Soft State Switching
A Proposal to Extend RSVP for Switching RSVP Flows

<draft-viswanathan-mpls-rsvp-00.txt>

Status of This Memo

Abstract

This memo describes a mechanism for establishing a switched path with guaranteed Quality of Service for RSVP [1] flows in a MultiProtocol Label Switching (MPLS) environment. It proposes an extension to the RSVP protocol that allows the establishment of a sequence of label switched hops along the hop-by-hop routed path by enabling adjacent nodes to exchange MPLS labels [11]. The labels may correspond to information that identifies a layer 2 virtual connection; for example, the VPI/VCI value in the case of an ATM-based infrastructure.

1. Introduction

A Label Switching Router (LSR) is a label switching node that has an IP Control Point (IP-CP) and implements an IP label switching technology [2-4]. LSRs form adjacencies using a well-known label switched path (LSP), also called the default path, that terminates at the adjacent LSR's IP-CP. This hop-by-hop LSP connectivity gives a network of LSRs the same nature as any ubiquitous IP internet. The objective is to label switch RSVP flows in such an environment.

This document proposes an extension to RSVP that introduces new objects to the existing RSVP messages. Using these objects, each downstream LSR provides its neighboring upstream LSR with the label on which it wishes to receive a RSVP flow. In an ATM-based LSR environment, this label would correspond to a VPI/VCI value for the ATM virtual circuit on which the LSR wishes to receive traffic from the RSVP flow. Then, using an approach similar to those outlined in [2], [3], and [4], the labels are spliced hop-by-hop to form an ingress-to-egress LSP. The data from the RSVP flow then traverses this LSP, and the RSVP signaling messages are forwarded hop-by-hop via default paths. By moving RSVP flows from the hop-by-hop routed path to a dedicated ingress-to-egress LSP, it is possible to leverage the QoS capabilities of the underlying switching technology to provide the type of service desired for the reserved flow.

The memo proposes a "one label per flow" approach, where a flow is synonymous with a particular sender (source address/source port) and session (destination address/protocol/destination port). It is assumed here that the LSRs on the edge of a MPLS network can either auto-learn or are configured to indicate that they are edge LSRs (on a per interface basis).

2. Soft State Switching

In soft state switching, the goal is to switch packets from a RSVP flow at layer 2 instead of having to forward them hop-by-hop as in conventional IP routers. By doing so, it is possible to leverage the high-performance switching and Quality of Service capabilities of the layer 2 technology. This is achieved when all neighboring LSRs along the routed path can exchange labels for establishing the switched path for RSVP flows. Then, the labels may be "spliced" hop-by-hop to set up an end-to-end (ingress-to-egress) LSP along the preferred routed path. By splicing, we refer to the process by which an incoming label is associated with an outgoing label at layer 2, without traffic encapsulated by the incoming label being processed at the network layer. For example, this can be achieved in ATM switches by establishing this association in the ATM switching tables. Once

the splicing is complete, the default path carrying best effort traffic between adjacent LSRs provides the IP forwarding path. The RSVP signaling messages are forwarded on the default path.

The labels are assumed to have only unidirectional significance. In other words, there exists a separate label space for each direction of flow on a link. Moreover, the downstream LSR is chosen to be the label space owner (allocator) on a link. The single owner approach keeps the label usage simple and manageable. If a label space had more than one owner, it would require that the owners synchronize their use of the labels or the space would have to be partitioned amongst the owners. For flexibility, the proposed extension to RSVP also supports the concept of "upstream on demand" allocation as described in [3]. In this method, the upstream LSR allocates labels when demanded by a downstream LSR. This enables co-existence with other protocols that consume labels.

3. Motivation

In this section, we discuss why the RSVP protocol is ideal for establishing a label switched path for reserved flows.

One motivating factor for using RSVP is that mapping the network-layer QoS request to a layer 2 virtual connection is simple. The RESV message carries the QoS requested by the receiver(s) of the RSVP flow. For example, this could correspond to one of the Integrated Service classes described in [6-8]. This QoS information is needed when layer 2 labels are set up and spliced; i.e., when the resource reservations are made. Otherwise, the LSP establishment protocol would have to carry its own QoS entity and/or map the label setup to RSVP tables at each LSR hop.

Another motivating reason for extending RSVP is multicast support. RSVP is designed to scale well for multicast sessions requiring resource reservation. RSVP also allows receivers to join existing sessions with different QoS requirements. An independent LSP establishment protocol should be able to handle such session "joins" equally well.

With the RSVP protocol the receivers can make sender selection through the provision of different filter styles. In this, multiple sender flows (as chosen by the receivers) in a RSVP session can be associated with a single reservation. In other words, sender flows in a RSVP session can be merged into a single downstream reservation. A new LSP establishment protocol would have to support a similar mechanism for seamless interoperability with the RSVP protocol.

Finally, any mechanism for setup of LSPs would, in any case, require extensive interfacing with the RSVP protocol and/or its state tables.

Due to these reasons, it is best if RSVP can be extended without changing its existing mechanics, to provide support for setting up the switched path for RSVP flows. This need not be viewed as "piggy-backing" another protocol on RSVP, but rather, a natural extension to RSVP to provide QoS in a MPLS environment.

4. L2 Label Exchange Mechanism

The proposed extension to RSVP calls for adding a new object to carry MPLS label information within RESV, PATH, and RESVERR messages. The egress LSR, say LSR A, (i.e. the "last" node in the MPLS environment, or the LSR through which the RSVP flow exits the MPLS environment) will place this object in any RESV message that it sends to the PHOP LSR for a flow (as stored in the Path state for this flow) -- call this LSR B. The RESV message is sent to LSR B via the default routed path.

If LSR B rejects the reservation (i.e., if the reservation is rejected by either policy or admission control, or due to an error), it then forwards a RESVERR message with the appropriate error code to LSR A. The RESVERR message includes the MPLS label object received from LSR A (RESVERR nack). Receipt of the RESVERR nack indicates that the upstream LSR will not forward the reserved flow on the requested LSP. In the event that this occurs, LSR A may choose to release its reservation or it may choose to classify and forward packets received on the default path from LSR B at the network layer.

If LSR B accepts the reservation, it will use the label in the RESV message to setup a LSP to LSR A (in this case, the egress LSR) on the outgoing interface. The QoS for this LSP corresponds to a mapping of the Integrated Service class specified in the RESV message to an appropriate set of QoS values for the layer 2 technology. LSR B will forward a PATH message for the reserved flow to LSR A which includes the MPLS label object allocated by LSR A (PATH ack). This MPLS label object will also be included in all subsequent PATH messages for the reserved flow sent to LSR A while the reservation remains in place.

LSR B will then choose a new label on the incoming interface through which the RSVP flow enters the LSR, and send this label to its own PHOP, LSR C, by passing the new MPLS label object in a RESV message. LSR B may optimistically choose to splice the label on the incoming interface from LSR C to the label on the outgoing interface to LSR A by modifying its layer 2 label swap table, or it may choose to wait for the receipt of a PATH ack from LSR C. If LSR C accepts the

reservation then it will forward a PATH ack to LSR B. If LSR C rejects the reservation, it will then send a RESVERR nack to LSR B. LSR B has the option of releasing its reservation (by transmitting a RESVERR nack downstream to LSR A) or of classifying the packets of the reserved flow on the default path from LSR C and forwarding them on the previously established QoS LSP to LSR A, while sending a RESVERR message without the label object to LSR A.

In the event of success at each PHOP LSR, the RESV will eventually reach the ingress LSR (the LSR through which the RSVP flow enters the MPLS environment). The ingress LSR will make necessary classifier entries to forward packets for this flow through the LSP identified by the label in the RESV message received from downstream. An ingress LSR will delete the MPLS label object before forwarding a RESV message to any of its PHOP nodes. The labels used for a RSVP reservation are released whenever the RSVP reservation is torn down or is timed-out.

Using this process, an end-to-end switched path is established for an RSVP flow through a MPLS network. The data packets from the RSVP flow are forwarded via this switched path, while RSVP control messages continue to use the default paths between LSRs.

5. Partial QoS Paths

The procedure described in Section 4 must be clarified in the event that the reserved traffic from a sender (source address/port) is transported initially across a LSP from the ingress to the egress LSR that has been established by an IP switching protocol [2-4, 9]. In this case, the best-effort packets are not forwarded along the hop-by-hop default path and processed at the network layer within each intermediate LSR, but are instead forwarded along a series of spliced label switched hops, and hence are not normally available for packet classification. If a reservation should succeed all the way back to the ingress LSR for a reserved flow, that LSR will classify the packets from the flow and move them onto the new ingress-to-egress QoS LSP.

However, if the reservation succeeds on some of the LSRs on the reverse path from the egress but not all the way back to the ingress, then QoS for the flow cannot be achieved on the path through the LSRs which accepted the reservation unless the farthest upstream LSR which accepted the reservation unsplices the best-effort LSP, classifies the packets of the reserved flow, and forwards them on the QoS LSP to the egress LSR. Note that the default behavior of RSVP is to allow partial QoS paths from the receiver back towards the sender by allowing reservations which have succeeded at a node to remain in

place in the event that the reservation fails further upstream.

Because it is likely that some LSRs will lack sufficient network-layer forwarding capability to unsplice and route many best-effort LSPs simultaneously, the behavior of a LSR which has accepted a reservation, established a QoS LSP on the appropriate downstream interface(s), but subsequently receives a RESVERR nack from upstream should be configurable. In the event that the LSR chooses to classify the reserved flow at the network layer by unsplicing the best-effort LSP, there are no required changes to the protocol exchange described in Section 4. However, if the LSR chooses to release the reservation, then it should transmit a RESVERR nack downstream and establish blockade state for the reservation. Subsequent reservations for the flow with an equal or greater flowspec should be rejected and blockaded until the blockade timer expires. This prevents the establishment of a potentially unused QoS LSP through the LSR until the blockade timer for the reservation expires. Reservations for the flow with a strictly smaller flowspec can be accepted and propagated upstream. Receipt of a RESVERR nack should be taken as definitive, even if it immediately follows (or precedes) a PATH ack.

Another alternative is to continue to propagate RESV messages and labels all the way to the ingress LSR, with an indication that the reservation has failed somewhere downstream, and that QoS need not be provided for the upstream segments of the LSP. These RESV messages would terminate at the ingress LSR without generating a RESVERR message on any node upstream of the reservation failure. This approach would entail modifications to the RSVP message processing rules.

6. Merging

RSVP scales by merging reservation requests as they propagate upstream towards senders, and by merging QoS handling state as the data flows propagate downstream towards the receivers. The ability to perform merging in a LSR environment is dependent on the switching capabilities of the LSRs.

There are several switching technologies available today (ATM, Frame Relay etc.) and perhaps more in the future. Moreover, the capabilities of a switch of a certain technology vary from vendor to vendor. Three basic characteristics are identified that determine how the underlying switching technology can be used in conjunction with this proposal to address merging of flows under the appropriate environment. They are:

- o Attribute A: Can correctly merge several upstream LSPs into a single downstream LSP ("VC merge"). Frame switches are typically able to do this in a straightforward manner. However, for ATM switches without appropriate functionality built in, cells from different AAL SDUs may become interleaved on the outgoing VC (LSP), thus corrupting the higher-layer information.

- o Attribute B: Can treat a set of labels as a single entity for QoS purposes. A switch with this property is able to treat all traffic from a set of labels in a like manner for purposes of scheduling, fair queueing etc. For example, an ATM switch that performs per-class queueing would assign all the VCs from a given set to a particular class. Then, cells from all the VCs in the sets would receive the QoS corresponding to that class.

- o Attribute C: Can demultiplex senders flows in a single LSP into a separate LSP for a sender. For example, using the label stack for L2 tunneling [3,4].

One logical candidate for flow merging would be support for shared explicit and wildcard reservations, where resources are shared among a set of multiple senders. The difficulty this poses is the potential need to demultiplex senders from the merged flow for downstream receivers which have made reservations for only a subset of the senders, as described in [10]. Merging of multiple sender LSPs into a single LSP (Attribute A) requires support for Attribute C in the LSRs to permit sender demultiplexing. Support for Attribute B permits LSRs to share QoS resources among a group of per-sender LSPs while still facilitating sender demultiplexing.

7. Multicast Support

7.1 Packet Replication

In order to support multicast sessions, at split points within the MPLS network, where data from upstream LSRs splits into multiple downstream flows, the LSR can perform the required duplication (at layer 2) of packets by utilizing the hardware multicast capability (for example, point-to-multipoint VC) of the switch, if available. Otherwise, the flow has to be processed at the network layer and multicast in the normal manner. Note that network layer forwarding is interoperable with all switch types.

7.2 Packet Duplication

In configurations where a per-source or shared multicast tree is

mapped to a point-to-multipoint LSP rooted at an ingress LSR and terminating at each egress LSR with one or more downstream receivers, packet duplication can occur if receivers make a reservation for a particular flow initially being carried on the multicast LSP. This occurs because the flow's packets are carried on both the best-effort and QoS LSP, which are delivered to each egress LSR on the multicast tree. This problem can be avoided if the packets of the reserved flow are removed from the best-effort multicast LSP and carried only on the QoS LSP.

7.3 Unreserved Receivers

When none of the receivers have made a reservation, the multicast session may flow through the default multicast LSP as best-effort traffic. But as soon as a receiver makes a reservation, and packets from the reserved flow are removed from the best-effort LSP, the data flow may stop to receivers that have not made a reservation. The receivers without a reservation only get PATH messages but no data (even at best-effort). This problem can be addressed in several different ways determined by the switch architecture.

This problem can be avoided for switches that support Attribute A. They can add the default best-effort LSP for the (source/)group as a branch in the point-to-multipoint per-flow QoS LSP by merging the QoS LSP back onto the best-effort LSP on those branches of the tree where there are no downstream receivers. If the switch architecture allows adding the local IP-CP to the point-to-multipoint QoS LSP, then the IP-CP can multicast the packets only to those interfaces from which there is no reservation but which are listed in the multicast table.

If the switch architecture does not support Attribute A, and can not efficiently perform the multicast forwarding in the IP-CP, then one approach is to build the per-flow QoS LSP to all egress LSRs on the multicast tree (whether they forwarded a RESV or not). The QoS on each branch of this point-to-multipoint LSP would be configured based on the amount of resources reserved on that branch. For best-effort branches, a UBR-like QoS would be used. The LSP construction could be performed under the control of the ingress LSR rooting the multicast tree. Another way to construct the LSP is to use a PATH message to perform the LSP establishment from the node downstream of which there are interfaces through which no reservation has been received. This would be initiated whenever there is at least one reservation in place at the node for the RSVP flow. This may not work in environments where upstream label allocation is not permitted.

7.4 Shared Media Label Allocation

This memo describes a RSVP extension for the MPLS environment where the downstream LSR is the label space owner. As discussed in [5] and [10], this can lead to an allocation deadlock if the downstream receivers on a shared media subnet cannot agree on the value for the label. One approach suggested in [10] is to permit a receiver to suggest a label by passing one upstream in a RESV message, but to allow the upstream node to select the definitive label and pass it downstream within a PATH ack.

Another alternative is to support upstream on demand allocation. In this case, a receiver forwards a RESV message using a NULL MPLS label object to indicate a request for label allocation. The upstream LSR will respond with a label for the RSVP flow in the PATH message to the downstream neighbors. The downstream receivers are responsible for using the label selected by the upstream node, and should include this label in all subsequent RESV messages. In the event that the label selected upstream is out-of-range for a particular receiver, then the receiver can forward a new RESV message with a NULL MPLS label object to trigger a new label allocation. Note that a PATHERR message is not suitable for communicating this error since it propagates all the way back to the sender.

The flexibility of upstream on demand label allocation is also useful in non-shared media environments as it allows co-existence with other IP switching protocols.

8. TTL Decrement

When IP packets flow through a switched path, the TTL value in the IP header cannot be decremented. The decrementing of the TTL value is used to delete packets in a routing loop to avoid/reduce congestion. For this purpose, the proposed LSR Hop Count Object carries a hop-count that counts the number of consecutive LSR hops. The LSRs increment the hop-count only if there is a switched path for that sender flow through that LSR. All LSRs maintain the hop count in the Path state. Only the egress LSR on which the LSP terminates would use the count to decrement the TTL on packets for that sender flow. The LSRs of a switching technology that have a TTL equivalent in the layer 2 header may choose not to use the LSR Hop Count Object.

9. Adjacency

LSR neighbors need some mechanism to establish adjacencies. This is required because the neighbors need to exchange the label range for

correct label allocation. They also need to elect the label allocator. The current version of this memo does not propose any extension to the RSVP protocol for this mechanism. It is assumed that adjacency would be established by another protocol (as proposed in [2], [3] or [4]) and such information would be made available to the RSVP module. In the absence of such a mechanism the LSRs would have to be configured with the required information to operate as described in this memo.

10. Object Formats

This section describes the object formats for the proposed extension. The label objects for ATM LSRs are defined below. Label formats for additional link-layer media will be proposed in a future revision of this memo.

o LSR HOP COUNT object: Class = x, C-Type = 1

```
          +-------------+-------------+-------------+-------------+
          |  Hop Count  |                Reserved                 |
          +-------------+-------------+-------------+-------------+
```

Hop Count
     Counts the length (in LSR hops) of the switched path.

o NULL Label Object: Class = y, C-Type = 1

o ATM RESV Label object: Class = y, C-Type = 2

```
          +-------------+-------------+-------------+-------------+
          |               IPv4 SrcAddress (4 bytes)               |
          +-------------+-------------+-------------+-------------+
          |    //////   |    //////   |          SrcPort          |
          +-------+-----+-------------+-------------+-------------+
          | Flags |       VPI         |            VCI            |
          +-------+-------------------+-------------+-------------+
          //                                                     //
          +-------------+-------------+-------------+-------------+
          |               IPv4 SrcAddress (4 bytes)               |
          +-------------+-------------+-------------+-------------+
          |    //////   |    //////   |          SrcPort          |
          +-------+-----+-------------+-------------+-------------+
          | Flags |       VPI         |            VCI            |
          +-------+-------------------+-------------+-------------+
```

IPv4 SrcAddress
     IPv4 address of the sender.

Flags - 4 bits
0x01 - Implies that the reservation is not in place in the node forwarding the RESVERR message and that the reserved traffic is not being forwarded via the VC (RESVERR nack).

VPI - 12 bits
Virtual Path Identifier. If less than 12 bits are significant, then it is right justified in this field.

VCI - 16 bits
Virtual Circuit Identifier. If less than 16 bits are significant, then it is right justified in this field.

o ATM PATH Label object: Class = y, C-Type = 3

```
+-------+-------------------+-------------+-------------+
| Flags |       VPI         |            VCI            |
+-------+-------------------+-------------+-------------+
```

Flags - 4 bits
0x01 - Implies that the PATH message is in response to an upstream on demand label allocation and may not be propagated any further.

0x02 - Implies that the PATH message is in response to a RESV message carrying a RESV Label object (PATH ack) and may not be propagated any further.

VPI - 12 bits
Virtual Path Identifier. If less than 12 bits are significant, then it is right justified in this field.

VCI - 16 bits
Virtual Circuit Identifier. If less than 16 bits are significant, then it is right justified in this field.

The IPv6 extension and error codes will be defined in a later revision of this memo.

The reader may have noticed that the new ATM RESV Label object has duplicated information already present in the FILTER_SPEC object. Another approach could be to extend the FILTER_SPEC object definition to carry the link-layer labels or insert the label object following the FILTER_SPEC object.

11. Security Considerations

Security considerations are not discussed in this memo.

12. Acknowledgements

The authors wish to acknowledge Shailendra Bhatnagar, Nancy Feldman, Liang Li, Steve Nadas, and Bruce Sinclair for their input.

13. References

[1] R. Braden, L. Zhang, S. Berson, S. Herzog, S. Jamin, Resource ReSerVation Protocol (RSVP) -- Version 1 Functional Specification. Internet Draft, draft-ietf-rsvp-spec-16, June 1997.

[2] P. Newman, W. L. Edwards, R. Hinden, E. Hoffman, F. Ching Liaw, T. Lyon, G. Minshall, Ipsilon Flow Management Protocol Specification for IPv4, Version 1.0. Internet RFC 1953, May 1996.

[3] Y. Rekhter, B. Davie, D. Katz, E. Rosen, G. Swallow, D. Farinacci, Tag Switching Architecture - Overview. Internet Draft, draft-rekhter-tagswitch-arch-00.txt, January 1997.

[4] A. Viswanathan, N. Feldman, R. Boivie, R. Woundy, ARIS: Aggregated Route-Based IP Switching. Internet Draft, draft-viswanathan-aris-overview-00.txt, March 1997.

[5] D. Farinacci, Partitioning Tag Space among Multicast Routers on a Common Subnet. Internet Draft, draft-farinacci-multicast-tag-part-00.txt, December 1996.

[6] S. Shenker, C. Partridge, R. Guerin, Specification of Guaranteed Quality of Service. Internet Draft, draft-ietf-intserv-guaranteed-svc-08.txt, February 1997.

[7] J. Wroclawski, Specification of the Controlled-Load Network Element Service. Internet Draft, draft-ietf-intserv-ctrl-load-svc-05.txt, May 1997.

[8] F. Baker, R. Guerin, D. Kandlur, Specification of Committed Rate Quality of Service. Internet Draft, draft-ietf-intserv-commit-rate-svc-00.txt, June 1996.

[9] K. Nagami, Y. Katsube, Y. Shobatake, A. Mogi, S. Matsuzawa,

T. Jinmei, H. Esaki, Flow Attribute Notification Protocol (FANP) Specification, Internet Draft, draft-rfced-info-nagami-00.txt, February 1997.

[10] B. Davie, Y. Rekhter, E. Rosen, Use of Label Switching With RSVP, Internet Draft, draft-davie-mpls-rsvp-00.txt, May 1997.

[11] R. Callon, P. Doolan, N. Feldman, A. Fredette, G. Swallow, A. Viswanathan, A Framework for Multiprotocol Label Switching, Internet Draft, draft-ietf-mpls-framework-00.txt, May 1997.

Author's Address

Arun Viswanathan
IBM Corporation
17 Skyline Drive
Hawthorne, NY 10532
Phone: +1 (914) 784-3273
Email: arunv@vnet.ibm.com

Vijay Srinivasan
IBM Corporation
PO Box 12195
Research Triangle Park, NC 27709
Phone: +1 (919) 254-2730
Email: vijay@raleigh.ibm.com

Steven Blake
IBM Corporation
PO Box 12195
Research Triangle Park, NC 27709
Phone: +1 (919) 254-2030
Email: slblake@raleigh.ibm.com