Resilient MPLS Rings

draft-kompella-mpls-rmr

Kireeti Kompella

IETF 91

JUNIPER NETWORKS

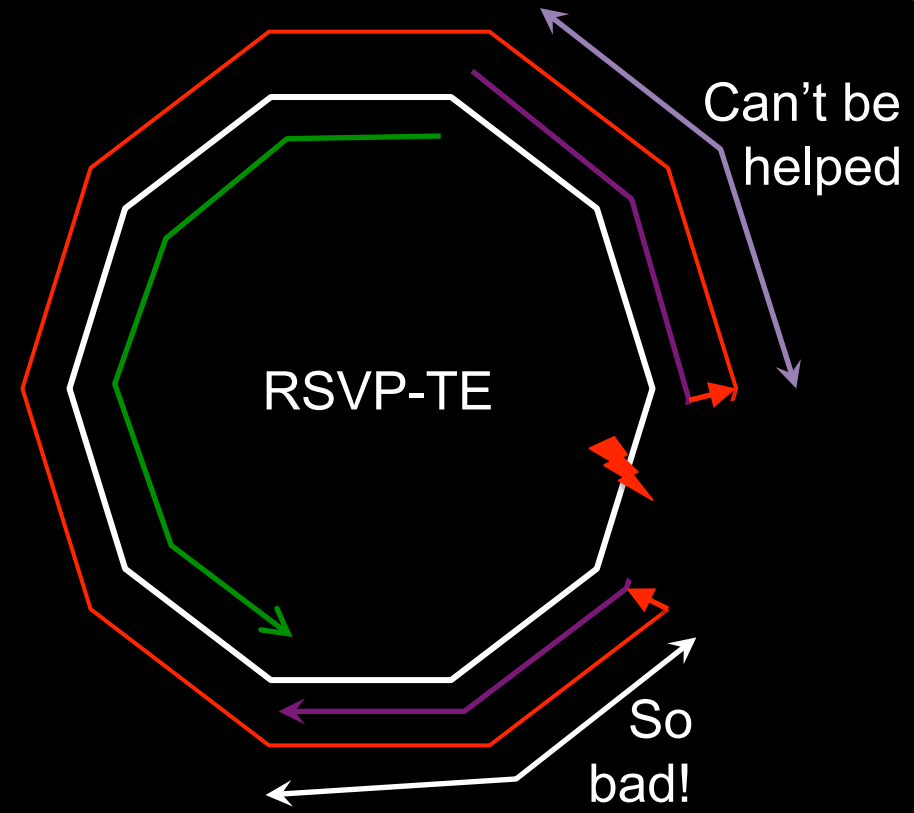
Using MPLS in Ring Topologies

- A ring is the most efficient topology that offers resilience …
  - … but MPLS resilience in rings is far from efficient
- Rings are often used in access and aggregation where bandwidth is precious …
  - … but pre-assigning bandwidth may be wasteful
- Rings are a simple topology, and there are lots of them …
  - … so configuring them should be as simple as possible

MPLS for Ring Transport

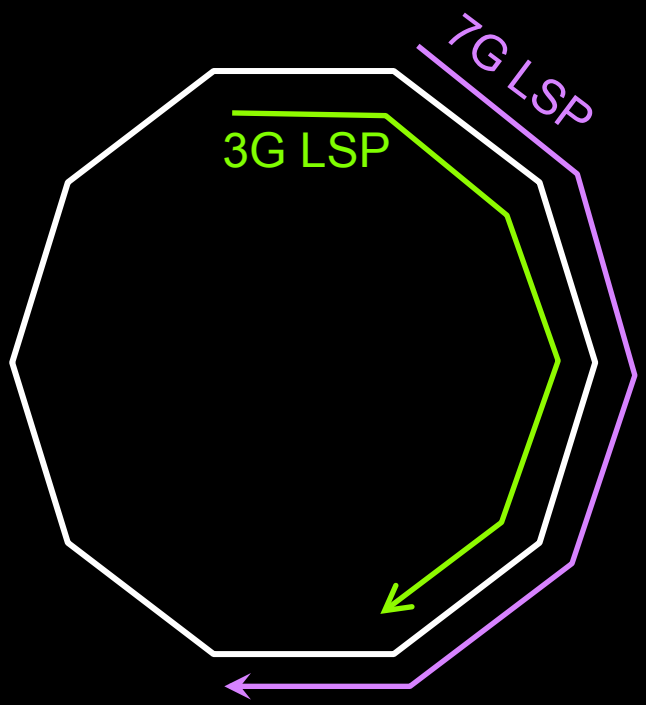
- The goal here is to identify and address issues in running MPLS as a transport protocol in access rings
  - Basically, to do what MPLS-TP set out to do, to replace TDM with packet, but in an efficient way
- To achieve this, we use IGP for ring discovery and RSVP-TE or LDP for signaling – but in a new way
  - Some of the differences from "traditional" signaling will be explained

Illustrations of Issues with MPLS in Rings



Consider all the extensions to the base LFA technique that there are to get to 100% coverage!

Bandwidth Allocation



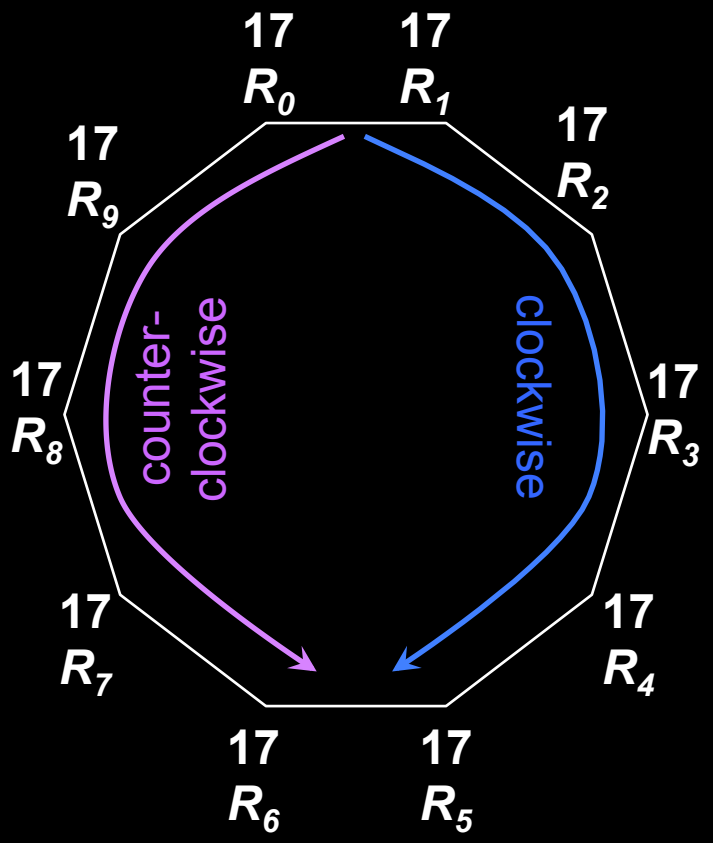
Assume a **10G** ring. If **7G** is allocated for the purple LSP, then the green LSP can only have **3G**. Is this the best allocation? Hard to say until the services using these LSPs are known.

For example, if the 7G LSP is not fully utilized, and more bandwidth is needed for the 3G LSP, *both LSPs* have to be disrupted to adjust the bandwidth!

New Paradigm: Resilient MPLS Rings

- Don't configure LSPs … configure MPLS rings
- Don't configure and signal n(n-1) LSPs …
  - LSPs come up on their own; no need for EROs
- Don't configure bandwidths …
  - bandwidths are deduced from traffic or services
- Don't configure protection paths, bypass LSPs or detours …
  - protection happens naturally
- Don't configure hierarchical LSPs …
  - hierarchy happens automatically

Configuring an MPLS Ring



What you see here is a ring with ring ID 17 and with 10 nodes R_0 to R_9

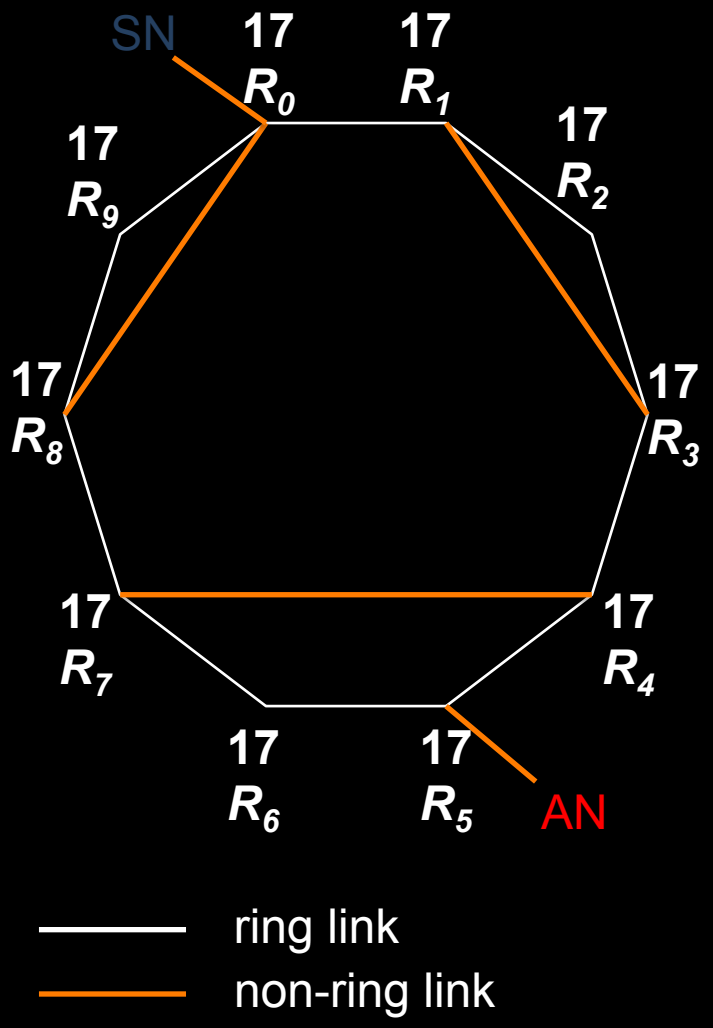
The *ring ID* is used to uniquely identify a ring within an AS.

The IGP is used to discover ring neighbors and ring links.

Links between a pair of ring nodes may belong to multiple rings

Links between a pair of ring nodes are automatically bundled into a single logical link.

Configuring an MPLS Ring



Note that rings are not “pure” – there will be non-ring links.

These can be “optical bypass” links, or links to access or service nodes.

Furthermore, there can be hierarchical rings, stacked rings, etc.

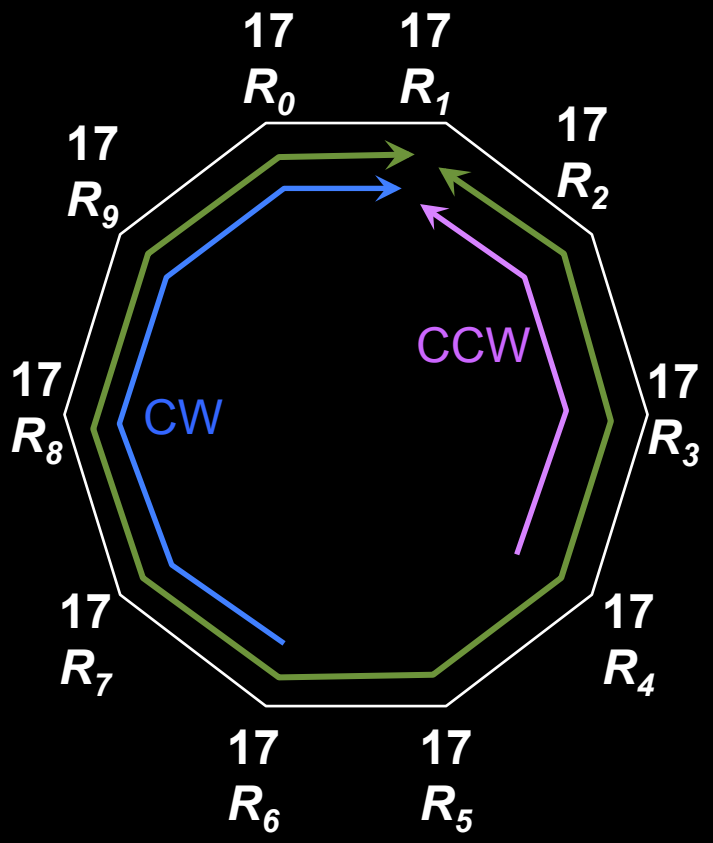
All these need to be taken into account in ring definition and discovery.

Ring Auto-discovery

Requirements: ring nodes are assigned to a *ring ID*

- Ring links are discovered and “auto-bundled”
  - Non-ring links are identified as such
- All nodes agree on *clockwise* and *counterclockwise*
- Each node knows its CW and CCW neighbor
- Node insertion and removal is handled

Ring LSPs: Basics



A ring LSP **starts** and **ends** at the same node – a pair of counter-rotating LSPs.

One direction is called *clockwise* (CW), and the other *counterclockwise* (CCW).

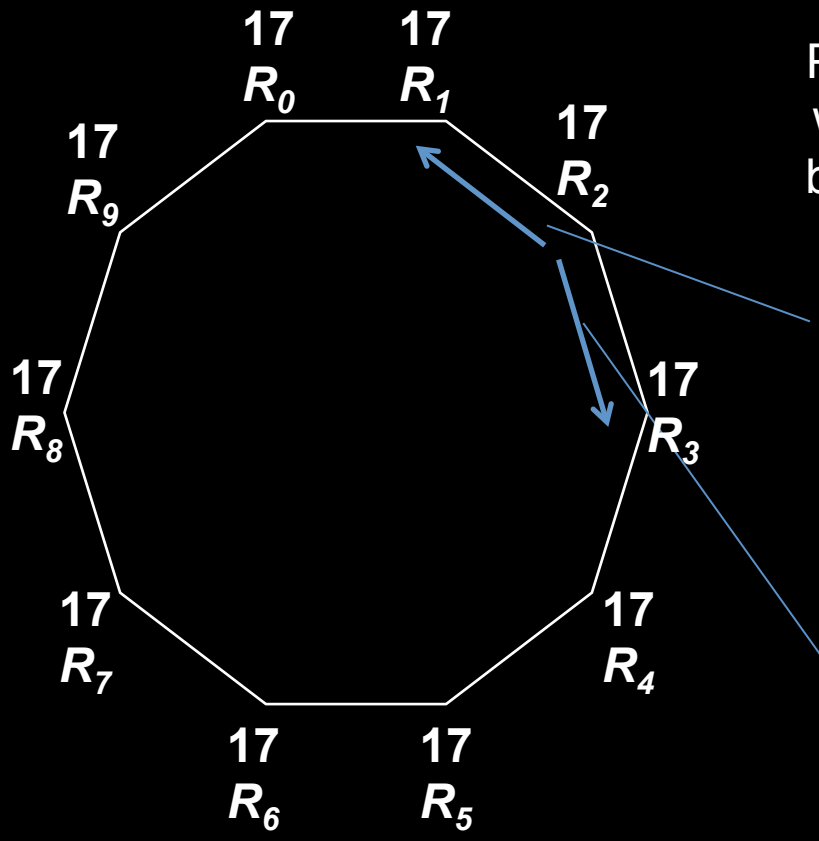
Ring LSP RL_1 starts and ends on R_1 and is a multipoint LSP with egress R_1

Each node can send traffic to R_1 either CW (e.g., R_6) or CCW (e.g., R_4).

Similarly, each node R_0, R_2, R_3, …, and R_9 has a ring LSP.

A ring of N nodes has N ring LSPs, not N*(N-1)!

Ring LSPs: Signaling

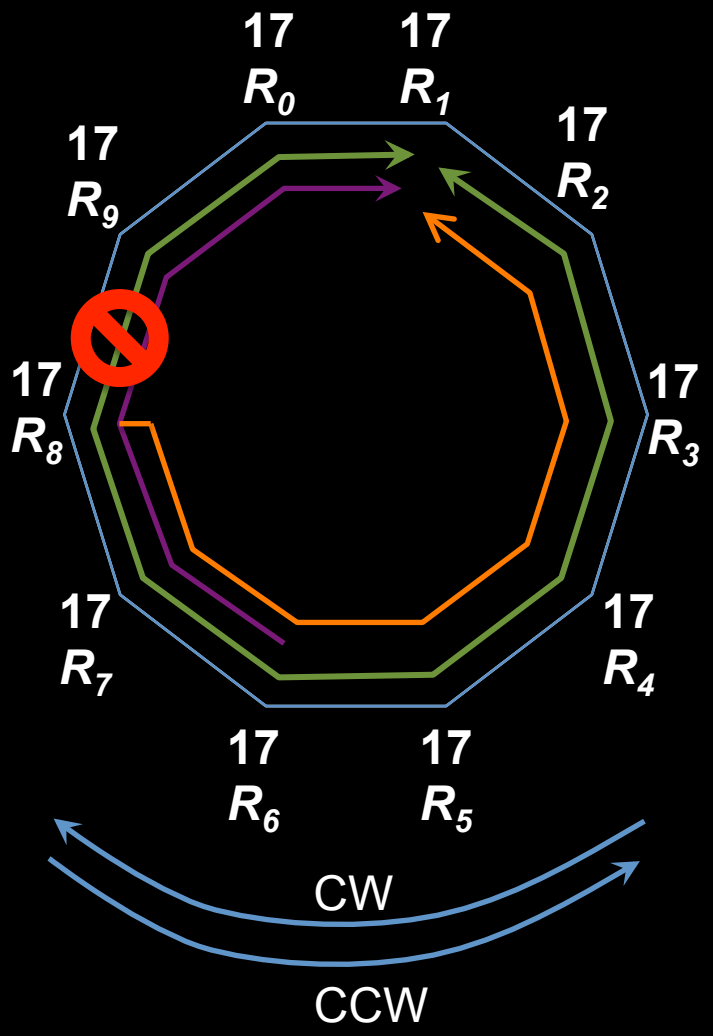


Path messages are automatically sent when an MPLS ring is configured, not because of specific LSP configuration.

Node 2 sends 10 Resv messages CCW, one for each of ring LSPs 1, 2, 3, …, 10. These contain CW labels, and establish the CW LSPs.

Node 2 also sends 10 Path messages CW, one for each of ring LSPs 1, 2, 3, …, 10. These contain CCW labels, and establish the CCW LSPs.

Ring LSPs: Protection



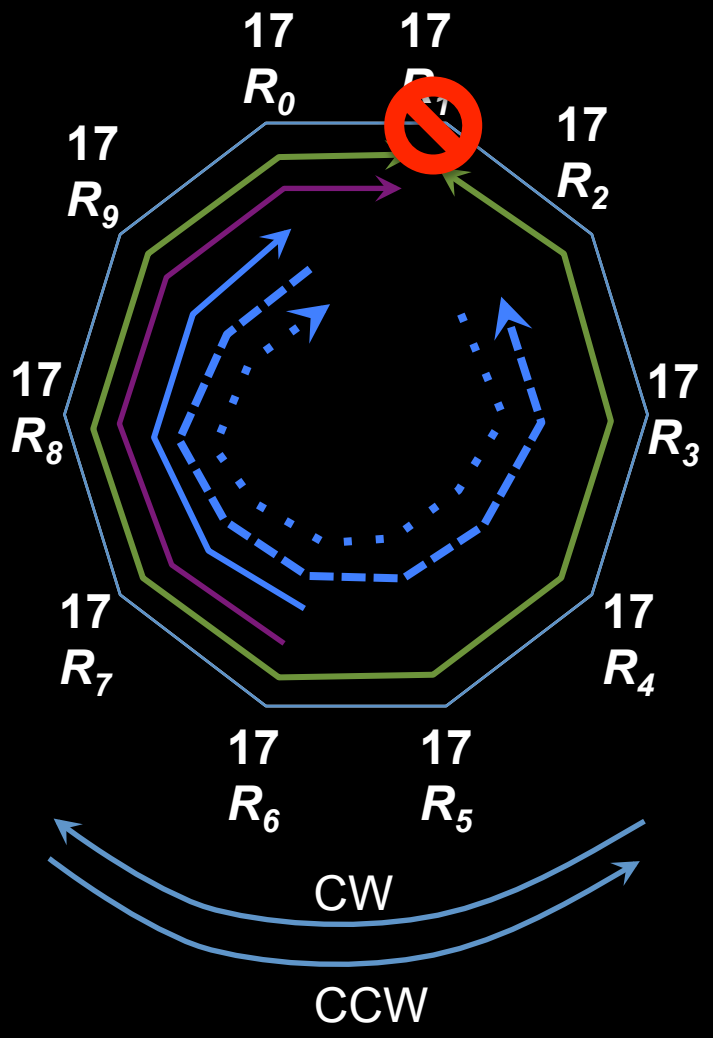
Since ring LSP RL1 is bidirectional, there is a path from node 8 to node 1 in both directions, CCW (via node 9) and CW (via node 7). This is used to protect ring LSP 1, say from node 6 to node 1.

If the link between node 8 and node 9 fails, traffic to node 1 is immediately put on the reverse LSP to node 1.

When the notification of the failure propagates to node 7, the traffic on RL1 is diverted at node 7 to the upstream direction.

When node 6 learns, it sends the traffic CCW to node 1. Effectively, the traffic has switched to the other direction.

Ring LSPs: Node Failure



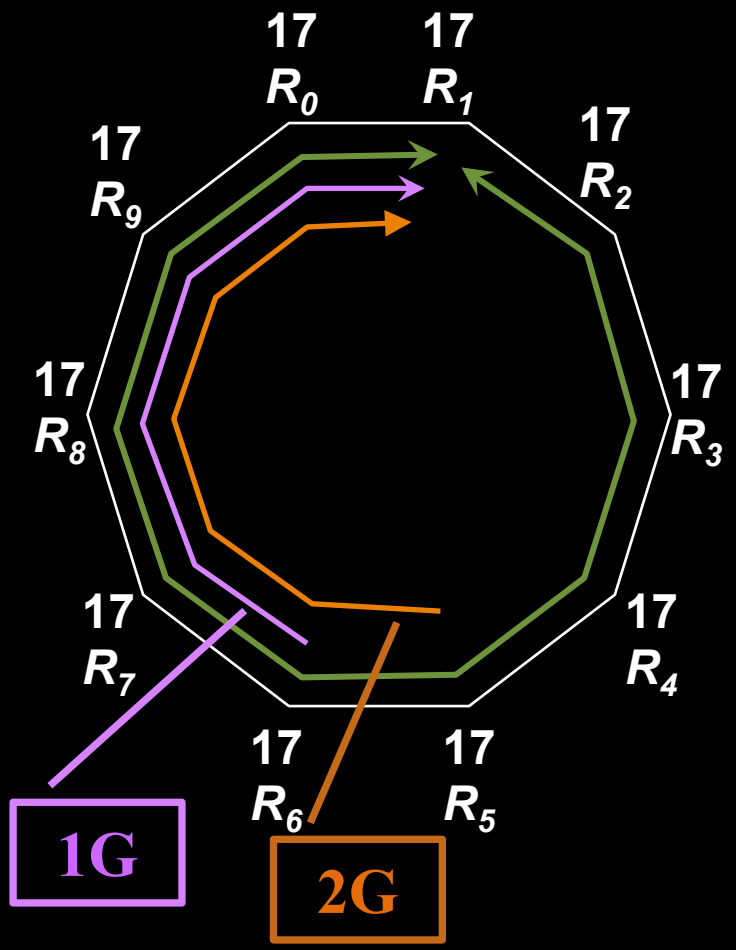
Node (say R_1) failure is similar to link failure – stuff just works. Of course, RL_1 clearly cannot recover – its egress has failed.

However, there is the danger of a loop:

1. R_0 protects by sending traffic CCW;
2. R_2 protects by sending traffic CW!

This can be dealt with by TTL …
… or by adding a new ESPL to indicate failure recovery

Ring LSPs: Bandwidth Management



Ring LSP RL_1 starts with 0 bandwidth. As services are provisioned over RL_1, their bandwidths are added to the LSP from where they enter the ring to the egress node.

Say a 1G PW is provisioned from node 6 to node 1. The LSP attempts to increase the bandwidth from node 6 to node 1. If successful, the service is accepted. Similarly for a 2G PW from node 5 to node 1.

The resulting signaled bandwidth in the CW direction for ring LSP 1 is 0 from node 1 to node 5; node 5 signals a bandwidth of 1G; node 6-10 signal a bandwidth of 3G.

Conclusion

- Rings are indeed a special topology
- MPLS on rings needs to be:
  - easy from configuration point of view;
  - efficient from protection PoV;
  - more flexible from a bandwidth PoV
- Ring LSPs appear to meet these requirements

- Thanks for your questions and comments on the list
  - Please keep them coming!