SPRING
Internet-Draft
Intended status: Standards Track
Expires: April 25, 2019

K. Kompella
A. Deshmukh
R. Torvi
Juniper Networks, Inc.

October 22, 2018

Resilient MPLS Rings
draft-kompella-spring-rmr-00

Abstract

This document describes the use of the SPRING MPLS data plane for Resilient MPLS Rings. It describes how to create the bidirectional ring LSPs with SPRING, and how protection works.

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

Copyright Notice

Table of Contents

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; [I-D.ietf-mpls-rmr] shows how this can be done. [I-D.ietf-teas-rsvp-rmr-extension] shows how RSVP-TE [RFC3209] can be used to signal RMR ring LSPs. [I-D.ietf-mpls-ldp-rmr-extensions] shows how LDP [RFC5036] can be used to signal RMR LSPs. This document shows how SPRING SID bindings can be used to create RMR LSPs, how the basic bidirectional LSPs are set up, and how protection works.

While RMR looks at rings potentially with "express links", this document focuses on simple rings. These are most common in access networks. Future revisions will look at more general rings.

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 <= i < n} of V. The directed edges {(R_i, R_i+1) and (R_i+1, R_i), 0 <= 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.

```
                     R0 . . . R1
                   .             .
                R7                 R2
   Anti-     |  .        Ring        .  |
   Clockwise |  .                    .  | Clockwise
             v  .      RID = 17      .  v
                R6                 R3
                   .             .
                     R5 . . . R4
```

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 size: The ring size for a given instantiation is N. This can change as nodes are added or removed, or go up or down.

Ring links: Links that connnect ring neighbors.

Express links: Links that connnect 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).

NS_k: Node SID for node R_k. Note that index arithmetic is done modulo the ring size N.

CAS_k, AAS_k: Clockwise adjacency SID at R_k, i.e., link R_k, R_k+1 and anticlockwise adjacency SID R_k, R_k-1 respectively. Note that index arithmetic is done modulo the ring size N.

CSS_jk: A clockwise node SID stack, typically with one or two SIDs, to be pushed by R_j to reach R_k in a clockwise direction.

ASS_jk: An anticlockwise node SID stack, typically with one or two SIDs, to be pushed by R_j to reach R_k in an anticlockwise direction.

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.

The goals of this document are to present mechanisms for improved 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 are achieved using extensions to existing IGP. This document shows how to do this using SPRING techniques, in particular, node SIDs. Note that in a simple ring topology, there is no need for complex algorithms to find loop-free protection paths.

3. Theory of Operation

We assume that a ring R has been configured, IGP advertisements have been made, and ring discovery is complete ([I-D.ietf-mpls-rmr]). We also assume that node and adjacency SIDs have been distributed.

3.1. Installing Primary LFIB Entries

Ring LSPs are not provisioned. Once a ring node R_i knows its RID, its ring links and directions, it kicks off ring LSP computation automatically. In particular, R_j computes clockwise and anticlockwise SID stacks CSS_jk and ASS_jk to node R_k. R_j then installs two FIB entries for R_k, CSS_jk and ASS_jk. It is up to an application to choose whether to go clockwise or anticlockwise from R_j to R_k.

R_j also computes CSS_jj and ASS_jj. Clearly, R_j does not act as ingress for its own LSPs. However, R_j can send OAM messages, for example, an MPLS ping or traceroute ([I-D.ietf-mpls-rfc4379bis]), using CSS_jj or ASS_jj, to test the entire ring LSP anchored at R_j in both directions.

3.2. Installing Protection LFIB Entries

At the same time that R_j sets up its primary clockwise and anticlockwise SID stacks, it sets up protection for each other node R_k. R_j does this by installing a protection SID stack for the node SID to R_k, NS_k. If the shortest path to R_k is clockwise, then the protection SID stack for NS_k is ASS_jk. Otherwise, it is CSS_jk.

Similarly, the protection entry for an adjacency SID CAS_j is ASS_j,j+1 and for AAS_j is CSS_j,j-1.

3.3. Protection

If a node R_j detects a failure from R_j+1 -- either all links to R_j+1 fail, or R_j+1 itself fails, R_j switches traffic on all CW node and adjacency SIDs to their protection LFIB entries. This switchover can be very fast, as the protection LFIB entries can be preprogrammed. Fast detection and fast switchover lead to minimal traffic loss.

R_j then sends an indication to R_j-1 that the CW direction is not working, so that R_j-1 can similarly switch traffic to the AC direction. This can be by an IGP update; other, potentially quicker, mechanisms would be preferable. These indications propagate AC until each traffic source on the ring AC of the failure is aware of the failure. Thus, within a short period, traffic will be flowing on the reverse path than that which was chosen, since there is a failure on the ring.

4. 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.

5. IANA Considerations

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

6. References

6.1. 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>.

6.2. Informative References

[I-D.ietf-mpls-ldp-rmr-extensions]
Esale, S. and K. Kompella, "LDP Extensions for RMR", 2018.

[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.

[I-D.ietf-mpls-rmr]
Kompella, K. and L. Contreras, "Resilient MPLS Rings", 2018.

[I-D.ietf-teas-rsvp-rmr-extension]
Deshmukh, A. and K. Kompella, "RSVP Extensions for RMR", 2018.

[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>.

[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>.

Authors' Addresses

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

Email: kireeti.kompella@gmail.com

Abhishek Deshmukh
Juniper Networks, Inc.
1133 Innovation Way
Sunnyvale, CA 94089
USA

Email: adeshmukh@juniper.net

Ravi Torvi
Juniper Networks, Inc.
1133 Innovation Way
Sunnyvale, CA 94089
USA

Email: rtorvi@juniper.net