Internet Draft
Expires: May 20002

Alia Atlas (Avici Systems)
Markus Jork (Avici Systems)

November 2001

MPLS RSVP-TE Interoperability for Local Protection/Fast Reroute

draft-atlas-rsvp-local-protect-interop-02.txt

Status of this Memo

This document is an Internet-Draft and is subject to all provisions of Section 10 of RFC2026. Internet-Drafts are working documents of the Internet Engineering Task Force (IETF), its areas, and its working groups. Note that other groups may also distribute working documents as Internet-Drafts.

Internet-Drafts are draft documents valid for a maximum of six months and may be updated, replaced, or obsoleted by other documents at any time. It is inappropriate to use Internet- Drafts as reference material or to cite them other than as "work in progress."

The list of current Internet-Drafts can be accessed at http://www.ietf.org/1id-abstracts.html

The list of Internet-Draft Shadow Directories can be accessed at http://www.ietf.org/shadow.html

Abstract

Our combined draft on fast-reroute, draft-ping-rsvp-fastreroute-00.txt, leaves several areas with future work required. One of those areas is the merging rules for detour LSPs. This draft describes some of the issues with the merging rules presented in draft-ping-rsvp-fastreroute-02.txt and proposes a solution which also enhances interoperability.

The implementation of detour LSPs described in draft-ping-rsvp-fastreroute-00.txt results in intermittent backup availability. A detour LSP will become temporarily unavailable when other detours with different EROs are merged into it; local protection will become temporarily unavailable if re-optimization of a backup path is implemented.

This draft describes examples of the above problems and proposes a solution. The solution also resolves issues with merged backups

which do not provide adequate protection, due to the use of SRLGs. The solution also finds backup paths in some topologies where a detour LSP could not be found, in the merging rules in draft-ping-rsvp-fastreroute-00.txt were followed.

Recommended behavior for supporting a revertive mode for local protection is specified.

Contents

1. Introduction

Our combined draft on fast-reroute, draft-ping-rsvp-fastreroute-00.txt, leaves several areas with future work required. One of those areas is the merging rules for detour LSPs. This draft describes some of the issues with the merging rules presented in draft-ping-rsvp-fastreroute-02.txt and proposes a solution which also enhances interoperability.

The implementation of detour LSPs described in [FAST-REROUTE] results in intermittent backup availability. A detour LSP will become temporarily unavailable when other detours with different EROs are merged into it; local protection will become temporarily unavailable if re-optimization of a backup path is implemented.

This draft describes examples of the above problems and proposes a solution. The solution also resolves issues with merged backups which do not provide adequate protection, due to the use of SRLGs. The solution also finds backup paths in some topologies where a detour LSP could not be found, in the merging rules in [FAST-REROUTE] were followed. Finally, the solution allows make-before-break to work on detour LSPs without causing the protection to become temporarily unavailable.

Recommended behavior for supporting a revertive mode for local protection is specified.

2. Backup Availability

In order for local protection to be useful in mission-critical networks, it is important that local protection, once created at a PLR for a given primary LSP, remains available at all times until a single fault has occurred. That fault could be physical or due to resource preemption and could occur on either the primary LSP or the backup LSP.

The fault against which the backup protects could occur at any time, including when the backup is temporarily unavailable. Similarly, at any time, traffic could be running across a backup - if it becomes temporarily unavailable then, traffic loss will result.

The following example demonstrates a case when following the merging rules given in [FAST-REROUTE] result in the backup becoming temporarily unavailable. In this example, the PLR of the backup which is temporarily unavailable is not aware that the backup is not functional during that period.

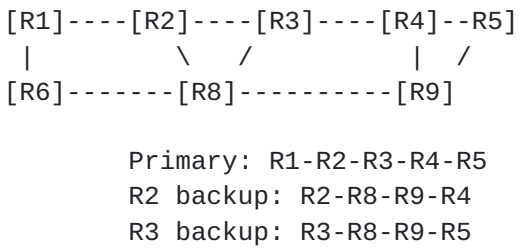


Figure 1: New Detour Causes Existing Detour to Fail

Assume that due to setup timing, R2's backup is created first. Then when the detour from R3 tries to be created, R8 must merge the two detours together. According to the merging rules in [FAST-REROUTE], R8 must select the ERO to R3's backup. When R8 does so, there is a period when the backup from R2 was "up" but isn't actually available unless R8 and R9 deal with changing the ERO in a make-before-break fashion.

The same problem will occur when and if the backup from R3 is torn down. Tearing down R3's backup will make R2's backup temporarily unavailable.

3. Backup Does Not Provide Protection

There are three examples of where the merging rules given in [FAST-REROUTE] result in either a final detour LSP which does not provide protection against failure or in no detour LSP being created.

3.1 Selected ERO Merges at PLR

In the following example, the merging rules in [FAST-REROUTE] require that the shorter ERO be selected.

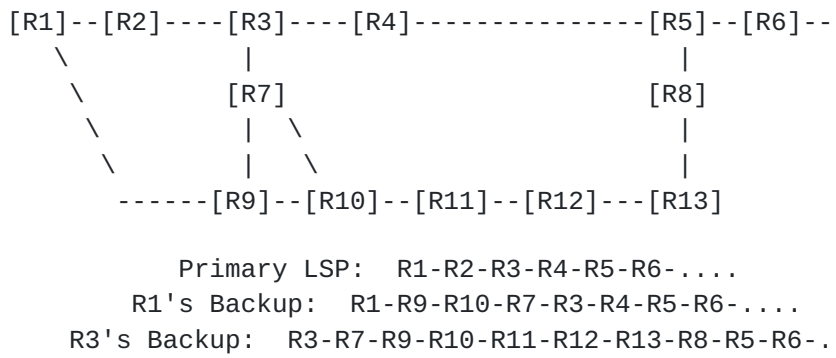


Figure 2: No Protection for PLR Because Merged Detour Ends at PLR

In this example, R9 must merge R1's backup and R3's backup. Because the shorter ERO must be chosen, R9 would select R1's backup's ERO; clearly R3 would not have an effective backup in this case.

This problem could be resolved by adding a merge rule which removes any ERO from consideration which merges with the primary LSP at a node which is the PLR for any detour LSP being merged.

3.2 Selected ERO Uses SRLG

In the following example, the selected ERO uses a link which belongs to an SRLG which one of the merged detour LSPs was avoiding.

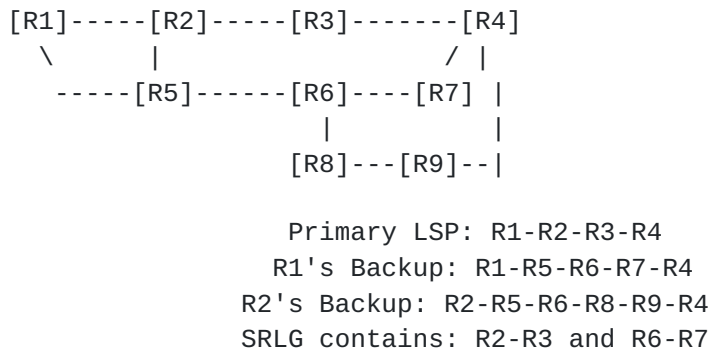
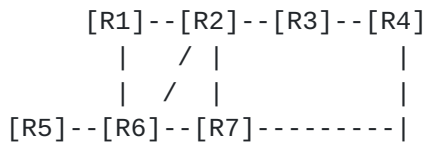


Figure 3: Merged Detour LSP Uses a Link Avoided Due to SRLG

In the above figure, the merging rules in [FAST-REROUTE] mean that R5 will select the ERO associated with R1's backup. Unfortunately, that means that the merged detour LSP will not protect against the failure of R2-R3 because the link R6-R7 is used in the final merged detour LSP and R6-R7 and R2-R3 are in a common SRLG.

3.3 No Backup Path For Detour LSP

The CSPF rules given in [FAST-REROUTE] require that the backup path selected does not cross a link in the same direction as the primary LSP does. This is to avoid merging problems when a detour LSP intersects its primary LSP and yet should not be merged. However, this can result in the lack of protection, due to a lack of paths, as described in the following example.



Primary: R5-R6-R7-R2-R3-R4
R2 Backup: R2-R6-R7-R4
[R2]-[R7] has insufficient bandwidth to support R2's backup

Figure 4: Detour LSP uses same link as primary LSP upstream from PLR

In the above example, there are two possible ways for R2 to have a backup. Either, it can use R2-R6-R7 or it can use R2-R7. In this scenario, the link R2-R7 does not have sufficient free bandwidth to admit the backup LSP.

4. Solution to Problems with Merging Rules

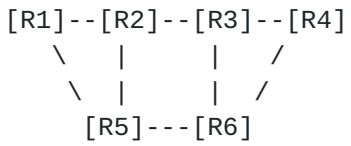
The problems with the merging rules can be solved by using a different SENDER_TEMPLATE for each backup LSP, instead of using the primary's SENDER_TEMPLATE, and by not merging Path messages with different EROs.

First, consider the solution of not merging Path messages with different EROs. This alone appears to resolve all three of the issues described in Section 3. However, simply not merging Path messages with different EROs introduces other problems when all detour LSPs for the same primary LSP have the same SENDER_TEMPLATE.

Consider the third issue described where the backup path must cross and use the same link as the primary LSP. Not merging the detour LSP and the primary LSP leads to the inability to determine whether a PathErr belongs to the Primary LSP or to a Backup LSP through the LSR with the same next hop. Figure 4 demonstrates the problem with this partial solution.

In Figure 4, if the link from R7-R4 breaks, R7 will send a PathErr to R6. If the SENDER_TEMPLATE for the Primary LSP and for R2's Backup LSP were the same, then R6 would be unable to determine whether the PathErr was for the Primary LSP or for R2's Backup LSP. This is the problem with not merging Path messages with different EROs when detour LSPs share the same SENDER_TEMPLATE.

To further elaborate, consider Figure 5 below, when one tries to NOT merge Path messages for detour LSPs with different EROs but with the same next link/hop.



```
Primary    : R1-R5-R6-R4
R1's Backup: R1-R2-R3-R6
R5's Backup: R5-R2-R3-R4
```

Figure 5: OverMerging Backup LSPs

If the SENDER_TEMPLATES for R1's backup and R5's backup are the same, then a RESV received for one would have a FILTER_SPEC which would match both R1's backup and R5's backup. Because the SENDER_TEMPLATEs are the same and the Detour object is not included in the RESV, there is no way at R2 to distinguish a RESV for R1's backup from a RESV for R5's backup.

From the preceeding, there are two alternatives if merging of Path messages with different EROs is not desired (as demonstrated via Figure 1). Either, the Detour object must be included in every Path, Resv, PathErr, ResvErr, PathTear, ResvTear, and ResvConf message which is for the detour LSP or the SENDER_TEMPLATEs for detour LSPs must be different. The former alternative is essentially expanding the SENDER_TEMPLATE to include the Detour object.

The recommendation is to simply use a different SENDER_TEMPLATE. The "IPv4 tunnel sender address" in the SENDER_TEMPLATE MUST contain an IP address of the PLR.

A backup LSP MUST be identified as follows. The SESSION object and the LSP_ID are copied from the primary LSP being protected. The IPv4 tunnel sender address MUST be set to an address of the PLR node. If the head-end of a tunnel is also acting as the PLR, it MUST choose an IP address different from the one used in the SENDER_TEMPLATE of the original LSP tunnel.

4.1 Detour Object Need Not be Understood

An additional benefit of having different SENDER_TEMPLATEs for detour LSPs is that the Detour object need not be rejected by LSRs which do not understand it. Instead, the Detour object can be silently passed along; its use is for managability.

This simplifies the interoperability scenarios between an LSR which implements only the facility backup method given in [FAST-REROUTE] and not the one-to-one detour LSP backup method given in [FAST-REROUTE].

5. Make-Before-Break

Supporting make-before-break for detour LSPs is important if

re-optimization of the backup paths is desired. It guarantees that the local protection, once created, will remain continuously available until a failure occurs. At a PLR, consider a single detour LSP for a given primary LSP. When network adaptivity, configuration, etc. dictate that a different path is preferred for the detour, a new detour LSP must be created. Once that new detour LSP is created, the fast-reroute protection should be moved to the new detour LSP; then the old detour LSP can finally be torn down.

The requirement is for local-protection to ALWAYS be available at a given PLR for a given primary LSP until a single failure occurs. That failure may occur on the backup or on the primary.

When signalling either detour LSP, the LSP ID used (sent in the SENDER_TEMPLATE and the FILTER_SPEC objects) is that of the primary LSP which the backup LSP is protecting. The SESSION object used when signalling the backup LSP is the same as the SESSION object of the primary LSP.

Therefore, the option of using a different LSP-ID, as described in [RSVP-TE] for a regular tunnel, is not available for make-before-break on a backup. As described in [RSVP-TE], when doing a make-before-break on a regular tunnel, the ingress will allocate a new LSP ID to be used when creating a new LSP.

To support make-before-break on a backup, it is necessary to have a way of distinguishing the current backup LSP from the new backup LSP. The content in the SENDER_TEMPLATE (and FILTER_SPEC) are the LSP ID and the "IPv4 (IPv6) tunnel sender address". The former cannot be modified; therefore it is necessary to change the address.

For detour LSPs, the "IPv4 (IPv6) tunnel sender address" will be filled with one of the PLR's address, which is different from that used when the LSR acts as an ingress, rather than a PLR. Additionally, for Detour Backup LSPs, the PLR will put that same address in the DETOUR Object's "Source ID".

An LSR distinguishes a backup LSP from a primary LSP based upon the "IPv4 (IPv6) tunnel sender address" in the SENDER_TEMPLATE (and FILTER_SPEC). Therefore, to signal two different backup LSPs from the same PLR for the same primary LSP, the PLR must use different IP addresses, which are put into the SENDER_TEMPLATE.

To effect a reroute or a bandwidth change on a backup, the PLR picks one of its IP addresses which is different from that used for the current backup LSP and which is different from that used for primary LSP. Then the PLR signals the new backup LSP and proceeds

with a make-before-break as described in [RSVP-TE].

To support make-before-break on backups in this manner, the PLR must have ownership of two distinct IP addresses. If the PLR is also ingress, then it requires a third distinct IP address, which it uses when signalling primary LSPs. Only the router ID needs to be routable. All three addresses MUST be unique within the MPLS domain.

Support of make-before-break on backups MAY be supported.

6. Revertive Behavior: Recovery from Failure

[MPLS-RECOVERY] describes some motivations for supporting revertive behavior. It is necessary to have a method for recovering back to the primary LSP after a failure has recovered. Ideally, as soon as the Ingress learned about the failure, a new primary LSP would have been created and the traffic moved onto it. Practically, it is not required to occur. A recomputation of the primary tunnel's path for a new primary LSP can guarantee the use of a newly available resource.

Revertive behavior MAY be supported. The determination of when a failure is over is as follows:

1) The locally detected failure, if any, has cleared (e.g. interface has come back up),

2) and a RESV message for the primary LSP has been received since the failure occured.

Practically, when a RESV for the primary LSP is received at a PLR and that PLR has local protection in use for that primary LSP, if no local failure is detectable, the PLR may revert to using the primary LSP.

When the RESV for the primary LSP is received, it is highly likely that a different label will be specified in the LABEL Object. This occurs if the downstream neighbor of the PLR has lost all knowledge of the primary LSP. When the PLR receives a different label, it MUST change the primary LSP to using that label without propogating a different label upstream.

Essentially, to revert to the primary LSP, the PLR should:

1) Update the primary LSP's out-segment to use the new label specified,

2) Move the traffic from the backup LSP's out-segment to the primary LSP's out-segment,

3) And clear the "local protection in use" flag from its IPv4 (IPv6) address subobject in the primary LSP's RRO. This change should be propogated back to the ingress as soon as feasible.

6.1 Local Failure

If the failure being recovered from is local, no PATH messages can be sent for the primary LSP until the affected link, etc, has recovered.

If the failure was resource preemption, revertive behavior may not be reasonable. If desired, then if a PATH message for the primary LSP succeeds in getting resources on the primary LSP's out-going interface, then and only then can the PLR forward a PATH message for the primary LSP downstream.

6.2 Remote Failure

If the failure being recovered from is remote, then the PLR may send PATH messages for the primary LSP immediately. However, in response to such a PATH message while the failure is occuring, the PATH message will be met with a PathErr. All such PathErrs must be ignored and not propogated upstream. The PLR is already aware that a failure exists and is repairing around that failure. The PLR should send a PATH message for the primary LSP no more frequently than the normal refresh interval.

7. Security Considerations

These procedures do not change the trust model of RSVP [RFC2205] and [RSVP-TE]. As such no additional security risks are posed.

8. References

[FAST-REROUTE] Pan, P. et al., "Fast Reroute Techniques in RSVP-TE", Internet Draft, draft-ping-rsvp-fastreroute-00.txt, November 2001

[MPLS-RECOVERY] Sharma, V. et al., "Framework for MPLS-based Recovery", Internet Draft, draft-ietf-mpls-recovery-frmwrk-03.txt,

July 2001

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

[RFC2205] Braden, R. et al., "Resource ReSerVation Protocol (RSVP) -- Version 1, Functional Specification", RFC 2205, September 1997.

[RSVP-TE] Awduche, D. et al., "RSVP-TE: Extensions to RSVP for LSP Tunnels", Internet Draft, draft-ietf-mpls-rsvp-lsp-tunnel-08.txt, February 2001.

9. Authors' Addresses

Alia Atlas
Avici Systems
101 Billerica Avenue
N. Billerica, MA 01862
Voice: +1 (978) 964-2070
Email: aatlas@avici.com

Markus Jork
Avici Systems
101 Billerica Avenue
N. Billerica, MA 01862
Voice: +1 (978) 964-2142
Email: mjork@avici.com