Implementing Cisco IP Routing (ROUTE)

Foundation Learning Guide

Foundation learning for the CCNP ROUTE 642-902 Exam

ciscopress.com

Diane Teare

Implementing Cisco IP Routing (ROUTE) Foundation Learning Guide

Foundation learning for the ROUTE 642-902 Exam

Diane Teare

Cisco Press

800 East 96th Street

Indianapolis, IN 46240

Implementing Cisco IP Routing (ROUTE) Foundation Learning Guide

Foundation learning for the ROUTE 642-902 Exam

Diane Teare

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

Printed in the United States of America

Fifth Printing: September 2012

Library of Congress Cataloging-in-Publication Data is on file.

ISBN-13: 978-1-58705-882-0

ISBN-10: 1-58705-882-0

Warning and Disclaimer

This book is designed to provide information about Cisco routing. 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 authors 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 or 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.

Corporate and Government Sales

The publisher offers excellent discounts on this book when ordered in quantity for bulk purchases or special sales, which may include electronic versions and/or custom covers and content particular to your business, training goals, marketing focus, and branding interests. For more information, please contact: U.S. Corporate and Government Sales 1-800-382-3419 corpsales@pearsontechgroup.com

For sales outside the United States please contact: International Sales international@pearsoned.com

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 make sure to include the book title and ISBN in your message.

We greatly appreciate your assistance.

Publisher: Paul Boger

Associate Publisher: Dave Dusthimer

Executive Editor: Mary Beth Ray

Managing Editor: Sandra Schroeder

Development Editor: Dayna Isley

Editorial Assistant: Vanessa Evans

Book Designer: Louisa Adair

Cover Designer: Sandra Schroeder

Composition: Mark Shirar

Business Operation Manager, Cisco Press: Anand Sundaram

Manager Global Certification: Erik Ullanderson

Copy Editor: Keith Cline

Proofreader: Leslie Joseph

Project Editor: Mandie Frank

Indexer: Tim Wright

Technical Editors: Sonya Coker, Jeremy Creech, Rick Graziani, Scott Hogg, David Kotfila, Wayne Lewis, Jim Lorenz, Snezhy Neshkova, Allan Reid, Jerold Swan, Bob Vachon

Americas Headquarters
Cisco Systems, Inc.
San Jose, CA

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

Europe Headquarters
Cisco Systems International BV
Amsterdam, The Netherlands

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

About the Author

Diane Teare is a professional in the networking, training, project management, and e-learning fields. She has more than 25 years of experience in designing, implementing, and troubleshooting network hardware and software, and has been involved in teaching, course design, and project management. She has extensive knowledge of network design and routing technologies, and is an instructor with one of the largest authorized Cisco Learning Partners. She was the director of e-learning for the same company, where she was responsible for planning and supporting all the company's e-learning offerings in Canada, including Cisco courses. Diane has a bachelor's degree in applied science in electrical engineering and a master's degree in applied science in management science. She currently holds her Cisco Certified Network Professional (CCNP), Cisco Certified Design Professional (CCDP), and Project Management Professional (PMP) certifications. She co-authored the Cisco Press titles *Designing Cisco Network Service Architectures (ARCH)*, Second Edition; *Campus Network Design Fundamentals*; the three editions of *Authorized Self-Study Guide Building Scalable Cisco Internetworks (BSCI)*; and *Building Scalable Cisco Networks.* Diane edited the two editions of the *Authorized Self-Study Guide Designing for Cisco Internetwork Solutions (DESGN)* and *Designing Cisco Networks*.

About the Contributor

Catherine Paquet is a practitioner in the field of Internetworking, Network Security, and Security Financials. So far, she has published eight books with Cisco Press. Catherine has in-depth knowledge of security systems, remote access, and routing technology. She is a Cisco Certified Security Professional (CCSP), a Cisco Certified Network Professional (CCNP), and a Certified Cisco Systems Instructor (CCSI) with the largest Cisco training partner, Global Knowledge. Catherine teaches many Cisco security classes such as Securing Networks with ASA (SNAF,SNAA). She also lectures directly with Cisco Systems in emerging countries on the business case for network Security. Her most recent consulting projects include conducting security assessments, performing network designs, configuring and implementing security solutions such as firewalls, virtual private networks, web filters, and intrusion prevention solutions.

About the Technical Reviewers

Sonya Coker has worked in the Cisco Networking Academy program since 1999 when she started a local academy. She has taught student and instructor classes locally and internationally in topics ranging from IT Essentials to CCNP. As a member of the Cisco Networking Academy development team, she has provided subject matter expertise on new courses and course revisions.

Jeremy Creech is a Learning and Development Manager for Cisco Systems with more than 13 years of experience in researching, implementing, and managing data and voice networks. Currently, he is a curriculum development manager for the Cisco Networking Academy Program, leveraging his experience as the Content Development Manager for CCNP Certification exams. He has recently completed curriculum development initiatives for ROUTE, SWITCH, TSHOOT and CCNA Security.

Scott Hogg has been a network computing consultant for more than 18 years. He has a B.S. in computer science, a M.S. in telecommunications, along with his CCIE (No. 5133) and CISSP (No. 4610). For the past ten years, Scott has been researching IPv6 technologies and recently has helped many organizations with their IPv6 planning. Scott has given numerous presentations and demonstrations of IPv6 technologies and authored the book titled *IPv6 Security*. He is also currently the Chair of the Rocky Mountain IPv6 Task Force.

Rick Graziani teaches computer science and computer networking courses at Cabrillo College in Aptos, California. Rick has worked and taught in the computer networking and information technology field for almost 30 years. Before teaching, Rick worked in IT for various companies, including Santa Cruz Operation, Tandem Computers, and Lockheed Missiles and Space Corporation. He holds a Master of Arts degree in computer science and systems theory from California State University Monterey Bay. Rick also does consulting work for Cisco Systems and other companies. When Rick is not working, he is most likely surfing. Rick is an avid surfer who enjoys surfing at his favorite Santa Cruz breaks.

David Kotfila, CCNA, CCDA, CCNP, CCDP, CCSP, CCVP, CCAI, teaches in the Computer Science department at Rensselaer Polytechnic Institute, Troy, New York. More than 550 of his students have received their CCNA, 200 have received their CCNP, and 14 have received their CCIE. David likes to spend time with his wife, Kate, his daughter, Charis, and his son, Chris. David enjoys hiking, kayaking, and reading.

Wayne Lewis has been a faculty member at Honolulu Community College since receiving a Ph.D. in math from the University of Hawaii at Manoa in 1992, specializing in finite rank torsion-free modules over a Dedekind domain. Since 1992, he has served as a math instructor, as the state school-to-work coordinator, and as the legal main contact for the Cisco Academy Training Center (CATC).

Dr. Lewis manages the CATC for CCNA, CCNP, and Security, based at Honolulu Community College, which serves Cisco Academies at universities, colleges, and high schools in Hawaii, Guam, and American Samoa. Since 1998, he has taught routing, multilayer switching, remote access, troubleshooting, network security, and wireless networking to instructors from universities, colleges, and high schools in Australia, Britain,

Canada, Central America, China, Germany, Hong Kong, Hungary, Indonesia, Italy, Japan, Korea, Mexico, Poland, Singapore, Sweden, Taiwan, and South America, both onsite and at Honolulu Community College.

Jim Lorenz is an instructor and curriculum developer for the Cisco Networking Academy Program. Jim has co-authored Lab Companions for the CCNA courses and the textbooks for the Fundamentals of UNIX course.

He has more than 25 years of experience in information systems, ranging from programming and database administration to network design and project management. Jim has developed and taught computer and networking courses for both public and private institutions. As the Cisco Academy Manager at Chandler-Gilbert College in Arizona, he was instrumental in starting the Information Technology Institute (ITI) and developed several certificates and degree programs.

Jim co-authored the CCNA Discovery online academy courses, Networking for Home and Small Businesses, and Introducing Routing and Switching in the Enterprise, with Allan Reid. Most recently, he developed the hands-on labs for the CCNA Security course and the CCNPv6 Troubleshooting course.

Snezhy Neshkova is a Cisco Certified Internetwork Expert (CCIE No. 11931) since 2003. She has more than 20 years of networking experience, including IT field services and support, management of information systems, and all aspects of networking education. Snezhy has developed and taught CCNA and CCNP networking courses to instructors from universities, colleges, and high schools in Canada, the United States, and Europe. Snezhy's passion is to empower students to become successful and compassionate lifelong learners. Snezhy holds a Master of Science degree in computer science from Technical University, Sofia, Bulgaria.

Allan Reid (CCNA, CCNA-W, CCDA, CCNP, CCDP, CCAI, MLS) is a professor in information and communications engineering technology and the lead instructor at the Centennial College CATC in Toronto, Canada. He has developed and taught networking courses for both private and public organizations and has been instrumental in the development and implementation of numerous certificate, diploma, and degree programs in networking. Outside of his academic responsibilities, Allan has been active in the computer and networking fields for more than 25 years and is currently a principal in a company specializing in the design, management, and security of network solutions for small and medium-sized companies. Allan is a curriculum and assessment developer for the Cisco Networking Academy program and has authored several Cisco Press titles.

Jerold Swan, CCIE No. 17783, CCSP, works as a senior network engineer for the Southern Ute Indian Tribe Growth Fund in southwest Colorado. Before that, he was a Cisco instructor for Global Knowledge. He has also worked in IT in the service provider and higher-education sectors. His areas of interest include routing protocols, security, and network monitoring. He is a graduate of Stanford University. His other interests include trail running, mountain biking, and volunteer search and rescue.

Bob Vachon, CCNP, CCNA-S, CCAI, is a professor in the Computer Systems Technology program at Cambrian College and has more than 20 years of experience in the networking field. In 2001, he began collaborating with the Cisco Networking Academy on various curriculum development projects, including CCNA, CCNA Security, and CCNP courses. For 3 years, Bob was also part of an elite team authoring CCNP certification exam questions. In 2007, Bob co-authored the *CCNA Exploration: Accessing the WAN* Cisco Press book.

Dedications

This book is dedicated to my inspirational husband, Allan Mertin, whose love and encouragement is so welcome; to our delightful and loving son, Nicholas, and his unending curiosity to discover everything about everything; and to my parents, Syd and Beryl, for their constant support and encouragement.

Acknowledgments

I want to thank many people for helping to put this book together:

The Cisco Press team: Mary Beth Ray, the executive editor, coordinated the whole project, steered the book through the necessary processes, and understood when the inevitable snags appeared. Patrick Kanouse, the managing editor, brought the book to production. Vanessa Evans was once again instrumental in organizing the logistics and administration. Dayna Isley, the development editor, has been invaluable in coordinating and producing a high-quality manuscript.

I also want to thank Mandie Frank, the project editor, and Keith Cline, the copy editor, for their excellent work in steering this book through the editorial process.

The Cisco ROUTE course development team: Many thanks to the members of the team who developed the ROUTE course.

The contributing author: I want to thank my friend and colleague, Catherine Paquet, for agreeing to contribute a chapter to this book, enriching it with her expertise and ensuring that the schedule did not suffer. I owe you, Catherine!

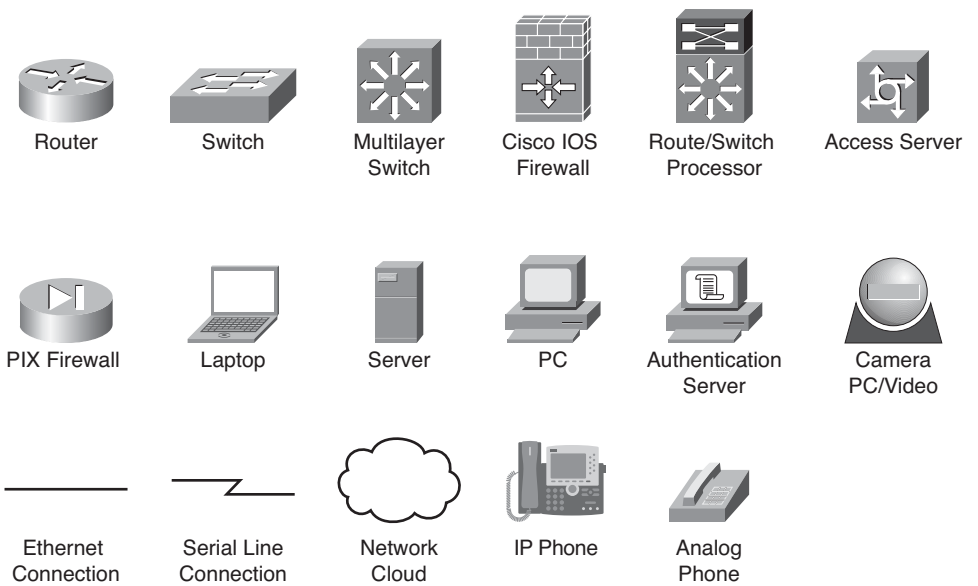
The technical reviewers: I want to thank the technical reviewers of this book for their thorough, detailed review and valuable input. Special thanks to Bob Vachon for his invaluable (and tedious, I'm sure) screen-capture work.

My family: Of course, this book would not have been possible without the endless understanding and patience of my family. They have always been there to motivate and inspire me. I am forever grateful.

Contents at a Glance

Contents

Icons Used in This Book



Command Syntax Conventions

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

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

Introduction

Networks continue to grow, becoming more complex as they support more protocols and more users. This book teaches you how to plan, configure, maintain, and scale a routed network. It focuses on using Cisco routers connected in LANs and WANs typically found at medium-to-large network sites. After completing this book, you will be able to select and implement the appropriate Cisco IOS services required to build a scalable, routed network.

In this book, you study a broad range of technical details on topics related to routing. First, complex enterprise network frameworks, architectures, and models are explored, and the process of creating, documenting, and executing an implementation plan is detailed. Internet Protocol (IP) routing protocol principles are examined in detail before the following IP Version 4 (IPv4) routing protocols are explored: Enhanced Interior Gateway Routing Protocol (EIGRP), Open Shortest Path First (OSPF), and Border Gateway Protocol (BGP). Manipulating routing updates and controlling the information passed between them are examined. Routing facilities for branch offices and mobile workers are explored. Finally, IP Version 6 (IPv6) is investigated in detail.

Configuration examples and sample verification outputs demonstrate troubleshooting techniques and illustrate critical issues surrounding network operation. Chapter-ending review questions illustrate and help solidify the concepts presented in this book.

This book starts you down the path toward attaining your CCNP, CCIP, or CCDP certification, providing in-depth information to help you prepare for the ROUTE exam (642-902).

The commands and configuration examples presented in this book are based on Cisco IOS Release 12.4.

Who Should Read This Book?

This book is intended for network architects, network designers, systems engineers, network managers, and network administrators who are responsible for implementing and troubleshooting growing routed networks. This book is also the official textbook for the Cisco Networking Academy program CCNP v6 ROUTE course.

If you are planning to take the ROUTE exam toward your CCNP, CCIP, or CCDP certification, this book provides you with in-depth study material. To fully benefit from this book, you should be CCNA certified or possess CCNA-level knowledge, including an understanding of the following topics:

- A working knowledge of the OSI reference model
- An understanding of internetworking fundamentals, including commonly used networking terms, numbering schemes, topologies, distance vector routing protocol operation, and when to use static and default routes
- The ability to operate and configure a Cisco router, including displaying and interpreting a router's routing table, configuring static and default routes, enabling a WAN serial connection using High-Level Data Link Control (HDLC) or Point-to-

Point protocol (PPP), configuring Frame Relay permanent virtual circuits (PVCs) on interfaces and subinterfaces, configuring IP standard and extended access lists, and verifying router configurations with available tools, such as **show** and **debug** commands

- Working knowledge of the TCP/IP stack, and configuring IP addresses and the Routing Information Protocol (RIP)

If you lack this knowledge and these skills, you can gain them by completing the Interconnecting Cisco Network Devices Part 1 (ICND1) and Interconnecting Cisco Network Devices Part 2 (ICND2) courses, or by reading the related Cisco Press books.

How This Book Is Organized

The chapters and appendixes in this book are as follows:

- Chapter 1, "Routing Services," describes the frameworks, architectures, and models used in complex enterprise network designs. The chapter also explores the process of creating, documenting, and executing a network implementation plan. The chapter concludes with a review of IP routing principles, including static and dynamic routing characteristics; classful and classless routing; the differences between distance vector, link-state, and advanced distance vector routing protocol behavior; the characteristics and configuration of RIP; and how Cisco routers populate their routing tables.
- Chapter 2, "Configuring the Enhanced Interior Gateway Routing Protocol," introduces EIGRP terminology and operations, and explains how to plan for, configure and verify EIGRP. Considerations for deploying EIGRP in enterprise WANs, and how to configure and verify EIGRP authentication are explored. The chapter concludes with a discussion of optimizing EIGRP implementations.
- Chapter 3, "Configuring the Open Shortest Path First Protocol," introduces the OSPF routing protocol. Basic configuration of OSPF, in both single and multiple areas is described. OSPF configuration over specific network types is also explored. The configuration and verification of advanced OSPF features are covered, including passive interface, default routes, summarization, virtual links, changing the cost metric, and special area types. The chapter concludes with a discussion of OSPF authentication configuration and verification.
- Chapter 4, "Manipulating Routing Updates," discusses network performance issues related to routing and using multiple IP routing protocols on a network. Implementing route redistribution between different routing protocols is described, and methods of controlling the routing information sent between these routing protocols are explored, including using route maps, distribute lists, and prefix lists. The chapter concludes with a comprehensive example of controlling routing updates.

- Chapter 5, "Implementing Path Control," describes path control, and details three tools for path control: offset lists, Cisco IOS IP service level agreements (SLAs), and policy-based routing (PBR). The chapter concludes with a discussion of advanced path control tools.
- Chapter 6, "Implementing a Border Gateway Protocol Solution for ISP Connectivity," introduces BGP terminology and concepts, and provides BGP configuration, verification, and troubleshooting techniques. The chapter also introduces route maps for manipulating BGP path attributes and filters for BGP routing updates.
- Chapter 7, "Implementing Routing Facilities for Branch Offices and Mobile Workers," discusses branch office implementation planning and the various services that can be implemented for branch office connectivity. The chapter explores how to route traffic to branch offices and discusses how mobile workers can connect to the corporate network.
- Chapter 8, "Implementing IPv6 in an Enterprise Network," introduces IPv6 and the IPv6 addressing scheme and explores how IPv6 addresses are configured. Routing protocols that support IPv6 are explored in detail, as is IPv6 policy routing and route redistribution. The chapter concludes with a discussion of how IPv4 networks can be transitioned to IPv6, including dual stack, tunneling, and translation techniques.
- Appendix A, "Answers to Review Questions," contains the answers to the review questions that appear at the end of each chapter.

You can find the following supplemental material at this book's companion website (http://www.ciscopress.com/title/9781587058820):

- Appendix B, "IPv4 Supplement," provides job aids and supplementary information that are intended for your use when working with IPv4 addresses. Topics include a subnetting job aid, a decimal-to-binary conversion chart, an IPv4 addressing review, an IPv4 access lists review, IP address planning, hierarchical addressing using variable-length subnet masks (VLSMs), route summarization, and classless interdomain routing (CIDR).
- Appendix C, "BGP Supplement," provides supplementary information on BGP covering the following topics: BGP route summarization, redistribution with interior gateway protocols (IGPs), communities, and route reflectors.
- "Acronyms and Abbreviations" identifies abbreviations, acronyms, and initialisms used in this book and in the internetworking industry.

This page intentionally left blank

Chapter 5

Implementing Path Control

This chapter discusses. It covers the following topics:

- Understanding Path Control
- Implementing Path Control Using Offset Lists
- Implementing Path Control Using Cisco IOS IP SLAs
- Implementing Path Control Using Policy-Based Routing
- Advanced Path Control Tools

This chapter starts by discussing path control fundamentals. Three tools for path control are detailed: offset lists, Cisco IOS IP service level agreements (SLAs), and policy-based routing (PBR). The chapter concludes with a discussion of advanced path control tools.

Understanding Path Control

This section introduces path control performance issues and introduces the tools available to control path selection.

Assessing Path Control Network Performance

This chapter is concerned with controlling the path that traffic takes through a network. In some cases, there might be only one way for traffic to go. However, many networks include redundant paths, by having redundant devices or redundant links. In these cases, the network administrator may want to control which way certain traffic flows.

The choice of routing protocol or routing protocols used in a network is one factor in defining how paths are selected; for example, different administrative distances, metrics, and convergence times may result in different paths being selected. As described in Chapter 4, “Manipulating Routing Updates,” when multiple routing protocols are implemented, inefficient routing may result. For example, two-way multipoint redistribution

requires careful planning and implementation to ensure that traffic travels the optimal way, and that there are no routing loops.

When a network includes redundancy, other considerations include the following:

- **Resiliency—**Having redundancy does not guarantee resiliency, the ability to maintain an acceptable level of service when faults occur. For example, having redundant links between two sites does not automatically result in the backup link being used if the primary link fails. Configuration is necessary to implement failover, and to use the backup link for load sharing if that is desired. (Even if failover is configured correctly, the redundant link may not operate when needed; for example, if it uses the same physical infrastructure as the primary link.)
- **Availability—**The time required for a routing protocol to learn about a backup path when a primary link fails is the *convergence time*. If the convergence time is relatively long, some applications may time out. Thus, using a fast-converging routing protocol, and tuning parameters to ensure that it does converge fast, is crucial for high-availability networks.
- **Adaptability—**The network can also be configured to adapt to changing conditions. For example, a redundant path could be brought up and used when the primary path becomes congested, not just when it fails.
- **Performance—**Network performance can be improved by tuning routers to load share across multiple links, making more efficient use of the bandwidth. For example, route advertisements for specific prefixes can be advertised on one link to change the balance of bandwidth use relative to other links.
- **Support for network and application services—**More advanced path control solutions involve adjusting routing for specific services, such as security, optimization, and quality of service (QoS). For example, to optimize traffic via a Cisco Wide Area Application Services (WAAS) Central Manager, traffic must be directed to flow through the Cisco WAAS device.

Note Cisco WAAS is a WAN optimization and application acceleration solution that optimizes application and video delivery over a WAN, and is illustrated briefly in the "Cisco Wide Area Application Services" section, later in this chapter.

- **Predictability—**The path control solution implemented should derive from an overall strategy, so that the results are deterministic and predictable. For example, traffic is bidirectional by nature; for every packet that goes out, a reply typically must come back. When configuring a routing protocol to deploy a path control strategy, consider both upstream and downstream traffic. For example, changing or tuning downstream advertisements toward a server farm could adversely affect upstream traffic flows from the server farm.

- **Asymmetric traffic—**Asymmetric traffic, traffic that flows one on path in one direction and on a different path in the opposite direction, occurs in many networks that have redundant paths. Asymmetry, far from being a negative trait, is often *desirable* network trait, because it uses available bandwidth effectively, such as on an Internet connection on which downstream traffic may require higher bandwidth than upstream traffic. Border Gateway Protocol (BGP) includes a good set of tools to control traffic in both directions on an Internet connection. However, in most routing protocols, there are no specific tools to control traffic direction.

  In a part of a network that includes devices or services such as stateful firewalls, Network Address Translation (NAT) devices, and voice traffic, which require symmetrical routing, traffic symmetry must be enforced or the services must be tuned to accommodate asymmetry. For example, asymmetry in voice networks may introduce jitter and QoS issues. In other areas of the network, though, it might be inefficient and undesirable to try to engineer artificial symmetry.

Optimal routing in terms of network utilization within specific requirements is typically a design goal. Those requirements should be considered within the context of the applications in use, the user experience, and a comprehensive set of performance parameters. These parameters include delay, bandwidth utilization, jitter, availability, and overall application performance. Even if the routing table on the routers includes the necessary prefixes, applications might still fail if the performance requirements are not met.

Path Control Tools

Unfortunately there is not a "one-command" solution to implement path control. Instead, many tools are available.

Path control tools include the following:

- **A good addressing design:** A good design should include summarizable address blocks and classless interdomain routing (CIDR) that align with the physical topology. These aspects are key to a stable network. As discussed in Chapter 1, "Routing Services," summarization hides addressing details, isolates routing issues, and defines failure domains. Controlling summarization in strategic areas of the network affects path control. For example, in the network in Figure 5-1, the 10.0.0.0/8 summary is advertised from both routers, and the more specific route for 10.1.80.0/24 is advertised from the router on the right, providing direct access to that subnet. The resulting traffic flows are deterministic and more resilient.
- **Redistribution and other routing protocol characteristics—**The capabilities of the routing protocol used can help implement a path control strategy more effectively, as summarized in Table 5-1. For example, Enhanced Interior Gateway Routing Protocol (EIGRP) automatically summarizes on network boundaries, and Open Shortest Path First (OSPF) can summarize only on Area Border Routers (ABRs) and Autonomous System Boundary Routers (ASBRs). Metrics can be changed and external routes can be tagged during redistribution between protocols. When multiple

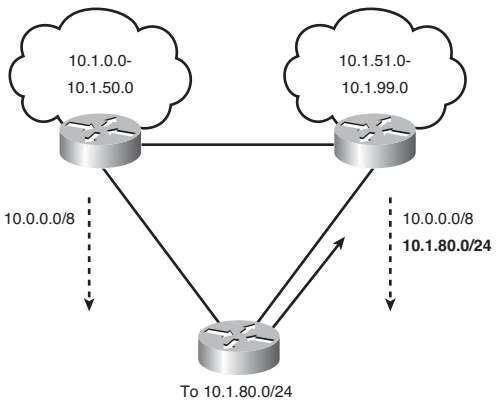


Figure 5-1 *Advertising Summaries and More-Specific Routes Affects Traffic Flow.*

routing protocols are used, routes must be redistributed between them carefully, as detailed in Chapter 4.

- **Passive interfaces—**As also described in Chapter 4, passive interfaces prevent a routing protocol's routing updates from being sent through the specified router interface.

Table 5-1 *Routing Protocol Characteristics*

Characteristic	OSPF	EIGRP
Route marking	Tags for external routes can be added at distribution points.	Tags for all routes can be configured.
Metric	Can be changed for external routes at redistribution points.	Can be set using route maps.
Next hop	Can be changed for external routes at redistribution points.	Can be set for all routes under various conditions.
Filtering	Summary information can be filtered at ABRs and ASBRs.	Can be configured anywhere for any routes.
Route summarization	Can be configured only on ABRs and ASBRs.	Can be configured anywhere for any routes. Autosummarization is on by default.[1]
Unequal-cost load balancing	Not available.	Available, with **variance** command.

[1]As mentioned in Chapter 1, the Cisco IOS documentation for EIGRP says that automatic summarization is now disabled by default. However, testing has confirmed it is still on, at least in some versions of the Cisco IOS. Thus, it would be prudent to confirm the autosummary configuration or to configure it explicitly.

Other tools include the following:

- Distribute lists
- Prefix lists
- Administrative distance
- Route maps
- Route tagging
- Offset lists
- Cisco IOS IP SLAs
- PBR

The first five of these tools were covered in Chapter 4; the others are the focus of the rest of this chapter.

Note Three other tools are covered in the "Advanced Path Control Tools" section, at the end of the chapter.

You can use all of these tools as part of an integrated strategy to implement path control, as illustrated in Figure 5-2. It is important to have a strategy before implementing specific path control tools and technologies.

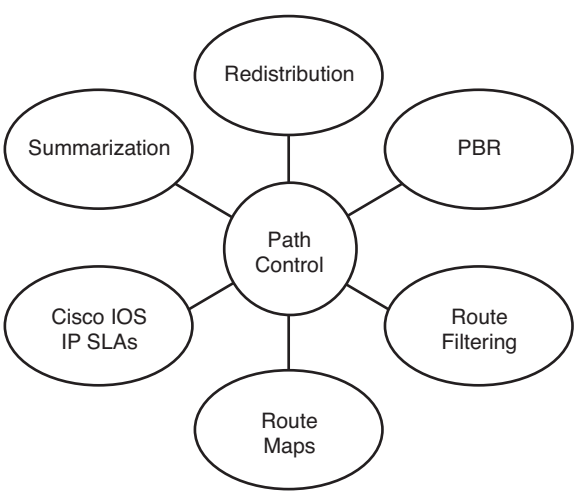


Figure 5-2 *Path Control Requires an Integrated Strategy.*

For example, filters allow specific control of routing updates and provide security mechanisms to hide specific destinations. In contrast, PBR can bypass the routing table and define a path based on static or dynamic information, forcing traffic to specific destinations such as security appliances, NAT devices, and WAN optimization elements.

As another example, by controlling and filtering routing updates in one direction, you can affect traffic flowing in the opposite direction and prevent that traffic from reaching those destinations

By tagging routes by using route maps, you can define priorities for specific destinations along multiple paths, allowing those paths to be used in a deterministic order. For example, on an Internet connection when multiple exit points exist out of a network, route maps can be used to tag and define priorities for specific destinations.

Implementing Path Control Using Offset Lists

This section introduces offset lists and how to configure and verify path control using offset lists.

Using Offset Lists to Control Path Selection

An offset list is the mechanism for increasing incoming and outgoing metrics to routes learned via EIGRP or Routing Information Protocol (RIP). (Offset lists are only used for distance vector routing protocols.) Optionally, an offset list can be limited by specifying either an access list or an interface.

Configuring Path Control Using Offset Lists

To add an offset to incoming and outgoing metrics to routes learned via EIGRP or RIP, use the **offset-list** {*access-list-number* | *access-list-name*} {**in** | **out**} *offset* [*interface-type interface-number*] router configuration command, as explained in Table 5-2.

Table 5-2 offset-list *Command*

Parameter	Description
access-list-number | *access-list-name*	Standard access list number or name to be applied. Access list number 0 indicates all access lists. If the *offset* value is 0, no action is taken.
in	Applies the access list to incoming metrics.
out	Applies the access list to outgoing metrics.
offset	Positive offset to be applied to metrics for networks matching the access list. If the offset is 0, no action is taken.
interface-type interface-number	(Optional) Interface type and number to which the offset list is applied.

The offset value is added to the routing metric. An offset list that specifies an interface type and interface number is considered to be an extended list and takes precedence over an offset list that is not extended. Therefore, if an entry passes the extended offset list and a normal offset list, the offset of the extended offset list is added to the metric.

Figure 5-3 illustrates an example network in which an organization is using RIP and is connected to the Internet service provider (ISP) via edge Routers R4 and R5. A subset of routes is received from each of the edge routers. The metric between Routers R2 and R5 is smaller than the metric between Routers R2 and R4, because it is only one hop. However, this is very slow link. An offset list can be used on Router R2 so that it prefers the path toward the edge Router R4 for a specific set of destinations.

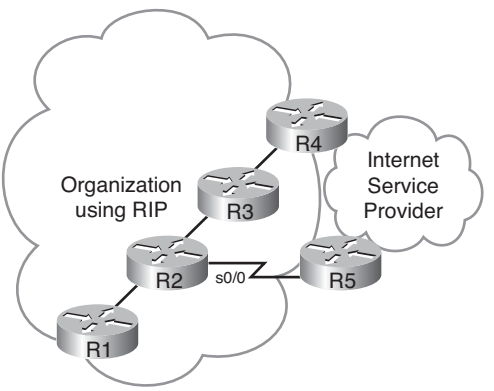


Figure 5-3 *An Offset List Can Be Used to Prefer a Faster Path.*

A partial configuration of Router R2 is shown in Example 5-1. In this example, the **offset-list 21 in 2 serial 0/0** command adds an offset of 2 to the metric of routes learned from interface serial 0/0 (connected to Router R5) that are permitted by access list 21. Access list 21 permits a specific set of routes (any in the 172.16.0.0/16 network) being learned from Router R5. This command is entered in RIP configuration mode on Router R2. This configuration results in the path toward Router R4 being considered better for the set of selected routes; R4 becomes the preferred way out toward the ISP for these routes.

Example 5-1 *Offset List Configuration for Router R2 in Figure 5-3*

```
router rip
 offset-list 21 in 2 serial 0/0
!
access-list 21 permit 172.16.0.0 0.0.255.255
```

Verifying Path Control Using Offset Lists

You can use the **traceroute** EXEC to verify that an offset list is affecting the path that traffic takes.

The routing table, viewed with the **show ip route** command, identifies the metrics for learned routes. You should compare these metrics to what was expected by the offset list configuration. For EIGRP, the EIGRP topology table can be examined using the **show ip eigrp topology** command. The topology table contains all routes learned from the router's EIGRP neighbors, and includes the metric information for those routes, including the best route and any other feasible routes that the router has learned about.

Note Recall that only successor and feasible successor routes are displayed with the **show ip eigrp topology** command. Add the **all-links** keyword to display all routes, including those not eligible to be successor or feasible successor routes.

You can use **debug** commands, such as **debug ip rip** and **debug ip eigrp**, to view the real-time processing of incoming and outgoing RIP routing updates, to ensure that the metric is being processed appropriately.

Caution Use caution when executing **debug** commands because they may consume a lot of router resources and could cause problems in a busy production network. Debugging output takes priority over other network traffic; too much debug output might severely reduce the performance of the router or even render it unusable in the worst case.

Implementing Path Control Using Cisco IOS IP SLAs

This section examines path control using Cisco IOS IP SLAs. A typical scenario for this solution is Internet branch office connectivity, with connections to two different ISPs, such as the network illustrated in Figure 5-4. In this case, the organization's edge router is configured to perform NAT, and has default routes for outbound traffic to the ISPs; branch offices, especially smaller ones, are not likely to run BGP or other routing protocols toward the ISP. The static default routes are likely to be equal cost, and the Cisco IOS will by default load balance over the links on a per-destination basis. NAT will be applied to the outbound traffic resulting from the load- balancing algorithm.

In this scenario, the edge router can detect if there is a direct failure on the link to one ISP, and in that case use the other ISP for all traffic. However, if the infrastructure within of one of the ISPs fails and the link to that ISP remains up, the edge router would continue to use that link because the static default route would still be valid.

There are multiple solutions to this issue. One approach is for the branch office router to run a dynamic routing protocol with the ISPs, so that the branch router learns the ISPs' networks in its routing table. The branch router will then be aware of any link failures within the ISPs' network. This solution is impractical for smaller branch offices, and in

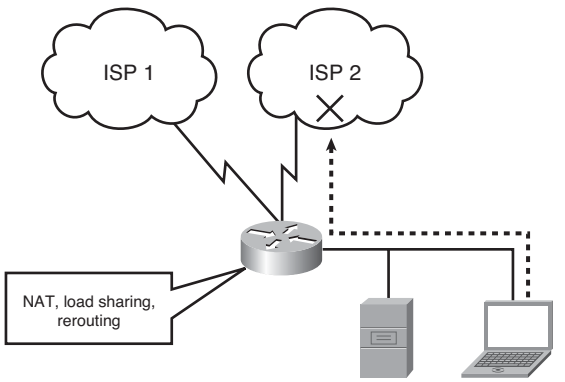


Figure 5-4 *A Branch Office Scenario.*

any case requires interaction and integration with the ISPs. It may, however, be the best solution for critical branch offices or those with large traffic volumes.

Another solution is to use either static routes or PBR, but make them subject to reachability tests toward critical destinations, such as the Domain Name System (DNS) servers within the ISP. If the DNS servers in one of the ISPs go down or are unreachable, the static route toward that ISP would be removed. These reachability tests can be performed with Cisco IOS IP SLAs that probe the DNS servers frequently and that are attached to the static routes.

The tools used for this solution include the following:

- **Object tracking—**The Cisco IOS object tracking tracks the reachability of specified objects (in this example, of DNS servers).
- **Cisco IOS IP SLAs probes—**The object tracking features can use Cisco IOS IP SLAs to send different types of probes toward the desired objects.
- **Route maps with PBR—**To associate the results of the tracking to the routing process, PBR with route maps can be used, allowing options to define specific traffic classes, such as voice, or specific applications.
- **Static routes with tracking options—**As an alternative to PBR, you can use static routes with tracking options. This solution is simpler and accommodates scenarios in which you want all outbound traffic to choose outbound exit points similarly.

Using Cisco IOS IP SLAs to Control Path Selection

This section introduces Cisco IOS IP SLAs and describes how this feature is used to control path selection.

Cisco IOS IP SLAs use active traffic monitoring, generating traffic in a continuous, reliable, and predictable manner, to measure network performance.

Cisco IOS IP SLAs, illustrated in Figure 5-5, send simulated data across the network and measure performance between multiple network locations or across multiple network paths. The information collected includes data about response time, one-way latency, jitter (interpacket delay variance), packet loss, voice-quality scoring, network resource availability, application performance, and server response time. In its simplest form, Cisco IOS IP SLAs verify whether a network element, such as an IP address on a router interface or an open TCP port on an IP host, is active and responsive.

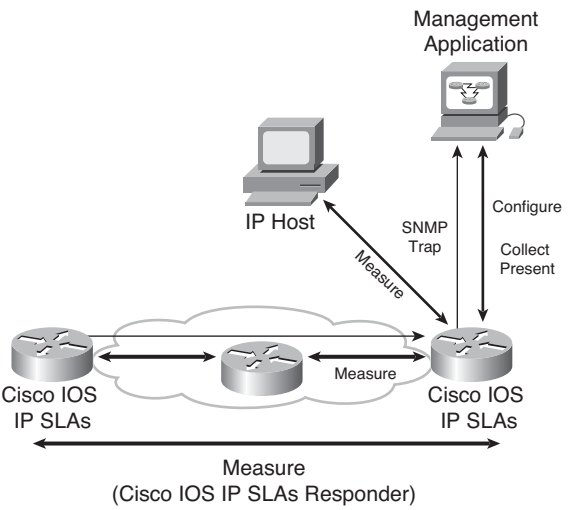


Figure 5-5 *Cisco IOS IP SLAs Measure Network Performance.*

Because Cisco IOS IP SLAs are accessible using Simple Network Management Protocol (SNMP), performance-monitoring applications, such as CiscoWorks Internetwork Performance Monitor (IPM) and other third-party Cisco partner performance-management products, can also use them.

Note For information about SNMP operation, see the SNMP chapter of Cisco's *Internetworking Technology Handbook*, available at http://www.cisco.com/en/US/docs/internetworking/technology/handbook/SNMP.html.

Cisco IOS IP SLAs use the Cisco Round-Trip Time Monitor (RTTMON) Management Information Base (MIB) for communication between the external Network Management System (NMS) applications and the Cisco IOS IP SLAs operations running on the Cisco devices.

As an additional feature, SNMP notifications based on the data gathered by a Cisco IOS IP SLAs operation allow the router to receive alerts when performance drops below a

specified level and when problems are corrected. These thresholds can trigger additional events and actions.

The following sections detail IP SLAs terminology and operation, before configuration, verification, and examples are provided in later sections.

Cisco IOS IP SLAs Operation

The embedded Cisco IOS IP SLAs measurement capability allows network managers to validate network performance, proactively identify network issues, and verify service guarantees by using active monitoring to generate probe traffic in a continuous, reliable, and predictable manner. This measurement capability also helps create a network that is "performance aware." Using IOS IP SLAs measurements, Cisco network equipment can verify service guarantees, validate network performance, improve network reliability, proactively identify network issues, and react to performance metrics with changes to the configuration and network.

The Cisco IOS IP SLAs feature allows performance measurements to be taken within and between Cisco devices, or between a Cisco device and a host, providing data about service levels for IP applications and services.

Cisco IOS IP SLAs measurements perform active monitoring by generating and analyzing traffic to measure performance between Cisco IOS Software devices or between a Cisco IOS device and a host, such as a network application server. With the IOS IP SLAs feature enabled, a router sends synthetic traffic to the other device, as illustrated in Figure 5-6.

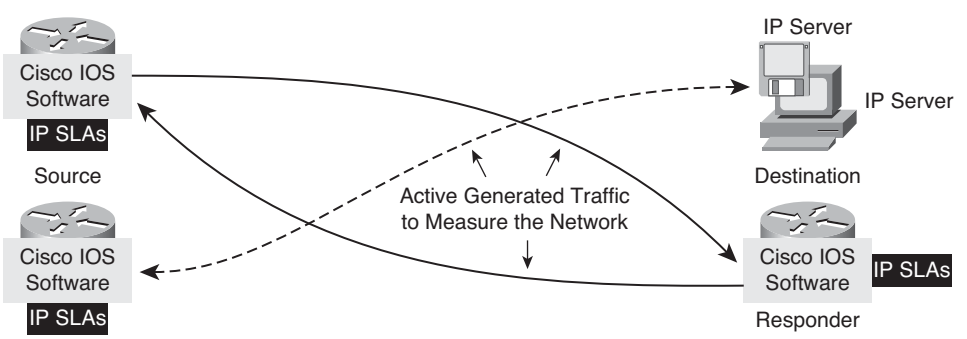


Figure 5-6 *IP SLAs Take Measurements Between a Cisco Device and Another Cisco Device or a Host.*

Cisco IOS IP SLAs Sources and Responders

All the IP SLAs measurement probe operations are configured on the IP SLAs *source*, either via the command-line interface (CLI) or through an SNMP tool that supports the operation of IP SLAs. The source sends probe packets to the *target*.

There are two types of IP SLAs operations: those in which the target device is running the IP SLAs *responder* component (such as a Cisco router), and those in which the target

device is not running the IP SLAs responder component (such as a web server or IP host). An IP SLAs responder is a component embedded in a Cisco IOS device that allows that device to anticipate and respond to IP SLAs request packets. A Cisco IOS device can be configured as an IP SLAs responder and will provide accurate measurements without the need for dedicated probes or any complex or per-operation configuration.

The IP SLAs measurement accuracy is improved when the target is an IP SLAs responder, as described in the upcoming "Cisco IOS IP SLAs Operation with Responders" section.

Cisco IOS IP SLAs Operations

An *IP SLAs operation* is a measurement that includes protocol, frequency, traps, and thresholds.

The network manager configures the IP SLAs source with the target device address, protocol, and User Datagram Protocol (UDP) or Transfer Control Protocol (TCP) port number, for each operation. When the operation is finished and the response has been received, the results are stored in the IP SLAs MIB on the source, and are retrieved using SNMP.

IP SLAs operations are specific to target devices. Operations such as DNS or HTTP can be sent to any suitable computer. For operations such as testing the port used by a database, there might be risks associated with unexpected effects on actual database servers, and therefore IP SLAs responder functionality on a router can be configured to respond in place of the actual database server.

Cisco IOS IP SLAs Operation with Responders

Using an IP SLAs responder provides enhanced measurement accuracy—without the need for dedicated third-party external probe devices—and additional statistics that are not otherwise available via standard Internet Control Message Protocol (ICMP)-based measurements.

When a network manager configures an IP SLAs operation on the IP SLAs source, reaction conditions can also be defined, and the operation can be scheduled to be run for a period of time to gather statistics. The source uses the IP SLAs control protocol to communicate with the responder before sending test packets. To increase security of IP SLAs control messages, message digest 5 (MD5) authentication can be used to secure the control protocol exchange.

The following sequence of events occurs for each IP SLAs operation that requires a responder on the target, as illustrated in Figure 5-7:

1. At the start of the control phase, the IP SLAs source sends a control message with the configured IP SLAs operation information to IP SLAs control port UDP 1967 on the target router (the responder). The control message includes the protocol, port number, and duration of the operation. In Figure 5-7, UDP port 2020 is used for the IP SLAs test packets.

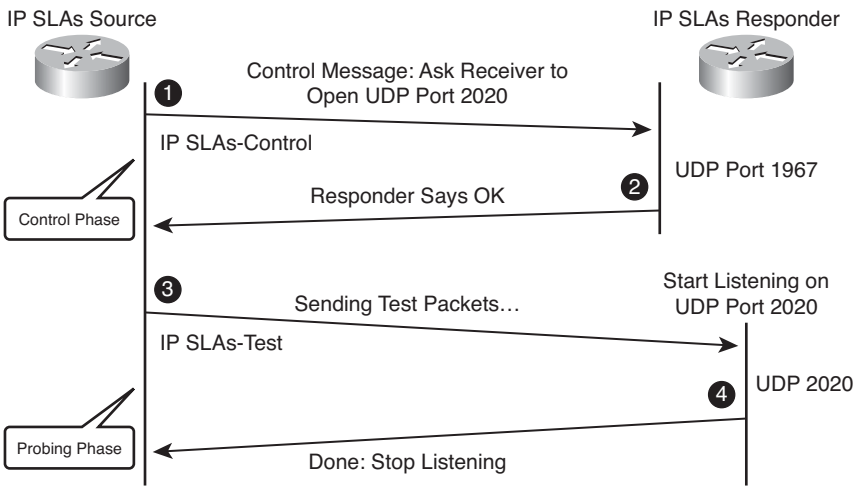


Figure 5-7 *IP SLAs Operation with a Responder.*

If MD5 authentication is enabled, the MD5 checksum is sent with the control message, and the responder verifies the MD5 checksum. If the authentication fails, the responder returns an "authentication failure" message.

2. If the responder processes the control message, it sends an "OK" message to the source and listens on the port specified in the control message for a specified duration. If the responder cannot process the control message, it returns an error. If the IP SLAs source does not receive a response from the responder, it tries to retransmit the control message. It will eventually time out if it does not receive a response.

Note The responder is capable of responding to multiple IP SLAs measurement operations that try to connect to the same port number.

3. If an "OK" message is returned, the IP SLAs operation on the source moves to the probing phase where it sends one or more test packets to the responder to compute response times. In Figure 5-7, the test messages are sent on control port 2020.

4. The responder accepts the test packets and responds. Based on the type of operation, the responder may add an "in" time stamp and an "out" time stamp in the response packet payload to account for the CPU time spent measuring unidirectional packet loss, latency, and jitter. These time stamps help the IP SLAs source make accurate assessments of one-way delay and processing time in target routers. The responder disables the user-specified port after it responds to the IP SLAs measurements packet or when the specified time expires.

Cisco IOS IP SLAs with Responder Time Stamps

Figure 5-8 illustrates the use of time stamps in round-trip calculations in an operation using an IP SLAs responder. The IP SLAs source uses four time stamps for the round-trip time (RTT) calculation.

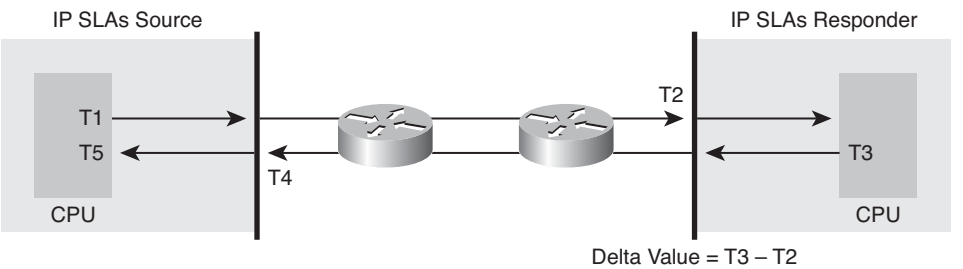


Figure 5-8 *Time Stamps in an IP SLAs Operation with a Responder.*

The IP SLAs source sends a test packet at time T1.

Because of other high-priority processes, routers might take tens of milliseconds to process incoming packets. For example, the reply to a test packet might be sitting in a queue waiting to be processed. To account for this delay, the IP SLAs responder includes both the receipt time (T2) and the transmitted time (T3) in the response packet. The time stamps are accurate to submilliseconds.

The IP SLAs source subtracts T2 from T3 to determine the delta value—the time spent processing the test packet in the IP SLAs responder. The delta value is subtracted from the overall RTT.

The same principle is applied by IP SLAs source. The incoming time stamp (T4) is taken at the interrupt level to allow for greater accuracy in the RTT calculation. The T4 time stamp, rather than the T5 time stamp (when the packet is processed), is used in the RTT calculation.

The two time stamps taken in the IP SLAs responder also allow one-way delay, jitter, and directional packet loss to be tracked. These statistics are critical for understanding asynchronous network behavior. To calculate these one-way delay measurements, the source and target need to be synchronized to the same clock source, and therefore, the Network Time Protocol (NTP) must be configured on both.

Configuring Path Control Using IOS IP SLAs

This section describes some of the commands used to configure path control using IOS IP SLAs.

The following steps are required to configure Cisco IOS IP SLAs functionality:

Step 1. Define one or more IP SLAs operations (or probes).

Step 2. Define one or more tracking objects, to track the state of IOS IP SLAs operations.

Step 3. Define the action associated with the tracking object.

These steps are detailed in the following sections.

Configuring Cisco IOS IP SLAs Operations

This section describes some of the configuration commands used to define IP SLAs operations.

Use the **ip sla** *operation-number* global configuration command (or the **ip sla monitor** *operation-number* global configuration command) to begin configuring a Cisco IOS IP SLAs operation and to enter IP SLA configuration mode (or rtr configuration mode). The *operation-number* is the identification number of the IP SLAs operation you want to configure.

Note Effective with Cisco IOS Release 12.4(4)T, 12.2(33)SB, and 12.2(33)SXI, the **ip sla monitor** command is replaced by the **ip sla** command.

Note From IP SLA configuration mode, a variety of commands can be entered, as shown here:

```
R1(config-ip-sla)#?
IP SLAs entry configuration commands:
  dhcp          DHCP Operation
  dns           DNS Query Operation
  exit          Exit Operation Configuration
  frame-relay   Frame-relay Operation
  ftp           FTP Operation
  http          HTTP Operation
  icmp-echo     ICMP Echo Operation
  icmp-jitter   ICMP Jitter Operation
  path-echo     Path Discovered ICMP Echo Operation
  path-jitter   Path Discovered ICMP Jitter Operation
  slm           SLM Operation
  tcp-connect   TCP Connect Operation
  udp-echo      UDP Echo Operation
  udp-jitter    UDP Jitter Operation
  voip          Voice Over IP Operation

R1(config-ip-sla)#
```

The ICMP echo operation is used to cause ICMP echo requests to be sent to a destination to check connectivity. Use the **icmp-echo** {*destination-ip-address* | *destination-hostname*} [**source-ip** {*ip-address* | *hostname*} | **source-interface** *interface-name*] IP SLA configuration mode command (or the **type echo protocol ipIcmpEcho** {*destination-ip-address* | *destination-hostname*} [**source-ipaddr** {*ip-address* | *hostname*} | **source-interface** *interface-name*] rtr configuration mode command) to configure an IP SLAs ICMP echo operation. The parameters of these commands are defined in Table 5-3.

Table 5-3 **icmp-echo** and **type echo protocol ipIcmpEcho** *Commands*

Parameter	Description
destination-ip-address | *destination-hostname*	Destination IPv4 or IPv6 address or hostname.
source-ip {*ip-address* | *hostname*} (or **source-ipaddr** {*ip-address* | *hostname*})	(Optional) Specifies the source IPv4 or IPv6 address or hostname. When a source IP address or hostname is not specified, the IP SLAs chooses the IP address nearest to the destination.
source-interface *interface-name*	(Optional) Specifies the source interface for the operation.

Note Effective with Cisco IOS Release 12.4(4)T, 12.2(33)SB, and 12.2(33)SXI, the **type echo protocol ipIcmpEcho** command is replaced by the **icmp-echo** command.

Use the **frequency** *seconds* IP SLA configuration submode command (or rtr configuration submode command) to set the rate at which a specified IP SLAs operation repeats. (For example, this command can be entered within the icmp-echo command mode.) The *seconds* parameter is the number of seconds between the IP SLAs operations; the default is 60.

Use the **timeout** *milliseconds* IP SLA configuration submode command (or rtr configuration submode command) to set the amount of time a Cisco IOS IP SLAs operation waits for a response from its request packet. (For example, this command can be entered within the icmp-echo command mode.) The *milliseconds* parameter is the number of milliseconds (ms) the operation waits to receive a response from its request packet. It is recommended that the value of the *milliseconds* parameter be based on the sum of both the maximum RTT value for the packets and the processing time of the IP SLAs operation.

After the Cisco IP SLAs operation is configured, it needs to be scheduled. Use the **ip sla schedule** *operation-number* [**life** {**forever** | *seconds*}] [**start-time** {*hh*:*mm*[:*ss*] [*month day* | *day month*] | **pending** | **now** | **after** *hh*:*mm*:*ss*}] [**ageout** *seconds*] [**recurring**] global configuration mode command (or the **ip sla monitor schedule** *operation-number* [**life** {**forever** | *seconds*}] [**start-time** {*hh:mm*[:*ss*] [*month day* | *day month*] | **pending** | **now** | **after** *hh:mm:ss*}] [**ageout** *seconds*] [**recurring**] global configuration mode command) to configure the scheduling parameters for a single Cisco IOS IP SLAs operation. The parameters of these commands are defined in Table 5-4.

Table 5-4 **ip sla schedule** and **ip sla monitor schedule** *Commands*

Parameter	Description
operation-number	Number of the IP SLAs operation to schedule.
life forever	(Optional) Schedules the operation to run indefinitely.
life *seconds*	(Optional) Number of seconds the operation actively collects information. The default is 3600 seconds (1 hour).
start-time	(Optional) Time when the operation starts.
hh:*mm*[:*ss*]	Specifies an absolute start time using hour, minute, and (optionally) second. Use the 24-hour clock notation. For example, start time 01:02 means "start at 1:02 a.m.," and start time 13:01:30 means "start at 1:01 p.m. and 30 seconds." The current day is implied unless you specify a *month* and *day*.
month	(Optional) Name of the month to start the operation in. If month is not specified, the current month is used. Use of this argument requires that a day be specified. You can specify the month by using either the full English name or the first three letters of the month.
day	(Optional) Number of the day (in the range 1 to 31) to start the operation on. If a day is not specified, the current day is used. Use of this argument requires that a month be specified.
pending	(Optional) No information is collected. This is the default value.
now	(Optional) Indicates that the operation should start immediately.
after *hh*:*mm*:*ss*	(Optional) Indicates that the operation should start *hh* hours, *mm* minutes, and *ss* seconds after this command was entered.
ageout *seconds*	(Optional) Number of seconds to keep the operation in memory when it is not actively collecting information. The default is 0 seconds (never ages out).
recurring	(Optional) Indicates that the operation will start automatically at the specified time and for the specified duration every day.

Note Effective with Cisco IOS Release 12.4(4)T, 12.2(33)SB, and 12.2(33)SXI, the **ip sla monitor schedule** command is replaced by the **ip sla schedule** command.

Configuring Cisco IOS IP SLAs Tracking Objects

This section examines some of the configuration commands used to define tracking objects, to track the state of IOS IP SLAs operations.

Use the **track** *object-number* **ip sla** *operation-number* {**state** | **reachability**} global configuration command (or the **track** *object-number* **rtr** *operation-number* {**state** | **reachability**} global configuration command) to track the state of an IOS IP SLAs operation, and enter track configuration mode. The parameters of these commands are defined in Table 5-5.

Table 5-5 **track ip sla** and **track rtr** *Commands*

Parameter	Description
object-number	Object number representing the object to be tracked. The range is from 1 to 500.
operation-number	Number used for the identification of the IP SLAs operation you are tracking.
state	Tracks the operation return code.
reachability	Tracks whether the route is reachable.

Note Effective with Cisco IOS Release 12.4(20)T, 12.2(33)SXI1, 12.2(33)SRE and Cisco IOS XE Release 2.4, the **track rtr** command is replaced by the **track ip sla** command.

Use the **delay** {**up** *seconds* [**down** *seconds*] | [**up** *seconds*] **down** *seconds*} track configuration command to specify a period of time to delay communicating state changes of a tracked object. The parameters of this command are defined in Table 5-6.

Table 5-6 **delay** *Commands*

Parameter	Description
up	Time to delay the notification of an up event.
down	Time to delay the notification of a down event.
seconds	Delay value, in seconds. The range is from 0 to 180. The default is 0.

Configuring the Action Associated with the Tracking Object

This section describes one of the configuration commands used to define the action associated with the tracking object.

Use the **ip route** *prefix mask* {*ip-address* | *interface-type interface-number* [*ip-address*]} [**dhcp**] [*distance*] [**name** *next-hop-name*] [**permanent** | **track** *number*] [**tag** *tag*]

global configuration command to establish a static route that tracks an object. The parameters of this command are defined in Table 5-7.

Table 5-7 **ip route** *Command*

Parameter	Description
prefix	IP route prefix for the destination.
mask	Prefix mask for the destination.
ip-address	IP address of the next hop that can be used to reach that network.
interface-type interface-number	Network interface type and interface number.
dhcp	(Optional) Enables a Dynamic Host Configuration Protocol (DHCP) server to assign a static route to a default gateway (option 3). Note that you specify the **dhcp** keyword for each routing protocol.
distance	(Optional) Administrative distance. The default administrative distance for a static route is 1.
name *next-hop-name*	(Optional) Applies a name to the specified route.
permanent	(Optional) Specifies that the route will not be removed, even if the interface shuts down.
track *number*	(Optional) Associates a track object with this route. Valid values for the *number* argument range from 1 to 500.
tag *tag*	(Optional) Tag value that can be used as a "match" value for controlling redistribution via route maps.

The next section introduces some of the commands used to verify path control using IOS IP SLAs. The section after that illustrates two examples of IOS IP SLAs configuration and verification.

Verifying Path Control Using IOS IP SLAs

This section describes some of the commands used to verify path control using IOS IP SLAs.

Use the **show ip sla configuration** [*operation*] command (or the **show ip sla monitor configuration** [*operation*] command) to display configuration values including all defaults for all Cisco IOS IP SLAs operations, or for a specified operation. The *operation* parameter is the number of the IP SLAs operation for which the details will be displayed.

Note Effective with Cisco IOS Release 12.4(20)T, 12.2(33)SXI1, 12.2(33)SRE and Cisco IOS XE Release 2.4, the **show ip sla monitor configuration** command is replaced by the **show ip sla configuration** command.

Use the **show ip sla statistics** [*operation-number*] [**details**] command (or the **show ip sla monitor statistics** [*operation-number*] [**details**] command) to display the current operational status and statistics of all Cisco IOS IP SLAs operations, or of a specified operation. The parameters of these commands are defined in Table 5-8.

Table 5-8 **show ip sla statistics** and **show ip sla monitor statistics** *Commands*

Parameter	Description
operation-number	(Optional) Number of the operation for which operational status and statistics are displayed.
details	(Optional) Operational status and statistics are displayed in greater detail.

Note Effective with Cisco IOS Release 12.4(20)T, 12.2(33)SXI1, 12.2(33)SRE and Cisco IOS XE Release 2.4, the **show ip sla monitor statistics** command is replaced by the **show ip sla statistics** command.

Examples of Path Control Using Cisco IOS IP SLAs

This section uses two examples to illustrate IOS IP SLAs configuration and verification.

Tracking Reachability to Two ISPs

Figure 5-9 illustrates a scenario in which Customer A is multihoming to two ISPs. Customer A is not using BGP with the ISPs; instead, it is using static default routes. Two default static routes with different administrative distances are configured, so that the link to ISP-1 is the primary link and the link to ISP-2 is the backup link. The static default route with the lower administrative distance will be preferred and injected into the routing table.

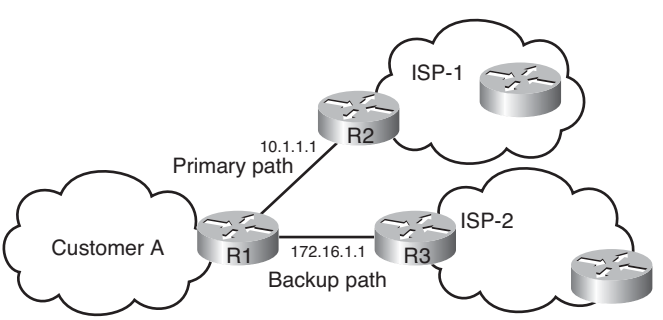


Figure 5-9 *Tracking Reachability to Two ISPs Example Network.*

However, if there is a problem with the ISP-1 router or with its connectivity toward the Internet but its interface to Customer A is still up, all traffic from Customer A will still go to that ISP. This traffic may then get lost within the ISP. The solution to this issue is the Cisco IOS IP SLAs functionality, which can be used to continuously check the reachability of a specific destination (such as a provider edge [PE] router interface, the ISP's DNS server, or any other specific destination) and conditionally announce the default route only if the connectivity is verified.

The Cisco IOS IP SLAs configuration of R1 is provided in Example 5-2.

Example 5-2 *Cisco IOS IP SLAs Configuration of Router R1 in Figure 5-9*

```
R1(config)#ip sla monitor 11
R1(config-rtr)#type echo protocol ipIcmpEcho 10.1.1.1 source-interface
FastEthernet0/0
R1(config-rtr-echo)#frequency 10
R1(config-rtr-echo)#exit
R1(config)#ip sla monitor schedule 11 life forever start-time now
R1(config)#track 1 rtr 11 reachability
R1(config-track)#exit
R1(config)#ip route 0.0.0.0 0.0.0.0 10.1.1.1 2 track 1


R1(config)#ip sla monitor 22
R1(config-rtr)#type echo protocol ipIcmpEcho 172.16.1.1 source-interface
FastEthernet0/1
R1(config-rtr-echo)#frequency 10
R1(config-rtr-echo)#exit
R1(config)#ip sla monitor schedule 22 life forever start-time now
R1(config)#track 2 rtr 22 reachability
R1(config-track)#exit
R1(config)#ip route 0.0.0.0 0.0.0.0 172.16.1.1 3 track 2
```

The first step in this configuration defines the probe; probe 11 is defined by the **ip sla monitor 11** command. The test defined with the **type echo protocol ipIcmpEcho 10.1.1.1 source-interface FastEthernet0/0** command specifies that the ICMP echoes are sent to destination 10.1.1.1 (R2) to check connectivity, with the Fast Ethernet 0/0 interface used as the source interface. The **frequency 10** command schedules the connectivity test to repeat every 10 seconds. The **ip sla monitor schedule 11 life forever start-time now** command defines the start and end time of the connectivity test for probe 11; the start time is now and it will continue forever.

The second step defines the tracking object, which is linked to the probe from the first step. The **track 1 rtr 11 reachability** command specifies that object 1 is tracked; it is linked to probe 11 (defined in the first step) so that the reachability of the 10.1.1.1 is tracked.

The last step defines an action based on the status of the tracking object. The **ip route 0.0.0.0 0.0.0.0 10.1.1.1 2 track 1** command conditionally configures the default route, via 10.1.1.1, with an administrative distance of 2, if the result of tracking object 1 is true. Thus, if 10.1.1.1 is reachable, a static default route via 10.1.1.1 with an administrative distance of 2, is installed in the routing table.

This scenario requires the configuration of two probes, two tracking objects, and two conditionally announced default routes. The second set of configuration commands in Example 5-2 is almost the same as the first set. Probe 22, defined by the **ip sla monitor 22** command, defines the test condition for the reachability of the backup ISP destination address 172.16.1.1, using Fast Ethernet 0/1 as the source address. The test is every 10 seconds, from now to forever. Tracking object 2 is related to the second probe, as defined by the **track 2 rtr 22 reachability** command. The default route configured, via 172.16.1.1, is using a higher administrative distance of 3, because the backup ISP is to be used only if the primary ISP is not available. This default route is offered to the routing table if the result of tracking object 2 is true.

Tracking DNS Server Reachability in the Two ISPs

Figure 5-10 illustrates the network for this example scenario. R3 represents a branch office connected to two ISPs. In this scenario Cisco IOS IP SLAs are used to track the reachability to the DNS servers (with IP addresses 10.0.8.1 and 10.0.8.2) and tie the results to the static default routes on R3. If there is a DNS server failure, the Cisco IOS IP SLAs probes will fail, the static default route to that DNS will be removed, and all traffic will be rerouted toward the other ISP.

Note This network was created in a lab to simulate a branch office scenario. The DNS server addresses are simulated by loopback 0 interfaces on R1 and R2. EIGRP is running between R1, R2, and R3.

The following steps detail the implementation and verification of Cisco IOS IP SLAs in this example:

Step 1. Verify reachability to the DNS servers.

Step 2. Configure Cisco IOS IP SLAs.

Step 3. Verify Cisco IOS IP SLAs operations.

Step 4. Configure tracking options.

Step 5. Configure static default routes or PBR that are tied to object tracking (the DNS servers).

Step 6. Verify dynamic operations and routing changes when the tracked objects fail.

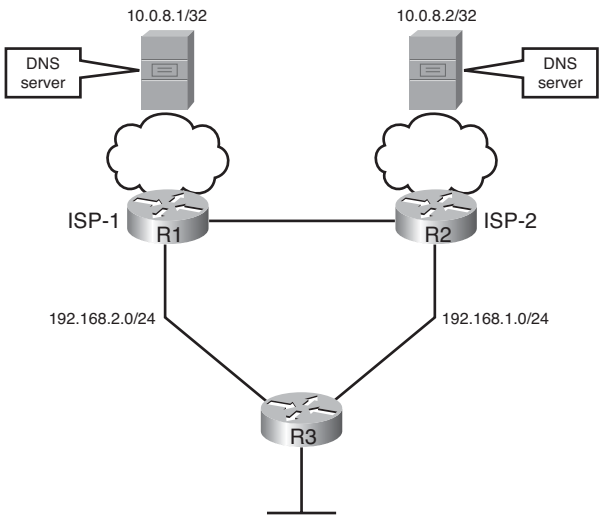


Figure 5-10 *Tracking Reachability to DNS Servers in the Two ISPs Example Network.*

Example 5-3 illustrates the results of the reachability verification tests from R3 to the DNS servers.

Example 5-3 *Results of Reachability Tests to DNS Servers from R3*

```
R3#ping 10.0.8.1

Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 10.0.8.1, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 28/31/36 ms
R3#ping 10.0.8.2

Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 10.0.8.2, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 28/29/32 ms
R3#
```

After confirming that the reachability tests are successful, the Cisco IOS IP SLAs are configured. The configuration is shown in Example 5-4. The **ip sla monitor 99** command is used to create an ICMP echo probe on R3 to the first DNS server; the operation number 99 is locally significant only to the router. (Note that there are many other types of

probes other than the ICMP echo probes that could be created.) The **frequency 10** command schedules the connectivity test to repeat every 10 seconds. The probe is scheduled to start now, and to run forever. A second probe, 100, is similarly created to test connectivity to the second DNS server.

Example 5-4 *Configuration of Router R3 in Figure 5-10*

```
ip sla monitor 99
 type echo protocol ipIcmpEcho 10.0.8.1
 frequency 10
ip sla monitor schedule 99 life forever start-time now

ip sla monitor 100
 type echo protocol ipIcmpEcho 10.0.8.2
 frequency 10
ip sla monitor schedule 100 life forever start-time now
```

The IP SLAs configuration is verified next, using the **show ip sla monitor configuration** command. The partial output is shown in Example 5-5, illustrating the details of the configuration of operation 99. This output confirms that the operation is an echo operation to 10.0.8.1 with a frequency of 10 seconds, and that it has already started (the start time has already passed).

Example 5-5 **show ip sla monitor configuration** *Output on R3*

```
R3(config)#do show ip sla monitor configuration
SA Agent, Infrastructure Engine-II
Entry number: 99
Owner:
Tag:
 Type of operation to perform: echo
 Target address: 10.0.8.1
Request size (ARR data portion): 28
Operation timeout (milliseconds): 5000
Type of Service parameters: 0x0
Verify data: No
 Operation frequency (seconds): 10
 Next Scheduled Start Time: Start Time already passed
Group Scheduled: FALSE
Life (seconds): Forever
Entry Ageout (seconds): never
Recurring (Starting Everyday): FALSE
Status of entry (SNMP RowStatus): Active
Threshold (milliseconds): 5000
Number of statistic hours kept: 2
```

```
Number of statistic distribution buckets kept: 1
Statistic distribution interval (milliseconds): 20
Number of history Lives kept: 0
Number of history Buckets kept: 15
— -More— -
```

The **show ip sla monitor statistics** command is used next, to display the number of successes, failures, and the results of the latest operations. The output is shown in Example 5-6, and it confirms that operation 99 has succeeded 16 times already, had no failures, and the last operation returned an "OK" result. Operation 100 has succeeded 15 times, had no failures, and its last operation also returned an "OK" result.

Example 5-6 **show ip sla monitor statistics** *Output on R3*

```
R3(config)#do show ip sla monitor statistics
Round trip time (RTT)   Index 99
         Latest RTT: 20 ms
Latest operation start time: *18:07:10.306 UTC Fri May 24 2002
 Latest operation return code: OK
 Number of successes: 16
 Number of failures: 0
Operation time to live: Forever

Round trip time (RTT)   Index 100
         Latest RTT: 19 ms
Latest operation start time: *18:07:12.006 UTC Fri May 24 2002
 Latest operation return code: OK
 Number of successes: 15
 Number of failures: 0
Operation time to live: Forever

R3(config)#
```

The next step is to configure tracking objects, as illustrated in Example 5-7. The first tracking object is tied to IP SLAs object 99 and has 10 seconds of down delay and 1 second of up delay, representing the level of sensitivity to changes of tracked objects. The delay helps to alleviate the affect of flapping objects, those that are going down and up rapidly. In this case, if the DNS server fails momentarily and comes back up within 10 seconds, there is no impact. The **ip route** command creates a static default route via 192.168.2.2 (R1) that appears or disappears, depending on the success or failure of the IP SLAs operation. Notice that this command reference the tracking object number 1, which in turn reference IP SLAs operation number 99.

The second tracking object is tied to IP SLAs object 100 and has a similar configuration.

Example 5-7 *Tracking Object Configuration of Router R3 in Figure 5-10*

```
track 1 rtr 99 reachability
 delay down 10 up 1
ip route 0.0.0.0 0.0.0.0 192.168.2.2 track 1

track 2 rtr 100 reachability
 delay down 10 up 1
ip route 0.0.0.0 0.0.0.0 192.168.1.2 track 2
```

Example 5-8 shows the static routes in the IP routing table. This output confirms that both static default routes currently appear in the routing table.

Example 5-8 *Routing Table on Router R3*

```
R3#show ip route static
S*   0.0.0.0 0.0.0.0 [1/0] via 192.168.2.2
                           via 192.168.1.2
```

To examine the routing behavior, IP routing debugging is enabled on R3, with the **debug ip routing** command. The DNS address on R2 is shut down. (Recall that in this example, the DNS address is simulated by interface loopback 0 on R2; thus a **shutdown** command on this interface is all that is required.)

Note The **debug ip routing** command may generate a significant amount of output.

The debug output on R3 is shown in Example 5-9. The EIGRP route to 10.0.8.2 is immediately deleted, and there are now no routes to 10.0.8.2. This is the object being tracked with the **track 2** command; it tracks reachability to IP SLAs object 100, which is an ICMP echo to 10.0.8.2. After about 10 seconds, the value specified in the **delay** command, the static default route via 192.168.1.2 (R2) is deleted.

Example 5-9 **debug ip routing** *Output on R3*

```
R3#
3w6d: RT: delete route to 10.0.8.2 via 192.168.1.2, eigrp metric [90/156160]
3w6d: RT: SET_LAST_RDB for 10.0.8.2 255.255.255.255
  OLD rdb: via 192.168.1.2, FastEthernet0/1

3w6d: RT: no routes to 10.0.8.2
3w6d: RT: NET-RED 10.0.8.2 255.255.255.255
3w6d: RT: delete subnet route to 10.0.8.2 255.255.255.255
3w6d: RT: NET-RED 10.0.8.2 255.255.255.255
R3#
```

```
3w6d: RT: del 0.0.0.0 via 192.168.1.2, static metric [1/0]
3w6d: RT: NET-RED 0.0.0.0 0.0.0.0
R3#
3w6d: RT: NET-RED 0.0.0.0 0.0.0.0
R3#
```

Debugging is disabled, and the statistics are viewed again, using the **show ip sla monitor statistics** command, as displayed in Example 5-10. This output confirms that there have been 11 failures on the IP SLAs object 100; these are failures in the ICMP echo to 10.0.8.2. The latest return code is "Timeout."

Example 5-10 show ip sla statistics *Output on R3*

```
R3#show ip sla monitor statistics
<Output omitted>
Round Trip Time (RTT) for             Index 100
         Latest RTT: NoConnection/Busy/Timeout
Latest operation start time: *17:29:26.572 UTC Sun Aug 2 2009
 Latest operation return code: Timeout
Number of successes: 80
 Number of failures: 11
Operation time to live: Forever


R3#
```

The static routes in the IP routing table now are shown in Example 5-11. This output confirms that only one static default remains, via 192168.2.2 (R1).

Example 5-11 show ip route static *Output on R3*

```
R3#show ip route static
S*   0.0.0.0 0.0.0.0 [1/0] via 192.168.2.2
R3#
```

To examine the routing behavior when connectivity to the R2 DNS is restored, IP routing debugging is enabled on R3 again, with the **debug ip routing** command, and the DNS address on R2 is enabled by performing a **no shutdown** command on the loopback 0 interface on R2.

The debug output on R3 is shown in Example 5-12. The EIGRP route to 10.0.8.2 comes up, and almost immediately the default static route via 192.168.1.2 (R2) comes up.

Example 5-12 debug ip routing *Output on R3*

```
3w6d: RT: SET_LAST_RDB for 10.0.8.2 255.255.255.255
  NEW rdb: via 192.168.1.2

3w6d: RT: add 10.0.8.2 255.255.255.255 via 192.168.1.2, eigrp metric [90/156160]
3w6d: RT: NET-RED 10.0.8.2 255.255.255.255
R3#
3w6d: RT: add 0.0.0.0 0.0.0.0 via 192.168.1.2, static metric [1/0]
3w6d: RT: NET-RED 0.0.0.0 0.0.0.0
3w6d: RT: NET-RED 0.0.0.0 0.0.0.0
R3#
3w6d: RT: NET-RED 0.0.0.0 0.0.0.0
R3#
```

The routing table now is shown in Example 5-13; both static default routes are there. Full connectivity has been restored.

Example 5-13 show ip route static *Output on R3*

```
R3#show ip route static
S*   0.0.0.0 0.0.0.0 [1/0] via 192.168.2.2
                           via 192.168.1.2
```

An alternative solution for this example network, using PBR, is presented at the end of the next section, after PBR is detailed.

In summary, there are many possibilities available with object tracking and Cisco IOS IP SLAs. As shown in these examples, you can base a probe on reachability, changing routing operations and path control based on the ability to reach an object. You can also use Cisco IOS IP SLAs with Cisco IOS Optimized Edge Routing (OER) to allow paths to be changed based on network conditions such as delay, load, and so forth. (Cisco IOS OER allows the best exit path to be selected, based on a defined policy, and is described briefly in the "Cisco IOS Optimized Edge Routing" section, later in this chapter.)

In deploying the Cisco IOS IP SLAs solution, the impact of the additional probe traffic being generated should also be considered, including how that traffic affects bandwidth utilization and congestion levels. Tuning the configuration (for example with the **delay** and **frequency** commands) becomes critical to mitigate possible issues related to excessive transitions and route changes in the presence of flapping tracked objects.

Implementing Path Control Using Policy-Based Routing

Chapter 4 describes route maps and how you can use them for route filtering. This section describes another use for route maps, with PBR. PBR enables the administrator to define a routing policy other than basic destination-based routing using the routing table. With PBR, route maps can be used to match source and destination addresses, protocol

types, and end-user applications. When a match occurs, a **set** command can be used to define items, such as the interface or next-hop address to which the packet should be sent.

Using PBR to Control Path Selection

In modern high-performance internetworks, organizations need the freedom to implement packet forwarding and routing according to their own defined policies in a way that goes beyond traditional routing protocol concerns.

Routers normally forward packets to destination addresses based on information in their routing tables. By using PBR, introduced in Cisco IOS Release 11.0, you can implement policies that selectively cause packets to take different paths based on source address, protocol types, or application types. Therefore, PBR overrides the router's normal routing procedures.

PBR also provides a mechanism to mark packets with different types of service (ToS). This feature can be used in conjunction with Cisco IOS queuing techniques so that certain kinds of traffic can receive preferential service.

PBR provides an extremely powerful, simple, and flexible tool to implement solutions in cases where legal, contractual, or political constraints dictate that traffic be routed through specific paths. Benefits you can achieve by implementing PBR include the following:

- **Source-based transit provider selection—**ISPs and other organizations can use PBR to route traffic originating from different sets of users through different Internet connections across policy routers.
- **QoS—**Organizations can provide QoS to differentiated traffic by setting the ToS values in the IP packet headers in routers at the periphery of the network and then leveraging queuing mechanisms to prioritize traffic in the network's core or backbone. This setup improves network performance by eliminating the need to classify the traffic explicitly at each WAN interface in the network's core or backbone.
- **Cost savings—**Using PBR, an organization can direct the bulk traffic associated with a specific activity to use a higher-bandwidth, high-cost link for a short time and to continue basic connectivity over a lower-bandwidth, low-cost link for interactive traffic.
- **Load sharing—**In addition to the dynamic load-sharing capabilities offered by destination-based routing that the Cisco IOS Software has always supported, network managers can implement policies to distribute traffic among multiple paths based on the traffic characteristics.

Configuring PBR

Configuring PBR involves configuring a route map with **match** and **set** commands and then applying the route map to the interface.

When configuring PBR, it is important to note that PBR is applied to *incoming* packets. Enabling PBR causes the router to evaluate all packets incoming on the interface using a route map configured for that purpose.

The steps required to implement path control include the following:

1. Choose the path control tool to use. Path control tools manipulate or bypass the IP routing table. For PBR, **route-map** commands are used.
2. Implement the traffic-matching configuration, specifying which traffic will be manipulated; **match** commands are used within route maps.
3. Define the action for the matched traffic, using **set** commands within route maps.
4. Optionally, fast-switched PBR or Cisco Express Forwarding (CEF)-switched PBR can be enabled. Fast-switched PBR must be enabled manually. CEF-switched PBR is automatically enabled when CEF switching is enabled (which it is by default in recent IOS versions) and PBR is enabled.
5. Apply the route map to incoming traffic or to traffic locally generated on the router.
6. Verify path control results, using **show** commands.

You can configure the route map statements used for PBR as **permit** or **deny**. The following defines how these options work:

- If the statement is marked as **deny**, a packet meeting the match criteria is not policy-based routed. Instead, it is sent through the normal forwarding channels; in other words, destination-based routing is performed.
- Only if the statement is marked as **permit** and the packet meets all the match criteria are the **set** commands applied.
- If no match is found in the route map, the packet is *not* dropped; it is forwarded through the normal routing channel, which means that destination-based routing is performed.
- If you do not want to revert to normal forwarding but instead want to drop a packet that does not match the specified criteria, configure a **set** statement to route the packets to interface null 0 as the last entry in the route map.

PBR match Commands

IP standard or extended access lists can be used to establish PBR match criteria using the **match ip address** {*access-list-number* | *name*} [*...access-list-number* | *name*] | **prefix-list** *prefix-list-name* [*..prefix-list-name*] route map configuration command, as explained in Table 5-9. You can use a standard IP access list to specify match criteria for a packet's

source address. You can use extended access lists to specify match criteria based on source and destination addresses, application, protocol type, and ToS.

Table 5-9 **match ip address** *Command*

Parameter	Description
access-list-number | *name*	The number or name of a standard or extended access list to be used to test incoming packets. If multiple access lists are specified, matching any one results in a match.
prefix-list *prefix-list-name*	Specifies the name of a prefix list to be used to test packets. If multiple prefix lists are specified, matching any one results in a match.

Use the **match length** *min max* route map configuration command, explained in Table 5-10, to establish criteria based on the packet length between specified minimum and maximum values. For example, a network administrator could use the match length as the criterion that distinguishes between interactive and file transfer traffic, because file transfer traffic usually has larger packet sizes.

Table 5-10 **match length** *Command*

Parameter	Description
min	The packet's minimum Layer 3 length, inclusive, allowed for a match
max	The packet's maximum Layer 3 length, inclusive, allowed for a match

PBR **set** Commands

If the **match** statements are satisfied, you can use one or more of the **set** statements described in this section to specify the criteria for forwarding packets through the router.

The router evaluates the first four **set** commands for PBR shown in this section in the order they are presented. As soon as a destination address or interface has been chosen, other **set** commands for changing the destination address or interface are ignored. Note, however, that some of these commands affect only packets for which there is an *explicit* route in the routing table, and others affect only packets for which there is *no explicit* route in the routing table.

By default, a packet that is not affected by any of the **set** commands in a route map statement it has matched is not policy routed and is forwarded normally; in other words, destination-based routing is performed.

set ip next-hop Command

The **set ip next-hop** *ip-address* [*...ip-address*] route map configuration command provides a list of IP addresses used to specify the adjacent next-hop router in the path toward the destination to which the packets should be forwarded. If more than one IP address is specified, the first IP address associated with a currently up and connected interface is used to route the packets. Table 5-11 explains the **set ip next-hop** command.

Table 5-11 set ip next-hop *Command*

Parameter	Description
ip-address	The IP address of the next hop to which packets are output. It must be the address of an adjacent router.

The **set ip next-hop** command affects all packet types and is always used if configured.

Note With the **set ip next-hop** command, the routing table is checked only to determine whether the next hop can be reached. It is not checked to determine whether there is an explicit route for the packet's destination address.

set interface Command

The **set interface** *type number* [*... type number*] route map configuration command provides a list of interfaces through which the packets can be routed. If more than one interface is specified, the first interface that is found to be up is used to forward the packets. Table 5-12 explains this command.

Table 5-12 set interface *Command*

Parameter	Description
type number	The interface type and number to which packets are output

If there is *no* explicit route for the destination address of the packet in the routing table (for example, if the packet is a broadcast or is destined for an unknown address), the **set interface** command has no effect and is ignored. A default route in the routing table is *not* considered an explicit route for an unknown destination address.

set ip default next-hop Command

The **set ip default next-hop** *ip-address* [*...ip-address*] route map configuration command provides a list of default next-hop IP addresses. If more than one IP address is specified, the first next hop specified that appears to be adjacent to the router is used. The optional specified IP addresses are tried in turn. Table 5-13 explains this command.

Table 5-13 **set ip default next-hop** *Command*

Parameter	Description
ip-address	The IP address of the next hop to which packets are output. It must be the address of an adjacent router.

A packet is routed to the next hop specified by the **set ip default next-hop** command only if there is *no* explicit route for the packet's destination address in the routing table. A default route in the routing table is *not* considered an explicit route for an unknown destination address.

set default interface Command

The **set default interface** *type number* [*...type number*] route map configuration command provides a list of default interfaces. If no explicit route is available to the destination address of the packet being considered for policy routing, it is routed to the first up interface in the list of specified default interfaces. Table 5-14 provides information about this command.

Table 5-14 **set default interface** *Command*

Parameter	Description
type number	The interface type and number to which packets are output.

A packet is routed to the next hop specified by the **set default interface** command only if there is *no* explicit route for the packet's destination address in the routing table. A default route in the routing table is *not* considered an explicit route for an unknown destination address.

PBR also provides a mechanism to mark packets using the **set ip tos** and **set ip precedence** commands, as shown in the next two sections.

set ip tos Command

The **set ip tos** [*number* | *name*] route map configuration command is used to set some of the bits in the IP ToS field in the IP packet. The ToS field in the IP header is 8 bits long, with 5 bits for setting the class of service (CoS) and 3 bits for the IP precedence. The CoS bits are used to set the delay, throughput, reliability, and cost.

The **set ip tos** command is used to set the 5 CoS bits. Values 0 through 15 are used (one of the bits is reserved). Table 5-15 provides the names and numbers of the defined ToS values used in this command.

Table 5-15 set ip tos *Command*

Parameter *number* | *name*	Description
0 | normal	Sets the normal ToS
1 | min-monetary-cost	Sets the min-monetary-cost ToS
2 | max-reliability	Sets the max reliable ToS
4 | max-throughput	Sets the max throughput ToS
8 | min-delay	Sets the min delay ToS

set ip precedence Command

The **set ip precedence** [*number* | *name*] route map configuration command enables you to set the 3 IP precedence bits in the IP packet header. With 3 bits, you have eight possible values for the IP precedence; values 0 through 7 are defined. This command is used when implementing QoS and can be used by other QoS services, such as weighted fair queuing (WFQ) and weighted random early detection (WRED). Table 5-16 provides the names and numbers of the defined IP precedence values used in this command.

Table 5-16 set ip precedence *Command*

Parameter *number* | *name*	Description
0 | routine	Sets the routine precedence
1 | priority	Sets the priority precedence
2 | immediate	Sets the immediate precedence
3 | flash	Sets the Flash precedence
4 | flash-override	Sets the Flash override precedence
5 | critical	Sets the critical precedence
6 | internet	Sets the internetwork control precedence
7 | network	Sets the network control precedence

You can use the **set** commands in conjunction with each other.

Configuring PBR on an Interface

To identify a route map to use for policy routing on an interface, use the **ip policy route-map** *map-tag* interface configuration command. Table 5-17 explains the parameter.

Table 5-17 **ip policy route-map** *Command*

Parameter	Description
map-tag	The name of the route map to use for policy routing. It must match a map tag specified by a **route-map** command.

Remember that policy-based routing is configured on the interface that *receives* the packets, not on the interface from which the packets are sent.

Packets originating on the router are not normally policy routed. *Local policy routing* enables packets originating on the router to take a route other than the obvious shortest path. To identify a route map to use for local policy routing, use the **ip local policy route-map** *map-tag* global configuration command. Table 5-18 explains the parameter. This command applies the specified route map to packets originating on the router.

Table 5-18 **ip local policy route-map** *Command*

Parameter	Description
map-tag	The name of the route map to use for local policy routing. It must match a map tag specified by a **route-map** command.

Since Cisco IOS Release 12.0, IP PBR can now be fast switched. Before this feature, policy routing could only be process switched, which meant that on most platforms, the switching rate was approximately 1000 to 10,000 packets per second. This was not fast enough for many applications. Users who need policy routing to occur at faster speeds can now implement policy routing without slowing down the router.

PBR must be configured before you configure fast-switched policy routing. Fast switching of policy routing is disabled by default. To enable it, use the **ip route-cache policy** interface configuration command.

Fast-switched PBR supports all the **match** commands and most of the **set** commands, except for the following restrictions:

- The **set ip default next-hop** and **set default interface** commands are not supported.
- The **set interface** command is supported only over point-to-point links unless a route-cache entry exists using the same interface specified in the **set interface** command in the route map. Also, when process switching, the routing table is checked to determine whether the interface is on an appropriate path to the destination. The software does not make this check during fast switching. Instead, if the packet matches, the software blindly forwards the packet to the specified interface.

Note The **ip route-cache policy** command is strictly for fast-switched PBR, and therefore, not required for a CEF-switched PBR.

Verifying PBR

To display the route maps used for policy routing on the router's interfaces, use the **show ip policy** EXEC command.

To display configured route maps, use the **show route-map** [*map-name*] EXEC command, where *map-name* is an optional name of a specific route map.

Use the **debug ip policy** EXEC command to display IP policy routing packet activity. This command shows in detail what policy routing is doing. It displays information about whether a packet matches the criteria and, if so, the resulting routing information for the packet.

Note Because the **debug ip policy** command generates a significant amount of output, use it only when traffic on the IP network is low, so that other activity on the system is not adversely affected.

To discover the routes that the packets follow when traveling to their destination from the router, use the **traceroute** EXEC command. To change the default parameters and invoke an extended **traceroute**, enter the command without a destination argument. You are then stepped through a dialog to select the desired parameters.

To check host reachability and network connectivity, use the **ping** EXEC command. You can use the **ping** command's extended command mode to specify the supported header options by entering the command without any arguments.

PBR Examples

This section provides three examples of PBR.

Using PBR When Connecting Two ISPs

In Figure 5-11, Router A provides Internet access for a private enterprise and is connected to two different ISPs. This router is advertising a 0.0.0.0 default route into the enterprise network to avoid large routing tables.

Therefore, when traffic from the enterprise networks 10.1.0.0 and 10.2.0.0 reaches Router A, it can go to either ISP A or ISP B. The company prefers to have ISP A and ISP B receive approximately equal amounts of traffic. PBR is implemented on Router A to shape, or load balance, traffic from Router A to each of the ISPs. All traffic sourced from the 10.1.0.0 subnet is forwarded to ISP A if there is no specific route to the destination in the routing table (the default route is not used). All traffic sourced from the 10.2.0.0 subnet is forwarded to ISP B if there is no specific route to the destination in the routing table.

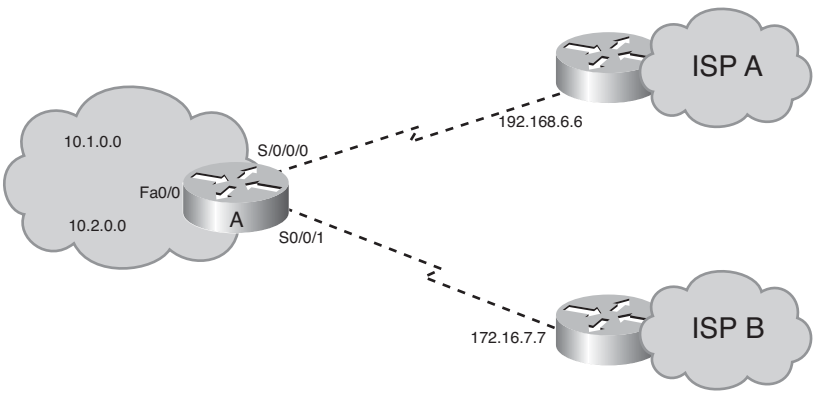


Figure 5-11 *Router A Is Connected to Two ISPs.*

Caution Remember, this policy provides for an outbound traffic policy from the enterprise to its ISPs only. It does not determine the inbound traffic policy for Router A. It is possible that traffic from 10.1.0.0 going out to ISP A will receive responses through ISP B.

Example 5-14 shows the configuration for Router A. Route map *equal-access* is configured.

Example 5-14 *Configuration of Router A in Figure 5-11*

```
RouterA(config)#access-list 1 permit 10.1.0.0 0.0.255.255
RouterA(config)#access-list 2 permit 10.2.0.0 0.0.255.255

RouterA(config)#route-map equal-access permit 10
RouterA(config-route-map)#match ip address 1
RouterA(config-route-map)#set ip default next-hop 192.168.6.6

RouterA(config-route-map)#route-map equal-access permit 20
RouterA(config-route-map)#match ip address 2
RouterA(config-route-map)#set ip default next-hop 172.16.7.7

RouterA(config-route-map)#route-map equal-access permit 30
RouterA(config-route-map)#set default interface null0
RouterA(config-route-map)#exit
RouterA(config)#interface FastEthernet 0/0
RouterA(config-if)#ip address 10.1.1.1 255.255.255.0
RouterA(config-if)#ip policy route-map equal-access
RouterA(config-if)#exit
RouterA(config)#interface Serial 0/0/0
RouterA(config-if)#ip address 192.168.6.5 255.255.255.0
RouterA(config-if)#exit
```

continues

Example 5-14 *Configuration of Router A in Figure 5-11 (continued)*

```
RouterA(config)#interface Serial 0/0/1
RouterA(config-if)#ip address 172.16.7.6 255.255.255.0
```

The **ip policy route-map equal-access** command is applied to the Fast Ethernet 0/0 interface, the *incoming* interface receiving the packets to be policy-routed.

Sequence number 10 in route map equal-access is used to match all packets sourced from any host in subnet 10.1.0.0. If there is a match, and if the router has no explicit route for the packet's destination, it is sent to next-hop address 192.168.6.6 (ISP A's router).

Sequence number 20 in route map equal-access is used to match all packets sourced from any host in subnet 10.2.0.0. If there is a match, and if the router has no explicit route for the packet's destination, it is sent to next-hop address 172.16.7.7 (ISP B's router).

Sequence number 30 in route map equal-access is used to drop all traffic not sourced from subnet 10.1.0.0 or 10.2.0.0. The null 0 interface is a route to nowhere; traffic is dropped.

The outputs shown in Examples 5-15, 5-16, and 5-17 are from Router A in Figure 5-11. Example 5-15 provides an example of **show ip policy** command output, indicating that the route map called equal-access is used for PBR on the router's Fast Ethernet 0/0 interface.

Example 5-15 **show ip policy** *on Router A in Figure 5-11*

```
RouterA#show ip policy
Interface          Route map
FastEthernet0/0    equal-access
```

Example 5-16 provides an example of **show route-map** command output, indicating that three packets have matched sequence 10 of the equal-access route map.

Example 5-16 **show route-map** *on Router A in Figure 5-11*

```
RouterA#show route-map
route-map equal-access, permit, sequence 10
  Match clauses:
    ip address (access-lists): 1
  Set clauses:
    ip default next-hop 192.168.6.6
 Policy routing matches: 3 packets, 168 bytes
route-map equal-access, permit, sequence 20
  Match clauses:
    ip address (access-lists): 2
  Set clauses:
    ip default next-hop 172.16.7.7
route-map equal-access, permit, sequence 30
```

```
Set clauses:
    default interface null0
```

Example 5-17 provides an example of the **debug ip policy** command output. The output indicates that a packet from 10.1.1.1 destined for 172.19.1.1 has been received on interface Fast Ethernet 0/0 and that it is policy-routed on Serial 0/0/0 to next hop 192.168.6.6 (because the source address of 10.1.1.1 matches line 10 of route map equal-access).

Example 5-17 **debug ip policy** *on Router A in Figure 5-11*

```
RouterA#debug ip policy
Policy routing debugging is on

11:51:25: IP: s=10.1.1.1 (FastEthernet0/0), d=172.19.1.1, len 100, policy match
11:51:25: IP: route map equal-access, item 10, permit
11:51:25: IP: s=10.1.1.1 (FastEthernet0/0), d=172.19.1.1 (Serial0/0/0), len 100,
policy routed
11:51:25: IP: FastEthernet0/0/0 to Serial0/0/0 192.168.6.6
```

Note The **show logging** command shows the logging buffer, including the output of the **debug** command.

Using PBR Based on Source Address

In Figure 5-12, Router A has a policy that packets with a source address of 192.168.2.1 (on the other side of Router B) should go out to Router C's interface Serial 0/0/1, 172.17.1.2 (via Router A's S0/0/1 interface). All other packets should be routed according to their destination address. Example 5-18 shows the relevant part of the configuration for Router A.

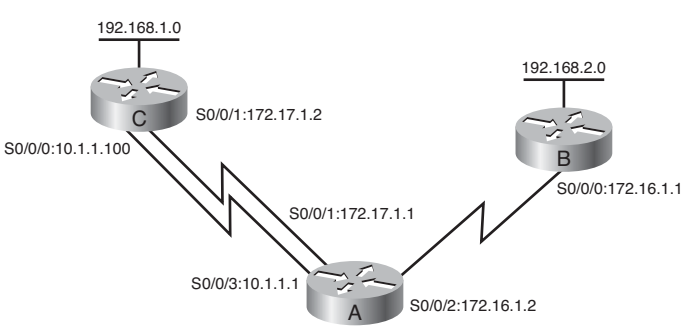


Figure 5-12 *Router A Has a Policy That Packets from 192.168.2.1 Go to Router C's Interface S0/0/1.*

Example 5-18 *Configuration of Router A in Figure 5-12*

```
RouterA(config)#interface Serial0/0/2
RouterA(config-if)#ip address 172.16.1.2 255.255.255.0
RouterA(config-if)#ip policy route-map test
RouterA(config-if)#route-map test permit 10
RouterA(config-route-map)#match ip address 1
RouterA(config-route-map)#set ip next-hop 172.17.1.2
RouterA(config-route-map)#exit
RouterA(config)#access-list 1 permit 192.168.2.1 0.0.0.0
```

Router A's Serial 0/0/2 interface, where packets from 192.168.2.1 go *into* Router A, is configured to do policy routing with the **ip policy route-map** command. The route map test is used for this policy routing. It tests the IP addresses in packets against access list 1 to determine which packets will be policy-routed.

Access list 1 specifies that packets with a source address of 192.168.2.1 are policy routed. Packets that match access list 1 are sent to the next-hop address 172.17.1.2, which is Router C's Serial 0/0/1 interface. All other packets are forwarded normally, according to their destination address. (Recall that access lists have an implicit **deny any** at the end, so no other packets are permitted by access list 1.)

The outputs shown in Examples 5-19, 5-20, and 5-21 are from Router A in Figure 5-12. Example 5-19 provides an example of the **show ip policy** command output. It indicates that the route map called test is used for policy routing on the router's interface Serial 0/0/2.

Example 5-19 **show ip policy** *Output on Router A in Figure 5-12*

```
RouterA#show ip policy
Interface           Route map
Serial0/0/2         test
```

The **show route-map** command, shown in Example 5-20, indicates that three packets have matched sequence 10 of the test route map.

Example 5-20 **show route-map** *Output on Router A in Figure 5-12*

```
RouterA#show route-map
route-map test, permit, sequence 10
  Match clauses:
    ip address (access-lists): 1
  Set clauses:
    ip next-hop 172.17.1.2
 Policy routing matches: 3 packets, 168 bytes
```

Example 5-21 provides an example of the output of the **debug ip policy** command. The output indicates that a packet from 172.16.1.1 destined for 192.168.1.1 was received on interface Serial 0/0/2 and that it was rejected by the policy on that interface. The packet is routed normally (by destination). Another packet, from 192.168.2.1 destined for 192.168.1.1, was later received on the same interface, Serial 0/0/2. This packet matched the policy on that interface and therefore was policy routed and sent out interface Serial 0/0/1 to 172.17.1.2.

Example 5-21 *Example of* **debug ip policy** *on Router A in Figure 5-12*

```
RouterA#debug ip policy
Policy routing debugging is on

...
11:50:51: IP: s=172.16.1.1 (Serial0/0/2), d=192.168.1.1 (Serial0/0/3), len 100,
  policy rejected — normal forwarding
...
11:51:25: IP: s=192.168.2.1 (Serial0/0/2), d=192.168.1.1, len 100, policy match
 11:51:25: IP: route map test, item 10, permit
11:51:25: IP: s=192.168.2.1 (Serial0/0/2), d=192.168.1.1 (Serial0/0/1), len 100,
  policy routed
 11:51:25: IP: Serial0/0/2 to Serial0/0/1 172.17.1.2
```

Alternative Solution IP SLAs Configuration Example Using PBR

This section presents an alternative solution to the configuration of the R3 router in Figure 5-10 given earlier in this chapter in the "Examples of Path Control Using Cisco IOS IP SLAs" section. A partial configuration is shown in Example 5-22, providing just the configuration for reachability to the R1 router. Explanatory comments are provided within the configuration. (Configuration for reachability to the R2 router would be similar.) Using PBR allows the configuration to be very granular, to support other options. In this example, PBR points to a next-hop address that is tracked via Cisco IOS IP SLAs.

Example 5-22 *Partial Alternative Configuration for Router R3 in Figure 5-10*

```
!Configure the object to be tracked; object 1 will be up if the router
!can ping 10.0.8.1
ip sla 99
  icmp-echo 10.0.8.1
    frequency 10
    timeout 5000
ip sla schedule 99 start-time now life forever
!
track 1 rtr 99 reachability
!
!Enable policy routing using route map IP-SLA
```

continues

Example 5-22 *Partial Alternative Configuration for Router R3 in Figure 5-10 continued*

```
interface FastEthernet 0/0
  ip address 10.2.8.1 255.255.255.0
  ip policy route-map IP-SLA
!
!Configure a route-map to set the next-hop to 192.168.2.1 (R1) if
! object 1 is up. If object 1 is down, then policy routing fails
! and unicast routing will route the packet.
route-map IP-SLA
  set ip next-hop verify-availability 192.168.2.1 10 track 1
```

This configuration uses the **set ip next-hop verify-availability** [*next-hop-address sequence* **track** *object*] route-map configuration command to configure policy routing to verify the reachability of the next hop of a route map before the router performs policy routing to that next hop. Table 5-19 explains the parameters of this command.

Table 5-19 **set ip next-hop verify-availability** *Command*

Parameter	Description
next-hop-address	(Optional) IP address of the next hop to which packets will be forwarded.
sequence	(Optional) Sequence of next hops. The acceptable range is from 1 to 65535.
track	(Optional) The tracking method is track.
object	(Optional) Object number that the tracking subsystem is tracking. The acceptable range is from 1 to 500.

Because of the use of route maps, this type of configuration allows you more granularity to define, via access lists or prefix lists, which traffic classes will be subject to changes based on the results of the object tracking. For example routes for voice, mission-critical data, and other traffic types could be changed.

Advanced Path Control Tools

This section provides a brief overview of additional path control mechanisms that you might encounter in your enterprise networks.

Cisco IOS Optimized Edge Routing

Cisco IOS OER is intended for sites using multiple Internet or WAN service providers. Cisco IOS OER uses tools such as Cisco IOS IP SLAs to automatically detect network service degradation and to make dynamic routing decisions and adjustments based on criteria such as response time, packet loss, jitter, path availability, traffic load distribution, and so forth.

In contrast, normal routing, using routing protocols, focuses on detecting a routing path using static routing metrics, rather than the condition of the service over that path.

An example is illustrated in Figure 5-13. The Cisco IOS OER edge routers, called border routers, monitor information about route prefixes (using traditional routing protocols) and gather performance statistics over each external interface (in this example, using Cisco IOS IP SLAs).

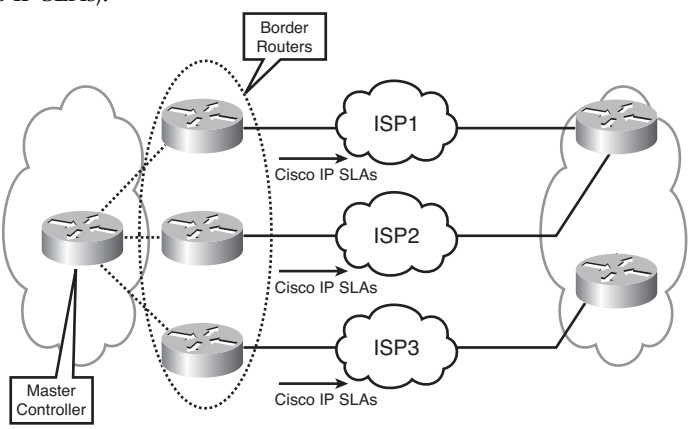


Figure 5-13 *Cisco IOS OER Operations.*

This information is periodically reported to another router called the master controller. If the prefixes and exit links comply with a configured policy based on performance and service metrics, routing remains as is. If not, the master controller makes a policy-based decision and notifies the border routers, which change the path, by such mechanisms as adding static routes or changing routing protocol parameters.

Virtualization

Virtualization is another advanced technology being used in enterprise networks that includes benefits such as traffic segregation across a common physical network infrastructure.

An example of virtualization is the use of virtual routing and forwarding (VRF) tables, which are virtual routing tables used to separate the routing function by group, on one physical router, as illustrated in Figure 5-14.

For example, employee routes could be kept separate from guest routes by using two different VRFs. These VRFs could also be associated with other virtualization and traffic segregation elements on the network, such as virtual LANs (VLANs), virtual private networks (VPNs), and generic routing encapsulation (GRE) tunnels, to provide an end-to-end, segregated path across the network. An example is illustrated in Figure 5-15, in which path control is based on a design decision to engineer different paths, end to end, with a variety of network virtualization technologies. In this figure, two business units are associated with two different VRFs on the end routers. These VRFs are associated with different VLANs and VPNs throughout the network, to provide an end-to-end segregated path across the network.

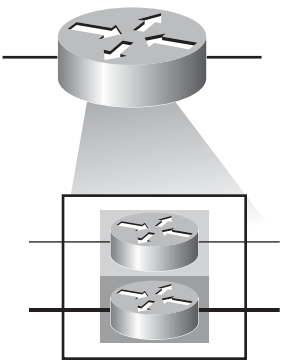


Figure 5-14 *VRF Creates Separate Virtual Routing Tables in One Physical Router.*

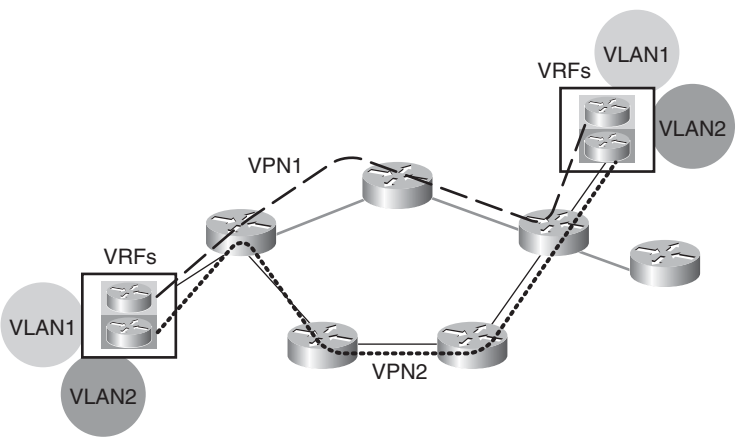


Figure 5-15 *Virtualization Technologies Used for Path Control.*

Cisco Wide Area Application Services

Cisco WAAS is a good example of the use of PBR to adjust the path of traffic based on advanced services for that traffic, to provide both scalability and high availability. Technologies such as Web Cache Communications Protocol (WCCP) perform a similar function, which is to have routers redirect normal traffic flows into Cisco WAAS devices, where a series of data reduction, flow optimization, and application acceleration services are implemented, and then have them route the flows back into their normal path across the WAN. This scenario is illustrated in the example in Figure 5-16. This use of path control is becoming common in networks with branch offices.

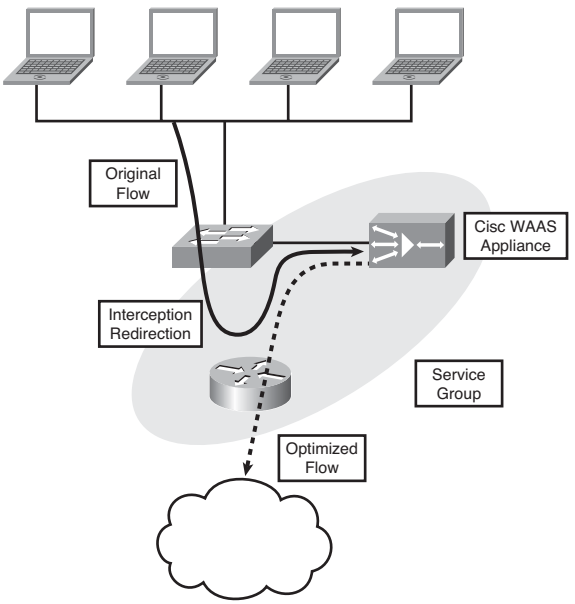


Figure 5-16 *WCCP Used for WAN Optimization.*

Summary

In this chapter, you learned about implementing path control. The chapter focused on the following topics:

- Redundant network considerations including resiliency, availability, adaptability, performance, support for network and application services, predictability, and asymmetric traffic.
- Path control tools including a good addressing design, redistribution and other routing protocol characteristics, passive interfaces, distribute lists, prefix lists, administrative distance, route maps, route tagging, offset lists, Cisco IOS IP SLAs, and PBR. (Advanced tools covered briefly include Cisco IOS OER, virtualization, and Cisco WAAS.)
- Offset lists, a mechanism for increasing incoming and outgoing metrics to routes learned via EIGRP or RIP. Configuration of offset lists is performed with the **offset-list** {*access-list-number* | *access-list-name*} {**in** | **out**} *offset* [*interface-type interface-number*] router configuration command. Verification of offset lists can be performed with the **traceroute** command, the **show ip route** command, and the **show ip eigrp topology** command.
- Cisco IOS IP SLAs, which use active traffic monitoring, generating traffic in a continuous, reliable, and predictable manner, to measure network performance. IOS IP

SLAs can be used in conjunction with other tools, including the following:

- Object tracking, to track the reachability of specified objects
- Cisco IOS IP SLAs probes, to send different types of probes toward the desired objects
- Route maps with PBR, to associate the results of the tracking to the routing process
- Static routes with tracking options, as a simpler alternative to PBR

- Cisco IOS IP SLAs terminology, including the following:
  - All the Cisco IOS IP SLAs measurement probe operations are configured on the IP SLAs *source*, either by the CLI or through an SNMP tool that supports IP SLAs operation. The source sends probe packets to the *target*.
  - There are two types of IP SLAs operations: those in which the target device is running the IP SLAs *responder* component, and those in which the target device is not running the IP SLAs responder component (such as a web server or IP host).
  - An *IP SLAs operation* is a measurement that includes protocol, frequency, traps, and thresholds.
- Configuring IOS IP SLAs, including the use of the following commands:
  - The **ip sla** *operation-number* global configuration command (or the **ip sla monitor** *operation-number* global configuration command) to begin configuring a Cisco IOS IP SLAs operation and enter IP SLA configuration mode (or rtr configuration mode).
  - The **icmp-echo** {*destination-ip-address* | *destination-hostname*} [**source-ip** {*ip-address* | *hostname*} | **source-interface** *interface-name*] IP SLA configuration mode command (or the **type echo protocol ipIcmpEcho** {*destination-ip-address* | *destination-hostname*} [**source-ipaddr** {*ip-address* | *hostname*} | **source-interface** *interface-name*] rtr configuration mode command) to configure an IP SLAs ICMP echo operation.
  - The **frequency** *seconds* IP SLA configuration submode command (or rtr configuration submode command) to set the rate at which a specified IP SLAs operation repeats.
  - The **timeout** *milliseconds* IP SLA configuration submode command (or rtr configuration submode command) to set the amount of time a Cisco IOS IP SLAs operation waits for a response from its request packet.
  - The **ip sla schedule** *operation-number* [**life** {**forever** | *seconds*}] [**start-time** {*hh*:*mm*[:*ss*] [*month day* | *day month*] | **pending** | **now** | **after** *hh*:*mm*:*ss*}] [**ageout** *seconds*] [**recurring**] global configuration mode command (or the **ip sla monitor schedule** *operation-number* [**life** {**forever** | *seconds*}] [**start-time**

{*hh:mm*[:*ss*] [*month day* | *day month*] | **pending** | **now** | **after** *hh:mm:ss*}] [**ageout** *seconds*] [**recurring**] global configuration mode command) to configure the scheduling parameters for a single Cisco IOS IP SLAs operation.

- The **track** *object-number* **ip sla** *operation-number* {**state** | **reachability**} global configuration command (or the **track** *object-number* **rtr** *operation-number* {**state** | **reachability**} global configuration command) to track the state of an IOS IP SLAs operation, and enter track configuration mode.
- The **delay** {**up** *seconds* [**down** *seconds*] | [**up** *seconds*] **down** *seconds*} track configuration command to specify a period of time to delay communicating state changes of a tracked object.
- The **ip route** *prefix mask* {*ip-address* | *interface-type interface-number* [*ip-address*]} [**dhcp**] [*distance*] [**name** *next-hop-name*] [**permanent** | **track** *number*] [**tag** *tag*] global configuration command to establish a static route that tracks an object.

- Verifying Cisco IOS IP SLAs, including the use of the **show ip sla configuration** [*operation*] command (or the **show ip sla monitor configuration** [*operation*] command), and the **show ip sla statistics** [*operation-number*] [**details**] command (or the **show ip sla monitor statistics** [*operation-number*] [**details**] command).
- Using PBR to control path selection, providing benefits including source-based transit provider selection, QoS, cost savings, and load sharing. PBR is applied to *incoming* packets; enabling PBR causes the router to evaluate all packets incoming on the interface using a route map configured for that purpose.
- Configuring and verifying PBR, including the following steps:
  - Choose the path control tool to use; for PBR, **route-map** commands are used.
  - Implement the traffic-matching configuration, specifying which traffic will be manipulated; **match** commands are used within route maps.
  - Define the action for the matched traffic, using **set** commands within route maps.
  - Optionally, fast-switched PBR or CEF-switched PBR can be enabled. Fast-switched PBR must be enabled manually. CEF-switched PBR is automatically enabled when CEF switching is enabled and PBR is enabled.
  - Apply the route map to incoming traffic or to traffic locally generated on the router.
  - Verify path control results, using **show** commands.
- PBR **match** commands, including the following:
  - The **match ip address** {*access-list-number* | *name*} [*...access-list-number* | *name*] route map configuration command
  - The **match length** *min max* route map configuration command

- PBR **set** commands, including the following four which are evaluated in this order (as soon as a destination address or interface has been chosen, other **set** commands for changing the destination address or interface are ignored):
  - The **set ip next-hop** *ip-address* [*...ip-address*] route map configuration command, which affects all packet types and is always used if configured.
  - The **set interface** *type number* [*...type number*] route map configuration command. If there is *no* explicit route for the destination address of the packet in the routing table (for example, if the packet is a broadcast or is destined for an unknown address), the **set interface** command has no effect and is ignored. A default route in the routing table is *not* considered an explicit route for an unknown destination address.
  - The **set ip default next-hop** *ip-address* [*...ip-address*] route map configuration command. A packet is routed to the next hop specified by the **set ip default next-hop** command only if there is *no* explicit route for the packet's destination address in the routing table. A default route in the routing table is *not* considered an explicit route for an unknown destination address.
  - The **set default interface** *type number* [*...type number*] route map configuration command. A packet is routed to the next hop specified by the **set default interface** command only if there is *no* explicit route for the packet's destination address in the routing table. A default route in the routing table is *not* considered an explicit route for an unknown destination address.
- Other PBR **set** commands, including the following:
  - The **set ip tos** [*number* | *name*] route map configuration command, used to set the 5 CoS bits. Values 0 through 15 are used; one of the bits is reserved.
  - The **set ip precedence** [*number* | *name*] route map configuration command, used to set the 3 IP precedence bits in the IP packet header.
  - The **set ip next-hop verify-availability** [*next-hop-address sequence* **track** *object*] route-map configuration command to configure policy routing to verify the reachability of the next hop of a route map before the router performs policy routing to that next hop.
- Commands to configure PBR on an interface, including the following:
  - The **ip policy route-map** *map-tag* interface configuration command, configured on the interface that *receives* the packets, not on the interface from which the packets are sent
  - The **ip local policy route-map** *map-tag* global configuration command, to apply a route map to packets originating on the router
- Commands to verify PBR, including the **show ip policy** command, the **show route-map** [*map-name*] command, the **debug ip policy** command, the **traceroute** command, and **ping** command.

- Advanced path control tools, including the following:
  - Cisco IOS OER, which uses tools such as Cisco IOS IP SLAs to automatically detect network service degradation and to make dynamic routing decisions and adjustments based on criteria such as response time, packet loss, jitter, path availability, traffic load distribution, and so forth
  - Virtualization, such as the use of VRF tables, VLANs, VPNs, and GRE tunnels
  - Cisco WAAS, including the use of WCCP to redirect normal traffic flows into Cisco WAAS devices

References

For additional information, see these resources:

- "Cisco IOS Software Releases 12.4 Mainline" support page: http://www.cisco.com/en/US/products/ps6350/tsd_products_support_series_home.html
- The Cisco IOS Command Reference: http://www.cisco.com/en/US/products/ps6350/prod_command_reference_list.html
- The Cisco IOS IP SLAs Command Reference: http://www.cisco.com/en/US/docs/ios/ipsla/command/reference/sla_book.html
- Cisco Optimized Edge Routing (OER) home page: http://www.cisco.com/en/US/tech/tk1335/tsd_technology_support_sub-protocol_home.html

Review Questions

Answer the following questions, and then see Appendix A, "Answers to Review Questions," for the answers.

1. List some considerations for redundant networks.
2. How does address summarization help keep a network stable?
3. List some path control tools.
4. Indicate whether each statement is referring to OSPF or EIGRP.

Statement	Routing Protocol
Metric can be changed only for external routes at redistribution points.	
Next hop can be set for all routes under various conditions.	
Can be configured only on ABRs and ASBRs.	
Unequal-cost load balancing is available.	
All routes can be tagged.	

5. Select the true statements.
   a. An empty prefix list denies all prefixes.
   b. Offset lists increase the incoming metric of routes.
   c. A distribute list allows an access list to be applied to routing updates.
   d. If a prefix is permitted, the route is used. If a prefix is denied, the route is not used.
   e. Offset lists decrease the incoming metric of routes.
6. In the **offset-list** command, what is the *access-list-number* or *access-list-name* parameter used for?
7. Fill in the blank: _____use active traffic monitoring, generating traffic in a continuous, reliable, and predictable manner, to measure network performance
8. What is a Cisco IOS IP SLAs responder?
9. Which ports are used when an IP SLAs source sends to an IP SLAs responder?
10. Select the true statements about IP SLAs.
    a. Operations are configured on the IP SLAs source.
    b. Operations are configured on the IP SLAs responder.
    c. A Cisco IOS device can be an IP SLAs responder.
    d. A Cisco IOS device can be an IP SLAs source.
    e. A web server can be an IP SLAs source.
11. Write the command to track the reachability of IOS IP SLAs operation number 100 with object number 2.
12. Write the command to start IP SLAs operation number 100 immediately and have it never end.
13. What are some benefits of policy-based routing (PBR)?
14. To which packets on an interface is PBR applied?
15. When a route map is used for PBR, which of the following are true statements?
    a. If the statement is marked as **deny**, a packet meeting the match criteria is sent through the normal forwarding channels.
    b. If the statement is marked as **deny**, a packet meeting the match criteria is dropped.
    c. If the statement is marked as **permit** and the packet meets all the match criteria, the **set** commands are applied.

d. If the statement is marked as **permit** and the packet meets all the match criteria, the packet is sent through the normal forwarding channels.

e. If no match is found in the route map, the packet is not dropped.

f. If no match is found in the route map, the packet is dropped.

16. In which order are the following **set** commands evaluated?

set default interface

set interface

set ip default next-hop

set ip next-hop

17. When is the **set default interface** command used?

18. When are the **ip policy route-map** and **ip local policy route-map** commands used?

19. What does the **set ip next-hop verify-availability** command do?

20. How does OER differ from normal routing?

21. What are virtual routing forwarding (VRF) tables?

This page intentionally left blank

Index

Numerics

A

B

C

D

E

F

G

H

I–J

K–L

M

N

O

P

Q

R

S

T

U

V

W–Z