Cisco Systems

CCIE® Professional Development

Troubleshooting IP Routing Protocols

The comprehensive, hands-on guide for resolving IP routing problems

Zaheer Aziz, CCIE No. 4127
Johnson Liu, CCIE No. 2637
Abe Martey, CCIE No. 2373
Faraz Shamim, CCIE No. 4131

ciscopress.com

Troubleshooting IP Routing Protocols

Zaheer Aziz, CCIE #4127
Johnson Liu, CCIE #2637
Abe Martey, CCIE #2373
Faraz Shamim, CCIE #4131

Cisco Press

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

Troubleshooting IP Routing Protocols

Zaheer Aziz, Johnson Liu, Abe Martey, Faraz Shamim

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

Printed in the United States of America 3 4 5 6 7 8 9 0

Third Printing January 2004

Library of Congress Cataloging-in-Publication Number: 2001086619

ISBN: 1-58705-019-6

Warning and Disclaimer

This book is designed to provide information about troubleshooting IP routing protocols, including RIP, IGRP, EIGRP, OSPF, IS-IS, PIM, and BGP. Every effort has been made to make this book as complete and as accurate as possible, but no warranty or fitness is implied.

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

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

Trademark Acknowledgments

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

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 e-mail at feedback@ciscopress.com. Please be sure to include the book title and ISBN in your message.

We greatly appreciate your assistance.

Publisher	John Wait
Editor-in-Chief	John Kane
Cisco Systems Management	Michael Hakkert
	Tom Geitner
	William Warren
Production Manager	Patrick Kanouse
Executive Editor	Brett Bartow
Acquisitions Editor	Amy Lewis
Development Editor	Christopher Cleveland
Project Editor	San Dee Phillips
Copy Editor	Krista Hansing
Technical Editors	Brian Morgan, Harold Ritter, John Tiso
Team Coordinator	Tammi Barnett
Book Designer	Gina Rexrode
Cover Designer	Louisa Adair
Composition	Publication Services, Inc.
Indexer	Tim Wright

CISCO SYSTEMS

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

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

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

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

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

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

Printed in the USA

About the Authors

Zaheer Aziz, CCIE #4127, is a network consulting engineer in the Internet Infrastructure Services group for Cisco Systems, Inc. Zaheer provides consulting services to major ISPs in the MPLS and IP routing protocols area. In his last five years at Cisco, Zaheer has been actively involved in speaking at Cisco Networkers conferences and at several Cisco events. Zaheer occasionally writes for *Cisco Packet* magazine and for *Spider Internet* magazine, Pakistan on topics of MPLS and BGP. He holds a master's degree in electrical engineering from Wichita State University, Wichita, KS and enjoys reading and playing cricket and Ping-Pong. Zaheer is married and has a loving five-year-old boy, Taha Aziz.

Johnson Liu, CCIE #2637, is a senior customer network engineer with the Advance Network Services Team for the enterprise in Cisco Systems. He obtained his MSEE degrees at the University of Southern California and has been with Cisco Systems for more than five years. He is the technical editor for other Cisco Press books, including *Internet Routing Architectures* and *Large-Scale IP Network Solutions*. Johnson has been involved in many large-scale IP network design projects involving EIGRP, OSPF, and BGP for large enterprise and service provider customers. Johnson is also a regular speaker for deploying and troubleshooting EIGRP at the Networkers conference.

Abe Martey, CCIE #2373, is a product manager of the Cisco 12000 Internet Router Series. Abe specializes in high-speed IP routing technologies and systems. Prior to this position, Abe worked as a support engineer in the Cisco Technical Assistance Center (TAC), specializing in IP routing protocols and later on the ISP Team (now Infrastructure Engineering Services Team), where he worked closely with tier one Internet service providers. Abe holds a master's degree in electrical engineering and has been with Cisco Systems for over six years. Abe is also the author of *IS-IS Design Solutions* from Cisco Press.

Faraz Shamim, CCIE #4131, is a network consulting engineer with the Advance Network Services Team for the Service Provider (ANS-SP) for Cisco Systems, Inc. He provides consulting services to his dedicated Internet service providers. Faraz wrote several documents, white papers, and technical tips for ODR, OSPF, RIP, IGRP, EIGRP, and BGP on Cisco Connection Online (CCO), (www.cisco.com). Faraz has also been engaged in developing and teaching the Cisco Internetworking Basic and Advance Bootcamp Training for Cisco new-hire engineers. He has also taught the Cisco Internetworking Bootcamp Course to MS students at the University of Colorado at Boulder (BU) and Sir Syed University of Engineering & Technology (SSUET), Karachi, Pakistan. Faraz has been a visiting faculty member for SSUET and also gave a lecture on OSPF to Lahore University of Management & Sciences (LUMS), Lahore, Pakistan. Faraz has been engaged in developing CCIE lab tests and proctoring the CCIE lab. Faraz actively speaks at the Networkers conference on the subject of OSPF. Like other authors of this book, he also started his career at the Cisco Technical Assistant Center (TAC) providing support for customers in IP routing protocols. Faraz has been with Cisco Systems for five years.

About the Technical Reviewers

Brian Morgan, CCIE #4865 , CCSI, is the Director of Data Network Engineering at Allegiance Telecom, Inc. He has been in the networking industry for more than 12 years. Before going to Allegiance, Morgan was an instructor/consultant teaching ICND, BSCN, BSCI, CATM, CVOICE, and BCRAN. He is a co-author of the *Cisco CCNP Remote Access Exam Certification Guide* and a technical editor of numerous Cisco Press titles.

Harold Ritter, CCIE # 4168 , is a network consulting engineer for Cisco Advanced Network Services. He is responsible for helping Cisco top-tier customers to design, implement, and troubleshoot routing protocols in their environment. He has been working as a network engineer for more than eight years.

John Tiso, CCIE #5162, is one of the senior technologists of NIS, a Cisco Systems Silver partner. He has a bachelor of science degree from Adelphi University. Tiso also holds the CCDP certification, Cisco Security and Voice Access Specializations, and Sun Microsystems, Microsoft, and Novell certifications. He has been published in several industry publications. He can be reached through e-mail at john@jtiso.com.

Dedications

Zaheer Aziz:

I would like to dedicate this book to my late father (may God bless his soul) for his struggling life for betterment of our life, to a person whose self-made, hardworking, and not-so-easy life history became a catalyst for the relatively little hard work I have put in my life. Undoubtedly, he would have tremendously enjoyed seeing this book, but he is not here. Truly, his Air Force blood would have rushed fast seeing this book, but he is not here. Verily, he would have immensely applauded me in seeing this book, but he is not here. Therefore, I want my mother, who has put in equal hard work in our life, to enjoy this accomplishment and success. She deserves equal credit in the success of our family, and I wish her a very long and happy life.

Johnson Liu:

I dedicate this book with my deepest love and affection to my wife, Cisco Liu, who has given me the inspiration and support to write this book.

Abe Martey:

I'd like to dedicate this book to all previous and current engineers of the Cisco Worldwide TAC for their remarkable enthusiasm, dedication, and excellence in providing technical and troubleshooting assistance to network operators in every corner of our planet and in space.

Faraz Shamim:

I would like to dedicate this book to my parents, whose favors I can never return and whose prayers I will always need. To my wife, who encouraged me when I felt too lazy to write, and to my sons, Ayaan and Ameel, who waited patiently for my attention on many occasions.

Acknowledgments

Faraz Shamim:

Alhamdulillah! I thank God for giving me the opportunity to write this book, which I hope will help many people in resolving their routing issues.

I would like to thank my manager, Srinivas Vegesna, and my previous manager and mentor, Andrew Maximov, for supporting me in this book project. Special thanks goes to Bob Vigil, who let me use some of his presentation material in the RIP and IGRP chapter. I would also like to thank Alex Zinin for clearing some of my OSPF concepts that I used in this book. I would like to thank my co-authors, Zaheer Aziz, Abe Martey, and Johnson Liu, who put up with my habit of reminding them of their chapter deadlines. I would also like to thank Chris Cleveland and Amy Lewis of Cisco Press for their understanding whenever we were late in submitting our chapters.

Zaheer Aziz:

All thanks to God for giving me strength to work on this book. I heartily thank my wife for her support, patience, and understanding that helped me put in many hours on this book. I appreciate the flexibility of my employer, Cisco Systems, Inc. (in particular, my manager, Srinivas Vegesna) for allowing me to work on this book while keeping my day job. Many thanks to Syed Faraz Shamim (lead author of this book), who invited me through a cell-phone call from San Jose to Washington, D.C., where I was attending IETF 46 in 1999, to co-author this book. Thanks to Moiz Moizuddin for independently reviewing the technical content of my chapters. I would like to credit my mentor, Syed Khalid Raza, for his continuous guidance and for showing me the world of BGP. Finally, I wish to thank Cisco Press, who made this book possible—in particular, Christopher Cleveland and Brian Morgan, whose suggestions greatly improved the quality of this book and made this process go smoothly.

Johnson Liu:

I would like to thank my friends and colleagues at Cisco Systems, with whom I spent many late hours with trying to troubleshoot P1 routing protocol problems. Their professionalism and knowledge are simply unparalleled. Special thanks to my managers, Andrew Maximow and Raja Sundaram, who have given me all their support throughout my career at Cisco Systems. Finally, I would like to thank my technical editors for their invaluable input and suggestions to improve this book.

Abe Martey:

First of all, I'd like to express sincere thanks to the co-authors and colleagues at work, Faraz, Johnson, and Zaheer for dreaming up this title and inviting me to participate in its materialization. We all worked on the Cisco Technical Assistance Center (TAC) Routing Protocol Team, giving us quite a bit of experience troubleshooting IP routing problems. This work is our attempt to generously share that experience with a larger audience beyond the Cisco Systems work environment.

I received a lot of support, mentorship, and training from many Cisco TAC and development engineers, as well as many direct and nondirect managers as a TAC Engineer. Hats off to this unique breed of talented individuals, women and men, who have committed their lives to keep the Internet running. I'd also like to thank these folks (too many of them to name here) for every bit of knowledge and wisdom that they've shared with me over the years.

Over time, I've developed great personal relationships with various networking professionals worldwide, all of whom I met as customers or through IETF, NANOG, IEEE, and other professional conferences and meetings. I'd like to sincerely thank them for sharing with me their knowledge and expertise, as well as their professional insights and visions into the future of networking technology.

I'd also like to express my sincerest gratitude to Amy Lewis and Chris Cleveland, both of Cisco Press, and the technical editors for their roles in helping bring this book to fruition. Many thanks to several close relatives for their support and encouragement all through this project.

Contents at a Glance

Table of Contents

Preface

Sitting in my office at Cisco on the third floor of building K, I read an e-mail from Kathy Trace from Cisco Press asking if I was interested in writing a book. She had read my technical tips that I had written for Cisco Connection Online and said that she wanted me as an author for Cisco Press. I was very enthusiastic about it and said to myself, "Yeah! It's a great idea! Let's write a book!" But on what subject?

One of the topics that I had in mind was OSPF. Johnson used to sit right in front of my office at that time. I asked him, "Hey, Johnson! You want to write a book with me?" He screamed, "A book!" I said, "Yeah, a book! What do you think?" He thought for a minute and said, "Well, what is left for us to write a book on? Cisco Press authors have written books on almost every routing topic. . . . But there *is* one subject that has not been covered in one single book—troubleshooting IP routing protocols."

Apparently, Johnson got the idea to write a troubleshooting book from his wife. Whenever Johnson's wife calls him at work, he has to put her on hold because he is busy troubleshooting a customer's problem. His wife, whose name is also Cisco, then gave him the idea of writing a troubleshooting book so that customers would have a troubleshooting guide on routing protocols that they can refer to so that they can successfully solve their problems before opening a case.

The idea was indeed great. No books had been written on this particular subject before. I then called Zaheer, who was attending IETF 46 in Washington, D.C., and told him about this; he also agreed that the idea was a good one. So now we had a team of three TAC engineers who had spent the last three to four years in TAC dealing with routing problems—and each one of us was an expert in one or two protocols. Our manager, Raja Sundaram, used to say, "I want you to pick up a protocol and become an expert in it." My area of expertise was OSPF, Johnson was a guru of EIGRP and multicasting, and Zaheer shone with his BGP knowledge. Very soon, we realized that we were missing one important protocol, IS-IS. Our exposure with IS-IS was not at a level that we could write a whole chapter on troubleshooting IS-IS, so Zaheer suggested Abe Martey for this job. Abe was already engaged in writing a book on IS-IS with Cisco Press, but after seeing our enthusiasm about this book, he agreed to become a member of our author team.

When we started working on these chapters, we realized that we were working on something that a routing network administrator had always dreamed of—a troubleshooting book that contains solutions for all the IP routing protocol problems. The data that we collected for this book came from the actual problems we have seen in customer networks in our combined 20 years of experience in troubleshooting IP networks. We wanted to make it a one-stop shop for troubleshooting guidance and reference. So, we provided the "understanding protocols" chapters along with troubleshooting to help you, the reader, go back to a specific protocol and refresh your memory. This book is also an excellent resource for preparation for the CCIE certification. This book should teach you how to tackle any IP routing problem that pops up in your network. All possible cases might not be discussed, but general guidelines and techniques teach a logical approach for solving typical problems that you might face.

Syed Faraz Shamim

Introduction

As the Internet continues to grow exponentially, the need for network engineers to build, maintain, and troubleshoot the growing number of component networks also has increased significantly. Because network troubleshooting is a practical skill that requires on-the-job experience, it has become critical that the learning curve necessary to gain expertise in internetworking technologies be reduced to quickly fill the void of skilled network engineers needed to support the fast-growing Internet. IP routing is at the core of Internet technology, and expedient troubleshooting of IP routing failures is key to reducing network downtime. Reducing network downtime is crucial as the level of mission-critical applications carried over the Internet increases. This book gives you the detailed knowledge to troubleshoot network failures and maintain the integrity of their networks.

Troubleshooting IP Routing Protocols provides a unique approach to troubleshooting IP routing protocols by focusing on step-by-step guidelines for solving a particular routing failure scenario. The culmination of years of experience with Cisco's TAC group, this book offers sound methodology and solutions for resolving routing problems related to BGP, OSPF, IGRP, EIGRP, IS-IS, RIP, and PIM by first providing an overview to routing and then concentrating on the troubleshooting steps that an engineer would take in resolving various routing protocol issues that arise in a network. This book offers you a full understanding of troubleshooting techniques and real-world examples to help you hone the skills needed to successfully complete the CCIE exam, as well as perform the duties expected of a CCIE-level candidate.

Who Should Read This Book?

This is an intermediate-level book that assumes that you have a general understanding of IP routing technologies and other related protocols and technologies used in building IP networks.

The primary audience for this book consists of network administrators and network operation engineers responsible for the high availability of their networks, or those who plan to become Cisco Certified Internetwork Experts.

How This Book Is Organized

Although this book could be read cover to cover, it is designed to be flexible and to allow you to easily move between chapters and sections of chapters to cover just the material that you need more work with.

- **Chapter 1, "Understanding IP Routing"**—This chapter provides an overview of IP routing protocols with focus on the following topics:
  - —IP addressing concepts
  - —Static and dynamic routes
  - —Dynamic routing
  - —Routing protocol administrative distance
  - —Fast forwarding in routers

The remaining chapters alternate between chapters that provides coverage of key aspects of a specific routing protocol and chapters devoted to practical, real-world troubleshooting methods for that routing protocol. The list that follows provides more detailed information:

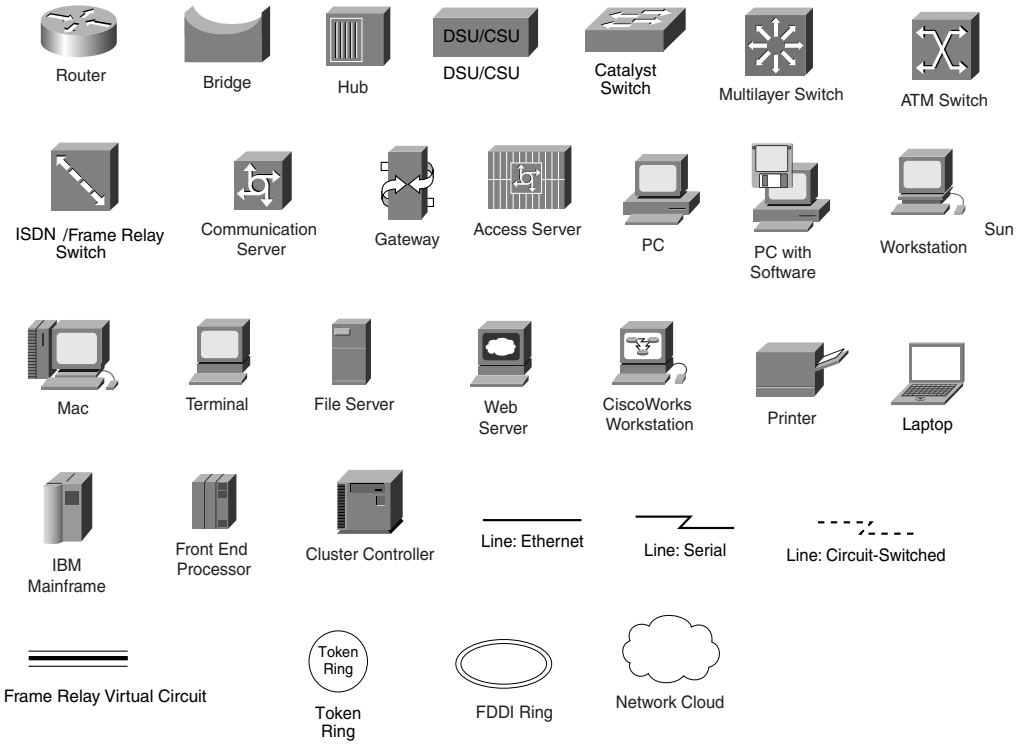
- **Chapter 2, "Understanding Routing Information Protocol (RIP)"**—This chapter focuses on the key aspects of RIP needed to confidently troubleshoot RIP problems. Topics include the following:
  - —Metrics
  - —Timers
  - —Split horizon
  - —Split horizon with poison reverse
  - —RIP-1 packet format
  - —RIP behavior
  - —Why RIP doesn't support discontiguous networks
  - —Why RIP doesn't support variable-length subnet masking (VLSM)
  - —Default routes and RIP
  - —Protocol extension to RIP
  - —Compatibility issues
- **Chapter 3, "Troubleshooting RIP"**—This chapter provides a methodical approach to resolving common RIP problems, which include the following:
  - —Troubleshooting RIP route installation
  - —Troubleshooting RIP route advertisement
  - —Troubleshooting routes summarization in RIP
  - —Troubleshooting RIP redistribution problems
  - —Troubleshooting dial-on-demand routing (DDR) issues in RIP
  - —Troubleshooting the route-flapping problem in RIP
- **Chapter 4, "Understanding Interior Gateway Routing Protocol (IGRP)"**—This chapter focuses on the key aspects of IGRP needed to confidently troubleshoot IGRP problems. Topics include the following:
  - —Metrics
  - —Timers
  - —Split horizon
  - —Split horizon and poison reverse
  - —IGRP packet format
  - —IGRP behavior
  - —Default route and IGRP
  - —Unequal-cost load balancing in IGRP

- **Chapter 5, "Troubleshooting IGRP"**—This chapter provides a methodical approach to resolving common IGRP problems, which include the following:
  - —Troubleshooting IGRP route installation
  - —Troubleshooting IGRP route advertisement
  - —Troubleshooting IGRP redistribution problems
  - —Troubleshooting dial-on-demand routing (DDR) issues in IGRP
  - —Troubleshooting route flapping in IGRP
  - —Troubleshooting variance problem
- **Chapter 6, "Understanding Enhanced Interior Gateway Routing Protocol (EIGRP)"**—This chapter focuses on the key aspects of EIGRP needed to confidently troubleshoot EIGRP problems. Topics include the following:
  - —Metrics
  - —EIGRP neighbor relationships
  - —The Diffusing Update Algorithm (DUAL)
  - —DUAL finite state machine
  - —EIGRP reliable transport protocol
  - —EIGRP packet format
  - —EIGRP behavior
  - —EIGRP summarization
  - —EIGRP query process
  - —Default route and EIGRP
  - —Unequal-cost load balancing in EIGRP
- **Chapter 7, "Troubleshooting EIGRP"**—This chapter provides a methodical approach to resolving common EIGRP problems, which include the following:
  - —Troubleshooting EIGRP neighbor relationships
  - —Troubleshooting EIGRP route advertisement
  - —Troubleshooting EIGRP route installation
  - —Troubleshooting EIGRP route flapping
  - —Troubleshooting EIGRP route summarization
  - —Troubleshooting EIGRP route redistribution
  - —Troubleshooting EIGRP dial backup
  - —EIGRP error messages

- **Chapter 8, "Understanding Open Shortest Path First (OSPF)"**—This chapter focuses on the key aspects of OSPF needed to confidently troubleshoot OSPF problems. Topics include the following:
  - —OSPF packet details
  - —OSPF LSA details
  - —OSPF areas
  - —OSPF media types
  - —OSPF adjacencies
- **Chapter 9, "Troubleshooting OSPF"**—This chapter provides a methodical approach to resolving common OSPF problems, which include the following:
  - —Troubleshooting OSPF neighbor relationships
  - —Troubleshooting OSPF route advertisement
  - —Troubleshooting OSPF route installation
  - —Troubleshooting redistribution problems in OSPF
  - —Troubleshooting route summarization in OSPF
  - —Troubleshooting CPUHOG problems
  - —Troubleshooting dial-on-demand routing (DDR) issues in OSPF
  - —Troubleshooting SPF calculation and route flapping
  - —Common OSPF error messages
- **Chapter 10, "Understanding Intermediate System-to-Intermediate System (IS-IS)"**—This chapter focuses on the key aspects of IS-IS needed to confidently troubleshoot IS-IS problems. Topics include the following:
  - —IS-IS protocol overview
  - —IS-IS protocol concepts
  - —IS-IS link-state database
  - —Configuring IS-IS for IP routing
- **Chapter 11, "Troubleshooting IS-IS"**—This chapter provides a methodical approach to resolving common IS-IS problems, which include the following:
  - —Troubleshooting IS-IS adjacency problems
  - —Troubleshooting IS-IS routing update problems
  - —IS-IS errors
  - —CLNS **ping** and **traceroute**
  - —Case study: ISDN configuration problem

- **Chapter 12, "Understanding Protocol Independent Multicast (PIM)"**—This chapter focuses on the key aspects of PIM needed to confidently troubleshoot PIM problems. Topics include the following:
  - — Fundamentals of IGMP Version 1, IGMP Version 2, and reverse path forwarding (RPF)
  - —PIM dense mode
  - —PIM sparse mode
  - —IGMP and PIM packet format
- **Chapter 13, "Troubleshooting PIM"**—This chapter provides a methodical approach to resolving common PIM problems, which include the following:
  - —IGMP joins issues
  - —PIM dense mode issues
  - —PIM sparse mode issues
- **Chapter 14, "Understanding Border Gateway Protocol Version 4 (BGP-4)"**—This chapter focuses on the key aspects of BGP needed to confidently troubleshoot BGP problems. Topics include the following:
  - —BGP-4 protocol specification and functionality
  - —Neighbor relationships
  - —Advertising routes
  - —Synchronization
  - —Receiving routes
  - —Policy control
  - —Scaling IBGP networks (route reflectors and confederations)
  - —Best-path calculation
- **Chapter 15, "Troubleshooting BGP"**—This chapter provides a methodical approach to resolving common BGP problems, which include the following:
  - —Troubleshooting BGP neighbor relationships
  - —Troubleshooting BGP route advertisement/origination and receiving
  - —Troubleshooting a BGP route not installing in a routing table
  - —Troubleshooting BGP when route reflectors are used
  - —Troubleshooting outbound traffic flow issues because of BGP policies
  - —Troubleshooting load-balancing scenarios in small BGP networks
  - —Troubleshooting inbound traffic flow issues because of BGP policies
  - —Troubleshooting BGP best-path calculation issues
  - —Troubleshooting BGP filtering

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:

- 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.
- **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.

The primary objective of this book is to provide elaborate guidance for troubleshooting Internet Protocol (IP) routing problems on Cisco routers. In this regard, the subsequent text covers well-known routing protocols such as the following:

- Open Shortest Path First Protocol (OSPF)
- Integrated Intermediate System-to-Intermediate System Protocol (IS-IS)
- Border Gateway Protocol (BGP)
- Protocol Independent Multicast (PIM) for multicast routing

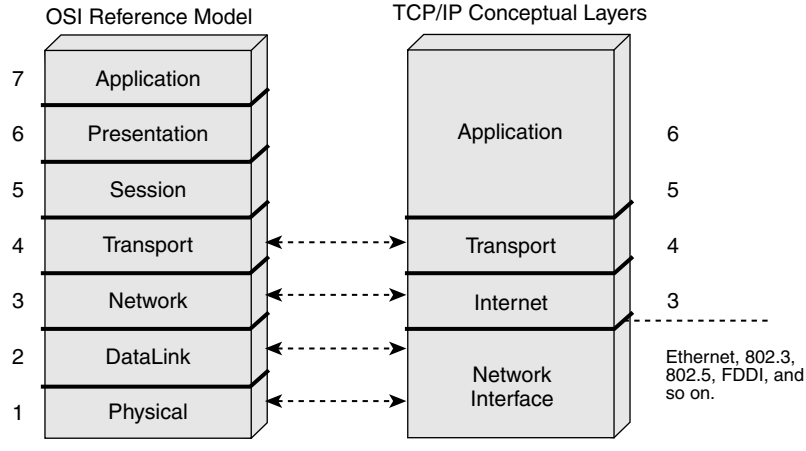
CHAPTER 1

Understanding IP Routing

This chapter presents an introduction to IP routing and provides insights to related concepts, such as IP addressing and various classifications of IP routing protocols. The chapter also provides a high-level overview of implementation and configuration concepts, such as route filtering and redistribution.

The Transmission Control Protocol/Internet Protocol (TCP/IP) suite of protocols is the underlying technology for information exchange on the Internet. TCP/IP uses a layering approach for computer communications similar to the Open System Interconnection (OSI) reference model, but with fewer than seven layers. Figure 1-1 shows the OSI reference model and the TCP/IP stack side by side. Related layers between the two stacks are indicated in the figure.

Figure 1-1 *OSI Reference Model and TCP/IP Stack*



IP operates at the Internet layer of the TCP/IP suite, which corresponds to the network layer of the OSI reference model. IP provides connectionless data-delivery services, which involve transmission of information from one part of a network to another in units of data known as packets or datagrams. The essence of the datagram delivery service model is that a permanent pre-established end-to-end path is not required for data transfer between two

points in a network. In a packet-based network, each router in the transmission path makes independent local decisions regarding the optimal forwarding path toward the destination for any transit packet. The decision-making is based on forwarding intelligence gathered either dynamically by means of a routing protocol or manually programmed static routes.

Addressing is an important aspect of the data-forwarding process. For any directed communication, there is a source and a destination. Addressing allows the target destination to be specified by the source and allows the destination node to also identify the source. Addressing is even more important in the datagram delivery mode of operation because, as in IP forwarding, the data path for any transmission is not nailed through the intermediate nodes between the source and destination.

As mentioned previously, within the IP datagram services infrastructure, information that is to be transmitted from one device to another first is broken down into packets. Each packet has an IP header, a transport layer (TCP or UDP) header, and a payload, which is a piece of the original information. Each IP packet is self-contained and independently is forwarded to the destination through the chain of intermediate devices that might be along the path of transmission.

The routers in the network depend on a routing protocol or static configuration to forward the datagrams in a stream to their intended destination. For any destination address, each node in the data path worries about only the outgoing interface or link along a locally determined optimal path to the destination (or as specified by a special forwarding policy). The IP forwarding process frequently is described as a *hop-by-hop destination-based forwarding mechanism*. This means that routers at each hop along the data path normally forward packets based on the destination address. However, modern routers also can use policy-based criteria, such as the source address in a packet to direct the forwarding.

At the destination, packets belonging to the same stream are reassembled into the original information. IP addressing is discussed in the next section, “IP Addressing Concepts.”

This process of forwarding a packet from one node to the other in a connectionless network based on the Layer 3 address (IP address, in this case) also is referred to as *routing*. Routers are specialized network devices with acquired routing intelligence.

So how do routers really decide where and how to forward packets traversing the internetwork? Well, this is done in a couple of ways. As alluded to previously, routers can be manually preprogrammed with predetermined path information known as static routes, or they can run applications that facilitate the learning and sharing of routing information automatically. Obtaining and propagating routing information by the latter method is referred to as *dynamic routing*.

IP Addressing Concepts

IP addressing is central to the operation of the IP protocol. The TCP/IP stack shown in Figure 1-1 features a network interface to the underlying physical and data-link layers, which allow the IP protocol to be media independent. Media independence is probably one of the critical advantages of the IP protocol that has promoted its wide acceptance and ubiquity. IP uses a native addressing scheme, in line with its media-independent architecture, that has no bearing on the underlying local-area network (LAN) or wide-area network (WAN) media interconnect IP devices. Therefore, IP successfully operates over heterogeneous network infrastructures consisting of several kinds of different media technology. This flexibility, together with a simple protocol stack, is the most critical instigator of its popularity.

IP addressing assigns addresses to individual network interfaces of a device (link-based approach) instead of using a single address for the whole device (host-based approach). The various interfaces of a device are connected to network links that are designated as subnetworks (or subnets) and are assigned subnet addresses. An interface's IP address is assigned from the subnet address space of the connecting link. The advantage of this link-based addressing approach is that it allows routers to summarize routing information by keeping track of only IP subnets in the routing tables instead of every host on the network. This is advantageous especially for broadcast links such as Ethernet that might have many devices connected at the same time. The Address Resolution Protocol (ARP) is used in IP networking for resolving the IP addresses of directly connected hosts to the corresponding data-link addresses.

Currently, two types of IP addresses exist: IP Version 4 addresses (IPv4) and IP Version 6 addresses (IPv6). IPv4 addressing, which was in place before IPv6 was adopted, uses 32 bits to represent each IP address. This 32-bit addressing scheme provides up to 2^{32} (4,294,967,295) unique host addresses, mathematically speaking. With the ever increasing size of the global Internet, the 32-bit IPv4 addressing scheme has turned out to be insufficient for the foreseeable future, prompting the introduction of the 128-bit IPv6 addressing scheme. This book covers practical troubleshooting of IP routing protocols deployed in IPv4 environments. Therefore, the ensuing text discusses only the IPv4 addressing structure and related concepts, most of which are applicable to IPv6. The following IPv4 addressing topics are covered in the subsequent sections:

- IPv4 address classes
- Private IPv4 address space
- IPv4 subnetting and variable-length subnet masking
- Classless interdomain routing

IPv4 Address Classes

As explained in the previous section, the 32-bit IPv4 addressing scheme allows a large number of host addresses to be defined. However, the link-based addressing scheme

adopted by IP requires network links to be associated with groups of addresses from which the connected hosts are assigned specific addresses. These address groups, described also as address prefixes, are referred to in classical IP terminology as *IP network numbers*.

Originally, IP network numbers were defined with rigid boundaries and grouped into address classes. The idea behind IP address classes was to enable efficient assignment of the IP address space by creating address groups that would support a varying number of hosts. Network links with fewer hosts then would be assigned an address from a class that supports an appropriate number of attached hosts. Another benefit of address classes was that they helped streamline the address-allocation process, making it more manageable.

Five address classes—A, B, C, D, and E—were defined and distinguished by the setting of the most significant bits of the most significant byte in the IP address. Each address class embraced a set of IPv4 address subnets, each of which supported a certain number of hosts. Table 1-1 shows the five IPv4 classes.

Table 1-1 *IP Address Classes and Representation*

Address Class	Bit Pattern of First Byte	First Byte Decimal Range	Host Assignment Range in Dotted Decimal
A	0xxxxxxx	1 to 127	1.0.0.1 to 126.255.255.254
B	10xxxxxx	128 to 191	128.0.0.1 to 191.255.255.255.254
C	110xxxxx	192 to 223	192.0.0.1 to 223.255.255.254
D	1110xxxx	224 to 239	224.0.0.1 to 239.255.255.254
E	11110xxx	240 to 255	240.0.0.1 to 255.255.255.255

As Table 1-1 shows, a specific bit pattern in the first byte of an IP address corresponds to a range of addresses and maps to a specific address class.

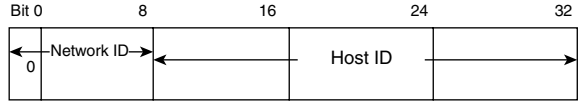
Of the five address classes, three—Class A, B, and C—were designated for unicast single source–to–single destination communication. Addresses in Class D were reserved for IP Multicast applications, which allows one-to-many communication. Class E addresses were reserved for experimental purposes.

To make the addresses in each of the unicast address classes (A, B, and C) support a specific maximum number of hosts, the 32-bit address field was delineated into network identifier (network ID) bits and host identifier bits (host ID) as follows:

- **Class A**—8-bit network ID, 24-bit host ID
- **Class B**—16-bit network ID, 16-bit host ID
- **Class C**—24-bit network ID, 8-bit host ID

Figure 1-2 shows the assignment of the 32 bits in a Class A address. The highest-order bit has a fixed value of 0, and the whole of the first byte is the network ID. The last 3 bytes are designated as host bits.

Figure 1-2 *Assignment of Class A Address Bits*



This notion of categorizing IP addresses into classes with rigid boundaries is also known as *classful addressing*. IP addresses use masks to delineate host bits from the network number bits. IP address structuring has evolved through various innovations, all geared toward making address allocation and actual assignment in real networks more efficient. You find out more about this in the section "Subnetting and Variable-Length Subnet Masks."

To make it easier for humans to work with IP addresses, these addresses are represented in a format known as *dotted-decimal notation*. In the dotted-decimal representation, the bits are grouped into octets and are separated by dots. Each octet of binary bits then is converted into the decimal equivalent. The last column of Table 1-1 shows the dotted-decimal notations for the range of addresses in each of the address classes.

Even though classful addressing was introduced to facilitate efficient use of the IPv4 address space, the rigid classful boundaries left a lot more to be desired. Because of its rigidity and inefficiency, classful addressing has been abandoned for the more efficient and flexible notion of *classless addressing*.

In classless addressing, any IP network number is interpreted as a prefix of a certain length. This interpretation provides more flexibility and results in a more efficient use of the IPv4 address space. A large classful block of addresses such as a Class A address can be split into multiple smaller blocks for allocation to multiple organizations instead of being allocated to a single organization under the classful notions. Conversely, classless addressing allows multiple Class C addresses to be aggregated and advertised as a single larger block instead of being treated as separate addresses. Aggregating addresses in this manner for the purposes of conserving resource in routers connected to the Internet is referred to as classless interdomain routing (CIDR), which is further discussed in a later section, "Classless Interdomain Routing (CIDR)."

IPv4 Private Address Space

Some address blocks in the unicast space were set aside and designated as private addresses. The private address space was intended for networks that are not connected to the public Internet. The following addresses are specific in RFC 1918 as part of the IPv4 private address space:

- 10.0.0.0 to 10.255.255.255
- 172.16.0.0 to 172.31.255.255
- 192.168.0.0 to 192.168.255.255

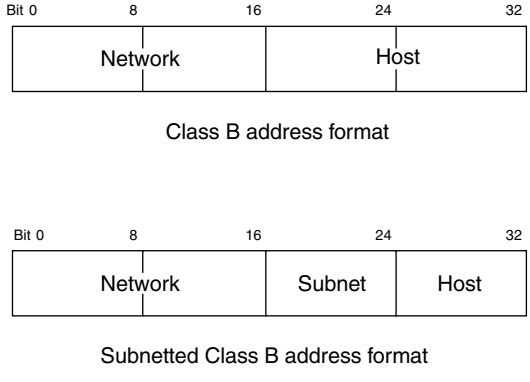
RFC 1700 provides general information on reserved or allocated parameters, including reserved addresses. Private internets that have deployed addresses from the private IPv4 space still can connect to the public Internet by using address Network Address Translation (NAT).

Subnetting and Variable-Length Subnet Masks

Before CIDR, each classful network number could be allocated for use in only a single organization. However, within an organization, it was possible to use subnetting to break up a classful address into multiple smaller address groups that could be applied to different segments of the same network domain.

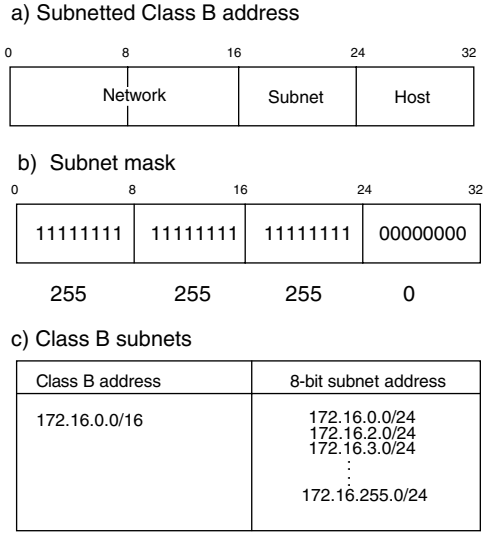
IP subnetting introduces another level of hierarchy into the structure of IP address classes by moving some of the host bits in a classful network number into the network ID field. The extended network ID is referred to as a subnetwork number or simply as an IP subnet. For example, one octet of the 2 octet host bits of a Class B address can be used to create 255 subnets, each with only an octet of host bits. This is illustrated in Figure 1-3.

Figure 1-3 *Class B Subnet Example*



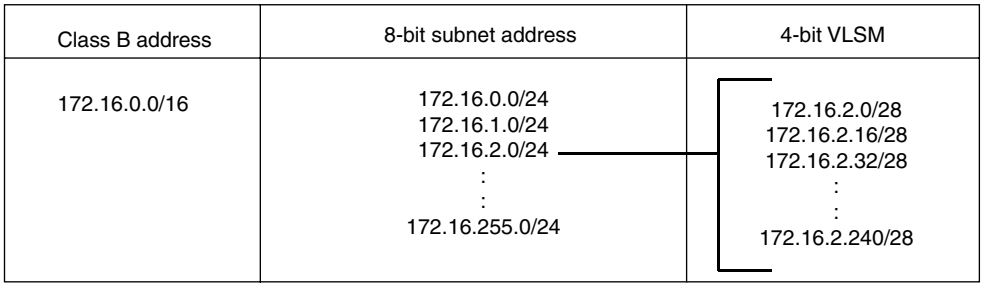
When an IP address is subnetted, the address mask is adjusted to reflect the new demarcation between the network and host bits. Figure 1-4 shows the new mask and the corresponding subnets that are created from a Class B address. A string of ones in the mask represent the network bits, and the zeros represent the host bits. A common way of representing an IP address is to indicate its prefix length, which is the number of 1 bits in the mask. This also represents the number of network bits in the address. For example, 172.16.1.0 255.255.255.0 can be represented as 172.16.1.0/24.

Figure 1-4 *Subnet Mask Example*



Even though classful addressing allows subnetting for more efficient assignment of addresses from a block, in a classful network environment only a consistent mask is allowed. VLSM extends the notion of subnetting to allow different masks to be applied to one network number, providing more flexibility in carving up an address into different block sizes for application to different segments in a network domain. This allows more efficient use of an allocated address block. For example, by using VLSM, the Class B address, 172.16.0.0/16, can be carved into smaller subnets with 24-bit subnet masks by using 8 host bits as subnet bits. You then can further subnet one of the first generation subnets—for example, 172.16.1.0/24—by using another 4 of the remaining host bits. This will result in much smaller blocks such as 172.16.1.0/28, 172.16.1.16/28, 172.16.1.32/28, and so on. VLSM can be used only in classless network environments in which the routing protocols and related routing software support classless addressing. Figure 1-5 illustrates subnetting with VLSMs.

Figure 1-5 *VLSM Example*

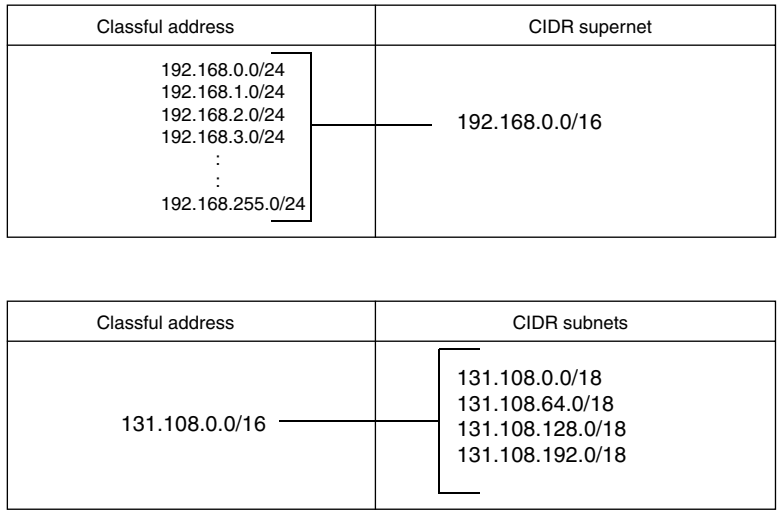


Classless Interdomain Routing

VLSM helps improve the efficiency of IP address usage for an assigned address block; however, it does not solve challenges with inefficient allocation of addresses to organizations. The imminent depletion of IP addresses as the result of inefficient use of classful blocks and the growing number of classful addresses in the global Internet routing tables as organizations were allocated multiples of a Class C address instead of a single Class B address led to the introduction of *classless interdomain routing (CIDR)*.

CIDR allows an IP network number to be any length, abandoning completely the fixed boundaries associated with classful concepts. The two benefits of CIDR are illustrated in the examples provided in Figure 1-6. By eliminating the notions of address classes, a block of addresses such as 192.168.0.0 to 192.168.255.0 consisting of an individual Class C address can be considered a uniform block that can be conveniently represented as 192.168.0.0/16. This essentially implies aggregation of 256 "old notion" Class C addresses into a single address block, referred to as a *CIDR block* or a *supernet*.

Figure 1-6 *Examples of CIDR Aggregation and Subnetting*



CIDR also allows network numbers to be flexibly subnetted and allocated to different organizations for interdomain routing exchange. For example, 131.108.0.0/16 can be divided into four subblocks (131.108.0.0/18, 131.108.64.0/18, 131.108.128.0/18, and 131.108.192.0/18) and allocated to four different organizations instead of one.

Static and Dynamic Routes

Static path information can be manually programmed into the router and simply force the router to utilize a particular interface or next-hop IP address for forwarding packets with matching destination addresses. Static routes potentially could match a broad range of network addresses. Yet another way to obtain routing information is to use distributed applications enabled on routers that allow automatic collection and sharing of routing information. These routing applications frequently are referred to as dynamic routing protocols because they are not only automated route-gathering tools; they also work in almost real time, tracking the state of connectivity in the network to provide routing information that is as current and as valid as possible.

Contrast this behavior with static routes, which are manual route entries and require manual intervention to reprogram the network routers in case of any path changes. Obviously, dynamic routing protocols provide more convenience to the network operator than static routes in managing routing information. The price for this convenience, however, is configuration and troubleshooting complexity. Operation of dynamic routing protocols also can be resource-intensive, requiring large amounts of memory and processing resources. Hence, working with dynamic routing protocols frequently requires advanced knowledge and sophisticated expertise for handling related network design, router configuration, tuning, and troubleshooting chores.

Even though static routing is less demanding on system resources and requires a lower level of technical skill to configure and troubleshoot, the sheer effort of manually entering routes for a sizeable network makes it a less attractive option. Obviously, static routing is not a good candidate for today's large enterprise and Internet service provider (ISP) IP-based networks. Another drawback to static routing is that it is less flexible for implementation of complicated routing policies. When it comes to routing policy implementation, there is no better substitute for the intelligence and flexibility provided by dynamic routing protocols, such as BGP, OSPF, and IS-IS. The next section further discusses dynamic routing protocols.

Dynamic Routing

The last section discusses the essence of IP routing and indicates that dynamic automatic routing is very necessary for large network deployments. This section discusses the characteristics and classification of various IP routing protocols. Although all routing protocols have a common goal of gathering routing information to support packet-forwarding decisions, they can be classified into two broad categories, unicast and multicast, based on the type of data traffic they are designed to provide forwarding information for.

The previous section indicates that IP provides an addressing scheme for identifying various locations or subnets in the network. The destination IP address in an IP packet

indicates the target address of the packet. The sender's address is stored in the source address field. An important concept to understand about IP addressing is IP subnetworks. IP subnetworks—or subnets, for short—are mentioned earlier in the section on IP addressing concepts. Physically, an IP subnet is a collection of interconnected network devices whose IP interface addresses share the same network ID and have a common mask.

The earlier section "IPv4 Address Classes" discusses unicast and multicast addresses. The unicast address space is used for addressing network devices, whereas addresses from the multicast space are used for specifying groups or users tuned in to receive information from the same multicast application.

For any IP unicast subnet, the last address, such as in 192.168.1.255/24, is known as the *broadcast address*. This address can be used to target all nodes on the subnet at the same time in what is referred to as a directed broadcast.

A unicast routing protocol is optimized for processing unicast network information and provides routing intelligence for forwarding IP packets to unicast destination addresses. Multicast forwarding is conceptually different and requires special routing applications to support forwarding of multicast packets.

Unicast Versus Multicast IP Routing

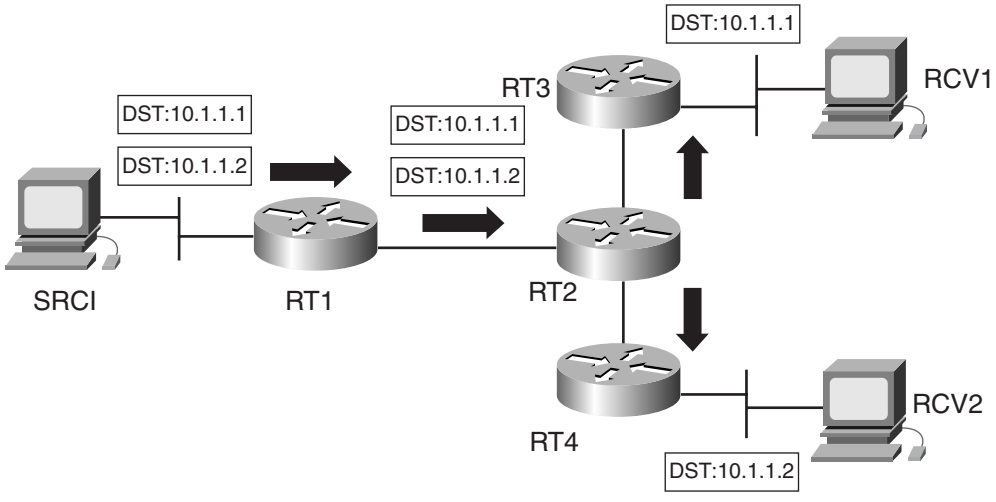
Two devices in an IP network normally communicate by sending unicast traffic to each other's IP address. An IP node might have many active interfaces, each of which needs to be configured with an IP address from the unicast space. The address on an interface uniquely defines the device on the subnet directly connected to that interface.

Cisco routers also support the concept of secondary logical subnets, many of which can be configured on a router's interface in addition to the primary address on that interface. Additionally, you can enable tunnel and loopback interfaces on a Cisco router, both of which provide it with unicast IP reachability. Packets with unicast addresses in their destination field are forwarded based on information in the IP routing table. The IP routing table on a Cisco router is displayed with the **show ip route** command.

If the address in the destination field of a packet is from the multicast address space (Class D), the packet is directed to a multicast group with potentially many receivers. Multicast forwarding uses special mechanisms that enable efficient utilization of network resources. If an application is designed for multidestination delivery, using unicast routing to forward packets of the application's data stream would require unnecessary replication at the source, resulting in a waste of network resources. This can be avoided by using multicast propagation, which replicates multicast packets only when necessary at branches in the network toward the location of receivers.

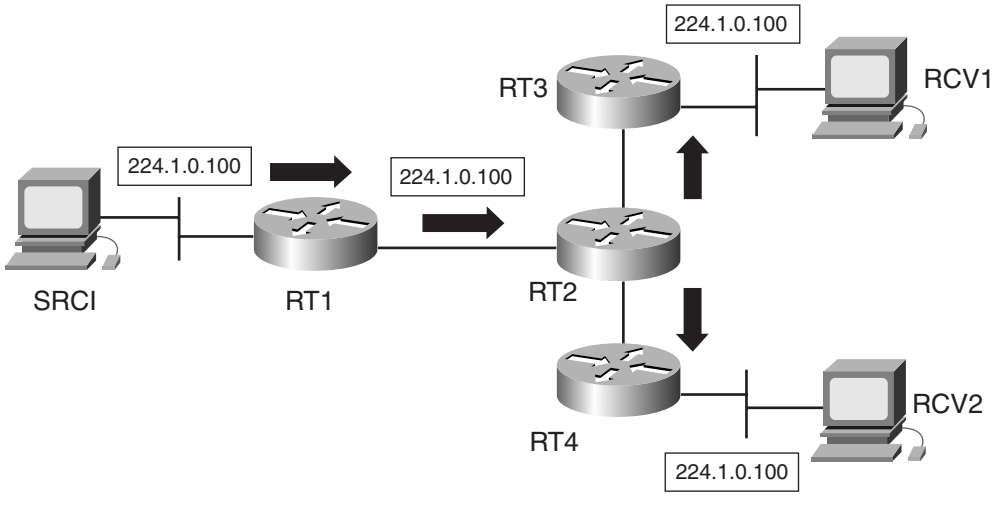
Figure 1-7 illustrates a situation in which a packet is forwarded from SRC1 to two separate destinations, RCV1 and RCV2, by unicast forwarding.

Figure 1-7 *Multidestination Unicast Forwarding*



In this case, SRC1 generates two identical streams of packets with destination addresses 10.1.1.1 and 10.1.1.2, respectively. Packets belonging to each stream are handled independently and are delivered through RT1 and RT2 to their respective destinations, consuming network resources (bandwidth and processing time) along the paths that they traverse. Contrast this scenario with that shown in Figure 1-8, where IP Multicast forwarding mechanisms are employed.

Figure 1-8 *Multicast Forwarding*



Multicast forwarding provides a more efficient way to deliver information by replicating packets only at fork points of the network where paths to receivers follow divergent directions. Therefore, as shown in the Figure 1-8, SRC1 originates only a single stream, and packets in this stream are forwarded through RT1 to RT2. They are then replicated at RT2 and fanned out to RCV1 and RCV2.

Multicast routing protocols are functionally different from unicast routing protocols, in that they build multicast forwarding state in the multicast-enabled routers by using a concept known as *reverse path forwarding (RPF)*. RPF is used to ensure that a multicast packet is received from the interface leading to the expected location of the multicast source, as dictated by the routing table in place.

RPF is discussed further in Chapter 12, "Understanding Protocol Independent Multicast (PIM)," which covers IP Multicast routing.

Table 1-2 shows a table of popular multicast and unicast routing protocols.

Table 1-2 *Unicast and Multicast Routing Protocols*

Unicast	Multicast
RIP (V1/V2)	DVMRP
IGRP	PIM
EIGRP	MOSPF
OSPF	MBGP
IS-IS	MSDP
BGP	

All the listed unicast routing protocols are supported in Cisco IOS Software; however, from the listed multicast routing protocols, only Protocol Independent Multicast (PIM) sparse mode/dense mode (SM/DM), Multicast Source Discovery Protocol (MSDP), and Multiprotocol BGP are supported.

Multicast routing environments also need an additional protocol called the Internet Gateway Multicast Protocol (IGMP). Multicast OSPR (MOSPF) is not supported at all, but IOS provides special capabilities for interoperability with the Distance Vector Multicast Routing Protocol (DVMRP).

As of this writing, multicast routing protocols are not widely deployed on the Internet. However, this situation obviously will change in the near future as more multicast-oriented applications, such as radio broadcasting, video streaming, remote training, videoconferencing, and gaming, become more popular on the Internet.

Classless Versus Classful IP Routing Protocols

The concepts of classless and classful IP routing protocols have roots in the manner in which IP addresses originally were defined.

Under classful addressing rules, a network number was assumed to retain its natural mask unless explicitly specified when subnetted into smaller blocks. However, earlier-generation routing protocols, such as the Routing Information Protocol (RIP), could handle only a single mask for any address throughout a network domain—the natural mask or a single consistent subnet mask. Routing protocols such as RIP that cannot handle more than one type of mask, as in the case of VLSMs, are referred to as *classful protocols* (see Table 1-3). The reason that classful protocols do not support VLSMs is that, by design, they do not advertise or carry the associated subnet mask with routes and, therefore, use simple intuitive mechanisms to determine the mask associated with a learned route.

The significant growth of the Internet to global dimensions called for more efficient use of the limited IPv4 address space. Available addresses in the IP address space therefore attained the status of a scarce commodity. The classless notions of VLSM and CIDR, discussed earlier, were invented to make address allocation and use more efficient. Routing protocols also were enhanced to support classless addressing environments. Routing protocols that are designed for operation in classless environments and that can handle VLSM address and CIDR are referred to as *classless routing protocols*.

Table 1-3 features a list of routing protocols categorized as classful and classless. RIP-1 and IGRP are grouped under classful protocols, whereas the more recently developed RIP-2, EIGRP, OSPF, IS-IS, and BGP fall in the classless category. The Exterior Gateway Protocol (EGP), the predecessor of the Border Gateway Protocol (BGP), which currently is considered obsolete, is also a classful protocol.

Table 1-3 *Classful and Classless IP Routing Protocols*

Classful	Classless
RIP-1	RIP-2
IGRP	EIGRP
EGP	OSPF
	Integrated IS-IS
	BGP

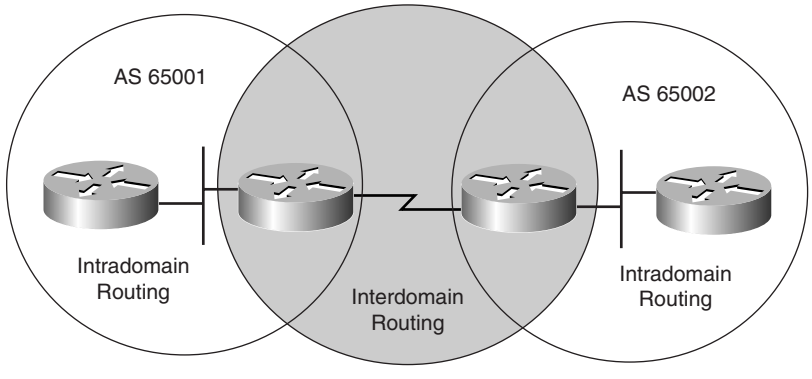
Interior Gateway Protocols Versus Exterior Gateway Protocols

Even though many unicast routing protocols were developed in the early days of the ARPANET (the predecessor to the Internet), Routing Information Protocol (RIP) emerged as the most popular. Many independent networks that were created at government research institutions and universities as a result of the remarkable success of

the ARPANET also adopted RIP for dynamic routing operations. The evolution of the ARPANET into the Internet required the numerous island networks to be interconnected using a more robust routing protocol. The Exterior Gateway Protocol (EGP) was selected for this purpose. EGP provided an efficient mechanism for routing among the various RIP domains. Therefore, RIP and EGP were optimized for distinct functions in the network based on their capabilities. RIP was used for intradomain routing, and EGP was used for interdomain routing. EGP later morphed into the Border Gateway Protocol (BGP), and other more robust protocols optimized for intradomain routing emerged in place of RIP. In particular, the Open Shortest Path First (OSPF) Protocol was developed in the Internet Engineering Task Force to provide capabilities that RIP lacked, such as more intelligent routing metrics, faster convergence, and operation in classless environments. So, here we are again with yet another classification of routing protocols: interior gateway routing protocols (for intradomain routing) and exterior gateway protocols (for interdomain routing).

Figure 1-9 shows two routing domains, AS 65001 and AS 65002, and an overlapping (shaded) region depicting the interconnection between border routers from each domain. In more current routing terminology, a routing domain also is referred to as an *autonomous system*. An autonomous system is an independent routing domain under the control of a single administrative authority.

Figure 1-9 *Intradomain and Interdomain Routing*



As noted before, an exterior gateway protocol provides the capability for sharing routing information between the two domains. Currently at version 4, BGP is the only IP interdomain protocol that is used for interconnecting the numerous autonomous systems in the global Internet. An interior gateway protocol provides routing intelligence within an autonomous system. Each of the autonomous systems in the Internet can run any suitable IGP. With the exception of EGP (the obsolete routing protocol) and BGP, all the other unicast protocols mentioned so far—IGRP, EIGRP, RIP, OSPF, and IS-IS—are IGPs (see Table 1-4).

Table 1-4 *IGP and EGP Classification*

Interior Gateway Protocols			Exterior Gateway Protocols
Distance Vector	**Advanced Distance Vector**	**Link-State**	**Path Vector**
RIP-1	EIGRP	OSPF	BGP
RIP-2		Integrated IS-IS	
IGRP			

The Interior Gateway Routing Protocol (IGRP) was invented by Cisco Systems to offer better metrics than the simple hop count supported by RIP. IGRP introduced a composite metric that consists of several parameters:

- Bandwidth
- Delay
- Reliability
- Load
- Maximum transmission unit (MTU)

Cisco evolved IGRP into the Enhanced Interior Gateway Routing Protocol (EIGRP). EIGRP provides faster convergence relative to IGRP by using backup routes, referred to as *feasible successor* routes, that are readily installed in the routing table when a preferred route is lost. Unlike IGRP, EIGRP supports VLSM.

OSPF and IS-IS are both popular IGPs used in very large IP networks. IS-IS originally was designed as a routing protocol for the Connectionless Network Protocol (CLNP) but later was adapted to route IP about the same time that the Open Shortest Path First (OSPF) protocol was being standardized in the Internet Engineering Task Force (IETF). OSPF and IS-IS are both link-state protocols, whereas RIP, IGRP, and EIGRP are distance vector protocols.

Also, OSPF and IS-IS are link-state protocols that use the shortest path first (SPF) algorithm (named after Dijkstra) for route computation, making them converge relatively fast in response to network changes.

Both protocols also support a two-level hierarchical routing architecture. OSPF and IS-IS are very similar protocols with almost identical capabilities. However, they have some architectural differences that are beyond the scope of this book.

An interesting point to note, however, is that OSPF was designed entirely for IP only, and OSPF packets are encapsulated in IP packets. In contrast, IS-IS was designed for CLNP and was adapted to support IP additionally. IS-IS packets are not encapsulated in IP packets but rather directly by the data link protocol.

The next section of this chapter looks at yet another routing protocol classification: distance vector and link-state protocols.

Distance Vector Versus Link-State Protocols

This section takes a look at routing protocol from a different perspective. In the previous sections, we considered general classification such as classful versus classless and also IGP versus EGP. This section discusses classification based on design and operation. The second row in Table 1-4 places the protocols discussed so far into four different categories, two of which stand out—distance vector and link-state. These two broad categories actually apply to IGP as shown in the table.

EIGRP is essentially a distance vector protocol just like IGRP, except that it is rightfully considered in its own class as an advanced distance vector protocol because it has more modern characteristics, such as support of classless routing and fast convergence. BGP is also in its own category, path vector protocol because, as an interdomain routing protocol, it uses the AS path attribute, which is made up of the list of autonomous systems that a route has traversed as a key measure for route comparison and selection.

Versions 1 and 2 of RIP (RIP-1 and RIP-2) and IGRP are classified as distance vector protocols because they use route-computation algorithms based on the Bellman-Ford algorithm. The Bellman-Ford algorithm is used in graph theory for calculating the shortest distance between two vertices in a directed graph. A directed graph is a collection of points, interconnected with directional links, such as the nodes and links in an internetwork. Routers running distance vector routing protocols use the Bellman-Ford algorithm for determining the shortest paths to all known locations in the network.

OSPF and Integrated IS-IS are both link-state protocols and use the shortest path first algorithm (Dijkstra) for route computation. Just like the Bellman-Ford algorithm, the Dijkstra algorithm provides an alternate method for computing the shortest distance between two points in a directed graph.

EIGRP uses a Cisco Systems–patented algorithm known as the Diffusing Update Algorithm (DUAL) to optimize route calculation, breaking away from its predecessor, IGRP, which is based on the Bellman-Ford algorithm.

The type of algorithm used by a protocol for route computation goes a long way toward affecting the efficiency of the protocol and how fast it converges. The following sections examine the concepts and operational principles behind distance vector protocols and link-state protocols.

Distance Vector Routing Concepts

This section reviews key concepts that underlie the operation of distance vector routing protocols, such as metrics, count to infinity, split horizon, holddowns, and triggered

updates. These concepts are evaluated in terms of general routing functionality, such as stability and speed of convergence and loop avoidance.

Distance Vector Metrics

In the Bellman-Ford algorithm, each router advertises the best paths to all known destinations, from its perspective, to all neighbors. The links between routers are assigned a measure known as cost or metric. The metric can be determined from characteristics of the links, such as hop count, bandwidth, delay, reliability, monetary value, and so on. The hop count associated with a link between two directly connected nodes is usually 1, even though arbitrary values can be administratively assigned. The metric associated with a specific path to a known destination from any router is the sum of all the metrics of links along that path. Usually, the path with the lowest metric is the best. A router might have many neighbors and, therefore, might receive multiple paths for the same destination. It then computes the metric associated with each of these paths and selects the best path by a criterion such as the lowest metric.

RIP uses hop count for metric, with the maximum possible number of hops to any reachable destinations being 15. A metric of 16 hops or more is considered to be infinity. Hence, a hop count of 15 defines the maximum width of reachability in a RIP network. This imposes a limit on the size of RIP-based networks, which also implies that RIP is suitable for only small, flat networks. Hop count actually pertains to the node count from a specific source to a destination and has no consideration for actual network characteristics, such as bandwidth, delay, or monetary costs.

IGRP, which is also a distance vector protocol, uses a metric system that takes into consideration relevant characteristics of the network, such as bandwidth, associated maximum transmission unit, reliability of links, and also path delay. The metric assigned to each link in the outgoing direction is calculated from a formula that takes into consideration all these characteristics. This sort of multifaceted metric is called a composite metric.

The Bellman-Ford algorithm uses a vector (distance vector), consisting of cost (metric) and next-hop information for each known route to determine best paths in the network from any standpoint. An iterative procedure calculates the cost of all paths for any received route and selects the vector with the best cost for each route. Hence, routing protocols that are based on the Bellman-Ford algorithm commonly are referred to as *distance vector protocols* (see Table 1-4).

Routing Convergence

When there is a topology change, a router might invalidate some of the previously known best paths. The router then uses new or existing information to determine an alternate best path for each affected destination. Recalculating routes to rediscover alternate routes as a result of network topology changes is referred to as *routing convergence*. Routing convergence may be

triggered by events such as router failures, link failures, or even administrative metric adjustments.

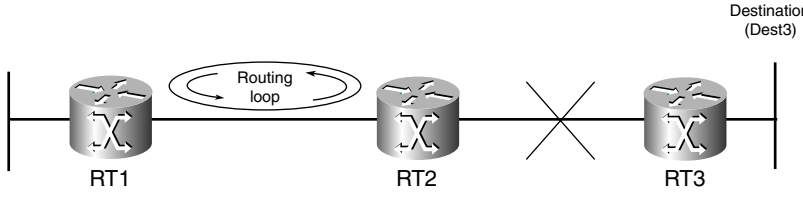
Distance vector protocols such as RIP and IGRP are relatively simple compared to their link-state counterparts. However, this simplicity comes with a price. Because each router bases its best-path determination on the best paths advertised by neighbors, such protocols are very prone to routing loops. A routing loop occurs when two nodes point to each other as the next hop along the path to the same destination. The most obvious effect of routing loops is that they prolong the time it takes for a router to determine a route is no longer available or to select an alternate path. Routing loops adversely impact convergence times. Therefore, it is desirable that unusable routes be removed from the network as soon as possible. The following sections discuss various methods employed by distance vector protocols to prevent or limit the effect of routing loops and improve convergence. The following is discussed:

- Counting to infinity
- Using holddown
- Using split horizon and poison reverse
- Using triggered updates

Loop Avoidance

Routers running distance vector protocols determine best paths for routes relative to neighbors that have advertised those routes to them. The mechanics of operation of distance vector protocols, specifically the way routes are advertised by distance vector protocols, makes such environments very susceptible to routing loops—for example, when a router running a distance vector protocol broadcasts routing updates over all interfaces activated for the protocol. When a router broadcasts all known routes in this manner, it may advertise a route back to the source it was heard from. Consequently, when there is a failure, it is possible for two neighboring nodes to think that the other is the next hop along the best path to a specific destination. This situation, which results in a routing loop, is elaborated in Figure 1-10.

Figure 1-10 *Routing Loops in Distance Vector Environments*



In Figure 1-10, RT1, RT2, RT3 are connected serially, and hop count is used as the measure for metric. A route associated with the destination link (Dest3) is advertised by RT3 to RT2, with a hop count of 1. RT2 assigns Dest3 a hop count of 2 and then advertises it to RT1. RT1 stores Dest3 with a hop count of 3 and with RT2 as the next hop. RT1 then might advertise Dest3 back to RT2. This route is not used by RT2 because it has a worse metric (four hops) than the original that came from RT3 (two hops). However, if the connection between RT2 and RT3 is broken, RT2 will remove the original route and install an alternate route to Dest3 with a metric of 4 and RT1 as the next hop. Meanwhile, RT1 has the same route pointing back to RT2 as the next hop. Thus, a loop situation is created and any packets from RT1 or RT2 to Dest3 will be caught up in a "ping pong" between the two routers for some time until their Time To Live (TTL) counters in the packets expire. Routing loops disrupt routing, and it is desirable to curtail them as quickly as they appear. To limit the effect of routing loops, distance vector protocols use a method known as *counting to infinity*. This principle is elaborated in the next section.

Counting to Infinity

To prevent routing loops of indefinite duration, distance vector protocols enforce limits on route metrics that allows routers to declare routes as unreachable after the associated metrics reach a certain value. In the loop situation described in Figure 1-10, RT1 and RT4 might advertise Dest3 to each other, each time increasing the associated hop count received from the other by 1 and before readvertising the route. Consequently, the metric associated with Dest3 will continue to increase. Counting to infinity places an upper bound on the metric beyond which it is considered infinity and the route is declared unusable. For RIP, this upper bound is 15.

Holddown

Holddown is used to dampen a route's response action to finding an alternate route when a primary route is no longer usable. When a router determines that a route is no longer available, it places the route in *holddown* state for a duration called the holddown time, during which it doesn't select an alternate route, even if available. The route in holddown state is advertised with a metric or value of infinity in an attempt to purge it from the network. Purging unusable routes helps reduce the incidence of routing loops.

To illustrate this using Figure 1-10, RT2 places Dest3 in holddown when it invalidates routes heard from RT3 because of the failure of the connection between them. With Dest3 in holddown state, RT2 does not use the alternate route from RT1; instead, it advertises Dest3 to RT1 again with a metric. This allows RT1 to withdraw Dest3 from its tables. By the expiration of the holddown time, both RT1 and RT2 are expected to have removed Dest3 from their routing tables, thus avoiding a potential routing loop.

Another benefit to using holddowns is that it prevents unnecessary reactions to equipment-related glitches that cause the link to flap. The downside is that it contributes significantly to the higher convergence times associated with distance vector routing protocols.

Split Horizon and Poison Reverse

Routing loops are primarily the result of routes being leaked back to their sources. For example, in Figure 1-10, the loop between RT1 and RT2 is caused by feedback of Dest3 back to RT2 by RT1, misleading RT2 to think that RT1 is the next hop on an alternate path to Dest3.

Split horizon prevents a router from advertising a route back out the interface through which it was received. With split horizon in effect, RT1 cannot advertise Dest3 back to RT3 over the link between them (see Figure 1-10).

Poison reverse is similar in principle to split horizon, except that it allows routes to be advertised back out the interfaces on which they were received as unreachable (metric of infinity assigned). That is, routes are "poisoned" in the reverse direction. Referring to Figure 1-10, with poison reverse enabled, RT1 advertises Dest3 back to RT2, but with a metric value of infinity (16 hops, in the case of RIP).

The approach adopted by poison reverse can result in undue waste of bandwidth if many poisoned routers must be advertised back out. However, this approach speeds up route convergence by eliminating the need for holddowns. In this case, the alternate route would have an obvious infinite metric when fed back to the source, hence simplifying the search for an alternative path, when the primary route is lost.

Periodic and Triggered Updates

Routers running distance vector routing protocols, such as RIP and IGRP, advertise all the contents of their routing tables at regular intervals. Periodic broadcasts of large routing tables are a major concern in large networks. For example, RIP broadcasts all known routes out of every active interface every 30 seconds, by default, even if there are no changes. IGRP uses a default update interval of 90 seconds.

If updates are advertised only periodically, changes in the network might not be communicated fast enough, impacting convergence times. Also, the holddown time typically is tied to the update interval. So a larger interval might result in less bandwidth consumption by routing updates yet might introduce higher convergence times.

Triggered (or flash) updates remove delays in convergence caused by periodic updates by sending updates immediately following a network change instead of waiting for the periodic update timer. Flash updates trickle through the network from one node to the other, resulting in an overall time gain in network-wide convergence, even if not very significant. Complicity

between periodically scheduled updates and triggered changes can result in unpredictable behavior.

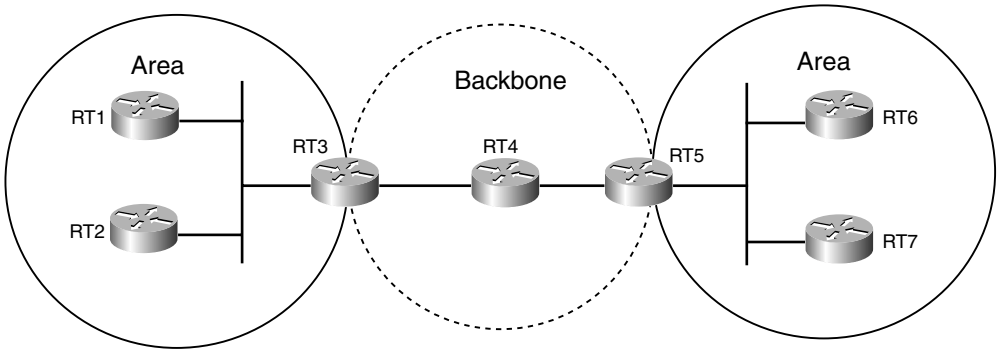
Link-State Protocols

Link-state protocols are relatively more modern and, therefore, incorporate capabilities into their design to overcome some of the shortcomings of distance vector protocols discussed previously. Hence, they are more sophisticated and require more memory and processing resources to operate effectively. By virtue of characteristics such as faster convergence, incremental updates, and a hierarchical architecture, link-state protocols are more suitable for deployment in large internetworks. Two popular link-state protocols used in IP networks are OSPF and IS-IS.

Unlike distance vector protocols, which share best-known routing information, link-state protocols allow routers to exchange topology (link-state) information that allows them to draw out the layout of the internetwork's topology. Routers in a link-state network converge relatively faster than their distance vector counterparts by responding immediately to changes in the topology, without the need for loop avoiding or limiting holddowns and counting to infinity. For example, RIP and IGRP typically feature convergence times in minutes, whereas OSPF and IS-IS converge in the order of seconds for comparable network changes.

Link-state protocols support hierarchy for scaling purposes by carving out a network into areas (see Figure 1-11). Routing within areas fall in the first level of the routing hierarchy. The areas are interconnected over a backbone area, and routing within the backbone constitutes the second level of the hierarchy.

Figure 1-11 *Areas and Hierarchy in Link-State Protocols*



Routers in the same area or the backbone share link-state information that is assembled into a link-state database. The topology of the area or the backbone is discerned by running the shortest path first algorithm over the respective databases. This procedure

also generates the best routes that are used in the IP routing and forwarding tables. Chapter 8, "Understanding Open Shortest Path First (OSPF)," and Chapter 10, "Understanding Intermediate System-to-Intermediate System (IS-IS)," describe the operation of the link-state routing protocols and their respective protocols in more detail.

Metrics in Link-State Protocols

Both OSPF and IS-IS use metrics, which are measures of link bandwidth. OSPF goes a step further, to provide autoconversion of the bandwidth on interface to a link cost. IS-IS metrics are 10, by default, on all interfaces. In both cases, the metric or cost associated with a link can be manually configured. The metric associated with a route is the sum of all the metrics on the outgoing links to the associated destination.

Chapters 8 and 10 provide more information on metrics in OSPF and IS-IS, respectively.

Routing Protocol Administrative Distance

The previous sections in this chapter provide a high-level overview of IP routing protocols from the perspectives of design, architecture, and operation. The section discusses briefly generic implementation-related issues that impact operation of these protocols on Cisco routers. Details of operation and configuration of each protocol are covered in the protocol-specific chapters.

Cisco IOS Software provides common command resources for configuring and enabling the capabilities of IP routing protocols. Commands such as **distance**, **distribute-list**, **redistribute**, **route-map**, **policy-map**, **access-list**, **prefix-list**, **offset-list**, and so forth frequently are referred to as *protocol-independent commands* because they can be used in diverse ways to enable many features in Cisco IOS Software, including routing protocol capabilities. In their application to routing protocols, protocol-independent commands are used for filtering routes, enabling redistribution, configuring default routes, and implementing various routing policies. You can find more detail on these commands online at www.cisco.com; however, this section discusses the **distance** command and the feature that it supports—*administrative distance.*

All the IP routing protocols discussed so far can operate concurrently and yet independently on Cisco routers if enabled together. Usually, only one IGP (OSPF or IS-IS) is required to run alongside BGP in an IP network. However, depending on the situation and the history of a network, more than one IGP might be operation to support routing requirements.

Administrative distance is a Cisco-specific method of distinguishing between routes obtained from different routing sources in the same network. It provides a simple mechanism to differentiate believability of routing information sources. Cisco IOS Software assigns numeric values to routing sources that allow routes from one routing source to be preferred over similar routes from another source. Sources with lower administrative dis-

tance values are preferred. When multiple protocols supply the same route, only the route from the source with the lower administrative distance will make it into the routing table. Table 1-5 lists the default administrative distances of IP routing sources. The **distance** command can be used to modify any of the defaults.

Table 1-5 *Administrative Distances of IP Routing Protocols*

Route Source	Administrative Distance
Connected interface	0
Static route out an interface	1
Static route to a next hop	1
EIGRP summary route	5
External BGP	20
Internal EIGRP	90
IGRP	100
OSPF	110
IS-IS	115
RIP-1/RIP-2	120
EGP	140
External EIGRP	170
Internal BGP	200
Unknown	255

Fast Forwarding in Routers

Even though this book is about routing protocols and how to troubleshoot routing-related problems, we would like to briefly mention in this introductory chapter that the high-speed forwarding requirements in today's networks have led to ingenious ways of packet processing on routers that extend beyond basic decision-making based on the IP routing table. The routing table remains critical for routing guidance, but instead of using the contents of the routing table directly, routers transform the routing information in the routing table for storage in data structures, optimized for high-speed packet forwarding. Cisco provides various high-speed forwarding mechanisms, such as fast switching, optimum switching, and Cisco Express Forwarding (CEF).

Frequently, troubleshooting routing problems requires investigation into the fast-forwarding tables, such as the CEF Forwarding Information Base (FIB) and the Adjacency Database. Detailed discussions of these fast-forwarding mechanisms are outside the scope of this book. More information on this subject matter is available at the Cisco site, www.cisco.com.

Summary

This introductory chapter reviews the concepts underlying IP routing and explains why routing is relevant for information transfer in a connectionless networking environment. You learned that protocols such as IP, which provide connectionless delivery of information, allow data to be transmitted in chunks of information, known as datagrams. IP datagrams also are referred to as packets. Packets consist of a payload and a header. The headers in IP packets contain target addresses that allow them to be independently routed over optimal paths in the network toward their destinations. IP is a network layer protocol; routers, which process and forward packets, run routing protocols that automate the gathering of routing information in internetworks.

Classful and classless notions of IP addressing are covered, leading to a discussion on VLSMs and CIDR. The relevance of CIDR and VLSMs as vehicles for efficient address allocation and use is covered as well.

The subsequent text of the chapter discusses various classifications of dynamic routing protocols, categorizing them into unicast versus multicast, classless versus classful, IGP versus EGP, and, finally, distance vector versus link-state. Key characteristics of distance vector and link-state protocols are discussed and compared.

Brief coverage of Cisco IOS Software protocol-independent commands led to the discussion of administrative distances associated with routing protocols. Administrative distance is defined as a mechanism for distinguishing between routing protocol sources and associating an IOS default trust factor with various routing protocols.

The final section briefly touches on how the routing information gathered by routing protocols actually is used in forwarding. It is pointed out that Cisco routers convert the information in a routing table into optimized data structures for high-speed packet forwarding.

Review Questions

1. What is connectionless data networking?
2. Why is routing needed in a connectionless networking environment? List two means by which routers obtain information for routing packets toward their destinations.
3. What is the difference between functionalities of Interior Gateway Protocols (IGPs) versus exterior gateway protocols (EGPs)?
4. List the two main groups of IP routing protocols based on the method of operation and routing algorithm. Also, list two examples of each type.
5. Briefly describe the operation of link-state routing protocols.

6 What is the key difference between classless and classful routing protocols? Give an example of each.

7 What is the use of routing protocol administrative distances on Cisco routers?

8 What are the values of administrative distance of IS-IS and OSPF, respectively?

9 If a router is running both OSPF and IS-IS protocols and has the same route from each of them, which protocol's information will be used in the IP routing table?

References

Bates, T., R. Chandra, Y. Rekhter, and D. Katz. "Multi-Protocol Extensions for BGP4." RFC 2858, 2000.

Bennett, Geoff. *Designing TCP/IP Internetworks*. New York, NY: John Wiley & Sons; 1997.

Callon, R. "Use of OSI IS-IS for Routing in TCP/IP and Dual Environments." RFC 1195. IETF 1990.

Fuller, V., T. Li, J. Yu, and K. Varadhan. "Classless Interdomain Routing (CIDR): An Address Assignment and Aggregation Strategy." RFC 1519. IETF 1992.

Hall, Eric A. *Internet Core Protocols: The Definitive Guide.* Sebastopol, CA: O'Reilly and Associates, 2000.

Hedrick, C. "Routing Information Protocol." STD 34, RFC 1058, 1988.

http://www.6bone.net/

http://www.cisco.com/warp/customer/701/3.html. "Understanding IP Addresses."

http://www.cisco.com/warp/public/103/index.shtml

Huitema, Christian. *Routing in the Internet, 2nd Edition.* Upper Saddle River, NJ: Prentice Hall, 2000.

ISO 10589. "Intermediate System-to-Intermediate System Intradomain Routing Information Exchange Protocol for Use in Conjunction with the Protocol for Providing the Connectionless-mode Network Service." (ISO 8473.)

Li, Rekhter. "Border Gateway Protocol Version 4 (BGP 4)." RFC 1771, 1995.

Maufer, Thomas. *Deploying IP Multicast in the Internet*. Upper Saddle River, NJ: Prentice Hall, 1997.

Miller, Philip. *TCP/IP Explained*. Woburn, MA: Digital Press, 1997.

Naugle, Mathew. *Network Protocol Handbook*. New York, NY: McGraw Hill, 1994.

Perlman, Radia. *Interconnections 2nd Edition.* Reading, MA: Addison Wesley, 1999.

Reynolds, J. and Postel, J. "Assigned Numbers." RFC 1700. IETF 1994.

Rekhter, Y., B. Moskowitz, D. Karrenberg, G. J. de Groot, and E. Lear. "Address Allocation for Private Internets." RFC 1918. IETF 1996.

This chapter covers the following key topics about Routing Information Protocol (RIP):

- Metric
- Timers
- Split horizon
- Split horizon with poison reverse
- RIP-1 packet format
- RIP behavior
- Why RIP doesn't support discontiguous networks
- Why RIP doesn't support variable-length subnet masking (VLSM)
- Default routes and RIP
- Protocol extension to RIP
- Compatibility issues

CHAPTER 2

Understanding Routing Information Protocol (RIP)

RIP is a distance vector protocol that uses hop count as its metric. This protocol is very simple and was intended for small networks. RIP is similar to gated, which was distributed by the FreeBSD version of UNIX. Before the RFC for RIP Version 1 (RIP-1) was written, several versions of RIP were floating around.

NOTE Hop count refers to the number of routers being traversed. For example, a hop count of 2 means that the destination is two routers away.

RIP is a classful protocol, which means that it doesn't carry subnet mask information in its routing update. Because it doesn't carry any subnet mask information, it is incapable of supporting variable-length subnet masking (VLSM) and discontiguous networks. RIP enables devices to exchange information about networks that they are directly connected to, as well as any other networks that they have learned from other RIP devices.

RIP sends its routing information every 30 seconds, which is the default update timer. This timer is configurable. The hold-down timer determines how long a router should wait before flushing the information from the routing table.

RFC 1058 was written to provide a standard for RIP, which uses the Bellman-Ford algorithm to compute its metric.

Metric

The RIP metric is based on hop count and can be between 1 and 15. The metric 16 is used for infinity, which means that if the route is unreachable, a metric of 16 is displayed. The question is, why was the metric chosen as 16? Why not 17 or 18? The metric filed in RIP-1 packet format clearly shows that it is 32 bits long. This means that, theoretically, RIP can support 2^{32} hops. Although this is a large number, the metric of 15 was chosen to avoid a *count to infinity* problem. (This is also referred to as a *routing loop.*) In a large network with a few hundred routers, a routing loop results in a long time for convergence if the *metric for infinity* has a large value. The number 16 was chosen to get a shorter convergence time.

The 15-hop limit was chosen also because RIP was intentionally designed for small networks. It was not intended for the large networks that potentially can have more than 15 hops.

Timers

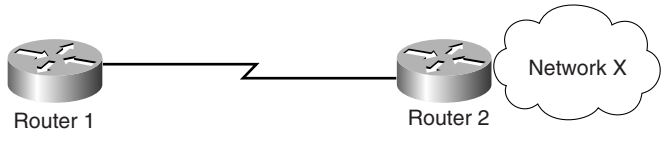
Like any distance vector protocol, RIP periodically sends an update every 30 seconds. This update consists of a broadcast of the entire routing table. The update timer controls this 30-second period. RIP uses the following timers:

- **Update**—The time between each update interval. This value is set to 30 seconds, by default, and is configurable.
- **Invalid**—The time after which a suspect route becomes invalid. This is set to 180 seconds, by default.
- **Hold-down**—The time used to suppress the possibility of defective routes being installed in the routing table. The default time is 180 seconds.
- **Flush**—The time after which a route is removed from the routing table. This is set to 240 seconds, by default.

Split Horizon

Split horizon is a technique used to avoid routing loops. With split horizon, when a route is learned on an interface, that route is not advertised back out on the same interface. For example, in Figure 2-1, Router 1 receives an update about Network X with a metric of 1 from the neighboring Router 2. Router 1 will not advertise Network X back to Router 2 if split horizon is enabled. If split horizon is disabled, however, Router 1 will advertise Network X with a metric of 2 to Router 2. If Network X fails, Router 2 will think that Router 1 has a better way to get to X, so it will send the packet destined to Network X toward Router 1, creating a black hole.

Figure 2-1 *An Example of Split Horizon*



Split Horizon with Poison Reverse

Another technique used to avoid routing loops is *split horizon with poison reverse*. With this technique, routes learned on an interface are advertised back on the same interface, but they are poisoned, which means that they have a metric of 16 (unreachable). In Figure 2-1, Router 1

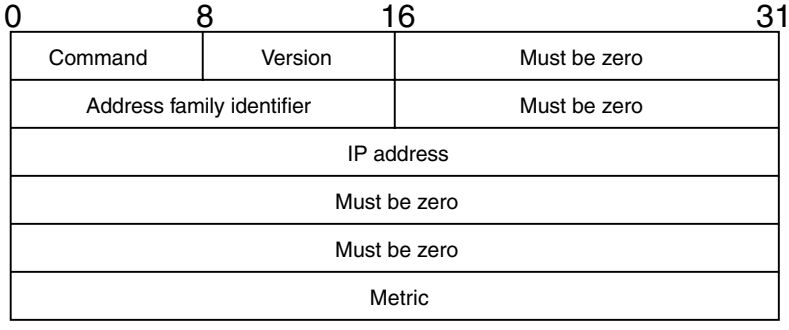
receives an update about Network X with a metric of 1 from neighboring Router 2. In the case of split horizon with poison reverse, Router 1 will advertise Network X back to Router 2, but with a metric of 16, which indicates infinity.

Split horizon with poison reverse is used only when a link failure occurs. It also can be used in a normal situation, but it is discouraged because it can potentially increase the size of the routing table.

RIP-1 Packet Format

The maximum datagram size in RIP is 512 octets. The first byte is used for commands such as **rip update request** and **rip update response**. The next byte is used for the Version field, which is set to 1 for RIP-1. The next 2 bytes must be 0. The 2-byte field after this is used for the address family identifier; the next 14 bytes are allocated for the network address, as shown in Figure 2-2. In the case of IP, only 4 bytes of those 14 are used for the IP address. The remaining 10 bytes are unused in RIP-1, although they are used in the RIP Version 2 (RIP-2) packet format. The next 4 bytes are used for the RIP metric, which can be up to 16. The portion from the address family identifier up to the Metric field can be repeated 25 times, to yield the maximum RIP packet size of 512 bytes.

Figure 2-2 *RIP-1 Packet Format*



RIP Behavior

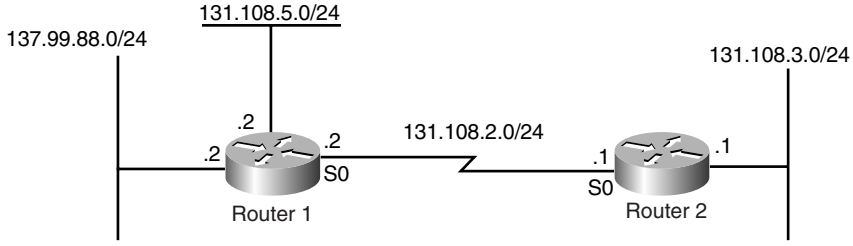
RIP follows certain rules when it sends and receives updates. This section covers the rules for sending and receiving updates.

RIP Rules for Sending Updates

When RIP sends an update, it performs several checks. In Figure 2-3, two routers are running RIP together. Router 1 is connected to two majornets, 131.108.0.0/16 and 137.99.0.0/16.

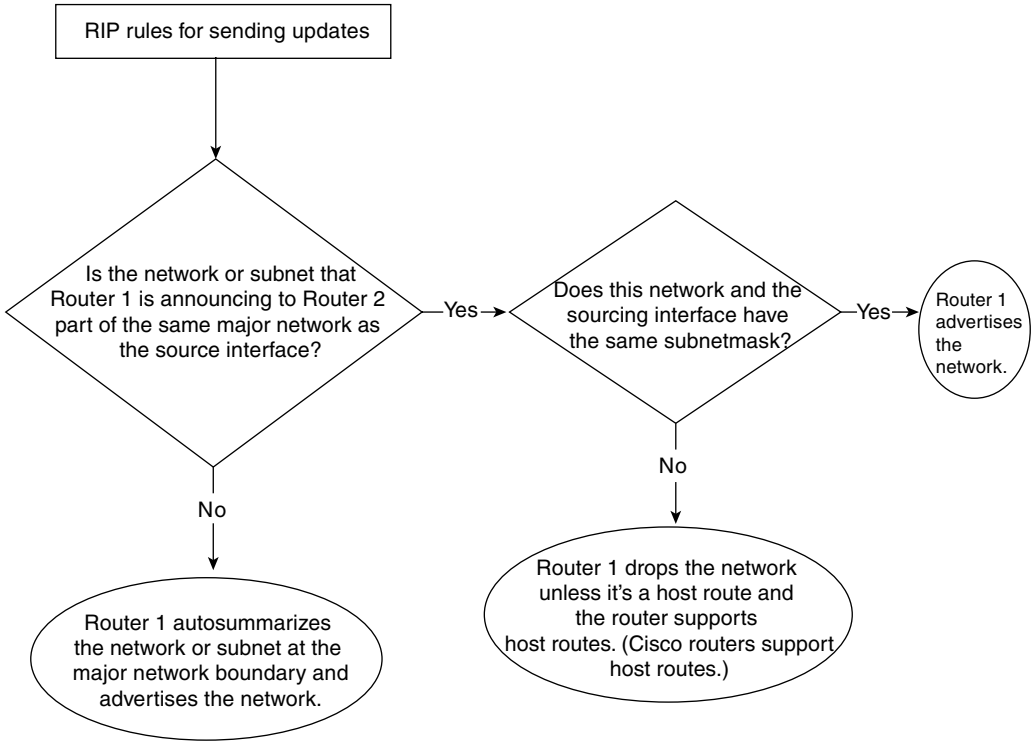
The majornet 131.108.0.0 is further divided into two subnets: 131.108.5.0/24 and 131.108.2.0/24, which is actually connected to Router 2.

Figure 2-3 *Example of RIP Behavior*



Before Router 1 sends a RIP update to Router 2, it performs the check as shown in Figure 2-4.

Figure 2-4 *Flowchart That Explains RIP Rules When Sending Updates*



When RIP sends the update, it checks to see whether the advertised network or subnet is on the same major network as the interface that is sourcing the RIP packet. If the advertised network or subnet is on a different major network from the interface sourcing the RIP packet, the network is autosummarized. In other words, RIP sends only the majornet information in its routing update. For example, in Figure 2-3, when Router 1 sends the RIP update to Router 2, it autosummarizes the subnet 137.99.88.0 into 137.99.0.0. If the advertised network or subnet is on the same major network as the router's interface sourcing the RIP packet, RIP determines whether the advertised subnet has the same mask as the interface that is sourcing the RIP update. If it has the same mask, RIP advertises that network; otherwise, RIP drops that network.

RIP Rules for Receiving Updates

When the receiving side gets an update from RIP, the update can contain either a subnet number, a host address, a network number, or all 0s (indicating the default route):

- Subnet number (such as 131.108.1.0)
- Host address (such as 131.108.1.1)
- Network number (such as 131.108.0.0)
- Default route (such as 0.0.0.0)

Figure 2-5 illustrates the checks performed by RIP on the receiving side.

When RIP receives the update, it determines whether the subnet received in the update belongs to the same major network as the receiving interface. If so, Router 2 applies the mask of the receiving interface. If the host bits are set in the host portion of the RIP update, the receiving router applies the host mask.

If that subnet belongs to a different major network, RIP checks whether any subnets of this major network already exist in the routing table and determines whether they are known from interfaces other than the one that received the update. Note that the network in this update should be a major network. If the answer is "yes," Router 2 ignores the update. If the answer is "no," Router 2 applies a classful mask.

If the update came across an unnumbered link, it might contain subnet information (bits in the subnet portion of the network address are set). Router 2 then applies a host mask. If the update carries subnet broadcast—for example, 131.108.5.127/25 or Class D or E—the RIP update must be ignored.

Example of Sending Updates

This section shows an example explaining RIP behavior when it sends an update. In Figure 2-6, two routers are running RIP. The link between Router 1 and Router 2 is in 131.108.0.0. The Ethernet interface on Router 1 is in 131.108.0.0 as well. Router 1 is also connected to another major network, which is 137.99.0.0.

Figure 2-5 *Flowchart That Explains RIP Rules When Receiving Updates*

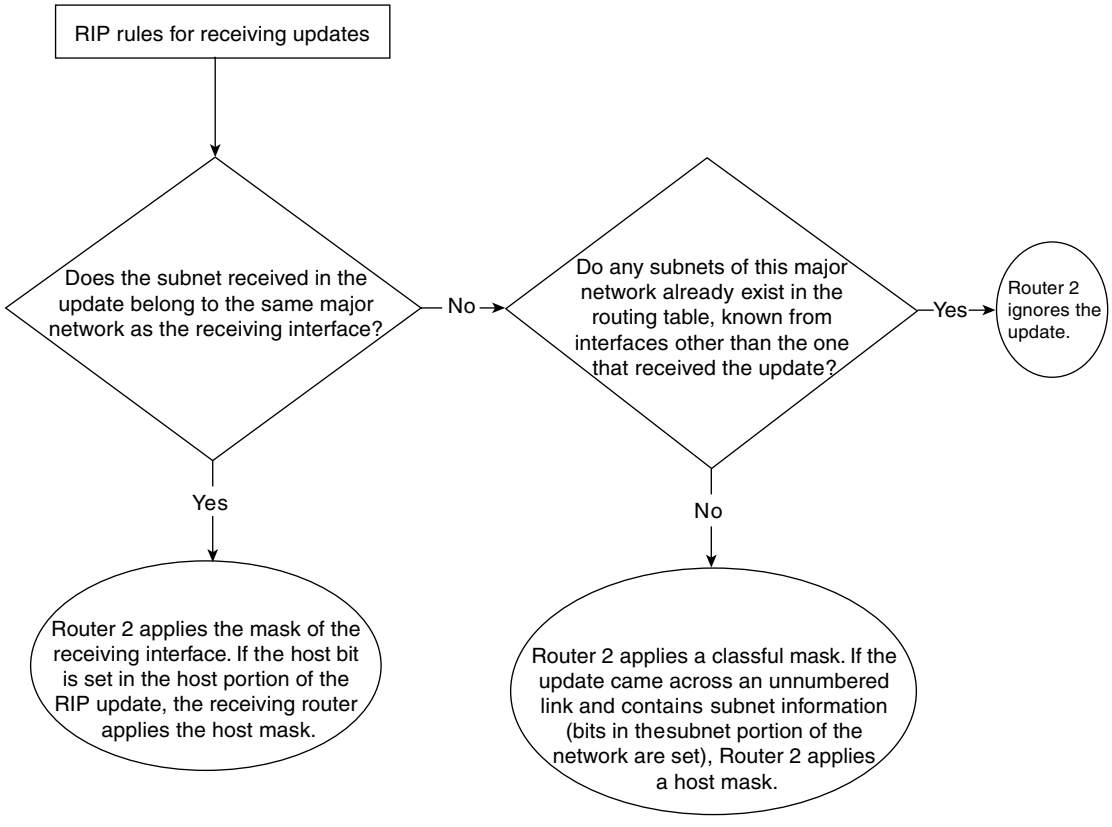
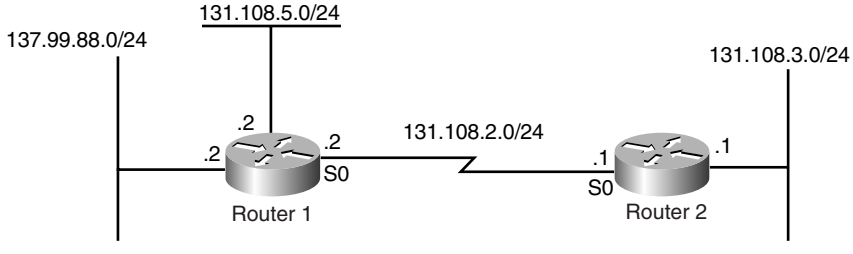


Figure 2-6 *An Example of RIP Behavior When Sending and Receiving Updates*



In Figure 2-6, when Router 1 sends an update to Router 2, it performs these checks:

1 Is 131.108.5.0/24 part of the same major network as 131.108.2.0/24, which is sourcing the update?

2. Yes. Does 131.108.5.0/24 have the same subnet mask 131.108.2.0/24, which is sourcing the update?
3. Yes. Router 1 advertises the network.
4. Is 137.99.88.0/24 part of the same major network as 131.108.2.0/24, which is sourcing the update?
5. No. Router 1 summarizes 137.99.88.0/24 at the major network boundary and advertises the route as 137.99.0.0.

This process results in Router 1 including 131.108.5.0 and 137.99.0.0 in its update to Router 2. You can see this in the output displayed using the **debug ip rip** command on Router 1, as demonstrated in Example 2-1.

Example 2-1 **debug ip rip** *Command Output Reveals RIP Update Information Sent*

```
Router1#debug ip rip
RIP: sending v1 update to 255.255.255.255 via Serial0 (131.108.2.2)
      subnet 131.108.5.0, metric 1
      network 137.99.0.0, metric 1
```

Example of Receiving Updates

Example 2-2 provides output from the **debug ip rip** command to display the routing update received on Router 2 from Router 1.

Example 2-2 **debug ip rip** *Command Output Reveals RIP Update Information Received*

```
Router2#debug ip rip
RIP: received v1 update from 131.108.2.2 on Serial0
      131.108.5.0 in 1 hops
      137.99.0.0 in 1 hops
```

Router 2 in Figure 2-6 performs the following checks to determine what mask to apply on a received network:

1. Is the received major network 137.99.0.0 the same as 131.108.2.0, which is the address assigned to the interface that received the update?
2. No. Do any subnets of this major network already exist in the routing table known from other interfaces?
3. No. Router 2 applies the natural mask (/16) because 137.99.0.0 is a Class B address.
4. Does subnet 131.108.5.0 belong to the same major network as subnet 131.108.2.0, which is the interface that received the update?
5. Yes. Router 2 applies the mask /24, which is the mask of the interface that received the update.

This process results in the networks and masks in Router 2's routing table, displayed using the **show ip route** command (see Example 2-3).

Example 2-3 **show ip route** *Command Output Reveals the Networks and Masks in Router 2's Routing Table*

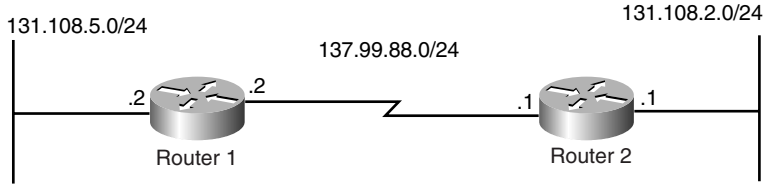
```
Router2#show ip route
R       137.99.0.0/16 [120/1] via 131.108.2.2, 00:00:07, Serial0
      131.108.0.0/24 is subnetted, 3 subnets
R       131.108.5.0 [120/1] via 131.108.2.2, 00:00:08, Serial0
C       131.108.2.0 is directly connected, Serial0
C       131.108.3.0 is directly connected, Ethernet0
```

Why RIP Doesn't Support Discontiguous Networks

A discontiguous network is comprised of a major network separated by another major network. In Figure 2-7, network 131.108.0.0 is separated by a subnet of network 137.99.0.0; here, 131.108.0.0 is a discontiguous network.

RIP is a classful protocol. Whenever RIP advertises a network across a different major network boundary, RIP summarizes the advertised network at the major network boundary. In Figure 2-7, when Router 1 sends an update containing 131.108.5.0 to Router 2 across 137.99.88.0, it converts 131.108.5.0/24 into 131.108.0.0/16. This process is called auto-summarization.

Figure 2-7 *An Example of a Discontiguous Network*



Router 1 takes the following steps before sending an update to Router 2:

1. Is 131.108.5.0/24 part of the same major network as 137.99.88.0/24, which is the subnet assigned to the interface that's sourcing the update?
2. No. Router 1 summarizes 131.108.5.0/24 and advertises the route 131.108.0.0/16.

The **debug ip rip** command output on Router 1 shows the update sent by Router 1, as demonstrated in Example 2-4.

Example 2-4 **debug ip rip** *Command Output Reveals RIP Update Information Sent by Router 1 in Figure 2-7*

```
Router1#debug ip rip
RIP: sending v1 update to 255.255.255.255 via Serial0 (137.99.88.2)
     network 131.108.0.0, metric 1
```

Router 2 goes through the following steps before accepting the update from Router 1:

1. Is the major network received (131.108.0.0) the same as the major network of 137.99.88.0/24, which is the subnet assigned to the interface that received the update?
2. No. Do any subnets of this major network already exist in the routing table known from interfaces other than that which received the update?
3. Yes. Router 2 ignores the update.

Again, **debug ip rip** command output on Router 2 shows the update received by Router 2, as demonstrated in Example 2-5.

Example 2-5 **debug ip rip** *Command Output Reveals RIP Update Information Received by Router 2 in Figure 2-7*

```
Router2#debug ip rip
RIP: received v1 update from 137.99.88.1 on Serial0
     131.108.0.0 in 1 hops
```

The routing table of Router 2, as demonstrated in the **show ip route** command output in Example 2-6, shows that the update was ignored. The only entry for any subnetwork or network on 131.108.0.0 is the one directly connected to Ethernet0.

Example 2-6 **show ip route** *Command Output Reveals That the Routing Table for Router 2 in Figure 2-7 Does Not Reflect the Advertised Route Sent by Router 1*

```
  137.99.0.0/24 is subnetted, 1 subnets
     C       137.99.88.0 is directly connected, Serial0
          131.108.0.0/24 is subnetted, 3 subnets
     C       131.108.2.0 is directly connected, Ethernet0
```

To avoid having updates ignored, configure a static route on both routers that points toward the specific subnets. For example, on Router 1, configure the following:

```
Router1(config)#ip route 131.108.2.0 255.255.255.0 137.99.88.1
```

On Router 2, configure the following:

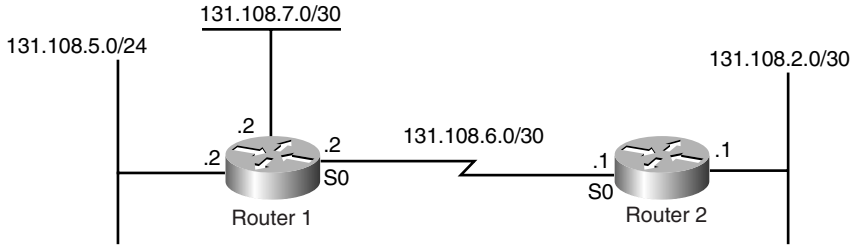
```
Router2(config)#ip route 131.108.5.0 255.255.255.0 137.99.88.2
```

Why RIP Doesn't Support Variable-Length Subnet Masking

The capability to specify a different subnet mask for the same network number is called *variable-length subnet masking (VLSM)*. RIP and IGRP are classful protocols and are incapable of carrying subnet mask information in their updates. Before RIP or IGRP sends an update, it performs a check against the subnet mask of the network that is about to be advertised, with the subnet mask of the interface sourcing the update. If the two subnet masks don't match, the update gets dropped.

The following example demonstrates this concept. In Figure 2-8, Router 1 has three subnets with two different masks (/24 and /30).

Figure 2-8 *An Example of a VLSM Network*



Router 1 goes through the following steps before sending an update to Router 2:

1. Router 1 checks to see if 131.108.5.0/24 is part of the same major network as 131.108.6.0/30, which is the network assigned to the interface that is sourcing the update.
2. It is part of the same major network, so Router 1 determines whether 131.108.5.0/24 has the same subnet mask as 131.108.6.0/30.
3. Because the subnet masks are not the same, Router 1 drops the network and doesn't advertise the route.
4. Router 1 now determines whether 131.108.7.0/30 is part of the same major network as 131.108.6.0/30, which is the network assigned to the interface that is sourcing the update.
5. It is part of the same major network, so Router 1 next determines whether 131.108.7.0/30 has the same subnet mask as 131.108.6.0/30.
6. Because the two subnet masks are the same, Router 1 advertises the network.

The preceding procedure determined that Router 1 includes only 131.108.7.0 in its update that is sent to Router 2. The **debug ip rip** command in Example 2-7 actually shows the update sent by Router 1.

Example 2-7 **debug ip rip** *Command Output Reveals RIP Update Information Sent by Router 1 to Router 2, as Illustrated in Figure 2-8*

```
RIP: sending v1 update to 255.255.255.255 via Serial0 (131.108.6.2)
            subnet 131.108.7.0, metric 1
```

Notice in the output in Example 2-7 that the only subnet included in the update is 131.108.7.0. The subnet 131.108.5.0 is not included because it has a different subnet mask.

This results in the following entry in Router 2's routing table displayed by the **show ip route** command (see Example 2-8).

Example 2-8 **show ip route** *Command Output Reveals That the Subnet 131.108.5.0/25 Is Missing from Router 2's Routing Table*

```
Router2#show ip route
          131.108.0.0/30 is subnetted, 3 subnets
     R       131.108.7.0 [120/1] via 131.108.2.2, 00:00:08, Serial0
     C       131.108.6.0 is directly connected, Serial0
     C       131.108.2.0 is directly connected, Ethernet0
```

To avoid eliminating subnets from routing updates, either use the same subnet mask over the entire RIP network or use static routes for networks with different subnet masks.

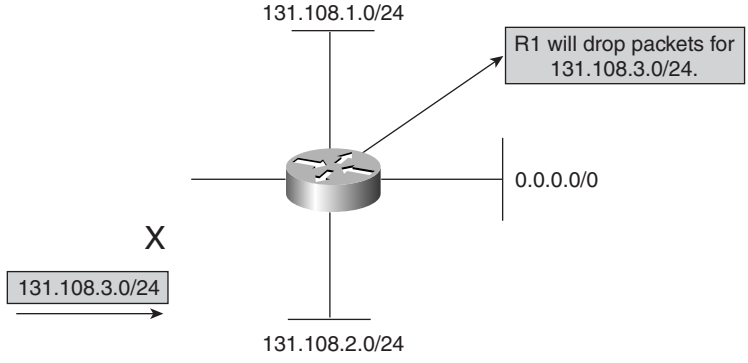
Default Routes and RIP

Cisco's RIP implementation supports the propagation of a default route, also known as 0.0.0.0/0. When RIP finds a default route in its routing table, it automatically advertises this in the RIP update.

One important thing to remember here is that the default route must have a valid metric. For example, if the default route is learned through OSPF and the metric is 20, RIP will advertise this router with a metric of infinity (16). So, for this situation, the **default-metric** command must be used under the **router rip** command to ensure that the proper metric is assigned to the update**.**

Classless and classful IP routing concepts play an important role, especially with default routes. With classful IP routing, if the router receives a packet destined for a subnet that it does not recognize and the network default route is missing in the routing table, the router discards the packet. Figure 2-9 explains this behavior.

Figure 2-9 *Classful IP Routing*



Here, Host X is sending traffic to the 131.108.3.0/24 subnet. Router R1 will discard these packets because it does not have a route for 131.108.3.0/24. Traffic will not be send to the default route because of the classful nature of routing.

If R1 enables IP classless routing, R1 will forward traffic to the default route.

Enabling IP classless routing is recommended when default network or default routes are used.

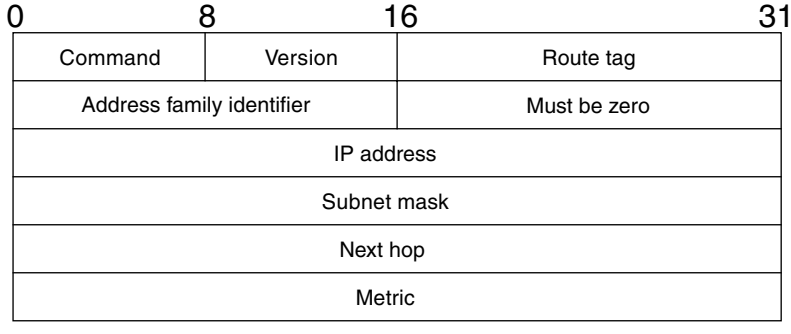
Protocol Extension to RIP

RIP Version 2 (RIP-2) made some improvements and enhancements to RIP-1. RIP-2 supports VLSM and discontiguous networks, and it offers the following enhancements:

- Route tag
- Subnet mask
- Next-hop metric
- Multicast capability
- Authentication

Figure 2-10 shows the RIP-2 packet format. The sections that follow discuss each of the enhancements and new packet fields in greater detail.

Figure 2-10 *RIP-2 Packet Format*



Route Tag

The Route Tag field is a 2-byte field that allows RIP routes to be assigned with a unique integer value. The routing table display shows the route tag for each RIP route, if assigned. This route tag plays an important role during redistribution with RIP. Any route that is redistributed into RIP gets tagged, to distinguish between internal RIP information and external RIP information.

When redistributed routes in RIP are assigned with route tags, it becomes easier to control redistribution of tagged routes into other protocols. Instead of matching against each route when redistributing into other protocols, RIP routes can simply be matched against the tag that they were assigned.

For example, consider that 10 static routes in a router are redistributed in RIP and are assigned a tag of 20. These static routes will be advertised in RIP as external routes with a tag of 20. If in some other router RIP is being redistributed into OSPF and OSPF wants only those 10 static routes to be redistributed, OSPF can simply match the tag information instead of listing each static route in its redistribution commands. In addition, if OSPF is being redistributed back into RIP at some other router, RIP should deny any routes that are tagged with 20. Matching against tags thus avoids IP routing loops as well.

Subnet Mask

Unlike RIP-1, RIP-2 carries subnet mask information along with the IP network number. If an IP network is variably subnetted, RIP-2 picks the subnet mask of each subnet and advertises to RIP-2 neighbors. RIP-2 routers in the network install routes with their respective subnets though a variable length of, say, /8, /15/, /24, and so on.

Support of VLSM also enables RIP-2 to understand discontiguous networks. In a discontiguous network, the IP supernet is divided by another IP block. Because RIP-2 can carry subnet mask information, each RIP-2 router has a route with the actual mask and routers can forward traffic properly.

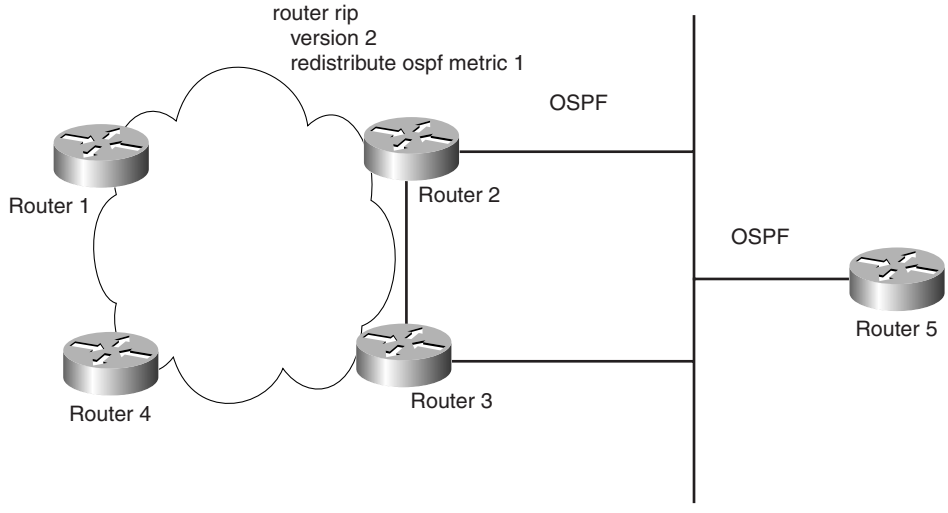
Next Hop

The Next Hop field was added to avoid an extra hop during packet forwarding. For those familiar with OSPF, the Next Hop field holds nearly the same role as the forwarding address for OSPF external routes.

In Figure 2-11, OSPF is enabled between Router 2 and Router 5. RIP is enabled on Router 2, Router 3, and all the other routers behind Router 2 and Router 3. Router 2 is doing redistribution between OSPF and RIP. Now when a packet from Router 1 is destined for OSPF networks and arrives at Router 2, it is forwarded to Router 5.

When a packet from Router 4 destined to the OSPF network arrives at Router 3, if there is no next-hop information (in case of RIP-1), Router 3 forwards the packet to the originator, Router 2. Then Router 2 forwards it to Router 5. This is an extra hop that Router 3 must take to get to the OSPF network. With the Next Hop field in the RIP packet, when a packet destined to the OSPF network arrives at Router 3, the RIP route for the destination network has its next hop set to Router 5 instead of Router 2. As a result, Router 3 does not forward the packet to Router 2—instead, it forwards the packet straight to Router 5.

Figure 2-11 *RIP-2 Packet Format*



Multicast Capability

RIP-2 uses multicast when sending an update to all its neighbors. This reduces unnecessary broadcast flooding on the wire. The multicast address that RIP-2 uses is 224.0.0.9.

All devices on the wire running RIP-2 listen for RIP-2 multicast packets on 224.0.0.9 at a multicast MAC address (01-00-5E-00-00-09). Devices not running RIP-2 simply discard RIP-2 messages on the wire, reducing unnecessary load.

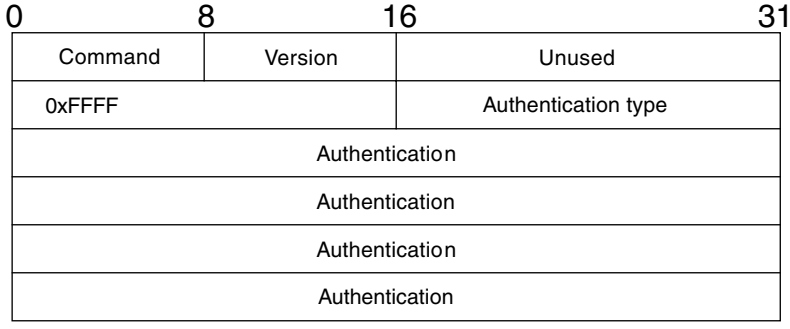
Authentication

RIP-2 supports simple password authentication, to validate trusted RIP-2 neighbors. RIP-2 speakers determine whether authentication is used by looking at the address family identifier (AFI) in RIP-2 packet. AFI in RIP-2 header indicates what kind of addresses are present in the rest of the packet.

If the AFI value is 0xFFFF, this means that the remainder of the entire RIP packet contains authentication information.

Figure 2-12 shows the packet format when authentication is used.

Figure 2-12 *RIP-2 Packet Format for Authentication*



Compatibility Issues

RIP-1 and RIP-2 can be run together in a network. You should be aware of a few things when running both protocols in your network:

- **Autosummarization**—RIP-1 and RIP-2 can be run together in a network. RFC 1723 for RIP-2 recommends disabling the autosummarization feature when using both RIP-1 and RIP-2.
- **Subnet advertisement**—If a more specific subnet is advertised to a RIP-1 router, the router might mistakenly take it as a host route update.
- **Queries**—When a RIP-2 router receives a query request from a RIP-1 router, it responds with a RIP-1 message. If the router is configured to send only RIP-2 messages, such a query request must be ignored.
- **Version field**—The Version field in the RIP packet determines how to handle RIP-1 and RIP-2 packets:
  - If version = 0 in the RIP packet, the packet is discarded, regardless of what version the receiving router is running.
  - If version = 1 in the RIP packet, all the “must be zero fields” are checked (refer to Figure 2-9). If the version is nonzero, the packet is discarded, regardless of what version the receiving router is running.
  - If version = 2 in the RIP packet and the receiving router is running RIP-1, the receiving router should look at only the related information in the packet. All the “must be zero fields” are ignored.

Summary

RIP is a distance vector protocol that uses the Bellman-Ford algorithm to compute IP routes dynamically. RIP is suitable to run in small IP networks because of its hop count limit of 15. RIP was designed as a simple IP routing protocol that exchanges a complete routing table at a fixed interval (30 seconds) with other routers running RIP. In larger networks with a large number of IP routes, sending a complete routing table every 30 seconds is not practical. This results in extra work for the sender and receiver, and it consumes unnecessary bandwidth and processing time. Therefore, RIP is used in smaller networks with a hop count of less than 15 and a small number of routes as well.

RIP offers a descent algorithm for loop avoidance by using split horizon and poison reverse. Split horizon takes care of the loops by not advertising any routes back to the interface where it learned the routes. Poison reverse causes routes to be advertised with the infinite RIP metric (16), thus removing RIP routes that might be looped or down.

Because any change in the network takes at least 30 seconds to propagate, the concept of holddown causes the RIP routing table to wait for three times the advertisement interval. This implementation is designed for when a RIP route is not advertised because it might have been down for a little over 30 seconds. The receiving routers should wait for 90 seconds to remove the route from the routing table. If a routes comes back before 90 seconds, it is reinstalled and is advertised throughout the network.

In the early days of IP networking, RIP was the protocol of choice in smaller IP networks. Since then, a lot of new IP protocols have been developed to be more robust and dynamic than RIP; they can scale up to a much larger number of routers than 15. The advent of these new protocols, such as OSPF, IS-IS, and EIGRP, resulted in almost complete phaseout of RIP from larger networks today. These new protocols have improved upon the limitations of RIP in terms of convergence and scalability, and they offer the support for VLSM and discontiguous networks that RIP-1 lacked.

Although RIP-2 improved RIP with new features, such as route tags, queries, subnet masks, next hops, multicasting, and authentication, larger networks still prefer OSPF, IS-IS, and EIGRP as IP routing protocols.

Review Questions

1. What is the maximum metric in RIP?
2. Why doesn't RIP support discontiguous networks?
3. Why doesn't RIP support VLSM?
4. What is the default update interval for RIP?
5. What transport protocol and port number do RIP use for sending updates?

6 What is the purpose of the split-horizon technique?

7 Does RIP Version 2 solve the discontiguous network problem by default?

8 Does RIP Version 2 also use broadcast for sending updates?

9 Does RIP support authentication?

Further Reading

Refer to the following RFCs for more information about RIP. You can access all RFCs online at www.isi.edu/in-notes/rfc*xxxx*.txt, where *xxxx* is the number of the RFC that you want to read.

RFC 1058, "Routing Information Protocol"
RFC 1723, "RIP Version 2"
RFC 2453, "RIP Version 2"
RFC 1582, "Extensions to RIP to Support Demand Circuits"
RFC 2091, "Triggered Extensions to RIP to Support Demand Circuits"
RFC 2082, "RIP-2 MD5 Authentication"

This chapter covers the following key topics:

- Troubleshooting RIP routes installation
- Troubleshooting RIP routes advertisement
- Troubleshooting routes summarization in RIP
- Troubleshooting RIP redistribution problems
- Troubleshooting dial-on-demand (DDR) routing issues in RIP
- Troubleshooting route flapping problem in RIP

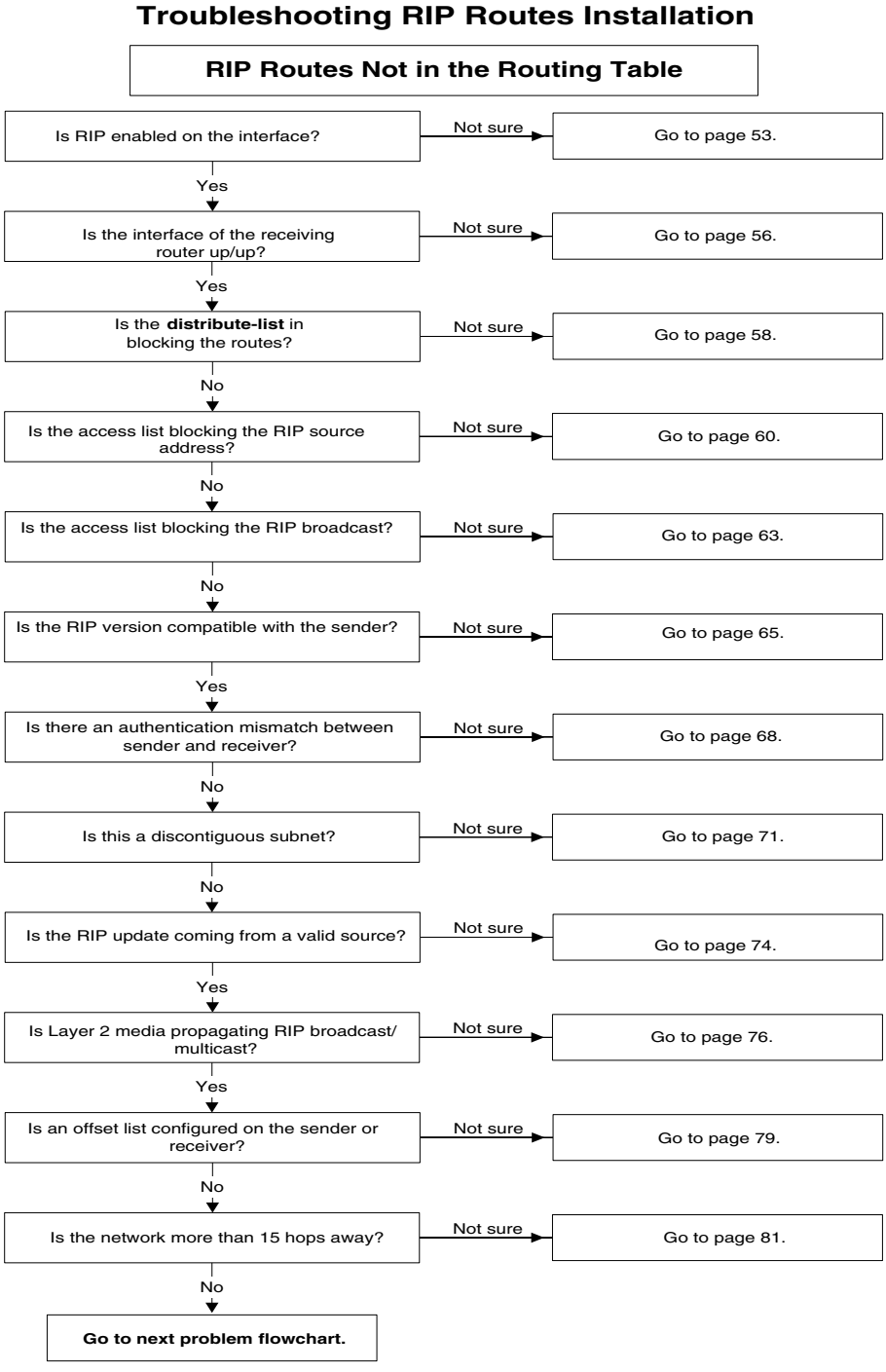
CHAPTER 3

Troubleshooting RIP

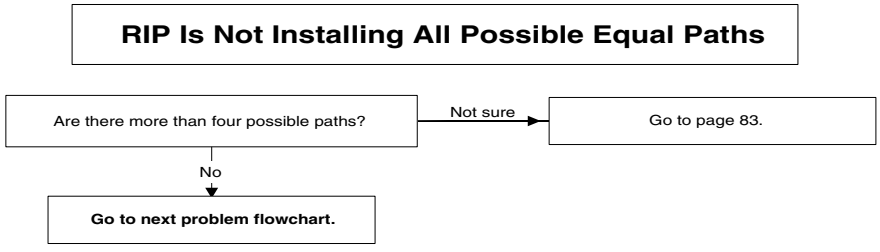
This chapter discusses some of the common problems in RIP and tells how to resolve those problems. At this time, no RIP error messages will help troubleshooting RIP problems. As a result, you will need to rely on debugs, configurations, and useful **show** commands, which we'll provide where necessary in this chapter. The flowcharts that follow document how to address common problems with RIP with the methodology used in this chapter.

Debugs sometimes can be very CPU-intensive and can cause congestion on your network. Therefore, we do not recommend turning on these debugs if you have a large network (that is, more than 100 networks or subnets in RIP). Sometimes, there could be multiple causes for the same problem—for example, Layer 2 is down, the **network** statement is wrong, and the sender is missing the **network** statement. Bringing up Layer 2 and fixing the **network** statement might not fix the network problem because the sender is still missing the **network** statement. Therefore, if one scenario doesn't fix the network problem, check into other scenarios. The word *RIP,* in general, refers to both RIP Version 1 (RIP-1) and RIP Version 2 (RIP-2). The problems discussed in this chapter are mostly related to RIP-1, unless specified as RIP-2.

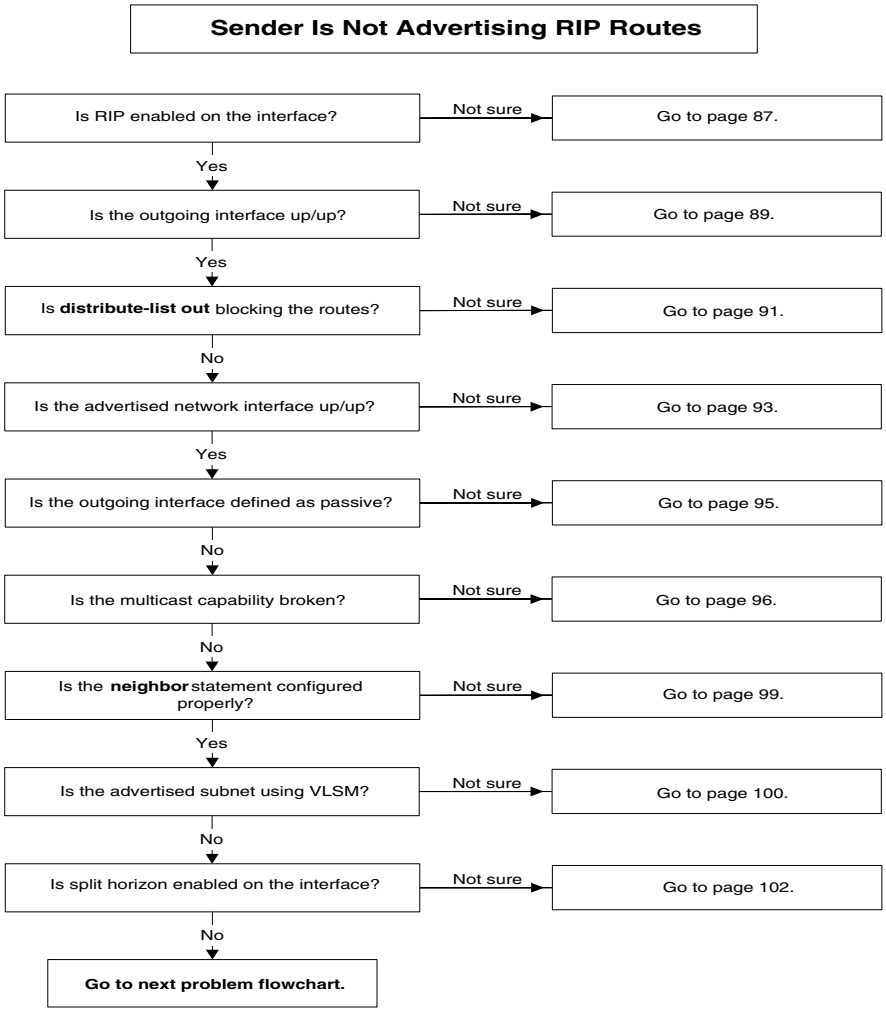
Flowcharts to Solve Common RIP Problems



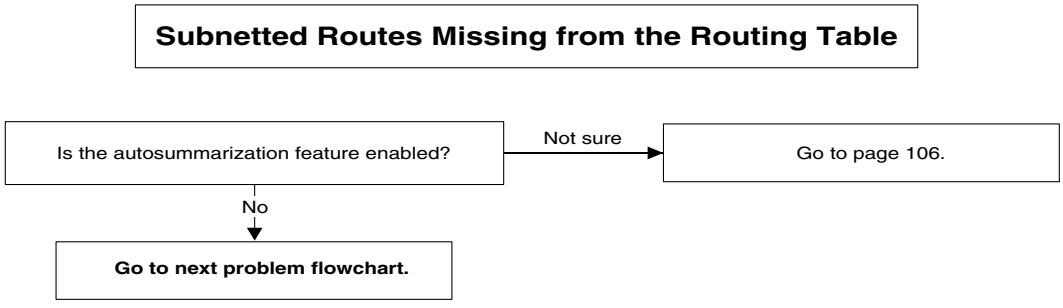
Troubleshooting RIP Routes Installation



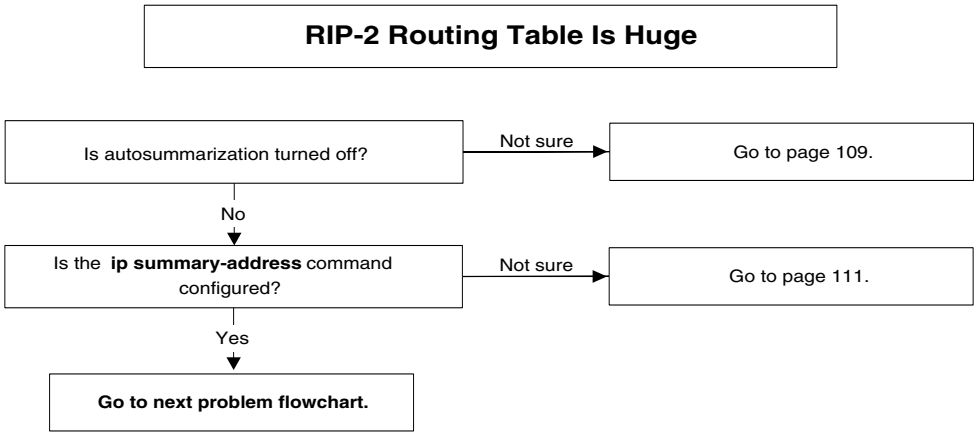
Troubleshooting RIP Route Advertisement



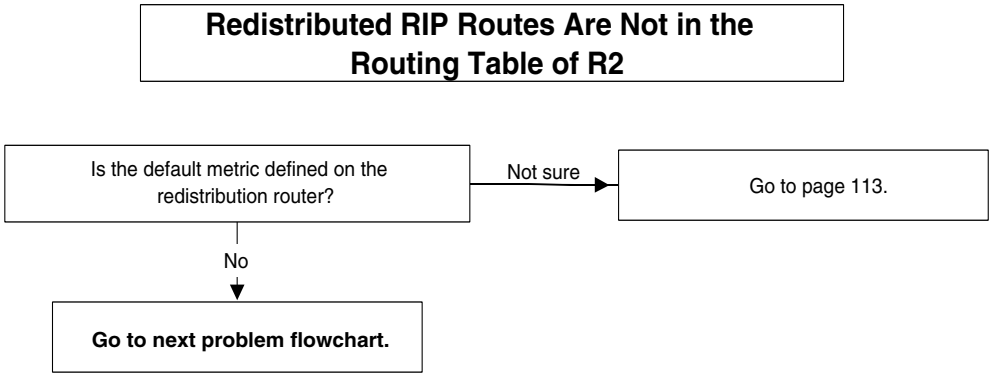
Troubleshooting RIP Route Advertisement



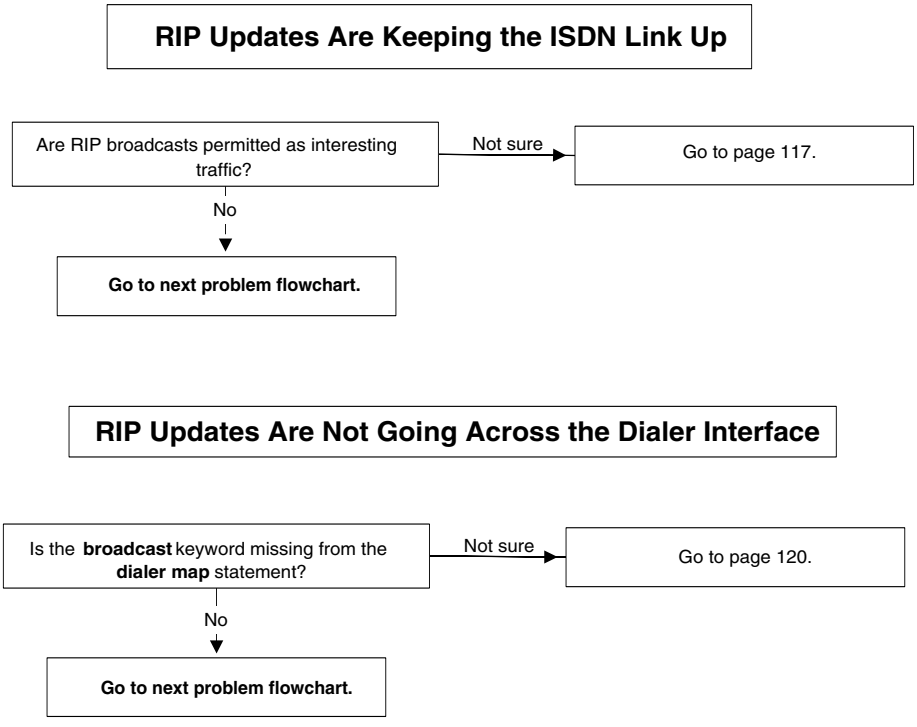
Troubleshooting Route Summarization in RIP



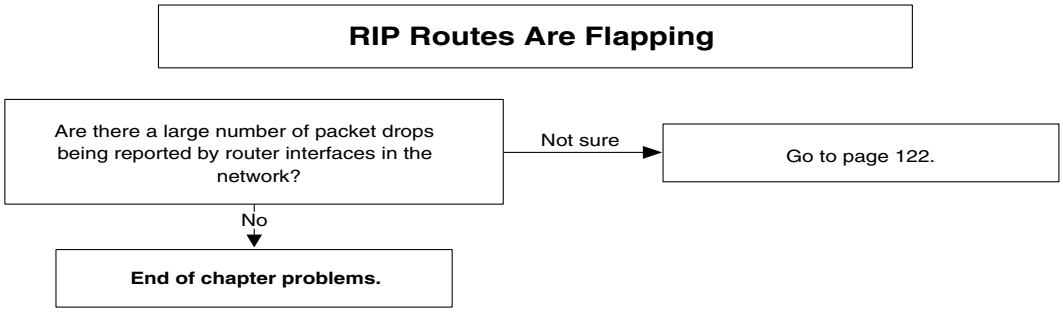
Troubleshooting RIP Redistribution Problems



Troubleshooting Dial-on-Demand Routing Issues in RIP



Troubleshooting Route Flapping Problems in RIP



Troubleshooting RIP Routes Installation

This section discusses several possible scenarios that can prevent RIP routes from getting installed in the routing table. This section is selected first in the troubleshooting list because the most common problem in RIP is that routes are not installed in the routing table.

If the routes are not installed in the routing table, the router will not forward the packets to destinations that are not in the routing table. When this happens, it creates reachability problems. Users start complaining that they cannot reach a server or a printer. When you investigate this problem, the first thing to ask is, "Do I have a route for this destination that users are complaining about?"

Three possibilities exist for routes not getting installed in the routing table:

- **Receiver's problem**—The router is receiving RIP updates but is not installing the RIP routes.
- **Intermediate media problem (Layer 2)**—Mostly related to Layer 2, the sender has sent the RIP updates, but they got lost in the middle and the receiver didn't receive them.
- **Sender's problem**—The sender is not even advertising RIP routes, so the receiving side is not seeing any RIP routes in the routing table.

The sender's problem will be discussed in the section "Troubleshooting RIP Route Advertisement." Two problems are related to RIP installation:

- RIP routes are not in the routing table.
- RIP is not installing all equal-cost path routes.

In the first problem, RIP is not installing any path to a specific network. In the second problem, RIP is not installing all paths to the network. Note that, in the second problem, the destination device is still reachable, but it's not listing all possible paths.

Problem: RIP Routes Not in the Routing Table

The routing table must have a network entry to send the packets to the desired destination. If there is no entry for the specific destination, the router will discard all the packets for this destination.

Example 3-1 shows that the routing table of R2 doesn't hold an entry for network 131.108.2.0.

Example 3-1 *Routing Table for R2 Shows No RIP Routes for Subnet 131.108.2.0*

```
R2#show ip route 131.108.2.0
% Subnet not in table
R2#
```

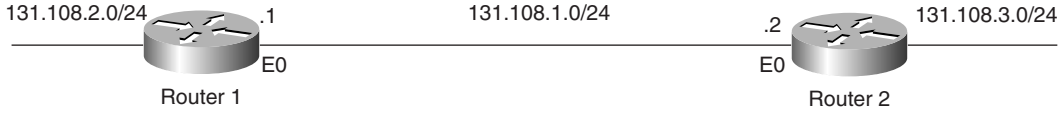
The possible causes for this problem are as follows:

- Missing or incorrect **network** statement
- Layer 2 down
- Distribute list blocking the route
- Access list blocking RIP source address
- Access list blocking RIP broadcast/multicast
- Incompatible version type
- Mismatch authentication key (RIP-2)
- Discontiguous network
- Invalid source
- Layer 2 problem (switch, Frame Relay, other Layer 2 media)
- Offset list with a large metric defined
- Routes that reached RIP hop-count limit
- Sender problem (discussed in the next chapter)

Figure 3-1 provides a network scenario that will be used as the basis for troubleshooting a majority of the aforementioned causes of the problem of RIP routes not in the routing table. The sections that follow carefully dissect how to troubleshoot this problem based on specific causes.

Figure 3-1 shows a setup in which Router 1 and Router 2 are running RIP between them.

Figure 3-1 *Example Topology for the Problem of RIP Routes Not in the Routing Table*



RIP Routes Not in the Routing Table—Cause: Missing or Incorrect network Statement

When you confirm that the route is missing from the routing table, the next step is to find out why. A route can be missing from the routing table for many reasons. The flowcharts at the beginning of this chapter can help isolate the cause that seems to fit most in your situation.

The obvious thing to check after discovering that the routes are not in the routing table is the router's configurations. Also check to see whether the **network** statement under **router rip** is properly configured.