

CISCO SYSTEMS

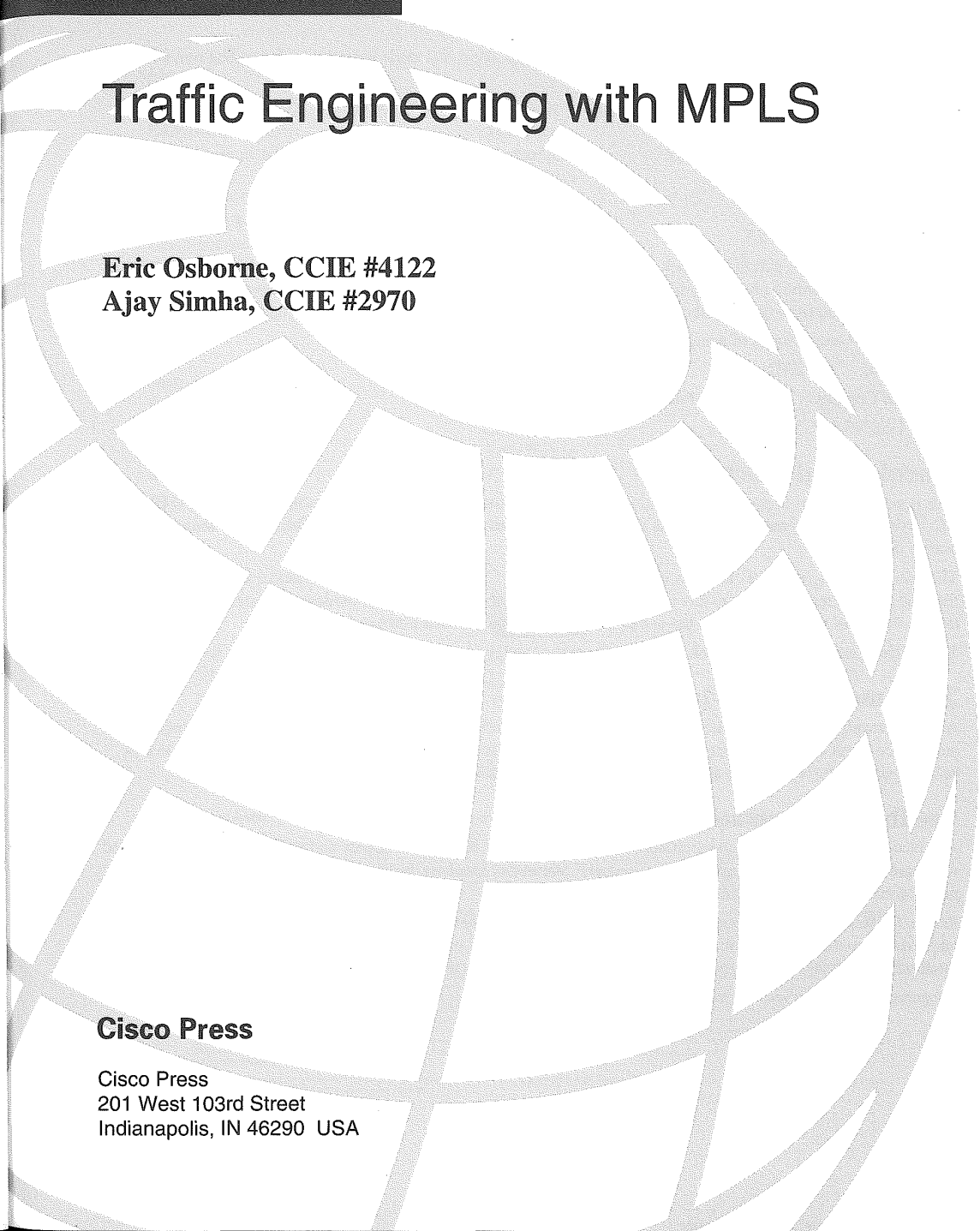


Traffic Engineering with MPLS

Design, configure, and manage MPLS TE
to optimize network performance

ciscopress.com

Eric Osborne, CCIE® #4122
Ajay Simha, CCIE #2970



Traffic Engineering with MPLS

Eric Osborne, CCIE #4122
Ajay Simha, CCIE #2970

Cisco Press

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

Traffic Engineering with MPLS

Eric Osborne

Ajay Simha

Copyright © 2003 Cisco Systems, Inc.

Published by:

Cisco Press

201 West 103rd Street

Indianapolis, IN 46290 USA

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

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

First Printing July 2002

Library of Congress Cataloging-in-Publication Number: 200-1086632

ISBN: 1-58705-031-5

Warning and Disclaimer

This book is designed to provide information about Multiprotocol Label Switching Traffic Engineering (MPLS TE). Every effort has been made to make this book as complete and as accurate as possible, but no warranty or fitness is implied.

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

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

Trademark Acknowledgments

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

Feedback Information

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

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

We greatly appreciate your assistance.

You can find additional information about this book, any errata, and Appendix B at www.ciscopress.com/1587050315.

Publisher	John Wait
Editor-In-Chief	John Kane
Cisco Systems Management	Michael Hakkert
	Tom Geitner
Executive Editor	Brett Bartow
Production Manager	Patrick Kanouse
Development Editor	Christopher Cleveland
Project Editor	Eric T. Schroeder
Copy Editor	Gayle Johnson
Technical Editors	Jim Guichard
	Alexander Marhold
	Jean-Philippe Vasseur
Team Coordinator	Tammi Ross
Book Designer	Gina Rexrode
Cover Designer	Louisa Klucznik
Compositor	Amy Parker
Indexer	Tim Wright



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

European Headquarters
Cisco Systems Europe
11 Rue Camille Desmoulins
92782 Issy-les-Moulineaux
Cedex 9
France
<http://www-europe.cisco.com>
Tel: 33 1 58 04 60 00
Fax: 33 1 58 04 61 00

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

Asia Pacific Headquarters
Cisco Systems Australia,
Pty., Ltd.
Level 17, 99 Walker Street
North Sydney
NSW 2059 Australia
<http://www.cisco.com>
Tel: +61 2 8448 7100
Fax: +61 2 9957 4350

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

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

Copyright © 2000, Cisco Systems, Inc. All rights reserved. Access Registrar, AccessPath, Are You Ready, ATM Director, Browse with Me, CCDA, CCDE, CCDP, CCIE, CCNA, CCNP, CCSI, CD-PAC, CiscoLink, the Cisco NetWorks logo, the Cisco Powered Network logo, Cisco Systems Networking Academy, Fast Step, FireRunner, Follow Me Browsing, FormShare, GigaStack, IQX, Intelligence in the Optical Core, Internet Quotient, IP/VC, IQ Breakthrough, IQ Expertise, IQ FastTrack, iQuick Study, IQ Readiness Scorecard, The IQ Logo, Kernel Proxy, MGX, Natural Network Viewer, Network Registrar, the Networkers logo, Packer, PIX, Point and Click Internetworking, Policy Builder, RateMUX, ReyMaster, ReyView, ScriptShare, Secure Script, Shop with Me, SlideCast, SMARTnet, SVX, TrafficDirector, TransPath, VlanDirector, Voice LAN, Wavelength Router, Workgroup Director, and Workgroup Stack are trademarks of Cisco Systems, Inc.; Changing the Way We Work, Live, Play, and Learn, Empowering the Internet Generation, are service marks of Cisco Systems, Inc.; and Aironet, ASIST, BFX, Catalyst, Cisco, the Cisco Certified Internetwork Expert logo, Cisco IOS, the Cisco IOS logo, Cisco Press, Cisco Systems, Cisco Systems Capital, the Cisco Systems logo, Collision Free, Enterprise/Solver, EtherChannel, EtherSwitch, FastHub, FastLink, FastPAD, IOS, IP/TV, IPX, LightStream, LightSwitch, MICA, NetRanger, Post-Routing, Pre-Routing, Registrar, StrainView Plus, Strain, SwitchProbe, TeleRouter, are registered trademarks of Cisco Systems, Inc. or its affiliates in the U.S. and certain other countries.

All other brands, names, or trademarks mentioned in this document or Web site are the property of their respective owners. The use of the word partner does not imply a partnership relationship between Cisco and any other company. (0010R)

About the Authors

Eric Osborne, CCIE #4122, has been doing Internet engineering of one sort or another since 1995. He's seen fire, he's seen rain, he's seen sunny days that he thought would never end. He joined Cisco in 1998 to work in the Cisco TAC, moved to the ISP Expert Team shortly after Ajay, and has been involved in MPLS since the Cisco IOS Software Release 11.1CT days. His BS degree is in psychology, which, surprisingly, is often more useful than you might think. He is a frequent speaker at Cisco's Networkers events in North America, having delivered the "Deploying MPLS Traffic Engineering" talk since 2000. He can be reached at eosborne@cisco.com.

Ajay Simha, CCIE #2970, graduated with a BS in computer engineering in India, followed by a MS in computer science. He joined the Cisco Technical Assistance Center in 1996 after working as a data communication software developer for six years. He then went on to support Tier 1 and 2 ISPs as part of Cisco's ISP Expert Team. His first exposure to MPLS TE was in early 1999. It generated enough interest for him to work with MPLS full-time. Simha has been working as an MPLS deployment engineer since October 1999. He has first-hand experience in troubleshooting, designing, and deploying MPLS. He can be reached at asimha@cisco.com.

About the Technical Reviewers

Jim Guichard, CCIE #2069, is an MPLS deployment engineer at Cisco Systems. In recent years at Cisco, he has been involved in the design, implementation, and planning of many large-scale WAN and LAN networks. His breadth of industry knowledge, hands-on experience, and understanding of complex internetworking architectures have enabled him to provide a detailed insight into the new world of MPLS and its deployment. He can be reached at jguichar@cisco.com.

Alexander Marhold, CCIE #3324, holds an MSC degree in industrial electronics and an MBA. He works as a senior consultant and leader of the Core IP Services team at PROIN, a leading European training and consulting company focused on service provider networks. His focus areas are core technologies such as MPLS, high-level routing, BGP, network design, and implementation. In addition to his role as a consultant, Marhold is also a CCSI who develops and holds specialized training courses in his area of specialization. His previous working experience also includes teaching at a polytechnic university for telecommunications, as well as working as CIM project manager in the chemical industry.

Jean-Philippe Vasseur has a French engineer degree and a master of science from the SIT (New Jersey, USA). He has ten years of experience in telecommunications and network technologies and worked for several service providers prior to joining Cisco. After two years with the EMEA technical consulting group, focusing on IP/MPLS routing, VPN, Traffic Engineering, and GMPLS designs for the service providers, he joined the Cisco engineering team. The author is also participating in several IETF drafts.

Protection and Restoration

Networks fail. More precisely, pieces of networks fail. Lots of things can cause something to fail in a network. They run the gamut from loosely connected cables to fiber cuts. Router crashes are another form of failure.

From a router's perspective, there are two kinds of failures in a network—link failures and node failures. It doesn't matter what the underlying cause is. A link failure can be a fiber cut, an ADM problem, or any number of other things. A node failure can be anything from a power problem to a router crash to a router being taken down for scheduled maintenance. No matter what the cause, all failures are either a link failure or a node failure.

It is highly desirable to reduce the negative effects of such failures, such as packet loss. As it turns out, MPLS TE and its ability to steer traffic away from the IGP-derived shortest path helps mitigate packet loss associated with link or node failures in the network. MPLS TE's ability to do this is known as Fast Reroute (FRR) or simply MPLS TE Protection.

This chapter covers the aspects of MPLS TE that deal with fast recovery from failures—what they are, how they work, and how you can configure them using Cisco IOS Software.

The Need for Fast Reroute

Network administrators have been dealing with link and node failures for as long as there have been networks. It has traditionally fallen to the Interior Gateway Protocol (IGP) to quickly route around failures, converging on the remaining topology. However, there are a few things the IGP doesn't do that well when it comes to convergence:

- In a large network, your IGP can take quite a few seconds to converge; until the entire network is converged, there is packet loss. It's not uncommon to see 5 to 10 seconds of packet loss when a core link flaps in a large network.
- A link failure can lead to congestion in some parts of the network while leaving other parts free of congestion.
- Configuring the IGP to converge quickly can make it overly sensitive to minor packet loss, causing false negatives and IGP convergence for no reason.

Also, assuming that the IGP is a link-state protocol, SPF has to be run once when the link goes down and then again when it comes back up. This problem is exacerbated with MPLS TE: If a link that is a part of an LSP fails, the LSP is torn down. After the headend recomputes a new path, SPF has to be run again for prefixes routed over the tunnel when autoroute is in place, thus making convergence times even worse than in a pure IP network.

IP networks that use SONET can also employ automatic protection switching (APS) to aid in quick recovery from link failures. The goal of APS is to switch over from an active link to a standby link within 50 milliseconds upon failure of the active link. However, if APS is run directly on a router, even after APS switches traffic over to the standby link, the IGP still needs to converge with the new neighbor on the other end of the link. Until the new IGP neighbors come up, packets might still be dropped.

APS also does not come without additional cost—the hardware cost of the add/drop multiplexer (ADM) required to achieve APS.

Luckily, there's an alternative to all this. You can use MPLS TE's FRR capabilities to minimize packet loss, without all the drawbacks of APS or fast IGP convergence.

RFC 2702, "Requirements for Traffic Engineering Over MPLS," describes the "resilience attribute" as the behavior of a traffic trunk under fault conditions:

A basic resilience attribute indicates the recovery procedure to be applied to traffic trunks whose paths are impacted by faults.

This is called *headend LSP reroute* or simply *headend reroute*.

At its simplest, headend rerouting is calculating a new path for an LSP after its existing path goes down. However, during the time required to perform this basic reroute, there can be significant traffic loss; the packet loss is potentially worse than with regular IP routing if you are autorouting over the TE tunnel. This is because you first need to signal a new TE LSP through RSVP and run SPF for destinations that need to be routed over the tunnel. It is desirable to be able to deal with a link or node failure in a way that has less loss than the basic headend LSP reroute.

Normally, when a link or node fails, this failure is signalled to the headends that had LSPs going through the failed link or node. The headends affected attempt to find new paths across the network for these tunnels.

Although a few seconds of loss is generally acceptable for data traffic, real-time applications such as voice, video, and some legacy applications might not be so forgiving. Many attempts have been made and drafts submitted to the IETF to solve this problem.

Within Cisco, the question arose as to how to make MPLS TE tunnels resistant to failures within the network. Mechanisms were developed to address this question, and they are collectively known as FRR, fast restoration mechanisms, or simply protection. Although it's impossible to have a completely lossless failure recovery mechanism, it's certainly possible to have mechanisms that minimize loss as much as possible.

Generally speaking, the goal for the different FRR mechanisms is to achieve as little packet loss as possible. Practically, this translates into anything from SONET-like recovery times (50 ms or less) to a few hundred milliseconds of loss before FRR kicks in.

What Is Protection?

Protection, in the context of fast restoration, is having procedures in place that, when applied to selected resources, ensures minimal loss of traffic upon failure. Protected resources could either be viewed as physical resources (link or nodes) or logical resources (the LSPs that cross a link or node). Regardless of how you look at it, network failures are always physical in origin—a link or a node goes down, taking some LSPs with it. What protection boils down to is protection of logical resources (LSPs) from physical failures (links or nodes).

For all subsequent discussions in this chapter (or, for that matter, anywhere in MPLS), the term *protection* should be associated with the fact that backup resources are *preestablished* and are not signalled after a failure has occurred, as you'd see in the case of headend LSP reroute. The preestablishment of protection resources is fundamental for any protection strategy. If protection resources weren't preestablished, they'd have to be set up after the failure was detected; by then, it's too late.

This chapter calls this preestablished LSP a *backup tunnel* or *protection tunnel*. They mean the same thing.

NOTE

Many people have asked, "When I have multiple path-option statements under my tunnel configuration, and a link/node failure causes my headend to pick the next path option in the list, isn't this protection?" No; no backup resources are precomputed or signalled before failure. Configuring multiple path options is merely a way to influence basic LSP rerouting. Unless the backup resources are signalled *before* any failure, there can be no fast protection.

Types of Protection

This section provides an overview of the three different types of protection schemes.

Protection can be broken into

- Path protection (sometimes called end-to-end protection)
- Local protection, which can be broken into two types:
 - Link protection
 - Node protection

Path Protection

Path protection is not currently available on Cisco routers. It is provided here for the sake of completeness and is not discussed further in this chapter. It is, however, discussed again in Chapter 9, "Network Design with MPLS TE," where the scalability of path protection is compared with that of link and node protection.

Path protection is essentially the establishment of an additional LSP in parallel with an existing LSP, where the additional LSP is used only in case of failure. This LSP is sometimes called the *backup*, *secondary*, or *standby* LSP. The backup LSP is not used to carry traffic except during a failure condition—hence, the term *standby*.

The backup LSP is built along paths that are as diverse as possible from the LSP they're protecting. This ensures that a failure along the path of the primary LSP does not also affect the backup LSP. Path protection is simple in concept. Each primary LSP is backed up by a standby LSP. Both the primary and backup LSPs are configured at the headend. Both are signalled ahead of time in the control plane.

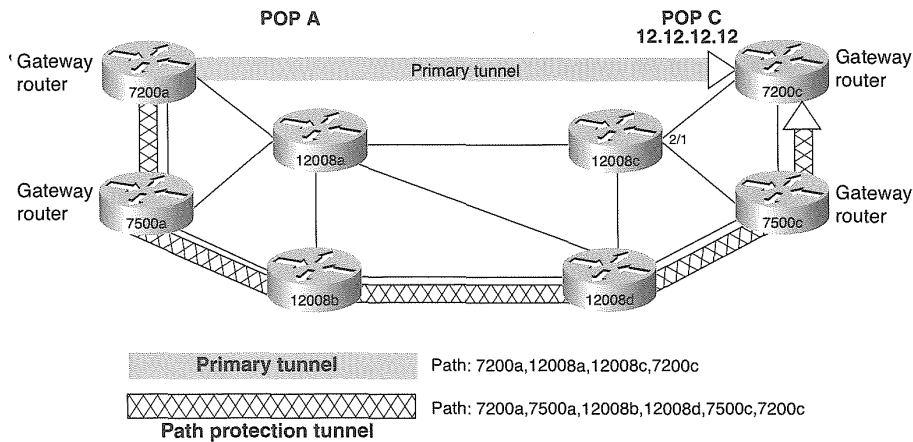
The primary and backup LSPs might have the same constraints. If the primary LSP has a bandwidth reservation of 100 Mbps, the backup LSP can also reserve 100 Mbps. This way, the end-to-end characteristics essentially remain the same, no matter whether the LSP used to carry traffic is the primary LSP or the protection LSP.

As mentioned in the preceding section, simply having a second path option under the tunnel interface does not make it path protection—it would be an LSP reroute. Path protection has better convergence than IGP convergence in an IP network or MPLS TE LSP reroute because it makes use of a presignalled LSP that is ready to go in case the primary LSP fails. With path protection, the relationship between the backup LSP and the number of primary LSPs it is protecting is 1:1. This makes the path protection scheme less scalable.

In other words, for every LSP you want to protect, you have to signal another LSP. If you want the primary and backup LSPs to share the same bandwidth characteristics, they need to reserve the same amount of bandwidth. Protection LSPs kick in only when there's a failure, and hopefully your network failure rate is far less than 50 percent, so you end up reserving backup bandwidth that you won't use most of the time and keeping other LSPs in the network from being able to use that bandwidth.

Figure 7-1 shows a primary tunnel going from 7200a to 7200c over 12008a and 12008c. The path protection tunnel is also from the headend 7200a to tail 7200c but goes over a diverse path (7200a→7500a→12008b→12008d→7500c→7200c).

Figure 7-1 Path Protection



Local Protection

Local protection is the term used when the backup or protection tunnel is built to cover only a segment of the primary LSP. Local protection, like path protection, requires the backup LSP to be presignalled.

In local protection, the backup LSP is routed around a failed link (in link protection) or node (in node protection), and primary LSPs that would have gone through that failed link or node are instead encapsulated in the backup LSP. Local protection has several advantages over path protection—faster failure recovery, 1:N scalability, and the consumption of less network state, to name a few.

Local Protection Terminology

Cisco's implementation of local protection is based on the bypass section of the IETF *draft-ietf-mpls-rsvp-lsp-fastreroute*, "Fast Reroute Extensions to RSVP-TE for LSP Tunnels," which might be an RFC by now.

Unlike path protection, for local protection, the relationship between the backup LSP and the number of primary LSPs it is protecting is 1:N. In other words, a single backup LSP can protect N primary LSPs, making it more scalable than path protection. This scalability makes the local protection scheme extremely attractive.

Before going any further into local protection, it is important to stop and understand some of the terminology. Figure 7-2 shows 12008a as the headend as of two backup tunnels—tunnel0, which terminates on 12008c, and tunnel1, which terminates on 7200c. Table 7-1 defines some of the commonly used terms.

Figure 7-2 Elements of Local Protection

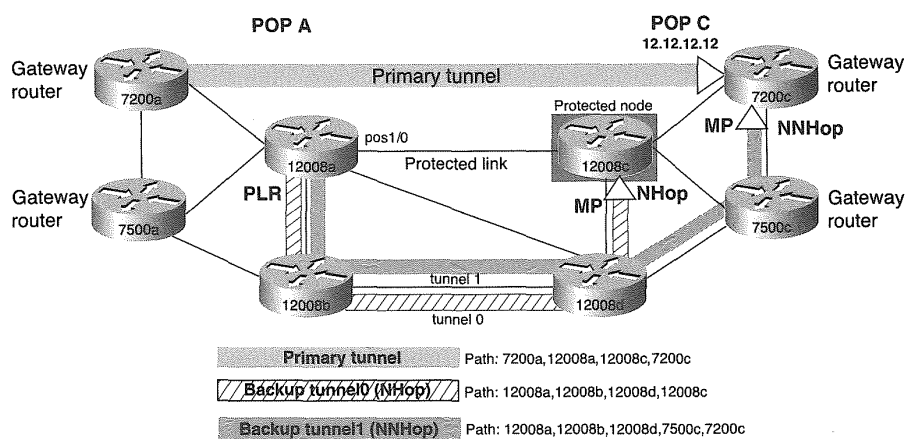


Table 7-1 Local Protection Terminology

Term	Definition
PLR	Point of Local Repair—The headend of the backup tunnel. In Figure 7-2, 12008a is the PLR.
MP	Merge Point—The merge point is where the backup tunnel terminates. In Figure 7-2, the merge point for tunnel0 is 12008c, and the merge point for tunnel1 is 7200c.
NHop	Next-hop router—A router that is one hop away from the PLR. This bypasses the protected link (12008a→12008c). 12008c in Figure 7-2 is the NHop for PLR 12008a.
NNHop	Next-next-hop router—A router that is two hops away from the PLR. 7200c is the NNHop for PLR 12008a in Figure 7-2. The NNHop bypasses a protected node (12008c).

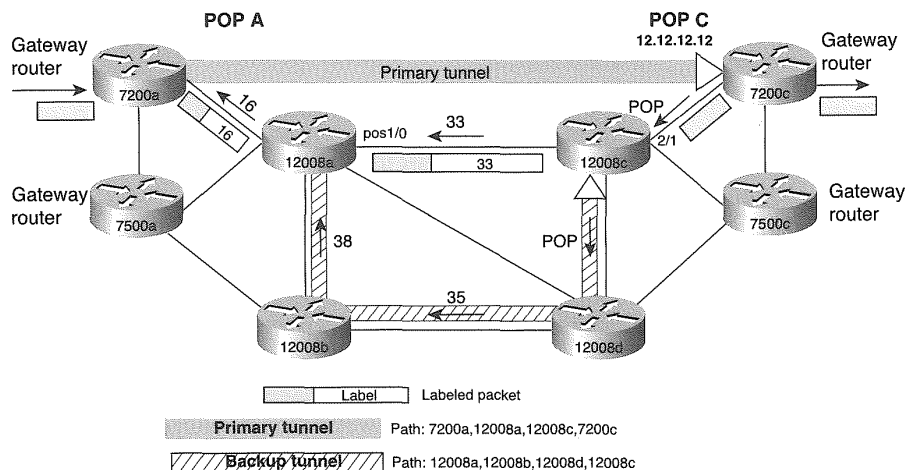
In addition to the terms “backup tunnel” and “protection tunnel,” you might see the terms “FRR tunnel” and “bypass tunnel” being used to refer to this presignalled tunnel. They all mean the same thing.

NOTE

The material presented so far might give you the impression that the primary tunnel headend and the PLR have to be two distinct things. This is not necessarily true in every case, even if it is the common situation. You might have configured link protection, protecting the link between the primary tunnel headend and its downstream neighbor. In this case, the primary tunnel headend is also the PLR. Basically, the PLR is where the backup tunnel begins.

The Need for Label Stacking

Consider Figure 7-3. It has two tunnels: *primary* and *backup*. The primary tunnel goes from 7200a to 7200c, whereas the backup tunnel goes from 12008a to 12008c, protecting link 12008a→12008c. Labels are distributed downstream to upstream (from 7200c toward 7200a). For the primary tunnel, routers 7200a, 12008a, and 12008c receive labels 16, 33, and POP (implicit null), respectively, in RSVP Resv messages from their downstream neighbors. Similarly for the backup tunnel, 12008a receives label 38 from 12008b. 12008b receives label 35 from 12008d, and 12008d receives the POP label from 12008c.

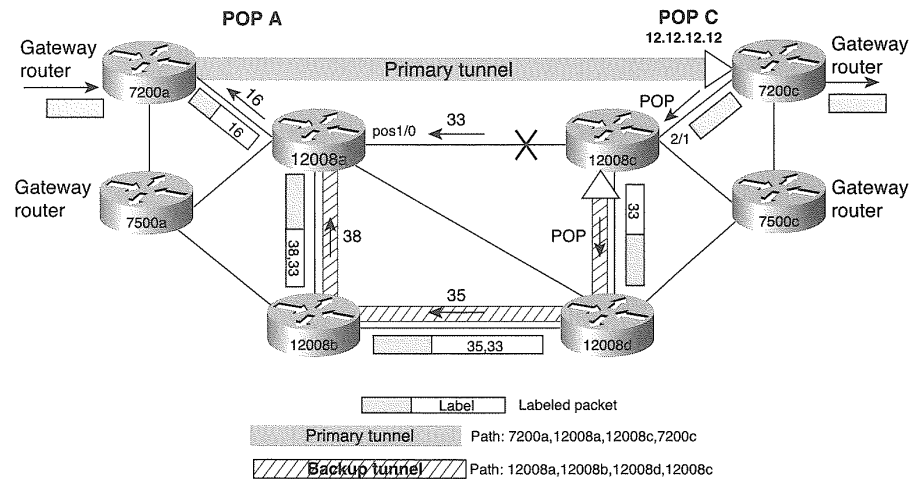
Figure 7-3 Label Forwarding Over the Primary Tunnel

In the forwarding plane, when the ingress router 7200a forwards a packet down the primary tunnel, it imposes a label of 16 and transmits the packet to 12008a. 12008a label-swaps 16 to 33 and sends it to 12008c. Because 12008c performs Penultimate Hop Popping (PHP), the top label 33 is removed, and the packet is sent to 7200c, which is the egress router.

If you now focus your attention on the link 12008a→12008c (the protected link), you'll notice that primary tunnel packets that are carried on this link have a label value of 33 as they enter 12008c.

Now consider Figure 7-4.

Figure 7-4 Label Stacking Over the Backup Tunnel After Failure of Protected Link



Before the protected link failed, the primary tunnel traffic entered 12008a with a label of 33.

When the protected link fails, if the primary tunnel traffic entering 12008a somehow reached 12008c with a label of 33, it would be as though the protected link never failed. 12008c would simply POP label 33 and send the packet to 7200c. This is because all routers in this figure are allocating labels from global label space. One of the benefits of global label space is that a router does the same thing with a label no matter what interface it comes in on.

This is exactly what happens with local protection. Because the backup tunnel is presignalled and is ready to carry traffic to 12008c, when the protected link fails, 12008a switches the traffic onto the backup path. Because 12008c is expecting the packets to arrive with a label value of 33, 12008a takes the packets destined for 12008c with a label value of 33 and stacks another label, 38, on top.

Label 38 is what 12008a is expected to impose on any packets entering the backup tunnel. As you can see in Figure 7-4, the packet leaving 12008a toward 12008b has a label stack {38,33}. As you'd expect, 12008b switches the packet based on the top-level label, 38. The packet leaving 12008b now has a label stack of {35,33} because 12008b label swapped the top label, 38, with label 35. When this packet reaches 12008d, PHP is performed, removing

the top label. 12008d performs PHP for the backup tunnel. With the top label removed, the packet has a single label of 33 as it enters 12008c. From the labels exchanged for the primary tunnel, 12008c performs PHP when it receives packets with a label of 33, and the packet is forwarded to 7200c.

Notice that the packet with label 33 enters 12008c over a different interface than it would have if the protected link did not go down. This is not an issue for 12008c because it is using global label space. 12008c does not care on which interface the packet enters. If it comes in with a label value of 33, it performs PHP and forwards the packet to 7200c. As you can see, for this scheme to work, both label stacking and global label space are required.

The other assumption made in local protection is that the failure is temporary. 12008a forwards the traffic down the backup tunnel upon failure and then informs the primary tunnel's headend (7200a) of the failure and its subsequent protection. 7200a then performs a make-before-break headend LSP reroute using normal CSPF.

NOTE

Because of the assumption that restoration in local protection is temporary, there is no need to reserve bandwidth on the backup tunnel. Reserving bandwidth for backup tunnels not only takes bandwidth away from primary tunnels, but also might influence the placement of other backup tunnels, and as such, should be avoided.

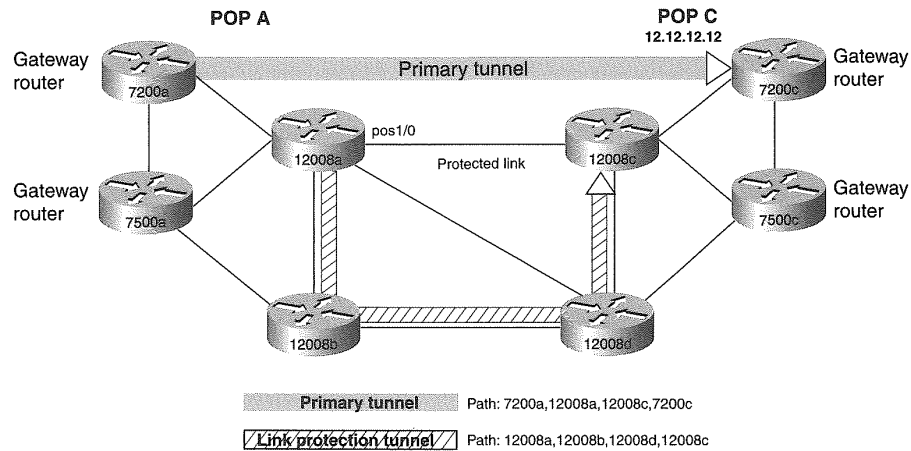
As previously mentioned, there are two types of local protection—*link* protection and *node* protection. They're similar in concept, but node protection is more complex than link protection.

This section briefly summarizes link and node protection. The rest of this chapter examines each of them in more detail.

Link Protection Overview

In many networks that are deployed today, it is common to see high-bandwidth links carrying traffic belonging to "important" flows and other flows that are not so important. If MPLS TE is deployed in such networks, "important flows" translates to "important LSPs." These LSPs might be carrying critical information or time-sensitive data that requires a real-time response. In such cases, it would be nice if all the "important LSPs" could be protected while ignoring the less-important LSPs. FRR allows you to protect some of your TE tunnels (just the ones you deem important) or all of your TE tunnels. With link protection, you can protect links that are carrying these important LSPs by using presignalled backup tunnels that bypass the protected link. Figure 7-5 depicts one such sample network.

Figure 7-5 Link Protection



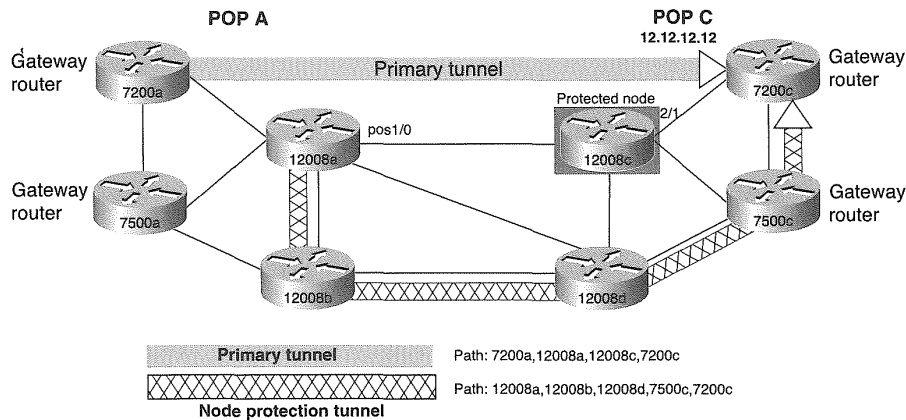
Here, link 12008a→12008c is considered the crucial link over which a primary tunnel is signalled. This link is called the *protected link*. In order to protect this link and the primary tunnel over it, a backup tunnel is signalled around the link so that the headend of the backup tunnel is the node that is immediately upstream of the protected link, and the tail of the backup tunnel is the node that is immediately downstream of the protected link. In Figure 7-5, the headend of the backup tunnel is 12008a (PLR), and the tail is 12008c (MP). As described in the preceding section, when the protected link fails, label stacking delivers the primary tunnel packets to 12008c so that 12008c sees the label it is expecting. Link protection relies on the fact that, although a protected link has gone down, the router at the other end of that protected link is still up. Link protection uses NHop backup tunnels; the backup tunnels terminate on the node on the other end of the protected link. As you can imagine, this allows you to protect against a link failure but not a node failure. If the node you terminate your backup tunnel on goes down, link protection can't help you.

Node Protection Overview

What if the node that is downstream of the protected link goes down, causing the protected link to fail? If this happens, it does you no good to try to deliver the packets to the node that just went down. In this case, you might want to safeguard the primary LSP from node failure on the other end of a link in addition to protecting the link itself.

Looking at this slightly differently, if you protect against failure of the downstream node, you have automatically protected against failure of the downstream link as well. Figure 7-6 shows the protection being moved to 12008c. 12008a is still the PLR, but the MP is now 7200c instead of 12008c. Node protection uses NNHop backup tunnels instead of NHop tunnels because it needs to protect against a failure of the NHop.

Figure 7-6 Node Protection



Node protection is similar to link protection in most ways. It differs from link protection in that the MP is not the NHop, but the NNHop. This has implications in label stacking as well. With link protection, the PLR knows what label the MP expects because it receives a label mapping directly from the MP for the primary tunnel. With node protection, however, the label the MP wants to see is never signalled through RSVP to the PLR. There has to be some other mechanism to learn about the label that MP is expecting for the primary tunnel. This mechanism is discussed in detail in the later section “Node Protection.”

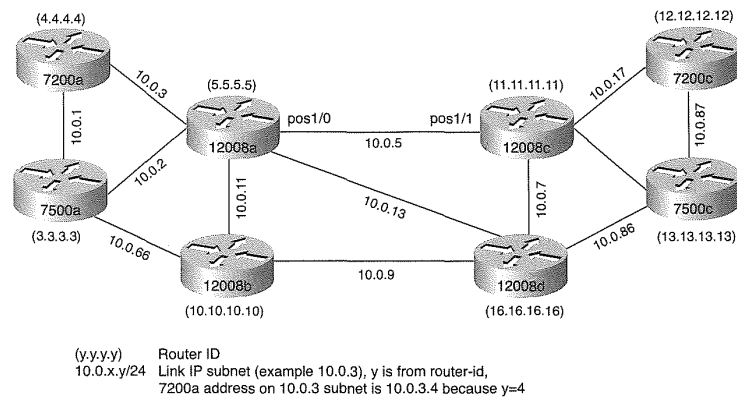
Link Versus Node Protection

Link and node protection scale differently, solve different problems, and have different restrictions. You don’t always need to protect everything in your network. You might want to protect only resources you consider critical (perhaps only the most important core links) or consider protecting links that are not being protected by, say, APS. Perhaps you need to protect LSPs carrying Voice over IP (VoIP) traffic or specific types of traffic from specific customers with whom you have service-level agreements (SLAs).

Where you deploy protection, and what type of protection you deploy, is up to you. Whatever you end up doing, you need to understand the scalability of various protection schemes as part of deciding what to protect. See Chapter 9 for a discussion of protection scalability.

The next two sections examine link and node protection in more detail—how they work, how they’re configured, and the differences between them.

All the figures in this chapter are based on the sample network shown in Figure 7-7. The router IDs (RIDs) (also loopback addresses) are in parentheses above or below the routers. For example, 7200a’s RID is 4.4.4.4.

Figure 7-7 IP Addressing Used in the Sample Network

Each link's subnet address is of the form 10.0.x.y, and each link has a 24-bit mask. For example, the link between 7200a and 12008a is 10.0.3.0/24. The IP address of 7200a's side of the link is 10.0.3.4 because 7200a's RID is 4.4.4.4, and 12008a's side of the link has the address 10.0.3.5 because 12008a's RID is 5.5.5.5.

Link Protection

Link protection can be divided into four sections:

- Prefailure configuration
- Failure detection
- Connectivity restoration
- Post-failure signalling

The following sections examine each of these topics in detail.

Prefailure Configuration

You need to note one subtle point: Link protection, like all other TE applications, is unidirectional. In order to protect LSPs flowing from A to B across a link, you need protection in the A→B direction. To protect LSPs flowing from B to A across the same link, you also need to build a protection LSP in the B→A direction. Most of the configurations or concepts presented in this chapter are unidirectional unless specified otherwise.

There are two places where you have to configure things related to link protection:

- At the headend on the tunnel interface you want to protect
- At the PLR

Enabling Link Protection on a Tunnel Interface

You might not want to protect all your primary tunnels. Remember that link protection means that, when a link goes down, the LSPs that would have gone over that link are instead sent across some other path in the network. If you end up protecting too much traffic (protecting an OC-48 with an LSP that goes down an OC-3, for example), you might end up making things not much better than if there was no protection.

That's why TE tunnels don't request protection by default. In order to have a TE tunnel request protection, you need to explicitly configure the tunnel to ask for protection. You do this using the **tunnel mpls traffic-eng fast-reroute** configuration under the primary tunnel interface at the headend, as shown in Example 7-1.

Example 7-1 Enabling Protection on 7200a

```
7200a#configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
7200a(config)#interface tunnel 1
7200a(config-if)#tunnel mpls traffic-eng fast-reroute
```

When you configure **tunnel mpls traffic-eng fast-reroute**, the headend sets the SESSION_ATTRIBUTE flag 0x01 ("Local protection desired") in the PATH message for that tunnel.

This can be observed in the output of **debug ip rsvp path detail** on the 7200a router, as demonstrated in Examples 7-2 and 7-3.

Example 7-2 shows the output *before* **tunnel mpls traffic-eng fast-reroute** is configured.

Example 7-2 debug ip rsvp path detail Output Before FRR Configuration

```
7200a#debug ip rsvp path detail
*Oct 16 14:38:56.460: SESSION_ATTRIBUTE    type 7 length 24:
*Oct 16 14:38:56.460: Setup Prio: 7, Holding Prio: 7
*Oct 16 14:38:56.460: Flags: SE Style
*Oct 16 14:38:56.460: Session Name: 7200a_t1
```

The only flag set is SE Style.

Example 7-3 shows the output *after* **tunnel mpls traffic-eng fast-reroute** is configured on the primary tunnel interface.

Example 7-3 debug ip rsvp path detail Output After FRR Configuration

```

7200a#debug ip rsvp path detail
*Oct 16 14:40:57.124: SESSION_ATTRIBUTE    type 7 length 24:
*Oct 16 14:40:57.124: Setup Prio: 7, Holding Prio: 7
*Oct 16 14:40:57.124: Flags: Local Prot desired, Label Recording, SE Style
*Oct 16 14:40:57.124: Session Name: 7200a_t1

```

After **tunnel mpls traffic-eng fast-reroute** is configured, three flags are now set, as detailed in Table 7-2.

Table 7-2 Flags Set After FRR Is Configured on the Primary Tunnel Interface

Flag	Description
Local Prot desired	This is how the headend indicates to any downstream nodes that it would like local protection of some sort (either link or node) for this LSP.
Label Recording	Not used in link protection, but as you will see later, it is used in node protection.
SE Style	Stays the same as before tunnel mpls traffic-eng fast-reroute was configured.

Enabling Link Protection at the PLR

Enabling FRR at the PLR involves two things:

- Creating a backup tunnel to the NHop
- Configuring the protected link to use the backup tunnel upon failure

Creating a Backup Tunnel to the NHop

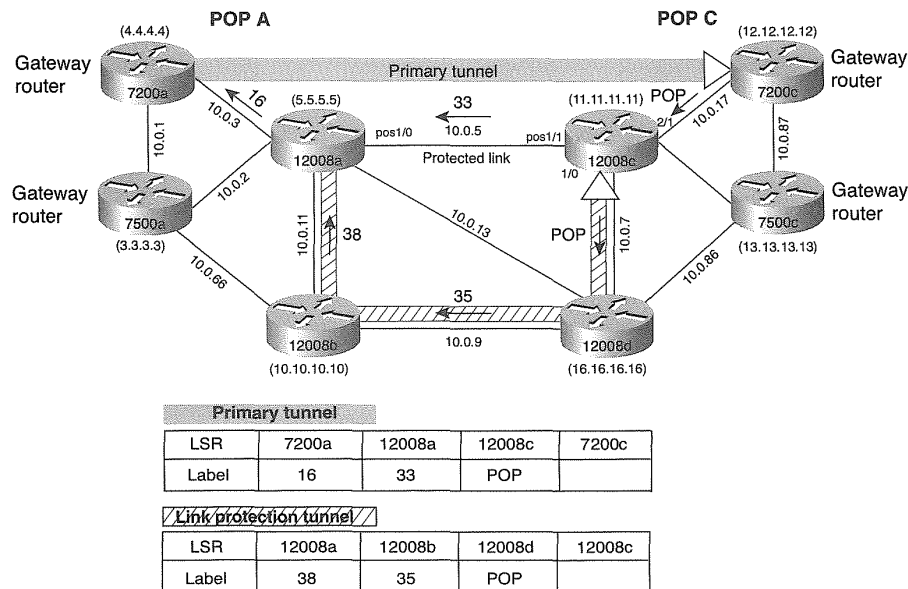
In local protection, a backup tunnel needs to be built from the PLR to the MP. In link protection, the MP is the node on the other end of the protected link, as depicted in Figure 7-8. The primary LSP in Figure 7-8 goes from 7200a to 7200c. The midpoints are 12008a and 12008c. The link that is being protected in this example is the POS link that goes between 12008a and 12008c in the 12008a→12008c direction.

If this link fails, the LSPs that are traversing this link need to be protected and rerouted locally to 12008c, which is the downstream end of this link.

You need to be sure to configure the backup tunnel so that it doesn't attempt to cross the link it's protecting. This would defeat the entire concept of local protection. You can use one of two methods to build the path for this backup tunnel:

- Use an explicitly routed path option to route the backup tunnel away from the link it is protecting.
- Use the **exclude-address** knob to tell a backup LSP to use any path it can find that avoids the link it's protecting.

Figure 7-8 Link Protection

**NOTE**

It is important to understand that, for link protection to work, the backup tunnel must originate at the router on the upstream end of the protected link and terminate on the router at the downstream end of the protected link. It is not important what path the protected link takes, as long as the protection LSP doesn't try to go out the link it's trying to protect. You might want to provision the backup LSP along a path that has available bandwidth comparable to the resources it's protecting.

Example 7-4 shows how to build the path for the backup tunnel using the exclude-address syntax.

Example 7-4 Building a Path for the Backup Tunnel

```
12008a#show running-config interface tunnel1
interface Tunnel1
description Link Protection Tunnel (Backup)
ip unnumbered Loopback0
no ip directed-broadcast
tunnel destination 11.11.11.11
tunnel mode mpls traffic-eng
tunnel mpls traffic-eng path-option 5 explicit name nhop
end
12008a#show ip explicit-paths name nhop
PATH link-protection-tunnel (loose source route, path complete, generation 8)
1: exclude-address 10.0.5.11
```

Example 7-4 excludes 12008c's POS1/1 interface address (10.0.5.11) to ensure that the protected link 12008a→12008c is not included in CSPF for the backup tunnel on 12008a. As you can see from Example 7-5, the backup tunnel comes up. If you examine the ERO, you can see that the protected tunnel is not a part of it.

Example 7-5 *show mpls traffic-eng tunnel tunnel1 Output Verifies that the Backup Tunnel Is Present*

```
12008a#show mpls traffic-eng tunnel tunnel1

Name: Link Protection Tunnel (Backup)      (Tunnel1) Destination: 11.11.11.11
Status:
  Admin: up          Oper: up          Path: valid          Signalling: connected

  path option 5, type explicit nhop (Basis for Setup, path weight 3)

Config Parameters:
  Bandwidth: 0          kbps (Global) Priority: 7 7  Affinity: 0x0/0xFFFF
  Metric Type: TE (default)
  AutoRoute: disabled LockDown: disabled Loadshare: 0          bw-based
  auto-bw: disabled(0/60) 0 Bandwidth Requested: 0

InLabel : -
OutLabel : POS1/1, 38
RSVP Signalling Info:
  Src 5.5.5.5, Dst 11.11.11.11, Tun_Id 1, Tun_Instance 93
RSVP Path Info:
  My Address: 5.5.5.5
  Explicit Route: 10.0.11.10 10.0.9.16 10.0.7.11 11.11.11.11
```

Tying the Protected Link to the Backup Tunnel

Just configuring the backup tunnel and calling the explicit path “backup” does not make traffic go over this tunnel when the protected link goes down. After you build the backup tunnel, you need to tell an interface to use that tunnel for protection.

How do you tie the protection of interface POS 1/0 (the outgoing interface of the protected link on 12008a) to this backup tunnel? You use the configuration highlighted in Example 7-6 on that interface.

Example 7-6 *Configuration for Backing Up Interface POS 1/0 with Tunnel1*

```
12008a#show running-config interface pos1/0
interface POS1/0
 ip address 10.0.5.5 255.255.255.0
 no ip directed-broadcast
 mpls traffic-eng tunnels
 mpls traffic-eng backup-path Tunnel1
 ip rsvp bandwidth 155000 155000
end
```

The command `mpls traffic-eng backup-path Tunnel1` protects 12008a's POS1/0 interface with Tunnel1.

The backup tunnel is now signalled and ready to go. If failure is detected on the protected link, all the traffic from LSPs requesting protection is forwarded over the backup tunnel. Until then, the backup LSP does not carry any traffic. Example 7-7 shows the output of **show mpls traffic-eng fast-reroute database** before any failure has occurred but after the backup tunnel is ready to protect the primary LSP.

Example 7-7 Verifying the FRR State

```
12008a#show mpls traffic-eng fast-reroute database
```

Tunnel head fast reroute information:					
Prefix	Tunnel	In-label	Out intf/label	FRR intf/label	Status
LSP midpoint frr information:					
LSP identifier		In-label	Out intf/label	FRR intf/label	Status
4.4.4.4 1 [1520]		16	PO1/0:33	Tu1:33	ready

Here is a detailed explanation of the key fields in this output:

- **LSP identifier**—This column shows the source address of the primary LSP being protected (4.4.4.4), its tunnel ID (1), and its LSP ID (1520).
- **In-label**—This column shows the incoming label that is being protected. This label corresponds to the tunnel listed in the **LSP identifier** column.
- **Out intf/label**—This column shows the interface of the protected link (PO1/0) and the tunnel's outgoing label on that interface (33).
- **FRR intf/label**—This column shows the backup tunnel that is being used to back up the primary tunnel. When POS1/0 fails, the packet that would have gone out POS1/0 with label 33 is instead sent down the backup tunnel, but with that same label value.
- **Status**—This column shows **ready**, meaning that FRR protection is ready to back up the primary tunnel, and that a failure has not yet occurred. If the **Status** field shows **active**, the protected link is currently down, and protection is *active* (sometimes referred to as *in place*). The **Status** field can also show **partial**, meaning that all the pieces required to back up a primary LSP are not yet available. A tunnel that has requested protection is not protected, generally because the interface is not configured with **mpls traffic-eng backup-path**, or the backup tunnel is not up.

If the backup tunnel exists and is up, the PLR (12008a) responds by setting the RRO subobject IPv4 flags to 0x01, indicating "Local protection available." This is specified in RFC 3209. You can see this from the output of **debug ip rsvp resv detail** on 7200a, as demonstrated in Example 7-8.

Example 7-8 debug ip rsvp resv detail Output on 7200a

```
7200a#debug ip rsvp resv detail
*Oct 17 08:38:04.216: RSVP:      version:1 flags:0000 type:RESV cksum:0000 ttl:255
  reserved:0 length:152
*Oct 17 08:38:04.220: SESSION                type 7 length 16:
*Oct 17 08:38:04.220:  Tun Dest 12.12.12.12 Tun ID 1 Ext Tun ID 4.4.4.4
*Oct 17 08:38:04.220: HOP                  type 1 length 12:
*Oct 17 08:38:04.220:  Hop Addr: 10.0.3.5 LIH: 0x0
```

continues

Example 7-8 debug ip rsvp resv detail Output on 7200a (Continued)

```

*Oct 17 08:38:04.220: TIME_VALUES          type 1 length 8 :
*Oct 17 08:38:04.220: Refresh Period (msec): 30000
*Oct 17 08:38:04.220: STYLE          type 1 length 8 :
*Oct 17 08:38:04.220: RSVP_SE_OPTION
*Oct 17 08:38:04.220: FLOWSPEC          type 2 length 36:
*Oct 17 08:38:04.220: version = 0 length in words = 7
*Oct 17 08:38:04.220: service id = 5, service length = 6
*Oct 17 08:38:04.220: tspec parameter id = 127, tspec flags = 0, tspec length = 5
*Oct 17 08:38:04.220: average rate = 0 bytes/sec, burst depth = 1000 bytes
*Oct 17 08:38:04.220: peak rate      = 2147483647 bytes/sec
*Oct 17 08:38:04.220: min unit = 0 bytes, max unit = 0 bytes
*Oct 17 08:38:04.220: FILTER_SPEC          type 7 length 12:
*Oct 17 08:38:04.220: Tun Sender: 4.4.4.4, LSP ID 1520
*Oct 17 08:38:04.220: LABEL          type 1 length 8 : 00000012
*Oct 17 08:38:04.220: RECORD_ROUTE          type 1 length 44:
*Oct 17 08:38:04.220: 10.0.5.5/32, Flags:0x1 (Local Prot Avail/to NHOP)
*Oct 17 08:38:04.220: Label record: Flags 0x1, ctype 1, incoming label 16
*Oct 17 08:38:04.220: 10.0.17.11/32, Flags:0x0 (No Local Protection)
*Oct 17 08:38:04.220: Label record: Flags 0x1, ctype 1, incoming label 33
*Oct 17 08:38:04.220: 10.0.17.12/32, Flags:0x0 (No Local Protection)
*Oct 17 08:38:04.220:
*Oct 17 08:38:04.220: RSVP 4.4.4.4_1520-12.12.12.12_1: RESV message arrived from
10.0.3.5 on POS3/0

```

The highlighted output in Example 7-8 also shows that protection is available to the PLR's NHop.

Entering **show ip rsvp sender detail** on 7200a and 12008a, as shown in Example 7-9, is useful to get a quick synopsis of the FRR states of the two routers.

Example 7-9 Determining the FRR States on 7200a and 12008a

```

7200a#show ip rsvp sender detail
PATH:
Tun Dest 12.12.12.12 Tun ID 1 Ext Tun ID 4.4.4.4
Tun Sender: 4.4.4.4, LSP ID: 1520
Path refreshes being sent to NHOP 10.0.3.5 on POS3/0
Session Attr::
Setup Prio: 7, Holding Prio: 7
Flags: Local Prot desired, Label Recording, SE Style
Session Name: 7200a_t1
ERO:
10.0.3.5 (Strict IPv4 Prefix, 8 bytes, /32)
10.0.5.11 (Strict IPv4 Prefix, 8 bytes, /32)
10.0.17.12 (Strict IPv4 Prefix, 8 bytes, /32)
12.12.12.12 (Strict IPv4 Prefix, 8 bytes, /32)
RR0:
Empty
Traffic params - Rate: 0G bits/sec, Max. burst: 1K bytes
Fast-Reroute Backup info:
Inbound FRR: Not active
Outbound FRR: No backup tunnel selected

```

Example 7-9 *Determining the FRR States on 7200a and 12008a (Continued)*

```

12008a#show ip rsvp sender detail
PATH:
  Tun Dest 12.12.12.12 Tun ID 1 Ext Tun ID 4.4.4.4
  Tun Sender: 4.4.4.4, LSP ID: 1520
  Path refreshes arriving on POS2/1 from PHOP 10.0.3.4
  Path refreshes being sent to NHOP 10.0.5.11 on POS1/0
  Session Attr::
    Setup Prio: 7, Holding Prio: 7
    Flags: Local Prot desired, Label Recording, SE Style
    Session Name: 7200a_t1
ERO:
  10.0.5.11 (Strict IPv4 Prefix, 8 bytes, /32)
  10.0.17.12 (Strict IPv4 Prefix, 8 bytes, /32)
  12.12.12.12 (Strict IPv4 Prefix, 8 bytes, /32)
RRO:
  10.0.3.4/32, Flags:0x0 (No Local Protection)
Traffic params - Rate: 0G bits/sec, Max. burst: 1K bytes
Fast-Reroute Backup info:
  Inbound FRR: Not active
  Outbound FRR: Ready -- backup tunnel selected
  Backup Tunnel: Tu1 (label 33)
  Bkup Sender Template:
    Tun Sender: 10.0.11.5, LSP ID: 506
  Bkup FilerSpec:
    Tun Sender: 10.0.11.5, LSP ID 506

```

Failure Detection

One of the most important pieces of any dynamic protocol is failure detection. This is doubly important in protection mechanisms because the longer it takes to detect a failure, the longer it takes to kick in the protection mechanism designed to circumvent that failure, and the less good your protection mechanism does you.

The methods for detecting these failures and the complexity involved in detecting them vary. It is easy to determine that a link has gone down if the PLR has had its interface administratively turned down, and harder to determine if the interface is down if there is a failure at or between the PLR and the other end of the circuit. Fortunately, detection of a failed link is nothing new. The mechanisms in place to aid the detection of these failures are

- Failure detection mechanisms specific to a particular physical layer, such as SONET
- For point-to-point links, PPP or HDLC keepalives
- RSVP hello extensions

Link protection needs to be able to detect that a directly connected link to the NHop has gone down. The next two sections examine detection using Layer 2 alarms and RSVP hellos in detail.

Failure Detection Using SONET Alarms

When link protection was designed, SONET APS 50-millisecond convergence was used as the metric. An important part of achieving this goal is rapid detection of an interface's failure. If it takes 10 seconds to figure out that a link is down, that's 10 seconds of lost traffic that was forwarded out that link, and that's not good. SONET's APS (and SDH's MSP) are triggered by various physical alarms; FRR merely hooks into those alarms and so detects interface failure as quickly as APS and MSP would.

NOTE Although FRR can be triggered by SONET alarms, it is supported only on POS interfaces, not ATM SONET interfaces.

NOTE Although FRR works off the same alarms that trigger APS, you do not need to configure APS to get FRR to work. They're two completely independent mechanisms that just happen to use the same failure trigger.

It is a good idea to enable `pos ais-shut` on both sides of any POS link; this sends a Line Alarm Indication Signal (LAIs) when the interface is administratively shut down. This is independent of whether you use FRR or not; `pos ais-shut` simply helps a router detect when the interface on the other side of a link has been shut down. Without `pos ais-shut`, the router needs to rely on PPP/HDLC keepalives timing out to figure out that an interface went down, and keepalive timeout can take a while.

Failure Detection Using RSVP Hellos

RFC 3209 defines RSVP hellos in Section 5. Here is part of the definition:

The RSVP Hello extension enables RSVP nodes to detect when a neighboring node is not reachable. The mechanism provides node to node failure detection. When such a failure is detected it is handled much the same as a link layer communication failure. This mechanism is intended to be used when notification of link layer failures is not available and unnumbered links are not used, or when the failure detection mechanisms provided by the link layer are not sufficient for timely node failure detection.

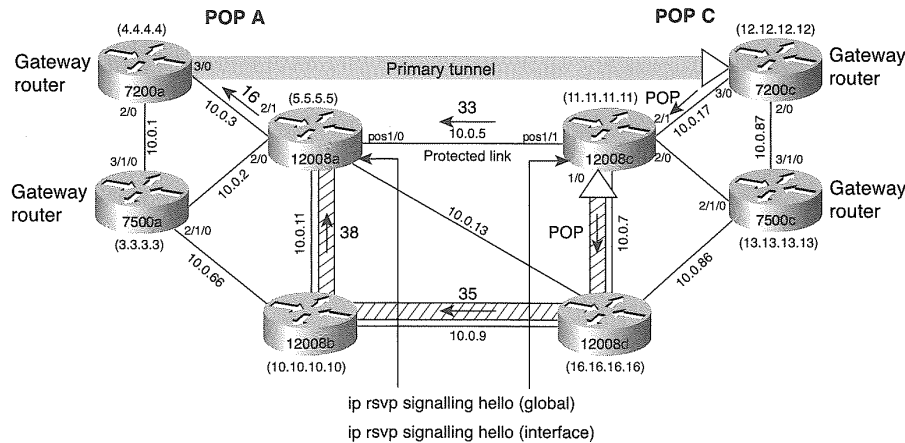
Physical layer alarms are not always available. For example, a Gigabit Ethernet interface that's connected to a switch doesn't always lose the link signal when a link in the same VLAN goes down.

If you want to detect failures on non-POS interfaces, you might want to use RSVP hellos. However, you need to know that RSVP hello-based failure detection is somewhat slower than Layer 2 alarm-based detection. It can take several hundred milliseconds to detect a neighbor failure using RSVP hellos, depending on how you tune them.

Even so, RSVP hello-based detection is considered sufficient for failure detection in local protection, and convergence is faster than plain IP or MPLS TE without FRR.

Figure 7-9 shows the configuration used for RSVP hellos.

Figure 7-9 Failure Detection Using RSVP Hellos



You need to configure RSVP hellos at both the global and interface levels, as shown in Example 7-10.

Example 7-10 Configuring RSVP Hellos at the Global and Interface Levels

```
12008a(config)#ip rsvp signalling hello
12008a(config)#interface pos 1/0
12008a(config-if)#ip rsvp signalling hello ?
    missed-acks      # missed Hello Acks which triggers neighbor down
    refresh-interval Time between sending Hello Requests, msec.
    <cr>
12008a(config-if)#ip rsvp signalling hello missed-acks ?
    <2-10> Hello missed
```

You control the refresh interval by entering the command shown in Example 7-11.

Example 7-11 Determining the Refresh Interval Setting

```
12008a(config-if)#ip rsvp signalling hello refresh-interval ?
    <10-10000> Hello interval
    <cr>
```

If no refresh interval is specified, it defaults to 100 milliseconds. Keep this default unless you have a good reason to change it. RSVP hellos need to be configured on *both* sides of a link.

You can examine the configuration of RSVP hellos using the **show ip rsvp hello instance detail** command, as demonstrated in Example 7-12.

Example 7-12 *Determining RSVP Hello Configuration Information*

```
mpls-12008a#show ip rsvp hello instance detail
Neighbor 10.0.5.11 Source 10.0.5.5
State: LOST
Type: ACTIVE (sending requests)
I/F: PO1/0
LSPs protecting: 3
Refresh Interval (msec)
  Configured: 100
Statistics:
  no stats collected
Src_instance 0xCFF52A98, Dst_instance 0x0
Counters:
  Communication with neighbor lost:
    Num times: 0
    Reasons:
      Missed acks: 0
      Bad Src_Inst received: 0
      Bad Dst_Inst received: 0
      I/F went down: 0
      Neighbor disabled Hello: 0
  Msgs Received: 0
    Sent: 24
    Suppressed: 0

mpls-12008a#
```

Using the **missed-acks** knob, you can control how many acknowledgments should be missed before the neighbor is considered down. This defaults to 4 if it is not configured explicitly.

Connectivity Restoration

As soon as the failure is detected, the PLR is responsible for switching traffic to the backup tunnel. The internal processing performed on the PLR involves the following:

- Making sure a presignalled backup LSP is in place. This includes the new label provided by a new downstream neighbor.
- New adjacency information (Layer 2 encapsulation) is computed based on the backup tunnel's outgoing physical interface.

This information is precomputed and ready to be installed in the FIB/LFIB as soon as the failure is detected so as to minimize packet loss; this is referred to as the *ready* state, as shown in Example 7-13. After the failure is detected, FRR kicks and is said to be in *active* state (see Example 7-17). These states can only be observed on the PLR (12008a, in this example).

Example 7-13 Verifying the FRR State

```

12008a#show mpls traffic-eng fast-reroute database
Tunnel head fast reroute information:
Prefix          Tunnel    In-label Out intf/label  FRR intf/label  Status
LSP midpoint frr information:
LSP identifier   In-label Out intf/label  FRR intf/label  Status
4.4.4.4 1 [1520]      16      PO1/0:33      Tu1:33          ready

```

The LSP identifier (4.4.4.4 1 [1520]) in Example 7-13 refers to the tunnel's source, the tunnel identifier, and the tunnel instance. This should agree with what you see in the output of `show mpls traffic-eng tunnel tunnel1` on 7200a, shown in Example 7-14.

Example 7-14 Tunnel State on Primary Tunnel Headend 7200

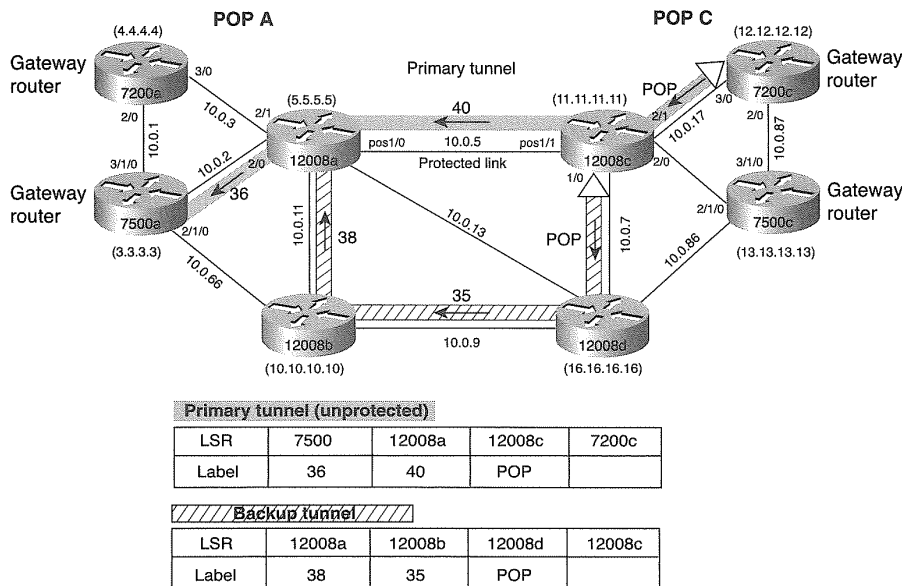
```

7200a#show mpls traffic-eng tunnel tunnel1
...
RSVP Signalling Info:
    Src 4.4.4.4, Dst 12.12.12.12, Tun_Id 1, Tun_Instance 1520
...

```

The output in Examples 7-13 and 7-14 demonstrates that protection is offered to the LSP originating from 7200a but not to the one originating from 7500a.

Figure 7-10 shows a tunnel from 7500a that is not requesting protection.

Figure 7-10 Unprotected LSP Over a Protected Link

You can see this in the configuration of the tunnel on 7500a in Example 7-15.

Example 7-15 7500a Tunnel Configuration

```
7500a#show running-config interface tunnel1
interface Tunnel1
description Primary tunnel(7500a->12008a->12008c->7200c)
ip unnumbered Loopback0
no ip directed-broadcast
ip route-cache distributed
tunnel destination 12.12.12.12
tunnel mode mpls traffic-eng
tunnel mpls traffic-eng autoroute announce
tunnel mpls traffic-eng path-option 5 explicit name primary
tunnel mpls traffic-eng path-option 6 dynamic
tunnel mpls traffic-eng record-route
end

7500a#show ip explicit-paths name primary
PATH primary (strict source route, path complete, generation 6)
  1: next-address 10.0.2.5
  2: next-address 10.0.5.11
  3: next-address 10.0.17.12
  4: next-address 12.12.12.12
```

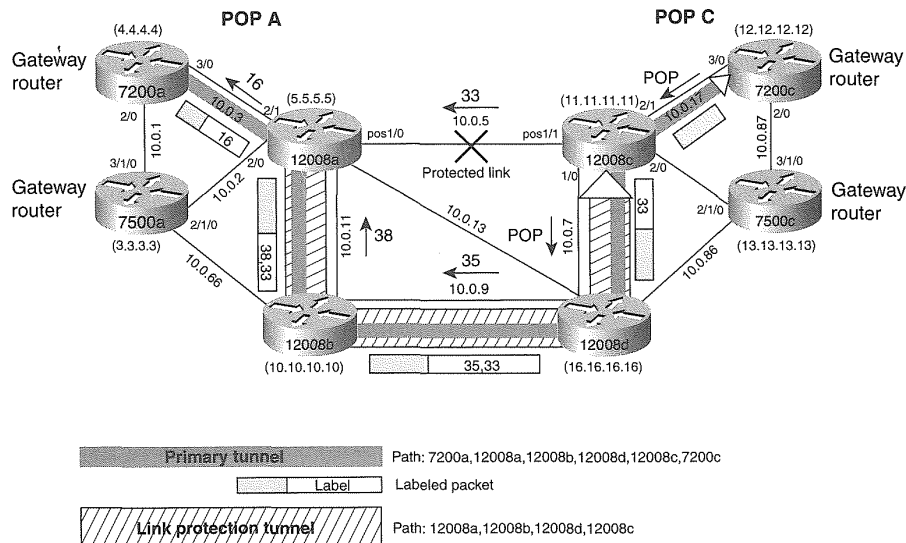
Example 7-15 shows that FRR is not configured under tunnel1 on 7500a.

As soon as a failure is detected, FRR quickly reacts to send traffic down the protection tunnel. This can be seen in the output from **show mpls traffic-eng fast-reroute database** (see Example 7-17) and **show mpls traffic-eng fast-reroute log reroutes** (see Example 7-20).

Figure 7-11 depicts the flow of traffic over the *link protection tunnel*. It is important to note that for local protection mechanisms, while the protection is active and the backup tunnel is forwarding traffic, the primary LSP continues to stay up. This is different from the end-to-end path protection scheme, in which the primary LSP goes down after a failure, gracefully switching over to standby LSP to minimize packet loss. This results in the tunnel's staying up.

When local protection is active, if you just examine the state of the primary tunnel on 7200a using the **show mpls traffic-eng tunnels tunnel1** command, you might not notice that protection has kicked in because the primary LSP is still up and you see only the original ERO, which does not include the backup path. As you can see from Figure 7-11, the primary tunnel traffic now goes over the backup path. Doing a **traceroute** to the tunnel destination when FRR is active shows you that the backup path is actually taken. Example 7-16 shows the output of the **traceroute** from 7200a to 7200c. 10.0.3.5 is the first hop to 12008a, 10.0.11.10 is the second hop to 12008b, 10.0.9.16 is the third hop to 12008d, 10.0.7.11 is the forth hop to 12008c, and 10.0.17.12 is the final hop to 7200c. You can also observe the label stack in this output.

Figure 7-11 Link Protection After the Protected Link Fails



Example 7-16 traceroute Output

```

7200a#traceroute 12.12.12.12
Type escape sequence to abort.
Tracing the route to 12.12.12.12

 1 10.0.3.5 [MPLS: Label 16 Exp 0] 0 msec 0 msec 0 msec
 2 10.0.11.10 [MPLS: Labels 38/33 Exp 0] 0 msec 0 msec 4 msec
 3 10.0.9.16 [MPLS: Labels 35/33 Exp 0] 0 msec 0 msec 0 msec
 4 10.0.7.11 [MPLS: Label 33 Exp 0] 0 msec 0 msec 0 msec
 5 10.0.17.12 4 msec * 0 msec

```

After failure, you can check if the FRR is active using the **show mpls traffic-eng fast-reroute** command at the PLR (12008a), as demonstrated in Examples 7-17 and 7-18.

Example 7-17 Determining Whether FRR Is Active

```

12008a#show mpls traffic-eng fast-reroute database
Tunnel head fast reroute information:
Prefix      Tunnel    In-label Out intf/label  FRR intf/label  Status
LSP midpoint fr information:
LSP identifier In-label Out intf/label  FRR intf/label  Status
4.4.4.4 1 [1520]      16      P01/0:33      Tu1:33          active

```

Example 7-18 *Details of the FRR Database*

```

12008a#show mpls traffic-eng fast-reroute database detail
LFIB FRR Database Summary:
  Total Clusters:      1
  Total Groups:        1
  Total Items:         1
Link 4: P01/0 (Down, 1 group)
  Group 28: P01/0->Tu1 (Up, 1 member)
    LSP identifier 4.4.4.4 1 [1520], active
    Input label 16, Output label P01/0:33, FRR label Tu1:33

```

Another useful command for checking FRR information is **show ip rsvp sender detail**. It shows what the RSVP neighbors have sent. Example 7-19 shows the output of this command executed on 12008a.

Example 7-19 *show ip rsvp sender detail Output*

```

12008a#show ip rsvp sender detail
PATH:
  Tun Dest: 12.12.12.12 Tun ID: 1 Ext Tun ID: 4.4.4.4
  Tun Sender: 4.4.4.4 LSP ID: 1520
  Path refreshes arriving on POS2/1 from PHOP 10.0.3.4
  Path refreshes being sent to NHOP 11.11.11.11 on Tunnel1
  Session Attr::
    Setup Prio: 7, Holding Prio: 7
    Flags: Local Prot desired, Label Recording, SE Style
    Session Name:Primary tunnel 7200a->12008a->12008c->7200c
  ERO:
    11.11.11.11 (Strict IPv4 Prefix, 8 bytes, /32)
    10.0.17.12 (Strict IPv4 Prefix, 8 bytes, /32)
    12.12.12.12 (Strict IPv4 Prefix, 8 bytes, /32)
  RRO:
    10.0.3.4/32, Flags:0x0 (No Local Protection)
  Traffic params - Rate: 0G bits/sec, Max. burst: 1K bytes
  Fast-Reroute Backup info:
    Inbound FRR: Not active
    Outbound FRR: Active -- using backup tunnel
      Backup Tunnel: Tu1 (label 33)
    Bkup Sender Template:
      Tun Sender: 10.0.11.5 LSP ID: 1632
    Bkup FilerSpec:
      Tun Sender: 10.0.11.5, LSP ID 1632
    Orig Output I/F: P01/0
    Orig Output ERO:
      10.0.5.11 (Strict IPv4 Prefix, 8 bytes, /32)
      10.0.17.12 (Strict IPv4 Prefix, 8 bytes, /32)
      12.12.12.12 (Strict IPv4 Prefix, 8 bytes, /32)

```

The **show mpls traffic-eng fast-reroute database** and **show ip rsvp sender detail** commands are good when they are applied *when FRR is active*. However, if you want to get

historical information about when FRR was active and for how long, you can use the **show mpls traffic-eng fast-reroute log reroutes** command, shown in Example 7-20.

Example 7-20 Obtaining Historical Information for Fast Reroute

12008a#show mpls traffic-eng fast-reroute log reroutes								
When	Interface	Event	Rewrites	Duration	CPU msecs	Suspends	Errors	
23:58:20	Tu1	Down	0	0 msecs	0	0	0	
23:58:04	P01/0	Down	1	0 msecs	0	0	0	
23:57:54	P01/0	Up	0	0 msecs	0	0	0	
01:05:47	P01/0	Down	1	0 msecs	0	0	0	
01:01:12	P01/0	Down	1	0 msecs	0	0	0	
00:34:39	P01/0	Down	1	0 msecs	0	0	0	
00:00:41	P01/0	Down	1	0 msecs	0	0	0	

Some useful commands for debugging FRR are as follows:

- **debug ip rsvp fast-reroute**
- **debug mpls lfib fast-reroute database**
- **debug mpls lfib fast-reroute events**
- **debug mpls lfib fast-reroute reroutes**

Post-Failure Signalling

As you understand by now, much of RSVP-based MPLS TE revolves around RSVP signalling. Fast restoration is not an exception. FRR concepts and implementation depend heavily on making further extensions to what is already defined in RFC 3209. These extensions are specified in *draft-ietf-mpls-rsvp-lsp-fastreroute*. This section talks about RSVP signalling that happens after FRR protection has kicked in. This can be broken into the following:

- Upstream signalling
- IGP notification
- Downstream signalling

Upstream Signalling

Recall from Chapter 4, “Path Calculation and Setup,” that when a link goes down along an LSP, the node that is upstream of the failed link signals a path error to the headends of the LSPs traversing the failed link. In Figure 7-12, after the link between 12008a and 12008c fails, it is the responsibility of 12008a to send a PathErr message to 7200a, which is the headend of the primary tunnel.

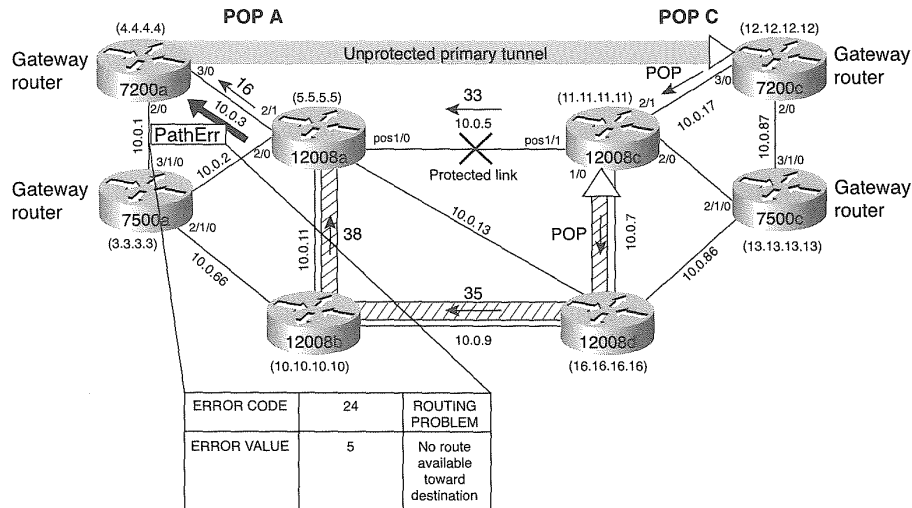
According to RFC 3209, “RSVP-TE: Extensions to RSVP for LSP Tunnels,” with no local protection configured, a node that is upstream of a failed link needs to send a PathErr for each LSP that crossed this link. This PathErr contains an error code of 24 (meaning

“Routing Problem”) with a value of 5 (indicating “No route available toward destination”), as shown in Figure 7-12. You can see this in the output of **debug ip rsvp path detail** on 7200a after shutting down the link between 12008a and 12008c, as demonstrated in Example 7-21.

Example 7-21 Using RSVP Path Debug to Observe PathErr

```
7200a#debug ip rsvp path detail
*Oct 16 12:54:09.469: RSVP:      version:1 flags:0000 type:PEER cksum:0000 ttl:25
*Oct 16 12:54:09.469: SESSION          type 7 length 16:
*Oct 16 12:54:09.469:   Tun Dest 12.12.12.12 Tun ID 1 Ext Tun ID 4.4.4.4
*Oct 16 12:54:09.469: ERROR_SPEC          type 1 length 12:
*Oct 16 12:54:09.469:   Error Node: 10.0.3.5
*Oct 16 12:54:09.469:   Error Code: 24 (Routing Problem)
*Oct 16 12:54:09.469:   Error Value: 0x5 (No route available toward destination)
```

Figure 7-12 PathErr Without Local Protection



When an LSP's headend gets the error shown in Example 7-21, it brings the tunnel interface down and then tries to find a new path for the LSP. The headend ignores the fact that local protection might be available around the broken link.

As a result, traffic along that LSP is blackholed until the LSP can be rerouted. This makes the backup LSP completely useless. Hence, you need a mechanism for 12008a to tell 7200a something like this: “My downstream link along the LSP is broken. I am temporarily rerouting the traffic. This path might no longer be the optimal path to the destination. Please compute an alternative path if one is available.” This is also referred to as the LSR 12008a *triggering reoptimization*.

To signal the nondestructive information, RFC 3209 specifies using a PathErr with an ERROR_SPEC containing error code 25, "Notification," and a subcode of 3, "Tunnel locally repaired."

When an LSP headend receives such a message, it knows that it does not need to immediately stop using its primary LSP, just that this LSP might be following a suboptimal path until it can be rerouted. The headend is free to reroute the LSP when it gets a chance to do so. What protection buys you here is that during the time before the headend can find a suitable alternative end-to-end path, traffic is still being delivered down the backup tunnel.

A headend that receives a notification of 25/3 attempts to calculate and signal a new path for that tunnel. After receiving the reservation (RESV) message for this new path, the label for the old path is replaced with the new label. Only then is the old LSP torn down. This achieves make-before-break and helps minimize packet loss.

If the headend cannot find a new path for an LSP that is currently being protected, the headend remains on the protected path. You might have an LSP that has an explicit path that doesn't allow it to get away from a failed link, or perhaps the necessary bandwidth for an LSP is available only along the link that has failed. As long as the protection tunnel is in place, the protected LSP remains up and passes traffic, and the headend periodically tries to find a new path for that tunnel to take.

Figure 7-13 shows the PathErr message when local protection has been enabled.

Figure 7-13 PathErr with Local Protection

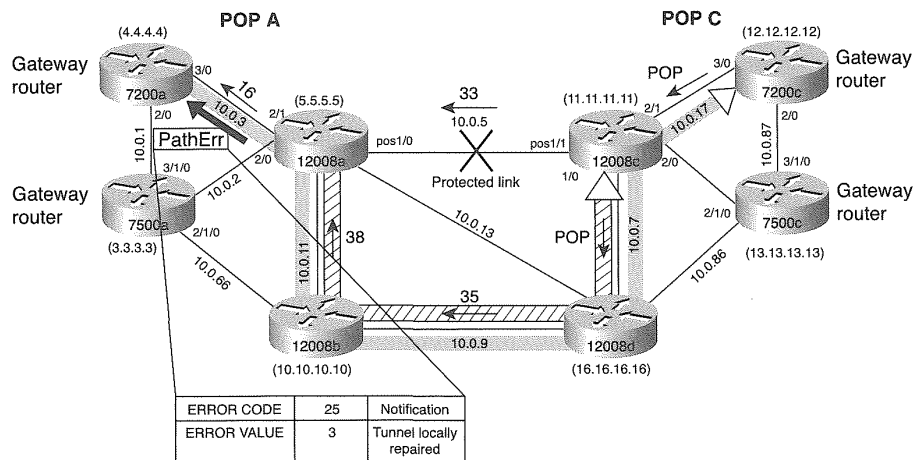


Figure 7-13 shows that 12008a (PLR) sends a PathErr to the primary tunnel headend 7200a. This is also captured in the debug output on 7200a using the **debug ip rsvp path detail** command, as shown in Example 7-22.

Example 7-22 PathErr Sent from 12008a to 7200a

```
7200a#debug ip rsvp path detail
*Oct 17 08:20:45.420: RSVP:      version:1 flags:0000 type:PERR cksum:0000 ttl:255
reserved:0 length:132
*Oct 17 08:20:45.420: SESSION                type 7 length 16:
*Oct 17 08:20:45.420:   Tun Dest 12.12.12.12 Tun ID 1 Ext Tun ID 4.4.4.4
*Oct 17 08:20:45.420: ERROR_SPEC                type 1 length 12:
*Oct 17 08:20:45.420:   Error Node: 10.0.3.5
*Oct 17 08:20:45.420:   Error Code: 25 (Notify)
*Oct 17 08:20:45.420:   Error Value: 0x3 (Tunnel locally repaired)
*Oct 17 08:20:45.420: SENDER_TEMPLATE           type 7 length 12:
*Oct 17 08:20:45.420:   Tun Sender: 4.4.4.4, LSP ID: 1520
*Oct 17 08:20:45.420: SENDER_TSPEC              type 2 length 36:
*Oct 17 08:20:45.420:   version=0, length in words=7
*Oct 17 08:20:45.420:   service id=1, service length=6
*Oct 17 08:20:45.420:   parameter id=127, flags=0, parameter length=5
*Oct 17 08:20:45.420:   average rate=0 bytes/sec, burst depth=1000 bytes
*Oct 17 08:20:45.420:   peak rate   =0 bytes/sec
*Oct 17 08:20:45.420:   min unit=0 bytes, max unit=0 bytes
*Oct 17 08:20:45.420: ADSPEC                    type 2 length 48:
*Oct 17 08:20:45.420:   version=0 length in words=10
*Oct 17 08:20:45.420: General Parameters break bit=0 service length=8
*Oct 17 08:20:45.420:                               IS Hops:1
*Oct 17 08:20:45.420:                               Minimum Path Bandwidth (bytes/sec):19375000
*Oct 17 08:20:45.420:                               Path Latency (microseconds):0
*Oct 17 08:20:45.420:                               Path MTU:4470
*Oct 17 08:20:45.420: Controlled Load Service break bit=0 service length=0
*Oct 17 08:20:45.420:
*Oct 17 08:20:45.420: RSVP 4.4.4.4_1520.12.12.12.12_1: PATH ERROR message for
12.12.12.12 (POS3/0) from 10.0.3.5
```

As shown in the highlighted output in Example 7-22, 10.0.3.5 happens to be the interface address of 12008a facing 7200a. This is highlighted so that you know where the RSVP messages are coming from.

When a protected link fails and is switched down the backup tunnel, the PLR also sends Path messages for the protected LSPs down the backup tunnel. This is so that the MP doesn't time out the protected tunnel, in the unlikely event that the protected tunnel headend can't reroute the LSP. See the later section "Downstream Signalling" for more details on why this is necessary.

Additionally, some changes are made to the body of the Path message itself. As you know, RSVP is a soft-state protocol. In order to keep sessions alive, RSVP refresh messages are sent periodically. These refresh messages are sent between RSVP neighbors by sending Path and Resv messages. LSP tunnels are identified by a combination of the SESSION and SENDER_TEMPLATE objects in these Path and Resv messages (see Chapter 4). The

SENDER_TEMPLATE object is modified by the PLR so that the sender IPv4 address now contains the PLR's IP address rather than that of the headend. Doing so allows the tail to see this Path message as coming from a new sender but still belonging to the same session.

From this point on, the refresh messages can flow over the backup tunnel. The original state maintained by the tail for this session is eventually torn down because of timeout (by any LSR downstream of the failed link, including the tail), but the altered Path message from the PLR is enough to effectively maintain the bandwidth reservation for as long as necessary. Example 7-23 shows the original ERO on 7200a before the protected link between 12008a and 12008c went down. Example 7-24 shows the change in ERO in the RSVP refresh message over the backup tunnel.

Example 7-23 Original ERO on Headend Router 7200a

```
7200a#show mpls traffic-eng tunnel tunnel1
Name: Primary tunnel 7200a->12008a->12... (Tunnel1) Destination: 12.12.12.12
Status:
  Admin: up      Oper: up      Path: valid      Signalling: connected

  path option 5, type explicit primary (Basis for Setup, path weight 3)

Config Parameters:
  Bandwidth: 100      kbps (Global) Priority: 7 7 Affinity: 0x0/0xFFFF
  Metric Type: TE (default)
  AutoRoute: enabled LockDown: disabled Loadshare: 100      bw-based
  auto-bw: disabled(0/258) 0 Bandwidth Requested: 100

InLabel : -
OutLabel : POS3/0, 16
RSVP Signalling Info:
  Src 4.4.4.4, Dst 12.12.12.12, Tun_Id 1, Tun_Instance 1520
RSVP Path Info:
  My Address: 4.4.4.4
  Explicit Route: 10.0.3.5 10.0.5.11 10.0.17.12 12.12.12.12
```

Example 7-24 Path Message for 7200c

```
12008a#debug ip rsvp path detail
RSVP 4.4.4.4_1520-12.12.12.12_1: PATH message for 12.12.12.12(POS2/1) from 10.0.3.4
RSVP 4.4.4.4_1520-12.12.12.12_1: PATH: fasteroute in progress
RSVP: version:1 flags:0000 type:PATH cksum:0000 ttl:254 reserved:0 length:216
SESSION      type 7 length 16:
  Destination 12.12.12.12, TunnelId 3, Source 3.3.3.3
HOP           type 1 length 12: 0A000203
              : 00000000
TIME_VALUES   type 1 length 8 : 00007530
EXPLICIT_ROUTE type 1 length 52:
  (#1) Strict IPv4 Prefix, 8 bytes, 10.0.2.5/32
  (#2) Strict IPv4 Prefix, 8 bytes, 10.0.11.10/32
  (#4) Strict IPv4 Prefix, 8 bytes, 10.0.7.11/32
  (#5) Strict IPv4 Prefix, 8 bytes, 10.0.17.12/32
  (#6) Strict IPv4 Prefix, 8 bytes, 12.12.12.12/32
```

Related to the refresh messages is the fact that Path messages would have to be forwarded down the backup tunnel by the PLR. But if the PLR did so using the contents of the ERO, as it would normally do, it would fail because the next IP address in the ERO would point to the failed link. This behavior has to change to make FRR work. In addition to the ERO, the RRO and phop objects are modified for refresh messages flowing over the backup tunnel according to the IETF draft.

IGP Notification

Although in many cases, RSVP messages reach either the headend or tailend ahead of any IGP notification, this is not guaranteed to be the case. When IGP information (such as OSPF/IS-IS LSA declaring a link down) for some reason makes it before the RSVP message, what consequences does this have on TE signalling?

In the absence of FRR, if the primary tunnel headend receives a link-down LSA for a link that was part of the primary LSP, the headend tears down the primary tunnel. After that, the headend can, if configured correctly, attempt to reroute the LSP.

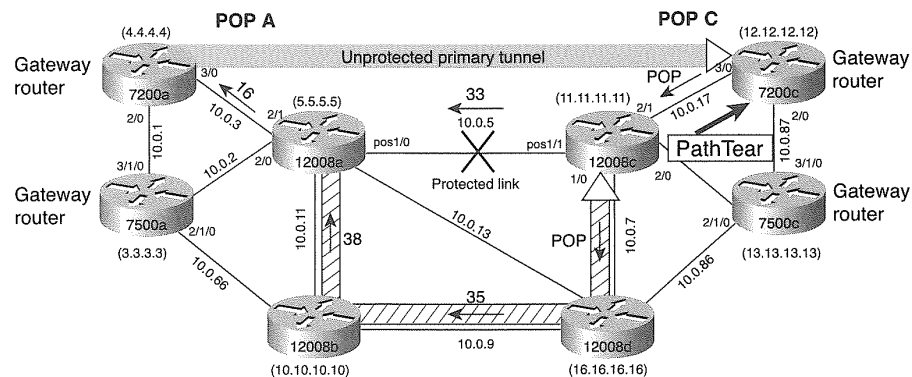
If the primary tunnel is configured for FRR, the link-down LSA has no effect. The headend tears down a protected LSP based only on RSVP error messages and ignores IGP's reporting a link down along the LSP. This is because a downed link doesn't necessarily mean a failed LSP because the LSP could be protected.

Downstream Signalling

You just saw the repercussions of link failure upstream of the failed link, both with and without local protection. This section examines what goes on downstream of the failed link.

Consider the topology shown in Figure 7-14.

Figure 7-14 *PathTear in the Absence of FRR*



When the link between 12008a and 12008c goes down (when no local protection is in place), 12008c sends a PathTear message to 7200c, as demonstrated in Example 7-25.

Example 7-25 PathTear Message from 12008c to 7200c

```
12008c#debug ip rsvp path detail
*Oct 22 12:59:41.185: RSVP:      version:1 flags:0000 type:PTEAR cksum:6CF7 ttl:2
52 reserved:0 length:132
*Oct 22 12:59:41.185: SESSION          type 7 length 16:
*Oct 22 12:59:41.189:  Tun Dest 12.12.12.12 Tun ID 1 Ext Tun ID 4.4.4.4
*Oct 22 12:59:41.189:  HOP              type 1 length 12:
*Oct 22 12:59:41.189:  Hop Addr: 10.0.17.11 LIH: 0x0
*Oct 22 12:59:41.189:  SENDER_TEMPLATE  type 7 length 12:
*Oct 22 12:59:41.189:  Tun Sender: 4.4.4.4, LSP ID: 1520
*Oct 22 12:59:41.189:  SENDER_TSPEC     type 2 length 36:
*Oct 22 12:59:41.189:  version=0, length in words=7
*Oct 22 12:59:41.189:  service id=1, service length=6
*Oct 22 12:59:41.189:  parameter id=127, flags=0, parameter length=5
*Oct 22 12:59:41.189:  average rate=0 bytes/sec, burst depth=1000 bytes
*Oct 22 12:59:41.189:  peak rate  =0 bytes/sec
*Oct 22 12:59:41.189:  min unit=0 bytes, max unit=0 bytes
*Oct 22 12:59:41.189:  ADSPEC           type 2 length 48:
*Oct 22 12:59:41.189:  version=0 length in words=10
*Oct 22 12:59:41.189:  General Parameters break bit=0 service length=8
*Oct 22 12:59:41.189:                               IS Hops:2
*Oct 22 12:59:41.189:                               Minimum Path Bandwidth (bytes/sec):19375000
*Oct 22 12:59:41.189:                               Path Latency (microseconds):0
*Oct 22 12:59:41.189:                               Path MTU:4470
*Oct 22 12:59:41.189:  Controlled Load Service break bit=0 service length=0
*Oct 22 12:59:41.189:
```

But if the LSP were protected because of FRR, this kind of message would have an adverse effect. It would result in the LSP's being torn down even though the LSP were being locally protected by the PLR. To prevent this, the PathTear message needs to be suppressed for primary LSPs that have the "Local Protection Desired" flag on, in spite of the fact that you don't receive Path messages for the protected primary tunnel on the original incoming interface anymore. As long as 12008c receives Path messages belonging to the original RSVP session on any interface, it does not time out and sends a PathTear on its downstream interface. If you recall from the "Upstream Signalling" section, the PLR sends Path messages for all protected tunnels down the protection tunnel.

As the tail of the primary tunnel, 7200c does not know that the protected tunnel has failed unless one of the following things happens:

- It receives an IGP update about the link failure.
- It receives a PathTear from MD 12008c.
- It does not receive an RSVP refresh message (Path) that keeps the session alive within a certain period of time.

This wait on Cisco routers by default is four keepalive periods. (See Chapter 4 for more information on RSVP timers.)

If the tail receives an IGP update about the link failure, it does not take any action from an MPLS TE perspective.

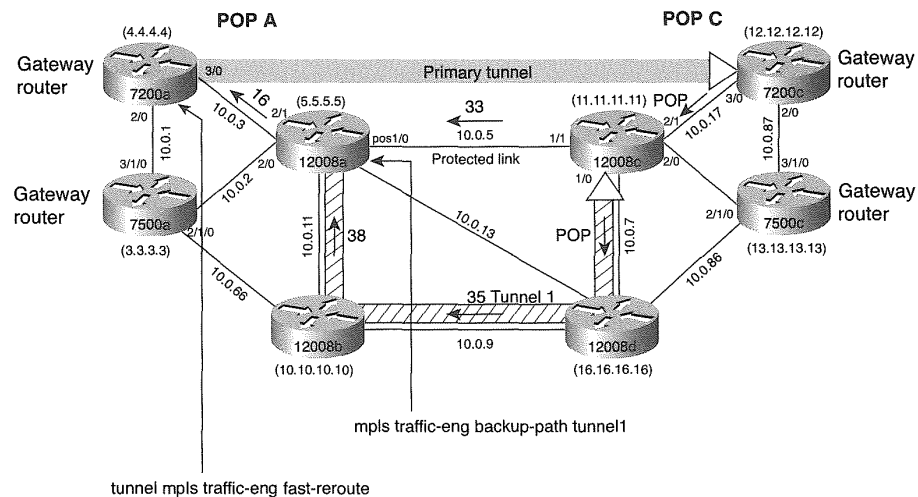
If the RSVP signalling state times out, the LSP is declared dead, and a ResvTear message is sent to the headend. This means that, apart from preventing PathTear from being sent by MP 12008c, you somehow need to make sure that the tail continues to receive the RSVP refresh messages even though one of the links that constituted the primary LSP is now down. This is achieved by making sure that the MP (12008c) continues to receive PATH messages for the primary LSP over the backup tunnel.

Link Protection Configuration Summary

The earlier sections delved into the pieces that make up local protection. The configuration and `show` commands used for each piece were provided and the concepts explained. This section summarizes all the configuration pieces needed to configure link protection.

The sample topology shown in Figure 7-15 explains the link protection configuration. In Figure 7-15, 7200a is the head of the primary tunnel that needs to be protected against failure of the protected link 12008a→12008c. 12008a is the PLR, and 12008c is the MP.

Figure 7-15 Link Protection Configuration



The configuration task for link protection can be broken into the following:

- Headend configuration
- PLR configuration

Headend Configuration

Example 7-26 shows the primary tunnel configuration for 7200a for enabling FRR.

Example 7-26 7200a Tunnel Configuration Before Enabling FRR

```
7200a#show running-config interface tunnel1
interface Tunnel1
description Primary tunnel 7200a->12008a->12008c->7200c
ip unnumbered Loopback0
tunnel destination 12.12.12.12
tunnel mode mpls traffic-eng
tunnel mpls traffic-eng autoroute announce
tunnel mpls traffic-eng path-option 5 explicit name primary
tunnel mpls traffic-eng path-option 6 dynamic
tunnel mpls traffic-eng fast-reroute

7200a#show ip explicit-paths name primary
PATH primary (strict source route, path complete, generation 6)
 1: next-address 10.0.3.5
 2: next-address 10.0.5.11
 3: next-address 10.0.17.12
 4: next-address 12.12.12.12
```

PLR Configuration

The PLR configuration consists of two pieces:

- Building a backup tunnel to NHop
- Configuring the protected interface to use the backup tunnel

Example 7-27 shows what the configuration of the backup tunnel on 12008a looks like.

Example 7-27 12008a Backup Tunnel Configuration

```
12008a#show running-config interface tunnel1
interface Tunnel1
description Link Protection Tunnel (Backup)
ip unnumbered Loopback0
no ip directed-broadcast
tunnel destination 11.11.11.11
tunnel mode mpls traffic-eng
tunnel mpls traffic-eng path-option 5 explicit name nhop

12008a#show ip explicit-paths name nhop
PATH nhop (strict source route, path complete, generation 24)
 1: next-address 10.0.11.10
 2: next-address 10.0.9.16
 3: next-address 10.0.7.11
```

Example 7-28 shows the configuration of protected interface POS 1/0, required for FRR.

Example 7-28 POS 1/0 Configuration After the **backup-path** Statement Is Added

```
12008a#show running-config interface pos 1/0
Building configuration...

Current configuration : 365 bytes
!
interface POS1/0
 ip address 10.0.5.5 255.255.255.0
 no ip directed-broadcast
 mpls label protocol ldp
 mpls traffic-eng tunnels
 mpls traffic-eng backup-path Tunnel1
 tag-switching ip
 crc 32
 clock source internal
 ip rsvp bandwidth 155000 155000
end
```

As you can see, enabling link protection is simple. It turns out that enabling node protection is just as easy, as you will see in the next section.

Node Protection

The earlier section “Node Protection Overview” provided you with a basic understanding of how node protection works. This section focuses on the details of node protection, particularly the areas in which node protection deviates from link protection.

Similarities Between Link Protection and Node Protection

Node protection and link protection share a few common characteristics and differ on a few others. The similarities are as follows:

- Enabling FRR at the primary tunnel headend
- Tying the protected link to the backup tunnel
- Failure detection
- Connectivity restoration
- Post-failure signalling

In node protection, although you are trying to protect against the failure of an NHop, the way you detect the failure is configured on the link connecting the PLR to the NHop, just as you would in case of link protection.

Differences Between Link Protection and Node Protection

This section examines the areas in which node protection differs from link protection:

- NNHop backup tunnel is configured at the PLR.
- Label recording is required for the NNHop backup tunnel.
- NNHop tunnel handles link and node failures.

NNHop Backup Tunnel Is Configured at the PLR

Creating a backup tunnel to the NNHop is similar to creating a backup tunnel to the NHop, except that an NNHop tunnel terminates on the NNHop instead of on the NHop. You have to build the backup tunnel to the NNHop instead of the NHop to protect against the failure of the NHop node. Example 7-29 shows the configuration of the NNHop backup tunnel on 12008a. This is depicted in Figure 7-16.

Example 7-29 NNHop Backup Tunnel for Node Protection

```
12008a#show running-config interface tunnel2
interface Tunnel2
  ip unnumbered Loopback0
  no ip directed-broadcast
  tunnel destination 12.12.12.12
  tunnel mode mpls traffic-eng
  tunnel mpls traffic-eng path-option 5 explicit name nnhop
  ip rsvp signalling dscp 0
end

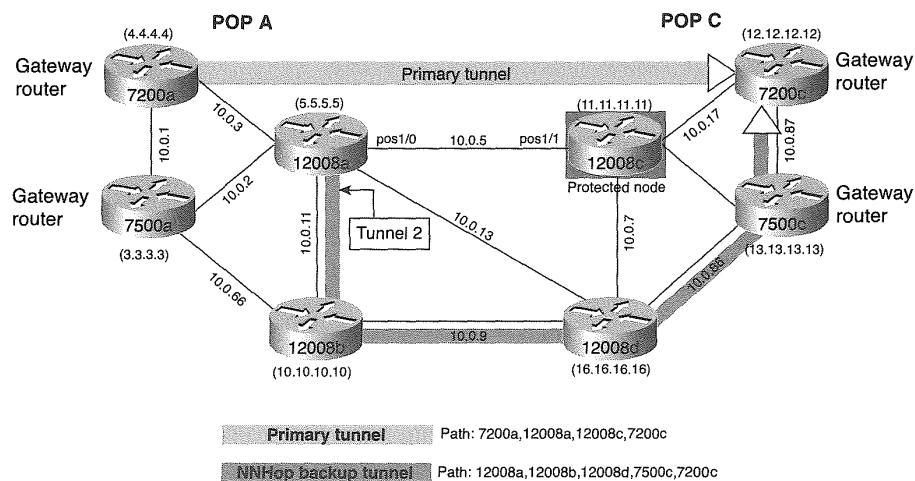
12008a#show ip explicit-paths name nnhop
ip explicit-path name nnhop enable
next-address 10.0.11.10
next-address 10.0.9.16
next-address 10.0.86.13
next-address 10.0.87.12
next-address 12.12.12.12
```

The hops specified in the explicit path list **nnhop** are as follows:

- **10.0.11.10**—The IP address of 12008b on link 12008a→12008b
- **10.0.9.16**—The IP address of 12008d on link 12008b→12008d
- **10.0.86.13**—The IP address of 7500c on link 12008d→7500c
- **10.0.87.12**—The IP address of 7200c on link 7500c→7200c

As you can see from Example 7-29, the destination of the tunnel is 12.12.12.12, which is the RID of 7200c, the NNHop node.

Figure 7-16 Node Protection



Label Recording Is Required for the NNHop Backup Tunnel

The output in Example 7-30 shows the **Label Recording** flag turned on. This flag is turned on whenever the headend configures **tunnel mpls traffic-eng fast-reroute**.

Example 7-30 debug ip path resv detail Output on 7200a Showing Label Recording

```
7200a#debug ip path resv detail
*Oct 16 14:40:57.124: SESSION_ATTRIBUTE type 7 length 24:
*Oct 16 14:40:57.124: Setup Prio: 7, Holding Prio: 7
*Oct 16 14:40:57.124: Flags: Local Prot desired, Label Recording, SE Style
*Oct 16 14:40:57.124: Session Name: 7200a_t1
```

When you build a tunnel to the NHRP because you are this node's upstream neighbor, you know what label the downstream neighbor expects because you received the label from the RSVP Resv message along the reservation. However, if you want to build a backup tunnel to the NNHop node, you do not know what label this node expects for a given LSP.

This is what the Label Recording flag is for. It records the incoming label used at each hop for an LSP so that a PLR doing node protection knows what label to use on a protected LSP when it's switched down the backup tunnel. Here is a quote from RFC 3902:

The recording of the Label subobject in the ROUTE_RECORD object is controlled by the label-recording-desired flag in the SESSION_ATTRIBUTE object. Since the Label subobject is not needed for all applications, it is not automatically recorded. The flag allows applications to request this only when needed.

Also, if you examine the output of `show mpls traffic-eng tunnel tunnel1` on 7200a, you'll see this:

```
Record Route: 10.0.5.5(16) 10.0.17.11(33)
              10.0.17.12(0)
```

The numbers 16, 33, and 0 in parentheses refer to the label values from the record label object.

Having this information available is also useful during troubleshooting.

Note that requesting label recording is controlled by the tunnel headend. Because the headend doesn't know whether the network has link protection or node protection, label recording is turned on but is not used if the network has no node protection.

NNHop Tunnel Handles Link and Node Failures

If you consider Figure 7-16 again, the backup tunnel bypasses both the NHop and the link that connects the PLR to the NHop. Even though the tunnel is primarily in place to protect against the failure of 12008c, the protected node, if the link 12008a→12008c fails, there is no way for the PLR (12008a) to tell whether the link went down because link itself went down or the link went down because the entire NHop node went down.

If you used an NHop backup tunnel, it would only guard against link failures. With NNHop backup tunnels, the moment a failure is detected on the POS 1/0 interface on 12008a, protection kicks in, sending all the packets of the primary tunnel over the backup tunnel. As you can see, node protection is both link and node protection rolled into one.

Advanced Protection Issues

The previous sections described how link and node protection work. This section covers some advanced FRR features that you can configure to make the protection you are using more effective. Here are the topics covered:

- Multiple backup tunnels to the same MP
- Backup bandwidth reservation
- Backup tunnel selection
- Promotion

Multiple Backup Tunnels

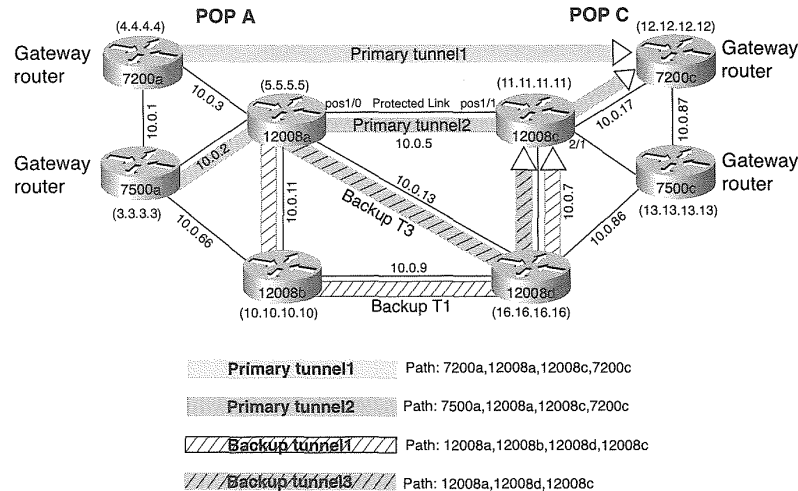
There are two types of multiple backup tunnels:

- Multiple backup tunnels to the same MP
- NHop versus NNHop backup tunnels

Multiple Backup Tunnels to the Same MP

Figure 7-17 shows a link that is protected by backup tunnel1 and backup tunnel2. This link carries two primary tunnels—tunnel1 and tunnel2.

Figure 7-17 Multiple Backup Tunnels to the Same MP



Suppose that the protected link—12008a→12008c in Figure 7-17—is an OC-12 link, but the other two downstream links from 12008a to 12008b and 12008d are only OC-3s. Suppose further that two LSPs exist that traverse the protected link: one from 7200a (Primary Tunnel1 in Figure 7-17) and one from 7500a (primary tunnel2 in Figure 7-17). Each LSP has a reservation of 150 Mbps, and both of them have FRR configured at the headend. If the protected link fails, it is not possible to offload both these LSPs onto a single backup tunnel without incurring traffic loss, because the two downstream links from 12008a over which you can build backup tunnels are both OC-3s. What can you do?

You can configure two backup tunnels from 12008a to 12008c—backup tunnel1 and backup tunnel3—such that each backup tunnel goes over diverse paths that have OC-3 worth of bandwidth. This protects the two primary tunnels without dropping traffic, as shown in Figure 7-17.

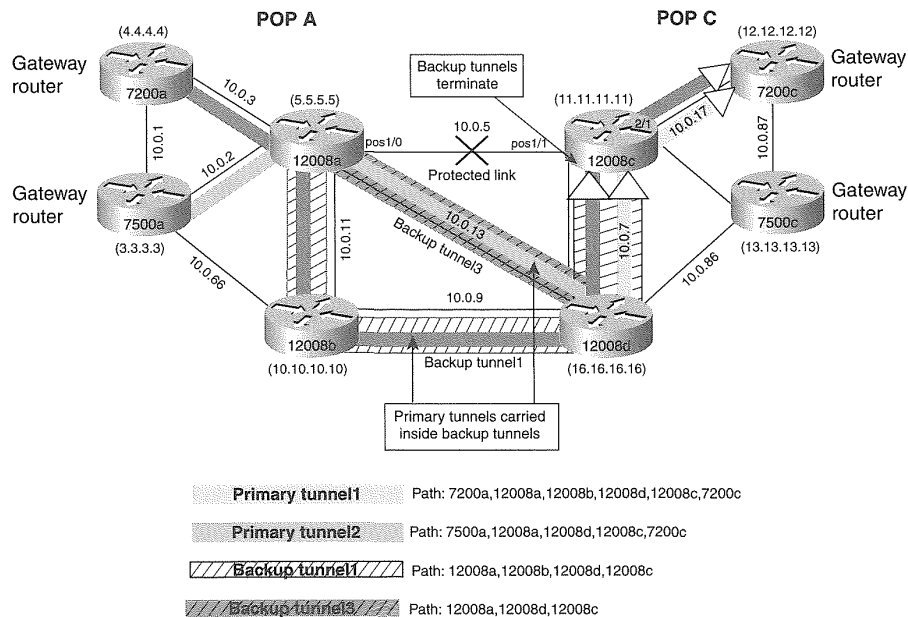
With multiple backup tunnels to the same MP, when the protected link or node fails, the traffic from primary tunnel1 and primary tunnel2 is assigned to these backup tunnels. This can be observed using the `show mpls traffic-eng fast-reroute database` command, as demonstrated in Example 7-31, which shows the FRR status on 12008a.

Example 7-31 FRR Status Information with Multiple Backup Tunnels

```
12008a#show mpls traffic-eng fast-reroute database
```

Tunnel head fast reroute information:						
Prefix	Tunnel	In-label	Out intf/label	FRR intf/label	Status	
LSP midpoint frr information:						
LSP identifier		In-label	Out intf/label	FRR intf/label	Status	
4.4.4.4 1 [1520]		16	PO1/0:33	Tu1:33	ready	
3.3.3.3 1 [902]		28	PO1/0:12319	Tu3: 19	ready	

The first line of output in Example 7-31 corresponds to Primary Tunnel1 (7200a), and the second one corresponds to primary tunnel2 (7500a). The out interface is POS1/0—the protected link/interface in both cases. Figure 7-18 illustrates the actual forwarding of the primary LSPs after failure.

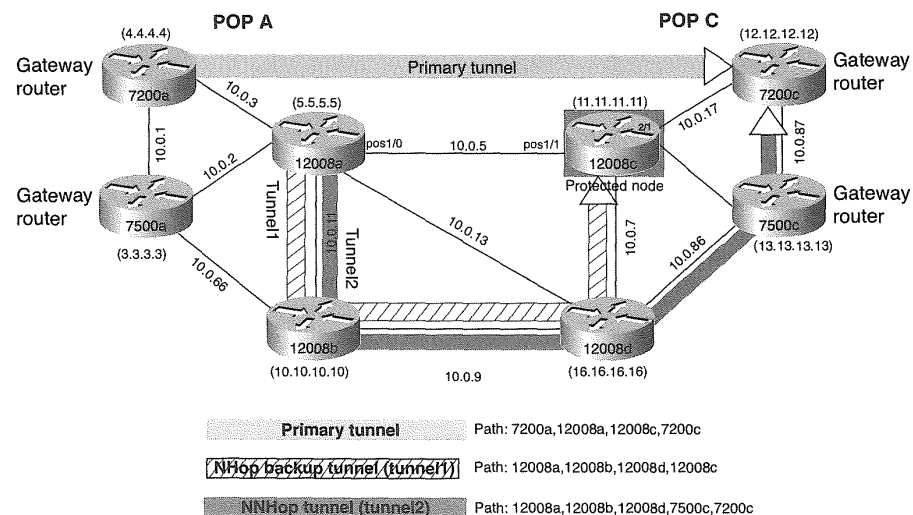
Figure 7-18 Load Balancing of Two Primary LSPs

NOTE

Parallel backup tunnels are meaningful only when they are destined for the same MP. Traffic belonging to a single primary tunnel is always carried on the same backup tunnel during failure. It is not split over multiple backup tunnels, even if they are configured. This is not CEF's load balancing, but instead merely a distribution of protected tunnels among protection tunnels. The later section "Backup Tunnel Selection Summary" covers this assignment in more detail.

NHop Versus NNHop Backup Tunnels

An NHop backup tunnel protects against failure of the link from the PLR to the NHop node, whereas an NNHop backup tunnel protects against both NHop node failure and the link connecting the PLR to the NHop. Hence, if both are configured, the PLR chooses the NNHop backup tunnel over the NHop backup tunnel when assigning protected LSPs to backup tunnels. Figure 7-19 depicts two backup tunnels configured on PLR 12008a—one going to NHop and the other going to NNHop.

Figure 7-19 *NHop Versus NNHop Backup Tunnels*

Example 7-32 shows that both primary LSPs are backed up using NNHop tunnel2, even though two backup tunnels are available—one to the NHop and the second to the NNHop. This can be observed by executing the `show mpls traffic-eng fast-reroute database` command.

Example 7-32 *NNHop Tunnel2 Being Used for Protection*

```

12008a#show mpls traffic-eng fast-reroute database
Tunnel head fast reroute information:
Prefix          Tunnel    In-label Out intf/label  FRR intf/label  Status
LSP midpoint frr information:
LSP identifier   In-label Out intf/label  FRR intf/label  Status
4.4.4.4 1 [1520]    16      P01/0:Pop tag   Tu2:tag-implicit ready
3.3.3.3 1 [902]     28      P01/0:19        Tu2:tag-implicit ready

```

Backup Bandwidth Reservation

The preceding section discussed the motivation for having multiple backup tunnels. This section discusses another FRR feature—*bandwidth* protection rather than just *link* (or node) protection. The idea is that, rather than placing all protected tunnels down a protection tunnel that might end up traversing a link that could be congested during failure, you can build bandwidth-aware protection tunnels. This is a fairly complex mechanism to employ on a large scale, but it gives you the best possible protection for your network.

Two new terms are introduced:

- Limited backup bandwidth
- Unlimited backup bandwidth

Limited backup bandwidth, as you might suspect, puts a limit on how much bandwidth can be allocated from a backup tunnel. It is a number that you need to configure, as shown in Example 7-33.

Example 7-33 *Backup Bandwidth Configuration for Global Pools and Subpools*

```

12008a(config-if)#tunnel mpls traffic-eng backup-bw ?
<1-4294967295> Amount of allocatable backup bw, any lsp may use
global-pool    Enter Allocatable amount for Global-Pool bw
sub-pool       Enter Allocatable amount for Sub-Pool bw

```

The limit can be a number between 1 and 4,294,967,295 specified in terms of kbps. The keywords **Global-Pool** and **Sub-Pool** refer to the dual pools created for DiffServ-Aware Traffic Engineering (DS-TE) (if you recall from Chapter 6, “Quality of Service with MPLS TE”). Again, for each of the pools, you can either limit the backup bandwidth using the configuration shown in Example 7-34, or you can use the keyword **unlimited**. If you enter a backup bandwidth without specifying the pool, the bandwidth protection offered by this backup tunnel is available to both types of primary LSPs. If you specify the pool type on the backup tunnel, only primary LSPs of that pool type can use the backup tunnel.

Example 7-34 *Configuring Limited Versus Unlimited Backup Bandwidth for the DS-TE Global Pool*

```

12008a(config-if)#tunnel mpls traffic-eng backup-bw global-pool ?
<1-4294967295> Amount of allocatable backup bw, only global-pool may use
Unlimited       Unlimited amount for Global-Pool bw

```

Unlimited backup bandwidth, as the name suggests, has no limits on how much backup bandwidth can be allocated from the backup tunnel. The same terms apply to subpools as well. Example 7-35 shows the configuration for unlimited backup bandwidth.

Example 7-35 *Configuring Limited Versus Unlimited Backup Bandwidth for the DS-TE Subpool*

```
12008a(config-if)#tunnel mpls traffic-eng backup-bw sub-pool ?
<1-4294967295> Amount of allocatable backup bw, only sub-pool may use
Unlimited      Unlimited amount for Sub-Pool bw
```

If you refer to Table 7-3 (in the next section), you can see that, if multiple backup tunnels are configured at a PLR, the backup tunnel selection process chooses a backup tunnel with *limited* backup bandwidth over one that has *unlimited* bandwidth configured. Among the tunnels that have *limited* bandwidth configured, the ones that terminate at NNHop are preferred over ones that terminate on NHop. It also prefers tunnels with subpool reservation over ones that have global pool reservation.

NOTE

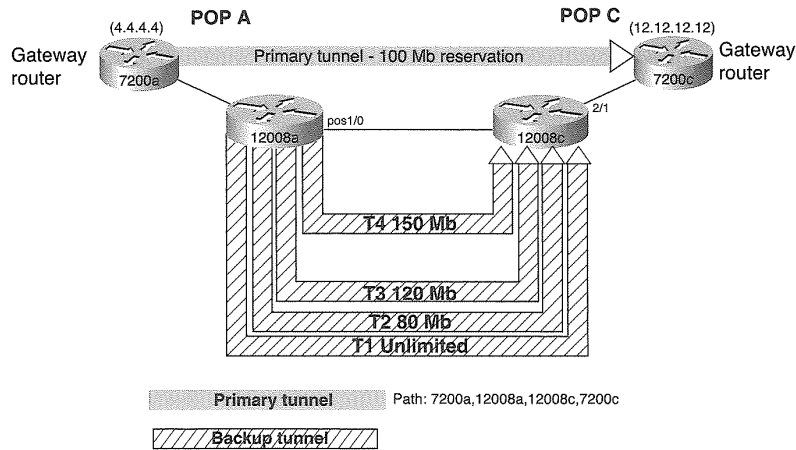
It is important to understand that backup bandwidth is only an associated bandwidth. It is not actually signalled. In other words, this scheme does not require the backup tunnels to successfully reserve the amount of bandwidth they offer (by configuring the backup-bw on the backup tunnels). You can configure the backup tunnel to reserve bandwidth if you want (using the **tunnel mpls traffic-eng bandwidth** command), but reserving bandwidth is a separate issue from protecting bandwidth.

Consider the example shown in Figure 7-20. The primary LSP has 100 Mbps of bandwidth reserved (from either pool), and four backup tunnels terminate at the same destination (12008c—PLR). Backup tunnel T1 has been configured with *unlimited* backup bandwidth and three tunnels (T2, T3, and T4) with *limited* backup bandwidth—80 Mbps, 120 Mbps, and 150 Mbps, respectively.

During the backup selection process, tunnel T1 is eliminated because other *limited* backup tunnels are going to the same destination that fit the demand. Tunnel T2 is eliminated because it has 80 Mbps and does not fulfill the bandwidth requirement of the primary LSP (100 Mbps). Both tunnel T3 and T4 meet the bandwidth requirement of the primary LSP. Backup tunnel T3 is selected because it has the least leftover bandwidth. After the primary LSP is assigned to backup tunnel T3, the backup bandwidth on tunnel T3 is reduced by 100 Mbps, so it now has a remaining protection capacity of 20 Mbps.

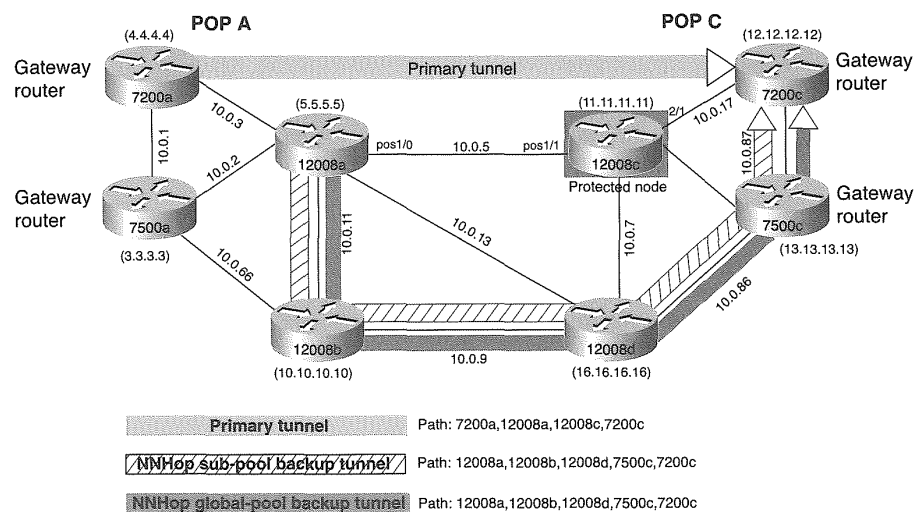
If backup tunnel T4 were selected, 50 Mbps of bandwidth would be wasted because it is not required by the primary LSP. If tunnel T3 is selected, only 20 Mbps is wasted, thus making it the best fit. Assignment is done this way in an attempt to limit backup bandwidth fragmentation.

Figure 7-20 Backup Tunnel Selection

**NOTE**

In the case of limited backup bandwidth, if more than one tunnel meets the bandwidth requirement, you choose the one that has the least *leftover* bandwidth. For unlimited, if you have more than one unlimited tunnel to choose from, you choose the one that has the least *assigned*.

Each primary LSP can reserve either global (best-effort) or subpool (premium/low-latency) bandwidth. As mentioned earlier, you can designate the pool type for the backup bandwidth—subpool to backup subpool primary LSPs, global-pool to backup global-pool primary LSPs and *any* pool (if you don't select either global or subpool). This means that the PLR tries to find a subpool backup tunnel for a subpool primary LSP with sufficient subpool bandwidth to cover the needs of that primary LSP. If such a subpool backup tunnel is not available, it tries to use the *any* pool (but not the global-pool tunnels). The same applies for global-pool backup tunnel selection. If a global-pool LSP needs backup, the PLR looks for a global-pool backup tunnel that meets its need. If it cannot find one, the PLR looks for an *any* pool tunnel but not subpool backup tunnels. Figure 7-21 shows two backup tunnels from the PLR (12008a) to the MP (7200c). One is a subpool backup tunnel and the other is a global-pool backup tunnel. Which one is used to backup the primary tunnel depends on the type of bandwidth being reserved by the primary tunnel.

Figure 7-21 Global Versus Subpool NNHop Backup Tunnels

Backup Tunnel Selection Summary

When multiple backup tunnels are configured, keeping track of which backup the PLR chooses can be confusing. The process of backup tunnel selection is covered in Table 7-3 and Table 7-4. The first column shows, from best to worst, what the PLR prefers. For example, an NNHop subpool tunnel with limited bandwidth is the best, and the NHop backup tunnel with unlimited bandwidth configuration is the worst.

Table 7-3 Tunnel Selection for Subpool Primary LSPs

Preference	Backup Tunnel Destination	Bandwidth Pool	Bandwidth Amount
Best	NNHop	Subpool	Limited
	NNHop	Subpool	Unlimited
	NNHop	Any	Limited
	NNHop	Any	Unlimited
	NHop	Subpool	Limited
	NHop	Subpool	Unlimited
	NHop	Any	Limited
Worst	NHop	Any	Unlimited

Table 7-4 Tunnel Selection for Global-Pool Primary LSPs

Preference	Backup Tunnel Destination	Bandwidth Pool	Bandwidth Amount
Best	NNHop	Global-pool	Limited
	NNHop	Global-pool	Unlimited
	NNHop	Any	Limited
	NNHop	Any	Unlimited
	NHop	Global-pool	Limited
	NHop	Global-pool	Unlimited
	NHop	Any	Limited
Worst	NHop	Any	Unlimited

NOTE

Complete details on configuration and the enhancements that were implemented for node protection are available at the Cisco web site at www.cisco.com. You can obtain information relevant to node protection enhancement by searching for "Fast Reroute enhancement."

Promotion

Because multiple backup tunnels can now be configured, a backup LSP that was chosen as a result of the selection process described earlier might no longer be the best due to a change in the environment, such as the configuration of a new backup tunnel or a primary LSP's resource requirement changing. To address this issue, the PLR performs reoptimization periodically. This periodic reoptimization is called *promotion*. It is done every five minutes on the PLRs by default. You can change this default using the **mpls traffic-eng fast-reroute timers** command. Setting the value of this timer to 0 disables promotion.

You can observe when the next promotion is due by using **show mpls traffic-eng tunnels summary**, as demonstrated in Example 7-36.

Example 7-36 Observing LSP Promotion Periods

```
12008a#show mpls traffic-eng tunnels summary
Signalling Summary:
  LSP Tunnels Process:      running
  RSVP Process:             running
  Forwarding:               enabled
  Head: 3 interfaces, 2 active signalling attempts, 2 established
        10 activations, 8 deactivations
```

continues

Example 7-36 *Observing LSP Promotion Periods (Continued)*

Midpoints:	2, Tails: 0
Periodic reoptimization:	every 3600 seconds, next in 3367 seconds
Periodic fastreroute:	every 300 seconds, next in 67 seconds
Periodic auto-bw collection:	every 300 seconds, next in 67 seconds

Configuring Multiple Backup Tunnels to Multiple NNHops

When doing link protection, there is, of course, only one MP because there's only one neighbor—at least on point-to-point links. However, in the case of node protection (or link protection of a multipoint subnet, such as an Ethernet subnet), you might need to configure multiple backup tunnels to different nodes. Because multipoint Ethernet segments are rare in a network core, let's consider node protection.

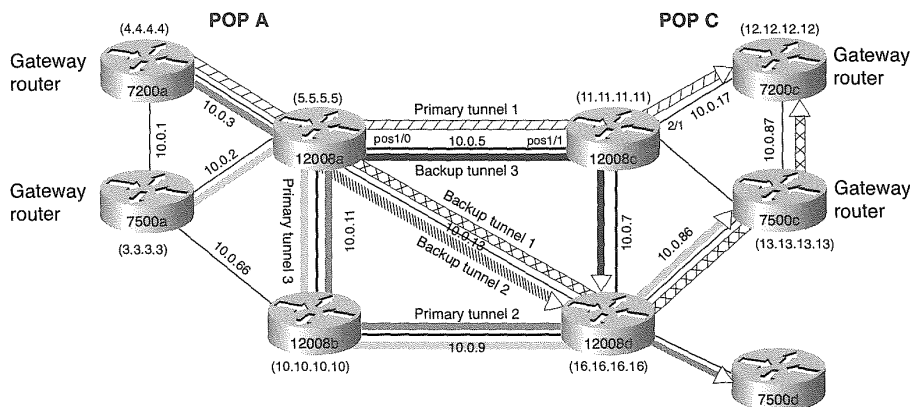
If you want to configure node protection on a PLR, and the NHop across that link has three neighbors of its own, the PLR needs at least three NNHop tunnels—one to each of the NNHops. Of course, you can use more than one backup tunnel to the same NNHop if you want.

Figure 7-22 illustrates multiple backup tunnels to different NNHops. Three primary tunnels go from 7200a and 7500a to 7200c, 7500c, and 7500d. These are labeled primary tunnel 1, primary tunnel 2, and primary tunnel 3. The PLR is 12008a. The NNHops that the PLR is trying to protect are 12008c, 12008b, and 12008d. Three backup tunnels—backup tunnel 1, backup tunnel 2, and backup tunnel 3—start at 12008a. Backup tunnel 1 terminates on 7200c, backup tunnel 2 terminates on 12008d, and backup tunnel 3 terminates on 12008c also. The rule you have to keep in mind when building these backup tunnels is that they need to

- Terminate on the PLR's NNHop node
- Avoid signalling the backup tunnel through the protected NHop

Notice that all the backup tunnels meet these criteria.

Figure 7-22 *Multiple Backup Tunnels to Multiple NNHops*



Multiple backup tunnels can be configured as shown in Example 7-37.

Example 7-37 Multiple Backup Tunnel Configuration

```
12008a#show running-config interface pos 1/0
interface POS1/0
 ip address 10.0.5.5 255.255.255.0
 no ip directed-broadcast
 mpls label protocol ldp
 mpls traffic-eng tunnels
 mpls traffic-eng backup-path Tunnel1
 mpls traffic-eng backup-path Tunnel2
 mpls traffic-eng backup-path Tunnel3
 crc 32
 clock source internal
 pos ais-shut
 ip rsvp bandwidth 155000 155000
 end
```

Adding the second and third backup paths is no different from adding the first one. Example 7-38 shows the configuration for all the backup tunnels on 12008a.

Example 7-38 Configuration for All the Backup Tunnels on 12008a

```
12008a#show running-config interface tunnel1
interface Tunnel1
 description Link Protection Tunnel (Backup)
 ip unnumbered Loopback0
 no ip directed-broadcast
 tunnel destination 12.12.12.12
 tunnel mode mpls traffic-eng
 tunnel mpls traffic-eng backup-bw global-pool unlimited
 tunnel mpls traffic-eng path-option 5 explicit name backup1
end
12008a#show running-config interface tunnel2
interface Tunnel2
 ip unnumbered Loopback0
 no ip directed-broadcast
 tunnel destination 13.13.13.13
 tunnel mode mpls traffic-eng
 tunnel mpls traffic-eng backup-bw global-pool unlimited
 tunnel mpls traffic-eng path-option 5 explicit name backup2
end
12008a#show running-config interface tunnel3
interface Tunnel3
 ip unnumbered Loopback0
 no ip directed-broadcast
 tunnel destination 14.14.14.14
 tunnel mode mpls traffic-eng
 tunnel mpls traffic-eng backup-bw global-pool unlimited
 tunnel mpls traffic-eng path-option 5 explicit name backup3
end
12008a#show ip explicit paths
ip explicit-path name backup1 enable
```

continues

Example 7-38 *Configuration for All the Backup Tunnels on 12008a (Continued)*

```
next-address 10.0.13.16
next-address 10.0.87.13
next-address 10.0.86.12
next-address 12.12.12.12

ip explicit-path name backup2 enable
next-address 10.0.13.16
next-address 16.16.16.16

ip explicit-path name backup3 enable
next-address 10.0.5.7
next-address 10.0.7.16
next-address 16.16.16.16
```

As you can see, node protection configuration is similar to link protection. The only difference is in the protection tunnel (or tunnels) on the protected interface—how many there are and where they terminate.

Summary

Both IP and MPLS TE reroute if there's a failure somewhere in the network. However, IGP reroute is often not quick enough for real-time applications, and the MPLS TE headend reroute is, if anything, slower than IGP reroute because there's more work to be done.

Real-time applications might require subsecond convergence. FRR is the way to achieve this. Loss is minimized because of the ability to use local failure detection and repair. In the absence of local failure detection and repair, signalling propagation delay might result in packet loss that is unsuitable for real-time applications.

There are two types of local protection: link protection and node protection. Link protection can help only when the link that is downstream of the PLR fails. Node protection protects both the link and the node that are immediately downstream of the PLR.

Local repair is highly scalable because multiple LSPs are protected by a single backup tunnel, as opposed to path protection, which needs one backup tunnel per LSP.

Chapter 9 discusses design issues related to protection.