

Internet Engineering Task Force  
Internet-Draft  
Expires: March, 2005

J. Manner  
T. Suihko  
M. Kojo  
M. Liljeberg  
K. Raatikainen  
September, 2004

**Localized RSVP**  
**<[draft-manner-lrsvp-04.txt](#)>**

Status of this Memo

By submitting this Internet-Draft, I certify that any applicable patent or other IPR claims of which I am aware have been disclosed, or will be disclosed, and any of which I become aware will be disclosed, in accordance with [RFC 3668](#).

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/lid-abstracts.html>

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

This document is a submission to the Transport Area Working Group of the Internet Engineering Task Force (IETF). Comments should be submitted to the [tsvwg@ietf.org](mailto:tsvwg@ietf.org) mailing list.

Abstract

Guaranteed QoS for multimedia applications is based on reserved resources in each node on the end-to-end path. Even if the correspondent node does not support QoS, it would be useful to be able to reserve at least local resources at the access network, especially wireless link resources. Additionally, for mobile nodes the continuity of QoS is disturbed due to end-to-end signaling or slow re-reservations of resources after each handover. This draft proposes a simple enhancement to RSVP, so that initial resource reservations and re-reservations due to host mobility can be done locally in an access network.

## Changes since -03

- Updated boilerplates
- Removed discussion on using the CAP Object (not really an option)
- Removed discussion and suggestion on using anycast or multicast addresses when sending the initial Path Request. This is more like a hack and trick, and not really good and clear engineering.
- Updated references and author information

## Changes since -02

- Clarified parts of the text, e.g. [Section 2.4](#)

## Changes since -01

- Some clarifications and editorial changes

## Table of Contents

<a href="#">1</a>	Introduction .....	<a href="#">2</a>
<a href="#">1.1</a>	Related work .....	<a href="#">3</a>
<a href="#">2</a>	Local QoS Support .....	<a href="#">4</a>
<a href="#">2.1</a>	Upstream transfers .....	<a href="#">5</a>
<a href="#">2.2</a>	Downstream transfers .....	<a href="#">6</a>
<a href="#">2.3</a>	Additional functionality .....	<a href="#">7</a>
<a href="#">2.4</a>	Addressing Issues .....	<a href="#">8</a>
<a href="#">2.5</a>	Interworking Issues .....	<a href="#">9</a>
<a href="#">3</a>	Fast local repair for mobile nodes .....	<a href="#">11</a>
<a href="#">4</a>	Security Consideration .....	<a href="#">13</a>
<a href="#">5</a>	IANA Considerations .....	<a href="#">14</a>
<a href="#">6</a>	Summary .....	<a href="#">14</a>
<a href="#">7</a>	Normative References .....	<a href="#">15</a>
<a href="#">8</a>	Non-Normative References .....	<a href="#">15</a>
<a href="#">9</a>	Authors' Addresses .....	<a href="#">16</a>

**[1. Introduction](#)**

Future mobile access networks will be based on IP technology. This implies that, on the network layer, all traffic, both traditional data and streamed data like audio or video, is transmitted as packets. Increasingly popular multimedia applications would benefit from better than best-effort service from the network, a forwarding service with strict Quality of Service (QoS) with guaranteed minimum bandwidth and bounded delay. Other applications, such as electronic commerce, network control and management, and remote login applications, would also benefit from a differentiated treatment.

The IETF has two main models for providing differentiated treatment of packets in routers. The Integrated Services (IntServ) model [\[1\]](#) together with the Resource Reservation Protocol (RSVP) [\[2\]](#) [\[3\]](#) provides per-flow guaranteed end-to-end transmission service. The Differentiated Services (DiffServ) framework [\[4\]](#) provides non-sigaled flow differentiation that usually provides, but does not guarantee, proper transmission service.



However, these architectures have weaknesses, for example, RSVP requires support from both communication end points and from the intermediate routers, and the protocol has performance issues in mobile environments. DiffServ can only provide statistical guarantees and is not well suited for dynamic environments. The Internet Architecture Board has outlined additional issues related to these two architectures [5].

Let us consider a scenario, where a fixed network correspondent node (CN) would be sending a multimedia stream to an end host behind a wireless link. If the correspondent node does not support RSVP it cannot signal its traffic characteristics to the network and request specific forwarding services. Likewise, if the correspondent node is not able to mark its traffic with a proper DiffServ Code Point (DSCP) to trigger service differentiation, the multimedia stream will get only best-effort service which may result in poor visual and audio quality in the receiving application. Even if the connecting wired network is over-provisioned, an end host would still benefit from local resource reservations, especially in wireless access networks, where the bottleneck resource is most probably the wireless link.

We propose a slight modification to RSVP that allows distinguishing local network reservations from the end-to-end reservations. The end host does not need to know the access network topology or the nodes that will reserve the local resources. The reservation message itself identifies the intention and the access network will find the correct network node(s) to respond to the reservation. Note that the scheme is not tied to only mobile networks but can be used in any access network that needs flexible local resource allocations. The first sketch of this solution appeared in [10] and [11], although some implementation ideas have changed since.

The localized RSVP signaling would fit well, for example, with a SIP session, where a call set up can have a pre-condition: if the request for local resources is successful, the end-host can send the COMET message and make the call "ring" [15]. Both ends would use their own local reservations.

All mobility-related terminology in this document are taken from [16]. Therefore, the commonly used term 'access router' (AR) means the 'default router'.

### **1.1. Related work**

Currently we can identify two ways to signal QoS requirements to an access network: DiffServ Code Points (DSCP) and RSVP with IntServ. In the DiffServ-based solution an end host can mark the upstream packets with proper DSCP values, but for the downstream the gateway on the edge of the access network must be able to identify and handle the preferred streams. This can be accomplished with default values for different micro flows in the Service Level Agreement negotiated between the client and the access network service provider.



An alternative way to make use of DiffServ could be through a Bandwidth Broker [6] [7] [8] that dynamically returns the proper code point for each flow: when the first packet of a flow arrives at the access network edge router, it requests the proper code point from the Bandwidth Broker. The router maintains a mapping from micro flow identification to the DiffServ code point (soft state) so that future packets can be directly identified and labeled. Still, this would require a protocol that the end host could use to dynamically adjust the mapping information stored at the access network edge for the incoming traffic.

RSVP can provide the signaling mechanism for QoS requirements to the access network. For upstream reservations, the end host would send the Path message to the access network edge router, which would return the Resv message and set up the reservations. The edge router would act as an RSVP proxy [9]. However, the reservation in the downlink direction is not as straightforward since the downlink reservation needs to be initiated by the RSVP proxy. We would need a way to trigger the proxy to initiate the RSVP signaling for a downlink flow.

Thus, these mechanisms do not seem to solve the problem entirely. The DiffServ mechanisms cannot provide explicit resource reservations. The problem with the RSVP proxy approach is that the proxy cannot automatically distinguish reservations that would be answered by the correspondent node and reservations that would require interception. Additionally, the RSVP proxy needs a way to know when to allocate resources for incoming flows. Mobile access networks also add to the problems, since mobile nodes can frequently change their point of attachment in the network and resource allocations need to be re-arranged, including changing the serving RSVP proxy.

## **2. Local QoS Support**

The usual signaling model of RSVP includes the data sender and receiver, and a network of RSVP routers. The data sender initiates the RSVP signaling by sending the Path message. This message is routed through the network, setting states in RSVP routers, and finally arriving at the data receiver. The receiver then responds to the signaling by sending the Resv message, which applies the reservation for the data stream.

If the data receiver is not RSVP-aware, it can not respond to the signaling and make the resource reservation happen. Similarly, if the receiver is RSVP-aware, but the sender is not, the sender can not initiate the signaling for the resource reservation.

In the Localized RSVP scheme, we expect the local end host to be RSVP-aware and we add an RSVP-aware application to a router in the local access network. This 'proxy' is called the Localized RSVP Proxy server (LRSVP proxy) and is located somewhere within the local access network - a good place would be the access network gateway. For local reservations, the proxy acts on behalf of correspondent nodes. In



this discussion, from a software engineering point of view, the proxy is its own process running on the router. Thus, the following three logical functions are kept separate: basic IP routing, basic RSVP message processing, and LRSVP proxy functionality.

Now, in order to distinguish local reservations from end-to-end reservations, we use one bit in the unused Flags field in the RSVP Session Object. The Local Indication (LI) bit (currently we use bit 0x8) is used to differentiate reservations that are internal to the access network. When the bit is set the RSVP message is part of local resource signaling and the RSVP router running the proxy will not forward the message to the next hop but instead give the message to the RSVP-application running on the router. A default value of zero indicates standard end-to-end signaling, where the proxy application is not concerned.

We also need a second bit, the Expedited Refresh (ER) bit (currently we use bit 0x4), to indicate that a Path message is sent as a refresh to a broken path and must be forwarded immediately. This indication is needed because each RSVP hop propagates a Path message before a timeout only if the Path state has changed - when a route changes at the receiver end of the data flow, a Path message is needed to set up again the Path state. This is discussed in more detail later.

When the local end host wants to make a resource reservation for a downstream flow, it needs a Path message from a node on the data path. If the data sender is not RSVP-aware, the local end host can trigger the LRSVP proxy to send the Path message on behalf of the data sender. A new message type called "Path Request", with an initial message type number 8, is used to request a Path message from the local RSVP proxy. This message has the same structure as a standard Path message.

A second message called "Path Request Tear", with an initial message type number 9, is used to tear down a downstream reservation. Note that due to the new bits and message types, all RSVP routers inside the access network must be upgraded with the LRSVP extension.

When a local end host wants to reserve resources in the local access network, it uses the LI flag in RSVP messages to indicate a local reservation. The structure of the RSVP messages follows the RSVP standard. When the router running the LRSVP proxy receives an RSVP message with the LI bit set it will notice that the flag was set and does not forward the message further to the next hop. The RSVP daemon on the router gives the message to the local RSVP application, which responds according to the following description.

### **2.1. Upstream transfers**

Setting local upstream reservations is straightforward and follows closely the standard RSVP functionality. The local end host sends the usual Path message, but sets the LI flag bit in the Session Object. The Session Object of the message defines the destination of the flow





that will eventually be transmitted from the local end host, and the Sender Template provides information about the local end host itself.

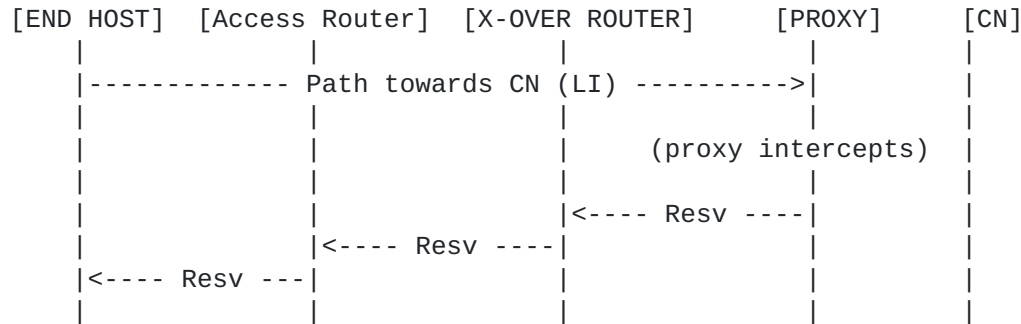


Figure 1: Local Upstream reservation

The Path message is routed within the access network and sets the Path state in RSVP routers. When the LRSVP proxy receives the Path message, it notes due to the LI bit that the reservation is meant to stay within the access network and responds with a Resv message. It will not forward the Path message further to the next hop. When a Resv message arrives at the local end host, the resources for the session defined in the Path message have been allocated.

## 2.2. Downstream transfers

Before setting a downstream resource reservation, the local end host needs to be aware of the data senders. For multimedia communications, a session is usually initiated with application layer protocols like SIP [15] or RTSP [12]. Based on this correspondence, the local end host knows the necessary information about the sender. Another, more coarse reservation could be set, for example, for browsing different audio servers; the local end host would indicate in the RSVP messages that the sender will use UDP. It is, therefore, possible to make resource reservations for any sender that wants to communicate with the local end host. However, in order to allow for more accurate QoS support, more information should be given. The way to find this information is out of scope of this document.

In order to set up a local downstream reservation we need a way to signal the LRSVP proxy to initiate the RSVP reservation set up on behalf of the sender(s), that is, to send a Path message. To achieve this, the local end host sends the Path Request message with the LI flag bit set (Fig. 2). The Path Request message is identical to a standard Path message apart from the message type field. The Session Object must include information about the recipient, the local end host in this case, and the Sender Template must define the expected sender(s). The Traffic specification (Tspec) can either be based on the local end user's wishes, a best estimate of the incoming traffic characteristics, or on application level session signaling prior to the transfer.



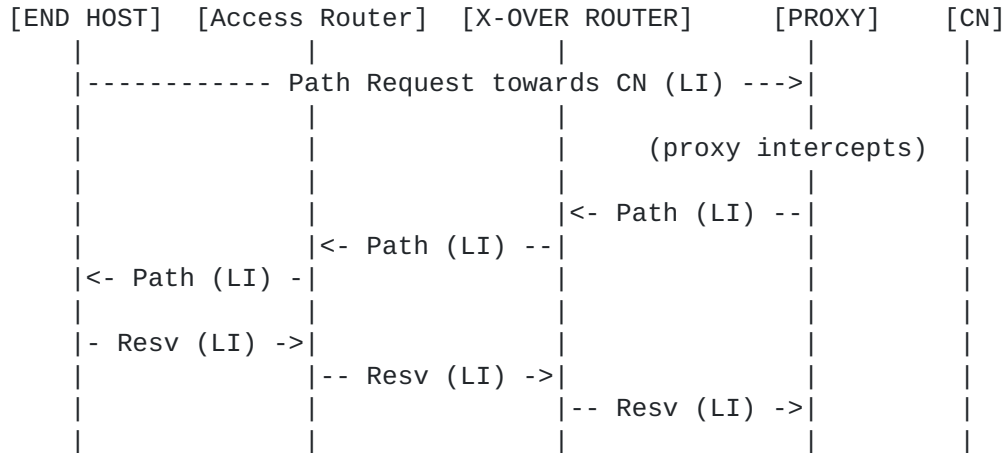


Figure 2: Local downstream reservation

When the LRSVP proxy receives this message, it detects that the message is meant to stay within the access network. The message type indicates that the proxy should initiate an RSVP reservation for a downstream flow and use the information in the message to fill the objects in a Path message. The proxy now generates a Path message that includes the parameter values in the Path Request message and sends it towards the local end host.

When the local end host receives the Path message, it responds with a Resv message with the LI flag set. This reserves the downstream resources within the access network for the senders originally identified by the local end host.

The Path Request message can also be used end-to-end, to request the correspondent node to initiate a resource reservation. In this case, the LI bit must not be set, since it would stop the message at the local access network.

### 2.3. Additional functionality

All the other features of RSVP are used with LRSVP in the standard way including the local repair mechanism and reservation tear down. Downstream reservations are torn down with the Path Request Tear message. The operation follows that of the Path Request: the message does not change states within the network, but only triggers the proxy to send a Path Tear message towards the end host for the specified session.

All messages used for local reservations must have the LI flag set in order to keep the signaling within the access network. Intermediate RSVP routers between the local end host and the LRSVP proxy should not process the Path Request message and they should forward it as an ordinary IP packet. An enhancement to the local repair changes this operation and is discussed later.

The proposed scheme also allows RSVP to be used to signal DiffServ



Code Points in a DiffServ access network using the RSVP DCLASS object [13]. The DCLASS object is used to represent and carry DiffServ code points within RSVP messages. The local end host can use the DCLASS object to instruct the LRSVP proxy to mark incoming traffic with certain DiffServ code points to trigger different forwarding behavior within the DiffServ access network. The local end host, however, needs to be aware of the different code point values and the related services, especially if other than standardized code points are used. This information can be part of host auto-configuration, for example.

Furthermore, the proposed signaling can be used at both ends of a data stream. For example, if the two end hosts in different access networks are communicating with each other, both of them could use the mechanism to allocate resources, for example, in their own access networks, independently of each other. This could happen, if the two access networks had a different view of QoS, one uses only IntServ and RSVP, while the other also uses DiffServ. In such a scenario, however, it may be more practical to use RSVP end-to-end, even if the core network connecting the two access networks does not support RSVP.

#### **2.4. Addressing Issues**

When the local end host sends Path or Path Request messages, it needs to use some IP address as the destination in the IP header. There are various options about which address the local end host can or can not use. The local end host must use in priority order (if known):

1. The address of the correspondent host,
2. The address of a proper LRSVP proxy,
3. The address of the next-hop RSVP router, or
4. The address of the default router.

The first option may not be possible, if the end host wants to allocate resources only for certain services regardless of the sender, HTTP, for example. The second possible address could be given through auto-configuration. The third option would be to send the IP packet to the next-hop RSVP router, if the end host has knowledge of it - ideally, this would be the default router, in a mobile access network, it would be the access router.

Finally, if any of the earlier addresses are not known, the end host may try to send the RSVP packet to the default router, and see if the router is also running an RSVP daemon and could handle the reservation attempt. If the default router is not running an RSVP daemon, it will return an ICMP error message.

A further concern arises if the access network has several inbound routes. It is possible that the local downstream reservation initiated by the end host is set on a wrong LRSVP proxy, one that will not get the packets arriving to the end host. This seems more of a network design issue and therefore the access network operator must locate the LRSVP proxies based on the packet routing - the proxies



could even be co-located at the access routers.

## **2.5. Interworking Issues**

The Localized RSVP makes use of two bits in the Session Object and adds two new message types. There can be situations where such a currently non-standard message arrives at a standard RSVP router.

According to the message processing rules in [RFC2209](#) [18], the Path Request and Path Request Tear messages would be forwarded intact by standard RSVP routers. However, for standard RSVP message, the bits used by LRSVP may or may not be kept between RSVP hops, and, thus, the indication of local signaling or the need for an expedited refresh may be lost. Therefore, all RSVP routers within an access network wanting to support local reservations must be set to keep the bits intact.

In one scenario, the local network of the end host would not understand the LRSVP extension or even standard RSVP. Thus, Path messages with the LI bit and Path Request message can be routed out of the local network. If the local network of the correspondent node has support for LRSVP, that LRSVP proxy gets the Path or Path Request message with the LI bit set from the external network. The proxy must drop the message and respond with a PathErr message and use a new error code called "LRSVP not supported". This would inform the host that LRSVP is not supported and it still can try end-to-end signaling.

Another interesting scenario arises when the correspondent node is a mobile node and the parties use route optimization. It can happen, that the correspondent node is actually in the same access network as the end host using LRSVP, and either or both nodes try to reserve local resources independently of each other. Now it is possible that Path and Path Request messages with the LI bit set are routed directly to the correspondent node, without going through a local network LRSVP proxy.

A solution would be that end hosts can also perform the same functions as an LRSVP proxy, that is, answer to Path messages with the LI bit set and, most importantly, handle Path Request messages as well:

- o If an end host receives an unsolicited Path message with the LI bit set, it should respond with a Resv message and not set the LI bit. The reason is that that if the LRSVP proxies drop Path messages with the LI bit set coming from external networks, the local end hosts can trust that if they receive such a message, it must have (if the network is properly configured) arrived from a node in the local access network. Now, if our end host that sent the Path message receives the Resv without the LI bit, it can use this as an indication that the correspondent node is in the local access network and may remove the LI bit in subsequent messages belonging to the same session.





o Similarly, if the correspondent node receives a Path Request message, it should respond with a Path message that does not have the LI bit set. Again, if our end host receives a Path message without the LI bit set in response to the local Path Request sent earlier, it can use this as an indication that the correspondent node is in the local domain and it may remove the LI bit in subsequent messages belonging to the same session.

Now, if the correspondent node moves again and changes access networks, the signaling is already set to standard end-to-end mode and reservations in the new RSVP-aware access network would be set in place. However, changing access networks implies most probably a change in the IP address assigned to the CN, which forces a re-reservation of resources.

In the latter scenario, it is quite possible, that the mobile correspondent node, located in the same access network as our end host, is not (L)RSVP aware. Thus, it cannot respond to the RSVP messages and local, actually, any kind of RSVP-based, reservations are not possible.

Moreover, the location of the LRSVP proxy may yet affect the signaling of two nodes within the same access network. Consider the case where each access router would also host an LRSVP proxy. Now if the two communicating nodes are connected to different access routers, the two nodes may use their own local reservations on the last-hop link, but also a standard end-to-end reservation is possible.

A further issue concerns the interactions between local and end-to-end reservations: could a local reservation be 'upgraded' into an end-to-end reservation? This should be possible for both directions:

o If the proxy receives a fully standard Path message from the local network with the same session information as an existing local reservation, it must forward the message as usual, but set a pending Path state indication for the end-to-end reservation. If a Resv arrives from the external network for this same session, it must change the reservation to an end-to-end reservation.

o If the proxy receives a Path Request message from the local network without the LI bit set, it must forward the message to the IP destination address. If the proxy receives later a Path message from the external network for an existing local session, it must set a pending state for the end-to-end reservation. If a Resv is received from the local end host without the LI bit set, the proxy must change its state for the session to 'end-to-end' (by removing a local-indication from its session structures) and forward the Resv message further to the external network.

Thus, it would be possible to upgrade a local session to end-to-end status. It is not clear whether there is a need for an end-to-end session to be 'downgraded' into a local session.



Note that when the resource signaling is going end-to-end, the local repair functionality may be affected. If both nodes use only local reservations, the local repair at either end is propagated at most to the local LRSVP proxy. With end-to-end reservations, the local repair may be propagated further away from the moving node and its access network.

### 3. Fast local repair for mobile nodes

The RSVP standard [2] defines that Path messages can perform a local repair of reservation paths. When the route between the communicating end hosts changes, a Path message will set the Path state of the reservation on the new route and a subsequent Resv message will apply the resource reservation. Therefore, by sending a Resv message a host cannot alone update the reservation, and thus perform a local repair, before a Path message has passed. The RSVP specification also says that in order to provide fast adaptation to routing changes without the overhead of short refresh periods, the local routing protocol module can notify the RSVP process of route changes for particular destinations. The RSVP process should use this information to trigger a quick refresh of state for these destinations, using the new route (Chapter 3.6, RFC2205 [2]). However, for example, Mobile IP, does not affect routing directly in routers, and thus mobility may not be noticed at intermediate RSVP routers.

When the mobile node has moved, it can send a Path message for each upstream resource reservation and initiate the standard local repair mechanism (Fig. 3). If there is no cross-over router, and the Path message arrives at a new LRSVP proxy, the proxy will reply to the Path with a Resv, similarly as it would for a new resource reservation request (what this actually looks like to the new proxy).

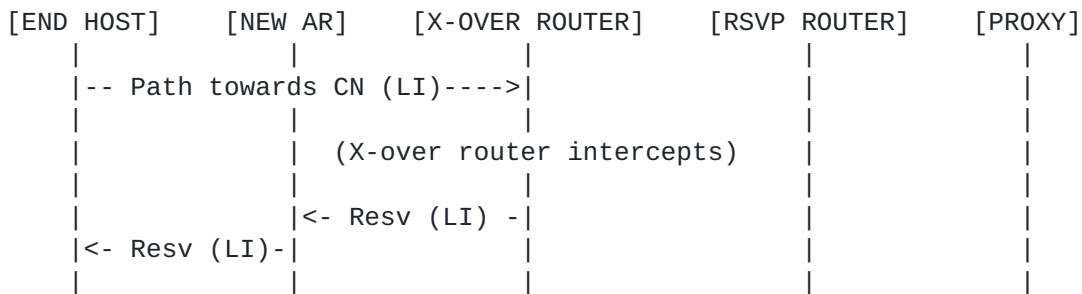


Figure 3: Fast upstream reservation

For the downstream, the mobile node will need to wait until it receives a Path message, setting up the Path state on the new route. Only after receiving the Path message, the mobile can send a Resv message to re-reserve the downstream resources.

The Path Request message can be used in mobile networks to initiate a faster local repair of downstream reservations on behalf of a mobile node that changed access routers during an ongoing RSVP session. When the mobile node changes its access router in the network, it should



send the Path Request message immediately after the handover (Fig 4).

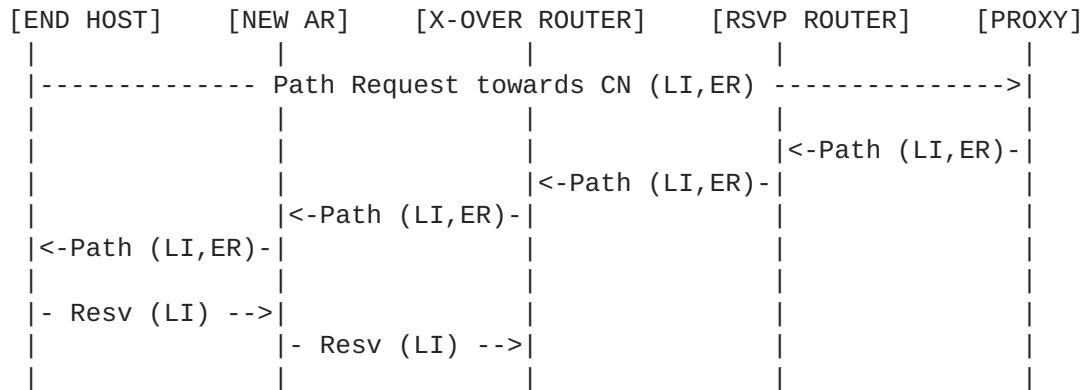


Figure 4: Fast downstream reservation

The message must have the ER bit set to indicate that the request is for an existing session and triggered due to movement. The Path Request message is forwarded through the intermediate RSVP routers until it arrives at the LRSVP proxy. The message would then instruct the proxy to initiate a local repair by sending the needed Path message. The proxy must set the ER bit in the Session Object to indicate that this Path message is not an ordinary refresh message but instead triggered by a routing change and therefore must be forwarded immediately to the next hop. If the ER bit is not set, the RSVP router in Fig. 4 would not forward the Path message immediately towards the mobile node but, instead, would wait until its own scheduled refresh timeout.

If the movement of the mobile node results in packets to flow through a new LRSVP proxy, the Path Request message would re-reserve the local resources for the new path. In this case, the proxy notes that the ER bit is set, but, since there is no existing state, it will initiate a new reservation. The ER bit must not be set in the following Path message since the reservation is a new one for this route.

An enhancement to the scheme would allow a cross-over RSVP router that has the reservation for the mobile node stored on a different interface to send the needed Path message (Fig. 5). RSVP routers inside the access network would look into Path Request messages. If the router notices it is the cross-over router, it sends a Path message towards the local end host. If an RSVP router sends the Path message, it must not send the Path Request message any further. This requires more processing from intermediate RSVP routers, but allows for faster state updates. This functionality would be especially important when the session is end-to-end instead of local: the Path Request message would not go to the correspondent node, but trigger the closer cross-over router to repair the local path of the reservation.



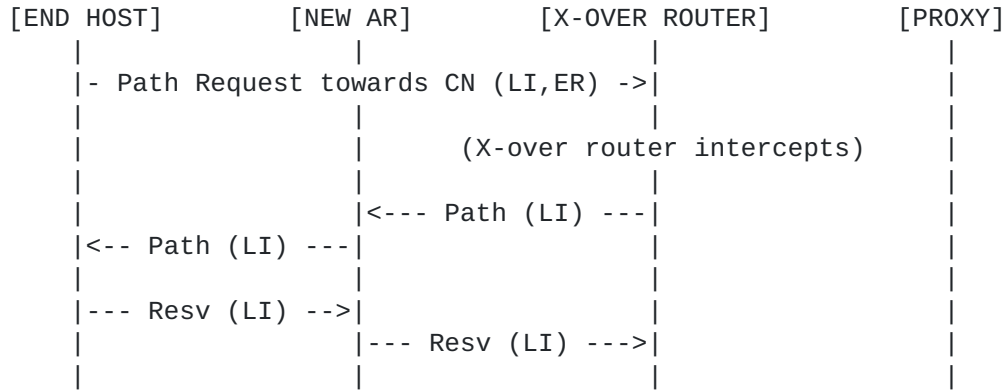


Figure 5: Enhancement of the fast downstream reservation

The LI flag must be set in all RSVP refresh messages if the reservation is set for the local access network. This will prevent a refresh message, like the Path Request message, to be routed out of the access network.

#### 4. Security Consideration

The Path Request message triggers most processing at the LRSVP proxy. This could potentially be used for Denial of Service attacks. A possible solution is to make RSVP daemons located on access routers make a sanity check on all Path Request (and Path Request Tear) messages: the receiver of the stream must be a node on a link connected to the AR. This has the benefit that the proxy may trust that the access router has validated the message; this can be seen as distributed processing of the authentication and authorization data. The same considerations apply for the Path message. In order to secure any RSVP signaling, a security association must be set up between the local end hosts and the access routers.

The RSVP daemon at the end hosts and LRSVP proxy must also modify their security/sanity checks to allow the end host to send the Path Request with apparently suspicious session information (identifying the correspondent node(s)). Also, the proxy must be able to send RSVP messages "on-behalf" of external network nodes.

The LRSVP proxy must be configured to identify its ingress and egress interfaces. If the proxy receives a Path or a Path Request message with the LI bit set from outside the access network, it must drop the message.

The two new messages can make use of the standard RSVP INTEGRITY and POLICY objects. This requires that the MN and ARs share the required keys. For the remaining functionality, the security considerations with standard RSVP apply, which are analyzed well by the NSIS WG in [17].





## 5. IANA Considerations

IANA needs to allocate the two flag bits in the RSVP Session Object, the LI and ER bits. In addition, IANA needs to give two RSVP message type numbers to the Path Request and Path Request Tear messages and one Error Type indicating that a local resource reservation is not allowed.

## 6. Summary

In summary, the Localized RSVP requires the following changes in the standard RSVP protocol:

- a) A new message type and number, named Path Request (initially number 8).
- b) A new message type and number, named Path Request Tear (initially number 9).
- c) A bit from the Session Object for the use as the Local Indication (LI) (initially bit 0x8).
- d) A bit from the Session Object for use as the Expedited Refresh (ER) indication (initially bit 0x4).
- e) An RSVP router must keep the LI bit set in all messages belonging to that session, if it encounters packets with the bit set.
- f) An RSVP router that is not also a proxy, must forward an RSVP packet with a message type "Path Request" without storing state.
- g) An RSVP router that is not also a proxy, must forward an RSVP packet with a message type "Path Request Tear" without affecting the stored state.
- h) An access network RSVP router should be able to use the Path Request as an indication of the need for a local repair. This can enable faster reservation set up following a handover. This affects point f), because the router that receives a Path Request must first check if it has the Path state stored on another network interface. If one is present, the Path Request message should not be forwarded to the next hop and instead the router must send a Path message towards the mobile node.
- i) A new error code to indicate that the access network does not support local reservations. If local resources cannot be requested, the end-host can always try to do end-to-end signaling.

To summarize, these changes are small and would make RSVP more appealing as a signaling protocol for IP access networks. The changes are required only within access networks, unless the Path Request message is deemed useful to use end-to-end through the Internet.



## 7. Normative References

- [1] R. Braden, D. Clark, S. Shenker, "Integrated Services in the Internet Architecture: an Overview". [RFC 1633](#), June 1994.
- [2] R. Braden, L. Zhang, S. Berson, S. Herzog, S. Jamin. "Resource ReSerVation Protocol (RSVP), Version 1 Functional Specification". [RFC 2205](#), September, 1997.
- [3] J. Wroclawski, "The Use of RSVP with IETF Integrated Services". [RFC 2210](#), September, 1997.
- [4] S. Blake, D. Black, M. Carlson, E. Davies, Z. Wang, W. Weiss, "An Architecture for Differentiated Services". [RFC 2475](#), December, 1998.
- [13] Y. Bernet, "Format of the RSVP DCLASS Object". [RFC 2996](#), November 2000.
- [18] R. Braden, "Resource ReSerVation Protocol (RSVP) -- Version 1 Message Processing Rules". [RFC 2209](#), September 1997.

## 8. Non-Normative References

- [5] G. Huston, "Next Steps for the IP QoS Architecture". [RFC 2990](#), November 2000.
- [6] K. Nichols, V. Jacobson, L. Zhang, "A Two-bit Differentiated Services Architecture for the Internet". [RFC 2638](#), July 1999.
- [7] R. Yavatkar, D. Hoffman, Y. Bernet, F. Baker, M. Speer, "SBM (Subnet Bandwidth Manager): A Protocol for RSVP-based Admission Control over IEEE 802-style networks". [RFC 2814](#), May 2000.
- [8] Phil Chimento and Ben Teitelbaum, "QBone Bandwidth Broker Architecture". The Internet2 initiative, June 2000. (<http://qbone.internet2.edu/bb/bboutline2.html>).
- [9] S. Gai, D. G. Dutt, N. Elfassy, Y. Bernet, "RSVP Proxy". Internet Draft (work in progress), March 2002 (expired) ([draft-ietf-rsvp-proxy-03.txt](#)).
- [10] Jukka Manner and Kimmo Raatikainen, "Extended Quality-of-Service for Mobile Networks". Proceedings of the 9th International Workshop on Quality of Service (IWQoS), June 2001, pp. 275-280. Published in the series Springer Lecture Notes in Computer Science (LNCS) 2092.
- [11] IST-BRAIN Project, "Deliverable D2.2: BRAIN architecture specifications and models, BRAIN functionality and protocol specification". March 2001, (available at: <http://www.ist-brain.org>).
- [12] H. Schulzrinne, A. Rao, R. Lanphier, "Real Time Streaming Protocol (RTSP)". [RFC 2326](#), April 1998.



[15] J. Rosenberg et al., "SIP: Session Initiation Protocol". [RFC 3261](#), June 2002

[16] J. Manner, and M. Kojo, "Mobility related terminology". [RFC 3753](#), June 2004.

[17] H. Tschofenig, "RSVP Security Properties". Internet Draft (work in progress), September 2004.

## 9. Authors' Addresses

Questions about this document may be directed to:

Jukka Manner  
Department of Computer Science  
University of Helsinki  
P.O. Box 68  
FIN-00014 HELSINKI  
Finland

Voice: +358-9-191-51298  
Fax: +358-9-191-51120  
E-Mail: [jmanner@cs.helsinki.fi](mailto:jmanner@cs.helsinki.fi)

Markku Kojo  
Department of Computer Science  
University of Helsinki  
P.O. Box 68  
FIN-00014 HELSINKI  
Finland

Voice: +358-9-191-51305  
Fax: +358-9-191-51120  
E-Mail: [kojo@cs.helsinki.fi](mailto:kojo@cs.helsinki.fi)

Kimmo Raatikainen  
Department of Computer Science  
University of Helsinki  
P.O. Box 68  
FIN-00014 HELSINKI  
Finland

Voice: +358-50-483-6275  
Fax: +358-9-191-51120  
E-Mail: [kraatika@cs.helsinki.fi](mailto:kraatika@cs.helsinki.fi)

Tapio Suihko  
VTT Information Technology  
P.O. Box 1203  
FIN-02044 VTT  
Finland



Voice: +358-9-456-6078  
Fax: +358-9-456-7028  
E-Mail: tapio.suihko@vtt.fi

Mika Liljeberg  
Nokia Research Center  
P.O. Box 407  
FIN-00045 Nokia Group  
Finland

Voice: +358-50-4836791  
E-Mail: Mika.Liljeberg@nokia.com

#### Acknowledgment

This work has been partially performed in the framework of the IST project IST-2000-28584 MIND, which is partly funded by the European Union. The authors would like to acknowledge the help of their colleagues in preparing this document, in particular Eleanor Hepworth and Alberto Lopez.

Copyright (C) The Internet Society (2004). This document is subject to the rights, licenses and restrictions contained in [BCP 78](#), and except as set forth therein, the authors retain all their rights.

This document and the information contained herein are provided on an "AS IS" basis and THE CONTRIBUTOR, THE ORGANIZATION HE/SHE REPRESENTS OR IS SPONSORED BY (IF ANY), THE INTERNET SOCIETY AND THE INTERNET ENGINEERING TASK FORCE DISCLAIM ALL WARRANTIES, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO ANY WARRANTY THAT THE USE OF THE INFORMATION HEREIN WILL NOT INFRINGE ANY RIGHTS OR ANY IMPLIED WARRANTIES OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE.



