IETF Draft
Multi-Protocol Label Switching

Ken Owens
Erlang Technology, Inc.

Expires: January 2002

Vishal Sharma
Metanoia, Inc.

Srinivas Makam
Ben Mack-Crane
Tellabs Operations, Inc.

Changcheng Huang
Carleton University

July 2001

A Path Protection/Restoration Mechanism for MPLS Networks

<draft-chang-mpls-path-protection-03.txt>

Status of this memo

Abstract

It is expected that MPLS-based recovery could become a viable option for obtaining faster restoration than layer 3 rerouting. To deliver reliable service, however, multi-protocol label switching (MPLS)[ ], [ ] requires a set of procedures to provide protection of the traffic carried on the label switched paths (LSPs). This imposes certain requirements on the path recovery process [ ], and requires

procedures for the configuration of working and protection paths, for the communication of fault information to appropriate switching elements, and for the activation of appropriate switchover actions. This document specifies a mechanism for path protection switching and restoration in MPLS networks.

Table of Contents

Page

1.0 Introduction

With the migration of real-time and high-priority traffic to IP networks, and with the need for IP networks to increasingly carry mission-critical business data, network survivability has become critical for future IP networks. Current routing algorithms, despite being robust and survivable, can take a substantial amount of time, to recover from a failure, on the order of several seconds to minutes, which can cause serious disruption of service in the interim. This is unacceptable for many applications that require a highly reliable service, and has motivated network providers to give serious consideration to the issue of network survivability.

Path-oriented technologies, such as MPLS, can be used to support advanced survivability requirements and enhance the reliability of IP networks. Different from legacy IP networks, MPLS networks establish label switched paths (LSPs), where packets with the same label follow the same path. This potentially allows MPLS networks to pre-establish protection LSPs for working LSPs, and achieve better protection switching times than those in legacy IP networks. With this objective in mind, the present contribution describes a mechanism to protect paths (or path segments) in MPLS networks. Before discussing the specifics of this contribution, we first outline the major components of a path protection solution, and point out those that are within the scope of this document. A complete solution for path protection requires the following components:

(i) A method for selecting the working and protection paths.

(ii) A method for signaling the setup of the working and protection paths.

(iii) A fault detection mechanism to detect faults along a path.

(iv) A fault notification mechanism, to convey information about the occurrence of a fault to a network entity responsible for reacting to the fault and taking appropriate corrective action.

(v) A switchover mechanism to move traffic over from the working path to the protection path.

(vi) A repair detection mechanism, to detect that a fault along a path has been repaired.

(vii) An (optional) switchback or restoration mechanism, for switching traffic back to the original working path, once it is discovered that the fault has corrected or has been repaired.

Observe that component (i) consists of algorithms and techniques that are used to select the working and protection paths based on specific criteria, established via policy or other constraints, and can be proprietary. It is therefore not subject to standardization, and is outside the scope of this draft. Therefore, the protection mechanism described later assumes that the working and protection paths are known to the LSR responsible for path setup, and are either communicated to it or are calculated by some intelligence at that LSR. Component (ii), which involves establishing the working and protection paths via signaling, is within the scope of the draft, and is discussed in Section 3.1.

A detailed specification of fault detection mechanisms is outside the scope of this draft, but the specification of how the path protection mechanism works with different fault detection methods is in scope, and is discussed in Section 5. In particular, we consider how the mechanism works for two practical cases of interest: (a) when only the end node of a path is responsible for detecting faults, and (b) when all the nodes along the path are responsible for detecting faults. The main focus of this draft is the specification of an efficient fault notification mechanism that

takes LSP merging into account (irrespective of whether they are physically or virtually merged). Switchover and switchback mechanisms also are also within the scope of the draft, but component (vi) is outside the scope of the draft, so the draft does not specify the details of the mechanisms used to detect that a fault has been repaired.

2.0 Motivation and Purpose

The framework document [3] lays out the various options for MPLS-based restoration/recovery. However, candidate approaches corresponding to various viable recovery options are still needed. Our work on proposing a path protection mechanism for MPLS networks is motivated by the belief that path protection (in conjunction with local repair) will be needed for truly reliable network operation. The purpose of this contribution is to propose a path protection mechanism that is:
(i) fast (compared to Layer 3, with the goal of being comparable to SONET),
(ii) scalable,
(iii)bandwidth efficient,
(iv)allows for path merging (i.e., is merging compatible), and
(v) works at the MPLS layer (that is, only uses knowledge that is commonly available to MPLS routing and signaling modules).

The major differences between this 02 version and the previous 01 version are:

-- Protection domain configuration details

-- Protection domain configuration information elements added

3.0 Key Features of the Proposed Mechanism

This contribution describes an MPLS-based path recovery mechanism that can facilitate fast protection switching. The mechanism currently supports 1:1 protection [3].
Bypass tunneling is for further study. First, because tunnel setup itself is not adequately defined yet, and second, because even assuming a tunnel could be setup, in the presence of tunnels (or tunneled segments) the mechanism still requires the ability to setup bi-directional tunnels, which is not defined yet. The mechanism has several timers to enable it to inter-work with protection mechanisms at other layers. Some of the key features of the protection mechanism are:

-- Special tree structure to efficiently distribute fault and/or recovery information.

Existing published proposals for MPLS recovery have not addressed the issue of fault notification in detail. Specifically, none of

these proposals has discussed how to perform fault notification for the label merging case. In this draft, we propose a new fault notification structure called the reverse notification tree (RNT), which makes fault notification efficient and scalable (we provide details of the RNT in subsequent sections).

-- Lightweight notification mechanism.

The lack of MPLS OAM functionality requires the definition of a lighweight stateless notification mechanism. Reliable transport mechanisms, such as TCP, are typically state-oriented and therefore difficult to scale. It is also very difficult to support point-to-multipoint communications based on reliable transport mechanisms. In our scheme, therefore, we use a stateless notification mechanism to achieve scalability. The notification is based on the transmission of packets that are sent periodically until the nodes responsible for switchover learn of the fault. Since no acknowledgements or handshaking between adjacent nodes is needed, the mechanism works only with timers and does not require the maintenance of state.

--Minimize delays of a recovery cycle.

An objective of the mechanism proposed in this draft is to minimize the duration of the recovery cycle. Thus a stateless transport mechanism together with high priority for control traffic minimizes notification delay. Likewise, a simple label merging approach to handle the traffic arriving on the working and protection paths eliminates the need for synchronization (or handshaking) between the LSRs at the two ends of a recovery path.

-- Work at the MPLS layer (that is, use information available to the MPLS signaling and routing modules at the nodes)

The mechanism is designed to operate using only MPLS constructs and the knowledge available to the MPLS modules at the nodes. Therefore, the mechanism assumes, by default, that the working and protection paths merge at a path merge LSR (PML) within the domain under consideration. However, since the mechanism does not depend on the path selection method, it also works in settings where a PML does not exist, and a path selection algorithm (outside the scope of this I-D) determines that the working and protection paths must terminate at different egress LSRs. Note, however, that for the path selection mechanism to be able to make this determination, it may need knowledge beyond that which is commonly available to the MPLS modules at a node. This is because determining whether a working path can be protected by another path with a different egress LSR requires Layer 3 knowledge to ascertain whether the LSR terminating the recovery path is acceptable. In the remainder of this document, we will focus on the PML case, with the understanding that the non-PML case is also supported.

In addition to the key features outlined above, some other characteristics of the mechanism are:

-- A liveness message to detect faults.

Although fault detection is outside the scope of this draft, we will allow the existence of a generic ''liveness'' message that can complement any fault detection mechanism. This liveness message may, for example, be provided as part of an user/control plane OAM function, or by an existing Hello message (as the RSVP "Hello" message) with an appropriately set timer value. We do not define specific liveness mechanisms in this draft, deferring instead to work on OAM in MPLS, which is where we expect such a liveness message to be defined.

Our assumption is that faults fall into different classes, and that different faults may be detected and corrected by different layers. Some faults (for example, the loss of signal or transmitter faults) may be detected and corrected by lower layer mechanisms (such as SONET), while others (for example, failure of the reverse link) may be detected (but may not be corrected) by lower layers and may be communicated to the MPLS layer. Still other faults (such as node failures or faults on the reverse link) may not be detected by lower layers, and will have to be detected and corrected at the MPLS layer. Therefore, we adopt the liveness message as a complementary fault detection mechanism.

We note that in this draft we confine our discussion of protection to a single MPLS domain, and do not consider protection/recovery across multiple MPLS domains or across multiple administrative boundaries. We note, however, that protection mechanisms in different domains may be concatenated, and that (at least initially) these mechanisms may work autonomously, across the (possibly) multiple points of attachment between two adjacent domains. However, coordination of protection mechanisms across multiple domains or across multiple transport technologies is currently out of the scope of this document.

4.0 Core MPLS Path Protection Components

This document assumes the terminology given in[1], [2], [3] , and introduces some additional terms. For the convenience of the reader, we repeat here some of the terminology from earlier documents.

Working Path
The protected path that carries traffic before the occurrence of a fault. The working path is the part of the LSP between the PSL and the PML (if any) or, in the absence of a PML, between the PSL and an egress LSR. A working path is denoted by the sequence of LSRs through which it passes. For example, in Fig. 1, the working path that starts at LSR 1 and terminates at LSR 7 is denoted by (1-2-3-4-

6-7).

Recovery Path
The path by which traffic is restored after the occurrence of a fault. In other words, the path along which traffic is directed by the recovery mechanism. The recovery path can either be an equivalent recovery path and ensure no reduction in quality of service or be a limited recovery path and thereby not guarantee the same quality of service (or some other criteria of performance) as the working path. A recovery path is also denoted by the sequence of LSRs through which it passes. Again, in Fig. 1, the recovery path that starts at LSR 1 and terminates at LSR 7 is denoted by (1-5-7).

Path Switch LSR (PSL)
An LSR that is the transmitter of both the working path traffic and its corresponding recovery path traffic. The PSL is responsible for switching of the traffic between the working path and the recovery path. The PSL is the origin of the recovery traffic, but may or may not be the origin of the working traffic (that is the working path may be transiting the PSL).

Path Merge LSR (PML)

An LSR that receives both working path traffic and its corresponding recovery path traffic, and either merges their traffic into a single outgoing path, or, it is itself the destination, passes the traffic on to the higher layer protocols. The PML is the destination of the recovery path but may or may not be the destination of the working path.

Intermediate LSR
An LSR on a working or recovery path that is neither a PSL nor a PML for that path.

FIS (Fault Indication Signal)
A signal that indicates that a fault along a path has occurred. It is relayed by each intermediate LSR to its upstream or downstream neighbor, until it reaches an LSR that is set up to perform MPLS recovery.

FRS (Fault Recovery Signal)
A signal that indicates that a fault along a path has been repaired. Like the FIS, it is relayed by each intermediate LSR to its upstream or downstream neighbor, until it reaches an LSR that performs a switchback to the path for which the FIS was received.

Liveness Message
A generic name for any message exchanged periodically between two adjacent LSRs that serves as a link probing mechanism. It provides an integrity check of the forward and backward directions of the link between the two LSRs as well as a check of neighbor liveness.

Path Continuity Test
A test that verifies the integrity and continuity of a path or a path segment. The details of such a test are beyond the scope of this draft. (This could be accomplished, for example, by sending a control message along the same links and nodes as those traversed by the data traffic.)

4.1 Reverse Notification Tree

Since LSPs are unidirectional entities and recovery requires the notification of faults to the LSR(s) responsible for switchover to the recovery path, a mechanism must be provided for the fault indication and the fault repair notification to travel from the point of occurrence of the fault back to the PSL(s). The situation is complicated in the following two cases:

(i) Physically merged LSPs: With label merging multiple working paths may converge to form a multipoint-to-point tree, with the PSLs as the leaves. In this case, therefore, the fault indication and -repair notification should be able to travel along a reverse path of the working path to all the PSLs affected by the fault. For example, in Fig. 1, for a fault along link 34 the affected PSLs are 1 and 9, where as for a fault along link 23, the only affected PSL is 1.

(ii) Virtually merged LSPs: When several LSPs originating at different LSRs share a common segment beyond some node, and share a common identifier (such as the SESSION ID in RSVP-TE), we call such LSPs virtually merged. In this case also, savings in notification can be realized by sending a single notification towards the affected PSLs along segments shared by the LSPs emanating from these PSLs, and allowing the notification to branch out at the merge node(s). For example, in Fig. 1, for a failure along link 67 a single notification could be sent for working paths 1-2-3-4-6-7 and 8-9-3-4-6-7 along their common segment 7-6-4-3. The notification would branch out at node 3, which is the node where the LSP from node **1 to node 7 and the LSP from node 8 to node 7 merge.**

In both the cases above, an appropriate notification path can be provided by the reverse notification tree (RNT which is a point-to-multipoint tree that is an exact mirror image of the converged working paths, along which the FIS and the FRS travel. (see Fig. 1). There are several advantages to using an RNT:

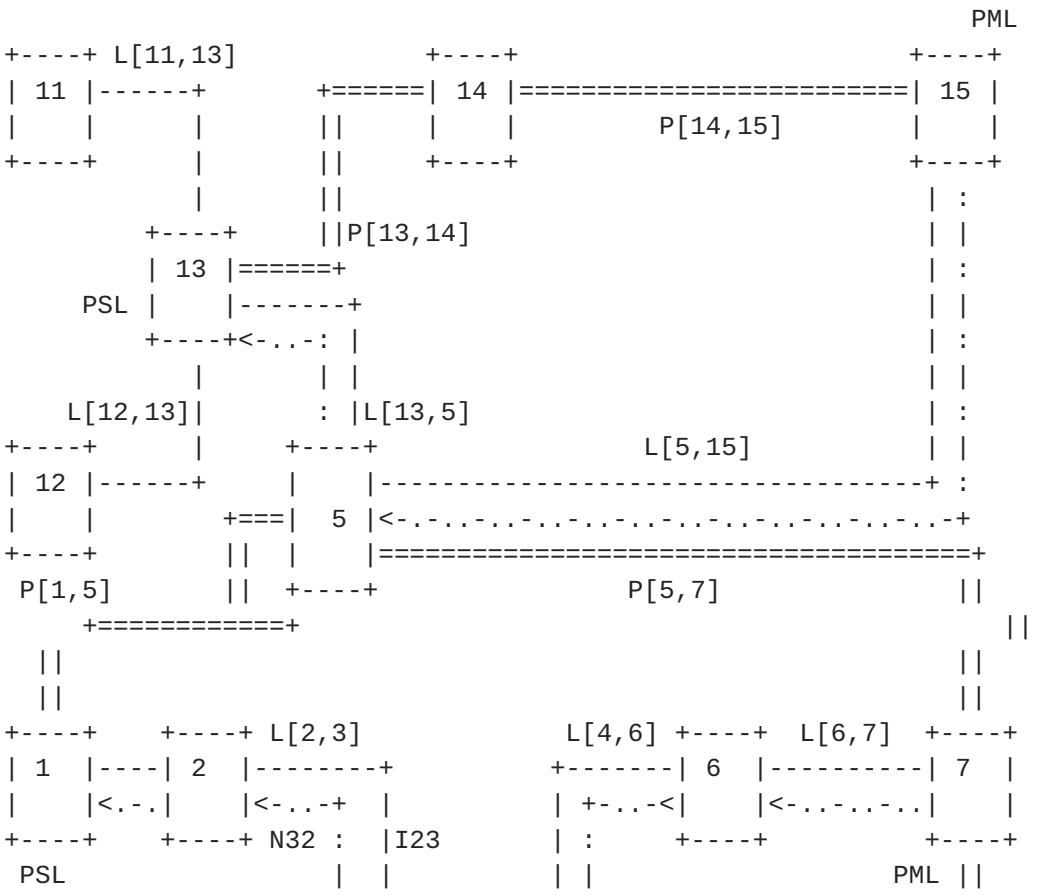
-- The RNT can be established in association with the working path(s), simply by making each LSR along a working path remember its upstream neighbor (or the collection of upstream neighbors whose working paths converge at the LSR and exit as one). Thus,

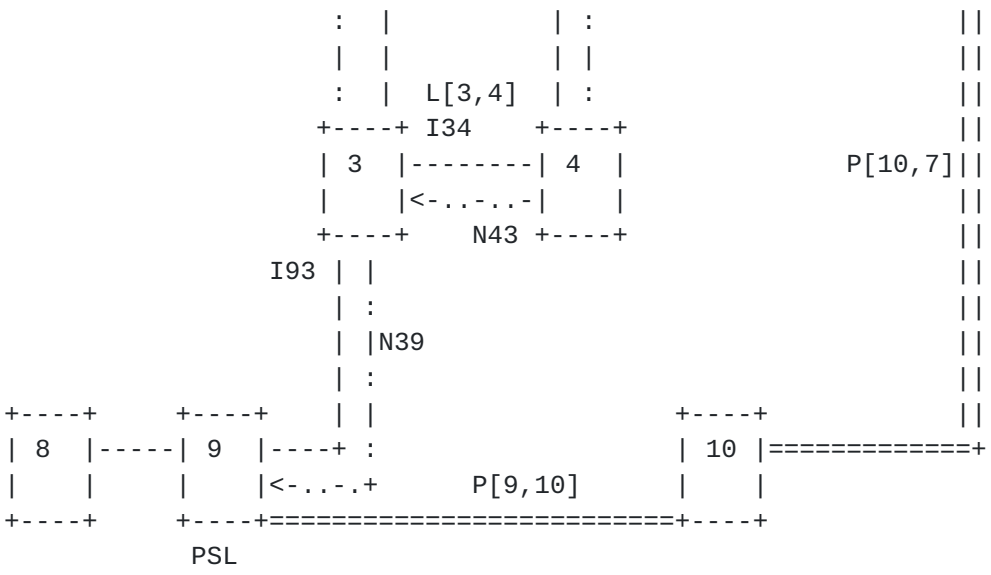
no multicast routing is required. We elaborate more on the RNT in Section 3.

-- Only one RNT is required for all the working paths that merge (either physically or virtually) to form the multipoint-to-point forward path. The RNT is rooted at an appropriately chosen LSR along the common segment of the merged working LSPs and is terminated at the PSLs. All intermediate LSRs on the converged working paths share the same RNT.

Therefore, the RNT enables a reduction in the signaling overhead associated with recovery. Unlike schemes that treat each LSP independently, and require signaling between a PSL and the PML individually for each LSP, the RNT allows for only one (or a small number of) signaling messages on the shared segments of the LSPs.

-- The RNT can be implemented either at Layer 3 or at Layer 2. In either case, the delay along the RNT needs to be carefully controlled. This may be ensured by giving the highest priority to the fault and repair notification packets, which travel along the RNT.





Legend:
--- = Working path
=== = Protection path
-..- = Reverse Notification Tree
---- = Working path
L[x,y] = Working path link between nodes x and y.
P[x,y] = Protection path link between nodes x and y.
Lxy = Label used by the LSP traversing link L[x,y] from x to y.
Nxy = Label used for RNT traffic sent from node x to node y.
Ixy = Interface between nodes x and y.

Figure 1: Illustration of MPLS protection configuration

4.2 Protection Domain

A protection domain is defined as the set of LSRs over which a working path and its corresponding recovery path are routed. Thus, a protection domain is bounded by the LSRs that provide the switching and (if needed) the merging functions for MPLS protection, namely, the PSL and the PML (if present), respectively.
In other words, a protection domain in bounded by the PSL at one end, and by the LSRs that form the end of the working or protection path at the other. In general, if the working and protection paths do not merge within the MPLS domain, the LSRs at the end of the working and protection paths may be egress LSRs. The PSL and the PML (alternatively, the end points of the working and protection paths within the MPLS domain under consideration) are identified during the setting up of an LSP, either via an offline algorithm or by an algorithm that runs at the head-end of an LSP to decide the specific nodes that the LSP must pass through. (Note that segments of the LSP between the PSL and the PML may be loosely routed, as long as the PSL and PML are known). Recovery should ideally be performed between the source and destination (end-to-end), but in some cases segment recovery may be desired (for example, when certain segments are more unreliable than others) or may be the only option (due to the

topology of the network, see Fig. 1). For example, in Fig. 1, the working path 8-9-3-4-6-7, can only have protection on the segment 9-3-4-6-7.

Note that when multiple LSPs merge into a single LSP or when multiple LSPs that share a common identifier follow the same path beyond some node, the working paths corresponding to these LSPs also converge. As explained in Section 4.4, an RNT can be used in this case for propagating the failure and repair notification back to the concerned PSL(s). We can therefore have a situation where different protection domains share a common RNT. A protection domain is denoted by specifying the working path and the recovery path. For example, in Fig. 1, the protection domain bounded by LSR 1 and LSR 7, is denoted by (1-2-3-4-6-7, 1-5-7).

4.2.1 Relationship between protection domains with different RNTs

When protection domains have different RNTs, two cases may arise, depending on whether or not any portions of the two domains overlap, that is, have nodes or links in common. If the protection domains do not overlap, the protection domains are independent (note that by virtue of the RNTs in the two domains being different, neither the working paths nor the RNTs in the two domains can overlap). In other words, failures in one domain do not interact with failures in the other domain. For example, the protection domain defined by (9-3-4-6-7, 9-10-7) is completely independent of the domain defined by (11-13-5-15, 11-13-14-15). As a result, as long as faults occur in independent domains, the network shown in Fig. 1 can tolerate multiple -faults (for example, simultaneous failures on the working path in each domain).

If protection domains with disjoint RNTs overlap, it implies that the protection path of one intersects the working path of the other. Therefore, although failures on the working paths of the two domains do not affect one another, failures on the protection path of one may affect the working path of the other and visa versa. For example, the protection domain defined by (1-2-3-4-6-7, 1-5-7) is not independent of the domain defined by (11-13-5-15, 11-13-14-15) since LSR 5 lies on the protection path in the former domain and on the working path in the latter domain.

4.2.2 Relationship between protection domains with the same RNT

When protection domains have the same RNT, different failures along the working paths may affect both paths differently. As shown in Fig. 1, for example, working paths 1-2-3-4-5-7 and 9-3-4-6-7 share the same RNT. As a result, for a failure on some segments of the working path, both domains will be affected, resulting in a protection switch in both (for example, the segment 3-4-6-7 in Fig. 1). Likewise, for failures on other segments of the working path, only one domain may be affected (for example, failure on segment 2-3

affects only the first working path 1-2-3-4-6-7, where as failure on the segment 9-3 affects only the second working path 9-3-4-6-7).

4.3 Multiple Faults

We note that transferring the working traffic to the recovery path is enough to take care of multiple faults on the working path. However, if multiple faults happen such that there is at least one failure on both the working and recovery paths, MPLS layer recovery may no longer suffice. In this case, the network will either have to allow for Layer 3 rerouting or have the PSL inform the administrator via an alarm, thus enabling the manual reconfiguration of a different working and backup path. (An OAM functionality could greatly simplify such communication.) Note that for a PSL to be able to generate an alarm, it must also have a mechanism for detecting faults on the recovery path, such as a RNT for the recovery path (to allow for the fault notification on the recovery path to be propagated to the PSL).

4.4 Timers and Thresholds

For its proper operation, the protection mechanism described in this contribution uses the following timers and thresholds:

5.0 Configuration

In the following sections, we describe the operation of the path protection mechanism, and explain the various steps involved with reference to Fig. 1.

Protection configuration consists of two aspects: establishing the protection domain and creating the reverse notification tree. The protection domain configuration involves either configuring the working and protection path pair or the protection path of an established working path. These aspects are discussed in this section.

5.1 Establishing a Protection Domain

The label distribution protocol encompasses negotiations in which two label distribution peers engage in order to learn of each other's MPLS capabilities. The label distribution protocol is used to establish a protection domain via signaling. The protection domain consists of a working path and a recovery/protection path pair. MPLS defines two methods for label distribution, Label Distribution Protocol (LDP/CR-LDP) and ReSerVation Protocol (RSVP). Our mechanism is designed to work with either of these label distribution protocols.

LDP/CR-LDP and RSVP allow the path to be setup loosely (each node determines it's next hop) or explicitly (each node has been given

it's next hop). We assume that protection paths will be setup explicitly, however there is no requirement for this. These protocols are being extended to provide a mechanism by which protection establishment can be signaled and created. The funtionality being introduced is:

-- Explicit Route Protection information to identify the PSL and PML, and thus the protection domain.

-- Path Protection information to configure the nodes in the protection domain.

The establishment of the protection domain requires the identification of the working path and the protection path. There are two separate cases to consider: non-merged (point-to-point) and merged (multipoint-to-point). The working and protection paths for RSVP/CR-LDP are identified as follows:

Non-merged:

-- RSVP: Same Sender Template (IP tunnel sender IP address, LSPID)

-- Cr-LDP: Same LSPID TLV (Ingress LSR Router ID and Local CR-LSP ID)

Merged:

-- RSVP: Same session object (IP tunnel end point address and Tunnel ID)

-- Cr-LDP: Same FEC TLV (Host Address and Prefix)

5.1.1 Explicit Route Protection Information

In order to identify the PSL, PML, and the nodes between the PSL and PML that make up a protection domain, anExplicit Route Protection fieldhas been added to the Explicit Route Field of CR-LDP and RSVP-TE [8][9]. The Explicit Route Protection field will first appear when the configuration message reaches the PSL. This denotes the start of a protection domain. When the PSL processes the Explicit Route Protection field, it will modify the configuration message with a Path Protection Field that is directly derived from the Explicit Route Protection Field and then forwards the configuration message.

The configuration message will continue along the path until the second Explicit Route Protection Field is evaluated at the PML. This denotes the end of the protection domain. When the PML processes the Explicit Route Protection Field, it will remove the Path Protection Field from the configuration message and then forward the message.

This same process would be perfomed for each protection domain along the configuration message path. For path protection it is critical to identify the PSL,PML, and nodes within the protection domain. The following attributes are specified in this field.

1. **The Router ID of the PSL or PML;**
2. **Whether the node processing the Explicit Route Protection field** at the current hop is a PSL or PML;
3. **What the protection type is 1+1, 1:1, etc.;**
4. **Whether this is the configuration message for the working or** protection path;
5. **If the protection path resources can be used for extra traffic** becides the protected traffic;
6. **Whether the RNT is implemented as a Hop-by-hop (Layer 3) LSP,** as an MPLS (Layer 2) LSP, or over SONET K1/K2 bytes;
7. **What to configure the hold-off and wait-to-restore timers; and**
8. **If the protection switching action is revertive.**

For example, the Explicit Route Field of the configuration message might look like the following:

```
        Ipv4 Address A
        Ipv4 Address B
        Explicit Route Protection (PSL, Router ID = current hop Ipv4
Router ID B)
        Ipv4 Address C
        Ipv4 Address D
  Ipv4 Address E
  Ipv4 Address F
  Explicit Route Protection (PML, Router ID = current Hop Ipv4
Router ID F)
        Ipv4 Address G
```

Denotes the Explicit Route path of two Ipv4 hops (A and B) with the second Ipv4 (B) hop identified as the PSL by the presence of the Explicit Route Protection field. The PSL signifies the beginning of the protection domain and as such creates the Path Protection Field in the configuration message and forwards the message to the next hop.

The configuration message continues for four more hops with the nodes processing the Path Protection Field. The fourth IPv4 (F)hop is identified as the PML by the presence of the Explicit Route Protection field. The PML signifies the end of the protection domain and as such removes the Path Protection Field from the configuration message prior to forwarding the message to the last hop. This process could continue if other protection domains are present after the PML.

5.1.2 Path Protection Information

The Path Protection specifies whether path protection is activated, identifies whether the path is the working path or protection path, and configures each node with in the protection domain[8][9]. The PSL node learns during a working/protection path configuration process, which working and protection paths are coupled together. The PML node learns during a working/protection path configuration process, which working and protection paths are merged to the outgoing network switch element. The PSL/PML pair constitute a protection domain.

The attributes required to establish the Protection Domain are defined in the framework[3]:

1 **RNT Type: Specifies whether the RNT is implemented as a Hop-by-hop** (Layer 3) LSP, as an MPLS (Layer 2) LSP, or over SONET K1/K2 bytes.
2 **Timer Options: Specifies the hold-off and wait-to-restore time** requirements.
3 **Revertive Option: Specifies whether the recovery action is** revertive.

5.2 Establishing a Protection/Recovery Path

The establishment of the recovery path requires the identification of the working path. There are two separate cases to consider: non-merged (point-to-point) and merged (point-to-multipoint). For path protection mechanisms, the working and protection paths for are identified as follows:

Non-merged:

-- RSVP: Same Sender Template (IP tunnel sender IP address, LSPID)

-- Cr-LDP: Same LSPID TLV (Ingress LSR Router ID and Local CR-LSP ID)

Merged:

-- RSVP: Same session object (IP tunnel end point address and Tunnel ID)

-- Cr-LDP: Same FEC TLV (Host Address and Prefix)

The Explicit Route Protection Field would only carry the protection path configuration information. The configuration of the protection path would be identical to the description provided in 5.1 for the protection path.

In most cases, the working path and its corresponding recovery path would be specified during LSP setup, either via a path selection algorithm (running at a centralized location or at an ingress LSR)

or via administrative configuration. Observe that the specification of the path, does not, strictly speaking, require the entire path to be explicitly specified. Rather, it requires only that the PSL and PML (or in the absence of a PML, the path egress points out of the MPLS domain) be specified, with the segments between them being loosely routed, if required. In other words, the path would be established between the two nodes at the boundaries of the protection domain via (possibly loose) explicit (or source) routing using LDP [ ], [ ] /RSVP [ ], [ ] signaling (alternatively, via constraint-based routing, or using manual configuration).

5.3 Creating the RNT

The RNT is used for propagating the FIS and the FRS, and can be created by a simple extension to the LSP setup process. Note: An MPLS OAM notification is preferable and could make use of the RNT. During the establishment of the working path, the signaling message carries with it the identity (address) of the upstream node that sent it (for example, via the path attribute in RSVP). Each LSR along the path simply remembers the identity of its immediately prior upstream neighbor on each incoming link. Through the neighbor discovery mechanism of the routing protocol, each LSR finds the interface connecting it to the upstream LSRs. (It is assumed in this draft that there is a bi-directional connection between two neighboring LSRs, such as a bi-directional SONET link, a bi-directional lower layer network link (e.g., an ATM VP), or a pair of bi-directional tunnels over an IP subnetwork.) The node then creates an ''inverse'' cross-connect table that for each protected outgoing LSP maintains a list of the incoming LSPs that merge into that outgoing LSP, together with the identity of the upstream node and incoming interface that each incoming LSP comes through. Upon receiving an FIS, an LSR extracts the labels contained in it (which are the labels of the protected LSPs that use the outgoing link that the FIS was received on) and checks whether the current LSR is the PSL for that LSP. If it is it terminates the FIS. Otherwise, it consults its inverse cross-connect table to determine the identity of the upstream nodes that the protected LSPs come from, and creates and transmits an FIS to each of them.

Therefore, based on whether the RNT is implemented at Layer 3 or Layer 2, two cases arise:

If the RNT is implemented by a point-to-multipoint LSP, then the working path can be bound to the ingress label and interface of the RNT LSP at a LSR. Note that the RNT only be a point-to-multipoint LSP in the case of mergeing, otherwise the RNT is implemented as a point-to-point LSP. The ingress label and interface can then be used as an index into the "inverse" cross-connect table to find the egress labels and interfaces of the RNT LSP as shown in Table 2. Upon receiving an FIS, an LSR extracts the labels and checks whether it is the PSL for that LSP. If it is, it terminates the FIS.

Otherwise, it consults its inverse cross-connect table to determine the outgoing labels and interfaces, performs a label swap and forwards the FIS to the appropriate upstream node(s). For example, consider Figure 1, and assume that a Layer 2 point-to-multipoint RNT, rooted at LSR 7 and extending to LSRs 1 and 9, is bound to the multipoint-to-point forward paths starting at LSRs 1 and 8 and terminating at LSR 7. Now in case of a fault on link L[4,6], LSR 3 receives an FIS on the RNT in a labeled packet with label N43. It uses this label as an index into its inverse cross-connect table, and learns that there are two previous nodes (namely those reachable via interfaces I23 and I93 respectively) that the FIS needs to be forwarded to. It encapsulates the received FIS into a labeled packets with labels N32 and N39, and dispatches them along interfaces I23 and I93 respectively.

Table 2. An example inverse cross-connect table for LSR 3 using MPLS (Layer 2) RNT

If the RNT is implemented by a hop-by-hop Layer 3 mechanism, using, for example, UDP packets (with a specific port number to identify notification message type), then the egress label and interface of the working path can be used as an index into the inverse cross-connect table to obtain the IP addresses of the previous hop(s) and the associated outgoing interface(s), as illustrated in Table 3. On each hop, the FIS carried in the UDP packet carries the label and interface of the working path for that hop. Thus, if the receiving node is not a PSL, the label and interface in the FIS can be extracted and can be used to access the inverse cross-connect table. The label and interface used by the working LSP on the hop(s) to the upstream node(s) are then inserted into FIS packet(s), and the FIS packet(s) transmitted to the appropriate upstream node(s) along the interface specified the inverse cross-connect table. Therefore, in the hop-by-hop mechanism the FIS packets are not forwarded by a node to its previous hops using its default layer 3 forwarding table. Rather, these packets are forwarded via the outgoing interface extracted from the node's inverse cross-connect table. As in the example above, in case of a fault on link L[4,6], LSR 3 receives an FIS from LSR 4 that contains the outgoing label L34 and the outgoing interface I34 of the LSP affected by the fault. LSR3 uses these to index its inverse cross-connect table (see Table 3), and learns, as before, that there are two previous nodes (those reachable via interfaces I23 and I93, respectively) that must receive an FIS. It then creates two FIS packets as follows. The first, for transmission along interface I23, contains the label L23 used by LSR 2 to transmit data to LSR 3 along the working LSP. The second, for transmission along interface I93, contains the label L93 used by LSR **9 to transmit data to LSR 3 along the working LSP.**

Table 3. An example inverse cross-connect table for LSR 3 using a

hop-by-hop (Layer 3) RNT

The roles of the various core protection/recovery components are:

PSL: The PSL must be able to correlate the RNT with the working and recovery paths. To this end, it maintains a table with a list of working LSPs protected by an RNT, and the identity of the recovery LSPs that each working path is to be switched to in the event of a failure on the working path. It need not maintain an inverse cross-connect table (for those LSPs and working paths for which it is the PSL).

PML: The PML, in general, has to remember all of its upstream neighbors and associate them with the appropriate working paths and RNTs. If the PML is also the root of the RNT, it has to associate each of its upstream nodes with a working path and RNT, but it need not maintain an inverse cross-connect table (for those LSPs and working paths for which it is a PML).

Intermediate LSR: An intermediate LSR has to only remember all of its upstream neighbors and associate them with the appropriate working paths and RNTs, and has to maintain an "inverse" cross-connect table.

5.4 Engineering a Protection Domain

For 1:1 protection, the bandwidth (if any) reserved for a protection/recovery path should be the same as the bandwidth reserved for its corresponding working path. This guarantees the same bandwidth for the protected traffic after protection switching. If the LSRs on the protection path support excess mode [3], the bandwidth reserved on the protection path for protecting high priority traffic can be used by other lower priority traffic streams. That is, lower priority traffic that has a segment in common with the recovery path, use the bandwidth of the recovery path, as long as the recovery path is not called into use. When the recovery path is pressed into service, the low priority traffic will be discarded to allow for the actual working traffic to take its place. Also, if delay, jitter or other QoS parameters are to be satisfied, the protection path in 1:1 protection should be chosen such that these requirements are satisfied.

Since the volume of signaling traffic (e.g., FIS/FRS messages, or liveness messages) is small, in general bandwidth need not be reserved for the signaling traffic provided that there exist other mechanisms that can ensure that the delay requirements of signaling messages are met (by using, for example, the highest priority for signaling messages).

For bypass tunneling protection, multiple working LSPs may share the

same protection bandwidth by tunneling protection LSPs over a common path. This requires that the working paths of these LSPs be disjoint, except at the PSL and PML, so that they can be assumed to not all fail at the same time. In this case, the bandwidth reserved on the tunnel will be the maximum of all individual paths. Otherwise, a bypass tunnel could be created to carry all the backup paths, with the bandwidth reserved for the tunnel being the maximum bandwidth required over all failure scenarios on the working LSPs.

5.5 Configuring Timers

The purpose of timers t1/t1' is to control the tradeoff between notification delay of the FIS/FRS and the resources consumed when sending the FIS/FRS. If t1/t1' is large, it may take a relatively long time for the node that initiated the FIS/FRS transmission to send the second the FIS/FRS if the first FIS/FRS message is lost, thereby increasing notification delay. On the other hand, if t1/t1' is small, the repetitive sending of FIS/FRS messages may waste bandwidth and processing power because the first message may already have reached the PSL(s).

It is assumed that after t2/t2' it is not necessary to do protection at MPLS layer, either because it is no longer useful or because by that time an upper layer protection mechanism will have been triggered.

The timers t4/t4' are used to control the frequency of liveness messages sent between neighboring LSRs, so their purpose is the same as those of timers t1/t1'. While frequent exchanges of liveness messages can unnecessarily consume network resources, too few exchanges may delay the discovery of faults. To accommodate delay jitter, t4' may be set at a slightly different value from t4.

The timers t5/t6 are used to allow lower layer protection to take effect before initiating MPLS layer recovery mechanisms (for example, an automatic protection switching between fibers that comprise a link between two LSRs). Following the detection of a fault/fault repair S/FRS packet, respectively. This allows for the lower layer protection to take effect and for the LSR to learn this through one of several ways: via an indication from a lower layer, or by the resumption of the reception of a liveness message, or by the lack of LF, LD, PF or PD conditions (see definitions in [3]).

The threshold K helps to minimize false alarms due to the occasional loss of a liveness message, which may occur, for example, either due to a temporary impairment in a link or a peer LSR or due to a buffer overflow.

6.0 Fault Detection

Each LSR must be able to detect certain types of faults, such as PF,

PD, LF, and LD [3] and propagate an FIS message towards the PSL. Here we consider unidirectional link faults, bi-directional (or complete) link faults, and node faults.

Essentially, the node upstream of the fault must be able to detect/learn about the fault. This motivates the need for a "liveness" message, which allows a node upstream of the fault to detect the fault either directly or implicitly. Also, the fault detection mechanism must provide the trigger for generating the FIS. Broadly, the types of mechanisms that could be triggers for the FIS are:

i) Lower layer mechanisms

ii) MPLS-based detection mechanisms, which may be used to detect link faults, via a liveness message for example.

iii) User-plane OAM mechanisms, such as a path continuity test, which may be used to detect other faults, such as mis-connections (arising from incorrect entries in the label forwarding table, for example).

The fault types that need to be detected are:

-- Unidirectional Link Fault: A uni-directional fault implies that only one direction of a bi-directional link has experienced a fault

-- Downlink Fault: A fault on a link in the downstream direction will be detected by the node downstream of the faulty link, either via the PF or PD condition being detected at the MPLS layer, or via LF or LD signals being propagated to the MPLS layer by the lower layer or via the absence of liveness messages.

-- Uplink Fault: A fault on a link in the upstream direction will be detected by a node upstream of the faulty link, either via a LF or LD being detected at the lower layer and propagated to the MPLS layer (if there was traffic on this reverse link), or via the PD or PF condition being detected at the MPLS layer, or via absence of liveness messages.

-- Bi-directional link fault or node fault: When both directions of the link have a fault (as in the case of a fiber cut), nodes at both ends of the link will detect the fault either due to the LF or PF signal or due to the absence of liveness messages.

7.0 Fault Notification

The rapid notification of a fault is effected by the propagation of the FIS message along the RNT. Due to the timers built into the FIS/FRS propagation mechanism, the transportation of FIS/FRS messages does not require a reliable mechanism like TCP. Any LSR may generate an FIS.

For instance, in Fig. 1 if link L23 fails, LSR 3 will detect it and transmit a FIS to LSR 2 (after waiting for time T2), its upstream neighbor along link L23. The FIS will contain the incoming labels (at node 3) of those LSPs on link L23 that have protection enabled. Upon receiving the FIS message, LSR 2 will consult its inverse-cross connect table and generate an FIS message for LSR 1, which on receiving the first FIS packet will wait for time t3 before performing a protection switch. The node which initiates the FIS will continue to send FIS messages at an interval of t1 until timer t2 expires. After t2 expires it is assumed that either upper layer protection will be triggered or enough number of FIS messages will have been sent to reach the desired reliability in conveying fault information to the PSL(s).

The roles of the various core protection switching components are:

PSL: The PSL does not generate a FIS message, but must be able to detect FIS packets.

PML: The PML must be able to generate the FIS packets in response to detecting failure, and should transmit them over the RNT. The PML begins FIS transmission after continuously detecting a fault for T2 time units, and does so every t1 time units for a maximum of t2 time units.

Intermediate LSR: An intermediate LSR must be able to generate/forward FIS packets, either as a result of continuously detecting a fault for T2 time units or in response to a received FIS packet. It must transmit these to all its affected upstream neighbors as per its inverse cross-connect table. Again, it does so every t1 time units for a maximum of t2 time units.

8.0 Switch Over

The switch over is the actual switching of the working traffic from the working path to the recovery path. This is performed by a PSL, t3 time units after the reception of the first FIS packet.

For example, in Fig. 1, consider protection domain (1-2-3-4-6-7, 1-5-7). When link L34 fails, the PSL LSR 1 on learning of the failure will perform a protection switch of the protected traffic from the working path 1-2-3-4-6-7 to the backup path 1-5-7. Notice that LSR 7 acts as a protection merge LSR, merging traffic from the working and backup paths. Since buffered packets from LSR 4 may continue to arrive at LSR 7 even after the protection switch (the dampening timer t43 at the PSL tends to mitigate this), a short-term misordering of packets may happen at LSR 7, until the buffers on the working path drain out.

The role of the core protection components is as follows:

PSL: Performs the protection switch upon receipt of the FIS message, but after waiting for time t3 following the first FIS message.

PML: The PML automatically merges protection traffic with working traffic. For a short period of time this may cause misordering of packets, since packets buffered at LSRs downstream of the fault may continue to arrive at the PML along the working path.

Intermediate LSR: The intermediate LSR has no special action.

9.0 Switch Back

Switch back or restoration is the transfer of working traffic from the recovery path to the working path, once the working path is repaired. This may be because the recovery path may be a limited recovery path [3], or because the working path is deemed to be preferred [3] in some respect. Restoration may be automatic or it may be performed by manual intervention (or not performed at all). In the revertive mode, restoration is performed upon the receipt of the FRS message. A path continuity test may be performed to ensure the integrity of the entire path before switching. I n the non-revertive mode it may be performed by operator intervention.

The role of the core protection components is similar here to what it is for protection switching. The PML does not need to do anything, unless it was the node that detected the failure, in which case it transmits a FRS upstream t6 time units after continuously detecting recover signal from lower layer or after detecting liveness messages from its peers. The intermediate LSR generates the FRS message if it was the node that detected the recovery or generates a FRS to relay the restoration status received from a downstream node. The PSL performs the restoration switch t3' seconds after receiving the first FIS message.

10.0 Protocol Specific Extensions

The signaling protocol specific extensions needed to implement the mechanism outlined in this draft are discussed in separate documents [ ],[9].

11.0 Security Considerations

The MPLS protection that is specified herein does not raise any security issues that are not already present in the MPLS architecture.

12.0 Intellectual Property Considerations

In accordance with the intellectual property rights procedures of the IETF standards process, to the extent that Tellabs has patents, pending applications and/or other intellectual property rights that

are essential to implementation of any subject matter submitted by Tellabs that is included in a standard, Tellabs is prepared to grant, on the basis of reciprocity (grantback), a license on such subject matter under terms and conditions that are reasonable and non-discriminatory.

13.0 Acknowledgements

We would like to thank our colleague Ben Mack-Crane, and members of the MPLS WG list, in particular Dave Allan, Bora Akyol, Neil Harrisson, Ping Pan, and J. Noel Chiappa, for suggestions, feedback, and corrections to the first version of this draft.

14.0 Authors' Addresses

Changcheng Huang
Vishal Sharma
Department of Systems and
Computer Engineering
Metanoia, Inc.
Carleton University
335 Elan Village Lane

1125 Colonel By Drive
Unit 203
Ottawa, Ontario K1S 5B6
San Jose, CA 95134-2539
Phone: (613) 520-2600 ext. 2477
Phone: 408-943-1794
Changcheng.huang@sce.carleton.ca
v.sharma@ieee.org

Srinivas Makam
Ken Owens
Tellabs Operations, Inc.
Erlang Technology, Inc.
4951 Indiana Avenue
1106 Fourth Street
Lisle, IL 60532
St. Louis, MO 63126
Phone: 630-512-7217
Phone: 314-918-1579
Srinivas.Makam@tellabs.com
keno@erlangtech.com

Ben Mack-Crane
Tellabs Operations, Inc.
4951 Indiana Avenue
Lisle, IL 60532

Ben.Mackcrane@tellabs.com
Ph: 630-848-7875

15.0 References

[1] Rosen, E., Viswanathan, A., and Callon, R., "Multiprotocol Label Switching Architecture", Work in Progress, Internet Draft <draft-ietf-mpls-arch-07.txt>, July 2000.

[2] Callon, R., Doolan, P., Feldman, N., Fredette, A., Swallow, G., Viswanathan, A., "A Framework for Multiprotocol Label Switching", Work in Progress, Internet Draft <draft-ietf-mpls-framework-05.txt>, September 1999.

[3] Makam, V., Sharma, V., Huang, C., Owens, K., Mack-Crane, B., et al, "A Framework for MPLS-based Recovery, " Work in Progress, Internet Draft <draft-ietf-mpls-recovery-frmwrk-00.txt>, September 2000.

[4] Andersson, L., Doolan, P., Feldman, N., Fredette, A., Thomas, B., "LDP Specification", Work in Progress, Internet Draft <draft-ietf-mpls-ldp-11.txt>, August 2000.

[5] Jamoussi, B. "Constraint-Based LSP Setup using LDP", Work in Progress, Internet Draft <draft-ietf-mpls-cr-ldp-04.txt>, July 2000.

[6] Braden, R., Zhang, L., Berson, S., Herzog, S., "Resource ReSerVation Protocol (RSVP) -- Version 1 Functional Specification", RFC 2205, September 1997.

[7] Awduche, D. et al "Extensions to RSVP for LSP Tunnels", Work in Progress, Internet Draft <draft-ietf-mpls-rsvp-lsp-tunnel-07.txt, August 2000.

[8] Huang, C., Sharma, V., Makam. V, and Owens, K., "Extensions to RSVP-TE for MPLS Path Protection, " Internet Draft, <draft-chang-rsvpte-path-protection-ext-01.txt>, November 2000.

[9] Owens, K., Sharma, V., Makam. V, and Huang, C., "Extensions to CR-LDP for MPLS Path Protection, " Internet Draft, <draft-owens-crldp-path-protection-ext-00.txt>, November, 2000.