CISCO

Official Cert Guide

Learn, prepare, and practice for exam success

CCIE Routing and Switching v5.0

Volume 1

Fifth Edition

NARBIK KOCHARIANS, CCIE® No. 12410

PETER PALÚCH, CCIE® No. 23527

ciscopress.com

CCIE Routing and Switching v5.0 Official Cert Guide, Volume 1

Fifth Edition

Narbik Kocharians, CCIE No. 12410
Peter Palúch, CCIE No. 23527

Cisco Press

800 East 96th Street

Indianapolis, IN 46240

CCIE Routing and Switching v5.0 Official Cert Guide, Volume 1, Fifth Edition

Narbik Kocharians, CCIE No. 12410

Peter Palúch, CCIE No. 23527

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

Printed in the United States of America

Second Printing February 2015

Library of Congress Control Number: 2014944345

ISBN-13: 978-1-58714-396-0

ISBN-10: 1-58714-396-8

Warning and Disclaimer

This book is designed to provide information about Cisco CCIE Routing and Switching Written Exam, No. 400-101. 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.

Special Sales

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

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

For questions about sales outside the U.S., please contact 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 email 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

Business Operation Manager, Cisco Press: Jan Cornelssen

Executive Editor: Brett Bartow

Managing Editor: Sandra Schroeder

Senior Development Editor: Christopher Cleveland

Senior Project Editor: Tonya Simpson

Copy Editor: John Edwards

Technical Editors: Paul Negron, Sean Wilkins

Editorial Assistant: Vanessa Evans

Cover Designer: Mark Shirar

Composition: Tricia Bronkella

Indexer: Tim Wright

Proofreader: Chuck Hutchinson

About the Authors

Narbik Kocharians, CCIE No. 12410 (Routing and Switching, Security, SP), is a Triple CCIE with more than 32 years of experience in the IT industry. He has designed, implemented, and supported numerous enterprise networks. Narbik is the president of Micronics Training Inc. (www.micronicstraining.com), where he teaches CCIE R&S and SP boot camps.

Peter Palúch, CCIE No. 23527 (Routing and Switching), is an assistant professor, Cisco Networking Academy instructor, and instructor trainer at the Faculty of Management Science and Informatics, University of Zilina, Slovakia. Peter has cooperated in various educational activities in Slovakia and abroad, focusing on networking and Linux-based network server systems. He is also active at the Cisco Support Community, holding the Cisco Designated VIP award in LAN & WAN Routing and Switching areas since the award program inception in 2011. Upon invitation by Cisco in 2012, Peter joined two Job Task Analysis groups that assisted defining the upcoming CCIE R&S and CCNP R&S certification exam topics. Peter holds an M.Sc. degree in Applied Informatics and a doctoral degree in the area of VoIP quality degradation factors. Together with his students, Peter has started the project of implementing the EIGRP routing protocol into the Quagga open-source routing software suite, and has been driving the effort since its inception in 2013.

About the Technical Reviewers

Paul Negron, CCIE No. 14856, CCSI No. 22752, has been affiliated with networking technologies for 17 years and has been involved with the design of core network services for a number of service providers, such as Comcast, Qwest, British Telecom, and Savvis to name a few. He currently instructs all the CCNP Service Provider–level courses, including Advanced BGP, MPLS, and the QoS course. Paul has six years of experience with satellite communications as well as ten years of experience with Cisco platforms.

Sean Wilkins is an accomplished networking consultant for SR-W Consulting (www.sr-wconsulting.com) and has been in the field of IT since the mid 1990s, working with companies such as Cisco, Lucent, Verizon, and AT&T as well as several other private companies. Sean currently holds certifications with Cisco (CCNP/CCDP), Microsoft (MCSE), and CompTIA (A+ and Network+). He also has a Master of Science in information technology with a focus in network architecture and design, a Master of Science in organizational management, a Master's Certificate in network security, a Bachelor of Science in computer networking, and Associates of Applied Science in computer information systems. In addition to working as a consultant, Sean spends most of his time as a technical writer and editor for various companies; check out this work at his author website: www.infodispersion.com.

Dedications

From Narbik Kocharians:

I would like to dedicate this book to my wife, Janet, for her love, encouragement, and continuous support, and to my dad for his words of wisdom.

From Peter Palúch:

To my family, students, colleagues, and friends.

Acknowledgments

From Narbik Kocharians:

First, I would like to thank God for giving me the opportunity and ability to write, teach, and do what I truly enjoy doing. Also, I would like to thank my family, especially my wife of 29 years, Janet, for her constant encouragement and help. She does such an amazing job of interacting with students and handling all the logistics of organizing classes as I focus on teaching. I also would like to thank my children, Chris, Patrick, Alexandra, and my little one, Daniel, for their patience.

A special thanks goes to Mr. Brett Bartow for his patience and our constant changing of the deadlines. It goes without saying that the technical editors and reviewers did a phenomenal job; thank you very much. Finally, I would like to thank all my students who inspire me every day, and you, for reading this book.

From Peter Palúch:

The opportunity to cooperate on the new edition of this book has been an honor and privilege beyond words for me. Wendell Odom, who has so gracefully and generously passed the torch to us, was the key person in introducing me to the Cisco Press representatives as a possible author, and I will be forever indebted to him for all the trust he has blessed us with. I have strived very much to live up to the unparalelled high level of content all previous authors have maintained throughout all editions of this book, and I would like to sincerely thank all of them for authoring such a great book that has significantly helped me achieve my certification in the first place.

My next immense thank you goes to Brett Bartow, the executive editor for this book. Brett's inviting and forthcoming attitude throughout the time of editing the book, compounded with his patience and understanding for my ever-moving (and constantly missed) deadlines, is second to none. He has done all in his power to help us, the authors, without compromising the quality of the work.

I would not have been able to complete my work on this volume without the endless support of my family. They have encouraged me, supported me, and gone out of their way to accommodate my needs. Words are not enough to express my gratitude.

Psalm 127, whose musical setting in works of Monteverdi, Handel, or Vivaldi I have come to admire, begins with words "Unless the Lord build the house, they labor in vain who build." Indeed, if it was not first and foremost the Lord's blessing and help throughout, this work would not have been finished successfully. To my Lord and Savior, Jesus Christ—thank you!

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

The Cisco Certified Internetwork Expert (CCIE) certification might be the most challenging and prestigious of all networking certifications. It has received numerous awards and certainly has built a reputation as one of the most difficult certifications to earn in all of the technology world. Having a CCIE certification opens doors professionally and typically results in higher pay and looks great on a resume.

Cisco currently offers several CCIE certifications. This book covers the version 5.0 exam blueprint topics of the written exam for the CCIE Routing and Switching certification. The following list details the currently available CCIE certifications at the time of this book's publication; check www.cisco.com/go/ccie for the latest information. The certifications are listed in the order in which they appear on the web page:

- CCDE
- CCIE Collaboration
- CCIE Data Center
- CCIE Routing & Switching
- CCIE Security
- CCIE Service Provider
- CCIE Service Provider Operations
- CCIE Wireless

Each of the CCDE and CCIE certifications requires the candidate to pass both a written exam and a one-day, hands-on lab exam. The written exam is intended to test your knowledge of theory, protocols, and configuration concepts that follow good design practices. The lab exam proves that you can configure and troubleshoot actual gear.

Why Should I Take the CCIE Routing and Switching Written Exam?

The first and most obvious reason to take the CCIE Routing and Switching written exam is that it is the first step toward obtaining the CCIE Routing and Switching certification. Also, you cannot schedule a CCIE lab exam until you pass the corresponding written exam. In short, if you want all the professional benefits of a CCIE Routing and Switching certification, you start by passing the written exam.

The benefits of getting a CCIE certification are varied and include the following:

- Better pay
- Career-advancement opportunities

- Applies to certain minimum requirements for Cisco Silver and Gold Channel Partners, as well as those seeking Master Specialization, making you more valuable to Channel Partners
- Better movement through the problem-resolution process when calling the Cisco TAC
- Prestige
- Credibility for consultants and customer engineers, including the use of the Cisco CCIE logo

The other big reason to take the CCIE Routing and Switching written exam is that it recertifies an individual's associate-, professional-, and expert-level Cisco certifications, regardless of his or her technology track. Recertification requirements do change, so please verify the requirements at www.cisco.com/go/certifications.

CCIE Routing and Switching Written Exam 400-101

The CCIE Routing and Switching written exam, at the time of this writing, consists of a two-hour exam administered at a proctored exam facility affiliated with Pearson VUE (www.vue.com/cisco). The exam typically includes approximately 100 multiple-choice questions. No simulation questions are currently part of the written exam.

As with most exams, everyone wants to know what is on the exam. Cisco provides general guidance as to topics on the exam in the CCIE Routing and Switching written exam blueprint, the most recent copy of which can be accessed from www.cisco.com/go/ccie.

Cisco changes both the CCIE written and lab blueprints over time, but Cisco seldom, if ever, changes the exam numbers. However, exactly this change occurred when the CCIE Routing and Switching blueprint was refreshed for v5.0. The previous written exam for v4.0 was numbered 350-001; the v5.0 written exam is identified by 400-101.

Table I-1 lists the CCIE Routing and Switching written exam blueprint 5.0 at press time. Table I-1 also lists the chapters that cover each topic.

Table I-1 *CCIE Routing and Switching Written Exam Blueprint*

Topics	Book Volume	Book Chapter
1.0 Network Principles		
1.1 Network theory		
1.1.a Describe basic software architecture differences between IOS and IOS XE		
1.1.a (i) Control plane and Forwarding plane	1	1
1.1.a (ii) Impact on troubleshooting and performance	1	1
1.1.a (iii) Excluding a specific platform's architecture	1	1

Topics	Book Volume	Book Chapter
1.1.b Identify Cisco Express Forwarding concepts		
1.1.b (i) RIB, FIB, LFIB, Adjacency table	1	6
1.1.b (ii) Load-balancing hash	1	6
1.1.b (iii) Polarization concept and avoidance	1	6
1.1.c Explain general network challenges		
1.1.c (i) Unicast flooding	1	4
1.1.c (ii) Out-of-order packets	1	4
1.1.c (iii) Asymmetric routing	1	4
1.1.c (iv) Impact of micro burst	1	4
1.1.d Explain IP operations		
1.1.d (i) ICMP unreachable, redirect	1	4
1.1.d (ii) IPv4 options, IPv6 extension headers	1	4
1.1.d (iii) IPv4 and IPv6 fragmentation	1	4
1.1.d (iv) TTL	1	4
1.1.d (v) IP MTU	1	4
1.1.e Explain TCP operations		
1.1.e (i) IPv4 and IPv6 PMTU	1	4
1.1.e (ii) MSS	1	4
1.1.e (iii) Latency	1	4
1.1.e (iv) Windowing	1	4
1.1.e (v) Bandwidth delay product	1	4
1.1.e (vi) Global synchronization	1	4
1.1.e (vii) Options	1	4
1.1.f Explain UDP operations		
1.1.f (i) Starvation	1	4
1.1.f (ii) Latency	1	4
1.1.f (iii) RTP/RTCP concepts	1	4
1.2 Network implementation and operation		
1.2.a Evaluate proposed changes to a network		
1.2.a (i) Changes to routing protocol parameters	1	7–10
1.2.a (ii) Migrate parts of a network to IPv6	1	4

Topics	**Book Volume**	**Book Chapter**
1.2.a (iii) Routing protocol migration	1	6
1.2.a (iv) Adding multicast support	2	8
1.2.a (v) Migrate Spanning Tree Protocol	1	3
1.2.a (vi) Evaluate impact of new traffic on existing QoS design	2	3, 4, 5
1.3 Network troubleshooting		
1.3.a Use IOS troubleshooting tools		
1.3.a (i) debug, conditional debug	1	4
1.3.a (ii) ping, traceroute with extended options	1	4
1.3.a (iii) Embedded packet capture	2	9
1.3.a (iv) Performance monitor	1	5
1.3.b Apply troubleshooting methodologies		
1.3.b (i) Diagnose the root cause of networking issues (analyze symptoms, identify and describe root cause)	1	11
1.3.b (ii) Design and implement valid solutions according to constraints	1	11
1.3.b (iii) Verify and monitor resolution	1	11
1.3.c Interpret packet capture		
1.3.c (i) Using Wireshark trace analyzer	2	9
1.3.c (ii) Using IOS embedded packet capture	2	9
2.0 Layer 2 Technologies		
2.1 LAN switching technologies		
2.1.a Implement and troubleshoot switch administration		
2.1.a (i) Managing the MAC address table	1	1
2.1.a (ii) errdisable recovery	1	3
2.1.a (iii) L2 MTU	1	1
2.1.b Implement and troubleshoot Layer 2 protocols		
2.1.b (i) CDP, LLDP	1	3
2.1.b (ii) UDLD	1	3
2.1.c Implement and troubleshoot VLAN		
2.1.c (i) Access ports	1	2
2.1.c (ii) VLAN database	1	2
2.1.c (iii) Normal, extended VLAN, voice VLAN	1	2

Topics	Book Volume	Book Chapter
2.1.d Implement and troubleshoot trunking		
2.1.d (i) VTPv1, VTPv2, VTPv3, VTP pruning	1	2
2.1.d (ii) dot1Q	1	2
2.1.d (iii) Native VLAN	1	2
2.1.d (iv) Manual pruning	1	2
2.1.e Implement and troubleshoot EtherChannel		
2.1.e (i) LACP, PAgP, manual	1	3
2.1.e (ii) Layer 2, Layer 3	1	3
2.1.e (iii) Load balancing	1	3
2.1.e (iv) EtherChannel misconfiguration guard	1	3
2.1.f Implement and troubleshoot spanning tree		
2.1.f (i) PVST+/RPVST+/MST	1	3
2.1.f (ii) Switch priority, port priority, path cost, STP timers	1	3
2.1.f (iii) PortFast, BPDU Guard, BPDU Filter	1	3
2.1.f (iv) Loop Guard, Root Guard	1	3
2.1.g Implement and troubleshoot other LAN switching technologies		
2.1.g (i) SPAN, RSPAN, ERSPAN	1	1
2.1.h Describe chassis virtualization and aggregation technologies		
2.1.h (i) Multichassis	1	1
2.1.h (ii) VSS concepts	1	1
2.1.h (iii) Alternatives to STP	1	1
2.1.h (iv) Stackwise	1	1
2.1.h (v) Excluding specific platform implementation	1	1
2.1.i Describe spanning-tree concepts		
2.1.i (i) Compatibility between MST and RSTP	1	3
2.1.i (ii) STP dispute, STP Bridge Assurance	1	3
2.2 Layer 2 multicast		
2.2.a Implement and troubleshoot IGMP		
2.2.a (i) IGMPv1, IGMPv2, IGMPv3	2	7
2.2.a (ii) IGMP snooping	2	7
2.2.a (iii) IGMP querier	2	7

Topics	**Book Volume**	**Book Chapter**
2.2.a (iv) IGMP filter	2	7
2.2.a (v) IGMP proxy	2	7
2.2.b Explain MLD	2	8
2.2.c Explain PIM snooping	2	8
2.3 Layer 2 WAN circuit technologies		
2.3.a Implement and troubleshoot HDLC	2	6
2.3.b Implement and troubleshoot PPP		
2.3.b (i) Authentication (PAP, CHAP)	2	6
2.3.b (ii) PPPoE	2	6
2.3.b (iii) MLPPP	2	6
2.3.c Describe WAN rate-based Ethernet circuits		
2.3.c (i) Metro and WAN Ethernet topologies	2	6
2.3.c (ii) Use of rate-limited WAN Ethernet services	2	6
3.0 Layer 3 Technologies		
3.1 Addressing technologies		
3.1.a Identify, implement, and troubleshoot IPv4 addressing and subnetting		
3.1.a (i) Address types, VLSM	1	4
3.1.a (ii) ARP	1	4
3.1.b Identify, implement, and troubleshoot IPv6 addressing and subnetting		
3.1.b (i) Unicast, multicast	1	4
3.1.b (ii) EUI-64	1	4
3.1.b (iii) ND, RS/RA	1	4
3.1.b (iv) Autoconfig/SLAAC, temporary addresses (RFC 4941)	1	4
3.1.b (v) Global prefix configuration feature	1	4
3.1.b (vi) DHCP protocol operations	1	4
3.1.b (vii) SLAAC/DHCPv6 interaction	2	10
3.1.b (viii) Stateful, stateless DHCPv6	1	4
3.1.b (ix) DHCPv6 prefix delegation	1	4
3.2 Layer 3 multicast		
3.2.a Troubleshoot reverse path forwarding		

Topics	**Book Volume**	**Book Chapter**
3.2.a (i) RPF failure	2	8
3.2.a (ii) RPF failure with tunnel interface	2	8
3.2.b Implement and troubleshoot IPv4 protocol independent multicast		
3.2.b (i) PIM dense mode, sparse mode, sparse-dense mode	2	8
3.2.b (ii) Static RP, auto-RP, BSR	2	8
3.2.b (iii) Bidirectional PIM	2	8
3.2.b (iv) Source-specific multicast	2	8
3.2.b (v) Group-to-RP mapping	2	8
3.2.b (vi) Multicast boundary	2	8
3.2.c Implement and troubleshoot multicast source discovery protocol		
3.2.c (i) Intra-domain MSDP (anycast RP)	2	8
3.2.c (ii) SA filter	2	8
3.2.d Describe IPv6 multicast		
3.2.d (i) IPv6 multicast addresses	2	7
3.2.d (ii) PIMv6	2	8
3.3 Fundamental routing concepts		
3.3.a Implement and troubleshoot static routing	1	6
3.3.b Implement and troubleshoot default routing	1	7–11
3.3.c Compare routing protocol types		
3.3.c (i) Distance vector	1	7
3.3.c (ii) Link state	1	7
3.3.c (iii) Path vector	1	7
3.3.d Implement, optimize, and troubleshoot administrative distance	1	11
3.3.e Implement and troubleshoot passive interface	1	7–10
3.3.f Implement and troubleshoot VRF lite	2	11
3.3.g Implement, optimize, and troubleshoot filtering with any routing protocol	1	11
3.3.h Implement, optimize, and troubleshoot redistribution between any routing protocols	1	11
3.3.i Implement, optimize, and troubleshoot manual and auto summarization with any routing protocol	1	7–10

Topics	Book Volume	Book Chapter
3.3.j Implement, optimize, and troubleshoot policy-based routing	1	6
3.3.k Identify and troubleshoot suboptimal routing	1	11
3.3.l Implement and troubleshoot bidirectional forwarding detection	1	11
3.3.m Implement and troubleshoot loop prevention mechanisms		
3.3.m (i) Route tagging, filtering	1	11
3.3.m (ii) Split horizon	1	7
3.3.m (iii) Route poisoning	1	7
3.3.n Implement and troubleshoot routing protocol authentication		
3.3.n (i) MD5	1	7–10
3.3.n (ii) Key-chain	1	7–10
3.3.n (iii) EIGRP HMAC SHA2-256bit	1	8
3.3.n (iv) OSPFv2 SHA1-196bit	1	9
3.3.n (v) OSPFv3 IPsec authentication	1	9
3.4 RIP (v2 and v6)		
3.4.a Implement and troubleshoot RIPv2	1	7
3.4.b Describe RIPv6 (RIPng)	1	7
3.5 EIGRP (for IPv4 and IPv6)		
3.5.a Describe packet types		
3.5.a (i) Packet types (hello, query, update, and so on)	1	8
3.5.a (ii) Route types (internal, external)	1	8
3.5.b Implement and troubleshoot neighbor relationship		
3.5.b (i) Multicast, unicast EIGRP peering	1	8
3.5.b (ii) OTP point-to-point peering	1	8
3.5.b (iii) OTP route-reflector peering	1	8
3.5.b (iv) OTP multiple service providers scenario	1	8
3.5.c Implement and troubleshoot loop-free path selection		
3.5.c (i) RD, FD, FC, successor, feasible successor	1	8
3.5.c (ii) Classic metric	1	8
3.5.c (iii) Wide metric	1	8
3.5.d Implement and troubleshoot operations		
3.5.d (i) General operations	1	8

Topics	**Book Volume**	**Book Chapter**
3.5.d (ii) Topology table, update, query, active, passive	1	8
3.5.d (iii) Stuck in active	1	8
3.5.d (iv) Graceful shutdown	1	8
3.5.e Implement and troubleshoot EIGRP stub		
3.5.e (i) Stub	1	8
3.5.e (ii) Leak-map	1	8
3.5.f Implement and troubleshoot load balancing		
3.5.f (i) equal-cost	1	8
3.5.f (ii) unequal-cost	1	8
3.5.f (iii) add-path	1	8
3.5.g Implement EIGRP (multiaddress) named mode		
3.5.g (i) Types of families	1	8
3.5.g (ii) IPv4 address-family	1	8
3.5.g (iii) IPv6 address-family	1	8
3.5.h Implement, troubleshoot, and optimize EIGRP convergence and scalability		
3.5.h (i) Describe fast convergence requirements	1	8
3.5.h (ii) Control query boundaries	1	8
3.5.h (iii) IP FRR/fast reroute (single hop)	1	8
3.5.h (iv) Summary leak-map	1	8
3.5.h (v) Summary metric	1	8
3.6 OSPF (v2 and v3)		
3.6.a Describe packet types		
3.6.a (i) LSA types (1, 2, 3, 4, 5, 7, 9)	1	9
3.6.a (ii) Route types (N1, N2, E1, E2)	1	9
3.6.b Implement and troubleshoot neighbor relationship	1	9
3.6.c Implement and troubleshoot OSPFv3 address-family support		
3.6.c (i) IPv4 address-family	1	9
3.6.c (ii) IPv6 address-family	1	9
3.6.d Implement and troubleshoot network types, area types, and router types		
3.6.d (i) Point-to-point, multipoint, broadcast, nonbroadcast	1	9

Topics	**Book Volume**	**Book Chapter**
3.6.d (ii) LSA types, area type: backbone, normal, transit, stub, NSSA, totally stub	1	9
3.6.d (iii) Internal router, ABR, ASBR	1	9
3.6.d (iv) Virtual link	1	9
3.6.e Implement and troubleshoot path preference	1	9
3.6.f Implement and troubleshoot operations		
3.6.f (i) General operations	1	9
3.6.f (ii) Graceful shutdown	1	9
3.6.f (iii) GTSM (Generic TTL Security Mechanism)	1	9
3.6.g Implement, troubleshoot, and optimize OSPF convergence and scalability		
3.6.g (i) Metrics	1	9
3.6.g (ii) LSA throttling, SPF tuning, fast hello	1	9
3.6.g (iii) LSA propagation control (area types, ISPF)	1	9
3.6.g (iv) IP FRR/fast reroute (single hop)	1	9
3.6.g (v) LFA/loop-free alternative (multihop)	1	9
3.6.g (vi) OSPFv3 prefix suppression	1	9
3.7 BGP		
3.7.a Describe, implement, and troubleshoot peer relationships		
3.7.a (i) Peer-group, template	2	1
3.7.a (ii) Active, passive	2	1
3.7.a (iii) States, timers	2	1
3.7.a (iv) Dynamic neighbors	2	1
3.7.b Implement and troubleshoot IBGP and EBGP		
3.7.b (i) EBGP, IBGP	2	1
3.7.b (ii) 4-byte AS number	2	1
3.7.b (iii) Private AS	2	1
3.7.c Explain attributes and best-path selection	2	1
3.7.d Implement, optimize, and troubleshoot routing policies		
3.7.d (i) Attribute manipulation	2	2
3.7.d (ii) Conditional advertisement	2	2
3.7.d (iii) Outbound route filtering	2	2

Topics	**Book Volume**	**Book Chapter**
3.7.d (iv) Communities, extended communities	2	2
3.7.d (v) Multihoming	2	2
3.7.e Implement and troubleshoot scalability		
3.7.e (i) Route-reflector, cluster	2	2
3.7.e (ii) Confederations	2	2
3.7.e (iii) Aggregation, AS set	2	2
3.7.f Implement and troubleshoot multiprotocol BGP		
3.7.f (i) IPv4, IPv6, VPN address-family	2	2
3.7.g Implement and troubleshoot AS path manipulations		
3.7.g (i) Local AS, allow AS in, remove private AS	2	2
3.7.g (ii) Prepend	2	2
3.7.g (iii) Regexp	2	2
3.7.h Implement and troubleshoot other features		
3.7.h (i) Multipath	2	2
3.7.h (ii) BGP synchronization	2	2
3.7.h (iii) Soft reconfiguration, route refresh	2	2
3.7.i Describe BGP fast convergence features		
3.7.i (i) Prefix independent convergence	2	2
3.7.i (ii) Add-path	2	2
3.7.i (iii) Next-hop address tracking	2	2
3.8 IS-IS (for IPv4 and IPv6)		
3.8.a Describe basic IS-IS network		
3.8.a (i) Single area, single topology	1	10
3.8.b Describe neighbor relationship	1	10
3.8.c Describe network types, levels, and router types		
3.8.c (i) NSAP addressing	1	10
3.8.c (ii) Point-to-point, broadcast	1	10
3.8.d Describe operations	1	10
3.8.e Describe optimization features		
3.8.e (i) Metrics, wide metric	1	10
4.0 VPN Technologies		

Topics	Book Volume	Book Chapter
4.1 Tunneling		
4.1.a Implement and troubleshoot MPLS operations		
4.1.a (i) Label stack, LSR, LSP	2	11
4.1.a (ii) LDP	2	11
4.1.a (iii) MPLS ping, MPLS traceroute	2	11
4.1.b Implement and troubleshoot basic MPLS L3VPN		
4.1.b (i) L3VPN, CE, PE, P	2	11
4.1.b (ii) Extranet (route leaking)	2	11
4.1.c Implement and troubleshoot encapsulation		
4.1.c (i) GRE	2	10
4.1.c (ii) Dynamic GRE	2	10
4.1.c (iii) LISP encapsulation principles supporting EIGRP OTP	1	8
4.1.d Implement and troubleshoot DMVPN (single hub)		
4.1.d (i) NHRP	2	10
4.1.d (ii) DMVPN with IPsec using preshared key	2	10
4.1.d (iii) QoS profile	2	10
4.1.d (iv) Pre-classify	2	10
4.1.e Describe IPv6 tunneling techniques		
4.1.e (i) 6in4, 6to4	2	8
4.1.e (ii) ISATAP	2	8
4.1.e (iii) 6RD	2	8
4.1.e (iv) 6PE/6VPE	2	8
4.1.g Describe basic Layer 2 VPN—wireline		
4.1.g (i) L2TPv3 general principles	2	10
4.1.g (ii) ATOM general principles	2	11
4.1.h Describe basic L2VPN—LAN services		
4.1.h (i) MPLS-VPLS general principles	2	10
4.1.h (ii) OTV general principles	2	10
4.2 Encryption		
4.2.a Implement and troubleshoot IPsec with preshared key		
4.2.a (i) IPv4 site to IPv4 site	2	10

Topics	Book Volume	Book Chapter
4.2.a (ii) IPv6 in IPv4 tunnels	2	10
4.2.a (iii) Virtual tunneling Interface (VTI)	2	10
4.2.b Describe GET VPN	2	10
5.0 Infrastructure Security		
5.1 Device security		
5.1.a Implement and troubleshoot IOS AAA using local database	2	9
5.1.b Implement and troubleshoot device access control		
5.1.b (i) Lines (VTY, AUX, console)	1	5
5.1.b (ii) SNMP	1	5
5.1.b (iii) Management plane protection	2	9
5.1.b (iv) Password encryption	1	5
5.1.c Implement and troubleshoot control plane policing	2	9
5.1.d Describe device security using IOS AAA with TACACS+ and RADIUS		
5.1.d (i) AAA with TACACS+ and RADIUS	2	9
5.1.d (ii) Local privilege authorization fallback	2	9
5.2 Network security		
5.2.a Implement and troubleshoot switch security features		
5.2.a (i) VACL, PACL	2	9
5.2.a (ii) Stormcontrol	2	9
5.2.a (iii) DHCP snooping	2	9
5.2.a (iv) IP source-guard	2	9
5.2.a (v) Dynamic ARP inspection	2	9
5.2.a (vi) port-security	2	9
5.2.a (vii) Private VLAN	1	2
5.2.b Implement and troubleshoot router security features		
5.2.b (i) IPv4 access control lists (standard, extended, time-based)	2	9
5.2.b (ii) IPv6 traffic filter	2	9
5.2.b (iii) Unicast reverse path forwarding	2	9
5.2.c Implement and troubleshoot IPv6 first-hop security		
5.2.c (i) RA guard	2	9

Topics	Book Volume	Book Chapter
5.2.c (ii) DHCP guard	2	9
5.2.c (iii) Binding table	2	9
5.2.c (iv) Device tracking	2	9
5.2.c (v) ND inspection/snooping	2	9
5.2.c (vii) Source guard	2	9
5.2.c (viii) PACL	2	9
5.2.d Describe 802.1x		
5.2.d (i) 802.1x, EAP, RADIUS	2	9
5.2.d (ii) MAC authentication bypass	2	9
6.0 Infrastructure Services		
6.1 System management		
6.1.a Implement and troubleshoot device management		
6.1.a (i) Console and VTY	1	5
6.1.a (ii) Telnet, HTTP, HTTPS, SSH, SCP	1	5
6.1.a (iii) (T)FTP	1	5
6.1.b Implement and troubleshoot SNMP		
6.1.b (i) v2c, v3	1	5
6.1.c Implement and troubleshoot logging		
6.1.c (i) Local logging, syslog, debug, conditional debug	1	5
6.1.c (ii) Timestamp	2	6
6.2 Quality of service		
6.2.a Implement and troubleshoot end-to-end QoS		
6.2.a (i) CoS and DSCP mapping	2	3
6.2.b Implement, optimize, and troubleshoot QoS using MQC		
6.2.b (i) Classification	2	3
6.2.b (ii) Network-based application recognition (NBAR)	2	3
6.2.b (iii) Marking using IP precedence, DSCP, CoS, ECN	2	3
6.2.b (iv) Policing, shaping	2	5
6.2.b (v) Congestion management (queuing)	2	4
6.2.b (vi) HQoS, subrate Ethernet link	2	3, 4, 5
6.2.b (vii) Congestion avoidance (WRED)	2	4

Topics	**Book Volume**	**Book Chapter**
6.2.c Describe Layer 2 QoS		
6.2.c (i) Queuing, scheduling	2	4
6.2.c (ii) Classification, marking	2	2
6.3 Network services		
6.3.a Implement and troubleshoot first-hop redundancy protocols		
6.3.a (i) HSRP, GLBP, VRRP	1	5
6.3.a (ii) Redundancy using IPv6 RS/RA	1	5
6.3.b Implement and troubleshoot Network Time Protocol		
6.3.b (i) NTP master, client, version 3, version 4	1	5
6.3.b (ii) NTP Authentication	1	5
6.3.c Implement and troubleshoot IPv4 and IPv6 DHCP		
6.3.c (i) DHCP client, IOS DHCP server, DHCP relay	1	5
6.3.c (ii) DHCP options	1	5
6.3.c (iii) DHCP protocol operations	1	5
6.3.c (iv) SLAAC/DHCPv6 interaction	1	4
6.3.c (v) Stateful, stateless DHCPv6	1	4
6.3.c (vi) DHCPv6 prefix delegation	1	4
6.3.d Implement and troubleshoot IPv4 Network Address Translation		
6.3.d (i) Static NAT, dynamic NAT, policy-based NAT, PAT	1	5
6.3.d (ii) NAT ALG	2	10
6.3.e Describe IPv6 Network Address Translation		
6.3.e (i) NAT64	2	10
6.3.e (ii) NPTv6	2	10
6.4 Network optimization		
6.4.a Implement and troubleshoot IP SLA		
6.4.a (i) ICMP, UDP, jitter, VoIP	1	5
6.4.b Implement and troubleshoot tracking object		
6.4.b (i) Tracking object, tracking list	1	5
6.4.b (ii) Tracking different entities (for example, interfaces, routes, IPSLA, and so on)	1	5
6.4.c Implement and troubleshoot NetFlow		

Topics	Book Volume	Book Chapter
6.4.c (i) NetFlow v5, v9	1	5
6.4.c (ii) Local retrieval	1	5
6.4.c (iii) Export (configuration only)	1	5
6.4.d Implement and troubleshoot embedded event manager		
6.4.d (i) EEM policy using applet	1	5
6.4.e Identify performance routing (PfR)		
6.4.e (i) Basic load balancing	1	11
6.4.e (ii) Voice optimization	1	11

To give you practice on these topics, and pull the topics together, Edition 5 of the *CCIE Routing and Switching v5.0 Official Cert Guide, Volume 1* includes a large set of CD questions that mirror the types of questions expected for the Version 5.0 blueprint. By their very nature, these topics require the application of the knowledge listed throughout the book. This special section of questions provides a means to learn and practice these skills with a proportionally larger set of questions added specifically for this purpose.

These questions will be available to you in the practice test engine database, whether you take full exams or choose questions by category.

About the *CCIE Routing and Switching v5.0 Official Cert Guide, Volume 1*, Fifth Edition

This section provides a brief insight into the contents of the book, the major goals, and some of the book features that you will encounter when using this book.

Book Organization

This volume contains four major parts. Beyond the chapters in these parts of the book, you will find several useful appendixes gathered in Part V.

Following is a description of each part's coverage:

- Part I, "LAN Switching" (Chapters 1–3)

  This part focuses on LAN Layer 2 features, specifically Ethernet (Chapter 1), VLANs and trunking (Chapter 2), and Spanning Tree Protocol (Chapter 3).

- Part II, "IP Networking" (Chapters 4–5)

  This part covers details across the spectrum of the TCP/IP protocol stack. It includes Layer 3 basics (Chapter 4) and IP services such as DHCP and ARP (Chapter 5).

- Part III, "IP IGP Routing" (Chapters 6–11)

  This part covers some of the more important topics on the exam and is easily the largest part of this volume. It covers Layer 3 forwarding concepts (Chapter 6), followed by three routing protocol chapters, one each about RIPv2, EIGRP, OSPF, and IS-IS (Chapters 7 through 10, respectively), and concludes with a discussion of IGP redistribution and routing information optimization (Chapter 11).

- Part IV, "Final Preparation"

  Chapter 12, "Final Preparation," contains instructions about using the testing software on the CD to verify your knowledge, presents suggestions on approaching your studies, and includes hints about further expanding your knowledge by participating in the Cisco Learning Network.

- Part V, "Appendixes"
  - Appendix A, "Answers to the 'Do I Know This Already?' Quizzes"—This appendix lists answers and explanations for the questions at the beginning of each chapter.
  - Appendix B, "Exam Updates"—As of the first printing of the book, this appendix contains only a few words that reference the web page for this book, at www.ciscopress.com/title/9781587143960. As the blueprint evolves over time, the authors will post new materials at the website. Any future printings of the book will include the latest newly added materials in printed form in Appendix B. If Cisco releases a major exam update, changes to the book will be available only in a new edition of the book and not on this site.

Note Appendixes C, D, E, F, and G and the Glossary are in printable, PDF format on the CD.

  - Appendix C, "Decimal to Binary Conversion Table" (CD-only)—This appendix lists the decimal values 0 through 255, with their binary equivalents.
  - Appendix D, "IP Addressing Practice" (CD-only)—This appendix lists several practice problems for IP subnetting and finding summary routes. The explanations to the answers use the shortcuts described in the book.
  - Appendix E, "Key Tables for CCIE Study" (CD-only)—This appendix lists the most important tables from the core chapters of the book. The tables have much of the content removed so that you can use them as an exercise. You can print the PDF file and then fill in the table from memory, checking your answers against the completed tables in Appendix F.
  - Appendix G, "Study Planner" (CD-only)—This appendix is a spreadsheet with major study milestones, where you can track your progress through your study.
  - Glossary (CD-only)—The Glossary contains the key terms listed in the book.

Book Features

The core chapters of this book have several features that help you make the best use of your time:

- **"Do I Know This Already?" Quizzes:** Each chapter begins with a quiz that helps you to determine the amount of time you need to spend studying that chapter. If you score yourself strictly, and you miss only one question, you might want to skip the core of the chapter and move on to the "Foundation Summary" section at the end of the chapter, which lets you review facts and spend time on other topics. If you miss more than one, you might want to spend some time reading the chapter or at least reading sections that cover topics about which you know you are weaker.
- **Foundation Topics:** These are the core sections of each chapter. They explain the protocols, concepts, and configuration for the topics in that chapter.
- **Foundation Summary:** The "Foundation Summary" section of this book departs from the typical features of the "Foundation Summary" section of other Cisco Press Exam Certification Guides. This section does not repeat any details from the "Foundation Topics" section; instead, it simply summarizes and lists facts related to the chapter but for which a longer or more detailed explanation is not warranted.
- **Key topics:** Throughout the "Foundation Topics" section, a Key Topic icon has been placed beside the most important areas for review. After reading a chapter, when doing your final preparation for the exam, take the time to flip through the chapters, looking for the Key Topic icons, and review those paragraphs, tables, figures, and lists.
- **Fill In Key Tables from Memory:** The more important tables from the chapters have been copied to PDF files available on the CD as Appendix E. The tables have most of the information removed. After printing these mostly empty tables, you can use them to improve your memory of the facts in the table by trying to fill them out. This tool should be useful for memorizing key facts. That same CD-only appendix contains the completed tables so that you can check your work.
- **CD-based practice exam:** The companion CD contains multiple-choice questions and a testing engine. The CD includes 200 questions unique to the CD. As part of your final preparation, you should practice with these questions to help you get used to the exam-taking process, as well as to help refine and prove your knowledge of the exam topics.
- **Key terms and Glossary:** The more important terms mentioned in each chapter are listed at the end of each chapter under the heading "Definitions." The Glossary, found on the CD that comes with this book, lists all the terms from the chapters. When studying each chapter, you should review the key terms, and for those terms about which you are unsure of the definition, you can review the short definitions from the Glossary.
- **Further Reading:** Most chapters include a suggested set of books and websites for additional study on the same topics covered in that chapter. Often, these references will be useful tools for preparation for the CCIE Routing and Switching lab exam.

This page intentionally left blank

Blueprint topics covered in this chapter:

This chapter covers the following subtopics from the Cisco CCIE Routing and Switching written exam blueprint. Refer to the full blueprint in Table I-1 in the Introduction for more details on the topics covered in each chapter and their context within the blueprint.

- Cisco Express Forwarding Concepts
- Routing Protocol Migration
- Policy-Based Routing

CHAPTER 6

IP Forwarding (Routing)

This chapter begins with coverage of the details of the forwarding plane—the actual forwarding of IP packets. This process of forwarding IP packets is often called *IP routing*, or simply *routing*. Also, many people also refer to IP routing as the *data plane*, meaning the plane (topic) related to the end-user data.

Chapters 7 through 11 cover the details of the IP *control plane*. In contrast to the term *data plane*, the control plane relates to the communication of control information—in short, routing protocols like OSPF and BGP. These chapters cover the routing protocols on the exam, plus an additional chapter on redistribution and route summarization.

"Do I Know This Already?" Quiz

Table 6-1 outlines the major headings in this chapter and the corresponding "Do I Know This Already?" quiz questions.

Table 6-1 *"Do I Know This Already?" Foundation Topics Section-to-Question Mapping*

Foundation Topics Section	Questions Covered in This Section	Score
IP Forwarding	1–6	
Multilayer Switching	7–9	
Policy Routing	10–11	
Total Score		

To best use this pre-chapter assessment, remember to score yourself strictly. You can find the answers in Appendix A, "Answers to the 'Do I Know This Already?' Quizzes."

1. What command is used to enable CEF globally for IPv4 packets?
   a. enable cef
   b. ip enable cef
   c. ip cef
   d. cef enable
   e. cef enable ip
   f. cef ip

2. What command is used to enable CEF globally for IPv6 packets?
    a. **enable cef6**
    b. **ipv6 enable cef**
    c. **ipv6 cef**
    d. **ip cef** (the command automatically enables CEF for IPv4 and IPv6)
3. Can CEF for IPv6 be enabled independently of CEF for IPv4?
    a. Yes
    b. No
4. Which of the following triggers an update to a CEF FIB?
    a. Receipt of an ICMPv6 Neighbor Advertisement message with previously unknown information
    b. Receipt of a LAN ARP reply message with previously unknown information
    c. Addition of a new route to the IP routing table by EIGRP
    d. Addition of a new route to the IP routing table by adding an **ip route** command
    e. The removal of a route from the IP routing table by EIGRP
5. Which of the following triggers an update to a CEF adjacency table?
    a. Receipt of a CDP multicast on the PVC connected to Router1
    b. Receipt of an ARP response with previously unknown information
    c. Receipt of a packet that needs to be routed to another router over a point-to-point interface
    d. Receipt of an ICMPv6 Neighbor Advertisement with previously unknown information
6. Which of the following packet-switching paths is considered to be the slowest?
    a. Process Switching
    b. Fast Switching
    c. Route Cache
    d. Cisco Express Forwarding
7. Which of the following commands is used on a Cisco IOS Layer 3 switch to use the interface as a *routed interface* instead of a *switched interface*?
    a. **ip routing** or **ipv6 unicast-routing** global command
    b. **ip routing** or **ipv6 unicast-routing** interface subcommand
    c. **ip address** interface subcommand
    d. **switchport mode routed** interface subcommand
    e. **no switchport** interface subcommand

8. On a Cisco Catalyst 3560 switch, the first line of the output of a **show interface vlan 55** command lists the state as "Vlan 55 is down, line protocol is down." Which of the following might be causing that state to occur?

- **a.** VLAN interface has not been **no shut** yet.
- **b.** The **ip routing** global command is missing from the configuration.
- **c.** On at least one interface in the VLAN, a cable that was previously plugged in has been unplugged.
- **d.** VTP mode is set to transparent.
- **e.** The VLAN has not yet been created on this switch, or is not in the active state.

9. On a Cisco Catalyst 3560 switch, the first line of the output of a **show interface vlan 55** command lists the state as "Vlan 55 is up, line protocol is down." Which of the following might be causing that state to occur?

- **a.** VLAN interface has not been **no shut** yet.
- **b.** The **ip routing** global command is missing from the configuration.
- **c.** There is no switch port on the switch with this VLAN allowed and in the STP forwarding state.
- **d.** STP has been administratively deactivated for this VLAN.
- **e.** The VLAN has not yet been created on this switch, or is not in the active state.

10. Imagine a route map used for policy routing, in which the route map has a **set default interface serial0/0** command. Serial0/0 is a point-to-point link to another router. A packet arrives at this router, and the packet matches the policy routing **route-map** clause whose only **set** command is the one just mentioned. Which of the following general characterizations is true?

- **a.** The packet will be routed out interface s0/0; if s0/0 is down, it will be routed using the default route from the routing table.
- **b.** The packet will be routed using the default route in the routing table; if there is no default, the packet will be routed out s0/0.
- **c.** The packet will be routed using the best match of the destination address with the routing table; if no match is found, the packet will be routed out s0/0.
- **d.** The packet will be routed out interface s0/0; if s0/0 is down, the packet will be discarded.

11. Router1 has an fa0/0 interface and two point-to-point WAN links back to the core of the network (s0/0 and s0/1, respectively). Router1 accepts routing information only over s0/0, which Router1 uses as its primary link. When s0/0 fails, Router1 uses policy routing to forward the traffic out the relatively slower s0/1 link. Which of the following **set** commands in Router1's policy routing route map could have been used to achieve this function?

- **a.** **set ip default next-hop**
- **b.** **set ip next-hop**
- **c.** **set default interface**
- **d.** **set interface**

Foundation Topics

IP Forwarding

IP forwarding, or *IP routing*, is the process of receiving an IP packet, making a decision of where to send the packet next, and then forwarding the packet. The forwarding process needs to be relatively simple, or at least streamlined, for a router to forward large volumes of packets. Ignoring the details of several Cisco optimizations to the forwarding process for a moment, the internal forwarding logic in a router works basically as shown in Figure 6-1.

Key Topic

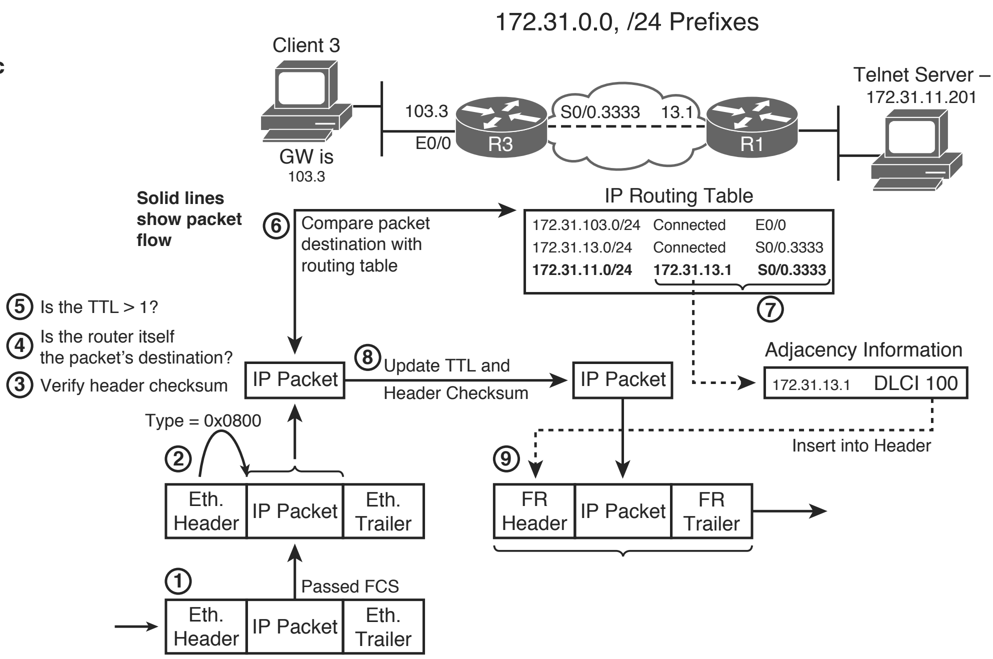


Figure 6-1 *Forwarding Process at Router3, Destination Telnet Server*

The following list summarizes the key steps shown in Figure 6-1:

Key Topic

1. A router receives the frame and checks the received frame check sequence (FCS); if errors occurred, the frame is discarded. The router makes no attempt to recover the lost packet.

2. If no errors occurred, the router checks the Ethernet Type field for the packet type and extracts the packet. The Data Link header and trailer can now be discarded.

3. Assuming an IPv4 packet, its header checksum is first verified. In case of mismatch, the packet is discarded. With IPv6 packets, this check is skipped, as IPv6 headers do not contain a checksum.

4. If the header checksum passed, the router checks whether the destination IP address is one of the addresses configured on the router itself. If it does, the packet has just arrived at its destination. The router analyzes the Protocol field in the IP header, identifying the upper-layer protocol, and hands the packet's payload over to the appropriate upper-protocol driver.
5. If the destination IP address does not match any of the router's configured addresses, the packet must be routed. The router first verifies whether the TTL of the packet is greater than 1. If not, the packet is dropped and an ICMP Time Exceeded message is sent to the packet's sender.
6. The router checks its IP routing table for the most specific prefix match of the packet's destination IP address.
7. The matched routing table entry includes the outgoing interface and next-hop router. This information is used by the router to look up the next-hop router's Layer 2 address in the appropriate mapping table, such as ARP, IP/DLCI, IP/VPI-VCI, dialer maps, and so on. This lookup is needed to build a new Data Link frame and optionally dial the proper number.
8. Before creating a new frame, the router updates the IP header TTL or Hop Count field, requiring a recomputation of the IPv4 header checksum.
9. The router encapsulates the IP packet in a new Data Link header (including the destination address) and trailer (including a new FCS) to create a new frame.

The preceding list is a generic view of the process. Next, a few words on how Cisco routers can optimize the routing process by using Cisco Express Forwarding (CEF).

Process Switching, Fast Switching, and Cisco Express Forwarding

Steps 6 and 7 from the generic routing logic shown in the preceding section are the most computation-intensive tasks in the routing process. A router must find the best route to use for every packet, requiring some form of table lookup of routing information. Also, a new Data Link header and trailer must be created, and the information to put in the header (like the destination Data Link address) must be found in another table.

Cisco has created several different methods to optimize the forwarding processing inside routers, termed *switching paths*. This section examines the two most likely methods to exist in Cisco router networks today: fast switching and CEF.

With fast switching, the first packet to a specific destination IP address is *process switched*, meaning that it follows the same general algorithm as shown in Figure 6-1. With the first packet, the router adds the results of this daunting lookup to the *fast-switching cache*, sometimes called the *route cache*, organized for fast lookups. The cache contains the destination IP address, the next-hop information, and the data-link header information that needs to be added to the packet before forwarding (as in Step 6 in Figure 6-1). Future packets to the same destination address match the cache entry, so it takes the router less time to process and forward the packet, as all results are already stored in the cache. This approach is also sometimes termed *route once, forward many times*.

Although it is much better than process switching, fast switching has significant drawbacks. The first packet must be process switched, because an entry can be added to the cache only when a packet is routed and the results of its routing (next hop, egress interface, Layer 2 rewrite information) are computed. A huge inflow of packets to destinations that are not yet recorded in the route cache can have a detrimental effect on the CPU and the router's performance, as they all need to be process switched. The cache entries are timed out relatively quickly, because otherwise the cache could get overly large as it has an entry per each destination address, not per destination subnet/prefix. If the routing table or Layer 3–to–Layer 2 tables change, parts of the route cache must be invalidated rather than updated, causing packets for affected destinations to become process switched again. Also, load balancing can only occur per destination with fast switching. Overall, fast switching was a great improvement at the time it was invented, but since that time, better switching mechanisms have been developed. One of them, Cisco Express Forwarding (CEF), has become the major packet-forwarding mechanism in all current Cisco IP routing implementations, with fast switching becoming practically unused. The support for unicast fast switching has therefore been discontinued and removed from IOS Releases 12.2(25)S and 12.4(20)T onward.

Key Topic

To learn the basic idea behind CEF as an efficient mechanism to perform routing decisions, it is important to understand that the crucial part of routing a packet through a router is finding out how to construct the Layer 2 frame header to allow the packet to be properly encapsulated toward its next hop, and forward the packet out the correct interface. Often, this operation is called a Layer 2 frame rewrite because that is what it resembles: A packet arrives at a router, and the router rewrites the Layer 2 frame, encapsulating the packet appropriately, and sends the packet toward the next hop. The packet's header does not change significantly—in IPv4, only the TTL and checksum are modified; with IPv6, only the Hop Count is decremented. An efficient routing mechanism should therefore focus on speeding up the construction of Layer 2 rewrite information and egress interface lookup. The process switching is highly inefficient in this aspect: The routing table lookup is relatively slow and might need recursive iterations until the directly attached next hop and egress interface are identified. The next-hop information must then be translated in ARP or other Layer 3–to–Layer 2 mapping tables to the appropriate Layer 2 address and the frame header must be constructed, and only then the packet can be encapsulated and forwarded. With each subsequent packet, this process repeats from the beginning.

Key Topic

One important observation is that while the routing table can hold tens of thousands of destination networks (prefixes), a router typically has only a handful of neighbors—the next hops toward all the known destinations. All destinations reachable through a particular next hop are using the same Layer 2 rewrite information. To reach any of the networks behind a particular *adjacent* next hop, the packets will be encapsulated into frames having the same Layer 2 header addresses and sent out the same egress interface. It makes sense, then, to trade memory for speed: Preconstruct the Layer 2 frame headers and egress interface information for each neighbor, and keep them ready in an *adjacency table* stored in the router's memory. This adjacency table can be constructed immediately as the routing table is populated, using IP addresses of next hops in the routing table and utilizing ARP or other Layer 3–to–Layer 2 mapping tables to translate next-hop

IP addresses into their corresponding Layer 2 addresses. A packet that is to be routed through a particular next hop will then simply use the preconstructed Layer 2 frame header for that next hop, without needing to visit the ARP or similar tables over and over again. The process of routing a packet will then transform itself to the process of deciding which entry from the adjacency table should be used to encapsulate and forward the packet. After the proper entry is selected, encapsulating the packet and forwarding it out the egress interface can be done in an extremely rapid way, as all necessary data is readily available.

Key Topic

Another important observation is that the routing table itself is not truly optimized for rapid lookups. It contains lots of information crucial to its construction but not that important for routing lookups, such as origin and administrative distances of routes, their metrics, age, and so on. Entries in the routing table can require recursive lookups: After matching a destination network entry, the next-hop information might contain only the IP address of the next hop but not the egress interface, so the next hop's IP address has to be looked up in the routing table in the next iteration—and the depth of this recursion is theoretically unlimited. Even after matching the ultimate entry in the routing table that finally identifies the egress interface, it does not really say anything about the Layer 2 rewrite that is necessary to forward the packet. The last found next-hop IP address during this lookup process has to be further matched in the ARP or similar mapping tables for the egress interface to find out how to construct the Layer 2 frame header. All these shortcomings can be improved, though: The destination prefixes alone from the routing table can be stored in a separate data structure called the *Forwarding Information Base*, or FIB, optimized for rapid lookups (usually, tree-based data structures meet this requirement). Instead of carrying the plain next hop's IP address from the routing table over into the FIB, each entry in the FIB that represents a destination prefix can instead contain a pointer toward the particular entry in the adjacency table that stores the appropriate rewrite information: Layer 2 frame header and egress interface indication. Any necessary recursion in the routing table is resolved while creating the FIB entries and setting up the pointers toward appropriate adjacency table entries. No other information needs to be carried over from the routing table into the FIB. In effect, the FIB stores only destination prefixes alone. The forwarding information itself is stored as Layer 2 rewrite information in the adjacency table, and entries in the FIB point toward appropriate entries in the adjacency table. All FIB entries that describe networks reachable through a particular next hop point to the same adjacency table entry that contains prepared Layer 2 header and egress information toward that next hop.

Key Topic

After the FIB and adjacency table are created, the routing table is not used anymore to route packets for which all forwarding information is found in the FIB/adjacency table. With FIB-based routers, the routing table can be used for packets that require more complex processing not available through straightforward Layer 2 rewrite; however, for plain packet routing, only the FIB and the adjacency table are used. The routing table therefore becomes more of a source of routing data to build the FIB and adjacency table contents but is not necessarily used to route packets anymore. Therefore, such a routing table is called the *Routing Information Base (RIB)*—it is the master copy of routing information from which the FIB and other structures are populated, but it is not necessarily used to route packets itself. Note that many routing protocols including Open Shortest Path First (OSPF) and Border Gateway Protocol (BGP) construct their own internal routing tables that are also called RIBs. These per-protocol RIBs are usually separate from the router's routing table and shall not be confused with the RIB discussed in this chapter.

Advantages of this approach should be immediately obvious. The FIB contains only the essential information to match a packet's destination address to a known prefix. A single lookup in the FIB immediately produces a pointer toward complete Layer 2 rewrite information for the packet to be forwarded. If the next hop for a destination changes, only the pointer in the respective FIB entry needs to be updated to point toward the new adjacency table entry; the FIB entry itself that represents the destination prefix is unchanged. Both FIB and adjacency tables can be readily constructed from the routing table and the available Layer 3–to–Layer 2 mapping tables, without requiring any packet flows as was the case in fast switching. To those readers familiar with database systems, the FIB can be seen as an index over the adjacency table, with IP prefixes being the lookup keys and the indexed data being the Layer 2 rewrite entries in the adjacency table.

These ideas are at the core of Cisco Express Forwarding, or CEF. Conceptually, CEF consists of two parts—the *Forwarding Information Base* and the *adjacency table*. The FIB contains all known destination prefixes from the routing table, plus additional specific entries, organized as a so-called *mtrie* or a *multiway prefix tree*. The adjacency table contains a Layer 2 frame header prepared for each known next hop or directly attached destination.

The CEF as just described can be implemented in a relatively straightforward way in software, and this is exactly what all software-based Cisco routers do: They implement CEF purely in software, as part of the operating system they run. Both FIB and adjacency tables are maintained in router's memory, and lookups in these structures are done by the CPU as part of interrupt handler executed when a packet is received. Figure 6-2, reused from the Cisco document "How to Choose the Best Router Switching Path for Your Network," Document ID 13706 available on the Cisco website, illustrates the concept.

Multilayer switches and high-end Cisco router platforms go even further, and instead of software-based FIB, they use specialized circuits (specifically, Ternary Content Addressable Memory [TCAM]) to store the FIB contents and perform even faster lookups. Using a TCAM, an address lookup is performed in an extremely short time that does not depend on the number of FIB entries, as the TCAM performs the matching on its entire contents in parallel. On these platforms, the CEF structures are distributed to individual linecards if present, and stored in TCAMs and forwarding ASICs.

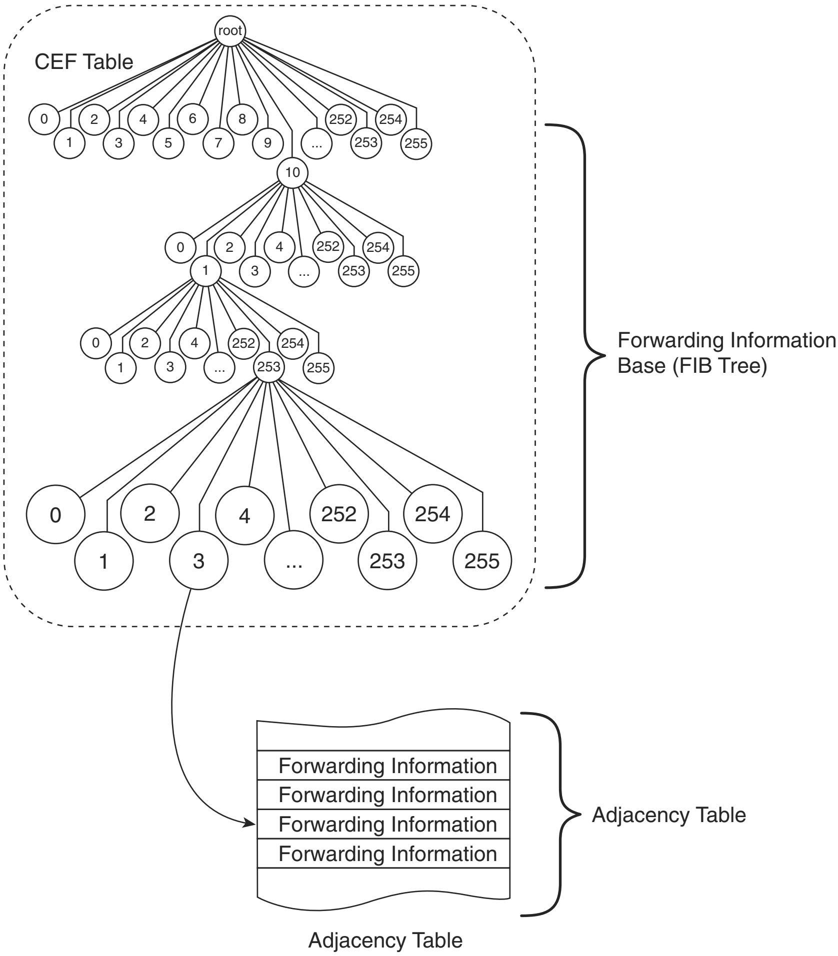


Figure 6-2 *Cisco Express Forwarding Basic Architecture*

To illustrate the CEF in action, consider the network in Figure 6-3 and related Example 6-1.

In this network, Router R1 is connected to two other routers and one multilayer switch. The Data Link Layer technologies interconnecting the devices are diverse: Between R1 and R2, HDLC is used; R1 and R3 are connected over a PPP link; R1 and MLS4 are using Ethernet interconnection in two VLANs—native VLAN and VLAN 2. OSPF is the routing protocol in use. R2 advertises networks 10.2.0.0/24 through 10.2.3.0/24 and FD00:2::/64 through FD00:2:3::/64. In a similar fashion, R3 advertises networks 10.3.4.0/24 through 10.3.7.0/24 and FD00:3:4::/64 through FD00:3:7::/64. MLS4 advertises networks 10.4.8.0/24 and 10.4.9.0/24, and FD00:4:8::/64 and FD00:4:9::/64, over both VLANs. Multiple interface encapsulations and multiple networks reachable over a single next hop are used in this example to show how potentially numerous destination prefixes map to a single adjacent next hop and how the Layer 2 rewrite information is built depending on the Data Link Layer technology. CEF is activated for both IPv4 and IPv6 using the **ip cef** and **ipv6 cef** global configuration commands on R1.

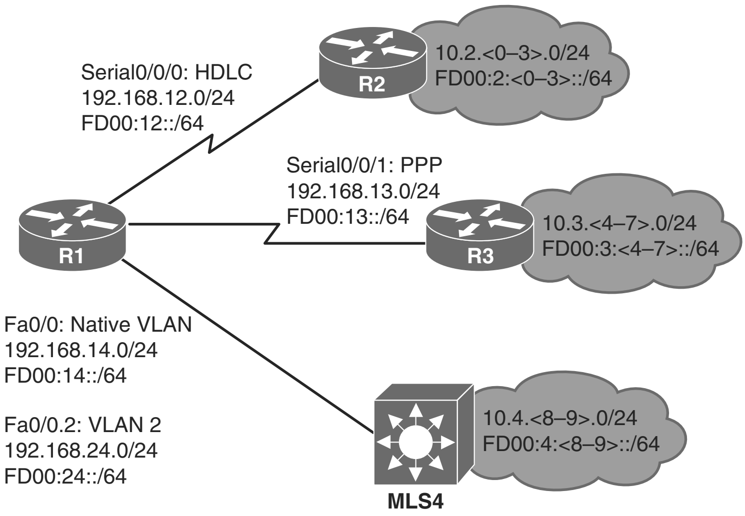


Figure 6-3 *Example Network Showcasing CEF Operation*

Example 6-1 *CEF FIB and Adjacency Table*

```
! On R1, show ip route ospf shows a portion of the RIB

R1# show ip route ospf
     10.0.0.0/8 is variably subnetted, 12 subnets, 2 masks
O       10.2.0.0/24 [110/782] via 192.168.12.2, 00:07:06, Serial0/0/0
O       10.2.1.0/24 [110/782] via 192.168.12.2, 00:07:06, Serial0/0/0
O       10.2.2.0/24 [110/782] via 192.168.12.2, 00:07:06, Serial0/0/0
O       10.2.3.0/24 [110/782] via 192.168.12.2, 00:07:06, Serial0/0/0
O       10.3.4.1/32 [110/782] via 192.168.13.3, 00:07:06, Serial0/0/1
O       10.3.5.0/24 [110/782] via 192.168.13.3, 00:07:06, Serial0/0/1
O       10.3.6.0/24 [110/782] via 192.168.13.3, 00:07:06, Serial0/0/1
O       10.3.7.0/24 [110/782] via 192.168.13.3, 00:07:06, Serial0/0/1
O       10.4.8.0/24 [110/2] via 192.168.24.4, 00:07:06, FastEthernet0/0.2
                    [110/2] via 192.168.14.4, 00:07:06, FastEthernet0/0
O       10.4.9.0/24 [110/2] via 192.168.24.4, 00:07:06, FastEthernet0/0.2
                    [110/2] via 192.168.14.4, 00:07:06, FastEthernet0/0

! Another crucial part of information is the ARP table that resolves
! next hop IP addresses of hosts connected via Ethernet to MAC addresses
! Serial interface technologies in this example are point-to-point and
! hence require no Layer 3-to-Layer 2 mapping tables. This information will
! be used in construction of adjacency table entries

R1# show ip arp
Protocol  Address          Age (min)  Hardware Addr   Type   Interface
Internet  192.168.14.4           41   0017.9446.b340  ARPA   FastEthernet0/0
```

```
Internet  192.168.14.1           -   0019.e87f.38e4  ARPA   FastEthernet0/0
Internet  192.168.24.1           -   0019.e87f.38e4  ARPA   FastEthernet0/0.2
Internet  192.168.24.4          41   0017.9446.b341  ARPA   FastEthernet0/0.2

! show ip cef shows the FIB contents. In the following output, only routes
! learned via OSPF are shown for brevity reasons. Note how a set of prefixes
! resolves through a particular adjacency (next hop IP and egress interface).

R1# show ip cef 10.0.0.0 255.0.0.0 longer-prefixes
Prefix              Next Hop            Interface
10.2.0.0/24         192.168.12.2        Serial0/0/0
10.2.1.0/24         192.168.12.2        Serial0/0/0
10.2.2.0/24         192.168.12.2        Serial0/0/0
10.2.3.0/24         192.168.12.2        Serial0/0/0
10.3.4.1/32         192.168.13.3        Serial0/0/1
10.3.5.0/24         192.168.13.3        Serial0/0/1
10.3.6.0/24         192.168.13.3        Serial0/0/1
10.3.7.0/24         192.168.13.3        Serial0/0/1
10.4.8.0/24         192.168.24.4        FastEthernet0/0.2
                    192.168.14.4        FastEthernet0/0
10.4.9.0/24         192.168.24.4        FastEthernet0/0.2
                    192.168.14.4        FastEthernet0/0

! Similarly, for IPv6, the relevant outputs are:

R1# show ipv6 route ospf
! Output shortened and reformatted for brevity
O   FD00:2::/64 [110/782]   via FE80::2, Serial0/0/0
O   FD00:2:1::/64 [110/782] via FE80::2, Serial0/0/0
O   FD00:2:2::/64 [110/782] via FE80::2, Serial0/0/0
O   FD00:2:3::/64 [110/782] via FE80::2, Serial0/0/0
O   FD00:3:4::/64 [110/782] via FE80::3, Serial0/0/1
O   FD00:3:5::/64 [110/782] via FE80::3, Serial0/0/1
O   FD00:3:6::/64 [110/782] via FE80::3, Serial0/0/1
O   FD00:3:7::/64 [110/782] via FE80::3, Serial0/0/1
O   FD00:4:8::/64 [110/2]   via FE80:24::4, FastEthernet0/0.2
                            via FE80:14::4, FastEthernet0/0
O   FD00:4:9::/64 [110/2]   via FE80:24::4, FastEthernet0/0.2
                            via FE80:14::4, FastEthernet0/0

R1# show ipv6 neighbors
IPv6 Address                              Age Link-layer Addr State Interface
FD00:14::4                                  1 0017.9446.b340  STALE Fa0/0
FD00:24::4                                  1 0017.9446.b341  STALE Fa0/0.2
FE80::3                                     - -               REACH Se0/0/1
FE80:14::4                                  2 0017.9446.b340  STALE Fa0/0
```

```
FE80:24::4                                  1 0017.9446.b341  STALE Fa0/0.2

R1# show ipv6 cef
! Output shortened and reformatted for brevity
FD00:2::/64   nexthop FE80::2 Serial0/0/0
FD00:2:1::/64 nexthop FE80::2 Serial0/0/0
FD00:2:2::/64 nexthop FE80::2 Serial0/0/0
FD00:2:3::/64 nexthop FE80::2 Serial0/0/0
FD00:3:4::/64 nexthop FE80::3 Serial0/0/1
FD00:3:5::/64 nexthop FE80::3 Serial0/0/1
FD00:3:6::/64 nexthop FE80::3 Serial0/0/1
FD00:3:7::/64 nexthop FE80::3 Serial0/0/1
FD00:4:8::/64 nexthop FE80:24::4 FastEthernet0/0.2
              nexthop FE80:14::4 FastEthernet0/0
FD00:4:9::/64 nexthop FE80:24::4 FastEthernet0/0.2
              nexthop FE80:14::4 FastEthernet0/0

! The show adjacency shows an abbreviated list of adjacency table entries
! Note that separate entries are created for IPv4 and IPv6 adjacencies,
! as the Protocol or EtherType field value in pre-constructed frame headers
! is different for IPv4 and IPv6

R1# show adjacency
Protocol Interface                 Address
IPV6     Serial0/0/0               point2point(12)
IP       Serial0/0/0               point2point(13)
IPV6     Serial0/0/1               point2point(10)
IP       Serial0/0/1               point2point(15)
IPV6     FastEthernet0/0.2         FE80:24::4(12)
IP       FastEthernet0/0           192.168.14.4(23)
IPV6     FastEthernet0/0           FE80:14::4(12)
IP       FastEthernet0/0.2         192.168.24.4(23)
IPV6     Serial0/0/1               point2point(4)
IPV6     FastEthernet0/0.2         FD00:24::4(5)
IPV6     FastEthernet0/0           FD00:14::4(7)

! Now focus on the adjacency table details. There are adjacencies via multiple
! interfaces. Serial0/0/0 is running HDLC. Note in the show adjacency detail
! command output the prepared HDLC header for all IPv6 prefixes (0F0086DD)
! and IP prefixes (0F000800) resolving through this adjacency.

R1# show adjacency s0/0/0 detail
Protocol Interface                 Address
IPV6     Serial0/0/0               point2point(12)
                                   0 packets, 0 bytes
                                   0F0086DD
```

```
                                    IPv6 CEF   never
                                    Epoch: 2
IP        Serial0/0/0               point2point(13)
                                    0 packets, 0 bytes
                                    0F000800
                                    CEF   expires: 00:02:43
                                          refresh: 00:00:43
                                    Epoch: 2

! Similar output can be achieved for Serial0/0/1 that runs PPP. In the following
! output, note the prepared PPP headers for IPv6 (FF030057) and IPv4 (FF030021)
! prefixes resolving through these adjacencies. There are two IPv6 adjacencies
! present as IPV6CP specifically installs an adjacency towards the neighbor's link
! local address.

R1# show adjacency s0/0/1 detail
Protocol Interface                  Address
IPV6      Serial0/0/1               point2point(10)
                                    0 packets, 0 bytes
                                    FF030057
                                    IPv6 CEF   never
                                    Epoch: 2
IP        Serial0/0/1               point2point(15)
                                    0 packets, 0 bytes
                                    FF030021
                                    CEF   expires: 00:02:30
                                          refresh: 00:00:30
                                    Epoch: 2
IPV6      Serial0/0/1               point2point(4)
                                    0 packets, 0 bytes
                                    FF030057
                                    IPv6 ND    never
                                    IPv6 ND    never
                                    Epoch: 2

! Adjacencies on Fa0/0 show preconstructed Ethernet headers for the neighbors
! 192.168.14.4, FE80:14::4 and FD00:14::4 - destination MAC, source MAC, EtherType.
! Compare the MAC addresses with contents of ARP and IPv6 ND tables above.

R1# show adjacency fa0/0 detail
Protocol Interface                  Address
IP        FastEthernet0/0           192.168.14.4(23)
                                    0 packets, 0 bytes
                                    00179446B3400019E87F38E40800
                                    ARP        02:29:07
                                    Epoch: 2
```

```
IPV6     FastEthernet0/0           FE80:14::4(12)
                                   0 packets, 0 bytes
                                   00179446B3400019E87F38E486DD
                                   IPv6 ND    never
                                   Epoch: 2
IPV6     FastEthernet0/0           FD00:14::4(7)
                                   0 packets, 0 bytes
                                   00179446B3400019E87F38E486DD
                                   IPv6 ND    never
                                   Epoch: 2

! Finally, adjacencies on Fa0/0.2 show preconstructed Ethernet headers for
! neighbors 192.168.24.4, FE80:24::4 and FD00:24::4 - destination MAC, source MAC,
! 802.1Q VLAN tag, EtherType. Compare the MAC addresses with contents of ARP and
! IPv6 ND tables.


R1# show adjacency fa0/0.2 detail
Protocol Interface                 Address
IPV6     FastEthernet0/0.2         FE80:24::4(12)
                                   0 packets, 0 bytes
                                   00179446B3410019E87F38E481000002
                                   86DD
                                   IPv6 ND    never
                                   Epoch: 2
IP       FastEthernet0/0.2         192.168.24.4(23)
                                   0 packets, 0 bytes
                                   00179446B3410019E87F38E481000002
                                   0800
                                   ARP        02:26:57
                                   Epoch: 2
IPV6     FastEthernet0/0.2         FD00:24::4(5)
                                   0 packets, 0 bytes
                                   00179446B3410019E87F38E481000002
                                   86DD
                                   IPv6 ND    never
                                   Epoch: 2
```

Table 6-2 summarizes a few key points about the three main options for router switching paths.

Key Topic

Table 6-2 *Matching Logic and Load-Balancing Options for Each Switching Path*

Switching Path	Structures That Hold the Forwarding Information	Load-Balancing Method
Process switching	Routing table	Per packet
Fast switching	Fast-switching cache (per flow route cache)	Per destination IP address
CEF	FIB tree and adjacency table	Per a hash of the packet source and destination, or per packet

The **ip cef** global configuration command enables CEF for all interfaces on a Cisco router. For IPv6, the **ipv6 cef** command is used to activate CEF support. Note that it is possible to run IPv4 CEF without IPv6 CEF, but the converse is not true: To run IPv6 CEF, IPv4 CEF must be active. The **no ip route-cache cef** interface subcommand can then be used to selectively disable CEF on an interface.

Load Sharing with CEF and Related Issues

One of major advantages of CEF is its native support for different load-sharing mechanisms, allowing the use of multiple paths toward a destination network if present in the FIB. CEF supports two modes of load sharing: per-packet and per-destination. With per-packet load sharing, packets destined to a destination network are distributed across multiple paths in a packet-by-packet fashion. With the per-destination mode, the CEF actually takes the source and destination IP address and optionally other data to produce a hash value that identifies the particular path to carry the packet. In effect, for a particular source and destination pair, all packets flow through a single path. Other particular source/destination address combinations toward the same destination network can produce a different hash and thus be forwarded over a different path. In fact, the per-destination load-sharing mode in CEF would be better called per-flow load sharing. The per-destination load-sharing mode is the default (hardware-based CEF implementations might not support the per-packet load sharing mode), and in general, it is preferred because it avoids packet reordering within a single conversation.

Per-destination load sharing in CEF is technically achieved by placing a so-called load-share table between the FIB and the adjacency table. This loadshare table contains up to 16 pointers to entries in the adjacency table, and the individual loadshare entries are populated so that the counts of loadshare pointers to particular adjacency entries are proportional to the costs of parallel routes toward the same destination. (That is, if there are two equal-cost paths to the same destination, eight loadshare entries will point to one next-hop adjacency entry while another eight loadshare entries will point to another next-hop adjacency entry. If there are three equal cost paths, only 15 loadshare entries will be populated, with each five loadshare entries pointing to one of the three next-hop adjacency entries.) When a packet arrives, the router performs a hashing operation over the packet's source and destination address fields, and uses the hash result value as an index into the loadshare table to select one of the possible paths toward the destination. This concept is illustrated in Figure 6-4, also taken from the Cisco document "How to Choose the Best Router Switching Path for Your Network," Document ID 13706.

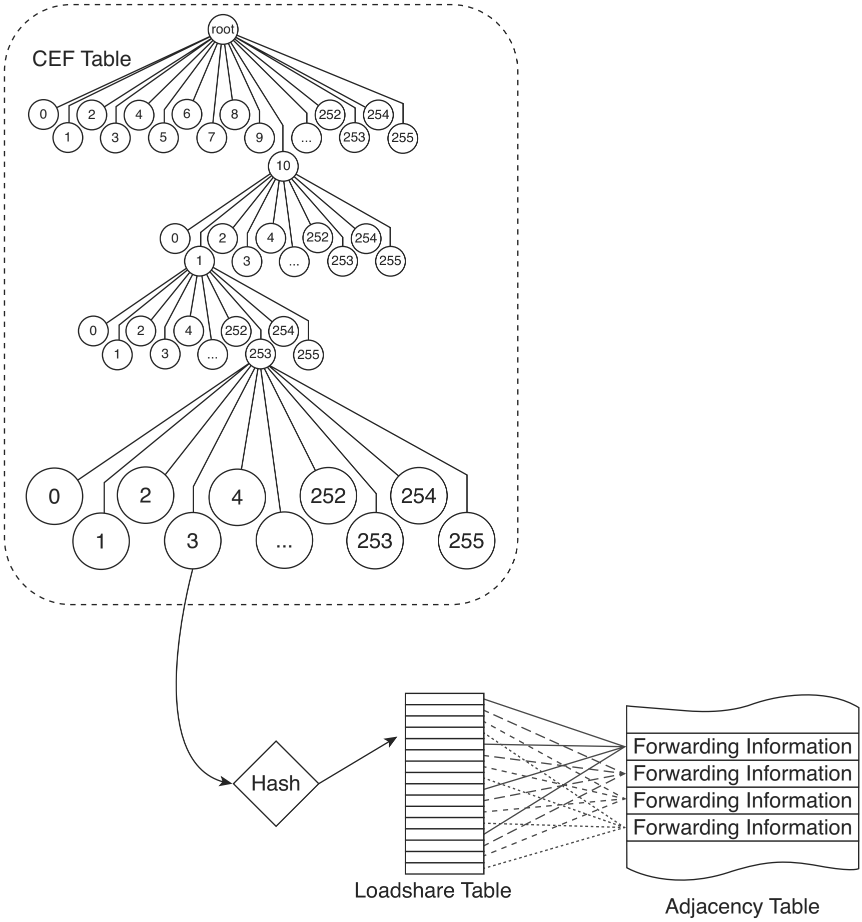


Figure 6-4 *CEF Load Balancing*

The particular method of per-packet or per-destination load sharing can be activated on *egress* interfaces of a router using the **ip load-share** { **per-destination** | **per-packet** } interface-level command. The availability of this command might be limited depending on the hardware capabilities of the device. Often, multilayer switches performing hardware-accelerated switching do not support this command while software-based ISR routers do.

With the hashing performed over fixed packet and/or segment address fields, a single hash function produces the same result for all packets in a flow. While this is desirable on a single router to always select a single path for a flow, it leads to unpleasant consequences in a network where multiple routers down a path to a destination have multiple routes toward it.

Consider the network shown in the Figure 6-5.

Key Topic

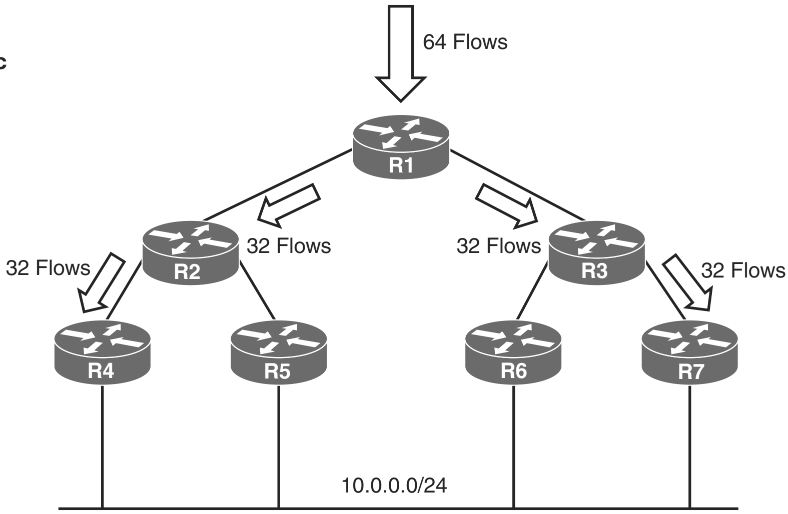


Figure 6-5 *CEF Polarization*

Key Topic

Router R1 has two neighbors, R2 and R3, toward the destination network 10.0.0.0/24. Let's assume that it is receiving 64 different flows destined to stations inside the network 10.0.0.0/24. Under ideal conditions, 32 flows will be forwarded from R1 through R2 and 32 other flows will be forwarded through R3. On R2, we now expect that it again balances the 32 received flows across its neighbors, forwarding 16 flows through R4 and another 16 flows through R5. However, if R2 is using the same hashing function as R1, this is no longer the case. All 32 flows received by R2 have produced the same hashing value on R1—that is why R2 is receiving all of them in the first place. Running the same hashing function over these 32 flows will again produce the same value for all of them, and as a result, R2 will no longer load-share them; rather, all 32 flows will be forwarded from R2 through a single path to the destination. Thus, no load sharing will occur farther down the path below R1. Quite the same fate will meet the remaining 32 flows on R3. This phenomenon is called CEF polarization, and will cause the advantage of load sharing to be lost quickly.

To avoid this, the basic CEF load-sharing mechanism has been enhanced. Each router chooses a random 4B-long number called a Universal ID (details of its selection are not public). This Universal ID is used as a seed in the hashing function used by CEF. Because with high probability, different routers will have unique Universal IDs, they will also produce different hashing results for a particular packet flow. As a result, a set of flows producing a single hashing value on one router might produce a set of different hashing values on another router, enabling the set of flows to be load-balanced again across multiple paths.

In recent IOSs, there are multiple variations of the CEF load-sharing algorithm:

- **Original algorithm:** As the name suggests, this is the original unseeded implementation prone to CEF polarization.
- **Universal algorithm:** An improved algorithm using the Universal ID to avoid the CEF polarization.
- **Tunnel algorithm:** A further improvement on the Universal algorithm especially suitable to environments where tunnels are extensively deployed, possibly resulting in a relatively small number of outer source/destination pairs. Avoids the CEF polarization. Might not be available for IPv6 CEF.
- **L4 port algorithm:** Based on the Universal algorithm while also taking the L4 source and/or destination ports into account. Avoids the CEF polarization.

Except from the Original algorithm, all other algorithms listed here avoid the CEF polarization issue by seeding the hash function using the Universal ID. This ID can be specified for these algorithms in the **ip cef load-sharing algorithm** and **ipv6 cef load-sharing algorithm** global configuration commands manually if necessary. This command is also used to select the particular load-sharing algorithm as described in the preceding list. To verify the current load-sharing mechanism and Universal ID value, the output of **show cef state**, **show ip cef summary**, or **show ip cef detail**, especially the heading, shall be examined (the output of these commands differs on different platforms).

The Catalyst 6500 platform (and some others that are directly derived from it, such as selected 7600 Series supervisors and linecards), enjoying a long history of existence during the time the details of CEF were fleshed out and perfected in software-based IOSs, has its own set of workarounds about the CEF polarization problem. On this platform, instead of the **ip cef load-sharing algorithm** command, the **mls ip cef load-sharing** command is used to select the load-sharing algorithm. The individual options are as follows:

- Default (**default mls ip cef load-sharing**): Uses source and destination IP, plus the Universal ID if supported by the hardware. Avoids CEF polarization.
- Full (**mls ip cef load-sharing full**): Uses source IP, destination IP, source L4 port, and destination L4 port. Does not use Universal ID. Prone to CEF polarization. However, to alleviate its impact, this load-balancing algorithm causes the traffic to split equally among multiple paths only if the number of paths is odd. With an even number of parallel paths, the ratio of traffic split will not be uniform.
- Simple (**mls ip cef load-sharing simple**): Uses source and destination IP only. Does not use Universal ID. Prone to CEF polarization.
- Full Simple (**mls ip cef load-sharing full simple**): Uses source IP, destination IP, source L4 port, and destination L4 port. Does not use Universal ID. Prone to CEF polarization. The difference from Full mode is that all parallel paths receive an equal weight, and fewer adjacency entries in hardware are used. This mode avoids unequal traffic split seen with Full mode.

Multilayer Switching

Multilayer Switching (MLS) refers to the process by which a LAN switch, which operates at least at Layer 2, also uses logic and protocols from layers other than Layer 2 to forward data. The term *Layer 3 switching* refers specifically to the use of the Layer 3 destination address, compared to the routing table (or equivalent), to make the forwarding decision. (The latest switch hardware and software from Cisco uses CEF switching to optimize the forwarding of packets at Layer 3.)

MLS Logic

Layer 3 switching configuration works similarly to router configuration—IP addresses are assigned to interfaces, and routing protocols are defined. The routing protocol configuration works just like a router. However, the interface configuration on MLS switches differs slightly from routers, using VLAN interfaces, routed interfaces, and Port-channel Layer 3 interfaces.

VLAN interfaces give a Layer 3 switch a Layer 3 interface attached to a VLAN. Cisco often refers to these interfaces as *switched virtual interfaces (SVI)*. To route between VLANs, a switch simply needs a virtual interface attached to each VLAN, and each VLAN interface needs an IP address in the respective subnets used on those VLANs.

Note Although it is not a requirement, the devices in a VLAN are typically configured in the same single IP subnet. However, you can use secondary IP addresses on VLAN interfaces to configure multiple subnets in one VLAN, just like on other router interfaces.

Key Topic

The operational state of SVI interfaces deserves a word on its own. For an MLS, an SVI is *the* Layer 3 interface that interconnects the internal "router" inside the MLS with the particular VLAN, much like an interface on a real router connects it to a particular network. An MLS can directly send packets to or through a particular VLAN by forwarding them over the corresponding SVI. These SVIs will be present in an MLS's routing table as egress interfaces for packets delivered into or through particular VLANs. The operational state of an SVI should therefore reflect the true ability of the MLS to directly forward packets into the corresponding VLAN. The SVI—despite being a virtual interface—must not be in the "up, line protocol up" state if the MLS is not truly capable of forwarding packets into the corresponding VLAN. In other words, the state of SVIs must mimic the behavior of ordinary routers. If an interface is not in the "up, line protocol up" state, the configured directly connected network on that interface and all routes formerly reachable over it must be removed from the routing table, and can be put back only if the interface becomes fully operational again.

There are two primary reasons why an MLS might be unable to forward packets into a particular VLAN: Either that VLAN is not created and active on the MLS, or the VLAN exists and is active but there is no physical Layer 2 interface on the switch allowing it to forward frames into that VLAN. Consequently, the state of an SVI can be one of the following:

Key Topic

- **Administratively down, line protocol down:** The SVI interface is shut down.
- **Down, line protocol down:** The corresponding VLAN does not exist, or is not in an active state (the **state suspend** or **shutdown** commands were issued in the VLAN's configuration).
- **Up, line protocol down:** The corresponding VLAN exists, but it is not allowed and in an STP forwarding state on any Layer 2 switch port (access or trunk).
- **Up, line protocol up:** The VLAN is created and the MLS is capable of forwarding frames (and hence packets) into that VLAN.

Key Topic

To avoid the "up, line protocol down," at least one of the following conditions must be true:

- At least one physical trunk that is itself in the "up, line protocol up" state must have this VLAN allowed, not VTP pruned, and in the STP forwarding state. This can be verified, for example, using the **show interfaces trunk** command (check the bottom-most section labeled with "Vlans in spanning tree forwarding state and not pruned").
- At least one physical switch port that is itself in the "up, line protocol up" state must have this VLAN configured as an access or voice VLAN and in the STP forwarding state. This can be verified, for example, using **show vlan** and **show spanning-tree** commands.

When using VLAN interfaces, the switch must take one noticeable but simple additional step when routing a packet. Like typical routers, MLS makes a routing decision to forward a packet. As with routers, the routes in an MLS routing table entry list an outgoing interface (a VLAN interface in this case), as well as a next-hop Layer 3 address. The adjacency information (for example, the IP ARP table or the CEF adjacency table) lists the VLAN number and the next-hop device's MAC address to which the packet should be forwarded—again, typical of normal router operation.

At this point, a true router would know everything it needs to know to forward the packet. An MLS switch, however, then also needs to use Layer 2 logic to decide which physical interface to physically forward the packet already encapsulated in a Layer 2 frame. The switch will simply find the next-hop device's MAC address in the CAM and forward the frame to that address based on the CAM.

Using Routed Ports and Port-channels with MLS

In some point-to-point topologies, VLAN interfaces are not required. For example, when an MLS switch connects to a router using a cable from a switch interface to a router's LAN interface, and the only two devices in that subnet are the router and that one physical interface on the MLS switch, the MLS switch can be configured to treat that one interface as a *routed port*. (Another typical topology for using routed ports is when two MLS switches connect for the purpose of routing between the switches, again creating a case with only two devices in the VLAN/subnet.)

A routed port on an MLS switch has the following characteristics:

- The interface is not placed into any user-defined VLAN (internally in an MLS switch, an *internal usage VLAN* is created for each individual routed port).
- On most Catalyst platforms, a routed port cannot be configured with subinterfaces.
- The switch does not keep any Layer 2 switching table information for the interface.
- Layer 3 settings, such as the IP address, are configured under the physical interface, just like a router.
- The adjacency table lists the outgoing physical interface or Port-channel, which means that Layer 2 switching logic is not required in these cases.

Key Topic

The *internal usage VLAN* created on behalf of a routed port deserves a special mention. For a VLAN-aware MLS, all operations are performed within the context of a VLAN in which the frame or packet is processed. The most natural way for these switches to implement a routed port is in fact to create a hidden, standalone, and dedicated VLAN for each separate routed port, and deactivate the typical Layer 2 control plane protocols on it. These dedicated VLANs are called *internal usage VLANs*. On Catalyst switches supporting an extended VLAN range, these internal usage VLANs are allocated from the extended range, depending on the setting of the **vlan internal allocation policy** { **ascending** | **descending** } global configuration command. If the **ascending** option is used, internal usage VLANs are allocated from VLAN ID 1006 upward. Conversely, if the **descending** option is used, internal usage VLANs are allocated from VLAN ID 4094 downward. On lower-end Catalyst platforms, this command is present in the configuration with the **ascending** option but cannot be modified.

The current allocation of internal usage VLANs can be displayed only using the **show vlan internal usage** command; they do not appear in common **show vlan** output. As an example, observe the output in the Example 6-2.

Example 6-2 *Internal Usage VLANs Created for Routed Ports*

```
! On this 3560G switch, ports GigabitEthernet0/12 and GigabitEthernet0/13 will
! be configured as routed ports, and the internal usage VLANs will be observed.
! The switch is configured with vlan internal allocation policy ascending

Switch(config)# do show vlan internal usage


VLAN Usage
---- --------------------


Switch(config)# interface gi0/12
Switch(config-if)# no switchport
Switch (config-if)# do show vlan internal usage
```

```
VLAN Usage
---- --------------------
1006 GigabitEthernet0/12

Switch(config-if)# exit
Switch(config)# interface gi0/13
Switch(config-if)# no switchport
Switch(config-if)# do show vlan internal usage

VLAN Usage
---- --------------------
1006 GigabitEthernet0/12
1007 GigabitEthernet0/13
```

Internal usage VLANs are internal to the switch, and regardless of the VTP mode, they are not stored in the VLAN database and are not advertised to any other switch in the VTP domain. The assignment of internal usage VLANs to routed ports is therefore only done at runtime and can differ between restarts of a switch, depending on the order that the routed ports are configured and on the unused extended VLAN IDs.

Because of the relatively discreet nature of internal usage VLANs (they are not visible in ordinary **show vlan** output), conflicts can ensue if an administrator tries to create an extended VLAN whose ID is—unknowingly to the administrator—already used by an internal usage VLAN, as shown in the Example 6-3.

Example 6-3 *Possible Internal Usage VLAN Conflict While Creating Extended VLANs*

```
! Building on the previous example, internal usage VLANs 1006 and 1007 exist
! on this switch. An administrator is not aware about their existence, though,
! and tries to create VLAN 1006 for its own use. Notice how the switch refuses
! to add the VLAN only after exiting the VLAN configuration.

Switch(config)# do show vlan internal usage

VLAN Usage
---- --------------------
1006 GigabitEthernet0/12
1007 GigabitEthernet0/13

Switch(config)# vlan 1006
Switch(config-vlan)# name SomeExtendedVLAN
Switch(config-vlan)# exit
% Failed to create VLANs 1006
VLAN(s) not available in Port Manager.
%Failed to commit extended VLAN(s) changes.
```

This problem can become especially unpleasant if VTPv3 is used that is capable of handling extended VLAN IDs. If the administrator creates an extended range VLAN on a VTP Primary Server switch, and the particular VLAN ID is already used by an internal usage VLAN on some other switch in the domain, VTP will fail to create this VLAN on that switch, resulting in connectivity issues. The conflict will be logged only on the switch experiencing the VLAN ID collision and so can elude the administrator's attention.

Key Topic

It is therefore generally recommended that if extended VLANs are used, they should be allocated from the end of the extended VLAN range that is opposite to the current internal VLAN allocation policy, to minimize the risk of creating VLAN ID collisions.

Keeping all these facts in mind, a routed port is practically equivalent to a switch port placed into a dedicated VLAN, with the Layer 2 control plane protocols deactivated on that port. From this viewpoint, a routed port is a syntactical device in the configuration to make the configuration quick and convenient, while the switch continues to handle the port internally as a switch port with a slightly modified operation.

The following two configuration snippets in Example 6-4 are practically equivalent; just the routed port is simpler to configure.

Example 6-4 *Routed Port and Its Internal Treatment by a Multilayer Switch*

```
! Following the previous example, assume the Gi0/12 is configured as follows:

Switch(config)# int gi0/12
Switch(config-if)# no switchport
Switch(config-if)# ip address 192.168.12.1 255.255.255.0
Switch(config-if)# do show vlan internal usage

VLAN Usage
---- --------------------
1006 GigabitEthernet0/12
```

```
! The above configuration is effectively equivalent to the following configuration:

Switch(config)# vlan 1006
Switch(config-vlan)# exit
Switch(config)# no spanning-tree vlan 1006
Switch(config)# no mac address-table learning vlan 1006
Switch(config)# interface GigabitEthernet0/12
Switch(config-if)# switchport mode access
Switch(config-if)# switchport access vlan 1006
Switch(config-if)# switchport nonegotiate
Switch(config-if)# no vtp
Switch(config-if)# exit
Switch(config)# interface Vlan1006
Switch(config-if)# ip address 192.168.12.1 255.255.255.0
```

Ethernet Port-channels can be used as routed interfaces as well. To do so, physical interfaces must be configured with the **no switchport** command *before* adding them to a channel group. The automatically created Port-channel interface inherits the configuration of the first physical interface added to the channel group; if that interface is configured as a routed interface, the entire Port-channel will be working as a routed port. An existing Layer 2 Port-channel cannot be changed from Layer 2 to Layer 3 operation and vice versa. If such a modification is necessary, it is first required to completely delete the entire Port-channel, unbundle the physical ports, reconfigure them into the desired mode of operation, and then add them into a channel group again, re-creating the Port-channel interface in the process. Also, when using a Port-channel as a routed interface, Port-channel load balancing should be based on Layer 3 addresses because the Layer 2 addresses will mostly be the MAC addresses of the two MLS switches on either end of the Port-channel. Port-channels can also be used as Layer 2 interfaces when doing MLS. In that case, VLAN interfaces would be configured with an IP address, and the Port-channel would simply act as any other Layer 2 interface.

Table 6-3 lists some of the specifics about each type of Layer 3 interface.

Key Topic

Table 6-3 *MLS Layer 3 Interfaces*

Interface	Forwarding to Adjacent Device	Configuration Requirements
VLAN interface	Uses Layer 2 logic and Layer 2 MAC address table	Create VLAN interface; VLAN must also exist
Physical (routed) interface	Forwards out physical interface	Use the **no switchport** command to create a routed interface
Port-channel (switched) interface	Not applicable; just used as another Layer 2 forwarding path	No special configuration; useful with VLAN interfaces
Port-channel (routed) interface	Balances across links in Port-channel	Needs the **no switchport** command to be used as a routed interface; optionally change load-balancing method

MLS Configuration

The upcoming MLS configuration example is designed to show all the configuration options. The network design is shown in Figures 6-6 and 6-7. In Figure 6-6, the physical topology is shown, with routed ports, VLAN trunks, a routed Port-channel, and access links. Figure 6-7 shows the same network, with a Layer 3 view of the subnets used in the network.

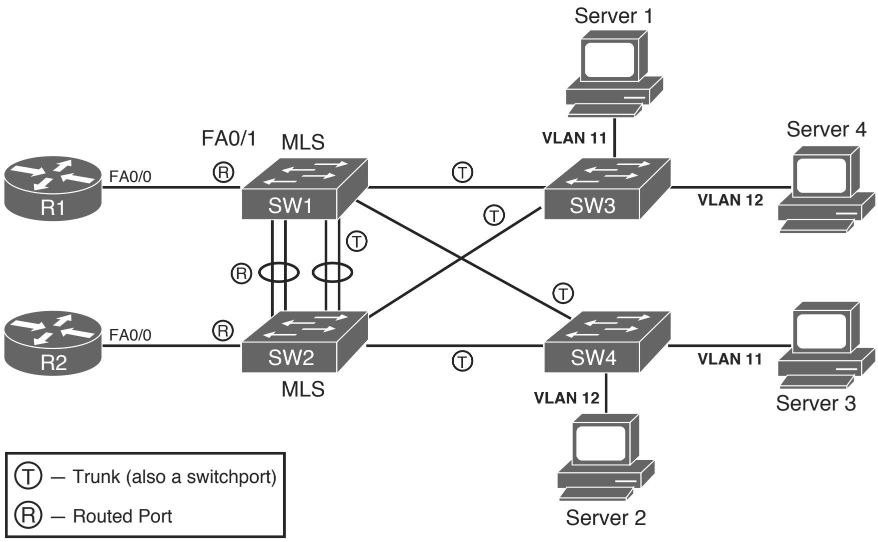


Figure 6-6 *Physical Topology: Example Using MLS*

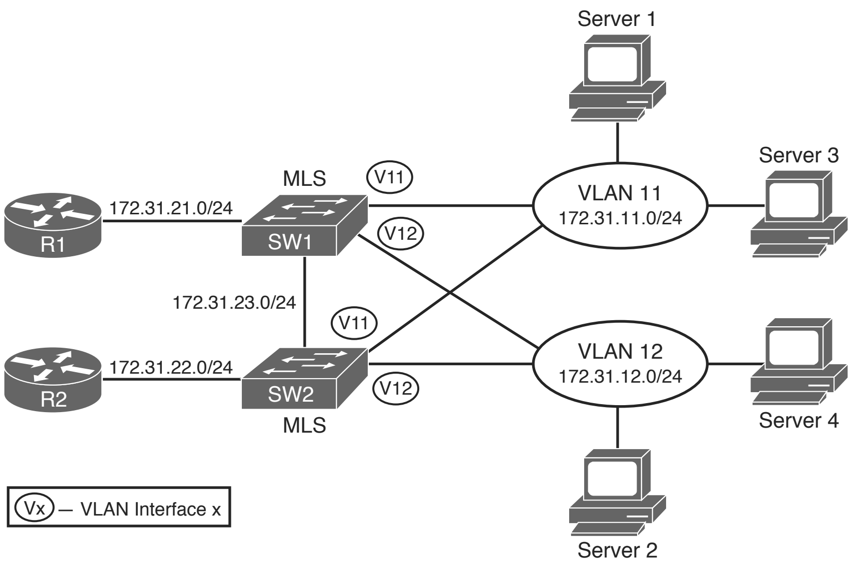


Figure 6-7 *Layer 3 Topology View: Example Using MLS*

A few design points bear discussion before jumping into the configuration. First, SW1 and SW2 need Layer 2 connectivity to support traffic in VLANs 11 and 12. In this particular example, a trunk is used between SW1 and SW2 as well as between SW1/SW2 and SW3/SW4. Focusing on the Layer 2 portions of the network, SW1 and SW2, both distribution MLS switches, connect to SW3 and SW4, which are access layer switches. SW1 and SW2 are responsible for providing full connectivity in VLANs 11 and 12. Having full Layer 2 connectivity between switches in a topology is the traditional approach. In newer deployments, a new approach is favored in which SW1 and SW2 are interconnected through a routed port (Layer 3 link) only, and the connections toward access layer switches are Layer 2 or even Layer 3. This allows for shrinking the size of Layer 2 domain and the resulting scope of STP operation. If only a routed link was left between SW1 and SW2, the Layer 2 topology between SW1/SW2 and SW3/SW4 would be physically loop-free and there would be no ports blocked by STP, requiring little or no reaction of STP if a link is added or removed.

Additionally, this design uses SW1 and SW2 as Layer 3 switches, so the hosts in VLANs 11 and 12 will use SW1 or SW2 as their default gateway. For better availability, the two switches should use HSRP, VRRP, or GLBP. Regardless of which protocol is used, both SW1 and SW2 need to be in VLANs 11 and 12, with connectivity in those VLANs, to be effective as default gateways.

In addition to a Layer 2 trunk between SW1 and SW2, to provide effective routing, it makes sense for SW1 and SW2 to have a routed path between each other as well. Certainly, SW1 needs to be able to route packets to Router R1, and SW2 needs to be able to route packets to Router R2. However, routing between SW1 and SW2 allows for easy convergence if R1 or R2 fails.

Figure 6-6 shows two alternatives for routed connectivity between SW1 and SW2, and one option for Layer 2 connectivity. For Layer 2 connectivity, a VLAN trunk needs to be used between the two switches. Figure 6-6 shows a pair of trunks between SW1 and SW2 (labeled with a circled T) as a Layer 2 Port-channel. The Port-channel would support the VLAN 11 and 12 traffic.

To support routed traffic, the figure shows two alternatives: Simply route over the Layer 2 Port-channel using VLAN interfaces or use a separate routed Port-channel. First, to use the Layer 2 Port-channel, SW1 and SW2 could simply configure VLAN interfaces in VLANs 11 and 12. The alternative configuration uses a second Port-channel that will be used as a routed Port-channel. However, the routed Port-channel does not function as a Layer 2 path between the switches, so the original Layer 2 Port-channel must still be used for Layer 2 connectivity. Upcoming Example 6-5 shows both configurations.

Finally, a quick comment about Port-channels is needed. This design uses Port-channels between the switches, but they are not required. Most links between switches today use at least two links in a Port-channel, for the typical reasons—better availability, better convergence, and less STP overhead. This design includes the Port-channel to point out a small difference between the routed interface configuration and the routed Port-channel configuration.

Example 6-5 shows the configuration for SW1, with some details on SW2.

Example 6-5 *MLS-Related Configuration on Switch1*

```
! Below, note that the switch is in VTP transparent mode, and VLANs 11 and 12 are
! configured, as required. Also note the ip routing global command, without which
! the switch will not perform Layer 3 switching of IP packets.

vlan 11
!
vlan 12

! The ip routing global command is required before the MLS will perform
! Layer 3 forwarding. Similarly, ipv6 unicast-routing is required for
! IPv6 routing to be enabled. On selected Catalyst platforms, the use of
! distributed keyword is required, as the CEF operates in distributed mode
! on these switches - over multiple ASICs or line cards.

ip routing
ipv6 unicast-routing distributed
!
vtp domain CCIE-domain
vtp mode transparent

! Next, the configuration shows basic Port-channel creation commands, with the
! no switchport command being required before bundling physical ports into
! a Port-channel. Note the Port-channel interface will be created automatically.

interface GigabitEthernet0/1
 no switchport
 no ip address
 channel-group 1 mode desirable
!
interface GigabitEthernet0/2
 no switchport
 no ip address
 channel-group 1 mode desirable


! Next, the Port-channel interface is assigned an IP address.

interface Port-channel1
 ip address 172.31.23.201 255.255.255.0

! Below, similar configuration on the interface connected to Router1.
```

```
interface FastEthernet0/1
 no switchport
 ip address 172.31.21.201 255.255.255.0

! Next, interface Vlan 11 gives Switch1 an IP presence in VLAN11. Devices in VLAN
! 11 can use 172.31.11.201 as their default gateway. However, using HSRP is
! better, so Switch1 has been configured to be HSRP primary in VLAN11, and Switch2
! to be primary in VLAN12, with tracking so that if Switch1 loses its connection
! to Router1, HSRP will fail over to Switch2.

interface Vlan11
 ip address 172.31.11.201 255.255.255.0
 standby 11 ip 172.31.11.254
 standby 11 priority 90
 standby 11 preempt
 standby 11 track FastEthernet0/1

! Below, VLAN12 has similar configuration settings, but with a higher (better)
! HSRP priority than Switch2's VLAN 12 interface.

interface Vlan12
 ip address 172.31.12.201 255.255.255.0
 standby 12 ip 172.31.12.254
 standby 12 priority 110
 standby 12 preempt
 standby 12 track FastEthernet0/1
```

Note For MLS switches to route using VLAN interfaces, the **ip routing** global command must be configured. MLS switches will not perform Layer 3 routing without the **ip routing** command, which is not enabled by default. Similar comments apply to IPv6 routing that needs to be enabled by **ipv6 unicast-routing**.

As stated earlier, the routed Port-channel is not required in this topology. It was included to show an example of the configuration, and to provide a backdrop from which to discuss the differences. However, as configured, SW1 and SW2 are Layer 3 adjacent over the routed Port-channel as well as through their VLAN 11 and 12 interfaces. So, they could exchange interior gateway protocol (IGP) routing updates over three separate subnets. In such a design, the routed Port-channel was probably added so that it would be the normal Layer 3 path between SW1 and SW2. Care should be taken to tune the IGP implementation so that this route is chosen instead of the routes over the VLAN interfaces.

Policy Routing

All the options for IP forwarding (routing) in this chapter had one thing in common: The destination IP address in the packet header was the only thing in the packet that was used to determine how the packet was forwarded. Policy routing (or Policy-Based Routing [PBR]) allows a router to make routing decisions based on information besides the destination IP address.

Policy routing's logic begins, depending on IPv4 or IPv6 in use, with the **ip policy** or **ipv6 policy** command on an interface. This command tells the IOS to process *incoming* packets on that interface with different logic before the normal forwarding logic takes place. (To be specific, policy routing intercepts the packet after Step 4, but before Step 5, in the routing process shown in Figure 6-1.) The IOS compares the received packets using the **route-map** referenced in the **ip policy** or **ipv6 policy** command. Figure 6-8 shows the basic logic.

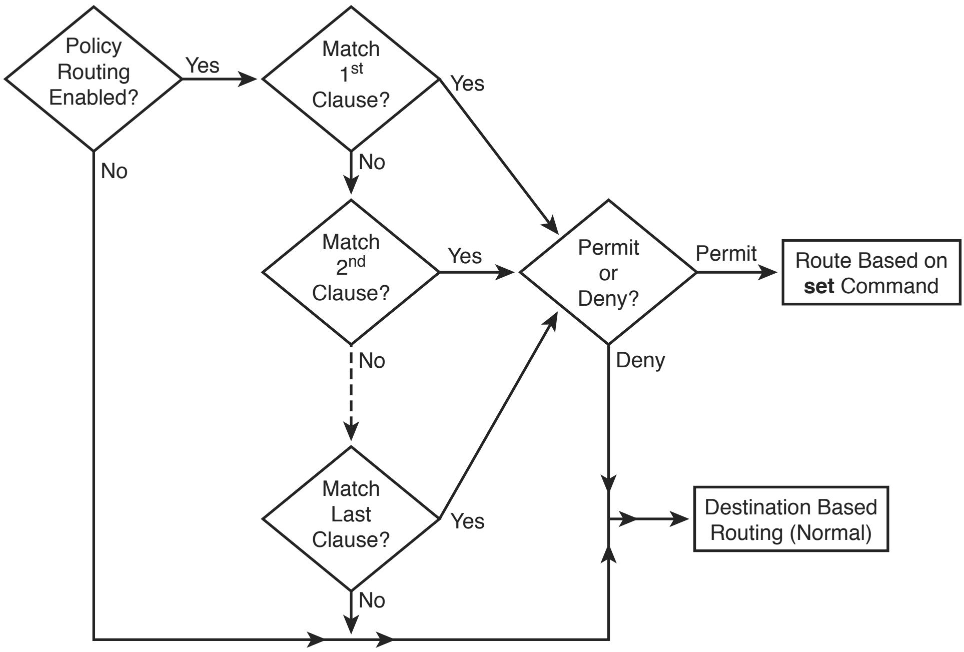


Figure 6-8 *Basic Policy Routing Logic*

Specifying the matching criteria for policy routing is relatively simple compared to defining the routing instructions using the **set** command. The route maps used by policy routing must match either based on referring to an ACL (numbered or named IPv4/IPv6 ACL, using the **match ip address** or **match ipv6** address command) or based on packet length (using the **match length** command). To specify the routing instructions—in other words, where to forward the packet next—use the **set** command. Table 6-4 lists the **set** commands and provides some insight into their differences.

Key Topic

Table 6-4 *Policy Routing Instructions (***set** *Commands)*

Command	Comments
set ip next-hop *ip-address* [.... *ip-address*]	Next-hop addresses must be in a connected subnet; forwards to the first address in the list for which the associated interface is up. Supported for both IPv4 and IPv6.
set ipv6 next-hop *ipv6-address [... ipv6-address]*	
set ip default next-hop *ip-address*[.... *ip-address*]	Same logic as previous command, except policy routing first attempts to route based on the routing table, and only if no match is found in the routing table, the packet will be handled by PBR. Default route in the routing table is ignored; that is, if the packet's destination is matched only by the default route, the packet will be handled by PBR. Supported for both IPv4 and IPv6.
set ipv6 default next-hop *ipv6-address* [... *ipv6-address*]	
set interface *interface-type interface-number* [.... *interface-type interface-number*]	Forwards packets using the first interface in the list that is up. Recommended only for point-to-point interfaces; strongly discouraged for multiaccess interfaces. Supported for both IPv4 and IPv6.
set default interface *interface-type interface-number* [. . . *interface-type interface-number*]	Same logic as previous command, except policy routing first attempts to route based on the routing table, and only if no match is found in the routing table, the packet will be handled by PBR. Default route in the routing table is ignored, that is, if the packet's destination is matched only by the default route, the packet will be handled by PBR. Recommended only for point-to-point interfaces; strongly discouraged for multiaccess interfaces. Supported for both IPv4 and IPv6.
set ip df *number*	Sets the IP DF bit; can be either 0 or 1. Supported only for IPv4.
set ip precedence *number* | *name*	Sets IP precedence bits; can be a decimal value in the range 0–7 or a textual name (IPv4 only). Supported for both IPv4 and IPv6.
set ipv6 precedence *number*	
set ip tos *number* | *name*	Sets the ToS bits (delay, throughput, reliability, monetary cost); can be decimal value or ASCII name. Supported for IPv4 only.

The first four **set** commands in Table 6-4 are the most important ones to consider. Essentially, you set either the next-hop IP address or the outgoing interface. Use the outgoing interface option *only* when it is of point-to-point technology type—for example, do not refer to a LAN interface or multipoint Frame Relay subinterface. This will almost certainly cause the policy-based routing to fail or act unexpectedly; details will be discussed later. Most importantly, note the behavior of the **default** keyword in the **set** commands. Use of the **default** keyword essentially means that policy routing tries the default (destination-based) routing first, and resorts to using the **set** command details only when the router finds no matching route in the routing table. Note that a default route is not considered a matching route by the **default** keyword. If a packet's destination is matched only by the default route, PBR treats this as if no match occurred, and the packet is eligible to be forwarded according to the **set** commands using the **default** keyword.

The remaining **set** commands set the bits inside the ToS byte of the packet; refer to Chapter 5, "Classification and Marking," in Volume II for more information about the ToS byte and QoS settings. Note that you can have multiple **set** commands in the same **route-map** clause. For example, you might want to define the next-hop IP address and mark the packet's ToS at the same time. A single route map entry can even contain multiple set statements specifying where the packet shall be forwarded. In such cases, the **set** statements are evaluated in the following order:

Key Topic

1. **set ip next-hop / set ipv6 next-hop**
2. set interface
3. **set ip default next-hop / set ipv6 default next-hop**
4. **set default interface**

The use of **set interface** and **set default interface** is strongly recommended only with point-to-point interfaces. Using multiaccess interfaces in these commands will lead to PBR failing in most cases. IPv6 PBR using **set** [**default**] **interface** with a multiaccess interface fails outright; differences in very selected cases have been observed under different IOS versions. IPv4 PBR under the same circumstances might appear to work but the background processes are unintuitive: The router first performs a normal routing table lookup for the packet's destination IP address to look for the connected next-hop address, and then tries to translate this next-hop address into the appropriate Layer 2 address on the multiaccess interface specified in the **set** [**default**] **interface** command. This can fail for obvious reasons: The routing table might provide no match for the packet's destination and thus the **set** [**default**] **interface** is skipped, or the next hop itself might be connected to a different interface. Even Proxy ARP, if applicable, is not going to help much—Cisco routers perform a validity check on received ARP responses similar to a unicast reverse path forwarding check. A router verifies using its routing table whether the sender IPv4 address in the ARP response's body (the address whose MAC address is being asked for) is reachable through the interface the ARP response came in. If this check fails, the router will drop the ARP response, claiming that it arrived over the "wrong cable" in the **debug arp** output. Once again, the use of **set** [**default**] **interface** is appropriate only with point-to-point interfaces. IOS Releases 15.x display an explicit warning if the command is used with multiaccess interface types.

The IPv6 PBR with **set interface** in particular has one more peculiarity: In some IOS versions, the router checks whether there is a matching route (ignoring the default route) for the packet's destination even if the packet is to be handled by PBR. If there is no matching route in the routing table, the **set interface** command is ignored. It is also noteworthy to mention that on some platforms, this behavior also depends on the state IPv6 CEF. The particular behavior of the IOS in question should therefore be verified using **debug ipv6 policy**.

Key Topic

If PBR is required on a multilayer switch, many lower-end switches, such as Catalyst 3550, 3560, or 3750, require that the TCAM in the switch is repartitioned in a different way, providing TCAM space for PBR entries while taking away space from entries of other types. On these platforms, the size of TCAM regions for individual applications cannot be configured individually; instead, a set of templates is prepared for typical switch deployments. A switch should be configured with an appropriate TCAM partitioning template that allocates the most space to the types of entries most required in the particular switch's mode of deployment. A template that provides space for PBR entries must be active before the PBR can be configured. These templates are called Switch Database Management templates, or SDM templates for short. Current SDM templates can be shown using the **show sdm prefer** command, also displaying an approximate space for different TCAM entry types. This command can be also used to view the TCAM allocation policy for different templates if the **show sdm prefer** *template-name* form is used. To allow for PBR usage on the switch models mentioned previously, either the **routing**, **access**, or **dual-ipv4-and-ipv6 routing** (if supported) SDM template needs to be used. On Catalyst 3650 and 3850 Series, the **advanced** SDM template is required. To activate a particular template, the **sdm prefer** *template-name* global configuration level command is used. After you issue this command, the switch must be reloaded. It is strongly recommended to consult the appropriate switch model documentation for the list of supported SDM templates and the individual features they activate.

Apart from PBR, changing the SDM template on an MLS might also be required if routing or IPv6 support are to be activated. One of indications that an inappropriate SDM template is currently active is very visible: The IOS CLI appears to lack the commands necessary to configure routing, PBR, or IPv6, even though the IOS should support these features and the appropriate licenses are in place.

Routing Protocol Changes and Migration

The proper selection of a routing protocol for a network is always a sensitive (and understandably difficult) task. Many factors need to be taken into consideration, ranging from the protocol's scalability and speed of convergence through advanced features, ending with compatibility issues especially in multivendor environments; all of these are related to the network's design and requirements. As the network evolves, it might become necessary to reevaluate the choice of a particular routing protocol, and if it is found to be inappropriate, it might need to be replaced.

Migrating from one routing protocol to another is always a disruptive change to the network. It requires careful planning to minimize the outages, and even then, they are inevitable, although their duration can be kept very low. Therefore, a routing protocol migration always requires a maintenance window.

Routing protocol migration is usually accomplished with the following steps:

Step 1. Plan the migration strategy.

Step 2. Activate the new routing protocol on all routers in the topology, raising its administrative distance (AD) above the ADs of the current IGP. If the new IGP is Routing Information Protocol (RIP) or Enhanced Interior Gateway Routing Protocol (EIGRP), redistribution from the current into the new IGP has to be configured on each router as well. The current IGP is left intact.

Step 3. Verify the new IGP's adjacencies and optionally the working database contents.

Step 4. Deactivate the current IGP in a gradual fashion.

Step 5. Remove the temporary settings from the new IGP.

We describe each of these steps in closer detail.

Planning the Migration Strategy

The deployment of a new routing protocol should be preplanned for the entire network, including the division of network into separate areas if and when a link-state IGP is to be used. Additionally, protocol features such as prefix summarization/filtration, stub features, and external information redistribution can further isolate areas of the network from one another. This planning should also involve the order in which routers will be migrated over from the current IGP to the new one. Ideally, routers should be migrated so that they form a contiguous, ever-growing part of the network running the new IGP, gradually shrinking the contiguous remainder of the network in which both the current and new IGP are run. If the current IGP is a link-state protocol, it is advisable to perform the migration in a per-area fashion. The backbone routers should be the last ones to migrate.

Activating New IGP While Keeping the Current IGP Intact

Key Topic

According to the planning in the previous step, the new IGP should be activated on the routers in the network, first setting its administrative distance (AD) to a higher value than the current IGP's AD, and only then adding interfaces and networks to the new IGP and activating selected features. The current IGP is left running and its configuration is unchanged throughout this entire step. If the current IGP uses various ADs for different network types (for example, EIGRP uses 90 and 170 for internal and external routes, respectively), the new IGP's AD should be reconfigured to be higher than the highest AD used by the existing IGP. As an example, if the current IGP is OSPF and the new IGP should be EIGRP, the ADs of EIGRP should, for the duration of the migration, be reconfigured to, say, 210 and 220 for internal and external EIGRP routes, respectively. This way, the new IGP can be deployed across the network, creating adjacencies between routers as usual but not influencing the routing tables and routing just yet. If the current IGP configuration includes redistribution from other sources (static routes, directly connected networks, and so on), the new IGP shall be configured similarly.

If the new IGP is a distance-vector routing protocol (RIP or EIGRP), each router must also be configured with redistribution from the current IGP into the new IGP. Reasons for this are explained later in the chapter.

Verifying New IGP Adjacencies and Working Database Contents

After the new IGP has been configured across the entire network, it should have created adjacencies in the usual fashion though the routing tables are not populated by its routes yet. These adjacencies should be verified to make sure that they are complete. After the current IGP is deactivated, these adjacencies are the only routing protocol adjacencies left between migrated routers, and so must be working as expected before the current IGP starts being removed.

Key Topic

It is often recommended to verify the contents of the working databases in the new IGP to check whether all expected networks are present, even though not placed into routing tables because of higher ADs. This step might be difficult to accomplish, though, because of two reasons. First, the amount and format of the data can be overwhelming to a human, requiring some kind of automated processing. The second reason is relevant only if the new IGP is a distance-vector protocol, that is, either RIP or EIGRP. These protocols advertise a learned route only if it is also installed in the routing table by the same protocol. This additional advertisement logic in distance-vector routing protocols is based on the fact that a router should not advertise a route it is not using itself. Because the AD of the new IGP has been configured to be higher than the current IGP's AD, routes learned by the new IGP will not be placed into the router's routing table as long as the current IGP is still running on the router, and hence will not be advertised further. As a result, if the new IGP is RIP or EIGRP, its working databases will contain only partial contents until the migration starts, making the verification before migration impossible. This behavior of distance-vector IGPs will be discussed in closer detail later in the chapter. Note that this additional advertisement logic does not apply to link-state IGPs such as OSPF and IS-IS, as the nature of routing information they generate and the flooding mechanism are strongly different from distance-vector IGPs and do not allow for such additional checks.

Deactivating Current IGP

The next step in the routing protocol migration involves the actual removal of the current IGP from a contiguous set of routers, one router at a time, allowing the new routing protocol to populate the routing table instead, and then proceeding to the next router. Alternatively, instead of plainly deleting the current IGP configuration from the router, it can be configured using the **passive-interface default** command that will effectively shut it down. In recent IOS versions, selected routing protocols even support the **protocol shutdown** or **shutdown** command. The obvious advantage of this approach is that the configuration of the current IGP is preserved, should it ever be necessary to activate it again quickly.

The removal or deactivation of the current IGP should be done in such a way that the network always consists of at most two regions. In one, both routing protocols are run (the unmigrated part of network), and in the other, only the new protocol is running (the migrated part of the network) and both regions are contiguous.

Key Topic

During a properly executed migration, the network consists of a contiguous region that runs both IGPs and of a contiguous region that runs the new IGP only. Traffic crossing the network enters either an unmigrated or a migrated router, and is destined to a network that is directly connected to a router that is again either migrated or unmigrated yet. These options have an impact on which IGPs carry the information about the destination and thus what source of routing information is used by routers along the way.

If traffic enters an *unmigrated* router and is destined to a network connected to an *unmigrated* router, the destination network is advertised in both IGPs but the new IGP has been configured with a higher AD, so it has no impact on the routing table contents. Consequently, the traffic completely follows the path provided by the current IGP, as if no migration was taking place.

If traffic enters a *migrated* router and is destined to a network connected to a *migrated* router, the destination network is advertised only in the new IGP, as the current IGP has been removed from the destination router. The current IGP does not advertise this network anymore and does not compete about this particular network with the new IGP (recall that it would otherwise be resolved in favor of the current IGP thanks to its lower AD). Consequently, all routers, both migrated and unmigrated, know about this destination only through the new IGP, and follow the path provided by the new IGP.

If traffic enters an *unmigrated* router and is destined to a network connected to a *migrated* router, the situation is very similar. As the current IGP has been removed from the destination router, the destination network is advertised only in the new IGP. All routers therefore know about this network through the new IGP only and follow the path provided by the new IGP.

Finally, if traffic enters a *migrated* router and is destined to a network connected to an *unmigrated* router, the situation is slightly more complex. The destination router advertises the network through both IGPs. Other unmigrated routers know the destination network through both IGPs and prefer the current IGP, while migrated routers, including the ingress router, know the network through the new IGP only. In the migrated path of the network, the traffic will be routed according to the new IGP until it is forwarded to the first unmigrated router. Starting with this router, all other routers on the path toward the destination still prefer the path provided by the current IGP. Therefore, beginning with this router, the traffic will be routed according to the current IGP.

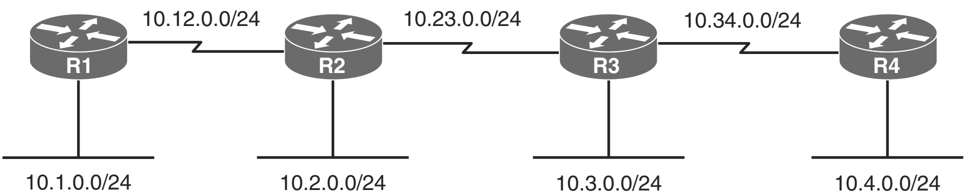
This analysis shows that during a properly executed migration, the network remains fully connected and destinations should remain fully reachable. Transient outages can occur at the moment when the current IGP is removed from a router, as the routes provided by the current IGP will need to be flushed from the routing table and replaced by routes learned through the new IGP.

Removing New IGP's Temporary Settings

After the network has been completely migrated to the new IGP and the previous IGP has been completely removed from all routers, the new IGP still contains temporary settings that were necessary for a seamless migration, especially the modified AD values, leftovers from redistribution of the previous IGP into the new IGP, and so on. These settings should be removed as the last step of the migration procedure. In link-state routing protocols, removing the temporary settings should not cause any additional interruptions in network service. However, in EIGRP, modifying the AD values causes the router to drop and reestablish its EIGRP adjacencies with neighboring routers, causing a transient disruption in network connectivity. These changes must therefore be also performed during a maintenance window.

Specifics of Distance-Vector Protocols in IGP Migration

Ideally, migrating to a different routing protocol should not involve any route redistribution between the current and the new IGP, as the redistribution involves additional complexity to the migration process. However, if the new IGP is a distance-vector protocol (such as RIP or EIGRP), a temporary redistribution is inevitable. The reason lies in the advertisement logic of these routing protocols: A learned route will be advertised further *only* if the router has placed that very learned route into the routing table as well. In other words, a learned route is advertised through the same routing protocol only if the router is using that route itself. As the migration process involves temporarily configuring the new IGP's administrative distance (AD) to be higher than the AD of the current IGP, none of the learned routes through the new IGP are going to be placed into the routing table if the current IGP is still running. If the new IGP happens to be RIP or EIGRP, any route learned through that protocol won't make it into the router's routing table and will not be advertised further as a result. To illustrate this concept, consider the network in Figure 6-9 (split horizon rules in EIGRP have been omitted for simplicity).



EIGRP	R1	R2	R3	R4
Advertises	10.1.0.0/24 10.12.0.0/23	10.12.0.0/24 10.2.0.0/24 10.23.0.0/24	10.23.0.0/24 10.3.0.0/24 10.34.0.0/24	10.34.0.0/24 10.4.0.0/24
Learns	10.2.0.0/24 10.23.0.0/24	10.1.0.0/24 10.3.0.0/24 10.34.0.0/24	10.12.0.0/24 10.2.0.0/24 10.4.0.0/24	10.23.0.0/24 10.3.0.0/24

Figure 6-9 *Example Network Topology for Routing Protocol Migration*

OSPF is the current routing protocol in this network, and the network is planned to be migrated to EIGRP. All four routers are therefore configured with EIGRP as well, the EIGRP AD is set to 210 for internal and 220 for external routes, and all interfaces are added to EIGRP on all routers. OSPF's operation is not influenced in any way, and because its AD remains at 110, routers still keep OSPP-learned routes in their routing table. If we focus on R1's operation and on the 10.1.0.0/24 network in particular, R1 advertises its directly connected networks, including 10.1.0.0/24 to R2 through EIGRP. R2 will have this route in its EIGRP topology table but will be unable to install it into the routing table because of EIGRP's modified AD of 210. As a result, R2 will not propagate the EIGRP-learned route 10.1.0.0/24 through EIGRP to R3, so neither R3 nor R4 will learn about this network through EIGRP. This limited propagation of networks in EIGRP will take place on each router in this topology: Each router will advertise its directly connected networks in EIGRP to its immediate neighbors, but these neighbors are prevented from advertising them further, as shown in Figure 6-9. Looking into EIGRP topology tables of all routers confirms this, as shown in Example 6-6.

Example 6-6 *Contents of EIGRP Topology Tables in Figure 6-9 Topology*

```
! On all routers in the topology from Figure 6-9, EIGRP is configured identically:

router eigrp 1
 network 10.0.0.0
 distance eigrp 210 220
 no auto-summary

! It is assumed that OSPF is also running on all four routers.

! show ip eigrp topology on R1:

R1# show ip eigrp topology
IP-EIGRP Topology Table for AS(1)/ID(10.12.0.1)

Codes: P - Passive, A - Active, U - Update, Q - Query, R - Reply,
       r - reply Status, s - sia Status

P 10.12.0.0/24, 1 successors, FD is 832000
        via Connected, Serial0/0/0
P 10.2.0.0/24, 0 successors, FD is Inaccessible
        via 10.12.0.2 (857600/281600), Serial0/0/0
P 10.1.0.0/24, 1 successors, FD is 281600
        via Connected, FastEthernet0/0
P 10.23.0.0/24, 0 successors, FD is Inaccessible
        via 10.12.0.2 (1344000/832000), Serial0/0/0

! show ip eigrp topology on R2:

R2# show ip eigrp topology
```

```
IP-EIGRP Topology Table for AS(1)/ID(10.23.0.2)

Codes: P - Passive, A - Active, U - Update, Q - Query, R - Reply,
       r - reply Status, s - sia Status

P 10.12.0.0/24, 1 successors, FD is 832000
        via Connected, Serial0/0/1
P 10.2.0.0/24, 1 successors, FD is 281600
        via Connected, FastEthernet0/0
P 10.3.0.0/24, 0 successors, FD is Inaccessible
        via 10.23.0.3 (857600/281600), Serial0/0/0
P 10.1.0.0/24, 0 successors, FD is Inaccessible
        via 10.12.0.1 (857600/281600), Serial0/0/1
P 10.23.0.0/24, 1 successors, FD is 832000
        via Connected, Serial0/0/0
P 10.34.0.0/24, 0 successors, FD is Inaccessible
        via 10.23.0.3 (1344000/832000), Serial0/0/0

! show ip eigrp topology on R3:

R3# show ip eigrp topology
IP-EIGRP Topology Table for AS(1)/ID(10.34.0.3)

Codes: P - Passive, A - Active, U - Update, Q - Query, R - Reply,
       r - reply Status, s - sia Status

P 10.12.0.0/24, 0 successors, FD is Inaccessible
        via 10.23.0.2 (1344000/832000), Serial0/0/1
P 10.2.0.0/24, 0 successors, FD is Inaccessible
        via 10.23.0.2 (857600/281600), Serial0/0/1
P 10.3.0.0/24, 1 successors, FD is 281600
        via Connected, FastEthernet0/0
P 10.4.0.0/24, 0 successors, FD is Inaccessible
        via 10.34.0.4 (857600/281600), Serial0/0/0
P 10.23.0.0/24, 1 successors, FD is 832000
        via Connected, Serial0/0/1
P 10.34.0.0/24, 1 successors, FD is 832000
        via Connected, Serial0/0/0

! show ip eigrp topology on R4:

R4# show ip eigrp topology
IP-EIGRP Topology Table for AS(1)/ID(10.34.0.4)

Codes: P - Passive, A - Active, U - Update, Q - Query, R - Reply,
       r - reply Status, s - sia Status
```

```
P 10.3.0.0/24, 0 successors, FD is Inaccessible
        via 10.34.0.3 (857600/281600), Serial0/0/1
P 10.4.0.0/24, 1 successors, FD is 281600
        via Connected, FastEthernet0/0
P 10.23.0.0/24, 0 successors, FD is Inaccessible
        via 10.34.0.3 (1344000/832000), Serial0/0/1
P 10.34.0.0/24, 1 successors, FD is 832000
        via Connected, Serial0/0/1
```

Note that on each router, only directly connected networks of its immediate neighbors are learned through EIGRP, and all these networks are marked with a "0 successors, FD is Inaccessible" indication in their heading, preventing them from being advertised further.

After OSPF is removed from R4's configuration as a step in the migration procedure, the OSPF-learned 10.1.0.0/24 will be removed from R4's routing table without being replaced by an EIGRP-learned route, as R2 is still running OSPF and does not advertise this route through EIGRP. This will cause connectivity outages: R4 will learn only about directly connected networks from R3 through EIGRP, missing all other networks, and R3—still running OSPF—will be unable to forward EIGRP-learned routes from R4 back to R2. Clearly, full connectivity in this network will be restored only after OSPF is completely removed.

The solution to this problem is to configure route redistribution from the current IGP into the new IGP on each router in the topology. In the example network, the situation will be significantly different, then: Because each router knows about all networks through OSPF, redistributing them from OSPF into EIGRP allows each router to advertise them all to each directly connected neighbor. While the neighbor will not be allowed to advertise them further if still running OSPF, its EIGRP topology database will nonetheless be populated with the full set of networks from its own neighbors. When OSPF is deactivated on a router, EIGRP-learned routes will take over—they will get installed into the routing table, and the router will be able to forward them further.

If the new IGP is a link-state protocol, this redistribution is unnecessary and shall not be configured. Flooding of topological information in link-state protocols is not constrained by routing table contents. Routers will always flood the routing information in a link-state protocol, regardless of whether routes derived from that information are installed into routing tables or not.

To analyze how this approach works, assume that the migration of the network in Figure 6-9 continues by gradual deactivation of OSPF, starting on R4 and proceeding router by router toward R1. Table 6-5 summarizes how the individual networks are visible in the routing tables of individual routers. Only the first two octets of each prefix are listed for brevity. Prefixes in the O row are learned by OSPF; prefixes in the D row are learned by EIGRP. Directly connected networks are not listed, as they are not influenced by changes in routing protocols.

Table 6-5 *Contents of Routing Tables in Different Migration Stages*

OSPF Run On		R1	R2	R3	R4
R1 to R4	O	10.2/24	10.1/24	10.1/24	10.1/24
		10.23/24	10.3/24	10.12/24	10.12/24
		10.3/24	10.34/24	10.2/24	10.2/24
		10.34/24	10.4/24	10.4/24	10.23/24
		10.4/24			10.3/24
	D	None	None	None	None
R1 to R3	O	10.2/24	10.1/24	10.1/24	None
		10.23/24	10.3/24	10.12/24	
		10.3/24	10.34/24	10.2/24	
		10.34/24			
	D	10.4/24	10.4/24	10.4/24	10.1/24 (EX)
					10.12/24 (EX)
					10.2/24 (EX)
					10.23/24
					10.3/24
R1 to R2	O	10.2/24	10.1/24	None	None
		10.23/24			
	D	10.3/24	10.3/24	10.1/24 (EX)	10.1/24 (EX)
		10.34/24	10.34/24	10.12/24	10.12/24
		10.4/24	10.4/24	10.2/24	10.2/24
				10.4/24	10.23/24
					10.3/24
R1 only	O	None	None	None	None
	D	10.2/24	10.1/24	10.1/24	10.1/24
		10.23/24	10.3/24	10.12/24	10.12/24
		10.3/24	10.34/24	10.2/24	10.2/24
		10.34/24	10.4/24	10.4/24	10.23/24
		10.4/24			10.3/24

Key observations about this table are as follows:

- Prefixes advertised from routers running both the original and new routing protocol are learned by the original routing protocol on all routers still running it.
- Prefixes advertised from routers running only the new routing protocol are learned by the new routing protocol across the entire network.
- At all times, all routers know about all prefixes.
- Traffic entering a router running both routing protocols and destined to a network on a router running both protocols is routed completely according to the original routing protocol without changes. This is because the network is advertised in both protocols and the new routing protocol's AD has been intentionally raised above the original protocol's AD.
- Traffic entering a router running the new routing protocol and destined to a network on a router running the new protocol is routed completely according to the new routing protocol. This is because the network in question is not injected into the original routing protocol anymore, so the only source of the information is the new protocol.
- Traffic entering a router running both routing protocols and destined to a network on a router running the new routing protocol is routed completely according to the new routing protocol. The reason is the same as in the previous item.
- Traffic entering a router running the new routing protocol and destined to a network on a router running both routing protocols will be routed according to the new routing protocol until it hits the first router that still runs both routing protocols. Afterward, it will be routed according to the original routing protocol. This is because in the migrated part of the network, routers run only the new routing protocol, while in the remaining part of network running both protocols, the original routing protocol is preferred.

The last four items are valid if the migration is performed in such a way that the network always consists of at most two contiguous regions. In one, both routing protocols are run (the unmigrated part of network), and in the other, only the new protocol is running (the migrated part of the network). Also, if this rule is maintained throughout the migration process, the boundary between the new and original routing protocol as described in the last item is crossed only once.

Foundation Summary

This section lists additional details and facts to round out the coverage of the topics in this chapter. Unlike most of the Cisco Press Exam Certification Guides, this "Foundation Summary" does not repeat information presented in the "Foundation Topics" section of the chapter. Please take the time to read and study the details in the "Foundation Topics" section of the chapter, as well as review items noted with a Key Topic icon.

Table 6-6 lists the protocols mentioned in or pertinent to this chapter and their respective standards documents.

Table 6-6 *Protocols and Standards for Chapter 6*

Name	Standardized In
Address Resolution Protocol (ARP)	RFC 826
IPv6 Neighbor Discovery	RFC 4861, RFC 5942
Differentiated Services Code Point (DSCP)	RFC 2474

Table 6-7 lists some of the key IOS commands related to the topics in this chapter. (The command syntax for switch commands was taken from the *Catalyst 3560 Multilayer Switch Command Reference, 15.0(2)SE*. Router-specific commands were taken from the IOS Release 15 mainline Command Reference.)

Table 6-7 *Command Reference for Chapter 6*

Command	Description
show ip arp	EXEC command that displays the contents of the IP ARP cache.
show ipv6 neighbors	EXEC command that displays the contents of the IPv6 neighbor cache.
[**no**] **switchport**	Switch interface subcommand that toggles an interface between a Layer 2 switched function (**switchport**) and a routed port (**no switchport**).
[**no**] **ip route-cache cef**	Interface subcommand that enables or disables CEF switching on an interface.
[**no**] **ip cef**	Global configuration command to enable (or disable) CEF on all interfaces.
[**no**] **ipv6 cef**	Global configuration command to enable (or disable) CEF for IPv6 on all interfaces. For IPv6 CEF to be activated, **ip cef** must also be present.

Command	Description
[**no**] **ip routing**	Enables IP routing; defaults to **no ip routing** and **no ipv6 unicast-routing** on a multilayer switch.
[**no**] **ipv6 unicast-routing**	
ip policy route-map *map-tag*	Router interface subcommand that enables policy routing for the packets entering the interface.
ipv6 policy route-map *map-tag*	

Refer to Table 6-4 for the list of **set** commands related to policy routing.

Memory Builders

The CCIE Routing and Switching written exam, like all Cisco CCIE written exams, covers a fairly broad set of topics. This section provides some basic tools to help you exercise your memory about some of the broader topics covered in this chapter.

Fill In Key Tables from Memory

Appendix E, "Key Tables for CCIE Study," on the CD in the back of this book, contains empty sets of some of the key summary tables in each chapter. Print Appendix E, refer to this chapter's tables in it, and fill in the tables from memory. Refer to Appendix F, "Solutions for Key Tables for CCIE Study," on the CD to check your answers.

Definitions

Next, take a few moments to write down the definitions for the following terms:

> policy routing, process switching, CEF, polarization, MLS, ARP, Proxy ARP, routed interface, fast switching, TTL, RIB, FIB, adjacency table, control plane, switched interface, data plane, IP routing, IP forwarding

Refer to the glossary to check your answers.

Further Reading

For a great overview of router switching paths, refer to www.cisco.com/en/US/tech/tk827/tk831/technologies_white_paper09186a00800a62d9.shtml.

For a good reference on load balancing with CEF, refer to http://cisco.com/en/US/tech/tk827/tk831/technologies_tech_note09186a0080094806.shtml.

Details on implementing and troubleshooting static routing can be found in numerous documents on the Cisco website. Recommended documents include "Specifying a Next Hop IP Address for Static Routes" (Document ID 27082), "Route Selection in Cisco Routers" (Document ID 8651), and "IOS Configuration Guide," in particular, the "IP Routing: Protocol-Independent Configuration Guide" section.

Index

Numerics

A

B

C

D

E

F

G

H

I

J-K

L

M

P

Q

R

S

T

U

V

W

X-Y-Z

This page intentionally left blank