IPv6 Fundamentals: A Straightforward Approach to Understanding IPv6

Second Edition

Rick Graziani

Cisco Press

800 East 96th Street

Indianapolis, IN 46240

IPv6 Fundamentals: A Straightforward Approach to Understanding IPv6, Second Edition

Rick Graziani

Published by:
Cisco Press
800 East 96th Street
Indianapolis, IN 46240 USA

Printed in the United States of America

1 17

Library of Congress Control Number: 2017931983

ISBN-13: 978-1-58714-477-6

ISBN-10: 1-58714-477-8

Warning and Disclaimer

This book is designed to provide information about IPv6 (Internet Protocol version 6). Every effort has been made to make this book as complete and as accurate as possible, but no warranty or fitness is implied.

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

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

Feedback Information

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

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

We greatly appreciate your assistance.

Trademark Acknowledgments

Special Sales

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

For government sales inquiries, please contact governmentsales@pearsoned.com.

For questions about sales outside the U.S., please contact intlcs@pearson.com.

Editor-in-Chief: Mark Taub

Product Line Manager: Brett Bartow

Business Operation Manager, Cisco Press: Ronald Fligge

Executive Editor: Mary Beth Ray

Managing Editor: Sandra Schroeder

Development Editor: Ellie Bru

Project Editor: Mandie Frank

Copy Editor: Kitty Wilson

Technical Editors: Jim Bailey, Tim Martin

Editorial Assistant: Vanessa Evans

Cover Designer: Chuti Prasertsith

Composition: codeMantra

Indexer: Cheryl Lenser

Proofreader: Larry Sulky

Americas Headquarters
Cisco Systems, Inc.
San Jose, CA

Asia Pacific Headquarters
Cisco Systems (USA) Pte. Ltd.
Singapore

Europe Headquarters
Cisco Systems International BV Amsterdam,
The Netherlands

Cisco has more than 200 offices worldwide. Addresses, phone numbers, and fax numbers are listed on the Cisco Website at **www.cisco.com/go/offices.**

About the Author

Rick Graziani has been an instructor of computer networking and computer science courses at Cabrillo College in Aptos, California since 1994. Rick also teaches networking courses in the Computer Engineering department at the University of California, Santa Cruz and is on the Curriculum Engineering team for Cisco Networking Academy. Prior to teaching, he worked in the information technology field for Santa Cruz Operation, Tandem Computers, Lockheed Missiles and Space Company, and served five years in the U.S. Coast Guard. When he is not working, he is most likely surfing at one of his favorite Santa Cruz surf breaks or hanging out with his dog, Luigi. You are welcome to use his instructional resources on his Cabrillo College website, www.cabrillo.edu/~rgraziani, for IPv6, CCNA, or CCNP information. You can email graziani@cabrillo.edu to obtain the username and password for all his materials.

Rick and Luigi

About the Technical Reviewers

Jim Bailey, CCIE No. 5275 (Routing & Switching; Service Provider) and CCDE No. 20090008, is a Solution Architect at Cisco Systems with more than 25 years of experience in networking. As part of the Cisco Advanced Services team, he works on the architecture, design, and implementation of networks for enterprise, service provider, and government customers. He has focused on IPv6 integration into those networks for more than 12 years. He has presented *IPv6 Planning, Deployment and Operational Considerations* at Cisco Live conferences. He has served as a technical reviewer for the *IPv6 for Enterprise Networks* and *IPv6 Design and Deployment LiveLessons* published by Cisco Press.

Tim Martin, CCIE No. 2020, is a dynamic presenter and a member of the Cisco Live Distinguished Speaker hall of fame. Tim frequently speaks at Cisco Live events in both the United States and Europe. Tim has been in the inter-networking industry for more than 34 years. Cisco Press recently published a title of his work *IPv6 Design & Deployment Live Lessons* (ISBN 9780134655512), a 6-hour video series that provides guidance on IPv6 enterprise design. In his current role at Cisco, he is a Solutions Architect focused on the U.S. public sector market. Tim achieved the distinction of a multi-protocol CCIE No. 2020 in June 1996 and has also attained the Gold Certified Engineer status from the IPv6 Forum. He has participated in numerous industry events related to IPv6 and contributes to the IETF IPv6 subcommittees. Tim is also a member of many different IPv6-related task forces, including the FEDv6TF, NAv6TF, TXv6TF, and RMv6TF.

Dedication

To brothers Frank and Mark. You are not only my brothers but you are my best friends. I love you both. Also, to all of my current and former students. I am humbled by the opportunity to teach such wonderful people. You make my job fun, and you are the reason I love to go to work every day.

Acknowledgments

First of all, I would like to thank my family their love and support. Family is the best.

I would like to thank all my friends and colleagues for their assistance. Mark Boolootian, Dave Barnett, and Jim Warner, thanks for the many years of discussing technologies and answering questions. We've drawn a lot of topologies on a lot of napkins over the years.

The technical editors Jim Bailey and Tim Martin at Cisco Systems deserve much more credit than the brief mention as technical reviewers for this book. They make me look a lot smarter than I am. They did an incredibly thorough job making sure that this book is as accurate and as up to date as possible. Their expertise and experience were invaluable in helping me author this book. They are both the unsung heroes of this project. Thank you for your dedication and your commitment.

I owe a great deal of gratitude to Gerlinde Brady, Mike Matera, and Rich Simms at Cabrillo College for their friendship and support. You made sure that the CS/CIS departments at Cabrillo College continued to run smoothly while I was engaged in the writing process. I feel very fortunate to work with all of you and all of our other friends in the CS/CIS department. Thank you David Hovey and Ahmad Allulu for lab support. Thank you Brad Smith, Patrick Mantey, J.J. Garcia-Luna Aceves, and Katia Obraczka for the privilege of teaching in the Computer Science and Engineering department at the University of California, Santa Cruz. Teaching students and working with faculty at UCSC have made this a much better book.

Writing this book has been one of many privileges I have received due to the honor of working with Dennis Frezzo, Jeremy Creech, Telethia Willis, and many others who work for the Cisco Networking Academy. Thank you all for the proud opportunity to be part of a program that has changed the lives of thousands of students around the world. More than colleagues, you are all friends, for which I am grateful.

Thank you, Pat Farley, for making sure I still get in my surf sessions. Now that the book is done, you'll see me in the lineup much more often. Thank you, Teri Graziani, for always being there and taking care of things while I was busy writing this book. I appreciate it more than you know.

Special thanks to Mary Beth Ray, executive editor for Cisco Press and a friend. Thank you for your patience and understanding through this long process. You always have that calm assurance and guidance.

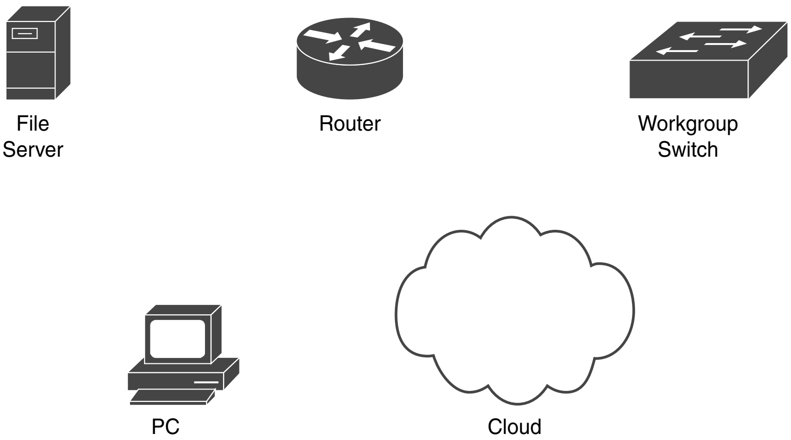
Thank you, Ellie Bru at Cisco Press, for working with me on a daily basis—weekdays and weekends—editing, formatting, and orchestrating the entire process. You were a pleasure to work with, and I am very grateful for all the hard work you put into this book. To Mandie Frank, Dhayanidhi, Kitty Wilson, and Larry Sulky, thank you for making me look like a better writer than I really am. And to everyone else at Cisco Press, I am extremely grateful for everything you have done. I am constantly amazed at the level of cooperation and teamwork required to produce a technical book. I am very thankful for all your help.

Finally, thank you to all my friends in Rome and Frascati, Italy, where I spent many months writing this book. Thank you, Giuseppe Cinque, for your many years of friendship and for convincing me to spend my sabbatical in Italy writing this book. Thank you, Mama and Papa Cinque, for making me part of your Positano family. Thank you, Levi Adam, Fidele Lassandro, Antonio Brancaccia, Daniel and Andrea and everyone at Tusculum Sport Center, Fermate N'Attimo, everyone at Il Borgo Verde, the Molinari family at Antico Forno, and everyone at Etabli for your friendship. And a special thank you to Alice, Mauro, Loredana, Marco, and Timmy Chialastri, for making Rome a second home. Your kindness made the time I spent in Rome some of the most enjoyable months I have ever had. Thank you for opening up your home and your hearts to me. I will forever be grateful for everything you did for me.

Contents at a Glance

Contents

Icons Used in This Book



Command Syntax Conventions

The conventions used to present command syntax in this book are the same conventions used in the IOS Command Reference. The Command Reference describes these conventions as follows:

- **Boldface** indicates commands and keywords that are entered literally as shown. In actual configuration examples and output (not general command syntax), boldface indicates commands that are manually input by the user (such as a **show** command).
- *Italics* indicate arguments for which you supply actual values.
- Vertical bars | separate alternative, mutually exclusive elements.
- Square brackets [ ] indicate optional elements.
- Braces { } indicate a required choice.
- Braces within brackets [{ }] indicate a required choice within an optional element.

Introduction

This book is intended to give you an in-depth understanding of IPv6 and related protocols. This book is for those who are new to the networking field, such as computer networking students, and also for network engineers with many years of experience administering larger enterprise networks. The only prerequisite is a basic understanding of networking protocols, including IPv4.

This second edition is a complete reorganization and almost a complete rewrite of the first edition, with a lot of new content. I have been a teacher for over 20 years, and this book is written for students of IPv6, as a self-study guide for learning IPv6. This book is designed to walk you through learning about IPv6 as if you were in a classroom, with the instructor explaining each concept. The information has been organized to help both those who want to read the book from cover to cover and also for those looking for specific information.

There is a great deal to learn about IPv6, and it is much more than just becoming familiar with a larger address. A brief look at the contents of this book will give you an idea of what is covered and what is needed to have a good understanding of IPv6.

My approach to writing this book was to do my best to explain each concept in a simple, step-by-step approach, as well as to include the critical details. It was a challenging balance between providing as much information as possible and not overwhelming you, the reader. IPv6 is not difficult to learn but involves multiple protocols and processes that might be new to some.

Don't be overwhelmed by all the details. For example, although I have included a brief description of each field in the protocols discussed in the book, it isn't necessarily important that you understand the details of each one. I mention this throughout the book. But I did feel it necessary not to leave out or hide these details from you.

RFCs are cited throughout the book. It was important to include these references for two reasons. First of all, I wanted to give you the authoritative source for the material in this book so that you have resources for more information. Second, IPv6 is currently and will continue to be a moving target for quite some time. Although it has been around for many years, additional development and fine-tuning are still taking place. If you are not familiar with reading RFCs, don't be intimidated. Most of them are not difficult to read, and they do their best to explain the topic clearly.

Review questions are included at the end of each chapter to help you understand some of the fundamental concepts discussed in the chapter. The review questions provide a high-level overview of some the key points discussed in the chapter. They are not meant as a detailed assessment of all the material covered in the chapter. An "IPv6 Command Quick Reference" has been included as Appendix B for Cisco IOS, Windows, Linux, and Mac OS X commands.

At times in this book I introduce a technology or concept but state that it is covered in more detail in a later chapter. I do this to explain the concept as it relates to the topic being discussed without getting lost in the details. The details are covered where appropriate. I feel it is better to revisit some of the more advanced topics after you have a more complete understanding of the entire IPv6 topic. At times I state that a concept is "beyond the scope of this book." I suggest resources for those who might be interested in learning more about those topics.

The objective of this book is to explain IPv6 as clearly as possible. At times it was like herding cats, trying to decide which topic to cover first. Chapter 2 is an IPv6 primer designed to give an overview of the main topics. Having this overview will make it easier as you progress through the rest of the book.

Readers are welcome to use the resources on my website, www.cabrillo.edu/~rgraziani, for IPv6, CCNA, or CCNP information. You can email me at graziani@cabrillo.edu to obtain the username and password for all my materials.

Goals and Methods

The most important goal of this book is to provide a thorough yet easy-to-understand introduction to IPv6. It is written for both computer networking students and seasoned network engineers. This book is also intended to provide a foundation in IPv6 that will allow you to build on. It explains topics that might be a little more challenging to grasp.

Another goal of this book is to be a resource for IPv6. The book is organized to make it as easy as possible to find information on specific topics. I have included command syntax, RFCs, and links to Cisco white papers to help guide you toward a better understanding of many of the topics.

Who Should Read This Book

This book is intended for anyone seeking a solid understanding of the fundamentals of IPv6, such as network engineers, network designers, network technicians, technical staff, and networking students, including those who are part of Cisco Networking Academy. You should have a basic familiarity with IPv4 and networking protocols before you begin reading this book.

Professionals deploying or planning to deploy IPv6 within a network will find this book useful. It provides examples, figures, IOS commands, and recommendations for configuring Cisco IOS IPv6 technology. Although Cisco devices are used in this book, those using non-Cisco equipment will also find this book helpful. The vast majority of protocols and technologies are IETF standards. Configuration and verification commands for Windows, Linux, and Mac OS are also included throughout the book.

How This Book Is Organized

If you are new to IPv6, you should read this book from cover to cover. However, if you have some knowledge of IPv6, it is designed to be flexible and allows you to easily move between chapters and sections of chapters to cover just the material you want to review. A common topology is used throughout the book except in a few cases.

Chapters 1 through 3 provide an introduction to IPv6, the reasons for moving to IPv6, an IPv6 primer, and a comparison of the IPv4 and IPv6 protocols. Chapters 4 through 7 discuss the different types of IPv6 address, including how to represent IPv6 addresses, global unicast addresses, link-local unicast addresses, and IPv6 multicast addresses. Chapters 8 through 11 discuss dynamic IPv6 addressing methods. Dynamic address allocation differs significantly in IPv6 and IPv4. These chapters discuss Stateless Address Autoconfiguration (SLAAC), stateless DHCPv6, and stateful DHCPv6. The chapter on SLAAC discusses the reason for permanent and temporary addresses and how to manage them using Cisco IOS and host operating systems. These chapters include the Cisco IOS commands and configuration examples for stateless and stateful DHCPv6. Chapters 12 and 13 discuss ICMPv6 and ICMPv6 Neighbor Discovery Protocol. These protocols and messages are introduced in earlier chapters, beginning with Chapter 2. Chapters 12 and 13 examine ICMPv6 and ICMPv6 Neighbor Discovery in more detail. Chapters 14 through 16 cover routing IPv6, including the IPv6 routing table, classic EIGRP for IPv6, EIGRP named mode, traditional OSPFv3, and OSPFv3 with address families. The last chapter, Chapter 17, introduces deploying IPv6 and transitioning from IPv4 to IPv6. In case you intend to read all the chapters, the order in the book is sequential.

The following list highlights the topics covered in each chapter and the book's organization:

- **Chapter 1, "Introduction to IPv6":** This chapter discusses how the Internet of today requires a new network layer protocol, IPv6, to meet the demands of its users. It also examines the limitations of IPv4 and describes how IPv6 resolves these issues while offering other advantages as well. This chapter examines the rationale of IPv6 and concerns regarding IPv4 address depletion. It presents a brief history of both IPv4 and IPv6. The IPv4 migration technologies CIDR and NAT are also discussed.
- **Chapter 2, "IPv6 Primer":** This chapter introduces some of the basic concepts and protocols that are explained in more detail throughout the rest of the book, including IPv6 address types, the basics of dynamic address allocation, and the hexadecimal number system, which is used to represent IPv6 addresses. This chapter gives an overview of some IPv6 concepts that are helpful in learning IPv6. This chapter also highlights many of the differences in IPv6.
- **Chapter 3, "Comparing IPv4 and IPv6":** This chapter compares and contrasts the IPv4 and IPv6 protocols. It also explores how fragmentation is handled. It discusses the IPv6 extension headers as well.
- **Chapter 4, "IPv6 Address Representation and Address Types":** This chapter introduces IPv6 addressing and address types. It discusses representation of IPv6 addresses, along with the different formats for representing IPv6 addresses and the rules for compressing the IPv6 notation. This chapter provides an introductory look at the different types of IPv6 addresses, including unicast, multicast, and anycast. It also discusses prefix length notation.
- **Chapter 5, "Global Unicast Address":** This chapter examines the global unicast address in detail. It examines the different parts of a global unicast address as well as

manual configuration of a global unicast address for Cisco IOS and host operating systems. The chapter also covers subnetting of IPv6, along with prefix allocation.

- **Chapter 6, "Link-Local Unicast Address":** This chapter examines link-local addresses and includes static and dynamic link-local address configuration examples. It explains the EUI-64 process, along with the significance of a link-local address in IPv6.
- **Chapter 7, "Multicast Addresses":** This chapter examines multicast addresses, including well-known and solicited-node multicast. It discusses the advantages of a multicast address over a broadcast address (the broadcast address does not exist in IPv6) and how IPv6 multicast addresses are mapped to Ethernet MAC addresses.
- **Chapter 8, "Basics of Dynamic Addressing in IPv6":** This chapter introduces and compares the three methods of dynamic address allocation: Stateless Address Autoconfiguration (SLAAC), stateless DHCPv6, and stateful DHCPv6. The chapters that follow discuss these methods in more detail.
- **Chapter 9, "Stateless Address Autoconfiguration (SLAAC)":** This chapter discusses the SLAAC process in detail. It includes a Wireshark examination of an ICMPv6 Router Advertisement message suggesting SLAAC. This chapter discusses the use of the privacy extension and temporary addresses with SLAAC-generated addresses, including the different states and lifetimes. It also explains how to manage privacy options on host operating systems.
- **Chapter 10, "Stateless DHCPv6":** This chapter examines SLAAC and other stateless DHCPv6 services. It covers DHCPv6 terminology and message types, along with the DHCPv6 process between the client and server. This chapter explains the rapid-commit option and relay agents.
- **Chapter 11, "Stateful DHCPv6":** This chapter examines stateful DHCPv6 services, similar to DHCP for IPv4. It also introduces a common method for providing IPv6 address space to homes using DHCPv6 with Prefix Delegation.
- **Chapter 12, "ICMPv6":** This chapter examines ICMPv6, which is a much more robust protocol than ICMPv4. It covers ICMPv6 error messages, including Destination Unreachable, Packet Too Big, Time Exceeded, and Parameter Problem. It also covers the ICMPv6 Echo Request and Echo Reply informational messages, along with Multicast Listener Discovery messages.
- **Chapter 13, "ICMPv6 Neighbor Discovery":** This chapter examines ICMPv6 Neighbor Discovery, including Router Solicitation, Router Advertisement, Neighbor Solicitation, Neighbor Advertisement, and Redirect messages. Not only does IPv6 resolve larger address space issues but ICMPv6 with Neighbor Discovery Protocol also presents a major change in network operations, including link-layer address resolution (ARP in IPv4), Duplicate Address Detection (DAD), Stateless Address Autoconfiguration (SLAAC), and Neighbor Unreachability Detection (NUD). This chapter discusses the IPv6 neighbor cache and neighbor cache states, similar to those of the IPv4 ARP cache.

- **Chapter 14, "IPv6 Routing Table and Static Routes":** This chapter examines the IPv6 routing table. It also discusses the configuration of IPv6 static routes, which are similar to static routes for IPv4. It explains IPv6 default routes and route summarization, as well as CEF for IPv6.
- **Chapter 15, "EIGRP for IPv6":** This chapter discusses EIGRP for IPv6. It begins with a comparison of EIGRP for IPv4 and EIGRP for IPv6. It discusses configuration and verification of classic EIGRP for IPv6 and EIGRP named mode (for IPv4 and IPv6).
- **Chapter 16, "OSPFv3":** This chapter discusses OSPFv3. It begins with a comparison of OSPFv2 (IPv4 only), traditional OSPFv3 (IPv6 only), and OSPFv3 with address families (IPv4 and IPv6). It also discusses configuration and verification of traditional OSPFv3 and OSPFv3 with address families.
- **Chapter 17, "Deploying IPv6 in the Network":** This chapter covers strategies for deploying IPv6, including creating an IPv6 address plan, configuring IPv6 VLANs, and implementing transparent failover at the first-hop router, using ICMPv6 or a first-hop redundancy protocol (FHRP). This chapter also discusses IPv4 and IPv6 integration and coexistence, including dual stacking, NAT64, and tunneling.
- **Appendix A, "Configuring NAT64 and IPv6 Tunnels":** This appendix provides configuration examples and additional information on NAT64 and IPv6 tunnels, introduced in Chapter 17.
- **Appendix B, "IPv6 Commands Quick Reference":** This appendix provides a summary of the Cisco IOS, Windows, Linux, and Mac OS commands used in this book.
- **Appendix C, "Answers to Review Questions":** This appendix provides the answers to the Review Questions at the end of each chapter.

Chapter 4

IPv6 Address Representation and Address Types

The most obvious and recognizable difference between IPv4 and IPv6 is the IPv6 address. An IPv4 address is 32 bits and expressed in dotted-decimal notation, whereas an IPv6 address is 128 bits in length and written in hexadecimal. However, there are many other differences between the two protocol addresses. IPv6 includes new address types as well as changes to familiar address types.

In this chapter, you will become familiar with reading IPv6 addresses. You will also learn how to represent many IPv6 addresses with fewer digits, using two simple rules.

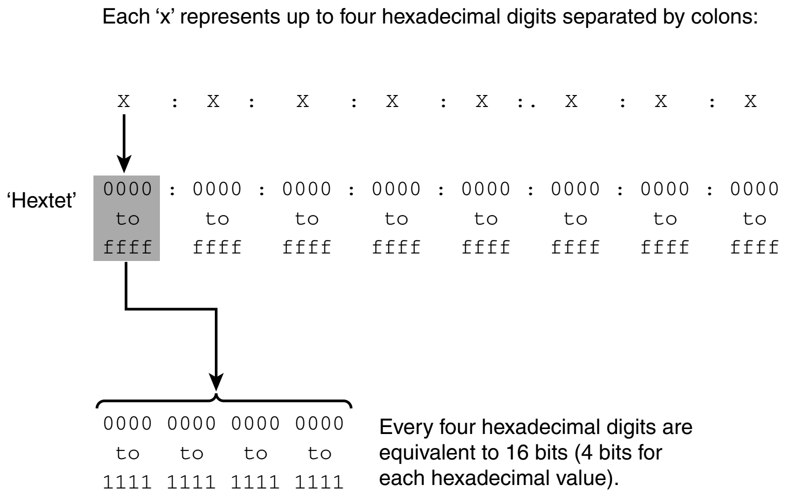
This chapter examines all the different types of IPv6 addresses in the unicast, multicast, and anycast categories. Some addresses, such as global unicast, link-local unicast, and multicast addresses, have more significance in IPv6. These addresses are examined more closely in Chapter 5, "Global Unicast Address," Chapter 6, "Link-Local Unicast Address," and Chapter 7, "Multicast Addresses."

Representation of IPv6 Addresses

IPv6 addresses are 128 bits in length and written as a string of hexadecimal digits. Every 4 bits can be represented by a single hexadecimal digit, for a total of 32 hexadecimal values (0_{16} [0000_2] through f_{16} [1111_2]). You will see later in this section how to possibly reduce the number of digits required to represent an IPv6 address. The alphanumeric characters used in hexadecimal are not case sensitive; therefore, uppercase and lowercase characters are equivalent. Although IPv6 address can be written in lowercase or uppercase, RFC 5952, *A Recommendation for IPv6 Address Text Representation*, recommends that IPv6 addresses be represented in lowercase.

Note If you are new to the hexadecimal number system, see Chapter 2, "IPv6 Primer," for information on this number system.

As described in RFC 4291, the preferred form is `x:x:x:x:x:x:x:x`. Each *x* is a 16-bit section that can be represented using up to four hexadecimal digits, with the sections separated by colons. The result is eight 16-bit sections, or hextets, for a total of 128 bits in the address, as shown in Figure 4-1. Figure 4-1 also shows an example of IPv6 addresses on a Windows host and a Mac OS host. These addresses and the format of these addresses will be explained in this chapter.




```
Windows-OS> ipconfig

Ethernet adapter Local Area Connection:
   Connection-specific DNS Suffix  . :
   IPv6 Address. . . . . . . . . . . : 2001:db8:cafe:1:d0f8:9ff6:4201:7086  ! IPv6 GUA
   Link-local IPv6 Address . . . . . : fe80::d0f8:9ff6:4201:7086%11         ! IPv6 Link-Local
   IPv4 Address. . . . . . . . . . . : 192.168.1.100
   Subnet Mask . . . . . . . . . . . : 255.255.255.0
   Default Gateway . . . . . . . . . : fe80::1%11                           ! IPv6 Default Gateway
                                       192.168.1.1
---------------------------------------------------------------------------------------------------
Mac-OS$ ifconfig

en1: flags=8863<UP,BROADCAST,SMART,RUNNING,SIMPLEX,MULTICAST> mtu 1500
          ether 60:33:4b:15:24:6f
          inet6 fe80::6233:4bff:fe15:246f%en1 prefixlen 64 scopeid 0x5          ! IPv6 Link-Local
          inet 192.168.1.111 netmask 0xffffff00 broadcast 192.168.1.255
          inet6 2001:db8:cafe:1:4bff:fe15:246f prefixlen 64 autoconf            ! IPv6 GUA
          Graphics: autoselect
          status: active
```

Figure 4-1 *Preferred Form of IPv6 Address*

The longest representation of the preferred form includes a total of 32 hexadecimal values. Colons separate the groups of 4-bit hexadecimal digits.

The unofficial term for a section of four hexadecimal values is a *hextet*, similar to the term *octet* used in IPv4 addressing. An IPv6 address consists of eight hextets separated by colons. As Figure 4-1 illustrates, each hextet, with its four hexadecimal digits, is equivalent to 16 bits. For clarity, the term *hextet* is used throughout this book when referring to individual 16-bit segments. The following list shows several examples of IPv6 addresses using the longest representation of the preferred form:

```
0000:0000:0000:0000:0000:0000:0000:0000
0000:0000:0000:0000:0000:0000:0000:0001
ff02:0000:0000:0000:0000:0000:0000:0001
fe80:0000:0000:0000:a299:9bff:fe18:50d1
2001:0db8:1111:000a:00b0:0000:9000:0200
2001:0db8:0000:0000:abcd:0000:0000:1234
2001:0db8:cafe:0001:0000:0000:0000:0100
2001:0db8:cafe:0001:0000:0000:0000:0200
```

At first glance, these addresses can look overwhelming. Don't worry, though. Later in this chapter, you will learn a technique that helps in reading and using IPv6 addresses. RFC 2373 and RFC 5952 provide two helpful rules for reducing the notation involved in the preferred format, which will be discussed next.

Rule 1: Omit Leading 0s

One way to shorten IPv6 addresses is to omit leading 0s in any hextet (that is, 16-bit section). This rule applies only to leading 0s and not to trailing 0s; being able to omit both leading and trailing 0s would cause the address to be ambiguous. Table 4-1 shows a list of preferred IPv6 addresses and how the leading 0s can be removed. The preferred form shows the address using 32 hexadecimal digits.

Table 4-1 *Examples of Omitting Leading 0s in a Hextet**

Format	IPv6 Address
Preferred	0000:0000:0000:0000:0000:0000:0000:0000
Leading 0s omitted	0: 0: 0: 0: 0: 0: 0: 0 or 0:0:0:0:0:0:0:0
Preferred	0000:0000:0000:0000:0000:0000:0000:0001
Leading 0s omitted	0: 0: 0: 0: 0: 0: 0: 1 or 0:0:0:0:0:0:0:1
Preferred	ff02:0000:0000:0000:0000:0000:0000:0001
Leading 0s omitted	ff02: 0: 0: 0: 0: 0: 0: 1 or ff02:0:0:0:0:0:0:1

Format	IPv6 Address
Preferred	fe80:**000**0:**000**0:**000**0:a299:9bff:fe18:50d1
Leading 0s omitted	`fe80:   0:   0:   0:a299:9bff:fe18:50d1` or fe80:0:0:0:a299:9bff:fe18:50d1
Preferred	2001:**0**db8:1111:**000**a:**00**b0:**000**0:9000:**0**200
Leading 0s omitted	`2001: db8:1111:   a:  b0:   0:9000: 200` or 2001:db8:1111:a:b0:0:9000:200
Preferred	2001:**0**db8:**000**0:**000**0:abcd:**000**0:**000**0:1234
Leading 0s omitted	`2001: db8:   0:   0:abcd:   0:   0:1234` or 2001:db8:0:0:abcd:0:0:1234
Preferred	2001:**0**db8:aaaa:**000**1:**000**0:**000**0:**000**0:**0**100
Leading 0s omitted	`2001: db8:aaaa:   1:   0:   0:   0: 100` or 2001:db8:aaaa:1:0:0:0:100
Preferred	2001:**0**db8:aaaa:**000**1:**000**0:**000**0:**000**0:**0**200
Leading 0s omitted	`2001: db8:aaaa:   1:   0:   0:   0: 200` or 2001:db8:aaaa:1:0:0:0:200

* In this table, the 0s to be omitted are in bold. Spaces are retained so you can better visualize where the 0s were removed.

It is important to remember that only leading 0s can be removed; if you deleted trailing 0s the address would be incorrect. To ensure that there is only one correct interpretation of an address, only leading 0s can be omitted, as shown in the following example:

- 0s omitted:

  2001:db8:100:a:0:bc:abcd:d0b

- Incorrect (trailing 0s):

  2001:db8**0**:100**0**:a**000**:0**000**:bc**00**:abcd:d0b**0**

- Correct (leading 0s):

  2001:**0**db8:**0**100:**000**a:**000**0:**00**bc:abcd:**0**d0b

Rule 2: Omit All-0s Hextets

The second rule for shortening IPv6 addresses is that you can use a double colon (::) to represent any single, contiguous string of two or more hextets (16-bit segments) consisting of all 0s. Table 4-2 illustrates the use of the double colon.

Table 4-2 *Examples of Omitting a Single Contiguous String of All-0s Hextets**

Format	IPv6 Address
Preferred	**0000:0000:0000:0000:0000:0000:0000:0000**
(::) All-0s segments	::
Preferred	**0000:0000:0000:0000:0000:0000:0000:**0001
(::) All-0s segments	::0001
Preferred	ff02:**0000:0000:0000:0000:0000:0000:**0001
(::) All-0s segments	ff02::0001
Preferred	fe80:**0000:0000:0000**:a299:9bff:fe18:50d1
(::) All-0s segments	fe80::a299:9bff:fe18:50d1
Preferred	2001:0db8:1111:000a:00b0:**0000**:0200
(::) All-0s segments	2001:0db8:1111:000a:00b0::0200
Preferred	2001:0db8:**0000:0000**:abcd:0000:0000:1234
(::) All-0s segments	2001:0db8::abcd:0000:0000:1234
Preferred	2001:0db8:aaaa:0001:**0000:0000:0000**:0100
(::) All-0s segments	2001:0db8:aaaa:0001::0100
Preferred	2001:0db8:aaaa:0001:**0000:0000:0000**:0200
(::) All-0s segments	2001:0db8:aaaa:0001::0200

* In this table, the 0s in bold in the preferred address are replaced by the double colon.

Only a single contiguous string of all-0s segments can be represented with a double colon; otherwise, the address would be ambiguous, as shown in this example:

- Incorrect address using two double colons:

  `2001::abcd::1234`

- Possible ambiguous choices:

```
2001:0000:0000:0000:0000:abcd:0000:1234
2001:0000:0000:0000:abcd:0000:0000:1234
2001:0000:0000:abcd:0000:0000:0000:1234
2001:0000:abcd:0000:0000:0000:0000:1234
```

As you can see, if two double colons are used, there are multiple possible interpretations, and you don't know which address is the correct one.

What happens if you have an address with more than one contiguous string of all-0s hextets—for example, 2001:0db8:0000:0000:abcd:0000:0000:1234? In that case, where should you use the single double colon (::)?

RFC 5952 states that the double colon should represent:

- The longest string of all-0s hextets.
- If the strings are of equal length, the first string should use the double colon (::) notation.

Therefore, 2001:0db8:0000:0000:abcd:0000:0000:1234 would be written 2001:0db8::abcd:0000:0000:1234. Applying both Rules 1 and 2, the address would be written 2001:db8::abcd:0:0:1234.

Note Most operating systems, including Cisco IOS and Microsoft Windows, accept the placement of a single double colon (::) in any valid location.

Combining Rule 1 and Rule 2

You can combine the two rules just discussed to reduce an address even further. Table 4-3 illustrates how this works, showing the preferred address, application of rule 1, and application of rule 2. Again, spaces are left so you can better visualize where the 0s have been removed.

Table 4-3 *Examples of Applying Both Rule 1 and Rule 2*

Format	IPv6 Address
Preferred	`0000:0000:0000:0000:0000:0000:0000:0000`
Leading 0s omitted	`   0:   0:   0:   0:   0:   0:   0:   0`
(::) All-0s segments	`::`
Preferred	`0000:0000:0000:0000:0000:0000:0000:0001`
Leading 0s omitted	`   0:   0:   0:   0:   0:   0:   0:   1`
(::) All-0s segments	`::1`

Format	IPv6 Address
Preferred	ff02:0000:0000:0000:0000:0000:0000:0001
Leading 0s omitted	ff02: 0: 0: 0: 0: 0: 0: 1
(::) All-0s segments	ff02::1
Preferred	fe80:0000:0000:0000:a299:9bff:fe18:50d1
Leading 0s omitted	fe80: 0: 0: 0:a299:9bff:fe18:50d1
(::) All-0s segments	fe80::a299:9bff:fe18:50d1
Preferred	2001:0db8:1111:000a:00b0:0000:9000:0200
Leading 0s omitted	2001: db8:1111: a: b0: 0:9000: 200
(::) All-0s segments	2001:db8:1111:a:b0::9000:200
Preferred	2001:0db8:0000:0000:abcd:0000:0000:1234
Leading 0s omitted	2001: db8: 0: 0:abcd: 0: 0:1234
(::) All-0s segments	2001:db8::abcd:0:0:1234
Preferred	2001:0db8:aaaa:0001:0000:0000:0000:0100
Leading 0s omitted	2001: db8:aaaa: 1: 0: 0: 0: 100
(::) All-0s segments	2001:db8:aaaa:1::100
Preferred	2001:0db8:aaaa:0001:0000:0000:0000:0200
Leading 0s omitted	2001: db8:aaaa: 1: 0: 0: 0: 200
(::) All-0s segments	2001:db8:aaaa:1::200

Table 4-4 shows the same examples as in Table 4-3, this time showing just the longest preferred form and the final compressed format after implementing both rules.

Table 4-4 *IPv6 Address Preferred and Compressed Formats*

Preferred Format	Compressed Format
0000:0000:0000:0000:0000:0000:0000:0000	::
0000:0000:0000:0000:0000:0000:0000:0001	::1
ff02:0000:0000:0000:0000:0000:0000:0001	ff02::1
fe80:0000:0000:0000:a299:9bff:fe18:50d1	fe80::a299:9bff:fe18:50d1
2001:0db8:1111:000a:00b0:0000:0000:0200	2001:db8:1111:a:b0::200
2001:0db8:0000:0000:abcd:0000:0000:1234	2001:db8::abcd:0:0:1234
2001:0db8:aaaa:0001:0000:0000:0000:0100	2001:db8:aaaa:1::100
2001:0db8:aaaa:0001:0000:0000:0000:0200	2001:db8:aaaa:1::200

Even after applying the two rules to compress the format, an IPv6 address can still look unwieldy. Don't worry! Chapter 5, "Global Unicast Address," shows a technique that I call the 3–1–4 rule. Using that rule makes IPv6 global unicast addresses (GUAs) easier to read than an IPv4 address and helps you recognize the parts of a GUA address.

Prefix Length Notation

In IPv4, the prefix (or network portion) of the address can be identified by a dotted-decimal netmask, commonly referred to as a *subnet mask*. For example, 255.255.255.0 indicates that the network portion, or prefix length, of the IPv4 address is the leftmost 24 bits. The 255.255.255.0 dotted-decimal netmask can also be written in CIDR notation as /24, indicating the 24 bits in the prefix.

IPv6 address prefixes can be represented much the same way that IPv4 address prefixes are written in CIDR notation. An IPv6 address prefix (the network portion of the address) is represented using the following format:

ipv6-address/prefix-length

The *prefix-length* is a decimal value indicating the number of leftmost contiguous bits of the address. It identifies the prefix (that is, the network portion) of the address. It is also used with unicast addresses to separate the prefix portion of the address from the Interface ID. Remember from Chapter 2 that the Interface ID is the equivalent to the host portion of an IPv4 address.

Let's look at an example using the address 2001:db8:aaaa:1111::100/64. The longest preferred form in Figure 4-2 illustrates how the /64 prefix length identifies the prefix, or network portion, of the address. The /64 prefix length leaves another 64 bits, which is the Interface ID portion of the address.

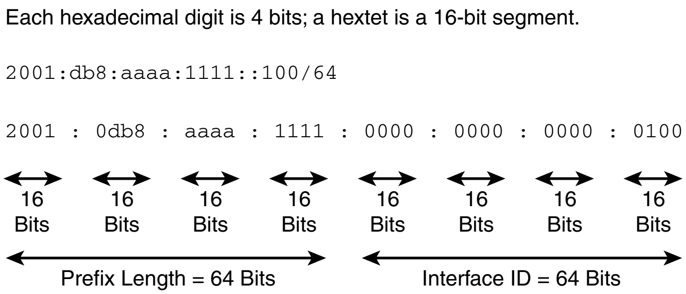


Figure 4-2 *IPv6 Prefix and Prefix Length*

In IPv6, just as in IPv4, the number of devices you can have on a network depends on the prefix length. However, due to the 128-bit length of an IPv6 address, there is no need to conserve address space as is needed with IPv4 public addresses.

In Figure 4-2, notice that the /64 prefix length results in an Interface ID of 64 bits. As we will discuss further in Chapter 5, this is a common prefix length for most end-user networks. A /64 prefix length gives us 18 quintillion devices on a single network (or subnet, if you prefer)!

Figure 4-3 shows several prefix length examples: /32, /48, /52, /56, /60, and /64. Notice that all of these examples fall on a *nibble boundary*, a multiple of 4 bits. Prefix lengths do not necessarily have to fall on a nibble boundary, although in most cases they do. Prefix lengths can also fall *within a nibble*—for example, /61, /62, or /63. We will discuss the prefix lengths, including within the nibble, more in Chapter 5 when we discuss the global unicast address, prefix allocation, and subnetting.

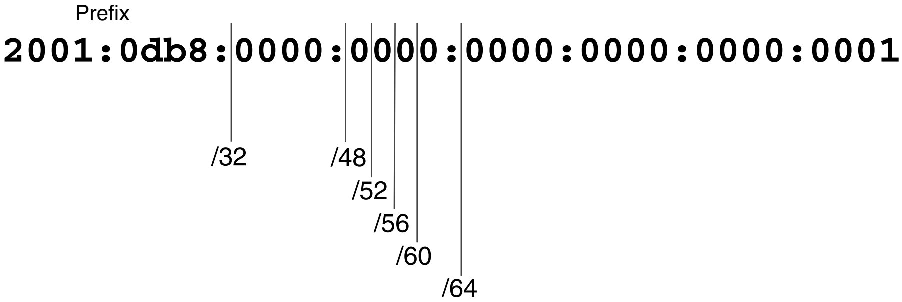


Figure 4-3 *IPv6 Prefix Length Examples*

IPv6 Address Types

We begin this section with a brief look at the IPv6 address space and how the different types of addresses are allotted within this space. Next, we examine the various addresses within three IPv6 address types: unicast, multicast, and anycast.

IPv6 address types are defined in RFC 4291, *IP Version 6 Addressing Architecture*. In this section, we examine the several types of unicast addresses, three types of multicast addresses, and the anycast address. We discuss some of these addresses in more detail than others. Global unicast addresses, link-local addresses, and multicast addresses are examined more closely in Chapters 5, 6, and 7.

Note IPv6 does not have a broadcast address. Other options exist in IPv6, such as a solicited-node multicast address and an all-IPv6 devices multicast address. Chapter 7 provides details on these types of addresses.

IPv6 Address Space

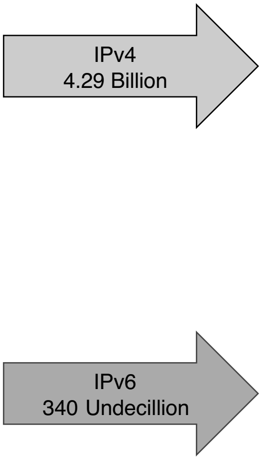
IPv4, with its 32-bit address space, provides for 4.29 billion (4,294,967,296) addresses. IPv6, with its 128-bit address space, provides for 340 undecillion addresses, or 340 trillion trillion trillion addresses. That's 340,282,366,920,938,463,463,374,607,431,768,211,456 addresses—a *lot* of addresses!

Many analogies have been made to help comprehend 340 undecillion (not all of which are completely accurate):

- "3,911,873,538,269,506,102 addresses per square meter of the surface of the planet Earth"[1]
- The number of grains of sand on Earth
- 10 nonillion addresses assigned to every person on Earth

As a disclaimer, I didn't do the math to calculate the number of square meters on the surface of Earth, and I haven't had a chance to count all the grains of sand on Earth either. And an argument can be made that this would be purely theoretical because of how addresses are allocated. Regardless, I think we can all agree that IPv6 provides an extremely large address space.

Figure 4-4 shows a chart of the powers of 10 to give a better idea of the tremendous increase in the IPv6 address space.



Number Name	Scientific Notation	Number of Zeros
1 Thousand	10^3	1,000
1 Million	10^6	1,000,000
1 Billion	10^9	1,000,000,000
1 Trillion	10^{12}	1,000,000,000,000
1 Quadrillion	10^{15}	1,000,000,000,000,000
1 Quintillion	10^{18}	1,000,000,000,000,000,000
1 Sextillion	10^{21}	1,000,000,000,000,000,000,000
1 Septillion	10^{24}	1,000,000,000,000,000,000,000,000
1 Octillion	10^{27}	1,000,000,000,000,000,000,000,000,000
1 Nonillion	10^{30}	1,000,000,000,000,000,000,000,000,000,000
1 Decillion	10^{33}	1,000,000,000,000,000,000,000,000,000,000,000
1 Undecillion	10^{36}	1,000,000,000,000,000,000,000,000,000,000,000,000
340,282,366,920,938,463,463,374,607,431,768,211,456		

Figure 4-4 *Powers of 10: Comparing IPv4 and IPv6 Address Space*

As mentioned in Chapter 1, "Introduction to IPv6," this means that we can now design IPv6 addressing schemas based on management and security plans, without the concern for public address depletion that we face with IPv4. (This will become even more evident in Chapter 5, when we discuss the global unicast address and subnetting.)

Table 4-5 shows the Internet Assigned Numbers Authority's (IANA's) allocation of the 128-bit IPv6 address space. Notice the allocations for global unicast, unique local unicast, link-local unicast, and multicast addresses. It may be a little difficult to visualize this using the table, so Figure 4-5 shows this same allocation in a pie chart to make it a little easier. Using the first 3 bits, the chart divides the IPv6 pie into eight slices (that is, 3 bits gives us eight possibilities). There are portions within the 000 and 111 slices used to indicate very small allocations (the chart shows them larger than the actual allocations) from this part of the address space.

Table 4-5 *IANA's Allocation of IPv6 Address Space**

Leading Bits	Address	Range of First Hextet	Allocation	Fraction of Space
000x		**0000** **1fff**		1/8
0000 0000	0000::/8	0000 00ff	Unspecified, loopback, embedded	1/256
0000 0001 through 0001 xxxx	0000::/3	0100 1fff	Reserved by IETF	Remaining 1/8
001x	2000::/3	2000 3fff	Global unicast	1/8
010x	4000::/3	4000 5fff	Reserved by IETF	1/8
011x	6000::/3	6000 7fff	Reserved by IETF	1/8
100x	8000::/3	8000 9fff	Reserved by IETF	1/8
101x	a000::/3	a000 bfff	Reserved by IETF	1/8

Leading Bits	Address	Range of First Hextet	Allocation	Fraction of Space	
110x	c000::/3	c000 dfff	Reserved by IETF	1/8	
111x				1/8	
1110 xxxx	e000::/4	e000 efff	Reserved by IETF		1/16
1111 0xxx	f000::/5	f000 f7ff	Reserved by IETF		1/32
1111 10xx	f800::/6	f800 fbff	Reserved by IETF		1/64
1111 110x	fc00::/7	fc00 fdff	Unique local unicast		1/128
1111 1110 0	fe00::/9	fe00 fe74	Reserved by IETF		1/512
1111 1110 10	fe80::/10	fe80 febf	Link-local unicast		1/1024
1111 1110 11	fec0::/10	fec0 feff	Reserved by IETF; previously site-local (deprecated)		1/1024
1111 1111	ff00::/8	ff00 ffff	Multicast		1/256

* In this table, the "Range of First Hextet" column does not show the complete range of the address space. For example, the actual range of the global unicast address space would be 2000:: through 3fff:ffff:ffff:ffff:ffff:ffff:ffff:ffff.

In both Table 4-5 and Figure 4-5, the IPv6 address space is divided into eighths, using the leading 3 bits (000, 001, 010, 011, 100, 101, 110, and 111). This information might be a little confusing right now, but it will become more obvious as you examine each of the IPv6 address types.

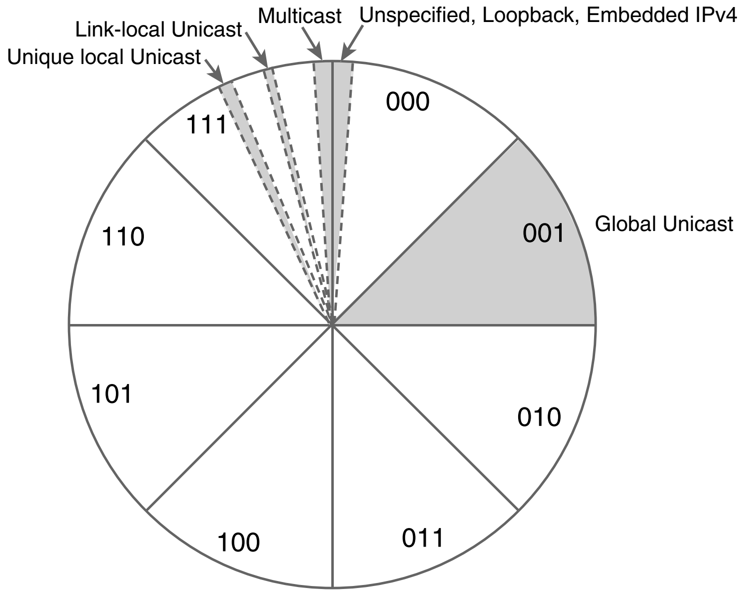


Figure 4-5 *IANA's Allocation of IPv6 Address Space in 1/8 Sections*

Unicast Addresses

Figure 4-6 diagrams the three types of addresses: unicast, multicast, and anycast. We begin by looking at unicast addresses. Don't be intimidated by all the different types of unicast addresses. The most significant types are global unicast addresses, which are equivalent to IPv4 public addresses, and link-local addresses. These address types are discussed in detail in Chapters 5 and 6.

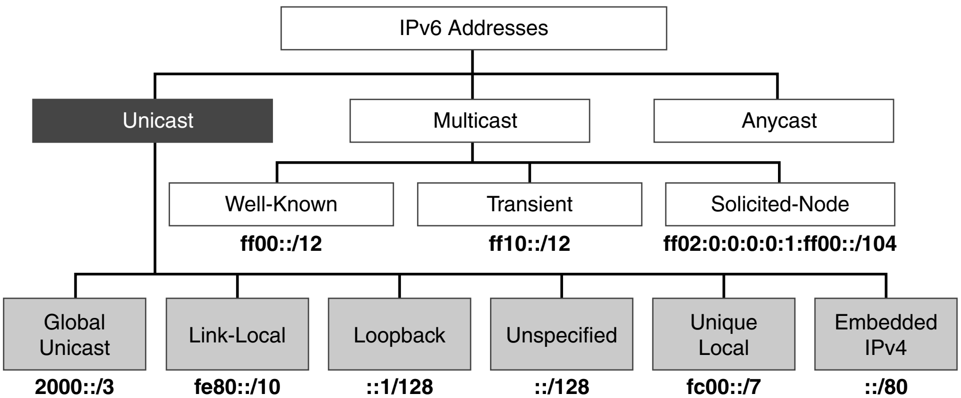


Figure 4-6 *IPv6 Address Types: Unicast Addresses*

A unicast address uniquely identifies an interface on an IPv6 device. A packet sent to a unicast address is received by the interface that is assigned to that address. Similar to IPv4, a source IPv6 addresses must be a unicast address.

Note Notice that there is no broadcast address shown in Figure 4-6. Remember that IPv6 does not include a broadcast address.

This section covers the different types of unicast addresses, as illustrated in Figure 4-6. The following is a quick preview of each type of unicast address discussed in this section:

- **Global unicast:** A routable address in the IPv6 Internet, similar to a public IPv4 address (covered in more detail in Chapter 5).
- **Link-local:** Used only to communicate with devices on the same local link (covered in more detail in Chapter 6).
- **Loopback:** An address not assigned to any physical interface that can be used for a host to send an IPv6 packet to itself.
- **Unspecified address:** Used only as a source address and indicates the absence of an IPv6 address.
- **Unique local:** Similar to a private address in IPv4 (RFC 1918) and not intended to be routable in the IPv6 Internet. However, unlike RFC 1918 addresses, these addresses are not intended to be statefully translated to a global unicast address.
- **IPv4 embedded:** An IPv6 address that carries an IPv4 address in the low-order 32 bits of the address.

Global Unicast Address

Global unicast addresses (GUAs), also known as *aggregatable global unicast addresses*, are globally routable and reachable in the IPv6 Internet. They are equivalent to public IPv4 addresses. They play a significant role in the IPv6 addressing architecture. One of the main motivations for transitioning to IPv6 is the exhaustion of its IPv4 counterpart. As you can see in Figure 4-6, a GUA address is only one of several types of IPv6 unicast addresses.

Figure 4-7 shows the generic structure of a GUA, which has three fields:

- **Global Routing Prefix:** The Global Routing Prefix is the prefix or network portion of the address assigned by the provider, such as an ISP, to the customer site.
- **Subnet ID:** The Subnet ID is a separate field for allocating subnets within the customer site. Unlike with IPv4, it is not necessary to borrow bits from the Interface ID (host portion) to create subnets. The number of bits in the Subnet ID falls between where the Global Routing Prefix ends and where the Interface ID begins. This makes subnetting simple and manageable.
- **Interface ID:** The Interface ID identifies the interface on the subnet, equivalent to the host portion of an IPv4 address. The Interface ID in most cases is 64 bits.

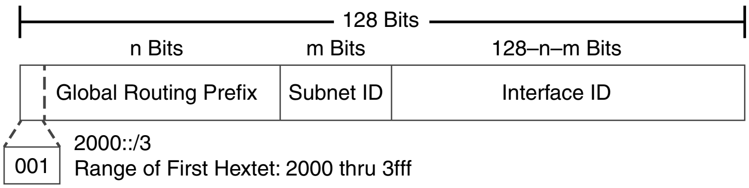


Figure 4-7 *Structure of a GUA Address*

Figure 4-7 illustrates the more general structure, without the specific sizes for any of the three parts. The first 3 bits of a GUA address begin with the binary value 001, which results in the first hexadecimal digit becoming a 2 or a 3. (We look at the structure of the GUA address more closely in Chapter 5.)

There are several ways a device can be configured with a global unicast address:

- Manually configured.
- Stateless Address Autoconfiguration.
- Stateful DHCPv6.

Example 4-1 demonstrates how to view the global unicast address on Windows and Mac OS operating systems, using the **ipconfig** and **ifconfig** commands, respectively. The **ifconfig** command is also used with the Linux operating system and provides similar output.

Note You may see multiple IPv6 global unicast addresses including one or more temporary addresses. You'll learn more about this in Chapter 9.

Example 4-1 *Viewing IPv6 Addresses on Windows and Mac OS*

```
Windows-OS> ipconfig
Ethernet adapter Local Area Connection:
   Connection-specific DNS Suffix  . :
   ! IPv6 GUA
   IPv6 Address. . . . . . . . . . . : 2001:db8:cafe:1:d0f8:9ff6:4201:7086
   ! IPv6 Link-Local
   Link-local IPv6 Address . . . . . : fe80::d0f8:9ff6:4201:7086%11
   IPv4 Address. . . . . . . . . . . : 192.168.1.100
   Subnet Mask . . . . . . . . . . . : 255.255.255.0
   ! IPv6 Default Gateway
   Default Gateway . . . . . . . . . : fe80::1%11
                                       192.168.1.1
-----------------------------------------------------------------------------------
Mac-OS$ ifconfig
en1: flags=8863<UP,BROADCAST,SMART,RUNNING,SIMPLEX,MULTICAST> mtu 1500
   ether 60:33:4b:15:24:6f
   ! IPv6 Link-Local
   inet6 fe80::6233:4bff:fe15:246f%en1 prefixlen 64 scopeid 0x5
   inet 192.168.1.111 netmask 0xffffff00 broadcast 192.168.1.255
   ! IPv6 GUA
   inet6 2001:db8:cafe:1:4bff:fe15:246f prefixlen 64 autoconf
   media: autoselect
   status: active
```

This section has provided just a brief introduction to global unicast addresses. Remember that IPv6 introduced a lot of changes to IP. Devices may obtain more than one GUA address for reasons such as privacy. For a network administrator needing to manage and control access within a network, having these additional addresses that are not administered through stateful DHCPv6 may be undesirable. Chapter 11 discusses devices obtaining or creating multiple global unicast addresses and various options to ensure that devices only obtain a GUA address from a stateful DHCPv6 server.

Link-Local Unicast Address

Link-local addresses are another type of unicast address as shown in Figure 4-6. A link-local address is a unicast address that is confined to a single link, a single subnet. Link-local addresses only need to be unique on the link (subnet) and do not need to be unique beyond the link. Therefore, routers do not forward packets with a link-local address. Devices can use Duplicate Address Detection (DAD) to determine whether or not the link-local address is unique.

Note Link-local unicast addresses are discussed in detail in Chapter 6. ICMPv6 DAD is examined in Chapter 13, "ICMPv6 Neighbor Discovery."

Figure 4-8 shows the format of a link-local unicast address, which is in the range fe80::/10. Using this prefix and prefix length results in the range of the first hextet being from fe80 to febf.

Note Using a prefix other than fe80 for a link-local address can result in unexpected behaviors. Although permitted by the RFC 4291, using a prefix other than fe80 should be tested prior to usage.

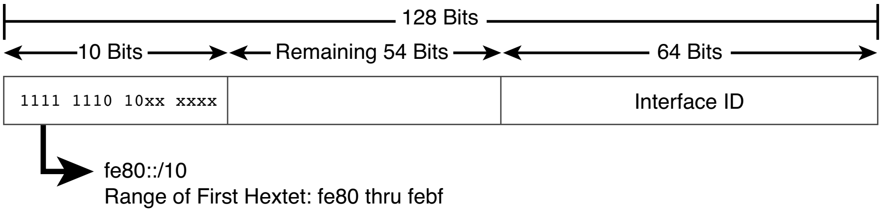


Figure 4-8 *Structure of a Link-Local Unicast Address*

In Chapter 6 we will examine the structure, uses, and configuration options for link-local addresses in much more detail. For now, here is a summary of some of the key points:

- To be an IPv6-enabled device, a device must have an IPv6 link-local address. The device doesn't have to have an IPv6 global unicast address, but it must have a link-local address.
- Link-local addresses are not routable off the link (IPv6 subnet). Routers do not forward packets with a link-local address.
- Link-local addresses only have to be unique on the link. It is very likely and sometimes even desirable for a device to use the same link-local address on different interfaces that are on different links.
- There can be only one link-local address per interface.

Configuration options for link-local addresses are (see Chapters 6 and 9 for more details):

- Devices dynamically (automatically) create their own link-local IPv6 address upon startup. This is the default on most operating systems, including Cisco IOS, Windows, Mac OS, and Linux.
- Link-local addresses can be manually configured.

The idea of a device creating its own IP address upon startup is really an amazing benefit in IPv6! Think of it. A device can create its own IPv6 link-local address completely on its own, without any kind of manual configuration or the services of a DHCP server. This means that the device can immediately communicate with any other device on its link (IPv6 subnet). A device may only need a link-local address because it only needs to communicate with other devices on its same network. Or it can use its link-local address to communicate with a device where it can obtain information for getting or creating a global unicast address, such as an IPv6 router or a DHCPv6 server. The device can then use this information to communicate with devices on other networks.

This solves the “Which came first, the chicken or the egg?” problem with IPv4. That is, “How do I ask a DHCP server for an IP address when I first need to have an IP address before I can communicate with the server to ask for one?” (DHCP for IPv4 uses a Discover message with an IPv4 source address of 0.0.0.0.) With IPv6, during startup the device automatically gives itself a link-local address that is unique on that subnet. It can then use this address to communicate with any device on the network, including an IPv6 router and, if necessary, a DHCPv6 server. Remember from Chapter 2 that an IPv6 router sends ICMPv6 Router Advertisement messages that allow the device to obtain a global unicast address, with or without the services of DHCPv6.

Example 4-1 demonstrates how to view the link-local address on Windows and Mac OS operating systems by using the **ipconfig** and **ifconfig** commands. These operating systems, as well as Linux, are enabled for IPv6 by default. So, even if the devices did not have a global unicast address, as shown in Example 4-1, you would still see the IPv6 link-local address. And as discussed in Chapter 2, this means client hosts are running IPv6 and, at a minimum, the network should be secured to prevent IPv6 attacks.

Note Notice the %11 and %en1 following the IPv6 link-local addresses in Example 4-1. These are known as *zone identifiers*, and they are used to identify the interface on the device. These are usually of little importance when referring to a link-local address, but they are highly significant for tying the address to the interface. Zone identifiers are discussed in Chapter 6.

The following are some of the ways IPv6 devices use a link-local address:

- When a device starts up, before it obtains a GUA address, the device uses its IPv6 link-local address as its source address to communicate with other devices on the network, including the local router.
- Devices use the router’s link-local address as their default gateway address.
- Routers exchange IPv6 dynamic routing protocol (OSPFv3, EIGRP for IPv6, RIPng) messages from their IPv6 link-local address.
- IPv6 routing table entries populated from dynamic routing protocols use the IPv6 link-local address as the next-hop address.

This section has provided just an introduction to the link-local address. We will explore all these topics in more detail in Chapter 6.

Loopback Addresses

A loopback address is another type of unicast address (refer to Figure 4-6). An IPv6 loopback address is ::1, an all-0s address except for the last bit, which is set to 1. It is equivalent to the IPv4 address block 127.0.0.0/8, most commonly the 127.0.0.1 loopback address.

Table 4-6 shows the different formats for representing an IPv6 loopback address.

Table 4-6 *IPv6 Loopback Address Representations*

Representation	IPv6 Loopback Address
Preferred	0000:0000:0000:0000:0000:0000:0000:0001
Leading 0s omitted	0:0:0:0:0:0:0:1
Compressed	::1

The loopback address can be used by a node to send an IPv6 packet to itself, typically when testing the TCP/IP stack. Loopback addresses have the following characteristics:

- A loopback address cannot be assigned to a physical interface.
- A packet with a loopback address, source address, or destination address should never be sent beyond the device.
- A router can never forward a packet with a destination address that is a loopback address.
- The device must drop a packet received on an interface if the destination address is a loopback address.

Unspecified Addresses

An unspecified unicast address is an all-0s address (refer to Figure 4-6). An unspecified unicast address is used as a source address to indicate the absence of an address. It cannot be assigned to an interface.

One example where an unspecified address can be used is as a source address in ICMPv6 Duplicate Address Detection (DAD). DAD is a process that a device uses to ensure that its unicast address is unique on the local link (network). DAD is discussed in Chapter 14.

Table 4-7 shows the different formats for representing an IPv6 unspecified address.

Table 4-7 *IPv6 Unspecified Address Representations*

Representation	IPv6 Unspecified Address
Preferred	0000:0000:0000:0000:0000:0000:0000:0000
Leading 0s omitted	0:0:0:0:0:0:0:0
Compressed	::

Unspecified addresses have the following characteristics:

- An unspecified source address indicates the absence of an address.
- An unspecified address cannot be assigned to a physical interface.
- An unspecified address cannot be used as a destination address.
- A router will never forward a packet that has an unspecified source address.

Unique Local Addresses

Figure 4-6 shows another type of IPv6 unicast address, the unique local address (ULA), which is the counterpart of IPv4 private addresses. Unique local addresses are also known as *private IPv6 addresses* or *local IPv6 addresses* (not to be confused with link-local addresses).

ULA addresses can be used similarly to global unicast addresses but are for private use and should not be routed in the global Internet. ULA addresses are only to be used in a more limited area, such as within a site or routed between a limited number of administrative domains. ULA addresses are for devices that never need access to the Internet and never need to be accessible from the Internet.

ULA addresses are defined in RFC 4193, *Unique Local IPv6 Unicast Addresses*. Figure 4-9 illustrates the format of a unique local unicast address.

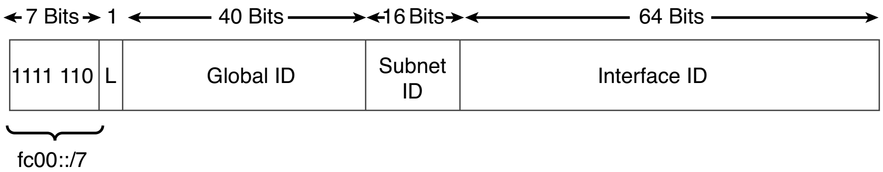


Figure 4-9 *Structure of a Unique Local Unicast Address*

Unique local addresses have the prefix fc00::/7, which results in the range of addresses from fc00::/7 to fdff::/7, as shown in Table 4-8.

Table 4-8 *Range of Unique Local Unicast Addresses*

Unique Local Unicast Address (Hexadecimal)	Range of First Hextet	Range of First Hextet in Binary
fc00::/7	fc00 to	**1111 110**0 0000 0000 to
	fdff	**1111 110**1 1111 1111

Unique local addresses have the following characteristics:

- They can be used just like global unicast addresses.
- They can be used for devices that never need access to or from the global Internet.
- They allow sites to be combined or privately interconnected without address conflicts and without requiring addressing renumbering. (Address conflicts are highly unlikely due to the large address space.)
- They are independent of any ISP and can be used within a site even without having Internet connectivity.

ULA and NAT

ULA and NAT is a bit of a tricky topic. The concept of translating a unique local address to a global unicast address is the subject of ongoing debate within the IPv6 community, and it fosters emotional opinions on both sides of the argument. The IAB published an informational RFC highlighting its thoughts on NAT and IPv6 in RFC 5902, *IAB Thoughts on IPv6 Network Address Translation*. In this RFC, the IAB summarizes the use of NAT as follows:

> Network address translation is viewed as a solution to achieve a number of desired properties for individual networks: avoiding renumbering, facilitating multihoming, making configurations homogenous, hiding internal network details, and providing simple security.

So, does this means NAT provides security, and ULA addresses can be translated to GUA addresses for this purpose? The simple answer is no. RFC 5902 goes on to state, "However, one should not confuse NAT boxes with firewalls. As discussed in [RFC 4864] Section 2.2, the act of translation does not provide security in itself."

Remember that the driving force for using NAT with IPv4 is not security but IPv4 address depletion. Although the IAB and the IETF did not intend for NAT to be used with IPv6 as it is with IPv4, NAT does provide mechanisms for translation where translation is necessary. These translation techniques include Network Prefix Translation version 6 (NPTv6), described in RFC 6296, *IPv6-to-IPv6 Network Prefix Translation*, and NAT66, described in an Internet draft RFC, *IPv6-to-IPv6 Network Address Translation* (long expired). Both of these RFCs focus on translation for address independence—and only where necessary. In RFC 6296, the IETF goes as far as stating, "For reasons discussed in [RFC 2993] and Section 5, the IETF does not recommend the use of Network Address Translation technology for IPv6."

Both NPTv6 and NAT66 are designed for address independence and not security. *Address independence* means that a site does not have to renumber its internal addresses if the ISP changes the site's external prefix or if the site changes ISPs and receives a different prefix.

NPTv6 and NAT66 are both stateless technologies, whereas NAT for IPv4 is stateful. It is the statefulness, not NAT itself, that provides the security. This means that internal devices are open to certain types of attacks that would not be possible in a NAT for IPv4 network. Without getting into the NAT-versus-security debate covered in Chapter 1, NAT for IPv4 is not security and introduces many problems and challenges.

If all this seems vague, complicated, and perhaps even contradictory, welcome to the discussion on NAT and IPv6.

Note For more information on ULA addresses with NAT66 or NPTv6, see Ed Horley's excellent articles on the topic, at www.howfunky.com. Horley has also written an excellent book, *Practical IPv6 for Windows Administrators*.

L Flag and Global ID

ULA addresses have the prefix fc00::/7, or the first 7 bits as 1111 110x. As shown in Figure 4-10, the eighth bit (x) is known as the L flag, or the local flag, and it can be either 0 or 1. This means that the ULA address range is divided into two parts:

- **fc00::/8** (1111 110**0**): When the L flag is set to 0, may be defined in the future.
- **fd00::/8** (1111 110**1**): When the L flag is set to 1, the address is locally assigned.

Because the only legitimate value for the L flag is 1, the only valid ULA addresses today are in the fd00::/8 prefix.

Another difference between ULA addresses and private IPv4 addresses is that ULA addresses can also be globally unique. This is helpful for ensuring that there won't be any conflicts when combining two sites using ULA addresses or just in case they get leaked out into the Internet.

The trick is that the global IDs must somehow be unique without being administered by a central authority. RFC 4193, *Sample Code for Pseudo-Random Global ID Algorithm*, defines a process whereby locally assigned Global IDs can be generated using a pseudorandom algorithm that gives them a very high probability of being unique. It is important that all sites generating Global IDs use the same algorithm to ensure that there is this high probability of uniqueness.

Note This section includes some information on the random Global ID algorithm for your reference. This information is not critical to your fundamental understanding of IPv6, and you can skip it if you prefer.

The algorithm defined in RFC 4193 is beyond the scope of this book, but these are the six steps from Section 3.2.2 of RFC 4193:

3.2.2. Sample Code for Pseudo-Random Global ID Algorithm

The algorithm described below is intended to be used for locally assigned Global IDs. In each case the resulting global ID will be used in the appropriate prefix as defined in Section 3.2.

1. Obtain the current time of day in 64-bit NTP format [NTP].
2. Obtain an EUI-64 identifier from the system running this algorithm. If an EUI-64 does not exist, one can be created from a 48-bit MAC address as specified in [ADDARCH]. If an EUI-64 cannot be obtained or created, a suitably unique identifier, local to the node, should be used (e.g., system serial number).
3. Concatenate the time of day with the system-specific identifier in order to create a key.
4. Compute an SHA-1 digest on the key as specified in [FIPS, SHA1]; the resulting value is 160 bits.
5. Use the least significant 40 bits as the Global ID.
6. Concatenate fc00::/7, the L bit set to 1, and the 40-bit Global ID to create a Local IPv6 address prefix.

Note The algorithm in RFC 4193 requires a /48 prefix. It does not work well if a larger prefix or contiguous prefixes are needed.

This algorithm will result in a Global ID that is reasonably unique and can be used to create a locally assigned local IPv6 address prefix. You can use the following website to generate and register your ULA address space: www.sixxs.net/tools/grh/ula.

Site-Local Addresses (Deprecated)

The original IPv6 specification allocated address space, similar to RFC 1918, *Private Address Space in IPv4*, for site-local addresses. Site-local addresses have since been deprecated (that is, made obsolete).

Site-local addresses, defined in RFC 3513, were given the prefix range fec0::/10. (You will most likely come across this prefix in older documentation.) The problem was that the term *site* was ambiguous. No one could really agree on what a site really meant. The other issue was that there was no guarantee that two sites within the same organization wouldn't end up using the same or overlapping site-local addresses, which kind of defeats the purpose of IPv6 and all this extra address space. Therefore, site-local addresses have been deprecated and replaced with unique local addresses.

IPv4 Embedded Address

The final unicast address types are IPv4 embedded addresses, as shown in Figure 4-6. IPv4 embedded addresses are IPv6 addresses used to aid the transition from IPv4 to IPv6. IPv4 embedded addresses carry an IPv4 address in the low-order 32 bits. These addresses are used to represent an IPv4 address inside an IPv6 address. RFC 4291 defines two types of IPv4 embedded addresses:

- IPv4-mapped IPv6 addresses
- IPv4-compatible IPv6 addresses (deprecated)

Special techniques such as tunnels are used to provide communications between islands of IPv6 devices over an IPv4-only network. To support this compatibility, IPv4 addresses can be embedded within an IPv6 address. This is easy to do because a 128-bit IPv6 address has plenty of room for the 32-bit IPv4 address. Basically, IPv6 just puts it at the end of the address and pads the front end. IPv4 and IPv6 packets are not compatible. Features such as NAT64 are required to translate between the two address families. See Chapter 17, "Deploying IPv6 in the Network," for more information.

IPv4-Mapped IPv6 Addresses

IPv4-mapped IPv6 addresses can be used by a dual-stack device that needs to send an IPv6 packet to an IPv4-only device. As shown in Figure 4-10, the first 80 bits are set to all 0s, and the 16-bit segment preceding the 32-bit IPv4 address is all 1s. The last 32 bits in the IPv4 address are represented in dotted-decimal notation. So, the first 96 bits are represented in hexadecimal, and the last 32 bits contain the IPv4 address in dotted-decimal notation.

With an IPv4-mapped IPv6 address, the IPv4 address does not have to be globally unique.

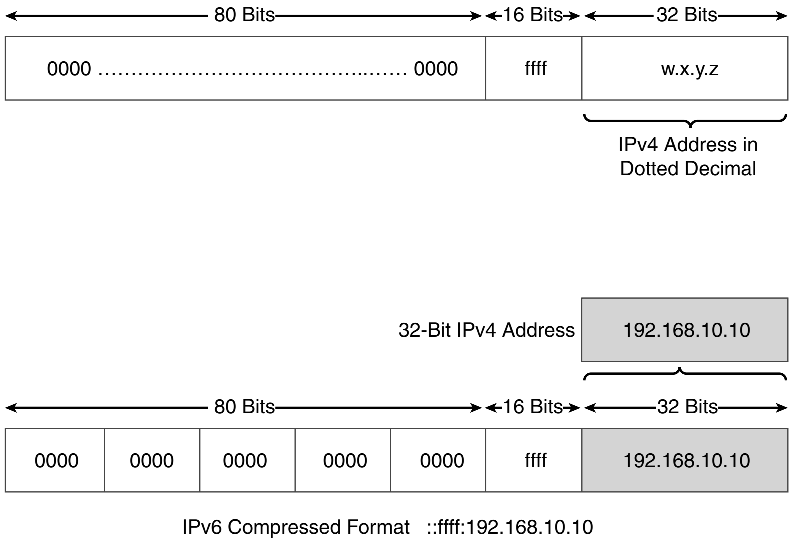


Figure 4-10 *IPv4-Mapped IPv6 Address*

Table 4-9 shows the various formats for representing an IPv4-mapped IPv6 address using the IPv4 address 192.168.10.10.

Table 4-9 *IPv4-Mapped IPv6 Address Representations*

Representation	IPv4-Mapped IPv6 Address
Preferred	0000:0000:0000:0000:0000:0000:ffff:192.168.10.10
Leading 0s omitted	0:0:0:0:0:0:ffff:192.168.10.10
Compressed	::ffff:192.168.10.10

Although there are many transition techniques available, the goal should always be native end-to-end IPv6 connectivity.

IPv4-Compatible IPv6 Addresses (Deprecated)

The deprecated IPv4-compatible IPv6 address is almost identical to an IPv4-mapped IPv6 address, except all 96 bits—including the 16-bit segment preceding the 32-bit IPv4 address—are all 0s. Another difference is that the IPv4 address used in the IPv4-compatible IPv6 address must be a globally unique IPv4 unicast address. The IPv4-compatible IPv6 address was rarely used and is now deprecated. Current IPv6 transition mechanisms no longer use this address type.

Note Chapter 18 discusses transition and coexistence strategies.

Multicast Addresses

Figure 4-11 shows the types of multicast addresses. Multicast is a technique in which a device sends a single packet to multiple destinations simultaneously (one-to-many). (Remember that a unicast address sends a single packet to a single destination [one-to-one].) Multiple destinations can actually be multiple interfaces on the same device, but they are typically different devices.

Note Figure 4-11 does not show all types of multicast addresses but is used to indicate the three multicast addresses this book focuses on.

An IPv6 multicast address defines a group of devices known as a *multicast group*. IPv6 multicast addresses use the prefix ff00::/8, shown in Table 4-10, which is equivalent to the IPv4 multicast address 224.0.0.0/4. A packet sent to a multicast group always has a unicast source address. A multicast address can never be the source address. Unlike IPv4, there is no broadcast address in IPv6. Instead, IPv6 uses multicast, including an all-IPv6 devices well-known multicast address and a solicited-node multicast address.

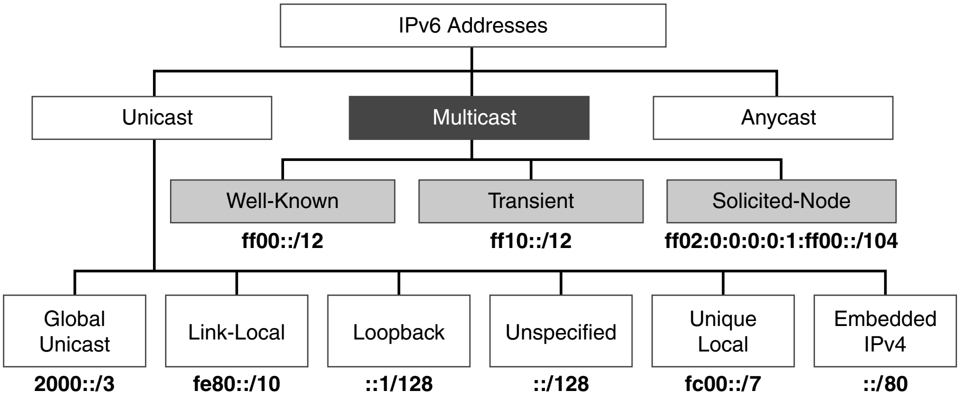


Figure 4-11 *Multicast Addresses*

Table 4-10 *IPv6 Multicast Address Representations*

Representation	IPv6 Multicast Address
Preferred	ff00:0000:0000:0000:0000:0000:0000:0000/8
Leading 0s omitted	ff00:0:0:0:0:0:0:0/8
Compressed	ff00::/8

Figure 4-12 shows the structure of an IPv6 multicast address. The first 8 bits are 1-bits (ff), followed by 4 bits allocated for flags and a 4-bit Scope field. The Scope field defines the range to which routers can forward the multicast packet. The next 112 bits represent the Group ID.

The 4 bits following 1111 1111 represent four different flags. The first three flags, 0 (reserved), R (rendezvous point), and P (network prefix), are beyond the scope of this book. The fourth flag, the least significant bit (LSB), or rightmost bit, is the transient flag (T flag). The T flag denotes the two types of multicast addresses:

- **Permanent (0):** These addresses, known as *predefined multicast addresses*, are assigned by IANA and include both well-known and solicited multicast.
- **Nonpermanent (1):** These are "transient" or "dynamically" assigned multicast addresses. They are assigned by multicast applications.

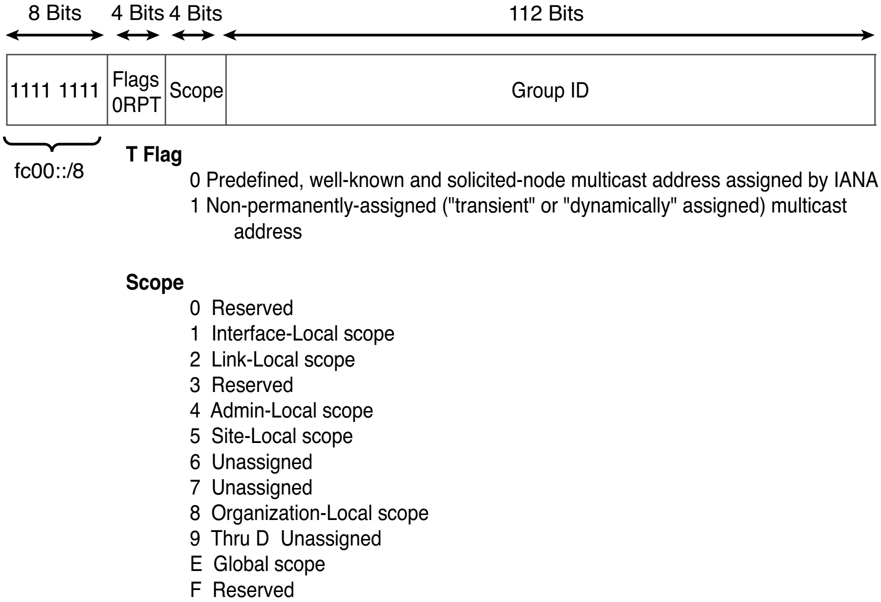


Figure 4-12 *IPv6 Multicast Address*

As shown in Figure 4-11, there are two types of predefined multicast addresses, both of which use the Flag field with a 0x0 value:

- Well-known multicast addresses
- Solicited-node multicast addresses

Note For additional information on IPv6 multicast and multicast routing, I highly suggest resources by Tim Martin, Cisco Systems, including the video *IPv6 Summit 2015: IPv6 Multicast Technologies*, at www.youtube.com/watch?v=H6bBiIPfYXM. Tim Martin also has an excellent Cisco Press LiveLessons video series, *IPv6 Design & Deployment LiveLessons* (see lesson 5).

Well-Known Multicast Addresses

Well-known multicast addresses have the prefix ff00::/12. As shown in Figure 4-12, this means that the third hexadecimal digit, the Flag field, is always set to 0. Well-known multicast addresses are predefined or reserved multicast addresses for assigned groups of devices. These addresses are equivalent to IPv4 well-known multicast addresses in the range 224.0.0.0 to 239.255.255.255. Some examples of IPv6 well-known multicast addresses include the following:

- **ff02::1:** All IPv6 devices
- **ff02::2:** All IPv6 routers
- **ff02::5:** All OSPFv3 routers
- **ff02::a:** All EIGRP (IPv6) routers

Solicited-Node Multicast Addresses

Solicited-node multicast addresses are used as a more efficient approach to IPv4's broadcast address. As discussed in Chapter 2, the solicited-node multicast is used in Layer 3-to-Layer 2 address resolution, similar to how Address Resolution Protocol (ARP) is used in IPv4. Solicited-node multicast addresses are automatically created using a special mapping of the device's unicast address with the solicited-node multicast prefix ff02:0:0:0:0:1:ff00::/104. Solicited-node multicast addresses are automatically created for every unicast address on a device.

Note Multicast addresses, the Scope field, assigned multicast, and solicited-node multicast are discussed in detail in Chapter 7.

Anycast Addresses

The last type of IPv6 address examined in this chapter is the anycast address (see Figure 4-13). An IPv6 anycast address is an address that can be assigned to more than one interface (typically different devices). In other words, multiple devices can have the same anycast address. A packet sent to an anycast address is routed to the "nearest" interface having that address, according to the router's routing table.

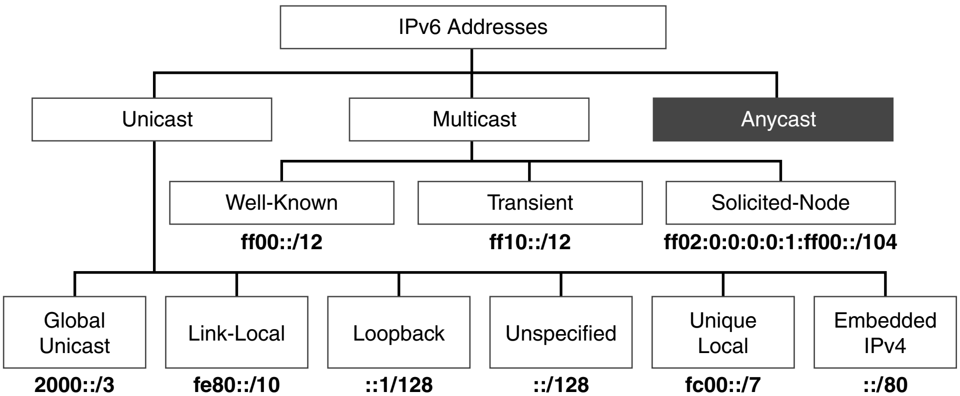


Figure 4-13 *Anycast Addresses*

Anycast addresses are available for both IPv4 and IPv6, initially defined in RFC 1546, *Host Anycasting Service*. Anycast was meant to be used for services such as DNS and HTTP but was never really implemented as designed.

There is no special prefix for an IPv6 anycast address. An IPv6 anycast address uses the same address range as global unicast addresses. Each participating device is configured to have the same anycast address. For example, servers A, B, and C in Figure 4-14 could be DHCPv6 servers with a direct Layer 3 connection into the network. These servers could advertise the same /128 address using OSPFv3. The router nearest the client request would then forward packets to the *nearest* server identified in the routing table.

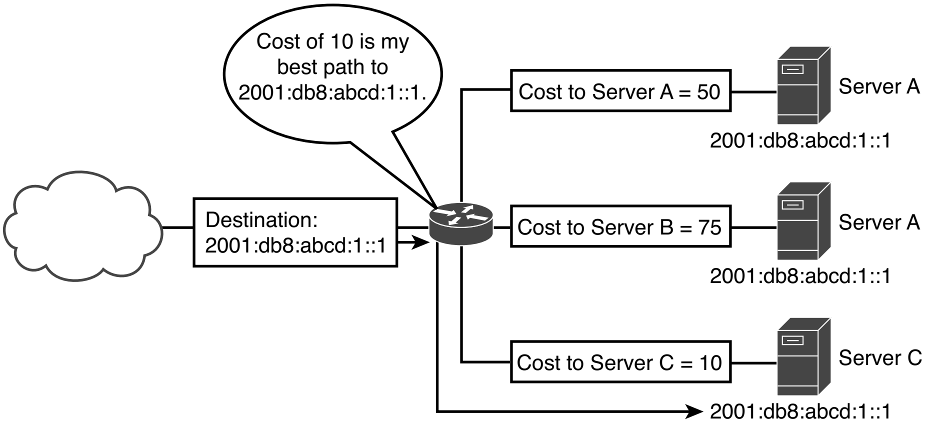


Figure 4-14 *Example of Anycast Addressing*

There are some reserved anycast address formats such as the subnet-router anycast address defined in RFC 4291 and RFC 2526. IPv6 anycast addressing is still somewhat in the experimental stages and beyond the scope of this book.

Summary

This chapter explains the basics of IPv6 addressing. The preferred format of an IPv6 128-bit address is written as eight 16-bit segments (hextets), separated by colons. The notation of the address can be reduced by omitting leading 0s and by using the double colon to replace contiguous hextets of 0s.

The IPv6 address space is extremely large. IPv6, with its 128-bit address space, provides for 340 undecillion addresses. Currently, only one-eighth of this space has been allocated for global unicast addresses, and a very small portion has been allocated for other unicast and multicast addresses.

This chapter introduces the three types of IPv6 addresses: unicast, multicast, and anycast. The following is a brief description of each of the addresses as discussed in this chapter:

- **Unicast addresses:** A unicast address uniquely identifies an interface on an IPv6 device. A source IPv6 address must be a unicast address. There are several types of unicast addresses:
  - **Global unicast addresses (GUAs):** Global unicast addresses are also known as an *aggregatable global unicast address*. These addresses are globally routable and reachable on the IPv6 Internet. They are equivalent to public IPv4 addresses. The current GUA address assignment from IANA begins with the binary value 001 or the prefix 2000::/3.
  - **Link-local addresses:** A link-local unicast address (fe80::/10) is a unicast address that is confined to a single link. The uniqueness of this address only has to be assured on that link because these packets are not routable off the link. An IPv6-enabled device must have a link-local address. Link-local unicast addresses are usually automatically created but can also be manually configured.
  - **Loopback addresses:** A loopback address is an all-0s address except for the last bit, which is set to 1. It is equivalent to the IPv4 loopback address, 127.0.0.1.
  - **Unspecified addresses:** An unspecified unicast address is an all-0s address. It cannot be assigned to an interface. An unspecified unicast address is used as a source address to indicate the absence of an address.
  - **Unique local addresses:** A unique local address (fc00::/7) is similar to the RFC 1918 private address space in IPv4. Unique local addresses should not be routable in the global Internet. They are to be used in more limited areas, such as within a site, or routed between a limited number of sites.
  - **IPv4 embedded addresses:** IPv6 addresses aid in the transition from IPv4 to IPv6. An IPv4 embedded address carries an IPv4 address in the low-order 32 bits. This type of address is used to represent an IPv4 address inside an IPv6 address. IPv4-mapped IPv6 addresses are the current type of IPv4 embedded addresses, with IPv4-compatible IPv6 addresses having been deprecated.
- **Multicast addresses:** Multicast is a technique used in which a device sends a single packet to multiple destinations simultaneously. This chapter introduces two types of multicast addresses:
  - **Well-known multicast addresses:** These multicast addresses are reserved for predefined groups of devices, such as all-IPv6 nodes and all-IPv6 routers multicast groups.
  - **Solicited-node addresses:** Every unicast address assigned to an interface also has a special multicast address known as a solicited-node multicast address. These multicast addresses are automatically created using a special mapping by prepending the solicited-node multicast prefix ff02:0:0:0:0:1:ff00::/104 to the last 24 bits of the unicast address. IPv6's solicited-node multicast address provides a way to reach every device on the link without all those devices needing to process the contents of the packet.

- **Anycast addresses:** An IPv6 anycast address is an address that can be assigned to more than one interface (typically different devices). In other words, multiple devices can have the same anycast address. A packet sent to an anycast address is routed to the "nearest" interface having that address, according to the router's routing table.

There is no broadcast address in IPv6. Instead, IPv6 uses multicast addresses such as the solicited-node multicast and all-IPv6 devices multicast.

Review Questions

1. Convert the following IPv6 address to its most compressed format, using the RFC 5952 standard for multiple strings of all-0s hextets:

   2001:0db8:cab0:0234:0034:0004:0000:0000
2. Convert the following IPv6 address to its most compressed format, using the RFC 5952 standard for multiple strings of all-0s hextets:

   2001:0db8:0cab:0000:0000:0000:0001:0000
3. Convert the following IPv6 address to its most compressed format, using the RFC 5952 standard for multiple strings of all-0s hextets:

   2001:0db8:0cab:1234:0230:1200:0034:0000
4. Convert the following IPv6 address to its most compressed format, using the RFC 5952 standard for multiple strings of all-0s hextets:

   fd00:0000:0000:0000:1234:0000:0000:0000
5. Convert the following IPv6 address to its most compressed format, using the RFC 5952 standard for multiple strings of all-0s hextets:

   2001:0db8:0000:0000:1234:0000:0000:1000
6. Convert this compressed IPv6 address to the complete address with 32 hexadecimal digits:

   2001:db8:cab::1
7. Convert this compressed IPv6 address to the complete address with 32 hexadecimal digits:

   2001:db8:0:0:234::
8. What is the prefix for the address 2001:db8:80f:f425::230/64?
9. What is the prefix for the address 2001:db8:80f:f425:250:56ff:fe83:ecc/64?
10. What is the prefix for the address fe80::250:56ff:fe83:ecc/64?
11. What is the prefix for the address 2001:db8:80f:f425:250:56ff:fe83:ecc/48?
12. What is the prefix for the address 2001:db8:80f:f425::230/48?
13. What is the prefix for the address 2001:db8:bb8a:f390::1/32?
14. What are the three fields in a global unicast address?
15. What is the range of the first hextet of a global unicast address?
16. Which type of address is required for a device to be IPv6-enabled?
17. What is the range of the first hextet of a link-local unicast address?

18. What are three characteristics of a link-local unicast address?
19. What unicast address is an all-0s address?
20. What are two characteristics of an unspecified unicast address?
21. What type of IPv6 unicast address is similar to IPv4 private addresses?
22. What is the range of the first hextet of a unique local address?
23. What is the difference between IPv6 unique local addresses and IPv4 private addresses in terms of NAT?
24. What are the first two hexadecimal digits in a multicast address?
25. What multicast address that is used in address resolution with IPv6 is similar to ARP with IPv4?

References

Endnote

1. R. Hinden, "IP Next Generation Overview," *Communications of the ACM*, Volume 39, Issue 6, June 1996, pp. 61–71.

RFCs

RFC 1546, *Host Anycasting Service*, C. Partridge, www.ietf.org/rfc/rfc1543.txt, November 1993.

RFC 1918, *Address Allocation for Private Internets*, Y. Rekhter, Cisco Systems, www.ietf.org/rfc/rfc1918.txt, February 1996.

RFC 2373, *IP Version 6 Addressing Architecture*, R. Hinden, Nokia, www.ietf.org/rfc/rfc2373.txt, July 1998.

RFC 2374, *An IPv6 Aggregatable Global Unicast Address Format*, R. Hinden, Nokia, www.ietf.org/rfc/rfc2374.txt, July 1998.

RFC 2375, *IPv6 Multicast Address Assignments*, R. Hinden, Ipsilon Networks, www.ietf.org/rfc/rfc2375.txt, July 1998.

RFC 2526, *Reserved IPv6 Subnet Anycast Addresses*, D. Johnson, Carnegie Mellon University, www.ietf.org/rfc/rfc2526.txt, March 1998.

RFC 2993, *Architectural Implications of NAT*, T. Hain, Microsoft, www.ietf.org/rfc/rfc2993.txt, November 2000.

RFC 3306, *Unicast-Prefix-Based IPv6 Multicast Addresses*, B. Haberman, www.ietf.org/rfc/rfc3306.txt, August 2002.

RFC 3513, *Internet Protocol Version 6 (IPv6) Addressing Architecture*, R. Hinden, Nokia, www.ietf.org/rfc/rfc3513.txt, April 2003.

RFC 3587, *IPv6 Global Unicast Address Format*, R. Hinden, Nokia, www.ietf.org/rfc/rfc3587.txt, March 2005.

RFC 4038 *Application Aspects of IPv6 Transition*, M-K Shin, ETRI/NIST, www.ietf.org/rfc/rfc4038.txt, August 2003.

RFC 4193, *Unique Local IPv6 Unicast Addresses*, R. Hinden, Nokia, www.ietf.org/rfc/rfc4193.txt, October 2005.

RFC 4291, *IP Version 6 Addressing Architecture*, R. Hinden, Nokia, www.ietf.org/rfc/rfc4291.txt, February 2006.

RFC 4861, *Neighbor Discovery for IP version 6 (IPv6)*, Y. Narten, IMB, www.ietf.org/rfc/rfc4861.txt, September 2007.

RFC 4864, *Local Network Protection for IPv6*, G. Van de Velde, www.ietf.org/rfc/rfc4864.txt, May 2007.

RFC 5902, *IAB Thoughts on IPv6 Network Address Translation*, D. Thaler, www.ietf.org/rfc/rfc5902.txt, July 2010.

RFC 5952, *A Recommendation for IPv6 Address Text Representation*, S. Kawamura, NEC Biglobe, Ltd., www.ietf.org/rfc/rfc5952.txt, August 2010.

RFC 6296, *IPv6-to-IPv6 Network Prefix Translation*, M. Wasserman, Painless Security, www.ietf.org/rfc/rfc6296.txt, June 2011.

IPv6-to-IPv6 Network Address Translation (NAT66), draft-mrw-behave-nat66-02.txt, M. Wasserman, Sandstorm Enterprises, tools.ietf.org/html/draft-mrw-behave-nat66-02, November 2008.

Websites

IANA, *Internet Protocol Version 6 Address Space*, www.iana.org/assignments/ipv6-address-space/ipv6-address-space.txt

IANA, *IPv6 Global Unicast Address Assignments*, www.iana.org/assignments/ipv6-unicast-address-assignments/ipv6-unicast-address-assignments.xml

Ed Horley's blog, www.howfunky.com, an excellent resource for IPv6.

Book

Practical IPv6 for Windows Administrators, by Ed Horley, Apress, December 2013.

Index

Numbers

A

B

C

D

E

F

G

J

L

M

N

O

P

R

S

T

U

V

W

Z