Microsoft

Understanding IPv6

3 THIRD EDITION

Covers Windows® 8 and Windows Server® 2012

Joseph Davies

Understanding IPv6

THIRD EDITION

Your essential guide to IPv6—now updated for Windows 8 and Windows Server 2012

Get in-depth technical information to put IPv6 technology to work—including networks with computers running Windows® 8 and Windows Server® 2012. Written by a networking expert, this reference explains IPv6 features and benefits, and provides detailed information to help you implement this protocol. You'll learn best practices for using IPv6 services in your Windows network, whether you're an IT professional, a network administrator, or an IT student.

Discover how to:

- Use Windows features and tools to implement IPv6 on your network
- Set up a test lab to experiment with IPv6 configuration and functionality
- Determine a practical IPv6 addressing and routing plan for your network
- Use IPv6 transition technologies to support both IPv4 and IPv6 during deployment
- Implement IPv6 security features and measures
- Deploy native IPv6 connectivity on an IPv4-only intranet
- Apply best practices from the Microsoft corporate network case study
- Test your understanding of IPv6 concepts with end-of-chapter quizzes

Web content includes:

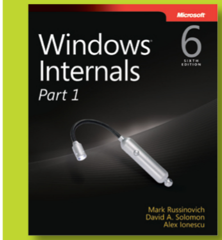
- Classroom-ready Microsoft® PowerPoint® slides for teaching IPv6
- IPv6 capture files that display packet structure and protocol processes (viewable with Microsoft Network Monitor 3.4)
- Ready to download at http://go.microsoft.com/FWLink/?Linkid=253018

*For **system requirements**, see the Introduction.*

About the Author

Joseph Davies is an award-winning author and instructor with 18 years of experience in TCP/IP, networking, and security technologies. The author of *Understanding IPv6, Second Edition* and *Windows Server 2008 TCP/IP Protocols and Services*, he also wrote "The Cable Guy," a monthly column for Microsoft TechNet and *TechNet Magazine*.

ALSO LOOK FOR



Windows Internals, Sixth Edition
Part 1

Mark Russinovich, David A. Solomon, and Alex Ionescu

ISBN 978-0-7356-4873-9

microsoft.com/mspress

ISBN: 978-0-7356-5914-8

9 780735 659148 90000

U.S.A. $49.99
Canada $52.99
[Recommended]

Operating Systems/Windows

Windows®

Microsoft®

Microsoft®

Understanding IPv6

Third Edition

Joseph Davies

Copyright © 2012 by Microsoft Corporation
All rights reserved. No part of the contents of this book may be reproduced or transmitted in any form or by any means without the written permission of the publisher.

ISBN: 978-0-7356-5914-8

1 2 3 4 5 6 7 8 9 LSI 7 6 5 4 3 2

Printed and bound in the United States of America.

Microsoft Press books are available through booksellers and distributors worldwide. If you need support related to this book, email Microsoft Press Book Support at *mspinput@microsoft.com*. Please tell us what you think of this book at *http://www.microsoft.com/learning/booksurvey*.

Microsoft and the trademarks listed at *http://www.microsoft.com/about/legal/en/us/IntellectualProperty/Trademarks/EN-US.aspx* are trademarks of the Microsoft group of companies. All other marks are property of their respective owners.

The example companies, organizations, products, domain names, email addresses, logos, people, places, and events depicted herein are fictitious. No association with any real company, organization, product, domain name, email address, logo, person, place, or event is intended or should be inferred.

This book expresses the author's views and opinions. The information contained in this book is provided without any express, statutory, or implied warranties. Neither the authors, Microsoft Corporation,
nor its resellers, or distributors will be held liable for any damages caused or alleged to be caused either directly or indirectly by this book.

Acquisitions and Developmental Editor: Ken Jones

Production Editor: Holly Bauer

Editorial Production: Octal Publishing, Inc.

Technical Reviewer: Ed Horley

Copyeditor: Richard Carey

Indexer: Ellen Troutman Zaig

Cover Design: Twist Creative • Seattle

Cover Composition: Karen Montgomery

Illustrator: Rebecca Demarest

For Kara:
My lady, my love, my life.

Contents at a Glance

	Foreword	*xxvii*
	Preface	*xxix*
	Introduction	*xxxi*
CHAPTER 1	**Introduction to IPv6**	**1**
CHAPTER 2	**IPv6 Protocol for Windows**	**17**
CHAPTER 3	**IPv6 Addressing**	**57**
CHAPTER 4	**The IPv6 Header**	**91**
CHAPTER 5	**ICMPv6**	**117**
CHAPTER 6	**Neighbor Discovery**	**131**
CHAPTER 7	**Multicast Listener Discovery and MLD Version 2**	**183**
CHAPTER 8	**Address Autoconfiguration**	**205**
CHAPTER 9	**IPv6 and Name Resolution**	**227**
CHAPTER 10	**IPv6 Routing**	**253**
CHAPTER 11	**IPv6 Transition Technologies**	**283**
CHAPTER 12	**ISATAP**	**301**
CHAPTER 13	**6to4**	**323**
CHAPTER 14	**Teredo**	**347**
CHAPTER 15	**IP-HTTPS**	**369**
CHAPTER 16	**NAT64/DNS64**	**377**
CHAPTER 17	**IPv6 Security Considerations**	**387**
CHAPTER 18	**DirectAccess**	**397**
CHAPTER 19	**Deploying IPv6 on an Intranet**	**413**
CHAPTER 20	**IPv6 on the Microsoft Corporate Network**	**437**

APPENDIX A IPv6 RFC Index 451
APPENDIX B Testing for Understanding Answers 457
APPENDIX C Setting Up an IPv6 Test Lab 487
APPENDIX D IPv6 Reference Tables 515
APPENDIX E Link-Layer Support for IPv6 519
APPENDIX F Windows Sockets Changes for IPv6 539
APPENDIX G Mobile IPv6 549
APPENDIX H Teredo Protocol Processes 605

Glossary *627*

Index *641*

Contents

Foreword . . . . *xxvii*

Preface . . . . *xxix*

Introduction . . . . *xxxi*

Chapter 1 Introduction to IPv6 1

Limitations of IPv4 1

Consequences of the Limited IPv4 Address Space 2

Features of IPv6 6

New Header Format 6

Large Address Space 6

Stateless and Stateful Address Configuration 7

IPsec Header Support Required 7

Better Support for Prioritized Delivery 7

New Protocol for Neighboring Node Interaction 8

Extensibility 8

Comparison of IPv4 and IPv6 8

IPv6 Terminology 9

The Case for IPv6 Deployment 12

IPv6 Solves the Address Depletion Problem 12

IPv6 Solves the Disjoint Address Space Problem 12

IPv6 Solves the International Address Allocation Problem 13

IPv6 Restores End-to-End Communication 13

IPv6 Uses Scoped Addresses and Address Selection 14

IPv6 Has More Efficient Forwarding 14

IPv6 Has Support for Security and Mobility 15

Testing for Understanding 15

What do you think of this book? We want to hear from you!

Microsoft is interested in hearing your feedback so we can continually improve our books and learning resources for you. To participate in a brief online survey, please visit:

microsoft.com/learning/booksurvey

Chapter 2 IPv6 Protocol for Windows 17

Architecture of the IPv6 Protocol for Windows. 17

Features of the IPv6 Protocol for Windows 19

Installed, Enabled, and Preferred by Default. .20

Basic IPv6 Stack Support. 21

IPv6 Stack Enhancements 21

GUI and Command-Line Configuration 22

Integrated IPsec Support 22

Windows Firewall Support 22

Temporary Addresses 23

Random Interface IDs 23

DNS Support. 24

Source and Destination Address Selection 24

Support for ipv6-literal.net Names. 24

LLMNR 25

PNRP. 25

Literal IPv6 Addresses in URLs 25

Static Routing 26

IPv6 over PPP 26

DHCPv6 27

ISATAP 27

6to4 27

Teredo 27

PortProxy. 28

IP-HTTPS 28

NAT64/DNS64 28

Group Policy Settings for Transition Technologies. 29

Application Support 30

Application Programming Interfaces 30

Windows Sockets 31

Winsock Kernel. 31

Remote Procedure Call 31

IP Helper 31

Win32 Internet Extensions 32

.NET Framework 32

Windows Runtime 32

Windows Filtering Platform 32

Windows Management Instrumentation Version 2 33

Manually Configuring the IPv6 Protocol. 33

Configuring IPv6 Through the Properties of Internet Protocol Version 6 (TCP/IPv6) 34

Configuring IPv6 with Windows PowerShell .37

Configuring IPv6 with the Netsh.exe Tool 37

Disabling IPv6 40

IPv6-Enabled Tools 42

Ipconfig 42

Route 43

Ping. 44

Tracert 45

Pathping. 46

Netstat 47

Displaying IPv6 Configuration with Windows PowerShell. 49

Get-NetIPInterface -AddressFamily IPv6 49

Get-NetIPAddress -AddressFamily IPv6 50

Get-NetRoute -AddressFamily IPv6 50

Get-NetNeighbor -AddressFamily IPv6 51

Displaying IPv6 Configuration with Netsh 51

Netsh interface ipv6 show interface 51

Netsh interface ipv6 show address 52

Netsh interface ipv6 show route 52

Netsh interface ipv6 show neighbors. 53

Netsh interface ipv6 show destinationcache. 53

References 53

Testing for Understanding. 54

Chapter 3 IPv6 Addressing 57

The IPv6 Address Space 57

IPv6 Address Syntax 58

Compressing Zeros 60

IPv6 Prefixes 60

Types of IPv6 Addresses. 61

Unicast IPv6 Addresses 62

Global Unicast Addresses 62

Link-Local Addresses 65

Unique Local Addresses 66

Special IPv6 Addresses 67

Transition Addresses 68

Multicast IPv6 Addresses. 68

Solicited-Node Address 70

Mapping IPv6 Multicast Addresses to Ethernet Addresses 71

Anycast IPv6 Addresses 72

Subnet-Router Anycast Address 73

IPv6 Addresses for a Host 73

IPv6 Addresses for a Router 74

Subnetting the IPv6 Address Space. 75

Step 1: Determining the Number of Subnetting Bits. 75

Step 2: Enumerating Subnetted Address Prefixes 76

IPv6 Address Allocation Strategies 81

IPv6 Interface Identifiers 82

EUI-64 Address-Based Interface Identifiers. .83

Temporary Address Interface Identifiers 87

IPv4 Addresses and IPv6 Equivalents 88

References 89

Testing for Understanding. 89

Chapter 4 The IPv6 Header 91

Structure of an IPv6 Packet 91

IPv4 Header 92

IPv6 Header 94

Values of the Next Header Field 96

Comparing the IPv4 and IPv6 Headers 97

IPv6 Extension Headers 99

Extension Headers Order 101

Hop-by-Hop Options Header 101

Destination Options Header. 105

Routing Header 107

Fragment Header 108

Authentication Header 112

Encapsulating Security Payload Header and Trailer. 112

IPv6 MTU 113

Upper-Layer Checksums 114

References 115

Testing for Understanding. 115

Chapter 5 ICMPv6 117

ICMPv6 Overview 117

Types of ICMPv6 Messages 118

ICMPv6 Header. 118

ICMPv6 Error Messages 119

Destination Unreachable 119

Packet Too Big 121

Time Exceeded 123

Parameter Problem 123

ICMPv6 Informational Messages 124

Echo Request. 124

Echo Reply 125

Comparing ICMPv4 and ICMPv6 Messages . . . 127
Path MTU Discovery . . . 127
Changes in PMTU . . . 128
References . . . 129
Testing for Understanding . . . 130

Chapter 6 Neighbor Discovery 131

Neighbor Discovery Overview . . . 131
Neighbor Discovery Message Format . . . 133
Neighbor Discovery Options . . . 134
- Source and Target Link-Layer Address Options . . . 134
- Prefix Information Option . . . 136
- Redirected Header Option . . . 139
- MTU Option . . . 141
- Route Information Option . . . 143

Neighbor Discovery Messages . . . 145
- Router Solicitation . . . 145
- Router Advertisement . . . 146
- Neighbor Solicitation . . . 150
- Neighbor Advertisement . . . 152
- Redirect . . . 155
- Summary of Neighbor Discovery Messages and Options . . . 157

Neighbor Discovery Processes . . . 158
- Conceptual Host Data Structures . . . 158
- Address Resolution . . . 159
- Neighbor Unreachability Detection . . . 163
- Duplicate Address Detection . . . 167
- Router Discovery . . . 170
- Redirect Function . . . 176

Host Sending Algorithm . . . 179
IPv4 Neighbor Messages and Functions and IPv6 Equivalents . . . 181
References . . . 181
Testing for Understanding . . . 181

Chapter 7 Multicast Listener Discovery and MLD Version 2 183

MLD and MLDv2 Overview 183

IPv6 Multicast Overview 184

Host Support for Multicast. 184

Router Support for Multicast 185

MLD Packet Structure. 188

MLD Messages. 189

Multicast Listener Query. 189

Multicast Listener Report 191

Multicast Listener Done 193

Summary of MLD 194

MLDv2 Packet Structure. 194

MLDv2 Messages 195

The Modified Multicast Listener Query 195

MLDv2 Multicast Listener Report 197

Summary of MLDv2 201

MLD and MLDv2 Support in Windows 201

References 202

Testing for Understanding. 203

Chapter 8 Address Autoconfiguration 205

Address Autoconfiguration Overview 205

Types of Autoconfiguration 205

Autoconfigured Address States 206

Autoconfiguration Process 207

DHCPv6. 210

DHCPv6 Messages 212

DHCPv6 Stateful Message Exchange 215

DHCPv6 Stateless Message Exchange 215

DHCPv6 Support in Windows 216

IPv6 Protocol for Windows Autoconfiguration Specifics 220

Autoconfigured Addresses for the IPv6 Protocol for Windows 221

References 224

Testing for Understanding. 225

Chapter 9 IPv6 and Name Resolution 227

Name Resolution for IPv6 227

DNS Enhancements for IPv6 227

LLMNR 228

Source and Destination Address Selection. 231

Source Address Selection Algorithm 233

Destination Address Selection Algorithm 235

Fixing IPv6 Brokenness in Windows Server 2012 and Windows 8. 237

Example of Using Address Selection. 238

Name Resolution Support in Windows. 240

Hosts File 240

DNS Resolver. 241

DNS Server Service 242

DNS Dynamic Update 243

DNS Zone Transfers 244

Source and Destination Address Selection .244

LLMNR Support 246

Support for ipv6-literal.net Names. 247

Peer Name Resolution Protocol 248

Name Resolution Policy Table 249

DNS Security Extensions. 250

References 250

Testing for Understanding. 251

Chapter 10 IPv6 Routing 253

Routing in IPv6 253

IPv6 Routing Table Entry Types 254

Route Determination Process 254

Strong and Weak Host Behaviors 255

Example IPv6 Routing Table for Windows. .256

End-to-End IPv6 Delivery Process 260
IPv6 on the Sending Host. 260
IPv6 on the Router. 262
IPv6 on the Destination Host. 265

IPv6 Routing Protocols. 268
Overview of Dynamic Routing. 268
Routing Protocol Technologies 269
Routing Protocols for IPv6 270

Static Routing with the IPv6 Protocol for Windows 272
Configuring Static Routing with Windows PowerShell 272
Configuring Static Routing with Netsh. 275
Configuring Static Routing with Routing and Remote Access 277

Dead Gateway Detection. 278

References 280

Testing for Understanding. 280

Chapter 11 IPv6 Transition Technologies 283

Overview. 283
Node Types 284
IPv6 Transition Addresses 284

Transition Mechanisms. 286
Using Both IPv4 and IPv6 286
IPv6-over-IPv4 Tunneling 288
DNS Infrastructure 290

Tunneling Configurations 291
Router-to-Router 292
Host-to-Router and Router-to-Host. 292
Host-to-Host. 293
Types of Tunnels. 294

Traffic Translation 296
NAT64/DNS64 296
PortProxy. 297

References 299

Testing for Understanding. 300

Chapter 12 ISATAP 301

ISATAP Overview 301

ISATAP Tunneling 302

ISATAP Tunneling Example 303

ISATAP Components 304

Router Discovery for ISATAP Hosts 306

Resolving the Name “ISATAP” 307

Using the ISATAP Router Name Group Policy Setting 311

Using the *Set-NetIsatapConfiguration -Router* Windows PowerShell Command 311

Using the *netsh interface isatap set router* Command 312

ISATAP Addressing Example 312

ISATAP Routing 313

ISATAP Communication Examples 314

ISATAP Host to ISATAP Host 314

ISATAP Host to IPv6 Host 315

Configuring an ISATAP Router 317

Example Using Windows PowerShell Commands 318

Example Using Netsh Commands 320

ISATAP in Windows Server 2012 and Windows 8 321

References 321

Testing for Understanding 321

Chapter 13 6to4 323

6to4 Overview 323

6to4 Tunneling 324

6to4 Tunneling Example 325

6to4 Components 327

6to4 Addressing Example 328

6to4 Routing 330

6to4 Support in Windows 331

6to4 Host/Router Support 331

6to4 Router Support 332

6to4 Communication Examples 336
6to4 Host to 6to4 Host/Router 336
6to4 Host to IPv6 Host 337
Example of Using ISATAP and 6to4 Together 341
Part 1: From ISATAP Host A to 6to4 Router A. 343
Part 2: From 6to4 Router A to 6to4 Router B 344
Part 3: From 6to4 Router B to ISATAP Host B 344
References 345
Testing for Understanding. 345

Chapter 14 Teredo 347

Introduction to Teredo. 347
Teredo Benefits. 348
Teredo Support in Microsoft Windows 348
Teredo and Protection from Unsolicited Incoming IPv6 Traffic. 349
Network Address Translators (NATs). 350
Teredo Components. 351
Teredo Client. 352
Teredo Server 352
Teredo Relay 353
Teredo Host-Specific Relay. 353
The Teredo Client and Host-Specific Relay in Windows 354
Teredo Addresses 356
Teredo Packet Formats. 360
Teredo Data Packet Format 360
Teredo Bubble Packets 360
Teredo Indicators 361
Teredo Routing 363
On-Link Teredo Client Destinations 364
Intersite Teredo Client Destinations 365
IPv6 Internet Destinations 365

Windows-Based Teredo Server and Relay 365

Configuring a Teredo Server 365

Configuring a Teredo Relay 366

References 367

Testing for Understanding. 367

Chapter 15 IP-HTTPS 369

Introduction to IP-HTTPS 369

IP-HTTPS Traffic. 370

IP-HTTPS Components. 371

Establishing an IP-HTTPS Connection. 372

IP-HTTPS Client Routing 373

Configuring IP-HTTPS Client Settings 374

IP-HTTPS Features in Windows Server 2012 and Windows 8 374

Summary. 375

References 375

Testing for Understanding. 375

Chapter 16 NAT64/DNS64 377

Introduction to NAT64/DNS64. 377

Limitations of NAT64/DNS64. 378

How NAT64/DNS64 Works 379

Configuration Requirements for NAT64/DNS64 379

DNS Name Query and Response 380

IPv6 Traffic from the IPv6-Only Node 382

Configuring NAT64/DNS64 in Windows Server 2012 384

Summary. 385

References 386

Testing for Understanding. 386

Chapter 17 IPv6 Security Considerations 387

IPv6 Security Considerations. 387

Authorization for Automatically Assigned Addresses and Configurations 388

Recommendations 388

Prevention of Rogue IPv6 Routers 389

Recommendations 389

Protection of IPv6 Packets. 389

Recommendations 390

Host Protection from Scanning and Attacks 390

Address Scanning. 390

Port Scanning 391

Recommendations 391

Control of Tunneled Traffic on Your Intranet 391

Recommendations 392

Control of What Traffic Is Exchanged with the Internet. 393

Recommendations 394

Summary. 395

References 395

Testing for Understanding. 396

Chapter 18 DirectAccess 397

Overview of DirectAccess 397

How DirectAccess Uses IPv6 398

DirectAccess Client Traffic over the IPv4 Internet 399

DirectAccess Client Traffic over the Intranet. 399

Force Tunneling 400

DirectAccess and IPv6 Routing 401

DirectAccess and the Role of IPsec 403

Encryption 404

Data Integrity 404

DirectAccess and the Role of the NRPT 405
NRPT Exemptions. 406
Network Location Detection. 406
Network Location Awareness 407
Network Location Detection Process 407
How DirectAccess Works 408
DirectAccess Client on the Intranet 408
DirectAccess Client on the Internet 409
Summary. 411
References 412
Testing for Understanding. 412

Chapter 19 Deploying IPv6 on an Intranet 413

Introduction 413
Planning for IPv6 Deployment 413
Platform Support for IPv6 414
Application Support for IPv6 415
Network Management Infrastructure Support for IPv6 415
Unicast IPv6 Addressing Architecture 416
Tunnel-Based IPv6 Connectivity 417
Other IPv6 Transition Technologies 420
Native IPv6 Connectivity 421
Name Resolution with DNS 422
Native IPv6 Addressing Allocation 423
Host-Based Security and IPv6 Traffic 424
Controlled or Prioritized Delivery for IPv6 Traffic 425
Deploying IPv6 427
Obtain Global Address Space 427
Set Up an IPv6 Test Network 427
Begin Application Migration 428
Configure DNS Infrastructure to Support AAAA Records and Dynamic Updates. 430
Upgrade IPv4-Only Hosts to IPv6/IPv4 Hosts. 430
Begin Deploying a Native IPv6 Infrastructure. 431

Connect Portions of Your Intranet over the IPv4 Internet 432
Connect Portions of Your Intranet over the IPv6 Internet 433
Summary. 433
References 434
Testing for Understanding. 435
Chapter 20 IPv6 on the Microsoft Corporate Network 437
Introduction 437
Characteristics of the Microsoft Corpnet. 438
History of IPv6 in Microsoft 438
Deployment Philosophy 439
Current Deployment of IPv6 on the Microsoft Corpnet 440
Short and Long-Term Plans for IPv6 on the Microsoft Corpnet. 443
Deployment Details 444
Addressing Plan and Routing Infrastructure. 444
DirectAccess 444
Security for IPv6 Traffic on the Microsoft Corpnet 445
Deployment Planning and Recommendations 446
Overall Planning. 446
Deployment Recommendations 448
Summary. 449
Testing for Understanding. 450
Appendix A IPv6 RFC Index 451
General 451
Addressing 451
Applications 452
Sockets API. 452
Transport Layer 452
Internet Layer. 453
Network Layer Security 454
Link Layer 454

Routing .454
IPv6 Transition Technologies .455

Appendix B Testing for Understanding Answers 457

Chapter 1: Introduction to IPv6 .457
Chapter 2: IPv6 Protocol for Windows .459
Chapter 3: IPv6 Addressing .461
Chapter 4: The IPv6 Header .464
Chapter 5: ICMPv6 .465
Chapter 6: Neighbor Discovery .466
Chapter 7: Multicast Listener Discovery and MLD Version 2 .469
Chapter 8: Address Autoconfiguration .470
Chapter 9: IPv6 and Name Resolution .472
Chapter 10: IPv6 Routing .473
Chapter 11: IPv6 Transition Technologies .475
Chapter 12: ISATAP .476
Chapter 13: 6to4 .477
Chapter 14: Teredo .478
Chapter 15: IP-HTTPS .479
Chapter 16: NAT64/DNS64 .480
Chapter 17: IPv6 Security Considerations .481
Chapter 18: DirectAccess .482
Chapter 19: Deploying IPv6 on an Intranet .483
Chapter 20: IPv6 on the Microsoft Corporate Network .484

Appendix C Setting Up an IPv6 Test Lab 487

IPv6 Test Lab Setup .487
Hardware and Software Requirements .489

Steps for Configuring the IPv6 Test Lab 489

Step 1: Set Up the Base Configuration Test Lab 490

Step 2: Configure the Test Lab for the Corpnet2 Subnet 490

Step 3: Demonstrate Default IPv6 Connectivity 495

Step 4: Demonstrate ISATAP-Based IPv6 Connectivity 496

Step 5: Demonstrate Native IPv6 Connectivity. 498

Step 6: Demonstrate 6to4-Based Connectivity. 501

Snapshot the Configuration 503

Additional IPv6 Test Lab Exercises 503

Demonstrate DHCPv6 503

Demonstrate DNS Zone Transfers over IPv6. 507

Demonstrate an IPv6-Only Environment. 509

Appendix D IPv6 Reference Tables 515

Appendix E Link-Layer Support for IPv6 519

Basic Structure of IPv6 Packets. 519

LAN Media 519

Ethernet: Ethernet II. 520

Network Monitor Capture 521

Ethernet: IEEE 802.3 SNAP 521

IEEE 802.11 523

Frame Control Field 525

Token Ring: IEEE 802.5 SNAP 526

FDDI 528

WAN Media 529

PPP 530

X.25. 531

Frame Relay. 533

ATM: Null Encapsulation. 534

ATM: SNAP Encapsulation 536

IPv6 over IPv4 537

References 538

Appendix F **Windows Sockets Changes for IPv6** 539

Added Constants. 540

Address Data Structures. 540

- in6_addr. 540
- sockaddr_in6 540
- sockaddr_storage 541

Wildcard Addresses 541

- *in6addr_loopback* and *IN6ADDR_LOOPBACK_INIT* 542

Core Sockets Functions 542

Name-to-Address Translation. 542

Address-to-Name Translation. 544

- Using *getaddrinfo* 545

Address Sorting 545

Address Conversion Functions 546

Socket Options 546

New Macros 547

References 547

Appendix G **Mobile IPv6** 549

Overview. 549

- Mobile IPv6 Components 549
- Mobile IPv6 Transport Layer Transparency 551

Mobile IPv6 Messages and Options 552

- Mobility Header and Messages 552
- Type 2 Routing Header 554
- Home Address Option for the Destination Options Header 555
- ICMPv6 Messages for Mobile IPv6 556
- Modifications to Neighbor Discovery Messages and Options. . . . 558

Mobile IPv6 Data Structures 561

- Binding Cache. 561
- Binding Update List 562
- Home Agents List. 563

Correspondent Registration 564

Return Routability Procedure. 565

Detecting Correspondent Nodes That Are Not Mobile IPv6–Capable 567

Mobile IPv6 Message Exchanges 567

Data Between a Mobile Node and a Correspondent Node 567

Binding Maintenance 574

Home Agent Discovery 578

Mobile Prefix Discovery 580

Mobile IPv6 Processes 582

Attaching to the Home Link. 582

Moving from the Home Link to a Foreign Link 583

Moving to a New Foreign Link 592

Returning Home. 594

Mobile IPv6 Host Sending Algorithm 597

Mobile IPv6 Host Receiving Algorithm. 600

References 603

Appendix H Teredo Protocol Processes 605

Initial Configuration for Teredo Clients. 606

Network Monitor Capture 608

Maintaining the NAT Mapping. 610

Initial Communication Between Teredo Clients on the Same Link 611

Initial Communication Between Teredo Clients in Different Sites 612

Cone NAT. 612

Restricted NAT 613

What do you think of this book? We want to hear from you!

Microsoft is interested in hearing your feedback so we can continually improve our books and learning resources for you. To participate in a brief online survey, please visit:

microsoft.com/learning/booksurvey

Initial Communication from a Teredo Client to a Teredo Host-Specific Relay 614

Cone NAT. 614

Restricted NAT 616

Initial Communication from a Teredo Host-Specific Relay to a Teredo Client 617

Cone NAT. 617

Restricted NAT 618

Initial Communication from a Teredo Client to an IPv6-Only Host 619

Cone NAT. 619

Restricted NAT 621

Initial Communication from an IPv6-Only Host to a Teredo Client 622

Cone NAT. 623

Restricted NAT 624

References 625

Glossary *627*

Index *641*

Foreword

When Joe first asked me to write a foreword for this latest edition of *Understanding IPv6*, I looked to forewords from previous editions as well as other networking books, hoping to draw some inspiration. As will become increasingly obvious, my writing skills are not as honed as Joe's.

Looking back was actually incredibly useful to me, because it clearly accentuated what has changed in the last few years. Microsoft has supported IPv6—the next generation of the Internet Protocol—since its inception. We built support into the Windows stack, rearchitected our platform to enable developers to take advantage of IPv6, and over the past 10 years have been extending support across the company.

The Internet Protocol is the routing and transit protocol for the Internet, the largest and most important assembly of computing infrastructure of our time. IPv6 is going to make the Internet better, by allowing direct connectivity between host, whether they be family members video chatting or business information zooming between data centers.

We frequently take the time to remind everyone of our commitment to the realization of the end-benefits of IPv6. We do this for multiple reasons. We take pride in our work, certainly, and it gives us great pleasure to do our part to make technology a bit better. But perhaps more important, these reminders are evangelical; they assure customers, partners, and readers that IPv6 is something worthy of attention, worthy of adoption.

For many years, this was a difficult task. People didn't want IPv6. The growth and maturation of IPv4 survival strategies, such as large-scale network address translation, threatened the inevitability of IPv6 adoption. That's the truth. Some in the networking world might try to revise the past 10 years: the broken routing equipment, the inadequate software, the legends of danger, IPv6 performance problems, IPv6 security problems, IPv6 money issues, and IPv6 zombies.

That darkness was real, but that darkness has past. In the past 24 months, we've made immense progress toward the goal of upgrading the Internet. IPv6 is no longer the next-generation Internet Protocol; it has become the now-generation Internet Protocol.

The World IPv6 Launch in June 2012 marked a key turning point in this transition. When you read this book, some of the most important web services in the world, not only from Microsoft but from across the technology community, are operational on the IPv6 Internet. Millions of users with IPv6-ready computers are using IPv6 to interact with these services and with one another. The apps, the operating systems, the routing infrastructure, the ISPs, and the services are not merely ready, they're activated.

Joe’s book, in all its editions, has always been *the* IPv6 reference; it's a fantastic medium for anyone interested in networking for understanding the Internet Protocol and its evolution. But as you read through this edition, I hope you gain not only the ability to understand and build networks by using IPv6, but also acquire a clearer perception of the changes happening all around you. The reality of how you search the Internet, play games with your friends, and access workplace resources is increasingly an IPv6 story.

When talking about the IPv6 story, we always note our commitment to ensuring that everyday users don’t notice any change, or sense that their experience has diminished as we transition. It shouldn’t matter whether your connection is over IPv4 or IPv6. You should have an Internet experience that is fast, reliable, and enjoyable, with the only evidence of the IPv6 transition being the lingering feeling that things simply got better.

After reading this book, you’ll likely be able to notice more than just that lingering feeling. The details, flags, and bits that make up IPv6 and the Internet’s evolution will become absolutely clear.

The Internet is going through an asynchronous, distributed, and transformative change at its very foundation. That change includes more than software or hardware; it involves a swath of people who work in networking, who use those systems, who architect networks, or who build apps.

By understanding IPv6 and this transformation, you contribute to its forward progress. Your journey becomes part of the greater tale of this technological evolution.

Thank you, and good luck.

Chris Palmer
IPv6 Program Manager,
Microsoft

Preface

The first edition of this book began in the spring of 1999. At that time, I developed a set of slides and presented an "Introduction to IPv6" course at Bellevue Community College in Bellevue, Washington, to four students. Although the turnout was not what I expected, the time spent learning IPv6, creating the slide presentation, and presenting IPv6 technology to these curious students proved to be an invaluable experience and prepared a firm foundation for future endeavors.

In 2000, as a technical writer for Windows, I wrote a white paper titled "Introduction to IP version 6" that is published on the Microsoft Windows IPv6 website (*www.microsoft.com/ipv6*) and generally inserted myself in any documentation task associated with IPv6. I also developed and delivered an internal course called "IPv6 Overview" with help on the topic of Windows Sockets from Tom Fout. Beginning in October 2000, this one-day course was taught to Microsoft software design engineers, software test engineers, program managers, and technical writers.

My transition to a program manager for technical content development afforded me the time, focus, and experience to turn the "IPv6 Overview" courseware and numerous other white papers and articles about IPv6 into *Understanding IPv6* (Microsoft Press, ISBN 978-0735612457), the first edition of this book. Between its first publication in November 2002 and January 2008, I continued to develop content for IPv6, supporting interim releases of IPv6 technology for Windows XP and the releases of Windows Server 2008 and Windows Vista, which have fully integrated IPv6 support for services and applications. The result of those efforts was the second edition of *Understanding IPv6*.

Between January 2008 and May 2012, I continued to follow the evolution of IPv6 in Windows, through Windows 7 and Windows Server 2008 R2 (writing detailed planning, deployment and troubleshooting information for DirectAccess) and on into the development of Windows Server 2012 and Windows 8. This third edition encapsulates all of these efforts.

It is my fervent hope that the work that I started in the spring of 1999 has culminated in a well-organized and readable text from which you can learn and understand the concepts, principles, and processes of IPv6.

Joseph Davies

Introduction

Due to the following recent events, the importance of Internet Protocol version 6 (IPv6) to the future of the Internet and organization intranets is now without question:

- On February 3, 2011, the Internet Corporation for Assigned Names and Numbers (ICANN) joined the Number Resources Organization (NRO), the Internet Architecture Board (IAB), and the Internet Society to announce that the pool of public Internet Protocol version 4 (IPv4) Internet addresses has now been completely allocated. Public IPv4 address space still exists to be assigned to organizations by regional address authorities, but there is no more public IPv4 address space in reserve.
- On June 8, 2011, Microsoft and other members of the Internet Society (ISOC) participated in World IPv6 Day to temporarily test connectivity and performance issues with dual-stack (IPv4 and IPv6) Internet properties.
- In April of 2012, the Internet Engineering Task Force (IETF) published Request for Comments (RFC) 6540, “IPv6 Support Required for All IP-Capable Nodes.” This Best Current Practice RFC advises that IPv6 support be required for all network nodes, in addition to IPv4.
- On June 6, 2012, Microsoft and other members of the ISOC participated in World IPv6 Launch to permanently enable dual stack on Internet properties.

The time has come to embrace, learn, and understand IPv6.

Pursuant to this need, this book is a straightforward discussion of the concepts, principles, and processes of IPv6 and how it is supported by the Microsoft Windows Server 2012, Windows Server 2008 R2, Windows Server 2008, Windows 8, Windows 7, and Windows Vista operating systems. Note that this book does not contain programming code–level details of the IPv6 protocol for these versions of Windows, such as structures, tables, buffers, or coding logic. These details are highly guarded Microsoft intellectual property that is of interest only to a relative handful of software developers. However, this book does contain details of how the Microsoft implementation of IPv6 in these versions of Windows works for described processes and how to modify default behaviors with Windows PowerShell and Netsh.exe tool commands, Group Policy settings, and registry values.

The purpose of this book is to provide an educational vehicle with which you can learn IPv6 to a fair level of technical depth—the terms, the addresses, the protocols, and the processes—to prepare you for planning, deployment, and operation of a native IPv6 infrastructure on your intranet.

> **Note** The contents of this book reflect the Internet standards for IPv6 and the feature set of the IPv6 protocol for Windows Server 2008 R2, Windows Server 2008, Windows 7, Windows Vista and the Release Preview versions of Windows Server 2012 and Windows 8. For information about changes in Internet standards and the IPv6 protocol for Windows Server 2012 and Windows 8 past the Release Preview version, go to the Microsoft Windows IPv6 website at *http://www.microsoft.com/ipv6*.

Who Should Read This Book

This book is intended for the following audiences:

- **Windows networking consultants and planners** This group includes anyone who will be planning for an eventual IPv6 migration with Windows.
- **Microsoft Windows network administrators** This group includes anyone who manages an IPv4-based network and wants to gain technical knowledge about IPv6 and its implementation in Windows.
- **Microsoft Certified Systems Engineers (MCSEs) and Microsoft Certified Trainers (MCTs)** Regardless of the eventual IPv6 content for Microsoft Official Curriculum (MOC) courseware for Windows Server, this book can be a standard reference for MCSEs and MCTs for IPv6 technology.
- **General technical staff** Because this book is mostly about IPv6 protocols and processes, independent of its implementation in Windows Server, general technical staff can use this book as an in-depth primer on IPv6 technologies.
- **Information technology students** This book originated as courseware for internal Microsoft software developers, testers, and program managers; thus it retains its capability as a textbook for IPv6 courses taught at an organization or educational institution, using Windows as the example IPv6 implementation.

What You Should Know Before Reading This Book

This book assumes a foundation of networking knowledge that includes basic networking concepts, widely used networking technologies, and sound knowledge of the TCP/IP suite. Wherever possible, I try to facilitate the reader's transition to IPv6 by comparing it with the corresponding feature, behavior, or component of IPv4.

For a firm foundation of knowledge of the TCP/IP protocol suite, let me recommend as a prerequisite that you read my other book, *Windows Server 2008 TCP/IP Protocols and Services* (Microsoft Press, 2008). Like this book, this resource is mostly about implementation-independent protocols and processes. As the author of this resource, I might be a bit biased; however, this book was written with *Windows Server 2008 TCP/IP Protocols and Services* in mind and builds upon it.

Organization of This Book

In this edition, I have organized the chapters into sections and the chapters within each section build upon each other in a logical fashion. For example, it is difficult to understand Neighbor Discovery processes without first understanding IPv6 addressing, the IPv6 header, and Internet Control Message Protocol for IPv6 (ICMPv6). Likewise, it's almost impossible to understand IPv6 transition technologies without first understanding IPv6 addressing, Neighbor Discovery processes, name resolution, and routing. The chapters lead to a discussion of planning for deployment, which requires an understanding of many elements of the preceding chapters.

Appendices of This Book

This book contains the following appendices:

- **Appendix A: IPv6 RFC Index** A listing of the RFCs and Internet drafts for IPv6 that are the most relevant to the IPv6 implementation in Windows at the time of this book's publication. This appendix is not designed to be an exhaustive list and will certainly be obsolete at some level after this book is printed.
- **Appendix B: Testing for Understanding Answers** Provides answers to the questions in the "Testing for Understanding" section for each chapter, which contain a series of review questions pertaining to the material in the chapter.

- **Appendix C: Setting Up an IPv6 Test Lab** This appendix answers the question, "How do I get it going so that I can play with it?" By using the instructions in this appendix, you can take five computers and create an IPv6 test lab to test address autoconfiguration, routing, and name resolution. At the end, you are left with a working IPv4 and IPv6 or IPv6-only test network with which you can experiment on your own.
- **Appendix D: IPv6 Reference Tables** A reprinting of the most relevant IPv6 tables of IPv6 protocol field values and other parameters.
- **Appendix E: Link-Layer Support for IPv6** A discussion of link-layer encapsulation of IPv6 packets for typical local area network (LAN) and wide area network (WAN) technologies.
- **Appendix F: Windows Sockets Changes for IPv6** A description of the enhancements to Windows Sockets to support both IPv6 and IPv4 at the same time.
- **Appendix G: Mobile IPv6** An in-depth discussion of Mobile IPv6, a protocol by which an IPv6 host can change locations and addresses while maintaining existing transport layer connections.
- **Appendix H: Teredo Protocol Processes** An in-depth discussion of the processes that a Teredo client uses to perform address autoconfiguration and initiate communication with other IPv6-capable hosts.

About the Companion Content

The companion content for this book, available at *http://www.microsoftpressstore.com/title/9780735659148*, includes the following:

- **Network Monitor captures** Throughout the book, packet structure and protocol processes are illustrated with actual IPv6 packets displayed by using Microsoft Network Monitor 3.4, a frame capturing and viewing program (also known as a network sniffer) that is provided free of charge by Microsoft. The display of the frames within the capture files depends on the version of Network Monitor that you are using. To install Network Monitor 3.4, see the Network Monitor blog at *http://blogs.technet.com/netmon/*.

- **Training slides** This is a set of Microsoft Office PowerPoint 2007 files that can be used along with this book to teach IPv6. For more information, see "A Special Note to Teachers and Instructors." To view the training slides, you need PowerPoint 2007 or later or the PowerPoint Viewer 2007. You can install PowerPoint Viewer 2007 from *Replace with http://www.microsoft.com/en-us/download/details.aspx?id=6*.

System Requirements

To view the book's capture files (*.cap), you must have Microsoft Network Monitor 3.4 or later. You can install Microsoft Network Monitor 3.4 from *http://blogs.technet.com/b/netmon/*.

IPv6 Protocol and Windows Product Versions

There are different versions of the Microsoft IPv6 protocol for Windows. In this book, I have chosen to confine the discussion to the IPv6 implementation in Windows Server 2012, Windows Server 2008 R2, Windows Server 2008, Windows 8, Windows 7, and Windows Vista. IPv6 in previous versions of Windows is typically not described, except as a point of contrast to IPv6 in these more recent versions of Windows.

A Special Note to Teachers and Instructors

This book originated from courseware and retains many of the inherent attributes, including objectives at the beginning of each chapter and review questions at the end of each chapter. If you are a teacher or instructor tasked with inculcating an understanding of IPv6 protocols and processes in others, I strongly urge you to consider using this book, the training slides found in the companion content for this book, and the IPv6 test lab instructions in Appendix C as a basis for your own IPv6 course.

The training slides are included to provide a foundation for your own slide presentation. The included slides contain either bulleted text or my original PowerPoint diagrams, which are synchronized with their chapter content. Because the slides were completed after the final book pages were done, there might be minor differences between the slides and the chapter content. These changes were made to enhance the ability to teach an IPv6 course based on the book.

The template I have chosen for the included slides is intentionally simple so that there are minimal issues with text and drawing color translations when you switch to a different template. Please feel free to customize the slides as you see fit.

If you are designing an implementation-independent IPv6 technology course, I suggest that you skip Chapter 2, "IPv6 Protocol for Windows," and cover Appendix E, "Link-Layer Support for IPv6," after Chapter 4, "The IPv6 Header."

For hands-on exercises, I encourage you to have your students build out the IPv6 test lab that is described in Appendix C, "Setting Up an IPv6 Test Lab." This can be done by each student on a server computer that can host five computers in a virtualized environment. The resulting test lab can be used for hands-on configuration exercises; analysis of IPv6 network traffic with Network Monitor (based on the captures provided with the companion content for this book or traffic captured on the test lab subnets); experimentation with IPv6 transition technologies and migration from an IPv4-only network to an IPv6-only network; and for application development and testing.

As a fellow instructor, I wish you success in your efforts to teach this interesting and important new technology to others.

Disclaimers and Support

This book represents a best-effort snapshot of information available at the time of its publication for IPv6 standards and the implementation of IPv6 and related protocols in Windows Server 2008 R2, Windows Server 2008, Windows 7, Windows Vista, and the Release Preview versions of Windows Server 2012 and Windows 8. Changes made to Windows Server 2012 and Windows 8 that were made after the Release Preview version or to IETF standards after May 30, 2012 are not reflected in this book.

To obtain the latest information about IETF standards for IPv6, go to the IETF website at *http://www.ietf.org*.

Acknowledgments

I would like to the thank the following people at Microsoft for participating in the technical reviews and for contributing content to the chapters and appendixes of the third edition of this book: Vivek Bhanu, Onur Filiz, Firat Kiyak, Darene Lewis, Bill Murray, Chidambaram Muthu, Tim Quinn, Pat Telford, Ben Schultz, Sean Siler, and Jeromy Statia. I would like to give special thanks to IPv6 Program Manager, Christopher Palmer, for the

Foreword. I would also like to give honorable mention to Dmitry Anipko, a senior software development engineer on the Windows Networking Core development team, who gave me very detailed feedback on both standards-based IPv6 and the implementation details of IPv6 in Windows Server 2012 and Windows 8.

To make this book a published reality, I would like to thank Ken Jones (Senior Editor at O'Reilly Media), my long-time professional colleague and IPv6 enthusiast Ed Horley for his great suggestions about current IPv6 industry trends and considerations (Technical Editor), Holly Bauer (O'Reilly Production Editor), and Richard Carey (Copyeditor).

And last, I would like to express my thanks and appreciation to my wife, Kara, and daughter, Katie, for their patience and tolerance for my time away during the last weeks of writing.

Support & Feedback

The following sections provide information on errata, book support, feedback, and contact information.

Errata

We've made every effort to ensure the accuracy of this book and its companion content. Any errors that have been reported since this book was published are listed on our Microsoft Press site:

http://www.microsoftpressstore.com/title/ 9780735659148

If you find an error that is not already listed, you can report it to us through the same page.

If you need additional support, email Microsoft Press Book Support at *mspinput@microsoft.com*.

Please note that product support for Microsoft software is not offered through the addresses above.

We Want to Hear from You

At Microsoft Press, your satisfaction is our top priority, and your feedback our most valuable asset. Please tell us what you think of this book at:

http://www.microsoft.com/learning/booksurvey

The survey is short, and we read every one of your comments and ideas. Thanks in advance for your input!

Stay in Touch

Let us keep the conversation going! We are on Twitter: http://twitter.com/MicrosoftPress

CHAPTER 1

Introduction to IPv6

At the end of this chapter, you should be able to do the following:

- Describe the shortcomings of Internet Protocol version 4 (IPv4) and the modern-day Internet, and describe how Internet Protocol version 6 (IPv6) addresses these shortcomings.
- Describe how the address depletion problem of IPv4 leads to the use of Network Address Translators (NATs) and problems with end-to-end communication.
- List and describe the features of IPv6.
- List and describe the key differences between IPv4 and IPv6.
- State the reasons for and business value of deploying IPv6.

Limitations of IPv4

The current version of IP (known as version 4 or IPv4) has not changed substantially since Request for Comments (RFC) 791, which was published in 1981. IPv4 has proven to be robust, easily implemented, and interoperable. It has stood up to the test of scaling an internetwork to a global utility the size of today's Internet. This is a tribute to its initial design.

However, the initial design of IPv4 did not anticipate the following:

- **The recent exponential growth of the Internet and the impending exhaustion of the IPv4 address space** Although the 32-bit address space of IPv4 allows for 4,294,967,296 addresses, previous and current allocation practices limit the number of public IPv4 addresses to a few hundred million. As a result, public IPv4 addresses have become relatively scarce, forcing many users and some organizations to use a NAT to map a small number of public IPv4 addresses to multiple private IPv4 addresses. Although NATs promote reuse of the private address space, they violate the fundamental design principle of the original Internet that all nodes have a unique, globally reachable address, thus preventing true end-to-end connectivity for all types of networking applications.

  Additionally, the rising prominence of Internet-connected devices and appliances ensures that the public IPv4 address space will eventually be depleted.

- **The need for simpler configuration** Most current IPv4 implementations must either be manually configured or use a stateful address configuration protocol such as Dynamic Host Configuration Protocol (DHCP). With more computers and devices using IP, there is a need for a simpler and more automatic configuration of addresses and routing configuration that does not rely on the administration of a DHCP infrastructure.

- **The requirement for security at the Internet layer** Private communication over a public medium such as the Internet requires cryptographic services that protect the data being sent from being viewed or modified in transit. Although a standard now exists for providing security for IPv4 packets (known as Internet Protocol security, or IPsec), this standard is optional for IPv4, and additional security solutions, some of which are proprietary, are prevalent.

- **The need for better support for prioritized and real-time delivery of data** Although standards for prioritized and real-time delivery of data—sometimes referred to as Quality of Service (QoS)—exist for IPv4, real-time traffic support relies on the 8 bits of the historical IPv4 Type of Service (TOS) field and the identification of the payload, typically using a User Datagram Protocol (UDP) or Transmission Control Protocol (TCP) port. Unfortunately, the IPv4 TOS field has limited functionality and, over time, has been redefined and has different local interpretations. The current standards for IPv4 use the TOS field to indicate a Differentiated Services Code Point (DSCP), a value set by the originating node and used by intermediate routers for prioritized delivery and handling. Additionally, payload identification that uses a TCP or UDP port is not possible when the IPv4 packet payload is encrypted.

To address these and other concerns, the Internet Engineering Task Force (IETF) has developed a suite of protocols and standards known as IP version 6 (IPv6). This new version, previously called IP-The Next Generation (IPng), incorporates the concepts of many proposed methods for updating the IPv4 protocol. IPv6 is designed to have minimal impact on upper-layer and lower-layer protocols and to avoid the random addition of new features.

Consequences of the Limited IPv4 Address Space

Because of the relative scarcity of public IPv4 addresses, NATs are being deployed to reuse the IPv4 private address space. In some configurations, there might be multiple levels of NATs between the client computer and the Internet. Although NATs do allow more clients to connect to the Internet, they also act as traffic bottlenecks and barriers to some types of communications.

Let’s examine the operation of a NAT to illustrate why network address translation is a nonscalable, stopgap solution that impairs end-to-end communication.

For example, say that a small business uses the 192.168.0.0/24 private IPv4 address prefix for its intranet and has been assigned the public IPv4 address of 131.107.47.119 by its Internet service provider (ISP). The NAT deployed at the edge of this network maps all private addresses on 192.168.0.0/24 to the public address of 131.107.47.119. The NAT uses dynamically chosen TCP and UDP ports to map internal (intranet) data streams to external (Internet) data streams. Figure 1-1 shows this example configuration.

If a private host assigned the private IPv4 address 192.168.0.10 uses a web browser to connect to the web server at 157.60.13.9, the private host creates an IPv4 packet with the following:

- Destination address: 157.60.13.9
- Source address: 192.168.0.10
- Destination TCP port: 80
- Source TCP port: 1025

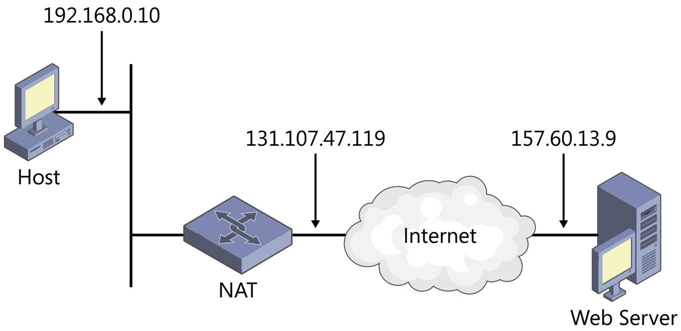


FIGURE 1-1 A NAT example.

This IPv4 packet is then forwarded to the NAT, which typically translates the source address and source TCP port of the outgoing packet to the following:

- Destination address: 157.60.13.9
- **Source address: 131.107.47.119**
- Destination TCP port: 80
- **Source TCP port: 5000**

The NAT keeps the mapping of {192.168.0.10, TCP 1025} to {131.107.47.119, TCP 5000} in a local translation table for future reference.

The translated IPv4 packet is sent over the Internet. The response is sent back by the web server and received by the NAT. When received, the packet contains the following:

- Destination address: 131.107.47.119
- Source address: 157.60.13.9
- Destination TCP port: 5000
- Source TCP port: 80

The NAT checks its translation table, locates the entry that was created when the initial packet was sent, translates the destination address and destination TCP port, and forwards the packet to the host at 192.168.0.10. The forwarded packet contains the following:

- **Destination address: 192.168.0.10**
- Source address: 157.60.13.9
- **Destination TCP port: 1025**
- Source TCP port: 80

For outgoing packets from the NAT, the source IPv4 address (a private address) is mapped to the ISP-assigned address (a public address), and the source TCP/UDP port numbers are mapped to different TCP/UDP port numbers. For incoming packets to the NAT, the destination IPv4 address (a public address) is mapped to the original intranet address (a private address), and the destination TCP/UDP port numbers are mapped back to their original TCP/UDP port numbers.

Normal network address translation relies on the following:

- **Address translation** Translation of the IPv4 addresses in the IPv4 header
- **Port translation** Translation of the TCP port numbers in the TCP header or of the UDP port numbers in the UDP header

Address and port translation lowers the forwarding performance of the NAT because of the additional operations that must be performed on each packet. As a result, NATs are typically not deployed in large-scale environments. However, recent development of standards such as carrier-grade NATs (CGNs) promise to scale NAT to enterprises and large ISPs.

To make modifications to the IPv4 packet beyond address or port translation requires additional processing and software components on the NAT called *NAT editors*. HyperText Transfer Protocol (HTTP) traffic on the World Wide Web does not require a NAT editor because all HTTP traffic requires only address and TCP port translation. However, NAT editors are required in the following situations:

- **An IPv4 address, TCP port, or UDP port is stored elsewhere in the payload.** For example, File Transfer Protocol (FTP) stores the dotted decimal representation of IPv4 addresses in the FTP header for the FTP PORT command. If the NAT does not properly translate the IPv4 address within the FTP header for the FTP PORT command and adjust the TCP sequence numbers in the data stream, connectivity and data transfer problems will occur.
- **TCP or UDP is not used to identify the data stream.** For example, Point-to-Point Tunneling Protocol (PPTP) tunneled data does not use a TCP or UDP header. Instead, PPTP uses a Generic Routing Encapsulation (GRE) header and the Call ID field of the GRE header to identify the data stream. If the NAT does not properly translate the Call ID field within the GRE header, connectivity problems will occur.

Most traffic can traverse a NAT because either the packets require only address or port translation or a NAT editor is present to modify the payload appropriately. However, some traffic cannot traverse a NAT. If the data requiring translation is in an encrypted part of the packet, translation is not possible. For IPsec-protected packets, address and port translation can invalidate the packet's integrity. IPsec NAT-Traversal (NAT-T) is a recent Internet standard that allows some types of IPsec-protected packets to be translated by a NAT.

An additional problem with NATs is their effect on peer-to-peer applications. In the peer-to-peer communication model, peers can act as either the client or the server and initiate communications to each other. If a peer is behind a NAT, two addresses are associated with it, one that is known to the peer behind the NAT (the private address) and one that is known in front of the NAT (the public address).

Let's examine a simple configuration in which NATs can cause problems for peer-to-peer applications. Figure 1-2 shows an intranet with a NAT at its edge.

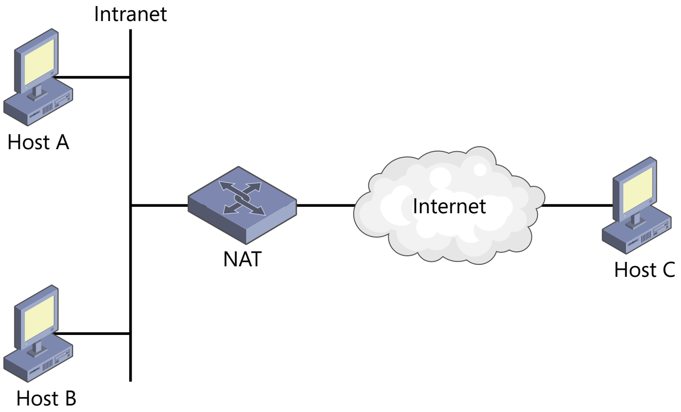


FIGURE 1-2 NAT and peer-to-peer applications.

For a peer-to-peer application running on all hosts, Host A can initiate a session with Host B (directly reachable on its link) and with Host C. However, Host A cannot inform Host C of the public address and port number of Host B because Host A does not know it. Also, Host C cannot initiate a session with either Host A or Host B without an existing translation table entry to translate the inbound connection-request packets to Host A's private address and port. Even with the table entry, Host C might not be able to initiate a session with both Host A and Host B because both hosts are known by the same public IPv4 address.

To make matters worse, it is a more common situation to have each Internet peer behind a NAT. To solve these problems, the peer-to-peer or multiple-party applications must be modified to be NAT-aware or use a NAT traversal technology, resulting in additional complexity. Also, some NAT-aware applications use an echo server to automatically discover their public address and port number, which adds costs for independent software vendors (ISVs) to deploy and maintain echo servers on the Internet.

NATs are a makeshift measure to extend the life of the IPv4 public address space, and they are not a solution to the IPv4 public address space problem. NATs work best for reusing the private address space for client computers and client/server-based communication when the client behind the NAT initiates the communication. Most server computers still need unambiguous public addresses. Although a server can be placed behind a NAT, the NAT must be configured manually with a static translation table entry to translate the inbound packets to the server's private address and port. In peer-to-peer communications, each end acts as both client and server and, therefore, peers separated by NATs might not operate correctly and must be modified for NAT awareness.

Features of IPv6

The following list summarizes the features of the IPv6 protocol:

- New header format
- Large address space
- Stateless and stateful address configuration
- IPsec header support required
- Better support for prioritized delivery
- New protocol for neighboring node interaction
- Extensibility

New Header Format

The IPv6 header has a new format that is designed to minimize header processing. This is achieved by moving both nonessential and optional fields to extension headers that are placed after the IPv6 header. The streamlined IPv6 header is more efficiently processed at intermediate routers.

IPv4 headers and IPv6 headers are not interoperable. IPv6 is not a superset of functionality that is backward compatible with IPv4. A host or router must use an implementation of both IPv4 and IPv6 to recognize and process both header formats. The new default IPv6 header is only twice the size of the default IPv4 header, even though the number of bits in IPv6 addresses is four times larger than in IPv4 addresses.

Large Address Space

IPv6 has 128-bit (16-byte) source and destination addresses. Although 128 bits can express over 3.4×10^{38} possible combinations, the large address space of IPv6 has been designed to allow for multiple levels of subnetting and address allocation, from the Internet backbone to the individual subnets within an organization.

Even with all of the addresses currently assigned for use by hosts, plenty of addresses are available for future use. With a much larger number of available addresses, address-conservation techniques, such as the deployment of NATs, are no longer necessary.

Stateless and Stateful Address Configuration

To simplify host configuration, IPv6 supports both stateful address configuration (such as address configuration in the presence of a DHCP for IPv6, or DHCPv6, server) and stateless address configuration (such as address and routing configuration in the absence of a DHCPv6 server). With stateless address configuration, hosts on a link automatically configure themselves with IPv6 addresses for the link (called *link-local addresses*), with IPv6 transition addresses, with addresses derived from prefixes advertised by local routers, and local subnet and default routes.

Both stateless and stateful addressing can be used at the same time. Even in the absence of a router, hosts on the same link can automatically configure themselves with link-local addresses and communicate without manual configuration. Link-local addresses are autoconfigured within seconds, and communication with neighboring nodes on the link is possible immediately. In comparison, some IPv4 hosts using DHCP must wait a full minute before abandoning DHCP configuration and self-configuring an IPv4 address.

IPsec Header Support Required

Support for the IPsec headers is an IPv6 protocol suite requirement. This requirement provides a standards-based solution for network protection needs and promotes interoperability between different IPv6 implementations. IPsec consists of two types of extension headers and a protocol to negotiate security settings. The Authentication header (AH) provides data integrity, data authentication, and replay protection for the entire IPv6 packet (excluding fields in the IPv6 header that must change in transit). The Encapsulating Security Payload (ESP) header and trailer provide data integrity, data authentication, data confidentiality, and replay protection for the ESP-encapsulated payload. The protocol typically used to negotiate IPsec security settings for unicast communication is the Internet Key Exchange (IKE) protocol.

The requirement to process IPsec headers does not make IPv6 inherently more secure. IPv6 packets are not required to be protected with IPsec, and IPsec is not a requirement of an IPv6 deployment. Additionally, the IPv6 standards do not require an implementation to support any specific encryption methods, hashing methods, or negotiation protocol (such as IKE).

Better Support for Prioritized Delivery

New fields in the IPv6 header define how traffic is handled and identified. Traffic is prioritized by using a Traffic Class field, which specifies a DSCP value just like IPv4. A Flow Label field in the IPv6 header allows routers to identify and provide special handling for packets that belong to a flow (a series of packets between a source and destination). Because the traffic is identified in the IPv6 header, support for prioritized delivery can be achieved even when the packet payload is encrypted with IPsec and ESP.

New Protocol for Neighboring Node Interaction

The Neighbor Discovery protocol for IPv6 is a series of Internet Control Message Protocol for IPv6 (ICMPv6) messages that manages the interaction of neighboring nodes (nodes on the same link). Neighbor Discovery replaces and extends the combination of the Address Resolution Protocol (ARP) (broadcast-based), ICMPv4 Router Discovery, and ICMPv4 Redirect messages with efficient multicast and unicast Neighbor Discovery messages.

Extensibility

IPv6 can easily be extended for new features by adding extension headers after the IPv6 header. Unlike options in the IPv4 header, which can support only 40 bytes of options, the size of IPv6 extension headers is constrained only by the size of the IPv6 packet.

Comparison of IPv4 and IPv6

Table 1-1 highlights some of the key differences between IPv4 and IPv6.

TABLE 1-1 Differences Between IPv4 and IPv6

IPv4	IPv6
Source and destination addresses are 32 bits (4 bytes) in length.	Source and destination addresses are 128 bits (16 bytes) in length. For more information, see Chapter 3, “IPv6 Addressing.”
IPsec header support is optional.	IPsec header support is required. For more information, see Chapter 4, “The IPv6 Header.”
No identification of packet flow for prioritized delivery handling by routers is present within the IPv4 header.	Packet flow identification for prioritized delivery handling by routers is present within the IPv6 header using the Flow Label field. For more information, see Chapter 4.
Fragmentation is performed by the sending host and at routers, slowing router performance.	Fragmentation is performed only by the sending host. For more information, see Chapter 4.
Has no link-layer packet-size requirements and must be able to reassemble a 576-byte packet.	Link layer must support a 1,280-byte packet and be able to reassemble a 1,500-byte packet. For more information, see Chapter 4.
Header includes a checksum.	Header does not include a checksum. For more information, see Chapter 4.
Header includes options.	All optional data is moved to IPv6 extension headers. For more information, see Chapter 4.
ARP uses broadcast ARP Request frames to resolve an IPv4 address to a link-layer address.	ARP Request frames are replaced with multicast Neighbor Solicitation messages. For more information, see Chapter 6, “Neighbor Discovery.”
Internet Group Management Protocol (IGMP) is used to manage local subnet group membership.	IGMP is replaced with Multicast Listener Discovery (MLD) messages. For more information, see Chapter 7, “Multicast Listener Discovery and MLD Version 2.”

IPv4	IPv6
ICMP Router Discovery is used to determine the IPv4 address of the best default gateway and is optional.	ICMPv4 Router Discovery is replaced with ICMPv6 Router Solicitation and Router Advertisement messages, and it is required. For more information, see Chapter 6.
Broadcast addresses are used to send traffic to all nodes on a subnet.	There are no IPv6 broadcast addresses. Instead, a link-local scope all-nodes multicast address is used. For more information, see "Multicast IPv6 Addresses" in Chapter 3.
Must be configured either manually or through DHCP for IPv4.	Does not require manual configuration or DHCP for IPv6. For more information, see Chapter 8, "Address Autoconfiguration."
Uses host address (A) resource records in the Domain Name System (DNS) to map host names to IPv4 addresses.	Uses AAAA records in the DNS to map host names to IPv6 addresses. For more information, see Chapter 9, "IPv6 and Name Resolution."
Uses pointer (PTR) resource records in the IN-ADDR.ARPA DNS domain to map IPv4 addresses to host names.	Uses pointer (PTR) resource records in the IP6.ARPA DNS domain to map IPv6 addresses to host names. For more information, see Chapter 9.

IPv6 Terminology

The following list of common terms for network elements and concepts provides a foundation for subsequent chapters. Figure 1-3 shows an IPv6 network.

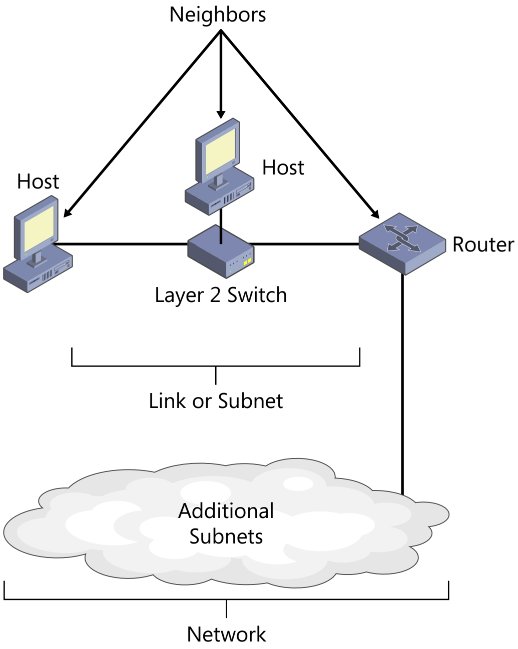


FIGURE 1-3 Elements of an IPv6 network.

IPv6 common terms and concepts are defined as follows:

- **Node** Any device that runs an implementation of IPv6. This includes routers and hosts.
- **Router** A node that can forward IPv6 packets not explicitly addressed to itself. On an IPv6 network, a router also typically advertises its presence and host configuration information.
- **Host** A node that cannot forward IPv6 packets not explicitly addressed to itself (a non-router). A host is typically the source and a destination of IPv6 traffic, and it silently discards traffic received that is not explicitly addressed to itself.
- **Upper-layer protocol** A protocol that uses IPv6 as its transport. Examples include Internet layer protocols such as ICMPv6 and Transport layer protocols such as TCP and UDP (but not Application layer protocols such as FTP and DNS, which use TCP and UDP as their transport).
- **Link** The set of network interfaces that are bounded by routers (or contains no routers) and that use the same 64-bit IPv6 unicast address prefix. Other terms for "link" are *subnet* and *network segment*. Many link-layer technologies are already defined for IPv6, including typical LAN technologies (such as Ethernet and Institute of Electrical and Electronics Engineers [IEEE] 802.11 wireless) and wide area network (WAN) technologies (such as the Point-to-Point Protocol [PPP] and Frame Relay). Additionally, IPv6 packets can be sent over logical links representing an IPv4 or IPv6 network, by encapsulating the IPv6 packet within an IPv4 or IPv6 header. For more information about LAN and WAN media support for IPv6, see Appendix E, "Link-Layer Support for IPv6."
- **Network** Two or more subnets connected by routers. Another term for network is *internetwork*.
- **Neighbors** Nodes connected to the same link. Neighbors in IPv6 have special significance because of IPv6 Neighbor Discovery, which can resolve neighbor link-layer addresses and detect and monitor neighbor reachability.
- **Interface** The representation of an attachment to a physical or logical link. An example of a physical interface is a network adapter. An example of a logical interface is a "tunnel" interface that is used to send IPv6 packets across an IPv4 network by encapsulating the IPv6 packet inside an IPv4 header.
- **Address** An identifier that can be used as the source or destination of IPv6 packets that is assigned at the IPv6 layer to an interface or set of interfaces.
- **Packet** The protocol data unit (PDU) that exists at the IPv6 layer and is composed of an IPv6 header and payload.
- **Link MTU** The maximum transmission unit (MTU)—the number of bytes in the largest IPv6 packet—that can be sent on a link. Because the maximum frame size includes the link-layer medium headers and trailers, the link MTU is not the same as the maximum frame size of the link. The link MTU is the same as the maximum payload size of the link-layer technology. For

example, for Ethernet using Ethernet II encapsulation, the maximum Ethernet frame payload size is 1,500 bytes. Therefore, the link MTU is 1500. For a link with multiple link-layer technologies (for example, a bridged link), the link MTU is the smallest link MTU of all the link-layer technologies present on the link.

- **Path MTU** The maximum-sized IPv6 packet that can be sent without performing host fragmentation between a source and destination over a path in an IPv6 network. The path MTU is typically the smallest link MTU of all the links in the path.

Figure 1-4 shows an IPv6-capable organization network and its relation to the IPv4 and IPv6 Internets.

A *site* is an autonomously operating IP-based network that is connected to the IPv6 Internet. Network architects and administrators within the site determine the addressing plan and routing policy for the organization network. An organization can have multiple sites. The actual connection to the IPv6 Internet can be either of the following types:

- **Direct** The connection to the IPv6 Internet uses a wide area network link (such as Frame Relay or T-Carrier) and connects to an IPv6-capable ISP (shown in Figure 1-4).
- **Tunneled** The connection to the IPv6 Internet uses an IPv6-over-IPv4 tunnel and connects to an IPv6 tunneling router.

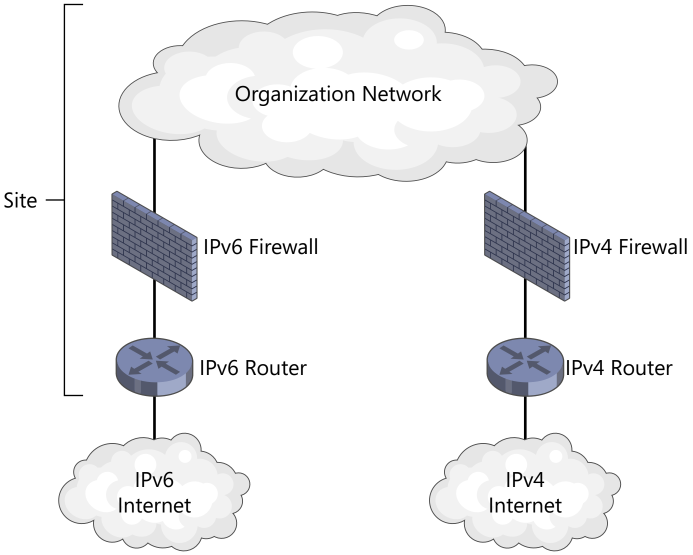


FIGURE 1-4 An IPv6-capable organization network and the IPv4 and IPv6 Internets.

For more information about how sites use IPv6 address prefixes, see Chapter 3.

The Case for IPv6 Deployment

Although the IPv6 protocol offers a host of technological advances and innovations, its use must still be justified from a business perspective and deployed by information technology (IT) staff in end-user organizations and ISPs. The deployment of native IPv6 support in the network infrastructure involves the planning and design of coexistence and migration strategies and the installation and maintenance of hardware and software. The resulting combination of IT staff, hardware and software resources, and time required for the transition makes the decision to deploy native IPv6 support a significant one, especially in light of other technology initiatives that might have higher visibility or better short-term benefits.

One must consider, however, that the Internet, once a pseudo-private network connecting educational institutions and United States government agencies, has become an indispensable worldwide communications medium that is an integral part of increased efficiency and productivity for commercial organizations and individuals, and it is now a major component of the world's economic engine. *Its growth must continue*.

To continue the growth of the Internet and private intranets, IPv4 must eventually be replaced. The sooner IPv4 is replaced, the sooner the benefits of its replacement protocol are realized. The following sections present the key technological and business benefits in the case to deploy IPv6.

IPv6 Solves the Address Depletion Problem

With the explosion in the popularity of the Internet has come the introduction of commerce-related activities that can now be done over the Internet by an ever-increasing number of devices. With IPv4, the number of public addresses available to new devices is limited and shrinking. IPv4 cannot continue to scale and provide global connectivity to all of the planned Internet-capable devices to be produced and connected in the next 10 years. Although these devices can be assigned private addresses, address and port translation introduces complexity to the devices that need to perform server, listening, or peer functionality. IPv6 solves the IPv4 public address depletion problem by providing an address space to last well into the twenty-first century.

The business benefit of moving to IPv6 is that mobile cell phones, personal data assistants (PDAs), automobiles, appliances, and even people can be assigned multiple globally reachable addresses. The growth of the devices connected to the Internet and the software that these devices run can proceed without restraint and without the complexity and cost of having to operate behind NATs.

IPv6 Solves the Disjoint Address Space Problem

With IPv4, there are typically two different addressing schemes for the home and the enterprise network. In the home, an Internet gateway device (IGD) is assigned a single public IPv4 address and the IGD assigns private IPv4 addresses to the hosts on the home network. An enterprise might have multiple public IPv4 addresses or a public address range and either assign public, private, or both types of addresses within the enterprise's intranet.

However, the public and private IPv4 address spaces are disjoint; that is, they do not provide symmetric reachability at the Network layer. Symmetric reachability exists when packets can be sent to and received from an arbitrary destination. With IPv4, there is no single addressing scheme that is applied to both networks that allows seamless connectivity. Connectivity between disjoint networks requires intermediate devices such as NATs or proxy servers. With IPv6, both homes and enterprises will be assigned global address prefixes and can seamlessly connect, subject to security restrictions such as firewall filtering and authenticated communication.

IPv6 Solves the International Address Allocation Problem

The Internet was principally a creation of educational institutions and government agencies of the United States of America. In the early days of the Internet, connected sites in the United States received IPv4 address prefixes without regard to summarizability or need. The historical result of this address allocation practice is that the United States has a disproportionate number of public IPv4 addresses.

With IPv6, public address prefixes are assigned to regional Internet registries, which, in turn, assign address prefixes to other ISPs and organizations based on justified need. This new address allocation practice ensures that address prefixes will be distributed globally based on regional connectivity needs rather than by historical origin. This makes the Internet more of a truly global resource rather than a United States–centric one. The business benefit to organizations across the globe is that they can rely on having available public IPv6 address space, without the current cost of obtaining IPv4 public address prefixes from their ISP or other source.

IPv6 Restores End-to-End Communication

With IPv4 NATs, there is a technical barrier for applications that rely on listening or peer-based connectivity because of the need for the communicating peers to discover and advertise their public IPv4 addresses and ports. The workarounds for the translation barrier might also require the deployment of echo or rendezvous servers on the Internet to provide public address and port configuration information.

With IPv6, NATs are no longer necessary to conserve public address space, and the problems associated with mapping addresses and ports disappear for developers of applications and gateways. More importantly, end-to-end communication is restored between hosts on the Internet by using addresses in packets that do not change in transit. This functional restoration has immense value when one considers the emergence of peer-to-peer telephony, video, and other real-time collaboration technologies for personal communications, and that the next wave of devices that are connected to the Internet include many types of peer-to-peer devices, such as mobile phones and gaming consoles.

By restoring global addressing and end-to-end connectivity, IPv6 has no barrier to new applications that are based on ad hoc connectivity and peer-based communication. Additionally, there is no need to deploy echo servers on the Internet. The business benefit for software developers is easier development of peer-based applications to share information, music, and media or to collaborate without having to work around the NAT translation barrier. An additional benefit to global addressing and end-to-end connectivity is that users can remotely access computers on their home networks rather than having to use intermediate hosts on the Internet.

IPv6 Uses Scoped Addresses and Address Selection

Unlike IPv4 addresses, IPv6 addresses have a *scope*, or a defined area of the network over which they are unique and relevant. For example, IPv6 has a global address that is equivalent to the IPv4 public address and a unique local address that is roughly equivalent to the IPv4 private address. Typical IPv4 routers do not distinguish a public address from a private address and will forward a privately addressed packet on the Internet. An IPv6 router, on the other hand, is aware of the scope of IPv6 addresses and will never forward a packet over an interface that does not have the correct scope.

There are different types of IPv6 addresses with different scopes. When multiple IPv6 addresses are returned in a DNS name query, the sending node must be able to distinguish their types and, when initiating communication, use a pair (source address and destination address) that is matched in scope and that is the most appropriate pair to use. For example, for a source and a destination that have been assigned both global (public) and link-local addresses, a sending IPv6 host would never use a global destination with a link-local source. IPv6 sending hosts include the address selection logic that is needed to decide which pair of addresses to use in communication. Moreover, the address selection rules are configurable. This allows you to configure multiple addressing infrastructures within an organization. Regardless of how many types of addressing infrastructures are in place, the sending host always chooses the "best" set of addresses. In comparison, IPv4 nodes have no awareness of address types and can send traffic to a public address from a private address.

The benefit of scoped addresses is that by using the set of addresses of the smallest scope, your traffic does not travel beyond the scope for the address, exposing your network traffic to fewer possible malicious hosts. The benefit of standardized and built-in address selection algorithms for ISVs is that they do not have to develop and test their own address selection schemes and can rely on the sorted list of addresses, resulting in lower software development costs.

IPv6 Has More Efficient Forwarding

IPv6 is a streamlined version of IPv4. Excluding prioritized delivery traffic, IPv6 has fewer fields to process and fewer decisions to make in forwarding an IPv6 packet. Unlike IPv4, the IPv6 header is a fixed size (40 bytes), which allows routers to process IPv6 packets faster. Additionally, the hierarchical and summarizable addressing structure of IPv6 global addresses means that there are fewer routes to analyze in the routing tables of organization and Internet backbone routers. The consequence is traffic that can be forwarded at higher data rates, resulting in higher performance for tomorrow's high-bandwidth applications that use multiple data types.

IPv6 Has Support for Security and Mobility

IPv6 has been designed to support security (IPsec, with AH and ESP header support required) and mobility (optionally, Mobile IPv6). Although one could argue that these features are available for IPv4, they are available on IPv4 as extensions and therefore have architectural or connectivity limitations that might not have been present if they had been part of the original IPv4 design. It is always better to design features in rather than bolt them on. Designing IPv6 with security and mobility in mind has resulted in an implementation that is a defined standard, has fewer limitations, and is more robust and scalable to handle the current and future communication needs of the users of the Internet.

The business benefit of requiring support for IPsec and using a single, global address space is that IPv6 can protect packets from end to end across the entire IPv6 Internet. Unlike IPsec on the IPv4 Internet, which must be modified and has limited functionality when the endpoints are behind NATs, IPsec on the IPv6 Internet is fully functional between any two endpoints.

Testing for Understanding

To test your understanding of IPv6, answer the following questions. See Appendix B, “Testing for Understanding Answers,” to check your answers.

1. What are the problems with IPv4 on today’s Internet?
2. How does IPv6 solve these problems?
3. How does IPv6 provide better prioritized delivery support?
4. Describe at least three ways in which IPv6 is more efficient than IPv4.
5. Explain how NATs prevent peer-to-peer applications from working properly.
6. What are the key technical benefits of deploying IPv6 now?
7. What are the key business benefits of deploying IPv6 now?

CHAPTER 4

The IPv6 Header

At the end of this chapter, you should be able to do the following:

- Describe the structure of an IPv6 packet.
- List and describe the fields in the IPv4 header.
- List and describe the fields in the IPv6 header.
- Compare and contrast the fields in the IPv4 header with the fields in the IPv6 header.
- List and describe each IPv6 extension header.
- Describe the IPv6 maximum transmission unit (MTU).
- Describe the new pseudo-header used for upper-layer checksums.

Structure of an IPv6 Packet

An Internet Protocol version 6 (IPv6) packet consists of an IPv6 header, extension headers, and an upper-layer protocol data unit. Figure 4-1 shows the structure of an IPv6 packet.

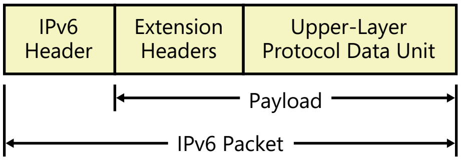


FIGURE 4-1 The structure of an IPv6 packet.

The components of an IPv6 packet are the following:

- **IPv6 Header** The IPv6 header is always present and is a fixed size of 40 bytes. The fields in the IPv6 header are described in the "IPv6 Header" section in this chapter.
- **Extension Headers** Zero or more extension headers can be present and are of varying lengths. If extension headers are present, a Next Header field in the IPv6 header indicates the first extension header. Within each extension header is another Next Header field, indicating the next extension header. The last extension header indicates the header for the upper-layer protocol—such as Transmission Control Protocol (TCP), User Datagram Protocol (UDP), or

Internet Control Message Protocol for version 6 (ICMPv6)—contained within the upper-layer protocol data unit.

The IPv6 header and extension headers replace the existing IPv4 header and its options. The new extension header format allows IPv6 to be enhanced to support future needs and capabilities. Unlike options in the IPv4 header, IPv6 extension headers have no maximum size and can expand to accommodate all the extension data needed for IPv6 communication. IPv6 extension headers are described in the "IPv6 Extension Headers" section in this chapter.

- **Upper-Layer Protocol Data Unit** The upper-layer protocol data unit (PDU) typically consists of an upper-layer protocol header and its payload (for example, an ICMPv6 message, a TCP segment, or a UDP message).

  The IPv6 packet payload is the combination of the IPv6 extension headers and the upper-layer PDU. Normally, it can be up to 65,535 bytes long. IPv6 packets with payloads larger than 65,535 bytes in length, known as *jumbograms*, can also be sent.

IPv4 Header

Before examining the IPv6 header, you might find it helpful, for contrasting purposes, to review the IPv4 header shown in Figure 4-2.

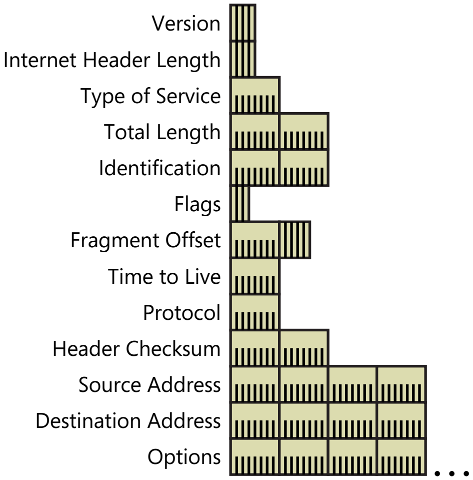


FIGURE 4-2 The structure of the IPv4 header.

Following is a list of the fields in the IPv4 header:

- **Version** The Version field indicates the version of IP and is set to 4. The size of this field is 4 bits.

- **Internet Header Length** The Internet Header Length (IHL) field indicates the number of 4-byte blocks in the IPv4 header. The size of this field is 4 bits. Because an IPv4 header is a minimum of 20 bytes in size, the smallest value of the IHL field is 5. IPv4 options can extend the minimum IPv4 header size in increments of 4 bytes. If an IPv4 option is not an integral multiple of 4 bytes in length, the remaining bytes are padded with padding options, making the entire IPv4 header an integral multiple of 4 bytes. With a maximum IHL value of 0xF, the maximum size of the IPv4 header, including options, is 60 bytes (15 × 4).

- **Type of Service** The Type of Service field indicates the desired service expected by this packet for delivery through routers across the IPv4 internetwork. The size of this field is 8 bits, including bits originally defined in RFC 791 for precedence, delay, throughput, reliability, and cost characteristics. RFC 2474 provides the modern definition as the Differentiated Services (DS) field. The high-order 6 bits of the DS field comprise the DS Code Point (DSCP) field. The DSCP field allows devices in a network to mark, unmark, and classify packets for forwarding. This is usually done based on the needs of an application. For example, Voice over IP and other real-time packets take precedence over e-mail in congested areas of the network. This is commonly referred to as Quality of Service (QoS). The low-order 2 bits of the Type of Service field are used for Explicit Congestion Notification (ECN), as defined in RFC 3168.

- **Total Length** The Total Length field indicates the total length of the IPv4 packet (IPv4 header + IPv4 payload) and does not include link-layer framing. The size of this field is 16 bits, which can indicate an IPv4 packet that is up to 65,535 bytes long.

- **Identification** The Identification field identifies this specific IPv4 packet. The size of this field is 16 bits. The Identification field is selected by the source node of the IPv4 packet. If the IPv4 packet is fragmented, all the fragments retain the Identification field value so that the destination node can group the fragments for reassembly.

- **Flags** The Flags field identifies flags for the fragmentation process. The size of this field is 3 bits; however, only 2 bits are defined for current use. There are two flags—one to indicate whether the IPv4 packet can be fragmented and another to indicate whether more fragments follow the current fragment.

- **Fragment Offset** The Fragment Offset field indicates the position of the fragment relative to the beginning of the original IPv4 payload. The size of this field is 13 bits.

- **Time-to-Live** The Time-to-Live (TTL) field indicates the maximum number of links on which an IPv4 packet can travel before being discarded. The size of this field is 8 bits. The TTL field was originally defined as a time count for the number of seconds the packet could exist on the network. An IPv4 router determined the length of time required (in seconds) to forward the IPv4 packet and decremented the TTL accordingly. Modern routers almost always forward an IPv4 packet in less than a second, and they are required by RFC 791 to decrement the TTL by at least one. Therefore, the TTL becomes a maximum link count with the value set by the sending node. When the TTL equals 0, an ICMPv4 Time Exceeded-Time to Live Exceeded in Transit message is sent to the source of the packet and the packet is discarded.

- **Protocol** The Protocol field identifies the upper-layer protocol. The size of this field is 8 bits. For example, a value of 6 in this field identifies TCP as the upper-layer protocol, a decimal value of 17 identifies UDP, and a value of 1 identifies ICMPv4. The Protocol field is used to identify the upper-layer protocol that is to receive the IPv4 packet payload.
- **Header Checksum** The Header Checksum field provides a checksum on the IPv4 header only. The size of this field is 16 bits. The IPv4 payload is not included in the checksum calculation because the IPv4 payload usually contains its own checksum. Each IPv4 node that receives IPv4 packets verifies the IPv4 header checksum and silently discards the IPv4 packet if checksum verification fails. When a router forwards an IPv4 packet, it must decrement the TTL. Therefore, the Header Checksum value is recomputed at each hop between source and destination.
- **Source Address** The Source Address field stores the IPv4 address of the originating host. The size of this field is 32 bits.
- **Destination Address** The Destination Address field stores the IPv4 address of an intermediate destination (in the case of source routing) or the destination host. The size of this field is 32 bits.
- **Options** The Options field stores one or more IPv4 options. The size of this field is a multiple of 32 bits (4 bytes). If an IPv4 option does not use all 32 bits, padding options must be added so that the IPv4 header is an integral number of 4-byte blocks that can be indicated by the IHL field.

IPv6 Header

The IPv6 header is a streamlined version of the IPv4 header. It eliminates fields that are either unneeded or rarely used, and it adds a field that provides better support for real-time traffic. Figure 4-3 shows the structure of the IPv6 header as described in RFC 2460.

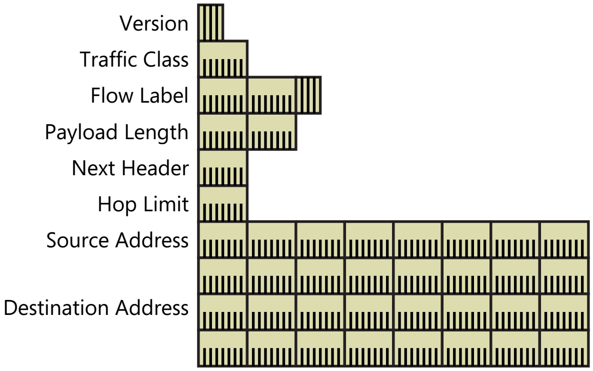


FIGURE 4-3 The structure of the IPv6 header.

Following is a list of the fields in the IPv6 header:

- **Version** The Version field indicates the version of IP and is set to 6. The size of this field is 4 bits. While the purpose of the Version field is defined in the same way for both IPv4 and IPv6, its value is not used to pass the packet to an IPv4 or IPv6 protocol layer. This identification is done through a protocol identification field in the link-layer header. For example, a common link-layer encapsulation for Ethernet, called Ethernet II, uses a 16-bit EtherType field to identify the Ethernet frame payload. For IPv4 packets, the EtherType field is set to 0x800. For IPv6 packets, the EtherType field is set to 0x86DD. Thus, the determination of the protocol of the Ethernet payload occurs before the packet is passed to the appropriate protocol layer.
- **Traffic Class** The Traffic Class field indicates the IPv6 packet's class or priority. The size of this field is 8 bits. This field provides functionality similar to the IPv4 Type of Service field. Like the Type of Service field in the IPv4 header, the first 6 bits of the Traffic Class field represent the DSCP field as defined in RFC 2474, and the last 2 bits are used for ECN as defined in RFC 3168.
- **Flow Label** The Flow Label field indicates that this packet belongs to a specific sequence of packets between a source and destination, requiring special handling by intermediate IPv6 routers. The size of this field is 20 bits. The flow label is used for prioritized delivery, such as delivery needed by real-time data (voice and video). For default router handling, the Flow Label field is set to 0. To distinguish a given flow, an intermediate router can use the packet's source address, destination address, and flow label. Therefore, there can be multiple flows between a source and destination, as distinguished by separate non-zero flow labels. The details of the use of the Flow Label field are described in RFC 3697.
- **Payload Length** The Payload Length field indicates the length of the IPv6 payload. The size of this field is 16 bits. The Payload Length field includes the extension headers and the upper-layer PDU. With 16 bits, an IPv6 payload of up to 65,535 bytes can be indicated. For payload lengths greater than 65,535 bytes, the Payload Length field is set to 0 and the Jumbo Payload option is used in the Hop-by-Hop Options extension header, which is described in the "Hop-by-Hop Options Header" section in this chapter.
- **Next Header** The Next Header field indicates either the type of the first extension header (if present) or the protocol in the upper-layer PDU (such as TCP, UDP, or ICMPv6). The size of this field is 8 bits. When indicating an upper-layer protocol, the Next Header field uses the same values that are used in the IPv4 Protocol field.
- **Hop Limit** The Hop Limit field indicates the maximum number of links over which the IPv6 packet can travel before being discarded. The size of this field is 8 bits. The Hop Limit field is similar to the IPv4 TTL field except that there is no historical relation to the amount of time (in seconds) that the packet is queued at the router. When Hop Limit equals 0 at a router, the router sends an ICMPv6 Time Exceeded-Hop Limit Exceeded in Transit message to the source and discards the packet.
- **Source Address** The Source Address field indicates the IPv6 address of the originating host. The size of this field is 128 bits.

- **Destination Address** The Destination Address field indicates the IPv6 address of the current destination node. The size of this field is 128 bits. In most cases, the Destination Address field is set to the final destination address. However, if a Routing extension header is present, the Destination Address field might be set to the address of the next intermediate destination.

Network Monitor Capture

Here is an example of an IPv6 header, as displayed by Network Monitor 3.4 (capture 04_01 in the companion content for this book):

```
 Frame:
+ Ethernet: Etype = IPv6
- Ipv6: Next Protocol = ICMPv6, Payload Length = 40
  - Versions: IPv6, Internet Protocol, DSCP 0
     Version:  (0110............................) IPv6, Internet Protocol, 6(0x6)
     DSCP:     (....000000......................) Differentiated services codepoint 0
     ECT:      (..........0.....................) ECN-Capable Transport not set
     CE:       (...........0....................) ECN-CE not set
     FlowLabel: (............00000000000000000000) 0
    PayloadLength: 40 (0x28)
    NextProtocol: ICMPv6, 58(0x3a)
    HopLimit: 128 (0x80)
    SourceAddress: FE80:0:0:0:260:97FF:FE02:6E8F
    DestinationAddress: FE80:0:0:0:260:97FF:FE02:6D3D
+ Icmpv6: Echo request, ID = 0x0, Seq = 0x18
```

This ICMPv6 Echo Request packet uses the default Traffic Class and Flow Label and a Hop Limit of 128, and it is sent between two hosts using link-local addresses.

Values of the Next Header Field

Table 4-1 lists typical values of the Next Header field for an IPv6 header or an IPv6 extension header. Each of the IPv6 extension headers is covered later in the chapter.

TABLE 4-1 Typical Values of the Next Header Field

Value (Decimal)	Header
0	Hop-by-Hop Options header
6	TCP
17	UDP
41	Encapsulated IPv6 header
43	Routing header
44	Fragment header
50	Encapsulating Security Payload header
51	Authentication header
58	ICMPv6
59	No next header
60	Destination Options header

For the most current list of the reserved values for the IPv4 Protocol and IPv6 Next Header fields, see *http://www.iana.org/assignments/protocol-numbers*.

In looking at the value of the Next Header field to indicate no next header, it would seem to make more sense to set its value to 0, rather than 59. However, the designers of IPv6 wanted to optimize the processing of IPv6 packets at intermediate routers. The only extension header that must be processed at every intermediate router is the Hop-by-Hop Options header. To optimize the test of whether the Hop-by-Hop Options header is present, its Next Header value is set to 0. In router hardware, it is easier to test for a value of 0 than to test for a value of 59.

Comparing the IPv4 and IPv6 Headers

In comparing the IPv4 and IPv6 headers, you can see the following:

- The number of fields has dropped from 12 (including options) in the IPv4 header to 8 in the IPv6 header.
- The number of fields that must be processed by an intermediate router has dropped from 6 to 4, making the forwarding of normal IPv6 packets more efficient.
- Seldom-used fields, such as fields supporting fragmentation and options in the IPv4 header, have been moved to extension headers in the IPv6 header.
- The size of the IPv6 header has doubled from 20 bytes for a minimum-sized IPv4 header to 40 bytes. However, the new IPv6 header contains source and destination addresses that are four times longer than IPv4 source and destination addresses.

Table 4-2 lists the individual differences between the IPv4 and IPv6 header fields.

TABLE 4-2 IPv4 Header Fields and Corresponding IPv6 Equivalents

IPv4 Header Field	IPv6 Header Field
Version	Same field but with a different version number.
Internet Header Length	Removed in IPv6. IPv6 does not include a Header Length field because the IPv6 header is always a fixed length of 40 bytes. Each extension header is either a fixed length or indicates its own length.
Type of Service	Replaced by the IPv6 Traffic Class field.
Total Length	Replaced by the IPv6 Payload Length field, which indicates only the size of the payload.
Identification Flags Fragment Offset	Removed in IPv6. Fragmentation information is not included in the IPv6 header. It is contained in a Fragment extension header.
Time-to-Live	Replaced by the IPv6 Hop Limit field.
Protocol	Replaced by the IPv6 Next Header field.
Header Checksum	Removed in IPv6. The link layer has a checksum that performs bit-level error detection for the entire IPv6 packet.
Source Address	The field is the same except that IPv6 addresses are 128 bits in length.
Destination Address	The field is the same except that IPv6 addresses are 128 bits in length.
Options	Removed in IPv6. IPv6 extension headers replace IPv4 options.

The one new field in the IPv6 header that is not included in the IPv4 header is the Flow Label field.

The result of the new IPv6 header is a reduction in the critical router loop, which is the set of instructions that must be executed to determine how to forward a packet. To forward a normal IPv4 packet, a router typically performs the following in its critical router loop:

1. Verify the Header Checksum field by performing its own checksum calculation and comparing its result with the result stored in the IPv4 header. Although this step is required by RFC 1812, modern high-speed routers commonly skip it.
2. Verify the value of the Version field. Although this step is not required by RFC 791 or 1812, performing this step saves network bandwidth because a packet containing an invalid version number is not propagated across the IPv4 internetwork only to be discarded by the destination node.
3. Decrement the value of the TTL field. If its new value is less than 1, send an ICMPv4 Time Exceeded-Time to Live Exceeded in Transit message to the source of the packet and then discard the packet. If not, place the new value in the TTL field.
4. Check for the presence of IPv4 header options. If present, process them.
5. Use the value of the Destination Address field and the contents of the local routing table to determine a forwarding interface and a next-hop IPv4 address. If a route is not found, send an ICMPv4 Destination Unreachable-Host Unreachable message to the source of the packet and discard the packet.
6. If the IPv4 MTU of the forwarding interface is less than the value of the Total Length field and the Don't Fragment (DF) flag is set to 0, perform IPv4 fragmentation. If the MTU of the forwarding interface is less than the value of the Total Length field and the DF flag is set to 1, send an ICMPv4 Destination Unreachable-Fragmentation Needed and DF Set message to the source of the packet and discard the packet.
7. Recalculate the new header checksum, and place its new value in the Header Checksum field.
8. Forward the packet by using the appropriate forwarding interface.

Note This critical router loop for IPv4 routers is a simplified list of items that an IPv4 router typically performs when forwarding. This list is not meant to imply any specific implementation nor an optimized order in which to process IPv4 packets for forwarding.

To forward a normal IPv6 packet, a router typically performs the following steps in its critical router loop:

1. Verify the value of the Version field. Although this step is not required by RFC 2460, performing it saves network bandwidth because a packet containing an invalid version number is not propagated across the IPv6 internetwork only to be discarded by the destination node.

2. Decrement the value of the Hop Limit field. If its new value is less than 1, send an ICMPv6 Time Exceeded-Hop Limit Exceeded in Transit message to the source of the packet and discard the packet. If not, place the new value in the Hop Limit field.
3. Check the Next Header field for a value of 0. If it is 0, process the Hop-by-Hop Options header.
4. Use the value of the Destination Address field and the contents of the local routing table to determine a forwarding interface and a next-hop IPv6 address. If a route is not found, send an ICMPv6 Destination Unreachable-No Route To Destination message to the source of the packet and then discard the packet.
5. If the link MTU of the forwarding interface is less than 40 plus the value of the Payload Length field, send an ICMPv6 Packet Too Big message to the source of the packet and discard the packet.
6. Forward the packet by using the appropriate forwarding interface.

> **Note** This critical router loop for IPv6 routers is a simplified list of items that an IPv6 router typically performs when forwarding. This list is not meant to imply any specific implementation nor an optimized order in which to process packets for forwarding.

As you can see, the process to forward an IPv6 packet is much simpler than for an IPv4 packet because it does not have to verify and recalculate a header checksum, perform fragmentation, or process options not intended for the router.

IPv6 Extension Headers

The IPv4 header includes all options. Therefore, each intermediate router must check for their existence and process them when present. This can cause performance degradation in the forwarding of IPv4 packets. With IPv6, delivery and forwarding options are moved to extension headers. The only extension header that must be processed at each intermediate router is the Hop-by-Hop Options extension header. This increases IPv6 header processing speed and improves the performance of forwarding IPv6 packets.

RFC 2460 specifies that the following IPv6 extension headers must be supported by all IPv6 nodes:

- Hop-by-Hop Options header
- Destination Options header
- Routing header
- Fragment header
- Authentication header
- Encapsulating Security Payload header

With the exception of the Authentication header and Encapsulating Security Payload header, all the IPv6 extension headers in the preceding list are defined in RFC 2460.

In a typical IPv6 packet, no extension headers are present. If special handling is required by either intermediate routers or the destination, the sending host adds one or more extension headers.

Each extension header must fall on a 64-bit (8-byte) boundary. Extension headers of a fixed size must be an integral multiple of 8 bytes. Extension headers of variable size contain a Header Extension Length field and must use padding as needed to ensure that their size is an integral multiple of 8 bytes.

The Next Header field in the IPv6 header and zero or more extension headers form a chain of pointers. Each pointer indicates the type of header that comes after the immediate header until the upper-layer protocol is ultimately identified. Figure 4-4 shows the chain of pointers formed by the Next Header field for various IPv6 packets.

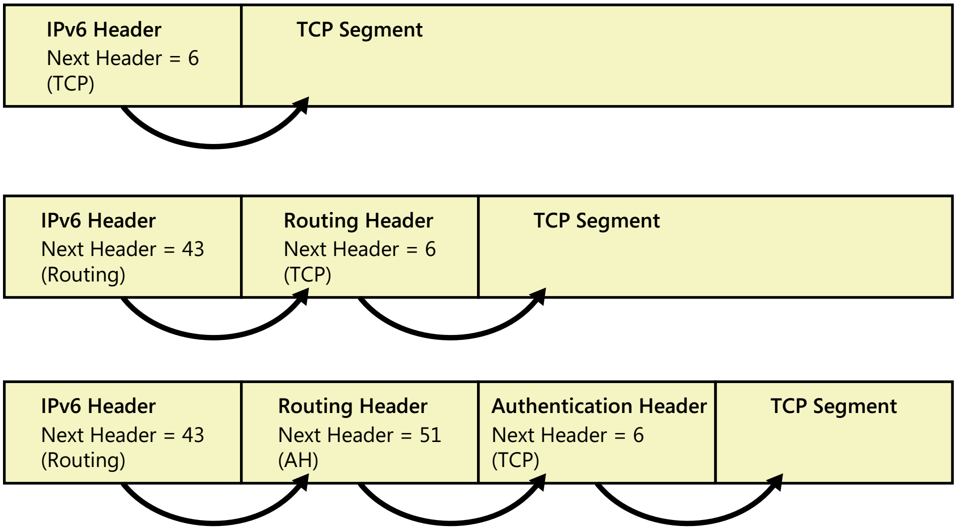


FIGURE 4-4 The chain of pointers formed by the Next Header field.

If an extension header contains an unrecognized or improper value of the Next Header field, the node discards the packet and sends an ICMP Parameter Problem-Unrecognized Next Header Type Encountered message to the packet source. An example of an improper value for any extension header is 0, because the Hop-by-Hop Options header must always be immediately after the IPv6 header.

Extension Headers Order

Extension headers are processed in the order in which they are present. Because the only extension header that is processed by every node on the path is the Hop-by-Hop Options header, it must be first. Similar rules apply for other extension headers. In RFC 2460, it is recommended that extension headers be placed after the IPv6 header in the following order:

1. Hop-by-Hop Options header
2. Destination Options header (for intermediate destinations when the Routing header is present)
3. Routing header
4. Fragment header
5. Authentication header
6. Encapsulating Security Payload header
7. Destination Options header (for the final destination)

Hop-by-Hop Options Header

The Hop-by-Hop Options header is used to specify delivery parameters at each hop on the path to the destination. It is identified by the value of 0 in the IPv6 header's Next Header field. Figure 4-5 shows the structure of the Hop-by-Hop Options header.

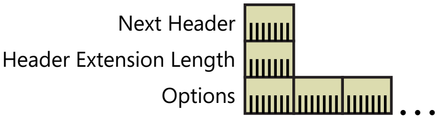


FIGURE 4-5 The structure of the Hop-by-Hop Options header.

The Hop-by-Hop Options header consists of a Next Header field, a Header Extension Length field, and an Options field that contains one or more options. The value of the Header Extension Length field is the number of 8-byte blocks in the Hop-by-Hop Options extension header, not including the first 8 bytes. Therefore, for an 8-byte Hop-by-Hop Options header, the value of the Header Extension Length field is 0. Padding options are used to ensure 8-byte boundaries.

An IPv6 Router Optimization

The interpretation of the Header Extension Length field in the Hop-by-Hop Options header is another example of how the designers of IPv6 wanted to optimize processing of IPv6 packets at intermediate routers. For packets with a Hop-by-Hop Options header, one of the first operations is to determine the size of the header. If the Header Extension Length field were defined to be the number of 8-byte blocks in the header, its minimum value would be 1 (the minimum-sized Hop-by-Hop Options header is 8 bytes long). To ensure robustness in an IPv6 forwarding implementation, a field whose valid values begin at 1 has to be checked for the invalid value of 0 before additional processing can be done.

With the current definition of the Header Extension Length field, 0 is a valid value and no testing of invalid values needs to be done. The number of bytes in the Hop-by-Hop Options header is calculated from the following formula: (header extension length + 1) × 8.

An option is a set of fields that either describes a specific characteristic of the packet delivery or provides padding. Options are sent in the Hop-by-Hop Options header and Destination Options header (described later in this chapter). Each option is encoded in the type-length-value (TLV) format that is commonly used in TCP/IP protocols. Figure 4-6 shows the structure of an option.

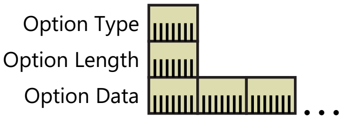


FIGURE 4-6 The structure of an option.

The Option Type field both identifies the option and determines the way it is handled by the processing node. The Option Length field indicates the number of bytes in the option, not including the Option Type and Option Length fields. The option data is the specific data associated with the option.

An option might have an alignment requirement to ensure that specific fields within the option fall on desired boundaries. For example, it is easier to process an IPv6 address if it falls on an 8-byte boundary. Alignment requirements are expressed by using the notation xn + y, indicating that the option must begin at a byte boundary equal to an integral multiple of x bytes plus y bytes from the start of the header. For example, the alignment requirement 4n + 2 indicates that the option must begin at a byte boundary of (an integral multiple of 4 bytes) + 2 bytes. In other words, the option must begin at the byte boundary of 6, 10, 14, and so on, relative to the start of the Hop-by-Hop Options or Destination Options headers. To accommodate alignment requirements, padding typically appears before an option and between each option when multiple options are present.

Option Type Field

Within the Option Type field, the two high-order bits indicate how the option is handled when the node processing the option does not recognize the option type. Table 4-3 lists the defined values of these two bits and their purpose.

TABLE 4-3 Values of the Two High-Order Bits in the Option Type Field

Value (Binary)	Action Taken
00	Skip the option.
01	Silently discard the packet.
10	Discard the packet, and send an ICMPv6 Parameter Problem message to the sender if the Destination Address field in the IPv6 header is a unicast or multicast address.
11	Discard the packet, and send an ICMPv6 Parameter Problem message to the sender if the Destination Address field in the IPv6 header is not a multicast address.

The third-highest-order bit of the Option Type indicates whether the option data can change (= 1) or not change (= 0) in the path to the destination.

Pad1 Option

The Pad1 option is defined in RFC 2460. It is used to insert a single byte of padding so that the Hop-by-Hop Options or Destination Options headers fall on 8-byte boundaries and to accommodate the alignment requirements of options. The Pad1 option has no alignment requirements. Figure 4-7 shows the Pad1 option.

Option Type
= 0

FIGURE 4-7 The structure of the Pad1 option.

The Pad1 option consists of a single byte; Option Type is set to 0, and it has no length or value fields. With Option Type set to 0, the option is skipped if not recognized, and it is not allowed to change in transit.

PadN Option

The PadN option is defined in RFC 2460. It is used to insert two or more bytes of padding so that the Hop-by-Hop Options or Destination Options headers fall on 8-byte boundaries and to accommodate the alignment requirements of options. The PadN option has no alignment requirements. Figure 4-8 shows the PadN option.

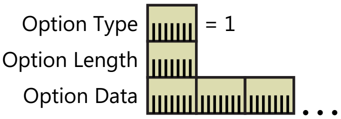


FIGURE 4-8 The structure of the PadN option.

The PadN option consists of the Option Type field (set to 1), the Length field (set to the number of padding bytes present), and 0 or more bytes of padding. With the Option Type field set to 1, the option is skipped if not recognized, and it is not allowed to change in transit.

Jumbo Payload Option

The Jumbo Payload option is defined in RFC 2675. It is used to indicate a payload size that is greater than 65,535 bytes. The Jumbo Payload option has the alignment requirement of 4n + 2. Figure 4-9 shows the Jumbo Payload option.

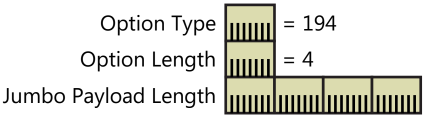


FIGURE 4-9 The structure of the Jumbo Payload option.

With the Jumbo Payload option, the Payload Length field in the IPv6 header no longer indicates the size of the IPv6 packet payload. Instead, the Jumbo Payload Length field in the Jumbo Payload option indicates the size, in bytes, of the IPv6 packet payload. With a 32-bit Jumbo Payload Length field, payload sizes of up to 4,294,967,295 bytes can be indicated. An IPv6 packet with a payload size greater than 65,535 bytes is known as a *jumbogram*. With the Option Type field set to 194 (0xC2 hexadecimal, binary 11000010), the packet is discarded and an ICMPv6 Parameter Problem message is sent if the option is not recognized and the destination address is not a multicast address; and the option is not allowed to change in transit.

The IPv6 protocol in Windows Server 2012, Windows Server 2008 R2, Windows Server 2008, Windows 8, Windows 7, and Windows Vista supports incoming jumbograms at the IPv6 layer. However, there is no support in UDP or TCP for sending or receiving jumbograms.

Router Alert Option

The Router Alert option (Option Type 5) is defined in RFC 2711 and is used to indicate to a router that the contents of the packet require additional processing. The Router Alert option has the alignment requirement of 2n + 0. Figure 4-10 shows the structure of the Router Alert option.

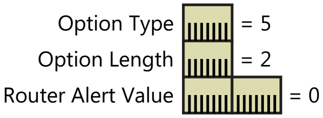


FIGURE 4-10 The structure of the Router Alert option.

The Router Alert option is used for Multicast Listener Discovery (MLD) and the Resource ReSerVation Protocol (RSVP). With the Option Type field set to 5, the option is skipped if not recognized, and it is not allowed to change in transit.

Network Monitor Capture

Here is an example of a Hop-by-Hop Options header as displayed by Network Monitor 3.4 (capture 04_02 in the companion content for this book):

```
 Frame:
+ Ethernet: Etype = IPv6
- Ipv6: Next Protocol = ICMPv6, Payload Length = 32
  + Versions: IPv6, Internet Protocol, DSCP 0
    PayloadLength: 32 (0x20)
    NextProtocol: HOPOPT, IPv6 Hop-by-Hop Option, 0(0)
    HopLimit: 1 (0x1)
    SourceAddress: FE80:0:0:0:2B0:D0FF:FEE9:4143
    DestinationAddress: FF02:0:0:0:0:1:FFE9:4143
  - HopbyHopHeader:
     NextHeader: ICMPv6
     ExtHdrLen: 0(8 bytes)
   - OptionRouterAlert:
    - OptionType: Router Alert
       Action:     (00......) Skip over this option
       C:          (..0.....) Option Data does not change en-route
       OptionType: (...00101) Router Alert
      OptDataLen: 2 bytes
      Value: Datagram contains a Multicast Listener Discovery message, 0 (0x0)
   - OptionPadN:
    - OptionType: PadN
       Action:     (00......) Skip over this option
       C:          (..0.....) Option Data does not change en-route
       OptionType: (...00001) PadN
      OptDataLen: 0 bytes
      OptionData: 0 bytes
+ Icmpv6: Multicast Listener Report
```

Notice the use of the Router Alert option (option type 5) and the PadN option (option type 1) to pad the entire Hop-by-Hop Options header to 8 bytes (1-byte Next Header field + 1-byte Option Length field + 4-byte Router Alert option + 2-byte PadN option).

Destination Options Header

The Destination Options header is used to specify packet delivery parameters for either intermediate destinations or the final destination. This header is identified by the value of 60 in the previous header's Next Header field. The Destination Options header has the same structure as the Hop-by-Hop Options header, as shown in Figure 4-11.

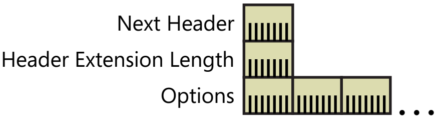


FIGURE 4-11 The structure of the Destination Options header.

The Destination Options header is used in two ways:

1. If a Routing header is present, it specifies delivery or processing options at each intermediate destination. In this case, the Destination Options header occurs before the Routing header.
2. If no Routing header is present or if this header occurs after the Routing header, this header specifies delivery or processing options at the final destination.

An example of a destination option is the Home Address destination option for Mobile IPv6.

Note The discussion of the Home Address destination option is here for example purposes only. Computers running Windows Server 2012, Windows Server 2008 R2, Windows Server 2008, Windows 8, Windows 7, or Windows Vista do not support Mobile IPv6 and act as correspondent nodes that are not Mobile IPv6–capable.

Home Address Option

The Home Address destination option (Option Type 201) is defined in RFC 6275 and is used to indicate the home address of the mobile node. The home address is an address assigned to the mobile node when it is attached to the home link and through which the mobile node is always reachable, regardless of its location on an IPv6 network. For information about when the Home Address option is sent, see Appendix G, "Mobile IPv6." The Home Address option has the alignment requirement of 8n + 6. Figure 4-12 shows the structure of the Home Address option.

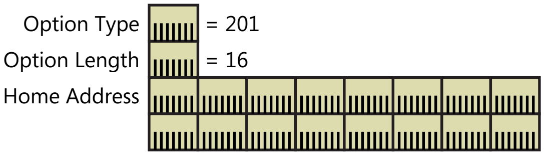


FIGURE 4-12 The structure of the Home Address option.

Following is a description of the fields in the Home Address option:

- **Option Type** With the Option Type field set to 201 (0xC9 hexadecimal, 11001001 binary), the packet is discarded and an ICMPv6 Parameter Problem message is sent if the option is not recognized and the destination address is not a multicast address; and the option is not allowed to change in transit.
- **Option Length** The Option Length field indicates the length of the option in bytes, not including the Option Type and Option Length fields. Because the only field past the Option Length field is the Home Address field to store an IPv6 address, the Option Length field is set to 16.
- **Home Address** The Home Address field indicates the home address of the mobile node. The size of this field is 128 bits.

For an example of the Home Address option in the Destination Options header, see the Network Monitor Capture 04_03 in the companion content for this book.

Summary of Option Types

Table 4-4 lists the different option types for options in Hop-by-Hop Options and Destination Options headers.

TABLE 4-4 Option Types

Option Type	Option and Where It Is Used	Alignment Requirement
0	Pad1 option: Hop-by-Hop and Destination Options headers	None
1	PadN option: Hop-by-Hop and Destination Options headers	None
194 (0xC2)	Jumbo Payload option: Hop-by-Hop Options header	4n + 2
5	Router Alert option: Hop-by-Hop Options header	2n + 0
201 (0xC9)	Home Address option: Destination Options header	8n + 6

Routing Header

IPv4 defines strict source routing, in which each intermediate destination must be only one hop away, and loose source routing, in which each intermediate destination can be one or more hops away. IPv6 source nodes can use the Routing header to specify a source route, which is a list of intermediate destinations for the packet to travel to on its path to the final destination. The Routing header is identified by the value of 43 in the previous header's Next Header field. Figure 4-13 shows the structure of the Routing header.

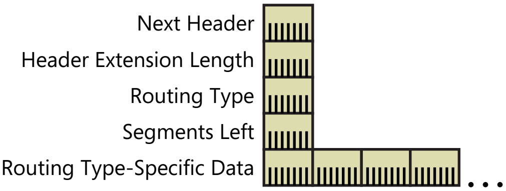


FIGURE 4-13 The structure of the Routing header.

The Routing header consists of a Next Header field, a Header Extension Length field (defined in the same way as the Hop-by-Hop Options extension header), a Routing Type field, a Segments Left field that indicates the number of intermediate destinations that are still to be visited, and routing type-specific data.

Note RFC 2460 also defines Routing Type 0, used for loose source routing. Because of security concerns, the Internet Engineering Task Force (IETF) has deprecated support for the Routing Type 0 header in RFC 5095. By default, IPv6 in Windows Server 2012, Windows Server 2008 R2, Windows 8, and Windows 7 will accept the packet if it has a complete routing header. If not, Windows drops the packet and sends an ICMP Parameter Problem-Erroneous Header Field Encountered message. This behavior is configurable with the *Set-NetIPv6Protocol –SourceRoutingBehavior* Windows PowerShell command (for Windows Server 2012 and Windows 8) and the *netsh interface ipv6 set global sourceroutingbehavior* command. Mobile IPv6 uses a Type 2 Routing header. For more information, see Appendix G.

Fragment Header

The Fragment header is used for IPv6 fragmentation and reassembly services. This header is identified by the value of 44 in the previous header's Next Header field. Figure 4-14 shows the structure of the Fragment header.

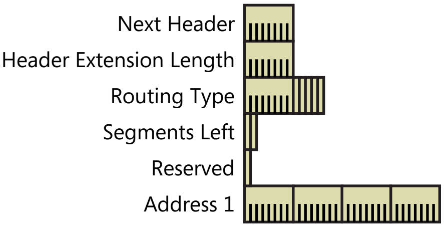


FIGURE 4-14 The structure of the Fragment header.

The Fragment header includes a Next Header field, a 13-bit Fragment Offset field, a More Fragments flag, and a 32-bit Identification field. The Fragment Offset, More Fragments flag, and Identification fields are used in the same way as the corresponding fields in the IPv4 header. Because the use of the Fragment Offset field is defined for 8-byte fragment blocks, the Fragment header cannot be used for jumbograms. The maximum number that can be expressed with the 13-bit Fragment Offset field is 8191. Therefore, Fragment Offset can be used to indicate only a fragment data starting position of up to 8191 × 8, or 65,528.

In IPv6, only source nodes can fragment payloads. If the payload submitted by the upper-layer protocol is larger than the link or path MTU, IPv6 fragments the payload at the source and uses the Fragment header to provide reassembly information. An IPv6 router will never fragment an IPv6 packet being forwarded.

Because the IPv6 internetwork does not transparently fragment payloads, data sent from applications that do not have an awareness of the destination path MTU cannot sense when data needing fragmentation by the source is discarded by IPv6 routers. This can be a problem for unicast or multicast traffic sent as a UDP message or other types of message streams that do not use TCP.

Differences in Fragmentation Fields

There are some subtle differences between the fragmentation fields in IPv4 and IPv6. In IPv4, the fragmentation flags are the three high-order bits of the 16-bit quantity composed of the combination of the fragmentation flags and the Fragment Offset field. In IPv6, the bits used for fragmentation flags are the three low-order bits of the 16-bit quantity composed of the combination of the fragmentation flags and the Fragment Offset field. In IPv4, the Identification field is 16 bits rather than 32 bits, as in IPv6, and in IPv6 there is no Don't Fragment flag. Because IPv6 routers never perform fragmentation, the Don't Fragment flag is always set to 1 for all IPv6 packets and therefore does not need to be included.

IPv6 Fragmentation Process

When an IPv6 packet is fragmented, it is initially divided into unfragmentable and fragmentable parts:

- The unfragmentable part of the original IPv6 packet must be processed by intermediate nodes between the fragmenting node and the destination. This part consists of the IPv6 header, the Hop-by-Hop Options header, the Destination Options header for intermediate destinations, and the Routing header.
- The fragmentable part of the original IPv6 packet must be processed only at the final destination node. This part consists of the Authentication header, the Encapsulating Security Payload header, the Destination Options header for the final destination, and the upper-layer PDU.

Next, the IPv6 fragment packets are formed. Each fragment packet consists of the unfragmentable part, a fragment header, and a portion of the fragmentable part. Figure 4-15 shows the IPv6 fragmentation process for a packet divided into three fragments.

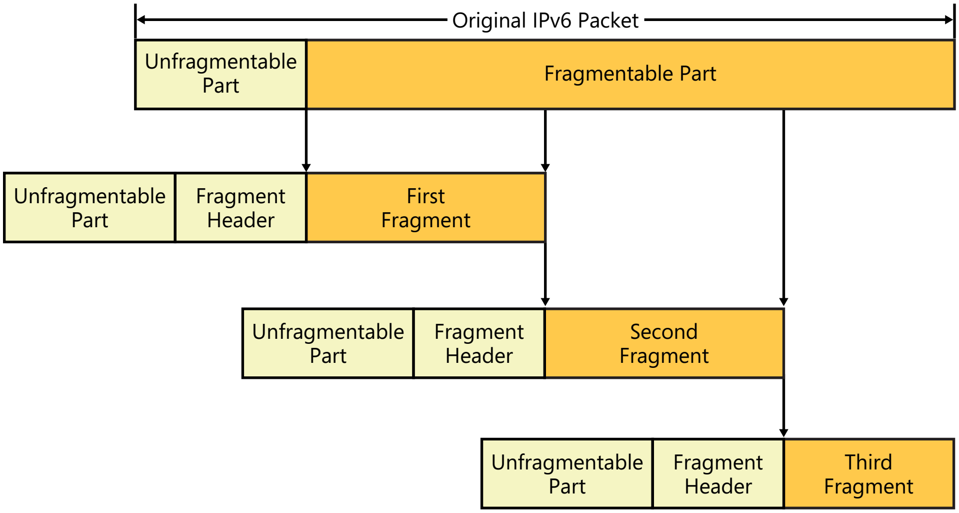


FIGURE 4-15 The IPv6 fragmentation process.

In each fragment, the Next Header field in the Fragment header indicates the first header or the upper-layer protocol in the original fragmentable part. The Fragment Offset field in the Fragment header indicates the offset, in 8-byte units known as *fragment blocks*, of this fragment relative to the original payload. The More Fragments flag is set on all fragment packets except the last fragment packet. All fragment packets created from the same IPv6 packet must contain the same Identification field value.

Fragmentation of IPv6 packets can occur when the upper-layer protocol of the sending host submits a packet to IPv6 that is larger than the path MTU to the destination. An example of IPv6 fragmentation is when a UDP application that is not aware of a path MTU sends large packets to a destination.

IPv6 packets sent to IPv4 destinations that undergo IPv6-to-IPv4 header translation might receive a path MTU update of less than 1280. In this case, the sending host sends IPv6 packets with a Fragment header in which the Fragment Offset field is set to 0 and the More Fragments flag is not set, and with a smaller payload size of 1272 bytes. The Fragment header is included so that the IPv6-to-IPv4 translator can use the Identification field in the Fragment header to perform IPv4 fragmentation to reach the IPv4 destination.

Network Monitor Capture

Here is an example of a Fragment header as displayed by Network Monitor 3.4 (frame 3 of capture 04_04 in the companion content for this book):

```
 Frame:
+ Ethernet: Etype = IPv6
- Ipv6: Next Protocol = ICMPv6, Payload Length = 1456
  + Versions: IPv6, Internet Protocol, DSCP 0
    PayloadLength: 1456 (0x5B0)
    NextProtocol: IPv6 Fragment header, 44(0x2c)
    HopLimit: 128 (0x80)
    SourceAddress: FE80:0:0:0:210:5AFF:FEAA:20A2
    DestinationAddress: FE80:0:0:0:250:DAFF:FED8:C153
  - FragmentHeader:
     NextHeader: ICMPv6
     Reserved: 0 (0x0)
   - FragmentInfor:
      FragmentOffset: 2896(0XB50)
      Reserved: (.............00.)
      M:        (...............1) More fragments
      Identification: 5 (0x5)
     FragmentData: Binary Large Object (1448 Bytes)
```

This is a fragment of a payload that uses the identification number of 5 and starts in byte position 2896 relative to the fragmentable portion of the original IPv6 payload.

Note You can create your own fragmented IPv6 traffic with the Ping.exe tool and the "–l" option by setting the buffer size so that IPv6 must fragment the ICMPv6-Echo Request message. Try this in the IPv6 test lab, as described in Appendix C, "Setting Up an IPv6 Test Lab," using Network Monitor 3.4 to capture the traffic.

IPv6 Reassembly Process

The fragment packets are forwarded by the intermediate IPv6 router or routers to the destination IPv6 address. The fragment packets can take different paths to the destination and arrive in a different order from which they were sent. To reassemble the fragment packets into the original payload, IPv6 uses the Source Address and Destination Address fields in the IPv6 header and the Identification field in the Fragment header to group the fragments. Figure 4-16 shows the IPv6 reassembly process.

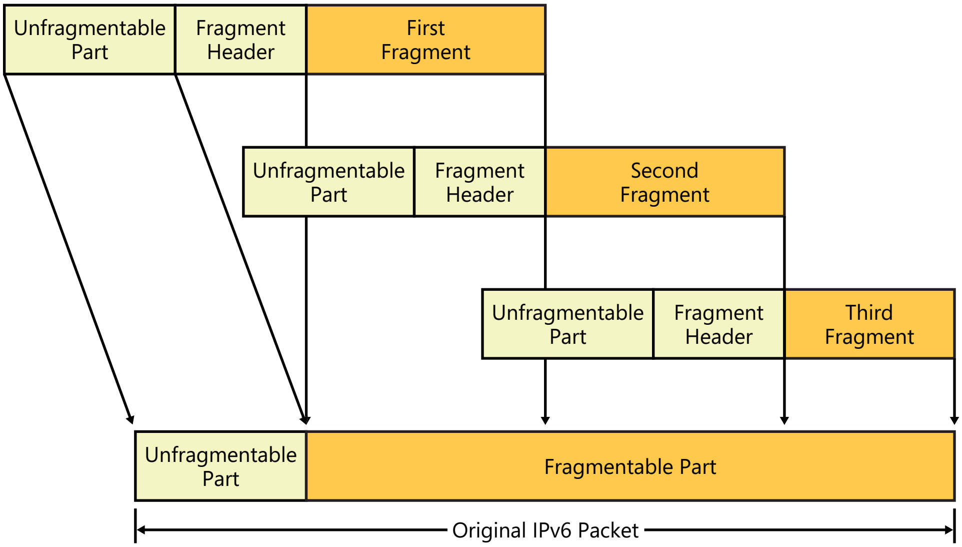


FIGURE 4-16 The IPv6 reassembly process.

After all the fragments arrive, the original payload length is calculated and the Payload Length field in the IPv6 header for the reassembled packet is updated. Additionally, the Next Header field of the last header of the unfragmentable part is set to the Next Header field of the Fragment header of the first fragment.

RFC 2460 recommends a reassembly time of 60 seconds before abandoning reassembly and discarding the partially reassembled packet. If the first fragment has arrived and reassembly has not completed, the reassembling host sends an ICMPv6 Time Exceeded-Fragment Reassembly Time Exceeded message to the source of the fragment.

Authentication Header

The Authentication header provides data authentication (verification of the node that sent the packet), data integrity (verification that the data was not modified in transit), and anti-replay protection (assurance that captured packets cannot be retransmitted and accepted as valid data) for the IPv6 packet, including the fields in the IPv6 header that do not change in transit across an IPv6 internetwork. The Authentication header, described in RFC 4302, is part of the security architecture for IP, as defined in RFC 4301. The Authentication header is identified by the value of 51 in the previous header's Next Header field. Figure 4-17 shows the structure of the Authentication header.

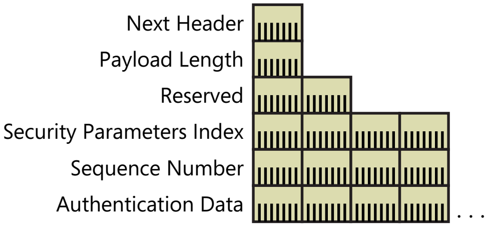


FIGURE 4-17 The structure of the Authentication header.

The Authentication header contains a Next Header field, a Payload Length field (the number of 4-byte blocks in the Authentication header, not counting the first two), a Reserved field, a Security Parameters Index (SPI) field that helps identify a specific IP Security (IPsec) security association (SA), a Sequence Number field that provides anti-replay protection, and an Authentication Data field that contains an integrity value check (ICV). The ICV provides data authentication and data integrity.

The Authentication header does not provide data confidentiality services for the upper-layer PDU by encrypting the data so that it cannot be viewed without the encryption key. To obtain data authentication and data integrity for the entire IPv6 packet and data confidentiality for the upper-layer PDU, you can use both the Authentication header and the Encapsulating Security Payload header and trailer.

Encapsulating Security Payload Header and Trailer

The Encapsulating Security Payload (ESP) header and trailer, described in RFC 4303, provide data confidentiality, data authentication, data integrity, and replay protection services to the encapsulated payload. The ESP header provides no security services for the IPv6 header or extension headers that occur before the ESP header. The ESP header and trailer are identified by the value of 50 in the previous header's Next Header field. Figure 4-18 shows the structure of the ESP header and trailer.

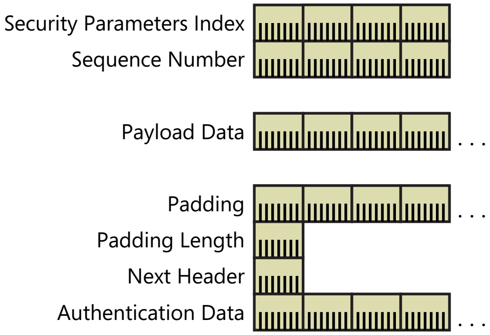


FIGURE 4-18 The structure of the ESP header and trailer.

The ESP header contains an SPI field that helps identify the IPsec SA, and a Sequence Number field that provides anti-replay protection. The ESP trailer contains the Padding, Padding Length, Next Header, and Authentication Data fields. The Padding field is used to ensure 4-byte boundaries for the ESP payload and appropriate data-block boundaries for encryption algorithms. The Padding Length field indicates the size of the Padding field in bytes. The Authentication Data field contains the ICV.

Details about how the ESP header and trailer provide data confidentiality, authentication, and integrity through cryptographic techniques are beyond the scope of this book.

IPv6 MTU

IPv6 requires that the link layer support a minimum MTU size of 1280 bytes. Link layers that do not support this MTU size must provide a link-layer fragmentation and reassembly scheme that is transparent to IPv6. For link layers that can support a configurable MTU size, RFC 2460 recommends that they be configured with an MTU size of at least 1500 bytes (the IPv6 MTU for Ethernet II encapsulation). An example of a configurable MTU is the Maximum Receive Unit (MRU) of a Point-to-Point Protocol (PPP) link.

Like IPv4, IPv6 provides a Path MTU Discovery process that uses the ICMPv6 Packet Too Big message described in the "Path MTU Discovery" section of Chapter 5, "ICMPv6." Path MTU Discovery allows the transmission of IPv6 packets that are larger than 1280 bytes.

IPv6 source hosts can fragment payloads of upper-layer protocols that are larger than the path MTU by using the process and Fragment header previously described. However, the use of IPv6 fragmentation is highly discouraged. An IPv6 node must be able to reassemble a fragmented packet that is at least 1500 bytes in size.

Table 4-5 lists commonly used local area network (LAN) and wide area network (WAN) technologies and their defined IPv6 MTUs.

TABLE 4-5 IPv6 MTUs for Common LAN and WAN Technologies

LAN or WAN Technology	IPv6 MTU
Ethernet (Ethernet II encapsulation)	1500, up to 9000 for jumbo frames
Ethernet (IEEE 802.3 SubNetwork Access Protocol [SNAP] encapsulation)	1492
IEEE 802.11	2312
Token Ring	Varies
Fiber Distributed Data Interface (FDDI)	4352
Attached Resource Computer Network (ARCNet)	9072
PPP	1500
X.25	1280
Frame Relay	1592
Asynchronous Transfer Mode (ATM) (Null or SNAP encapsulation)	9180

For more information about LAN and WAN encapsulations for IPv6 packets, see Appendix E, "Link-Layer Support for IPv6."

Upper-Layer Checksums

The current implementation of TCP, UDP, and ICMP for IPv4 incorporates into their checksum calculation a pseudo-header that includes both the IPv4 Source Address and Destination Address fields. This checksum calculation must be modified for TCP, UDP, and ICMPv6 traffic sent over IPv6 to include IPv6 addresses. Figure 4-19 shows the structure of the new IPv6 pseudo-header that must be used by TCP, UDP, and ICMPv6 checksum calculations. IPv6 uses the same algorithm as IPv4 for computing the checksum value.

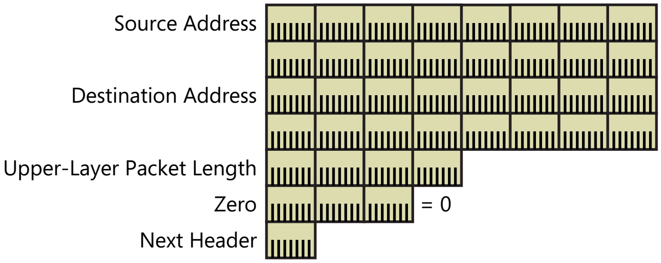


FIGURE 4-19 The structure of the new IPv6 pseudo-header.

The IPv6 pseudo-header includes the Source Address field, the Destination Address field, an Upper Layer Packet Length field that indicates the length of the upper-layer PDU, and a Next Header field that indicates the upper-layer protocol for which the checksum is being calculated.

References

The following references were cited in this chapter:

- **RFC 791** “Internet Protocol”
- **RFC 1812** “Requirements for IP Version 4 Routers”
- **RFC 2460** “Internet Protocol, Version 6 (IPv6)”
- **RFC 2474** “Definition of the Differentiated Services Field (DS Field)”
- **RFC 2675** “IPv6 Jumbograms”
- **RFC 2711** “IPv6 Router Alert Option”
- **RFC 3168** “The Addition of Explicit Congestion Notification (ECN) to IP”
- **RFC 3697** “IPv6 Flow Label Specification”
- **RFC 4301** “Security Architecture for the Internet Protocol”
- **RFC 4302** “IP Authentication Header”
- **RFC 4303** “IP Encapsulating Security Payload (ESP)”
- **RFC 5095** “Deprecation of Type 0 Routing Headers in IPv6”
- **RFC 6275** “Mobility Support in IPv6”

You can obtain these RFCs from *http://www.ietf.org/rfc.html*.

Testing for Understanding

To test your understanding of the IPv6 header, answer the following questions. See Appendix B, “Testing for Understanding Answers,” to check your answers.

1. Why does the IPv6 header not include a checksum?
2. What is the IPv6 equivalent to the IHL field in the IPv4 header?
3. How does the combination of the Traffic Class and Flow Label fields provide better support for prioritized traffic delivery?
4. Which extension headers are fragmentable and why? Which extension headers are not fragmentable and why?

5. Describe a situation that results in an IPv6 packet that contains a Fragment Header in which the Fragment Offset field is set to 0 and the More Fragments flag is not set to 1.

6. Describe how the new upper-layer checksum calculation affects transport layer protocols such as TCP and UDP.

7. If the minimum MTU for IPv6 packets is 1280 bytes, how are 1280-byte packets sent on a link that supports only 512-byte frames?

Index

Symbols

0s, compressing in representation of IPv6 addresses, 60
6to4, 20, 221, 295, 323–346, 391
- addresses, 323
- addressing example, 328–331
- blocking IPv4-encapsulated 6to4 traffic, 369
- communication examples, 336–341
  - 6to4 host to 6to4 host/router, 336
  - 6to4 host to IPv6 host, 337
- components, 327
- demonstrating 6to4 connectivity, 501–503
- DirectAccess server used as relay, 408
- disabled on Microsoft Corpnet computers with DirectAccess, 445
- disabling, 392
- host-to-router tunneling, 294
- route for DirectAccess server on IPv4 Internet, 403
- routes for DirectAccess client on IPv4 Internet, 401
- routing, 330
- routing functionality in edge devices, 347
- support in IPv6 for Windows, 27
- support in Windows, 331–335
  - host/router support, 331
  - manual router configuration, 333
- tunneling, 324–327
  - example, 325
- unicast connectivity for IPv6/IPv4 hosts or IPv6 sites, 418
- use for DirectAccess client traffic over IPv4 Internet, 399
- using in your intranet, reasons it's not recommended, 418
- using with ISATAP, 341–344
  - packet sent from 6to4 Router B to ISATAP Host B, 344
  - packet sent from ISATAP Host A to 6to4 Router A, 343

6to4 addresses, 285
6to4 Relay Name Group Policy setting, 331
6to4 State Group Policy setting, 392
127-bit prefixes on point-to-point router links, 73
128-bit IPv6 address space, 57
802.1X authentication, 388, 389
802.3 SNAP encapsulation, 520, 521–523
802.11 wireless, 10, 228, 523–529
802 addresses (IEEE), 83
- conversion to IPv6 interface identifiers, 86
- mapping to EUI-64 addresses, 84

1394 IEEE standard, 82
:: (double colon)
- compression of colon hexadecimal format to, 60
- determining number of 0 blocks or bits in, 60

A

AAAA records in the DNS, 9, 227, 291
- automatic registration of, 243
- configuring DNS infrastructure to support, 430
- DNS64 name resolution process for, 380
- environment containing bot A records and, 24
- manually creating for DNS Server service, 242
- mapping to address record (A) queries and responses, 28
- mapping to IPv4 A (address) records with DNS64, 377
- querying and processing of, IPv6 in Windows, 24
- support by DNS infrastructure, 422

A (address) records, 9
- DNS64 name resolution process for, 380
- mapping to AAAA records with DNS64, 377

AAL5 trailer, 535
A bits in Teredo addresses, 357
ABR (area border router), 271
Active Directory domains, DNS clients not members of, 422

Active Directory Domain Services (AD DS), 447
Active Directory Sites and Services
  configuring for IPv6 subnets, 449
  using to control/prioritize IPv6 traffic, 425
address autoconfiguration, 132, 205–226
  autoconfiguration process, 207
    for hosts in IPv6, 208–210
  autoconfiguration specifics, IPv6 for Windows, 220
  autoconfigured addresses for IPv6 for Windows, 221–224
  autoconfigured address states, 206
  DHCPv6, 210–219
    messages, 212–214
    stateful message exchange, 215
    stateless message exchange, 215
    support in Windows, 216–225
  different states for autoconfigured addresses, 231
  types of autoconfiguration, 205
addresses
  6to4, 323
    addressing example, 328–331
    tunneling interface, 325
  definition of address in IPV6, 10
  depletion problem, IPv6 as solution, 12
  duplicate address detection, 132, 158, 167–170
  example of selection on Microsoft Corpnet, 442
  international allocation problem, IPV6 as solution, 13
  IPv6, 57–90
    address allocation strategies, 81–82
    address space, 57
    anycast, 72
    configuring with Netsh.exe tool, 38
    configuring with Windows PowerShell, 37
    displaying for interfaces with netsh, 52
    for a host, 73
    interface identifiers, 82–88
    IPv4 address equivalents, 88
    manually configuring in Windows, 33
    multicast, 68–72
    multiple IPv6 addresses, Advanced TCP/IP Settings dialog, 36
    prefixes, 60
    for a router, 74
    scoped addresses and address selection, 14
    syntax, 58
    types of, 61
    unicast, 62–68
  IPv6 transition addresses, 284
  ISATAP addressing example, 312
  link-local ISATAP addresses, 302, 304
  manual configuration, support in IPv6 for Windows, 224
  selection rules for, 291
  source and destination, selection in IPv6 for Windows, 24
  stateless and stateful configuration in IPv6, 7
  Teredo, 356–359
    addressing example, 358
    obscured external address in, 357
  translation of IPv4 addresses to IPv6 with NAT64/DNS64, 381
addressing
  IPV6 addressing RFCs and Internet drafts, 451
  native IPv6 addressing allocation, 423
  plan for Microsoft Corpnet, 444
  unicast IPv6 addressing architecture, 416
address records, 291
address record types, MLDv2 Multicast Listener Report message, 200
address resolution, 132, 158, 159–163, 261
  differences from duplicate address detection, 167
  example, Part 1, 160
  example, Part 2, 161
  performed in redirect process, 177
Address Resolution Protocol. *See* ARP
address scanning, 390
address space
  disjoint, IPv6 as solution to problem, 12
  IPv4
    consequences of limited address space, 2–6
    impending exhaustion of, 1
  IPv6, 57
    large address space, 6
    subnetting, 75–81
address translation in NATs, 4
Advanced TCP/IP Settings dialog box for IPv6, 35
  DNS tab, 36
  IP Settings tab, 35
Advertisement Interval option, 134, 560
  Router Advertisement messages, 149
AF_INET6 constant, 540
A flag. *See* Autonomous flag, Prefix Information option
AH. *See* Authentication header
AI_ADDRCONFIG flag, 543
AI_ALL flag, 543

AI_CANONNAME flag, 543
AI_NUMERICHOST flag, 543
AI_NUMERICSRV flag, 543
AI_PASSIVE flag, 543
AI_V4MAPPED flag, 543
all-nodes multicast addresses, 185
American Registry for Internet Numbers (ARIN), 64, 416
answers to the "Testing for Understanding" sections, 457–486
anycast addresses, 61, 72
  interfaces on IPv6 router, 74
  IPv4 prefix for 6to4 relays, 327
  Subnet-Router, 73
APIPA (Automatic Private IP Addressing), 65
APIs (application programming interfaces)
  Windows APIs with IPv6 support, 30–33
application migration to support IPv6, 428
  inventorying applications before, 428
  scoping the work and scheduling migration, 429
applications
  inventorying for IPv6 technology gaps, 447
  IPv6 application RFCs, 452
application support for IPv6, 415
area border router (ABR), 271
A records in the DNS, 24, 291
ARIN (American Registry for Internet Numbers), 64, 416
ARP (Address Resolution Protocol), 8, 131
  differences between IPv4 and IPv6, 8
  DoS attacks bsed on, for IPv4, 388
  gratuitous ARP, 167
ATM
  null encapsulation of IPv6 packets, 534
  SNAP encapsulation of IPv6 packets, 536
Authenticated IP (AuthIP), 389
authentication
  for DHCPv6 messages, 388
  IEEE 802.1X authentication, 388
    preventing rougue IPv6 routers, 389
  IPsec tunnels between DirectAccess client and intranet, 404
  new mode for IP-HTTPS on Windows, 375
Authentication Data field, 112
Authentication header (AH), 7, 99, 101, 109, 112, 271
  for IPsec-protected communications, 405
  structure of, 112
Authentication indicator, Teredo packets, 361
  fields in, 361
  no client identifier or authentication value, 362
Automatic Private IP Addressing (APIPA), 65
automatic tunnels, 295
Autonomous flag, Prefix Information option, 137, 208, 211, 274, 275, 277, 424
Auxiliary Data field, MLDv2 Multicast Listener Report messages, 199
Auxiliary Data Length field, MLDv2 Multicast Listener Report messages, 199

B

Berkeley Sockets, 31
BGP-4 (Border Gateway Protocol version 4), 272
BGP (Border Gateway Protocol), 269
binary numbers
  conversion between binary, hexadecimal, and decimal numbers, 59
  enumerating subnetted address prefixes, 77
  IPv6 address in, 58
bind( ) function, 541, 542
binding cache, 561
binding update list, 562
bit boundaries, subnetting along, 82
bit-level subnetting, 76
Border Gateway Protocol (BGP), 269
Border Gateway Protocol version 4 (BGP-4), 272
broadcast addresses (IPv4), 70
bubble packets (Teredo), 360
  multicast bubble cache on clients, 364
business benefit of moving to IPv6, 12
business need, determining when planning IPv6 deployment, 446

C

Call ID field (GRE header), 4
care-of address (CoA), 568
carrier-grade NATs (CGNs), 4, 347
C (Conflict) flag, 230, 246
certificates
  authentication, IP-HTTPS client, 372, 374
  issued to DirectAccess computers, 408
Checksum field
  Destination Unreachable messages, 120
  ICMPv6 header, 119
  MLDv2 Multicast Listener Report messages, 198
  modified Multicast Listener Query messages, 196
  Multicast Listener Done messages, 194
  Multicast Listener Query messages, 190
  Multicast Listener Report messages, 191
  Neighbor Advertisement messages, 153

Neighbor Solicitation messages, 151
Packet Too Big messages, 122
Redirect messages, 156
Router Advertisement messages, 147
Router Solicitation messages, 146
checksums
Header Checksum field, IPv4, 94
IPv4 versus IPv6 headers, 8
upper-layer checksums in IPv6, 114
Checkv4.exe tool, 428, 429
Classless Inter-Domain Routing (CIDR) notation for IPv4, 60
Client Identification field (Teredo), 361
Client ID Length field (Teredo), 361
clients
DirectAccess
on the Internet, 409–411
on the intranet, 408
settings for DirectAccess behavior, 408
traffic over intranet, 399
traffic over IPv4 Internet, 399
inventorying IPv6 technology gaps on, 446
IP-HTTPS, 372
configuring settings, 374
new state for, 375
routing, 373
Teredo, 349, 352
and host-specific relay in Windows, 354
initial configuration for, 606–610
on-link default route, 364
on-link Teredo client destinations, 364
state types, 355
client state, Teredo client, 355
CLNP (Connectionless Network Protocol), 271
clouds, use by PNRP, 248
CNA (correspondent node address), 568
CoA (care-of address), 568
Code field
Destination Unreachable messages, 120
Echo Request messages, 125
ICMPv6 header, 119
modified Multicast Listener Query messages, 196
Multicast Listener Done messages, 194
Multicast Listener Query messages, 190
Multicast Listener Report messages, 191
Neighbor Advertisement messages, 153
Neighbor Solicitation messages, 151
Packet Too Big messages, 122
Parameter Problem messsages, 124
Redirect messages, 156
Router Advertisement messages, 147
Router Solicitation messages, 146
Time Exceeded messages, 123
colon hexadecimal, representation of IPv6 addresses, 58
compressing to double colon (::), 60
command line
IPv6-enabled tools for Windows, 42–49
Ipconfig, 42
Netstat tool, 47
Pathping tool, 46
Ping tool, 44
Route tool, 43
Tracert tool, 45
Windows IPv6 settings configuration, 22
company ID, 23, 83, 84
conceptual host data structures, 158
Cone flag, Teredo addresses, 357
cone NATs, 351, 612, 614, 617
initial communication from IPv6-only host to Teredo client, 623
initial communication from Teredo client to IPv6-only host, 619
configured tunnels, 294
configuring IPv6 protocol manually in Windows, 33
Conflict (C) flag, 230, 246
congestion control
Explicit Congestion Notification (ECN), 21, 93
ICMPv4 Source Quench messages and, 466
Connectionless Network Protocol (CLNP), 271
Connection Security rules in Windows, 403
configuring to specify IPsec protection for IPv6 traffic, 424
for DirectAccess, 406
DirectAccess client settings, 408
use for IPv6 traffic on Microsoft Corpnet, 445
connections to IPv6 Internet, 11
controlling or prioritizing IPv6 traffic, 425
correspondent registration in mobile IPv6, 564–567
CRL Distribution Points field, network location server's certificate, 409
Current Hop Limit field, Router Advertisement messages, 148, 275, 277

D

DAD. *See* duplicate address detection
data integrity, IPsec used with DirectAccess, 404
dead gateway detection, 166, 278–279
in IPv6 for Windows, 21

- decimal numbers
  - conversion between binary, hexadecimal, and decimal numbers, 59
  - method of enumerating subnetted address prefixes, 79
  - representing IPv4 addresses, 59
- default router
  - configuring, 274
  - link-local address of, 258
- default router list, 158
- Default Router Preference field, Router Advertisement messages, 148, 275, 277
- default routes, 254
- default state, Teredo client, 355
- DELAY state, 166
- denial of service (DoS) attacks
  - ND-based, protection against, 388
  - and verification of care-of address, 566
- deploying IPv6 on intranet, 413–436
  - application support for IPv6, 415
  - deployment of IPv6 on IPv4 intranet, 427–433
    - beginning application migration, 428
    - beginning deployment of native IPv6 infrastructure, 431
    - configuring DNS infrastructure, 430
    - connecting portions of intranet over IPv4 Internet, 432
    - connecting portions of intranet over IPv6 Internet, 433
    - obtaining global address space, 427
    - setting up test for IPv6 network, 427
    - starting rollout of IPv6 connectivity, 432
    - steps in process, 427
    - upgrading IPv4-only hosts to IPv6/IPv4, 430
  - network management infrastructure support for IPv6, 415
  - planning for IPv6 deployment, 413
  - unicast IPv6 addressing architecture, 416–426
    - controlled or prioritized delivery for IPv6, 425
    - host-based security and IPv6 traffic, 424
    - native IPv6 addressing allocation, 423
    - native IPv6 connectivity, 421
    - other transition technologies, 420
    - tunnel-based IPv6 connectivity, 417–420
  - Windows platforms' support for IPv6, 414
- Deprecated state, autoconfigured addresses, 206
- Destination Address field
  - Ethernet header of Neighbor Advertisement messages, 153
  - Ethernet header of Neighbor Solicitation messages, 150
  - Ethernet header of Redirect messages, 155
  - Ethernet header of Router Advertisement messages, 147
  - Ethernet header of Router Solicitation messages, 145
  - IPv4, 94, 114
  - IPv6 header, 96, 111
    - LLMNR query messages, 246
    - MLDv2 Multicast Listener Report message, 197
    - modified Multicast Listener Query message, 195
    - Multicast Listener Done message, 193
    - Multicast Listener Query message, 189
    - Multicast Listener Report message, 191
    - Neighbor Advertisement message, 153
    - Neighbor Solicitation message, 151
    - Redirect message, 155
    - Router Advertisement message, 147
    - Router Solicitation message, 145
  - IPv6 pseudo-headers, 114
  - Redirect messages, 156
- destination address selection algorithm, 232, 235–237
  - Windows support for, 244
- destination cache, 158, 253, 263
  - checking in host sending algorithm, 179
  - displaying with netsh interface, 53
  - viewing and checking destination address, 261
- Destination field, 288
- Destination Options header, 99, 101, 105–107, 109
  - Home Address destination option, 106
  - Home Address option in, 555
  - option types, 516
  - structure of, 105
  - summary of option types, 107
  - unrecognized options, 124
  - uses of, 106
- Destination Unreachable messages, 119–121, 263, 517
  - Code field, values of, 120
  - comparison for ICMPv6 and ICMPv4, 127
  - No Route to Destination message, example of, 121
  - structure of, 120
- DHCP (Dynamic Host Configuration Protocol), 2
  - intranet DNS servers configured through, 405
  - on Microsoft corporate network, 444
  - Windows-based server, registering address records for clients, 422

DHCP Server service in Windows, 424
DHCPv6 Advertise message, 213
DHCPv6 Confirm message, 213
DHCPv6 Decline message, 213
DHCPv6 (Dynamic Host Configuration Protocol for IPv6), 208, 210–219, 241, 253, 388
  authentication for message exchanges, 388
  clients, servers, and relay agents, 211
  configuring DNS clients for IPv6 addresses of DNS servers, 422
  configuring IPv6 addresses of IPv6-enabled DNS server for Windows, 24
  considerations when using, 423
  demonstrating in test lab, 503–505
  demonstrating use by client in test lab, 505
  deploying on intranet, 431
  interface identifiers assigned via, 83
  key difference in DHCPv6 and DHCP for IPv4, 211
  messages, 212–214
    fields in, 212
    options, 213
    between relay agents and servers, 214
  Router Advertisement messages, 210
  situations for use, 423
  stateful message exchange, 215
  stateless message exchange, 215
  support in Windows, 27, 216
    DHCPv6 client, 216
    DHCPv6 relay agent, 216
    DHCPv6 stateless and stateful server, 218
DHCPv6 Information-Request message, 213
DHCPv6 Rebind message, 213
DHCPv6 Reconfigure message, 213
DHCPv6 Relay Agent Properties dialog box, 217
DHCPv6 Relay-Forward message, 213
DHCPv6 Relay-Reply message, 213
DHCPv6 Release message, 213
DHCPv6 Renew message, 213
DHCPv6 Reply message, 213
DHCPv6 Request message, 213
DHCPv6 Solicit message, 213
dial-up connections
  IPv4 address for, 87
  IPv6-based, temporary address interface IDs, 87
Differentiated Services Code Point (DSCP), 2, 426
Differentiated Services (DS) field, 426
DirectAccess, 397–412
  benefits of, 397
  how it works, 408–411
    DirectAccess client on Internet, 409–411
    DirectAccess client on intranet, 408
  IP-HTTPS clients, 372
  NAT64/DNS64 component configured as DirectAccess server, 378
  NAT64/DNS64 services for, on Microsoft Corpnet, 441
  network location detection, 406
  and role of IPsec, 403
    data integrity, 404
    encryption, 404
  and role of NRPT, 405
    NRPT exemptions, 406
  Technology and Solution page, 398
  Teredo server and relay configuration, 365
  use of IPv6, 398
    client traffic over intranet, 399
    client traffic over IPv4 Internet, 399
    force tunneling, 400
    IPv6 routing, 401–403
  use of NAT64/DNS64 only at network edge for clients, 449
  use of NRPT, 249
  use on Microsoft Corpnet, 444
direct connections to IPv6 Internet, 11
DisabledComponents registry value, 40, 419
disabled state, Teredo client and host-specific relay, 355
disabling IPv6 in Windows systems, 40
  configuration combinations and DisabledComponents registry value, 41
distance vector routing protocols, 269
DNS (Domain Name System)
  adding DNS servers with Netsh.exe tool, 39
  adding DNS servers with Windows PowerShell, 37
  configuring infrastructure for IPv6 deployment, 430
  configuring IPv6 settings in Windows, 35
  demonstrating zone transfers over IPv6, 507
  DNS64, 296
  DNS64 in Windows IPv6, 20, 28
  DNS Client service, 246
  DNS dynamic update, 243–244
  DNS Recursive Name Server option, 241
  DNS resolver, 241
  DNS resolver cache, 241
  DNS Server service, 242–243
    GlobalNames zone and functionality, 449
    NAT64/DNS64 and, 379

DNS tab of Advanced TCP/IP Settings dialog box, 36
DNS zone transfers, 244
dynamic update of AAAA and PTR records, 446
enhancements for IPv6, 227
information stored in NRPT, 405
infrastructure on Microsoft Corpnet, 444
mapping of host names to IPv4 and IPv6 records, 9
Name Query Response messages, 232
Name Query Response with IPv6 and IPv4 addresses, on Windows, 20
name resolution for IPv6, 422
name resolution for Microsoft 6to4 relay, 327
name resolution on single-subnet and ad-hoc wireless networks via LLMNR, 25
normal name resolution, 246
resolving host name "ISATAP", 307
secure dynamic updates and non-domain joined Windows hosts, 422
servers specified in DirecAccess client's TCP/IP settings, 406
support for IPv6 in Windows, 24
transition infrastructure, 290
DNS Security Extensions (DNSSEC), 250
DNS64 and, 379
use of NRPT, 249
Don’t Fragment flag, 109
dormant state, Teredo client, 354
dotted-decimal format, representation of IPv4 addresses, 58
double colon (::)
compression of zeros in IPv6 addresses, 60
determining number of 0 blocks or bits in, 60
DSCP (Differentiated Services Code Point), 2, 426
dual IP layer architecture, 286
types of packets, 287
dual-stack architecture, 287
types of packets, 288
duplicate address detection, 132, 158, 167–170
DAD proxy on IP-HTTPS server, 374
example, Part 1, 168
example, Part 2, 169
using in address autoconfiguration, 207
Dynamic Host Configuration Protocol. *See* DHCP; DHCPv6
dynamic mappings, NAT, 350
dynamic routing, 268
dynamic updates for DNS records, 430

E

Echo messages (ICMP), 44
Echo Reply messages, 126
structure of, 125
Echo Request messages, 124
encapsulated with IPv4 header, 290
example, displayed by Network Monitor Capture, 125
Echo Request packet, 96
use of echo servers by NAT-aware applications, 5
ECN (Explicit Congestion Notification), 95
support in IPv6 for Windows, 21
edge devices
filtering of IPv4 and IPv6 traffic by edge routers, 425
tunneling of IPv6 packets between, using 6to4, 348
edge traversal, enabling for Teredo, 349
Encapsulating Security Payload (ESP)
encryption of DirectAccess client traffic, 404
Encapsulating Security Payload (ESP) header and trailer, 7, 112, 271
structure, 113
Encapsulating Security Payload header, 99, 101, 109
encryption
DirectAccess client traffic over Internet, 404
end-to-end connectivity, restoring with IPv6, 13
end-to-end IPv6 delivery process, 260–267
IPv6 on destination host, 265–267
receiving host process, 267
IPv6 on sending host, 260–262
sending host process, 262
IPv6 on the router, 262–265
router forwarding process, 264
enterprise client state, Teredo client, 355
error messages (ICMPv6), 118, 119–124
comparision to ICMPv4 messages, 127
Destination Unreachable, 119–121
Packet Too Big, 121, 128
Parameter Problem messsages, 123
Time Exceeded, 123
ESP. *See* Encapsulating Security Payload (ESP) header and trailer
ESP-NULL setting, IPsec-protected communications, 405

Ethernet, 10
- Ethernet header of Neighbor Advertisement message, 153
- Ethernet header of Neighbor Solicitation message, 150
- Ethernet header of Redirect message, 155
- Ethernet header of Router Advertisement message, 147
- Ethernet header of the Router Solicitation message, 145
- Ethernet II encapsulation, 128, 520
- IEEE 802 addresses, 83
- modules for IEEE 802.3 in Windows, 18
- SNAP encapsulation, 114, 516, 521–524
- Source and Target Link-Layer Address ND options, 135

Ethernet addresses, mapping IPv6 multicast addresses to, 71

EtherType of 0x86DD, 394

Explicit Congestion Notification (ECN), 95
- support in IPv6 for Windows, 21

Extended Unique Identifier (EUI)-64 addresses, 82, 84
- conversion to IPv6 interface identifiers, 85, 431
- creating from IEEE 802 addresses, 84
- interface identifiers based on, 83, 87, 390

extension headers, 91, 92, 99–113
- Authentication header, 112
- Destination Options header, 105–107
- Encapsulating Security Payload (ESP) header and trailer, 112–116
- Fragment header, 108–112
- Hop-by-Hop Options header, 101
  - Option Type field, 103
- IPv6 MTU, 113
- order of, 101
- Routing header, 107

external IPv4 address, obscured, in Teredo addresses, 357

external UDP port, obscrued, in Teredo addresses, 357

F

FDDI (Fiber Distributed Data Interface), 141
- encapsulation of IPv6 packets, 528

files and folders (network), UNC path name for, 24

firewalls
- host-based, 425
- host-based stateful firewalls, 391
- IPv6 functionality, 415
- protection from unwanted incoming IPv6 traffic, 349
- router-based, 425

FireWire, 82

flags
- and indicators in LLMNR header, 230
- Teredo addresses, 357

Flags field
- IPv4, 93
- IPv6 multicast addresses, 69

Flow Label field, IPv6 header, 95, 98

force tunneling, 400

Forefront Unified Access Gateway (UAG) 2010, 378

forwarding of packets
- increased efficiency in IPv6, 14
- inefficiencies caused by IPv4 NATs, 4
- IPv6 multicast traffic, 186

FQDN (fully qualified domain name)
- resolving in network location detection, 407
- resolving with Windows Sockets Getaddrinfo( ) function, 25

fragmentation process in IPv6, 109

Fragment header, 99, 101, 108–112
- differences in IPv4 and IPv6 fragmentation fields, 109
- example, displayed by Network Monitor Capture, 110
- IPv6 fragmentation process, 109
- reassembly process for fragment packets in IPv6, 111
- structure of, 108

Fragment Offset field, 108, 109
- IPv4, 93

Fragment Reassembly Time Exceeded, 123

Frame Control field, IEEE 802.11 header, 525

Frame Relay, 533

Framing layer, implementation in Windows, 18

FTP (File Transfer Protocol), IPv6 capabilities on Windows, 30

fully qualified domain name (FQDN)
- resolving in network location detection, 407
- resolving with Windows Sockets Getaddrinfo( ) function, 25

G

gateways
- configuring default router gateway with Netsh.exe tool, 38
- configuring default router gateway with Windows PowerShell, 37
- dead gateway detection, 166, 278–279
- dead gateway detection in IPv6 for Windows, 21
- default router gateway, configuring in Windows, 35
- multiple default gateways setting, Advanced TCP/IP Settings dialog, 36

general query, Multicast Listener Query messages, 189

Generic Routing Encapsulation (GRE) headers, 4

Getaddrinfo( ) function, 249, 415

Gethostbyname( ) function, 415

Getnameinfo( ) function, 429

Get-NetIPAddress –AddressFamily IPv6 Windows PowerShell command, 50, 222

Get-NetIPConfiguration Windows PowerShell cmdlet, 43

Get-NetIPInterface –AddressFamily IPv6 Windows PowerShell command, 49

Get-NetIPInterface Windows PowerShell cmdlet, 45

Get-NetNeighbor –AddressFamily IPv6 Windows PowerShell command, 51

Get-NetNeighbor Windows PowerShell cmdlet, 159, 261

Get-NetPrefixPolicy Windows PowerShell cmdlet, 245, 291

Get-NetRoute –AddressFamily IPv6 Windows PowerShell command, 50

Get-NetRoute Windows PowerShell cmdlet, 159, 256

Getpeername( ) function, 542

Getsocketname( ) function, 542

global cloud, 248

Global ID, unique local addresses, 66, 416

global unicast addresses, 62
- assigned to interfaces in IPv6 host, 74
- automatically registered AAAA records, 243
- fields in, 63
- obtaining global address space from regional registry, 427
- obtaining global IPv6 address space for Microsoft Corpnet, 439
- structure defined in RFC 3587, 62
- subnetting subnet ID portion, 75
- topological structure, 64
- unicast addresses for subnets, 416

gratuitous ARP, 167

GRE (Generic Routing Encapsulation) headers, 4

Group ID field, IPv6 multicast addresses, 70

Group Policy settings
- 6to4 Relay Name, 331
- 6to4 State, 392
- IP-HTTPS State, 374
- for IPv6 transition technologies on Windows, 22, 29
- ISATAP Router Name, 305
  - using, 311
- ISATAP State, 392
- Teredo Client Port, 356
- Teredo Default Qualified, 356
- Teredo Refresh Rate, 354
- Teredo Server Name, 366
- Teredo State, 355, 393
- for transition technologies, 20

GUI (graphical user interface), Windows, configuring IPv6 settings, 22

H

Happy Eyeballs, 20. *See also* IPv6 brokenness; RFC 6555

hardware requirements for IPv6 test lab, 489

Header Checksum field, IPv4, 94

Header Extension Length field, 100
- in Hop-vy-Hop Options header, 101
- Routing header, 107

headers, 91–116
- extension headers for IPv6, 8
- ICMPv6, 118
- IPsec header support requirement in IPv6, 7
- IPv4, 92–94
- IPv6, 94
  - comparing to IPv4 headers, 97–99
  - example of, 96
  - extension headers, 99–113
  - fields in, 95
  - LLMNR query messages, 246
  - MLDv2 Multicast Listener Report message, 197
  - modified Multicast Listener Query message, 195
  - Multicast Listener Done message, 193
  - Multicast Listener Query message, 189
  - Multicast Listener Report message, 191
  - Neighbor Advertisement message, 153
  - Neighbor Solicitation message, 151

new format, 6
Redirect message, 155
RFC 2460, support on Windows, 21
Router Advertisement message, 147
Router Solicitation message, 145
upper layer checksums, 114
values of Next Header field, 96
LLMNR, 230
structure of IPv6 packet, 91
Teredo packets, 360
Authentication and Origin indicators, 361
hexadecimals
colon hexadecimal representation of IPv6 addresses, 58
conversion between binary, hexadecimal, and decimal numbers, 59
method for enumerating subnetted address prefixes, 78
numbering system choice for IPv6, 59
subnetting along nibble boundaries, 75
Home Address destination option, 106, 555
home addresses
destinations with source addresses that are, 236
for mobile IPv6 nodes, 234
Home Address field, 106
Home Agent Address Discovery Reply message, 557, 584
Home Agent Address Discovery Request message, 556, 584
Home Agent flag, Router Advertisement messages, 148
Home Agent Information option, 134, 149
home agents list, 563
Hop-by-Hop Options header, 99, 101–105, 109
example, displayed by Network Monitor Capture, 105
Option Type field, 103
option types, 516
order of, 101
Router Alert option and Padding option, 193
structure of, 101
summary of option types, 107
unrecognized options, 124
Hop-Count field, 214
Hop Limit Exceeded in Transit, 123
Hop Limit field, 260, 263
IPv6 header, 95, 123
default setting for, 171
LLMNR query messages, 246
MLDv2 Multicast Listener Report message, 197
modified Multicast Listener Query message, 195
Multicast Listener Done message, 193
Multicast Listener Query message, 189
Multicast Listener Report message, 191
Neighbor Advertisement message, 153
Neighbor Solicitation message, 151
Redirect message, 155
Router Advertisement message, 147
Router Solicitation message, 145
host address (A) resource records. *See* A (address) records
host IDs
IPv4 addresses, 82
in subnetted address prefixes, 75
host receiving algorithm, 600–602
host/router, 6to4, 327
6to4 host to 6to4 host/router communication, 336
addressing example, 329
Windows support for, 331
host routes, 254
hosts
6to4, 327
routes used, 330
6to4 host to 6to4 host/router communication, 336
6to4 host to IPv6 host communications, 337
display by Network Monitor capture, 340
address autoconfiguration process in IPv6, 207, 208–210
diagram of, 209
communication between ISATAP hosts, 314
configuring for IPv4 address of ISATAP router, 319
defined, 10
dynamic registration of IPv6 host records in the DNS, on Windows, 24
intranet, upgrading IPv4-only hosts to IPv6/IPv4 hosts, 430
inventorying IPv6 technology gaps on, 446
IPv6 addresses for, 73
IPv6 multicast support, 184
IPv6 on destination host, 265–267
IPv6 on sending host, 260–262
ISATAP, communication between, in 6to4 sites, 342
ISATAP host to IPv6 host communications, 315
LLMNR, 229

MLDv2-capable, types of address records sent, 200
security for IPv6 traffic, 424
sending Multicast Listener Query messages to, 187
strong and weak behaviors, 255
strong host model for sending and receiving, IPv6 for Windows, 22
on subnet, identification in IPv4 and IPv6, 61
Teredo host-specific relay, 353
routes, 363
Teredo client and, 354–356
Windows-based IPv6, zone IDs for link-local addresses, 65
host sending algorithm, 179, 597–599
diagram of, 180
Hosts file, 240–241
host-to-host tunneling, 293
ISATAP, 304
host-to-router tunneling, 292
ISATAP, 304
HTTPS, 369. *See also* IP-HTTPS
IP over Secure Hypertext Transfer Protocol (HTTPS) (IP-HTTPS), 20

I

IANA (Internet Assigned Numbers Authority)
current list of permanently assigned IPv6 multicast addresses, 70
permanently assigned (well-known) multicast addresses, 69
ICMP (Internet Control Message Protocol)
Echo messages, 44
ICMPv4 Redirect messages, 8
ICMPv4 Router Discovery, 8
NAT64 providing basic translation for, 378
router discovery in ICMPv4, 170
router discovery, replacement by Neighbor Discovery, 131
ICMPv6 (Internet Control Message Protocol for IPv6), 8, 117–130
changes in PMTU, 128
comparing with ICMPv4 messages, 127
Destination Unreachable messages, 263
Echo Request and Echo Reply messages
use by Teredo tunneling interface, 365
Echo Request messages
encapsulated with IPv4 header, 290
Echo Request packet, 96
error messages, 119–124
header, 118
informational messages, 124–126
messages, 516
for mobile IPv6, 556–558
overview, 117
Packet Too Big message, 113
Packet Too Big messages, 263
Path MTU Discovery process, 127
support in IPv6 for Windows, 21
types of messages, 118
ICS (Internet Connection Sharing), 332, 418
ICV (integrity value check), 112
Identification field, 108
in Fragment header, 110
use in reassembly process, 111
IPv4 headers, 93
IPv4 versus IPv6, 109
Identifier field, Echo Request messages, 125
IDSs (intrusion detection devices)
detection of malicious IPv6 traffic, 393
IDSs (intrusion detection systems)
analyzing for IPv6 technology gaps, 448
support for analyzing native and tunneled IPv6 traffic, 425
IEEE (Institute of Electrical and Electronics Engineers)
801.11 wireless standard, 18
802.1X authentication, 388
802.3 Ethernet standard, 18
802.3 Sub-Network Access Protocol (SNAP) encapsulation, 520, 521–523
802.5 SNAP encapsulation, 526
802.11 wireless, 10, 523–529
802.11 wireless network, 228
802 addresses, 83, 390
conversion to IPv6 interface identifiers, 86
802.x SNAP header, 536
1394 standard (FireWire), 82
EUI-64 addresses, 82, 84
Individual/Group (I/G) bit, 186
mapping of 802 addresses to EUI-64 addresses, 84
IETF (Internet Engineering Task Force), 2
deprecation of Routing Type 0 header, 108
IPV6 RFCs, 451–456
Mobility extension header, 552
NAT64/DNS64, 377
Request for Comments webpage, 455
RFCs and Internet drafts, information on, 54
standards for IPv6 protocol stack functionality, 21

IGDs (Internet gateway devices), 12
I/G (Individual/Group) bit, 84, 186
IGMPv2 (Internet Group Management Protocol version 2), 183, 202
  Leave Group messages, 193
  messages and MLD equivalents, 194
IGMPv3 (Internet Group Management Protocol version 3), 183
  messages and MLDv2 equivalents, 201
IGP (interior gateway protocol), 444
IIS (Internet Information Services), FTP Publishing and SMTP Server services, 30
IKE (Internet Key Exchange) protocol, 7, 389
in6_addr data structure, 541
in6addr_loopback data structure, 542
IN6_IS_ADDR_LINKLOCAL macro, 547
IN6_IS_ADDR_LOOPBACK macro, 547
IN6_IS_ADDR_MC_GLOBAL macro, 547
IN6_IS_ADDR_MC_LINKLOCAL macro, 547
IN6_IS_ADDR_MC_NODELOCAL macro, 547
IN6_IS_ADDR_MC_ORGLOCAL macro, 547
IN6_IS_ADDR_MC_SITELOCAL macro, 547
IN6_IS_ADDR_MULTICAST macro, 547
IN6_IS_ADDR_SITELOCAL macro, 547
IN6_IS_ADDR_UNSPECIFIED macro, 547
IN6_IS_ADDR_V4COMPAT macro, 547
IN6_IS_ADDR_V4MAPPED macro, 547
INCOMPLETE state, 165
indicators, Teredo packets, 361–363
Individual/Group (I/G) bit, 186
INetFwPolicy2( ) function, 349
informational messages (ICMPv6), 118, 124–126
  comparision to ICMPv4 messages, 127
  Echo Reply, 125
  Echo Request, 124
Institute of Electrical and Electronics Engineers. *See* IEEE
Integrated Intermediate System-to-Intermediate System (IS-IS) IGP, 444
  intranet based on, configuring Active Directory Sites and Services for, 449
Integrated IS-IS for IPv6, 271
integrity value check (ICV), 112
interface identifiers, 82–88
  6to4-derived, 221
  deriving permanent interface ID, 390
  disabling on infrastructure servers, 431
  EUI-64 address-based, 83, 84
  EUI-64–derived, 207
  for global address prefixes, 231
  global unicast addresses, 64
  Interface ID field, global unicast addresses, 63
  ISATAP–derived, 221
  link-local addresses, 65
  obtaining for IPv6 unicast addresses, 85–87
  temporary addresses, 87
  unique local addresses, 66
Interface ID field
  6to4 addresses, 68
  global unicast addresses, 63
  Teredo flags field in, 357
interfaces
  6to4 tunneling interface
    determining name of, 325
    enabling forwarding on, 334
  definition in IPv6, 10
  displaying list of IPv6 interfaces with netsh, 51
  IP-HTTPS tunneling interface, 372
  listing IPv6 interfaces with PowerShell, 49
  random interface IDs generated on Windows, 23
  Teredo tunneling interface, 349, 353
    packet forwarding, 364
interior gateway protocol (IGP), 444
Internet
  access to/from Microsoft Corpnet, 445
  IPv6-capable organization network and, 11
  mapping of internal (intranet) data streams to, 2
  preventing unwanted traffic from, 393
Internet Connection Sharing (ICS), 332, 418
Internet Control Message Protocol. *See* ICMP; ICMPv6
Internet draft, "Operational Guidance for IPv6 Deployment in IPv4 Sites using ISATAP", 320, 321
Internet draft, "Routing IPv6 with IS-IS", 272
Internet draft, "Update to RFC 3484 Default Address Selection for IPv6", 24, 54, 233, 244, 245
Internet Engineering Task Force. *See* IETF
Internet Explorer
  literal IPv6 addresses in URLs, 25
  use of WinInet to access IPv6-enabled websites, 32
Internet Group Management Protocol version 2 (IGMPv2), 183
Internet Group Management Protocol version 3 (IGMPv3), 183
Internet Header Length (IHL) field, IPv4, 93
Internet Information Services (IIS), FTP Publishing and SMTP Server services, 30
Internet Key Exchange (IKE) protocol, 7, 389

Internet layer
- IPv4, in Windows TCP/IP protocols implementation, 18
- IPv6, in Windows TCP/IP protocols implementation, 18
- IPv6 RFCs for, 453

Internet Protocol Control Protocol (IPCP), 87
Internet Protocol security. *See* IPsec
Internet Protocol Version 6 (TCP/IPv6). *See* TCP/IPv6
Internet Protocol Version 6 (TCP/IPv6) Properties dialog box, 34
- General tab, 34
  - Advanced TCP/IP Settings dialog box, 35

Internet registry, obtaining global prefix from, 416
intranets. *See also* deploying IPv6 on intranet
- controlling tunelled traffic on, 391
- DirectAccess client on, 408
- DirectAccess client traffic over, 399
- DirectAccess server on, routes for, 402
- ISATAP on, 27
- mapping data streams to Internet data streams, 2
- routes for traffic from DirectAccess client to intranet resource, 401
- routes for traffic from intranet resource to DirectAccess client, 402

Intra-Site Automatic Tunnel Addressing Protocol. *See* ISATAP
intrusion detection devices. *See* IDSs
intrusion prevention systems (IPSs), analyzing for IPv6 technology gaps, 448
Invalid state, autoconfigured addresses, 206
IP6.ARPA domain, 9, 228, 244
- PTR records for IPv6-only and IPv6/IPv4 nodes, 291

Ipconfig tool, 42
- displaying zone ID of link-local addresses, 66
- ipconfig command on computer running Windows Server 2012, 42
- Windows PowerShell equivalent to ipconfig command, 43

IPCP (Internet Protocol Control Protocol), 87
IP Helper, 31
IP-HTTPS, 20, 369–376, 420
- centrally configuring Group Policy settings for, 29
- client routing, 373
- components, 371–372
- configuring client settings, 374
- DirectAccess server as server for, 408
- establishing a connection, 372
- features in Windows Server 2012 and Windows 8, 374
- IPv6 packet encapsulated with, 370
- route for DirectAccess server on IPv4 Internet, 403
- routes for DirectAccess client on IPv4 Internet, 402
- support in IPv6 for Windows, 28
- termination of SSL session for IP-HTTPS server, 371
- use by DirectAccess client for traffic over IPv4 Internet, 399
- use on Microsoft Corpnet computers with DirectAccess, 445
- Windows support for, 370

IP-HTTPS State Group Policy setting, 374
IPng (IP-The Next Generation), 2
IP over Secure Hypertext Transfer Protocol (HTTPS). *See* IP-HTTPS
IPsec (Internet Protocol security), 2
- cryptographic protection of IPv6 traffic, 424
- DirectAccess and, 403
  - data integrity, 404
  - encryption of client traffic, 404
  - Gateway component on DirectAccess server, 404
- header support, requirement in IPv6, 7
- integrated support in IPv6 for Windows, 22
- with IPv6, Windows support for, 403
- IPv6 versus IPv4, 15
- Layer Two Tunneling Protocol with IPsec (L2TP/IPsec), 432
- protection of IPv6 packets, 389
- protection of router discovery messages, 389
- security association (SA), 112
- support in IPv6 on Windows, 19

IPsec NAT-Traversal (NAT-T), 4
IPSs (intrusion prevention systems), analyzing for IPv6 technology gaps, 448
IP-The Next Generation (IPng), 2
IPv4
- addressing concepts and IPv6 equivalents, 88
- broadcast addresses, 70
- connectivity on Windows with IPv6 enabled and preferred, 20
- consequences of limited address space, 2–6
- critical router loop, 98
- disjoint public and private address space, 12
- expression of addresses with prefix length, 60
- fragmentation fields, 109

- headers, 92–94
  - comparing to IPv6 headers, 97–99
- ICMPv4
  - comparison with ICMPv6 messages, 127
- key differences from IPv6, 8
- limitations of, 1
- link-local addresses defined in RFC 3927, 65
- loopback address, 67
- NATs (Network Address Trnaslators), 350
- neighbor messages and functions, 181
- packet translated from IPv6 by NAT64/ DNS64, 382
- possible number of sites connected to Internet, 64
- subnet masks in, 60
- subnetting in, 75
- Teredo server address, 357
- unspecified address, 67
- using both IPv4 and IPv6, 286–288

IPv4-compatible addresses, 284

IPv4-encapsulated IPv6 traffic
- NATs and, 348
- security monitoring of, 448

IPv4-mapped address, 285

IPv6
- better support for prioritized delivery, 7
- case for deployment, 12
  - more efficient forwarding, 14
  - restoring end-to-end communication, 13
  - scoped addresses and address selection, 14
  - solving address depletion problem, 12
  - solving disjoint address space problem, 12
  - solving international address allocation problem, 13
  - support for security and mobility, 15
- extensibility, 8
- host communications with 6to4 host, 337
- IPsec header support requirement, 7
- key differences from IPv4, 8
- large address space, 6
- list of features, 6
- Neighbor Discovery protocol, 8
- new header format, 6
- stateless and stateful address configuration, 7
- terminology, 9

IPv6 brokenness, 237

IPV6CP (IPv6 Control Protocol), 26

ipv6-literal.net names
- support in IPv6 for Windows, 24
- Windows support for, 247

IPv6-over-IPv4 traffic, 302

IPv6-over-IPv4 tunneling, 288–290

IPv6 over IPv6, 537–538

IPv6 routing table, 253
- entry types, 254
- using in route determination, 254

IPv6 Static Route dialog box, 278

IPv6 technology gaps, determining, 446

ISATAP (Intra-Site Automatic Tunnel Addressing Protocol), 20, 221, 295, 301–322, 391, 399, 417
- addresses, 285
- addressing example, 312
- centrally configuring Group Policy settings for, 29
- communication examples, 314–317
  - ISATAP host to IPv6 host, 315
  - ISATAP host to ISATAP host, 314
- components, 304
- configuring ISATAP router, 317–320
  - example using PowerShell commands, 318
- configuring to control traffic from Internet, 394
- connectivity across entire Microsoft Corpnet, 441
- demonstrating ISATAP-based IPv6 connectivity, 496
- deployment on Microsoft Corpnet, 438
- disabling, 302, 392
- enabling on specific computers on a network, 305
- hosts, 304
- host-to-router tunneling, 294
- in Windows Server 2012 and Windows 8, 321
- netsh interface isatap set router command, 312
- projected removal from Microsoft Corpnet, 441
- reasons deployment is not recommended, 417
- receiving Router Advertisement message from ISATAP router, 222
- router discovery for hosts, 306–312
- routes from intranet resource on IPv6-capable intranet, 403
- routing, 313
- tunneling, 302
  - example, 303
- unicast connectivity for IPv6/IPv4 hosts on IPv4-only intranet, 417
- using ISATAP Router Name Group Policy setting, 311

using with 6to4, 341–344
packet sent from 6to4 Router B to ISATAP Host B, 344
packet sent from ISATAP Host A to 6to4 Router A, 343
Windows support for, 27
ISATAP Router Name Group Policy setting, 305, 306
ISATAP State Group Policy setting, 303, 392
ISO 10589, 271, 280

J

join latency, 188
jumbo Ethernet frames, 114, 516
jumbograms, 92, 104, 108
Jumbo Payload option, 104

L

L2TP (Layer Two Tunneling Protocol), 289
support by IPV6CP in Windows, 26
LANs (local area networks)
connection properties in Network Connections folder (Windows), 18
interface IDs for interfaces, 82
IPV6 MTUs for common technologies, 113
link-layer technologies in IPv6, 10
media, 519
MTUs for common technologies, 516
multiple addresses assigned to interfaces, 231
Large Scale NAT (LSN), 347
Layer Two Tunneling Protocol (L2TP), 289
support by IPV6CP in Windows, 26
Layer Two Tunneling Protocol with IPsec (L2TP/IPsec), 432
Length field
MTU option, 142
Prefix Information option, 137
Redirected Header option, 140
Route Information option, 143
lifetime for temporary addresses, setting on Windows, 23
Link-Address field, 214
link layer
IPv6 RFCs and Internet drafts for, 454
support for IPv6, 519–538
link-layer address resolutions, 70
link-layer encapsulation, 289
IPv6-over-IPv4 tunneling, 289
link-layer technologies in bridged or Layer-2 switched environments, 141
link-local addresses, 7, 65–66
assigned to interfaces on IPv6 host, 74
autoconfiguration of, 206
derivation in address autocinfiguration, 207
display by ipconfig, 43
interfaces on IPv6 router, 74
ISATAP addresses, 302
example, 304
zone ID, 45
zone IDs for, 65
link-local cloud, 248
Link-Local Multicast Name Resolution. *See* LLMNR
link MTU (maximum transmission unit), 10, 128
links, defined, 10
link state–based routing protocols, 269
link state database (LSDB), 271
literal IPv6 addresses in URLs
use by Internet Explorer, 32
Windows' support for, 25
LLMNR (Link-local Multicast Name Resolution), 19, 228–231
message structure, 229–231
fields and flags defined for LLMNR, 230
headers, 230
Windows support for, 25, 246
Lmhosts file, 307
Local (L) flag in unique local addresses, 66
logical interfaces, 10
in Windows Framing Layer, 18
loopback address, 67
assigned to interfaces on IPv6 hosts, 74
interface on IPv6 router, 74
LSDB (link state database), 271
LSN (Large Scale NAT), 347

M

MAC (Media Access Control) addresses, 390
destination MAC addresses on Ethernet networks, 71
destination multicast address, mapping to IPv6 multicast address, 193
IEEE 802 addresses, 83
mapping of, IPv6 interface IDs and, 82
registering multicast MAC address with network adapter, 185
table of interesting destination addresses on network adapter, 186

Managed Address Configuration flag, Router Advertisement message, 148, 208, 210, 215
Managed Address Configuration flag, Router Advertisement Message, 424
managed code, 429
management systems and infrastructure, examining for IPv6 technology gaps, 447
manufacturer ID. *See* company ID
Maximum Receive Unit (MRU), 531
  Point-to-Point Protocol (PPP) link, 113
Maximum Response Code field, modified Multicast Listener Query messages, 196
Maximum Response Delay field
  Multicast Listener Done messages, 194
  Multicast Listener Query messages, 190
  Multicast Listener Report messages, 191
maximum transmission units. *See* MTUs
MAX_RTR_SOLICITATIONS variable, 172
Message Body field, ICMPv6, 119
messages
  Neighbor Discovery (ND), 133
  summary of Neighbor Discovery (ND) options, 157
M flag. *See* Managed Address Configuration flag, Router Advertisement message; *See* Managed Address Configuration flag, Router Advertisement Message
Microsoft
  IP-HTTPS features in Windows Server 2012 and Windows 8, 375
  IP-HTTPS specification, 370
  IP-HTTPS Tunneling Protocol Specification, 375
Microsoft corporate network, IPv6 on, 437–450
  characteristics of Microsoft Corpnet, 438
  current deoployment of IPv6, 440–443
    example of address selection, 442
  deployment details, 444–445
    addressing plan and routing infrastructure, 444
    DirectAccess, 444
    security for IPv6 traffic, 445
  deployment philosophy, 439
  deployment planning and recommendations, 446–449
    deployment recommendations, 448
    determining IPv6 technology gaps, 446
    overall planning, 446
    training operations staff, 448
  history of IPv6 on the Corpnet, 438–439
  short and long-term plans for IPv6, 443
Microsoft Exchange Server
  IPv6-capable SMTP service, 30
Microsoft Network Monitor, 186
Microsoft Windows Software Development Kit (SDK), 428
migration of applications to IPv6, 428
  inventorying applications beforehand, 428
  scoping the work and scheduling migration, 429
MLD (Multicast Listener Discovery), 117, 183–204
  IPv6 multicast overview, 184
    host suport for multicast, 184
    router support for multicast, 185–189
  MLD and MLDv2 support in Windows, 201
  MLD messages and IGMP2 equivalents, 194
  MLDv2 messages, 195
    IGMPv3 messages and, 201
    modified Multicast Listener Query, 195–197
    Multicast Listener Report messages, 197–204
  MLDv2 packet structure, 194
  Multicast Listener Done messages, 193–195
  Multicast Listener Query messages, 189
  Multicast Listener Report messages, 191–193
  overview of MLD and MLDv2, 183
  packet structure, 188
  Router Aler option used for, 104
  support in IPv6 for Windows, 21
mobile broadband links, module in Windows Framing Layer, 18
mobile IPv6, 549–604
  attaching to home link, 582
  components, 549–551
  correspondent registration, 564–567
  data structures, 561–563
  Home Address destination option, 106
  host receiving algorithm, 600–602
  host sending algorithm, 597–599
  message exchanges, 567–582
  messages and options, 552–561
  mobile node changes home address, 591
  mobile node initiating communication with new correspondent node, 586–588
  moving from home link to foreign link, 583–586
  moving to new foreign link, 592–594
  new correspondnet node communicates with mobile node, 588–589
  returning home, 594–596
  Transport layer transparency, 551
Mobile Prefix Advertisement message, 558
  sent from mobile node, 581
Mobile Prefix Solicitation message, 557, 580
Mobility extension header and messages, 552–554

mobility, IPv6 support for, 15
More Fragments flag, 108, 110
MRU (Maximum Receive Unit), 531
  Point-to-Point Protocol (PPP) link, 113
Msg-Type field, 212
MTU field, MTU option in ND, 142
MTUs (maximum transmission units)
  for common LAN and WAN technologies, 516
  IPv6, 113
    for common LAN and WAN technologies, 113
  link MTU, 10
  link MTU of interface over which packet is forwarded, 122
  link MTU of next-hop interface, 263
  minimim IPv6 MTU, 120
  MTU option in ND, 141
    Router Advertisement messages, 149
  and Packet Too Big messages, 121
  path maximum transmission unit (PMTU), 128
  path MTU, 11
multicast addresses, 61
  interfaces on IPv6 host listening for traffic, 74
  interfaces on IPv6 router listening for traffic, 74
  IPv6, 68–72
    current list of permanently assigned IPv6 multicast addresses, 70
    mapping to Ethernet multicast addresses, 71
  IPv6 multicast address, 229
  IPv6, prefix, 184
  solicited-node address, 70
Multicast Address field, 189
  MLDv2 Multicast Listener Report messages, 199
  modified Multicast Listener Query messages, 196
  Multicast Listener Done messages, 194
  Multicast Listener Query messages, 190
  Multicast Listener Report messages, 191
Multicast Address Record 1 to n fields, 198
multicast-address-specific queries, 189
Multicast Listener Discovery. *See* MLD
Multicast Listener Done messages, 187, 193–195
  IPv6 header, 193
  structure of, 194
Multicast Listener Query messages, 187, 189
  MLDv2, 195
    structure of, 196
  structure of, 190
Multicast Listener Report messages, 187, 191–193
  example of MLD message, 192
  IPv6 header, 191
  MLD2, 197–204
    example, 200
    IPv6 header, 197
    multicast address record types, 200
    structure of, 198
    structure of multicast address record, 198
  structure of, 191
multicast promiscuous mode, 186
multicast routing, native IPv6 connectivity, 421

N

Name Query Request messages, 229, 246
Name Query Response messages, 229, 232, 247
name resolution, 227–252
  host name ISATAP to IPv4 address, 307
  for IPv6, 227–231
    DNS enhancements, 227–228
    LLMNR, 228–231
  source and destination address selection, 231–240
  support in Windows, 240–250
Name Resolution Policy Table. *See* NRPT
namespaces (DNS), stored in NRPT, 405
NAP (Network Access Protection) platform, 389
  additional protection for IPsec-protected communications, 390
NAT64, 296
NAT64/DNS64, 377–386, 400, 420
  configuration requirements, 379
  configuring in Windows Server 2012, 384
  for DirectAccess on Microsoft Corpnet, 441
  DirectAccess server on IPv4-only intranet acting as, 402
  DNS64, 377
  DNS name query and response, 380
  IPsec-protected traffic, inability to travel across NAT64, 403
  IPv6-to-IPv4 translation solution, 377
  IPv6 traffic from IPv6-only node, 382–386
    inital packet from IPv6-only node, 382
    response IPv4 packet from IPv4-only node, 383
    translated IPv4 packet to IPv4-only node, 382
    translated response packet to IPv6-only node, 384
  key differences from traditional IPv4-based protocol translation, 379
  limitations of, 378
  NAT64, 378
  using only at network edge for DirectAccess clients, 449

National Institute of Standards and Technology (NIST), information on IPv6 security, 396
native IPv6
  addressing allocation, 423
  connectivity, 421
  demonstrating native IPv6 connectivity, 498–501
  traffic over site-to-site VPN connections, 432
  using instead of transition technologies, 448
NAT (Network Address Translation), 1, 2–6, 350
  cone NAT, 612, 614, 617
    initial communication from IPv6-only host to Teredo client, 623
    initial communication from Teredo client to IPv6-only host, 619
  editors, 4
  IPv4-encapsulated IPv6 traffic and, 289
  IPv4 packet-filtering function of, 349
  multilayered NAT, Internet connectivity and, 347
  NAT64/DNS64 in Windows IPv6, 20, 28
  problem for peer-to-peer applications, 13
  requirement for NAT editors, 4
  restricted NAT, 613, 616, 618
    initial communication from IPv6-only host to Teredo client, 624
    initial communication from Teredo client to IPv6-only host, 621
  reuse of IPv4 private address space, 2
  Teredo and, 349
  Teredo clients maintaining NAT mapping, 610–611
  Teredo NAT traversal for IPv6 traffic, 348
  types of, 351
NAT-PT (Network Address Translation/Protocol Translation), 378
ND. *See* Neighbor Discovery
Neighbor Advertisement messages, 71, 133, 152–155, 159
  example, 155
  multicast, sent in duplicate address detection, 169
  reply to duplicate address detection Neighbor Solicitation message, 167
  sent during address autoconfiguration, 207
  sent in address resolution process, 161
  structure of, 153
neighbor cache, 158
  checking for entry matching next-hop address, 179, 261, 263
  entry states, 164
Neighbor Discovery (ND), 8, 117, 131–182
  address resolution process, 159–163
  conceptual host data structures, 158
  duplicate address detection, 167–170
  host sending algorithm, 179
  IPv4 neighbor messages and functions with IPv6 equivalents, 181
  IPv6 hosts obtaining address configurations via, 388
  message format, 133
  messages, 145
    changes to messages and options, mobile IPv6, 558
    Neighbor Advertisement messages, 152–155
    Neighbor Solicitation messages, 150–152
    Redirect messages, 127, 155–157
    Router Advertisement messages, 146
    Router Solicitation messages, 145
  messages and possible options, 518
  monitoring on IP-HTTPS server, 374
  MTU option, 141
  neighbor unreachability detection, 163–167
  option types, 518
  overview, 131
  processes, 158
  Redirected Header option, 139
  redirect function, 176–182
  Route Information option, 143
  router discovery, 170–175
  SEND (SEcure Neighbor Discovery), 388
  Source and Target Link-Layer Address options, 134–136
  summary of messages and options, 157
  support in IPv6 for Windows, 21
neighbors, 10
  displaying with netsh interface command, 53
  Get-NetNeighbor –AddressFamily IPv6 PowerShell command, 51
  unreachability detection in IPv6 for Windows, 21
Neighbor Solicitation messages, 70, 133, 150–152, 159
  example, 152
  multicast, sent in address resolution process, 160
  multicast, sent in duplicate address detection, 168
  sent during address autoconfiguration, 207
  structure of, 151
  Surce Link-Layer Address option in, 136
  Target Address field, 167

neighbor unreachability detection, 132, 158, 163–167, 279
  and dead gateway detection, 166
  example of exchange of ND messages to establish, 164
  ISATAP interfaces' support for, 321
  neighbor cache entry states, 164
.NET Framework, 32
NetBIOS, 35
  deprecating use of applications for IPv6-predominat intranet, 449
  resolving host name "ISATAP", 307
NetBIOS Name Query Response messages, 228
NetBIOS over TCP/IP (NetBT), 228
netsh interface
  6to4 host/router configuration, 332
  6to4 router configuration, 334
  adding DNS servers, 39
  complete list of commands, 39
  configuring default gateway, 38
  configuring IPv6 addresses, 37
  configuring IPv6 router to set M flag to 1 in router advertisements, 212
  configuring IPv6 router to set O flag to 1 in router advertisements, 212
  configuring ISATAP router, 320
  configuring static routing, 275–277
  creating configured tunnels for IPv6, 294
  disabling 6to4, 393
  disabling ISATAP, 392
  displaying IPv6 configuration, 51–53
    ipv6 show address command, 52
    ipv6 show destinationcache command, 53
    ipv6 show interface command, 51
    ipv6 show neighbors command, 53
    ipv6 show route command, 52
  enabling forwarding and advertising on ISATAP tunneling interface, 317
  enabling ISATAP on individual computers, 305
  enabling sending of router advertisements, 26
  enabling weak host model, 256
  IP-HTTPS client configuration, 374
  ipv6 add dnsserver command, 24, 241
  ipv6 add|set|delete prefixpolicy commands, 291
  ipv6 add|set route commands, 26
  ipv6 install command, 430
  ipv6 set global mldversion=version2, 201
  ipv6 set global mldversion=version3, 202
  ipv6 set global multicastforwarding=enable, 202
  ipv6 set global randomizeidentifiers=disabled command, 23, 83
  ipv6 set global sourceroutingbehavior command, 108
  ipv6 set interface InterfaceNameOr Index forwarding=enabled command, 26
  ipv6 set privacy enabled command, 23, 221
  ipv6 show destinationcache command, 261
  ipv6 show interface command, 45
  ipv6 show neighbors command, 261
  ipv6 show prefixpolicies command, 245
  ipv6 show route command, 256, 257
    example routing table entries, 258
  isatap set router AddressOrName command, 318
  isatap set router command, 312
  isatap set state disabled command, 303
  manual configuration of IPv6 settings, 33
  portproxy add v6tov4 command, 298
  randomized interface IDs, disabling on servers, 431
  Teredo client state configuration, 355, 393
  Teredo client UDP port configuration, 356
  Teredo refresh rate, setting, 354
  Teredo relay configuration, 366
  Teredo server configuration, 365
    name or first IPv4 address, 366
Netstat tool, 47
Network Access Protection (NAP) platform, 389
  additional protection for IPsec-protected communications, 390
Network Address Translation. *See* NAT
Network Address Translation/Protocol Translation (NAT-PT), 378
Network Connections folder, 18, 241
  DNS dynamic update for IPv6, 244
  IPv6 installed and enabled by default, 20
  manually configuring IPv6 settings via, 34
  Sharing tab, enabling ICS, 332
Network Device Interface Specification (NDIS), 18
network hardware, inventorying for IPv6 technology gaps, 447
networking applications
  updating on Windows for IPv6 connectivity, 20
  Windows support for operation over IPv6, 30
Network layer
  implementation in Windows, 18
  OSI IS-IS protocol suite, 271
Network Location Awareness (NLA) service, 407

network location detection, 406
  process, summary of, 407
network management infrastructure support for IPv6, 415
Network Monitor, 186
networks
  definition in IPv6, 10
  elements of IPv6 network, 9
  IPv6-capable network and IPv4 and IPv6 Internet, 11
  IPv6-only network on Windows, capabilities of, 30
network sniffers, 186
New Host dialog box for AAAA records, 242
New-NetRoute Windows PowerShell cmdlet, 37
Next Header field, IPv6 header, 95, 266
  Authentication Header, 112
  chain of pointers formed by, 100
  Encapsulating Security Payload (ESP) header and trailer, 112
  Fragment header, 108, 110, 111
  identifying ICMPv6 header, 118
  Routing header, 107
  typical values for, 515
  typical values of, 96
  upper-layer checksums or pseudo-headers, 114
next-hop address, 179, 253, 255
  determining from route in routing table, 256, 258
  obtaining from destination cache, 261
next-hop determination, 132
next-hop interface, 253, 255
  obtaining from destination cache, 261
nibble boundaries, subnetting along, 75, 81
NI_DGRAM flag, 545
NI_NAMEREQD flag, 544
NI_NOFQDN flag, 544
NI_NUMERICHOST flag, 544
NI_NUMERICSERV flag, 544
NLA (Network Location Awareness) service, 407
nodes
  defined, 10
  identifying in IPv6 multicast addresses, 70
  solicited-node multicast address, 70
  types of, 284
NRPT (Name Resolution Policy Table), 249, 444
  rules for DirectAccess, 406
  rules for intranet namespace, on DirectAccess client, 408
  using with DirectAccess, 405
    NRPT exemptions, 406
NUD. *See* neighbor unreachability detection
Number of Multicast Address Records field, MLDv2 Multicast Listener Report messages, 198
Number of Sources field
  MLDv2 Multicast Listener Report messages, 199
  modified Multicast Listener Query messages, 197

O

obscured external address, Teredo addresses, 357
obscured external port, Teredo addresses, 357
Obscured Origin Address field (Teredo), 363
Obscured Origin Port Number field (Teredo), 363
O flag. *See* Other Stateful Configuration flag, Router Advertisement message
On-Link flag, Prefix Information option, 137, 208, 211, 215, 274, 424
Opcode field, 230
Open Shortest Path First (OSPF), 269
Open Shortest Path First version 3 (OSPFv3), 444
Option-Code field, 213
Option-Data field, 213
Option-Len field, 213
Option Length field, 102
  Home Address destination option, 106
options
  Neighbor Discovery (ND), 134
  Options field, IPv4 header, 94
  summary of Neighbor Discovery (ND) options, 157
  types for Hop-by-Hop and Destination Options headers, 107
  unrecognized options, ICMPv6 error message for, 124
Options field, DHCPv6 messages, 212
Option Type field, 102
  Home Address destination option, 106
  settings of two high-order bits, 124
  values of two high-order bits in, 103
Origin indicator, Teredo packets, 361, 362
  structure and fields in, 362
OSPF for IPv6, 271
OSPF (Open Shortest Path First), 269
Other Stateful Configuration flag, Router Advertisement message, 148, 208, 211, 424
Override flag, Neighbor Advertisement messages, 154

P

Packet Layer Protocol (PLP), 532
packets
  basic structure of IPv6 packets, 519
  definition in IPv6, 10
  fragmentation process in IPv6, 109
  fragment, reassembly process in IPv6, 111
  MLD message packet structure, 188
  MLDv2 packet structure, 194
  more efficient forwarding in IPv6, 14
  protection of IPv6 packets, 389
  structure of IPv6 packet, 91
  Teredo, formats of, 360–363
  types in dual IP layer architecture, 287
  types in dual-stack architecture, 287
Packet Too Big messages, 119, 121, 128, 263
  comparison of ICMPv6 and ICMPv4, 127
  example, displayed by Network Monitor Capture, 122
  structure of, 121
Pad1 option, 103
Padding option, 193
PadN option, 103, 105
parameter discovery, 132
Parameter Problem messages, 119, 123, 517
  comparison for IPv4 and IPv6, 127
  structure of, 123
  Unrecognized IPv6 Option Encountered message, 124
path MTU, 11, 110, 128
  changes in, 128
  IPv6-over-IPv4 tunneling, 289
Path MTU Discovery process, 113, 127
  changes in, 128
  changes in PMTU, 128
  Packet Too Big messages, 122
path names, UNC (Uniform Naming Convention), 24
Pathping tool, 46
  display on computer running Windows Server 2012, 46
path vector–based routing protocols, 270
Payload Length field, IPv6 header, 95, 111
  Authentication header, 112
PDU (protocol data unit), 10
Peer-Address field, 214
Peer Name Resolution Protocol. *See* PNRP
peer-to-peer applications
  IPv4 NATs as problem for, 13
  problem with NATs, 5
  Windows Peer-to-Peer Networking platform, 25
permanent interface identifiers, 83
PF_INET6 constant, 540
physical interfaces, 10
PIM (Protocol Independent Multicast), 188
Ping tool, 44
  command-line options supporting IPv6, 44
  creating fragmented IPv6 traffic with, 111
  testing connectivity between 6to4 hosts, 326
  testing connectivity for ISATAP hosts, 304
  zone ID: ping fe80::2b0:d0ff:fee9:4143%3, 66
PKI (public key infrastructure), computer certificates in DirectAccess, 408
PLP (Packet Layer Protocol), 532
PMTU. *See* path MTU
PNRP (Peer Name Resolution Protocol), 19, 248, 336
  resource for further information, 249
  support in IPv6 for Windows, 25
Pointer field, Parameter Problem messages, 124
pointer (PTR) resource records, 9, 228, 242, 291
  DNS infrastructure support for, 422
  DNS servers' support of IPv6 PTR records and dynamic updates, 430
  querying and processing of, IPv6 in Windows, 24
  registration via DNS dynamic update, 244
pointer queries, 228
point-to-point router links, 127-bit prefixes on, 73
Point-to-Point Tunneling Protocol. *See* PPTP
Portion of Redirected Packet field, 140
PortProxy, 20, 28, 297, 421, 429
  configuring, 298
  protocols it works with, 299
port scanning, 391
port translation in NATs, 4
port translation or proxy solutions, 429
PowerShell. *See* Windows PowerShell
  IP-HTTPS client configuration, 374
PPP over Ethernet (PPPoE), 26
PPP (Point-to-Point Protocol), 10, 530
  interface identifiers generated when configuring interfaces, 83
  IPv6 over PPP in Windows, 26
  module in Windows Framing Layer, 18
  MRU (Maximum Receive Unit) of link, 113
PPTP (Point-to-Point Tunneling Protocol), 4, 289
  site-to-site VPN connections with, 432
  support by IPV6CP in built-in remote access client, 26
Preferred Lifetime field, Prefix Information option, 138
Preferred state, autoconfigured addresses, 206

prefix discovery, 132
prefixes
  NAT64/DNS64 configured with, 380
  prefix list, 158
  Teredo prefix, 356
prefixes for IPv4 addresses, 64
prefixes for IPv6 addresses, 60
  global address prefix, 416
  global and unique local prefixes for specific subnet, 67
  IPv6 address blocks with prefix length smaller than 48 bits, 64
  multicast address prefix, 184
  solicited-node multicast address, 70
  subnetted addresses, 75
    enumerating prefixes for, 76–81
  unique local addresses, 416
  unique local unicast addresses, 66
Prefix field
  Prefix Information option, 138
  Route Information option, 144
Prefix Information option, 136–139
  Autonomous flag, 274, 275, 277
  example, displayed by Network Monitor Capture, 139
  fields in, 137
  modification for mobile IPv6, 559
  On-link flag, 274
  On-Link flag, 211
  Router Advertisement messages, 149, 208
Prefix Length field
  Prefix Information option, 137
  Route Information option, 143
prefix policy table, 232, 291
  adding, modifying, and removing entries, 246
  changing entries in, 245
  default, 245
printers
  DHCPv6-reserved addresses for, 423
  network printer, UNC path name for, 24
prioritized delivery of data
  better support in IPv6, 7
  IPv6 traffic, 425
  limitations of IPv4, 2
private addresses, 4
PROBE state, 166
promiscuous mode, 186
protocol analyzers, 186
protocol data unit (PDU), 10
Protocol field, IPv4, 94, 288, 290
  setting to 41 for ISATAP and 6to4 traffic, 393
  set to 41, NAT translation and, 348
Protocol Independent Multicast (PIM), 188
protocols
  IPv6 routing protocols, 268–272
  multicast routing protocols, 188
Provider Assigned (PA), 416
Provider Independent (PI) space from regional registry, 416
proxied name queries, using NAT64/DNS64, 380
pseudo-headers in IPv6, 114
PTR records. *See* pointer (PTR) resource records
public and private addresses
  disjoint address space in IPv4, 12
  global scope for, in Windows systems, 244
  IPv4 public address of Teredo server, 357
  IPv6 global addresses as equivalent to IPv4 public addresses, 62
  NAT mapping of private addresses to public IPv4 addresses, 350
  NATs and peer-to-peer applications, 5
  public addresses, 4
    random interface IDs for IPv6 addresses on Windows, 23
  public IPv4 address, use by 6to4, 347
  scarcity of IPv4 public addresses, 1
public key infrastructure (PKI), computer certificates in DirectAccess, 408
public topology, global unicast addresses, 64

Q

QoS (Quality of Service), 2, 426
QQIC (Querier's Query Interval Code) field, 197
QR (Query/Response) flag, 230
QRV (Querier's Robustness Variable) field, 197
qualified state, Teredo client, 354
  configuring Teredo client to be in, 356
Quality of Service (QoS), 2, 426
Question Type field, 228

R

random number for Global ID of unique local prefixes, 416
RA (Router Advertisement) Guard, 389
rate limitations
  Redirect messages, 176

rate-limiting ICMPv6 error messages, 119
R bit in Teredo addresses, 357
RCODE (Return Code) field, 231
REACHABLE state, 165
Reachable Time field, Router Advertisement messages, 149
reassembly process for fragment packets, 111
Record Type field, 199
recvfrom( ) function, 542
Redirected Header option, Redirect messages, 156
redirect function, 133, 158, 176–182
- example, Part 1, 177
- example, Part 2, 177
- example, Part 3, 178
- steps in IPv6 process, 176

Redirect messages, 155–157, 176
- example, 157
- ICMPv6, 133
- IPv4, 131
- options, 156
- Redirected Header option, 139
- sent by router in redirect process, 178
- structure of, 156

reference tables for IPv6, 515–518
registry
- DisabledComponents, 40, 419
- disabling 6to4, 393
- disabling automatic tunneling, 419
- disabling ISATAP, 303, 392
- network location URL stored in, 407
- selectively disabling IPv6 components and configuring behaviors, 40

relay router. *See* relays, 6to4
relays
- 6to4, 327
  - determining IPv4 address for, 329
  - DirectAccess server as, 408
  - routes used by, 331
- Teredo, 353
  - configuring, 366
  - routes, 363

remote procedure call (RPC), 31
remote subnet or network routes, 254
Requests for Comments. *See* RFCs
Reserved1 field
- Prefix Information option, 138
- Route Information option, 143

Reserved2 field
- Prefix Information option, 138
- Route Information option, 144

Reserved field, 112
- MLDv2 Multicast Listener Report messages, 198
- modified Multicast Listener Query messages, 196
- MTU option, 142
- Multicast Listener Done messages, 194
- Multicast Listener Query messages, 190
- Multicast Listener Report messages, 191
- Neighbor Advertisement messages, 154
- Neighbor Solicitation messages, 151
- Redirected Header option, 140
- Redirect messages, 156
- Router Advertisement messages, 148
- Router Solicitation messages, 146

Resource ReSerVation Protocol (RSVP), 104
restricted NATs, 351, 613, 616, 618
- initial communication from IPv6-only host to Teredo client, 624
- initial communication from Teredo client to IPv6-only host, 621

Retransmission Timer field, Router Advertisement messages, 149
Return Code (RCODE) field, 231
Return Routability procedure in mobile IPv6, 565
reverse queries, 228
RFC 791, “Internet Protocol”, 114, 115
RFC 1035, “Domain Names - Implementation and Specification”, 229, 250
RFC 1191, “Path MTU Discovery”, 127, 129
RFC 1256, “ICMP Router Discovery Messages”, 170, 181
RFC 1356, “Multiprotocol Interconnect on X.25 and ISDN in the Packet Mode”, 538
RFC 1661, “The Point to Point Protocol (PPP)”, 538
RFC 1662, “PPP in HDLC-like Framing”, 538
RFC 1723, “RIP Version 2”, 280
RFC 1752, “The Recommendation for the IP Next Generation Protocol”, 283, 299
RFC 1981, “Path MTU Discovery for IP version 6”, 128, 129
RFC 2080, “RIPng for IPv6”, 279, 280
RFC 2236, “Internet Group Management Protocol, Version 2”, 183, 202
RFC 2328, “OSPF Version 2”, 279, 280
RFC 2460, “Internet Protocol, Version 6 (IPv6)”, 21, 53, 94, 115, 129, 603
- extension headers, 99
- MTU size, 113
- order of extension headers, 101
- reassembly time for fragment packets, 112
- Routing Tyoe 0, 108

RFC 2464, “Transmission of IPv6 Packets over Ethernet Networks”, 135, 181, 538
RFC 2467, “Transmission of IPv6 Packets over FDDI Networks”, 538
RFC 2470, “Transmission of IPv6 Packets over Token Ring Networks”, 538
RFC 2474, “Definition of the Differentiated Services Field (DS Field)”, 95, 115
RFC 2491, “IPv6 over Non-Broadcast Multiple Access (NBMA) Networks”, 538
RFC 2492, “IPv6 over ATM Networks”, 538
RFC 2526, “Reserved IPv6 Subnet Anycast Addresses”, 603
RFC 2529, “Transmission of IPv6 over IPv4 Domains without Explicit Tunnels”, 299
RFC 2545, “Use of BGP-4 Multiprotocol Extensions for IPv6 Inter-Domain Routing”, 279, 280
RFC 2590, “Transmission of IPv6 Packets over Frame Relay Networks Specification”, 538
RFC 2675, “IPv6 Jumbograms”, 103, 104
RFC 2710, “Multicast Listener Discovery (MLD) for IPv6”, 21, 53, 183, 202
RFC 2711, “IPv6 Router Alert Option”, 202
RFC 3022, “The IP Network Address Translator (NAT)”, 350, 367
RFC 3056, “Connection of IPv6 Domains via IPv4 Clouds”, 221, 224, 285, 299, 323, 328, 345
RFC 3068, “An Anycast Prefix for 6to4 Relay Routers”, 345
RFC 3118, “Authentication for DHCP Messages”, 388, 395
RFC 3168, “The Addition of Explicit Congestion Notification (ECN) to IP”, 21, 53, 95, 115
RFC 3306, “Unicast-Prefix-based IPv6 Multicast Addresses”, 89
RFC 3315, “Dynamic Host Configuration Protocol for IPv6 (DHCPv6)”, 26, 27, 210, 212, 224, 388, 395
RFC 3376, “Internet Group Management Protocol, Version 3”, 183, 202
RFC 3484, “Default Address Selection for Internet Protocol version 6 (IPv6)”, 53, 54, 232, 250, 291, 299
RFC 3493, “Basic Socket Interface Extensions for IPv6”, 53, 54, 547
RFC 3531, information on managing bits of IPv6 address block, 82, 89
RFC 3542, “Advanced Sockets Application Program Interface (API) for IPv6”, 53, 54, 547
RFC 3587, “IPv6 Global Unicast Address Format”, 62, 89
RFC 3596, “DNS Extensions to support IP version 6”, 250
RFC 3627, discouraging use of 127-bit prefixes on point-to-point router links, 73
RFC 3646, “DNS Configuration Options for Dynamic Host Configuration Protocol for IPv6 (DHCPv6)”, 224, 241, 250
RFC 3697, “IPv6 Flow Label Specification”, 95, 115
RFC 3736, “Stateless Dynamic Host Configuration Protocol (DHCP) Service for IPv6”, 27, 54
RFC 3776, “Using IPsec to Protect Mobile IPv6 Signaling Between Mobile Nodes and Home Agents”, 603
RFC 3810, “Multicast Listener Discovery Version 2 (MLDv2) for IPv6”, 53, 54, 183, 196, 202
RFC 3879, “Deprecating Site Local Addresses”, 62, 89
RFC 3927, “Dynamic Configuration of IPv4 Link-Local Addresses”, 88, 89
RFC 3956, “Embedding the Rendezvous Point (RP) Address in an IPv6 Multicast Address”, 88, 89
RFC 3971, “SEcure Neighbor Discovery (SEND)”, 388, 395
RFC 3986, “Uniform Resource Identifier (URI): Generic Syntax”, 25, 54
RFC 4007, “IPv6 Scoped Address Architecture”, 65, 89
RFC 4033, “DNS Security Introduction and Requirements”, 249, 250
RFC 4034, “Resource Records for the DNS Security Extensions”, 249, 250
RFC 4035, “Protocol Modifications for the DNS Security Extensions”, 249, 250
RFC 4191, “Default Router Preferences and More-Specific Routes”, 21, 54, 134, 143, 181, 279, 280, 388, 395
RFC 4193, “Unique Local IPv6 Unicast Addresses”, 66, 89
RFC 4213, “Transition Mechanisms for IPv6 Hosts and Routers”, 284, 291, 294, 299
RFC 4271, “A Border Gateway Protocol 4 (BGP-4)”, 279, 280
RFC 4291, “IP Version 6 Addressing Architecture”, 21, 54, 61, 62, 63, 89
    anycast addresses, 73
    interface IDs for LAN interfaces, 82
    Scope field values in multicast addresses, 69
RFC 4301, “Security Architecture for the Internet Protocol”, 389, 395
RFC 4302, “IP Authentication Header”, 112, 115, 389, 395

RFC 4303, “IP Encapsulating Security Payload (ESP)”, 112, 115, 389, 395
RFC 4380, “Teredo: Tunneling IPv6 over UDP through Network Address Translations (NATs)”, 285, 299, 347, 367, 625
RFC 4443, “Internet Control Message Protocol (ICMPv6) for the Internet Protocol Version 6 (IPv6)”, 21, 54, 117, 124, 129
RFC 4760, “Multiprotocol Extensions for BGP-4”, 279, 280
RFC 4787, “Network Address Translation (NAT) Behavioral Requirements for Unicast UDP”, 378, 386
RFC 4795, “Link-local Multicast Name Resolution (LLMNR)”, 228, 229, 250
RFC 4861, “Neighbor Discovery for IP Version 6 (IPv6)”, 21, 54, 134, 158, 181, 388, 395, 603
  states for neighbor cache entries, 165
RFC 4862, “IPv6 Stateless Address Autoconfiguration”, 21, 54, 181, 207, 208, 220, 224
RFC 4884, “Extended ICMP to Support Multi-Part Messages”, 118, 129
RFC 4941, “Privacy Extensions for Stateless Address Autoconfiguration in IPv6”, 83, 87, 89
RFC 5072, “IP Version 6 over PPP”, 26, 54, 83, 89, 538
RFC 5095, deprecation of Routing Type 0 header, 108
RFC 5214, “Intra-Site Automatic Tunnel Addressing Protocol (ISATAP)”, 221, 224, 285, 299, 301, 321
RFC 5308, “Routing IPv6 with IS-IS”, 279, 280
RFC 5340, “OSPF for IPv6”, 280
RFC 5382, “NAT Behavioral Requirements for TCP”, 378, 386
RFC 5508, “NAT Behavioral Requirements for ICMP”, 378, 386
RFC 5991, “Teredo Security Updates”, 357, 367
RFC 6052, “IPv6 Addressing of IPv4/IPv6 Translators”, 386
RFC 6105, “IPv6 Router Advertisement Guard”, 389, 395
RFC 6146, “Stateful NAT64: Network Address and Protocol Translation from IPv6 Clients to IPv4 Servers”, 296, 299, 377, 386
RFC 6147, “DNS64: DNS Extensions for Network Address Translation from IPv6 Clients to IPv4 Servers”, 296, 299, 377, 386
RFC 6164, “Using 127-Bit IPv6 Prefixes on Inter-Router Links”, 73, 89
RFC 6275, “Mobility Support in IPv6”, 106, 115, 134, 137, 181, 603
RFC 6286, “Autonomous-System-Wide Unique BGP Identifier for BGP-4”, 280
RFC 6343, “Advisory Guidelines for 6to4 Deployment”, 335, 345
RFC 6555, “Happy Eyeballs: Success with Dual-Stack Hosts”, 20, 24, 54, 250
RFCs
  IETF Request for Comments webpage, 455
  IPv6 RFC index, 451–456
RIP Next Generation (RIPng) for IPv6, 270
RIP (Routing Information Protocol), 269
Route Information option, 143, 149
Route Lifetime field, Route Information option, 144
route print command, 256, 259
  example routing table entries, 260
Router Address flag, Prefix Information option, 137, 560
Router Advertisement messages, 133, 146, 253, 276
  Advertisement Interval option, 560
  advertising 6to4 router as default to ISATAP hosts, 343
  Current Hop Limit field, 275, 277
  Default Router Preference field, 275, 277
  example, 150
  IPv4-encapsulated, example of, 309
  Managed Address Configuration flag, 208, 210
  mobile IPv6 modifications to, 559
  MTU option, 141
  multicast, sent during router discovery, 173–175
  On-Link and Autonomous (A) flags, 424
  options, 149
  Other Stateful Configuration flag, 208, 211
  Prefix Information option, 136–139
  receives in address autoconfiguration, 208, 215
  receiving from ISATAP router, 222
  Route Information option, 143
  Router Information option, 274
  Router Lifetime field, 171, 275
  sent in address autoconfiguration, 205
  sent in router discovery process, 171
  structure of, 147
Router Advertisement (RA) Guard, 389
Router Alert option, 104, 105, 193
router discovery, 132, 158, 170–175
  configurations performed by, 171
  example, Part 1, 172
  example, Part 2, 173–175
  for ISATAP hosts, 306–312
  performing with ISATAP router, 308
  processes in IPv6, 171

Router flag, Neighbor Advertisement messages, 154
Router Information option, 274
Router Lifetime field, Router Advertisement messages, 148, 171, 275, 277
routers
- 6to4, 327
  - addressing example, 329
  - manual configuration, 333
  - routes used, 330
  - Windows support for, 332
- computer running Windows as static unicast IPv6 router, 431
- configuring ISATAP router, 317–320
  - using Netsh commands, 320
  - using PowerShell commands, 318
- defined, 10
- firewalls based in, 425
- host/router, 6to4, 327
- inventorying for IPv6 technology gaps, 447
- IPv6/IPv4, 6to4 relay, 327
- IPv6 traffic over site-to-site VPN connections, 432
- ISATAP, 304
  - becoming routing bottleneck, 417
- ISATAP Router Name Group Policy setting, 311
- preventing rogue IPv6 routers, 389
- readying for IPv6 unicast traffic, 431
- readying for multicast IPv6 traffic, 431
- support for IPv6 muilticast routing, 421
- Teredo relay, 353

Router Solicitation messages, 133, 145
- IPv4-encapsulated, example of, 309
- multicast, sent during router discovery, 172
- sent by ISATAP host to 6ot4 router, 342
- sent in address autoconfiguration, 205, 208
- sent in router discovery process, 171
- sent to ISATAP router, 306

router-to-host tunneling, 292
router-to-router tunneling, 292
routing, 253–282
- 6to4, 330
- asymmetric routing paths, 445
- critical router loop to determine packet forwarding, 98
- data needing fragmentation discarded by routers, 108
- dead gateway detection, 278–279
- default router list, 158
- DirectAccess and IPv6 routing, 401–403
- displaying list of routes in IPv6 routing table, 50
- displaying routes in IPv6 routing table with netsh, 52
- end-to-end IPv6 delivery process, 260–267
  - IPv6 on destination host, 265–267
  - IPv6 on sending host, 260–262
  - IPv6 on the router, 262–265
- enhancements in IPv6 for Windows, 21
- global routing prefix, 63
- Hop Limit Exceeded in Transit, Code field of Time Exceeded message, 123
- host informing local subnet routers that it's listening for multicasts, 185
- in IPv6, 253
  - example routing table for Windows, 256
  - route determination process, 254
  - routing table, 253
  - routing table entry types, 254
  - strong and weak host behavior, 255
- IP-HTTPS client, 373
- IPv6 addresses for a router, 74
- IPv6 on intranets, 416
- IPv6 router optimization, 102
- IPv6 routing protocols, 268–272
  - BGP-4, 272
  - distance vector protocols, 269
  - Integrated IS-IS for IPv6, 271
  - link-state based protocols, 269
  - OSPF for IPv6, 271
  - overview of dynamic routing, 268
  - path-vector based protocols, 270
  - RIPng for IPv6, 270
- IPv6 routing RFCs, 454
- ISATAP, 313
  - ISATAP host to IPv6 host communication, 316
- multicast, native IPv6 connectivity, 421
- plan for IPv6 deployment on Microsoft Corpnet, 444
- prefixes for IPv6 summarized routes, 60
- preventing forwarding of tunneled IPv6 traffic, 393
- route metrics, Advanced TCP/IP Settings dialog, 36
- router discovery (ICMP), replacement by Neighbor Discovery, 131
- router discovery, IPv6 interface IDs via, 87
- routers identified in multicast IPv6 addresses, 70
- router support for multicast in IPv6, 185–189
  - communicating group membershipt to other multicast routers, 188
  - forwarding IPv6 multicast traffic, 186

querying attached subnets for host membership, 187
receiving all IPv6 multicast traffic, 186
receiving and processing Lstener report and messages, 187
Route tool, IPv6-enabled, 43
security considerations for IPv6 nodes, 388
static routing with IPv6 for Windows, 26, 272–278
configuring with Netsh tool, 275–277
configuring with Windows PowerShell, 272–275
configuring wtih Routing and Remote Access, 277
Subnet-Router anycast address, 73
Teredo, 363–365
intersite Teredo client destinations, 365
IPv6 Internet destinations, 365
on-link Teredo client destinations, 364
unicast, with native IPv6 connectivity, 421
uses of Neighbor Discovery by routers, 132
Routing And Remote Access Server Setup Wizard, 216
Routing and Remote Access service, 424
Routing and Remote Access snap-in, 217
configuring static routing with, 277
Routing header, 99, 101, 107, 109
Destination Options header and, 106
structure of, 107
Routing Information Protocol (RIP), 269
Routing Type field, 107
RPC (remote procedure call), 31
RSVP (Resource ReSerVation Protocol), 104

S

SAs (Security Associations), 389
scope
different scopes, LAN interface on IPv6 host, 231
Scope field in IPv6 multicast addresses, 69
Scope field for IPv6 multicast addresses, 515
scope ID. *See* zone ID
scope, IPv6 addresses, 14
Secure Neighbor Discovery (SEND), 117
Secure Socket Tunneling Protocol (SSTP), 370
security
analyzing for IPv6 technology gaps, 448
host-based, and IPv6 traffic, 424
for IPv6 traffic on Microsoft Corpnet, 445
IPv6 support for, 15
requirement for Internet layer, 2
systems monitoring network traffic, IPv6 functionality, 415
Teredo, 357
security association (SA), 112, 389
security considerations for IPv6, 387–396
aspects of security for IPv6 traffic, 387
authorizations for automatically assigned addresses and configurations, 388
controlling traffic exchange with Internet, 393
control of tunneled traffin on intranet, 391
preventing rogue IPv6 routers, 389
protection from scanning and attacks, 390
protection of IPv6 packets, 389
Security Parameters Index (SPI) field, 112
Segments Left field, 107
SEND (SEcure Neighbor Discovery), 388
Sequence Number field, 112
Echo Request messages, 125
Server Manager administrative tool, 216, 218
servers
DirectAccess
infrastructure tunnel to, 410
on IPv4 Internet, routes for, 403
routes for, 402
settings for DirectAccess behavior, 408
disabling randomized interface IDs, 431
DNS, support for operation over IPv6, 430
inventorying IPv6 technology gaps on, 446
IP-HTTPS, 371
DAD proxy on, 374
Neighbor Discovery monitoring on, 374
Teredo, 352
configuring Windows-based server, 365
IPv4 address, 357
routes, 363
Set-DnsClientServerAddress Windows PowerShell cmdlet, 24, 37, 241
Set-NetIPv6Protocol MldVersion=2 command, 201
Set-NetIPv6Protocol MldVersion=3 command, 202
Set-NetIPv6Protocol -RandomizeIdentifiers Disabled command, 83
Set-NetIPv6Protocol –SourceRoutingBehavior command, 108
Set-NetIPv6Protocol Windows PowerShell cmdlet, 23
disabling generation of random interface IDs, 23
Set-NetTCPSetting -EcnCapability Enabled Windows PowerShell command, 21

S flag, modified Multicast Listener Query messages, 197
shunt functionality, Teredo relay on Windows Server 2012, 366
sites, 11
  number able to connect to IPv6 Internet, 64
site topology, global unicast addresses, 64
SLAAC. *See* stateless address autoconfiguration
SMTP (Simple Mail Transfer Protocol), IPv6 capabilities on Windows, 30
SNAP (Sub-Network Access Protocol) encapsulation
  ATM SNAP encapsulation of IPv6 packets, 536
  IEEE 802.3, 520, 521–523
  IEEE 802.5, 526
SO 10589, 280
Sockets API RFCs, 452
software requirements for IPv6 test lab, 489
Solicited flag, Neighbor Advertisement messages, 154
solicited-node address, 70
  for each unicast address assigned to interface on IPv6 host, 74
  for each unicast address assigned to interface on IPv6 router, 75
Source Address 1 to n fields
  MLDv2 Multicast Listener Report messages, 199
  modified Multicast Listener Query messages, 197
Source Address field
  Ethernet header of Neighbor Advertisement messages, 153
  Ethernet header of Neighbor Solicitation messages, 150
  Ethernet header of Redirect messages, 155
  Ethernet header of Router Advertisement messages, 147
  Ethernet header of Router Solicitation messages, 145
  IPv4, 94, 114
  IPv6 header
    LLMNR query messages, 246
    MLDv2 Multicast Listener Report message, 197
    modified Multicast Listener Query message, 195
    Multicast Listener Done message, 193
    Multicast Listener Query message, 189
    Multicast Listener Report message, 191
    Neighbor Advertisement message, 153
    Neighbor Solicitation message, 151, 169
    Redirect message, 155
    Router Advertisement message, 147
    Router Solicitation message, 145
  IPv6 pseudo-headers, 114
Source Address field, IPv6 header, 95, 111
source address selection algorithm, 232, 233–235
source and destination addresses, 255
  anycast addresses as destination addresses, 73
  destination MAC addresses on Ethernet networks, 71
  IPV4 versus IPv6, 8
  mapping of destination IPv6 and MAC multicast addresses, 193
  multicast addresses, 68
  in NAT translation for IPv4, 3
  selection in IPv6 for Windows, 24
  selection of, 231–240
    destination address selection algorithm, 235–237
    example of using address selection, 238–240
    for typical IPv4-only hosts, 231
    for typical IPv6 hosts, 232
    prefix policy table, 232
    source address selection algorithm, 233–235
    Winsow support for, 244–246
Source field, 288
Source Link-Layer Address option, 134–136
  example, displayed by Network Monitor Capture, 136
  Neighbor Solicitation messages, 151, 160
  Router Advertisement messages, 149
  Router Solicitation messages, 146
SPI (Security Parameters Index) field, 112
SSL (Secure Sockets Layer)
  HTTP over (HTTPS), 369
  termination of sessions for IP-HTTPS server, 371
  use in network location detection process, 407
SSTP (Secure Socket Tunneling Protocol), 370
stacks
  basic IPv6 stack support on Windows, 21
  dual-stack architecture, 287
  IPv6 enhancements on Windows, 21
STALE state, 165
stateful address autoconfiguration, 7, 206, 423
  interface identifiers assigned during, 83
  support by DHCPv6 on Windows, 27
stateful message exchange, DHCPv6, 215
stateless address autoconfiguration, 7, 148, 205, 423
  support by DHCPv6 on Windows, 27
  sutoconfiguration support in IPv6 for Windows, 21
  use on Microsoft corporate network, 444

stateless message exchange, DHCPv6, 215
static mappings, NAT, 350
static routing, 268
  with IPv6 for Windows, 272–278
  support in IPv6 for Windows, 26
strong host model, 255
  for sending and receiving in IPv6 for Windows, 22
Subnet ID field
  global and unique local unicast addresses, 67
  global unicast addresses, 63
subnet masks in IPv4, 60
subnet or network routes
  directly attached, 254
  remote, 254
Subnet-Router anycast address, 73
  interfaces of IPv6 router, 74
subnets
  addressing for IPv6, 416
  determining boundaries of IPv6 subnets, 416
  ISATAP, 304
  prefixes for IPv6 subnets, 60
  prefix length, configuring in Windows, 35
subnetting IPv6 address space, 75–81
  address allocation strategies, 81
  determining number of subnetting bits, 75
  enumerating subnetted address prefixes, 76–81
Sub-Network Access Protocol encapsulation. *See* SNAP encapsulation
switched environments (Layer 2), MTU option, 141
switching hardware, inventorying for IPv6 technology gaps, 447
symmetric NATs, 351

T

Target Address field
  Neighbor Advertisement messages, 154
  Neighbor Solicitation messages, 151, 167
  Redirect messages, 156, 176
Target Link-Layer Address option, 135
  Neighbor Advertisement messages, 154
  Redirect messages, 156
TCAM (Ternary Content Addressable Memory) entries, 447
TCP/IP
  architecture for Windows, 18
  installing Microsoft version 6 protocol component, 430
  in Windows systems, 286
Tcpip.sys file, 18
TCP/IPv6
  configuring IPv6 through properties of, 34
  manual configuration of IPv6 through properties of, 33
  Properties dialog box, 34
    General tab, 34
TCP (Transmission Control Protocol), 2, 286
  implementation in Windows, 17
  IP-HTTPS client connection, 372
  NAT64 providing basic translation for, 378
TCP/UDP port numbers, mapping in IPv4 NATs, 4
TC (Truncation) flag, 231
temporary addresses, 231
  display by ipconfig, 43
  enabling use of, 221
  interface identifiers, 83, 87
  support in IPv6 for Windows, 23
Tentative state, autoconfigured addresses, 206
Tentative (T) flag, 231
Teredo, 20, 295, 347–368, 391
  addresses, 285, 356–359
    addressing example, 358
  benefits of using, 348
  centrally configuring Group Policy settings for, 29
  client, 352
  client and host-specific relay in Windows, 354–356
  components, 351
  configuring to control traffic from Internet, 394
  DirectAccess server as server for, 408
  host-specific relay, 353
  NATs (Network Address Translators), 350
  packet formats, 360–363
  and protection from unsolicited incoming IPv6 traffic, 349
  protocol processes, 605–626
    initial communication from client to host-specific relay, 614–617
    initial communication from client to IPv6-only host, 619–622
    initial communication from host-specific relay to client, 617–619
    initial communication from IPv6-only host to Teredo client, 622–625
    initial communication of clients in different sites, 612–614
    initial communication of clients on same link, 611

initial configuration for clients, 606–610
maintaining NAT mapping, 610
relay, 353
route for DirectAccess server on IPv4 Internet, 403
routes for DirectAccess client on IPv4 Internet, 401
routing, 363–365
server, 352
setting for client state, 392, 393
support in IPv6 for Windows, 27
UDP-encapsulated traffic, blocking, 369
unicast connectivity across IPv4 Internet between IPv6/IPv4 hosts, 419
use for DirectAccess client traffic over IPv4 Internet, 399
use on Microsoft Corpnet computers with DirectAccess, 445
Windows-based server and relay, 365
Windows systems support, 348
Teredo Client Port Group Policy setting, 356
Teredo Default Qualified Group Policy setting, 356
Teredo Refresh Rate Group Policy setting, 354
Teredo Server Name Group Policy setting, 366
Teredo State Group Policy setting, 355, 393
Ternary Content Addressable Memory (TCAM) entries, 447
"Testing for Understanding" sections, answers for, 457–486
test lab for IPv6, setting up, 487–514
demonstrating DHCPv6, 503–505
demonstrating DNS zone transfers over IPv6, 507
demonstrating IPv6-only environment, 509–514
demonstrating use of DHCPv6 by client, 505
hardware and software requirements, 489
resource for further information, 514
restoring to original configuration, 506
steps for configuring test lab, 489–503
test lab setup, 487–489
Time Exceeded messages, 119, 123
comparision of IPv6 and IPv4 messages, 127
structure of, 123
Time-to-Live (TTL) field, IPv4, 93
TLV. *See* type-length-value (TLV) format for options
token bucket, 119
Token Ring networks, IEEE 802.5 SNAP encapsulation, 526
TOS (Type of Service) field, 2
Total Length field, IPv4, 93
Tracert tool, 45
command-line options supporting IPv6, 45
Traffic Class field, 95, 426
traffic monitoring systems, analyzing for IPv6 technology gaps, 448
traffic translation, 296–299
NAT64/DNS64, 296
PortProxy, 297
training operations staff for IPv6 deployment, 448
Transaction Identifier (ID) field, 212, 230
transition addresses, 68
transition technologies, 283–300, 420
IPv6 RFCs for, 455
overview, 283
IPv6 transition addresses, 284
node types, 284
reduction of reliance on, 439
settings on DirectAccess client, 408
settings on DirectAccess server, 408
settings on Windows, 22, 29
traffic translation, 296–299
transition mechanisms, 286–291
DNS infrastructure, 290–291
dual-stack architecture, 287
IPv6-over-IPv4 tunneling, 288–290
using both IPv4 and IPv6, 286–288
tunneling configurations, 291–295
used by Microsoft Corpnet computers with DirectAccess, 444
using native IPv6 instead of, 448
Transport layer, 17
IPv6 RFCs for, 452
protocols, 286
Truncation (TC) flag, 231
T (Tentative) flag, 231
TTL (Time-to-Live) field, IPv4, 93
tunneled connections to IPv6 Internet, 11
tunneled data, PPTP, 4
tunneling
6to4, 323, 324–327
enabling forwarding on interface, 334
example, 325
configurations, 291–295
host-to-host tunneling, 293
host-to-router tunneling, 292
router-to-host tunneling, 292
router-to-router tunneling, 292
control of tunneled traffin on intranet, 391
disabling tunneling technologies, 419

enabling forwarding and advertising on ISATAP interface, 317
force tunneling, 400
interface names displayed by ipconfig, 43
interface names displayed by netsh, 51
IP-HTTPS tunneling interface, 372
IPv6-over-IPv4 tunneling, 288–290
IPv6 traffic using Teredo, 348
ISATAP-based IPv6 traffic, 302, 316
example, 303
multiple addresses assigned to interfaces, 231
sending tunneled IPv6 traffic over site-to-site VPN connection, 432
support in IPV6CP on Windows, 26
tunnel-based IPv6 connectivity, 417–420
6to4, 418
ISATAP, 417
manually configured tunnels, 419
Teredo, 419
tunnels
automatic tunnels, 295
configured tunnels, 294
IPv4-based, in Windows Framing Layer, 18
manually configured, 419
types of, 294
Type 2 Routing header, 554
Type field
Destination Unreachable messages, 120
Echo Request messages, 125
for ICMPv6 error messages, 118
for ICMPv6 header, 119
for ICMPv6 information messages, 118
MLDv2 Multicast Listener Report messages, 198
modified Multicast Listener Query messages, 196
MTU option, 142
Multicast Listener Done messages, 194
Multicast Listener Query messages, 190
Multicast Listener Report messages, 191
Neibhbor Discovery (ND) options, 134
Neighbor Advertisement messages, 153
Neighbor Solicitation messages, 151
Packet Too Big messages, 122
Parameter Problem messages, 124
Prefix Information option, 137
Redirected Header option, 140
Redirect messages, 156
Route Information option, 143
Router Advertisement messages, 147
Router Solicitation messages, 146
Source Link-Layer Address option, 135
Target Link-Layer Address option, 135
Time Exceeded messages, 123
type-length-value (TLV) format for options, 102, 134
DHCPv6 options, 213
Type of Service field, IPv4, 93
Type of Service (TOS) field, 2

U

UDP Destination Port field for LLMNR queries, 246
UDP Source Port field for LLMNR queries, 246
UDP (User Datagram Protocol), 286
configuring UDP port for Teredo client, 356
implementation in Windows, 17
NAT64 providing basic translation for, 378
UDP port 3544 for Teredo traffic, 393
ULA. *See* unique local addresses
U/L (Universal/Local) bit, 84
complementing in EUI-64 address, 85
U/L (Universal/Local flag), Teredo addresses, 357
UNC (Uniform Naming Convention) path names, 24, 247
unicast addresses
assigned to interfaces on IPv6 router, 74
global unicast addresses, 62–64
in IPv6, 61
interfaces on IPv6 host, 73
IPv4 address of ISATAP router, 306
IPv6 global addresses, 62
IPv6 prefixes and, 60
link-local addresses, 65–66
mapping to solicited-node multicast address, 70
special IPv6 addresses, 67
subnetting in IPv6, 75
determining number of subnetting bits, 75
support in IPv6 for Windows, 21
transition addresses, 68
types in IPv6, 62
unique local addresses, 66
Uniform Naming Convention (UNC) path names, 24, 247
unique local addresses, 66
assigned to interfaces in IPv6 host, 74
automatically registered AAAA records, 243
subnetting subnet ID portion, 75
unicast addresses for subnets, 416
unspecified address, 67
Unused field
Destination Unreachable messages, 120
Time Exceeded messages, 123

Upper Layer Packet Length field, 114
upper-layer protocols, 10
  checksums in IPv6, 114
  indicating forward progress of communications, 164
  upper-layer protocol data unit (PDU), 91, 92
    Authentication header and, 112
    on IPv6 destination host, 266
URLs
  literal IPv6 addresses in. *See* literal IPv6 addresses in URLs
  network location URL for DirectAccess clients, 407

V

Valid Lifetime field, Prefix Information option, 138
Valid state, autoconfigured addresses, 206
Value field, ND options, 134
Version field
  IPv4 header, 92
  IPv6 header, 95, 262
virtual IP addresses (VIPs), 321, 335
VPNs (virtual private networks), 289
  access to Microsoft Corpnet from Internet, 445
  PPTP and L2TP connections, support in IPV6CP on Windows, 26
  site-to-site VPN connections, 432

W

WAN optimizers
  IPv6 functionality, 415
  support for compression of IPv6 traffic, 425
WANs (wide area networks)
  IPV6 MTUs for common technologies, 113
  link-layer technologies in IPv6, 10
  media, 529–537
  MTUs for common technologies, 516
weak host model, 255
  enabling in Windows, 256
WFP (Windows Filtering Platform), 32
Win32 Internet Extensions (WinInet) API, 25, 32
Windows Filtering Platform (WFP), 32
Windows Firewall, 391
  Advanced Security snap-in
    configuring connection security rules for IPv6 traffic, 390
  protection from unwanted incoming IPv6 traffic, 349
  support of IPv6 traffic and configuration of exceptions, 22
  use for IPv6 traffic on Microsoft Corpnet, 445
Windows Internet Name Service (WINS), 35
Windows Management Instrumentation (WMI) version 2, 33
Windows Network Load Balancing (NLB), 321
Windows Peer-to-Peer Networking platform, 336
Windows PowerShell
  6to4 host/router configuration, 332
  6to4 router configuration, 334
  cmdlet help (cmdlet_name -?), 385
  complete list of commands, online reference, 37
  configuration of firewall rules, 23
  configuration of IPv6 settings for Windows, 22
  configuring IPv6 router to set M flag to 1 in router advertisements, 212
  configuring IPv6 router to set O flag to 1 in router advertisements, 212
  configuring IPv6 with, 37
  configuring ISATAP router, 318
  configuring static routes, 26
  configuring static routing, 272–275
  disabling 6to4, 393
  disabling ISATAP, 303, 392
  displaying IPv6 configuration, 49–51
    Get-NetIPInterface –AddressFamily IPv6 command, 49
    Get-NetNeighbor –AddressFamily IPv6 command, 51
    Get-NetRoute –AddressFamily IPv6 command, 50
  enabling an interface for forwarding, 26
  enabling forwarding and advertising on ISATAP tunneling interface, 317
  enabling ISATAP on individual computers, 305
  enabling multicast forwarding, 202
  enabling sending of router advertisements, 26
  enabling temporary addresses, 221
  enabling weak host model, 256
  Get-NetIPAddress -AddressFamily ipv6 command, 222
  Get-NetIPConfiguration cmdlet, 43
  Get-NetIPInterface cmdlet, 45
  Get-NetNeighbor cmdlet, 159, 261
  Get-NetPrefixPolicy cmdlet, 245, 291
  Get-NetRoute cmdlet, 159, 256
  manually configuring IPv6, 33

NAT64/DNS64, configuring in Windows 2012, 384
randomized interface identifiers (IDs), disabling, 431
Set-DnsClientServerAddress cmdlet, 24, 241
Set-NetIPv6Protocol cmdlet, lifetime for temporary addresses, 23
Set-NetIPv6Protocol MldVersion=2 cmdlet, 201
Set-NetIPv6Protocol MldVersion=3 cmdlet, 202
Set-NetIPv6Protocol -RandomizeIdentifiers Disabled command, 83
Set-NetIPv6Protocol –SourceRoutingBehavior command, 108
Set-NetTCPSetting -EcnCapability Enabled command, 21
Teredo client configuration, qualified state, 356
Teredo client state configuration, 355, 393
Teredo client UDP port configuration, 356
Teredo refresh rate, setting, 354
Teredo relay configuration, 366
Teredo server configuration, 365
name or first IPv4 address, 366
Windows Runtime, 32
Windows Server
IPv6 protocol for, 17
IPv6 protocol for Windows Server 2003, 19
temporary addresses, 23
Windows Server 2012, detecting lack of connectivity to IPv6, 20
Windows Sockets, 18, 31
applications using, updating for IPv6 connectivity, 20
changes for IPv6, 539–548
added constants, 540
address data structures, 540
address sorting, 545
address-to-name translation, 544
core socket functions, 542
resources for further information, 539
socket options, 546
wildcard addresses, 541
Getaddrinfo( ) function, 25, 249
INetFwPolicy2( ) function, 349
name resolution functions, 232
scanning application code for IPv4-specific calls, 428
support for IPv6, 415
Windows systems
6to4 support, 331–335
host/router support, 331
manual router configuration, 333
router support, 332
802.1X authentication support, 389
APIPA (Automatic Private IP Addressing), 65
APIs not dependent on IPv4 or IPv6, 429
augoconfigured addresses for IPv6, 220, 221–224
automatic tunneling, IPv6 for Windows, 295
configured tunnels for IPv6 protocol, 294
configuring NAT64/DNS64 in Windows Server 2012, 384
detecting lack of connectivity to IPv6, 20
DHCPv6 on, 432
DHCPv6 support, 216
DHCPv6 client, 216
DHCPv6 relay agent, 216
DHCPv6 stateless and stateful server, 218
DirectAccess setup on, 411
DirectAccess, simplified configuration on Windows Server 2012, 404
dual IP layer architecture, 287
dual-stack architecture, 287
fixing IPv6 brokenness in Windows Server 2012 and Windows 8, 237–238
force tunneling for DirectAccess, 401
host-based security for IPv6 traffic, 424
HTTPS support, 370
IP-HTTPS features in Windows Server 2012 and Windows 8, 374
IPsec support for IPv4 and IPv6, 389
IPsec with IPv6, support for, 403
IPv4-only hosts, upgrading to IPv6/IPv4, 430
IPv6 protocol for, 17–56
6to4 support, 27
APIs, 30–33
application support, 30
command-line tools, 42–49
configuring with Netsh.exe tool, 37–39
configuring with PowerShell, 37
DHCPv6, 27
disabling IPv6, 40
displaying IPv6 configuration with netsh, 51–53
displaying IPv6 configuration with PowerShell, 49–51
DNS support, 24
features, 19

group policy settings for transition technologies, 29
GUI and command-line configuration, 22
installed and enabled by default, 20
integrated IPsec support, 22
IP-HTTPS, 28
IPv6 over PPP, 26
ISATAP support, 27
literal IPv6 addresses in URLs, 25
LLMNR, 25
manually configuring IPv6, 33
NAT64/DNS64, 28
PortProxy, 28
random interface IDs, 23
references for RFCs cited, 53
source and destination address selection, 24
stack enhancements, 21
static routing, 26
support for ipv6-literal.net names, 24
temporary addresses, 23
Teredo support, 27
Windows XP and Windows Server 2003, 19
ISATAP in Windows Server 2012 and Windows 8, 321
ISATAP Router Name Group Policy setting, 311
ISATAP tunneling with IPv6 protocol, 302
MLD and MLDV2 support, 201
name resolution support for IPv6, 240–250
DNS dynamic update, 243–244
DNS resolver, 241
DNS Security Extensions (DNSSEC), 250
DNS Server service, 242–243
DNS zone transfers, 244
Hosts file, 240
ipv6-literal.net names, 247
LLMNR support, 246
Name Resolution Protocol Table (NRPT), 249
Peer Name Resolution Protocol (PNRP), 248
source and destination address selection, 244–246
permanent interface IDs, 83
resolving host name "ISATAP", 307
routing table, IPv6 (example), 256
Routing Type 0 headers, 108
secure DNS dynamic updates and non-domain joined hosts, 422
site-to-site VPN connection with PPTP or L2TP/IPsec, 433
static routing with IPv6, 272–278
configuring with Netsh tool, 275–277
configuring with Routing and Remote Access, 277
configuring with Windows PowerShell, 272–275
strong host model for sent IPv6 packets, 256
support for IPv6, 414
TCP/IP protocol, 286
Teredo client, 352
Teredo client and host-specific relay, 354–356
Teredo host-specific relay support, 354
Teredo relay support, 353
Teredo server and relay on, 365
Teredo support, 348
tunneling technologies, disabling, 419
WinInet. *See* Win32 Internet Extensions
WINS, 35
deprecating use of, for IPv6 intranet, 449
Winsock. *See* Windows Sockets
Winsock Kernel (WSK), 31
Wired Ethernet Connection, enabling forwarding and advertising over, 273
wireless networks, ad hoc IEEE 802,11 network, 228
WSAConnectByName( ) function, 232
WSARecvMsg( ) function, 542
WSASendMsg( ) function, 542
WSK (Winsock Kernel), 31

X

X.25, 531
encapsulation of IPv6 packets, 532

Z

zeros, compressing in representation of IPv6 addresses, 60
zone ID
for link-local address, 43
for link-local addresses, 65
specifying as part of IPv6 address for command-line tools, 45
zone transfers (DNS), 244
denonstrating over IPv6, 507

About the Author

JOSEPH DAVIES is an award-winning author and instructor with 18 years' experience in TCP/IP, networking, and security technologies. His books include *Understanding IPV6, Second Edition* and *Windows Server 2008 TCP/IP Protocols and Services*, and he writes the monthly column "The Cable Guy" for Microsoft TechNet.