

MPLS WG  
Internet-Draft  
Intended status: Standards Track  
Expires: July 15, 2019

K. Kompella  
Juniper Networks, Inc.  
L. Contreras  
Telefonica  
January 11, 2019

Resilient MPLS Rings  
draft-ietf-mpls-rmr-09

## Abstract

This document describes the use of the MPLS control and data planes on ring topologies. It describes the special nature of rings, and proceeds to show how MPLS can be effectively used in such topologies. It describes how MPLS rings are configured, auto-discovered and signaled, as well as how the data plane works. Companion documents describe the details of discovery and signaling for specific protocols.

## Requirements Language

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in [[RFC2119](#)].

## Status of This Memo

This Internet-Draft is submitted in full conformance with the provisions of [BCP 78](#) and [BCP 79](#).

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF). Note that other groups may also distribute working documents as Internet-Drafts. The list of current Internet-Drafts is at <https://datatracker.ietf.org/drafts/current/>.

Internet-Drafts are draft documents valid for a maximum of six months and may be updated, replaced, or obsoleted by other documents at any time. It is inappropriate to use Internet-Drafts as reference material or to cite them other than as "work in progress."

This Internet-Draft will expire on July 15, 2019.

## Copyright Notice

Copyright (c) 2019 IETF Trust and the persons identified as the document authors. All rights reserved.

Internet-Draft

Resilient MPLS Rings

January 2019

This document is subject to [BCP 78](https://trustee.ietf.org/license-info) and the IETF Trust's Legal Provisions Relating to IETF Documents (<https://trustee.ietf.org/license-info>) in effect on the date of publication of this document. Please review these documents carefully, as they describe your rights and restrictions with respect to this document. Code Components extracted from this document must include Simplified BSD License text as described in Section 4.e of the Trust Legal Provisions and are provided without warranty as described in the Simplified BSD License.

## Table of Contents

<a href="#">1.</a>	Introduction . . . . .	<a href="#">2</a>
<a href="#">1.1.</a>	Definitions . . . . .	<a href="#">3</a>
<a href="#">2.</a>	Motivation . . . . .	<a href="#">5</a>
<a href="#">3.</a>	Theory of Operation . . . . .	<a href="#">5</a>
<a href="#">3.1.</a>	Provisioning . . . . .	<a href="#">5</a>
<a href="#">3.2.</a>	Ring Nodes . . . . .	<a href="#">6</a>
<a href="#">3.3.</a>	Ring Links and Directions . . . . .	<a href="#">6</a>
<a href="#">3.3.1.</a>	Express Links . . . . .	<a href="#">6</a>
<a href="#">3.4.</a>	Ring LSPs . . . . .	<a href="#">7</a>
<a href="#">3.5.</a>	Installing Primary LFIB Entries . . . . .	<a href="#">7</a>
<a href="#">3.6.</a>	Installing FRR LFIB Entries . . . . .	<a href="#">7</a>
<a href="#">3.7.</a>	Protection . . . . .	<a href="#">8</a>
<a href="#">4.</a>	Autodiscovery . . . . .	<a href="#">9</a>
<a href="#">4.1.</a>	Overview . . . . .	<a href="#">9</a>
<a href="#">4.2.</a>	Ring Announcement Phase . . . . .	<a href="#">10</a>
<a href="#">4.3.</a>	Mastership Phase . . . . .	<a href="#">10</a>
<a href="#">4.4.</a>	Ring Identification Phase . . . . .	<a href="#">11</a>
<a href="#">4.5.</a>	Ring Changes . . . . .	<a href="#">11</a>
<a href="#">5.</a>	Ring Signaling . . . . .	<a href="#">12</a>
<a href="#">6.</a>	Ring OAM . . . . .	<a href="#">12</a>
<a href="#">7.</a>	Advanced Topics . . . . .	<a href="#">12</a>
<a href="#">7.1.</a>	Half-rings . . . . .	<a href="#">12</a>
<a href="#">7.2.</a>	Hub Node Resilience . . . . .	<a href="#">12</a>
<a href="#">8.</a>	Security Considerations . . . . .	<a href="#">13</a>
<a href="#">9.</a>	Acknowledgments . . . . .	<a href="#">13</a>
<a href="#">10.</a>	IANA Considerations . . . . .	<a href="#">13</a>
<a href="#">11.</a>	References . . . . .	<a href="#">13</a>
<a href="#">11.1.</a>	Normative References . . . . .	<a href="#">13</a>
<a href="#">11.2.</a>	Informative References . . . . .	<a href="#">13</a>
	Authors' Addresses . . . . .	<a href="#">14</a>

## 1. Introduction

Rings are a very common topology in transport networks. A ring is the simplest topology offering link and node resilience. Rings are nearly ubiquitous in access and aggregation networks. As MPLS

increases its presence in such networks, and takes on a greater role in transport, it is imperative that MPLS handles rings well; this is not the case today.

This document describes the special nature of rings, and the special needs of MPLS on rings. It then shows how these needs can be met in several ways, some of which involve extensions to protocols such as IS-IS [[RFC5305](#)], OSPF [[RFC3630](#)], RSVP-TE [[RFC3209](#)] and LDP [[RFC5036](#)]. RMR LSPs can also be signaled with SPRING [[RFC8402](#)]; that will be described in a future document.

The intent of this document is to handle rings that "occur naturally". Many access and aggregation networks in metros have their start as a simple ring. They may then grow into more complex topologies, for example, by adding parallel links to the ring, or by adding "express" links. The goal here is to discover these rings (with some guidance), and run MPLS over them efficiently. The intent is not to construct rings in a mesh network, and use those for protection.

### 1.1. Definitions

A (directed) graph  $G = (V, E)$  consists of a set of vertices (or nodes)  $V$  and a set of edges (or links)  $E$ . An edge is an ordered pair of nodes  $(a, b)$ , where  $a$  and  $b$  are in  $V$ . (In this document, the terms node and link will be used instead of vertex and edge.)

A ring is a subgraph of  $G$ . A ring consists of a subset of  $n$  nodes  $\{R_i, 0 \leq i < n\}$  of  $V$ . The directed edges  $\{(R_i, R_{i+1}) \text{ and } (R_{i+1}, R_i), 0 \leq i < n-1\}$  must be a subset of  $E$  (note that index arithmetic is done modulo  $n$ ). We define the direction from node  $R_i$  to  $R_{i+1}$  as "clockwise" (CW) and the reverse direction as "anticlockwise" (AC). As there may be several rings in a graph, we number each ring with a distinct ring ID RID.

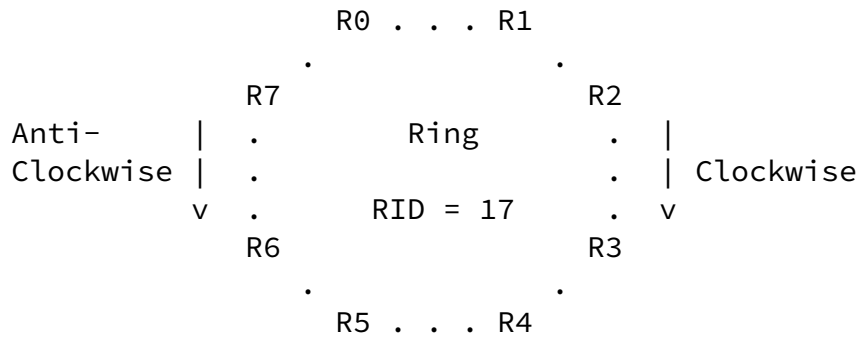


Figure 1: Ring with 8 nodes

The following terminology is used for ring LSPs:

**Ring ID (RID):** A non-zero number that identifies a ring; this is unique in some scope of a Service Provider's network. A node may belong to multiple rings.

**Ring node:** A member of a ring. Note that a device may belong to several rings.

**Node index:** A logical numbering of nodes in a ring, from zero upto one less than the ring size. Used purely for exposition in this document.

**Ring master:** The ring master initiates the ring identification process. Mastership is indicated in the IGP by a two-bit field.

**Ring neighbors:** Nodes whose indices differ by one (modulo ring size).

**Ring links:** Links that connect ring neighbors.

**Express links:** Links that connect non-neighboring ring nodes.

**Ring direction:** A two-bit field in the IGP indicating the direction of a link. The choices are:

UN: 00 undefined link

CW: 01 clockwise ring link

AC: 10 anticlockwise ring link

EX: 11 express link

Ring Identification: The process of discovering ring nodes, ring links, link directions, and express links.

The following notation is used for ring LSPs:

$R_k$ : A ring node with index  $k$ .  $R_k$  has AC neighbor  $R_{(k-1)}$  and CW neighbor  $R_{(k+1)}$ .

$RL_k$ : A (unicast) Ring LSP anchored on node  $R_k$ .

$CL_{jk}$ : A label allocated by  $R_j$  for  $RL_k$  in the CW direction.

$AL_{jk}$ : A label allocated by  $R_j$  for  $RL_k$  in the AC direction.

$P_{jk}$  ( $Q_{jk}$ ): A Path (Resv) message sent by  $R_j$  for  $RL_k$ .

## [2.](#) Motivation

A ring is the simplest topology that offers resilience. This is perhaps the main reason to lay out fiber in a ring. Thus, effective mechanisms for fast failover on rings are needed. Furthermore, there are large numbers of rings. Thus, configuration of rings needs to be as simple as possible. Finally, bandwidth management on access rings is very important, as bandwidth is generally quite constrained here.

The goals of this document are to present mechanisms for improved MPLS-based resilience in ring networks (using ideas that are reminiscent of Bidirectional Line Switched Rings), for automatic bring-up of LSPs, better bandwidth management and for auto-hierarchy. These goals can be achieved using extensions to existing IGP and MPLS signaling protocols, using central provisioning, or in other ways.

## [3.](#) Theory of Operation

Say a ring has ring ID RID. The ring is provisioned by choosing one or more ring masters for the ring and assigning them the RID. Other

nodes in the ring may also be assigned this RID, or may be configured as "promiscuous". Ring discovery then kicks in. When each ring node knows its CW and AC ring neighbors and its ring links, and all express links have been identified, ring identification is complete.

Once ring identification is complete, each node signals one or more ring LSPs  $RL_i$ .  $RL_i$ , anchored on node  $R_i$ , consists of two counter-rotating unicast LSPs that start and end at  $R_i$ . A ring LSP is "multipoint": any node  $R_j$  can use  $RL_i$  to send traffic to  $R_i$ ; this can be in either the CW or AC directions, or both (i.e., load balanced). Both of these counter-rotating LSPs are "active"; the choice of direction to send traffic to  $R_i$  is determined by policy at the node where traffic is injected into the ring. The default is to send traffic along the shortest path. Bidirectional connectivity between nodes  $R_i$  and  $R_j$  is achieved by using two different ring LSPs:  $R_i$  uses  $RL_j$  to reach  $R_j$ , and  $R_j$  uses  $RL_i$  to reach  $R_i$ .

### [3.1.](#) Provisioning

The goal here is to provision rings with the absolute minimum configuration. The exposition below aims to achieve that using auto-discovery via a link-state IGP (see [Section 4](#)). Of course, auto-discovery can be overridden by configuration. For example, a link that would otherwise be classified by auto-discovery as a ring link might be configured not to be used for ring LSPs.

### [3.2.](#) Ring Nodes

Ring nodes have a loopback address, and run a link-state IGP and an MPLS signaling protocol. To provision a node as a ring node for ring RID, the node is simply assigned that RID. A node may be part of several rings, and thus may be assigned several ring IDs.

To simplify ring provisioning even further, a node  $N$  may be made "promiscuous" by being assigned an RID of 0. A promiscuous node listens to RIDs in its IGP neighbors' link-state updates. For every non-zero RID  $N$  hears from a neighbor,  $N$  joins the corresponding ring by taking on that RID. In many situations, the use of promiscuous mode means that only one or two nodes in a ring needs to be provisioned; everything else is auto-discovered.

A ring node indicates in its IGP updates the ring LSP signaling protocols it supports. This can be LDP and/or RSVP-TE. Ideally, each node should support both.

### 3.3. Ring Links and Directions

Ring links must be MPLS-capable. They are by default unnumbered, point-to-point (from the IGP point of view) and "auto-bundled". The last attribute means that parallel links between ring neighbors are considered as a single link, without the need for explicit configuration for bundling (such as a Link Aggregation Group). Note that each component may be advertised separately in the IGP; however, signaling messages and labels across one component link apply to all components. Parallel links between a pair of ring nodes is often the result of having multiple lambdas or fibers between those nodes. RMR is primarily intended for operation at the packet layer; however, parallel links at the lambda or fiber layer result in parallel links at the packet layer.

A ring link is not provisioned as belonging to the ring; it is discovered to belong to ring RID if both its adjacent nodes belong to RID. A ring link's direction (CW or AC) is also discovered; this process is initiated by the ring's ring master. Note that the above two attributes can be overridden by provisioning if needed; it is then up to the provisioning system to maintain consistency across the ring.

#### 3.3.1. Express Links

Express links are discovered once ring nodes, ring links and directions have been established. As defined earlier, express links are links joining non-neighbor ring nodes; often, this may be the

result of optically bypassing ring nodes. The use of express links will be described in a future version of this document.

### 3.4. Ring LSPs

Ring LSPs are not provisioned. Once a ring node  $R_i$  knows its RID, its ring links and directions, it kicks off ring LSP signaling automatically.  $R_i$  allocates CW and AC labels for each ring LSP

RL<sub>k</sub>. R<sub>i</sub> also initiates the creation of RL<sub>i</sub>. As the signaling propagates around the ring, CW and AC labels are exchanged. When R<sub>i</sub> receives CW and AC labels for RL<sub>k</sub> from its ring neighbors, primary and fast reroute (FRR) paths for RL<sub>k</sub> are installed at R<sub>i</sub>. More details are given in [Section 5](#).

For RSVP-TE LSPs, bandwidths may be signaled in both directions. However, these are not provisioned either; rather, one does "reverse call admission control". When a service needs to use an LSP, the ring node where the traffic enters the ring attempts to increase the bandwidth on the LSP to the egress. If successful, the service is admitted to the ring.

### [3.5](#). Installing Primary LFIB Entries

In setting up RL<sub>k</sub>, a node R<sub>j</sub> sends out two labels: CL<sub>jk</sub> to R<sub>j-1</sub> and AL<sub>jk</sub> to R<sub>j+1</sub>. R<sub>j</sub> also receives two labels: CL<sub>j+1,k</sub> from R<sub>j+1</sub>, and AL<sub>j-1,k</sub> from R<sub>j-1</sub>. R<sub>j</sub> can now set up the forwarding entries for RL<sub>k</sub>. In the CW direction, R<sub>j</sub> swaps incoming label CL<sub>jk</sub> with CL<sub>j+1,k</sub> with next hop R<sub>j+1</sub>; these allow R<sub>j</sub> to act as LSR for RL<sub>k</sub>. R<sub>j</sub> also installs an LFIB entry to push CL<sub>j+1,k</sub> with next hop R<sub>j+1</sub> to act as ingress for RL<sub>k</sub>. Similarly, in the AC direction, R<sub>j</sub> swaps incoming label AL<sub>jk</sub> with AL<sub>j-1,k</sub> with next hop R<sub>j-1</sub> (as LSR), and an entry to push AL<sub>j-1,k</sub> with next hop R<sub>j-1</sub> (as ingress).

Clearly, R<sub>k</sub> does not act as ingress for its own LSPs. However, R<sub>k</sub> can send OAM messages, for example, an MPLS ping or traceroute ([\[I-D.ietf-mpls-rfc4379bis\]](#)), using labels CL<sub>k,k+1</sub> and AL<sub>k-1,k</sub>, to test the entire ring LSP anchored at R<sub>k</sub> in both directions. Furthermore, if these LSPs use UHP, then R<sub>k</sub> installs LFIB entries to pop CL<sub>k,k</sub> for packets received from R<sub>k-1</sub> and to pop AL<sub>k,k</sub> for packets received from R<sub>k+1</sub>.

### [3.6](#). Installing FRR LFIB Entries

At the same time that R<sub>j</sub> sets up its primary CW and AC LFIB entries, it can also set up the protection forwarding entries for RL<sub>k</sub>. In the CW direction, R<sub>j</sub> sets up an FRR LFIB entry to swap incoming label CL<sub>jk</sub> with AL<sub>j-1,k</sub> with next hop R<sub>j-1</sub>. In the AC direction,

R<sub>j</sub> sets up an FRR LFIB entry to swap incoming label AL<sub>jk</sub> with



CL<sub>j+1,k</sub> with next hop R<sub>j+1</sub>. Again, R<sub>k</sub> does not install FRR LFIB entries in this manner.

### 3.7. Protection

In this scheme, there are no protection LSPs as such -- no node or link bypass LSPs, no standby LSPs, no detours, and no LFA-type protection. Protection is via the "other" direction around the ring, which is why ring LSPs are in counter-rotating pairs. Protection works in the same way for link, node and ring LSP failures.

If a node R<sub>j</sub> detects a failure from R<sub>j+1</sub> -- either all links to R<sub>j+1</sub> fail, or R<sub>j+1</sub> itself fails, R<sub>j</sub> switches traffic on all CW ring LSPs to the AC direction using the FRR LFIB entries. If the failure is specific to a single ring LSP, R<sub>j</sub> switches traffic just for that LSP. In either case, this switchover can be very fast, as the FRR LFIB entries can be preprogrammed. Fast detection and fast switchover lead to minimal traffic loss.

R<sub>j</sub> then sends an indication to R<sub>j-1</sub> that the CW direction is not working, so that R<sub>j-1</sub> can similarly switch traffic to the AC direction. For RSVP-TE, this indication can be a PathErr or a Notify; other signaling protocols have similar indications. These indications propagate AC until each traffic source on the ring AC of the failure uses the AC direction. Thus, within a short period, traffic will be flowing in the optimal path, given that there is a failure on the ring. This contrasts with (say) bypass protection, where until the ingress recomputes a new path, traffic will be suboptimal.

Note that the failure of a node or a link will not necessarily affect all ring LSPs. Thus, it is important to identify the affected LSPs (and switch them), but to leave the rest alone.

One point to note is that when a ring node, say R<sub>j</sub>, fails, RL<sub>j</sub> is clearly unusable. However, the above protection scheme will cause a traffic loop: R<sub>j-1</sub> detects a failure CW, and protects by sending CW traffic on RL<sub>j</sub> back all the way to R<sub>j+1</sub>, which in turn sends traffic to R<sub>j-1</sub>, etc. There are three proposals to avoid this:

1. Each ring node acting as ingress sends traffic with a TTL of at most 2\*n, where n is the number of nodes in the ring.
2. A ring node sends protected traffic (i.e., traffic switched from CW to AC or vice versa) with TTL just large enough to reach the egress.

3. A ring node sends protected traffic with a special purpose label below the ring LSP label. A protecting node first checks for the presence of this label; if present, it means that the traffic is looping and MUST be dropped.

It is recommended that (2) be implemented. The other methods are optional.

#### 4. Autodiscovery

##### 4.1. Overview

Auto-discovery proceeds in three phases. The first phase is the announcement phase. The second phase is the mastership phase. The third phase is the ring identification phase.

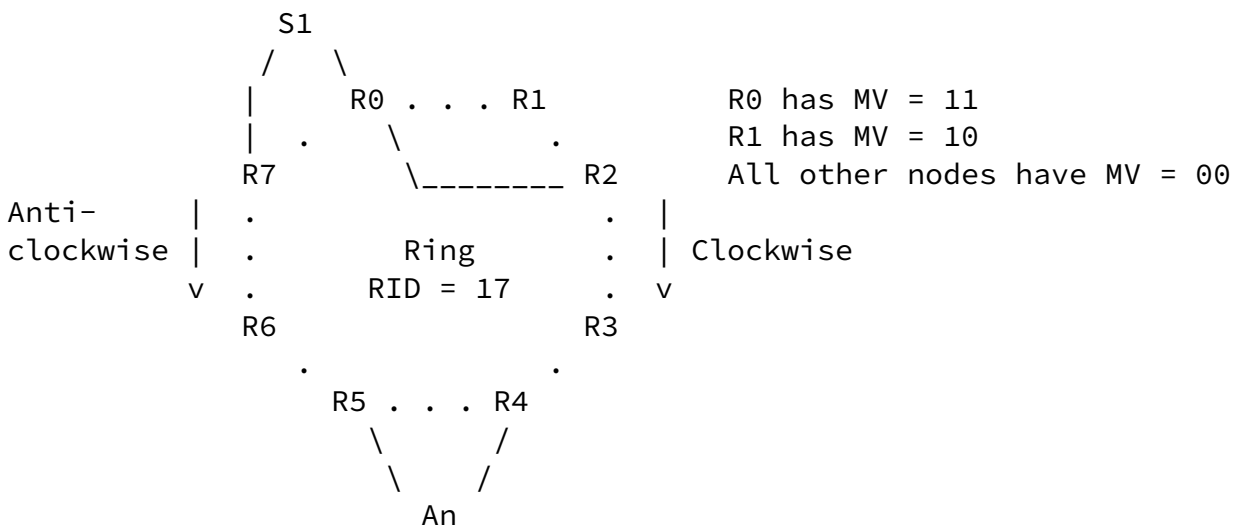


Figure 2: Ring with non-ring nodes and links

The format of an RMR Node Type-Length-Value (TLV) is given below. It consists of information pertaining to the node and optionally, sub-TLVs. A Neighbor sub-TLV contains information pertaining to the node's neighbors. Other sub-TLVs may be defined in the future. Details of the format specific to IS-IS and OSPF will be given in the corresponding IGP documents.

[RMR Node Type][RMR Node Length][RID][Node Flags][sub-TLVs]

Ring Node TLV Format

[RMR Nbr Type][RMR Nbr Length][Nbr Address][Nbr Flags]

```

0                               1
0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5
+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
|MV | SS  | SO  |  MBZ    |SU |M|
+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
MV: Mastership Value
SS: Supported Signaling Protocols
   (100 = RSVP-TE; 010 = LDP; 001 = SPRING)
SO: Supported OAM Protocols (100 = BFD; 010 = CFM; 001 = EFM)
SU: Signaling Protocol to Use (00 = none; 01 = LDP; 10 = RSVP-TE)
M : Elected Master (0 = no, 1 = yes)

```

Flags for a Ring Node TLV

```

0                               1
0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5
+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
|RD |OAM|          MBZ          |
+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
RD: Ring Direction
OAM: OAM Protocol to use (00 = none; 01 = BFD; 10 = CFM; 11 = EFM)

```

Flags for a Ring Neighbor TLV

[4.2.](#) Ring Announcement Phase

Each node participating in an MPLS ring is assigned an RID; in the example, RID = 17. A node is also provisioned with a mastership value. Each node advertises a ring node TLV for each ring it is participating in, along with the associated flags. It then starts timer T1.

A node in promiscuous mode doesn't advertise any ring node TLVs. However, when it hears a ring node TLV from an IGP neighbor, it joins that ring, and sends its own ring node TLV with that RID.

The announcement phase allows a ring node to discover other ring nodes in the same ring so that a ring master can be elected.

### [4.3.](#) Mastership Phase

When timer T1 fires, a node enters the mastership phase. In this phase, each ring node N starts timer T2 and checks if it is master. If it is the node with the lowest loopback address of all nodes with the highest mastership values, N declares itself master by readvertising its ring node TLV with the M bit set.

When timer T2 fires, each node examines the ring node TLVs from all other nodes in the ring to identify the ring master. There should be exactly one; if not, each node restarts timer T2 and tries again. The nodes that set their M bit should be extra careful in advertising their M bit in subsequent tries.

### [4.4.](#) Ring Identification Phase

When there is exactly one ring master M, M enters the Ring Identification Phase. M indicates that it has successfully completed this phase by advertising ring link TLVs. This is the trigger for M's CW neighbor to enter the Ring Identification Phase. This phase passes CW until all ring nodes have completed ring identification.

In the Ring Identification Phase, a node X that has two or more IGP neighbors that belong to the ring picks one of them to be its CW ring neighbor. If X is the ring master, it also picks a node as its AC ring neighbor. If there are exactly two such nodes, this step is trivial. If not, X computes a ring that includes all nodes that have completed the Ring Identification Phase (as seen by their ring link TLVs) and further contains the maximal number of nodes that belong to the ring. Based on that, X picks a CW neighbor and inserts ring link TLVs with ring direction CW for each link to its CW neighbor; X also inserts a ring link TLV with direction AC for each link to its AC neighbor. Then, X determines its express links. These are links connected to ring nodes that are not ring neighbors. X advertises ring link TLVs for express links by setting the link direction to "express link".

### [4.5.](#) Ring Changes

The main changes to a ring are:

ring link addition;  
ring link deletion;  
ring node addition; and  
ring node deletion.

The main goal of handling ring changes is (as much as possible) not to perturb existing ring operation. Thus, if the ring master hasn't changed, all of the above changes should be local to the point of change. Link adds just update the IGP; signaling should take advantage of the new capacity as soon as it learns. Link deletions in the case of parallel links also show up as a change in capacity (until the last link in the bundle is removed.)

The removal of the last ring link between two nodes, or the removal of a ring node is an event that triggers protection switching. In a simple ring, the result is a broken ring. However, if a ring has express links, then it may be able to converge to a smaller ring with protection. Details of this process will be given in a future version.

The addition of a new ring node can also be handled incrementally. Again, the details of this process will be given in a future version.

## [5. Ring Signaling](#)

A future version of this document will specify protocol-independent details about ring LSP signaling.

## [6. Ring OAM](#)

Each ring node should advertise in its ring node TLV the OAM protocols it supports. Each ring node is expected to run a link-level OAM over each ring link. This should be an OAM protocol that both neighbors agree on. The default hello time is 3.3 millisecond.

Each ring node also sends OAM messages over each direction of its ring LSP. This is a multi-hop OAM to check LSP liveness; typically, BFD would be used for this. The node chooses the hello interval; the

default is once a second.

## [7.](#) Advanced Topics

### [7.1.](#) Half-rings

In some cases, a ring H may be incomplete, either because H is permanently missing a link (not just because of a failure), or because the link required to complete H is in a different IGP area. Either way, the ring discovery algorithm will fail. We call such a ring a "half-ring". Half-rings are sufficiently common that finding a way to deal with them effectively is a useful problem to solve. This topic will not be addressed in this document; that task is left for a future document.

### [7.2.](#) Hub Node Resilience

Let's call the node(s) that connect a ring to the rest of the network "hub node(s)" (usually, there are a pair of hub nodes.) Suppose a ring has two hub nodes H1 and H2. Suppose further that a non-hub ring node X wants to send traffic to some node Z outside the ring. This could be done, say, by having targeted LDP (T-LDP) sessions from H1 and H2 to X advertising LDP reachability to Z via H1 (H2); there

would be a two-label stack from X to reach Z. Say that to reach Z, X prefers H1; thus, traffic from X to Z will first go to H1 via a ring LSP, then to Z via LDP.

If H1 fails, traffic from X to Z will drop until the T-LDP session from H1 to Z fails, the IGP reconverges, and H2's label to Z is chosen. Thereafter, traffic will go from X to H2 via a ring LSP, then to Z via LDP. However, this convergence could take a long time. Since this is a very common and important situation, it is again a useful problem to solve. However, this topic too will not be addressed in this document; that task is left for a future document.

## [8.](#) Security Considerations

It is not anticipated that either the notion of MPLS rings or the extensions to various protocols to support them will cause new security loopholes. As this document is updated, this section will also be updated.

## 9. Acknowledgments

Many thanks to Pierre Bichon whose exemplar of self-organizing networks and whose urging for ever simpler provisioning led to the notion of promiscuous nodes.

## 10. IANA Considerations

There are no requests as yet to IANA for this document.

## 11. References

### 11.1. Normative References

[RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", [BCP 14](#), [RFC 2119](#), DOI 10.17487/RFC2119, March 1997, <<https://www.rfc-editor.org/info/rfc2119>>.

### 11.2. Informative References

[I-D.ietf-mpls-rfc4379bis]  
Kompella, K., Swallow, G., Pignataro, C., Kumar, N., Aldrin, S., and M. Chen, "Detecting Multi-Protocol Label Switched (MPLS) Data Plane Failures", [draft-ietf-mpls-rfc4379bis-09](#) (work in progress), October 2016.

[RFC3209] Awduche, D., Berger, L., Gan, D., Li, T., Srinivasan, V., and G. Swallow, "RSVP-TE: Extensions to RSVP for LSP Tunnels", [RFC 3209](#), DOI 10.17487/RFC3209, December 2001, <<https://www.rfc-editor.org/info/rfc3209>>.

[RFC3630] Katz, D., Kompella, K., and D. Yeung, "Traffic Engineering (TE) Extensions to OSPF Version 2", [RFC 3630](#), DOI 10.17487/RFC3630, September 2003, <<https://www.rfc-editor.org/info/rfc3630>>.

[RFC5036] Andersson, L., Ed., Minei, I., Ed., and B. Thomas, Ed.,

"LDP Specification", [RFC 5036](#), DOI 10.17487/RFC5036, October 2007, <<https://www.rfc-editor.org/info/rfc5036>>.

[RFC5305] Li, T. and H. Smit, "IS-IS Extensions for Traffic Engineering", [RFC 5305](#), DOI 10.17487/RFC5305, October 2008, <<https://www.rfc-editor.org/info/rfc5305>>.

[RFC8402] Filsfils, C., Ed., Previdi, S., Ed., Ginsberg, L., Decraene, B., Litkowski, S., and R. Shakir, "Segment Routing Architecture", [RFC 8402](#), DOI 10.17487/RFC8402, July 2018, <<https://www.rfc-editor.org/info/rfc8402>>.

#### Authors' Addresses

Kireeti Kompella  
Juniper Networks, Inc.  
1133 Innovation Way  
Sunnyvale, CA 94089  
USA

Email: [kireeti.kompella@gmail.com](mailto:kireeti.kompella@gmail.com)

Luis M. Contreras  
Telefonica  
Ronda de la Comunicacion  
Sur-3 building, 3rd floor  
Madrid 28050  
Spain

Email: [luismiguel.contrerasmurillo@telefonica.com](mailto:luismiguel.contrerasmurillo@telefonica.com)

URI: <http://people.tid.es/LuisM.Contreras/>