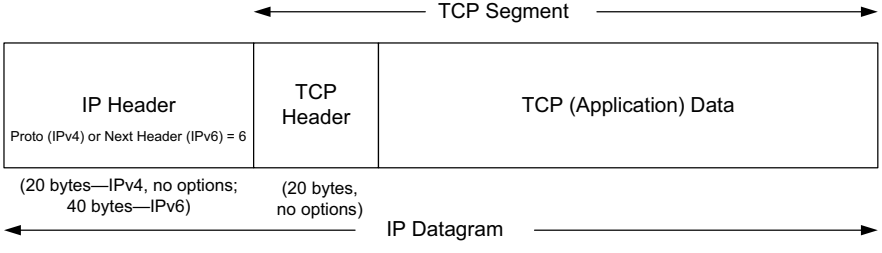


Figure 12-2 The TCP header appears immediately following the IP header or last IPv6 extension header and is often 20 bytes long (with no TCP options). With options, the TCP header can be as large as 60 bytes. Common options include Maximum Segment Size, Timestamps, Window Scaling, and Selective ACKs.

The header itself is considerably more complicated than the header we saw for UDP in Chapter 10. This is not very surprising, as TCP is a significantly more complicated protocol that must keep each end of the connection informed (synchronized) about the current state. It is shown in Figure 12-3.

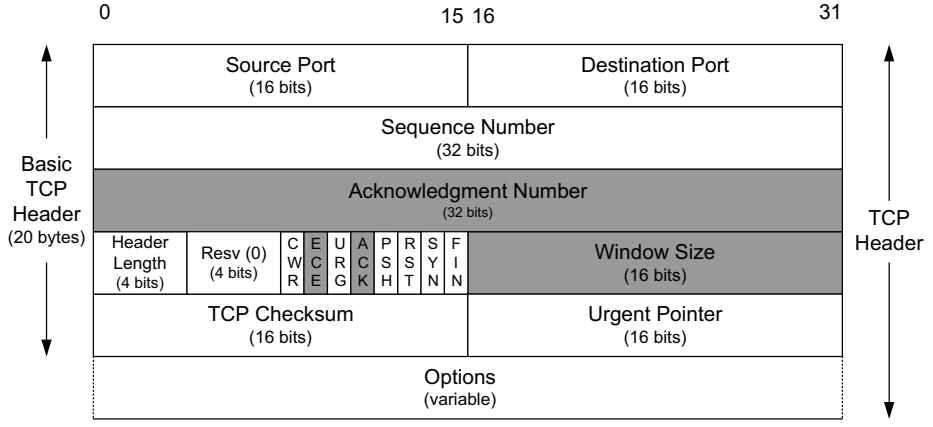


Figure 12-3 The TCP header. Its normal size is 20 bytes, unless options are present. The *Header Length* field gives the size of the header in 32-bit words (minimum value is 5). The shaded fields (*Acknowledgment Number*, *Window Size*, plus *ECE* and *ACK* bits) refer to the data flowing in the opposite direction relative to the sender of this segment.

Each TCP header contains the source and destination port number. These two values, along with the source and destination IP addresses in the IP header, uniquely identify each connection. The combination of an IP address and a port number is sometimes called an *endpoint* or *socket* in the TCP literature. The latter term appeared in [RFC0793] and was ultimately adopted as the name of the Berkeley-derived programming interface for network communications (now frequently called "Berkeley sockets"). It is a *pair* of sockets or endpoints (the 4-tuple consisting of the client IP address, client port number, server IP address, and server port number) that uniquely identifies each TCP connection. This fact will become important when we look at how a TCP server can communicate with multiple clients (see Chapter 13).

The *Sequence Number* field identifies the byte in the stream of data from the sending TCP to the receiving TCP that the first byte of data in the containing segment represents. If we consider the stream of bytes flowing in one direction between two applications, TCP numbers each *byte* with a sequence number. This sequence number is a 32-bit unsigned number that wraps back around to 0 after reaching $(2^{32}) - 1$. Because every byte exchanged is numbered, the *Acknowledgment Number* field (also called the *ACK Number* or *ACK* field for short) contains the next sequence number that the sender of the acknowledgment expects to receive. This is therefore the sequence number of the last successfully received byte of data plus 1. This field is valid only if the *ACK* bit field (described later in this section) is on,

which it usually is for all but initial and closing segments. Sending an ACK costs nothing more than sending any other TCP segment because the 32-bit *ACK Number* field is *always* part of the header, as is the *ACK* bit field.

When a new connection is being established, the *SYN* bit field is turned on in the first segment sent from client to server. Such segments are called *SYN segments*, or simply *SYNs*. The *Sequence Number* field then contains the first sequence number to be used on that direction of the connection for subsequent sequence numbers and in returning ACK numbers (recall that connections are all bidirectional). Note that this number is *not* 0 or 1 but instead is another number, often randomly chosen, called the *initial sequence number* (ISN). The reason for the ISN not being 0 or 1 is a security measure and will be discussed in Chapter 13. The sequence number of the first byte of data sent on this direction of the connection is the ISN plus 1 because the *SYN* bit field *consumes* one sequence number. As we shall see later, consuming a sequence number also implies reliable delivery using retransmission. Thus, SYNs and application bytes (and FINs, which we will see later) are reliably delivered. ACKs, which do not consume sequence numbers, are not.

TCP can be described as "a sliding window protocol with cumulative positive acknowledgments." The *ACK Number* field is constructed to indicate the largest byte received in order at the receiver (plus 1). For example, if bytes 1–1024 are received OK, and the next segment contains bytes 2049–3072, the receiver cannot use the regular *ACK Number* field to signal the sender that it received this new segment. Modern TCPs, however, have a *selective acknowledgment* (SACK) option that allows the receiver to indicate to the sender out-of-order data it has received correctly. When paired with a TCP sender capable of *selective repeat*, a significant performance benefit may be realized [FF96]. In Chapter 14 we will see how TCP uses *duplicate acknowledgments* (multiple segments with the same *ACK* field) to help with its congestion control and error control procedures.

The *Header Length* field gives the length of the header in 32-bit words. This is required because the length of the *Options* field is variable. With a 4-bit field, TCP is limited to a 60-byte header. Without options, however, the size is 20 bytes.

Currently eight bit fields are defined for the TCP header, although some older implementations understand only the last six of them.[1] One or more of them can be turned on at the same time. We briefly mention their use here and discuss each of them in more detail in later chapters.

1. ***CWR***—Congestion Window Reduced (the sender reduced its sending rate); see Chapter 16.
2. ***ECE***—ECN Echo (the sender received an earlier congestion notification); see Chapter 16.
3. ***URG***—Urgent (the *Urgent Pointer* field is valid—rarely used); see Chapter 15.

1. Note that [RFC3540], an experimental RFC, also defines the least significant of the Resv bits as a nonce sum (NS). See Section 16.12.

4. ***ACK***—Acknowledgment (the *Acknowledgment Number* field is valid—always on after a connection is established); see Chapters 13 and 15.
5. ***PSH***—Push (the receiver should pass this data to the application as soon as possible—not reliably implemented or used); see Chapter 15.
6. ***RST***—Reset the connection (connection abort, usually because of an error); see Chapter 13.
7. ***SYN***—Synchronize sequence numbers to initiate a connection; see Chapter 13.
8. ***FIN***—The sender of the segment is finished sending data to its peer; see Chapter 13.

TCP's flow control is provided by each end advertising a window size using the *Window Size* field. This is the number of bytes, starting with the one specified by the ACK number, that the receiver is willing to accept. This is a 16-bit field, limiting the window to 65,535 bytes, and thereby limiting TCP's throughput performance. In Chapter 15 we will look at the Window Scale option that allows this value to be scaled, providing much larger windows and improved performance for high-speed and long-delay networks.

The *TCP Checksum* field covers the TCP header and data and some fields in the IP header, using a pseudo-header computation similar to the one used with ICMPv6 and UDP that we discussed in Chapters 8 and 10. It is mandatory for this field to be calculated and stored by the sender, and then verified by the receiver. The TCP checksum is calculated with the same algorithm as the IP, ICMP, and UDP ("Internet") checksums.

The *Urgent Pointer* field is valid only if the *URG* bit field is set. This "pointer" is a positive offset that must be added to the *Sequence Number* field of the segment to yield the sequence number of the *last* byte of urgent data. TCP's urgent mechanism is a way for the sender to provide specially marked data to the other end.

The most common *Option* field is the Maximum Segment Size option, called the MSS. Each end of a connection normally specifies this option on the first segment it sends (the ones with the *SYN* bit field set to establish the connection). The MSS option specifies the maximum-size segment that the sender of the option is willing to receive in the reverse direction. We describe the MSS option in more detail in Chapter 13 and some of the other TCP options in Chapters 14 and 15. Other common options we investigate include SACK, Timestamp, and Window Scale.

In Figure 12-2 we note that the data portion of the TCP segment is optional. We will see in Chapter 13 that when a connection is established, and when a connection is terminated, segments are exchanged that contain only the TCP header (with or without options) but no data. A header without any data is also used to acknowledge received data, if there is no data to be transmitted in that direction (called a *pure ACK*), and to notify the communication peer of a change in the window size (called a *window update*). There are also some cases resulting from timeouts when a segment can be sent without any data.

12.4 Summary

The problem of providing reliable communications over lossy communication channels has been studied for years. The two primary methods for dealing with errors include error-correcting codes and data retransmission. The protocols using retransmissions must also handle data loss, usually by setting a timer, and must also arrange some way for the receiver to signal the sender what it has received. Deciding how long to wait for an ACK can be tricky, as the appropriate time may change as network routing or load on the end systems varies. Modern protocols estimate the round-trip time and set the retransmission timer based on some function of these measurements.

Except for setting the retransmission timer, retransmission protocols are simple when only one packet may be in the network at one time, but they perform poorly for networks where the delay is high. To be more efficient, multiple packets must be injected into the network before an ACK is received. This approach is more efficient but also more complex. A typical approach to managing the complexity is to use sliding windows, whereby packets are marked with sequence numbers, and the window size bounds the number of such packets. When the window size varies based on either feedback from the receiver or other signals (such as dropped packets), both flow control and congestion control can be achieved.

TCP provides a reliable, connection-oriented, byte stream, transport-layer service built using many of these techniques. We looked briefly at all of the fields in the TCP header, noting that most of them are directly related to these abstract concepts in reliable delivery. We will examine them in detail in the chapters that follow. TCP packetizes the application data into segments, sets a timeout anytime it sends data, acknowledges data received by the other end, reorders out-of-order data, discards duplicate data, provides end-to-end flow control, and calculates and verifies a mandatory end-to-end checksum. It is the most widely used protocol on the Internet. It is used by most of the popular applications, such as HTTP, SSH/TLS, NetBIOS (NBT—NetBIOS over TCP), Telnet, FTP, and electronic mail (SMTP). Many distributed file-sharing applications (e.g., BitTorrent, Shareaza) also use TCP.

12.5 References

[FF96] K. Fall and S. Floyd, "Simulation-Based Comparisons of Tahoe, Reno and SACK TCP," *ACM Computer Communications Review*, July 1996.

[J90] R. Jain, "Congestion Control in Computer Networks: Issues and Trends," *IEEE Network Magazine*, May 1990.

[K75] L. Kleinrock, *Queuing Systems, Volume 1: Theory* (Wiley-Interscience, 1975).

[K97] S. Keshav, *An Engineering Approach to Computer Networking* (Addison-Wesley, 1997). (Note: A second edition is being developed.)

[RFC0793] J. Postel, "Transmission Control Protocol," Internet RFC 0793/STD 0007, Sept. 1981.

[RFC1122] R. Braden, ed., "Requirements for Internet Hosts—Communication Layers," Internet RFC 1122/STD 0003, Oct. 1989.

[RFC2861] M. Handley, J. Padhye, and S. Floyd, "TCP Congestion Window Validation," Internet RFC 2861 (experimental), June 2000.

[RFC2883] S. Floyd, J. Mahdavi, M. Mathis, and M. Podolsky, "An Extension to the Selective Acknowledgement (SACK) Option for TCP," Internet RFC 2883, July 2000.

[RFC3168] K. Ramakrishnan, S. Floyd, and D. Black, "The Addition of Explicit Congestion Notification (ECN) to IP," Internet RFC 3168, Sept. 2001.

[RFC3390] M. Allman, S. Floyd, and C. Partridge, "Increasing TCP's Initial Window," Internet RFC 3390, Oct. 2002.

[RFC3517] E. Blanton, M. Allman, K. Fall, and L. Wang, "A Conservative Selective Acknowledgment (SACK)-Based Loss Recovery Algorithm for TCP," Internet RFC 3517, Apr. 2003.

[RFC3540] N. Spring, D. Wetherall, and D. Ely, "Robust Explicit Congestion Notification (ECN) Signaling with Nonces," Internet RFC 3540 (experimental), June 2003.

[RFC3649] S. Floyd, "HighSpeed TCP for Large Congestion Windows," Internet RFC 3649 (experimental), Dec. 2003.

[RFC3708] E. Blanton and M. Allman, "Using TCP Duplicate Selective Acknowledgement (DSACKs) and Stream Control Transmission Protocol (SCTP) Duplicate Transmission Sequence Numbers (TSNs) to Detect Spurious Retransmissions," Internet RFC 3708 (experimental), Feb. 2004.

[RFC3782] S. Floyd, T. Henderson, and A. Gurtov, "The NewReno Modification to TCP's Fast Recovery Algorithm," Internet RFC 3782, Apr. 2004.

[RFC4015] R. Ludwig and A. Gurtov, "The Eifel Response Algorithm for TCP," Internet RFC 4015, Feb. 2005.

[RFC4782] S. Floyd, M. Allman, A. Jain, and P. Sarolahti, "Quick-Start for TCP and IP," Internet RFC 4782 (experimental), Jan. 2007.

[RFC5044] P. Culley, U. Elzur, R. Recio, S. Bailey, and J. Carrier, "Marker PDU Aligned Framing for TCP Specification," Internet RFC 5044, Oct. 2007.

[RFC5382] S. Guha, ed., K. Biswas, B. Ford, S. Sivakumar, and P. Srisuresh, "NAT Behavioral Requirements for TCP," Internet RFC 5382/BCP 0142, Oct. 2008.

[RFC5482] L. Eggert and F. Gont, "TCP User Timeout Option," Internet RFC 5482, Mar. 2009.

[RFC5562] A. Kuzmanovic, A. Mondal, S. Floyd, and K. Ramakrishnan, "Adding Explicit Congestion Notification (ECN) Capability to TCP's SYN/ACK Packets," Internet RFC 5562 (experimental), June 2009.

[RFC5681] M. Allman, V. Paxson, and E. Blanton, "TCP Congestion Control," Internet RFC 5681, Sept. 2009.

[RFC5682] P. Sarolahti, M. Kojo, K. Yamamoto, and M. Hata, "Forward RTO-Recovery (F-RTO): An Algorithm for Detecting Spurious Retransmission Timeouts with TCP," Internet RFC 5682, Sept. 2009.

[RFC5690] S. Floyd, A. Arcia, D. Ros, and J. Iyengar, "Adding Acknowledgement Congestion Control to TCP," Internet RFC 5690 (informational), Feb. 2010.

[RFC5827] M. Allman, K. Avrachenkov, U. Ayesta, J. Blanton, and P. Hurtig, "Early Retransmit for TCP and Stream Control Transmission Protocol (SCTP)," Internet RFC 5827 (experimental), May 2010.

[RFC5926] G. Lebovitz and E. Rescorla, "Cryptographic Algorithms for the TCP Authentication Option (TCP-AO)," Internet RFC 5926, June 2010.

[RFC5927] F. Gont, "ICMP Attacks against TCP," Internet RFC 5927 (experimental), July 2010.

[RFC6056] M. Larsen and F. Gont, "Recommendations for Transport-Protocol Port Randomization," Internet RFC 6056/BCP 0156, Jan. 2011.

[RFC6093] F. Gont and A. Yourtchenko, "On the Implementation of the TCP Urgent Mechanism," Internet RFC 6093, Jan. 2011.

[RFC6182] A. Ford, C. Raiciu, M. Handley, S. Barre, and J. Iyengar, "Architectural Guidelines for Multipath TCP Development," Internet RFC 6182 (informational), Mar. 2011.

[RFC6298] V. Paxson, M. Allman, J. Chu, and M. Sargent, "Computing TCP's Retransmission Timer," Internet RFC 6298, June 2011.

[S48] C. Shannon, "A Mathematical Theory of Communication," *Bell System Technical Journal*, July/Oct. 1948.

[SP00] J. Stone and C. Partridge, "When the CRC and TCP Checksum Disagree," *Proc. ACM SIGCOMM*, Aug./Sept. 2000.

This page intentionally left blank

Index

Symbols

* (Wildcard)
  domain names and, 526
  local IP address restrictions in server design, 500–501
.in-addr.arpa
  classless delegation, 539
  special domain for IPv4, 537–538
.ip6.arpa, 537–538

Numbers

2.4GHz band, Wi-Fi, 124–126
3DES. *See* Triple-DES
3GPP (3rd Generation Partnership Project), 275, 933
5GHz band, Wi-Fi, 124–126
6rd (IPv6 Rapid Deployment), 339, 933
6to4
  definition of, 933
  IPv4 to IPv6 transition, 482

A

A (address) records
  definition of, 529, 933
  overview of, 529–530
  querying, 531
  translating DNS from IPv4 to IPv6, 569
A-MPDU (aggregated MAC protocol data unit)
  definition of, 934
  frame aggregation support, 118–119
A-MSDU (aggregated MAC service data unit)
  definition of, 934
  frame aggregation support, 118
AAA (authentication, authorization, and accounting), 833–834, 933
AAAA (address) records
  definition of, 933
  DNS resource record types, 529–530
  translating DNS from IPv4 to IPv6, 569
Abbreviated handshake, TLS, 881
ABC (Appropriate Byte Counting), in TCP, 733, 933
Abortive release, of TCP connections, 627
Abstract Syntax Notation One (ASN.1), 935
Access categories (ACs), in EDCA, 123
Access control
  NAC (Network Access Control), 833–837
  RADIUS server for, 141
Access control lists. *See* ACLs (access control lists)
Access Network Discovery and Selection Function (ANDSF), 275, 934
Access points. *See* APs (access points)
ACCM (Asynchronous Control Character Map), in PPP
  definition of, 933
  escaping characters and, 134–135
ACD (Address Conflict Detection), 176–177, 933
ACFC (Address and Control Field Compression), in PPP, 132, 933
ACK clock, in TCP, 731
ACK division, attack against TCP, 785
Acknowledge Number field, in GRE tunnels, 150
ACKs (acknowledgement)
  clocking congestion via, 730–731
  combined with SYN segments (SACK), 607
  cumulative in TCP, 586–587
  definition of, 933
  duplicate ACK threshold in fast retransmit, 667
  establishing TCP connections and, 597, 602–603
  NAT and TCP and, 307–308
  requesting connection to nonexistent TCP port, 626
  retransmission and, 580–581
  retransmission timeout settings, 584
  stretch ACKs in recovery from local congestion, 754–757
  TCP header field, 588–589
  TCP segments and, 701
  in Wi-Fi control frames, 116
  window update and, 583
ACLs (access control lists)
  definition of, 933
  in packet-filtering firewalls, 300
  rules in, 335
ACs (access categories), in EDCA, 123
ACs (attribute certificates)
  as alternative to public key certificates, 831
  definition of, 933

ACSII characters, escaping in PPP operations, 134–135
Actions, in ACL rules, 335
Active attacks, threats to network communication, 807–809
Active closer, FIN segments and, 597
Active open, in TCP connections, 597
Active opener (client)
  RST segments, 631
  simultaneous open and, 600
  in TCP connections, 596, 599
Active queue management (AQM), 782–785, 935
Ad hoc mode, Wi-Fi, 112
Ad-Hoc On-Demand Distance Vector (AODV)
  definition of, 934
  Wi-Fi mesh and, 130
Additive increase/additive decrease (AIAD)
  congestion control and, 777
  definition of, 934
Additive increase/multiplicative decrease (AIMD)
  congestion control and, 769
  definition of, 934
Address (A) records. *See* A (address) records
Address (AAAA) records. *See* AAAA (address) records
Address (Addr) field, in PPP frames, 132
Address and Control Field Compression (ACFC), in PPP, 132, 933
Address autoconfiguration. *See* SLAAC (stateless address autoconfiguration)
Address behavior, in NAT, 311–313
Address Conflict Detection (ACD), 176–177, 933
Address Family Transition Router (AFTR), in DS-Lite, 340, 934
Address management, DHCP for, 235
Address pools, DHCP, 235–236
Address realms
  IP addresses, 299
  proxy firewalls supporting private address realms, 301
Address Resolution Protocol (ARP), 165
Address selection, in IP host models
  destination address selection algorithm, 224–225
  overview of, 222–223
  source address selection algorithm, 223–224
Address unreachable message, ICMPv6, 364
Admin-scope boundaries, in router configuration, 53
Administrative prohibition, ICMP messages and, 365
Administrative scope, in multicast addresses, 53
ADSP (Author Domain Signing Practices), in DKIM, 916–917, 933
Advanced Encryption Standard. *See* AES (Advanced Encryption Standard)
ADVERTISE message, DHCPv6, 262–264
Advertised window. *See* awnd (advertised window)
Advertisement Interval option, neighbor discovery in IPv6, 412
Advertisement messages, in MRD, 394–395
AEAD (authenticated encryption with associated data), 820, 934
AES (Advanced Encryption Standard)
  definition of, 934
  standardized for Internet use, 819
  as symmetric encryption algorithm, 811
  in Wi-Fi security, 129
AES-MAC, 819
AF (Assured Forwarding), 190, 934
AFTR (Address Family Transition Router), in DS-Lite, 340, 934
Aggregated MAC service data unit. *See* A-MSDU (aggregated MAC service data unit)
Aggregation
  route aggregation, 50
  of Wi-Fi frames, 116–119
Agile probing, in TCP, 779
AH (Authentication Header)
  authentication and integrity protection with, 856, 858
  definition of, 934
  fields in, 856
  in IPSec, 841
  NAT updates and, 866
  overview of, 454–455
  transport and tunnel modes, 856–857
AIA (Authority Information Access)
  certificate extension, 828
  definition of, 934
AIAD (additive increase/additive decrease)
  congestion control and, 777
  definition of, 934
AIMD (additive increase/multiplicative decrease)
  congestion control and, 769
  definition of, 934
Alert protocol, TLS handshaking, 880
ALGs (application layer gateways). *See also* Gateways
  definition of, 934
  IP routers, 20
  IPv4/IPv6 translation, 340–345
  NAT Traversal as alternative to, 316
  proxy firewalls and, 301
Allocation of IP addresses
  multicast addresses, 65
  overview of, 62
  unicast addresses, 62–65
  to users and organizations, 31

Alternate ports
RSTP, 110
STP, 104–105
Amplification attacks, DNS-related attacks, 571
ANDSF (Access Network Discovery and Selection Function), 275, 934
Answer, authority, and additional information section, of DNS message, 526–527
Any-source multicast (ASM)
definition of, 935
as multicast service model, 54
Anycast addresses, 62
AODV (Ad-Hoc On-Demand Distance Vector)
definition of, 934
Wi-Fi mesh and, 130
APIPA (Automatic Private IP Addressing)
definition of, 934
SLAAC and, 276, 284
APIs (Application Programming Interfaces)
definition of, 934
design and, 22
Application design
APIs in, 22
client/server design pattern, 20–21
peer-to-peer design pattern, 21–22
Application layer
full-duplex TCP service to, 587
of OSI model, 10
TCP and UDP services for, 585
Application-managed keepalives, 794
Application Programming Interfaces. *See* APIs (Application Programming Interfaces)
Application protocols, NAT and, 304
Application-unique strings. *See* AUS (application-unique strings)
Appropriate Byte Counting (ABC), in TCP, 733, 933
APs (access points)
definition of, 934
ICMP fast handover messages and, 388
Wi-Fi, 112
APSD (automatic power save delivery), 120, 935
AQM (active queue management), 782–785, 935
Architecture, protocol
end-to-end argument, 6
error control and flow control, 7–8
fate sharing, 6–7
packets, connections, and datagrams, 3–6
principles of, 2–3
of protocol suite, 1
ARM (ARPANET Reference Model), 1–2, 13–16
ARP (Address Resolution Protocol)
ACD (Adddress Conflict Detection), 176–177
announcement packets, 176
`arp` command, 177–178
attacks related to, 178–179
cache, 169–170
cache timeout, 174
definition of, 935
determining MAC addresses, 442
direct delivery and, 167–169
example of use, 166–167
frame format, 170–171
gratuitous ARP, 175–176
interaction between IP fragmentation and ARP/ND, 496–497
introduction to, 165–166
IPv4 and, 13
operation of, 171–173
Proxy ARP, 174–175
request to nonexistent host, 173–174
setting IPv4 address for embedded device, 178
summary and references, 179–180
TCP connection timeouts and, 604
`arp` command
ARP cache timeout and, 174
examining ARP cache, 169–170
options, 177–178
ARP hack, 175
ARP poisoning, attacks on ICMP, 429
ARP probe, ACD defining, 176
ARP reply frames, 168
ARP request frames
direct delivery and, 167
Proxy ARP and, 174–175
request to nonexistent host, 173–174
ARPANET Reference Model (ARM), 1–2, 13–16
ARQ (Automatic Repeat Request), 579–581, 935
AS (Authentication Server), in PANA, 935
AS (autonomous system)
definition of, 935
multicast addresses based on, 55
ASM (any-source multicast)
definition of, 935
as multicast service model, 54
ASN.1 (Abstract Syntax Notation One), 935
Assignment of unicast addresses
to devices, 32
multiple providers/multiple networks/multiple addresses, 68–70
overview of, 65–66
single provider/multiple networks/multiple addresses, 67–68
single provider/no network/single address, 66–67
single provider/single network/single address, 67
Assignment policies, IA (Identity Association) based on, 255–256

Assured Forwarding (AF), 190, 934
Asymmetric (public) key ciphers. *See also* Public key cryptography, 809–812
Asynchronous Control Character Map (ACCM)
  definition of, 933
  escaping characters and, 134–135
Attacks
  ARP, 178–179
  DNS, 571–572
  ICMP, 428–429
  Internet architecture, 25–26
  IP address, 70–71
  IP protocol, 226
  link layer, 154–156
  on NAT and firewall, 345–346
  system configuration, 292
  TCP, 640–643
  TCP congestion control and, 785–786
  TCP keepalive and, 802
  TCP timeout/retransmission and, 687
  TCP window management and, 723
  UDP, 507–508
Attribute certificates (ACs)
  as alternative to public key certificates, 831
  definition of, 933
Attribution, of datagrams, 26
Auditability, ESP and, 863
Augmented message, CRC, 86
AUS (application-unique strings)
  definition of, 935
  ENUM records and, 551–552
  NAPTR records and, 549
AUTH (authentication packets), in IKE, 935
Authenticated encryption, 814–815
Authenticated encryption with associated data (AEAD), 820, 934
Authenticated nonexistence, DNSSEC, 895
Authentication
  AH (Authentication Header). *See* AH (Authentication Header)
  basic principles of security, 806–807
  DHCP and, 271–273
  EAP methods for, 838
  PPP and, 140–141
  PSKs (preshared keys) for, 129–130
  SAE (Simultaneous Authentication of Equals), 130
  SHA-1 algorithm in, 268
  spoofing attacks and, 226
  TCP-AO (Authentication Option), 612
Authentication, authorization, and accounting (AAA), 833–834, 933
Authentication Header. *See* AH (Authentication Header)
Authentication Option (TCP-AO)
  definition of, 959
  TCP header, 612
Authentication Server (AS), in PANA, 55, 935
Author Domain Signing Practices (ADSP), 916–917, 933
Authorities, in allocation of IP addresses, 62
Authority Information Access (AIA)
  certificate extension, 828
  definition of, 934
Authority Key Identifier, for identifying public keys, 828, 830
Auto-proxy ARP, 175
Automatic power save delivery (APSD), 120, 935
Automatic Repeat Request (ARQ), 579–581, 935
Autonegotiation, in Ethernet
  duplex mismatch and, 96
Autonomous system (AS)
  definition of, 935
  multicast addresses based on, 55
Autotuning TCP receive windows, 715–716
Availability, in CIA triad, 806
awnd (advertised window) in TCP
  overview of, 729–730
  slow start algorithm and, 733–734, 736
AXFR (full zone transfer) messages, in DNS, 559–561, 935

B

B4 (Bridging Broadband), 340, 935
Backoff factor, RTO and, in TCP, 655
Backoff time, in MAC, 121–122
Backup ports, STP, 104–105
BACP (Bandwidth Allocation Control Protocol), 139, 935
Bandwidth Allocation Protocol (BAP), 139
Bandwidth (capacity)
  allocating in MP, 139
  buffer bloat and, 781
  connections and, 3
Bandwidth-delay product. *See* BDP (bandwidth-delay product)
Bandwidth on demand (BOD), 139, 936
Bandwidth-scalable TCPs, 773
Bank teller's algorithm, 138
BAP (Bandwidth Allocation Protocol), 139
Baran, Paul, 1
Basic Encoding Rules (BER), 935
Basic service set. *See* BSS (basic service set)
BCMCS (Broadcast and Multicast Service Controller), 935

BCP (best current practice) category, RFCs and, 23
BDP (bandwidth-delay product)
  congestion control and, 730
  high-speed networks and, 770
  HSTCP (HighSpeed TCP) and, 772
BER (Basic Encoding Rules), 935
BER (bit error rate)
  data frame fragmentation, 117–118
  definition of, 935
Berkeley Internet Name Domain v. 9 (BIND9), 935
Berkeley sockets
  half-close support, 598
  incoming connection queue and, 636, 639–640
  popular APIs, 22
  restrictions on foreign endpoints, 635
  state transitions, 618
  TCP ports, 588
  TCP_NODELAY option for disabling Nagle algorithm, 700
Berkeley Software Distribution. *See* BSD (Berkeley Software Distribution)
Best current practice (BCP) category, RFCs and, 23
Best-effort delivery, of packets, 7
Better-than-Nothing Security (BTNS), 852, 936
BGP (Border Gateway Protocol), 935
BI (binary increase), 774
BIC (Binary Increase Congestion Control)
  BIC-TCP, 773–774
  overview of, 772–773
Bidirectional tunneling, in mobile IP, 216–217
Big endian byte ordering, 183
Binary additive increase algorithm, 773–774
Binary exponential backoff, retransmission and, 650
Binary increase (BI), 774
Binary Increase Congestion Control (BIC)
  BIC-TCP, 773–774
  overview of, 772–773
Binary notation
  expressing IP addresses in, 32–33
  prefixes, 48
  of subnet masks, 39
Binary phase shift keying (BPSK)
  definition of, 936
  higher throughput (802.11n) support and, 128
Binary search increase algorithm, BIC-TCP and, 773–774
BIND9 (Berkeley Internet Name Domain v. 9), 935
Binding method, in STUN, 321
Binding, MNs (mobile nodes), 216–217
Binding Update (BU), in MIP, 936
Bit error rate (BER)
  data frame fragmentation, 117–118
  definition of, 935
Bit flipping attack, 918
Bit stuffing, in PPP frames, 132
BITS (Bump in the Stack), in IPsec, 840, 935
BITW (Bump in the Wire), in IPsec, 840, 936
Bitwise AND operation, used with subnet masks, 40
BL (Bulk Leasequery)
  DCHP relay agents, 269–270
  definition of, 936
Black hats, attacks related to Internet architecture, 26
Black holes, in PMTUD, 613
Blackhole route messages, ICMPv6, 372
Block ciphers, 811
Blocking route messages, ICMPv6, 372
Bloop attacks, 429
BOD (bandwidth on demand), 139, 936
Bombs, ICMP attacks, 428
Bonding, link aggregation and, 92–93
Boot Server Discovery Protocol (BSDP), 246, 936
BOOTP (Internet Bootstrap Protocol)
  compatibility with DHCP, 236–238
  definition of, 936
  DHCP based on, 235
  options, 238–239
  relay agents, 268
BOOTREQUEST, 239, 242
Border Gateway Protocol (BGP), 209, 935
Bot attacks, 26
Bot herders, 806
Botnets
  attacks related to Internet architecture, 26
  taking control of computers, 806
BPDUs (Bridge PDUs)
  building the spanning tree, 107
  definition of, 936
  RSTP (Rapid Spanning Tree Protocol), 110–111
  STP and, 104
  structure of, 105–107
  viewing with Wireshark, 109
BPSK (binary phase shift keying)
  definition of, 936
  higher throughput (802.11n) support and, 128
Bridge PDUs. *See* BPDUs (Bridge PDUs)
Bridges
  layer 2 relay agents, 270
  overview of, 98–102
  STP. *See* STP (Spanning Tree Protocol)
Bridging Broadband (B4), in DS-Lite, 340, 935
Broadcast addresses
  overview of, 15
  setting/finding, 437–439
  structure of, 42–43
Broadcast and Multicast Service Controller (BCMCS), in cellular networks, 239, 935

Broadcast domain, link-layer broadcast, 167
Broadcasting
  introduction to, 435–436
  overview of, 436–437
  sending broadcast datagrams, 439–441
  setting/finding broadcast addresses, 437–439
Brute-force attacks, 816
BSD (Berkeley Software Distribution)
  definition of, 936
  standards and, 24
  Tahoe release, 737
BSDP (Boot Server Discovery Protocol), 246, 936
BSS (basic service set)
  definition of, 936
  QoS BBS, 122
  Wi-Fi, 112
BTNS (Better-than-Nothing Security), in IPsec, 852, 936
BU (Binding Update), in MIP, 936
Buffer bloat, TCP congestion control and, 781–782
Buffer overflow, worms, 805
Buffers
  large buffers and auto-tuning, 715–719
  packets stored in, 4
Bulk data, in TCP communication, 692
Bulk Leasequery (BL)
  DCHP relay agents, 269–270
  definition of, 936
Bump in the Stack (BITS), in IPsec, 840, 935
Bump in the Wire (BITW),in IPsec, 840, 936
Bundles, of PPP links, 137, 139
Byte stuffing, in PPP frames, 132

C

Cache
  ARP cache, 169–170
  ARP cache timeouts, 174
Cache poisoning, DNS-related attacks, 572
Caching servers, DNS, 517–518
CALIPSO (Common architecture Label IPv6 Security Option), 199, 936
Callback Control Protocol (CBCP), in PPP
  definition of, 936
  negotiation of callbacks in LCP, 136
Callback, PPP supporting, 136
CAMELLIA, standardized for Internet use, 819
Candidate sets (CS), in source address selection, 223–224
Candidate transport addresses, in ICE, 333
Canonical name records. *See* CNAME (canonical name) records
Canonical ordering, of RRset in DNSSEC, 902–903
Capture, network communication, 808
Capturing portals, link layer attacks, 155
Care-of address (CoA)
  definition of, 937
  in Mobile IP, 216–217
Care-of Test (CoT)
  definition of, 937
  mobility messages in RRP, 218–219
Care-of Test Init (CoTI)
  definition of, 937
  mobility messages in RRP, 218–219
Carrier-grade NAT (CGN), 315
Carrier sense, 120
Carrier sense, multiple access with collision avoidance. *See* CSMA/CA (carrier sense, multiple access with collision avoidance)
Carrier sensed, multiple access with collision detection (CSMA/CD)
  definition of, 938
  Ethernet interface and, 80–81
CAs (certification authorities). *See also* Certificates (public key)
  definition of, 936
  PKI (Public Key Infrastructure) and, 821–822
Catenet. *See also* Internetwork, 1
CBC (cipher block chaining)
  block ciphers in encryption algorithms, 820
  definition of, 936
CBC-MAC (cipher block chaining message authentication code), 129
CBCP (Callback Control Protocol)
  definition of, 936
  negotiation of callbacks in LCP, 136
CCA (clear channel assessment), in Wi-Fi
  definition of, 936
  for physical carrier sense, 121
CCITT (Comité Consultatif International Télégraphique), 24, 936
CCM (counter mode)
  CCMP algorithm based on, 129
  definition of, 936
CCMP (counter mode with CBC Message Authentication Code)
  definition of, 936
  in Wi-Fi security, 129–130
CCP (Compression Control Protocol), in PPP,
  definition of, 936
  MPPE and, 145
  overview of, 139–140
ccTLDs (country code TLDs)
  definition of, 937
  in DNS name space, 512–514
CDN (content delivery networks), 535
CDP (CRL Distribution Point), 828–829, 937

CE (Congestion Experienced) bit, in IP header, 783–784
CERT (Computer Emergency Response Team), 937
Certificate Policies (CP), 828
Certificate Request (CERTREQ) payload
  definition of, 937
  in IKE, 869–870
Certificate revocation lists. *See* CRLs (certificate revocation lists)
Certificates, ACs (attribute certificates), 831
Certificates option, in ND, 417
Certificates (public key)
  CAs and PKIs and, 821–822
  extensions, 827–828
  for identifying four types of Internet entities, 823
  validating, 826
  validating and revoking certificates, 828–831
  viewing preconfigured, 823–826
Certificates, SEND, 403
Certification authorities. *See* CAs (certification authorities)
Certification Path Advertisement message, ICMP Send messages, 407
Certification Path Solicitation message, ICMP Send messages, 406–407
Certification Practice Statement (CPS), 937
CERTREQ (Certificate Request) payload
  definition of, 937
  in IKE, 869–870
CGAs (cryptographically generated addresses)
  definition of, 937
  Handover Key Request/Reply options, 422–423
  neighbor discovery options in IPv6, 414–415
  RSA Signature option, 415–416
  securing IPv6 Neighbor Discovery, 292
  SEND (Secure Neighbor Discovery) and, 403–406
  verification of, 405
CGN (carrier-grade NAT), 315
Chaddr (Client Hardware Address) field
  DHCP/BOOTP message format, 238
  MAC addresses in, 244
Challenge-response protocols, 816
Change Cipher Spec protocol, 878
Channels
  in SSM multicast service model, 54
  TURN, 327
  Wi-Fi, 124
CHAP (Challenge-Handshake Authentication Protocol)
  definition of, 937
  for PPP authentication, 140–141
Character stuffing, in PPP frames, 132
Checkpointing, saving work, 10
Checksums
  (Generic Routing Encapsulation), 150
  applying at application layer, 601
  for detecting packet errors, 580
  IP header fields, 185
  TCP, 586, 590
  UDP, 475–478
  UDP-Lite, 487–488
  verifying message integrity, 817
  WKP checksum neutrality, 341
CIA (confidentiality, integrity, and availability)
  AH and, 856, 858
  definition of, 937
  ESP and, 860
  overview of, 806
ciaddr (Client IP address) field, 237
CIDR (Classless Inter-Domain Routing)
  definition of, 937
  developed to alleviate pressure on available IPv4 addresses, 47–48
  masks, 47
  routing scalability addressed by, 303
Cipher-based MAC (CMAC), 819–820, 937
Cipher block chaining (CBC)
  block ciphers in encryption algorithms, 820
  definition of, 936
Cipher block chaining message authentication code (CBC-MAC), 129
Cipher Change protocol, TLS handshaking, 880
Cipher suite rollback attacks, 919
Cipher suites. *See* CS (cipher suites)
Ciphertext
  encrypting cleartext message, 810
  TLS, 878–879
Civic location, location information in DHCP, 274
Clark, D., 3
Class of Service (CoS), 937
Class selector code points, 190
Class Selector (CS), 938
Classes, IP address
  Class D addresses reserved for IPv4 multicast, 54–55
  overview of, 34–36
  prefix length and, 47–48
Classic RTO method, 651–652
Classless Inter-Domain Routing. *See* CIDR (Classless Inter-Domain Routing)
Classless routes, DHCP and, 246
Classless Static Route (CSR) parameter, 246
Clear channel assessment (CCA)
  definition of, 936
  for physical carrier sense, 121
Clear to send. *See* CTS (clear to send)

Client alive messages, 802
Client Hardware Address (Chaddr) field
  DHCP/BOOTP message format, 238
  MAC addresses in, 244
Client IP address (Ciaddr) field, in DHCP, 237
Client/server design pattern, 20–21
Client state machine, DHCP, 251–252
ClientHello message, in TLS, 887
Clients
  keepalives detecting state of client host, 795
  setting keepalive time for Windows client, 797–799
Clock granularity, RTO bounds and, 654
Clock recovery, Ethernet frames, 84
CLOSED state, in TCP,
  sharing connection state, 768
  simultaneous open and close transitions, 625
  TCP state transitions, 618
CLOSE_WAIT state, TCP state transitions, 618
CMAC (cipher-based MAC), 819–820, 937
CNAME (canonical name) records
  definition of, 937
  DNS resource record types, 534–536
  translating DNS from IPv4 to IPv6, 569
CNs (correspondent nodes), in MIP, 216–218, 937
CoA (care-of address)
  definition of, 937
  in Mobile IP, 216–217
Coding theory, 579
Collision Count field, CGAs, 404–405
Comité Consultatif International Télégraphique (CCITT), 24, 936
Common architecture Label IPv6 Security Option (CALIPSO), 199, 936
Communication protocols
  ARQ (Automatic Repeat Request), 579–581
  congestion control, 583–584
  flow control, 583
  introduction to, 579
  retransmission settings, 584
  sliding windows, 582
  TCP. *See* TCP (Transmission Control Protocol)
  windows of packets, 581–582
Compound TCP (CTCP) algorithm, 779–781, 938
Compression
  ACFC (Address and Control Field Compression), 132
  CCP (Compression Control Protocol), 139–140
  header compression, 139, 142–143
  MPPC (Microsoft Point-to-Point Compression Protocol), 140
  PFC (Protocol Field Compression), 133
  VJ (Van Jacobson) compression, 141
Compression Control Protocol. *See* CCP (Compression Control Protocol)
Compression labels, DNS names and, 523–524
Compression-optional attributes, STUN, 321
Computer Emergency Response Team (CERT), 937
Computer Systems Research Group (CSRG), 24, 938
Concurrent servers, 21
Confidentiality, in CIA triad, 806
Confidentiality, integrity, and availability. *See* CIA (confidentiality, integrity, and availability)
Configuration data delivery, DHCP for, 235
Configuration Payload (CP)
  definition of, 937
  IKE, 849
Congestion avoidance algorithm
  classic algorithms for TCP congestion, 734–736
  comparing with slow start, 736–737
Congestion collapse state, 728
Congestion control. *See also* Flow control
  in communication protocols, 583–584
  in TCP. *See* TCP congestion control
  in UPD server design, 505
Congestion Experienced (CE) indicator, in IP header, 783–784
Congestion indicator, ECN, 188
Congestion Manager, 768
Congestion window. *See* cwnd (congestion window)
Congestion Window Reducing. *See* CWR (Congestion Window Reducing)
Congestion Window Validation (CWV), in TCP, 742–744, 938
Connection completion, TCP congestion control, 766–767
Connection-oriented networks, 5
Connection-oriented protocols, 595
Connection-oriented service, 585
Connection refused error, in TCP and UDP, 626
Connection state, TCP, 595
Connectionless networks, 5, 181
Connectionless protocols, 595
Connections, in protocol architecture, 3–6
Connections, TCP
  aborting, 627–628
  attacks related to, 640–643
  establishing and terminating, 595–598
  example of PMTUD with, 613–616
  example showing packet-level details, 602–604
  FIN_WAIT_2 state, 625
  half-close operation, 598–599
  half-open connections, 628–630
  header options, 605–606

incoming connection queue, 636–640
introduction to, 595
ISN (initial sequence number), 601–602
MSS (Maximum Segment Size) option, 606–607
PAWS (Protection against Wrapped Sequence Numbers), 610–611
PMTUD (Path MTU Discovery) and, 612–613
port numbers and, 632–634
quiet time concept, 624
requesting connection to nonexistent port, 626
reset segments, 625–626
restrictions on foreign endpoints, 635–636
restrictions on local IP addresses, 634–635
SACK (selective acknowledgement) option, 607
server operation and, 631–632
simultaneous open and close, 599–601
simultaneous open and close transitions, 625
state transition diagrams, 617–618
summary and references, 643–645
TCP-AO (Authentication Option), 612
timeout settings, 604–605
Timestamps option, 608–610
TIME_WAIT state (2MSL), 618–624
translating addresses and port numbers, 605
TWA (TIME-WAIT Assassination), 630–631
UTO (User Timeout) option, 611–612
WSCALE (Window Scale) option, 608
Conservation of packets, 731
Content delivery networks (CDN), 535
Content filters, web proxies operating as, 302
Contention window (CW)
definition of, 938
in MAC, 122
Control field, in PPP frames, 132
Control frames, Wi-Fi, 115–116
Cooks, C., 814
Coordinated Universal Time (UTC), 961
Correspondent nodes (CNs), in MIP, 216–218, 937
Correspondent registration, in RO, 218
CoS (Class of Service), 937
CoT (Care-of Test)
definition of, 937
mobility messages in RRP, 218–219
CoTI (Care-of Test Init)
definition of, 937
mobility messages in RRP, 218–219
Counter (CTR) mode
definition of, 938
operating modes of encryption algorithms, 820
Counter mode (CCM)
CCMP algorithm based on, 129
definition of, 936
Counter mode with CBC Message Authentication Code (CCMP)
definition of, 936
in Wi-Fi security, 129–130
Counters, IGMP/MLD, 467–468
Country code TLDs (ccTLDs)
definition of, 937
in DNS name space, 512–514
CP (Certificate Policies), 828
CP (Configuration Payload)
definition of, 937
IKE, 849
CPS (Certification Practice Statement), 937
CRCs (Cyclic Redundancy Checks)
applying at application layer, 601
compared with Internet checksum, 186
definition of, 937
for detecting errors in packets, 580
host address filtering and, 449, 451
integrity checking in Ethernet frames, 86–88
TCP reliability and, 586
CREATE_CHILD_SA exchange, IKE protocol, 852–853
CRL Distribution Point (CDP), 828–829, 937
CRLs (certificate revocation lists)
definition of, 937
distribution point for, 828–829
extensions, 830
Cryptographic suites, 819–821
Cryptographically generated addresses. *See* CGAs (cryptographically generated addresses)
Cryptographically strong PRNGs (CSPRNGs), 816, 938
Cryptography. *See also* Encryption
attacks, 918
cryptographic and cipher suites, 819–821
cryptosystems, 809–812
DH (Diffie-Hellman-Merkle Key Agreement), 813–814
ECC (Elliptic Curve Cryptography), 815
hash functions and message digests, 817–818
message authentication codes, 818–819
message syntax, 823
nonces and salt, 816
overview of, 809
PFS (Perfect Forward Secrecy), 815
pseudorandom numbers, generators, and function families, 815–816
RSA (Rivest, Shamir, and Adleman) public key cryptography, 812–813
signcryption, 814–815
CS (candidate sets), in source address selection, 223–224

CS (cipher suites)
definition of, 938
overview of, 819–821
in TLS, 878
CS (Class Selector), 938
CSMA/CA (carrier sense, multiple access with collision avoidance)
DCF as form of, 120
definition of, 938
WLANs (wireless LANs), 84
CSMA/CD (carrier sensed, multiple access with collision detection)
definition of, 938
Ethernet interface and, 80–81
CSPRNGs (cryptographically strong PRNGs), 816, 938
CSR (Classless Static Route) parameter, 246
CSRG (Computer Systems Research Group), 24, 938
CTCP (Compound TCP) algorithm, 779–781, 938
CTR (Counter) mode
definition of, 938
operating modes of encryption algorithms, 820
CTS (clear to send)
carrier sense and, 121
definition of, 938
Wi-Fi control frames, 115
CUBIC algorithm, for
TCP congestion control, 775–776
Current-state records, IGMP/MLD group membership reports, 457
CW (contention window)
definition of, 938
in MAC, 122
cwnd (congestion window) in TCP
comparing slow start with congestion avoidance, 736–737
congestion avoidance algorithm and, 734–736
CWV (Congestion Window Validation), 742–744
definition of, 938
Eifel Response Algorithm and, 744–745
FACK (forward acknowledgment) and, 741
overview of, 729–730
SACK congestion control and, 740–741
slow start algorithm and, 732–734
standard TCP algorithm and, 738
Tahoe, Reno, and Fast Recovery and, 737
undoing changes in, 762–766
CWR (Congestion Window Reducing)
CWR bit, 784
definition of, 938
fast retransmit events, 761–762
local congestion events, 764
sender pause and, 753
CWV (Congestion Window Validation), 742–744, 938
Cyclic Redundancy Checks. *See* CRCs (Cyclic Redundancy Checks)

D

DAD (duplicate address detection)
definition of, 938
DHCPv6, 259–260
IPv6 addresses and, 277–278
MLD messages and, 457
Neighbor Discovery protocol and, 253
RA and RS messages and, 280–282
Daemen, Joan, 811
Data Encryption Standard. *See* DES (Data Encryption Standard)
Data flow, TCP. *See* TCP data flow
Data frames
fragmentation and aggregation, 94–95
Wi-Fi, 116–119
Data labels, DNS names and, 523
Data-link layer, of OSI model, 9
Data types, resource record categories, 528
Datagram Congestion Control Protocol. *See* DCCP (Datagram Congestion Control Protocol)
Datagram TLS. *See* DTLS (Datagram TLS)
Datagrams
attribution of, 26
fragmenting, 148
important concepts in development of network architecture, 5–6
in protocol architecture, 3–6
receiving multicast, 447–449
sending multicast, 446–447
spoofing attacks and, 25–26
TLS with. *See* DTLS (Datagram TLS)
Datagrams, IP
direct delivery of, 167
DNS messages using IPv4 datagrams, 525
fragmentation of, 488
of ICMP messages within, 354–355
IPv4, 182
TCP encapsulation in, 587
Datagrams, UDP
DNS messages using, 525
encapsulation, 474
fragmentation of, 488–492
maximum size, 497–498
translating UDP/IPv4 and UDP/IPv6 datagrams, 505–506
truncation of, 498
Davies, Donald, 1
Day, J., 2

DCCP (Datagram Congestion Control Protocol)
definition of, 938
NAT and, 309
transport protocols in TCP/IP suite, 16
DCF (distributed coordinating function), in Wi-Fi
collision avoidance/backoff procedure, 121–122
definition of, 938
options for controlling sharing of wireless medium, 120
DDDS (Dynamic Delegation Discovery System)
definition of, 938
ENUM records and, 551–552
DNS NAPTR records and, 549
URI/URN resolution and, 553
DDNS (Dynamic DNS)
mapping DNS to DHCP addresses, 286
supporting DNS Update, 567
DDoS (distributed DoS)
attacks related to Internet architecture, 26
definition of, 939
Deadlock, Nagle algorithm resulting in, 699
Decimal notation, expressing IP addresses in, 32
Default router, IP forwarding, 208
Deferred authentication, in DHCP, 272
Defragmentation, of data frames. *See also* Fragmentation, 117
Delay-based congestion control
buffer bloat and, 782
CTCP (Compound TCP) algorithm, 779–781
FAST TCP algorithm, 778–779
overview of, 777
TCPW (TCP Westwood) algorithm, 779
Vegas TCP algorithm, 777–778
Delayed ACKs
interaction with Nagle algorithm, 699
with piggybacking, 692
in TCP data flow, 695–696
Delegated path discovery (DPD)
certificate validation and, 831
definition of, 940
Delegated path validation (DPV)
certificate validation and, 831
definition of, 940
Delegation
classless .in-addr.arpa delegation, 539
DNS zones and, 516
Delegation signer (DS) resource record
definition of, 940
DNSSEC, 897–898
Delivery to multiple locations, broadcasting and multicasting for, 435
Demilitarized zones. *See* DMZ (demilitarized zones)
Demultiplexing
identifiers in, 11
implementation and design and, 10–13
TCP/IP suite and, 16–17
Denial-of-service. *See* DoS (denial-of-service)
DER (Distinguished Encoding Rules), 869, 939
DES (Data Encryption Standard). *See also* Triple-DES
definition of, 939
dictionary attacks and, 816
as symmetric encryption algorithm, 811
Destination address selection algorithm, in IP host models, 224–225
Destination cache, 403
Destination (DST) address, in Ethernet frame format, 85
Destination IP address
host processing of IP datagrams, 220–221
in IP datagrams, 186
in IP forwarding, 209
IPv6 header fields, 196
Routing header fields, 201
selection by hosts, 222–223
Teredo tunneling and, 485
Destination metrics, TCP timeout/retransmission, 685–686
Destination Options, IPv6, 196
Destination unreachable messages, ICMP
overview of, 364
PTB (Packet Too Big), 612
requesting connection to nonexistent ports, 626
UDP datagram and, 480
Detecting Network Attachment (DNA), 241, 939
Detection algorithm, for spurious timeouts and retransmissions, 677
Detection, of congestion, 728–729
DF (Don't Fragment), 939
DH (Diffie-Hellman-Merkle Key Agreement), 813–814, 939
DHCP (Dynamic Host Configuration Protocol)
address pools and leases, 235–236
attacks related to, 292
authentication, 271–273
automatic address assignment, 67
BOOTP message format, 236–238
BOOTP options, 238–239
definition of, 939
DHCP/DNS interaction, 285–286
DHCPACK message, 250
DHCPDISCOVER message, 244–247
DHCPNAK message, 243
DHCPOFFER message, 247–248
DHCPREQUEST message, 241–243, 248–249

DHCP (Dynamic Host Configuration Protocol), *continued*
location information, 274–275
manual configuration, 265–266
mobility and handoff information, 275
operation of, 239–241
overview of, 234–235
PPPoE (PPP over Ethernet) and, 286–291
Rapid Commit option, 273–274
reconfigure extension, 273
relays, 267–271
snooping, 276
state machine, 251–252
subnet mask configuration, 39
summary and references, 292–298
DHCP Unique Identifier. *See* DUID (DHCP Unique Identifier)
DHCPACK message, 241, 250
DHCPDECLINE message, 241
DHCPDISCOVER message, 240, 244–247
DHCPINFORM message, 241
DHCPLEASEQUERY message, 269
DHCPNAK message, 241, 243
DHCPOFFER message, 240, 247–248
DHCPRELEASE message, 241
DHCPREQUEST message, 239–243, 248–249
DHCPv6
DAD (Neighbor Solicitation), 259–260
DUID (DHCP Unique Identifier), 256–257
IA (Identity Association), 255–256
IPv6 address lifecycle, 252–253
manual configuration, 250–251
message format, 253–255
operation of, 257–258
overview of, 252
prefix delegation, 266–267
REQUEST message, 264–265
router solicitation and advertisement, 260–263
Dictionary attacks
DES and, 816
security protocol-related, 918
Differentiated Services Code Point (DSCP), 188–190, 940
Differentiated Services field. *See* DS (Differentiated Services) field
Diffie-Hellman-Merkle Key Agreement (DH), 813–814, 939
DIFS (distributed inter-frame space), in Wi-Fi
carrier sense and, 120–121
definition of, 939
Digest challenge, STUN mechanisms and, 325
Digests, message digests, 817–818
Digital, Intel, Xerox (DIX)
definition of, 939
Ethernet, 82
Digital Living Network Alliance (DLNA), 939
Digital Object Identifier (DOI), 939
Digital Signature Algorithm (DSA), 821, 940
Digital Signature Standard (DSS), 821, 940
Digital signatures
in cipher suites, 821
in public key cryptography, 812
RSA Signature option, 416
Digital subscriber line. *See* DSL (digital subscriber line)
Direct delivery
IP forwarding, 210–212
with IPv4, 167–169
Directed broadcast, 43
Direction specification, in firewall rules, 335
Discard Request messages, in LCP operation, 134
Discovery problem, in p2p networks, 22
Discrete log problem, in DH (Diffie-Hellman) encryption, 814
Distinguished Encoding Rules (DER), 869, 939
Distributed coordinating function. *See* DCF (distributed coordinating function)
Distributed DoS (DDoS)
attacks related to Internet architecture, 26
definition of, 939
Distributed inter-frame space (DIFS)
carrier sense and, 120–121
definition of, 939
Distribution service (DS)
definition of, 940
Wi-Fi, 112
Distributions, TCP/IP suite, 24–25
DIX (Digital, Intel, Xerox)
definition of, 939
Ethernet, 82
DKIM (DomainKeys Identified Mail)
definition of, 939
DKIM signatures, 916
example using, 916–918
overview of, 915–916
DLNA (Digital Living Network Alliance), 939
DMZ (demilitarized zones)
definition of, 939
DNS queries and, 565–567
packet-filtering firewalls and, 300
unicast addresses and, 67–68
DNA (Detecting Network Attachment), 241, 939
DNAME resource records, DNS, 536, 939
DNS-0x20, 572

DNS (Domain Name System)
address and name server records, 529–530
answer, authority, and additional information section formats, 526–527
attacks related to, 571–572
AXFR (full zone transfer) messages, 559–561
caching and, 517–518
CNAME (canonical name) records, 534–536
definition of, 939
DHCP and, 233, 285–286
DNS notify, 564–565
dynamic updates, 555–558
ENUM records, 551–552
example using resource record types, 530–534
extension format (EDNSO), 524–525
introduction to, 511–512
IXFR (incremental zone transfer) messages, 561–563
LDAP and, 570–571
LLMNR and mDNS, 569–570
mDNS (Multicast DNS), 444–445
message format, 520–524
MX (mail exchanger) records, 544–545
name resolution process, 518–520
name servers and zones, 516–517
name space, 512–514
naming syntax, 514–516
NAPTR (name authority pointer) records, 549–551
open DNS servers and DDNS, 567
OPT (option) pseudo records, 547–548
PTR (pointer) records, 536–541
question (query) and zone section format, 526
resource record types, 527–529
S-NAPTR and U-NAPTR, 554–555
security. *See* DNSSEC (DNS Security)
SIP records, 552
SOA (start of authority) records, 541–544
sort lists, round-robin, and split DNS, 565–567
SPF (sender policy framework) and TXT records, 545–547
SRV (service) records, 548–549
summary and references, 572–578
TCP/IP suite and, 19
translating DNS from IPv4 to IPv6, 568–569
transparency and extensibility, 567–568
URI/URN resolution, 553–554
well-known ports, 18, 525–526
zone transfers, 558–559
DNS Notify
initiating zone transfers, 525
necessity of zone transfers, 518
DNS proxy, 568
DNS Security. *See* DNSSEC (DNS Security)
DNS servers
caching, 517–518
gTLD servers, 519–520
primary and secondary, 517
response to DNS queries, 565–567
root servers, 518
zones, 516–517
DNS Update, DDNS support for, 567
DNS64
definition of, 939
DNSSEC with, 915
translating DNS from IPv4 to IPv6, 568–569
DNSKEY resource record
definition of, 939
DNSSEC and, 896–897
signed zones and zone cuts, 903
DNSSEC (DNS Security)
canonical orderings and forms, 902–903
definition of, 939
DNS-related attacks and, 571
DNS64 and, 915
DNSKEY resource record in, 896–897
DS (delegation signer) resource record in, 897–898
NSEC (NextSECure) resource record in, 898–901
operation of, 902
overview of, 894–896
resolver operation example, 903–911
resource records, 896
RRSIG (Resource Record Signature) resource record in, 901–902
signed zones and zone cuts, 903
transaction authentication, 911–915
DNSSL (DNS Search List) option, in ND, 422–423
DOI (Digital Object Identifier), 939
Domain hacks, 514
Domain Keys Identified Mail. *See* DKIM (Domain Keys Identified Mail)
Domain Name System. *See* DNS (Domain Name System)
Domain names, DNS Search List option, 422–423
Domain-specific keys (DSRK)
definition of, 940
key derivation in EAP, 838
Domain-specific usage-specific root keys (DSUSRKs)
definition of, 940
key derivation in EAP, 838
Domains
DNS and, 19
in DNS name space, 512
Done message, ICMP, 388–390
Don't Fragment (DF), 939

DoS (denial-of-service)
definition of, 939
DNS attacks, 571
DTLS (Datagram TLS) protection, 894
IGMP or MLD attacks, 469–470
Internet architecture attacks, 26
system configuration attacks, 292
TCP attacks, 640
TCP timeout/retransmission attacks, 686
types of threats to network communication, 808
UDP attacks, 506
Dotted-decimal notation
IP addresses in, 32, 537
of subnet masks, 39
DPD (delegated path discovery)
certificate validation and, 831
definition of, 940
DPV (delegated path validation)
certificate validation and, 831
definition of, 940
Drive-by attacks, 806
Drop precedence, assigned to datagrams, 191
DS (delegation signer) resource record
definition of, 940
DNSSEC, 897–898
DS (Differentiated Services) field
definition of, 940
ICMP Parameter Problem and, 379
in IP header, 183
in IP protocol, 188–192
DS (distribution service)
definition of, 940
Wi-Fi, 112
DS-Lite (Dual-Stack Lite)
definition of, 940
IPv4/IPv6 translation, 339–340
DSA (Digital Signature Algorithm), 821, 940
DSACK (duplicate SACK) extension
definition of, 940
Eifel Detection Algorithm and, 679
Eifel Response Algorithm and, 681
overview of, 677–679
DSCP (Differentiated Services Code Point), 188–190, 940
DSL (digital subscriber line)
buffer bloat and, 781
definition of, 940
overview of, 4
PPPoE and, 286–287
DSRK (domain-specific keys)
definition of, 940
key derivation in EAP, 838
DSS (Digital Signature Standard), 821, 940
DST (Destination) address, in Ethernet frame format, 85
DSUSRKs (domain-specific usage-specific root keys)
definition of, 940
key derivation in EAP, 838
DTCP (Dynamic Tunnel Configuration Protocol), 154
DTLS (Datagram TLS)
definition of, 940
DoS protection, 894
example of use of, 884–891
handshake protocol, 892–894
overview of, 876–877
record layer, 891–892
Dual-Stack Lite (DS-Lite)
definition of, 940
IPv4/IPv6 translation, 339–340
DUID (DHCP Unique Identifier)
definition of, 940
DHCPDISCOVER message and, 246
types of, 256–257
DupACK spoofing, TCP congestion control attacks, 785–786
Duplex modes
duplex mismatch, 96
overview of, 94–96
Duplicate ACK threshold (dupthresh), in TCP
in fast retransmit, 667
NewReno algorithm and, 739
packet reordering and, 683
Duplicate acknowledgements, congestion control and, 589
Duplicate address detection. *See* DAD (duplicate address detection)
Duplicate SACK. *See* DSACK (duplicate SACK) extension
Dupthresh. *See* Duplicate ACK threshold (dupthresh)
Duration field, in frame transmission, 121
DWORD value, 518
Dynamic Delegation Discovery System. *See* DDDS (Dynamic Delegation Discovery System)
Dynamic DNS (DDNS)
mapping DNS to DHCP addresses, 286
supporting DNS Update, 567
Dynamic Host Configuration Protocol. *See* DHCP (Dynamic Host Configuration Protocol)
Dynamic/private ports, 18
Dynamic Tunnel Configuration Protocol (DTCP), 154
Dynamic updates, DNS, 555–558

E

E-mail, PGP (Pretty Good Privacy) encryption for, 822
EAP (Extensible Authentication Protocol)
  attacks related to, 918
  definition of, 940
  ERP (EAP Re-authentication Protocol), 839
  IKE protocol and, 851–852
  key derivation, 838
  layers of, 836–837
  methods, 837–838
  network security and, 833–834
  packet format, 835
  for PPP authentication, 141
  request/response messages, 835–836
  in Wi-Fi security, 129
EAP-FAST (EAP-Flexible Authentication via Security Tunneling), 940
EAP Re-authentication Protocol (ERP), 839, 941
EAP-TTLS (EAP-Tunneled Transport Layer Security), 838, 940
EAPoL (EAP over LAN), 833–834, 940
Eavesdropping
  link layer attacks, 155
  types of threats to network communication, 808
EC2N (Elliptic Curve groups modulo a power 2), 821, 941
ECC (Elliptic Curve Cryptography), 815, 821
ECC (Error Correcting Code), 579, 941
ECDSA (Elliptic Curve Digital Signature Algorithm), 941
ECE (ECN Echo), 589, 941
Echo Request/Reply (ping) messages
  example of broadcasting, 438–439
  ICMP, 380–383
  in LCP operation, 134
  Redirect message and, 374
  sending from link-local unicast address, 445
  translating ICMPv6 to ICMPv4, 426
ECN-Capable Transport (ECT), 783, 941
ECN Echo (ECE), 589, 941
ECN (Explicit Congestion Notification)
  delay-based congestion control and, 777
  detecting congestion, 728
  ECN-Echo bit, 784
  ICMP Parameter Problem and, 379
  IP header fields, 183, 188–192
  TCP congestion control attacks and, 786
  TCP header fields, 782–785
ECP (Elliptic Curve groups modulo a Prime), 821, 941
EDCA (enhanced DCF channel access)
  definition of, 941
  UPs (user priorities), 123
  Wi-Fi mesh and, 130
Edge ports, RSTP (Rapid Spanning Tree Protocol), 110
Editors, NAT, 315
EDNS0 (Extension format for DNS)
  definition of, 941
  DNS (Domain Name System), 524–525
  DNSSEC, 895–896
EF (Expedited Forwarding), 191, 941
EFO (Extended Flags option)
  definition of, 941
  Router Advertisement Flags Extension option, 420–421
Eifel Detection Algorithm, 679–680
Eifel Response Algorithm
  handling spurious RTOs in congestion control, 744–745
  responding to spurious transmissions, 680–682
EIFS (extended interframe space), in Wi-Fi
  carrier sense and, 121
  definition of, 941
Eligible rate estimate (ERE)
  definition of, 941
  in TCPW congestion control, 779
Elliptic Curve Cryptography (ECC), 815, 821
Elliptic Curve Digital Signature Algorithm (ECDSA), 941
Elliptic Curve groups modulo a power 2 (EC2N), 821, 941
Elliptic Curve groups modulo a Prime (ECP), 821, 941
Elliptic curves, 815
Embedded devices, setting IPv4 addresses for, 178
EMSKs (extended MSKs)
  definition of, 941
  key derivation in EAP, 838
Encapsulating Security Payload. *See* ESP (Encapsulating Security Payload)
Encapsulation
  definition of, 10
  of ICMP messages within IP datagrams, 354–355
  IGMP/MLD, 453–454
  implementation and design and, 10–13
  TCP encapsulation in IP datagrams, 587
  TCP/IP suite and, 16–17
  tunnel encapsulation limits in IPv6, 198
  of UDP datagram, 474
Encryption. *See also* Cryptography
  MPPE (Microsoft Point-to-Point Encryption), 145
  spoofing attacks and, 226
  TCP-AO (Authentication Option), 612
  in Wi-Fi security, 129

End system, for protocol suites, 12
End-to-end argument, in protocol architecture, 6
End-to-end checksum, in UDP, 473, 475
Endpoint discriminator, LCP options, 138
Endpoints, TCP, 588, 595–596
Enforcement Point (EP), in PANA, 839, 941
Enhanced DCF channel access. *See* EDCA (enhanced DCF channel access)
ENUM records
  definition of, 941
  DNS resource record types, 551–552
Enumeration attacks, 919
EOL (End of List), TCP header options, 605
EP (Enforcement Point), in PANA, 941
Ephemeral port numbers, 18
EQM (equal modulation)
  definition of, 941
  higher throughput (802.11n) support and, 127
Equation-based rate control, 768
ERE (eligible rate estimate)
  definition of, 941
  in TCPW congestion control, 779
ERP (EAP Re-authentication Protocol), 839, 941
Error control, in protocol architecture, 7–8
Error Correcting Code (ECC), 579, 941
Error messages, ICMP
  destination unreachable, 364–372
  extended and multipart messages, 363–364
  overview of, 309, 361–363
  Parameter Problem message, 379–380
  Redirect message, 372–375
  time exceeded message, 375–378
  translating ICMPv4 to ICMPv6, 424–425
  translating ICMPv6 to ICMPv4, 427
ESN (Extended Sequence Number), in IPsec, 856, 942
ESP (Encapsulating Security Payload)
  definition of, 942
  ESP-NULL, WESP, and traffic visibility, 863–864
  in IPSec, 217, 841
  overview of, 858
  transport and tunnel modes, 858–863
ESP-NULL, 863–864
ESS (extended service set), Wi-Fi, 112
ESSID (extended service set identifiers)
  definition of, 942
  Wi-Fi, 112
Established connections, TCP, 596
ESTABLISHED state in TCP
  half-open connections and, 628
  incoming connection queue and, 636
  simultaneous open and close transitions, 625
  TCP port numbers and, 632–634
  TCP state transitions, 618
Ethernet (IEEE 802.3)
  autonegotiation in, 95
  converting IP multicast addresses to MAC addresses, 442–444
  flow control, 98
  frames, 84–86
  frames sizes, 88–89
  integrity checking on frames, 86–88
  LAN/MAN standards, 82–84
  MAC addresses, 16
  MTU (maximum transmission unit), 506
  overview of, 80–82
  power saving, 96–97
  speeds, 81
  supporting broadcasting at link layer, 437
Ethernet interfaces, Promiscuous mode, 155
Ethernet type field, 16
`ethtool`, Linux program for checking full duplex support, 94
EUI (extended unique identifier)
  definition of, 942
  formats of IPv6 addresses, 44–45
EV-DO (Evolution, Data Optimized (or Only)), 942
EV (Extended Validation), 942
Exchange of database records. *See* Zone transfers
Expedited Forwarding (EF), 191, 941
Experimental category, RFCs and, 23
Experimental Values, neighbor discovery in IPv6, 423
Explicit Congestion Notification. *See* ECN (Explicit Congestion Notification)
Explicit sending, in congestion control, 583
Explicit signaling, in congestion control, 728
Exponential backoff
  binary, 650
  SWS (silly windows syndrome) and, 713
  in TCP connection timeout, 604
Extended and multipart messages, ICMP, 363–364
Extended Flags option (EFO)
  definition of, 941
  Router Advertisement Flags Extension option, 420–421
Extended interframe space (EIFS)
  carrier sense and, 121
  definition of, 941
Extended MSKs (EMSKs)
  definition of, 941
  key derivation in EAP, 838
Extended Sequence Number (ESN), in IPsec, 856, 942
Extended service set (ESS), Wi-Fi, 112
Extended service set identifiers (ESSID)
  definition of, 942
  Wi-Fi, 112

Extended unique identifier (EUI)
definition of, 942
formats of IPv6 addresses, 44–45
Extended Validation (EV), certificates, 942
Extensibility, DNS, 567–568
Extensible Authentication Protocol. *See* EAP (Extensible Authentication Protocol)
Extensible Markup Language (XML)
common use with Web pages, 338
definition of, 962
Extensible Messaging and Presence Protocol (XMPP), 333, 962
Extension data structure, appended to ICMP messages, 363–364
Extensions
DNS. *See* EDNS0
identifiers, 44
IP header, 182–183
IPv6 header, 194–196
TLS, 883–884
Extranets, 20

F

F-RTO (Forward-RTO Recovery), in TCP
definition of, 942
for detecting spurious transmissions, 680
FACK (forward acknowledgment), in TCP
definition of, 942
for TCP congestion control, 741–742
Fast Recovery algorithm, 737–738
Fast retransmit
event, 761
example of, 668–671
introduction to, 647
local congestion and, 759–762
overview of, 667–668
SACK recovery and, 757–759
FAST TCP algorithm, 778–779
Fate sharing, in protocol architecture, 6–7
FCFS (first-come-first-served)
definition of, 942
packet processing, 4
FCS (Frame Check Sequence)
definition of, 942
integrity checking in Ethernet frames, 88
verifying message integrity, 817
FEC (forward error correction)
definition of, 942
higher throughput (802.11n) support and, 128
FIFO (first-in-first-out)
definition of, 942
queue management and, 782
scheduling packets, 4
File (Boot File Name) field, DHCP/BOOTP message format, 238–239
File Transfer Protocol. *See* FTP (File Transfer Protocol)
Filter-mode-change records, IGMP/MLD group membership reports, 457
Filters
host address filtering, 449–451
IGMP/MLD processing and, 456
`iptables`, 335
NAT, 312–313
packet-filtering firewalls, 300
web proxies operating as content filters, 302
FIN segments, in TCP
active and passive closers and, 597–598
connection completion and, 766
definition of, 942
half-close operation and, 598–599
half-open connections and, 628
sequence numbers in, 603
Fingerprinting, TCP keepalive attacks, 802
Fingerprints, or digest of message, 817–818
FIN_WAIT_1 state, TCP state transitions, 618, 625
FIN_WAIT_2 state, TCP state transitions, 618, 625
Firewalls
attacks related to, 345–346
configuring, 334
direct interaction with, 338–339
IP addresses and, 67
overview of, 300
packet-filtering firewalls, 300–301
proxy firewalls, 301–303
rules for packet-filtering, 335–336
summary and references, 345–346
First-come-first-served (FCFS)
definition of, 942
packet processing, 4
First-in-first-out. *See* FIFO (first-in-first-out)
Flags
DHCP/BOOTP message format, 236
GRE tunnels and, 150
IPv6 multicast addresses, 57, 58
Flooding attacks, 102, 428
Flow control. *See also* Congestion control
link layer and, 98
in protocol architecture, 7–8
rate-based and window-based, 583
in UPD server design, 505
Flow control, in TCP
example of dynamic window size adjustment and flow control, 705–708
large buffers and auto-tuning, 715–719
overview of, 700–701

Flow control, in TCP, *continued*
sliding window protocol, 701–704
SWS (silly windows syndrome), 708–715
TCP header and, 590
zero windows and TCP persistent timer, 704–705
FMIP (Mobile IP with Fast Handovers), 388, 942
Foreign IP addresses
TCP port numbers and, 632
TCP server restrictions on foreign endpoints, 635–636
Forgery, message authentication codes protecting against, 818
Forward acknowledgment (FACK), in TCP
definition of, 942
for TCP congestion control, 741–742
Forward error correction (FEC)
definition of, 942
higher throughput (802.11n) support and, 128
Forward-RTO Recovery (F-RTO)
definition of, 942
for detecting spurious transmissions, 680
Forwarding actions, IP forwarding, 209–210
Forwarding datagrams, 14
Forwarding tables, IP forwarding, 208–209
Four-message exchange operations, of DHCPv6, 265–266
FQDN (fully qualified domain names)
definition of, 942
DHCPv6 and, 260
vs. unqualified domain names, 515
Fraggle attacks, UDP-related attacks, 506
Fragment header, IPv6 protocol, 203–208
Fragment number, data frame fragmentation, 117
Fragment Offset field
in IPv6 Fragment header, 203–205
in UDP fragmentation, 489
Fragmentation
of datagrams, 14
of IGMP packets, 470
of Internet traffic, 506
IP fragmentation. *See* IP fragmentation
UDP/IPv4, 488–492
of Wi-Fi frames, 116–119
Frame Check Sequence. *See* FCS (Frame Check Sequence)
Frame Control Word
in MPDU, 113
PSM (power save mode) and, 119
Retry bit and, 116
Frame Relay, best-effort delivery, 7
Frames, Ethernet
802.3 standard, 84–86
ARP frame format, 170–171
ARP reply frames, 168
ARP request frames, 167, 173–174
integrity checking on, 86–88
link-layer PDUs, 14
payload of, 16
sizes, 88–89
Frames, PPP, 131–132
Frames, Wi-Fi
control frames, 115–116
data frames, 116–119
management frames, 113–115
overview of, 113
FreeBSD
broadcast addresses, 441
incoming connection queue and, 638
standards and, 24
Frequencies, Wi-Fi, 124
Freshness property, authentication protocols and, 816
FTP (File Transfer Protocol)
definition of, 942
NAT and, 304
TCP/IP suite and, 13
Full duplex
support, 94–95
TCP service to application layer, 587
Fully qualified domain names. *See* FQDN (fully qualified domain names)

G

Galois/Counter Mode. *See* GCM (Galois/Counter Mode)
Galois MAC (GMAC), 819–820, 943
Gateway or Router IP Address (Giaddr) field
DHCP/BOOTP message format, 238
LDRAs and, 271
Gateways
application layer. *See* ALGs (application layer gateways)
between packet-switching networks, 1
between protocols, 20
RED (Random Early Detection) gateways, 783–785
GCKSs (group controller/key servers)
definition of, 942
multicast support in IPSec, 864–865
GCM (Galois/Counter Mode)
AES and, 819
definition of, 942
operating modes of encryption algorithms, 820
GDOI (Group Domain of Interpretation), 942
GENA (General Event Notification Architecture), 338, 942
Generator polynomial, CRC and, 86–87
Generic Attribute Registration Protocol (GVRP), 943

Generic Multicast Registration Protocol (GMRP), 943
Generic Routing Encapsulation. *See* GRE (Generic Routing Encapsulation)
Generic Security Services API (GSS-API), 943
Generic top-level domains (gTLD), 512–514, 943
GI (guard interval)
  definition of, 942
  higher throughput (802.11n) support and, 127
Giaddr (Gateway or Router IP Address) field
  DHCP/BOOTP message format, 238
  LDRAs and, 271
GKM (group key management)
  definition of, 943
  multicast support in IPSec, 864
Global Internet. *See* Internet
Global scope
  configuring global addresses with SLAAC, 278
  of IPv6 addresses, 43
  of multicast addresses, 53
GLOP addressing, IP multicast, 54–56
GMAC (Galois MAC), 819–820, 943
GMRP (Generic Multicast Registration Protocol), 943
GPAD (Group PAD), 864, 943
Granularity, of TCP clock, 654
Gratuitous ARP, 175–176
GRE (Generic Routing Encapsulation)
  definition of, 943
  establishing tunneling with, 149–153
  link layer attacks and, 156
  NAT and tunneled packets, 310
Greenfield mode, 802.11n operating modes, 128
Group addresses. *See* Multicast addresses
Group controller/key servers (GCKSs)
  definition of, 942
  multicast support in IPSec, 864–865
Group Domain of Interpretation (GDOI), 942
Group key management (GKM)
  definition of, 943
  multicast support in IPSec, 864
Group members (Group Member Part), IGMP/MLD processing by, 454–457
Group membership
  displaying IP group membership, 446–447
  IGMP membership reports, 455
Group PAD (GPAD), 864, 943
Group Secure Association Key Management (GSAKMP), 943
Group security associations (GSAs)
  definition of, 943
  multicast support in IPSec, 864
Group SPD (GSPD), 943
GSAKMP (Group Secure Association Key Management), 943
GSAs (group security associations)
  definition of, 943
  multicast support in IPSec, 864
GSPD (Group SPD), 943
GSS-API (Generic Security Services API), 943
gTLD (generic top-level domains), 512–514, 943
Guard interval (GI)
  definition of, 942
  higher throughput (802.11n) support and, 127
GVRP (Generic Attribute Registration Protocol), 943

H

HAIO (Home Agent Information Option), 943
Hairpinning (HP), 314, 486
Half-close operation, TCP connections, 598–599
Half-open connections, keepalives detecting, 794
Handoff information, DHCP, 275
Handover Key Request/Reply options, in ND, 422–423
Handshaking protocols
  DTLS (Datagram TLS), 892–894
  three-way handshake in TCP, 597, 640
  TLS, 880–883
HAs (home agents)
  definition of, 943
  Home Agent Information option in ND, 412–413
  ICMP Home Agent Discovery Request message, 386
  ICMP Mobile Prefix Solicitation message, 387–388
  in Mobile IP, 216–217
Hash-based addresses (HBAs), 405
Hash functions
  CGAs (cryptographically generated addresses) and, 404–405
  cryptographic, 817–818
  initial sequence numbers in TCP and, 601–602
  TCP-AO (Authentication Option), 612
HBAs (hash-based addresses), 405
HC (hybrid coordinator), in HCCA, 123
HCCA (HFCA-controlled channel access), 123
HCF (hybrid coordination function)
  definition of, 943
  options for controlling sharing of wireless medium, 120
  QoS and, 122–123
HDLC (High-Level Data Link Control)
  Address and Control fields, 132
  definition of, 943
  LCP links based on, 131
Header compression, in PPP
  CCP (Compression Control Protocol), 139
  PPP (Point-to-Point Protocol), 142–143

Header, ICMP
ICMP Parameter Problem and, 379–380
Redirected Header option in ND, 411
Header, IP
CE (Congestion Experienced) indicator in, 783–784
Checksum field, 186–188
DS and ECN fields, 188–192
extensions, 182–183
header compression, 142–143
IPv6 extensions, 194–196
IPv6 Fragment header, 203–208
IPv6 RH (Router header), 200–203, 955
Mobile IP, 216
overview of, 183–186
Header, TCP
connection options, 605–606
fields in, 588–590
MSS (Maximum Segment Size) option, 606–607
PAWS (Protection against Wrapped Sequence Numbers), 610–611
SACK (selective acknowledgement) option, 607
TCP-AO (Authentication Option), 612
Timestamps option, 608–610
UTO (User Timeout) option, 611–612
WSCALE (Window Scale) option, 608
Header, UDP, 474–476, 481–482
HELD (HTTP-enabled Location Delivery)
definition of, 943
location information in DHCP, 274
Hexadecimal notation
expressing IP addresses in, 32–33
IPv6 addresses in, 537
HFCA-controlled channel access (HCCA), 123
Hierarchical routing, 48
High-Level Data Link Control. *See* HDLC (High-Level Data Link Control)
High-speed environments, congestion control in, 770
High-Speed Packet Access (HSPA), 944
High Throughput (HT), 128, 944
HighSpeed TCP (HSTCP), 770–772, 944
Hijacking attacks
firewalls and NATs and, 345
TCP-related attacks, 641
HIP (Host Identity Protocol)
definition of, 943
Identifier/locator separating protocols, 70
Historic category, RFCs and, 23
HK-LIFETIME, Handover Keys, 422–423
Hlen (HW Len) field, DHCP/BOOTP message format, 236
HMAC (keyed-hash message authentication code), 818–819, 943
HoA (home address)
definition of, 944
of IPv6 node, 199
in Mobile IP, 216–218
Hole punching, in NAT traversal, 317
Home address. *See* HoA (home address)
Home Agent Discovery Request message, ICMP, 386
Home Agent Information Option (HAIO), 943
Home Agent Information option, in ND, 412–413
Home agents. *See* HAs (home agents)
Home Test (HoT)
definition of, 944
mobility messages in RRP, 218–219
Home Test Init (HoTI)
definition of, 944
mobility messages in RRP, 218–219
Hop-by-hop
IP forwarding, 209
protocols, 12
Hop-by-hop options (HOPOPTs)
definition of, 944
IPv6, 196
Hop Limit field
ICMP Time Exceeded message, 375
IPv6, 199
MRD (Multicast Router Discovery) and, 394
ND messages and, 396
HOPOPTs (hop-by-hop options)
definition of, 944
IPv6, 196
Hops field, DHCP/BOOTP message format, 236
Host addresses, 35
Host fields, in IP addresses, 37
Host Identity Protocol (HIP)
definition of, 943
Identifier/locator separating protocols, 70
Host models, IP
address selection, 222–223
destination address selection algorithm, 224–225
overview of, 220–222
source address selection algorithm, 223–224
Host names, 19
Host number, in IP addresses, 35
Host Requirements RFCs, 23
Host unreachable message, ICMP, 364
Hosts
ARP request to nonexistent, 173–174
host address filtering, 449–451
keepalives detecting state of peer host, 795
server host crashes and does not reboot (keepalive scenarios), 796
server host crashes and reboots (keepalive scenarios), 797–799

server host unreachable (keepalive scenarios), 799–800
in small networks, 11–13
HoT (Home Test)
definition of, 944
mobility messages in RRP, 218–219
HoTI (Home Test Init)
definition of, 944
mobility messages in RRP, 218–219
HP (Hairpinning), 314, 486
HSPA (High-Speed Packet Access), 944
HSTCP (HighSpeed TCP), 770–772, 944
HT (High Throughput), 128, 944
HTML (Hyper-Text Markup Language), 944
HTTP-enabled Location Delivery (HELD)
definition of, 943
location information in DHCP, 274
HTTP (Hypertext Transfer Protocol)
definition of, 944
proxy firewalls, 302–303
well-known port for, 18
HTTP over SSL/TLS (HTTPS), 944
HTTP using UDP (HTTPMU), 338, 944
HTTPMU (HTTP using UDP), 338, 944
HTTPS (HTTP over SSL/TLS), 944
Htype (HW Type) field, DHCP/BOOTP message format, 236
HWRP (Hybrid Wireless Routing Protocol)
definition of, 944
Wi-Fi mesh and, 130
Hybrid coordination function. *See* HCF (hybrid coordination function)
Hybrid coordinator (HC), in HCCA, 123
Hybrid cryptosystems, 812
Hybrid Wireless Routing Protocol (HWRP)
definition of, 944
Wi-Fi mesh and, 130
Hyper-Text Markup Language (HTML), 944
Hypertext Transfer Protocol. *See* HTTP (Hypertext Transfer Protocol)

I

IA (Identity Association)
definition of, 944
in DHCPv6, 255–256
IAB (Internet Architecture Board), 22, 944
IAID (Identity Association Identifier)
definition of, 944
in DHCP, 246, 255–256
IANA (Internet Assigned Numbers Authority)
allocation of IP addresses and, 62
allocation of IPv6 multicast addresses, 58
definition of, 944
IKE registry of values, 849
OUI (Organizationally Unique Identifier), 442–444
port number assignments, 18, 632
registry for Sec values, 405
registry for SRV values, 548–549
URI, 553
IANA Service Name and Transport Protocol Port Number (ISPR), 548
IBSS (independent basic service set)
definition of, 944
Wi-Fi, 112
ICANN (Internet Corporation for Assigned Names and Numbers)
definition of, 944
DNSSEC zones list, 904
TLD management, 512
ICE (Interactive Connectivity Establishment), 332–334, 944
ICMP fix-up, 309
ICMP (Internet Control Message Protocol)
Advertisement Interval option in ND, 412
attacks related to, 428–429
Certificate option in ND, 417
CGA options in ND, 414–415
definition of, 944
destination unreachable, 364–372
DNS Search List option in ND, 422–423
Echo Request/Reply messages, 380–383
encapsulation of messages within IP datagrams, 354–355
error messages, 361–363
Experimental Values in ND, 423
extended and multipart messages, 363–364
Handover Key Request/Reply options in ND, 422–423
Home Agent Discovery Request message, 386
Home Agent Information option in ND, 412–413
ICMPv4 messages, 356–357
ICMPv6 messages, 358–360
IND (Inverse Neighbor Discovery), 401–402
introduction to, 353–354
IP Address/Prefix option in ND, 417–418
Link-Layer Address (LLA) option in ND, 418–419
MIPv6 fast handover messages, 388
MLD extension messages, 390–394
Mobile Prefix Solicitation message, 387–388
MRD (Multicast Router Discovery), 394–395
MTU option in ND, 411–412
NAT and, 309
ND options, 407–409
ND support, 395–396
Nonce option in ND, 416–417
NS (Neighbor Solicitation) message, 398–401

ICMP (Internet Control Message Protocol), *continued*
NUD (Neighbor Unreachability Detection), 402–403
Parameter Problem message, 379–380
Prefix Information option in ND, 410–411
processing messages, 360–361
queries/informational messages, 380
Recursive DNS Server option in ND, 420
Redirected Header option in ND, 411
redirection of messages, 372–375
Route Information option in ND, 420
Router Advertisement Flags Extension option in ND, 420–421
router solicitation and advertisement messages in ICMPv4, 383–385
router solicitation and advertisement messages in ICMPv6, 396–398
RSA Signature option in ND, 415–416
SEND (Secure Neighbor Discovery), 403–407
sending broadcast datagrams, 439
Source Link-Layer Address option in ND, 409–410
Source Quench messages, 785
Source/Target Address List options in ND, 413–414
summary and references, 430–434
TCP-related attacks, 641–642
time exceeded message, 375–378
Timestamp option in ND, 416
translating ICMPv4 to ICMPv6, 424–426
translating ICMPv6 to ICMPv4, 426–428
Trust Anchor option in ND, 417
use in layers of TCP/IP suite, 15
ICS (Internet Connection Sharing)
assignment of unicast addresses, 67
definition of, 944
NAT in Windows OS context, 337
ICV (Integrity Check Value), 856, 945
ID (identification payload), in IKE, 945
Id/loc split protocols, 70
Identification field
IP header fields, 185, 203–204
in UDP fragmentation, 489
Identification (ID payload), in IKE, 945
Identification messages, in LCP operation, 134
Identifier/locator separating protocols, 70
Identifiers
in demultiplexing, 11
IP addresses as, 70
in multiplexing, 10
Identity Association (IA)
definition of, 944
in DHCPv6, 255–256
Identity Association Identifier (IAID)
definition of, 944
in DHCP, 246, 255–256
IDN ccTLS (Internationalized ccTLDs), 512
IDNs (internationalized domain names), 512, 945
IEEE (Institute of Electrical and Electronics Engineers)
definition of, 945
Ethernet. *See* Ethernet (802.3)
interface standards, 44
LAN/MAN standards (802), 82–84
link aggregation (802.1AX), 92–93
Logical Link Control (802.2), 84
Multiple Registration Protocol (802.1ak), 111
network security (802.1x), 833–834
quality of service (802.1p), 90
standards of, 24
for VLANs (802.1q), 89–92
wireless (802.11). *See* Wi-Fi (wireless fidelity-802.11)
IESG (Internet Engineering Steering Group), 22–23, 945
IETF (Internet Engineering Task Force)
definition of, 945
for Internet standards, 22–23
ROAD (ROuting and ADdressing) group, 47
securing IPv6 Neighbor Discovery, 292
`ifconfig` command, in UNIX and Linux
setting/finding broadcast addresses, 437
viewing active multicast addresses, 66
viewing format of link-local IPv6 address, 45–46
IGD (Internet Gateway Device), in UPnP
definition of, 945
NAT and, 338–339
IGDDC (Internet Gateway Device Discovery and Control), 337
IGMP (Internet Group Management Protocol)
attacks related to, 469–470
counters and variables, 467–468
definition of, 945
examples, 459–464
lightweight IGMP3, 464–465
MLD as translation of IGMPv3 to IPv6, 390
MRD (Multicast Router Discovery) and, 394
in multicast addressing, 15
overview of, 451–453
processing by group members (Group Member Part), 454–457
processing by multicast routers (Multicast Router Part), 457–459
robustness of, 465–467
snooping, 468–469
summary and references, 470–472

IHL (Internet Header Length), in IPv4
attacks related to, 919
definition of, 945
ICMP Parameter Problem and, 379
IP header and, 183
IIDs (interface identifiers)
as basis for unicast IPv6 addresses, 43–46
definition of, 945
for link-scoped IPv6 addresses, 58
IKE (Internet Key Exchange), in IPsec
algorithm selection and application, 849–850
BTNS (Better-than-Nothing Security), 852
CREATE_CHILD_SA exchange, 852–853
definition of, 945
EAP and, 851–852
example using, 867–876
IKE_AUTH exchange, 850–851
IKE_SA_INIT exchange, 846–847
INFORMATIONAL exchange, 853–854
KE (Key Exchange) and Ni, Nr (Nonce) payloads, 848
message formats, 844–846
MOBIKE (Mobile IKE), 854
N (Notification) and CP (Configuration) payloads, 849
NAT updates and, 866
overview of, 842–843
SA (Security Association) payloads and proposals, 847–848
traffic selectors, 851
IKE_AUTH exchange, 850–851, 871–874
IKE_SA_INIT exchange, 846–847, 867–870
IMAP (Interactive Mail Access Protocol)
definition of, 945
SRV record providing IMAP service, 549
well-known port for, 18
IMAPS (IMAP over SSL/TLS), 18, 945
Implementation architecture
design and, 8
layering, 8–10
multiplexing, demultiplexing, and encapsulation, 10–13
Implementations, TCP/IP suite, 24–25
Implicit sending, in congestion control, 583
IN (Internet class name), 945
Incoming connection queue, TCP servers, 636–640
Incremental zone transfer (IXFR) messages, in DNS, 561, 946
IND (Inverse Neighbor Discovery)
definition of, 945
neighbor discovery in IPv6, 401–402
Source/Target Address List options in ND, 413–414
Independent basic service set (IBSS)
definition of, 944
Wi-Fi, 112
Indication transactions, STUN, 320
Indirect delivery, example of IP forwarding, 212–215
Industrial, Scientific, and Medical (ISM), 124
Information disclosure attacks, ICMP and, 428
Information security. *See also* Security, 806
Information theory, 579
Informational category, RFCs and, 23
INFORMATIONAL exchange, IKE protocol, 853–854, 873, 875
Informational messages, ICMP, 309
Initial sequence number. *See* ISN (initial sequence number)
Initial window (IW) value
definition of, 946
in slow start algorithm, 732
Institute of Electrical and Electronics Engineers (IEEE). *See* IEEE (Institute of Electrical and Electronics Engineers)
Integrated Services Digital Network (ISDN), 946
Integrity Check Value (ICV), 856, 945
Integrity, in CIA triad, 806
Integrity protection, AH (Authentication Header), 856, 858
Inter-Packet Gap (IPG), in Ethernet, 89, 945
Inter-Switch Link (ISL)
definition of, 946
VLAN trunking, 90
Interactive communication, TCP data flow, 692–695
Interactive Connectivity Establishment (ICE), 332–334, 944
Interactive data, in TCP communication, 692
Interactive keystrokes, TCP and, 692–693
Interactive Mail Access Protocol. *See* IMAP (Interactive Mail Access Protocol)
Interface address, IP addresses and, 35
Interface identifiers. *See* IIDs (interface identifiers)
Interface, in IP forwarding, 209
Intermediate system, for protocol suites, 12
Intermediate System to Intermediate System (IS-IS), 946
International Organization for Standardization (ISO), 8–9
International Telecommunication Union. *See* ITU (International Telecommunication Union)
Internationalized ccTLDs (IDN ccTLS), 512
Internationalized domain names (IDNs), 512, 945
Internet
attacks related to Internet architecture, 25–26
internationalization of, 512
overview of, 19–20

Internet, *continued*
TCP/IP suite forming basis of, 2
UDP in, 506–507
WWW compared to, 2–3
Internet Architecture Board (IAB), 22, 944
Internet Assigned Numbers Authority. *See* IANA (Internet Assigned Numbers Authority)
Internet checksum
algorithm for computing Internet-related checksum, 185
mathematics of, 187–188
overview of, 186–187
Internet class name (IN), 945
Internet Connection Sharing. *See* ICS (Internet Connection Sharing)
Internet Control Message Protocol. *See* ICMP (Internet Control Message Protocol)
Internet Corporation for Assigned Names and Numbers. *See* ICANN (Internet Corporation for Assigned Names and Numbers)
Internet Engineering Steering Group (IESG), 22–23, 945
Internet Engineering Task Force. *See* IETF (Internet Engineering Task Force)
Internet Gateway Device Discovery and Control (IGDDC), 337
Internet Gateway Device (IGD)
definition of, 945
NAT and, 338–339
Internet Group Management Protocol. *See* IGMP (Internet Group Management Protocol)
Internet Header Length. *See* IHL (Internet Header Length)
Internet Key Exchange. *See* IKE (Internet Key Exchange)
Internet Protocol Control Protocol (IPCP)
MPPE and, 145
types of NCPs used on PPP links, 141
Internet Registry Information Service (IRIS), 554, 946
Internet Research Task Force (IRTF), 23, 946
Internet Security Association and Key Management Protocol (ISAKMP), 867, 946
Internet Service Providers. *See* ISPs (Internet Service Providers)
Internet Society (ISOC), 23, 946
Internetwork layer. *See* Network (internetwork) layer
Intra-Site Automatic Tunnel Addressing Protocol (ISATAP), 440, 946
Intranets, 20
Inverse Neighbor Discovery. *See* IND (Inverse Neighbor Discovery)
IP address pooling behavior, NAT and, 312
IP Address/Prefix option, in ND, 417–418
IP addresses
Address realms, 299
allocation of, 62
allocation of multicast addresses, 65
allocation of unicast addresses, 62–65
anycast addresses, 62
assigning unicast addresses, 65–66
attacks involving, 70–71
broadcast addresses, 42–43
CIDR, 47–48
classful addressing, 34–36
converting IP multicast addresses to MAC addresses, 442–444
definition of, 14
in DHCP/BOOTP message format, 237–238
expressing, 32–34
foreign IP address restrictions in server design, 502–503
host names and, 19
introduction to, 31–32
IPv4/IPv6 translators, 52–53
IPv4 multicast addresses, 54–57
IPv6 addresses and interface identifiers, 43–46
IPv6 multicast addresses, 57–61
multicast addresses, 53–54
multiple addresses in UDP server design, 501–502
multiple providers/multiple networks/multiple addresses, 68–70
route aggregation, 48–50
single provider/multiple networks/multiple addresses, 67–68
single provider/no network/single address, 66–67
single provider/single network/single address, 67
spanning IP address families in server design, 504
special use addresses, 50–52
spoofing attacks and, 226
structure of, 34
subnet addressing, 36–39
subnet masks, 39–41
summary and references, 71–77
TCP/IP suite and, 19
TCP server restrictions on local, 634–635
unicast and broadcast, 15
in UPD server design, 499–501
VSLM (variable-length subnet masks), 41–42
IP datagrams. *See also* Packets
fragmenting, 184
of ICMP messages within IP datagrams, 354–355
LCP (Link Control Protocol) and, 131–137
PPP and, 130–131

source and destination IP addresses in, 186
in TCP/IP suite, 14
IP forwarding
direct delivery, 210–212
forwarding actions, 209–210
forwarding table, 208–209
indirect delivery, 212–215
overview of, 208
IP fragmentation
example of UPD/IPv4 fragmentation, 488–492
interaction between IP fragmentation and ARP/ND, 496–497
of IP datagrams, 184
IP performing, 148
IPv6 Fragment header, 203–208
overview of, 488
reassembly timeout, 492
UDP-related attacks, 506
IP (Internet Protocol)
address selection by hosts, 222–223
attacks, 226
best-effort delivery, 7
bidirectional tunneling in mobile IP, 216–217
destination address selection algorithm, 224–225
discussion of issues in mobile IP, 220
DS field and ECN, 188–192
examples of forwarding, 210–215
forwarding. *See* IP forwarding
Fragment header, 203–208
header. *See* Header, IP
host models, 220–222
ICMP addressing limits in, 353
Internet checksum, 186–188
introduction to, 181–183
IPv6 extension headers, 194–196
IPv6 options, 196–199
mobile IP. *See* MIP (Mobile IP)
options, 192–194
RO (route optimization) in mobile IP, 217–219
Routing header, 200–203
source address selection algorithm, 223–224
summary and references, 226–231
IP masquerading
`ipchains` command for configuring, 345
NAT in Linux context, 337
IP routers, 20
`ipchains` command, configuring IP masquerading in Linux, 345
`ipconfig` command in Windows
manual management of DHCP information, 250–251
manual management of DHCPv6 information, 265–266
IPCP (Internet Protocol Control Protocol), in PPP
MPPE and, 145
types of NCPs used on PPP links, 141
IPG (Inter-Packet Gap), in Ethernet, 89, 945
IPSec (IP Security)
attacks related to, 918
Authentication Header. *See* AH (Authentication Header)
definition of, 945
Encapsulating Security Payload. *See* ESP (Encapsulating Security Payload)
example using IKE, 867–876
GRE tunnels and, 150
Internet Key Exchange protocol. *See* IKE (Internet Key Exchange)
L2TP/IPSec, 865
layer 3 security with, 840–842
multicast support, 864
NAT traversal, 865–867
`iptables`, 335
IPv4 addresses
allocation of multicast addresses, 65
allocation of unicast addresses, 62–65
configuring with SLAAC, 276
example of subnet addressing, 37
examples of subnet masks, 39
expressing, 32–33
foreign IP address restrictions in server design, 502–503
IPv4-converted addresses, 341
IPv4-embedded IPv6 address, 52
IPv4/IPv6 translators, 52–53
IPv4-translatable addresses, 341
local address restrictions in UDP server design, 500–501
multicast addresses, 54–57
multiple addresses in UDP server design, 501–502
multiple servers per port in UPD server design, 503–504
setting for embedded devices, 178
spanning IP address families in UDP server design, 504
special-use addresses, 50–51
tunneling IPv6 packets over UDP/IPv4 packets, 154
using ARP to map to hardware addresses, 165
IPv4/IPv6 translation
DS-Lite (Dual-Stack Lite), 339–340
IPv4-converted and IPv4 translatable addresses, 341–342
NAT and ALG for, 340–345
overview of, 340–341

IPv4/IPv6 translation, *continued*
stateful translation, 344–345
stateless translation, 342–344
IPv4 protocol
ARP and, 13
Checksum field, 186–188
computing Internet checksum, 186–187
direct delivery with, 167–169
DS and ECN fields, 188–192
encapsulation of ICMP messages in, 354–355
header, 182–183
header fields, 183–186
ICMP messages related to, 356–357
ToS (Type of Service) byte in, 188–189
translation to/from IPv6. *See* IPv4/IPv6 translation
tunneling IPv6 through, 482–487
IPv6 addresses
allocation of multicast addresses, 65
allocation of unicast addresses, 62–65
configuring global addresses with SLAAC, 278
configuring with SLAAC, 276–277
DAD (Duplicate Address Detection), 277–278
examples of subnet masks, 40
expressing, 32–34
interface identifiers and, 43–46
IPv4/IPv6 translators, 52–53
lifecycle of, 252–253
multicast addresses, 57–61
multiple servers per port in UPD server design, 503–504
spanning IP address families in UDP server design, 504
special-use addresses, 51–52
tunneling IPv6 packets over UDP/IPv4 packets, 154
ULAs (Unique Local IPv6 Unicast Addresses), 225
wildcard address, 632
IPv6 protocol
Checksum field, 186–188
DS and ECN fields, 188–192
encapsulation of ICMP messages in, 354–355
extension headers, 194–196
Fragment header, 203
header, 182–183
header fields, 183–186
ICMP messages related to, 358–360
jumbogram support, 481
NAT and, 310–311
neighbor discovery. *See* ND (Neighbor Discovery Protocol)
options, 196–199
Rapid Deployment (6rd), 339, 933
Routing header, 200–203
Traffic Class byte in, 188–189
translating to/from IPv4. *See* IPv4/IPv6 translation
tunneling through IPv4 networks, 482–487
UDP and, 481–482
IPv6 Remote-ID, DCHP relay agents, 268
IPV6CP (IPv6 Control Protocol), in PPP, 141, 945
IRIS (Internet Registry Information Service), 554, 946
IRTF (Internet Research Task Force), 23, 946
IS-IS (Intermediate System to Intermediate System), 946
ISAKMP (Internet Security Association and Key Management Protocol), 867, 946
ISATAP (Intra-Site Automatic Tunnel Addressing Protocol), 440, 946
ISDN (Integrated Services Digital Network), 946
ISL (Inter-Switch Link)
definition of, 946
VLAN trunking, 90
ISM (Industrial, Scientific, and Medical), 124
ISN (initial sequence number)
definition of, 946
SYN segment containing, 597
in TCP connection establishment, 601–602
TCP header and, 589
ISO (International Organization for Standardization), 8–9
ISOC (Internet Society), 23, 946
ISPR (IANA Service Name and Transport Protocol Port Number), 548
ISPs (Internet Service Providers)
ACs (access concentrator), 287
allocation of IP addresses and, 32, 62–63
definition of, 946
MTU (maximum transmission unit), 494
Iterative servers, 21
ITU (International Telecommunication Union)
definition of, 946
standards organizations, 24
X.509 standard. *See* X.509 standard
IW (initial window) value, in TCP
definition of, 946
in slow start algorithm, 732
`iwconfig` command, setting Wi-Fi control frame variables, in UNIX and Linux, 115
IXFR (incremental zone transfer) messages, in DNS, 561–563, 946

J

Jacobson, V., 652–654, 731
Jumbograms
IPv6 options for, 198
IPv6 supporting, 481

K

Kaminsky attacks, DNS-related, 572
Kamoun, F., 48
Karn's algorithm, 655
KDF (key derivation function), 815
KE (Key Exchange)
  definition of, 946
  IKE payloads, 848
Keepalive interval
  changing values of, 796
  definition of, 795
Keepalive probes
  changing values of, 796
  definition of, 795
  overview of, 793
Keepalive time
  changing values of, 796–797
  definition of, 795
  server host crashes and reboots, 799–800
  server host unreachable, 800–802
Keepalive timers, 793
Keepalives, TCP. *See* TCP keepalive
Key derivation function (KDF), 815
Key Exchange (KE)
  definition of, 946
  IKE payloads, 848
Key Hash field, RSA Signature option, 416
Key management, in cryptography, 821
Key signing keys. *See* KSKs (key signing keys)
Keyed-hash message authentication code (HMAC), 818–819, 943
Keys, cryptographic, 810
Kleinrock, Leonard, 1, 48
KSKs (key signing keys)
  definition of, 946
  DNSSEC, 897, 905

L

L2TP/IPSec, 865
L2TP (Layer 2 Tunneling Protocol)
  definition of, 946
  establishing tunneling with, 149
LaBrea tarpit, attacks related to window management, 723
LACP (Link Aggregation Control Protocol), 92–93, 947
LAGs (link aggregation groups), 93, 947
Land attacks, ICMP attacks, 428
LANs (local area networks)
  definition of, 947
  Ethernet standards, 82–84
  virtual. *See* VLANs (virtual LANs)
Large-scale NAT (LSN), IPv6 transition, 315
Last Member Query Interval (LMQI), in IGMP/MLD, 468, 947
Last Member Query Time (LMQT), 466–467, 947
LAST_ACK state
  simultaneous open and close transitions, 625
  TCP state transitions, 618
Latency, connections and, 4
Layer 2 devices, DCHP relay agents, 270–271
Layer 2 Tunneling Protocol (L2TP)
  definition of, 946
  establishing tunneling with, 149
Layering
  implementation and design and, 8–10
  security protocols, 832–833
  TCP/IP suite and, 14
Layering violation, transport layer, 476
LCG (Linear Congruential Generator), 816, 947
LCI (Location Configuration Information), 274, 947
LCI (logical channel identifier), 5
LCN (logical channel number), 5, 947
LCP (Link Control Protocol)
  definition of, 947
  operation of, 133–134
  options, 134–137
  overview of, 131–134
LDAP (Lightweight Directory Access Protocol)
  definition of, 947
  DNS and, 570–571
  SRV record providing LDAP service, 548–549
  well-known port for, 18
LDRAs (lightweight DHCP relay agents), 271, 947
LEAP (Lightweight EAP), 947
Leasequery, DCHP relay agents, 269–270
Leases
  in BOOTP, 235
  duration of DHCP leases, 235–236
Length field
  in Ethernet frame format, 85–86
  UDP header, 475, 481–482
  Payload Length field, in IPv6 header, 184, 198, 204–205
  Total Length field, in IPv4 header, 183–184, 207
Licklider, J.C.R., 2
Lightweight DHCP relay agents (LDRAs), 271, 947
Lightweight Directory Access Protocol. *See* LDAP (Lightweight Directory Access Protocol)
Lightweight EAP (LEAP), 947
Lightweight IGMP3, 464–465
Lightweight MLD (LW-MLD), 394, 948
Lightweight MLDv2, 464–465
Limited slow start, 772
Limited transmit approach, to congestion control in TCP, 742

Linear Congruential Generator (LCG), 816, 947
Link Address field, in DHCPv6 message format, 254
Link aggregation (802.1AX), 92–93
Link Aggregation Control Protocol (LACP), 92–93, 947
Link aggregation groups (LAGs), 93, 947
Link Control Protocol. *See* LCP (Link Control Protocol)
Link discriminator, LCP options, 139
Link layer
  address resolution and, 165–166
  attacks on, 154–156
  autonegotiation in Ethernet interface, 95
  bridges and switches, 98–102
  broadcasting ARP frames, 167
  duplex mismatch, 96
  Ethernet (802.3) frames, 84–86
  Ethernet frame sizes, 88–89
  Ethernet supporting broadcasting at, 437
  flow control (802.1X) in Ethernet interface, 98
  full duplex Ethernet, 94–95
  IEEE 802 LAN/MAN standards for, 82–84
  integrity checking on Ethernet frames, 86–88
  introduction to, 79
  link aggregation (802.1AX), 92–93
  MRP (Multiple Registration Protocol), 111
  MTU (maximum transmission unit), 148
  Point-to-Point Protocol. *See* PPP (Point-to-Point Protocol)
  standards, 80–82
  STP (Spanning Tree Protocol). *See* STP (Spanning Tree Protocol)
  summary and references, 156–163
  tunneling, 149–153
  UDLs (unidirectional links), 153–154
  VLANs and QoS tagging, 89–92
  wireless LANs. *See* Wi-Fi (wireless fidelity-802.11)
  WoL (Wake-on LAN), power saving, and magic packets, 96–97
Link-Layer Address. *See* LLA (Link-Layer Address)
Link Local Multicast Name Resolution (LLMNR), 445, 569–570, 947
Link-local scope
  configuring IPv4 addresses with SLAAC, 276
  configuring IPv6 addresses with SLAAC, 276–277
  of IPv6 addresses, 43
  IPv6 multicast addresses, 58–59
  multicast addresses, 53
  viewing format of link-local IPv6 address, 45
Link Quality Reports (LQRs)
  definition of, 947
  in PPP operations, 136
Links, unidirectional, 153–154
Linux OSs
  autotuning TCP receive windows, 715
  `ethtool` program for checking full duplex support, 94
  IP masquerading, 337
  IPSec implementations, 867
  quick acknowledgments, 733
  rate limiting of ICMP messages, 369–370
  RTT estimation, 657–661
  standards and, 24
  vconfig command for manipulating 802.1p/q information, 90–91
LISTEN state, in TCP
  TCP port numbers and, 632–633
  TCP state transitions and, 618
LLA (Link-Layer Address)
  definition of, 947
  Link-Layer Address (LLA) option in ND, 418–419
  Source Link-Layer Address option in ND, 409–410
  Source/Target Address List options in ND, 413–414
LLC (Logical Link Control)
  802.2 standard defining, 84
  definition of, 947
  relationship of link layer frames to data frames, 116
LLMNR (Link Local Multicast Name Resolution), 445, 569–570, 947
LMQI (Last Member Query Interval), in IGMP/MLD, 468, 947
LMQT (Last Member Query Time), 466–467, 947
LNP (Local Network Protection), 310, 947
Local area networks. *See* LANs (local area networks)
Local congestion, in Linux TCP example,
  fast retransmit and, 759–762
  sender pause and, 750–754
  stretch ACKs and recovery from, 754–757
Local IP addresses
  restrictions in UDP server design, 500–501
  TCP server restrictions on local, 634–635
Local net (limited) broadcast, 43
Local Network Protection (LNP), 310, 947
Location Configuration Information (LCI), 274, 947
Location-to-Service Translation (LoST), 274–275
Locators, IP addresses as, 70
Logical channel identifier (LCI), 5
Logical channel number (LCN), 5, 947
Logical Internet addresses, translating to physical hardware addresses, 167
Logical Link Control. *See* LLC (Logical Link Control)
Long options, DHCP/BOOTP options, 239

Long-term credential mechanism, STUN, 325–326
Long-Term Evolution (LTE), 948
Longest matching prefix algorithm, 69, 209–210
Lookup, IP addresses form host names, 19
Loopback
  hairpinning (NAT loopback), 314, 486
  implementing loopback capacity, 145–148
  PPP issues and, 134
LoST (Location-to-Service Translation), 274–275
Low-rate DoD attacks, TCP timeout/retransmission attacks, 686
LQRs (Link Quality Reports), in PPP
  definition of, 947
  in PPP operations, 136
LSN (large-scale NAT), IPv6 transition, 315
LTE (Long-Term Evolution), 948
LW-MLD (Lightweight MLD), 948

M

MAC layer, sublayer of link layer, 84
MAC (Media Access Control)
  addresses in Ethernet frames, 16
  chaddr (Client Hardware Address) field, 244
  DCF collision avoidance/backoff procedure, 121–122
  definition of, 948
  HCF (hybrid coordination function), 122–123
  multicast addresses, 437
  overview of, 120–121
  protocols, 81
  virtual carrier sense and physical carrier sense, 121
MAC (message authentication codes), 818–820, 948
MAC PDU (MPDU), 113, 949
MAC Services Data Unit (MSDU), in 802.11n, 949
Magnification attacks, UDP-related attacks, 506
Mail exchanger (MX) records, in DNS
  definition of, 950
  DNS resource record types, 544–545
Mailboxes, port numbers as, 474
Malware (malicious software)
  attacks related to Internet architecture, 26
  definition of, 805–806
Man-in-the-middle attacks. *See* MITM (man-in-the-middle) attacks
Management frames, Wi-Fi, 113–115
Manchester Phase Encoding (MPE)
  clock recovery in Ethernet frames, 85
  definition of, 949
MANs (metropolitan area networks)
  cable TV and DSL, 79
  definition of, 948
  Ethernet standards, 82–84
Mapping timer, clearing NAT state, 308–309
MAPs (mesh APs), 130
Masks, in IP forwarding, 209
Masquerading attacks
  firewalls and NATs and, 345
  ICMP attacks and, 429
  link layer attacks, 155
  types of threats to network communication, 808
Master session keys (MSK), in EAP
  definition of, 949
  key derivation in EAP, 838
Maximum pad value (MPV), 137, 949
Maximum probing, 774
Maximum received unit (MRU)
  definition of, 949
  length of LCP packet and, 134
Maximum segment life (MSL). *See also* TIME_WAIT state (2MSL)
  SYN segments, 610
  TIME_WAIT state (2MSL) and, 618
Maximum Segment Size. *See* MSS (Maximum Segment Size)
Maximum transmission unit. *See* MTU (maximum transmission unit)
MCS (Modulation and coding scheme)
  definition of, 948
  higher throughput (802.11n) support and, 127
MD (Message Digest Algorithms)
  definition of, 948
  MD-5 (Message Digest Algorithm 5), 817–818
  TLS extensions, 883
mDNS (Multicast DNS), 444–445, 570, 948
Mean deviation, in RTT estimation, 653
Mechanisms
  DNS TXT records and DKIM, 546
  STUN, 325–326
Media Access Control. *See* MAC (Media Access Control)
Media independent handoff (MIH)
  definition of, 948
  mobility and handoff information in DHCP, 275
Member links, in PPP bundles, 137
Mesh (802.11s), Wi-Fi, 130
Mesh APs (MAPs), 130
Mesh deterministic access, 130
Mesh Point (MP), 949
Mesh points (MPs), 130, 949
Mesh STAs (mesh stations), 130
Message authentication codes (MAC), 818–820, 948
Message boundaries, datagrams and, 5–6
Message digests
  Message Digest Algorithms. *See* MD (Message Digest Algorithms)
  overview of, 817–818

Message formats
DHCP and BOOTP, 236–238
DHCPv6, 253–255
DNS, 520–524
ESP, 862
IKE protocol, 844–846
STUN, 320
Message stream modification (MSM)
definition of, 949
types of threats to network communication, 808
Meta types, resource record categories, 528
Metcalfe's Law, 19
Metropolitan area networks. *See* MANs (metropolitan area networks)
MF (More Fragments) field, in UDP fragmentation, 489
Michael, for message integrity checking, 129
Microsoft CHAP (MS-CHAP), 949
Microsoft Point-to-Point Compression Protocol (MPPC), 140, 949
Microsoft Point-to-Point Encryption (MPPE), 145, 150
MIH (media independent handoff)
definition of, 948
mobility and handoff information in DHCP, 275
MIME (Multipurpose Internet Mail Extensions), 948
MIMO (multiple input, multiple output)
definition of, 948
higher throughput (802.11n) support, 126–127
MIP (Mobile IP)
bidirectional tunneling, 216–217
definition of, 948
DHCP and, 233
discussion of issues in, 220
handling IP nodes, 199
ICMP fast handover message in MIPv6, 388
ICMP Home Agent Discovery Request message, 386
ICMP Mobile Prefix Solicitation message, 387–388
Mobile IP with Fast Handovers (FMIP), 942
Mobile IPv6, 70
overview of, 215–216
RO (route optimization), 217–219
Routing header and, 200
Misbehaving receivers, TCP congestion control attacks, 785
MITM (man-in-the-middle) attacks
ICMP attacks and, 428
security protocol-related, 918–919
types of threats to network communication, 808
MLD (Multicast Listener Discovery)
attacks related to, 469–470
counters and variables, 467–468
DAD and, 278
definition of, 948
examples, 459–464
ICMP extension messages, 390–394
lightweight MLDv2, 464–465
overview of, 15, 451–453
processing by group members (Group Member Part), 454–457
processing by multicast routers (Multicast Router Part), 457–459
query/report/done messages in ICMP, 388–390
robustness of, 465–467
snooping, 468–469
summary and references, 470–472
MLPP (Multilevel Precedence and Preemption), 189, 948
MMRP (Multiple MAC Registration Protocol), 111, 948
MNs (mobile nodes)
definition of, 948
in Mobile IP, 216–218
MOBIKE (Mobile IKE), 854, 865, 949
Mobile IP. *See* MIP (Mobile IP)
Mobile nodes (MNs)
definition of, 948
in Mobile IP, 216–218
Mobile Prefix Advertisement messages, ICMP, 410–411
Mobile Prefix Solicitation message, ICMP, 387–388
Mobility header, 216
Mobility messages, in RRP, 218–219
Mobility Services (MoS)
definition of, 949
mobility and handoff information in DHCP, 275
Modifiers, DNS TXT records, 546
MODP (Modulo-P groups), in DH, 821, 949
Modulation and coding scheme (MCS), in 802.11n
definition of, 948
higher throughput (802.11n) support and, 127
Modulo-P groups (MODP), in DH, 821, 949
More Frag field, data frame fragmentation, 117
MoS (Mobility Services)
definition of, 949
mobility and handoff information in DHCP, 275
MP (Multilink PPP)
definition of, 949
overview of, 137–139
MPDU (MAC PDU), 113, 949
MPE (Manchester Phase Encoding)
clock recovery in Ethernet frames, 85
definition of, 949
MPLS (Multi-Protocol Label Switching), 215, 949
MPPC (Microsoft Point-to-Point Compression Protocol), 140, 949

MPPE (Microsoft Point-to-Point Encryption), 145, 150
MPs (mesh points), 130, 949
MPV (maximum pad value), 137, 949
MRD (Multicast Router Discovery)
  definition of, 949
  designing location of multicast routers, 469
  overview of, 394–395
MRP (Multiple Registration Protocol)
  802.1ak, 111
  definition of, 949
MRRU (multilink maximum received reconstructed unit), 138, 949
MRU (maximum received unit)
  definition of, 949
  length of LCP packet and, 134
MS-CHAP (Microsoft CHAP), 949
MSDU (MAC Services Data Unit), in 802.11n, 949
MSK (master session keys)
  definition of, 949
  key derivation in EAP, 838
MSL (maximum seqment life), in TCP. *See also* TIME_WAIT state (2MSL)
  SYN segments, 610
  TIME_WAIT state (2MSL) and, 618
MSM (message stream modification) attack
  definition of, 949
  types of threats to network communication, 808
MSS (Maximum Segment Size), in TCP
  definition of, 950
  SWS (silly windows syndrome) and, 709
  TCP header, 590, 605–607
MSTP (Multiple Spanning Tree Protocol), in bridges, 111
MTU (maximum transmission unit)
  definition of, 950
  for Ethernet, 86, 506
  frame formats and, 79
  ISPs (Internet Service Providers) and, 494
  link layer and, 148
  neighbor discovery options in IPv6, 411–412
  PPPoE (PPP over Ethernet) and, 614
  preventing fragmentation of TCP datagrams, 612–613
  PTB messages in ICMPv6, 370–371
Multi-access networks, 9
Multi-Protocol Label Switching (MPLS), 215, 949
Multi6 architecture, 70
Multicast addresses
  allocation of, 65
  IPv4 multicast addresses, 54–57
  IPv6 multicast addresses, 57–61
  NAT and, 310
  overview of, 53–54
Multicast DNS (mDNS), 444–445, 570, 948
Multicast groups
  overview of, 15
  RP (rendezvous point), 60
Multicast Listener Discovery. *See* MLD (Multicast Listener Discovery)
Multicast Listener Query, Report, and Done messages, ICMP messages, 388–390
Multicast Router Discovery. *See* MRD (Multicast Router Discovery)
Multicast routers
  designing location of, 469
  IGMP/MLD processing by, 457–459
  overview of, 452–454
  querier election, 466
  query message options, 459
Multicast state, 441
Multicasting
  converting IP multicast addresses to MAC addresses, 442–444
  example of, 444–446
  host address filtering, 449–451
  introduction to, 435–436
  IPSec supporting, 864–865
  overview of, 441–442
  receiving multicast datagrams, 447–449
  sending multicast datagrams, 446–447
Multihomed systems
  IPv6 and, 70
  overview of, 12–13
  unicast addresses and, 67–68
Multilevel Precedence and Preemption (MLPP), 189, 948
Multilink maximum received reconstructed unit (MRRU), 138, 949
Multilink PPP (MP), 137–139, 949
Multiple input, multiple output (MIMO)
  definition of, 948
  higher throughput (802.11n) support, 126–127
Multiple MAC Registration Protocol (MMRP), 948
Multiple Registration Protocol (MRP)
  802.1ak, 111
  definition of, 949
Multiple Spanning Tree Protocol (MSTP), 111
Multiple VLAN Registration Protocol (MVRP), 111, 950
Multiplexing
  implementation and design and, 10–13
  important concepts in development of network architecture, 4
  TCP/IP suite and, 16–17
Multipurpose Internet Mail Extensions (MIME), 948
MVRP (Multiple VLAN Registration Protocol), 111, 950

MX (mail exchanger) records, in DNS
definition of, 950
DNS resource record types, 544–545

N

N (Notification), IKE payloads, 849, 873
NA (Neighbor Advertisement)
ICMPv6 messages, 277
IND (Inverse Neighbor Discovery), 401
main components of ND, 396
neighbor discovery in IPv6, 398–401
NAC (Network Access Control), 833–837, 950
NACK (Negative ACKs), 111, 950
Nagle algorithm, in TCP
Delayed ACK interaction with, 699
disabling, 699–700
overview of, 696–698
reducing number of packets across WANs, 692
Nagle, John, 696
Name authority pointer records. *See* NAPTR (name authority pointer) records
Name resolution. *See also* DNS (Domain Name System)
of host names into IP addresses, 511
process of, 518–519
Name server (NS) records
definition of, 951
DNS resource record types, 529–530
Name servers
caching, 517–518
DNS, 516–517
Name Service Caching Daemon (NSCD), 518, 951
Name space, DNS, 512–514
Naming syntax, DNS, 514–516
NAP (Network Access Protection)
definition of, 950
DHCP leases and, 246
NAPT (Network Address Port Translation), 305–306, 950
NAPTR (name authority pointer) records
definition of, 950
DNS resource record types, 549–551
NTN (non-terminal NAPTR), 551–552
S-NAPTR and U-NAPTR, 554–555
URI/URN resolution and, 553–554
NAR (New Access Router), 417, 950
NAT (network address translation)
address and port translation behavior, 311–313
address pools, 312
attacks related to, 345–346
configuring, 334
definition of, 950
direct interaction with, 338–339
DS-Lite (Dual-Stack Lite) and, 339–340
editors, 315
filtering behavior, 312–313
firewall rules, 335–336
hairpinning (NAT loopback), 314, 486
ICE and, 332–334
ICMP and, 309
IPSec NAT traversal, 865–867
IPv4/IPv6 translation, 340–345
IPv6 and, 310–311
mapping, 307
multicast and, 310
NAPT (Network Address Port Translation), 305–306
other transport protocols and, 309
overview of, 303–305
pinholes and hole punching, 317
port forwarding and port mapping and, 314
private addresses and, 51
rules, 337–338
session, 307–308
SPNAT (service provider NAT), 315–316
STUN (Session Traversal Utilities for NAT), 319–326
summary and references, 346–352
TCP and, 306–308
Teredo tunneling and, 485–486
translating TCP and UDP addresses and port numbers, 605
traversal, 316
tunneled packets and, 310
TURN (Traversal Using Relays around NAT), 326–332
UDP and, 308–309
UNSAF (unilateral self-address fixing), 317–319
NAT-PMP (NAT Port Mapping Protocol)
definition of, 950
direct interaction with NAT and firewalls, 338–339
NAT Port Mapping Protocol (NAT-PMP)
definition of, 950
direct interaction with NAT and firewalls, 338–339
NAT-PT (NAT with Port Translation), 950
NAT Traversal
ICE (Interactive Connectivity Establishment), 332–334
overview of, 316
pinholes and hole punching, 317
STUN (Session Traversal Utilities for NAT), 319–326
TURN (Traversal Using Relays around NAT), 326–332
UNSAF (unilateral self-address fixing), 317–319
NAT with Port Translation (NAT-PT), 950

NAT64, IPv6/IPv4 translation, 344, 950
NAV (Network Allocation Vector), 121, 950
NBMA (non-broadcast multiple access)
  ICMP Redirect message used with, 375
  ND and, 396
  overview of, 167
NCoA (New Care-of Address), 419, 950
NCPs (Network Control Protocols), 131, 141–142, 950
ND (Neighbor Discovery Protocol)
  Advertisement Interval option, 412
  Certificate option, 417
  CGA options, 414–415
  definition of, 951
  DNS Search List option, 422–423
  Experimental Values in, 423
  Handover Key Request/Reply options, 422–423
  Home Agent Information option, 412–413
  IND (Inverse Neighbor Discovery), 401–402
  interaction between IP fragmentation and ARP/ND, 496–497
  IP Address/Prefix option, 417–418
  Link-Layer Address (LLA) option, 418–419
  MTU option in, 411–412
  Nonce option, 416–417
  NS (Neighbor Solicitation) message, 398–401
  NUD (Neighbor Unreachability Detection), 402–403
  options, 407–409
  overview of, 395–396
  Prefix Information option, 410–411
  Recursive DNS Server option, 420
  Redirected Header option, 411
  Route Information option, 420
  Router Advertisement Flags Extension option, 420–421
  router solicitation and advertisement messages in ICMPv6, 396–398
  RSA Signature option, 415–416
  SEND (Secure Neighbor Discovery), 403–407
  Source Link-Layer Address option, 409–410
  Source/Target Address List options, 413–414
  Teredo tunneling and, 486–487
  Timestamp option, 416
  Trust Anchor option, 417
NDP. *See* ND (Neighbor Discovery Protocol)
Negative ACKs (NACK), 133, 950
Negative caching, DNS servers and, 517
Neighbor Advertisement. *See* NA (Neighbor Advertisement)
Neighbor Discovery Protocol. *See* ND (Neighbor Discovery Protocol)
Neighbor Solicitation messages, ICMPv6, 277
Neighbor Unreachability Detection (NUD)
  definition of, 951
  neighbor discovery in IPv6, 402–403
NEMO (Network Mobility), 216, 951
Net number, in IP addresses, 35
NetBoot service, from Apple, 246
`netsh` command, viewing in-use multicast groups in Windows OSs, 447–448
`netstat` command
  displaying IP group membership, 446–447
  restricting local IP addresses, 634
  viewing active multicast addresses, 66
  viewing forwarding table with, 446–447
  viewing IPv4 UDP servers, 500–501
Network Access Control (NAC), 833–837, 950
Network Address Port Translation (NAPT), 305–306, 950
Network address translation. *See* NAT (network address translation)
Network Allocation Vector (NAV), in Wi-Fi, 121, 950
Network architecture
  APIs, 22
  ARM (ARPANET Reference Model), 1–2, 13–16
  attacks related to Internet architecture, 25–26
  client/server design pattern, 20–21
  end-to-end argument, 6
  error control and flow control, 7–8
  fate sharing, 6–7
  implementation and design, 8
  implementations and distributions, 24–25
  Internet, intranets, extranets, 19–20
  layering, 8–10
  multiplexing, demultiplexing, and encapsulation, 10–13, 16–17
  names, addresses, and DNS, 19
  packets, connections, and datagrams, 3–6
  peer-to-peer design pattern, 21–22
  port numbers, 17–19
  principles of, 2–3
  standardization of, 22–24
  summary and references, 26–30
  TCP/IP suite, 13
Network byte order, TCP/IP headers, 183
Network communication, threats to, 807–809
Network Control Protocols (NCPs), 131, 141–142, 950
Network File System (NFS), 478
Network interface cards. *See* NICs (network interface cards)
Network (internetwork) layer
  address resolution and, 165–166
  of OSI model, 9
  passing datagram to transport layer, 17
  TCP and UDP in, 585

Network Mobility (NEMO), 216, 951
Network Time Protocol. *See* NTP (Network Time Protocol)
New Access Router (NAR), 417, 950
New Care-of Address (NCoA), 419, 950
NewReno algorithm, 739–740
Next Header field
  ICMP Parameter Problem and, 379–380
  IPv6, 194–195
  Mobile IP, 216
  UDP header, 476
  UDP-Lite, 487
Next-hop, in IP forwarding, 209
Next Server IP Address (Siaddr) field, DHCP/BOOTP message format, 238, 246
NextSECure resource record. *See* NSEC (NextSECure) resource record
NFS (Network File System), 478
NICs (network interface cards)
  definition of, 951
  host address filtering, 449–451
  overview of, 92
No Route to Destination, ICMPv6 messages, 365
Node-local scope
  of IPv6 addresses, 43
  multicast addresses, 53
Node Requirements RFCs, 23
Nominees, in ICE, 333
Non-broadcast multiple access. *See* NBMA (non-broadcast multiple access)
Non-HT mode, 802.11n operating modes, 128
Non-terminal NAPTR (NTN), 551, 951
Nonce (number used once)
  cryptographic nonces and, 816
  definition of, 951
  IKE payloads, 848
  in ND, 416–417, 486–487
Nonportable addresses, allocation of IP addresses and, 62
Nonrepudiation, basic principles of security, 806–807
NOP (No Operation), TCP header options, 605
Notification (N), IKE payloads, 849, 873
NPTv6, 310, 951
NRO (Number Resource Organization)
  allocation of IP addresses and, 62–63
  definition of, 951
NS (name server) records
  definition of, 951
  DNS resource record types, 529–530
NS (Neighbor Solicitation)
  definition of, 951
  IND (Inverse Neighbor Discovery), 401–402
  main components of ND, 396
  neighbor discovery in IPv6, 398–401
NSCD (Name Service Caching Daemon), 518, 951
NSEC (NextSECure) resource record
  canonical ordering of, 902
  definition of, 951
  DNSSEC, 898–901
  signed zones and zone cuts, 903
NSEC Parameters (NSEC3PARAM), in DNSSEC, 898, 951
NSEC3PARAM (NSEC Parameters), in DNSSEC, 900, 951
NTN (non-terminal NAPTR), 551, 951
NTP (Network Time Protocol)
  definition of, 951
  multicast group in, 54
  variable-scope IPv6 multicast addresses, 58
NUD (Neighbor Unreachability Detection)
  definition of, 951
  neighbor discovery in IPv6, 402–403
Nuke class, bombs attacks on ICMP, 428
NULL encryption algorithm, 819
Number Resource Organization (NRO)
  allocation of IP addresses and, 62–63
  definition of, 951
Number used once. *See* Nonce (number used once)

O

Object Identifier (OID), 951
OCSP (Online Certification Status Protocol)
  definition of, 951
  validating and revoking certificates, 829–831
OFDM (Orthogonal frequency division multiplexing)
  definition of, 951
  MIMO and, 127
OID (Object Identifier), 951
OLSR (Optimized Link State Routing)
  definition of, 951
  Wi-Fi mesh and, 130
Online Certification Status Protocol (OCSP)
  definition of, 951
  validating and revoking certificates, 829–831
OOB (Out of Band), 719, 952
Op field, DHCP/BOOTP message format, 236
Open DNS servers, 567
Open Systems Interconnection (OSI)
  definition of, 952
  layering and, 8
Open systems, TCP/IP suite as, 2
OpenPGP, 822, 883
OPT (option) pseudo records, DNS resource record types, 547–548

Optimal window size, 730
Optimistic ACKing, TCP congestion control attacks, 785–786
Optimistic DAD, 253
Optimistic state, IPv6 addresses, 253
Optimized Link State Routing (OLSR)
  definition of, 951
  Wi-Fi mesh and, 130
Option overloading, DHCP/BOOTP, 239
Option Request Option (ORO), in DHCP, 952
Orderly release, aborting TCP connections, 627
Organizationally Unique Identifier. *See* OUI (Organizationally Unique Identifier)
ORO (Option Request Option), in DHCP, 952
Orthogonal frequency division multiplexing (OFDM)
  definition of, 951
  MIMO and, 127
OSI (Open Systems Interconnection)
  definition of, 952
  layering and, 8
OUI (Organizationally Unique Identifier)
  definition of, 952
  formats of IPv6 addresses, 44–45
  IP multicasting and, 442–444
Out of Band (OOB), 719, 952
Overlay networks
  p2p and, 22
  tunneling allowing formation of, 149

P

P2P (Peer-to-Peer)
  definition of, 952
  design pattern, 21–22
  discovery problem in, 22
PA (provider-aggretable) addresses
  allocation of IP addresses and, 62
  definition of, 952
PAA (PANA Authentication Agent), 839, 952
PaC (PANA client), 839, 952
Packet duplication, TCP timeout/retransmission, 684–685
Packet-filtering firewalls
  overview of, 300–301
  rules, 335–336
Packet-filtering, NAT functions, 305
Packet reordering, TCP timeout/retransmission, 682–684
Packet size threshold, in Wi-Fi control frames, 115
Packet sniffing, 26, 156
Packet-switching, 4
Packet-switching networks, 1
Packet Too Big. *See* PTB (Packet Too Big)
Packetization, TCP reliability and, 586
Packets
  AQM (active queue management), 782–783
  conservation of, 731
  IEEE, 84
  in protocol architecture, 3–6
  retransmission and, 580–581
  sliding window protocol, 582
  in TCP/IP suite, 14
  windows of packets, 581–582
PACP (port access control protocol), 835
PAD (peer authorization database)
  definition of, 952
  in IPSec, 841
PAD (PPPoE Activity Discovery) messages, 288
Padded payload, in Ethernet frame format, 86
Padding
  block size in PPP, 136–137
  IPv6 options for, 197–198
Padding oracle, security protocol-related attacks and, 919
PANA Authentication Agent (PAA), 839, 952
PANA client (PaC), 839, 952
PANA (Protocol for Carrying Authentication for Network Access), 839–840, 952
PAP (Password Authentication Protocol)
  definition of, 952
  for PPP authentication, 140
Parameter Problem messages
  ICMP, 379–380
  translating ICMPv4 to ICMPv6, 426
  translating ICMPv6 to ICMPv4, 426–427
Partial ACKs, in TCP, 739
Passive attacks, types of threats to network communication, 807–809
Passive closer, FIN segments and, 597
Passive open, in TCP connections, 597
Passive opener (server)
  RST segments, 631
  in TCP connections, 599
Password Authentication Protocol (PAP)
  definition of, 952
  for PPP authentication, 140
Path MTU, 148
Path MTU Discovery. *See* PMTUD (Path MTU Discovery)
Path MTU (PMTU), 953
Pattern-match-criteria, in ACL rules, 335
PAWS (Protection against Wrapped Sequence Numbers)
  definition of, 952
  TCP header, 610–611

Payloads
of frames, 16, 86
IKE protocol, 847–849
IP header, 184–185, 203–206
jumbogram options, 198
UDP header, 475, 482
PCF (point coordination function), in Wi-Fi
definition of, 952
options for controlling sharing of wireless medium, 120
PCO (phased coexistence operation)
definition of, 952
higher throughput (802.11n) support and, 128
PCoA (Previous Care-of Address), 419, 952
PCP (Port Control Protocol)
definition of, 952
direct interaction with NAT and firewalls, 339
PD (prefix delegation), 266–267
PDUs (protocol data units)
definition of, 952
encapsulation and, 10
link layer. *See* Frames, Ethernet
PEAP (Protected EAP), 838, 952
Peer Address field, in DHCPv6 message format, 254
Peer authorization database (PAD)
definition of, 952
in IPSec, 841
Peer-to-Peer (P2P)
definition of, 952
design pattern, 21–22
discovery problem in, 22
PEN (Private Enterprise Number), 257, 952
Per-association (per-connection) basis, for NAT connections, 303–304
Per-flow state, LCI and LCN and, 5
Per-hop behavior (PHB)
definition of, 953
forwarding and, 189
Perfect Forward Secrecy (PFS), 815, 953
PFC (Protocol Field Compression)
definition of, 953
PPP and, 133
PFS (Perfect Forward Secrecy), 815, 953
PGP (Pretty Good Privacy), 822
Phased coexistence operation (PCO)
definition of, 952
higher throughput (802.11n) support and, 128
PHB (per-hop behavior)
definition of, 953
forwarding and, 189
Phishing attacks, 806
PHY (physical) layer, 953
Physical addresses, translating logical addresses to, 167
Physical carrier sense, 121
Physical layer
802.11 standard describing, 123–124
channels and frequencies, 124–126
of OSI model, 9
PI (provider-independent) addresses
allocation of IP addresses and, 63
definition of, 953
Piggybacking, Delayed ACKs and, 692, 695
PIM (Protocol Independent Multicast), 953
ping. *See* Echo Request/Reply (ping) messages
Ping of death attacks
ICMP attacks, 428
UDP attacks, 506–507
Pinholes, in NAT traversal, 317
PIO (Prefix Information option)
definition of, 953
neighbor discovery options in IPv6, 410–411
PKCs (public key certificates). *See also* Certificates, 831, 953
PKCS (Public Key Cryptography Standards), 953
PKI (Public Key Infrastructure)
CGAs not requiring, 404
definition of, 953
overview of, 822
Plain old telephone service (POTS)
definition of, 953
DSL and, 287
Plaintext
encrypting cleartext message, 810
TLS, 878
PLCP (Physical Layer Convergence Procedure), 113, 953
PLPMTUD (Packetization Layer Path MTU Discovery), 612–613
Plug and Play, 337–339
PMTU (Path MTU), 148, 953
PMTUD (Path MTU Discovery)
definition of, 953
example of use with TCP, 613–616
example using UDP, 493–496
link layer and, 148
PTB messages in ICMPv6, 370–371
TCP connections and, 612–613
TCP-related attacks, 641–642
using UDP for, 493
PNAC (Port-Based Network Access Control), 833, 953
PoE (power-over-Ethernet)
definition of, 953
higher throughput (802.11n) support and, 128

Point coordination function (PCF), in Wi-Fi
definition of, 952
options for controlling sharing of wireless medium, 120
Point-to-Point Protocol. *See* PPP (Point-to-Point Protocol)
Point-to-Point Tunneling Protocol. *See* PPTP (Point-to-Point Tunneling Protocol)
Pointer (PTR) records, DNS resource record types, 536–541
POP3, SRV record providing POP3 service, 549
Port access control protocol (PACP), 835
Port-Based Network Access Control (PNAC), 833, 953
Port Control Protocol (PCP)
definition of, 952
direct interaction with NAT and firewalls, 339
Port forwarding, in NAT, 314
Port mapping, in NAT, 314
Port numbers
binding options available to TCP server, 635
as mailboxes, 474
TCP/IP suite and, 17–19
TCP servers, 632–634
in UPD server design, 499–500
Port overloading, in NAT, 313
Port parity, in NAT, 313
Port preservation, NAT and TCP and, 307
Port-Preserving Symmetric NAT (PP), 486
Port states, STP, 104–105
Port translation behavior, in NAT, 311–313
Port unreachable message, ICMP, 365–370
Ports
multiple UDP servers per port, 503–504
requesting connection to nonexistent TCP port, 626
in TCP header, 588
POTS (plain old telephone service)
definition of, 953
DSL and, 287
Pouzin, Louis, 1
Power-over-Ethernet (PoE)
definition of, 953
higher throughput (802.11n) support and, 128
Power save mode (PSM), 119–120, 954
Power Save Multi-Poll (PSMP), 120, 954
Power saving, Ethernet (802.3) and, 96–97
PP (Port-Preserving Symmetric NAT), 486
PPP over Ethernet. *See* PPPoE (PPP over Ethernet)
PPP (Point-to-Point Protocol)
authentication, 140–141
CCP (Compression Control Protocol) and, 139–140
definition of, 953
example of, 143–145
header compression, 142–143
MP (Multilink PPP), 137–139
NCPs (Network Control Protocols), 141–142
overview of, 130–131
PPPMux, 137
PPPMuxCP (PPP Mux Control Protocol), 137
PPPoE Activity Discovery (PAD) messages, 288
PPPoE (PPP over Ethernet)
definition of, 953
Discovery and PPP Session phases, 288
DSL and, 287
example of use of, 289–291
message format, 288–289
MTU (maximum transmission unit) and, 614
overview of, 286
PPTP (Point-to-Point Tunneling Protocol)
definition of, 953
establishing sessions, 151–153
establishing tunneling with, 149–153
NAT editors and, 315
Preamble, of Ethernet frame, 84
Preferred lifetime
IA (Identity Association) and, 255
IPv6 addresses, 252
Prefix delegation (PD), 266–267
Prefixes
in IPv4 and IPv6 address management, 47–48
route aggregation and, 50
subnet mask formats, 39
Presentation layer, of OSI model, 10
Preshared keys. *See* PSKs (preshared keys)
Pretty Good Privacy (PGP), 822
Previous Care-of Address (PCoA), 419, 952
PRFs (pseudorandom functions), 816, 954
PRGs (pseudorandom generators), 815–816, 954
Primary DNS servers, 517
Priority fields, QoS 802.1p, 90
Private Enterprise Number (PEN), 257, 952
PRNGs (pseudorandom numbers)
definition of, 954
overview of, 815–816
Probing, NAT session, 308
Promiscuous ARP, 175
Promiscuous mode, Ethernet interfaces, 155
Protected EAP (PEAP), 838, 952
Protection against Wrapped Sequence Numbers (PAWS)
definition of, 952
TCP header, 610–611
Protocol data units. *See* PDUs (protocol data units)

Protocol field
IP header, 185
in PPP frames, 132–133
UDP header, 476
Protocol Field Compression (PFC)
definition of, 953
PPP and, 133
Protocol for Carrying Authentication for Network Access (PANA), 839–840, 952
Protocol identifiers, in multiplexing, 10
Protocol Independent Multicast (PIM), 953
Protocol multiplexing, 10
Protocol suites
definition of, 1
end and intermediate systems, 12
Protocols, 1
Provider-aggretable (PA) addresses
allocation of IP addresses and, 62
definition of, 952
Provider-independent (PI) addresses
allocation of IP addresses and, 63
definition of, 953
Proxy ARP, 174–175
Proxy firewalls
overview of, 301–303
types of firewalls, 300
Proxy Router Solicitation (RtSolPr), 388
Proxy routers, 388
PrRtAdv (Proxy Router Advertisement), 388
Pseudorandom functions (PRFs), 816, 954
Pseudorandom generators (PRGs), 815–816, 954
Pseudorandom numbers. *See* PRNGs (pseudorandom numbers)
PSH bit
sender pause and local congestion (event 1), 750
in TCP communication, 694–695
PSKs (preshared keys)
definition of, 954
encryption and, 129–130
vulnerability to dictionary attacks, 918
PSM (power save mode), 119–120, 954
PSMP (Power Save Multi-Poll), 120, 954
PTB (Packet Too Big)
definition of, 954
destination unreachable and, 612
ICMP attacks and, 429
ICMPv6 messages, 364, 370–371
TCP-related attacks, 641
PTR (pointer) records, DNS resource record types, 536–541
Public key certificates. *See also* Certificates
DNSKEY resource record and, 896
overview of, 822
Public key cryptography
asymmetric (public) key ciphers, 809–812
Handover Key Request/Reply options, 422–423
RSA (Rivest, Shamir, and Adleman), 812–813
Public Key Infrastructure. *See* PKI (Public Key Infrastructure)
Pure ACK, TCP header, 590

Q

QAM (quadrature amplitude modulation), 128, 954
QAPs (QoS access points), 122
QBSS (QoS BSS), 122, 954
QI (Query Interval), in IGMP and MLD, 468, 954
QoS access points (QAPs), 122
QoS BSS (QBSS), 122, 954
QoS (quality of service)
802.1p standard, 90
APSD and, 120
definition of, 954
HCF (hybrid coordination function) and, 122–123
tagging, 89–92
in VLANs, 86
QoS stations (QSTAs), 122–123, 954
QPSK (quadrature phase shift keying), 128, 954
QQI (Querier's Query Interval), in IGMP/MLD, 954
QQIC (Querier's Query Interval Code)
definition of, 954
in MLD, 390
MLD and, 459
QRI (Query Response Interval), in IGMP/MLD, 468, 954
QRV (querier robustness variable), in IGMP/MLD, 466–467, 954
QS (Quick-Start)
definition of, 954
IPv6 options for, 199
QSTAs (QoS stations), 122–123, 954
Quadrature amplitude modulation (QAM), 128, 954
Quadrature phase shift keying (QPSK), 128, 954
Qualifiers, DNS TXT records, 546
Quality of service. *See* QoS (quality of service)
Querier election, multiple multicast routers and, 466
Querier robustness variable (QRV), 466–467, 954
Querier Robustness Variable (QVR), 390
Querier's Query Interval Code. *See* QQIC (Querier's Query Interval Code)
Querier's Query Interval (QQI), in IGMP/MLD, 954
Queries, DNS, 526
Queries/informational messages, ICMP
Echo Request/Reply messages, 380–383
Home Agent Discovery Request message, 386
MIPv6 fast handover messages, 388
MLD extension messages, 390–394

MLD query/report/done messages, 388–390
Mobile Prefix Solicitation message, 387–388
MRD (Multicast Router Discovery), 394–395
overview of, 380
router solicitation and advertisement messages in ICMPv4, 383–385
Query Interval (QI), in IGMP and MLD, 468, 954
Query message
ICMP, 388–390
IGMP, 457–459
Query/response, in DNS protocol, 518
Query Response Interval (QRI), in IGMP/MLD, 468, 954
Query types, resource record categories, 528
Question (query) and zone section format, 526
Queueing theory, in congestion control, 583
Queues
packets stored in, 4
TCP server incoming connection queue, 636–640
Quick acknowledgments, Linux, 733
Quick-Start (QS)
definition of, 954
IPv6 options for, 199
Quiet time concept, TCP state transitions, 624
QVR (Querier Robustness Variable), 390

R

RA (Router Advertisement)
Advertisement Interval option in ND, 412
definition of, 954
DHCPv6, 260–263
Home Agent Information option in ND, 412–413
ICMP attacks and, 429
ICMP messages, 383–385
ICMPv6 messages, 280–281
link with NA (Neighbor Advertisement), 396
MTU option in ND, 411–412
neighbor discovery in IPv6, 396–398
Prefix Information option, 410–411
Route Information option in ND, 420
Router Advertisement Flags Extension option, 420–421
Trust Anchor option, 417
RADIUS (Remote Authentication Dial-In User Service)
for access control, 141
definition of, 955
RAIO (Relay Agent Information Option)
definition of, 955
DHCP, 268
Random Early Detection (RED) gateways
AQM (active queue management) and, 783–785
definition of, 955
Random numbers, in ND, 416–417
Rapid Commit option, DHCP/BOOTP message format, 273–274
Rapid Spanning Tree Protocol (RSTP), 103, 110–111, 956
RARP (reverse ARP), 166, 955
RASs (remote access servers)
control decisions by, 141
definition of, 955
Rate-based flow control, 583
Rate halving, for TCP congestion control, 741–742
Rate-Halving with Bounding Parameters (RHBP)
definition of, 955
for TCP congestion control, 741–742
Rate limiting, of ICMP messages in Linux server, 369–370
RC4 algorithm
definition of, 955
in Wi-Fi security, 129–130
RD (Router Discovery)
definition of, 955
overview of, 383–385
RDATA, in DNS resource record, 527, 955
RDNSS (Recursive DNS Server)
definition of, 955
neighbor discovery in IPv6, 420
Real-Time Protocol (RTP), 313
Reassembly
fragmentation and, 488
of fragmented datagrams, 14, 205
timeout, 492
Rebinding time (T2), for DHCP messages, 240
Receive window structure, sliding window protocol, 701
Reconfigure extension, DHCP, 273
Record layer, in TLS
DTLS (Datagram TLS), 891–892
TLS (Transport Layer Security), 877
Record markers, datagrams and, 5–6
Record protocol, TLS, 878–880
Recovery point, in TCP retransmission, 671
Recur field, GRE tunnels and, 150
Recursive DNS Server (RDNSS)
definition of, 955
neighbor discovery in IPv6, 420
RED (Random Early Detection) gateways
AQM (active queue management) and, 783–785
definition of, 955
Redirect messages, ICMP
ICMP attacks and, 428
overview of, 372–375
Redirected Header option in ND, 411
Reference model, of protocol suite, 1

Referenced connections, TCP reset segments and, 625
`regedit` program in Windows, setting keepalive time with, 797
Regional Internet registries (RIRs)
  allocation of IP addresses and, 62–63
  definition of, 955
Registered ports, 18
Reject route message, ICMPv6, 372
Relative fairness, congestion control schemes and, 769
Relay Agent Information Option (RAIO)
  definition of, 955
  DHCP, 268
Relay agents, DHCP
  layer 2 devices, 270–271
  leasequery and bulk leasequery, 269–270
  overview of, 267
  RAIO (Relay Agent Information Option), 268
  Remote-ID and IPv6 Remote-ID, 268
  Server Identifier Override, 268–269
Relayed transport address, TURN, 326
Reliability, TCP, 586–587
Remote access servers (RASs)
  control decisions by, 141
  definition of, 955
Remote Authentication Dial-In User Service (RADIUS)
  for access control, 141
  definition of, 955
Remote-ID, DCHP relay agents, 268
Remote procedure call (RPC)
  definition of, 955
  SOAP and, 338
Rendezvous point (RP)
  definition of, 955
  for multicast groups, 60
Renegotiation, of cryptographic connection parameters in TLS, 884
Renewal time (T1), for DHCP messages, 240
Reno algorithm, 737–738
Renumbering, allocation of IP addresses and, 63
Repacketization, in TCP
  overview of, 586
  TCP timeout/retransmission, 686–687
Replay attacks
  Nonce option in ND countering, 416–417
  types of threats to network communication, 808
Report message, ICMP, 388–390
REQUEST message, DHCPv6, 264–265, 269
Request/response transactions, in STUN, 320
Request to send. *See* RTS (request to send)
Reserved addresses, IPv6 multicast, 61
Reserved field, in IPv6 Fragment header, 203–204
Reset (RST) segments, TCP
  aborting connections, 627–628
  definition of, 956
  half-open connections, 628–630
  incoming connection queue and, 637
  overview of, 625–626
  requesting connection to nonexistent port, 626
  TWA (TIME-WAIT Assassination), 630–631
Resolver
  accessing DNS with, 511
  DNSSEC example of operation, 903–911
  UDP and, 525–526
  validating security aware resolver, 895
Resource Record Set. *See* RRSet (Resource Record Set)
Resource Record Signature resource record. *See* RRSIG (Resource Record Signature) resource record
Resource records. *See* RRs (resource records)
Resource utilization attacks
  attacks related to IGMP or MLD, 469–470
  UDP-related attacks, 506
Response algorithm, for spurious timeouts and retransmissions, 677
Restart Window (RW), 739, 956
Retransmission
  ARQ and, 580
  of packets, 7
  in TCP. *See* TCP timeout/retransmission
  timeout settings in communication protocols, 584
Retransmission ambiguity problem, 655, 679
Retransmission (RTX), 894, 956
Retransmission timeout. *See* RTO (retransmission timeout)
Retry bit, Frame Control Word, 116
Return Routability Procedure (RRP), in MIP
  definition of, 956
  in RO, 218–219
Reverse ARP (RARP), 166, 955
Reverse DNS queries, 536
Reverse lookup, host names from IP addresses, 19
Reverse Path Forwarding (RPF), 956
RFC (Request for Comments), 23–24, 955
RGMP (Router-port Group Management Protocol), 469, 955
RH (Routing Header)
  definition of, 955
  in IPv6, 200–203
RHBP (Rate-Halving with Bounding Parameters)
  definition of, 955
  for TCP congestion control, 741–742
RIID field, IPv6 multicast addresses, 60–61
Rijmen, Vincent, 811

Rijndael algorithm. *See* AES (Advanced Encryption Standard)
RIP (Router Information Protocol), 955
RIRs (regional Internet registries)
  allocation of IP addresses and, 62–63
  definition of, 955
Rivest, Shamir, and Adleman. *See* RSA (Rivest, Shamir, and Adleman)
`rlogin` (UNIX), precursor to SSH, 692
RO (route optimization), in MIP
  definition of, 955
  in mobile IP, 217–219
ROAD (Running Out of Address Space), 955
Robust Header Compression (ROHC), 143, 955
Robust Security Network access (RSNA)
  definition of, 956
  in Wi-Fi security, 129
Robust Security Network (RSN)
  definition of, 956
  in Wi-Fi security, 129
Robustness/reliability, of IGMP and MLD, 465–467
ROHC (Robust Header Compression), 143, 955
Roles, STP, 104–105
Root bridge, building the spanning tree in STP, 107
Root certificates, 822
Root ports, STP, 104–105
Rouge RAs, ICMP attacks and, 429
Round-robin, DNS, 565–567
Round-trip time. *See* RTT (round-trip time)
Round-trip-time
  estimation, 584
  `traceroute` measuring, 377
Route aggregation, 48–50
Route Information option, in ND, 420
Route optimization (RO), in MIP
  definition of, 955
  in mobile IP, 217–219
Route Type identifiers, 201
Router Advertisement. *See* RA (Router Advertisement)
Router Alert, IPv6 options for, 198
Router Discovery (RD)
  definition of, 955
  overview of, 383–385
Routing Header (RH)
  definition of, 955
  IPv6, 200–203
Router Information Protocol (RIP), 955
Router Requirements RFC, 23
Router Solicitation. *See* RS (Router Solicitation)
Router solicitation and advertisement messages in ICMPv4, ICMP, 383–385
Routers
  congestion of, 727–728
  crashes, 226
  default router, 208
  IGMP/MLD processing by multicast routers, 457–459
  IP routers, 20
  multicast routing, 452–454
  between packet-switching networks, 1
  in small networks, 11–13
Routing Policy Specification Language (RPSL), 65, 956
Routing protocols, 209
Routing tables, 208, 439–441
RP (rendezvous point), in IP Multicast
  definition of, 955
  for multicast groups, 60
RPC (remote procedure call)
  definition of, 955
  SOAP and, 338
  Reverse Path Forwarding (RPF), 956
RPSL (Routing Policy Specification Language), 65, 956
RRP (Return Routability Procedure), in MIP
  definition of, 956
  in RO, 218–219
RRs (resource records), in DNS
  address and name server records, 529–530
  CNAME (canonical name) records, 534–536
  definition of, 956
  in DNS message format, 520–521
  dynamic DNS updates and, 555–557
  ENUM records, 551–552
  example using resource record types, 530–534
  MX (mail exchanger) records, 544–545
  NAPTR (name authority pointer) records, 549–551
  OPT (option) pseudo records, 547–548
  overview of, 527–529
  PTR (pointer) records, 536–541
  S-NAPTR and U-NAPTR, 554–555
  SIP records, 552
  SOA (start of authority) records, 541–544
  SPF (sender policy framework) and TXT records, 545–547
  SRV (service) records, 548–549
  translating DNS from IPv4 to IPv6, 569
  transparency and, 568
  URI/URN resolution, 553–554
RRs (resource records), DNSSEC
  DNSKEY resource record, 896–897
  DS (delegation signer) resource record, 897–898
  NSEC (NextSECure) resource record, 898–901

RRs (resource records), DNSSEC, *continued*
overview of, 896
RRSIG (Resource Record Signature) rescource record, 901–902
RRset (Resource Record Set)
canonical ordering of, 902–903
definition of, 956
dynamic DNS updates and, 555–557
overview of, 527
RRSIG (Resource Record Signature) resource record
definition of, 956
DNSSEC, 901–902
signed zones and zone cuts, 903
RS (Router Solicitation)
definition of, 956
DHCPv6, 260–263
ICMP attacks and, 429
ICMP messages, 383–385
ICMPv6 messages, 280
link with NS (Neighbor Solicitation), 396
neighbor discovery in IPv6, 396–398
RSA (Rivest, Shamir, and Adleman)
in cipher suites, 821
definition of, 956
ECC as alternative to, 815
overview of, 812–813
TLS extensions, 883
RSA Signature option, in ND, 415–416
RSN (Robust Security Network)
definition of, 956
in Wi-Fi security, 129
RSNA (Robust Security Network access)
definition of, 956
in Wi-Fi security, 129
RST. *See* Reset (RST) segments, TCP
RSTP (Rapid Spanning Tree Protocol), 103, 110–111, 956
RTO (retransmission timeout), in TCP
classic method, 651–652
clock granularity and RTO bounds, 654
definition of, 956
initial values, 654
introduction to, 647
Linux RTT estimation, 657–661
retransmission ambiguity and Karn's algorithm, 655
robustness of RTTM to loss and reordering, 662–664
RTT estimation behaviors, 661–662
RTTM (RTT Measurement) with Timestamps option, 656–657
setting, 651
slow start algorithm and, 732
spurious. *See* Spurious timeouts and retransmissions, in TCP
standard method, 652–654
TCP connections and, 611
RTP (Real-Time Protocol), 313
RTS (request to send)
carrier sense and, 121
definition of, 956
Wi-Fi control frames, 115
RtSolPr (Proxy Router Solicitation), 388
RTT (round-trip time)
classic method of RTT estimation, 651–652
clock granularity and RTO bounds, 654
definition of, 956
estimation behaviors, 661–662
HSTCP (HighSpeed TCP) and, 773
initial values in RTO, 654
Linux estimation of, 657–661
Nagle algorithm and, 696–697
retransmission timeout settings and, 584–585
RTO based on, 648, 651
standard method of estimating, 652–654
"stop and wait' protocol and, 581
STUN messages, 320
TCP Timestamp option and, 610
RTTM (RTT Measurement)
robustness to loss and reordering, 662–664
with Timestamps option, 656–657, 956
RTTVAR (RTT Variance), in TCP, 685–686, 956
RTX (Retransmission), 894, 956
RW (Restart Window), in TCP, 739, 956

S

S-NAPTR (straightforward NAPTR)
definition of, 958
DNS resource record types, 554
SACK (selective acknowledgement)
definition of, 957
DSACK (duplicate SACK) extension, 677–679
example of retransmission with, 673–676
fast retransmit and SACK recovery, 757–759
receiver behavior, 672
retransmission with, 647, 671–672
sender behavior, 673
for TCP congestion control, 740–741
in TCP header, 589, 607
SAD (security association database)
definition of, 957
in IPSec, 841–842
SAE (Simultaneous Authentication of Equals)
definition of, 957
Wi-Fi mesh and, 130
Salt, in cryptography, 816

SAP (Session Announcement Protocol)
definition of, 957
for multicast sessions, 55
SAs (Security Associations), in IPsec
CREATE_CHILD_SA exchange, 852–853
definition of, 956
GSAs (group security associations), 864
in IPSec, 841
payloads and proposals, 847–848
proposed algorithms, 867–869
Scalability, of DNS, 516
Scope
ICMPv6 error (Beyond Scope of Source Address), 371
of IPv6 addresses, 43
IPv6 multicast addresses, 57–58
of multicast addresses, 53
SCSV (Signaling Cipher Suite Value), 884, 957
SCTP (Stream Control Transmission Protocol)
definition of, 957
NAT and, 309
transport protocols in TCP/IP suite, 16
SCVP (Server-Based Certificate Validation Protocol)
certificate validation and, 831
definition of, 957
SDID (Signing Domain Identifier), 916, 957
SDLC (Synchronous Data Link Control)
based on HDLC, 131
definition of, 957
SDOs (standards-defining organizations), 23, 957
SDP (Session Description Protocol)
definition of, 957
ICE and, 332–333
IP multicast and, 55
Secondary DNS servers, 517
Secret Key Transaction Authentication for DNS (TSIG), 911–914
Secs field, DHCP/BOOTP message format, 236
Secure Entry Point (SEP) bit, DNSSEC, 896, 905, 957
Secure Hash Algorithm 1. *See* SHA 1 (Secure Hash Algorithm 1)
Secure hash function, 404
Secure Neighbor Discovery. *See* SEND (Secure Neighbor Discovery)
Secure Real-Time Protocol (SRTP), 883, 958
Secure Remote Password (SRP), 883, 958
Secure Shell. *See* SSH (Secure Shell)
Secure Sockets Layer (SSL). *See also* TLS (Transport Layer Security), 876–877, 958
Security
ACs (attribute certificates), 831
basic principles, 806–807
certificates, CAs, and PKIs, 821–822
cryptographic and cipher suites, 819–821
cryptographic nonces and salt, 816
cryptosystems, 809–812
DH (Diffie-Hellman-Merkle Key Agreement), 813–814
ECC (Elliptic Curve Cryptography), 815
hash functions and message digests, 817–818
introduction to, 805–806
message authentication codes, 818–819
PFS (Perfect Forward Secrecy), 815
protocols. *See* Security protocols
pseudorandom numbers, generators, and function families, 815–816
public key certificates, CAs, and X.509, 822–828
RSA (Rivest, Shamir, and Adleman) public key cryptography, 812–813
signcryption, 814–815
summary and references, 919–932
threats to network communication, 807–809
validating and revoking certificates, 828–831
Wi-Fi, 129–130
Security association database (SAD)
definition of, 957
in IPSec, 841–842
Security Associations. *See* SAs (Security Associations)
Security Gateway (SG), in IPsec, 840, 957
Security Parameter Index (SPI), in IPsec
definition of, 958
IKE protocol, 844
Security policy database (SPD), in IPsec
definition of, 958
in IPSec, 841–842
Security protocols
attacks on, 918–919
DKIM (Domain Keys Identified Mail), 915–918
DNS. *See* DNSSEC (DNS Security)
EAP methods, 837–838
ERP (EAP Re-authentication Protocol), 839
Internet Key Exchange. *See* IKE (Internet Key Exchange)
IPSec (IP Security), 840–842
IPSec NAT traversal, 865–867
L2TP/IPSec, 865
layering and, 832–833
NAC (Network Access Control), 833–837
PANA (Protocol for Carrying Authentication for Network Access), 839–840
transport layer. *See* TLS (Transport Layer Security)
Segments Left field, in Routing header, 201–202

Segments, TCP, 586
Selective acknowledgement. *See* SACK (selective acknowledgement)
Selective retransmission, 673
Self-clocking
  ACKs and, 731
  Nagle algorithm and, 696
Self-describing padding, 137
Send maximum segment size. *See* SMSS (send maximum segment size)
SEND (Secure Neighbor Discovery)
  Certificate option, 417
  certification path solicitation/advertisement, 406–407
  CGAs (cryptographically generated addresses), 403–406
  definition of, 957
  Handover Key Request/Reply options, 422–423
  ICMP attacks and, 429
  neighbor discovery options in IPv6, 414–415
  Nonce option in ND, 416–417
  overview of, 403
  RSA Signature option, 415–416
  securing IPv6 Neighbor Discovery, 292
  Timestamp option, 416
  Trust Anchor option, 417
  as variant on ND, 396
Send window structure, sliding window protocol, 701
Sender pause and local congestion (event 1), TCP congestion control, 750–754
Sender policy framework (SPF) records, DNS resource record types, 545–547
SEP (Secure Entry Point) bit, DNSSEC, 896, 905, 957
Sequence Control field, data frame fragmentation, 117
Sequence numbers
  for avoiding duplicate packets, 580
  data frame fragmentation and, 117
  GRE, 150
  PPP, 138
  TCP, 587–588
  TCP-related attacks, 641
  TCP segments, 701
  URG, 590
Sequencing header, in MP, 138
Sequential Port-Symmetric NAT (SP), 486
Server alive messages, TCP keepalive attacks, 802
Server-Based Certificate Validation Protocol (SCVP), 831
Server Identifier Override, DCHP relay agents, 268–269
Server Load Reduction (SLR), 486
ServerHello message, in TLS, 887–889
Servers
  accessing servers behind NAT, 314
  iterative and concurrent, 21
  server host crashes and does not reboot (keepalive scenarios), 796
  server host crashes and reboots (keepalive scenarios), 797–799
  server host unreachable (keepalive scenarios), 799–800
Service model, TCP, 585–586
Service provider NAT. *See* SPNAT (service provider NAT)
Service set identifiers (SSID)
  definition of, 958
  Wi-Fi, 112
Service sets, Wi-Fi, 112
Service (SRV) records, DNS resource record types, 548–549
Session Announcement Protocol (SAP)
  definition of, 957
  for multicast sessions, 55
Session Description Protocol. *See* SDP (Session Description Protocol)
Session Initiation Protocol (SIP)
  definition of, 957
  ENUM records and, 551–552
Session keys, in public key cryptography, 812
Session layer, of OSI model, 10
Session timers, NAT, 307–308
Session Traversal Utilities for NAT. *See* STUN (Session Traversal Utilities for NAT)
SFD (start frame delimiter), in link layer protocols
  clock recovery in Ethernet frames, 84
  definition of, 957
SG (Security Gateway), in IPsec, 840, 957
SHA 1 (Secure Hash Algorithm 1)
  for authentication in DHCP, 268
  definition of, 957
  overview of, 817–818
  TLS extensions, 883
Shannon, Claude, 579
Sharing connection state, 767–768
Shim6 protocol, 70
Short Interframe Space (SIFS), in Wi-Fi
  definition of, 957
  in MAC, 122
Short sequence number, LCP options, 138
Short-term credential mechanism, STUN, 325
Siaddr (Next Server IP Address) field, DHCP/BOOTP message format, 238, 246

SIFS (Short Interframe Space), in Wi-Fi
definition of, 957
in MAC, 122
Signaling Cipher Suite Value (SCSV), 884, 957
Signature verification, CGAs for, 405–406
Signed zones, DNSSEC, 903
Signing Domain Identifier (SDID), 916, 957
SIIT (Stateless IP/ICMP Translation)
definition of, 957
IPv4/IPv6 translation, 342–344
Silly windows syndrome. *See* SWS (silly windows syndrome)
Simple Mail Transfer Protocol. *See* SMTP (Simple Mail Transfer Protocol)
Simple Network Management Protocol (SNMP)
definition of, 958
well-known port for, 18
Simple Object Access Protocol (SOAP)
definition of, 958
GENA using, 338
Simple Service Discovery Protocol. *See* SSDP (Simple Service Discovery Protocol)
Simple Tunneling of UDP through NATs, 319
Simultaneous Authentication of Equals (SAE)
definition of, 957
Wi-Fi mesh and, 130
Simultaneous close, in TCP connections
overview of, 600–601
state transition, 625
Simultaneous open, in TCP connections
defined, 597
overview of, 599–600
state transition, 625
SIP Outbound mechanism, in ICE, 333
SIP records, DNS resource record types, 552
SIP (Session Initiation Protocol)
definition of, 957
ENUM records and, 551–552
SLAAC (stateless address autoconfiguration)
configuring IPv4 link-local addresses, 276
configuring IPv6 link-local addresses, 276–277
deciding whether to use, 244
definition of, 957
example of, 278–283
IPv6 DAD (Duplicate Address Detection), 277–278
IPv6 global addresses, 278
overview of, 276
stateless DHCP and, 283–284
utility/benefit of, 284–285
Sliding window protocol
movement of windows, 702–704
in packet communication, 582
send and receive structures, 701
TCP as, 589
SLLAO (Source Link-Layer Address Option), 409–410, 958
Slot time, in MAC, 122
Slow start algorithm, in TCP
classic algorithms for TCP congestion, 732–734
comparing with congestion avoidance, 736–737
limited, 772
viewing slow start behavior with Wireshark, 749–750
slow start threshold. *See* ssthresh (slow start threshold)
SLR (Server Load Reduction), 486
Smack attacks, ICMP attacks and, 429
Smoothed RTT. *See* SRTT (smoothed RTT)
SMSS (send maximum segment size)
definition of, 958
SWS (silly windows syndrome) and, 709
TCP connections and, 613
SMTP (Simple Mail Transfer Protocol)
definition of, 958
MX (mail exchanger) records and, 544
SRV record providing SMTP service, 549
well-known port for, 18
Smurf attacks, ICMP, 428
SNA (System Network Architecture)
definition of, 958
SDLC in, 131
Sname (Server Name) field, DHCP/BOOTP message format, 238–239
SNAP (Subnetwork Access Protocol), 958
Sniffing, 808
SNMP (Simple Network Management Protocol)
definition of, 958
well-known port for, 18
Snooping
DHCP, 276
IGMP/MLD, 468–469
SNS (Symmetric NAT Support), 486
SOA (start of authority) records
definition of, 958
DNS resource record types, 541–544
SOAP (Simple Object Access Protocol)
definition of, 958
GENA using, 338
`sock` program
creating UDP datagram, 493, 496
generating UDP datagram with, 478–481
restricting local IP addresses, 634
Sockets
popular APIs, 22
in TCP connections, 595–596
TCP ports, 588

SOCKS proxy firewalls, 302–303
Soft state
  ARP cache timeout and, 174
  multicast information and, 441
SOLICIT message, DHCPv6, 260, 269
Solicitation messages, in MRD, 394–395
Solicitation of servers, by clients, 435
Sort lists, DNS, 565–567
Source address selection algorithm, in IP host models, 223–224
Source IP addresses
  address selection by hosts, 222–223
  host processing of IP datagrams, 220–221
  ICMPv6 errors, 371–372
  in IP datagrams, 186
Source Link-Layer Address Option (SLLAO), 409–410, 958
Source Quench messages, TCP congestion control attacks, 785
Source/Target Address List options, in ND, 413–414
SP (Sequential Port-Symmetric NAT), 486
Spam
  DNS resource record for fighting, 545–547
  as malware, 806
Spanning tree, building, 107
Spanning Tree Protocol. *See* STP (Spanning Tree Protocol)
Spatial multiplexing, power save mode, 120
Spatial streams, higher throughput (802.11n) support, 126
SPD (security policy database)
  definition of, 958
  in IPSec, 841–842
Spear phishing attacks, 806
Special-use IP addresses
  for IPv4, 50–51
  for IPv6, 51–52
  local net (limited) broadcast, 43
SPF (sender policy framework) records, DNS resource record types, 545–547
SPI (Security Parameter Index)
  definition of, 958
  IKE protocol, 844
Split DNS, 565–567
SPNAT (service provider NAT)
  definition of, 958
  DS-Lite and, 339
  overview of, 315–316
Spoofing attacks
  ICMP, 429
  Internet architecture, 25
  IP addresses, 70, 226
  TCP, 640–642
  TCP keepalive attacks, 802
Spurious association attacks, 808
Spurious timeouts and retransmissions, in TCP
  congestion control and, 744–745
  DSACK (duplicate SACK) extension, 677–679
  Eifel Detection Algorithm, 679–680
  Eifel Response Algorithm, 680–682
  F-RTO (Forward-RTO Recovery), 680
  overview of, 677
SRC (Source) address, in Ethernet frame format, 85
SRP (Secure Remote Password), 883, 958
SRTP (Secure Real-Time Protocol), 883, 958
SRTT (smoothed RTT)
  classic method of RTT estimation, 651–652
  definition of, 958
  destination metrics and, 685–686
SRV (service) records, DNS resource record types, 548–549
SSDP (Simple Service Discovery Protocol)
  definition of, 958
  direct interaction with NAT and firewalls, 338
  viewing in-use multicast groups in Windows OSs, 448
SSH (Secure Shell)
  for application-managed keepalives, 794
  definition of, 958
  TCP data flow and, 692
  tracing RTT of TCP connection, 697–698
  well-known port for, 18
  well-known ports for, 632
SSID (service set identifiers)
  definition of, 958
  Wi-Fi, 112
SSL (Secure Sockets Layer). *See also* TLS (Transport Layer Security), 876–877, 958
SSM (source-specific multicast)
  attacks related to IGMP or MLD, 470
  definition of, 959
  IGMP and MLD supporting, 452
  MLD supporting, 390
  as multicast service model, 54
ssthresh (slow start threshold), in TCP congestion control
  comparing slow start with congestion avoidance, 736
  Eifel Response Algorithm and, 744–745
  overview of, 733
  standard TCP algorithm and, 738
Standard RTO method, in TCP, 652–654
Standard TCP congestion control algorithm, 728–739
Standards
  IETF (Internet Engineering Task Force) in, 22–23
  link layer, 80–82
  other organizations in, 23–24
  RFC (Request for Comments) and, 23–24

Standards-defining organizations (SDOs), 23, 957
Standards-track category, RFCs and, 23
Start frame delimiter (SFD)
clock recovery in Ethernet frames, 84
definition of, 957
Start of authority (SOA) records, in DNS
definition of, 958
DNS resource record types, 541–544
STAs (stations), in Wi-Fi
definition of, 959
Wi-Fi, 112
State-change records, IGMP/MLD group membership reports, 457
State machine, DHCP, 251–252
State, storing in connection switches, 5
State transitions, TCP
FIN_WAIT_2 state, 625
overview of, 616
quiet time concept, 624
simultaneous open and close transitions, 625
state transition diagrams, 617–618
TIME_WAIT state (2MSL), 618–624
Stateful translation, IPv4/IPv6, 344–345
Stateless address autoconfiguration. *See* SLAAC (stateless address autoconfiguration)
Stateless IP/ICMP Translation (SIIT)
definition of, 957
IPv4/IPv6 translation, 342–344
Stateless mode, DHCPv6, 283–284
Static multiplexing, 4
Station-to-Station (STS) protocol, relation to DH (Diffie-Hellman), 814
Statistical multiplexing, 4
STODER, repacketization and, 686
"Stop and wait" protocol
communication protocols and, 581
Nagle algorithm and, 697
TCP and, 696
STP (Spanning Tree Protocol), in bridges
BPDU structure, 105–107
building the spanning tree, 107
definition of, 959
example of, 107–109
handling topology changes, 107
overview of, 102–104
port states and roles, 104–105
RSTP (Rapid Spanning Tree Protocol), 110–111
Straightforward NAPTR (S-NAPTR)
definition of, 958
DNS resource record types, 554
Stream ciphers, symmetric key ciphers, 811
Stream Control Transmission Protocol. *See* SCTP (Stream Control Transmission Protocol)
Stretch ACKs, 754–757
Strong host model, 220
STS (Station-to-Station) protocol, relation to DH (Diffie-Hellman), 814
STUN (Session Traversal Utilities for NAT)
attributes defined by TURN, 328
binding method, 321
definition of, 959
ICE making use of, 332–334
mechanisms, 325–326
message formats, 320
Teredo servers compared with, 482
Subdomains, in DNS hierarchy, 514
Subnet addressing, 36–39
Subnet broadcast addresses. *See* Broadcast addresses
Subnet fields, in IP addresses, 37
Subnet masks
overview of, 39–41
VLSM (variable-length subnet masks), 41–42
Subnetwork Access Protocol (SNAP), 105, 958
Subnetworks, 37
Switches and bridges
attacks on, 155
layer 2 relay agents and, 270
link layer and, 98–102
in small networks, 11–13
VLAN, 90
SWS (silly windows syndrome)
definition of, 959
example of avoiding, 709–715
overview of, 708
rules for avoiding, 708–709
Symmetric key encryption
cryptographic algorithms, 809–811
KDF (key derivation function) in, 815
Symmetric NAT Support (SNS), 486
SYN bit field, TCP header, 589–590
SYN cookies, in TCP
attacks related to window management and, 723
TCP-related attacks, 640–641
SYN floods, TCP-related attacks, 640
SYN segments, in TCP
combined with ACKs (SACK), 607
definition of, 959
establishing TCP connections and, 602–603
MSL (maximum seqment life), 610
NAT and TCP, 307–308
requesting connection to nonexistent TCP port, 626
in TCP connections, 596–597
TCP header and, 589
WSCALE (Window Scale) option and, 608

Synchronous Data Link Control (SDLC)
based on HDLC, 131
definition of, 957
SYN_RCVD state, in TCP
incoming connection queue and, 636
simultaneous open and close transitions, 625
TCP state transitions, 618
SYN_SENT state, in TCP
simultaneous open and close transitions, 625
TCP state transitions, 618
System configuration options
attacks related to system configuration, 292
autoconfiguration. *See* SLAAC (stateless address autoconfiguration)
DHCP (Dynamic Host Configuration Protocol). *See* DHCP (Dynamic Host Configuration Protocol)
introduction to, 233–234
summary and references, 292–298
System Network Architecture (SNA) from IBM
definition of, 958
SDLC in, 131

T

T1 (Renewal time), for DHCP messages, 240
Tahoe algorithm, TCP congestion control, 737–738
Tarpits, attacks related to window management, 723
Tayor, Bob, 2
`tc` program, for packet scheduling and traffic control subsystem in Linux, 752
TC (Topology Change), in BPDU structure, 106
TCA (Topology Change Acknowledgment), 106
TCN (topology change notification), 107
TCP-AO (Authentication Option)
definition of, 959
TCP header, 612
TCP congestion control
active queue management and ECN, 782–785
attacks related to, 785–786
BIC (Binary Increase Congestion Control), 772–774
buffer bloat, 781–782
classic algorithms for, 730–732
comparing slow start with congestion avoidance, 736–737
congestion avoidance algorithm, 734–736
connection completion and, 766–767
CTCP (Compound TCP) algorithm, 779–781
CUBIC, 775–776
CWV (Congestion Window Validation), 742–744
delay-based, 777
example of handling, 745–749
FACK (forward acknowledgment) and rate halving for, 741–742
fast retransmit and local congestion, 759–762
fast retransmit and SACK recovery, 757–759
FAST TCP algorithm, 778–779
handling spurious RTOs, 744–745
in high-speed environments, 770
HSTCP (HighSpeed TCP), 770–772
introduction to, 727–728
limited transmit approach to, 742
NewReno algorithm for, 739–740
SACK (selective acknowledgement) for, 740–741
sender pause and local congestion (event 1), 750–754
sharing connection state, 767–768
slow start algorithm, 732–734
slow start behavior, 749–750
slowing down TCP senders, 729–730
standard TCP algorithm, 728–739
stretch ACKs and recovery from local congestion, 754–757
summary and references, 786–792
Tahoe, Reno, and Fast Recovery algorithms, 737–738
TCPW (TCP Westwood) algorithm, 779
TFRC (TCP Friendly Rate Control), 768–770
timeouts, retransmissions, and undoing `cwnd` changes, 762–766
Vegas TCP algorithm, 777–778
TCP data flow
attacks related to window management, 723
delayed ACK interaction with Nagle algorithm, 699
delayed ACKs, 695–696
disabling Nagle algorithm, 699–700
example of dynamic window size adjustment and flow control, 705–708
example using urgent mechanism, 720–722
flow control, 700–701
interactive communication, 692–695
introduction to, 691
large buffers and auto-tuning, 715–719
Nagle algorithm, 696–698
sliding window protocol, 701–704
summary and references, 723–725
SWS (silly windows syndrome), 708–715
urgent mechanism, 719–720
zero windows and TCP persistent timer, 704–705
TCP Friendly Rate Control (TFRC), 768–770, 959
TCP/IP suite
ARPANET Reference Model, 13–16
based on ARPANET, 1

implementations and distributions, 24–25
layering, 14
multiplexing, demultiplexing, and encapsulation, 16–17
names, addresses, and DNS, 19
OSI model compared with, 8–9
overview of, 13
port numbers, 17–19
TCP keepalive
attacks related to, 802
description of, 795–797
introduction to, 793–794
server host crashes and does not reboot, 797–799
server host crashes and reboots, 799–800
server host unreachable, 800–802
summary and references, 802–803
TCP segments, 15
TCP servers
incoming connection queue, 636–640
overview of, 631–632
port numbers and, 632–634
restrictions on foreign endpoints, 635–636
restrictions on local IP addresses, 634–635
TCP timeout/retransmission
attacks related to, 687
classic RTO method, 651–652
clock granularity and RTO bounds, 654
congestion control and, 762–766
connection establishment and, 604–605
destination metrics, 685–686
DSACK (duplicate SACK) extension, 677–679
Eifel Detection Algorithm, 679–680
Eifel Response Algorithm, 680–682
example of, 648–651
example of fast retransmit, 668–671
example of retransmission with SACK, 673–676
example of timer-based retransmission, 665–667
F-RTO (Forward-RTO Recovery), 680
fast retransmit, 667–668
introduction to, 647–648
Linux RTT estimation, 657–661
packet duplication, 684–685
packet reordering, 682–684
repacketization, 686–687
retransmission ambiguity and Karn's algorithm, 655
retransmission with SACK, 671–672
robustness of RTTM, 662–664
RTO (retransmission timeout) setting, 651
RTT estimation behaviors, 661–662
RTTM (RTT Measurement) with Timestamps option, 656–657
SACK receiver behavior, 672
SACK sender behavior, 673
spurious timeouts and retransmissions, 677
standard RTO method, 652–654
summary and references, 688–690
timer-based retransmission, 664–665
TCP (Transmission Control Protocol)
ARQ as basis of, 579
connection management. *See* Connections, TCP
definition of, 959
encapsulation in IP datagrams, 587
flow control and, 7–8
header fields, 588–590
introduction to, 584–585
NAT and, 306–308
reliability, 586–587
service model, 585–586
STUN and, 320
summary and references, 591–593
transport protocols in TCP/IP suite, 15
well-known ports for, 525–526
TCP Westwood+ (TCPW+) algorithm, 777
TCP Westwood (TCPW) algorithm, 779
`tcpdump` command
connecting to Web server on host, 171
ICMP destination unreachable messages, 480
not converting IP addresses to machine names, 479
viewing UDP fragmentation, 490–491
TCP_NODELAY option, for disabling Nagle algorithm, 700
`tcptrace`, connection statistics with, 745–747
TCPW+ (TCP Westwood+) algorithm, 777
TCPW (TCP Westwood) algorithm, 779
TDM (time-division multiplexing), 4, 959
Teardrop attacks
ICMP, 428
UDP, 506
`telnet` command
connecting to Web server on host, 171
establishing TCP connections, 602
Telnet program
SSH replacing, 692
well-known port for, 18
Temporal Key Integrity Protocol (TKIP), in Wi-Fi, 129–130
Temporary addresses, in DHCPv6, 255–256
Tentative state, IPv6 addresses, 253
Teredo, tunneling IPv6 over IPv4
IPv4/IPv6 translation, 339
relays and servers, 482
tunneling, 154, 482–487

Termination messages
in LCP operation, 134
in MRD, 394–395
Termination, of TCP connections, 595–598
TFC (Traffic Flow Confidentiality), 858
TFN (Tribe Flood Network), 429
TFRC (TCP Friendly Rate Control), 768–770, 959
TFTP (Trivial File Transfer Protocol)
definition of, 959
ICMP port unreachable messages and, 366–370
Threats, to network communication, 807–809
Three-way handshake, 597, 640
Throughput (802.11n), Wi-Fi, 126–128
Time-division multiplexing (TDM), 4, 959
Time exceeded message, ICMP, 375–378
Time-Remaining messages, in LCP operation, 134
Time sync function. *See* TSF (time sync function)
Time-to-live. *See* TTL (Time-to-live)
TIME-WAIT Assassination (TWA), 630–631, 960
Timed wait (MSL), 618
Timeouts, TCP. *See* TCP timeout/retransmission
Timer-based retransmission
example of, 665–667
introduction to, 647
overview of, 664–665
Timestamp Echo Reply. *See* TSER (Timestamp Echo Reply)
Timestamp Request/Replay message, ICMP attacks and, 429
Timestamp Value. *See* TSV (Timestamp Value)
Timestamps option. *See also* TSOPT (timestamps option)
neighbor discovery in IPv6, 416
TCP header, 608–610
TIME_WAIT state (2MSL), in TCP
overview of, 618–624
TCP state transitions, 624
TWA (TIME-WAIT Assassination), 630–631
Tinygrams, 696
TKIP (Temporal Key Integrity Protocol), in Wi-Fi, 129–130
TLDs (top-level domains)
definition of, 959
in DNS name space, 512
name servers for, 517
TLS (Transport Layer Security)
with datagrams (DTLS), 884–891
definition of, 959
DTLS DoS protection, 894
DTLS handshake protocol, 892–894
DTLS record layer, 891–892
example of use of, 884–891
extensions, 883–884
handshaking protocols, 880–883
HTTP/HTTPS and, 18
overview of, 876–877
Record protocol, 878–880
renegotiation of cryptographic connection parameters, 884
TCP with, 320
TLS 1.2, 877–878
TLV (type-length-value) sets
definition of, 959
IPv6 options held as, 196–197
Top-level domains. *See* TLDs (top-level domains)
Topology Change Acknowledgment (TCA), 106
Topology change notification (TCN), 107
Topology Change (TC), in BPDU structure, 106
Topology changes, STP handling, 107
ToS (Type of Service) byte
definition of, 959
ICMP Parameter Problem and, 379
in IPv4, 183, 188–189
redefined as DSCP/ECN fields, 379
Total Length field
ICMP Parameter Problem and, 379
in IP header, 183–184
TPDU (transport PDU), 10
`traceroute`, for determining routing path, 376–378
Traffic analysis, types of threats to network communication, 808
Traffic Class byte, in IPv6, 183, 188–189
Traffic Flow Confidentiality (TFC), 858
Traffic selectors (TS)
definition of, 960
IKE, 851, 873
Traffic specification (TSPEC), in Wi-Fi QoS
definition of, 960
in HCCA, 123
traffic visibility, ESP (Encapsulating Security Payload), 863–864
Transacation authentication, in DNS, 911–915
Transaction Signatures (TSIG), in DNS
definition of, 960
transaction authentication in DNSSEC, 911–914
Transient session keys (TSKs), 838
Translating
DNS from IPv4 to IPv6, 568–569
ICMPv4 to ICMPv6, 424–426
ICMPv6 to ICMPv4, 426–428
IPv4 to IPv6, 482
UDP/IPv4 and UDP/IPv6 datagrams, 505–506
Translation behavior, NAT, 312

Translation functions, NAT, 305
Translators, TCP connections, 605
Transmission Control Protocol. *See* TCP (Transmission Control Protocol)
Transmit opportunities (TXOPs), in Wi-Fi QoS
  in DCF, 123
  definition of, 960
Transparency, DNS, 567–568
Transport layer
  layering violation, 476
  of OSI model, 9–10
  security. *See* TLS (Transport Layer Security)
  transport protocols in TCP/IP suite, 15–16
  UDP checksum, 475–476
Transport Layer Security. *See* TLS (Transport Layer Security)
Transport PDU (TPDU), 10
Transport protocols, 309
Traversal, NAT, 316
Traversal Using Relays around NAT. *See* TURN (Traversal Using Relays around NAT)
Tribe Flood Network (TFN), 429
Triple-DES (3DES)
  definition of, 959
  standardized for Internet use, 819
  as symmetric encryption algorithm, 811
Trivial File Transfer Protocol (TFTP)
  definition of, 959
  ICMP port unreachable messages and, 366–370
Trunking, VLAN switches and, 90
Trust anchors
  CAs (certification authorities) and, 822
  in ND, 417
  SEND (Secure Neighbor Discovery), 403
TS (traffic selectors), in IPsec
  definition of, 960
  IKE, 851, 873
TSER (Timestamp Echo Reply), in TCP
  definition of, 960
  Eifel Detection Algorithm and, 679
  TCP Timestamp option and, 609
  timer-based retransmission and, 665–666
TSF (time sync function), in Wi-Fi
  in 802.11 specification, 119–120
  definition of, 960
  Wi-Fi frames and, 114
TSIG (Transaction Signatures), in DNS
  definition of, 960
  transaction authentication in DNS, 911–914
TSKs (transient session keys), 838
TSOPT (timestamps option), in TCP
  definition of, 960
  Eifel Detection Algorithm using, 679
  Linux RTT estimation and, 657
  robustness of RTTM to loss and reordering, 662–664
  RTTM (RTT Measurement) with, 656–657
  TCP header, 608–610
TSPEC (traffic specification), in Wi-Fi QoS
  definition of, 960
  in HCCA, 123
TSV (Timestamp Value), in TCP
  definition of, 960
  Eifel Detection Algorithm and, 679
  RTTM with Timestamps option, 656
  TCP Timestamp option and, 608–609
TTL (Time-to-live)
  definition of, 960
  ICMP Time Exceeded message, 375, 378
  IP header fields, 184
  MRD (Multicast Router Discovery) and, 394
  name servers, 517
  QS (Quick-Start) TTL, 199
  SYN segments, 611
Tunnel endpoint, IPv6 traffic and, 46
Tunneled packets, NAT and, 310
Tunneling
  IPv4/IPv6 translation, 339
  IPv6 options for, 198
  link layer and, 149–153
  link layer attacks and, 156
Tunneling proxy servers, 302
TURN (Traversal Using Relays around NAT)
  definition of, 960
  ICE making use of, 332–334
  overview of, 326–332
  Teredo relays compared with, 482
TWA (TIME-WAIT Assassination), 630–631, 960
TXOPs (transmit opportunities), in Wi-Fi QoS
  in DCF, 123
  definition of, 960
TXT records
  definition of, 960
  DNS resource record types, 545–547
Type field, in Ethernet frame format, 85–86
Type-length-value (TLV) sets
  definition of, 959
  IPv6 options held as, 196–197
Type of Service byte. *See* ToS (Type of Service) byte

U

U-NAPTR (URI-enabled NAPTR)
  definition of, 960
  DNS resource record types, 555

U-NII (Unlicensed National Information Infrastructure)
5GHz band for, 124
definition of, 960
UBM (unicast-prefix-based multicast)
allocation of IPv4 addresses, 56
definition of, 960
UDLs (unidirectional links)
definition of, 960
link layer and, 153–154
UDP-Lite, 487–488
UDP servers
designing, 498–499
flow control and congestion control in server design, 505
foreign IP address restrictions in server design, 502–503
IP addresses and port numbers in server design, 499–500
local IP address restrictions in server design, 500–501
multiple addresses in server design, 501–502
multiple servers per port, 503–504
spanning IP address families in server design, 504
UDP (User Datagram Protocol)
attacks related to, 507–508
broadcast overhead and, 451
checksum, 475–478
connection refused error, 626
as connectionless protocols, 595
definition of, 960
examples, 478–481
flow control and congestion control in server design, 505
foreign IP address restrictions in server design, 502–503
header, 474–475
ICE and, 332
interaction between IP fragmentation and ARP/ND, 496–497
in the Internet, 506–507
introduction to, 473–474
IP addresses and port numbers in server design, 499–500
IP fragmentation and, 488–492
IPv6 and, 481–482
local IP address restrictions in server design, 500–501
maximum UDP datagram size, 497–498
multiple addresses in server design, 501–502
multiple servers per port, 503–504
NAT and, 308–309
PMTUD (Path MTU Discovery) with, 493–496
reassembly timeout, 492
sending broadcast datagrams, 439
server design, 498–499
spanning IP address families in server design, 504
STUN and, 320
summary and references, 508–510
Teredo tunneling and, 482–487
translating UDP/IPv4 and UDP/IPv6 datagrams, 505–506
transport protocols in TCP/IP suite, 15
UDP-Lite, 487–488
well-known ports for, 525–526
UEQM (unequal modulation), in 802.11n
definition of, 960
higher throughput (802.11n) support and, 127
ULAs (Unique Local IPv6 Unicast Addresses)
definition of, 960
NAT and, 310
overview of, 225
Unauthorized access attacks, 26
Unequal modulation (UEQM), in 802.11n
definition of, 960
higher throughput (802.11n) support and, 127
Unicast addresses
allocation of, 62–65
anycast addresses, 62
assigning, 65–66
C class spaces for, 35
definition of, 34
Echo Request message sent from link-local unicast address, 445–446
IIDs as basis for unicast IPv6 addresses, 43–46
multiple providers/multiple networks/multiple addresses, 68–70
overview of, 15
single provider/multiple networks/multiple addresses, 67–68
single provider/no network/single address, 66–67
single provider/single network/single address, 67
Unicast-prefix-based IPv6 multicast addresses, 58
Unicast-prefix-based multicast (UBM)
allocation of IPv4 addresses, 56
definition of, 960
Unicode, internationalization of Internet, 512
Unidirectional links (UDLs)
definition of, 960
link layer and, 153–154
Uniform Resource Locator (URL), 961
Unilateral self-address fixing. *See* UNSAF (unilateral self-address fixing)

Unique Local IPv6 Unicast Addresses. *See* ULAs (Unique Local IPv6 Unicast Addresses)
Universal Plug and Play (UPnP) framework
  definition of, 961
  direct interaction with NAT and firewalls, 337–339
Universal Resource Identifier. *See* URI (Universal Resource Identifier)
UNIX
  Berkeley version. *See* BSD (Berkeley Software Distribution)
  `rlogin`, 692
Unlicensed National Information Infrastructure (U-NII)
  5GHz band for, 124
  definition of, 960
Unreachable hosts, keepalives detecting, 795–796
UNSAF (unilateral self-address fixing)
  definition of, 961
  overview of, 317–319
  STUN (Session Traversal Utilities for NAT), 319–326
Updates
  DNS Update, 567
  dynamic DNS updates, 555–558
UPnP (Universal Plug and Play) framework
  definition of, 961
  direct interaction with NAT and firewalls, 337–339
Upper layer, TLS (Transport Layer Security), 877
UPs (user priorities), in Wi-Fi QoS, 123, 961
URG (Urgent Mechanism), in TCP
  definition of, 961
  example working with urgent data, 720–722
  overview of, 719–720
  TCP header, 590
URI-enabled NAPTR (U-NAPTR)
  definition of, 960
  DNS resource record types, 555
URI (Universal Resource Identifier)
  definition of, 961
  ENUM records and, 551–552
  NAPTR records and, 549
  URI/URN resolution, 553–554
URL (Uniform Resource Locator), 961
URN resolution, 553–554
Usage-specific keys (USRK), in EAP
  definition of, 961
  key derivation in EAP, 838
User Datagram Protocol. *See* UDP (User Datagram Protocol)
User priorities (UPs), in Wi-Fi QoS, 123, 961
User Timeout (UTO) option, in TCP
  definition of, 961
  TCP header, 611–612
USRK (usage-specific keys)
  definition of, 961
  key derivation in EAP, 838
UTC (Coordinated Universal Time), 961
UTO (User Timeout) option, in TCP
  definition of, 961
  TCP header, 611–612

V

Valid lifetime
  IA (Identity Association) and, 255
  IPv6 addresses, 252
Validating certificates, 828–831
Validating security aware resolver, in DNSSEC, 895
Variable-length subnet masks (VLSM), 41–42, 961
Variable-scope addresses, IPv6 multicast, 58
Variables, IGMP/MLD, 467–468
`vconfig` command, for manipulating 802.1p/q information in Linux, 90–91
VCs (virtual circuits)
  definition of, 961
  multiplexing and, 4
Vegas TCP algorithm, 777–778
Vendor Extension field, DHCP/BOOTP message format, 238, 246
VENONA system, 918
Virtual carrier sense, 121
Virtual circuits (VCs)
  definition of, 961
  multiplexing and, 4
Virtual LANs. *See* VLANs (virtual LANs)
Virtual private networks. *See* VPNs (virtual private networks)
Viruses, 806
VJ (Van Jacobson) compression, 141–142
VLAN identifier, 90
VLAN tag, 90
VLANs (virtual LANs)
  definition of, 961
  multicast routing, 452
  overview of, 89–92
  QoS tagging and, 145–148
VLSM (variable-length subnet masks), 41–42, 961
VoIP (Voice over IP), 961
VPNs (virtual private networks)
  connecting to Internet via, 20
  definition of, 961
  tunneling, 149

W

W3C (World Wide Web Consortium), 24, 961
Wake-on LAN (WoL), 96–97, 962

WANs (wide area networks)
definition of, 961
Internet as, 2
PPPoE and, 286–287
War driving, link layer attacks, 155
Weak host model, 220
Web caches, web proxies operating as, 302
Web of trust, public key certificates and, 822
Web proxies, 302
Web Proxy Auto-Discovery Protocol (WPAD), 302, 962
Web Proxy Autodiscovery Protocol (WRED), 962
Weighted RED (WRED), 783, 962
Well-known ports
overview of, 18
for SSH (Secure Shell), 632
for UDP or TCP, 525–526
Well-Known Prefix (WKP), in algorithmic address translation, 341, 962
WEP (wired equivalent privacy) in Wi-Fi
attacks related to, 918
definition of, 961
Wi-Fi attacks and, 155
for Wi-Fi security, 129–130
WESP (Wrapped ESP), in IPsec, 863–864, 961
White hats, 26
WHOIS service, 63–64
Wi-Fi Multimedia (WMM), in Wi-Fi QoS, 122, 962
Wi-Fi Protected Access. *See* WPA (Wi-Fi Protected Access)
Wi-Fi (wireless fidelity- IEEE 802.11)
attacks, 155
attacks related to, 918
channels and frequencies (802.11b/g), 124–125
control frames, 115–116
CSMA/CA and, 84
data frames, fragmentation, and aggregation, 116–119
definition of, 962
frames, 113–115
higher throughput (802.11n), 126–128
mesh (802.11s), 130
overview of, 111–112
physical layer, 123–124
physical layer channels and frequencies, 124–126
power save mode and time sync function, 119–120
RSTP (Rapid Spanning Tree Protocol), 120–123
security, 129–130
throughput (802.11n), 116, 126–128
Wide area networks. *See* WANs (wide area networks)
Wildcard (*)
domain names and, 526
local IP address restrictions in server design, 500–501
Wildcard address, IPv6 addresses, 632
WiMAX (Worldwide Interoperability for Microwave Access), 79, 82–83, 962
Window advertisement (window update)
in window-based flow control, 583
WSCALE (Window Scale) option in TCP, 608
Window-based flow control, 583
Window management, TCP
example of dynamic window size adjustment and flow control, 705–708
flow control and, 700–701
large buffers and auto-tuning, 715–719
sliding window protocol, 701–704
SWS (silly windows syndrome), 708–715
zero windows and TCP persistent timer, 704–705
Window probes, in TCP window management, 704
Window Size field
cwnd (congestion window), 729
flow control and, 727
TCP segments, 701
Window update
SWS (silly windows syndrome) and, 711
TCP header and, 590
window management and, 706
Windows of packets, in communication protocols, 581–582
Windows OS (Microsoft)
autotuning TCP receive windows, 715
ICS (Internet Connection Sharing), 337
IPSec implementations, 867
Wired equivalent privacy. *See* WEP (wired equivalent privacy)
Wireless fidelity. *See* Wi-Fi (wireless fidelity-802.11)
Wireless LANs (WLANs). *See also* Wi-Fi (wireless fidelity-802.11), 962
Wireshark
Flow graph, 749
monitoring TCP keepalives, 797–798
TCP Stream Graph, 707, 747–748
TCP ZeroWindowProbe, 710
viewing BPDUs with, 109
WKP (Well-Known Prefix), 341, 962
WLANs (wireless LANs). *See also* Wi-Fi (wireless fidelity-802.11), 962
WMM (Wi-Fi Multimedia), in Wi-Fi QoS, 122, 962
WoL (Wake-on LAN), 96–97, 962
WOPT. *See* WSOPT (Window Scale Option)
World Wide Web Consortium (W3C), 24, 961

World Wide Web (WWW)
definition of, 962
Internet compared to, 2–3
Worldwide Interoperability for Microwave Access (WiMAX), 79, 82–83, 962
Worms
attacks related to window management, 723
buffer overflow and, 805
types of malware, 806
WPA (Wi-Fi Protected Access)
attacks related to, 918
definition of, 962
Wi-Fi attacks and, 155
for Wi-Fi security, 129–130
WPAD (Web Proxy Auto-Discovery Protocol), 302, 962
Wrapped ESP (WESP), in IPsec, 863–864, 961
WRED (Weighted RED), 783, 962
WSOPT (Window Scale Option)
definition of, 962
TCP header, 608, 610
WWW (World Wide Web)
definition of, 962
Internet compared to, 2–3

X

X.25 protocol
definition of, 962
VCs (virtual circuits) and, 4–5
X.509 standard
certificate extensions, 827–828
file or encoding formats in, 822–823
TLS extensions, 883
validating and revoking certificates, 828–831
viewing preconfigured certificates, 823–826
XML (Extensible Markup Language)
common use with Web pages, 338
definition of, 962
XMPP (Extensible Messaging and Presence Protocol), 333, 962

Y

Yiaddr (Your IP address) field, DHCP/BOOTP message format, 237, 247
Your IP address (Yiaddr) field, DHCP/BOOTP message format, 237, 247

Z

Zero window advertisement
example of dynamic window size adjustment and flow control, 705–708
large buffers and auto-tuning, 717–719
probes, 710
SWS (silly windows syndrome) and, 711–713
TCP persistent timer and, 704–705
Zombie attacks, 26
Zone cuts, DNS and DNSSEC, 903
Zone enumeration, NSEC chain and, 900
Zone signing key (ZSK)
definition of, 962
DNSSEC, 896, 905
Zone transfers, in DNS
AXFR (full zone transfer) messages, 559–561
DNS notify and, 558–559, 564–565
initiating, 525
IXFR (incremental zone transfer) messages, 561–563
overview of, 517–518
Zones, DNS
dynamic updates, 555–558
overview of, 516–517
Zones, DNSSEC, 903
ZSK (zone signing key)
definition of, 962
DNSSEC, 896, 905