

Network Working Group
Request for Comments: 2205
Category: Standards Track

R. Braden, Ed.
ISI
L. Zhang
UCLA
S. Berson
ISI
S. Herzog
IBM Research
S. Jamin
Univ. of Michigan
September 1997

Resource ReSerVation Protocol (RSVP) --

Version 1 Functional Specification

Status of this Memo

This document specifies an Internet standards track protocol for the Internet community, and requests discussion and suggestions for improvements. Please refer to the current edition of the "Internet Official Protocol Standards" (STD 1) for the standardization state and status of this protocol. Distribution of this memo is unlimited.

Abstract

This memo describes version 1 of RSVP, a resource reservation setup protocol designed for an integrated services Internet. RSVP provides receiver-initiated setup of resource reservations for multicast or unicast data flows, with good scaling and robustness properties.

Table of Contents

1. Introduction	4
1.1 Data Flows	7
1.2 Reservation Model	8
1.3 Reservation Styles	11
1.4 Examples of Styles	14
2. RSVP Protocol Mechanisms	19
2.1 RSVP Messages	19
2.2 Merging Flowspecs	21
2.3 Soft State	22
2.4 Teardown	24
2.5 Errors	25
2.6 Confirmation	27
2.7 Policy Control	27
2.8 Security	28
2.9 Non-RSVP Clouds	29
2.10 Host Model	30
3. RSVP Functional Specification	32
3.1 RSVP Message Formats	32
3.2 Port Usage	47
3.3 Sending RSVP Messages	48
3.4 Avoiding RSVP Message Loops	50
3.5 Blockade State	54
3.6 Local Repair	56
3.7 Time Parameters	57
3.8 Traffic Policing and Non-Integrated Service Hops	58
3.9 Multihomed Hosts	59
3.10 Future Compatibility	61
3.11 RSVP Interfaces	63
4. Acknowledgments	76
APPENDIX A. Object Definitions	77
APPENDIX B. Error Codes and Values	92
APPENDIX C. UDP Encapsulation	98
APPENDIX D. Glossary	102
REFERENCES	111
SECURITY CONSIDERATIONS	111
AUTHORS' ADDRESSES	112

What's Changed

This revision contains the following very minor changes from the ID14 version.

- o For clarity, each message type is now defined separately in [Section 3.1](#).
- o We added more precise and complete rules for accepting Path messages for unicast and multicast destinations ([Section 3.1.3](#)).
- o We added more precise and complete rules for processing and forwarding PathTear messages ([Section 3.1.5](#)).
- o A note was added that a SCOPE object will be ignored if it appears in a ResvTear message ([Section 3.1.6](#)).
- o A note was added that a SENDER_TSPEC or ADSPEC object will be ignored if it appears in a PathTear message ([Section 3.1.5](#)).
- o The obsolete error code Ambiguous Filter Spec (09) was removed, and a new (and more consistent) name was given to error code 08 (Appendix B).
- o In the generic interface to traffic control, the Adspec was added as a parameter to the AddFlow and ModFlow calls (3.11.2). This is needed to accommodate a node that updates the slack term (S) of Guaranteed service.
- o An error subtype was added for an Adspec error (Appendix B).
- o Additional explanation was added for handling a CONFIRM object ([Section 3.1.4](#)).
- o The rules for forwarding objects with unknown class type were clarified.
- o Additional discussion was added to the Introduction and to [Section 3.11.2](#) about the relationship of RSVP to the link layer. ([Section 3.10](#)).
- o [Section 2.7](#) on Policy and Security was split into two sections, and some additional discussion of security was included.
- o There were some minor editorial improvements.

1. Introduction

This document defines RSVP, a resource reservation setup protocol designed for an integrated services Internet [RSVP93, [RFC 1633](#)]. The RSVP protocol is used by a host to request specific qualities of service from the network for particular application data streams or flows. RSVP is also used by routers to deliver quality-of-service (QoS) requests to all nodes along the path(s) of the flows and to establish and maintain state to provide the requested service. RSVP requests will generally result in resources being reserved in each node along the data path.

RSVP requests resources for simplex flows, i.e., it requests resources in only one direction. Therefore, RSVP treats a sender as logically distinct from a receiver, although the same application process may act as both a sender and a receiver at the same time. RSVP operates on top of IPv4 or IPv6, occupying the place of a transport protocol in the protocol stack. However, RSVP does not transport application data but is rather an Internet control protocol, like ICMP, IGMP, or routing protocols. Like the implementations of routing and management protocols, an implementation of RSVP will typically execute in the background, not in the data forwarding path, as shown in Figure 1.

RSVP is not itself a routing protocol; RSVP is designed to operate with current and future unicast and multicast routing protocols. An RSVP process consults the local routing database(s) to obtain routes. In the multicast case, for example, a host sends IGMP messages to join a multicast group and then sends RSVP messages to reserve resources along the delivery path(s) of that group. Routing protocols determine where packets get forwarded; RSVP is only concerned with the QoS of those packets that are forwarded in accordance with routing.

In order to efficiently accommodate large groups, dynamic group membership, and heterogeneous receiver requirements, RSVP makes receivers responsible for requesting a specific QoS [[RSVP93](#)]. A QoS request from a receiver host application is passed to the local RSVP process. The RSVP protocol then carries the request to all the nodes (routers and hosts) along the reverse data path(s) to the data source(s), but only as far as the router where the receiver's data path joins the multicast distribution tree. As a result, RSVP's reservation overhead is in general logarithmic rather than linear in the number of receivers.

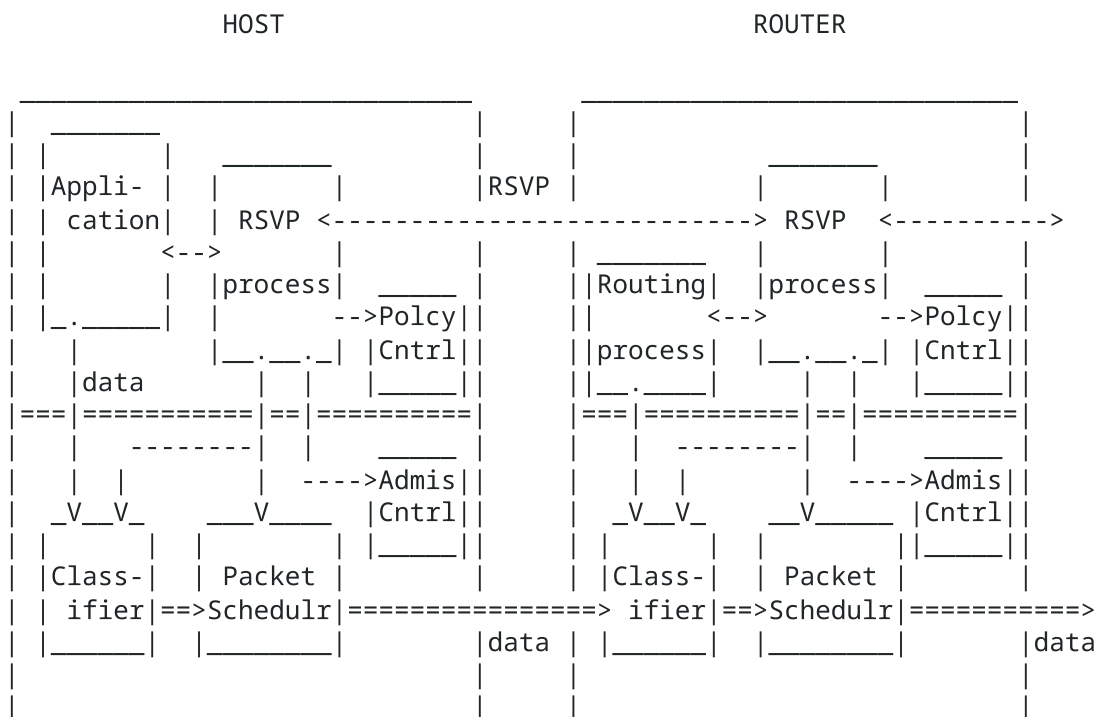


Figure 1: RSVP in Hosts and Routers

Quality of service is implemented for a particular data flow by mechanisms collectively called "traffic control". These mechanisms include (1) a packet classifier, (2) admission control, and (3) a "packet scheduler" or some other link-layer-dependent mechanism to determine when particular packets are forwarded. The "packet classifier" determines the QoS class (and perhaps the route) for each packet. For each outgoing interface, the "packet scheduler" or other link-layer-dependent mechanism achieves the promised QoS. Traffic control implements QoS service models defined by the Integrated Services Working Group.

During reservation setup, an RSVP QoS request is passed to two local decision modules, "admission control" and "policy control". Admission control determines whether the node has sufficient available resources to supply the requested QoS. Policy control

Explore Litigation Insights

Docket Alarm provides insights to develop a more informed litigation strategy and the peace of mind of knowing you're on top of things.

Real-Time Litigation Alerts



Keep your litigation team up-to-date with **real-time alerts** and advanced team management tools built for the enterprise, all while greatly reducing PACER spend.

Our comprehensive service means we can handle Federal, State, and Administrative courts across the country.

Advanced Docket Research



With over 230 million records, Docket Alarm's cloud-native docket research platform finds what other services can't. Coverage includes Federal, State, plus PTAB, TTAB, ITC and NLRB decisions, all in one place.

Identify arguments that have been successful in the past with full text, pinpoint searching. Link to case law cited within any court document via Fastcase.

Analytics At Your Fingertips



Learn what happened the last time a particular judge, opposing counsel or company faced cases similar to yours.

Advanced out-of-the-box PTAB and TTAB analytics are always at your fingertips.

API

Docket Alarm offers a powerful API (application programming interface) to developers that want to integrate case filings into their apps.

LAW FIRMS

Build custom dashboards for your attorneys and clients with live data direct from the court.

Automate many repetitive legal tasks like conflict checks, document management, and marketing.

FINANCIAL INSTITUTIONS

Litigation and bankruptcy checks for companies and debtors.

E-DISCOVERY AND LEGAL VENDORS

Sync your system to PACER to automate legal marketing.