Network Working Group   Pierre Francois
Internet Draft   Individual Contributor
Intended status: Informational   Clarence Filsfils
Expires: June 2017   Ahmed Bashandy
                                                           Cisco Systems, Inc.
                                                           Stephane Litkowski
                                                                        Orange
                                                             December 14, 2016

Loop avoidance using Segment Routing
draft-francois-rtgwg-segment-routing-uloop-01

Abstract

This document presents a mechanism aimed at providing loop avoidance in the case of IGP network convergence event. The solution relies on the temporary use of SR policies ensuring loop-freeness over the post-convergence paths from the converging node to the destination.

Copyright Notice

Table of Contents

1. Introduction

Forwarding loops happen during the convergence of the IGP, as a result of transient inconsistency among forwarding states of the nodes of the network.

This document provides a mechanism leveraging Segment Routing to ensure loop-freeness during the IGP reconvergence process following a link-state change event.

We use Figure 1 to illustrate the mechanism. In this scenario, all the IGP link metrics are 1, excepted R3-R4 whose metric is 100. We consider the traffic from S to D.

```
                       +-------R1------R2---+
                      /                      \
                     /                        \
                S---R0                         R3-----D
                     \                        /
                      \                      /
                       +-------R5------R4---+
                                          100
```

Figure 1 Illustrative scenario, failure of link R2-R3

When the link between R2 and R3 fails, traffic sent from S to D, initially flowing along S-R0-R1-R2-R3-D is subject to transient forwarding loops while routers update their forwarding state for destination D. For example, if R0 updates its FIB before R5, packets for D may loop between R0 and R5. If R5 updates its FIB before R4, packets for D may loop between R5 and R4.

Using segment routing, a headend can enforce an explicit path without creating any state along the desired path. As a result, a converging node can enforce traffic on the post-convergence path in a loop-free manner, using a list of segments (typically short). We suggest that the converging node enforces its post-convergence path to the destination when applying this behavior to ease operation (predictability of path, less capacity planning issues ...); nodes converge over their new optimal path, but temporarily use an SR policy to ensure loop-freeness over that path.

In our example, R0 can temporarily steer traffic destined to D over SR path [NodeSID(R4), AdjSID(R4->R3), D]. By doing so, packets for D will be forwarded by R5 as per NodeSID(R4), and by R4 as per AdjSID(R4->R3). From R3 on, the packet is forwarded as per destination D. As a result, traffic follows the desired path, regardless of the forwarding state for destination D at R5 and R4. After some time, the normal forwarding behavior (without using an SR policy) can be applied; routers will converge to their final forwarding state, still consistently forwarding along the post-convergence paths across the network.

1.1. Conventions used in this document

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 RFC-2119

In this document, these words will appear with that interpretation only when in ALL CAPS. Lower case uses of these words are not to be interpreted as carrying RFC-2119 significance.

2. Loop-free two-stage convergence process

Upon a topology change, when a node R converging for destination D does not trust the loop-freeness of its post-convergence path for destination D, it applies the following two-stage convergence process for destination D.

Stage 1: After computing the new path to D, for a configured amount of time C, R installs a FIB entry for D that steers packets to D via a loop-free SR path. C is assumed to be configured as per the worst-case convergence time of a node, network-wide. The SR path is computed when the event occurs.

Stage 2: After C elapses, R installs the normal post-convergence FIB entry for D, i.e. without any additional segments inserted that ensure the loop-free property.

Loop-freeness is ensured during this process, because:

1. Paths made of non up-to-date routers are loop-free.

Routers which forward as per the initial state of the network are consistent.

2. A packet reaching a node in stage 1 is ensured to reach its destination.

When a packet reaches a router in stage 1, it is steered on a SR path ensuring a loop-free post-convergence path, whatever the state of other routers on the path.

3. Paths made of a mix of routers in stage 1 and stage 2 are consistent.

After C milliseconds, all routers are forwarding as per their post-convergence paths, either expressed classically or as a loop-free SR path.

In our example, when R2-R3 fails, R0 forwards traffic for destination D over SR Path [NodeSID(R4), AdjSID(R4->R3), D], for C milliseconds. During that period, packets sent by R0 to D are loop-free as per the application of the policy. When C elapses, R0 now uses its normal post-convergence path to the destination, forwarding packets for D as is to R5.

R5 also implements loop avoidance, and has thus temporarily used a loop-avoiding SR policy for D. This policy is [AdjSID(R4->R3), D], oif R5->R4. If R5 is still applying the stage 1 behavior, the packet will be forwarded using this policy, and will thus safely reach the destination. If R5 also had moved to stage 2, it forwards the packet as per its normal post-convergence path, via R4. The forwarding state of R4 for D at stage 1 and stage 2 are the same: oif R4->R3, as forwarding packets for destination D as is to R3 ensures a loop-free post-convergence path.

3. Computing loop-avoiding SR policies

The computation to turn a post-convergence path into a loop-free list of segments is outside the scope of this document. It is a local behavior at a node.

In a future revision of this document, we may provide a reference approach to compute loop-avoiding policies for link up, link metric increase, link down, link metric decrease, node up, and node down events. TI-LFA Repair Tunnel

4. Analysis

In this section, we review the main characteristics of the proposed solution. These characteristics are illustrated in [3].

4.1. Incremental Deployment

There is no requirement for a full network upgrade to get benefits from the solution.

(1) Nodes that are upgraded bring benefit for the traffic passing through them.

(2) Nodes that are not upgraded to support SR-based loop-avoidance will cause the micro-loops that they were causing before, unless they get avoided by the local behavior of a node supporting the behavior.

4.2. No impact on capacity planning

By ensuring loop-free post-convergence paths, the behavior remains in line with the natural expected convergence process of the IGP. Enabling SR-based loop-avoidance hence does not require consideration for capacity planning, compared to any loop avoidance mechanism that lets traffic follow a different path than the post-convergence one. The behavior is local. Nothing is expected from remote nodes except the basic support of Prefix and Adjacency SID's.

5. Security Considerations

The behavior described in this document is internal functionality to a router that result in the ability to explicitly steer traffic over the post convergence path after a remote topology change in a manner that guarantees loop freeness. As such no additional security risk is introduced by using the mechanisms proposed in this document.

6. IANA Considerations

No requirements for IANA

7. Contributors

Additional contributors: Bruno Decraene and Peter Psenak.

8. References

8.1. Normative References

8.2. Informative References

[1] Filsfils, C., Previdi, S., Decraene, B., Litkowski, S., and R. Shakir, "Segment Routing Architecture", draft-ietf-spring-segment-routing-10 (work in progress), November 2016.

[2] Shand, M. and S. Bryant, "IP Fast Reroute Framework", RFC 5714, January 2010.

[3] Litkowski, S., "Avoiding micro-loops using Segment Routing", MPLS World Congress , 2016.

9. Acknowledgments

This document was prepared using 2-Word-v2.0.template.dot.

Authors' Addresses

Pierre Francois
pfrpfr@gmail.com

Ahmed Bashandy
Cisco Systems
170 West Tasman Dr, San Jose, CA 95134, USA
Email: bashandy@cisco.com

Clarence Filsfils
Cisco Systems
Brussels, Belgium
Email: cfilsfil@cisco.com

Stephane Litkowski
Orange
Email: stephane.litkowski@orange.com