Network Working Group   A. Bashandy
Internet Draft   C. Filsfils
Intended status: Standard Track   Cisco Systems
Expires: July 2018   Bruno Decraene
                                                  Stephane Litkowski
                                                              Orange
                                                     Pierre Francois
                                              Individual Contributor
                                                    January 19, 2018

          Topology Independent Fast Reroute using Segment Routing
             draft-bashandy-rtgwg-segment-routing-ti-lfa-02

Abstract

   This document presents Topology Independent Loop-free Alternate Fast
   Re-route (TI-LFA), aimed at providing protection of node and
   adjacency segments within the Segment Routing (SR) framework. This
   Fast Re-route (FRR) behavior builds on proven IP-FRR concepts being
   LFAs, remote LFAs (RLFA), and remote LFAs with directed forwarding
   (DLFA). It extends these concepts to provide guaranteed coverage in
   any IGP network. A key aspect of TI-LFA is the FRR path selection
   approach establishing protection over post-convergence paths from
   the point of local repair, dramatically reducing the operational
   need to control the tie-breaks among various FRR options.

Status of this Memo

Copyright Notice

Table of Contents

1. Introduction

   Segment Routing aims at supporting services with tight SLA
   guarantees [1]. By relying on segment routing this document
   provides a local repair mechanism for standard IGP shortest path
   capable of restoring end-to-end connectivity in the case of a
   sudden directly connected failure of a network component. Non-SR
   mechanisms for local repair are beyond the scope of this document.
   Non-local failures are addressed in a separate document [5].

   For each destination in the network, TI-LFA prepares a data-plane
   switch-over to be activated upon detection of the failure of a
   link used to reach the destination. TI-LFA provides protection in
   the event of any one of the following: single link failure,
   single node failure, or single local SRLG failure. In link
   failure mode, the destination is protected assuming the failure of
   the link. In node protection mode, the destination is protected
   assuming that the neighbor connected to the primary link has
   failed. In local SRLG protecting mode, the destination is
   protected assuming that a configured set of links sharing fate
   with the primary link has failed (e.g. a linecard).

   Protection applies to traffic which traverses the PLR. Traffic
   which does NOT traverse the PLR remains unaffected.

   Using segment routing, there is no need to establish TLDP sessions
   with remote nodes in order to take advantage of the applicability
   of remote LFAs (RLFA) or remote LFAs with directed forwarding
   (DLFA)[2]. As a result, preferring LFAs over RLFAs or DLFAs, as
   well as minimizing the number of RLFA or DLFA repair nodes is not
   required. This allows for a protection path selection approach
   meeting operational needs rather than a topologically constrained
   one.

   Using SR, there is no need to create state in the network in order
   to enforce an explicit FRR path. As a result, we can use
   optimized detour paths for each specific destination and for each
   type of failure without creating additional forwarding state.
   Also, the mode of protection (link, node, SRLG) is not constrained
   to be network wide or node wide, but can be managed on a per
   interface basis.

   Building on such an easier forwarding environment, the FRR
   behavior suggested in this document tailors the repair paths over
   the post-convergence path from the PLR to the protected
   destination, given the enabled protection mode for the interface.

   As the capacity of the post-convergence path is typically planned
   by the operator to support the post-convergence routing of the
   traffic for any expected failure, there is much less need for the
   operator to tune the decision among which protection path to
   choose. The protection path will automatically follow the natural
   backup path that would be used after local convergence. This also
   helps to reduce the amount of path changes and hence service
   transients: one transition (pre-convergence to post-convergence)
   instead of two (pre-convergence to FRR and then post-convergence).

```
                     L        ____
                   S----F--{____}----D
                  /\      |           /
                  |  |    | _____  /
                  |__}---Q{_____}
```
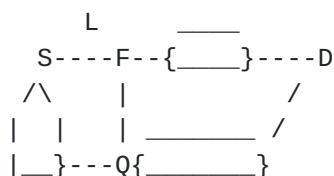
                    Figure 1 TI-LFA Protection

   We use Figure 1 to illustrate the TI-LFA approach.

   The Point of Local Repair (PLR), S, needs to find a node Q (a repair
   node) that is capable of safely forwarding the traffic to a
   destination D affected by the failure of the protected link L, a set
   of adjacent links including L (local SRLG), or the node F itself.
   The PLR also needs to find a way to reach Q without being affected
   by the convergence state of the nodes over the paths it wants to use
   to reach Q.

   In Section 2 we define the main notations used in the document.
   They are in line with [2].

   In Section 3, we suggest to compute the P-Space and Q-Space
   properties defined in Section 2, for the specific case of nodes
   lying over the post-convergence paths towards the protected
   destinations.

Using the properties defined in Section 3, we describe how to
compute protection lists that encode a loopfree post-convergence
towards the destination, in Section 4.

Finally, we define the segment operations to be applied by the PLR
to ensure consistency with the forwarding state of the repair node,
in Section 5.

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. Terminology

We define the main notations used in this document as the following.

We refer to "old" and "new" topologies as the LSDB state before and
after the considered failure.

SPT_old(R) is the Shortest Path Tree rooted at node R in the initial
state of the network.

SPT_new(R, X) is the Shortest Path Tree rooted at node R in the
state of the network after the resource X has failed.

Dist_old(A,B) is the distance from node A to node B in SPT_old(A).

Dist_new(A,B, X) is the distance from node A to node B in
SPT_new(A,X).

Similarly to [4], we rely on the concept of P-Space and Q-Space for
TI-LFA.

The P-Space P(R,X) of a node R w.r.t. a resource X (e.g. a link S-F,
a node F, or a local SRLG) is the set of nodes that are reachable
from R without passing through X. It is the set of nodes that are
not downstream of X in SPT_old(R).

The Extended P-Space P'(R,X) of a node R w.r.t. a resource X is the
set of nodes that are reachable from R or a neighbor of R, without
passing through X.

The Q-Space Q(D,X) of a destination node D w.r.t. a resource X is
the set of nodes which do not use X to reach D in the initial state
of the network. In other words, it is the set of nodes which have D
in their P-Space w.r.t. S-F, F, or a set of links adjacent to S).

A symmetric network is a network such that the IGP metric of each
link is the same in both directions of the link.

3. Intersecting P-Space and Q-Space with post-convergence paths

In this section, we suggest to determine the P-Space and Q-Space
properties of the nodes along the post-convergence paths from the
PLR to the protected destination and compute an SR-based explicit
path from P to Q when they are not adjacent. Such properties will
be used in Section 4 to compute the TI-LFA repair list.

3.1. P-Space property computation for a resource X

A node N is in P(R, X) if it is not downstream of X in SPT_old(R).
X can be a link, a node, or a set of links adjacent to the PLR. A
node N is in P'(R,X) if it is not downstream of X in SPT_old(N),
for at least one neighbor N of R.

3.2. Q-Space property computation for a link S-F, over post-convergence paths

We want to determine which nodes on the post-convergence path from
the PLR to the destination D are in the Q-Space of destination D
w.r.t. link S-F.

This can be found by intersecting the post-convergence path to D,
assuming the failure of S-F, with Q(D, S-F).

3.3. Q-Space property computation for a set of links adjacent to S, over post-convergence paths

We want to determine which nodes on the post-convergence path from
the PLR to the destination D are in the Q-Space of destination D
w.r.t. a set of links adjacent to S (S being the PLR). That is, we
aim to find the set of nodes on the post-convergence path that use
none of the members of the protected set of links, to reach D.

This can be found by intersecting the post-convergence path to D,
assuming the failure of the set of links, with the intersection
among Q(D, S->X) for all S->X belonging to the set of links.

[3.4](). Q-Space property computation for a node F, over post-convergence
   paths

   We want to determine which nodes on the post-convergence from the
   PLR to the destination D are in the Q-Space of destination D w.r.t.
   node F.

   This can be found by intersecting the post-convergence path to D,
   assuming the failure of F, with Q(D, F).

[4](). TI-LFA Repair Tunnel

   The TI-LFA repair tunnel consists of an outgoing interface and a
   list of segments (repair list) to insert on the SR header. The
   repair list encodes the explicit post-convergence path to the
   destination, which avoids the protected resource X.

   The TI-LFA repair tunnel is found by intersecting P(S,X) and Q(D,X)
   with the post-convergence path to D and computing the explicit SR-
   based path EP(P, Q) from P to Q when these nodes are not adjacent
   along the post convergence path. The TI-LFA repair list is
   expressed generally as (Node_SID(P), EP(P, Q)).

   Most often, the TI-LFA repair list has a simpler form, as described
   in the following sections.

[4.1](). The repair node is a direct neighbor

   When the repair node is a direct neighbor, the outgoing interface is
   set to that neighbor and the repair segment list is empty.

   This is comparable to a post-convergence LFA FRR repair.

[4.2](). The repair node is a PQ node

   When the repair node is in P(S,X), the repair list is made of a
   single node segment to the repair node.

   This is comparable to a post-convergence RLFA repair tunnel.

[4.3](). The repair is a Q node, neighbor of the last P node

   When the repair node is adjacent to P(S,X), the repair list is made
   of two segments: A node segment to the adjacent P node, and an
   adjacency segment from that node to the repair node.

   This is comparable to a post-convergence DLFA repair tunnel.

[4.4](). Connecting distant P and Q nodes along post-convergence paths

In some cases, there is no adjacent P and Q node along the post-
convergence path. However, the PLR can perform additional
computations to compute a list of segments that represent a loopfree
path from P to Q.

[5](). Protecting segments

In this section, we explain how a protecting router S processes the
active segment of a packet upon the failure of its primary outgoing
interface for the packet, S-F.

The behavior depends on the type of active segment to be protected.

[5.1](). The active segment is a node segment

The active segment is kept on the SR header, unchanged (1). The
repair list is inserted at the head of the list. The active segment
becomes the first segment of the inserted repair list.

Note (1): If the SRGB at the repair node is different from the SRGB
at the PLR, then the active segment must be updated to fit the SRGB
of the repair node.

In [Section 5.3](), we describe the node protection behavior of PLR S,
for the specific case where the active segment is a prefix segment
for the neighbor F itself.

[5.2](). The active segment is an adjacency segment

We define hereafter the FRR behavior applied by S for any packet
received with an active adjacency segment S-F for which protection
was enabled. We distinguish the case where this active segment is
followed by another adjacency segment from the case where it is
followed by a node segment.

[5.2.1](). Protecting [Adjacency, Adjacency] segment lists

If the next segment in the list is an Adjacency segment, then the
packet has to be conveyed to F.

To do so, S applies a "NEXT" operation on Adj(S-F) and then two
consecutive "PUSH" operations: first it pushes a node segment for F,
and then it pushes a protection list allowing to reach F while
bypassing S-F. For details on the "NEXT" and "PUSH" operations,
refer to [[6]()].

Upon failure of S-F, a packet reaching S with a segment list
matching [adj(S-F),adj(M),...] will thus leave S with a segment list
matching [RT(F),node(F),adj(M)], where RT(F) is the repair tunnel
for destination F.

In [Section 5.3.2](#), we describe the TI-LFA behavior of PLR S when
node protection is applied and the two first segments are Adjacency
Segments.

5.2.2. Protecting [Adjacency, Node] segment lists

If the next segment in the stack is a node segment, say for node T,
the packet segment list matches [adj(S-F),node(T),...].

A first solution would consist in steering the packet back to F
while avoiding S-F. To do so, S applies a "NEXT" operation on
Adj(S-F) and then two consecutive "PUSH" operations: first it pushes
a node segment for F, and then it pushes a repair list allowing to
reach F while bypassing S-F.

Upon failure of S-F, a packet reaching S with a segment list
matching [adj(S-F),node(T),...] will thus leave S with a segment
list matching [RT(F),node(F),node(T)].

Another solution is to not steer the packet back via F but rather
follow the new shortest path to T. In this case, S just needs to
apply a "NEXT" operation on the Adjacency segment related to S-F,
and push a repair list redirecting the traffic to a node Q, whose
path to node segment T is not affected by the failure.

Upon failure of S-F, packets reaching S with a segment list matching
[adj(L), node(T), ...], would leave S with a segment list matching
[RT(Q),node(T), ...]. Note that this second behavior is the one
followed for node protection, as described in [Section 5.3.1](#).

5.3. Protecting SR policy midpoints against node failure

As planned in the previous version of this document, we describe the
behavior of a node S configured to interpret the failure of link S-
>F as the node failure of F, in the specific case where the active
segment of the packet received by S is a Prefix SID of F represented
as "F"), or an Adjacency SID for the link S-F (represented as "S-
>F").

5.3.1. Protecting {F, T, D} or {S->F, T, D}

We describe the protection behavior of S when

1. the active segment is a prefix SID for a neighbor F, or an
   adjacency segment S->F

2. the primary interface used to forward the packet failed

3. the segment following the active segment is a prefix SID (for
   node T)

4. node protection is active for that interface.

The TILFA Node FRR behavior becomes equivalent to:

1. Pop; the segment F or S->F is removed

2. Confirm that the next segment is in the SRGB of F, meaning that
   the next segment is a prefix segment, e.g. for node T

3. Identify T (as per the SRGB of F)

4. Pop the next segment and push T's segment based on the local SRGB

5. forward the packet according to T.

5.3.2. Protecting {F, F->T, D} or {S->F, F->T, D}

We describe the protection behavior of S when

1. the active segment is a prefix SID for a neighbor F, or an
   adjacency segment S->F

2. the primary interface used to forward the packet failed

3. the segment following the active segment is an adjacency SID (F-
   >T)

4. node protection is active for that interface.

The TILFA Node FRR behavior becomes equivalent to:

1. Pop; the segment F or S->F is removed

2. Confirm that the next segment is an adjacency SID of F, say F->T

3. Identify T (as per the set of Adjacency Segments of F)

4. Pop the next segment and push T's segment based on the local SRGB

5. forward the packet according to T.

6. Security Considerations

   The techniques described in this document are internal
   functionality to a router that result in the ability to guarantee
   an upper bound on the time taken to restore traffic flow upon the
   failure of a directly connected link or node. As these techniques
   steer traffic to the post-convergence path as quickly as possible,
   this serves to minimize the disruption associated with a local
   failure which can be seen as a modest security enhancement.

7. IANA Considerations

   No requirements for IANA

8. Conclusions

   This document proposes a mechanism that is able to pre-calculate a
   backup path for every primary path so as to be able to protect
   against the failure of a directly connected link, node, or SRLG.
   The mechanism is able to calculate the backup path irrespective of
   the topology as long as the topology is sufficiently redundant.

9. References

9.1. Normative References

9.2. Informative References

   [1]   Filsfils, C., Previdi, S., Decraene, B., Litkowski, S., and R.
          Shakir, "Segment Routing Architecture", draft-ietf-spring-
          segment-routing-08 (work in progress), May 2016.

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

   [3]   Filsfils, C., Francois, P., Shand, M., Decraene, B., Uttaro,
          J., Leymann, N., and M. Horneffer, "Loop-Free Alternate (LFA)
          Applicability in Service Provider (SP) Networks", RFC 6571,
          June 2012.

   [4]   Bryant, S., Filsfils, C., Previdi, S., Shand, M., and N. So,
          "Remote Loop-Free Alternate (LFA) Fast Reroute (FRR)", RFC
          7490, DOI 10.17487/RFC7490, April 2015, <http://www.rfc-
          editor.org/info/rfc7490>.

   [5]   Bashandy, A., Filsfils, C., and Litkowski, S., " Loop
         avoidance using Segment Routing", draft-bashandy-rtgwg-
         segment-routing-uloop-00, (work in progress), May 2017

   [6]   Filsfils, C., Previdi, S., Decraene, B., Litkowski, S., and
         Shakir, R, "Segment Routing Architecture", draft-ietf-spring-
         segment-routing-11 (work in progress), February 2017

10. Acknowledgments

   We would like to give Les Ginsberg special thanks for the valuable
   comments and contribution

   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

   Bruno Decraene
   Orange
   Issy-les-Moulineaux
   FR
   Email: bruno.decraene@orange.com

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