Cisco Router Configuration Handbook

Second Edition

The single-source guide to configuring the most popular Cisco router features

David Hucaby, CCIE® No. 4594
Steve McQuerry, CCIE No. 6108
Andrew Whitaker

ciscopress.com

Cisco Router Configuration Handbook

Second Edition

Dave Hucaby, CCIE No. 4594
Steve McQuerry, CCIE No. 6108
Andrew Whitaker

Cisco Press
800 East 96th Street
Indianapolis, IN 46240

Cisco Router Configuration Handbook, Second Edition

Dave Hucaby, Steve McQuerry, Andrew Whitaker

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

Printed in the United States of America

Second Printing August 2012

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

ISBN-13: 978-1-58714-116-4

ISBN-10: 1-58714-116-7

Warning and Disclaimer

This book is designed to provide information about configuring Cisco routers. Every effort has been made to make this book as complete and as accurate as possible, but no warranty or fitness is implied.

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

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

Trademark Acknowledgments

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

Corporate and Government Sales

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

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

Feedback Information

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

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

We greatly appreciate your assistance.

Publisher: Paul Boger

Associate Publisher: Dave Dusthimer

Executive Editor: Brett Bartow

Managing Editor: Sandra Schroeder

Copy Editor: Apostrophe Editing Services

Editorial Assistant: Vanessa Evans

Book Designer: Louisa Adair

Composition: Mark Shirar

Manager, Global Certification: Erik Ullanderson

Business Operation Manager, Cisco Press: Anand Sundaram

Senior Development Editor: Christopher Cleveland

Project Editor: Seth Kerney

Technical Editors: Steve Kalman, Joe Harris

Indexer: WordWise Publishing Services

Proofreaders: Sheri Cain and Water Crest Publishing

Americas Headquarters
Cisco Systems, Inc.
San Jose, CA

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

Europe Headquarters
Cisco Systems International BV
Amsterdam, The Netherlands

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

Dedications

Dave Hucaby: This book is dedicated to my wife, Marci, and my daughters, Lauren and Kara. I am blessed to have three wonderful girls in the house; their love, encouragement, and support carry me along. God is good!

Steve McQuerry: I dedicate this work to my beautiful wife and love of my life, Becky. Also, to my wonderful children, Katie, Logan, and Cameron. You are all my inspiration. Your patience, love, and support give me the courage and strength needed to spend the required time and energy on a project like this. Even through the long hours, I want you to know I love you all very much.

About the Authors

David Hucaby, CCIE #4594, is a lead network engineer for the University of Kentucky, where he designs, implements, and maintains campus networks using Cisco products. Prior to his current position, he was a senior network consultant, providing design and implementation consulting, focusing on Cisco-based VPN and IP telephony solutions. Hucaby has bachelor of science and master of science degrees in electrical engineering from the University of Kentucky. He is also the author of *CCNP Switching Exam Certification Guide* by Cisco Press.

Stephen McQuerry, CCIE #6108, is an instructor and consultant with more than ten years of networking industry experience. He is a certified Cisco Systems instructor (CCSI) and a course director/developer, teaching routing and switching concepts for Global Knowledge. McQuerry regularly teaches Cisco Enterprise courses. Additionally, he has developed and taught custom Cisco switching courses. McQuerry holds a bachelor of science degree in engineering physics from Eastern Kentucky University. He is also the author of *Interconnecting Cisco Network Devices* by Cisco Press.

Andrew Whitaker has been teaching and developing Cisco courses for more than seven years and holds the following certifications: CCNP, CCVP, CCSP, CCDP, CCNA:Security, MCT, CEI, CISSP, LPT, CEH, ECSA, MCTS, MCSE, CNE, A+, Network+, Security+, Convergence+, CTP, CICP, CHFI, EMCPA. He is the author of several books, including *Penetration Testing and Network Defense* by Cisco Press.

About the Technical Reviewers

Steven Kalman is the principal officer at Esquire Micro Consultants, which offers lecturing, writing, and consulting services. He has more than 30 years of experience in data processing, with strengths in network design and implementation. Kalman is an instructor and author for Learning Tree International. He has written and reviewed many networking-related titles. He holds CCNA, CCDA, ECNE, CEN, and CNI certifications.

Joe Harris, CCIE No. 6200 (R/S, Security & SP), is a Triple CCIE working for Cisco as a systems engineer within the Wireline and Emerging Providers organization, where he specializes in security and MPLS-related technologies. With more than 16 years of extensive experience focusing on advance technologies within the IT arena, Joe has been primarily focused on supporting various enterprise-sized networks revolving around all aspects of Cisco technology. He has also provided high-end consulting for both large and small corporations, as well as local government and federal agencies. Joe holds a bachelor of science degree from Louisiana Tech University and resides with his wife and two children in Frisco, Texas.

Acknowledgments

Dave Hucaby: I am very grateful for another opportunity to work on a Cisco Press project. Getting to dabble in technical writing has been great fun, a highlight in my career, and a lot of work, too! Naturally, these good folks at Cisco Press have gone the extra mile to make writing enjoyable and achievable: Brett Bartow, who kindly accepted my idea for a book like this and kindly prodded us along to meet deadlines we didn't think we could, and Chris Cleveland, who is a superb development editor. As a matter of fact, every Cisco Press person I have met along the way has been so nice, encouraging, and excited about their work!

Thanks to our technical reviewers: Steve Kalman and Joe Harris. Working on a book of this nature has been challenging. The sheer volume and scope of the Cisco IOS Software commands and features are a little overwhelming. I truly appreciate reviewers who can help us see a bigger picture of better organization and accuracy while we're writing in the depths of configuration commands. This book is also a testimony to the great number of things you can do with a router, thanks to the Cisco IOS Software. I don't know how many hundreds of commands we have covered in this book, but we had to leave out many more lesser-used commands just to keep a handle on the book's size and scope. I'm amazed at the robustness of the software and its dynamic nature.

I would like to express my thanks to my friend and coauthor Steve McQuerry. We've followed each other around for many years, and it has been great to work on this project with him. Hopefully, we Kentucky boys can work on more things like this.

Lastly, I would like to acknowledge the person who stole my laptop computer halfway through the first edition of this book project. Whoever you are, you left me a victim of my own lack of current backups. I made up a silly joke many years ago: "A backup is worth a million bytes, especially if you have to type them all back in." Indeed.

Steve McQuerry: About 20 years ago, the late Rodger Yockey gave me an opportunity as a field engineer in the computer industry. Since then, several people have been there at key moments to help my career go in certain directions. I owe a great debt to these people, as they have helped me reach the level I am at today. It is not often that one has the opportunity to thank those who have been instrumental in molding his career. In addition to Rodger, I would like to take a moment to also thank Ted Banner for his guidance and mentoring. I would also like to thank Chuck Terrien for giving me the opportunity to work as an instructor in the Cisco product line. I would like to thank Brett Bartow for the opportunity to begin sharing my experiences with the network community by writing for Cisco Press. Last but not least, I have to thank my friend and coauthor, Dave Hucaby. This book was his concept, and I thank him for the opportunity work with him once again. I hope we will always find a way to continue working together in the future.

Since I began working on book and course projects a couple of years ago, I have a newfound respect for what it takes to edit, coordinate, publish, and basically keep authors on track. Behind every Cisco Press book is an incredible staff, and I would be remiss if I did not acknowledge their work. Chris Cleveland, again it has been great working with you. I hope that we can work together again in the future.

Without the following individuals behind the book, it would be no more than a collection of jumbled notes and napkin sketches of networking configurations:

The sharp eyes of all our technical editors on the first and this edition: Joe Harris, Steve Kalman, Alexander Marhold, and Kevin Turek.

All my students and fellow instructors at Global Knowledge. Your challenges and questions provide me with the drive to have a better understanding.

My wife and children for their never-ending patience and understanding during this and all of my projects.

Most important, God, for giving me the skills, talents, and opportunity to work in such a challenging and exciting profession.

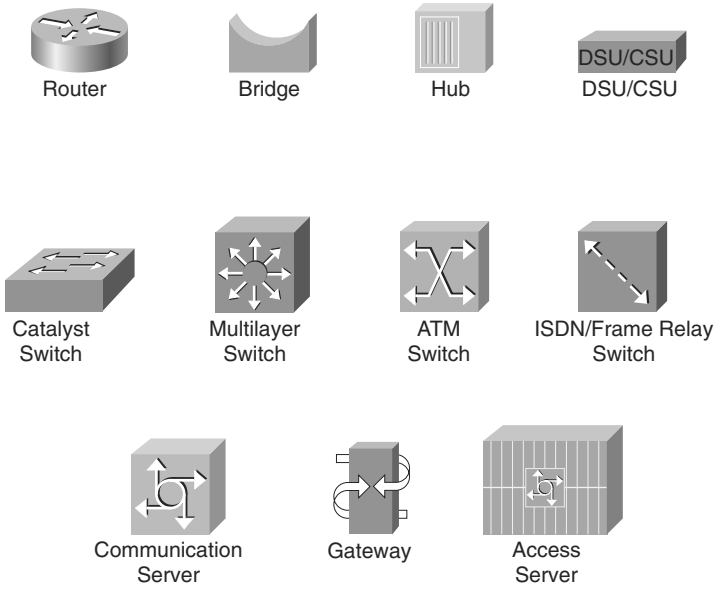
Andrew Whitaker: I would like to express my thanks to both Dave Hucaby and Steve McQuerry for this opportunity. Brett Bartow and Chris Cleveland, it is great to work with both of you again. Finally, to Steve Kalman and Joe Harris, I appreciate how diligently you worked to ensure a quality book.

Contents at a Glance

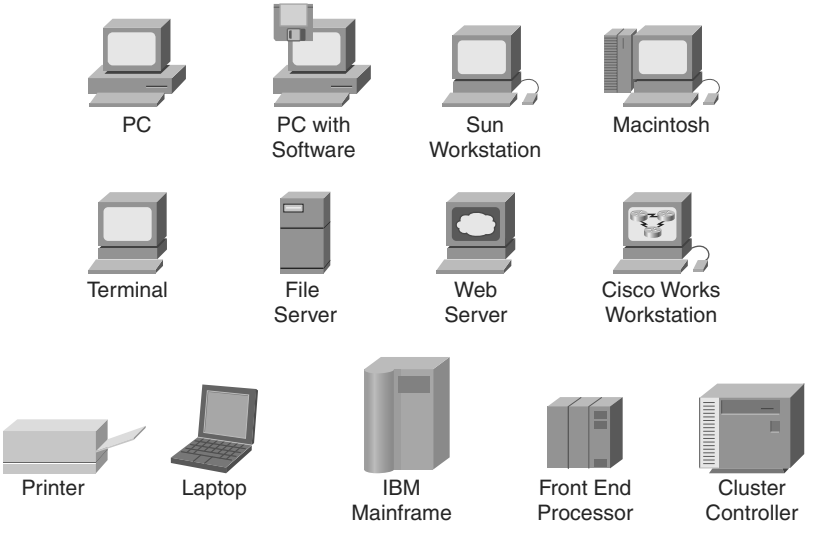
Contents

Icons Used in This Book

Throughout this book, you see the following icons used for networking devices:



The following icons are used for peripherals and other devices:



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

There are many sources of information and documentation for configuring Cisco networking devices, but few provide a quick and portable solution for networking professionals. This book is designed to provide a quick-and-easy reference guide for a wide range of commonly used features that can be configured on Cisco routers. In essence, the subject matter from an entire bookshelf of Cisco IOS Software documentation, along with other networking reference material, has been "squashed" into one handy volume that you can take with you.

This idea for this book began with my study habits for the CCIE written and lab exam. Over time, I found that I had put together a whole notebook of handwritten notes about how to configure a variety of Cisco router features. I also found that I began carrying this notebook with me into the field as a network consultant. When you're on the job and someone requires you to configure a feature you're not too familiar with, it's nice to have your handy reference notebook in your bag! Hopefully, this book will be that handy reference for you.

Features and Organization

This book is meant to be used as a tool in your day-to-day tasks as a network administrator or engineer. As such, we have avoided presenting a large amount of instructional informa-tion or theory on the operation of protocols or commands. That is better handled in other textbooks dedicated to a more limited subject matter.

Instead, this book is divided into parts that present quick facts, configuration steps, and explanations of configuration options for each feature in the Cisco IOS Software.

How to Use This Book

All the information in this book has been designed to follow a quick-reference format. If you know what feature or technology you want to use, you can turn right to the section that deals with it. Sections are numbered with a quick-reference index, showing both chapter and section number. For example, 13-3 is Chapter 13, Section 3. You'll also find shaded index tabs on each page, listing the section number, the chapter subject, and the topic dealt with in that section.

Facts About a Feature

Each section in a chapter includes a bulleted list of quick facts about the feature, technol-ogy, or protocol. Refer to these lists to quickly learn or review how the feature works. Immediately following, we have placed a note that details what protocol or port number the feature uses. If you are configuring filters or firewalls and you need to know how to al-low or block traffic from the feature, look for these notes.

Configuration Steps

Each feature covered in a section includes the required and optional commands used for common configuration. The difference is that the configuration steps are presented in an outline format. If you follow the outline, you can configure a complex feature or technol-ogy. If you find that you don't need a certain feature option, skip over that level in the out-line.

Sample Configurations

Each section includes an example of how to implement the commands and their options. We have tried to present the examples with the commands listed in the order you would actually enter them to follow the outline. Many times, it is more difficult to study and un-derstand a configuration example from an actual router, because the commands are dis-played in a predefined order, not in the order you entered them. Where possible, the ex-amples have also been trimmed to show only the commands presented in the section.

Further Reading

Each chapter ends with a recommended reading list to help you find more in-depth sources of information for the topics discussed.

Chapter 10

Multiprotocol Label Switching

This chapter covers the background and configuration of frame mode Multiprotocol Label Switching (MPLS). The following common configurations and features are discussed:

- **10-1: Configuring Basic MPLS—**MPLS is a high-performance packet forwarding technology that is based on Layer 2 switching instead of relying only on Layer 3 routing. Routers use labels added between the Layer 2 and Layer 3 headers for forwarding decisions.
- **10-2: MPLS Traffic Engineering—**Service providers can use traffic engineering on MPLS-enabled routers to route a customer's network traffic based on throughput and delay. Tunnels are created for label switch paths (LSP) using the Resource Reservation Protocol (RSVP).
- **10-3: MPLS Virtual Private Networks (VPN)—**MPLS VPNs have separate virtual routing and forwarding (VRF) instances for each customer in addition to a global routing table that is used to reach other routers in the provider network. Each VRF has a 64-bit route distinguisher (RD) to keep each customer's IP subnet separate from other routing and forwarding tables. BGP route target communities are used in exchanging route information between routers.

10-1: Configuring Basic MPLS

Multiprotocol Label Switching (MPLS) is a high-performance packet forwarding technology based on Layer 2 switching instead of Layer 3 routing. Routers are configured for MPLS forward frames based on labels instead of traditional Layer 3 IP unicast routing that performs Layer 3 lookups on the destination address at each hop. MPLS labels are inserted between the Layer 2 and Layer 3 headers and can be pushed (added), popped (removed), swapped, or aggregated (removing the top label and performing a Layer 3 lookup).

- MPLS is attractive to service providers because of its scalability and features. Providers can scale easier with MPLS than using Asynchronous Transfer Mode (ATM) or Frame Relay permanent virtual circuits (PVC). The traffic engineering features of MPLS enable providers to route traffic not just on a destination address but also on other factors such as bandwidth requirements and quality of service (QoS).
- A label switch router (LSR) is any router or switch that implements label distribution and can forward packets based on labels. An edge-LSR is any router that performs label imposition (push) or label disposition (pop). An ATM LSR is an ATM switch that can act as an LSR. With ATM LSRs, cell switching is used for the label forwarding table.
- MPLS-enabled routers communicate with each other using the Label Distribution Protocol (LDP). LSRs discover each other using hello packets, and peer relationships are maintained using keepalives. LSRs share label binding information with other LSRs. The label bindings build a forwarding equivalence class (FEC). A forwarding equivalence class is a group of IP packets forwarded in the same manner, over the same path, with the same forwarding treatment.
- Routers still rely on Layer 3 routing protocols for determining the path a packet should take. However, routing information, along with VPN, traffic engineering, and QoS information, is sent to the data plane to build a label forwarding information base (LFIB) for optimal forwarding performance.

Note The commands that follow are based on the IETF Label Distribution Protocol (LDP) that uses a different syntax for commands than the older proprietary Tag Distribution Protocol (TDP). Prior to Cisco IOS Software Release 12.4(2)T, LDP-related commands were saved in the configuration in the older syntax. Starting with Cisco IOS Software Release 12.4(2)T, commands are saved in the configuration as they are entered.

Configuration

1. (Required) Enable Cisco Express Forwarding (CEF):

```
(global) ip cef [distributed]
```

CEF must be enabled on all routers running MPLS and on interfaces receiving unlabeled IP packets. Core routers do not perform CEF switching but must have CEF enabled globally to exchange labels. Enter the **distributed** keyword if your router supports distributed CEF (dCEF). Use dCEF when you want your line cards (for example, VIP cards) to perform the express forwarding so that the route processor (RP) can handle routing protocols.

Note dCEF is not compatible with the RSVP. If you plan on configuring MPLS traffic engineering with RSVP, you must use CEF and not dCEF.

2. (Required) Start MPLS packet switching:

```
(global and interface) mpls ip
```

You must enable MPLS forwarding both globally and on the router interfaces for which you want to participate in MPLS forwarding. This command enables label switching of IPv4 packets according to normally routed paths. (Additional configuration is needed to support traffic engineering, QoS, and VPNs.) When this command is entered, Label Distribution Protocol (LDP) hello and keepalives are sent and received on the interfaces enabled for MPLS.

3. (Optional) Select the distribution protocol either globally or on a particular interface:

```
(global) mpls label protocol {ldp | tdp}
```

-or-

```
(interface) mpls label protocol {ldp | tdp | both}
```

Starting with Cisco IOS Software Release 12.4(3), the default protocol changed from Cisco Tag Distribution Protocol (TDP) to the IETF LDP. When changing the protocol on an interface, you have the option of enabling both TDP and LDP. LSRs must run the same distribution protocol to establish a peer session and exchange label information.

4. (Optional) Manually configure the LDP identifier:

```
(global) mpls ldp router-id interface [force]
```

MPLS-enabled routers identify themselves in LDP messages using an identifier. By default, the identifier is the highest IP address of all loopback interfaces. If there are no loopback interfaces, the router uses the highest IP address of all active interfaces. You can manually configure the LDP identifier with this command. The router then uses whatever IP address is configured on the interface you specify. The IP address entered must be reachable by adjacent LSRs. A common symptom of having an unreachable LDP ID IP address is that the forwarding information base (FIB) is populated, but there is no information in the label information forwarding base (LFIB). By default, a router ID is not changed until the interface currently used for the router ID is shut down, the IP address on that interface changes or is removed, or the router is rebooted. You can use the **force** keyword to force the router to change the router ID.

5. (Optional) Enable the distribution of labels associated with the IP default route:

```
(global) mpls ip default-route
```

By default, Cisco routers will not distribute labels for the IP default route. Enter this command to enable dynamic switching of labels for a router's default route.

6. (Optional) Enable MD5 authentication between peers:

```
(global) mpls ldp neighbor ip-address password password-string
```

MD5 authentication can be configured to verify TCP communication between two LDP peers. Both peers must be configured to use the same password.

7. (Optional) Enable the MPLS LDP autoconfiguration feature for OSPF interfaces:

```
(router) mpls ldp autoconfig [area area-id]
```

Normally, you must enter the **mpls ip** command both globally and on each interface for which you want to send and receive LDP packets. This process might be time-consuming and prone to human errors when configuring a router with many interfaces. The autoconfiguration feature helps with these issues by automatically enabling LDP on every interface associated with an OSPF or IS-IS instance. Note that you still need to enable LDP globally. When configuring LDP autoconfiguration for OSPF, you can choose to only enable LDP for interfaces belonging to a particular area by entering the **area** keyword followed by the area number.

8. (Optional) Enable the MPLS LDP autoconfiguration feature for IS-IS interfaces:

```
(router) mpls ldp autoconfig [level-1 | level-2]
```

Like autoconfiguration for OSPF, you must first enter the global **mpls ip** command before you can allow the autoconfiguration feature. Optionally, you can allow autoconfiguration on interfaces configured for level-1 or level-2 routing by entering the **level-1** or **level-2** keywords, respectively.

Example

Figure 10-1 is used for this example. The three routers are configured for MPLS with manually configured LDP identifiers and MD5 peer authentication. Additionally, RouterB is configured for OSPF autoconfiguration for Area 0, and LDP is disabled on RouterA's FastEthernet0/0 interface.

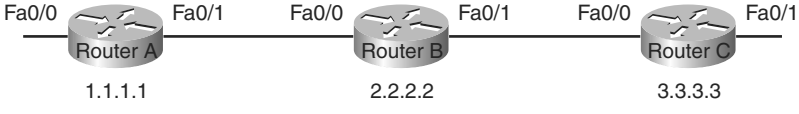


Figure 10-1 *Basic MPLS Configuration Example*

```
RouterA
hostname RouterA
!
ip cef
mpls ip
mpls ldp router-id 1.1.1.1
mpls ldp neighbor 2.2.2.2 password C1sc0
!
interface fastethernet0/0
 ip address 172.16.0.1 255.255.0.0
 no mpls ip
!
```

```
interface fastethernet0/1
 ip address 172.17.0.1 255.255.0.0
 mpls ip
!
interface loopback 0
 ip address 10.1.1.1 255.255.255.255
!
interface loopback 1
 ip address 1.1.1.1 255.255.255.255
!
router ospf 1
 router-id 10.1.1.1
 network 172.16.0.0 0.0.255.255 area 0
 network 172.17.0.0 0.0.255.255 area 0
 network 10.1.1.1 0.0.0.0 area 0
 network 1.1.1.1 0.0.0.0 area 0


RouterB
hostname RouterB
!
mpls ip
mpls ldp router-id 2.2.2.2
mpls ldp neighbor 1.1.1.1 password C1sc0
mpls ldp neighbor 3.3.3.3 password C1sc0
!
interface fastethernet0/0
 ip address 172.17.0.2 255.255.0.0
!
interface fastethernet0/1
 ip address 172.18.0.1 255.255.0.0
!
interface loopback0
 ip address 10.2.2.2 255.255.255.255
!
interface loopback1
 ip address 2.2.2.2 255.255.255.255
!
router ospf 1
 router-id 10.2.2.2
 network 172.17.0.0 0.0.255.255 area 0
 network 172.18.0.0 0.0.255.255 area 0
 network 10.2.2.2 0.0.0.0 area 0
 network 2.2.2.2 0.0.0.0 area 0
 mpls ldp autoconfig area 0
```

```
RouterC
hostname RouterC
!
mpls ip
mpls ldp router-id 3.3.3.3
mpls ldp neighbor 2.2.2.2 password C1sc0
!
interface fastethernet0/0
 ip address 172.18.0.2 255.255.0.0
 mpls ip
!
interface fastethernet0/1
 ip address 172.19.0.1 255.255.0.0
 mpls ip
!
interface loopback0
 ip address 10.3.3.3 255.255.255.255
!
interface loopback1
 ip address 3.3.3.3 255.255.255.255
!
router ospf 1
 router-id 10.3.3.3
 network 172.18.0.0 0.0.255.255 area 0
 network 172.19.0.0 0.0.255.255 area 0
 network 10.3.3.3 0.0.0.0 area 0
 network 3.3.3.3 0.0.0.0 area 0
```

10-2: MPLS Traffic Engineering

Service providers can use traffic engineering on MPLS-enabled routers to route a customer's network traffic based on throughput and delay. Traffic engineering (TE) tunnels are created for label switch paths (LSP) using the RSVP. A tunnel interface represents the head of the LSP and is configured with a set of resource requirements (for example, bandwidth requirements, media requirements, and priority):

- Topology and resource information is flooded using either IS-IS or OSPF. Traffic engineering extensions are added to the routing process to enable MPLS traffic engineering.
- IS-IS routers must be configured to use the new style of IS-IS metric to support new type, length, and value objects (TLV) for traffic engineering. The new TLVs are 22 and 135 and have sub-TLVs that enable you to add properties to a link for purposes of traffic engineering.

- The Shortest Path First (SPF) algorithm used by IS-IS and OSPF chooses the tunnel interface before choosing an alternative path over main interfaces.
- Multiple tunnels might be configured to load-share traffic.

Configuration

1. (Required) Allow CEF and LDP as explained in Section 10-1.
2. (Required) Allow the MPLS traffic engineering feature:

   ```
   (global) mpls traffic-eng tunnels
   ```

   Before you can allow interfaces for traffic engineering, you must first allow traffic engineering globally.
3. (Required) Allow traffic engineering on interfaces:

   ```
   (interface) mpls traffic-eng tunnels
   ```

   When you have allowed MPLS traffic engineering globally, you can then allow it on interfaces. Configuring this command causes its resource information to be flooded into the appropriate interior gateway protocol (IGP) link-state database (either IS-IS or OSPF). This command also enables the interface to accept traffic engineering tunnel signaling requests.
4. (Required when using CAC with RSVP) Allow RSVP on an interface and specify the amount of bandwidth to reserve:

   ```
   (interface) ip rsvp bandwidth [bandwidth]
   ```

   This command enables the resource reservation protocol (RSVP) on the interface. You must enter this command if you use Call Admission Control (CAC) with RSVP. If you do not specify a bandwidth value, a default bandwidth value of 75 percent will be used.
5. Configure OSPF for MPLS traffic engineering.

   a. (Required) Allow traffic engineering for each area:

      ```
      (router) mpls traffic-eng area area-number
      ```

   b. (Required) Set the traffic engineering router identifier:

      ```
      (router) mpls traffic-eng router-id interface
      ```

      Unlike OSPF and MPLS router IDs, MPLS traffic engineering will not automatically set a router ID. The router ID will be the IP address on the interface you enter in this command. It is recommended that you use a loopback interface and that it match that of the interface used for the OSPF process router ID. The router ID is flooded to other routers and is used as the tunnel destination IP address.

6. Configure IS-IS for MPLS traffic engineering.

   a. (Required) Allow traffic engineering:

      ```
      (router) mpls traffic-eng {level-1 | level-2}
      ```

      You can choose to flood traffic engineering into IS-IS level-1 or level-2.

   b. (Required) Set the traffic engineering router identifier:

      ```
      (router) mpls traffic-eng router-id interface
      ```

   c. (Required) Configure the router to generate and accept new-style TLVs:

      ```
      (router)metric-style wide
      ```

      The original metric used by IS-IS does not support traffic engineering. You must enter this command to support the new type, length, value objects (TLV), and sub-TLVs used for traffic engineering.

Note See RFC 3784, "Intermediate System to Intermediate System (IS-IS) Extensions for Traffic Engineering (TE)," for more on the new TLVs and sub-TLVs used for traffic engineering.

7. Create a tunnel interface:

   ```
   (global) interface tunnel number
   ```

   This command creates a new tunnel interface.

8. Assign an IP address to the tunnel interface:

   ```
   (interface) ip unnumbered interface
   ```

   Tunnel interfaces should be unnumbered because the tunnel interface represents a unidirectional link. It is recommended that you use a loopback interface for the *interface* value.

9. Set the tunnel destination IP address:

   ```
   (interface) tunnel destination ip-address
   ```

   The tunnel destination IP address should be set to the far-end router's IP address that was configured with the **mpls traffic-eng router-id** command under the IS-IS or OSPF router process.

10. Set the tunnel type to MPLS traffic engineering:

    ```
    (interface) tunnel mode mpls traffic-eng
    ```

    There are many types of tunnel interfaces (for example, GRE, 6to4, and so on), so you must manually set the interface type.

11. Configure the bandwidth for the tunnel:

    ```
    (interface) tunnel mpls traffic-eng bandwidth bandwidth
    ```

The range of valid *bandwidth* values is 1–4294967295.

12. Configure a dynamic or explicit path option:

```
(interface) tunnel mpls traffic-eng path-option number { dynamic | explicit
{name path-name | identifier path-number}}
```

You can configure the tunnel to be dynamic, meaning the router will rely on the IS-IS or OSPF routing information to determine the best path for the tunnel. You can also manually define the path using the **explicit** option. If you choose to explicitly map out the tunnel path, you need to enter either a name or identifier to reference the configured path. The steps for configuring an explicit path follow:

a. Configure the explicit path (if applicable).

- Enter IP explicit path configuration mode:

```
(global) ip explicit-path {name path-name | identifier path-number}
```

The *path-name* or *path-number* should match what you configured with the **tunnel mpls traffic-eng path-option** command earlier.

- Enter the IP addresses for each hop:

```
(explicit-path-configuration) next-address [loose | strict] ip-address
```

Enter the next IP addresses one at a time for the path you want the tunneled traffic to take. The **loose** and **strict** keywords are optional. The **loose** option tells the router that the previous address in the explicit path does not need to be directly connected to the next address. The router is therefore free to determine the path from the previous address to the next. The **strict** option tells the router that the previous address must be directly connected to the **next** address.

Note The next IP address can be either the link (interface) IP address of the next-hop router or the MPLS node address (set with the **mpls traffic-eng router-id** *interface* router process command). However, for Cisco IOS releases 12.4(24)T and earlier, you cannot use a link address as the first hop followed by node addresses for subsequent hops. You do not have this restriction for releases after 12.4(24).

13. Allow the IGP (that is, IS-IS or OSPF) to use the tunnel when performing its SPF route calculation:

```
(interface) tunnel mpls traffic-eng autoroute announce
```

The only way to forward traffic onto a tunnel is to allow this feature or to configure a static route to the interface.

14. (Optional) Manually configure the metric used by the SPF calculation:

```
(interface) tunnel mpls traffic-eng autoroute metric absolute metric
```

You can allow the IGP to automatically assign the metric for the tunnel or manually configure the metric value using this command.

Example

Figure 10-2 is used for this example. RouterA is configured for MPLS traffic engineering. An explicit tunnel is configured with a backup dynamic tunnel. The tunnel is configured with a metric of 10 and announced to IS-IS so that IS-IS will consider the tunnel when it performs its SPF calculation.

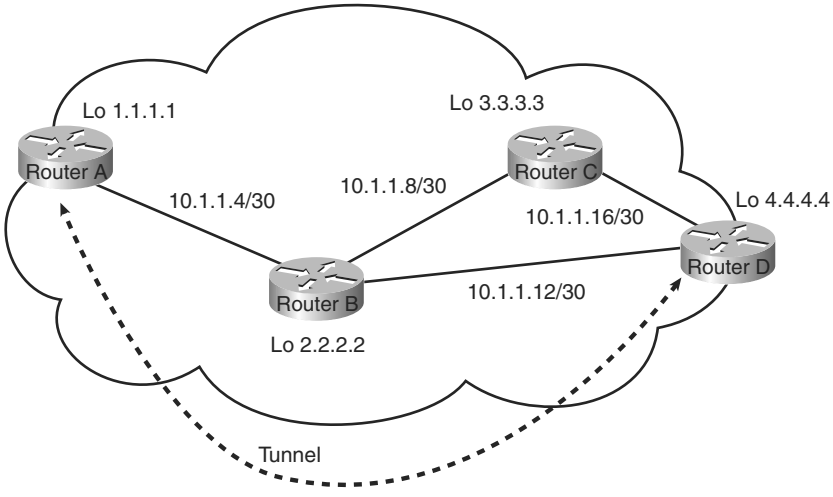


Figure 10-2 *MPLS Traffic Engineering Example*

```
hostname RouterA
!
ip cef
mpls ip
mpls ldp router-id 1.1.1.1
mpls traffic-eng tunnels
!
interface fastethernet0/0
 ip address 10.1.1.5 255.255.255.252
 mpls ip
 ip router isis
 mpls traffic-eng tunnels
 ip rsvp bandwidth 512
!
interface loopback 0
 ip address 1.1.1.1 255.255.255.255
```

```
 ip router isis
!
router isis
 net 49.0001.0000.0000.0001.00
 is-type level-1
 metric-style wide
 mpls traffic-eng router-id loopback 0
 mpls traffic-eng level-1
!
interface tunnel 0
 ip unnumbered loopback 0
 tunnel destination 4.4.4.4
 tunnel mpls traffic-eng bandwidth 512
 tunnel mode mpls traffic-eng
 tunnel mpls traffic-eng autoroute announce
 tunnel mpls traffic-eng autoroute metric absolute 10
 tunnel mpls traffic-eng path-option 1 explicit name TUNNEL
 tunnel mpls traffic-eng path-option 2 dynamic
!
ip explicit-path name TUNNEL
 next-address 2.2.2.2
 next-address 4.4.4.4
```


10-3: MPLS Virtual Private Networks (VPN)

- MPLS virtual private networks (VPN) offer greater scalability than Frame Relay or ATM overlay VPN solutions.
- MPLS VPNs have a separate routing and forwarding (VRF) instance for each customer.
- Each VRF has a 64-bit route distinguisher (RD) to keep each customer's IP subnet separate from other routing and forwarding tables.
- Routers maintain a global routing table that is used to reach other routers in the provider's network.
- BGP route target communities are used to exchange route information between routers.
- There is a separate CEF and routing table for each VRF.

Configuration

1. Configure MPLS according to Section 10-1.
2. Create the VRF and enter into VRF configuration mode:

   ```
   (global) ip vrf name
   ```

   Every customer will have their own VRF.

Table 10-1 *RD Formats*

Format	Example
16-bit autonomous system number: a 32-bit number	65501:1
32-bit IP address: a 16-bit number	192.168.0.1:1

3. Create a route distinguisher for the VRF:

```
(vrf) rd route-distinguisher
```

You must configure a route distinguisher for the VRF to be functional. This command adds an 8-byte value to an IPv4 prefix to create a VPN IPv4 prefix. The RD can be entered in one of the two formats shown in Table 10-1.

4. Create a route target for import, export, or both import and export:

```
(vrf) route-target {import | export | both} route-target-community-number
```

This command creates a list of import and export route target extended communities for the VRF. Learned routes that carry the same extended community number as the *route-target-community-number* you configure can be either imported into the VRF or exported out of the VRF (or both). Extended communities follow the same format as RDs.

5. Associate the VRF with an interface facing a customer edge (CE) router:

```
(interface) ip vrf forwarding vrf-name
```

This command associates the VRF instance you created earlier in Step 2 with an interface. Configuring this command removes the IP address so you need to reconfigure the IP address after applying this command.

6. Configure the VRF instance under the BGP process:

   a. Enter BGP configuration mode:

   ```
   (global) router bgp autonomous-system-number
   ```

   b. Enter the address family configuration mode for the VRF instance:

   ```
   (router) address-family ipv4 vrf vrf-name
   ```

   Enter the VRF name you created earlier in Step 2.

   c. Configure the customer edge neighbor:

   ```
   (address-family) neighbor ip-address remote-as autonomous-system-number
   ```

   This command goes on the provider edge (PE) router and associates the VRF created in Step 2 with the BGP neighbor defined here under the VRF address family mode.

d. Activate the CE BGP neighbor:

```
(address-family) neighbor ip-address activate
```

7. Configure the provider edge (PE) to provider edge (PE) routing:

a. Enter BGP configuration mode:

```
(global) router bgp autonomous-system-number
```

b. Configure the PE BGP neighbor:

```
(router) neighbor ip-address remote-as autonomous-system-number
```

There is a separate VRF instance for each customer and a global routing table. The global routing is created here by defining the BGP peers within the provider.

c. Activate the PE BGP neighbor:

```
(router) neighbor ip-address activate
```

d. Enter the vpn4 unicast address family:

```
(router) address-family vpnv4 unicast
```

This address family configures an IPv4 unicast VPN routing instance that enables the PE routers to exchange BGP information with each other while still remaining separate from the customers' VRF instances.

e. Define and activate the PE BGP neighbors:

```
(address-family) neighbor ip-address remote-as autonomous-system-number
(address-family) neighbor ip-address activate
```

f. Allow extended communities for the PE BGP neighbor:

```
(address-family) neighbor ip-address send-community extended
```

Configuring this command activates support for extended communities. Extended communities are necessary for route targets to work with VRFs.

Note For more on BGP configuration, see Chapter 6, "IP Routing Protocols."

Example

Figure 10-3 is used in this example. The configuration that follows shows the configuration on the PE1 router. A VRF instance is created for Customer_A and Customer_B. For Customer_A, the router is configured to both import and export all routes tagged with the extended community 100:1. For Customer_B, the router is configured to both import and export all routes tagged with the extended community 100:2:

```
hostname PE1
!
ip cef
!
```

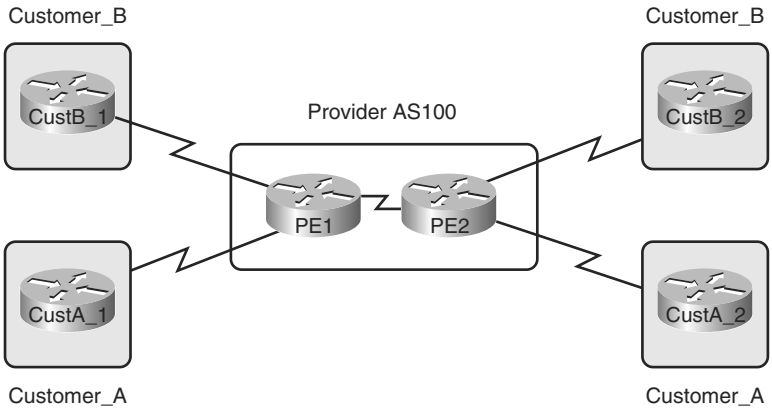


Figure 10-3 *MPLS VPN Example*

```
ip vrf Customer_A
 rd 100:1
 route-target both 100:1
!
ip vrf Customer_B
 rd 100:2
 route-target both 100:2
!
interface serial0/0
 ip address 192.168.1.5 255.255.255.252
 mpls ip
 description Link to PE
!
interface serial0/1
 ip address 192.168.1.9 255.255.255.252
 description Link to Customer_A
 ip vrf forwarding Customer_A
!
interface serial0/2
 ip address 192.168.1.13 255.255.255.252
 description Link to Customer_B
 ip vrf forwarding Customer_B
!
router bgp 100
 neighbor 192.168.1.6 remote-as 100
 neighbor 192.168.1.6 activate
 address-family vpnv4 unicast
 neighbor 192.168.1.6 remote-as 100
 neighbor 192.168.1.6 activate
 neighbor 192.168.1.6 send-community extended
```

```
address-family ipv4 unicast vrf Customer_A
redistribute connected
neighbor 192.168.1.10 remote-as 65535
neighbor 192.168.1.10 activate
address-family ipv4 unicast vrf Customer_B
redistribute connected
neighbor 192.168.1.14 remote-as 65534
neighbor 192.168.1.14 activate
```

References

Refer to the following recommended sources for further technical information about the MPLS topics covered here:

Definitive MPLS Network Designs, by Jim Guichard, Francois Le Faucheur, and Jean-Philippe Vasseur, Cisco Press, ISBN 1587051869.

MPLS Configuration on Cisco IOS Software, by Umesh Lakshman and Lancy Lobo, Cisco Press, ISBN 1587051990.

MPLS Fundamentals, by Luc DeGhein, Cisco Press, ISBN 1587051974.

MPLS and VPN Architectures, by Ivan Pepelnjak, Cisco Press, ISBN 1487050021.

MPLS and VPN Architectures, Volume II, by Ivan Pepelnjak, Jim Guichard, and Jeff Apcar, Cisco Press, ISBN 1587051125.

RFC 2547, "BGP/MPLS VPNs."

RFC 2702, "Requirements for Traffic Engineering Over MPLS."

RFC 2917, "A Core MPLS IP VPN Architecture."

RFC 3031, "Multiprotocol Label Switching Architecture."

RFC 3034, "Use of Label Switching on Frame Relay Networks Specification."

RFC 3036, "LDP Specification."

RFC 3063, "MPLS Loop Prevention Mechanism."

RFC 3032, "MPLS Label Stack Encoding."

RFC 3209, "RSVP-TE: Extensions to RSVP for LSP Tunnels."

RFC 3784, "Intermediate System to Intermediate System (IS-IS) Extensions for Traffic Engineering."

This page intentionally left blank

Index

Numerics

A

B

C

D

E

F

G

H

I

L

M

N

O

P

Q

R

S

T

U

V

W