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
Troubleshooting IP Routing Protocols

Faraz Shamim, CCIE #4131, 
Zaheer Aziz, CCIE #4127, 
Johnson Liu, CCIE #2637, 
and Abe Martey, CCIE #2373
Troubleshooting IP Routing Protocols

Faraz Shamim, Zaheer Aziz, Johnson Liu, Abe Martey

Copyright© 2002 Cisco Systems, Inc.

Published by:
Cisco Press
201 West 103rd Street
Indianapolis, IN 46290 USA

All rights reserved. No part of this book may be reproduced or transmitted in any form or by any means, electronic or mechanical, including photocopying, recording, or by any information storage and retrieval system, without written permission from the publisher, except for the inclusion of brief quotations in a review.

Printed in the United States of America 1 2 3 4 5 6 7 8 9 0
First Printing May 2002

Library of Congress Cataloging-in-Publication Number: 2001086619

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.
About the Authors

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.

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.

About the Technical Reviewers

Brian Morgan, CCIE #4865, CCSL, 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 go 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

Introduction xxxiv

Chapter 1 Understanding IP Routing 3
Chapter 2 Understanding Routing Information Protocol (RIP) 29
Chapter 3 Troubleshooting RIP 47
Chapter 4 Understanding Interior Gateway Routing Protocol (IGRP) 127
Chapter 5 Troubleshooting IGRP 137
Chapter 6 Understanding Enhanced Interior Gateway Routing Protocol (EIGRP) 207
Chapter 7 Troubleshooting EIGRP 227
Chapter 8 Understanding Open Shortest Path First (OSPF) 295
Chapter 9 Troubleshooting OSPF 341
Chapter 10 Understanding Intermediate System-to-Intermediate System (IS-IS) 533
Chapter 11 Troubleshooting IS-IS 585
Chapter 12 Understanding Protocol Independent Multicast (PIM) 625
Chapter 13 Troubleshooting PIM 643
Chapter 14 Understanding Border Gateway Protocol Version 4 (BGP-4) 659
Chapter 15 Troubleshooting BGP 719
Appendix Answers to Review Questions 839
Index 849
# Table of Contents

Introduction xxxiv

**Chapter 1  Understanding IP Routing  3**

- IP Addressing Concepts 5
  - IPv4 Address Classes  5
  - IPv4 Private Address Space 7
  - Subnetting and Variable-Length Subnet Masks 8
  - Classless Interdomain Routing 10

- Static and Dynamic Routes 11

  - Dynamic Routing 11
    - Unicast Versus Multicast IP Routing 12
    - Classless Versus Classful IP Routing Protocols 15
    - Interior Gateway Protocols Versus Exterior Gateway Protocols 15
    - Distance Vector Versus Link-State Protocols 18
      - Distance Vector Routing Concepts 18
      - Link-State Protocols 23
  
- Routing Protocol Administrative Distance 24
  
- Fast Forwarding in Routers 25
  
- Summary 26
  
- Review Questions 26
  
- References 27

**Chapter 2 Understanding Routing Information Protocol (RIP)  29**

- Metric 29
  
- Timers 30
  
- Split Horizon 30
  
- Split Horizon with Poison Reverse 30

- RIP-1 Packet Format 31

  - RIP Behavior 31
    - RIP Rules for Sending Updates 31
    - RIP Rules for Receiving Updates 33
    - Example of Sending Updates 33
    - Example of Receiving Updates 35

- Why RIP Doesn’t Support Discontiguous Networks 36
Why RIP Doesn’t Support Variable-Length Subnet Masking 37
Default Routes and RIP 39
Protocol Extension to RIP 40
  Route Tag 40
  Subnet Mask 41
  Next Hop 41
  Multicast Capability 42
  Authentication 42
Compatibility Issues 43
Summary 44
Review Questions 44
Further Reading 45

Chapter 3  Troubleshooting RIP 47
Flowcharts to Solve Common RIP Problems 48
Troubleshooting RIP Routes Installation 52
Problem: RIP Routes Not in the Routing Table 52
  RIP Routes Not in the Routing Table—Cause: Missing or Incorrect network
    Statement 53
      Debugs and Verification 54
      Solution 55
  RIP Routes Not in the Routing Table—Cause: Layer 1/2 Is Down 56
    Debugs and Verification 57
    Solution 58
  RIP Routes Not in the Routing Table—Cause: distribute-list in Is Blocking the
    Route 58
    Debugs and Verification 58
    Solution 59
  RIP Routes Not in the Routing Table—Cause: Access List Blocking RIP Source
    Address 60
      Debugs and Verification 60
      Solution 62
  RIP Routes Not in the Routing Table—Cause: Access List Blocking RIP Broadcast
    or Multicast (in Case of RIP-2) 63
      Debugs and Verification 63
      Solution 64
  RIP Routes Not in the Routing Table—Cause: Incompatible RIP Version Type 65
    Debugs and Verification 65
    Solution 67
RIP Routes Not in the Routing Table—Cause: Mismatch Authentication Key
(RIP-2) 68
  - Debrids and Verification 69
  - Solution 70
RIP Routes Not in the Routing Table—Cause: Discontiguous Network 71
  - Debrids and Verification 72
  - Solution 73
RIP Routes Not in the Routing Table—Cause: Invalid Source 74
  - Debrids and Verification 74
  - Solution 76
RIP Routes Not in the Routing Table—Cause: Layer 2 Problem (Switch, Frame
  Relay, Other Layer 2 Media) 76
  - Debrids and Verification 77
  - Solution 78
RIP Routes Not in the Routing Table—Cause: Offset List Has a Large Metric
  Defined 79
  - Debrids and Verification 80
  - Solution 81
RIP Routes Not in the Routing Table—Cause: Routes Reached RIP Hop Count
  Limit 81
  - Debrids and Verification 82
  - Solution 83
Problem: RIP Is Not Installing All Possible Equal-Cost Paths—Cause: maximum-path
  Command Restricts RIP from Installing More Than One Path 83
  - Debrids and Verification 85
  - Solution 85
Troubleshooting RIP Routes Advertisement 86
Problem: Sender Is Not Advertising RIP Routes 86
  Sender Is Not Advertising RIP Routes—Cause: Missing or Incorrect network
    Statement 87
    - Debrids and Verifications 88
    - Solution 88
  Sender Is Not Advertising RIP Routes—Cause: Outgoing Interface Is Down 89
    - Debrids and Verification 90
    - Solution 91
  Sender Is Not Advertising RIP Routes—Cause: distribute-list out Is Blocking the
    Route 91
    - Debrids and Verification 91
    - Solution 92
  Sender Is Not Advertising RIP Routes—Cause: Advertised Network Interface Is
    Down 93
    - Debrids and Verification 94
    - Solution 94
Sender Is Not Advertising RIP Routes—Cause: Outgoing Interface Is Defined Passive 95
  Debugs and Verification 95
  Solution 96

Sender Is Not Advertising RIP Routes—Cause: Broken Multicast Capability (Frame Relay) 96
  Debugs and Verification 97
  Solution 98

Sender Is Not Advertising RIP Routes—Cause: Misconfigured neighbor Statement 99
  Debugs and Verification 99
  Solution 100

Sender Is Not Advertising RIP Routes—Cause: Advertised Subnet Is VLSM 100
  Debugs and Verification 101
  Solution 101

Sender Is Not Advertising RIP Routes—Cause: Split Horizon Is Enabled 102
  Debugs and Verification 104
  Solution 105

Problem: Subnetted Routes Missing from the Routing Table of R2—Cause: Autosummarization Feature Is Enabled 106
  Debugs and Verification 108
  Solution 108

Troubleshooting Routes Summarization in RIP 109

Problem: RIP-2 Routing Table Is Huge—Cause: Autosummarization Is Off 109
  Debugs and Verification 110
  Solution 111

Problem: RIP-2 Routing Table Is Huge—Cause: ip summary-address Is Not Used 111
  Debugs and Verification 112
  Solution 112

Troubleshooting RIP Redistribution Problems 113
  Debugs and Verification 115
  Solution 115

Troubleshooting Dial-on-Demand Routing Issues in RIP 116

Problem: RIP Broadcast Is Keeping the ISDN Link Up—Cause: RIP Broadcasts Have Not Been Denied in the Interesting Traffic Definition 117
  Debugs and Verification 118
  Solution 119

Problem: RIP Updates Are Not Going Across the Dialer Interface—Cause: Missing broadcast Keyword in a dialer map Statement 120
  Debugs and Verification 121
  Solution 122
Chapter 4  Understanding Interior Gateway Routing Protocol (IGRP)  127

Metrics  127
Timers  129
Split Horizon  130
Split Horizon with Poison Reverse  130
IGRP Packet Format  131
IGRP Behavior  131
Default Route and IGRP  132
Unequal-Cost Load Balancing in IGRP  133
Summary  135
Review Questions  135

Chapter 5  Troubleshooting IGRP  137

Flowcharts to Solve Common IGRP Problems  138
Troubleshooting IGRP Route Installation  142

Problem: IGRP Routes Not in the Routing Table  142
IGRP Routes Not in the Routing Table—Cause: Missing or Incorrect network Statement  143
   Debugs and Verification  144
   Solution  145
IGRP Routes Not in the Routing Table—Cause: Layer 1/2 Is Down  147
   Debugs and Verification  147
   Solution  148
IGRP Routes Not in the Routing Table—Cause: distribute-list in Is Blocking the Route  149
   Debugs and Verification  150
   Solution  150
IGRP Routes Not in the Routing Table—Cause: Access List Blocking IGRP Source Address  151
   Debugs and Verification  151
   Solution  152
IGRP Routes Not in the Routing Table—Cause: Access List Blocking IGRP Broadcast  153
   Debugs and Verification  154
   Solution  155
IGRP Routes Not in the Routing Table—Cause: Discontiguous Network 155
  Debugs and Verification 156
  Solution 157
IGRP Routes Not in the Routing Table—Cause: Invalid Source 159
  Debugs and Verification 160
  Solution 160
IGRP Routes Not in the Routing Table—Cause: Layer 2 Problem (Switch, Frame
  Relay, Other Layer 2 Media) 161
  Debugs and Verification 162
  Solution 162
IGRP Routes Not in the Routing Table—Cause: Sender’s AS Mismatch 163
  Debugs and Verification 164
  Solution 165

Problem: IGRP Is Not Installing All Possible Equal-Cost Paths—Cause: maximum-
  paths Restricts IGRP to a Maximum of Four Paths by Default 166
  Debugs and Verification 167
  Solution 168

Troubleshooting IGRP Routes Advertisement 168

Problem: Sender Is Not Advertising IGRP Routes 169
  Sender Is Not Advertising IGRP Routes—Cause: Missing or Incorrect network
  Statement 169
    Debugs and Verification 170
    Solution 170
  Sender Is Not Advertising IGRP Routes—Cause: Outgoing Interface Is
  Down 171
    Debugs and Verification 172
    Solution 172
  Sender Is Not Advertising IGRP Routes—Cause: distribute-list out Is Blocking the
  Route 173
    Debugs and Verification 174
    Solution 174
  Sender Is Not Advertising IGRP Routes—Cause: Advertised Network Interface Is
  Down 175
    Debugs and Verification 175
    Solution 176
  Sender Is Not Advertising IGRP Routes—Cause: Outgoing Interface Is Defined as
  Passive 176
    Debugs and Verification 177
    Solution 178
  Sender Is Not Advertising IGRP Routes—Cause: Broken Broadcast Capability
  (Encapsulation Failure in Layer 2) 178
    Debugs and Verification 179
    Solution 180
Sender Is Not Advertising IGRP Routes—Cause: Misconfigured neighbor
  Statement 180
    Debugs and Verification 181
    Solution 181
Sender Is Not Advertising IGRP Routes—Cause: Advertised Subnet Is
  VLSM 182
    Debugs and Verification 183
    Solution 183
Sender Is Not Advertising IGRP Routes—Cause: Split Horizon Is Enabled 184
    Debugs and Verification 186
    Solution 187
Problem: Candidate Default Is Not Being Advertised—Cause: ip default-network
  Command Is Missing 188
    Debugs and Verification 189
    Solution 190
Troubleshooting IGRP Redistribution Problems 191
Problem: Redistributed Routes Are Not Getting Installed in the Routing Table—Cause:
  Metric Is Not Defined During Redistribution into IGRP 191
    Debugs and Verification 192
    Solution 193
Troubleshooting Dial-on-Demand Routing (DDR) Issues in IGRP 194
Problem: IGRP Broadcast Is Keeping the ISDN Link Up—Cause: IGRP Broadcasts
  Have Not Been Denied in the Interesting Traffic Definition 194
    Debugs and Verification 195
    Solution 196
Problem: IGRP Updates Are Not Going Across the Dialer Interface—Cause: Missing
  Broadcast Keyword in a dialer map Statement 197
    Debugs and Verification 197
    Solution 198
Troubleshooting Route Flapping Problem in IGRP 198
Problem: IGRP Routes Are Flapping—Cause: Packet Drops on Sender’s or Receiver’s
  Interface 199
    Debugs and Verification 200
    Solution 201
Troubleshooting Variance Problem 201
Problem: IGRP Not Using Unequal-Cost Path for Load Balancing—Cause: variance
  Command Is Missing or Misconfigured 202
    Debugs and Verification 203
    Solution 204
Chapter 6  Understanding Enhanced Interior Gateway Routing Protocol (EIGRP)  207

Metrics 208
EIGRP Neighbor Relationships 209
The Diffusing Update Algorithm 211
DUAL Finite-State Machine 213
EIGRP Reliable Transport Protocol 214
EIGRP Packet Format 215
EIGRP Behavior 218
EIGRP Summarization 219
EIGRP Query Process 220
Default Routes and EIGRP 221
Unequal-Cost Load Balancing in EIGRP 221
Summary 223
Review Questions 224

Chapter 7  Troubleshooting EIGRP 227

Troubleshooting EIGRP Neighbor Relationships 227
Consulting the EIGRP Log for Neighbor Changes 228
EIGRP Neighbor Problem—Cause: Unidirectional Link 230
EIGRP Neighbor Problem—Cause: Uncommon Subnet 233
Misconfiguration of the IP Address on the Interfaces 234
Primary and Secondary IP Addresses of the Neighboring Interface Don’t Match 234
Switch or Hub Between EIGRP Neighbor Connection Is Misconfigured or Is Leaking Multicast Packets to Other Ports 235
EIGRP Neighbor Problem—Cause: Mismatched Masks 235
EIGRP Neighbor Problem—Cause: Mismatched K Values 237
EIGRP Neighbor Problem—Cause: Mismatched AS Number 239
EIGRP Neighbor Problem—Cause: Stuck in Active 240
Reviewing the EIGRP DUAL Process 240
Determining Active/Stuck in Active Routes with show ip eigrp topology active 242
Methodology for Troubleshooting the Stuck in Active Problem 244

Troubleshooting EIGRP Route Advertisement 250
EIGRP Is Not Advertising Routes to Neighbors When the Network Administrators Think That It Should 251
EIGRP Is Not Advertising Routes to Its Neighbors—Cause: Distribute List 251
EIGRP Is Not Advertising Routes to Its Neighbors—Cause: Discontiguous Networks 252
EIGRP Is Not Advertising Routes to Neighbors—Cause: Split-Horizon Issues 253
EIGRP Is Advertising Routes to Neighbors When the Network Administrators Think That It Shouldn’t 257
EIGRP Is Advertising Routes with Unexpected Metric 259

Troubleshooting EIGRP Route Installation 264
EIGRP Is Not Installing Routes—Cause: Auto or Manual Summarization 265
EIGRP Is Not Installing Routes—Cause: Higher Administrative Distance 267
EIGRP Is Not Installing Routes—Cause: Duplicate Router IDs 268

Troubleshooting EIGRP Route Flapping 271

Troubleshooting EIGRP Route Summarization 276
EIGRP Summarization Route Problem—Cause: Subnetworks of Summary Route Don’t Exist in Routing Table 276
EIGRP Summarization Route Problem—Cause: Too Much Summarization 278

Troubleshooting EIGRP Redistribution Problems 280

Troubleshooting EIGRP Dial Backup Problem 286

EIGRP Error Messages 291

Summary 292

Chapter 8  Understanding Open Shortest Path First (OSPF) 295

OSPF Packet Details 295
Hello Packets 297
Database Description Packets 299
Link-State Request Packets 300
Link-State Update Packets 301
Link-State Acknowledgment Packet 301

OSPF LSA Details 302
Router LSA 304
   Router LSA Example 305
Network LSA 307
   Network LSA Example 308
Summary LSA 309
   Summary LSA Example 310
External LSA 313
   External LSA Example 314

OSPF Areas 315
Normal Areas 319
Chapter 9  Troubleshooting OSPF  341

Flowcharts to Solve Common OSPF Problems  342

Troubleshooting OSPF Neighbor Relationships  351

Problem: OSPF Neighbor List Is Empty  351
  OSPF Neighbor List Is Empty—Cause: OSPF Not Enabled on the Interface  352
    Debugs and Verification  353
    Solution  354
  OSPF Neighbor List Is Empty—Cause: Layer 1/2 Is Down  354
    Debugs and Verification  355
    Solution  355
OSPF Neighbor List Is Empty—Cause: Interface Is Defined as Passive Under OSPF 356
  Debugs and Verification 357
  Solution 358

OSPF Neighbor List Is Empty—Cause: Access List Blocking OSPF Hellos on Both Sides 358
  Debugs and Verification 359
  Solution 360

OSPF Neighbor List Is Empty—Cause: Mismatched Subnet Number/Mask over a Broadcast Link 361
  Debugs and Verification 361
  Solution 362

OSPF Neighbor List Is Empty—Cause: Mismatched Hello/Dead Intervals 362
  Debugs and Verification 363
  Solution 364

OSPF Neighbor List Is Empty—Cause: Mismatched Authentication Type 364
  Debugs and Verification 365
  Solution 366

OSPF Neighbor List Is Empty—Cause: Mismatched Authentication Key 366
  Debugs and Verification 367
  Solution 368

OSPF Neighbor List Is Empty—Cause: Mismatched Area ID 368
  Debugs and Verification 368
  Solution 369

OSPF Neighbor List Is Empty—Cause: Mismatched Stub/Transit/NSSA Area Options 370
  Debugs and Verification 371
  Solution 371

OSPF Neighbor List Is Empty—Cause: OSPF Adjacency Over Secondary IP Address 372
  Debugs and Verification 373
  Solution 374

OSPF Neighbor List Is Empty—Cause: OSPF Adjacency over Asynchronous Interface 375
  Debugs and Verification 376
  Solution 377

OSPF Neighbor List Is Empty—Cause: No Network Type or Neighbor Defined over NBMA 377
  Debugs and Verification 378
  Solution 379

OSPF Neighbor List Is Empty—Cause: Frame Relay/Dialer Interface Missing the broadcast Keyword on Both Sides 380
  Debugs and Verification 381
  Solution 382

Problem: OSPF Neighbor Stuck in ATTEMPT 383
OSPF Neighbor Stuck in ATTEMPT—Cause: Misconfigured neighbor
  Statement 384
  Debugs and Verification 384
  Solution 385

OSPF Neighbor Stuck in ATTEMPT—Cause: Unicast Connectivity Is Broken on NBMA 385
  Debugs and Verification 386
  Solution 386

Problem: OSPF Neighbor Stuck in INIT 387
  OSPF Neighbor Stuck in INIT—Cause: Access List on One Side Is Blocking OSPF Hellos 387
    Debugs and Verification 388
    Solution 389
  OSPF Neighbor Stuck in INIT—Cause: Multicast Capabilities Are Broken on One Side (6500 Switch Problem) 389
    Debugs and Verification 390
    Solution 390
  OSPF Neighbor Stuck in INIT—Cause: Authentication Is Enabled Only on One Side 391
    Debugs and Verification 391
    Solution 392
  OSPF Neighbor Stuck in INIT—Cause: The frame-relay map/dialer-map Statement on One Side Is Missing the broadcast Keyword 393
    Debugs and Verification 394
    Solution 395
  OSPF Neighbor Stuck in INIT—Cause: Hellos Are Getting Lost on One Side at Layer 2 396
    Debugs and Verification 396
    Solution 397

Problem: OSPF Neighbor Stuck in 2-WAY—Cause: Priority 0 Is Configured on All Routers 398
  Debugs and Verification 400
  Solution 400

Problem: OSPF Neighbor Stuck in EXSTART/EXCHANGE 401
  OSPF Neighbor Stuck in EXSTART/EXCHANGE—Cause: Mismatched Interface MTU 401
    Debugs and Verification 402
    Solutions 403
  OSPF Neighbor Stuck in EXSTART/EXCHANGE—Cause: Duplicate Router IDs on Neighbors 404
    Debugs and Verification 405
    Solution 406
OSPF Neighbor Stuck in EXSTART/EXCHANGE—Cause: Can’t Ping Across with More Than Certain MTU Size 406
  Debugs and Verification 408
  Solution 408
OSPF Neighbor Stuck in EXSTART/EXCHANGE—Cause: Unicast Connectivity Is Broken 409
  Debugs and Verification 410
  Solutions 410
OSPF Neighbor Stuck in EXSTART/EXCHANGE—Cause: Network Type Is Point-to-Point Between PRI and BRI/Dialer 414
  Debugs and Verification 415
  Solution 416

Problem: OSPF Neighbor Stuck in LOADING 417
OSPF Neighbor Stuck in LOADING—Cause: Mismatched MTU Size 418
  Debugs and Verification 418
  Solution 419
OSPF Neighbor Stuck in LOADING—Cause: Link-State Request Packet Is Corrupted 420
  Debugs and Verification 421
  Solution 422

Troubleshooting OSPF Route Advertisement 422

Problem: OSPF Neighbor Is Not Advertising Routes 422
OSPF Neighbor Is Not Advertising Routes—Cause: OSPF Not Enabled on the Interface That Is Supposed to Be Advertised 423
  Debugs and Verification 424
  Solution 425
OSPF Neighbor Is Not Advertising Routes—Cause: Advertising Interface Is Down 426
  Debugs and Verification 427
  Solution 428
OSPF Neighbor Is Not Advertising Routes—Cause: Secondary Interface Is in a Different Area Than the Primary Interface 429
  Debugs and Verification 430
  Solution 431

Problem: OSPF Neighbor (ABR) Not Advertising the Summary Route 432
OSPF Neighbor (ABR) Not Advertising the Summary Route—Cause: Area Is Configured as Totally Stubby Area 432
  Debugs and Verification 433
  Solution 434
OSPF Neighbor (ABR) Not Advertising the Summary Route—Cause: ABR Is Not Connected to Area 0 434
  Debugs and Verification 435
  Solution 436
OSPF Neighbor (ABR) Not Advertising the Summary Route—Cause:
Discontiguous Area 0 437
  Debugs and Verification 438
  Solution 439

Problem: OSPF Neighbor Is Not Advertising External Routes 441
OSPF Neighbor Is Not Advertising External Routes—Cause: Area Is Configured as
a Stub Area or NSSA 441
  Debugs and Verification 442
  Solution 442
OSPF Neighbor Is Not Advertising External Routes—Cause: NSSA ABR Not
  Translating Type 7 LSAs into Type 5 LSAs 444
  Debugs and Verification 445
  Solution 449

Problem: OSPF Neighbor Not Advertising Default Routes 450
OSPF Neighbor Not Advertising Default Routes—Cause: Missing default-
  information originate Commands 451
  Debugs and Verification 452
  Solution 454
OSPF Neighbor Not Advertising Default Routes—Cause: Default Route Missing
  from the Neighbor’s Routing Table 455
  Debugs and Verification 455
  Solution 456
OSPF Neighbor Not Advertising Default Routes—Cause: Neighbor Trying to
  Inject a Default into a Stub Area 458
  Debugs and Verification 459
  Solution 459
OSPF Neighbor Not Advertising Default Routes—Cause: NSSA ABR/ASBR Not
  Originating Type 7 Default 460
  Debugs and Verification 462
  Solution 462

Troubleshooting OSPF Route Installation 463
Problem: OSPF Not Installing Any Routes in the Routing Table 463
  OSPF Not Installing Any Routes in the Routing Table—Cause: Network Type
    Mismatch 464
    Debugs and Verification 464
    Solution 466
OSPF Not Installing Any Routes in the Routing Table—Cause: IP Addresses Are
  Flipped in Dual Serial-Connected Routers 467
    Debugs and Verification 468
    Solution 469
OSPF Not Installing Any Routes in the Routing Table—Cause: One Side Is a Numbered and the Other Side Is an Unnumbered Point-to-Point Link 469
  Debugs and Verification 471
  Solution 472

OSPF Not Installing Any Routes in the Routing Table—Cause: Distribute List Is Blocking the Route Installation 473
  Debugs and Verification 474
  Solution 474

OSPF Not Installing Any Routes in the Routing Table—Cause: Broken PVC in a Fully Meshed Frame Relay Network with Broadcast Network Type 475
  Debugs and Verification 476
  Solution 478

Problem: OSPF Not Installing External Routes in the Routing Table 479
  OSPF Not Installing External Routes in the Routing Table—Cause: Forwarding Address Is Not Known Through the Intra-Area or Interarea Route 480
  Debugs and Verification 481
  Solution 483

OSPF Not Installing External Routes in the Routing Table—Cause: ABR Not Generating Type 4 Summary LSA 484
  Debugs and Verification 486
  Solution 486

Troubleshooting Redistribution Problems in OSPF 488

Problem: OSPF Neighbor Is Not Advertising External Routes 488
  OSPF Neighbor Is Not Advertising External Routes—Cause: Subnets Keyword Missing from the ASBR Configuration 489
  Debugs and Verification 490
  Solution 490

OSPF Neighbor Is Not Advertising External Routes—Cause: distribute-list out Is Blocking the Routes 491
  Debugs and Verification 492
  Solution 493

Troubleshooting Route Summarization in OSPF 494

Problem: Router Is Not Summarizing Interarea Routes—Cause: area range Command Is Not Configured on ABR 495
  Debugs and Verification 496
  Solution 496

Problem: Router Is Not Summarizing External Routes—Cause: summary-address Command Is Not Configured on ASBR 497
  Debugs and Verification 498
  Solution 499
Troubleshooting CPUHOG Problems  499

Problem: CPUHOG Messages During Adjacency Formation—Cause: Router Is Not Running Packet-Pacing Code  500
  Debugs and Verification  501
  Solution  501

  Debugs and Verification  502
  Solution  502

Troubleshooting Dial-on-Demand Routing Issues in OSPF  503

Problem: OSPF Hellos Are Bringing Up the Link—Cause: OSPF Hellos Are Permitted as Interesting Traffic  503
  Debugs and Verification  504
  Solution  505

Problem: Demand Circuit Keeps Bringing Up the Link  505
  Demand Circuit Keeps Bringing Up the Link—Cause: A Link Flap in the Network  506
    Debugs and Verification  507
    Solution  508
  Demand Circuit Keeps Bringing Up the Link—Cause: Network Type Defined as Broadcast  508
    Debugs and Verification  509
    Solution  510
  Demand Circuit Keeps Bringing Up the Link—Cause: PPP Host Routes Are Getting Redistributed into the OSPF Database  511
    Debugs and Verification  512
    Solution  513
  Demand Circuit Keeps Bringing Up the Link—Cause: One of the Routers Is Not Demand Circuit–Capable  514
    Debugs and Verification  515
    Solution  516

Troubleshooting SPF Calculation and Route Flapping  517

SPF Running Constantly—Cause: Interface Flap Within the Network  518
  Debugs and Verification  519
  Solution  520

SPF Running Constantly—Cause: Neighbor Flap Within the Network  520
  Debugs and Verification  522
  Solution  523
SPF Running Constantly—Cause: Duplicate Router ID 524
  Debugs and Verification 525
  Solution 527

Common OSPF Error Messages 528

“Unknown routing protocol” Error Message 528

OSPF: “Could not allocate router id” Error Message 529

“%OSPF-4-BADLSATYPE: Invalid Isa: Bad LSA type” Type 6 Error Message 529

“OSPF-4-ERRRCV” Error Message 529
  Mismatched Area ID 529
  Bad Checksum 530
  OSPF Not Enabled on the Receiving Interface 531

Chapter 10 Understanding Intermediate System-to-Intermediate System (IS-IS) 533

IS-IS Protocol Overview 533
  IS-IS Routing Protocol 535

IS-IS Protocol Concepts 535
  IS-IS Nodes, Links, and Areas 536
  Adjacencies 537
    ES-IS Adjacencies 538
    IS-IS Adjacencies 538
  Hierarchical Routing 541
  IS-IS Packets 542
    Generic IS-IS Packet Format 543
  IS-IS Metrics 545
  IS-IS Authentication 548
  ISO CLNP Addressing 548
    NSAP Format 549
    NSAP Examples 550
    Guidelines for Defining NSAP Addresses 551

IS-IS Link-State Database 552
  Overview of the IS-IS Link-State Database 552
  Flooding and Database Synchronization 555
  Shortest Path First (SPF) Algorithm and IS-IS Route Calculation 558

Configuring IS-IS for IP Routing 559
  Configuring IS-IS on Point-to-Point Serial Links 559
    show clns protocol Command 562
    show clns neighbors detail Command 563
    show clns interface Command 564
    show isis topology Command 565
    show isis database Command 565
Chapter 11  Troubleshooting IS-IS  585

Troubleshooting IS-IS Adjacency Problems  587
Problem 1: Some or All of the Adjacencies Are Not Coming Up  590
  Step 1: Checking for Link Failures  591
  Step 2: Verifying Basic Configuration  593
  Step 3: Checking for Mismatched Level 1 and Level 2 Interfaces  593
  Step 4: Checking for Area Misconfiguration  594
  Step 5: Checking for Misconfigured IP Subnets  595
  Step 6: Check for Duplicate System IDs  596
Problem 2: Adjacency in INIT State  596
  Mismatched MTU  600
  IS-IS Hello Padding  602
  Misconfigured Authentication  604
Problem 3: Only ES-IS Adjacency Instead of IS-IS Adjacency Formed  605

Troubleshooting IS-IS Routing Update Problems  606
Route Advertisement Problems  607
  Local Routes Not Being Advertised to Remote  609
  Solution Summary  611
Route Redistribution and Level 2–to–Level 1 Route-Leaking Problems  611
Route-Flapping Problem  612
  Solution Summary  616

IS-IS Errors  616
CLNS ping and traceroute  617
Case Study: ISDN Configuration Problem  619
IS-IS Troubleshooting Command Summary  622
Summary  623
Chapter 12  Understanding Protocol Independent Multicast (PIM)  625

Fundamentals of IGMP Version 1, IGMP Version 2, and Reverse Path Forwarding  626
  IGMP Version 1  626
  IGMP Version 2  627
  Multicast Forwarding (Reverse Path Forwarding)  628

PIM Dense Mode  630

PIM Sparse Mode  632

IGMP and PIM Packet Format  635
  IGMP Packet Format  635
  PIM Packet/Message Formats  636

Summary  640

Review Questions  641

Chapter 13  Troubleshooting PIM  643

Troubleshooting IGMP Joins  643
  Solution to IGMP Join Problem  645

Troubleshooting PIM Dense Mode  646
  Solution to PIM Dense Mode Problem  650

Troubleshooting PIM Sparse Mode  651
  Solution to PIM Sparse Mode Problem  656

Summary  656

Chapter 14  Understanding Border Gateway Protocol Version 4 (BGP-4)  659

BGP-4 Protocol Specification and Functionality  662

Neighbor Relationships  663
  External BGP Neighbor Relationships  665
  Internal BGP Neighbor Relationships  667

Advertising Routes  668
  Synchronization Rule  671

Receiving Routes  672

Policy Control  672
  Policy Control Using BGP Attributes  674
    LOCAL_PREF Attribute  675
    MULTI_EXIT_DISC (MED) Attribute  677
    AS_PATH Attribute  682
    NEXT_HOP Attribute  685
    ORIGIN Attribute  685
Directly Connected External BGP Neighbors Not Coming Up—Cause: Layer 2 Is Down, Preventing Communication with Directly Connected BGP Neighbor 729
  Debugs and Verification 729
  Solution 730
Directly Connected External BGP Neighbors Not Coming Up—Cause: Incorrect Neighbor IP Address in BGP Configuration 731
  Debugs and Verification 731
  Solution 732

Problem: Nondirectly Connected External BGP Neighbors Not Coming Up 732
Nondirectly Connected External BGP Neighbors Not Coming Up—Cause: Route to the Nondirectly Connected Peer Address Is Missing from the Routing Table 733
  Debugs and Verification 734
  Solution 736
Nondirectly Connected External BGP Neighbors Not Coming Up—Cause: ebgp-multihop Command Is Missing in BGP Configuration 736
  Debugs and Verification 737
  Solution 738
  Debugs and Verification 739
  Solution 741

Problem: Internal BGP Neighbors Not Coming Up 741

Problem: BGP Neighbors (External and Internal) Not Coming Up—Cause: Interface Access List Blocking BGP Packets 741
  Debugs and Verification 742
  Solution 742

Troubleshooting BGP Route Advertisement/Origination and Receiving 743

Problem: BGP Route Not Getting Originated 743
  BGP Route Not Getting Originated—Cause: IP Routing Table Does Not Have a Matching Route 744
    Debugs and Verification 744
    Solution 746
  BGP Route Not Getting Originated—Cause: Configuration Error 746
    Debugs and Verification 746
    Solution 749
  BGP Route Not Getting Originated—Cause: BGP Is Autosummarizing to Classful/Network Boundary 749
    Debugs and Verification 750
    Solution 751
Problem in Propagating/Originating BGP Route to IBGP/EBGP Neighbors—Cause:
Misconfigured Filters 752
Debugs and Verification 753
Solution 754

Problem in Propagating BGP Route to IBGP Neighbor but Not to EBGP Neighbor—
Cause: BGP Route Was from Another IBGP Speaker 754
Debugs and Verification 755
Solution 757
   IBGP Full Mesh 757
   Designing a Route-Reflector Model 757
   Designing a Confederation Model 758

Problem in Propagating IBGP Route to IBGP/EBGP Neighbor—Cause: IBGP Route
Was Not Synchronized 761
Debugs and Verification 762
Solution 762

Troubleshooting BGP Route Not Installing in Routing Table 762

Problem: IBGP-Learned Route Not Getting Installed in IP Routing Table 763
   IBGP-Learned Route Not Getting Installed in IP Routing Table—Cause: IBGP
   Routes Are Not Synchronized 763
   Debugs and Verification 764
   Solution 765

IBGP-Learned Route Not Getting Installed in IP Routing Table—Cause: IBGP Next
Hop Not Reachable 766
Debugs and Verification 768
Solution 769
   Announce the EBGP Next Hop Through an IGP Using a Static Route or
   Redistribution 769
   Change the Next Hop to an Internal Peering Address 770

Problem: EBGP-Learned Route Not Getting Installed in IP Routing Table 771
   EBGP-Learned Route Not Getting Installed in IP Routing Table—Cause: BGP
   Routes Are Dampered 771
   Debugs and Verification 772
   Solution 774

EBGP-Learned Route Not Getting Installed in IP Routing Table—Cause: BGP
Next Hop Not Reachable in Case of Multihop EBGP 774
   Debugs and Verification 775
   Solution 777

EBGP-Learned Route Not Getting Installed in the Routing Table—Cause:
Multiexit Discriminator (MED) Value Is Infinite 777
   Debugs and Verification 778
Troubleshooting BGP Route-Reflection Issues 778

Problem: Configuration Mistakes—Cause: Failed to Configure IBGP Neighbor as a Route-Reflector Client 779
  Debugs and Verification 779
  Solution 780

Problem: Route-Reflector Client Stores an Extra BGP Update—Cause: Client-to-Client Reflection 780
  Debugs and Verification 782
  Solution 782

Problem: Convergence Time Improvement for RR and Clients—Cause: Use of Peer Groups 783
  Debugs and Verification 784
  Solution 785

Problem: Loss of Redundancy Between Route Reflectors and Route-Reflector Client—Cause: Cluster List Check in RR Drops Redundant Route from Other RR 785
  Debugs and Verification 787
  Solution 788

Troubleshooting Outbound IP Traffic Flow Issues Because of BGP Policies 790

Problem: Multiple Exit Points Exist but Traffic Goes Out Through One or Few Exit Routers—Cause: BGP Policy Definition Causes Traffic to Exit from One Place 791
  Solution 793

Problem: Traffic Takes a Different Interface from What Shows in Routing Table—Cause: Next Hop of the Route Is Reachable Through Another Path 795
  Debugs and Verification 797
  Solution 798

Problem: Multiple BGP Connections to the Same BGP Neighbor AS, but Traffic Goes Out Through Only One Connection—Cause: BGP Neighbor Is Influencing Outbound Traffic by Sending MED or Prepended AS_PATH 798
  Solution 800
    Request AS 110 to Send the Proper MED for Each Prefix 800
    Don’t Accept MED from AS 110 801
    Manually Change LOCAL_PREFERENCE for P1, P2, and P3 at All the Exit Points X, Y, and Z 801

Problem: Asymmetrical Routing Occurs and Causes a Problem Especially When NAT and Time-Sensitive Applications Are Used—Cause: Outbound and Inbound Advertisement 802
  Debugs and Verification 803
  Solution 804

Troubleshooting Load-Balancing Scenarios in Small BGP Networks 806
Problem: Load Balancing and Managing Outbound Traffic from a Single Router When Dualhomed to Same ISP—Cause: BGP Installs Only One Best Path in the Routing Table 806

Solution 808

Problem: Load Balancing and Managing Outbound Traffic in an IBGP Network—Cause: By Default, IBGP in Cisco IOS Software Allows Only a Single Path to Get Installed in the Routing Table Even Though Multiple Equal BGP Paths Exist 809

Solution 810

Troubleshooting Inbound IP Traffic Flow Issues Because of BGP Policies 812

Problem: Multiple Connections Exist to an AS, but All the Traffic Comes in Through One BGP Neighbor, X, in the same AS—Cause: Either BGP Neighbor at X Has a BGP Policy Configured to Make Itself Preferred over the Other Peering Points, or the Networks Are Advertised to Attract Traffic from Only X 813

Solution 818

Problem: Multiple Connections Exist to Several BGP Neighbors, but Most of the Traffic from Internet to 100.100.100.0/24 Always Comes in Through One BGP Neighbor in AS 110—Cause: Route Advertisements for 100.100.100.0/24 in AS 109 Attract Internet Traffic Through That BGP Neighbor in AS 110 819

Solution 819

Troubleshooting BGP Best-Path Calculation Issues 820

Problem: Path with Lowest RID Is Not Chosen as Best 821

Solution 823

Problem: Lowest MED Not Selected as Best Path 824

Solution 826

Troubleshooting BGP Filtering 828

Problem: Standard Access List Fails to Capture Subnets 828

Solution 830

Problem: Extended Access Lists Fails to Capture the Correct Masked Route 831

Solution 833

Extended Access List Solution 833
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 provide 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.
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
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

**RIP Routes Not in the Routing Table**

- Is RIP enabled on the interface?  
  - Yes  
  - No

- Is the interface of the receiving router up/up?  
  - Yes  
  - No

- Is the `distribute-list` in blocking the routes?  
  - Yes  
  - No

- Is the access list blocking the RIP source address?  
  - Yes  
  - No

- Is the access list blocking the RIP broadcast?  
  - Yes  
  - No

- Is the RIP version compatible with the sender?  
  - Yes  
  - No

- Is there an authentication mismatch between sender and receiver?  
  - Yes  
  - No

- Is this a discontiguous subnet?  
  - Yes  
  - No

- Is the RIP update coming from a valid source?  
  - Yes  
  - No

- Is Layer 2 media propagating RIP broadcast/multicast?  
  - Yes  
  - No

- Is an offset list configured on the sender or receiver?  
  - Yes  
  - No

- Is the network more than 15 hops away?  
  - Yes  
  - No

Go to next problem flowchart.
Troubleshooting RIP Routes Installation

### RIP Is Not Installing All Possible Equal Paths

- Are there more than four possible paths? [Not sure] [Go to page 83.]
  - No [Go to next problem flowchart.]

### Troubleshooting RIP Route Advertisement

### Sender Is Not Advertising RIP Routes

- Is RIP enabled on the interface? [Not sure] [Go to page 87.]
  - Yes [Go to next problem flowchart.]
    - Is the outgoing interface up/up? [Not sure] [Go to page 89.]
      - Yes [Go to next problem flowchart.]
        - Is distribute-list out blocking the routes? [Not sure] [Go to page 91.]
          - No [Go to next problem flowchart.]
        - Is the advertised network interface up/up? [Not sure] [Go to page 93.]
          - Yes [Go to next problem flowchart.]
            - Is the outgoing interface defined as passive? [Not sure] [Go to page 95.]
              - No [Go to next problem flowchart.]
        - Is the multicast capability broken? [Not sure] [Go to page 96.]
          - No [Go to next problem flowchart.]
        - Is the neighbor statement configured properly? [Not sure] [Go to page 99.]
          - Yes [Go to next problem flowchart.]
            - Is the advertised subnet using VLSM? [Not sure] [Go to page 100.]
              - No [Go to next problem flowchart.]
        - Is split horizon enabled on the interface? [Not sure] [Go to page 102.]
          - No [Go to next problem flowchart.]
Troubleshooting RIP Route Advertisement

Subnetted Routes Missing from the Routing Table

Is the autosummarization feature enabled?
- No → Go to next problem flowchart.
- Not sure → Go to page 106.

Troubleshooting Route Summarization in RIP

RIP-2 Routing Table Is Huge

Is autosummarization turned off?
- No → Go to next problem flowchart.
- Not sure → Go to page 109.

Is the `ip summary-address` command configured?
- Yes → Go to next problem flowchart.
- Not sure → Go to page 111.

Troubleshooting RIP Redistribution Problems

Redistributed RIP Routes Are Not in the Routing Table of R2

Is the default metric defined on the redistribution router?
- No → Go to next problem flowchart.
- Not sure → Go to page 113.
**Troubleshooting Dial-on-Demand Routing Issues in RIP**

**RIP Updates Are Keeping the ISDN Link Up**

Are RIP broadcasts permitted as interesting traffic?  
Not sure  
Go to page 117.

No  
Go to next problem flowchart.

**RIP Updates Are Not Going Across the Dialer Interface**

Is the `broadcast` keyword missing from the `dialer map` statement?  
Not sure  
Go to page 120.

No  
Go to next problem flowchart.

**Troubleshooting Route Flapping Problems in RIP**

**RIP Routes Are Flapping**

Are there a large number of packet drops being reported by router interfaces in the network?  
Not sure  
Go to page 122.

No  
End of chapter problems.
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**

```
131.108.2.0/24  131.108.1.0/24  131.108.3.0/24
           .1                    .2
Router 1   E0                  E0
           131.108.0.0/24
```

**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.
When the `network` statement is configured, it does two things:

- Enables RIP on the interface and activates the capability to send and receive RIP updates
- Advertises that network in a RIP update packet

If the `network` statement under `router rip` command is not configured or misconfigured, it can cause this problem.

Figure 3-2 shows the flowchart to follow to solve this problem based on this cause.

---

**Debugs and Verification**

Example 3-2 shows the configuration for Router R2 (as illustrated in Figure 3-1). The loopback interface is used in this example and many other examples throughout the chapter. If the loopback interface is replaced with any other interface, it will not change the meaning. We suggest that you treat the loopback as any interface that is up and functional and that has a valid IP address.

**Example 3-2  Configuration for Router R2 from Figure 3-1**

```plaintext
interface Loopback0
 ip address 131.108.3.2 255.255.255.0
!
interface Ethernet0
 ip address 131.108.1.2 255.255.255.0
!
router rip
 network 131.107.0.0
!
```

Refer back to Figure 3-1 and compare it to the configuration for R2 in Example 3-2. You notice that network 131.108.0.0 is missing from R2’s configurations.
Example 3-3 shows the output of the `show ip protocols` command on R2. This output shows that the routing information source is also not displaying 131.108.1.1 as a gateway.

Example 3-3  *show ip protocols*  *Missing Gateway Information for Routing Information Source*

```
R2#show ip protocols
Routing Protocol is "rip"
    Sending updates every 30 seconds, next due in 11 seconds
    Invalid after 180 seconds, hold down 180, flushed after 240
    Outgoing update filter list for all interfaces is
    Incoming update filter list for all interfaces is
    Redistributing: rip
    Default version control: send version 1, receive any version
    Automatic network summarization is in effect
    Routing for Networks:
        131.107.0.0
    Routing Information Sources:
        Gateway         Distance      Last Update
        Distance: (default is 120)
```

**Debug Commands**

Example 3-4 shows the `debug ip rip` output. In this debug, R2 is ignoring the RIP updates coming from R1 because RIP is not enabled on Ethernet 0. This is because of the lack of a `network` statement for 131.108.0.0 under `router rip` in the router configuration mode.

Example 3-4  *debug ip rip*  *Command Output Displays That RIP Updates from Router R1 Are Being Ignored*

```
R2#debug ip rip
RIP protocol debugging is on

R2#RIP: ignored v1 packet from 131.108.1.1 (not enabled on Ethernet0)
```

**Solution**

Because the `network` statement is missing on Router 2, as shown in Example 3-2, it ignores RIP updates arriving on its Ethernet 0 interface, as seen in the debug output in Example 3-4. This problem can also happen if incorrect `network` statements are configured. Take a Class C address, for example. Instead of configuring 209.1.1.0, you configure 209.1.0.0, assuming that 0 will cover anything in the third octet. RIP-1 is a classful protocol, and it assumes the classful `network` statements. If a `cidr` statement is configured instead, RIP will not function properly.

To correct this problem, you must add the `network` statement in the configurations.

Example 3-5  *New Configuration of R2 That Solves the Problem*

```
interface Loopback0
 ip address 131.108.3.2 255.255.255.0
!
interface Ethernet0
 ip address 131.108.1.2 255.255.255.0
```

*continues*
Chapter 3: Troubleshooting RIP

Example 3-6 shows the output of `show ip protocols` on R2. This output displays the gateway information now.

Example 3-6  show ip protocols Showing Gateway Set to the R1’s Interface IP Address

Example 3-7 shows the output of `show ip route`, which shows that Router R2 is learning the RIP route after the configuration change.

Example 3-7  show ip route Displays the Route Being Learned After Fixing the Problem

RIP Routes Not in the Routing Table—Cause: Layer 1/2 Is Down

One cause for routes not in the routing table is Layers 1 or 2 being down. If Layers 1 or 2 are down, it’s not a RIP problem. The following is a list of the most common things to check if the interface or line protocol is down:

- Unplugged cable
- Loose cable
Problem: RIP Routes Not in the Routing Table

- Bad cable
- Bad transceiver
- Bad port
- Bad interface card
- Layer 2 problem at telco, in case of a WAN link
- Missing `clock` statement, in case of back-to-back serial connection

Figure 3-3 shows the flowchart to follow to solve this problem based on this cause.

**Figure 3-3  Flowchart to Solve Why RIP Routes Don’t Show Up in a Routing Table**

Debugs and Verification

Example 3-8 shows that the Ethernet interface’s line protocol is down, indicating that something is wrong at Layer 1 or Layer 2.

**Example 3-8  show interface output Displays That the Line Protocol Is Down**

<table>
<thead>
<tr>
<th>Command</th>
<th>Output</th>
</tr>
</thead>
</table>
| `R2#show interface ethernet 0` | Ethernet0 is up, line protocol is down  
Hardware is Lance, address is 0000.0c70.d41e (bia 0000.0c70.d41e)  
Internet address is 131.108.1.2/24 |

Debugs

Example 3-9 shows the output of `debug ip rip`. In this debug, R2 is not sending or receiving any RIP updates because Layer 2 is down.

**Example 3-9  debug ip rip Command Output Shows Nothing Is Being Sent**

<table>
<thead>
<tr>
<th>Command</th>
<th>Output</th>
</tr>
</thead>
</table>
| `R2#debug ip rip` | RIP protocol debugging is on  
R2# |
Solution

RIP runs above Layer 2. RIP cannot send or receive any routes if Layer 2 is down.

The Layer 2 problem must be fixed. Sometimes, the problem could be as simple as loose cables, or it could be as complex as bad hardware; in which case, the hardware must be replaced.

Example 3-10 shows the output of `show interface Ethernet 0` on R2 after the Layer 2 problem is fixed. The output shows that the line protocol is now up.

Example 3-10  `show interface Output After Fixing the Layer 1/2 Problem Shows the Interface Ethernet0 Is Now Up`

```
R2#show interface Ethernet0
Ethernet0 is up, line protocol is up
    Hardware is Lance, address is 0000.0c70.d41e (bia 0000.0c70.d41e)
    Internet address is 131.108.1.2/24
```

Example 3-11 shows the output of `show ip route`, which illustrates that the RIP route is being learned after fixing the Layer 1/2 problem.

Example 3-11  `Routing Table Entry After Fixing the Layer 1/2 Problem`

```
R2#show ip route 131.108.2.0
Routing entry for 131.108.2.0/24
    Known via "rip", distance 120, metric 1
    Redistributing via rip
    Last update from 131.108.1.1 on Ethernet0, 00:00:07 ago
    Routing Descriptor Blocks:
      * 131.108.1.1, from 131.108.1.1, 00:00:07 ago, via Ethernet0
        Route metric is 1, traffic share count is 1
```

RIP Routes Not in the Routing Table—Cause: distribute-list in Is Blocking the Route

A distribute list is a filtering mechanism for routing updates. The distribute list calls an access list and checks to see which networks are supposed to be permitted. If the access list doesn’t contain any network, the routing update will be automatically denied. A distribute list can be applied on either incoming routing updates or outgoing routing updates.

In this example, the `distribute-list in` is configured; however, the access list doesn’t contain the `permit` statement for 131.108.0.0, so R2 is not installing these routes in the routing table.

Figure 3-4 shows the flowchart to follow to solve this problem based on this cause.

Debugs and Verification

Example 3-12 shows the current configuration of Router R2. In this configuration, `access-list 1` is used to permit network 131.107.0.0; however, there is an implicit `deny` at the end of every access list, so 131.108.0.0 will also be denied. In the access list configuration, network 131.108.0.0 is not permitted, so the router is not installing any subnets of the 131.108.0.0 network.
Figure 3-4  Flowchart to Solve Why RIP Routes Don’t Show Up in a Routing Table

Example 3-12  \textit{R2’s Configuration Shows That Network 131.108.0.0 Is Being Blocked with an Implicit \textbf{deny} Under access-list 1}

```plaintext
interface Loopback0
  ip address 131.108.3.2 255.255.255.0
!
interface Ethernet0
  ip address 131.108.1.2 255.255.255.0
!
router rip
  network 131.108.0.0
!
distribute-list 1 in
!
access-list 1 permit 131.107.0.0 0.0.255.255
```

Solution

When a distribute list is used, you should always double-check your access list to make sure that the networks that are supposed to be permitted actually \textit{are} permitted in the access list. The access list in Example 3-12 permits only 131.107.0.0 and denies everything else because there is an implicit \textbf{deny} at the end of each access list. To fix this problem, permit 131.108.0.0 in \textbf{access-list 1}.

Example 3-13 shows the new configuration of Router R2 with the access list to permit 131.108.0.0.

Example 3-13  \textit{Correcting the Configuration on R2 to Fix the Problem}

```plaintext
interface Loopback0
  ip address 131.108.3.2 255.255.255.0
!
interface Ethernet0

continues
```
Example 3-13  Correcting the Configuration on R2 to Fix the Problem (Continued)

Example 3-14 shows that Router R2 is learning RIP routes after the configuration change.

Example 3-14  R2 Routing Table Is Learning the RIP Routes After the Correction

RIP Routes Not in the Routing Table—Cause: Access List Blocking RIP Source Address

Access lists are used to filter the traffic based on the source address. Extended access lists are used to filter the traffic based on the source or destination address, T-2. To filter the incoming and outgoing traffic, these access lists may be applied on the interface with this interface-level command:

```
ip access-group access-list number {in | out}
```

When the access list is applied in a RIP environment, always make sure that it doesn’t block the source address of the RIP update. In this example, R2 is not installing RIP routes in the routing table because access-list 1 is not permitting the source address of RIP updates from R1.

Figure 3-5 shows the flowchart to follow to solve the problem based on this cause.

Debugs and Verification

Example 3-15 shows the current configuration of router R2. The access list in R2 is not permitting the source address of RIP updates, that is, 131.108.1.1. In Figure 3-1, 131.108.1.1 is the source address of R1 RIP updates. Because there is an implicit deny at the end of each access list, 131.108.1.1 will be automatically denied.
**Figure 3-5** Flowchart to Solve Why RIP Routes Don’t Show Up in a Routing Table

[Flowchart]

RIP routes are not in the routing table of R2.

Is the access list blocking the RIP source address?  
Not sure

If the source address is not permitted in the input access list, RIP will not install any routes. Go to “Debugs and Verification” section.

---

**Example 3-15** access-list 1 Is Not Permitting the Source Address

R2#

```
interface Loopback0
ip address 131.108.3.2 255.255.255.0
!
interface Ethernet0
ip address 131.108.1.2 255.255.255.0
ip access-group 1 in
!
router rip
network 131.108.0.0
!
access-list 1 permit 131.107.0.0 0.0.255.255
```

---

**Debugs**

The output of `debug ip rip` in Example 3-16 shows that RIP is only sending the updates, not receiving anything, because the source address 131.108.1.1 is not permitted in the input access list of R2.

**Example 3-16** debug ip rip Output Reveals That R2 Is Not Receiving Any RIP Updates

R2#debug ip rip

```
RIP: sending v1 update to 255.255.255.255 via Ethernet0 (131.108.1.2)
RIP: build update entries
   subnet 131.108.3.0 metric 1
RIP: sending v1 update to 255.255.255.255 via Loopback0 (131.108.3.1)
RIP: build update entries
   subnet 131.108.1.0 metric 1
RIP: sending v1 update to 255.255.255.255 via Ethernet0 (131.108.1.2)
RIP: build update entries
   subnet 131.108.3.0 metric 1
```

continues
Solution

The standard access list specifies the source address. In this case, the source address is 131.108.1.1, which is the sending interface address of R1. This source address is not permitted in the standard access list of R2, so RIP routes will not get installed in the routing table of R2. To solve this problem, permit the source address in access list 1.

Example 3-17 shows the new configuration change to fix this problem.

Example 3-17  The Modified Access List Permits the Source Address

```
Example 3-17
R2#

interface Loopback0
ip address 131.108.3.2 255.255.255.0
!
interface Ethernet0
ip address 131.108.1.2 255.255.255.0
ip access-group 1 in
!
router rip
network 131.108.0.0
!
access-list 1 permit 131.107.0.0 0.0.255.255
access-list 1 permit 131.108.1.1 0.0.0.0
```

This problem can also happen when using extended access lists if the RIP source address is not permitted in the access list. This solution also can be used in the case of an extended access list. The idea here is to permit the source address of RIP update.

Example 3-18 shows the configuration with an extended access list.

Example 3-18  The Correct Extended Access List Configuration, if Used

```
Example 3-18
R2#

interface Loopback0
ip address 131.108.3.2 255.255.255.0
!
interface Ethernet0
ip address 131.108.1.2 255.255.255.0
ip access-group 100 in
!
router rip
network 131.108.0.0
!
access-list 100 permit ip 131.107.0.0 0.0.255.255 any
access-list 100 permit ip host 131.108.1.1 any
```

Example 3-19 shows the routing table of Router R2, which shows that it has learning RIP routes after the configuration change.
RIP Routes Not in the Routing Table—Cause: Access List Blocking RIP Broadcast or Multicast (in Case of RIP-2)

Access lists are used to filter certain types of packets. When using access lists on the interface inbound, always make sure that they are not blocking the RIP broadcast or UDP port 520, which is used by RIP-1 and RIP-2 (or the RIP multicast address, in cases of RIP-2).

If these addresses are not permitted in the access list that is applied on the interface inbound, RIP will not install any routes in the routing table learned on that interface.

Figure 3-6 shows the flowchart to follow to solve this problem based on this cause.

Debugs and Verification

Example 3-20 shows the current configuration of R2. In this configuration, RIP’s destination address of 255.255.255.255 is not being permitted. This will result in no RIP routes being installed in R2’s routing table. The RIP updates sent from R1 to the destination of 255.255.255.255 will be blocked by R2.
Example 3-20  
**R2 Configuration Does Not Permit RIP-1 Broadcast Addresses**

<table>
<thead>
<tr>
<th>Command</th>
</tr>
</thead>
<tbody>
<tr>
<td>R2#</td>
</tr>
<tr>
<td>interface Loopback0</td>
</tr>
<tr>
<td>ip address 131.108.3.2 255.255.255.0</td>
</tr>
<tr>
<td>interface Ethernet0</td>
</tr>
<tr>
<td>ip address 131.108.1.2 255.255.255.0</td>
</tr>
<tr>
<td>ip access-group 100 in</td>
</tr>
<tr>
<td>router rip</td>
</tr>
<tr>
<td>network 131.108.0.0</td>
</tr>
<tr>
<td>access-list 100 permit ip 131.107.0.0 0.0.255.255 any</td>
</tr>
<tr>
<td>access-list 100 permit ip host 131.108.1.1 host 131.108.1.2</td>
</tr>
</tbody>
</table>

**Solution**

RIP-1 broadcasts its routing updates on 255.255.255.255. This address must be permitted in the input access list of the receiving router so that it can receive the RIP updates.

Example 3-21 shows the new configuration for Router R2. *access-list 100* is modified so that it can permit the RIP broadcast address that was being blocked before.

Example 3-21  **Configuring Router R2’s Input Access List to Accept RIP-1 Broadcasts**

<table>
<thead>
<tr>
<th>Command</th>
</tr>
</thead>
<tbody>
<tr>
<td>interface Loopback0</td>
</tr>
<tr>
<td>ip address 131.108.3.2 255.255.255.0</td>
</tr>
<tr>
<td>interface Ethernet0</td>
</tr>
<tr>
<td>ip address 131.108.1.2 255.255.255.0</td>
</tr>
<tr>
<td>ip access-group 100 in</td>
</tr>
<tr>
<td>router rip</td>
</tr>
<tr>
<td>network 131.108.0.0</td>
</tr>
<tr>
<td>access-list 100 permit ip 131.107.0.0 0.0.255.255 any</td>
</tr>
<tr>
<td>access-list 100 permit ip host 131.108.1.1 host 131.108.1.2</td>
</tr>
<tr>
<td>access-list 100 permit ip host 131.108.1.1 host 255.255.255.255</td>
</tr>
</tbody>
</table>

In cases of RIP-2, the configuration will change slightly. The multicast address needs to be permitted instead of the broadcast address, as shown in Example 3-22.

Example 3-22  **Configuring Router R2’s Input Access List to Accept RIP-2 Multicast**

<table>
<thead>
<tr>
<th>Command</th>
</tr>
</thead>
<tbody>
<tr>
<td>interface Loopback0</td>
</tr>
<tr>
<td>ip address 131.108.3.2 255.255.255.0</td>
</tr>
<tr>
<td>interface Ethernet0</td>
</tr>
<tr>
<td>ip address 131.108.1.2 255.255.255.0</td>
</tr>
<tr>
<td>ip access-group 100 in</td>
</tr>
<tr>
<td>router rip</td>
</tr>
<tr>
<td>version 2</td>
</tr>
<tr>
<td>network 131.108.0.0</td>
</tr>
<tr>
<td>access-list 100 permit ip 131.107.0.0 0.0.255.255 any</td>
</tr>
<tr>
<td>access-list 100 permit ip host 131.108.1.1 host 131.108.1.2</td>
</tr>
<tr>
<td>access-list 100 permit ip host 131.108.1.1 host 224.0.0.9</td>
</tr>
</tbody>
</table>
Example 3-23 shows the routing table of R2 after correcting the problem.

**Example 3-23**  \textit{R2 Routing Table After Correcting the Access List Shows That the RIP Routes Are Being Learned}

```plaintext
R2# show ip route 131.108.2.0
Routing entry for 131.108.2.0/24
  Known via "rip", distance 120, metric 1
  Redistributing via rip
  Last update from 131.108.1.1 on Ethernet0, 00:00:07 ago
Routing Descriptor Blocks:
  * 131.108.1.1, from 131.108.1.1, 00:00:07 ago, via Ethernet0
    Route metric is 1, traffic share count is 1
```

**RIP Routes Not in the Routing Table—Cause: Incompatible RIP Version Type**

When RIP is configured on a router, it is run by default as Version 1, which means that all its interfaces will send and receive RIP-1 packets only. To run Version 2 of RIP, you must add the \textit{version 2} line under \texttt{router rip}. When a router running Version 1 receives a RIP update from a router running Version 2, it ignores the updates and does not install any routes in the routing table. For a router to accept a Version 2 packet, the interface must be configured to accept the RIP-2 updates.

Figure 3-7 shows the flowchart to follow to solve this problem based on this cause.

**Figure 3-7**  \textit{Flowchart to Solve Why RIP Routes Don’t Show Up in a Routing Table}

**Debugs and Verification**

Example 3-24 shows the configuration of Router R2. In this configuration, RIP is configured to send and receive Version 1 packets only.
Chapter 3: Troubleshooting RIP

Example 3-24  **R2 Configuration Shows That It Is Configured for RIP-1, Which Is the Default**

```
R2# interface Loopback0
    ip address 131.108.3.2 255.255.255.0
!
R2# interface Ethernet0
    ip address 131.108.1.2 255.255.255.0
!
R2# router rip
    network 131.108.0.0
```

Example 3-25 shows the output of the `debug ip rip` command. This command reveals that R2 is receiving a RIP packet from R1, which is configured to send Version 2 updates.

**Example 3-25  **debug ip rip** Command Output Shows the Version Incompatible Message on R2**

```
R2# debug ip rip
RIP protocol debugging is on
RIP: ignored v2 packet from 131.108.1.1 (illegal version)
```

Example 3-26 shows the output of the `show ip protocols` command, which indicates that the Ethernet0 interface is sending and receiving RIP-1 packets. This means that if a Version 2 packet is received on Ethernet 0 of R2, it will be ignored because the interface can send and receive only Version 1 packets.

**Example 3-26  **show ip protocols** Command Output Reveals the RIP Sends Out and Receives Only RIP Version 1 Packets on Ethernet0**

```
R2# show ip protocols
Routing Protocol is "rip"
    Sending updates every 30 seconds, next due in 9 seconds
    Invalid after 180 seconds, hold down 180, flushed after 240
    Outgoing update filter list for all interfaces is
    Incoming update filter list for all interfaces is
    Redistributing: rip
    Default version control: send version 1, receive version 1
    
    Interface  Send  Recv  Key-chain
    Ethernet0  1     1
    Loopback0  1     1

R2# show ip protocols
Routing for Networks:
    131.108.0.0
Routing Information Sources:
    Gateway  Distance  Last Update
    131.108.1.1  120  00:01:34
    Distance: (default is 120)
```

Example 3-27 shows the configuration of R1. This shows that sender R1 is configured to send Version 2 packets. The command `version 2` enables a router to send and accept only RIP-2 packets.
Example 3-27  *R1’s Configuration Reveals That It Is Configured for RIP Version 2 Packets*

<table>
<thead>
<tr>
<th>R1#</th>
</tr>
</thead>
<tbody>
<tr>
<td>router rip</td>
</tr>
<tr>
<td>version 2</td>
</tr>
<tr>
<td>network 131.108.0.0</td>
</tr>
</tbody>
</table>

Example 3-28  *show ip protocols Command Output Reveals That R1 Is Sending and Receiving Only RIP Version 2 Packets*

<table>
<thead>
<tr>
<th>R1#show ip protocols</th>
</tr>
</thead>
<tbody>
<tr>
<td>Routing Protocol is &quot;rip&quot;</td>
</tr>
<tr>
<td>Sending updates every 30 seconds, next due in 13 seconds</td>
</tr>
<tr>
<td>Invalid after 180 seconds, hold down 180, flushed after 240</td>
</tr>
<tr>
<td>Outgoing update filter list for all interfaces is</td>
</tr>
<tr>
<td>Incoming update filter list for all interfaces is</td>
</tr>
<tr>
<td>Redistributing: rip</td>
</tr>
<tr>
<td>Default version control: send version 2, receive version 2</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Interface</th>
<th>Send</th>
<th>Recv</th>
<th>Key-chain</th>
</tr>
</thead>
<tbody>
<tr>
<td>Ethernet0/1</td>
<td>2</td>
<td>2</td>
<td></td>
</tr>
<tr>
<td>Loopback1</td>
<td>2</td>
<td>2</td>
<td></td>
</tr>
</tbody>
</table>

Routing for Networks:
- 131.108.0.0

Routing Information Sources:
- Gateway: 131.108.1.2, Distance: 120, Last Update: 00:04:09

| Distance: (default is 120) |

Solution

If the receiver R2 is configured to receive only RIP Version 1 packets, it will ignore the RIP Version 2 updates. You must configure Router R1 on the sender’s side so that it will send both Version 1 and Version 2 packets. When R2 receives the Version 1 packet, it will install the routes in the routing table. R2 will ignore RIP-2 packets because it is configured for RIP-1.

Example 3-29 shows the new configuration for R1. In this configuration, the sender (R1’s Ethernet interface) is configured to send and receive both RIP-1 and RIP-2 packets.

Example 3-29  *New Configuration of R1 to Send and Receive Version 1 and Version 2 Packets*

<table>
<thead>
<tr>
<th>R1#</th>
</tr>
</thead>
<tbody>
<tr>
<td>interface Loopback0</td>
</tr>
<tr>
<td>ip address 131.108.2.1 255.255.255.0</td>
</tr>
<tr>
<td>!</td>
</tr>
<tr>
<td>interface Ethernet0</td>
</tr>
<tr>
<td>ip address 131.108.1.1 255.255.255.0</td>
</tr>
<tr>
<td>ip rip send version 1 2</td>
</tr>
<tr>
<td>ip rip receive version 1 2</td>
</tr>
</tbody>
</table>

continues
Chapter 3: Troubleshooting RIP

Example 3-29  New Configuration of R1 to Send and Receive Version 1 and Version 2 Packets (Continued)

```
! router rip
  version 2
  network 131.108.0.0
```

Example 3-30 shows the output of `show ip protocols`, which indicates that the Ethernet0 interface is sending and receiving Version 1 and Version 2 packets. The advantage of sending both Version 1 and Version 2 updates is that, if any devices on this Ethernet segment are running Version 1 only or Version 2 only, those devices will be capable of communicating with R1 on Ethernet.

Example 3-30  `show ip protocols` Command Output Reveals the RIP Version 1 and 2 Packets Being Sent and Received by R1’s Ethernet0 Interface

```
R1# show ip protocols
Routing Protocol is "rip"
  Sending updates every 30 seconds, next due in 4 seconds
  Invalid after 180 seconds, hold down 180, flushed after 240
  Outgoing update filter list for all interfaces is
  Incoming update filter list for all interfaces is
  Redistributing: rip
  Default version control: send version 2, receive version 2

+-------+--------+--------+-------+-------------------+-------+--------+--------+--------+        +-------+--------+--------+-------+----------+--------+--------+--------+        +-------+--------+--------+-------+----------+--------+--------+--------+        +-------+--------+--------+-------+----------+--------+--------+--------+        +-------+--------+--------+-------+----------+--------+--------+--------+        +-------+--------+--------+-------+----------+--------+--------+--------+        +-------+--------+--------+-------+----------+--------+--------+--------+
| Interface        | Send | Recv | Key-chain |
| Ethernet0        | 1    | 2    | 1 2       |
| Loopback0        | 2    | 2    |
| Routing for Networks: |
| 131.108.0.0 |
| Routing Information Sources: |
| Gateway | Distance | Last Update |
| 131.108.1.2 | 120 | 00:00:07 |

Distance: (default is 120)
```

Example 3-31 shows R2’s routing table after the configuration change.

Example 3-31  R2 Routing Table After R1 Is Configured to Send and Receive RIP-1 and RIP-2 Packets

```
R2# show ip route 131.108.2.0
Routing entry for 131.108.2.0/24
  Known via "rip", distance 120, metric 1
  Redistributing via rip
  Last update from 131.108.1.1 on Ethernet0, 00:00:07 ago
  Routing Descriptor Blocks:
  * 131.108.1.1, from 131.108.1.1, 00:00:07 ago, via Ethernet0
    Route metric is 1, traffic share count is 1
```

RIP Routes Not in the Routing Table—Cause: Mismatch Authentication Key (RIP-2)

One of the options in RIP-2 is that the RIP-2 updates can be authenticated for increased security. When authentication is used, a password must be configured on both sides. This
password is called the *authentication key*. If this key does not match with the key on the other side, the RIP-2 updates will be ignored on both sides.

Figure 3-8 shows the flowchart to follow to solve this problem based on this cause.

**Figure 3-8** Flowchart to Solve Why RIP Routes Don’t Show Up in a Routing Table

Debugs and Verification

Example 3-32 shows the configurations of routers R1 and R2 when this problem happens. In this configuration, a different RIP authentication key is configured on R1 and R2. The R2 Ethernet interface is configured with the key *cisco1*, whereas R1 is configured with the key *Cisco*. These two keys do not match, so they ignore each other’s update and routes will not be installed in the routing table.

**Example 3-32** Configurations for R1 and R2 Show That Different Authentication Keys Are Configured on Each Side

```
R2#
interface Loopback0
ip address 131.108.3.2 255.255.255.0
!
interface Ethernet0
ip address 131.108.1.2 255.255.255.0
ip rip authentication key-chain cisco1
!
routing rip
version 2
network 131.108.0.0
```

continues
Chapter 3: Troubleshooting RIP

Example 3-32  Configurations for R1 and R2 Show That Different Authentication Keys Are Configured on Each Side (Continued)

<table>
<thead>
<tr>
<th>R1#</th>
</tr>
</thead>
<tbody>
<tr>
<td>interface Loopback0</td>
</tr>
<tr>
<td>ip address 131.108.2.1 255.255.255.0</td>
</tr>
<tr>
<td>interface Ethernet0</td>
</tr>
<tr>
<td>ip address 131.108.1.1 255.255.255.0</td>
</tr>
<tr>
<td>ip rip authentication key-chain cisco</td>
</tr>
<tr>
<td>!</td>
</tr>
<tr>
<td>router rip</td>
</tr>
<tr>
<td>version 2</td>
</tr>
<tr>
<td>network 131.108.0.0</td>
</tr>
<tr>
<td>!</td>
</tr>
</tbody>
</table>

Example 3-33 shows the output from the `debug ip rip` command on R2 that indicates that R2 is receiving a RIP packet that has invalid authentication. This means that the authentication key between sender and receiver doesn’t match.

Example 3-33  debug ip rip Command Output Reveals Invalid Authentication for a RIP-2 Packet Received on R2

<table>
<thead>
<tr>
<th>R2#debug ip rip</th>
</tr>
</thead>
<tbody>
<tr>
<td>RIP protocol debugging is on</td>
</tr>
<tr>
<td>RIP: ignored v2 packet from 131.108.1.1 (invalid authentication)</td>
</tr>
</tbody>
</table>

Solution

When using authentication in RIP, make sure that the sender and the receiver are configured with the same authentication key. Sometimes, adding a `space` at the end of the key can cause the `invalid authentication` problem because a `space` will be taken as a literal key entry. As a result, this causes a problem that cannot be corrected just by looking at the configurations.

Debugs will show that there is a problem with the authentication key. To solve this problem, configure the same keys on both sender and receiver, or retype the authentication key, making sure that no `space` is being added at the end.

Example 3-34 shows the new configuration to correct this problem. The authentication key is reconfigured on Router R2 to match Router the key on R1.

Example 3-34  R2 Configuration with the Corrected Authentication Key

<table>
<thead>
<tr>
<th>R2#</th>
</tr>
</thead>
<tbody>
<tr>
<td>interface Loopback0</td>
</tr>
<tr>
<td>ip address 131.108.3.2 255.255.255.0</td>
</tr>
<tr>
<td>interface Ethernet0</td>
</tr>
<tr>
<td>ip address 131.108.1.2 255.255.255.0</td>
</tr>
<tr>
<td>ip rip authentication key-chain cisco</td>
</tr>
<tr>
<td>!</td>
</tr>
<tr>
<td>router rip</td>
</tr>
<tr>
<td>version 2</td>
</tr>
<tr>
<td>network 131.108.0.0</td>
</tr>
<tr>
<td>!</td>
</tr>
</tbody>
</table>
Example 3-35 shows the routing table of R2 after the configuration change.

**Example 3-35  R2 Routing Table After Reconfiguring the Authentication Key on R2**

```
R2#show ip route 131.108.2.0
Routing entry for 131.108.2.0/24
    Known via "rip", distance 120, metric 1
    Redistributing via rip
    Last update from 131.108.1.1 on Ethernet0, 00:00:07 ago
    Routing Descriptor Blocks:
    * 131.108.1.1, from 131.108.1.1, 00:00:07 ago, via Ethernet0
      Route metric is 1, traffic share count is 1
```

**RIP Routes Not in the Routing Table—Cause: Discontiguous Network**

When a major network is separated by another major network in the middle, this is called a discontiguous network. Chapter 2, “Understanding Routing Information Protocol (RIP),” provides a detailed explanation of why RIP does not support discontiguous networks. Enabling RIP with this topology causes problems.

Figure 3-9 shows an example of a discontiguous network that exists when a major network is separated by another major network.

**Figure 3-9  An Example of a Discontiguous Network**

![Figure 3-9 An Example of a Discontiguous Network]

Figure 3-10 shows the flowchart to follow to solve this problem based on this cause.

**Figure 3-10  Flowchart to Solve Why RIP Routes Don’t Show Up in a Routing Table**

```
RIP routes are not in the routing table of R2.

Is this a discontiguous subnet?  Not sure

No

If RIP receives a summarized route for a discontiguous network, it will not install it in the routing table. Go to “Debugs and Verification” section.

Go to next cause.
```
Chapter 3: Troubleshooting RIP

Debugs and Verification

Example 3-36 shows the configuration of Router R1 and Router R2. RIP is enabled on the Ethernet interfaces of R1 and R2 with the correct `network` statement.

Example 3-36  Configuration of R1 and R2 in a Discontiguous Network Environment

```
R2#
  interface Loopback0
  ip address 137.99.3.2 255.255.255.0
#
  interface Ethernet0
  ip address 131.108.1.2 255.255.255.0
#
  router rip
  network 131.108.0.0
  network 137.99.0.0
#
R1#
  interface Loopback0
  ip address 137.99.2.1 255.255.255.0
#
  interface Ethernet0
  ip address 131.108.1.1 255.255.255.0
#
  router rip
  network 131.108.0.0
  network 137.99.0.0
#
```

Example 3-37 shows the `debug ip rip` output for routers R1 and R2. Both debugs shows that the network 137.99.0.0 is being sent across.

Example 3-37  debug ip rip Output Showing That Both Routers Are Sending Summarized Major Network Addresses Across

```
R2#debug ip rip
RIP protocol debugging is on
RIP: received v1 update from 131.108.1.1 on Ethernet0
  137.99.0.0 in 1 hops
RIP: sending v1 update to 255.255.255.255 via Ethernet0 (131.108.1.2)
RIP: build update entries
  network 137.99.0.0 metric 1
R2#
R1#debug ip rip
RIP protocol debugging is on
R1#
  RIP: received v1 update from 131.108.1.2 on Ethernet0
    137.99.0.0 in 1 hops
  RIP: sending v1 update to 255.255.255.255 via Ethernet0 (131.108.1.1)
  RIP: build update entries
    network 137.99.0.0 metric 1
```

As a result, both routers will ignore the 137.99.0.0 update from each other. Because R1 and R2 are already connected to this major network, they will ignore the update.
Solution

RIP is not installing the route 137.99.0.0 in the routing table because RIP doesn’t support discontiguous networks, as discussed in the beginning of the chapter. Several solutions to this problem exist. The quick solution is to configure a static route to the more specific subnets of 137.99.0.0 on each router. The second solution is to enable Version 2 of RIP. Another solution is to replace RIP with another IP routing protocol, such as OSPF, IS-IS, EIGRP, and so on, that supports discontiguous networks.

Example 3-38 shows the configuration change that is required for both Router R1 and Router R2 to fix the problem. This configuration adds the static route for the discontiguous subnets. Because you cannot pass the subnet information across in case of discontiguous networks in RIP-1, the only solution is to patch it with static routes.

Example 3-38  Static Route Configuration Should Solve This Problem

<table>
<thead>
<tr>
<th>R1#</th>
<th>R2#</th>
</tr>
</thead>
</table>
| interface Loopback0  
ip address 137.99.2.1 255.255.255.0  |
| !  | !  |
| interface Ethernet0  
ip address 131.108.1.1 255.255.255.0  |
| !  | !  |
| router rip  

network 131.108.0.0  

network 137.99.0.0  |
| !  | !  |
| ip route 137.99.3.0 255.255.255.0 131.108.1.2  |

<table>
<thead>
<tr>
<th>R2#</th>
</tr>
</thead>
</table>
| interface Loopback0  
ip address 137.99.3.2 255.255.255.0  |
| !  |
| interface Ethernet0  
ip address 131.108.1.2 255.255.255.0  |
| !  |
| router rip  

network 131.108.0.0  

network 137.99.0.0  |
| !  | !  |
| ip route 137.99.2.0 255.255.255.0 131.108.1.1  |

Example 3-39 shows the alternate solution to fix this problem, in the case of RIP-2. The solution is to run RIP-2 with no auto-summary configured. With the no-auto summary command added, RIP-2 will not autosummarize when crossing a major network boundary. The specific subnet information will be sent across.

Example 3-39  Configuration That Works Under RIP-2 in a Discontiguous Network Environment

<table>
<thead>
<tr>
<th>router rip</th>
</tr>
</thead>
<tbody>
<tr>
<td>version 2</td>
</tr>
<tr>
<td>network 131.108.0.0</td>
</tr>
<tr>
<td>network 137.99.0.0</td>
</tr>
<tr>
<td>no auto-summary</td>
</tr>
</tbody>
</table>
Example 3-40 shows the routing table of R2 after fixing this problem.

**Example 3-40**  
*R2 Routing Table Shows That 137.99.2.0/24 Is Learned Through RIP-2 After Configuring the **no-auto summary** Command*

```
R2#show ip route 137.99.2.0
Routing entry for 137.99.2.0/24
    Known via "rip", distance 120, metric 1
    Redistributing via rip
    Last update from 131.108.1.1 on Ethernet0, 00:00:07 ago
    Routing Descriptor Blocks:
        * 131.108.1.1, from 131.108.1.1, 00:00:07 ago, via Ethernet0
          Route metric is 1, traffic share count is 1
```

**RIP Routes Not in the Routing Table—Cause: Invalid Source**

When RIP tells the routing table to install the route, it performs a source-validity check. If the source is not on the same subnet as the local interface, RIP ignores the update and does not install routes in the routing table coming from this source address.

Figure 3-11 shows the network diagram for invalid source problem.

**Figure 3-11**  
*Network Diagram for Invalid Route Source*

In Figure 3-11, Router 1’s Serial 0 interface is unnumbered to Loopback 0. Router 2’s serial interface is numbered. When Router 2 receives a RIP update from Router 1, it complains about the source validity because the source address is not on the same subnet as Router 2’s Serial 0 interface.

Figure 3-12 shows the flowchart to follow to solve this problem based on this cause.

**Debugs and Verification**

Example 3-41 shows the configuration of both Router R2 and Router R1. In this configuration, R1’s Serial 0 interface is unnumbered to Loopback 0. R2’s Serial 0 interface is numbered.
The debug ip rip output in Example 3-42 shows that R2 is ignoring the RIP update from R1 because of a source validity check. The RIP update coming from R1 is not on the same subnet, so R2 will not install any routes in the routing table.

Example 3-42 debug ip rip Message Shows That R2 Is Receiving RIP Updates from a Different Source Address Than Its Own Interface

R2#debug ip rip
RIP protocol debugging is on
RIP: ignored v1 update from bad source 131.108.2.1 on Serial0
R2#
Solution

When one side is numbered and the other side is unnumbered, this check must be turned off. This is usually the case in a dialup situation when remotes are dialing into an access router. The access router’s dialup interface is unnumbered, and all remote routers get an IP address assigned on their dialup interfaces.

Example 3-43 shows the new configuration change on Router R2 to fix this problem.

Example 3-43  Configuration of R2 to Turn Off the Source Validity Check

```bash
R2# interface Loopback0
ip address 131.108.3.2 255.255.255.0
!
interface Serial0
ip address 131.108.1.2 255.255.255.0
!
router rip
no validate-update-source
network 131.108.0.0
```

Example 3-44 shows that after changing the configurations of R2, the route gets installed in the routing table.

Example 3-44  R2 Routing Table After Turning Off Source Validity Check

```bash
R2# show ip route 131.108.2.0
Routing entry for 131.108.2.0/24
Known via "rip", distance 120, metric 1
Redistributing via rip
  Last update from 131.108.1.1 00:00:01 ago
Routing Descriptor Blocks:
* 131.108.1.1, from 131.108.1.1, 00:00:07 ago
    Route metric is 1, traffic share count is 1
```

RIP Routes Not in the Routing Table—Cause: Layer 2 Problem (Switch, Frame Relay, Other Layer 2 Media)

Sometimes, multicast/broadcast capability is broken at Layer 2, which further affects Layer 3 multicast. As a result, RIP fails to work properly. The Layer 3 broadcast/multicast is further converted into Layer 2 broadcast/multicast. If Layer 2 has problems in handling Layer 2 multicast/broadcast, the RIP updates will not be propagated. The debugs show that broadcast or multicast is being originated at one end but is not getting across.

Figure 3-13 shows the network diagram for Frame Relay problems while running RIP.

In Figure 3-13, Router 1 and Router 2 are connected through any Layer 2 media—for example, Frame Relay, X.25, Ethernet, FDDI, and so on.

Figure 3-14 shows the flowchart to follow to solve this problem based on this cause.
Problem: RIP Routes Not in the Routing Table

Figure 3-13  *Two Routers Running RIP in a Frame Relay Environment*

![Diagram of two routers running RIP](image)

Figure 3-14  *Flowchart to Solve Why RIP Routes Don’t Show Up in a Routing Table*

- RIP routes are not in the routing table of R2.
- Is Layer 2 media propagating RIP broadcast/multicast?
  - Yes: Go to next cause.
  - Not sure: RIP-1 sends an update on broadcast address 255.255.255.255, and RIP-2 sends an update on multicast address 224.0.0.9. These two addresses must be permitted through Layer 2 media. Go to “Debugs and Verification” section.

Debugs and Verification

Example 3-45 shows the output of the `debug ip rip` command, which shows that R1 is sending and receiving RIP updates without any problem. On R2, RIP updates are being sent but not received. This means that the RIP update is being lost at Layer 2.

**Example 3-45  `debug ip packet` Against access-list 100 Shows That R1 Is Sending RIP Updates on the Wire, and R2 Is Not Receiving It**

```
R1#debug ip packet 100 detail
IP packet debugging is on (detailed) for access list 100
R1#

IP: s=131.108.1.1 (Ethernet0), d=255.255.255.255, len 132, sending broadcast/multicast
  UDP src=520, dst=520
```

continues
Chapter 3: Troubleshooting RIP

Example 3-45  **debug ip packet Against access-list 100** Shows That R1 Is Sending RIP Updates on the Wire, and R2 Is Not Receiving It (Continued)

Example 3-46 shows **access-list 100**, which is used against the debug to look at the RIP broadcast/multicast specifically.

Example 3-46  **access-list 100 Is Used Against the Debs to Minimize the Traffic**

Example 3-47  **Multicast Pings Are Failing, Which Means That R2’s Multicast Is Getting Lost at Layer 2**

Solution

RIP-1 sends an update on a broadcast address of 255.255.255.255. In the case of RIP-2, the update is sent on a multicast address of 224.0.0.9. If these two addresses get blocked at Layer 2 or are not being propagated at Layer 2, RIP will not function properly. Layer 2 could be a simple Ethernet switch, a Frame Relay cloud, a bridging cloud, and so on. Fixing the Layer 2 problem is beyond the scope of this book.

Example 3-48 shows that after fixing the Layer 2 problem, RIP routes get installed in the routing table.

Example 3-48  **R2 Is Installing RIP Routes After Fixing the Layer 2 Problems**
RIP Routes Not in the Routing Table—Cause: Offset List Has a Large Metric Defined

Offset lists are used to increase the metric value of RIP updates coming in or going out. The use of an offset list can directly influence the routing table. This list can be applied on selected networks that can be defined in an access list. If the offset value is a large number, such as 14 or 15, the RIP metric will reach infinity when it crosses a couple of routers. That’s why the offset list value should be kept to a minimum value.

Figure 3-15 shows a network setup that can produce a problem in the case of a misconfigured offset list.

Example 3-49 shows that the specific router 131.108.6.0 is not in the routing table of R2.

Example 3-49  R2’s Routing Table Missing the Subnet That Is Off R3

<table>
<thead>
<tr>
<th>R2#show ip route 131.108.6.0</th>
</tr>
</thead>
<tbody>
<tr>
<td>% Subnet not in table</td>
</tr>
</tbody>
</table>

Figure 3-16 shows the flowchart to follow to solve this problem based on this cause.

Figure 3-16  Flowchart to Solve Why RIP Routes Don’t Show Up in a Routing Table

---

RIP routes are not in the routing table of R2.

Is an offset list configured on the sender or receiver?  

When an offset list is applied, the metric should be kept low; otherwise, it can reach the limit of 16 and the route will not get installed. Go to “Debugs and Verification” section.
Debugs and Verification

Troubleshooting should be done to investigate RIP’s normal behavior.

Example 3-50 shows that R2 is receiving other RIP routes, but not 131.108.6.0/24.

Example 3-50  **R2 Is Missing 131.108.6.0/24 from Its Routing Table**

```
R2#show ip route RIP
131.108.0.0/24 is subnetted, 4 subnets
R    131.108.5.0 [120/1] via 131.108.1.1, 00:00:06, Ethernet1
R    131.108.3.0 [120/1] via 131.108.1.1, 00:00:06, Ethernet1
```

This shows that problem is with 131.108.6.0/24, not with RIP in general. The reason is that R3 is receiving other RIP routes from R1, so the RIP update that is coming from R1 is working fine.

Example 3-51 shows the routing table of R1, where 131.108.6.0/24 is present in the routing table.

Example 3-51  **R1 Sees 131.108.6.0/24 in Its Routing Table**

```
R1#show ip route 131.108.6.0
Routing entry for 131.108.6.0/24
Known via "rip", distance 120, metric 1
```

So why is R2 not installing 131.108.6.0/24? This could be because of one of the following reasons:

- R1 is not advertising to R2.
- R1 is advertising, but R2 is not receiving.
- R2 is receiving but is discarding it because of an infinite metric.

The simplest way to troubleshoot such problems is quick configuration examination.

Example 3-52 shows the configuration of Router R1.

Example 3-52  **The Offset List Has a Large Value Configured on R1 for 131.108.6.0/24**

```
R1#router rip
version 2
offset-list 1 out 15 Ethernet0/1
network 131.108.0.0
!
access-list 1 permit 131.108.6.0
```

The administrator has configured an offset list with a very large metric. The offset list is used to change the metric of RIP update.

From the configuration, you can surmise that any update that passes **access-list 1** will have 15 added in the metric. In Example 3-52, **access-list 1** permits 131.108.6.0. This means that the metric of 131.108.6.0 is 16, which, to RIP, is an infinite metric; upon receiving it, R2 will reject it.

To verify this, run the **debug ip rip** command, as demonstrated in Example 3-53.
Problem: RIP Routes Not in the Routing Table

Because 16 is the infinite metric for RIP, R2 will reject 131.108.6.0/24 from going in the routing table.

Solution

Typically, offset lists are not used in RIP networks. When the network has redundant equal-hop (cost) paths and the administrator wants one route preferred over another, an offset list can be used.

For example, suppose that two links exist between R1 and R2. One of the links could be either congested or experiencing delay.

The administrator might want to shift the IP traffic for certain destination subnets to a noncongested link for a short time, to get better throughput and to alleviate some of the congestion. An offset list is an easy way to achieve this by making the RIP metric higher for the subnets on the congested interface.

Example 3-54 shows the new configuration of Router R1.

Example 3-55 shows the routing table of Router R2 after fixing the problem.

RIP Routes Not in the Routing Table—Cause: Routes Reached RIP Hop Count Limit

The RIP metric can go up to a maximum of 15 hops. If a network has more than 15 hops, RIP is not a suitable protocol for it.
Figure 3-17 shows a network setup that produces a RIP hop-count limit problem.

**Figure 3-17  Network Setup That Can Produce a RIP Hop-Count Limit Problem**

R2 is receiving an update for a RIP route, which is several (more than 15) hops away. R2 doesn’t install that route in the routing table, as demonstrated in the output in Example 3-56.

**Example 3-56  R2’s Routing Table Is Missing the Route for 131.108.6.0**

```
R2# show ip route 131.108.6.0
% Subnet not in table
```

Figure 3-18 shows the flowchart to solve this problem.

**Figure 3-18  Flowchart to Solve Why RIP Routes Don’t Show Up in a Routing Table**

Debugs and Verification

The most logical way to start troubleshooting this problem is to look at R1 and determine whether R1 is receiving 131.108.6.0/24.

Example 3-57 shows that Router R1 is receiving RIP routes for 131.108.6.0/24.
Problem: RIP Is Not Installing All Possible Equal-Cost Paths

Example 3-57  *R1’s Routing Table Has 131.108.6.0/24 with a Metric of 15 (Maximum RIP Metric)*

```
R1# show ip route 131.108.6.0
Routing entry for 131.108.6.0/24
Known via "rip", distance 120, metric 15
```

R1 is receiving the route in question, but with a metric of 15. R1 will add 1 more to 15 when advertised to R2, which will result in an infinite metric, consequently preventing the route from being placed in the routing table.

To prove this, in R1, you can run the `debug ip rip` command to view the process in real time.

Example 3-58 shows the output of `debug ip rip` on Router R1.

Example 3-58  *debug ip rip Output Shows That R1 Is Advertising 131.108.6.0 with a Metric of 16 (Infinity)*

```
R1# debug ip rip
RIP protocol debugging is on
RIP: sending v2 update to 224.0.0.9 via Ethernet1 (131.108.1.1)
  131.108.6.0/24 -> 0.0.0.0, metric 16, tag 0
```

Example 3-59 shows the output of `debug ip rip` on Router R2. Router R2 receives this update and discards it because the metric shows that this network is infinitely far away and, therefore, unreachable.

Example 3-59  *debug ip rip Output on R2 Shows That R2 Is Receiving Routes with an Infinite Metric*

```
R2# debug ip rip
RIP protocol debugging is on
RIP: received v2 update from 131.108.1.1 on Ethernet1
  131.108.6.0/24 -> 0.0.0.0 in 16 hops (inaccessible)
```

Solution

This is a classical RIP problem in which a route passes through more than 15 devices. IP networks these days usually have more than 15 routers. There is no way to overcome this behavior other than to pick a routing protocol that does not have a 15-hop limitation. You should use OSPF, EIGRP, or IS-IS instead.

Problem: RIP Is Not Installing All Possible Equal-Cost Paths—Cause: maximum-path Command Restricts RIP from Installing More Than One Path

By default, Cisco routers support only four equal paths for the purpose of load balancing. The `maximum-path` command can be used for up to six equal-cost paths. If the command
is not configured properly, it can cause a problem, as discussed in this section. When configured improperly, the `maximum-path` command allows only one path to the destination, even though more than one path exists. Configuring the command as `maximum-path 1` should be done only when load balancing is not desired.

Figure 3-19 and Example 3-60 provide a network scenario that will be used as the basis for troubleshooting when the `maximum-path` command restricts RIP from installing more than one path, resulting in the omission of all possible equal-cost paths. The sections that follow carefully dissect how to troubleshoot this problem.

Figure 3-19 shows the network setup that produces the problem of RIP not installing all possible equal-cost paths.

**Figure 3-19  RIP Network Vulnerable to an Equal-Cost Path Problem**

Example 3-60 shows the routing table of Router R1. Only one route is being installed in the routing table. By default, any routing protocol supports equal-cost multipaths (load balancing). If more than one equal path exists, it must be installed in the routing table.

**Example 3-60  R1 Installs Only One Path for 131.108.2.0/24**

```
R1# show ip route rip
131.108.0.0/24 is subnetted, 1 subnets
 R 131.108.2.0 [120/1] via 131.108.5.3, 00:00:09, Ethernet2
```

Figure 3-20 shows the flowchart to follow to solve this problem based on this cause.
Problem: RIP Is Not Installing All Possible Equal-Cost Paths

Figure 3-20  Flowchart to Solve Why RIP Routes Don’t Show Up in a Routing Table

Debugs and Verification

Example 3-61 shows the output of `debug ip rip` on Router R1. The output shows that Router R1 is receiving two equal-cost routes.

Example 3-61  `debug ip rip` Output on R1 Shows R1 Receiving Two Updates for the 131.108.2.0 Network

```
R1#debug ip rip
RIP protocol debugging is on
R1#
RIP: received v2 update from 131.108.5.3 on Ethernet2
  131.108.2.0/24 -> 0.0.0.0 in 1 hops
RIP: received v2 update from 131.108.1.2 on Ethernet1
  131.108.2.0/24 -> 0.0.0.0 in 1 hops
```

Only one route is installed in the routing table. You see only one route in the routing table instead of two because operator has configured `maximum-paths 1` in the configuration.

Example 3-62 shows the current configuration for Router R1.

Example 3-62  `R1 Is Configured with maximum-path 1`

```
R1#
router rip
version 2
network 131.108.0.0
maximum-paths 1
```

Solution

By default, Cisco IOS Software allows up to four equal-cost routes to be installed in the routing table. This could be increased up to six routes if configured as in Example 3-63.
Example 3-63 shows the configuration that installs six equal-cost path routes in the routing table.

**Example 3-63  Allowing the Maximum of Six Paths in the Routing Table**

```
R1# router rip
maximum-paths 6
```

This example makes more sense when you have more than four paths and only four are getting installed in the routing table. Because four equal-cost routes is a default, `maximum-paths` needs to be increased to accommodate the fifth and possibly sixth route.

## Troubleshooting RIP Routes Advertisement

All the problems discussed so far deal with the problem on the receiving end or the problem in the middle (Layer 2).

A third possible cause exists when routes are not being installed in the routing table. The sender could be having a problem sending RIP updates for some reason. As a result, the receiver cannot install the RIP routes in the routing table. This section talks about the things that can go wrong on the sender’s side.

This section discusses some of the possible scenarios that can prevent RIP routes from being advertised. Some cases overlap with router installation problems—for example, missing `network` statement(s) or an interface that is down. This section assumes that, after troubleshooting the problems previously addressed in the “Troubleshooting RIP Routes Installation” section, the problems persist. This section presents recommendations on where to go next to resolve those issues.

Two of the most prevalent problems that can go wrong on the sender’s end deal with RIP route advertisement:

- The sender is not advertising RIP routes.
- Subnetted routes are missing.

## Problem: Sender Is Not Advertising RIP Routes

Typically, an IP network running RIP has routers that have a consistent view of the routing table. In other words, all routers have routing tables that contain reachability information for all the IP subnets of the network. This might differ in cases when filtering of certain subnets is done at some routers and not at others. Ideally, all RIP routers have routes of the complete network.
When the routing information differs from one router to the other, one of two possibilities could exist:

- Some routers are not advertising the RIP routes.
- Some routers are not receiving the RIP routes.

This section deals with problems in sending RIP routes.

Figure 3-21 provides a network scenario that will be used as the basis for troubleshooting a majority of following causes of the problem of the sender not advertising RIP routes:

- Missing or incorrect network statement
- Outgoing interface that is down
- distribute-list out blocking the routes
- Advertised network interface that is down
- Outgoing interface defined as passive
- Broken multicast capability (encapsulation failure in Frame Relay)
- Misconfigured neighbor statement
- Advertised subnet is VLSM
- Split horizon enabled

Figure 3-21 shows the network setup in which Router R1 is not sending RIP routes toward R2.

The sections that follow carefully dissect how to troubleshoot this problem based on specific causes.

**Sender Is Not Advertising RIP Routes—Cause: Missing or Incorrect network Statement**

One of the requirements for enabling RIP on a router’s interface is to add the network statement under the router rip command. The network statement decides which interface RIP should be enabled on. If the network statement is misconfigured or not configured, RIP will not be enabled on that interface and RIP routes will not be advertised out that interface.

Figure 3-22 shows the flowchart to follow to fix this problem.
Figure 3-22  *Flowchart to Solve Why the Sender Is Not Advertising RIP Routes*

![Flowchart](image)

**Debugs and Verifications**

Example 3-64 shows the current configuration for R1.

**Example 3-64  *R1 Configuration Shows the Misconfigured network Statement***

```
R1#
interface Loopback0
 ip address 131.108.2.1 255.255.255.0
!
interface Ethernet0
 ip address 131.108.1.1 255.255.255.0
!
router rip
 network 131.107.0.0
```

The **network** statement is incorrectly configured under **router rip** in Example 3-64. Instead of 131.108.0.0, 131.107.0.0 is configured. This will not enable RIP on the interface, and no updates will be sent.

**Solution**

Sometimes, a classless statement is configured under **router rip**, assuming that it will cover all the networks—for example:

```
 router rip
 network 131.0.0.0
```

The **network** statement will not cover 131.0.0.0 through 131.255.255.255 because 131.0.0.0 is a classless network and RIP is a classful protocol. Similarly, if you have multiple Class C addresses, you cannot use one **network** statement to cover all the
addresses that you own. For example, suppose that you own 200.1.1.0 through 200.1.4.0. This doesn’t mean that you can use the following command syntax:

```bash
router rip
network 200.1.0.0
```

The `network` statement here is meaningless for RIP-1 because RIP-1 is a classful protocol. The correct way to advertise all four networks in RIP is as follows:

```bash
router rip
network 200.1.1.0
network 200.1.2.0
network 200.1.3.0
network 200.1.4.0
```

Example 3-65 shows the corrected configuration for R1.

**Example 3-65 Correcting the `network` Statement in the R1 Configuration**

```bash
R1#
interface Loopback0
ip address 131.108.2.1 255.255.255.0
!
interface Ethernet0
ip address 131.108.1.1 255.255.255.0
!
router rip
network 131.108.0.0
```

Example 3-66 shows the routing table of Router R2, showing the learned RIP route.

**Example 3-66 R2 Routing Table Shows That the RIP Routes Are Learned After Correcting the `network` Statement**

```bash
R2#show ip route 131.108.2.0
Routing entry for 131.108.2.0/24
Known via "rip", distance 120, metric 1
  Redistributing via rip
  Last update from 131.108.1.1 on Ethernet0, 00:00:11 ago
  Routing Descriptor Blocks:
    * 131.108.1.1, from 131.108.1.1, 00:00:11 ago, via Ethernet0
  Route metric is 1, traffic share count is 1
```

**Sender Is Not Advertising RIP Routes—Cause: Outgoing Interface Is Down**

RIP is the routing protocol that runs on Layer 3. RIP cannot send updates across an interface if the outgoing interface is down. There can be a variety of possible causes for the outgoing interface being down:

- Interface is up, line protocol is down
- Interface is down, line protocol is down
- Interface is administratively down, line protocol is down
If the outgoing interface shows any of these symptoms, RIP will not be capable of sending any updates across the network. The main thing to note here is that, with any of these potential causes, the line protocol will always show down. This is the most important information to determine Layer 2 connectivity.

Figure 3-23 shows the flowchart to follow to solve this problem based on this cause.

**Figure 3-23**  *Flowchart to Solve Why the Sender Is Not Advertising RIP Routes*

---

**Debugs and Verification**

Example 3-67 shows that the interface Ethernet 0 is down.

**Example 3-67  *Outgoing Interface Ethernet 0 of R1 Shows That the Line Protocol Is Down***

```
R1#show interface ethernet 0
Ethernet0 is up, line protocol is down
Hardware is Lance, address is 0000.0c70.d31e (bia 0000.0c70.d31e)
Internet address is 131.108.1.1/24
```

Example 3-68 shows the `debug ip rip` output. In this debug, R1 is not sending or receiving any RIP updates because Layer 2 is down.

**Example 3-68  *debug ip rip Output Reveals That Nothing Is Being Sent or Received on R1’s Ethernet0 Interface***

```
R1#debug ip rip
RIP protocol debugging is on
R1#
```

In the debug, there are no outputs because of this problem.
Solution

RIP runs above Layer 2. RIP cannot send or receive any routes if Layer 2 is down.

To correct this problem, Layer 2 or Layer 1 must be corrected. Sometimes, the problem could be as simple as loose cables or a bad cable that must be replaced, or it could be as complex as bad hardware, in which case hardware must be replaced.

Example 3-69 shows the interface Ethernet 0 after fixing the Layer 2 problem.

Example 3-69  *Example 3-69  R1’s Outgoing Interface Ethernet0 Is Up After Fixing the Layer 2 Issue*

<table>
<thead>
<tr>
<th>R1#</th>
</tr>
</thead>
<tbody>
<tr>
<td>Ethernet0 is up, line protocol is up</td>
</tr>
<tr>
<td>Hardware is Lance, address is 0000.0c70.d31e (bia 0000.0c70.d31e)</td>
</tr>
<tr>
<td>Internet address is 131.108.1.1/24</td>
</tr>
</tbody>
</table>

Example 3-70 shows the routing table of R2.

Example 3-70  *Example 3-70  R1’s Ethernet0 Interface Is Up, So RIP Is Sending Updates and R2 Has RIP Routes in Its Routing Table*

<table>
<thead>
<tr>
<th>R2#</th>
</tr>
</thead>
<tbody>
<tr>
<td>show ip route 131.108.2.0</td>
</tr>
<tr>
<td>Routing entry for 131.108.2.0/24</td>
</tr>
<tr>
<td>Known via &quot;rip&quot;, distance 120, metric 1</td>
</tr>
<tr>
<td>Redistributing via rip</td>
</tr>
<tr>
<td>Last update from 131.108.1.1 on Ethernet0, 00:00:07 ago</td>
</tr>
<tr>
<td>Routing Descriptor Blocks:</td>
</tr>
<tr>
<td>* 131.108.1.1, from 131.108.1.1, 00:00:07 ago, via Ethernet0</td>
</tr>
<tr>
<td>Route metric is 1, traffic share count is 1</td>
</tr>
</tbody>
</table>

Sender Is Not Advertising RIP Routes—Cause: distribute-list out Is Blocking the Route

distribute-list out is used to filter any routes that will be sent out an interface. If a receiver is complaining about missing routes that should be received, make sure that the routes are not being filtered through distribute-list out. If this is the case, you must modify the access list.

Figure 3-24 shows the flowchart to follow to fix this problem.

Debugs and Verification

Example 3-71 shows the configuration of Router R1. In this configuration, access-list 1 does not explicitly permit the 131.108.0.0 network, so R1 will not be allowed to advertise any 131.108.X.X network, including 131.108.2.0/24.
When using a distribute list, you should always double-check your access list to make sure that the networks that are supposed to be permitted are explicitly permitted in the access list. If not, they will be denied. In the configuration example in Example 3-72, the access list is permitting only 131.107.0.0. An implicit `deny any` at the end of each access list causes the 131.108.0.0 network to be denied. To fix this problem, permit 131.108.0.0 in `access-list 1`, as shown in Example 3-72.

Example 3-71  access-list 1 Does Not Permit the 131.108.0.0 Network

```
R1#
interface Loopback0
 ip address 131.108.2.1 255.255.255.0
!
interface Ethernet0
 ip address 131.108.1.1 255.255.255.0
!
router rip
 network 131.108.0.0
distribute-list 1 out
!
access-list 1 permit 131.107.0.0 0.0.255.255
```

Solution

When using a distribute list, you should always double-check your access list to make sure that the networks that are supposed to be permitted are explicitly permitted in the access list. If not, they will be denied. In the configuration example in Example 3-72, the access list is permitting only 131.107.0.0. An implicit `deny any` at the end of each access list causes the 131.108.0.0 network to be denied. To fix this problem, permit 131.108.0.0 in `access-list 1`, as shown in Example 3-72.

Example 3-72  Reconfiguring access-list 1 to Permit Network 131.108.0.0

```
interface Loopback0
 ip address 131.108.2.1 255.255.255.0
!
interface Ethernet0
 ip address 131.108.1.1 255.255.255.0
!
```
Example 3-72  *Reconfiguring access-list 1 to Permit Network 131.108.0.0 (Continued)*

```
router rip
network 131.108.0.0
distribute-list 1 out
!
access-list 1 permit 131.108.0.0 0.0.255.255
```

Example 3-73  *R2 Routing Table Shows the Entry for the 131.108.2.0 Network After Permitting It in access-list 1*

```
R2# show ip route 131.108.2.0
Routing entry for 131.108.2.0/24
Known via "rip", distance 120, metric 1
    Distributing via rip
    Last update from 131.108.1.1 on Ethernet0, 00:00:07 ago
Routing Descriptor Blocks:
    * 131.108.1.1, from 131.108.1.1, 00:00:07 ago, via Ethernet0
      Route metric is 1, traffic share count is 1
```

**Sender Is Not Advertising RIP Routes—Cause: Advertised Network Interface Is Down**

The network that is being advertised might be down, and the connected route has been removed from the routing table. In this situation, RIP will start advertising that network with an infinite metric of 16; after the hold-down timer has expired, it will no longer advertise this network. As soon as the advertised network comes up, RIP will start advertising it again in its updates.

Figure 3-25 shows the flowchart to follow to fix this problem.

**Figure 3-25  Flowchart to Solve Why the Sender Is Not Advertising RIP Routes**

RIP routes are not being advertised by Router R1.

Is the advertised network interface up/up?

- No  Go to next cause.
  - Not sure  The interface's network number will not be advertised if the interface that represents the network is down. Go to “Debugs and Verification” section.
Debups and Verification

Example 3-74 shows that the line protocol of R1’s Ethernet 1 interface is down, indicating that there is something wrong at Layer 2. This is the interface that is directly attached to the network that needs to be advertised. Therefore, that network cannot be advertised to neighboring routers.

Example 3-74  show interface Output Displays That the Line Protocol of the Advertised Network Is Down

<table>
<thead>
<tr>
<th>Command Output</th>
</tr>
</thead>
<tbody>
<tr>
<td>R1#show interface Ethernet 1</td>
</tr>
<tr>
<td>Ethernet1 is up, line protocol is down</td>
</tr>
<tr>
<td>Hardware is Lance, address is 0000.0c70.d51e (bia 0000.0c70.d51e)</td>
</tr>
<tr>
<td>Internet address is 131.108.2.1/24</td>
</tr>
</tbody>
</table>

When the advertised network’s interface goes down, RIP will detect the down condition. RIP will no longer advertise that network in the RIP update. In Example 3-74, interface Ethernet 1 is down, so RIP will no longer advertise 131.108.2.0/24 in its update.

Solution

You must correct this problem at Layer 2 or Layer 1. Sometimes, the problem could be as simple as loose cables, or it could be as complex as bad hardware, in which case the hardware must be replaced. After fixing the Layer 2 problem, reissue the `show interface` command to view the current status, to verify that it has changed state to up.

Example 3-75 shows that the advertised network interface line protocol is up.

Example 3-75  show interface Output Displays That the Line Protocol of Ethernet1 Is Up After Fixing the Layer 2 Issue

<table>
<thead>
<tr>
<th>Command Output</th>
</tr>
</thead>
<tbody>
<tr>
<td>R1#show interface Ethernet 1</td>
</tr>
<tr>
<td>Ethernet1 is up, line protocol is up</td>
</tr>
<tr>
<td>Hardware is Lance, address is 0000.0c70.d51e (bia 0000.0c70.d51e)</td>
</tr>
<tr>
<td>Internet address is 131.108.2.1/24</td>
</tr>
</tbody>
</table>

When the interface is active again, RIP will begin to advertise that network in its periodic updates. Example 3-76 shows that the route that was down is back in the routing table of R2.

Example 3-76  show ip route Output Displays That R2’s Routing Table Indicates the Network Again After the Layer 2 Issue Is Resolved

<table>
<thead>
<tr>
<th>Command Output</th>
</tr>
</thead>
<tbody>
<tr>
<td>R2#show ip route 131.108.2.0</td>
</tr>
<tr>
<td>Routing entry for 131.108.2.0/24</td>
</tr>
<tr>
<td>Known via &quot;rip&quot;, distance 120, metric 1</td>
</tr>
<tr>
<td>Redistributing via rip</td>
</tr>
<tr>
<td>Last update from 131.108.1.1 on Ethernet0, 00:00:07 ago</td>
</tr>
<tr>
<td>Routing Descriptor Blocks:</td>
</tr>
<tr>
<td>* 131.108.1.1, from 131.108.1.1, 00:00:07 ago, via Ethernet0</td>
</tr>
<tr>
<td>Route metric is 1, traffic share count is 1</td>
</tr>
</tbody>
</table>
Sender Is Not Advertising RIP Routes—Cause: Outgoing Interface Is Defined Passive

A situation might arise in which a router has a complete RIP routing table, but it is not advertising to other routers running RIP. This occurs when not all routers in a RIP network have complete routing tables, resulting in lacking IP connectivity from one part of the network to the other. If the outgoing interface is defined as passive, it will not advertise any RIP updates on that interface.

Figure 3-26 shows the flowchart to follow to fix this problem.

Figure 3-26 Flowchart to Solve Why the Sender Is Not Advertising RIP Routes

Debugs and Verification

Example 3-77 shows the output of `show ip protocols`, which shows that the outgoing interface is defined as a passive interface.

Example 3-77 show ip protocols Output Reveals That the Outgoing Interface on R1 Is Passive
Example 3-77  show ip protocols  Output Reveals That the Outgoing Interface on R1 Is Passive (Continued)

<table>
<thead>
<tr>
<th>Routing Information Sources:</th>
<th>Gateway</th>
<th>Distance</th>
<th>Last Update</th>
</tr>
</thead>
<tbody>
<tr>
<td>131.108.1.2</td>
<td>120</td>
<td>00:00:26</td>
<td></td>
</tr>
<tr>
<td>Distance: (default is 120)</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Example 3-78 shows the configuration of Router R1, which shows that the outgoing interface is defined as passive.

Example 3-78  Configuring the passive interface Command in RIP

```
router rip
passive-interface Ethernet0
network 131.108.0.0
```

Solution

When an interface is defined as a passive interface under RIP, RIP will receive updates on that interface but will not send any updates.

In Example 3-78, the interface Ethernet 0 is defined as passive, so R1 is not sending any updates on Ethernet 0. Sometimes, some networks should be advertised and others should be filtered. In this type of situation, passive interfaces should not be used. Distribute lists, used to selectively filter updates, are a better solution in that case.

Assume that `passive-interface` was configured by mistake. Take this command out of the configuration to solve this problem using the `no` form of the command.

Example 3-79 shows the new configuration to solve this problem.

Example 3-79  Correcting the passive-interface Problem

```
router rip
network 131.108.0.0
```

Example 3-80 shows the routing table of R2 after fixing the problem.

Example 3-80  R2 Routing Table After Removing the passive-interface Command

```
R2#show ip route 131.108.2.0
Routing entry for 131.108.2.0/24
  Known via "rip", distance 120, metric 1
  Routing Descriptor Blocks:
          * 131.108.1.1, from 131.108.1.1, 00:00:07 ago, via Serial0
            Route metric is 1, traffic share count is 1
```

Sender Is Not Advertising RIP Routes—Cause: Broken Multicast Capability (Frame Relay)

In some networking scenarios, router interfaces do not automatically propagate multicast and broadcast traffic unless configured to do so. This could be a major problem because
RIP-1 updates are sent at a broadcast address and RIP-2 uses multicast to exchange routes. No routing information will propagate across the network unless broadcast and multicast features are enabled on such interfaces. Nonbroadcast multiaccess (NBMA) Frame Relay is a prime example of a networking environment in which interfaces exhibit this behavior.

Figure 3-27 shows a network setup that is deliberately configured with broken multicast to illustrate the example of how Frame Relay RIP updates will not go across R1.

In Figure 3-27, Router 1 and Router 2 are connected through Frame Relay. Router 1 is not advertising RIP routes toward Router 2.

**Figure 3-27**  *NBMA Frame Relay Network Vulnerable to Broken Multicast Capability Problems*

![Diagram of NBMA Frame Relay Network Vulnerable to Broken Multicast Capability Problems](image)

Figure 3-28 shows the flowchart to follow to solve this problem based on this cause.

**Figure 3-28**  *Flowchart to Solve Why the Sender Is Not Advertising RIP Routes*

```
RIP routes are not being advertised by Router R1.

Is the multicast capability broken?  Not sure

No  Go to next cause.

If a Layer 2 NBMA network such as Frame Relay or ISDN is configured with static mapping, make sure that the broadcast keyword is added in the map statement. Go to "Debugs and Verification" section.
```

**Debugs and Verification**

Example 3-81 shows the configuration of Router R1. In this example, Frame Relay provides the Layer 2 encapsulation. In this configuration, the `frame-relay map` statement doesn’t have the keyword `broadcast` at the end. As a result, all broadcast/multicast traffic will be prohibited from crossing the NBMA network. The `broadcast` keyword tells the router to replicate the necessary broadcasts and send them across the specified circuits.
Chapter 3: Troubleshooting RIP

Example 3-81  *RI's frame-relay map Statement Lacks the broadcast Keyword*

```
R1# interface Serial3  
   ip address 131.108.1.1 255.255.255.0  
   encapsulation frame-relay  
   frame-relay map ip 131.108.1.2 16  
```

Example 3-83 shows output from `debug ip packet`. This debug includes only the broadcast traffic source from R1. As shown in Example 3-82, R1 is configured with `access-list 100`.

Example 3-82  *Configuration in R1 of access-list 100 to Limit debug Output*

```
R1#: access-list 100 permit ip host 131.108.1.1 host 255.255.255.255
```

R1 is configured with `access-list 100`, which permits all packets from source 131.108.1.1 destined to the broadcast address of 255.255.255.255. In Example 3-83, R1 runs `debug ip packet detail` with `access-list 100` to limit traffic destined to 255.255.255.255 with R1 as the source. The debug output in Example 3-83 shows that there are encapsulation failures, indicating that they cannot be placed in the appropriate Layer 2 frame.

Example 3-83  *debug ip packet Output on R1 Reveals Encapsulation Failure for RIP Updates*

```
R1# debug ip packet 100 detail  
IP packet debugging is on (detailed) for access list 100  
R1#  
IP: s=131.108.1.1 (local), d=255.255.255.255 (Serial3), len 112, sending broad/multicast  
   UDP src=520, dst=520  
IP: s=131.108.1.1 (local), d=255.255.255.255 (Serial3), len 112, encapsulation failed  
   UDP src=520, dst=520
```

Solution

When RIP is running in a Frame Relay (NBMA) environment, Layer 2 must be configured to support broadcast traffic; otherwise, RIP updates will not get across. When static map-ping is used, make sure to add the `broadcast` keyword at the end of a `frame-relay map` statement.

Example 3-84 shows the new configuration of Router R1 with the corrected `frame-relay map` statement.

Example 3-84  *Corrected Configuration to Enable Broadcast Traffic to Go Across an NBMA Environment*

```
R1#:  
   interface Serial3  
   ip address 131.108.1.1 255.255.255.0  
   encapsulation frame-relay  
   frame-relay map ip 131.108.1.2 16 broadcast  
```

Example 3-85 shows the routing table of R2 with RIP routes.
Sender Is Not Advertising RIP Routes—Cause: Misconfigured neighbor Statement

In a nonbroadcast environment, RIP utilizes a unicast method to send RIP updates. To send unicast RIP updates, **neighbor** statements must be configured carefully. If the neighbor address is configured incorrectly in the **neighbor** statement, RIP will not send the unicast update to the neighbor.

Figure 3-29 shows the flowchart to follow to solve this problem based on this cause.

**Figure 3-29**  Flowchart to Solve Why the Sender Is Not Advertising RIP Routes

Debrids and Verification

Example 3-86 shows the RIP configuration in Router R1. The configuration shows that the **neighbor** statement is configured incorrectly. Instead of 131.108.1.2, it’s pointing to 131.108.1.3, which doesn’t exist.

**Example 3-86**  **Router R1 RIP Configuration with Incorrectly Configured neighbor Statement**
Solution

In Example 3-86, RIP is sending a unicast update to a neighbor address of 131.108.1.3, which doesn’t exist.

To solve the problem, the `neighbor` statement must be configured properly.

Example 3-87 shows the corrected configuration of Router R1.

**Example 3-87**  *Router R1 Configuration with the Correct `neighbor` Statement*

```
R1# router rip
network 131.108.0.0
neighbor 131.108.1.2
```

Example 3-88 shows the RIP routes installed in R2’s routing table.

**Example 3-88**  *R2 Routing Table Shows the RIP Entry After Correcting the RIP `neighbor` Statement*

```
R2# show ip route 131.108.2.0
Routing entry for 131.108.2.0/24
Known via "rip", distance 120, metric 1
Redistributing via rip
Last update from 131.108.1.1 on Serial0, 00:00:07 ago
Routing Descriptor Blocks:
* 131.108.1.1, from 131.108.1.1, 00:00:07 ago, via Serial0
   Route metric is 1, traffic share count is 1
```

**Sender Is Not Advertising RIP Routes—Cause: Advertised Subnet Is VLSM**

In almost all IP networks, IP addresses are efficiently utilized by doing variable-length subnet masking (VLSM) of the original IP block. Because RIP-1 does not support VLSM routing, routing VLSM routes becomes a common issue with RIP running networks.

Figure 3-30 shows the network setup, which produces problems because of the existence of a VLSM. The figure shows that Router 1 has an interface whose mask is /24. Note that 131.108.0.0 is variably subnetted to two different masks, 131.108.1.0/24 and 131.108.2.0/25.

**Figure 3-30**  *VLSM Network Example Producing Problems with RIP*

RIP-1 cannot advertise the mask of a subnet, so it cannot support VLSM and cannot advertise /25 to an RIP interface whose mask is /24.

Figure 3-31 shows the flowchart to follow to correct this problem.
Problem: Sender Is Not Advertising RIP Routes

Example 3-89 shows that a loopback interface on R1 is configured for a /25 (255.255.255.128) subnet mask; the interface that will be sourcing RIP update has a /24 (255.255.255.0) mask.

Example 3-89  Configuration to Show VLSM Subnets

<table>
<thead>
<tr>
<th>R1#</th>
</tr>
</thead>
<tbody>
<tr>
<td>interface Loopback0</td>
</tr>
<tr>
<td>ip address 131.108.2.1 255.255.255.128</td>
</tr>
<tr>
<td>interface Ethernet0</td>
</tr>
<tr>
<td>ip address 131.108.1.1 255.255.255.0</td>
</tr>
<tr>
<td>router rip</td>
</tr>
<tr>
<td>network 131.108.0.0</td>
</tr>
</tbody>
</table>

Solution

RIP-1 is not designed to carry subnet mask information. Therefore, any subnet that is using a different mask than the interface that will be sourcing the RIP update will not be advertised by RIP. RIP actually performs a check before sending an update, to make sure that the subnet that will be advertised by RIP has the same subnet mask as the interface that will be sourcing the RIP update. If the mask is different, RIP actually drops the update and will not advertise it.

To solve the problem, either change the subnet mask so that it matches the interface that will be sourcing the RIP update or change the protocol to RIP-2, which does support VLSM.
Example 3-90 shows the configuration changes that correct the problem.

Example 3-90  *Configuring RIP to Advertise VLSM Routes*

```
R1#:
  interface Loopback0
  ip address 131.108.2.1 255.255.255.0
!
  interface Ethernet0
  ip address 131.108.1.1 255.255.255.0
!
  router rip
  version 2
  network 131.108.0.0
```

Example 3-91 shows the routing table of Router R2 after correcting the problem.

Example 3-91  *Router R2 Routing Table After Resolving the VLMS Support Problem*

```
R2#:
  show ip route 131.108.2.0
  Known via "rip", distance 120, metric 1
  Routing entry for 131.108.2.0/25
    Redistributing via rip
    Last update from 131.108.1.1 on Ethernet0, 00:00:07 ago
    Routing Descriptor Blocks:
      * 131.108.1.1, from 131.108.1.1, 00:00:07 ago, via Ethernet0
        Route metric is 1, traffic share count is 1
```

**Sender Is Not Advertising RIP Routes—Cause: Split Horizon Is Enabled**

Split horizon is a feature in RIP to control routing loops. In some situations, it is necessary to enable split horizon to avoid loops. For example, split horizon is necessary in a normal situation when a RIP update is received on an interface and is not sent out on the same interface. Split horizon must be disabled in other environments, such as a hub-and-spoke Frame Relay environment in which spokes have no circuit between them and they go through the hub router, as shown in Figure 3-32.

Another unique situation worth mentioning is one in which a router has an external route that has a next-hop address also known through some interface where other RIP routers are sitting. When those external routes are redistributed into RIP, the router doesn’t advertise that route out the same interface because split horizon is enabled. Also, if a secondary address is configured under an interface, split horizon must be turned off on that interface; otherwise, that secondary address will not be advertised out that interface to other routers.

Figure 3-33 shows the network setup that produces problems when split horizon is enabled. Router 1 is not advertising all RIP routes to Router 3.
Problem: Sender Is Not Advertising RIP Routes

Figure 3-32  Hub-and-Spoke Frame Relay Network Requiring Disabling Split Horizon

Figure 3-33  Split Horizon–Enabled Network Vulnerable to RIP Problems
Figure 3-34 shows the flowchart to follow to fix this problem.

Figure 3-34  *Flowchart to Solve Why the Sender Is Not Advertising RIP Routes*

![Flowchart](image)

- **RIP routes are not being advertised by Router R1.**
  - **Is split horizon enabled on the interface?**
  - **Not sure**
  - **Split-horizon might need to be turned off when advertising secondary addresses or redistributed routes known via the same interface. Go to “Debugs and Verification” section.**
  - **No**
  - **Go to next cause.**

**Debugs and Verification**

Example 3-92 shows the current configuration of R1.

**Example 3-92  *166.166.166.0/24 Is Being Redistributed into RIP on R1***

```
R1#
router rip
redistribute static
   network 131.108.0.0

ip route 155.155.0.0 255.255.0.0  10.10.10.4
ip route 166.166.166.0 255.255.255.0 131.108.1.3
```

Example 3-93 shows that the route 166.166.166.0/24 is not in the routing table of Router R2; however, 155.155.155.0/24 does show up in the routing table.

**Example 3-93  *R2 Routing Table Does Not Show Route 166.166.166.0/24***

```
R2#show ip route rip
R    155.155.0.0/16 [120/1] via 131.108.1.1, 00:00:07, Ethernet0
```

Example 3-94 shows the **debug ip rip** output on Router R1. R1 is advertising only 155.155.0.0/16, not 166.166.166.0/24. In R2’s routing table, no route exists for 166.166.166.0/24.

**Example 3-94  *debug ip rip Output Displays 166.166.166.0 Is Not Being Advertised by R1***

```
R1#debug ip rip
RIP protocol debugging is on
RIP: sending v1 update to 255.255.255.255 via Ethernet0 (131.108.1.1)
RIP: build update entries
network 155.155.0.0 metric 1
```
Solution

This problem occurs because the next hop of 166.166.166.0/24 is 131.108.1.2. With split horizon, RIP will suppress this update from going out the same interface that 166.166.166.0/24 is learned. Notice that the route 155.155.155.0/24 was advertised by R1 because the next-hop address of that route was 10.10.10.4, which is a different interface on R1.

The solution lies in turning off split horizon on the Ethernet 0 interface of R1.

A similar situation would arise if 166.166.166.0/24 was defined as a secondary interface address on R1, which will not advertise this secondary interface address in its RIP update unless split horizon is turned off.

Example 3-95 shows the new configuration on Router R1 to solve this problem.

Example 3-95  *Disabling Split-Horizon on R1’s Ethernet 0 Interface*

```
R1#
interface Ethernet0
 ip address 131.108.1.1 255.255.255.0
 no ip split-horizon
```

Example 3-96 shows that after making the configuration changes, R2 is receiving 166.166.166.0/24 in the RIP updates.

Example 3-96  *R2 Routing Table After Split Horizon Has Been Disabled Confirms That RIP Updates Reflect the 166.166.166.0/24 Route*

```
R2#
show ip route rip
 R    155.155.0.0/16 [120/1] via 131.108.1.1, 00:00:08, Ethernet0
 R    166.166.0.0/16 [120/1] via 131.108.1.1, 00:00:08, Ethernet0
```

This problem can also be seen when interfaces are configured with secondary IP addresses.

Example 3-97 shows the interface configuration with secondary IP address.

Example 3-97  *Interface Configuration with Secondary Addresses*

```
R1#
interface Ethernet0
 ip address 131.108.2.1 255.255.255.0 secondary
 ip address 131.108.1.1 255.255.255.0
```

If split horizon is enabled, this secondary address will not be advertised on Ethernet0.

Similarly, imagine a situation in which there are three routers—R1, R2, and R3—on the same Ethernet, as shown in Figure 3-35.

R1 and R3 are running OSPF. R1 and R2 are running RIP, as in the preceding example. Now, R3 advertises certain routes through OSPF to R1 that R1 must redistribute in RIP. R1 will not advertise those OSPF routes to R2 because of split horizon. The solution is again to disable split horizon.
Basicly, these are the three main reasons for turning off split horizon. Any other situation might create a routing loop if split horizon is turned off.

Problem: Subnetted Routes Missing from the Routing Table of R2—Cause: Autosummarization Feature Is Enabled

In some situations, subnetted routes are not advertised in RIP. Whenever RIP sends an update across a major network boundary, the update will be autosummarized. This is not really a problem; this is done to reduce the size of the routing table.

Figure 3-36 shows a network setup in which R1 has subnets of 155.155.0.0, but R2 shows none of these subnets in its routing table. Either R1 is not advertising them to R2, or R2 is not receiving them. The chances of R1 not advertising more specific subnets of 155.155.0.0/16 is more favorable.

Example 3-98 shows that the subnetted route of 155.155.0.0/16 is missing from the routing table of R2, but the major network route is present. This means that R1 is advertising the routes but is somehow summarizing the subnets to go as 15.155.0.0/16.
Problem: Subnetted Routes Missing from the Routing Table of R2

Figure 3-36  *RIP Network Vulnerable to Autosummarization Problems*  

Example 3-98  *R2’s Routing Table Reflects That the Subnetted Route Is Missing*

```
R2# show ip route 155.155.155.0 255.255.255.0
  % Subnet not in table

R2# show ip route 155.155.0.0
  Routing entry for 155.155.0.0/16
  Known via "rip", distance 120, metric 1
  Redistributing via rip (self originated)
  Last update from 131.108.1.1 on Ethernet0, 00:00:01 ago
  Routing Descriptor Blocks:
  * 131.108.1.1, from 131.108.1.1, 00:00:01 ago, via Ethernet0
    Route metric is 1, traffic share count is 1
```

Figure 3-37 shows the flowchart to fix this problem based on the autosummarization feature being enabled.

Figure 3-37  *Flowchart to Solve Why the Sender Is Not Advertising RIP Routes*

RIP-2 routes are not being advertised by Router R1.

Is the autosummarization feature enabled?

Not sure

No

Go to next cause.

When RIP crosses a major network border, it automatically summarizes to classful boundaries. Go to “Debugs and Verification” section.
Debugs and Verification

Example 3-99 shows the configuration of R1 in the case of RIP-1. RIP-1 is a classful protocol and always summarizes to classful boundaries for nondirectly connected major networks.

Example 3-99  R1 Configuration with RIP Version 1

```
R1#
interface Loopback1
  ip address 131.108.2.1 255.255.255.0
!
interface Loopback3
  ip address 155.155.155.1 255.255.255.0
!
interface Ethernet0
  ip address 131.108.1.1 255.255.255.0
!
router rip
  network 131.108.0.0
  network 155.155.0.0
```

Example 3-100 shows the routing table in Router R2. Notice that R2 is receiving 155.155.0.0/16, not 155.155.155.0/24, as configured on R1. Also note that R2 is receiving a /24 route of 131.108.2.0, the route of the same major network as that of interface Ethernet 0, which connects R1 to R2.

Example 3-100  R2 Routing Display to Show How Subnetted Routes Are Summarized to Classful Boundaries

```
R2#show ip route RIP
R    155.155.0.0/16 [120/1] via 131.108.1.1, 00:00:22, Ethernet0
     131.108.0.0/24 is subnetted, 3 subnets
R    131.108.2.0 [120/1] via 131.108.1.1, 00:00:22, Ethernet0
```

Solution

In RIP-1, there is no workaround for this problem because RIP-1 is a classful routing protocol. RIP-1 automatically summarizes any update to a natural class boundary when that update goes over an interface configured with a different major network.

As indicated by R2’s routing table in Example 3-100, 155.155.155.0/24 is advertised over an interface configured with 131.108.0.0. This summarizes 155.155.155.0/24 to a Class B boundary as 155.155.0.0/16.

In RIP-1, this is not a problem because RIP-1 is a classful protocol and the network should be designed with this understanding. With RIP-2, however, Cisco routers can be configured to stop the autosummarization process.

For example, R1’s configurations can be changed to run a RIP-2 process rather than a RIP-1 process.

Example 3-101 shows the configuration that solves this problem for RIP-2.

```
Example 3-101  R2 Configuration with RIP Version 2
```

```
R2#show ip route RIP
R    155.155.0.0/16 [120/1] via 131.108.1.1, 00:00:22, Ethernet0
     131.108.0.0/24 is subnetted, 3 subnets
R    131.108.2.0 [120/1] via 131.108.1.1, 00:00:22, Ethernet0
```
Problem: RIP-2 Routing Table Is Huge—Cause: Autosummarization Is Off

Example 3-101  *Disabling Autosummarization in RIP-2*

```
router rip
version 2
network 131.108.0.0
network 155.155.0.0
no auto-summary
```

Example 3-102  *Router R2’s Routing Table Shows That It Is Receiving the Subnetted Route 155.155.155.0/24*

```
R2#show ip route 155.155.0.0
155.155.0.0/24 is subnetted, 1 subnets
R 155.155.155.0 [120/1] via 131.108.1.1, 00:00:21, Ethernet0
131.108.0.0/24 is subnetted, 3 subnets
R 131.108.2.0 [120/1] via 131.108.1.1, 00:00:21, Ethernet0
```

**Troubleshooting Routes Summarization in RIP**

Route summarization refers to summarizing or reducing the number of routes in a routing table. For example, 131.108.1.0/24, 131.108.2.0/24 and 131.108.3.0/24 can be reduced to one route entry (that is, 131.108.0.0/16 or 131.108.0.0/22), the latter of which will cover only these three subnets. Route summarization (autosummarization and manual summarization, both of which are addressed in this section) is used to reduce the size of the routing table. This section discusses the most significant problem related to the route summarization—the RIP-2 routing table is huge. Two of the most common causes for this are as follows:

- Autosummarization is off.
- **ip summary-address** is not used.

Figure 3-38 shows a network setup that could produce a large routing table.

**Figure 3-38  Network Setup That Could Generate a Large Routing Table**

![Network Setup That Could Generate a Large Routing Table](chart)

Problem: RIP-2 Routing Table Is Huge—Cause: Autosummarization Is Off

When a RIP update crosses a major network, it summarizes to the classful boundary. For example, 131.108.1.0, 131.108.2.0, and 131.108.3.0 will be autosummarized to 131.108.0.0/16.
when advertised to a router with no 131.108.X.X addresses on its inter-faces. Disabling the autosummarization feature increases the size of the routing table. In some situations, this feature must be turned off (for example, if discontiguous networks exist, as discussed earlier).

Figure 3-39 shows the flowchart to follow to solve this problem based on this cause.

**Figure 3-39  Flowchart to Resolve a Large RIP-2 Routing Table**

Debugs and Verification

Example 3-103 shows the configuration on R2 that produces this problem. In this configuration, R2 has autosummary turned off.

**Example 3-103  Disabling Autosummarization Under RIP for R2**

```
R2# router rip
    version 2
    network 132.108.0.0
    network 131.108.0.0
    no auto-summary
```

Example 3-104 shows R1’s routing table. This routing table has only four routes, but in a real network with the configuration in Example 3-103, there could be several hundred routes. R1 is receiving every subnet of 131.108.0.0/16. In this example, these are only three, but it can be much, much worse.

**Example 3-104  Router R1 Routing Table Shows Subnetted Routes in the Routing Table**

```
R1# show ip route rip
  131.108.0.0/24 is subnetted, 3 subnets
    R 131.108.3.0 [120/1] via 132.108.1.2, 00:00:24, Serial3
    R 131.108.2.0 [120/1] via 132.108.1.2, 00:00:24, Serial3
    R 131.108.1.0 [120/1] via 132.108.1.2, 00:00:24, Serial3
R1#```
Solution

Because the autosummarization feature is disabled under the RIP configuration of R2, R1 sees the subnetted routes in the routing table. When this feature is enabled, all the subnetted routes will go away.

Example 3-105 shows the altered configuration of R2. In this configuration, autosummarization is on, to reduce the size of the routing table. Because this is the default, you will not see it in the configuration. The command to enable autosummarization is `auto-summary` under `router rip`.

Example 3-105  \textit{R2 Uses Autosummarization to Reduce Routing Table Size}

```
R2#
router rip
version 2
network 132.108.0.0
network 131.108.0.0
```

Example 3-106 shows the reduced size of the routing table.

Example 3-106  \textit{Autosummary Reduces the Routing Table Size for Router R1}

```
R1#show ip route rip
R    131.108.0.0/16 [120/1] via 132.108.1.2, 00:00:01, Serial3
```

Problem: RIP-2 Routing Table Is Huge—Cause: ip summary-address Is Not Used

Figure 3-40 shows the network setup that could produce a large routing table.

**Figure 3-40  Network Setup That Could Generate a Large Routing Table**

Figure 3-40 shows that R2 is announcing several subnets of 131.108.0.0 network. Notice that the link between R1 and R2 is also part of the 131.108.0.0 network, so autosummarization cannot play any role to solve the problem of receiving a subnet route that could be summarized. The autosummarization feature could have worked only if the R1, R2 link was in a different major network.

Figure 3-41 shows the flowchart to follow to solve this problem based on this cause.
Chapter 3: Troubleshooting RIP

Example 3-107 shows that in the configuration of R2, the `ip summary-address` command is not used under the Serial 1 interface to summarize the routes.

Example 3-108 shows the routing table of R1. In this example, there are only three routes. In a real network, however, the number could be worse based on the configuration in Example 3-107.

Solution

In the situation described in the preceding section, autosummary is on but is not helpful because the whole network is within one major network. Imagine a network with Class B address space with thousands of subnets. Autosummary cannot play any role here because...
no major network boundary is crossed. A new feature of summarization was introduced in RIP starting with Cisco IOS Software Release 12.0.7T. This feature is similar to EIGRP manual summarization.

Example 3-109 shows the new configuration that solves this problem. This configuration reduces the size of the routing table. This command can be used with different masks so that, if a network has contiguous blocks of a subnet, the router could be configured to summarize subnets into smaller blocks. This then would reduce the routes advertised to the RIP network.

Based on the preceding configuration, R2 will summarize the RIP route on the Serial 1 interface. Any network subnet that falls in the 131.108.0.0 network will be summarized to one 131.108.0.0 major network, and its mask will be 255.255.252.0. This means that R2 will announce only a single summarize route of 131.108.0.0/22 and will suppress the subnets of 131.108.0.0.

Example 3-110 shows the routing table of Router R1 with a reduced number of entries as a result of summarization.

This section talks about problems that can happen during redistribution in RIP. Redistribution refers to the case when another routing protocol or a static route or connected route is being injected into RIP. Special care is required during this process to avoid any routing loops. In addition, metric (hop count) should be defined during this process, to avoid problems.

The most prevalent problem encountered with RIP redistribution is that redistributed routes are not being installed in the routing table of the RIP routers receiving these routes. When destination routes are not present in a routing table, no data can reach those destinations. The most common cause of this is a metric that is not defined during redistribution into RIP.

In RIP, the metric for a route is treated as a hop count that shows the number of routers that exist along this route. As discussed in Chapter 2, 15 is the maximum hop count that RIP supports; anything greater than 15 is treated as the infinite metric and, upon receipt, is dropped.
Figure 3-42 shows the network setup that could produce the problem in which redistributed routes do not get installed in the routing table of the receiver.

**Figure 3-42** *Network Vulnerable to Redistributed Route Problems*

R1 and R3 are running OSPF in Area 0, whereas R1 and R2 are running RIP. R3 is announcing 131.108.6.0/24 through OSPF to R1. In R1, OSPF routes are being redistributed into RIP, but R2 is not receiving 131.108.6.0/24 through RIP.

Figure 3-43 shows the flowchart to follow to solve this problem based on this cause.

**Figure 3-43** *Flowchart to Resolve Redistributed Route Problems*

Redistributed RIP routes are not on the routing table of R2.

- Is the default metric defined on the redistribution router? 
  - Yes: Go to next problem.
  - No: Go to next problem.

When redistributing into RIP, the metric must be defined between 1 and 15; otherwise, RIP advertises the redistributed route with a metric of 16 (infinity). Go to “Debugs and Verification” section.
Debugs and Verification

To troubleshoot this problem, you need to investigate whether R1 is receiving 131.108.6.0/24.

Example 3-111 shows that R3 is advertising 131.108.6.0/24 through OSPF to R1.

Example 3-111  show ip route  Output Confirms That OSPF Is Working Fine and That R1 Is Receiving 131.108.6.0/24

<table>
<thead>
<tr>
<th>R1#show ip route 131.108.6.0</th>
</tr>
</thead>
<tbody>
<tr>
<td>Routing entry for 131.108.6.0/24</td>
</tr>
<tr>
<td>Known via &quot;ospf 1&quot;, distance 110, metric 20, type intra area</td>
</tr>
</tbody>
</table>

R1 must be configured to redistribute OSPF routes in RIP. Example 3-112 shows that R1 is redistributing OSPF in RIP.

Example 3-112  Configuring R1 So That OSPF Is Redistributed in RIP

<table>
<thead>
<tr>
<th>R1#</th>
</tr>
</thead>
<tbody>
<tr>
<td>router rip</td>
</tr>
<tr>
<td>version 2</td>
</tr>
<tr>
<td>redistribute ospf 1</td>
</tr>
<tr>
<td>network 131.108.0.0</td>
</tr>
</tbody>
</table>

Now, you must first investigate R2 whether 131.108.6.0/24 is coming.

Example 3-113 shows that, in R2, 131.108.6.0/24 is not present in the RIP routing table.

Example 3-113  R2 Routing Table Does Not Reflect That 131.108.6.0/24 Is Present

<table>
<thead>
<tr>
<th>R2#show ip route 131.108.6.0</th>
</tr>
</thead>
<tbody>
<tr>
<td>% Subnet not in table</td>
</tr>
</tbody>
</table>

There are two basic ways to view this issue. The first is a simple show run on R1. The second is to run the debug ip rip on R2 command to watch the process.

Example 3-114 shows the output of debug ip rip.

Example 3-114  debug ip rip  Output Shows That 131.108.6.0/24 Is Inaccessible

<table>
<thead>
<tr>
<th>R2#debug ip rip</th>
</tr>
</thead>
<tbody>
<tr>
<td>RIP: received v2 update from 131.108.1.1 on Ethernet1</td>
</tr>
<tr>
<td>131.108.6.0/24 -&gt; 0.0.0.0 in 16 hops (inaccessible)</td>
</tr>
</tbody>
</table>

Solution

In RIP-1 or RIP-2, 16 is considered to be an infinite metric. Any update with a metric greater than 15 will not be considered for entry into the routing table.

In this example, the OSPF route in R1 for 131.108.6.0/24 has a metric of 20. When OSPF is redistributed into RIP in R1, OSPF advertised 131.108.6.0/24 with a metric of 20, which exceeds the maximum metric allowed in RIP. OSPF knows only cost as a metric, whereas
RIP utilizes hop count. No metric translation facility exists, so the administrator must configure a metric to be assigned to redistributed routes.

Without the default metric configuration in R1, R2, upon receiving this update, complains about the excessive metric and marks it as (inaccessible), as shown in Example 3-114.

To correct this problem, R1 needs to assign a valid metric through configuration when doing the redistribution, as done for R1 in Example 3-115.

**Example 3-115**  Assigning a Valid Metric for Successful Redistribution

```plaintext
R1#
router rip
version 2
redistribute ospf 1 metric 1
network 131.108.0.0
```

In the configuration of Example 3-155, all redistributed routes from OSPF in RIP get a metric of 1. This metric is treated as hop count by R2.

Example 3-116 shows that R2 is receiving the correct route with a metric of 1.

**Example 3-116**  debug ip rip Reveals That the New Configuration for R1 Works and That R2 Is Receiving the Correct Route

```plaintext
R2#debug ip rip
RIP: received v2 update from 131.108.1.1 on Ethernet1
131.108.6.0/24 -> 0.0.0.0 in 1 hops
```

Example 3-117 shows that the route gets installed in the routing table of R2.

**Example 3-117**  R2 Routing Table Reflects That the Redistribution for Route 131.108.6.0/24 Is Successful

```plaintext
R2#show ip route 131.108.6.0
Routing entry for 131.108.6.0/24
Known via "rip", distance 120, metric 1
```

## Troubleshooting Dial-on-Demand Routing Issues in RIP

Dial-on-demand routing (DDR) is common in scenarios in which the ISDN or similar dialup links are used as a backup link. When the primary link goes down, this backup link comes up. RIP begins sending and receiving updates on this link as long as the primary link is down.

The dialup links can be used as a backup for the primary link in two ways:

- Use the **backup interface** command.
- Use a floating static route with a dialer list that defines interesting traffic.
The first method is very simple: The command is typed under the dial interface, indicating that it’s a backup for a primary interface.

The second method requires a floating static route with a higher administrative distance than RIP (for example, 130 or above). It also requires defining interesting traffic that should bring up the link. The RIP broadcast address of 255.255.255.255 must be denied in the dialer list, so it shouldn’t bring up the link unnecessarily.

When running RIP under DDR situations, there are a number of issues to consider. Some problems are related to the ISDN line or an async line in which RIP updates keep bouncing. Some problems are related to the configuration. This section talks about the two most common dialup problems:

- A RIP broadcast is keeping the link up.
- RIP updates are not going across the dialer interface.

**Problem: RIP Broadcast Is Keeping the ISDN Link Up**—**Cause: RIP Broadcasts Have Not Been Denied in the Interesting Traffic Definition**

ISDN links are typically used as backup links when primary links go down. Cisco IOS Software requires that a router be instructed on which kind of traffic can bring up the ISDN link and keep it up. Such traffic is referred to as *interesting traffic*. Network operators typically want data traffic to be considered as interesting traffic to bring and keep the ISDN link up. RIP or other routing protocol updates should not be defined as interesting traffic. If this is not done, when the ISDN link comes up, it stays up as long as routing updates (RIP, in this case) are sent on a regular basis. That is not be the desired behavior because ISDN provides low-speed connectivity, and some data actually might go over the slow link even though the primary faster link is available.

Figure 3-44 shows the network setup that produces these particular DDR issues.

*Figure 3-44  Network Setup Vulnerable to DDR Problems*
Chapter 3: Troubleshooting RIP

Figure 3-45 shows the flowchart to follow to fix this problem.

Figure 3-45  Flowchart to Solve the RIP Broadcast Keeping the ISDN Link Up Problem

Debugs and Verification

Example 3-118 shows the configuration on Router R1 that produces this problem. In this configuration, only TCP traffic is denied. In other words, TCP traffic will not bring up and sustain the link. RIP broadcasts utilize UDP port 520. Because the `permit ip any any` command allows UDP port 520 to go through, RIP traffic is considered interesting traffic.

In Example 3-118, interface BRI 3/0 is configured to dial via the `dialer-map` command to the router with an IP address of 192.168.254.14 (R2). The number of dial is 57654. The `dialer-group` command defines `dialer-list 1`, which relies on `access-list 100` to define the interesting traffic. In this example, `access-list 100` denies all TCP traffic and permits all IP traffic. In other words, TCP traffic will not bring up and keep up the ISDN link, whereas other traffic, including RIP, can do so.

Example 3-118  Configuring the ISDN Interface with `dialer-group` to Define Interesting Traffic

```
R1#
interface BRI3/0
ip address 192.168.254.13 255.255.255.252
encapsulation ppp
dialer map ip 192.168.254.14 name R2 broadcast 57654
dialer-group 1
isdn switch-type basic-net3
ppp authentication chap

access-list 100 deny tcp any any
access-list 100 permit ip any any
dialer-list 1 protocol ip list 100
```
Example 3-119 shows the output of `show dialer`, which shows that the reason for the link coming up is a RIP broadcast.

**Example 3-119  show dialer Output Reveals That a RIP Broadcast Is Keeping the ISDN Link Up**

```
R1#show dialer
BRI1/1:1 - dialer type = ISDN
Idle timer (120 secs), Fast idle timer (20 secs)
Wait for carrier (30 secs), Re-enable (2 secs)
Dialer state is data link layer up
Dial reason: ip (s=192.168.254.13, d=255.255.255.255)
Current call connected 00:00:08
Connected to 57654 (R2)
```

In Example 3-119, **Dial reason** section 255.255.255.255 is the destination IP address, which is the address where RIP-1 advertisements will go on BRI1/1:1. **Dial reason** indicates that the interesting traffic is RIP, which has caused this ISDN to dial in the first place.

**Solution**

When running RIP and DDR, define an access list for interesting traffic. In Example 3-118, the access list is denying only the TCP traffic and permitting all the IP traffic. RIP uses an IP broadcast address of 255.255.255.255 to send the routing updates. This address must be denied in the access list so that RIP doesn’t bring up the link every 30 seconds. Denying 255.255.255.255 as a destination will block all broadcast traffic from bringing up the link. Blocking UDP port 520 will block RIP-1 and RIP-2 updates specifically. When the link is up, RIP can flow freely across the link. However, it will not keep the link up because it’s not part of the interesting traffic definition.

Example 3-120 shows the correct configuration change in Router R1. In this configuration, all traffic destined to 255.255.255.255 address is denied. This covers all broadcast traffic, so RIP-1 will not bring up the link after this configuration change.

**Example 3-120  Correct Configuration for Router R1 in access-list 100 to Deny Traffic from the RIP-1 Broadcast IP Address**

```
R1#
access-list 100 deny ip any 255.255.255.255
access-list 100 permit ip any any
dialer-list 1 protocol ip list 100
```

One important thing to know here is that RIP-1 uses the 255.255.255.255 address for sending RIP updates. RIP-2, on the other hand, uses 224.0.0.9. So, when dealing with RIP-2, you need to deny traffic from the multicast address of 224.0.0.9 as interesting traffic, as demonstrated in Example 1-21.

**Example 3-121  Configuration for Router R1 in access-list 100 to Deny Traffic from the RIP-2 Broadcast IP Address**

```
R1#
access-list 100 deny ip any 224.0.0.9
access-list 100 permit ip any any
```
Also, in a situation in which both RIP-1 and RIP-2 are running, both of these broadcast addresses should be denied in the access list, as demonstrated in Example 3-122.

Example 3-122  Configuration for Router R1 in access-list 100 to Deny Traffic from the RIP-1 and RIP-2 Broadcast IP Addresses

```
access-list 100 deny ip any 255.255.255.255
access-list 100 deny ip any 224.0.0.9
access-list 100 permit ip any any
```

Because both RIP-1 and RIP-2 use UDP port 520, it would be most efficient to deny this port if RIP-1 and RIP-2 are not considered interesting traffic. Example 3-123 demonstrates this.

Example 3-123  Configuring access-list 100 for R1 to Deny Traffic from the RIP-1 and RIP-2 UDP Port

```
R1#
access-list 100 deny udp any any eq 520
access-list 100 permit ip any any
```

The final configuration of R1 would like Example 3-124.

Example 3-124  Efficient Configuration of R1 when RIP-1 and RIP-2 Are Both Denied as Interesting Traffic

```
R1#
interface BRI3/0
ip address 192.168.254.13 255.255.255.252
encapsulation ppp
dialer map ip 192.168.254.14 name R2 broadcast 57654
dialer-group 1
isdn switch-type basic-net3
ppp authentication chap
!
access-list 100 deny udp any any eq 520
access-list 100 permit ip any any
!
dialer-list 1 protocol ip list 100
```

Problem: RIP Updates Are Not Going Across the Dialer Interface—Cause: Missing broadcast Keyword in a dialer map Statement

When a dialer interface (ISDN, for example) comes up, you might want to run a routing protocol over this link. Static routes might do the job, but in networks with a large number of routes, static routes might not scale. Therefore, running a dynamic routing protocol such as RIP is necessary. In some situations, the ISDN link might be up, but no routing information is going across. Without a routing protocol, no destination addresses can be learned and no traffic can be sent to those destinations. This problem must be fixed because the ISDN interface is of no use when it is not carrying any traffic.
Problem: RIP Updates Are Not Going Across the Dialer Interface

Figure 3-46 shows the flowchart to follow to solve this problem based on this cause.

**Figure 3-46**  Flowchart to Solve the RIP Updates Not Going Across the Dialer Interface Problem

![Flowchart](image)

**Debugs and Verification**

Example 3-125 shows the configuration on R1 that produces this problem.

**Example 3-125**  Configuring R1 When No Routing Updates Will Go on the ISDN Link

```plaintext
R1# interface BRI3/0
ip address 192.168.254.13 255.255.255.252
encapsulation ppp

**dialer map**
ip 192.168.254.14 name R2 57654
isdn switch-type basic-net3
ppp authentication chap
```

Example 3-126 shows that RIP is sending the broadcast update toward R2. You can see that it’s failing because of the **encapsulation failed** message. Also in Example 3-126, R1 is running a `debug ip packet` command with `access-list 100` to display only the UDP port 520 output. RIP-1 and RIP-2 use UDP port 520 to exchange updates with other RIP running routers.

**Example 3-126**  Discovering Why RIP Routes Are Not Going Across an ISDN Interface

```plaintext
R1#
access-list 100 permit udp any any eq 520
access-list 100 deny ip any any

R1#debug ip packet 100 detail
IP: s=192.168.254.13 (local), d=255.255.255.255 (BRI3/0), len 46, sending
    broadcast/multicast
    UDP src=520, dst=520
IP: s=192.168.254.13 (local), d=255.255.255.255 (BRI3/0), len 72, **encapsulation failed**
    UDP src=520, dst=520
```
Solution

The root of the issue is RIP’s use of broadcasts to send its routing updates. In DDR, `dialer map` statements are necessary to associate the next-hop protocol address to the phone number dialed to get to the destination. The `broadcast` keyword must be used in the `dialer map` statements; otherwise, the broadcast will encounter the encapsulation failure message demonstrated by Example 3-126. To correct this problem, add the `broadcast` keyword in the `dialer map` statement, as demonstrated in Example 3-127 for Router R1.

**Example 3-127**  Corrected Configuration of R1 to Enable RIP Updates to Go Across the ISDN Interface

```
interface BRI3/0
ip address 192.168.254.13 255.255.255.252
encapsulation ppp
dialer map ip 192.168.254.14 name R2 broadcast 57654
dialer-group 1
isdn switch-type basic-net3
ppp authentication chap
```

Troubleshooting Routes Flapping Problem in RIP

Running RIP in a complex environment can sometimes cause flapping of routes. Route flapping refers to routes coming into and going out of the routing table. To check whether the routes are indeed flapping, check the routing table and look at the age of the routes. If the ages are constantly getting reset to 00:00:00, this means that the routes are flapping. Several reasons exist for this condition. This section discusses one of the common reasons—packet loss because the packet is dropping on the sender’s or receiver’s interface. The example in this section considers Frame Relay because it is the most common medium in which this problem occurs. The packet loss can be verified through the interface statistics by looking at the number of packet drops and determining whether that number is constantly incrementing.

Figure 3-47 shows the network setup that can produce RIP route flapping.

Figure 3-48 shows the flowchart to follow to solve this problem.

Debugs and Verification

In a large RIP network, especially, in a Frame Relay environment, there is a high possibility that RIP updates are lost in the Frame Relay cloud or that the RIP interface dropped the update. Again, the symptoms can be present in any Layer 2 media, but Frame Relay is the focus here. This situation causes RIP to lose a route for a while. If RIP does not receive a route for 180 seconds, the route is put in a holddown for 240 seconds and then is purged. This situation is corrected by itself (and time), but, in some cases, configuration changes can be required. For example, consider the output in Example 3-128, where no RIP update has been received for 2 minutes and 8 seconds. This means that four RIP updates have been missed, and we are 8 seconds into the fifth update.
Figure 3-47  *Network Vulnerable to RIP Route Flapping*

![Diagram showing network vulnerable to RIP Route Flapping](image)

Figure 3-48  *Flowchart to Solving the RIP Route Flapping Problem*

```
RIP routes are flapping.

Are there a large number of packet drops being reported by router interfaces in the network?

Not sure

No

This is the end of all the problems in this chapter.
```

Example 3-128  *Routing Table of the Hub Router Showing That the Last RIP Update Was Received 2:08 Minutes Ago*

```
Hub#show ip route rip
R  155.155.0.0/16 [120/1] via 131.108.1.1, 00:02:08, Serial0
R  166.166.0.0/16 [120/1] via 131.108.1.1, 00:02:08, Serial0
```
Example 3-129 shows that there are a large number of broadcast drops on the interface.

**Example 3-129**  
*show interfaces serial 0*  
*Output Reveals a Large Number of Broadcast Drops*

```
Hub# show interfaces serial 0
Serial0 is up, line protocol is up
Hardware is MK5025
Description: Charlotte Frame Relay Port DLCI 100
MTU 1500 bytes, BW 1024 Kbit, DLY 20000 usec, rely 255/255, load 44/255
Encapsulation FRAME-RELAY, loopback not set, keepalive set (10 sec)
LMI enq sent 7940, LMI stat recvd 7937, LMI upd recvd 0, DTE LMI up
LMI enq recvd 0, LMI stat sent 0, LMI upd sent 0
LMI DLCI 1023 LMI type is CISCO frame relay DTE
Broadcast queue 64/64, broadcasts sent/dropped 1769202/1849660, interface broadcasts 3579215
```

**Solution**

The *show interfaces serial 0* command further proves that there is some problem at the interface level. Too many drops are occurring at the interface level. This is the cause of the route flapping. In the case of Frame Relay, the Frame Relay broadcast queue must be tuned. Tuning the Frame Relay broadcast queue is out of the scope of this book; several papers at Cisco’s Web site discuss how to tune the Frame Relay broadcast queue.

In a non-Frame Relay situation, the input or output hold queue might need to be increased.

Example 3-130 shows that after fixing the interface drop problem, route flapping disappears.

**Example 3-130**  
*Serial Interface Output After Adjusting the Broadcast Queue*

```
Hub# show interfaces serial 0
Serial0 is up, line protocol is up
Hardware is MK5025
Description: Charlotte Frame Relay Port DLCI 100
MTU 1500 bytes, BW 1024 Kbit, DLY 20000 usec, rely 255/255, load 44/255
Encapsulation FRAME-RELAY, loopback not set, keepalive set (10 sec)
LMI enq sent 7940, LMI stat recvd 7937, LMI upd recvd 0, DTE LMI up
LMI enq recvd 0, LMI stat sent 0, LMI upd sent 0
LMI DLCI 1023 LMI type is CISCO frame relay DTE
Broadcast queue 0/256, broadcasts sent/dropped 1769202/0, interface broadcasts 3579215
```

In Example 3-131, the *show ip routes* output displays that the routes are stable in the routing table and that the timers are at a value lower than 30 seconds.

**Example 3-131**  
*show ip routes Output Reveals Stable RIP Routes*

```
Hub# show ip route rip
R 155.155.0.0/16 [120/1] via 131.108.1.1, 00:00:07, Serial0
R 166.166.0.0/16 [120/1] via 131.108.1.1, 00:00:07, Serial0
```
This page intentionally left blank
Symbols

%OSPF-4-BADLSATYPE error messages, 529

Numerics

128-bit addressing scheme, IPv6, 5
2-way state, OSPF neighbors, 336
   getting stuck, 398–400
32-bit addressing scheme, IPv4, 5
   classes, 5–7
   private address space, 7–8
   subnetting, 8

A

ABRs (area border routers), generating default summary routes, 326–327
access lists
   distribute lists
      IGRP uninstalled routes, 149–150
      unadvertised IGRP routes, 173–174
filtering BGP traffic
   unfiltered masked routes, 831–835
   unfiltered subnets, 828–830
inbound interfaces, blocked RIP broadcast, 63–65
source address, blocked RIP route installation, 60–63
unintentional TCP packet blockages, 741–742
Acknowledgment field (EIGRP packets), 216
active routes (EIGRP), 213
Active state (BGP-4), 663
AD (administrative distance), 24, 661
Address Resolution Protocol (ARP), 5
   addresses
      CLNP, 548
      NSAPs, 536
      defining, 551
      format, 549–551
addressing, 4
classless, 7
CLNP NSAPs, 536, 548
   defining, 551
   format, 549–551
hop-by-hop destination-based forwarding, 4
IPv4
   CIDR, 10
   classes, 5–7
   private address space, 7–8
   subnet, 8
media independence, 5
adjacencies, 334–335
   2-way state, 336
   Attempt state, 336
   Down state, 335
   ES-IS, 538, 589
      formation in IS-IS network, 605
   Exchange state, 337
   Exstart state, 336
   Full state, 338
   Init state, 336
   IS-IS, 537–540, 587–589
      absence of, 590–596
      INIT state, 596–605
   Loading state, 338
OSPF
   optional capability mismatches, 370–372
   Stuck in 2-WAY state, 398–400
   Stuck in ATTEMPT, 383–386
   Stuck in EXSTART/EXCHANGE state, 401–417
   Stuck in INIT, 387–398
   Stuck in LOADING state, 417–422
administrative distance, 24, 661
advanced distance vector routing protocols
   EIGRP
      dial backup, 286–290
      error messages, 291
      mismatched AS numbers, 239
      mismatched K values, 237
      mismatched masks, 235–237
      neighbor relationships, 227–232
      redistribution, 280–286
      route advertisement, 250–256
      route flapping, 271–275
route installation, 264–270
route summarization, 276–280
SIA errors, 240–250
uncommon subnets, 233–235
unexpected metrics, 259–263
unexpected route advertisement, 257–259

Advertising
BGP-4 routes, 668–670
synchronization rule, 671–672

IGRP routes
broadcast capability, 178–180
distribute lists, 173–174
misconfigured neighbor statement, 180–181
misconfigured network statement, 169–171
network interface, 175–176
outgoing interface, 171–172
passive outgoing interface, 176–178
split horizon, 184, 187–188
troubleshooting, 169
VLSM, 182–184

RIP routes, 86
blocked routes, 91–93
broken multicast capability, 96–98
down network interface, 93–94
down outgoing interface, 89–91
misconfigured neighbor statement, 99–100
missing network statement, 87–89
passive outgoing interface, 95–96
split horizon, troubleshooting, 102, 105–106
VLSM routes, troubleshooting, 100–102

Advertising Router field
OSPF link-state request packets, 300
LSAs, 303
AFI (address family identifier), 42
aggregate-address command, configuring BGP route origination, 747–749

Area 0, 315
Area ID field (OSPF packets), 297
areas, 315–318
IS-IS, 536–537
hierarchical routing, 541
normal, 319

NSSAs, 321–324
configuring, 322–324
default routes, 326–327
filtering Type 7 LSAs, 326
injecting external routes, 325–326
totally NSSAs, 324–326
stub, 319–320
totally stubby, 321

ARP (Address Resolution Protocol), 5

AS (autonomous system), 659
confederations, designing, 758–761
inbound traffic flows, 812
underutilized links, 813–818
outbound IP traffic flows, 790
asymmetrical routing, 802–806
dual-homed connections, 798–802
reachability, 795–798
single exit point, 791–795

AS confederations, 711–712

AS_PATH attribute
filtering, 835
policy control, 682, 684–685

ASSEQUENCE attribute, 685

AS_SET attribute, 685

ASBR (autonomous system boundary router), 660
assert mechanism, 640
PIM dense mode, 631–632
assigning default IGRP metric for redistribution, 191–194
asymmetrical routing, 802–806
Attached Bit field (LSPs), 554
Attached Router field (Network LSAs), 307
ATTEMPT state, OSPF neighbors, 336
getting stuck, 383–386

attributes (BGP)

AS_PATH
filtering, 835
policy control, 682–685

COMMUNITIES, policy control, 697–699
interpreting from command output, 688–690
LOCAL_PREF policy control, 675–676

MED
infinite setting, 777
policy control, 677–682

NEXT_HOP, policy control, 685
ORIGIN, policy control, 685
authentication
IS-IS, 548
keys, 69
null authentication, 364
RIP, 42
Authentication field (OSPF packets), 297
auto-cost reference-bandwidth command, 305
Autonomous System Number field, EIGRP packets, 216
autosummarization
EIGRP, 219–220
IP routes into Level 1/Level2, 573–574
missing RIP subnetted routes, 106–109
RIP, 43
AuType field (OSPF packets), 297
avoiding routing loops, split horizon, 130

B
backbone
  indication LSAs, 332–333
  IS-IS networks, 537
  virtual links, 316–318
Backup Designated Router field, OSPF Hello packets, 299
backup interface command, 194
backup links, dialup, 194–198
Bad LSA type error messages, 529
bandwidth, calculating IGRP metric, 128–129
behavior, IGRP, 131
best-path calculation, 661, 668
  BGP, 820–827
  BGP-4, 713–716
  IS-IS SPF algorithm, 558
BGP-4 (Border Gateway Protocol), 659
  AD, 661
  advertising routes, 668–670
  AS confederations, 711–712
  AS-PATH, filtering, 835
  attributes, interpreting from command output, 688–690
  best-path calculation, 661, 713–716, 820
    selection of lowest MED value, 824–827
    selection of wrong path, 821–823
cold potato, 661
confederations, designing, 758–761
EBGP multihop, misconfiguration, 736–740
Established state, 663
extended access lists, unfiltered masked routes, 831–835
external neighbor relationships
  incorrect IP address assignment, 731–732
  IP connectivity, 728
  Layer 2 problems, 729–731
  nondirectly connected, 732–733
hot potato, 661
inbound traffic flows, 812
  underutilized links, 813–818
internal neighbor relationships, route propagation, 754–762
load balancing, dual-homed outbound traffic, 806–812
neighbor relationships, 663–664, 659
  advertised routes, 726
  external, 665–667
  internal, 667
  statistics, displaying, 726
nondirectly connected external neighbor relationships, missing routing table addresses, 733–736
peers, 659–660
policies, 661
policy control, 672–674
  AS_PATH attribute, 682–685
  communities, 697–699
  distribute lists, 695–696
  filter lists, 695
  LOCAL_PREF attribute, 675–676
  MED attribute, 677–682
  NEXT_HOP attribute, 685
  ORF, 700–702
  ORIGIN attribute, 685
  prefix lists, 696
  route maps, 690–694
  WEIGHT knob, 686–688
policy routing, outbound IP traffic flows, 790–806
private peering, 660
protocol specifications, 662
public peering, 660
RFC 1771, 662
route dampening, 702–706
route origination
- classful network advertisements, 749–751
- misconfiguration, 746–749
- misconfigured distribute lists, 752–754
- missing routing table entries, 743–746
route reflection, 707–711, 777
- client-to-client reflection, 780–782
- identical cluster IDs, 785, 788–790
- misconfiguration, 779–780
- peer groups, 783–785
routing table
- uninstalled EBGP-learned routes, 771–777
- uninstalled IBGP-learned routes, 763–771
- standard access lists, unfiltered subnets, 828–830
- synchronization rule, 671–672
- synchronization, disabling, 766
BGP feed, 659
Bit B field (router LSAs), 305
Bit E field (external LSAs), 313
Bit E field (router LSAs), 304
Bit V field (router LSAs), 304
black holes, 671–672
boundaries, EIGRP queries, 220
broadcast addresses, 12
broadcast capability, unadvertised IGRP routes, 178–180
broadcast links, IS-IS, 536
broadcast mode (NBMA), 329–330

causes of uninstalled RIP routes
- access lists blocking RIP broadcast, 63–65
- access lists blocking source address, 60–63
- discontiguous networks, 71–74
- distribute list incoming routes, 58–60
- equal cost paths, 83–86
- hop count exceeded, 81–83
- incompatible RIP version, 65–68
- incorrect network statement, 53–56
- invalid source, 74–76
- Layer 1/2 down, 56–58
- Layer 2 problems, 76–78
- mismatched authentication key, 68–71
- offset value too high, 79–81
characteristics
- normal areas, 319
- NSSAs, 322
- stub areas, 320
- totally stubby areas, 321
Checksum field
- EIGRP packets, 216
- IGMP packets, 636
- LSPs, 554
- OSPF packets, 297
checksum operation, LSAs, 303
CIDR (Classless Interdomain Routing), 7, 10
classes of IP addresses, 5–7
classful networks
- masks, 8
- redistribution into BGP, 749–751
- subnetting, 8
classful routing protocols, 29
- versus classless, 15
classless addressing, 7
CIDR, 10
classless routing protocols versus classful, 15
clear isis command, 587
clients, route reflection, 757
client-to-client reflection, 780–782
CLNP (Connectionless Network Protocol), 534, 586
addressing, 548
- IS-IS, 586
- NSAPs, format, 549–551
- NSAPs, defining, 551
CLNS (connectionless network services), 533
C
clusters
identical cluster IDs between RR, 785, 788–790
route reflector client/servers, designing, 757–758
cold potato routing, 661, 682
cold potato routing, 661, 682
commands
aggregate-address, configuring BGP route
origination, 747–749
auto-cost reference-bandwidth, 305
backup interface, 194
clear isis, 587
debug ip bgp update, 727
debug ip bgp updates, 727
debug ip rip, 35, 55
ip default-network, 132, 221
ip summary-address eigrp, 219
match as_path, implementing in route maps, 692–693
match community, implementing in
route maps, 692
match ip address, implementing in
route maps, 691
output, interpreting BGP attributes, 688, 690
ping clns, 617–619
set as-path prepend, implementing in route
maps, 693
set community, implementing in
route maps, 693
set local preference, implementing in route
maps, 694
set metric command, implementing in route
maps, 694
show clns interface, 564, 586
show clns neighbors, 586
show clns neighbors detail, 563
field definitions, 588
show clns protocol, 562–563
show ip bgp, 726
show ip bgp neighbor, 726
show ip bgp neighbors, 726–727
show ip bgp summary, 726
show ip eigrp neighbor, 210
show ip eigrp topology active, 242–245
show ip protocols, 56
show ip route, 12, 56, 134, 142
show isis database, 586
show isis topology, 565, 586
timers basic, 130
traceroute, 617–619
variance, 133–134, 221–223
community, BGP-4 policy control, 697–699
command
classless and classful routing protocols, 15
interior and exterior gateway protocols, 15, 17–18
IPv4 and IPv6, 5
link-state and distance vector protocols, 18
TCP/IP and OSI reference model, 3
Type 7 and Type 5 LSAs, 322
confederations (BGP), designing, 758–761
configuring
EIGRP manual summarization, 219
IS-IS
ATM connectivity, 566–568
autosummarization, 573–574
case study, 619–622
default route advertisement, 569
IP routing, 560, 566–573
redistribution, 570–573
routing on point-to-point links, 559–566
NSSA areas, 322–324
virtual links, 316–318
Connect state (BGP-4), 663
connectivity
external BGP neighbor relationships, 728
IS-IS
adjacencies, 587–605
ping clns command, 617–619
traceroute command, 617–619
constant SPF calculations in OSPF networks,
troubleshooting, 518–528
convergence, 129
BGP networks, peer groups, 783–785
diffused computation, 213
distance vector protocols, 19–20
EIGRP, 207
query process, 220
local computation, 213
cost
IS-IS default metric, 546
OSPF metric calculation, overriding, 305
“could not allocate route id” error messages, 529
count to infinity, 29
distance vector protocols, 21
CPU utilization, displaying IS-IS statistics, 613–615
CPUHOG messages, OSPF, 499–503
CSNP Interval timer, IS-IS link-state database, 557
CSNPs (complete sequence number packets), 556

D
dampening routes, modifying parameters, 771–774
data-forward process, addressing, 4
datagram delivery service model, 3
DBD (Database Description) packets, OSPF, 299
Sequence Number field, 300
DC bit, 332
dDR (dial-on-demand routing)
IGRP, 194
dial backup, 194–198
OSPF, 503–516
RIP, 116–122
debug ip bgp update command, 727
debug ip bgp updates command, 727
debug ip rip command, 35, 55
debugging SPF problems (IS-IS), 615–616
decision process, IS-IS SPF algorithm, 558
default routes
EIGRP, 221
IGRP, 132–133
unadvertised candidates, 188–191
OSPF
unadvertised, 450–462
NSSAs, 326–327
RIP, 39–40
unadvertised, 450–462
defining
metric for IGRP redistribution, 191–194
NSAPs, 551
delay, calculating IGRP metric, 128–129
delay metric, IS-IS, 546
demand circuits (OSPF), 331–334
dense mode (PIM), 625, 630–631
assert mechanism, 631–632
prune mechanism, 631
troubleshooting, 646–651
Designated Router field (OSPF Hello packets), 299
designing
BGP confederations, 758–761
route reflector model, 757–758
devices
interfaces, link-based addressing, 5
Layer 2 media, troubleshooting uninstalling
IGRP routes, 161–163
dial backup
EIGRP, 286–290
IGRP, 194–198
diffused computation, 213
Dijkstra algorithm, 295
direct inspection of LSPs, 606–607
directed broadcasts, 12
directly connected external BGP neighbors, IP
connectivity, 728–729
DIS (designated intermediate system), 539
disabling BGP synchronization, 766
discontiguous networks
EIGRP, 252–253
routing behavior, 218–219
RIP, 36–37
troubleshooting behavior, 71–74
uninstalled IGRP routes, 155–158
displaying
BGP neighbor statistics, 726
IS-IS CPU utilization statistics, 613–615
IS-IS link-state database, 565
IS-IS topology, 565
prefixes, assigned attributes, 726
distance vector protocols
convergence, 19–20
counting to infinity, 21
holddown, 21
IGRP, 133–134
behavior, 131
default routes, 132–133
defining metric for redistribution, 191–194
dial backup links, 194–198
flapping routes, 198–201
metrics, 127–129
packets, 131
split horizon, 130
split horizon with poison reverse, 130
timers, 129–130
unadvertised default route candidates, troubleshooting, 188–191
uninstalled equal-cost paths, troubleshooting, 166–168
uninstalled routes, troubleshooting, 142–188
variance, 201–204
loop avoidance, 20–21
metrics, 19
periodic updates, 22
poison reverse, 22
RIP
authentication, 42
compatibility issues, 43
DDR, 116, 118–122
default routes, 39–40
discontiguous networks, 36–37
flapping routes, 122–124
multicast, 42
Next Hop fields, 41
packet behavior, 31
receiving updates, 33–35
redistribution, 113–116
route advertisements, 86–106
    passive outgoing interface, 96
route tag field, 40–41
route installation, 52–85
sending updates, 31–33
split horizon, 30
    with poison reverse, 31
subnet masks, 41
summarization, 109–113
timers, 30
uninstalled routes, 52
version field, 43
VLSM, 37–39
split horizon, 22
triggered updates, 22
versus link-state, 18
distribute lists, 58
BGP-4
    misconfiguration, 752–754
    policy control, 695–696
blocked routes, troubleshooting, 91–93
IGRP uninstalled routes, 149–150
incoming routes, blocked RIP route installation, 58–60
misconfiguration, 251–252
unadvertised IGRP routes, troubleshooting, 173–174
dotted-decimal notation, IP address representation, 7
Down state, OSPF neighbors, 335
Doyle, Jeff, 214
dropped packets, IGRP, 199–201
DRs (designated routers), network LSAs, 307–308
DUAL (Diffusing Update Algorithm), 207, 211, 240–241
    active routes, 213
    FC, 211
    FD, 211
    FSM, 213–214
    feasible successors, 212
    passive routes, 213
    RD, 211
    successors, 211
dual addressing scheme, IS-IS, 586
dual homing
    load balancing, 806–812
    outbound BGP traffic flows, 798–802
duplicate router IDs, EIGRP, 268–270
dynamic routing, 4, 11
    classless versus classful, 15
    interior versus exterior gateway, 15–18
    link-state versus distance vector protocols, 18
    multicast versus unicast, 12–14
    versus static routing, 11

E

EBGP (External BGP), 660
multihop
    misconfiguration, 736–740
    resolving nondirectly connected neighbor relationships, 732
unreachable next hop, 774–777
underutilized links between two ASes, 813–818
uninstalled routes, 771
dampened BGP routes, 771–774
infinite MED value, 777
EIGRP (Enhanced Interior Gateway Routing Protocol), 17, 207
convergence, 207
diffused computation, 213
local computation, 213
default routes, 221
dial backup, 286–290
DUAL, 207, 211, 240–241
active routes, 213
FC, 211
FD, 211
feasible successors, 212
passive routes, 213
RD, 211
successors, 211
error messages, 291
hold time, 209
metrics, 208–209
neighbor relationships, 209–210, 227
log, reviewing, 228–229
mismatched AS numbers, 239
mismatched k values, 237
mismatched masks, 235–237
one-way, 230–232
SIA errors, 240–250
uncommon subnets, 233–235
packets, TLV, 216–217
redistribution, 280–286
reliable packets, 214
route advertisement, 250
discontiguous networks, 252–253
misconfigured distribute lists, 251–252
split-horizon, 253–256
unexpected advertisements, 257–259
unexpected metrics, 259–263
route flapping, 271–275
route installation, 264–270
route summarization, 276–280
routing behavior, 218–219
RTP, 214
summarization, 219–220
unequal-cost load balancing, 221–223
empty neighbor lists, OSPF, 351–383
empty routing tables (IGRP), troubleshooting, 142–166
Enhanced Interior Gateway Routing Protocol. See EIGRP
equal-cost paths, route installation
IGRP, 166–168
RIP, 83–86
error messages, EIGRP, 291
error metric, IS-IS, 546
ES-IS (End System-to-Intermediate System), 589
adjacencies, 538
misidentification in IS-IS networks, 605
Established state (BGP-4), 663–664
route advertisements, 668–670
synchronization rule, 671–672
establishing IGRP dial backup, 194–198
Exchange state, OSPF neighbors, 337
getting stuck, 401–417
expense metric, IS-IS, 546
Exstart state, OSPF neighbors, 336
getting stuck, 401–417
extended access lists
BGP filtering, unfiltered masked routes, 831–835
dump ip bgp update command output, 727
uninstalled IGRP routes, troubleshooting, 151–155
exterior gateway protocols, versus interior, 15–18
external LSAs (Type 5), 302, 313–315
external metrics, IS-IS, 547
external neighbor relationships, BGP-4, 665–667
incorrect IP address assignment, 731–732
IP connectivity, 728
Layer 2 problems, 729–731
nondirectly connected, 732–733
misconfiguration, 736–740
missing routing table addresses, 733–736
external routes
injecting into NSSAs, 325–326
OSPF, unadvertised, 441–449
redistribution into IS-IS, 570–573
summarization, 497–499
uninstalled, 479–487
Hybrid routing protocols 857

F

FC (feasible condition), 211
FD (feasible distance), 211
feasible successor routes, 17, 212
field definitions
EIGRP packets, 216–217
external LSAs, 313
IGMP packets, 635
IS-IS packets, TLVs, 543–545
LSP headers, 553–555
show clns neighbors command output, 588
OSPF
Hello packets, 298–299
LSA headers, 303–304
packets, 296
summary LSAs, 310
filter lists, BGP-4 policy control, 695
filtering traffic
BGP traffic
    AS_PATH, 835
    unfiltered masked routes, 831–835
    unfiltered subnets, 828–830
extended access lists, troubleshooting
uninstalled IGRP routes, 151–155
Type 7 LSAs, 326
Flags field (EIGRP packets), 216
flapping routes
dampening, 702–706
EIGRP, 271–275
IGRP, 198–201
IS-IS, 612–616
RIP, 122–124
flooding, 552
IS-IS, 555–557
flush timers
IGRP, 130
RIP, 30
format of NSAPs, 549–551
Forwarding Address field (External LSAs), 313
forwarding packets
    high-speed, 25
    multicast, 12–14
full feed, 659
Full state, OSPF neighbors, 338

gateway of last resort, IGRP, 188
generating default summary routes, 326–327
generic format, IS-IS packets, 543–545
Group Address field, IGMP packets, 636

H

headers
    LSAs, 303–304
    LSPs, 553–555
    OSPF packets, 296
Hello Interval field, OSPF Hello packets, 298
hellos
    IIHs, 538–539
    OSPF, 297–299
hierarchical Route-Reflection, 709
hierarchical routing, IS-IS, 541
hold time
    BGP-4, 663
    EIGRP, 209
holddown, distance vector protocols, 21
hold-down timers
    IGRP, 130
    RIP, 30
hop count, 29
troubleshooting RIP route installation, 81–83
hop-by-hop destination-based forwarding
    mechanism, 4
hot potato, 661
hybrid routing protocols, EIGRP, 207
coreference, 207
default routes, 221
DUAL, 207, 211–213
IP Internal Route TLV (EIGRP), 216–217
metrics, 208–209
neighbor relationships, 209–210
packet fields, 216–217
query process, 220
routing behavior, 218–219
RTP, 214
summarization, 219–220
unequal-cost load balancing, 221–223
I Bit field (OSPF DBD packets), 299
IBGP (Internal BGP), 660
    AS confederations, 711–712
    black holes, 671–672
    neighbor relationships
        client-to-client reflection, 780–782
        misconfiguration, 779–780
    route reflection, 707–711
    uninstalled routes, 763–766
        unreachable next-hops, 766–771
identical cluster IDs in RR session, troubleshooting, 785, 788–790
identifying routers attached to transit links, 308
Idle state (BGP-4), 663
IETF standardization of IS-IS, 535
IGMP (Internet Group Management Protocol)
    version 1, 626
    version 2, 627
        joins, 643–645
        leave mechanism, 627–628
        packets, 635
        querier election mechanism, 627
IGRP (Interior Gateway Routing Protocol)
    behavior, 131
    DDR, 194
        dial backup links, 194–198
        default routes, 132–133
        intermediate media problems, 142
        load balancing, 168
        metrics, 127–129
        packets, 131
    redistribution
        into NSSA area, 321
        metric, defining, 191–194
    route flapping, 198–201
    route installation, 142–166
    split horizon, 130
    split horizon with poison reverse, 130
    timers, 129–130
    unadvertised default route candidates, 188–191
    unequal-cost load balancing, 133–134
    uninstalled equal-cost paths, 166–168
    uninstalled routes, sender problems, 142, 168–188
    variance, 201–204
IIHs (intermediate system-to-intermediate system hellos), 538–539
improving BGP network convergence, 783–785
inbound BGP traffic flows, 812
    underutilized links, 813–818
incompatible RIP versions, troubleshooting RIP
    route installation, 65–68
incorrect network statements, RIP route installation, 53–56
indication LSAs, 332–333
INIT state, 336
    getting stuck, 382, 387–398
    IS-IS adjacencies, 596–605
injecting external routes into NSSAs, 325–326
installing RIP routes, 52
    access lists
        blocked RIP broadcast, 63–65
        blocked source addresses, 60–63
discontiguous networks, 71–74
distribute lists, blocked RIP routes, 58–60
    equal-cost paths, 83–86
    hop count exceeded, 81–83
    incompatible RIP version, 65–68
    incorrect network statements, 53–56
    invalid sources, 74–76
    Layer 2 problems, 76–78
    line protocols, down state, 56–58
    mismatched authentication key, 68–71
    offset list value too high, 79–81
Integrated IS-IS. See IS-IS
interarea routes, summarization, 495–496
interesting traffic, 117
Interface MTU field, OSPF DBD packets, 299
    interfaces
        link-based addressing, 5
        olist, 630–631
Interior Gateway Protocols
    EIGRP, 17
        versus exterior, 15–18
    intermediate media problems, 142
    internal metrics, IS-IS, 547
    internal neighbor relationships
        BGP-4, 667
        route propagation, 754–761
        synchronization, 761–762
unintentional TCP packet blockages, 741–742
interpreting BGP attributes from command output, 688–690
Invalid LSA error messages, 529
invalid sources, troubleshooting uninstalled routes
  IGRP, 159–161
  RIP, 74–76
invalid timers
  IGRP, 129
  RIP, 30
IP addressing
  broadcast addresses, 12
  CIDR, 10
  dynamic routing, 11
  classless versus classful routing protocols, 15
  interior versus exterior gateway, 15–18
  link-state versus distance vector protocols, 18–24
  unicast versus multicast routing, 12–14
  subnets, 12
ip default-network command, 132, 221
IP Internal Route TLV (EIGRP), 216–217
IP network numbers, 6
IP prefix, 659
IP routing, IS-IS configuration, 560, 566–573
  ATM connectivity, 566–568
  autosummarization, 573–574
  default route advertisement, 569
  point-to-point links, 559–566
  redistribution, 570–573
ip summary-address eigrp command, 219
IPv4
  address classes, 5–7
  private address space, 7–8
  subnetting, 8
  versus IPv6, 5
IS Type field (LSPs), 555
IS-IS (Intermediate System-to-Intermediate System), 533–535
  adjacencies, 538–540, 587–589
  absence of, 590–596
  INIT state, 596–605
  misidentified ES-IS adjacencies, 605
  three-way reliable, 540
  areas, 536–537
  authentication, 548
  backbone, 537
broadcast links, 536
CLNP addressing, 548, 586
  NSAP format, 549–551
  NSAPs, defining, 551
DIS, 539
errors, 616–617
eexternal routes, redistribution to Level 1, 611
flapping routes, 612–616
flooding, 552
hierarchical routing, 541
IETF standardization, 535
IP routing
  ATM configuration, 566–568
  autosummarization, 573–574
  configuring, 560, 566–573
  default route advertisement, 569
  point-to-point links, 559–566
  redistribution, 570–573
level 1, 535
links, 536–537
link-state database, 552–555
  displaying, 565
  flooding, 555–557
  synchronization, 555–557
  update timers, 556–557
LSPs, header fields, 553–555
metrics, 545–548
  external, 547
  internal, 547
nodes, 536–537
NSAPs, 536
packets, 542
  generic format, 543–545
PDUs, 542
ping clns command, 617–619
point-to-point links, 536
PSN, 539
route advertisements, misconfiguration, 607–611
routing domains, 536
routing updates, 606–607
SNPs, 556
SPF algorithm, 558
  triggers, 614–615
traceroute command, 617–619
update process, 555
ISO CLNS (International Organization for Standardization Connectionless Networking Services)

- CLNP, 534
- IS-IS, 533–535
  - adjacencies, 537
  - areas, 536–537
  - ATM configuration, 566–568
  - autosummarization, 573–574
  - backbone, 537
  - broadcast links, 536
  - CLNP addressing, 548–551
  - default route advertisement, 569
  - DIS, 539
  - ES-IS adjacencies, 538
  - flooding, 552
  - header fields, 553–555
  - hierarchical routing, 541
  - IETF standardization, 535
  - IP routing configuration, 559–573
  - IS-IS adjacencies, 538–540
  - Level 1, 535
  - links, 536–537
  - link-state database, 552–557
  - metrics, 545–548
  - nodes, 536–537
  - NSAPs, 536
  - packets, 542–545
  - point-to-point links, 536
  - PSN, 539
  - redistribution, 570–573
  - routing domains, 536
  - SPF algorithm, 558
  - three-way reliable adjacencies, 540
  - update process, 555
- SNPs, 556
- ISPs, BGP-4 659
  - advertising routes, 668–672
  - best path calculation, 713–716
  - external neighbor relationships, 665–667
  - internal neighbor relationships, 667
  - neighbor relationships, 663–664
  - policy control, 672–688
  - protocol specifications, 662
  - route dampening, 702–706
- IXP (Internet Exchange Point), 660

J-K-L

- join mechanism
  - IGMP version2, 627–628, 643–645
  - PIM sparse mode, 633
- knobs, policy control, 686–688
- Layer 1 (OSI), IGRP uninstalled routes, 147–149
- Layer 2 (OSI)
  - connectivity between external BGP neighbors, 729–731
  - IGRP uninstalled routes, 161–163
  - RIP route installation, 76–78
- layered protocol suites, TCP/IP, 3
- leave mechanism, IGMP version 2, 627–628
- Length field (LSAs), 304
- Level 1 LAN IIHs, 539
- Level 2 LAN IIHs, 539
- line protocols
  - RIP route installation, 56–58
  - uninstalled IGRP routes, troubleshooting, 147–149
- Link Data field (router LSAs), 305
- link flaps, 506
- Link ID field (router LSAs), 305
- link-based addressing, 5
- links
  - IS-IS, 536–537
  - OSPF, 305
- link-state acknowledgment packets (OSPF), 301
- link-state database (IS-IS), 552–555
  - displaying, 565
  - flooding, 555–557
  - synchronization, 555–557
  - update timers, 556–557
- Link-State ID field
  - LSAs, 303
  - OSPF link-state request packets, 300
- link-state protocols, 23
- IS-IS, 533, 535
  - adjacencies, 537, 587–605
  - areas, 536–537
  - ATM configuration, 566–568
  - authentication, 548
  - autosummarization, 573–574
  - backbone, 537
broadcast links, 536
CLNP addressing, 548–551, 586
configuring, case study, 619–622
confusion with ES-IS adjacencies, 605
default route advertisement, 569
DIS, 539
displaying CPU utilization statistics, 613–615
errors, 616–617
ES-IS adjacencies, 538
external routes, redistribution into Level 1, 611
flapping routes, 612–616
flooding, 552
header fields, 553–555
hierarchical routing, 541
IETF standardization, 535
IP routing configuration, 559–573
IS-IS adjacencies, 538–540
Level 1, 535
links, 536–537
link-state database, 552–557
metrics, 545–548
nodes, 536–537
NSAPs, 536
packets, 542–545
ping clns command, 617–619
point-to-point links, 536
PSN, 539
redistribution, 570–573
route advertisements, 607–611
routing domains, 536
routing updates, 606–607
SNPs, 556
SPF algorithm, 558
SPF process triggers, 614–615
three-way reliable adjacencies, 540
trace route command, 617–619
update process, 555
metrics, 24
OSPF, 295
“%OSPF-4-BADLSATYPE error messages, 529
adjacencies, 334–338
areas, 315–320
constant SPF calculations, troubleshooting, 518–528
“could not allocate route id” error messages, 529
CPUHOG messages, 499–503
DDR, 503–516
debugs, CPU utilization, 341
demand circuits, 331–334
Dijkstra algorithm, 295
empty neighbor lists, 351–383
external LSAs, 313–315
LSAs, 302–304
multiaccess media, 327–328
NBMA media, 329–331
network LSAs, 307–308
NSSAs, 321–326
null authentication, 364
“OSPF-4-ERRRCV” error messages, 529–531
packets, 295–301
point-to-point media, 328
redistribution, 488–494
router LSAs, 304–306
Stuck in 2-WAY state, 398–400
Stuck in ATTEMPT state, 383–386
Stuck in EXSTART/EXCHANGE state, 401–417
Stuck in INIT, state, 387–398
Stuck in LOADING state, 417–422
summarization, 494–499
summary LSAs, 309–312
totally stubby areas, 321
unadvertised default routes, 450–462
unadvertised external routes, 441–449
unadvertised routes, troubleshooting, 422–429, 431
unadvertised summary routes, 432–440
uninstalled external routes, 479–487
uninstalled routes, 463–478
“unknown routing protocol” error messages, 528
virtual links, 316–317
versus distance vector, 18
link-state request packets (OSPF), 300
link-state update packets (OSPF), 301
load balancing
dual-homed outbound BGP traffic, 806–812
IGRP, 168
  uninstall equal-cost paths, 166–168
  variance, 201–204
Loading state, OSPF neighbors, 338
  getting stuck, 417–422
local computation, 213
LOCAL_PREF attribute, policy control, 675–676
loop avoidance, distance vector protocols, 20–21
loopback interfaces, BGP peering, 732–733
LS Age field (LSAs), 303
LS Checksum field (LSAs), 303
LS Sequence Number field (LSAs), 303
LS Type field (OSPF link-state request packets), 300
LS Type field (LSAs), 303
LSA Header field (OSPF DBD packets), 300
LSAs, 295, 302–303
  external LSAs (Type 5), 313–315
  header fields, 303–304
  indication LSAs, 332–333
  network LSAs (Type 2), 307–308
  NSSA, P bit, 324
  periodic, 332
  router LSAs (Type 1), 304–306
  summary LSAs (Type 3/4), 309–312
  fields, 310
  Type 7, filtering, 326
LSP Database Overload Bit field (LSPs), 555
LSP Identifier field (LSPs), 553
LSP number field (LSPs), 554
LSP Refresh Interval timer, IS-IS link-state
database, 557
LSP Retransmit Interval timer, IS-IS link-state
database, 557
LSPs (link-state packets), 553
  direct inspection, 606–607
  flooding, 552, 555–557
  header fields, 553–555
  TLVs, 547
  Transmission Interval timer, 557

M

M Bit field, OSPF DBD packets, 299
manual summarization, EIGRP, 219–220
  masks, 8
match as_path command, implementing in route
  maps, 692–693
match community command, implementing in route
  maps, 692
match ip address command, implementing in route
  maps, 691
Maxage timer, IS-IS link-state database, 556
Maximum Response Time field, IGMP packets, 636
MED (multi-exit discriminator) attribute
  best-path selection, 824–827
  infinite value, 777
  policy control, 677–682
media independence of IP addressing, 5
media types
  Layer 2, uninstalled IGRP routes,
  troubleshooting, 161–163
OSPF
  demand circuits, 331–334
  multiaccess media, 327–328
  NBMA media, 329–331
  point-to-point media, 328
messages
  “%OSPF-4-BADLSATYPE
    Invalid lsa Bad LSA type”, 529
  “could not allocate route id”, 529
  CPUHOG (OSPF), 499–503
  “OSPF-4-ERRRCV”, 529–531
  “unknown routing protocol”, 528
  PIM, 638–640
Metric field (router LSAs), 305
metric field (summary LSAs), 310
metrics, 29–30
  cost (OSPF), overriding, 305
distance vector protocols, 19
  EIGRP, 208–209
  IGRP, 127–129
defining for redistribution, 191–194
  IS-IS, 545–548
  link-state protocols, 24
misconfiguration
access lists, troubleshooting uninstalled IGRP routes, 149–155

BGP
distribute lists, 752–754
neighbor addresses, 731–732
route origination, 746–749

IS-IS
adjacencies, 589–605
case study, 619–622
redistribution, 611
route advertisements, 607–611
neighbor statements, 99–100
unadvertised IGRP routes, 180–181
network statements, unadvertised IGRP routes, 169–171
route reflection
IBGP neighbors, 779–780
identical cluster IDs, 785–790
routers, troubleshooting uninstalled IGRP routes, 143–146
mismatched AS numbers (EIGRP), 239
mismatched authentication key, troubleshooting RIP route installation, 68–71
mismatched K values, EIGRP, 237
mismatched masks, EIGRP, 235–237
mismatched sender AS number, troubleshooting uninstalled IGRP routes, 163–166
modifying BGP dampening parameters, 771–772, 774

MS Bit field, OSPF DBD packets, 300
multiaccess media, OSPF networks, 327–328
multicast, 625

IGMP
joins, 643–645
leave mechanism, 627–628
packet format, 635
querier election mechanism, 627
version 1, 626
version 2, 627

PIM
dense mode, 625, 630–632, 646–651
messages, 638–640
packets, 636–638
sparse mode, 625, 632–634, 651–656
RIP addresses, 42
RPF, 628–630

multicast routing, versus unicast, 12–14
multihoming, 552, 660

N

NBMA media, OSPF networks, 329
broadcast mode, 329–330
point-to-multipoint mode, 331
point-to-point mode, 330
Neighbor field (OSPF Hello packets), 299
neighbor relationships
BGP-4, 663–664
external, 665–667
external, incorrect IP address assignment, 731–732
internal, 667
route advertisements, 668–672
EIGRP, 209–210, 227
mismatched AS numbers, 239
mismatched K values, 237
mismatched masks, 235–237
one-way, 230, 232
reviewing documented changes, 228–229
SIA error, 240–250
uncommon subnets, 233–235

external (BGP)
IP connectivity, 728
Layer 2 problems, 729–731
IBGP, client-to-client reflection, 780–782
internal
misconfiguration, 779–780
route propagation, 754–762
unintentional TCP packet blockages, 741–742

OSPF
2-way state, 336
Attempt state, 336
Down state, 335
empty neighbor lists, 351–383
ES-IS, 538
Exchange state, 337
Exstart state, 336
Full state, 338
Init state, 336
IS-IS, 538–540
Loading state, 338
Stuck in 2-WAY state, 398–400
Stuck in ATTEMPT, 383–386
Stuck in EXSTART/EXCHANGE state, 401–417
Stuck in INIT, 387–398
Stuck in LOADING state, 417–422
unadvertised default routes, 450–462
unadvertised external routes, 441–449
unadvertised summary routes, 432–440
network convergence time, 129
network interfaces, unadvertised IGRP routes,
troubleshooting, 175–176
network LSAs (Type 2), 302, 307–308
Network Mask field
  External LSAs, 313
  Network LSAs, 307
  OSPF hello packets, 298
  summary LSAs, 310
Next Hop field (RIP packets), 41
NEXT_HOP attribute, policy control, 685
nodes, IS-IS, 536–537
nondirectly connected external BGP neighbor
  relationships, 732–733
  misconfiguration, 736–740
  missing routing table addresses, 733–736
normal areas, 319
NSAPs (network service access points), 536
  defining, 551
  format, 549–551
NSSAs, 321–324
  configuring, 322–324
  default routes, 326–327
  injecting external routes, 325–326
  totally NSSAs, 324–326
Type 7 LSAs, 302
  filtering, 326
  P bit, 324
null authentication, 364
Null0 route, advertising, 670
Number of Links field (router LSAs), 305

doctets, IP address representation, 7
offset list values
  troubleshooting RIP route installation, 79–81
oilist, 630–631
one-way neighbor relationships, EIGRP, 230–232
Opcode field, EIGRP packets, 216
OpenConfirm state (BGP-4), 664
OpenSent state (BGP-4), 664
optional capability mismatch, 370–372
Options field
  OSPF DBD packets, 299
  OSPF Hello packets, 298–299
Options field (LSAs), 303
ORF, BGP-4 policy control, 700–702
ORIGIN attribute, policy control, 685
originating BGP routes
  classful network advertisements, 749–751
  misconfiguration, 746–749
  misconfigured distribute lists, 752–754
  missing routing table entries, 743–746
OSI reference model versus TCP/IP model, 3
OSPF, 295
%OSPF-4-BADLSATYPE error messages, 529
adjacencies, 334–335
  2-way state, 336
  Attempt state, 336
  Down state, 335
  Exchange state, 337
  Exstart state, 336
  Full state, 338
  Init state, 336
  Loading state, 338
  optional capability mismatches, 370–372
areas, 315–316, 318
  normal, 319
  NSSAs, 321–326
  stub, 319–320
  totally stubby, 321
backbone
  indication LSAs, 332–333
  virtual links, 316–317
“could not allocate route id” error messages, 529
DDR, 503–516
debugs, CPU utilization, 341
demand circuits, 331–334
Dijkstra algorithm, 295
external routes, summarization, 497–499
link types, 305
LSAs, 295, 302–303
  external LSAs (Type 5), 313–315
  header fields, 303–304
network LSAs (Type 2), 307–308
router LSAs (Type 1), 304–306
summary LSAs (Type 3/4), 309–312
multiaccess media, 327–328
NBMA media, 329
  broadcast mode, 329–330
  point-to-multipoint mode, 331
  point-to-point mode, 330
neighbor relationships
  empty neighbor lists, 351–383
unadvertised default routes, 450–462
unadvertised external routes, 441–449
unadvertised summary routes, 432–440
null authentication, 364
“OSPF-4-ERRRCV” error messages, 529–531
packets, 295–297
  DBD, 299
  fields, 296
  Hello, 297–299
  link-state acknowledgment, 301
  link-state request, 300
  link-state update, 301
point-to-point media, 328
redistribution, 488–494
SPF calculations, 518–528
Stuck in 2-WAY state, 398–400
Stuck in ATTEMPT, 383–386
Stuck in EXSTART/EXCHANGE state, 401–417
Stuck in INIT state, 387–398
Stuck in LOADING state, 417–422
summarization, 494–496
unadvertised routes, 422–431
uninstalled external routes, 479–487
uninstalled routes, 463–478
“unknown routing protocol” error messages, 528
outbound dual-homed BGP traffic, outbound traffic flows (BGP)
  asymmetrical routing, 802–806
dual-homed, 798–802
  load balancing, 806–812
reachability, 795–798
single exit point from AS, 791–795
outgoing interface, unadvertised IGRP routes, troubleshooting, 171–172
overriding OSPF metric calculation, 305

P

P bit, 324
packet drops, IGRP, 199–201
Packet Length field (OSPF packets), 297
packets, 3, 533
data-forwarding process, addressing, 4
EIGRP
  IP Internal Route TLV, 216–217
  Q count, 232
  query process, 220
  reliable, 214
  TLV, 216–217
forwarding, high-speed, 25
hop-by-hop destination-based forwarding, 4
IGMP Type field, 635
IGRP, 131
IIHs, 538–539
IS-IS, 542
generic format, 543–545
TLV fields, 543–545
LSPs
direct inspection, 606–607
flooding, 552
header fields, 553–555
multicast, 625
multicast routing, 12–14
OSPF, 295–297
  DBD, 299
  Hello, 297–299
  link-state acknowledgment, 301
  Link-State Request, 300
  link-state update, 301
PIM, 636–638
RIP
  AFI, 42
  Next Hop field, 41
SNPs, 556
TCP, unintentional blockages, 741–742
parameters for BGP dampening, modifying, 771–774
partial feed, 659
Partition Bit field (LSPs), 554
passive outgoing interfaces
  RIP route advertisement, 95–96
  unadvertised IGRP routes, 176–178
passive routes (EIGRP), 213
payload, 4
PDUs (protocol data units), 542
peer groups, improving BGP convergence, 783–785
peering
  between nondirectly connected external neighbors, missing routing table addresses, 733–736
BGP-4
  external neighbor relationships, 665–667
  internal neighbor relationships, 667
  route advertisements, 668–672
periodic LSAs, 332
periodic updates, distance vector protocols, 22
PIM (Protocol Independent Multicast)
  dense mode, 625, 630–631
  assert mechanism, 631–632
  troubleshooting, 646–651
IGMP joins, 643–645
messages, 638–640
packets, 636–638
sparse mode, 625, 632, 651, 654–656
discovery process, 632–633
join mechanism, 633
register process, 634
RPs, 632
ping clns command, 617–619
point-to-multipoint mode (NBMA), 331
point-to-point IIHs, 539
point-to-point links, IS-IS, 536
point-to-point media, OSPF networks, 328
point-to-point mode (NBMA), 330
point-to-point serial links, IS-IS configuration, 559–565
poison reverse, distance vector protocols, 22
poison updates, 130
policies, BGP, 661
policy control, 672–674
  AS_PATH attribute, 682–685
  communities, 697–699
  distribute lists, 695–696
  filter lists, 695
  LOCAL_PREF attribute, 675–676
  MED attribute, 677–682
  NEXT_HOP attribute, 685
  ORF, 700–702
  ORIGIN attribute, 685
prefix lists, 696
route maps, 690–694
  match as_path command, 692–693
  match community command, 692
  match ip address command, 691
  set as-path prepend command, 693
  set community command, 693
  set local preference command, 694
  set metric command, 694
WEIGHT knob, 686–688
policy routing (BGP), outbound IP traffic flows, 790
asymmetrical routing, 802–806
dual-homed connections, 798–802
reachability, 795–798
single exit point, 791–795
prefix lists, BGP-4 policy control, 696
prefixes
  advertising, 668–670
  synchronization rule, 671–672
assigned attributes, displaying, 726
origination
  classful network advertisements, 749–752
  distribute lists, 752–754
  misconfiguration, 746–749
prepending AS_PATH, 682–685
preventing routing loops
  DUAL, 207, 211
  active routes, 213
  FC, 211
  FD, 211
  feasible successors, 212
  passive routes, 213
  RD, 211
  successors, 211
private IPv4 address space, 7–8
private peering, 660
protocol header format, OSPF packets, 296
protocol specifications, BGP-4, 662
  external neighbor relationships, 665–667
  internal neighbor relationships, 667
  neighbor relationships, 663–664
protocol-independent commands, 24
prune mechanism, PIM dense-mode, 631
Pseudonode identifier field (LSPs), 554
PSN (pseudonodes), 539
PSNPs (partial sequence number packets), 556
public peering, 660

Q-R

Q count, 232
querier election mechanism, IGMP version 2, 627
queries
  EIGRP, 214, 220
  RIPs, 43
RD (reported distance), 211
reachability
  EBGP multihop next hops, 774–777
  IBGP next-hops, 766–771
  IS-IS
    ping clns command, 617–619
    traceroute command, 617–619
  IS-IS TLVs, 547
  outbound BGP traffic flows, 795–798
  RIP route installation, 52
receiver problems, IGRP uninstalled routes, 142
receiving updates, RIP, 33–35
redistribution
  EIGRP, 280–286
  external routes into IS-IS, 570–573
  IGRP metric, defining, 191–194
  into NSSAs, filtering Type 7 LSAs, 326
  IS-IS, misconfiguration, 611
  OSPF, 488–494
  RIP, 113–116

redundancy
  dial backup links
    EIGRP, 286–290
    IGRP, 194–198
  virtual links, 316
register message (PIM), 638
register process, PIM sparse mode, 634
reliable EIGRP packets, 214
Remaining Lifetime field (LSPs), 553
replies (EIGRP), 214
resolving EIGRP SIA errors, 240–250
reviewing EIGRP neighbor changes, 228–229
RFC 1771, synchronization rule, 671–672
RIDs (router IDs), 659
  best-path selection, 821–823
RIP (Routing Information Protocol)
  authentication, 42
  compatibility issues, 43
  DDR, 116–122
  default routes, 39–40
  discontiguous networks, 36–37
  flapping routes, 122–124
  hop count, 29
  metrics, 29–30
  missing subnetted routes, 106–109
  Next Hop field, 41
  packet behavior, 31
  packets, AFI, 42
  redistribution, 113–116
  route advertisements, 86
    blocked routes, 91–93
    broken multicast capability, 96–98
    down network interface, 93–94
    down outgoing interface, 89–91
    misconfigured neighbor statement, 99–100
    passive outgoing interface, 95–96
    sending RIP routes, 86
    split horizon, 102, 105–106
    VLSM routes, 100–102
  route installation, 52
  split horizon, 30–31
  subnet masks, 41
  summarization, 109–113
  timers, 30
uninstalled routes, causes of, 52
  blocked RIP broadcast, 63–65
  blocked source address, 60–63
  discontiguous networks, 71–74
  distribute list incoming routes, 58–60
  equal-cost paths, 83–86
  hop count exceeded, 81–83
  incompatible RIP version, 65–68
  incorrect network statements, 53–56
  invalid sources, 74–76
  Layer 2 problems, 76–78
  line protocol in down state, 56–58
  mismatched authentication key, 68–71
  offset list value too high, 79–81
updates
  receiving, 33–35
  sending, 31–33
version field, 43
VLSM, 37–39
RIP-2
  multicast, 42
  Route Tag field, 40–41
route advertisements
  EIGRP, 250
    discontiguous networks, 252–253
    misconfigured distribute lists, 251–252
    split-horizon, 253–256
    unexpected advertisements, 257–259
    unexpected metrics, 259–263
  IGRP, 168–188
  OSPF, troubleshooting unadvertised routes, 422–431
route dampening, 702–706
route flapping, EIGRP, 271–275
route installation
  EIGRP, 264
    summarization, 265–270
  OSPF
    uninstall external routes, 479–487
    uninstall routes, 463–478
route maps, BGP-4 policy control, 690–694
  communities, 697–699
  distribute lists, 695–696
  filter lists, 695
  match as-path command, 692–693
  match community command, 692
  match ip address command, 691
ORF, 700–702
  prefix lists, 696
  set as-path prepend command, 693
  set community command, 693
  set local-preference command, 694
  set metric command, 694
route origination (BGP)
  classful network advertisements, 749–751
  misconfiguration, 746–749
  misconfigured distribute lists, 752–754
  missing routing table entries, 743–746
route redistribution, OSPF, 488–494
route reflection, 707–711, 778
  client-to-client, 780–782
  cluster design, 757–758
  identical cluster IDs, 785, 788–790
  misconfigured IBGP neighbor, 779–780
  peer groups, 783–785
route summarization
  EIGRP, 276–280
  OSPF
    external routes, 497–499
    troubleshooting, 494–496
route tag field (RIP-2), 40–41
route-flapping, IS-IS, 612–616
Router Dead Interval field (OSPF Hello packets), 299
Router ID field (OSPF packets), 297
router LSAs (Type 1), 302–306
routers
  convergence, 129
  high-speed packet forwarding, 25
OSPF
  adjacencies, 334–338
  link types, 305
routes, poisoned, 22
routing domains, IS-IS, 536
routing loops, 30
DUAL, 207, 211
  active routes, 213
  FC, 211
  FD, 211
  feasible successors, 212
  passive routes, 213
  RD, 211
  successors, 211
IGRP
  split horizon, 130
  split horizon with poison reverse, 130
routing policies, BGP-4, 672–674
  AS_PATH attribute, 682–685
  LOCAL_PREF attribute, 675–676
  MED attribute, 677–682
  NEXT_HOP attribute, 685
  ORIGIN attribute, 685
  WEIGHT knob, 686–688
routing protocols
  administrative distance, 24
  classful, 29
  distance vector
    convergence, 19–20
    counting to infinity, 21
    holddown, 21
    loop avoidance, 20–21
    metrics, 19
    periodic updates, 22
    poison reverse, 22
    split horizon, 22
    triggered updates, 22
EIGRP, 207
  behavior, 218–219
  convergence, 207
  default routes, 221
  DUAL, 207, 211–213
  IP Internal Route TLV, 216–217
  metrics, 208–209
  neighbor relationships, 209–210
  packet fields, 216–217
  query process, 220
  RTP, 214
  summarization, 219–220
  unequal-cost load balancing, 221–223
IGRP
  behavior, 131
  default routes, 132–133
  metrics, 127–129
  packets, 131
  split horizon, 130
  split horizon with poison reverse, 130
  timers, 129–130
  unequal-cost load balancing, 133–134
  link-state, 23–24
OSPF, 295
  adjacencies, 334–338
  areas, 315–326
  demand circuits, 331–334
  external LSAs, 313–315
  LSAs, 302–304
  multaccess media, 327–328
  NBMA media, 329–331
  network LSAs, 307–308
  packets, 295–301
  router LSAs, 304–306
  summary LSAs, 309–312
  virtual links, 316–317
  unicast versus multicast, 13–14
routing tables
  BGP
    uninstalled EBGP-learned routes, 771–777
    uninstalled IBGP-learned routes, 763–771
  EIGRP, nonexistent summarized routes, 276–278
  IGRP
    flapping routes, 198–201
    uninstalled routes, troubleshooting, 142–166
OSPF
  uninstalled external routes, 479–487
  uninstalled routes, 463–478
routing updates, 606–607
  IS-IS
    redistribution into Level 1, 611
    route advertisements, 607–611
  RPF (reverse path forwarding), 628–630
  RPF check failure, 650
  RPs (rendezvous points), 632
  RTO (retransmission timeout), 232
  RTP (Reliable Transfer Protocol), 214
  Rtr Pri field (OSPF Hello packets), 299
scalability, IBGP
  AS confederations, 711–712
  route reflectors, 707–711
security
  authentication, null authentication, 364
  IS-IS, authentication, 548
selecting BGP best-path, 821–827
sender problems
  IGRP uninstalled routes, troubleshooting, 142, 168–188
  uninstalled IGRP routes, 163–166
sending RIP routes, 31–33, 86
  blocked routes, 91–93
  broken multicast capability, 96–98
  down network interface, 93–94
  down outgoing interface, 89–91
  misconfigured neighbor statement, 99–100
  missing network statement, 87–89
  passive outgoing interface, 95–96
  split horizon, 102, 105–106
  VLSM routes, 100–102
Sequence field (EIGRP packets), 216
Sequence Number field (LSPs), 554
serial links (point-to-point), IS-IS configuration, 559–565
servers, route reflectors, 757
set as-path prepend command, implementing in route maps, 693
set community command, implementing in route maps, 693
set local preference command, implementing in route maps, 694
set metric command, implementing in route maps, 694
show clns interface command, 564, 586
show clns neighbors command, 586
show clns neighbors detail command, 563
  field definitions, 588
show clns protocol command, 562–563
show ip bgp command, 726
show ip bgp neighbor command, 726
show ip bgp neighbors command, 726–727
show ip bgp summary command, 726
show ip eigrp neighbor command, 210
show ip eigrp topology active command, 242–245
show ip protocols command, 56
show ip route command, 12, 56, 134, 142
show isis database command, 586
show isis topology command, 565, 586
SIA (stuck in active) route errors, 220
EIGRP, 240–250
SNPs (sequence number packets), 556
source validity checks, failure on IGRP networks, 159–161
sparse mode (PIM), 625, 632
  discovery process, 632–633
  join mechanism, 633
  register process, 634
  RPs, 632
  troubleshooting, 651, 654–656
SPF algorithm, 17
  IS-IS decision process, 558
  OSPF, 518–528
  triggers, 614–615
split horizon, 130–131
distance vector protocols, 22
EIGRP, 253–256
RIP, 30
  poison reverse, 31
  route advertisement, 102, 105–106
unadvertised IGRP routes, troubleshooting, 184, 187–188
SRTT (smooth round-trip time), 232
standard access lists
  BGP filtering, unfiltered subnets, 828–830
  debug ip bgp update command output, 727
static routing, 4, 11
stub areas, 319–320
Stuck in 2-WAY state, 398–400
Stuck in ATTEMPT, 383–386
Stuck in EXSTART/EXCHANGE state, 401–417
Stuck in INIT state, 382, 387–398
Stuck in LOADING state, 417–422
subnet masks, RIP, 41
subnets, 12
  autosummarization, 106–109
  discontiguous, uninstalled IGRP routes, 155–158
  masks, 8
successors (EIGRP), 211
summarization
   EIGRP, 276–280
   OSPF
      external routes, 497–499
      troubleshooting, 494–496
   RIP, 109
      excessively large routing table, 110–113
summary ASBR LSAs (Type 4), 302
summary LSAs (Type 3), 309–312
summary network LSAs (Type 3), 302
summary routes, unadvertised, 432–440
supernets, 10
synchronization
   BGP routes, disabling, 766
   IS-IS, 555–557
synchronization rule (BGP-4), 671–672
synchronized BGP routes, propagating to neighbor, 761–762
System identifier field (LSPs), 554

TCP (Transmission Control Protocol), unintentional packet blockages, 741–742
TCP/IP (Transmission Control Protocol/Internet Protocol)
   IP addressing
      CIDR, 10
      classes, 5–7
      media independence, 5
      private address space, 7–8
      subnets, 12
      subnetting, 8
   IP protocol, 3
      versus OSI reference model, 3
   three-way reliable adjacencies, 540
   timers, RIP, 30
   timers basic command, 130
TLV (Type/Length/Value), 216–217
   IP Internal Route, 216–217
   IS-IS packets, 543–545
   LSP TLVs, 547
      metric information, 545–548
topologies, IS-IS, displaying, 565
topology table (EIGRP)
      convergence, 19–20
      diffused computation, 213
      local computation, 213
ToS field (router LSAs), 305
ToS field (summary LSAs), 310
ToS Metric field (router LSAs), 305
ToS metric field (summary LSAs), 310
totally NSSAs, 324–326
totally stubby areas, 321
traceroute command, 617–619
traffic
   EIGRP, unequal-cost load balancing, 221–223
   IGRP
      load balancing, 168
      unequal-cost load balancing, 133–134
   transit links, identifying attached routers, 308
   transit peering, 660
   triggered updates, distance vector protocols, 22
   Type 1 LSAs, 304–306
   Type 2 LSAs, 307–308
   Type 3 LSAs, 309–312
   Type 4 LSAs, 309–312
   Type 5 LSAs, 313–315
      comparing to Type 7, 322
   Type 7 LSAs
      filtering, 326
      NSSAs, 321–324
   Type field
      IGMP packets, 635
      OSPF packets, 296
      router LSAs, 305
unadvertised IGRP routes
  broadcast capability, 178–180
  default route candidates, 188–191
distribute lists, 173–174
misconfigured neighbor statement, 180–181
misconfigured network statement, 169–171
network interface, 175–176
outgoing interface, 171–172
passive outgoing interface, 176–178
split horizon, 184, 187–188
troubleshooting, 169
VLSM, 182–184
uncommon subnets, EIGRP, 233–235
unequal-cost load balancing, 133–134
  EIGRP, 221–223
  IGRP, 133–134
  variance, 201–204
unexpected metrics, EIGRP, 259–263
unexpected route advertisements, EIGRP, 257–259
unicast routing versus multicast, 12–14
unicasts, 625
unidirectional links, EIGRP, 230, 232
uninstalled routes
  EBGP
    dampening, 771–774
    infinite MED value, 777
    unreachable multihop EBGP next hop, 774–777
  EIGRP, 265–270
  external (OSPF), 479, 481–487
  IBGP
    synchronization, 763–766
    unreachable next-hops, 766–771
IGRP
  receiver problems, 142
  sender problems, 168–188
  troubleshooting, 142–166
OSPF, 463–478
RIP, 52
“unknown routing protocol” error messages, 528
unreachable IBGP next hops, 766–771
unreliable EIGRP packets, 214
update packets (EIGRP), 214
update process (IS-IS), 555

update timers
  IGRP, 129
  IS-IS, 556–557
  RIP, 30
updates
  IGRP, poison updates, 130
  RIP
    receiving, 33–35
    sending, 31–33
utilization (CPU), OSPF, 499–503

V
variable-length subnet mask. See VLSM
variance, 201–204
variance command, 133–134, 221–223
Version field
  EIGRP packets, 216
  RIP, 43
Version Number field (OSPF packets), 296
virtual links, 316–318
VLSM (variable-length subnet mask), 9
  RIP, 37–39
    route advertisement, 100–102
unadvertised IGRP routes, troubleshooting, 182–184

W
WEIGHT knob, policy control, 686–688