Network Working Group   S. Bryant
Internet Draft   M. Shand
Expiration Date: June 2007   S. Previdi
                                                           Cisco Systems

                                                              Dec 2006

IP Fast Reroute Using Not-via Addresses
<draft-ietf-rtgwg-ipfrr-notvia-addresses-00.txt>

Status of this Memo

Abstract
   This draft describes a mechanism that provides fast reroute in an IP
   network through encapsulation to "not-via" addresses. A single level
   of encapsulation is used. The mechanism protects unicast, multicast
   and LDP traffic against link, router and shared risk group failure,
   regardless of network topology and metrics.

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 [RFC2119].

Table of Contents

Introduction

   When a link or a router fails, only the neighbors of the failure are
   initially aware that the failure has occurred. In a network
   operating IP fast reroute [IPFRR], the routers that are the
   neighbors of the failure repair the failure. These repairing routers
   have to steer packets to their destinations despite the fact that
   most other routers in the network are unaware of the nature and
   location of the failure.

   A common limitation in most IPFRR mechanisms is an inability to
   steer the repaired packet round an identified failure. The extent to
   which this limitation affects the repair coverage is topology
   dependent. The mechanism proposed here is to encapsulate the packet
   to an address that explicitly identifies the network component that
   the repair must avoid. This produces a repair mechanism, which,
   provided the network is not partitioned by the failure, will always
   achieve a repair.

1. Overview of Not-via Repairs

   The purpose of a repair is to deliver packets to their destination
   without traversing a known failure in the network, i.e. to deliver
   the packet not via the failure. A special address is assigned to
   each protected component. This address is called the not-via
   address. The semantics of a not-via address are that a packet
   addressed to a not-via address must be delivered to the router
   advertising that address, not via the protected component (link,
   node, SRLG etc.) with which that address is associated.

   A simple example would be node repair in which an additional address
   is assigned to each interface in the network. To repair a failure,
   the repairing router encapsulates the packet to the not-via address
   of the router interface on the far side of the failure. The routers
   on the repair path then know to which router they must deliver the
   packet, and which network component they must avoid. The network
   fragment shown in Figure 1 illustrates a not-via repair for the case
   of a router failure.

```
             A
             |                      Bp is the address to use to get
             |                        a packet to B not-via P
             |
   S---------P----------B. . . . . . . . . .D
    \        |          Bp^
     \       |           |
      \      |           |
       \     C           |
        \                |
         ----------------+
           Repair to Bp
```
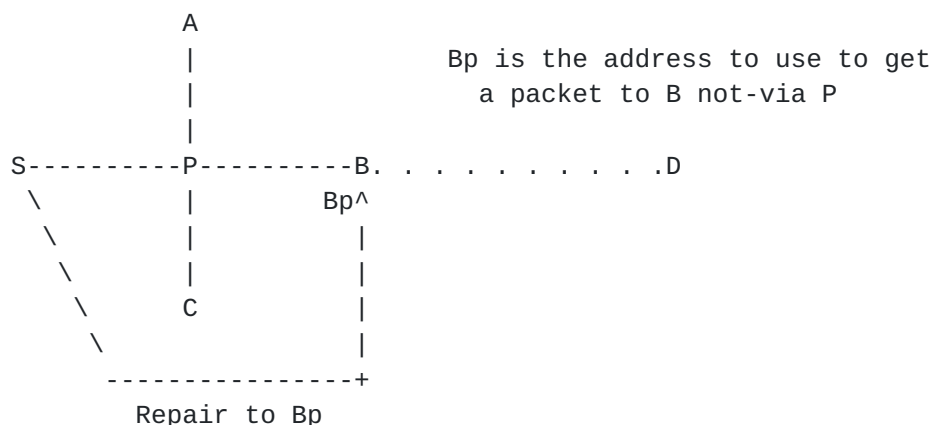
Figure 1: Not-via repair of router failure

Assume that S has a packet for some destination D that it would
normally send via P and B, and that S suspects that P has failed. S
encapsulates the packet to Bp. The path from S to Bp is the shortest
path from S to B not going via P. If the network contains a path
from S to B that does not transit router P, i.e. the network is not
partitioned by the failure of P, then the packet will be
successfully delivered to B. When the packet addressed to Bp arrives
at B, B removes the encapsulation and forwards the repaired packet
towards its final destination.

Note that if the path from B to the final destination includes one
or more nodes that are included in the repair path, a packet may
back track after the encapsulation is removed. However, because the
decapsulating router is always closer to the packet destination than
the encapsulating router, the packet will not loop.

For complete protection, all of P's neighbors will require a not-via
address that allows traffic to be directed to them without
traversing P. This is shown in Figure 2.

```
             A
             |Ap
             |
   Sp      Pa|Pb
   S---------P----------B
           Ps|Pc        Bp
             |
           Cp|
             C
```

Figure 2: The set of Not-via P Addresses

1.1 Use of Equal Cost Multi-Path

A router can use an equal cost multi-path (ECMP) repair in place of a not-via repair.

A router computing a not-via repair path MAY subject the repair to ECMP.

1.2 Use of LFA repairs

The not-via approach provides complete repair coverage and therefore may be used as the sole repair mechanism. There are, however, advantages in using not-via in combination with loop free alternates (LFA) as documented in [LFA].

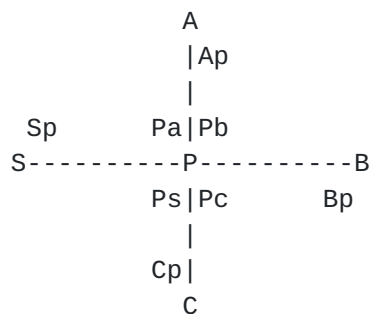LFAs are computed on a per destination basis and in general, only a subset of the destinations requiring repair will have a suitable LFA repair. In this case, those destinations which are repairable by LFAs are so repaired and the remainder of the destinations are repaired using the not-via encapsulation. This has the advantage of reducing the volume of traffic that requires encapsulation. On the other hand, the path taken by an LFA repair may be less optimal than that of the equivalent not-via repair. The description in this document assumes that LFAs will be used where available, but the distribution of repairs between the two mechanisms is a local implementation choice.

2. Not-via Repair Path Computation

The not-via repair mechanism requires that all routers on the path from S to B (Figure 1) have a route to Bp. They can calculate this by failing node P, running an SPF, and finding the shortest route to B.

A router has no simple way of knowing whether it is on the shortest path for any particular repair. It is therefore necessary for every router to calculate the path it would use in the event of any possible router failure. Each router therefore "fails" every router in the network, one at a time, and calculates its own best route to each of the neighbors of that router. In other words, with reference to Figure 2, some router X will consider each router in turn to be P, fail P, and then calculate its own route to each of the not-via P addresses advertised by the neighbors of P. i.e. X calculates its route to Sp, Ap, Bp, and Cp, in each case, not via P.

To calculate the repair paths a router has to calculate n-1 SPFs where n is the number of routers in the network. This is expensive to compute. However, the problem is amenable to a solution in which each router (X) proceeds as follows. X first calculates the base

topology with all routers functional and determines its normal path

to all not-via addresses. This can be performed as part of the
normal SPF computation. For each router P in the topology, X then
performs the following actions:-

   1. Removes router P from the topology.

   2. Performs an incremental SPF [ISPF] on the modified topology.
      The iSPF process involves detaching the sub-tree affected by
      the removal of router P, and then re-attaching the detached
      nodes. However, it is not necessary to run the iSPF to
      completion. It is sufficient to run the iSPF up to the point
      where all of the nodes advertising not-via P addresses have
      been re-attached to the SPT, and then terminate it.

   3. Reverts to the base topology.

This algorithm is significantly less expensive than a set of full
SPFs. Thus, although a router has to calculate the repair paths for
n-1 failures, the computational effort is much less than n-1 SPFs.

Experiments on a selection of real world network topologies with
between 40 and 400 nodes suggest that the worst-case computational
complexity using the above optimizations is equivalent to performing
between 5 and 13 full SPFs. Further optimizations are described in
section 6.

3. Operation of Repairs

This section explains the basic operation of the not-via repair of
node and link failure.

3.1 Node Failure

When router P fails (Figure 2) S encapsulates any packet that it
would send to B via P to Bp, and then sends the encapsulated packet
on the shortest path to Bp. S follows the same procedure for routers
A and C in Figure 2. The packet is decapsulated at the repair target
(A, B or C) and then forwarded normally to its destination. The
repair target can be determined as part of the normal SPF by
recording the "next-next-hop" for each destination in addition to
the normal next-hop.

Notice that with this technique only one level of encapsulation is
needed, and that it is possible to repair ANY failure regardless of
link metrics and any asymmetry that may be present in the network.
The only exception to this is where the failure was a single point
of failure that partitioned the network, in which case ANY repair is
clearly impossible.

3.2 Link Failure

The normal mode of operation of the network would be to assume
router failure. However, where some destinations are only reachable
through the failed router, it is desirable that an attempt be made
to repair to those destinations by assuming that only a link failure
has occurred.

To perform a link repair, S encapsulates to Ps (i.e. it instructs
the network to deliver the packet to P not-via S). All of the
neighbors of S will have calculated a path to Ps in case S itself
had failed. S could therefore give the packet to any of its
neighbors (except, of course, P). However, S should preferably send
the encapsulated packet on the shortest available path to P. This
path is calculated by running an SPF with the link SP failed. Note
that this may again be an incremental calculation, which can
terminate when address Ps has been reattached.

It is necessary to consider the behavior of IPFRR solutions when a
link repair is attempted in the presence of node failure. In its
simplest form the not-via IPFRR solution prevents the formation of
loops forming as a result of mutual repair, by never providing a
repair path for a not-via address. Referring to Figure 2, if A was
the neighbor of P that was on the link repair path from S to P, and
P itself had failed, the repaired packet from S would arrive at A
encapsulated to Ps. A would have detected that the AP link had
failed and would normally attempt to repair the packet. However, no
repair path is provided for any not-via address, and so A would be
forced to drop the packet, thus preventing the formation of loop.

3.3 Multi-homed Prefix

A multi-homed Prefix (MHP) is a prefix that is reachable via more
than one router in the network. Some of these may be repairable
using LFAs as described in [LFA]. Only those without such a repair
need be considered here.

When IPFRR router S (Figure 3) discovers that P has failed, it needs
to send packets addressed to the MHP X, which is normally reachable
through P, to an alternate router, which is still able to reach X.

```
    X                        X                        X
    |                        |                        |
    |                        |                        |
    |            Sp          |Pb                       |
    Z...............S---------P---------B...............Y
                   Ps|Pc        Bp
                     |
                   Cp|
                     C
```
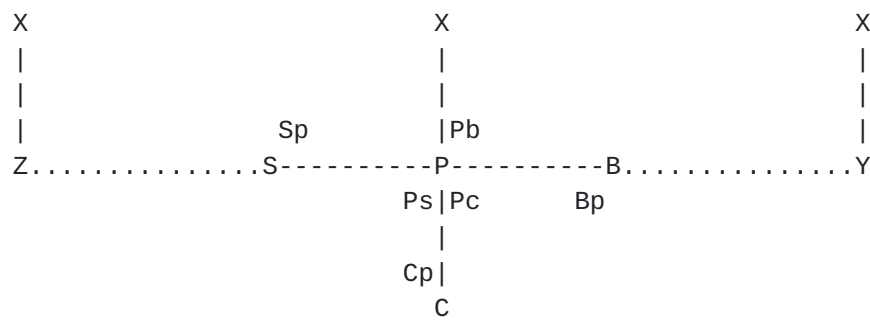
                     Figure 3: Multi-homed Prefixes

     S should choose the closest router that can reach X during the
     failure as the alternate router. S determines which router to use as
     the alternate while running the SPF with P failed. This is
     accomplished by the normal process of re-attaching a leaf node to
     the core topology (this is sometimes known as a "partial SPF").

     First, consider the case where the shortest alternate path to X is
     via Z. S can reach Z without using the failed router P. However, S
     cannot just send the packet towards Z, because the other routers in
     the network will not be aware of the failure of P, and may loop the
     packet back to S. S therefore encapsulates the packet to Z (using a
     normal address for Z). When Z receives the encapsulated packet it
     removes the encapsulation and forwards the packet to X.

     Now consider the case where the shortest alternate path to X is via
     Y, which S reaches via P and B. To reach Y, S must first repair the
     packet to B using the normal not-via repair mechanism. To do this S
     encapsulates the packet for X to Bp. When B receives the packet it
     removes the encapsulation and discovers that the packet is intended
     for MHP X. The situation now reverts to the previous case, in which
     the shortest alternate path does not require traversal of the
     failure. B therefore follows the algorithm above and encapsulates
     the packet to Y (using a normal address for Y). Y removes the
     encapsulation and forwards the packet to X.

     It may be that the cost of reaching X using local delivery from the
     alternate router (i.e. Z or Y) is greater than the cost of reaching
     X via P. Under those circumstances, the alternate router would
     normally forward to X via P, which would cause the IPFRR repair to
     loop. To prevent the repair from looping the alternate router must
     locally deliver a packet received via a repair encapsulation. This
     may be specified by using a special address with the above
     semantics. Note that only one such address is required per node.

     Notice that using the not-via approach, only one level of

encapsulation was needed to repair MHPs to the alternate router.

3.4 **Installation of Repair Paths**

The following algorithm is used by node S (Figure 3) to pre-calculate and install repair paths in the FIB, ready for immediate use in the event of a failure.

For each neighbor P, consider all destinations which are reachable via P in the current topology:-

1. For all destinations with an ECMP or LFA repair (as described in [LFA]) install that repair.

2. For each destination (DR) that remains, identify in the current topology the next-next-hop (H) (i.e. the neighbor of P that P will use to send the packet to DR).

3. For each next-next-hop node H for which S has a path to the not-via address Hp (H not via P), identify each destination with current next-next-hop H and install a not-via repair to Hp for that destination.

4. Identify all remaining destinations (M) which can still be reached when node P fails. These will be multi-homed prefixes that are not repairable by LFA, for which the normal attachment node is P, or a router for which P is a single point of failure, and have an attachment point that is reachable after P has failed.

5. For each multi-homed prefix (M) identified in step (4):-

   a. Identify the new attachment node (as shown in Figure 3). This may be:-

      i. Y, where the next hop towards Y is P, or

      ii. Z, where the next hop towards Z is not P.

   b. If the attachment node is Z, install the repair for M as a tunnel to Z' (where Z' is the address of Z that is used to force local forwarding).

   c. For the subset of prefixes (M) that remain (having attachment point Y), install the repair path previously installed for destination Y.

6. For each destination (DS) that remains, install a not-via repair to Ps (P not via S). Note, these are destinations for which node P is a single point of failure, and can only be repaired by assuming that the apparent failure of node P was

simply a failure of the S-P link.

[4](#). Compound failures

[4.1](#) Shared Risk Link Groups

   A Shared Risk Link Group (SRLG) is a set of links whose failure can
   be caused by a single action such as a conduit cut or line card
   failure. When repairing the failure of a link that is a member of an
   SRLG, it must be assumed that all the other links that are also
   members of the SRLG have also failed. Consequently, any repair path
   must be computed to avoid not just the adjacent link, but also all
   the links which are members of the same SRLG.

   In Figure 4 below, the links S-P and A-B are both members of
   SRLG "a". The semantics of the not-via address Ps changes from
   simply "P not-via the link S-P" to be "P not-via the link S-P or any
   other link with which S-P shares an SRLG" In Figure 4 this is the
   links that are members of SRLG "a". I.e. links S-P and A-B. Since
   the information about SRLG membership of all links is available in
   the Link State Database, all nodes computing routes to the not-via
   address Ps can infer these semantics, and perform the computation by
   failing all the links in the SRLG when running the iSPF.

   Note that it is not necessary for S to consider repairs to any other
   nodes attached to members of the SRLG (such as B). It is sufficient
   for S to repair to the other end of the adjacent link (P in this
   case).

```
           a    Ps
       S----------P---------D
       |          |
       |    a     |
       A----------B
       |          |
       |          |
       C----------E
```


          Figure 4: Shared Risk Link Group
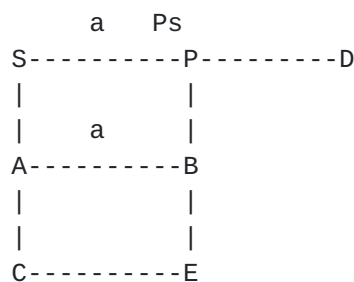
   In some cases, it may be that the links comprising the SRLG occur in
   series on the path from S to the destination D, as shown in Figure
   5. In this case, multiple consecutive repairs may be necessary. S
   will first repair to Ps, then P will repair to Dp. In both cases,
   because the links concerned are members of SRLG "a" the paths are
   computed to avoid all members of SRLG "a".

```
          a   Ps    a   Dp
       S----------P---------D
       |          |         |
       |    a     |         |
       A----------B         |
       |          |         |
       |          |         |
       C----------E---------F
```
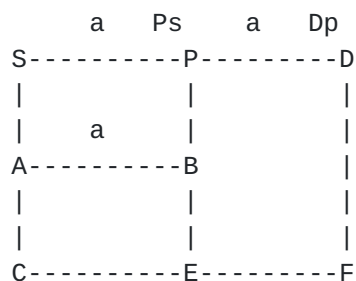
Figure 5: Shared Risk Link Group members in series

While the use of multiple repairs in series introduces some
additional overhead, these semantics avoid the potential
combinatorial explosion of not-via addresses that could otherwise
occur.

Note that although multiple repairs are used, only a single level of
encapsulation is required. This is because the first repair packet
is de-capsulated before the packet is re-encapsulated using the not-
via address corresponding to the far side of the next link which is
a member of the same SRLG. In some cases the de-capsulation and re-
encapsulation takes place (at least notionally) at a single node,
while in other cases, these functions may be performed by different
nodes. This scenario is illustrated in Figure 6 below.

```
          a   Ps              a  Dg
       S----------P---------G--------D
       |          |         |        |
       |    a     |         |        |
       A----------B         |        |
       |          |         |        |
       |          |         |        |
       C----------E---------F--------H
```
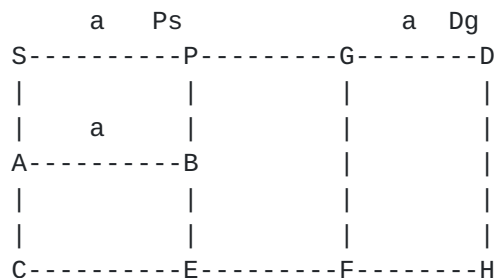
Figure 6: Shared Risk Link Group members in series

In this case, S first encapsulates to Ps, and node P decapsulates
the packet and forwards it "native" to G using its normal FIB entry
for destination D. G then repairs the packet to Dg.

It can be shown that such multiple repairs can never form a loop
because each repair causes the packet to move closer to its
destination.

It is often the case that a single link may be a member of multiple
SRLGs, and those SRLG may not be isomorphic. This is illustrated in
Figure 7 below.

```
        ab  Ps                a  Dg
      S----------P---------G--------D
      |          |         |        |
      |     a    |         |        |
      A----------B         |        |
      |          |         |        |
      |     b    |         |   b    |
      C----------E---------F--------H
      |          |
      |          |
      J----------K
```

             Figure 7: Multiple Shared Risk Link Groups

   The link SP is a member of SRLGs "a" and "b". When a failure of the
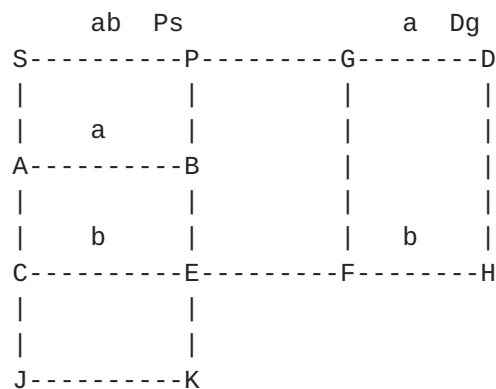   link SP is detected, it must be assumed that BOTH SRLGs have failed.
   Therefore the not-via path to Ps must be computed by failing all
   links which are members of SRLG "a" or SRLG "b". I.e. the semantics
   of Ps is now "P not-via any links which are members of any of the
   SRLGs of which link SP is a member". This is illustrated in Figure 8
   below.

```
        ab  Ps                a  Dg
      S----/-----P---------G---/----D
      |          |         |        |
      |     a    |         |        |
      A----/-----B         |        |
      |          |         |        |
      |     b    |         |   b    |
      C----/-----E---------F---/----H
      |          |
      |          |
      J----------K
```
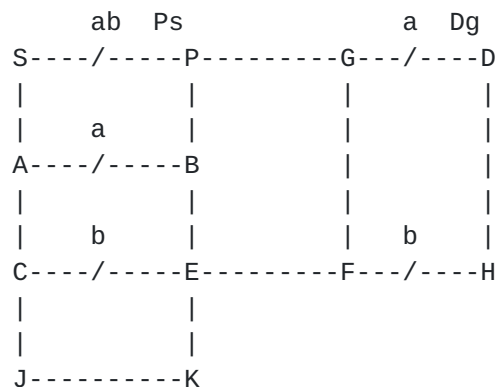
      Figure 8: Topology used for repair computation for link S-P

   In this case, the repair path to Ps will be S-A-C-J-K-E-B-P. It may
   appear that there is no path to D because GD is a member of SRLG "a"
   and FH is a member of SRLG "b". This is true if BOTH SRLGs "a" and
   "b" have in fact failed. But that would be an instance of multiple
   uncorrelated failures which are out of scope for this design. In
   practice it is likely that there is only a single failure, i.e.
   either SRLG "a" or SRLG "b" has failed, but not both. These two
   possibilities are indistinguishable from the point of view of the
   repairing router S and so it must repair on the assumption that both

are unavailable. However, each link repair is considered
independently. The repair to Ps delivers the packet to P which then
forwards the packet to G. When the packet arrives at G, if SRLG "a"
has failed it will be repaired around the path G-F-H-D. This is
illustrated in Figure 9 below. If, on the other hand, SRLG "b" has
failed, link GD will still be available. In this case the packet
will be delivered as normal across the link GD.

```
          ab  Ps                  a  Dg
      S----/-----P---------G---/----D
      |          |         |        |
      |    a     |         |        |
      A----/-----B         |        |
      |          |         |        |
      |    b     |         |   b    |
      C----------E---------F--------H
      |          |
      |          |
      J----------K
```


Figure 9: Topology used for repair computation for link G-D

A repair strategy that assumes the worst-case failure for each link
can often result in longer repair paths than necessary. In cases
where only a single link fails, rather than the full SRLG, this
strategy may occasionally fail to identify a repair even though a
viable repair path exists in the network. The use of sub-optimal
repair paths is an inevitable consequence of this compromise
approach. The failure to identify any repair is a serious
deficiency, but is a rare occurrence in a robustly designed network.
This problem can be addressed by:-

   1. Reporting that the link in question is irreparable, so that the
      network designer can take appropriate action.

   2. Modifying the design of the network to avoid this possibility.

   3. Using some form of SRLG diagnostic (for example, by running BFD
      over alternate repair paths) to determine which SRLG member(s)
      has actually failed and using this information to select an
      appropriate pre-computed repair path. However, aside from the
      complexity of performing the diagnostics, this requires
      multiple not-via addresses per interface, which has poor
      scaling properties.

4.1.1 Use of LFAs with SRLGs

Section 4.1 above describes the repair of links which are members of
one or more SRLGs. LFAs can be used for the repair of such links
provided that any other link with which S-P shares an SRLG is
avoided when computing the LFA. This is described for the simple
case of "local-SRLGs" in [LFA].

4.2 Local Area Networks

LANs are a special type of SRLG and are solved using the SRLG
mechanisms outlined above. With all SRLGs there is a trade-off
between the sophistication of the fault detection and the size of
the SRLG. Protecting against link failure of the LAN link(s) is
relatively straightforward, but as with all fast reroute mechanisms,
the problem becomes more complex when it is desired to protect
against the possibility of failure of the nodes attached to the LAN
as well as the LAN itself.

```
             +--------------Q------C
             |
             |
             |
   A--------S-------(N)-------------P------B
             |
             |
             |
             +--------------R------D
```
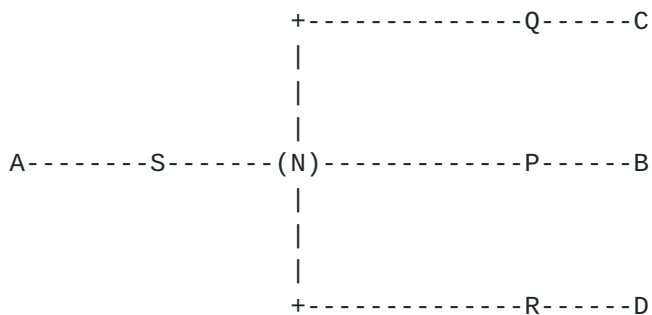
                Figure 10: Local Area Networks

Consider the LAN shown in Figure 10. For connectivity purposes, we
consider that the LAN is represented by the pseudonode (N). To
provide IPFRR protection, S must run a connectivity check to each of
its protected LAN adjacencies P, Q, and R, using, for example BFD
[BFD].

When S discovers that it has lost connectivity to P, it is unsure
whether the failure is:

   . its own interface to the LAN,

   . the LAN itself,

   . the LAN interface of P,

   . the node P.

4.2.1 Simple LAN Repair

A simple approach to LAN repair is to consider the LAN and all of
its connected routers as a single SRLG. Thus, the address P not via
the LAN (Pl) would require P to be reached not-via any router
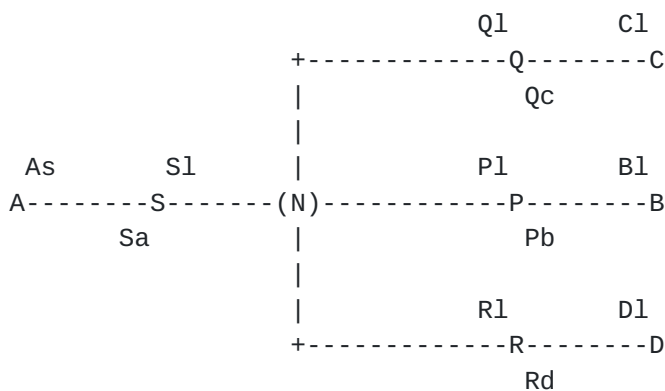connected to the LAN. This is shown in Figure 11.

```
                           Ql         Cl
                 +-------------Q--------C
                 |            Qc
                 |
   As       Sl   |            Pl        Bl
    A-------S-------(N)------------P--------B
        Sa       |            Pb
                 |
                 |            Rl        Dl
                 +-------------R--------D
                           Rd
```

                Figure 11: Local Area Networks - LAN SRLG

In this case, when S detected that P had failed it would send
traffic reached via P and B to B not-via the LAN or any router
attached to the LAN (i.e. to Bl). Any destination only reachable
through P would be addressed to P not-via the LAN or any router
attached to the LAN (except of course P).

Whilst this approach is simple, it assumes that a large portion of
the network adjacent to the failure has also failed. This will
result in the use of sub-optimal repair paths and in some cases the
inability to identify a viable repair.

4.2.2 LAN Component Repair

In this approach, possible failures are considered at a finer
granularity, but without the use of diagnostics to identify the

specific component that has failed. Because S is unable to diagnose
the failure it must repair traffic sent through P and B, to B not-
via P,N (i.e. not via P and not via N), on the conservative
assumption that both the entire LAN and P have failed. Destinations
for which P is a single point of failure must as usual be sent to P
using an address that avoids the interface by which P is reached
from S, i.e. to P not-via N. Similarly for routers Q and R.

Notice that each router that is connected to a LAN must, as usual,
advertise one not-via address for each neighbor. In addition, each
router on the LAN must advertise an extra address not via the
pseudonode (N).

Notice also that each neighbor of a router connected to a LAN must
advertise two not-via addresses, the usual one not via the neighbor
and an additional one, not via either the neighbor or the
pseudonode. The required set of LAN address assignments is shown in
Figure 12 below. Each router on the LAN, and each of its neighbors,
is advertising exactly one address more than it would otherwise have
advertised if this degree of connectivity had been achieved using
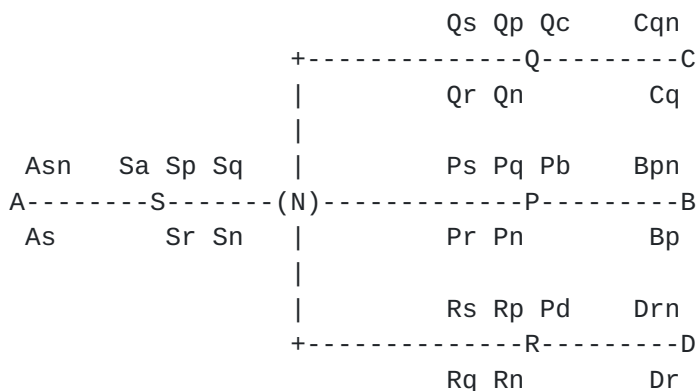point-to-point links.

```
                              Qs Qp Qc     Cqn
                  +--------------Q---------C
                  |             Qr Qn          Cq
                  |
    Asn    Sa Sp Sq  |             Ps Pq Pb     Bpn
    A--------S-------(N)-------------P---------B
     As         Sr Sn  |             Pr Pn          Bp
                  |
                  |             Rs Rp Pd     Drn
                  +--------------R---------D
                              Rq Rn          Dr
```

Figure 12: Local Area Networks

[4.2.3](#) LAN Repair Using Diagnostics

A more specific LAN repair can be undertaken by using diagnostics.
In order to explicitly diagnose the failed network component, S
correlates the connectivity reports from P and one or more of the
other routers on the LAN, in this case, Q and R. If it lost
connectivity to P alone, it could deduce that the LAN was still
functioning and that the fault lay with either P, or the interface
connecting P to the LAN. It would then repair to B not via P (and P
not-via N for destinations for which P is a single point of failure)
in the usual way. If S lost connectivity to more than one router on

the LAN, it could conclude that the fault lay only with the LAN, and
could repair to P, Q and R not-via N, again in the usual way.

5. Multiple Simultaneous Failures

The failure of a node or an SRLG can result in multiple correlated
failures, which may be repaired using the mechanisms described in
this design. This design will not correctly repair a set of
unanticipated multiple failures. Such failures are out of scope of
this design and are for further study.

It is important that the routers in the network are able to
discriminate between these two classes of failure, and take
appropriate action.

6. Optimizing not-via computations using LFAs

If repairing node S has an LFA to the repair endpoint it is not
necessary for any router to perform the incremental SPF with the
link SP removed in order to compute the route to the not-via address
Ps. This is because the correct routes will already have been
computed as a result of the SPF on the base topology. Node S can
signal this condition to all other routers by including a bit in its
LSP or LSA associated with each LFA protected link. Routers
computing not-via routes can then omit the running of the iSPF for
links with this bit set.

When running the iSPF for a particular link AB, the calculating
router first checks whether the link AB is present in the existing
SPT. If the link is not present in the SPT, no further work is
required. This check is a normal part of the iSPF computation.

If the link is present in the SPT, this optimization introduces a
further check to determine whether the link is marked as protected
by an LFA in the direction in which the link appears in the SPT. If
so the iSPF need not be performed. For example, if the link appears
in the SPT in the direction A->B and A has indicated that the link
AB is protected by an LFA no further action is required for this
link.

If the receipt of this information is delayed, the correct operation
of the protocol is not compromised provided that the necessity to
perform a not-via computation is re-evaluated whenever new
information arrives.

This optimization is not particularly beneficial to nodes close to
the repair since, as has been observed above, the computation for
nodes on the LFA path is trivial. However, for nodes upstream of the
link SP for which S-P is in the path to P, there is a significant

reduction in the computation required.

7. Multicast

Multicast traffic can be repaired in a similar way to unicast. The multicast forwarder is able to use the not-via address to which the multicast packet was addressed as an indication of the expected receive interface and hence to correctly run the required RPF check.

In some cases, all the destinations, including the repair endpoint, are repairable by an LFA. In this case, all unicast traffic may be repaired without encapsulation. Multicast traffic still requires encapsulation, but for the nodes on the LFA repair path the computation of the not-via forwarding entry is unnecessary since, by definition, their normal path to the repair endpoint is not via the failure.

A more complete description of multicast operation will be provided in a future version of this draft.

8. Fast Reroute in an MPLS LDP Network.

Not-via addresses are IP addresses and LDP [LDP] will distribute labels for them in the usual way. The not-via repair mechanism may therefore be used to provide fast re-route in an MPLS network by first pushing the label which the repair endpoint uses to forward the packet, and then pushing the label corresponding to the not-via address needed to effect the repair. Referring once again to Figure 1, if S has a packet destined for D that it must reach via P and B, S first pushes B's label for D. S then pushes the label that its next hop to Bp needs to reach Bp.

Note that in an MPLS LDP network it is necessary for S to have the repair endpoint's label for the destination. When S is effecting a link repair it already has this. In the case of a node repair, S either needs to set up a directed LDP session with each of its neighbor's neighbors, or it needs to use the next-next hop label distribution mechanism proposed in [NNHL].

9. Encapsulation

Any IETF specified IP in IP encapsulation may be used to carry a not-via repair. IP in IP [IPIP], GRE [GRE] and L2TPv3 [L2TPv3], all have the necessary and sufficient properties. The requirement is that both the encapsulating router and the router to which the encapsulated packet is addressed have a common ability to process the chosen encapsulation type.

When an MPLS LDP network is being protected, the encapsulation would normally be an additional MPLS label. In an MPLS enabled IP network an MPLS label may be used in place of an IP in IP encapsulation in

the case above.

10. Routing Extensions

   IPFRR requires IGP extensions. Each IPFRR router that is directly
   connected to a protected network component must advertise a not-via
   address for that component. This must be advertised in such a way
   that the association between the protected component (link, router
   or SRLG) and the not-via address can be determined by the other
   routers in the network.

   It is necessary that not-via capable routers advertise in the IGP
   that they will calculate not-via routes.

   It is necessary for routers to advertise the type of encapsulation
   that they support (MPLS, GRE [GRE], L2TPv3 etc). However, the
   deployment of mixed IP encapsulation types within a network is
   discouraged.

11. Incremental Deployment

   Incremental deployment is supported by excluding routers that are
   not calculating not-via routes (as indicated by their capability
   information flooded with their link state information) from the base
   topology used for the computation of repair paths. In that way
   repairs may be steered around islands of routers that are not IPFRR
   capable.

   Routers that are protecting a network component need to have the
   capability to encapsulate and de-capsulate packets. However, routers
   that are on the repair path only need to be capable of calculating
   not-via paths and including the not-via addresses in their FIB i.e.
   these routers do not need any changes to their forwarding mechanism.

12. IANA considerations

   There are no IANA considerations that arise from this draft.

13. Security Considerations

   The repair endpoints present vulnerability in that they might be
   used as a method of disguising the delivery of a packet to a point
   in the network. The primary method of protection should be through
   the use of a private address space for the not-via addresses. These
   addresses MUST NOT be advertised outside the area, and SHOULD be
   filtered at the network entry points. In addition, a mechanism might
   be developed that allowed the use of the mild security available
   through the use of a key [GRE] [L2TPv3]. With the deployment of such
   mechanisms, the repair endpoints would not increase the security
   risk beyond that of existing IP tunnel mechanisms.

An attacker may attempt to overload a router by addressing an
excessive traffic load to the de-capsulation endpoint. Typically,
routers take a 50% performance penalty in decapsulating a packet.
The attacker could not be certain that the router would be impacted,
and the extremely high volume of traffic needed, would easily be
detected as an anomaly.

If an attacker were able to influence the availability of a link,
they could cause the network to invoke the not-via repair mechanism.
A network protected by not-via IPFRR is less vulnerable to such an
attack than a network that undertook a full convergence in response
to a link up/down event.

Intellectual Property Statement

Full copyright statement

Normative References

   There are no normative references.

Informative References

   Internet-drafts are works in progress available from
   <http://www.ietf.org/internet-drafts/>

   [BFD]   Katz, D., Ward, D., "Bidirectional Forwarding
                  Detection", < draft-ietf-bfd-base-05.txt>,
                  June 2006, (work in progress).

   [GRE]   S. Hanks, T. Li, D. Farinacci, P. Traina.
                  "Generic Routing Encapsulation (GRE)",
                  RFC 1701,. October 1994.

   [IPFRR]   Shand, M., Bryant, S., "IP Fast-reroute
                  Framework",
                  <draft-ietf-rtgwg-ipfrr-framework-06.txt>,
                  October 2006, (work in progress).

   [IPIP]   Perkins, C., "IP encapsulation within IP", RFC
                  2003, October 1996

   [ISPF]   McQuillan, J., I. Richer and E. Rosen, "ARPANET
                  Routing Algorithm Improvements", BBN Technical
                  Report 3803, April 1978.

   [L2TPv3]   J. Lau, Ed., et al., "Layer Two Tunneling
                  Protocol - Version 3 (L2TPv3)", RFC 3931,
                  March 2005.

   [LDP]   Andersson, L., Doolan, P., Feldman, N.,

Fredette, A. and B. Thomas,

                      "LDP Specification", RFC 3036, January 2001.

    [LFA]   A. Atlas, Ed, A. Zinin, Ed, "Basic
                      Specification for IP Fast-Reroute: Loop-free
                      Alternates",
                      <draft-ietf-rtgwg-ipfrr-spec-base-05.txt>,
                      Feb 2006, (work in progress).

    [MPLS-TE]   Ping Pan, et al, "Fast Reroute Extensions to
                      RSVP-TE for LSP Tunnels", RFC 4090, May 2005.

    [NNHL]   Shen, N., et al "Discovering LDP Next-Nexthop
                      Labels",
                      <draft-shen-mpls-ldp-nnhop-label-02.txt>,
                      May 2005, (work in progress)

    [RFC2119]   Bradner, S., "Key words for use in RFCs to
                      Indicate Requirement Levels", RFC 2119
                      (BCP 14), March 1997

Acknowledgements

Authors' Addresses

    Stewart Bryant
    Cisco Systems,
    250, Longwater Avenue,
    Green Park,
    Reading, RG2 6GB,
    United Kingdom.   Email: stbryant@cisco.com

    Stefano Previdi
    Cisco Systems
    Via Del Serafico, 200
    00142 Rome,
    Italy   Email: sprevidi@cisco.com

    Mike Shand
    Cisco Systems,
    250, Longwater Avenue,
    Green Park,
    Reading, RG2 6GB,
    United Kingdom.   Email: mshand@cisco.com