                 Remote-LFA Node Protection and Manageability

Abstract

   The loop-free alternates (LFAs) computed following the current
   remote-LFA specification guarantees only link protection. The
   resulting remote-LFA next hops (also called "PQ-nodes") may not
   guarantee node protection for all destinations being protected by it.

   This document describes an extension to the remote-loop-free-based IP
   fast reroute mechanisms that specifies procedures for determining
   whether or not a given PQ-node provides node protection for a
   specific destination. The document also shows how the same procedure
   can be utilized for the collection of complete characteristics for
   alternate paths. Knowledge about the characteristics of all
   alternate paths is a precursor to applying the operator-defined
   policy for eliminating paths not fitting the constraints.

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 7841.

   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/rfc8102.

Table of Contents

1. Introduction

   The Remote-LFA specification [RFC7490] provides loop-free alternates
   that guarantee only link protection. The resulting remote-LFA
   alternate next hops (also referred to as the "PQ-nodes") may not
   provide node protection for all destinations covered by the same
   remote-LFA alternate, in case of failure of the primary next-hop
   node, and it does not provide a means to determine the same.

   Also, the LFA Manageability document [RFC7916] requires a computing
   router to find all possible alternate next hops (including all
   possible remote-LFA), collect the complete set of path
   characteristics for each alternate path, run an alternate-selection
   policy (configured by the operator), and find the best alternate
   path. This will require that the remote-LFA implementation gathers
   all the required path characteristics along each link on the entire
   remote-LFA alternate path.

   With current LFA [RFC5286] and remote-LFA implementations, the
   forward SPF (and reverse SPF) is run with the computing router and
   its immediate one-hop routers as the roots. While that enables
   computation of path attributes (e.g., Shared Risk Link Group (SRLG)
   and Admin-groups) for the first alternate path segment from the
   computing router to the PQ-node, there is no means for the computing
   router to gather any path attributes for the path segment from the
   PQ-node to the destination. Consequently, any policy-based selection
   of alternate paths will consider only the path attributes from the
   computing router up until the PQ-node.

   This document describes a procedure for determining node protection
   with remote-LFA. The same procedure is also extended for the
   collection of a complete set of path attributes, enabling more
   accurate policy-based selection for alternate paths obtained with
   remote-LFA.

1.1. Abbreviations

   This document uses the following list of abbreviations:

      LFA: Loop-Free Alternates

      RLFA or R-LFA: Remote Loop-Free Alternates

      ECMP: Equal-Cost Multiple Path

      SPF: Shortest Path First graph computations

      NH: Next-Hop node

1.2. 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].

2. Node Protection with Remote-LFA

   Node protection is required to provide protection of traffic on a
   given forwarding node against the failure of the first-hop node on
   the primary forwarding path. Such protection becomes more critical
   in the absence of mechanisms like non-stop routing in the network.
   Certain operators refrain from deploying non-stop-routing in their
   network, due to the required complex state synchronization between
   redundant control plane hardwares it requires, and the significant
   additional computation and performance overheads it comes along with.
   In such cases, node protection is essential to guarantee
   uninterrupted flow of traffic, even in the case of an entire
   forwarding node going down.

   The following sections discuss the node-protection problem in the
   context of remote-LFA and propose a solution.

2.1. The Problem

   To better illustrate the problem and the solution proposed in this
   document, the following topology diagram from the remote-LFA document
   [RFC7490] is being re-used with slight modification.

```
                         D1
                        /
                     S-x-E
                    /     \
                   N       R3--D2
                    \     /
                    R1---R2
```

                       Figure 1: Topology 1
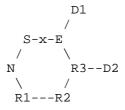
   In the above topology, for all (non-ECMP) destinations reachable via
   the S-E link, there is no standard LFA alternate. As per the remote-
   LFA [RFC7490] alternate specifications, node R2 being the only PQ-
   node for the S-E link provides the next hop for all of the above
   destinations. Table 1 shows all possible primary and remote-LFA
   alternate paths for each destination.

```
+-------------+-------------+---------+------------------------+
| Destination | Primary Path | PQ-node | Remote-LFA Backup Path |
+-------------+-------------+---------+------------------------+
| R3          | S->E->R3    | R2      | S=>N=>R1=>R2->R3        |
| E           | S->E        | R2      | S=>N=>R1=>R2->R3->E     |
| D1          | S->E->D1    | R2      | S=>N=>R1=>R2->R3->E->D1 |
| D2          | S->E->R3->D2 | R2     | S=>N=>R1=>R2->R3->D2    |
+-------------+-------------+---------+------------------------+
```

             Table 1: Remote-LFA Backup Paths via PQ-Node R2

A closer look at Table 1 shows that, while the PQ-node R2 provides
link protection for all the destinations, it does not provide node
protection for destinations E and D1. In the event of the node-
failure on primary next hop E, the alternate path from the remote-LFA
next hop R2 to E and D1 also becomes unavailable. So, for a remote-
LFA next hop to provide node protection for a given destination, the
shortest path from the given PQ-node to the given destination MUST
NOT traverse the primary next hop.

In another extension of the topology in Figure 1, let us consider an
additional link between N and E with the same cost as the other
links.

```
                         D1
                        /
                     S-x-E
                    /   / \
                   N---+   R3--D2
                    \      /
                     R1---R2
```

                         Figure 2: Topology 2

In the above topology, the S-E link is no longer on any of the
shortest paths from N to R3, E, and D1. Hence, R3, E, and D1 are
also included in both the extended P-space and the Q-space of E (with
respect to the S-E link). Table 2 shows all possible primary and
R-LFA alternate paths via PQ-node R3 for each destination reachable
through the S-E link in the above topology. The R-LFA alternate
paths via PQ-node R2 remain the same as in Table 1.

```
+-------------+--------------+---------+-----------------------+
| Destination | Primary Path | PQ-node | Remote-LFA Backup Path |
+-------------+--------------+---------+-----------------------+
| R3          | S->E->R3     | R3      | S=>N=>E=>R3           |
| E           | S->E         | R3      | S=>N=>E=>R3->E        |
| D1          | S->E->D1     | R3      | S=>N=>E=>R3->E->D1    |
| D2          | S->E->R3->D2 | R3      | S=>N=>E=>R3->D2       |
+-------------+--------------+---------+-----------------------+
```

                Table 2: Remote-LFA Backup Paths via PQ-Node R3

   Again, a closer look at Table 2 shows that, unlike Table 1 where the
   single PQ-node R2 provided node protection for destinations R3 and
   D2, if we choose R3 as the R-LFA next hop, it no longer provides node
   protection for R3 and D2. If S chooses R3 as the R-LFA next hop and
   if there is a node-failure on primary next hop E, then one of the
   parallel ECMP paths between N and R3 also becomes unavailable on the
   alternate path from S to R-LFA next hop R3. So, for a remote-LFA
   next hop to provide node protection for a given destination, the
   shortest paths from S to the chosen PQ-node MUST NOT traverse the
   primary next-hop node.

2.2. Additional Definitions

   This document adds and enhances the following definitions, extending
   the ones mentioned in the Remote-LFA specification [RFC7490].

2.2.1. Link-Protecting Extended P-Space

   The Remote-LFA specification [RFC7490] already defines this. The
   link-protecting extended P-space for a link S-E being protected is
   the set of routers that are reachable from one or more direct
   neighbors of S, except primary node E, without traversing the S-E
   link on any of the shortest paths from the direct neighbor to the
   router. This MUST exclude any direct neighbor for which there is at
   least one ECMP path from the direct neighbor traversing the link
   (S-E) being protected.

   For a cost-based definition for link-protecting extended P-space,
   refer to Section 2.2.6.1.

2.2.2. Node-Protecting Extended P-Space

   The node-protecting extended P-space for a primary next-hop node E
   being protected is the set of routers that are reachable from one or
   more direct neighbors of S, except primary node E, without traversing
   node E. This MUST exclude any direct neighbors for which there is at

least one ECMP path from the direct neighbor traversing the node E
being protected.

For a cost-based definition for node-protecting extended P-space,
refer to Section 2.2.6.2.

2.2.3. Q-Space

The Remote-LFA document [RFC7490] already defines this. The Q-space
for a link S-E being protected is the set of nodes that can reach
primary node E, without traversing the S-E link on any of the
shortest paths from the node itself to primary next hop E. This MUST
exclude any node for which there is at least one ECMP path from the
node to the primary next hop E traversing the link (S-E) being
protected.

For a cost-based definition for Q-Space, refer to Section 2.2.6.3.

2.2.4. Link-Protecting PQ-Space

A node Y is in a link-protecting PQ-space with respect to the link
(S-E) being protected if and only if Y is present in both link-
protecting extended P-space and the Q-space for the link being
protected.

2.2.5. Candidate Node-Protecting PQ-Space

A node Y is in a candidate node-protecting PQ-space with respect to
the node (E) being protected if and only if Y is present in both the
node-protecting extended P-space and the Q-space for the link being
protected.

Please note that a node Y being in a candidate node-protecting PQ-
space does not guarantee that the R-LFA alternate path via the same,
in entirety, is unaffected in the event of a node failure of primary
next-hop node E. It only guarantees that the path segment from S to
PQ-node Y is unaffected by the same failure event. The PQ-nodes in
the candidate node-protecting PQ-space may provide node protection
for only a subset of destinations that are reachable through the
corresponding primary link.

2.2.6. Cost-Based Definitions

This section provides cost-based definitions for some of the terms
introduced in Section 2.2 of this document.

2.2.6.1. Link-Protecting Extended P-Space

   Please refer to Section 2.2.1 for a formal definition of link-
   protecting extended P-space.

   A node Y is in a link-protecting extended P-space with respect to the
   link (S-E) being protected if and only if there exists at least one
   direct neighbor of S (Ni) other than primary next hop E that
   satisfies the following condition.

   D_opt(Ni,Y) < D_opt(Ni,S) + D_opt(S,Y)

   Where,
     D_opt(A,B) : Distance on the most optimum path from A to B.
           Ni   : A direct neighbor of S other than primary
                  next hop E.
            Y   : The node being evaluated for link-protecting
                  extended P-Space.

            Figure 3: Link-Protecting Ext-P-Space Condition

2.2.6.2. Node-Protecting Extended P-Space

   Please refer to Section 2.2.2 for a formal definition of node-
   protecting extended P-space.

   A node Y is in a node-protecting extended P-space with respect to the
   node E being protected if and only if there exists at least one
   direct neighbor of S (Ni) other than primary next hop E, that
   satisfies the following condition.

   D_opt(Ni,Y) < D_opt(Ni,E) + D_opt(E,Y)

   Where,
     D_opt(A,B) : Distance on the most optimum path from A to B.
            E   : The primary next hop on the shortest path from S
                  to destination.
           Ni   : A direct neighbor of S other than primary
                  next hop E.
            Y   : The node being evaluated for node-protecting
                  extended P-Space.

            Figure 4: Node-Protecting Ext-P-Space Condition

   Please note that a node Y satisfying the condition in Figure 4 above
   only guarantees that the R-LFA alternate path segment from S via
   direct neighbor Ni to the node Y is not affected in the event of a
   node failure of E. It does not yet guarantee that the path segment

from node Y to the destination is also unaffected by the same failure
event.

2.2.6.3. Q-Space

Please refer to Section 2.2.3 for a formal definition of Q-Space.

A node Y is in Q-space with respect to the link (S-E) being protected
if and only if the following condition is satisfied:

D_opt(Y,E) < D_opt(S,E) + D_opt(Y,S)

Where,
   D_opt(A,B) : Distance on the most optimum path from A to B.
           E : The primary next hop on the shortest path from S
                to destination.
           Y : The node being evaluated for Q-Space.

                      Figure 5: Q-Space Condition

2.3. Computing Node-Protecting R-LFA Path

The R-LFA alternate path through a given PQ-node to a given
destination is comprised of two path segments as follows:

1. Path segment from the computing router to the PQ-node (Remote-LFA
    alternate next hop), and

2. Path segment from the PQ-node to the destination being protected.

So, to ensure that an R-LFA alternate path for a given destination
provides node protection, we need to ensure that none of the above
path segments are affected in the event of failure of the primary
next-hop node. Sections 2.3.1 and 2.3.2 show how this can be
ensured.

2.3.1. Computing Candidate Node-Protecting PQ-Nodes for Primary Next
        Hops

To choose a node-protecting R-LFA next hop for a destination R3,
router S needs to consider a PQ-node from the candidate node-
protecting PQ-space for the primary next hop E on the shortest path
from S to R3. As mentioned in Section 2.2.2, to consider a PQ-node
as a candidate node-protecting PQ-node, there must be at least one
direct neighbor Ni of S, such that all shortest paths from Ni to the
PQ-node do not traverse primary next-hop node E.

Implementations SHOULD run the inequality in Section 2.2.6.2,
Figure 4 for all direct neighbors, other than primary next-hop node
E, to determine whether a node Y is a candidate node-protecting PQ-
node. All of the metrics needed by this inequality would have been
already collected from the forward SPFs rooted at each of direct
neighbor S, computed as part of standard LFA [RFC5286]
implementation. With reference to the topology in Figure 2, Table 3
shows how the above condition can be used to determine the candidate
node-protecting PQ-space for S-E link (primary next hop E).

Candidate PQ-node (Y)	Direct Nbr (Ni)	D_opt (Ni,Y)	D_opt (Ni,E)	D_opt (E,Y)	Condition Met
R2	N	2 (N,R2)	1 (N,E)	2 (E,R2)	Yes
R3	N	2 (N,R3)	1 (N,E)	1 (E,R3)	No

Table 3: Node-Protection Evaluation for R-LFA Repair Tunnel to PQ-
Node

As seen in the above Table 3, R3 does not meet the node-protecting
extended p-space inequality; so, while R2 is in candidate node-
protecting PQ-space, R3 is not.

Some SPF implementations may also produce a list of links and nodes
traversed on the shortest path(s) from a given root to others. In
such implementations, router S may have executed a forward SPF with
each of its direct neighbors as the SPF root, executed as part of the
standard LFA computations [RFC5286]. So, S may re-use the list of
links and nodes collected from the same SPF computations to decide
whether or not a node Y is a candidate node-protecting PQ-node. A
node Y shall be considered as a node-protecting PQ-node if and only
if there is at least one direct neighbor of S, other than the primary
next hop E for which the primary next-hop node E does not exist on
the list of nodes traversed on any of the shortest paths from the
direct neighbor to the PQ-node. Table 4 is an illustration of the
mechanism with the topology in Figure 2.

```
+-------------+-------------------------+-----------+-----------+
| Candidate   | Repair Tunnel Path      | Link      | Node      |
| PQ-node     | (Repairing router to PQ-| Protection| Protection|
|             | node)                   |           |           |
+-------------+-------------------------+-----------+-----------+
| R2          | S->N->R1->R2            | Yes       | Yes       |
| R2          | S->E->R3->R2            | No        | No        |
| R3          | S->N->E->R3             | Yes       | No        |
+-------------+-------------------------+-----------+-----------+
```

            Table 4: Protection of Remote-LFA Tunnel to the PQ-Node

   As seen in the above Table 4, while R2 is a candidate node-protecting
   remote-LFA next hop for R3 and D2, it is not so for E and D1, since
   the primary next hop E is on the shortest path from R2 to E and D1.

2.3.2. Computing Node-Protecting Paths from PQ-Nodes to Destinations

   Once a computing router finds all the candidate node-protecting PQ-
   nodes for a given directly attached primary link, it shall follow the
   procedure as proposed in this section to choose one or more node-
   protecting R-LFA paths for destinations reachable through the same
   primary link in the primary SPF graph.

   To find a node-protecting R-LFA path for a given destination, the
   computing router needs to pick a subset of PQ-nodes from the
   candidate node-protecting PQ-space for the corresponding primary next
   hop, such that all the path(s) from the PQ-node(s) to the given
   destination remain unaffected in the event of a node failure of the
   primary next-hop node. To determine whether a given PQ-node belongs
   to such a subset of PQ-nodes, the computing router MUST ensure that
   none of the primary next-hop nodes are found on any of the shortest
   paths from the PQ-node to the given destination.

   This document proposes an additional forward SPF computation for each
   of the PQ-nodes to discover all shortest paths from the PQ-nodes to
   the destination. This will help determine whether or not a given
   primary next-hop node is on the shortest paths from the PQ-node to
   the given destination. To determine whether or not a given candidate
   node-protecting PQ-node provides node-protecting alternate for a
   given destination, all the shortest paths from the PQ-node to the
   given destination have to be inspected to check if the primary next-
   hop node is found on any of these shortest paths. To compute all the
   shortest paths from a candidate node-protecting PQ-node to one or
   more destinations, the computing router MUST run the forward SPF on
   the candidate node-protecting PQ-node. Soon after running the
   forward SPF, the computer router SHOULD run the inequality in
   Figure 6 below, once for each destination. A PQ-node that does not

qualify the condition for a given destination does not guarantee node
protection for the path segment from the PQ-node to the specific
destination.

D_opt(Y,D) < D_opt(Y,E) + Distance_opt(E,D)

Where,
   D_opt(A,B) : Distance on the most optimum path from A to B.
            D : The destination node.
            E : The primary next hop on the shortest path from S
                 to destination.
            Y : The node-protecting PQ-node being evaluated

     Figure 6: Node-Protecting Condition for PQ-Node to Destination

All of the above metric costs, except D_opt(Y, D), can be obtained
with forward and reverse SPFs with E (the primary next hop) as the
root, run as part of the regular LFA and remote-LFA implementation.
The Distance_opt(Y, D) metric can only be determined by the
additional forward SPF run with PQ-node Y as the root. With
reference to the topology in Figure 2, Table 5 shows that the above
condition can be used to determine node protection with a node-
protecting PQ-node R2.

Destination (D)	Primary-NH (E)	D_opt (Y, D)	D_opt (Y, E)	D_opt (E, D)	Condition Met
R3	E	1 (R2,R3)	2 (R2,E)	1 (E,R3)	Yes
E	E	2 (R2,E)	2 (R2,E)	0 (E,E)	No
D1	E	3 (R2,D1)	2 (R2,E)	1 (E,D1)	No
D2	E	2 (R2,D2)	2 (R2,E)	1 (E,D2)	Yes

     Table 5: Node-Protection Evaluation for R-LFA Path Segment between
                       PQ-Node and Destination

As seen in the example above, R2 does not meet the node-protecting
inequality for destination E and D1. And so, once again, while R2 is
a node-protecting remote-LFA next hop for R3 and D2, it is not so for
E and D1.

In SPF implementations that also produce a list of links and nodes
traversed on the shortest path(s) from a given root to others, the
inequality in Figure 6 above need not be evaluated. Instead, to
determine whether or not a PQ-node provides node protection for a
given destination, the list of nodes computed from forward SPF that
run on the PQ-node for the given destination SHOULD be inspected. In
case the list contains the primary next-hop node, the PQ-node does
not provide node protection. Else, the PQ-node guarantees the node-
protecting alternate for the given destination. Below is an
illustration of the mechanism with candidate node-protecting PQ-node
R2 in the topology in Figure 2.

Destination	Shortest Path (Repairing router to PQ-node)	Link Protection	Node Protection
R3	R2->R3	Yes	Yes
E	R2->R3->E	Yes	No
D1	R2->R3->E->D1	Yes	No
D2	R2->R3->D2	Yes	Yes

            Table 6: Protection of Remote-LFA Path between PQ-node and
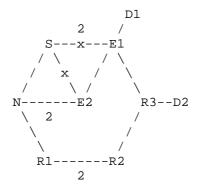   Destination

As seen in the above example, while R2 is a candidate node-protecting
R-LFA next hop for R3 and D2, it is not so for E and D1, since the
primary next hop E is on the shortest path from R2 to E and D1.

The procedure described in this document helps no more than to
determine whether or not a given remote-LFA alternate provides node
protection for a given destination. It does not find out any new
remote-LFA alternate next hops, outside the ones already computed by
the standard remote-LFA procedure. However, in the case of
availability of more than one PQ-node (remote-LFA alternates) for a
destination where node protection is required for the given primary
next hop, this procedure will eliminate the PQ-nodes that do not
provide node protection and choose only the ones that do.

2.3.3. Computing Node-Protecting R-LFA Paths for Destinations with
        Multiple Primary Next-Hop Nodes

In certain scenarios, when one or more destinations may be reachable
via multiple ECMP (equal-cost-multi-path) next-hop nodes and only
link protection is required, there is no need to compute any
alternate paths for such destinations. In the event of failure of
one of the next-hop links, the remaining primary next hops shall
always provide link protection. However, if node protection is

required, the rest of the primary next hops may not guarantee node
protection. Figure 7 below shows one such example topology.

```
                        D1
                 2     /
               S---x---E1
              / \     / \
             /   x   /   \
            /     \ /     \
           N-------E2     R3--D2
            \   2         /
             \           /
              \         /
              R1-------R2
                   2
```

```
Primary Next hops:
  Destination D1 = [{ S-E1, E1}, {S-E2, E2}]
  Destination D2 = [{ S-E1, E1}, {S-E2, E2}]
```

Figure 7: Topology with Multiple ECMP Primary Next Hops

In the above example topology, costs of all links are 1, except the
following links:

    Link: S-E1, Cost: 2

    Link: N-E2: Cost: 2

    Link: R1-R2: Cost: 2

In the above topology, on computing router S, destinations D1 and D2
are reachable via two ECMP next-hop nodes E1 and E2. However, the
primary paths via next-hop node E2 also traverse via the next-hop
node E1. So, in the event of node failure of next-hop node E1, both
primary paths (via E1 and E2) become unavailable. Hence, if node
protection is desired for destinations D1 and D2, alternate paths
that do not traverse any of the primary next-hop nodes E1 and E2 need
to be computed. In the above topology, the only alternate neighbor N
does not provide such an LFA alternate path. Hence, one or more
R-LFA node-protecting alternate paths for destinations D1 and D2,
needs to be computed.

In the above topology, the link-protecting PQ-nodes are as follows:

    Primary Next Hop: E1, Link-Protecting PQ-Node: { R2 }

    Primary Next Hop: E2, Link-Protecting PQ-Node: { R2 }

To find one (or more) node-protecting R-LFA paths for destinations D1
and D2, one (or more) node-protecting PQ-node(s) need to be
determined first. Inequalities specified in Sections 2.2.6.2 and
2.2.6.3 can be evaluated to compute the node-protecting PQ-space for
each of the next-hop nodes E1 and E2, as shown in Table 7 below. To
select a PQ-node as a node-protecting PQ-node for a destination with
multiple primary next-hop nodes, the PQ-node MUST satisfy the
inequality for all primary next-hop nodes. Any PQ-node that is NOT a
node-protecting PQ-node for all the primary next-hop nodes MUST NOT
be chosen as the node-protecting PQ-node for the destination.

Primary Next Hop (E)	Candidate PQ-node (Y)	Direct Nbr (Ni)	D_opt (Ni,Y)	D_opt (Ni,E)	D_opt (E,Y)	Condition Met
E1	R2	N	3 (N,R2)	3 (N,E1)	2 (E1,R2)	Yes
E2	R2	N	3 (N,R2)	2 (N,E2)	3 (E2,R2)	Yes

Table 7: Computing Node-Protected PQ-Nodes for Next Hop E1 and E2

In SPF implementations that also produce a list of links and nodes
traversed on the shortest path(s) from a given root to others, the
tunnel-repair paths from the computing router to candidate PQ-node
can be examined to ensure that none of the primary next-hop nodes are
traversed. PQ-nodes that provide one or more Tunnel-repair paths
that do not traverse any of the primary next-hop nodes are to be
considered as node-protecting PQ-nodes. Table 8 below shows the
possible tunnel-repair paths to PQ-node R2.

Primary-NH (E)	PQ-Node (Y)	Tunnel-Repair Paths	Exclude All Primary-NH
E1, E2	R2	S==>N==>R1==>R2	Yes

Table 8: Tunnel-Repair Paths to PQ-Node R2

From Tables 7 and 8 in the example above, R2 is a node-protecting PQ-
node for both primary next hops E1 and E2 and should be chosen as the
node-protecting PQ-node for destinations D1 and D2 that are both
reachable via the primary next-hop nodes E1 and E2.

Next, to find a node-protecting R-LFA path from a node-protecting PQ-
node to destinations D1 and D2, inequalities specified in Figure 6
should be evaluated to ensure that R2 provides a node-protecting
R-LFA path for each of these destinations, as shown below in Table 9.
For an R-LFA path to qualify as a node-protecting R-LFA path for a
destination with multiple ECMP primary next-hop nodes, the R-LFA path
from the PQ-node to the destination MUST satisfy the inequality for
all primary next-hop nodes.

Destinat ion (D)	Primary- NH (E)	PQ- Node (Y)	D_opt (Y, D)	D_opt (Y, E)	D_opt (E, D)	Condition Met
D1	E1	R2	3 (R2, D1)	2 (R2, E1)	1 (E1, D1)	No
D1	E2	R2	3 (R2, D1)	3 (R2, E2)	2 (E2, D1)	Yes
D2	E1	R2	2 (R2, D2)	2 (R2, E1)	2 (E1, D2)	Yes
D2	E2	R2	2 (R2, D2)	2 (R2, E2)	3 (E2, D2)	Yes

Table 9: Finding Node-Protecting R-LFA Path for
Destinations D1 and D2

In SPF implementations that also produce a list of links and nodes
traversed on the shortest path(s) from a given root to others, the
R-LFA paths via a node-protecting PQ-node to the final destination
can be examined to ensure that none of the primary next-hop nodes are
traversed. One or more R-LFA paths that do not traverse any of the
primary next-hop nodes guarantees node protection in the event of
failure of any of the primary next-hop nodes. Table 10 shows the
possible R-LFA-paths for destinations D1 and D2 via the node-
protecting PQ-node R2.

```
+-------------+-----------+---------+-----------------+-----------+
| Destination | Primary-NH | PQ-Node |   R-LFA Paths   | Exclude   |
|     (D)     |    (E)    |   (Y)   |                 |   All     |
|             |           |         |                 | Primary-NH|
+-------------+-----------+---------+-----------------+-----------+
|     D1      |  E1, E2   |   R2    | S==>N==>R1==>R2 |    No     |
|             |           |         | -->R3-->E1-->D1 |           |
|             |           |         |                 |           |
|     D2      |  E1, E2   |   R2    | S==>N==>R1==>R2 |    Yes    |
|             |           |         |   -->R3-->D2    |           |
+-------------+-----------+---------+-----------------+-----------+
```

             Table 10: R-LFA Paths for Destinations D1 and D2

   From Tables 9 and 10 in the example above, the R-LFA path from R2
   does not meet the node-protecting inequality for destination D1,
   while it does meet the same inequality for destination D2. So, while
   R2 provides a node-protecting R-LFA alternate for D2, it fails to
   provide node protection for destination D1. Finally, while it is
   possible to get a node-protecting R-LFA path for D2, no such node-
   protecting R-LFA path can be found for D1.

2.3.4. Limiting Extra Computational Overhead

   In addition to the extra reverse SPF computations suggested by the
   Remote-LFA document [RFC7490] (one reverse SPF for each of the
   directly connected neighbors), this document proposes a forward SPF
   computation for each PQ-node discovered in the network. Since the
   average number of PQ-nodes found in any network is considerably more
   than the number of direct neighbors of the computing router, the
   proposal of running one forward SPF per PQ-node may add considerably
   to the overall SPF computation time.

   To limit the computational overhead of the approach proposed, this
   document specifies that implementations MUST choose a subset from the
   entire set of PQ-nodes computed in the network, with a finite limit
   on the number of PQ-nodes in the subset. Implementations MUST choose
   a default value for this limit and may provide the user with a
   configuration knob to override the default limit. This document
   suggests 16 as a default value for this limit. Implementations MUST
   also evaluate some default preference criteria while considering a
   PQ-node in this subset. The exact default preference criteria to be
   used is outside the scope of this document and is a matter of
   implementation. Finally, implementations MAY also allow the user to
   override the default preference criteria, by providing a policy
   configuration for the same.

This document proposes that implementations SHOULD use a default
preference criteria for PQ-node selection that will put a score on
each PQ-node, proportional to the number of primary interfaces for
which it provides coverage, its distance from the computing router,
and its router-id (or system-id in case of IS-IS). PQ-nodes that
cover more primary interfaces SHOULD be preferred over PQ-nodes that
cover fewer primary interfaces. When two or more PQ-nodes cover the
same number of primary interfaces, PQ-nodes that are closer (based on
metric) to the computing router SHOULD be preferred over PQ-nodes
farther away from it. For PQ-nodes that cover the same number of
primary interfaces and are the same distance from the computing
router, the PQ-node with smaller router-id (or system-id in case of
IS-IS) SHOULD be preferred.

Once a subset of PQ-nodes is found, a computing router shall run a
forward SPF on each of the PQ-nodes in the subset to continue with
procedures proposed in Section 2.3.2.

3. Manageability of Remote-LFA Alternate Paths

3.1. The Problem

With the regular remote-LFA [RFC7490] functionality, the computing
router may compute more than one PQ-node as usable remote-LFA
alternate next hops. Additionally, [RFC7916] specifies an LFA (and a
remote-LFA) manageability framework, in which an alternate selection
policy may be configured to let the network operator choose one of
them as the most appropriate remote-LFA alternates. For such a
policy-based alternate selection to run, the computing router needs
to collect all the relevant path characteristics (as specified in
Section 6.2.4 of [RFC7916]) for each of the alternate paths (one
through each of the PQ-nodes). As mentioned before in Section 2.3,
the R-LFA alternate path through a given PQ-node to a given
destination is comprised of two path segments. Section 6.2.4 of
[RFC7916] specifies that any kind of alternate selection policy must
consider path characteristics for both path segments while evaluating
one or more RLFA alternate paths.

The first path segment (i.e., from the computing router to the PQ-
node) can be calculated from the regular forward SPF done as part of
standard and remote LFA computations. However, without the mechanism
proposed in Section 2.3.2 of this document, there is no way to
determine the path characteristics for the second path segment (i.e.,
from the PQ-node to the destination). In the absence of the path
characteristics for the second path segment, two remote-LFA alternate
paths may be equally preferred based on the first path segment
characteristics only, although the second path segment attributes may
be different.

3.2. The Solution

   The additional forward SPF computation proposed in Section 2.3.2
   shall also collect links, nodes, and path characteristics along the
   second path segment. This shall enable the collection of complete
   path characteristics for a given remote-LFA alternate path to a given
   destination. The complete alternate path characteristics shall then
   facilitate more accurate alternate path selection while running the
   alternate selection policy.

   As already specified in Section 2.3.4, to limit the computational
   overhead of the proposed approach, forward SPF computations must be
   run on a selected subset from the entire set of PQ-nodes computed in
   the network, with a finite limit on the number of PQ-nodes in the
   subset. The detailed suggestion on how to select this subset is
   specified in the same section. While this limits the number of
   possible alternate paths provided to the alternate-selection policy,
   this is needed to keep the computational complexity within affordable
   limits. However, if the alternate-selection policy is very
   restrictive, this may leave few destinations in the entire topology
   without protection. Yet this limitation provides a necessary
   tradeoff between extensive coverage and immense computational
   overhead.

   The mechanism proposed in this section does not modify or invalidate
   any part of [RFC7916]. This document specifies a mechanism to meet
   the requirements specified in Section 6.2.5.4 of [RFC7916].

4. IANA Considerations

   This document does not require any IANA actions.

5. Security Considerations

   This document does not introduce any change in any of the protocol
   specifications. It simply proposes to run an extra SPF rooted on
   each PQ-node discovered in the whole network.

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,
              <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,
              DOI 10.17487/RFC5286, September 2008,
              <http://www.rfc-editor.org/info/rfc5286>.

   [RFC7490] 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>.

6.2. Informative References

   [RFC7916] Litkowski, S., Ed., Decraene, B., Filsfils, C., Raza, K.,
              Horneffer, M., and P. Sarkar, "Operational Management of
              Loop-Free Alternates", RFC 7916, DOI 10.17487/RFC7916,
              July 2016, <http://www.rfc-editor.org/info/rfc7916>.

Acknowledgements

Authors' Addresses

   Pushpasis Sarkar (editor)
   Arrcus, Inc.

   Email: pushpasis.ietf@gmail.com

   Shraddha Hegde
   Juniper Networks, Inc.
   Electra, Exora Business Park
   Bangalore, KA   560103
   India

   Email: shraddha@juniper.net

   Chris Bowers
   Juniper Networks, Inc.
   1194 N. Mathilda Ave.
   Sunnyvale, CA   94089
   United States of America

   Email: cbowers@juniper.net

   Hannes Gredler
   RtBrick, Inc.

   Email: hannes@rtbrick.com

   Stephane Litkowski
   Orange

   Email: stephane.litkowski@orange.com