Internet Engineering Task Force (IETF)
Request for Comments: 7490
Category: Standards Track
ISSN: 2070-1721

S. Bryant
C. Filsfils
S. Previdi
Cisco Systems
M. Shand
Independent Contributor
N. So
Vinci Systems

April 2015

Remote Loop-Free Alternate (LFA) Fast Reroute (FRR)

Abstract

This document describes an extension to the basic IP fast reroute mechanism, described in RFC 5286, that provides additional backup connectivity for point-to-point link failures when none can be provided by the basic mechanisms.

Status of This Memo

This is an Internet Standards Track document.

This document is a product of the Internet Engineering Task Force (IETF). It represents the consensus of the IETF community. It has received public review and has been approved for publication by the Internet Engineering Steering Group (IESG). Further information on Internet Standards is available in Section 2 of RFC 5741.

Information about the current status of this document, any errata, and how to provide feedback on it may be obtained at http://www.rfc-editor.org/info/rfc7490.

Table of Contents

1. Introduction

RFC 5714 [RFC5714] describes a framework for IP Fast Reroute (IPFRR) and provides a summary of various proposed IPFRR solutions. A basic mechanism using Loop-Free Alternates (LFAs) is described in [RFC5286] that provides good repair coverage in many topologies [RFC6571], especially those that are highly meshed. However, some topologies, notably ring-based topologies, are not well protected by LFAs alone. This is because there is no neighbor of the Point of Local Repair (PLR) that has a cost to the destination via a path that does not traverse the failure that is cheaper than the cost to the destination via the failure.

The method described in this document extends the LFA approach described in [RFC5286] to cover many of these cases by tunneling the packets that require IPFRR to a node that is both reachable from the PLR and can reach the destination.

2. Terminology

This document uses the terms defined in [RFC5714]. This section defines additional terms that are used in this document.

Repair tunnel:
: A tunnel established for the purpose of providing a virtual neighbor that is a Loop-Free Alternate.

P-space:
: The P-space of a router with respect to a protected link is the set of routers reachable from that specific router using the pre-convergence shortest paths without any of those paths (including equal-cost path splits) transiting that protected link.

  For example, the P-space of S with respect to link S-E is the set of routers that S can reach without using the protected link S-E.

Extended P-space:
: Consider the set of neighbors of a router protecting a link. Exclude from that set of routers the router reachable over the protected link. The extended P-space of the protecting router with respect to the protected link is the union of the P-spaces of the neighbors in that set of neighbors with respect to the protected link (see Section 5.2.1.2).

Q-space:
The Q-space of a router with respect to a protected link is the set of routers from which that specific router can be reached without any path (including equal-cost path splits) transiting that protected link.

PQ node:
A PQ node of a node S with respect to a protected link S-E is a node that is a member of both the P-space (or the extended P-space) of S with respect to that protected link S-E and the Q-space of E with respect to that protected link S-E. A repair tunnel endpoint is chosen from the set of PQ-nodes.

Remote LFA (RLFA):
The use of a PQ node rather than a neighbor of the repairing node as the next hop in an LFA repair [RFC5286].

In this document, the notation X-Y is used to mean the path from X to Y over the link directly connecting X and Y while the notation X->Y refers to the shortest path from X to Y via some set of unspecified nodes including the null set (i.e., including over a link directly connecting X and Y).

2.1. 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 RFC 2119 [RFC2119].

3. Overview of Solution

The problem of LFA IPFRR reachability in some networks is illustrated by the network fragment shown in Figure 1 below.

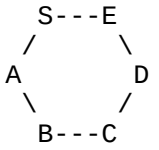


Figure 1: A Simple Ring Topology

If all link costs are equal, traffic that is transiting link S-E cannot be fully protected by LFAs. The destination C is an Equal-Cost Multipath (ECMP) from S, and so traffic to C can be protected when S-E fails but traffic to D and E are not protectable using LFAs.

This document describes extensions to the basic repair mechanism in which tunnels are used to provide additional logical links that can then be used as loop-free alternates where none exist in the original topology. In Figure 1, S can reach A, B, and C without going via S-E; these form S's extended P-space with respect to S-E. The routers that can reach E without going through S-E will be in E's Q-space with respect to link S-E; these are D and C. B has equal-cost paths to E via B-A-S-E and B-C-D-E, and so the forwarder at S might choose to send a packet to E via link S-E. Hence, B is not in the Q-space of E with respect to link S-E. The single node in both S's extended P-space and E's Q-space is C; thus, node C is selected as the repair tunnel's endpoint. Thus, if a tunnel is provided between S and C as shown in Figure 2, then C, now being a direct neighbor of S, would become an LFA for D and E. The definition of (extended) P-space and Q-space are provided in Section 2, and details of the calculation of the tunnel end points are provided in Section 5.2.

The non-failure traffic distribution is not disrupted by the provision of such a tunnel since it is only used for repair traffic and MUST NOT be used for normal traffic. Note that Operations, Administration, and Maintenance (OAM) traffic used specifically to verify the viability of the repair MAY traverse the tunnel prior to a failure.

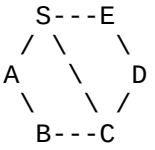


Figure 2: The Addition of a Tunnel

The use of this technique is not restricted to ring-based topologies but it is a general mechanism that can be used to enhance the protection provided by LFAs. A study of the protection achieved using remote LFA in typical service provider core networks is provided in Section 9, and a side-by-side comparison between LFA and remote LFA is provided in Section 9.4.

Remote LFA is suitable for incremental deployment within a network, including a network that is already deploying LFA. Computation of the repair path requires acceptable CPU resources and takes place exclusively on the repairing node. In MPLS networks, the targeted LDP protocol needed to learn the label binding at the repair tunnel endpoint (Section 8) is a well understood and widely deployed technology.

The technique described in this document is directed at providing repairs in the case of link failures. Considerations regarding node failures are discussed in Section 7. This memo describes a solution to the case where the failure occurs on a point-to-point link. It covers the case where the repair first hop is reached via a broadcast or non-broadcast multi-access (NBMA) link such as a LAN and the case where the P or Q node is attached via such a link. It does not, however, cover the more complicated case where the failed interface is a broadcast or NBMA link.

This document considers the case when the repair path is confined to either a single area or to the level two routing domain. In all other cases, the chosen PQ node should be regarded as a tunnel adjacency of the repairing node, and the considerations described in Section 6 of [RFC5286] should be taken into account.

4. Repair Paths

As with LFA FRR, when a router detects an adjacent link failure, it uses one or more repair paths in place of the failed link. Repair paths are precomputed in anticipation of later failures so they can be promptly activated when a failure is detected.

A tunneled repair path tunnels traffic to some staging point in the network from which it is known that, in the absence of a worse-than-anticipated failure, the traffic will travel to its destination using normal forwarding without looping back. This is equivalent to providing a virtual loop-free alternate to supplement the physical loop-free alternates; hence the name "remote LFA FRR". In its simplest form, when a link cannot be entirely protected with local LFA neighbors, the protecting router seeks the help of a remote LFA staging point. Network manageability considerations may lead to a repair strategy that uses a remote LFA more frequently [LFA-MANAGE].

Examples of worse failures are node failures (see Section 7), the failure of a Shared Risk Link Group (SRLG), the independent concurrent failures of multiple links, or broadcast or NBMA links (Section 3); protecting against such failures is out of scope for this specification.

4.1. Tunnels as Repair Paths

Consider an arbitrary protected link S-E. In LFA FRR, if a path to the destination from a neighbor N of S does not cause a packet to loop back over the link S-E (i.e., N is a loop-free alternate), then S can send the packet to N and the packet will be delivered to the destination using the pre-failure forwarding information. If there is no such LFA neighbor, then S may be able to create a virtual LFA

by using a tunnel to carry the packet to a point in the network that is not a direct neighbor of S from which the packet will be delivered to the destination without looping back to S. In this document, such a tunnel is termed a repair tunnel. The tail end of this tunnel (the repair tunnel endpoint) is a "PQ node", and the repair mechanism is a "remote LFA". This tunnel MUST NOT traverse the link S-E.

Note that the repair tunnel terminates at some intermediate router between S and E, and not E itself. This is clearly the case, since if it were possible to construct a tunnel from S to E, then a conventional LFA would have been sufficient to effect the repair.

4.2. Tunnel Requirements

There are a number of IP-in-IP tunnel mechanisms that may be used to fulfill the requirements of this design, such as IP-in-IP [RFC1853] and Generic Routing Encapsulation (GRE) [RFC1701].

In an MPLS-enabled network using LDP [RFC5036], a simple label stack [RFC3032] may be used to provide the required repair tunnel. In this case, the outer label is S's neighbor's label for the repair tunnel endpoint, and the inner label is the repair tunnel endpoint's label for the packet destination. In order for S to obtain the correct inner label, it is necessary to establish a targeted LDP session [RFC5036] to the tunnel endpoint.

The selection of the specific tunneling mechanism (and any necessary enhancements) used to provide a repair path is outside the scope of this document. The deployment in an MPLS/LDP environment is relatively simple in the data plane, as an LDP Label Switched Path (LSP) from S to the repair tunnel endpoint (the selected PQ node) is readily available and hence does not require any new protocol extension or design change. This LSP is automatically established as a basic property of LDP behavior. The performance of the encapsulation and decapsulation is efficient, as encapsulation is just a push of one label (like conventional MPLS-TE FRR) and the decapsulation is normally configured to occur at the penultimate hop before the repair tunnel endpoint. In the control plane, a Targeted LDP (TLDP) session is needed between the repairing node and the repair tunnel endpoint, which will need to be established and the labels processed before the tunnel can be used. The time to establish the TLDP session and acquire labels will limit the speed at which a new tunnel can be put into service. This is not anticipated to be a problem in normal operation since the managed introduction and removal of links is relatively rare, as is the incidence of failure in a well-managed network.

When a failure is detected, it is necessary to immediately redirect traffic to the repair path. Consequently, the repair tunnel used MUST be provisioned beforehand in anticipation of the failure. Since the location of the repair tunnels is dynamically determined, it is necessary to automatically establish the repair tunnels. Multiple repair tunnels may share a tunnel endpoint.

5. Construction of Repair Paths

5.1. Identifying Required Tunneled Repair Paths

Not all links will require protection using a tunneled repair path. Referring to Figure 1, if E can already be protected via an LFA, S-E does not need to be protected using a repair tunnel since all destinations normally reachable through E must therefore also be protectable by an LFA; such an LFA is frequently termed a "link LFA". Tunneled repair paths (which may be calculated per prefix) are only required for links that do not have a link or per-prefix LFA.

It should be noted that using the Q-space of E as a proxy for the Q-space of each destination can result in failing to identify valid remote LFAs. The extent to which this reduces the effective protection coverage is topology dependent.

5.2. Determining Tunnel Endpoints

The repair tunnel endpoint needs to be a node in the network reachable from S without traversing S-E. In addition, the repair tunnel endpoint needs to be a node from which packets will normally flow towards their destination without being attracted back to the failed link S-E.

Note that once released from the tunnel, the packet will be forwarded, as normal, on the shortest path from the release point to its destination. This may result in the packet traversing the router E at the far end of the protected link S-E, but this is obviously not required.

The properties that are required of repair tunnel endpoints are as follows:

- The repair tunneled point MUST be reachable from the tunnel source without traversing the failed link; and
- when released from the tunnel, packets MUST proceed towards their destination without being attracted back over the failed link.

Provided both these requirements are met, packets forwarded over the repair tunnel will reach their destination and will not loop after a single link failure.

In some topologies it will not be possible to find a repair tunnel endpoint that exhibits both the required properties. For example, if the ring topology illustrated in Figure 1 had a cost of four for the link B-C while the remaining links were the cost of one, then it would not be possible to establish a tunnel from S to C (without resorting to some form of source routing).

5.2.1. Computing Repair Paths

To compute the repair path for link S-E, it is necessary to determine the set of routers that can be reached from S without traversing S-E and match this with the set of routers from which the node E can be reached by normal forwarding without traversing the link S-E.

The approach used in this memo is as follows:

- The method of computing the set of routers that can be reached from S on the shortest path tree without traversing S-E is described. This is called the S's P-space with respect to the failure of link S-E.

- The distance of the tunnel endpoint from the PLR is increased by noting that S is able to use the P-space of its neighbors with respect to the failure of link S-E since S can determine which neighbor it will use as the next hop for the repair. This is called the S's extended P-space with respect to the failure of link S-E. The use of extended P-space allows greater repair coverage and is the preferred approach.

- Finally, two methods of computing the set of routers from which the node E can be reached by normal forwarding without traversing the link S-E. This is called the Q-space of E with respect to the link S-E.

The selection of the preferred node from the set of nodes that are in both extended P-space and Q-space with respect to the S-E is described in Section 5.2.2.

A suitable cost-based algorithm to compute the set of nodes common to both extended P-space and Q-space with respect to the S-E is provided in Section 5.3.

5.2.1.1. P-space

The set of routers that can be reached from S on the shortest path tree without traversing S-E is termed the P-space of S with respect to the link S-E. This P-space can be obtained by computing a Shortest Path Tree (SPT) rooted at S and excising the subtree reached via the link S-E (including those routers that are members of an ECMP that includes link S-E). The exclusion of routers reachable via an ECMP that includes S-E prevents the forwarding subsystem from attempting to execute a repair via the failed link S-E. Thus, for example, if the Shortest Path First (SPF) computation stores at each node the next hops to be used to reach that node from S, then the node can be added to P-space if none of its next hops are link S-E. In the case of Figure 1, this P-space comprises nodes A and B only. Expressed in cost terms, the set of routers {P} are those for which the shortest path cost S->P is strictly less than the shortest path cost S->E->P.

5.2.1.2. Extended P-space

The description in Section 5.2.1.1 calculated router S's P-space rooted at S itself. However, since router S will only use a repair path when it has detected the failure of the link S-E, the initial hop of the repair path need not be subject to S's normal forwarding decision process. Thus, the concept of extended P-space is introduced. Router S's extended P-space is the union of the P-spaces of each of S's neighbors (N). This may be calculated by computing an SPT at each of S's neighbors (excluding E) and excising the subtree reached via the path N->S->E. Note this will excise those routers that are reachable through all ECMPs that include link S-E. The use of extended P-space may allow router S to reach potential repair tunnel endpoints that were otherwise unreachable. In cost terms, a router (P) is in extended P-space if the shortest path cost N->P is strictly less than the shortest path cost N->S->E->P. In other words, once the packet is forced to N by S, it is a lower cost for it to continue on to P by any path except one that takes it back to S and then across the S->E link.

Since in the case of Figure 1 node A is a per-prefix LFA for the destination node C, the set of extended P-space nodes with respect to link S-E comprises nodes A, B, and C. Since node C is also in E's Q-space with respect to link S-E, there is now a node common to both extended P-space and Q-space that can be used as a repair tunnel endpoint to protect the link S-E.

5.2.1.3. Q-space

The set of routers from which the node E can be reached, by normal forwarding without traversing the link S-E, is termed the Q-space of E with respect to the link S-E. The Q-space can be obtained by computing a reverse Shortest Path Tree (rSPT) rooted at E, with the subtree that might traverse the protected link S-E excised (i.e., those nodes that would send the packet via S-E plus those nodes that have an ECMP set to E with one or more members of that ECMP set traversing the protected link S-E). The rSPT uses the cost towards the root rather than from it and yields the best paths towards the root from other nodes in the network. In the case of Figure 1, the Q-space of E with respect to S-E comprises nodes C and D only. Expressed in cost terms, the set of routers {Q} are those for which the shortest path cost Q<-E is strictly less than the shortest path cost Q<-S<-E. In Figure 1, the intersection of the E's Q-space with respect to S-E with S's P-space with respect to S-E defines the set of viable repair tunnel endpoints, known as "PQ nodes". As can be seen in the case of Figure 1, there is no common node and hence no viable repair tunnel endpoint. However, when the extended P-space (Section 5.2.1.2) at S with respect to S-E is considered, a suitable intersection is found at C.

Note that the Q-space calculation could be conducted for each individual destination and a per-destination repair tunnel end point determined. However, this would, in the worst case, require an SPF computation per destination that is not currently considered to be scalable. Therefore, the Q-space of E with respect to link S-E is used as a proxy for the Q-space of each destination. This approximation is obviously correct since the repair is only used for the set of destinations which were, prior to the failure, routed through node E. This is analogous to the use of link LFAs rather than per-prefix LFAs.

5.2.2. Selecting Repair Paths

The mechanisms described above will identify all the possible repair tunnel endpoints that can be used to protect a particular link. In a well-connected network, there are likely to be multiple possible release points for each protected link. All will deliver the packets correctly, so arguably, it does not matter which is chosen. However, one repair tunnel endpoint may be preferred over the others on the basis of path cost or some other selection criteria.

There is no technical requirement for the selection criteria to be consistent across all routers, but such consistency may be desirable from an operational point of view. In general, there are advantages in choosing the repair tunnel endpoint closest (shortest metric) to

S. Choosing the closest maximizes the opportunity for the traffic to be load balanced once it has been released from the tunnel. For consistency in behavior, it is RECOMMENDED that the member of the set of routers {PQ} with the lowest cost S->P be the default choice for P. In the event of a tie, the router with the lowest node identifier SHOULD be selected.

It is a local matter whether the repair path selection policy used by the router favors LFA repairs over RLFA repairs. An LFA repair has the advantage of not requiring the use of a tunnel; however, network manageability considerations may lead to a repair strategy that uses a remote LFA more frequently [LFA-MANAGE].

As described in [RFC5286], always selecting a PQ node that is downstream to the destination with respect to the repairing node prevents the formation of loops when the failure is worse than expected. The use of downstream nodes reduces the repair coverage, and operators are advised to determine whether adequate coverage is achieved before enabling this selection feature.

5.3. A Cost-Based RLFA Algorithm

The preceding text has described the computation of the remote LFA repair target (PQ) in terms of the intersection of two reachability graphs computed using an SPF algorithm. This section describes a method of computing the remote LFA repair target for a specific failed link using a cost-based algorithm. The pseudocode provided in this section avoids unnecessary SPF computations; for the sake of readability, it does not otherwise try to optimize the code. The algorithm covers the case where the repair first hop is reached via a broadcast or NBMA link such as a LAN. It also covers the case where the P or Q node is attached via such a link. It does not cover the case where the failed interface is a broadcast or NBMA link. To address that case it is necessary to compute the Q-space of each neighbor of the repairing router reachable through the LAN, i.e., to treat the pseudonode [RFC1195] as a node failure; this is because the Q-spaces of the neighbors of the pseudonode may be disjoint and require use of a neighbor-specific PQ node. The reader is referred to [NODE-PROTECTION] for further information on the use of RLFA for node repairs.

The following notation is used:

- D_opt(a,b) is the shortest distance from node a to node b as computed by the SPF.
- dest is the packet destination.

- fail_intf is the failed interface (S-E in the example).
- fail_intf.remote_node is the node reachable over interface fail_intf (node E in the example).
- intf.remote_node is the set of nodes reachable over interface intf.
- root is the root of the SPF calculation.
- self is the node carrying out the computation.
- y is the node in the network under consideration.
- y.pseudonode is true if y is a pseudonode.

```
////////////////////////////////////////////////////////////////
//
//   Main Function



////////////////////////////////////////////////////////////////
//
// We have already computed the forward SPF from self to all nodes
// y in network and thus we know D_opt (self, y).  This is needed
// for normal forwarding.
// However, for completeness:

Compute_and_Store_Forward_SPF(self)

// To extend P-space, we compute the SPF at each neighbor except
// the neighbor that is reached via the link being protected.
// We will also need D_opt(fail_intf.remote_node,y), so we
// compute that at the same time.

Compute_Neighbor_SPFs()

// Compute the set of nodes {P} reachable other than via the
// failed link.

Compute_Extended_P_Space(fail_intf)

// Compute the set of nodes that can reach the node on the far
// side of the failed link without traversing the failed link.

Compute_Q_Space(fail_intf)
```

```
    // Compute the set of candidate RLFA tunnel endpoints.

    Intersect_Extended_P_and_Q_Space()

    // Make sure that we cannot get looping repairs when the
    // failure is worse than expected.

    if (guarantee_no_looping_on_worse_than_protected_failure)
        Apply_Downstream_Constraint()

    //
    //  End of Main Function
    //
    ////////////////////////////////////////////////////////////////////

    ////////////////////////////////////////////////////////////////////
    //
    //  Procedures
    //


    ////////////////////////////////////////////////////////////////////
    //
    // This computes the SPF from root and stores the optimum
    // distance from root to each node y.

    Compute_and_Store_Forward_SPF(root)
        Compute_Forward_SPF(root)
        foreach node y in network
            store D_opt(root,y)



    ////////////////////////////////////////////////////////////////////
    //
    // This computes the optimum distance from each neighbor (other
    // than the neighbor reachable through the failed link) and
    // every other node in the network.
    //
    // Note that we compute this for all neighbors, including the
    // neighbor on the far side the failure.  This is done on the
    // expectation that more than one link will be protected and
    // that the results are stored for later use.
    //

    Compute_Neighbor_SPFs()
        foreach interface intf in self
            Compute_and_Store_Forward_SPF(intf.remote_node)
```

```
////////////////////////////////////////////////////////////////
//
// The reverse SPF computes the cost from each remote node to
// root. This is achieved by running the normal SPF algorithm
// but using the link cost in the direction from the next hop
// back towards root in place of the link cost in the direction
// away from root towards the next hop.

Compute_and_Store_Reverse_SPF(root)
    Compute_Reverse_SPF(root)
    foreach node y in network
        store D_opt(y,root)



////////////////////////////////////////////////////////////////
//
// Calculate Extended P-space
//
// Note that the "strictly less than" operator is needed to
// avoid ECMP issues.

Compute_Extended_P_Space(fail_intf)
    foreach node y in network
        y.in_extended_P_space = false
        // Extend P-space to the P-spaces of all reachable
        // neighbors
        foreach interface intf in self
            // Exclude failed interface, noting that
            // the node reachable via that interface may be
            // reachable via another interface (parallel path)
            if (intf != fail_intf)
                foreach neighbor n in intf.remote_node
                    // Apply RFC 5286 Inequality 1
                    if ( D_opt(n, y) <
                            D_opt(n,self) + D_opt(self, y))
                        y.in_extended_P_space = true

////////////////////////////////////////////////////////////////
//
// Compute the Nodes in Q-space
//

Compute_Q_Space(fail_intf)
    // Compute the cost from every node in the network to the
    // node normally reachable across the failed link
    Compute_and_Store_Reverse_SPF(fail_intf.remote_node)
```

```
        // Compute the cost from every node in the network to self
        Compute_and_Store_Reverse_SPF(self)

        foreach node y in network
            if ( D_opt(y,fail_intf.remote_node) < D_opt(y,self) +
                    D_opt(self,fail_intf.remote_node) )
                y.in_Q_space = true
            else
                y.in_Q_space = false



    ////////////////////////////////////////////////////////////////
    //
    // Compute Set of Nodes in Both Extended P-space and in Q-space

    Intersect_Extended_P_and_Q_Space()
        foreach node y in network
            if ( y.in_extended_P_space && y.in_Q_space &&
                    y.pseudonode == False)
                y.valid_tunnel_endpoint = true
            else
                y.valid_tunnel_endpoint = false


    ////////////////////////////////////////////////////////////////
    //
    // A downstream route is one where the next hop is strictly
    // closer to the destination.  By sending the packet to a
    // PQ node that is downstream, we know that if the PQ node
    // detects a failure it will not loop the packet back to self.
    // This is useful when there are two failures or when a node has
    // failed rather than a link.

    Apply_Downstream_Constraint()
        foreach node y in network
            if (y.valid_tunnel_endpoint)
                Compute_and_Store_Forward_SPF(y)
                if ((D_opt(y,dest) < D_opt(self,dest))
                    y.valid_tunnel_endpoint = true
                else
                    y.valid_tunnel_endpoint = false


  //
  ////////////////////////////////////////////////////////////////
```

5.4. Interactions with IS-IS Overload, RFC 6987, and Costed Out Links

Since normal link state routing takes into account the IS-IS overload bit, OSPF stub router advertisement [RFC6987], and costed out links (as described in Section 3.5 of [RFC5286]), the forward SPFs performed by the PLR rooted at the neighbors of the PLR also need to take this into account. A repair tunnel path from a neighbor of the PLR to a repair tunnel endpoint will generally avoid the nodes and links excluded by the IGP overload/costing-out rules. However, there are two situations where this behavior may result in a repair path traversing a link or router that should be excluded:

1. One situation is when the first hop on the repair tunnel path (from the PLR to a direct neighbor) does not follow the IGP shortest path. In this case, the PLR MUST NOT use a repair tunnel path whose first hop is along a link that has a cost or reverse cost equal to MaxLinkMetric (for OSPF) or the maximum cost (for IS-IS) or whose first hop has the overload bit set (for IS-IS).

2. The other situation is when the IS-IS overload bit and the mechanism of [RFC6987] only prevent transit traffic from traversing a node; they do not prevent traffic destined to a node. The per-neighbor forward SPFs using the standard IGP overload rules will not prevent a PLR from choosing a repair tunnel endpoint that is advertising a desire to not carry transit traffic. Therefore, the PLR MUST NOT use a repair tunnel endpoint with the IS-IS overload bit set or where all outgoing interfaces have the cost set to MaxLinkMetric for OSPF.

6. Example Application of Remote LFAs

An example of a commonly deployed topology that is not fully protected by LFAs alone is shown in Figure 3. Provider Edge (PE)1 and PE2 are connected in the same site. P1 and P2 may be geographically separated (intersite). In order to guarantee the lowest latency path from/to all other remote PEs, normally the shortest path follows the geographical distance of the site locations. Therefore, to ensure this, a lower IGP metric (5) is assigned between PE1 and PE2. A high metric (1000) is set on the P-PE links to prevent the PEs being used for transit traffic. The PEs are not individually dual-homed in order to reduce costs.

This is a common topology in Service Provider (SP) networks.

When a failure occurs on the link between PE1 and P1, PE1 does not have an LFA for traffic reachable via P1. Similarly, by symmetry, if the link between PE2 and P2 fails, PE2 does not have an LFA for traffic reachable via P2.

Increasing the metric between PE1 and PE2 to allow the LFA would impact the normal traffic performance by potentially increasing the latency.

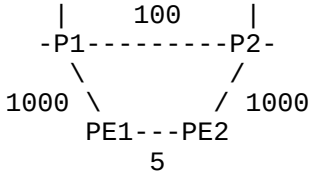


Figure 3: Example SP Topology

Clearly, full protection can be provided using the techniques described in this document by PE1 choosing P2 as the remote LFA repair target node and PE2 choosing P1 as the remote LFA repair target.

7. Node Failures

When the failure is a node failure rather than a point-to-point link failure, there is a danger that the RLFA repair will loop. This is discussed in detail in [IP-FRR]. In summary, the problem is that two or more of E's neighbors, each with E as the next hop to some destination D, may attempt to repair a packet addressed to destination D via the other neighbor and then E, thus causing a loop to form. A similar problem exists in the case of a shared risk link group failure where the PLR for each failure attempts to repair via the other failure. As will be noted from [IP-FRR], this can rapidly become a complex problem to address.

There are a number of ways to minimize the probability of a loop forming when a node failure occurs, and there exists the possibility that two of E's neighbors may form a mutual repair.

1. Detect when a packet has arrived on some interface I that is also the interface used to reach the first hop on the RLFA path to the remote LFA repair target, and drop the packet. This is useful in the case of a ring topology.

2. Require that the path from the remote LFA repair target to destination D never passes through E (including in the ECMP case), i.e., only use node protecting paths in which the cost from the remote LFA repair target to D is strictly less than the cost from the remote LFA repair target to E plus the cost E to D.

3. Require that where the packet may pass through another neighbor of E, that node is down stream (i.e., strictly closer to D than the repairing node). This means that some neighbor of E (X) can repair via some other neighbor of E (Y), but Y cannot repair via X.

Case 1 accepts that loops may form and suppresses them by dropping packets. Dropping packets may be considered less detrimental than looping packets. This approach may also lead to dropping some legitimate packets. Cases 2 and 3 above prevent the formation of a loop but at the expense of a reduced repair coverage and at the cost of additional complexity in the algorithm to compute the repair path. Alternatively, one might choose to assume that the probability of a node failure is sufficiently rare that the issue of looping RLFA repairs can be ignored.

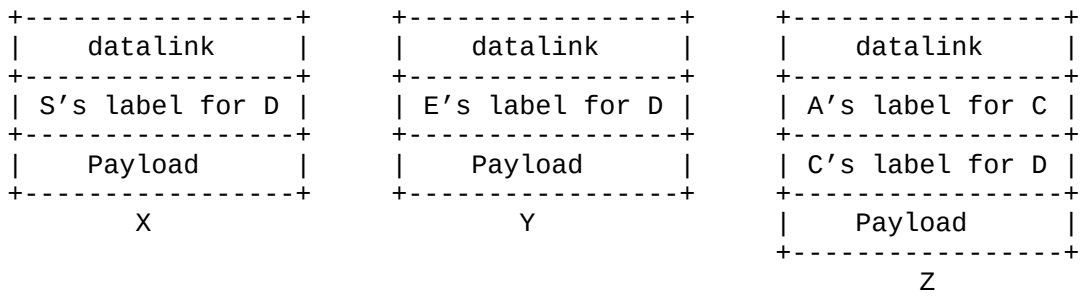
The probability of a node failure and the consequences of node failure in any particular topology will depend on the node design, the particular topology in use, and the strategy adopted under node failure. It is recommended that a network operator perform an analysis of the consequences and probability of node failure in their network and determine whether the incidence and consequence of occurrence are acceptable.

This topic is further discussed in [NODE-PROTECTION].

8. Operation in an LDP Environment

Where this technique is used in an MPLS network using LDP [RFC5036], and S is a transit node, S will need to swap the top label in the stack for the remote LFA repair target's (PQ's) label to the destination and to then push its own label for the remote LFA repair target.

In the example, S in Figure 2 already has the first hop (A) label for the remote LFA repair target (C) as a result of the ordinary operation of LDP. To get the remote LFA repair target's label (C's label) for the destination (D), S needs to establish a targeted LDP session with C. The label stack for normal operation and RLFA operation is shown below in Figure 4.



X = Normal label stack packet arriving at S
Y = Normal label stack packet leaving S
Z = RLFA label stack to D via C as the remote LFA repair target

Figure 4

To establish a targeted LDP session with a candidate remote LFA repair target node, the repairing node (S) needs to know what IP address the remote LFA repair target is willing to use for targeted LDP sessions. Ideally, this is provided by the remote LFA repair target advertising this address in the IGP in use. Which address is used, how this is advertised in the IGP, and whether this is a special IP address or an IP address also used for some other purpose is out of scope for this document and must be specified in an IGP-specific RFC.

In the absence of a protocol to learn the preferred IP address for targeted LDP, an LSR should attempt a targeted LDP session with the Router ID [RFC2328] [RFC5305] [RFC5340] [RFC6119] [OSPF-RI] unless it is configured otherwise.

No protection is available until the TLDP session has been established and a label for the destination has been learned from the remote LFA repair target. If for any reason the TLDP session cannot be established, an implementation SHOULD advise the operator about the protection setup issue through the network management system.

9. Analysis of Real World Topologies

This section gives the results of analyzing a number of real world service provider topologies collected between the end of 2012 and early 2013.

9.1. Topology Details

The figure below characterizes each topology (topo) studied in terms of:

- the number of nodes (# nodes) excluding pseudonodes;
- the number of bidirectional links (# links) including parallel links and links to and from pseudonodes;
- the number of node pairs that are connected by one or more links (# pairs);
- the number of node pairs that are connected by more than one (i.e., parallel) link (# para); and
- the number of links (excluding pseudonode links, which are by definition asymmetric) that have asymmetric metrics (# asym).

topo	# nodes	# links	# pairs	# para	# asym
1	315	570	560	10	3
2	158	373	312	33	0
3	655	1768	1314	275	1195
4	1281	2326	2248	70	10
5	364	811	659	80	86
6	114	318	197	101	4
7	55	237	159	67	2
8	779	1848	1441	199	437
9	263	482	413	41	12
10	86	375	145	64	22
11	162	1083	351	201	49
12	380	1174	763	231	0
13	1051	2087	2037	48	64
14	92	291	204	64	2

9.2. LFA Only

The figure below shows the percentage of protected destinations (% prot) and the percentage of guaranteed node protected destinations (% gtd N) for the set of topologies characterized in Section 9.1 achieved using only LFA repairs.

These statistics were generated by considering each node and then considering each link to each next hop to each destination. The percentage of such links across the entire network that are protected against link failure was determined. This is the percentage of protected destinations. If a link is protected against the failure of the next hop node, this is considered Guaranteed Node Protecting (GNP) and the percentage of guaranteed node protected destinations is calculated using the same method used for calculating the link protection coverage.

GNP is identical to node-protecting as defined in [RFC6571] and does not include the additional node protection coverage obtained by the de facto node-protecting condition described in [RFC6571].

topo	% prot	% gtd N
1	78.5	36.9
2	97.3	52.4
3	99.3	58
4	83.1	63.1
5	99	59.1
6	86.4	21.4
7	93.9	35.4
8	95.3	48.1
9	82.2	49.5
10	98.5	14.9
11	99.6	24.8
12	99.5	62.4
13	92.4	51.6
14	99.3	48.6

9.3. RLFA

The figure below shows the percentage of protected destinations (% prot) and % guaranteed node protected destinations (% gtd N) for RLFA protection in the topologies studies. In addition, it shows the percentage of destinations using an RLFA repair (% PQ) together with the total number of unidirectional RLFA targeted LDP sessions established (# PQ), and the number of PQ sessions that would be

required for complete protection but that could not be established because there was no PQ node, i.e., the number of cases whether neither LFA or RLFA protection was possible (no PQ). It also shows the 50 (p50), 90 (p90), and 100 (p100) percentiles for the number of individual LDP sessions terminating at an individual node (whether used for TX, RX, or both).

For example, if there were LDP sessions that required A->B, A->C, C->A, and C->D, these would be counted as 2, 1, 2, and 1 at nodes A, B, C, and D respectively because:

A has two sessions (to nodes B and C);

B has one session (to node A);

C has two sessions (to nodes A and D); and

D has one session (to node D).

In this study, remote LFA is only used when necessary, i.e., when there is at least one destination that is not reparable by a per destination LFA and a single remote LFA tunnel is used (if available) to repair traffic to all such destinations. The remote LFA repair target points are computed using extended P-space and choosing the PQ node that has the lowest metric cost from the repairing node.

topo	% prot	% gtd N	% PQ	# PQ	no PQ	p50	p90	p100
1	99.7	53.3	21.2	295	3	1	5	14
2	97.5	52.4	0.2	7	40	0	0	2
3	99.999	58.4	0.7	63	5	0	1	5
4	99	74.8	16	1424	54	1	3	23
5	99.5	59.5	0.5	151	7	0	2	7
6	100	34.9	13.6	63	0	1	2	6
7	99.999	40.6	6.1	16	2	0	2	4
8	99.5	50.2	4.3	350	39	0	2	15
9	99.5	55	17.3	428	5	1	2	67
10	99.6	14.1	1	49	7	1	2	5
11	99.9	24.9	0.3	85	1	0	2	8
12	99.999	62.8	0.5	512	4	0	0	3
13	97.5	54.6	5.1	1188	95	0	2	27
14	100	48.6	0.7	79	0	0	2	4

Another study [ISOCORE2010] confirms the significant coverage increase provided by remote LFAs.

9.4. Comparison of LFA and RLFA results

The table below provides a side-by-side comparison of the LFA and the remote LFA results. This shows a significant improvement in the percentage of protected destinations and normally a modest improvement in the percentage of guaranteed node protected destinations.

topo	LFA % prot	RLFA %prot	LFA % gtd N	RLFA % gtd N
1	78.5	99.7	36.9	53.3
2	97.3	97.5	52.4	52.4
3	99.3	99.999	58	58.4
4	83.1	99	63.1	74.8
5	99	99.5	59.1	59.5
6	86.4	100	21.4	34.9
7	93.9	99.999	35.4	40.6
8	95.3	99.5	48.1	50.2
9	82.2	99.5	49.5	55
10	98.5	99.6	14.9	14.1
11	99.6	99.9	24.8	24.9
12	99.5	99.999	62.4	62.8
13	92.4	97.5	51.6	54.6
14	99.3	100	48.6	48.6

As shown in the table, remote LFA provides close to 100% prefix protection against link failure in 11 of the 14 topologies studied and provides a significant improvement in two of the remaining three cases. Note that in an MPLS network, the tunnels to the PQ nodes are always present as a property of an LDP-based deployment.

In the small number of cases where there is no intersection between the (extended) P-space and the Q-space, a number of solutions to providing a suitable path between such disjoint regions in the network have been discussed in the working group. For example, an explicitly routed LSP between P and Q might be set up using RSVP-TE or using Segment Routing [SEGMENT-ROUTING]. Such extended repair methods are outside the scope of this document.

10. Management and Operational Considerations

The management of LFA and remote LFA is the subject of ongoing work within the IETF [LFA-MANAGE], to which the reader is referred. Management considerations may lead to a preference for the use of a remote LFA over an available LFA. This preference is a matter for the network operator and not a matter of protocol correctness.

When the network reconverges, micro-loops [RFC5715] can form due to transient inconsistencies in the forwarding tables of different routers. If it is determined that micro-loops are a significant issue in the deployment, then a suitable loop-free convergence method, such as one of those described in [RFC5715], [RFC6976], or [ULOOP-DELAY], should be implemented.

11. Historical Note

The basic concepts behind remote LFA were invented in 2002 and were later included in [IP-FRR], submitted in 2004.

[IP-FRR] targeted a 100% protection coverage and hence included additional mechanisms on top of the remote LFA concept. The addition of these mechanisms made the proposal very complex and computationally intensive, and it was therefore not pursued as a working group item.

As explained in [RFC6571], the purpose of the LFA FRR technology is not to provide coverage at any cost. A solution for this already exists with MPLS-TE FRR. MPLS-TE FRR is a mature technology that is able to provide protection in any topology thanks to the explicit routing capability of MPLS-TE.

The purpose of LFA FRR technology is to provide for a simple FRR solution when such a solution is possible. The first step along this simplicity approach was "local" LFA [RFC5286]. This specification of "remote LFA" is a natural second step.

12. Security Considerations

The security considerations of [RFC5286] also apply.

Targeted LDP sessions and MPLS tunnels are normal features of an MPLS network, and their use in this application raises no additional security concerns.

IP repair tunnel endpoints (where used) SHOULD be assigned from a set of addresses that are not reachable from outside the routing domain; this would prevent their use as an attack vector.

Other than OAM traffic used to verify the correct operation of a repair tunnel, only traffic that is being protected as a result of a link failure is placed in a repair tunnel. The repair tunnel MUST NOT be advertised by the routing protocol as a link that may be used to carry normal user traffic or routing protocol traffic.

13. References

13.1. Normative References

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

[RFC5286] Atlas, A., Ed. and A. Zinin, Ed., "Basic Specification for IP Fast Reroute: Loop-Free Alternates", RFC 5286, September 2008, <http://www.rfc-editor.org/info/rfc5286>.

[RFC5714] Shand, M. and S. Bryant, "IP Fast Reroute Framework", RFC 5714, January 2010, <http://www.rfc-editor.org/info/rfc5714>.

13.2. Informative References

[IP-FRR] Bryant, S., Filsfils, C., Previdi, S., and M. Shand, "IP Fast Reroute using tunnels", Work in Progress, draft-bryant-ipfrr-tunnels-03, November 2007.

[ISOCORE2010] So, N., Lin, T., and C. Chen, "LFA (Loop Free Alternates) Case Studies in Verizon's LDP Network", 13th Annual MPLS Conference, 2010.

[LFA-MANAGE] Litkowski, S., Decraene, B., Filsfils, C., Raza, K., Horneffer, M., and P. Sarkar, "Operational management of Loop Free Alternates", Work in Progress, draft-ietf-rtgwg-lfa-manageability-08, March 2015.

[NODE-PROTECTION] Sarkar, P., Gredler, H., Hegde, S., Bowers, C., Litkowski, S., and H. Raghuveer, "Remote-LFA Node Protection and Manageability", Work in Progress, draft-ietf-rtgwg-rlfa-node-protection-01, December 2014.

[OSPF-RI] Xu, X., Chunduri, U., and M. Bhatia, "Carrying Routable IP Addresses in OSPF RI LSA", Work in Progress, draft-ietf-ospf-routable-ip-address-02, April 2015.

[RFC1195] Callon, R., "Use of OSI IS-IS for routing in TCP/IP and dual environments", RFC 1195, December 1990, <http://www.rfc-editor.org/info/rfc1195>.

[RFC1701] Hanks, S., Li, T., Farinacci, D., and P. Traina, "Generic Routing Encapsulation (GRE)", RFC 1701, October 1994, <http://www.rfc-editor.org/info/rfc1701>.

[RFC1853] Simpson, W., "IP in IP Tunneling", RFC 1853, October 1995, <http://www.rfc-editor.org/info/rfc1853>.

[RFC2328] Moy, J., "OSPF Version 2", STD 54, RFC 2328, April 1998, <http://www.rfc-editor.org/info/rfc2328>.

[RFC3032] Rosen, E., Tappan, D., Fedorkow, G., Rekhter, Y., Farinacci, D., Li, T., and A. Conta, "MPLS Label Stack Encoding", RFC 3032, January 2001, <http://www.rfc-editor.org/info/rfc3032>.

[RFC5036] Andersson, L., Ed., Minei, I., Ed., and B. Thomas, Ed., "LDP Specification", RFC 5036, October 2007, <http://www.rfc-editor.org/info/rfc5036>.

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

[RFC5340] Coltun, R., Ferguson, D., Moy, J., and A. Lindem, "OSPF for IPv6", RFC 5340, July 2008, <http://www.rfc-editor.org/info/rfc5340>.

[RFC5715] Shand, M. and S. Bryant, "A Framework for Loop-Free Convergence", RFC 5715, January 2010, <http://www.rfc-editor.org/info/rfc5715>.

[RFC6119] Harrison, J., Berger, J., and M. Bartlett, "IPv6 Traffic Engineering in IS-IS", RFC 6119, February 2011, <http://www.rfc-editor.org/info/rfc6119>.

[RFC6571] Filsfils, C., Ed., Francois, P., Ed., 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, <http://www.rfc-editor.org/info/rfc6571>.

[RFC6976] Shand, M., Bryant, S., Previdi, S., Filsfils, C., Francois, P., and O. Bonaventure, "Framework for Loop-Free Convergence Using the Ordered Forwarding Information Base (oFIB) Approach", RFC 6976, July 2013, <http://www.rfc-editor.org/info/rfc6976>.

[RFC6987] Retana, A., Nguyen, L., Zinin, A., White, R., and D. McPherson, "OSPF Stub Router Advertisement", RFC 6987, September 2013, <http://www.rfc-editor.org/info/rfc6987>.

[SEGMENT-ROUTING]
Filsfils, C., Previdi, S., Bashandy, A., Decraene, B., Litkowski, S., Horneffer, M., Shakir, R., Tantsura, J., and E. Crabbe, "Segment Routing Architecture", Work in Progress, draft-ietf-spring-segment-routing-01, February 2015.

[ULOOP-DELAY]
Litkowski, S., Decraene, B., Filsfils, C., and P. Francois, "Microloop prevention by introducing a local convergence delay", Work in Progress, draft-litkowski-rtgwg-uloop-delay-03, February 2014.

Acknowledgements

The authors wish to thank Levente Csikor and Chris Bowers for their contribution to the cost-based algorithm text. The authors thank Alia Atlas, Ross Callon, Stephane Litkowski, Bharath R, Pushpasis Sarkar, and Adrian Farrel for their review of this document.

Authors' Addresses

Stewart Bryant
Cisco Systems
9-11 New Square,
Bedfont Lakes,
Feltham,
Middlesex TW14 8HA
United Kingdom

EMail: stbryant@cisco.com

Clarence Filsfils
Cisco Systems
De Kleetlaan 6a
1831 Diegem
Belgium

EMail: cfilsfil@cisco.com

Stefano Previdi
Cisco Systems

EMail: sprevidi@cisco.com

Mike Shand
Independent Contributor

EMail: imc.shand@gmail.com

Ning So
Vinci Systems

EMail: ningso@vinci-systems.com