Network Working Group
Internet Draft
Expiration Date: May 2001

Robert Goguen
Cisco Systems, Inc.

George Swallow
Cisco Systems, Inc.

November 2000

RSVP Label Allocation for Backup Tunnels

draft-swallow-rsvp-bypass-label-01.txt

Status of this Memo

Abstract

This document describes the use of RSVP, to establish backup tunnels for local repair of LSP tunnels.

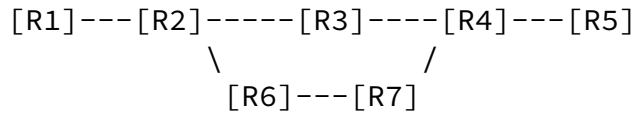
Contents

1. Introduction

This document describes the use of RSVP[RFC2205], to establish backup tunnels for local repair of LSP tunnels. The RSVP extensions for setting up LSP tunnels are described in [2].

In order to meet the needs of realtime applications such as Voice over IP, it is highly desirable to be able to repair LSP tunnels in 10s of milliseconds. We use the term local repair to when referring to techniques which accomplish this. There are two basic strategies for effecting local repair. These are one to one backup and one to many backup.

In the one to one case, a label switched path is established which intersects the original tunnel somewhere downstream of the point of local repair. For each LSP which is backed up, another backup LSP is established.



For example, suppose that in the simple topology above, R1 creates a tunnel to R5 via the path [R1->R2->R3->R4->R5]. R2 can provide local repair by creating a partial backup tunnel [R2->R6->R7->R4] which merges with the original tunnel [R1->R2->R3->R4->R5] at R4.

A second means of backing up LSPs is to take advantage of the label stack. Instead of creating a separate LSP for every backed-up LSP tunnel, a single LSP is created which serves to backup up a set of tunnels. We call such a tunnel a bypass tunnel. The bypass tunnel itself is established just like any other LSP tunnel.

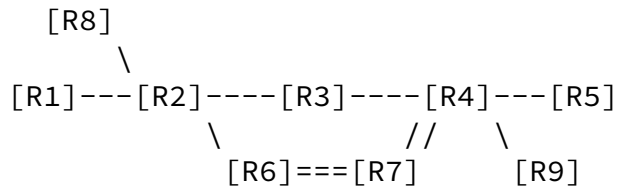
Again the bypass tunnel must intersect the original tunnel(s) somewhere downstream of the point of local repair. This of course implies that the set of tunnels being backed up all pass through some

common downstream node. All tunnels which pass through the point of local repair and through this common node which do not use the facilities being bypassed are candidates for this set of tunnels.

To effect the repair of the backed up tunnels, packets belonging to a repaired tunnel are redirected onto the bypass tunnel. An additional label representing the bypass tunnel is stacked onto the redirected packets. At the penultimate hop of the bypass tunnel, the label for the bypass tunnel is popped off the stack, revealing the label which represents the tunnel being backed up.

Returning to the above example, R2 in this case would build a bypass

tunnel [R2->R6->R7->R4]. The doubled lines represent this tunnel. The backup path for [R1->R2->R3->R4->R5] again rejoins the original path at R4, but its path is now [R1->R2->R4->R5] with the bypass tunnel as the connection between R2 and R4.



The scalability improvement comes in that this bypass tunnel can also be a backup for tunnels from any of R1, or R2, R8 to any of R4, R5, or R9 which traverse the link R2->R3.

2. Labels for backup tunnels

In this section we consider the issues of binding labels to these various backup entities. In this discussion we refer to the point of local repair as PLR or the PLR node depending on context. Similarly, we refer to the point at which the backup tunnel merges back into the original tunnel as the merge point or merge node. Note that it is possible that the merge node is also the tunnel tail.

In MPLS, a node may have a single label space which is global to the node, or it may have multiple label spaces, some of which are specific to particular interfaces.

Whether one to one backup or many to one backup is used, a means for obtaining labels which properly represent the backed up tunnels is required.

The simplest case is when the merge node is the next hop of both the original tunnel and of the backup tunnel and the merge node has a global label space. In the one to one case this situation would arise as follows. There exist multiple links between the local-repair node and the merge node. An LSP tunnel crosses one of these links. A parallel link will be used as the backup should the primary link fail. Since the merge node is using a global label space, it would properly recognize packets which were sent on the parallel link.

In the many to one case, the above situation arises in a more general way. Consider any two neighbor nodes, L and M. If L wishes to backup the tunnels crossing the link to M it could establish a backup tunnel to M via any available route which avoids the backed up link. Again, since M is using a global label space, to effect the bypass, L

would simply do the normal label swap, and then push on the label for the backup tunnel. The penultimate hop of the backup tunnel would pop the label for the backup tunnel, revealing the label which M expects.

When a failure occurs and the backup comes into use, a means of maintaining the RSVP state is required. In this simple case the Path messages can simply be sent via the backup link or tunnel as the case may be.

Except for these specific cases, additional labels must be assigned and a means of signaling is required.

Returning to the bypass tunnel example above, if R4 does not use a global label space then a label must be assigned out of the label space used by R4 on the interface terminating the link from R7. if, however, R4 has a global label space then the label which R2 uses during the backup could be the same label value which is used by R3.

In the next section we describe a general means of obtaining backup labels. In the following section we describe an efficient means communicating labels where global label spaces are used.

3. Signaling for backup tunnels

A number of objectives must be met to obtain a satisfactory signaling solution. These are summarized as follows:

1. Unambiguously and uniquely identify backup tunnels
2. Unambiguously associate primary tunnels with their backup tunnels
3. Work with both global and non-global label spaces
4. Allow for merging of backup tunnels
5. Maintain RSVP state during and after fail-over.

3.1. Identification and association of backup LSP tunnels

LSP tunnels are identified by a combination of the SESSION and SENDER_TEMPLATE objects. The relevant fields are:

IPv4 tunnel end point address

IPv4 address of the egress node for the tunnel.

Tunnel ID

A 16-bit identifier used in the SESSION that remains constant over the life of the tunnel.

Extended Tunnel ID

A 32-bit identifier used in the SESSION that remains constant over the life of the tunnel. Normally set to all zeros. Ingress nodes that wish to narrow the scope of a SESSION to the ingress-egress pair may place their IPv4 address here as a globally unique identifier.

IPv4 tunnel sender address

IPv4 address for a sender node

LSP ID

A 16-bit identifier used in the SENDER_TEMPLATE and the FILTER_SPEC that can be changed to allow a sender to share resources with itself.

The first three of these are in the SESSION object and are the basic identification of the tunnel. The last two are in the SENDER_TEMPLATE.

The LSP_ID is used to differentiate multiple LSPs during a tunnel reroute procedure. During this procedure, multiple LSPs each with their own LSP_ID may be active simultaneously. It is quite possible that a node which is downstream of the point of local repair on the LSP being backed up is also upstream of the point of local repair for some other LSP associated with the tunnel. It is thus necessary to properly associate the LSP_ID of a backup tunnel with the LSP_ID of the tunnel being backed up.

We therefore propose that backup tunnels be identified as follows. The SESSION object and the LSP_ID are copied from the LSP tunnel being backed up. The IPv4 tunnel sender address is 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.

3.2. ERO contents

The PLR creates an ERO for the backup LSP tunnel. It does this by operating on the original ERO or on the contents of a received RRO. Abstract nodes which precede the merge point are removed. In the case of signaling over a bypass tunnel, the procedure is now complete. In the one-to-one backup case the, an explicit route from the PLR node to the merge node is prepended to the ERO.

4. Efficient signaling for the global label case

When global labels are in use and bypass tunnels are used as the means of local repair, a more efficient means of backup label distribution can be employed. When using a bypass tunnel, the backed up tunnels are only one LSP hop from rejoining their primary LSPs. If the merge node is using a global label space, then the label necessary for the back up tunnel is the label already assigned by the merge node.

By using the label record option of the ROUTE_RECORD procedure, downstream labels can be known by upstream nodes without further protocol interactions. In this case no Path message is sent down the bypass tunnel prior to a failure. When a failure occurs, the PLR updates the ERO as above and begins refreshing the LSP down the bypass tunnel. This ensures that the LSP state is refreshed in the merge node and nodes further downstream.

5. Notification of local repair

In many situations, the route used during a local repair will be less than optimal. The point of the local repair is to keep high priority and loss sensitive traffic flowing while a more optimal rerouting of the tunnel can be effected by the head-end of the tunnel. Thus the head-end needs to know of the failure.

To provide this notification, the point of local repair SHOULD send a PathErr message with error code of "Notify" and an error value of "Notification of local repair".

6. Security Considerations

These procedures do not change the trust model of RSVP [RFC2205] and draft-ietf-mpls-rsvp-tunnel-07.txt[RSVP-TE]. As such no additional security risks are posed.

7. Intellectual Property Considerations

Cisco Systems may have intellectual property rights claimed in regard to some or all of the specification contained in this document.

8. References

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

[RSVP-TE] Awduche, D. et al., "Extensions to RSVP for LSP Tunnels", Internet Draft, draft-ietf-mpls-rsvp-lsp-tunnel-07.txt, August, 2000.

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

9. Authors' Addresses

Robert Goguen
Cisco Systems, Inc.
250 Apollo Drive
Chelmsford, MA 01824
Voice: +1 978 244 8095
Email: rgoguen@cisco.com

George Swallow
Cisco Systems, Inc.
250 Apollo Drive
Chelmsford, MA 01824
Voice: +1 978 244 8143
Email: swallow@cisco.com