Performance Evaluation of the Extensions for Control Message Retransmissions in RSVP

Michael Menth [*], Rüdiger Martin

Department of Distributed Systems, Institute of Computer Science,
University of Würzburg, Am Hubland, 97074 Würzburg, Germany
{menth,martin}@informatik.uni-wuerzburg.de

Abstract. Quality of Service (QoS) for real-time transmission can be achieved by resource reservation in the routers along the path. In recent years, several protocols and extensions to them have been designed for signaling resource reservation in IP networks. This work reviews various protocols that exhibit different signaling concepts. Then, we study the impact of control message retransmissions (CMR) and the control option on the reservation establishment delay (RED) and the reservation teardown delay (RTD). Numerical results quantify the resulting performance gain in different networking scenarios.

1 Introduction

Future communication networks will guarantee seamless quality of service (QoS) data transportation from the sender to the receiver. The network must provide sufficient resources to forward the data in an adequate way to meet the loss and delay requirements of the traffic. To achieve that goal, massive overprovisioning can be applied as well as intelligent traffic engineering techniques. One of them is admission control (AC): When the network's capacity does not suffice to transport all offered traffic, AC shelters the network from overload by admitting only a limited number of reservation requests. Thus, QoS for the flows in place is maintained at the expense of blocked flows. In order to perform AC in a network entity, the amount of requested resource of each flow must be known beforehand and is usually delivered by a resource reservation protocol.

In recent years, several resource reservation protocols have been designed for IP networks. In this paper we would like to give an overview over the most prominent protocols: RSVP, RSVP refresh overhead reduction extensions, aggregation of RSVP reservations, Boomerang, YESSIR, BGRP, and stateless reservation protocols. These protocols distinguish in syntax and semantics and reveal also different information passing concepts.

In the past, protocol performance has been studied using software implementations [1–4]. In [5] the reliability of RSVP was studied. In our investigation we focus on the control message retransmission (CMR) option which is suggested in the refresh overhead reduction extensions of RSVP [6]. We evaluate its impact on the reservation establishment delay (RED) and the reservation teardown delay (RTD). We perform these

[*] This work was partially funded by the Bundesministerium für Bildung und Forschung of the Federal Republic of Germany (Förderkennzeichen 01AK045). The authors alone are responsible for the content of the paper.

studies in various networking scenarios and take signaling paradigms from standard RSVP and other reservation protocols into account.

This paper is structured as follows. In Section 2 we present the above mentioned protocols. Then, we study the response time of resource reservation protocols with special respect to the control message retransmission (CMR) feature in the RSVP extensions. Numerical results illustrate their behavior and quantify the performance gain. In Section 4 we summarize this work.

2 An Overview of Resource Reservation Protocols

Real-time applications like voice over IP (VoIP) [7] or video conference require signaling protocols for the application layer as well as for the network and the transport layer.

Applications need to identify and locate their communication peers and to negotiate session parameters. Codecs have to be agreed on and translators can be involved in case of incompatible end systems. These and other tasks are performed by standards like H.323, the Session Initiation Protocol (SIP), and the Media Gateway Control Protocol (MGCP). The Real-time Transport Protocol (RTP) provides means for data synchronization to avoid distorted time lines for presentation. On the transport layer we have the User Datagram Protocol (UDP) and the Transmission Control Protocol (TCP). They ensure that IP packets are associated with the correct ports in the end systems, and TCP provides means to detect and repair packet loss. The network layer forwards the data to their destination. It consists of the Internet Protocol (IP) and the routing protocols.

The link layer offers QoS mechanisms as it has the control over the overall capacity of a link. These resources do not only relate to mere bandwidth but also to CPU processing time, buffer space, and others. Resource reservation can be performed per hop on all intermediate links between sender and receiver. The control messages are transported using regular IP packets (network layer) but their information relates to the link layer.

This work focuses on signaling for resource reservation. In this section we present various existing protocols and concentrate on their information forwarding paradigm.

2.1 Resource Reservation Protocol (RSVP)

RSVP has been conceived by the IETF for the signaling of reservation requests within an Integrated Services network [8, 9]. Both unicast and multicast applications are supported and different reservation styles (e. g. shared reservations) are possible.

Connection Establishment. To initiate a reservation with RSVP, the sending node issues a so-called PATH message that establishes a PATH state (PS) in the intermediate hops on the way to the desired destination machine. The state of a flow is the information related to it in a router. The destination router responds with a RESV message that visits the intermediate routers in the reverse direction using the previous hop information of the PS (cf. Figure 1). On that way, the RESV states (RS) are established. This ensures that resources are reserved in each router in downstream direction. The first pass from

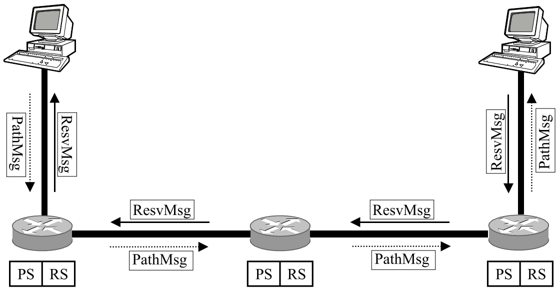

Fig. 1. Control message flow in RSVP.

the sender to the receiver (PATH msg.) collects advertising information that is delivered to the receiver to enable the receiver to make appropriate reservation requests. The actual reservation is made on the way back to the sender (RESV msg.). Hence, RSVP uses a two-pass signaling approach, also known as one-pass with advertising (OPWA). Explicit PATHERR and RESVERR messages indicate errors, and TEARDOWN messages tear down the connection and remove the states in the routers.

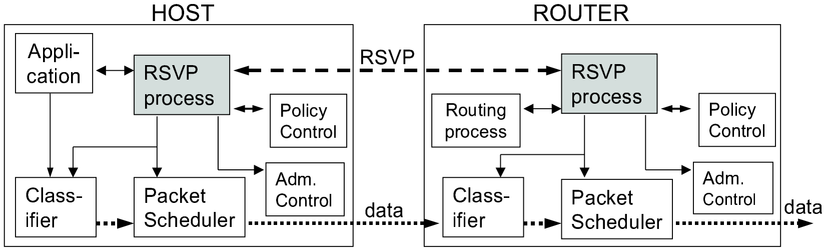

Fig. 2. The establishment of a reservation using RSVP.

When a RESV message is received by a router, the required actions are taken to set up the reservations for the respective data flow (cf. Figure 2). The admission decision is based on the flow and reservation specifiers. If the request succeeds, the classifier and the scheduler are configured to forward the data flow messages.

Soft States. RSVP control messages are usually sent directly in IP or UDP datagrams. This communication is unreliable in both cases. The control messages do not belong to a user data flow and are not protected by the reservation. In addition, the end systems may go down without notifying the network. Thus, a mechanism is required to remove the unused states. RSVP uses a soft state approach to cope with that: The states time out and disappear after a cleanup time L unless they are refreshed by another PATH or RESV message. To keep the connection alive, every participating node sends periodically PATH and RESV messages to its neighboring hops with refresh period R. RSVP [8] suggests to set L to $3 \cdot R$. If the source stops without tearing down the connection or in case of routing changes, the PATH and RESV states will eventually time out in all unused routers.

2.2 RSVP Refresh Overhead Reduction Extensions

RSVP messages consist of different standardized objects that carry flow related information. The RSVP refresh overhead reduction extensions [6, 10] introduce several new objects to RSVP that reduce the overhead due to the transmission of refresh messages. Apart from that, RSVP is enhanced by control message retransmission (CMR) capabilities.

BUNDLE Messages. RSVP nodes send one PATH and RESV message per refresh interval and connection. Since several connections are carried on the same link, their refresh messages can be handled within one single BUNDLE message where just the different message bodies are assembled. This yields just a reduction of the mere control packet frequency but not a reduction of the control message frequency. The bandwidth consumed by control messages is negligible and BUNDLE messages hardly reduce it. There are also as many operations required by the router as in normal RSVP.

Complexity Reduction by MESSAGE_IDs. PATH and RESV messages are sent periodically per RSVP connection and their content rarely changes. The receiver of such a message identifies the corresponding flow using the traffic descriptor and refreshes or updates its state. The introduction of a MESSAGE_ID object alleviates this task. If a control message is expected to change the state in the next router, it is equipped with a unique MESSAGE_ID number. Consecutive control messages that just refresh this state are equipped with the same MESSAGE_ID. The desired state is then identified by a hash value using the MESSAGE_ID and can be refreshed without processing the whole control message. MESSAGE_IDs speed up the lookup time for a flow. The impact on the flow processing time can be studied by comparing the performance of RSVP implementations.

Summary Refresh Extensions. The MESSAGE_IDs do not require to be sent with their related control message. A MESSAGE_ID LIST object may contain only MESSAGE_ID objects instead of the whole control message that would only be used in the failure case. This reduces also the required bandwidth for signaling, however, this has no impact on the network utilization since the fraction of signaling traffic is small anyway. To handle cases where the receiver encounters an inconsistent state view, the receiver may order new PATH or RESV messages by issuing a MESSAGE_ID_NACK that refers to the corrupted RSVP connection.

Control Message Retransmissions (CMR). Last but not least, it is possible to set an ACK_DESIRED flag within a MESSAGE_ID object to indicate the receiver to send a MESSAGE_ID_ACK to acknowledge the receipt of a control message. If the sender has not yet received the MESSAGE_ID_ACK object after R_f time, it retransmits the respective control message. This makes the communication more robust against packet loss which is an important issue, e. g., in wireless networks. To adjust this mechanism to potential network overload situations, an exponential backoff algorithm is introduced to avoid unnecessary control messages. On the one hand, CMR make RSVP implementations more complex (additional timers and control messages) but on the other hand, they make the communication on a lossy link more reliable which reduces the response time of the distributed RSVP processes. The impact depends on the networking scenario and will be investigated in the second part of that work.

2.3 Aggregation of RSVP Reservations

The above mentioned modifications to RSVP tend to reduce the protocol overhead per RSVP control message and allow better performing implementations of the RSVP state machine. However, they are not able to solve the fundamental scaling problem: The processing costs in a router grow linearly with the number of supported reservations which is feasible in the access network with only a few QoS flows but not in a core network. Therefore, [11] suggests an aggregator at a border router of a network that summarizes many individual RSVP reservations into one aggregated reservation. The aggregated reservations share the same path through the network and they are deaggregated at the egress point. This reduces the number of reservation states drastically within the network and relieves the core routers from processing individual reservation requests. The number of end-to-end tunnel reservations rises quadratically with the number of aggregators and deaggregators in a network. The same objective can be achieved by using the aggregation capabilities of Multiprotocol Label Switching (MPLS) [12–14].

2.4 Boomerang

The Boomerang protocol [15] aims at reducing some part of the overhead that is induced by RSVP. There are no PATH and RESV messages. The sender generates a Boomerang message that is forwarded hop by hop to the receiver. Along that path, the routers that understand Boomerang perform a reservation for a flow. As soon as the message arrives at the receiver, the reservation is already in place. The receiver does not even need to process the message, it just bounces the message back to the sender to notify the establishment. Optionally, the return channel of a bidirectional session may be reserved on the way back. Note that a different path may be taken for that purpose (cf. Figure 3).

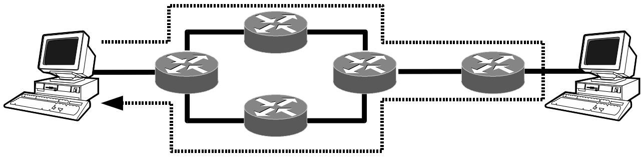

Fig. 3. A bidirectional reservation setup by Boomerang.

If a reservation request fails or if a session terminates, the reservation states can be torn down with a reservation request of size zero. Only the sending node generates signaling messages, therefore, the Boomerang approach is simpler than RSVP since the major complexity and processing is located at the sender. The concept is also based on soft states and requires refreshes to keep the reservation alive. If the Boomerang message does not return to the initiating node within a certain time, it is considered to be lost, so that the sender can take appropriate actions. The Boomerang protocol induces clearly less burden on the routers than conventional RSVP implementations [3].

2.5 YESSIR - YEt another Sender Session Internet Reservation

YESSIR [16] is a reservation protocol that is based on RTP [7, 17]. RTP is usually a wrapper for application data and adds sequence numbers, time stamps and other identifiers. Each session is controlled by the Real-time Transport Control Protocol (RTCP). Senders and receivers send periodically sender and receiver reports (SR, RR). SRs contain throughput and other information about the last report interval and allow, e. g., the derivation of the current round-trip time in the network. RRs indicate packet loss and delay statistics among others. This is extremely useful for adaptive applications. YESSIR works like RSVP also in a unicast and a multicast environment and offers also different reservation styles.

YESSIR reservation messages are piggybacked at the end of RTCP SR or RR messages, possibly enhanced by additional YESSIR-specific data, carried in IP packets with router-alert option, i. e. they are intercepted by routers and processed by those supporting this option. As with Boomerang, reservations are triggered by the sender. If a router along the way is not able to provide the requested resources, the exact reasons for the reservation failure can be remarked. This helps the end systems to either drop the session or to lower the reservation request. The rate for the reservation can be given explicitly, it may be deduced from codec types in the RTP payload or it may also be inferred from the size of the payload and the corresponding time stamps. YESSIR also relies on the soft state approach. As with Boomerang, only the sender issues refreshes and the session can be torn down with an explicit RTCP BYE message. Unlike RSVP, the intermediate nodes are not able to issue error messages, failure situations have to be recognized by the receiver and reported via RRs to the sender.

2.6 Border Gateway Reservation Protocol (BGRP)

BGRP [18] has been conceived for inter-domain use and to work in cooperation with the Border Gateway Protocol (BGP) for routing. It is used for reservations between border routers only. BGRP addresses the scalability problem directly since it is designed to aggregate all inter-domain reservations with the same autonomous system (AS) gateway as destination into a single funnel reservation, no matter of their origin.

We explain briefly how signaling with BGRP works to set up a sink tree reservation (cf. Figure 4). A PROBE message is sent from a source border router to a destination border router and collects the visited border routers. Upon the reception of a PROBE message, the border routers check for available resources, and forward the PROBE packet towards the destination. The destination border router terminates this process. It

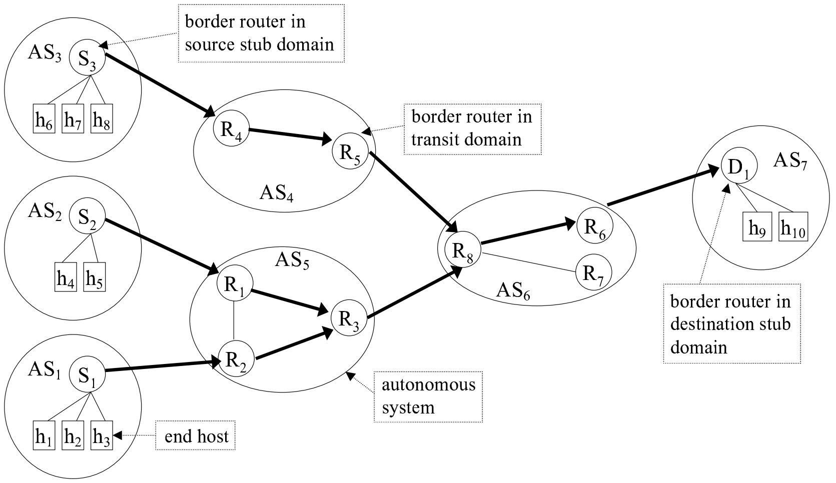
border router in source stub domain
AS3
S3
h6
h7
h8
R4
R5
AS4
border router in transit domain
AS2
S2
h4
h5
AS5
R1
R3
R2
AS1
S1
h1
h2
h3
end host
autonomous system
R8
R6
R7
AS6
D1
AS7
h9
h10
border router in destination stub domain

Fig. 4. Signaling in BGRP.

converts the PROBE message into a GRAFT message and inserts an ID that identifies its sink tree. The GRAFT message travels back on the collected path: The required reservation states are established and marked with the ID, or they are updated if they are already existent. The PROBE and GRAFT messages contain only a relative reservation offset, therefore, the communication for GRAFT messages must be reliable (e. g. using TCP). BGRP is a soft state protocol, therefore, neighboring routers exchange explicit REFRESH messages to keep the reservation alive.

Here, we have also receiver based reservations but in contrast to RSVP, the information is not stored per flow but per aggregate which is characterized by the same destination AS. Like with RSVP reservation aggregation, the advantage of that approach is the state scalability in the routers. In addition, there is only one sink tree reservation for every destination AS in each border router. Hence, the number of BGRP reservations scales linearly with the number of AS.

2.7 Measurement Based Admission Control

The above described reservation protocols suffer from a common disadvantage. They keep a state for every reservation, either for an individual or for an aggregate reservation. Their information is used to derive the already reserved capacity on the link and the AC decision is based on the remaining capacity.

As an alternative, measurement based AC (MBAC) gains this information by an estimation of the current network load by traffic measurements. Therefore, routers do not store any flow related information and MBAC has no scaling problems. However, MBAC might not be a good solution if strict guarantees are required. To support this architecture, the signaling protocols may be the same as in the conventional case, however, some special purpose protocols for MBAC have been also conceived [19–21].

MBAC is problematic if flows do not send continuously since this leads to wrong traffic estimations.

2.8 Stateless Reservation Protocols

A special MBAC approach are stateless reservation protocols. We only describe their basic architecture, for further details the reader is referred to [22–24].

End systems equip the packets that belong to reserved traffic with a special priority tag. Instead of keeping a record for different reservations, the intermediate routers analyze the packet streams on each outgoing link. They count the packets with the reservation tag within a given interval and infer the reserved rate R_{res}. For a correct measurement, this requires that the holder of a reservation sends also packets when the application is idle, otherwise, the reservation is reduced.

A sender can reserve resources from the network by sending request packets at a certain rate to the destination. The newly admitted rate R_{new} is also recorded over this interval. The sum of $R_{res} + R_{new}$ is an upper bound on the reserved rate on an output port and may not exceed the link capacity R_{link}. The intermediate hops drop the request packets intentionally if there is not enough capacity ($R_{res} + R_{new} > R_{link}$) to transport another flow. Otherwise, these packets are forwarded. A new reservation is only admitted if its request packets pass all AC tests in the intermediate routers and the destination signals the success back to the source. This is only the basic mechanism that does not reveal the manifold implementation problems.

3 Performance Evaluation of Control Message Retransmissions

Retransmissions for control messages (CMR) make the communication for RSVP control traffic more robust against packet loss. This is crucial in scenarios with high packet loss, e. g. in wireless networks. CMR lead to faster reactions of the RSVP processes in the distributed routers. The reaction time of the remote processes affects the reservation establishment delay (RED) and the reservation teardown delay (RTD). The establishment of a reservation is a prerequisite for the start of real-time applications, e. g. video streaming, and it is annoying for the user if he or she has to wait. Therefore, a short RED is important for good QoS perception by the user (post dial delay). Unused and blocked capacity is not profitable, therefore, resources should be released very quickly after session termination by the application layer. This requires a short RTD. In this section, we give a brief description of the options under study and illustrate their influence on RED and RTD.

3.1 Model Description

In our investigation we consider a general signaling protocol with different options. We borrow most of the nomenclature from RSVP but we do not limit our experiments to configurations in RSVP. We neglect the message processing times in the routers and focus on the effect of the mere transmission times and involved timeout values. The performance measures RED and RTD do not depend on the characteristics of the user traffic but on packet loss probability on a link and the path length. The analytical calculations are lengthy but straightforward, so we omit them in this presentation.

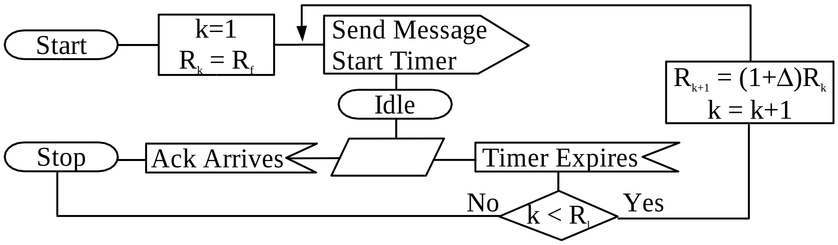

Fig. 5. A flowchart of the retransmission algorithm.

Control Message Retransmissions. We recall briefly the concept of CMR and point out the influencing parameters. The reservation process of a sender issues a control message and sets the retransmission timer. The receiver of this message is required to immediately return an acknowledgement. If the sender does not receive an acknowledgement before the timer expires, the control message is retransmitted. The retransmission timer value R_k depends on the k^{th} retransmission interval: $R_k = (1 + \Delta)^{k-1} \cdot R_f$ for $1 \leq k < R_l$. R_k scales linearly with the rapid retransmission interval R_f. An exponential backoff is applied and Δ (we use $\Delta = 1$) governs the speed at which the sender increases the timer value. This avoids unnecessary retransmissions due to links with long transmission delays. The rapid retry limit R_l is an upper bound on the number of control message transmissions. A flowchart of the algorithm is depicted in Figure 5. This concept reduces only the response time of RSVP but it does not yield reliable communication. The parameter $R_l = 1$ corresponds to signaling without CMR. In case that no acknowledgement returns, the sender tries again after the refresh interval R. If a node has not received an update message after $3 \cdot R$, it faces a soft state timeout and sends a TEARDOWN control message to indicate the end of the session to its neighboring nodes.

Endpoint versus Common Control. In Boomerang or YESSIR, only the endpoints (sender, receiver) trigger control messages that are also processed in the other nodes. In RSVP, all participating routers control the connection, i. e. they do not only forward the control messages when they arrive, they also create refresh (PATH, RESV) messages when they do not receive them in time. We call the first approach “endpoint control” (EC) and the second one “common control” (CC). With CC and CMR, the ACKs are created and returned by neighboring hosts and not as under EC by the receiving peer over many intermediate hops. CC seems to make a reservation more robust against loss of control messages, especially in combination with CMR.

One-Pass versus Two-Pass. In RSVP, OPWA is used for establishing a connection which is in fact a two-pass approach: One pass is needed for setting up the PATH states in the routers and one pass back is required for setting up the reservations. The same holds for BGRP's PROBE and GRAFT messages. With Boomerang and YESSIR this is different. The reservation is done with the first pass from the sender to the receiver and a successful session setup may be notified to the source or not. Therefore, this is a true one-pass approach. With two-pass, the signaling takes on average twice as long as with one-pass. This is a relatively trivial result. To simplify the analysis, we concentrate only on the one-pass approach. As a consequence, the results for RED in our study must be doubled in case of RSVP OPWA.

Network Parameters. The effects of the retransmission timers depend certainly on the networking scenario. We make the following assumptions. We set the packet transmission delay per link to 10 milliseconds. The packet loss probability p_l on a single link influences the system as well as the number of links n in the reservation path. Therefore, we conduct studies varying theses parameters. If theses parameters are constant, we assume a path length of $n = 10$ hops and a link packet loss probability of $p_l = 10^{-2}$ which occurs in congested networks. For $n = 10$ hops, this yields an end-to-end packet loss probability of about 10%. Transatlantic routes can have more than 20 hops and the overall packet loss probability is sometimes more than 25%. We observed these values with the statistic tool for UDP traffic in realaudio or realvideo. Especially under these circumstances, reservations are crucial for real-time applications.

3.2 Reservation Establishment Delay

We are interested in the influence of the refresh interval R, the rapid retry limit R_l, and the control option on the RED.

First, we study the configuration EC without CMR as we find it in YESSIR. In Figure 6 the mean of the RED (E[RED]) is shown for a fixed path length ($n = 10$). For small link loss probabilities, the difference for various R is negligible. E[RED] rises with increasing loss rates and the difference between various values for R becomes visible. For high packet loss probabilities ($p_l = 10^{-2}$) the refresh interval R dominates RED almost linearly since R is several orders of magnitude larger than the packet transmission delay.

We set $p_l = 10^{-2}$ and observe the reservation for different path lengths. Figure 7 shows that the RED grows linearly with the path length. At first sight this seems to be a consequence of the summation of link transmission times but this is not the case because the size of a round trip time is in the order of hundred milliseconds. This phenomenon is rather due to the fact that a longer route exhibits a larger end-to-end loss probability. Because of the high loss probabilities, the influence of the retransmission timer R is linear. In the following studies, we set R to 30 seconds.

The CMR option has been standardized [6] to reduce the signaling delay for RSVP in lossy networks. The retransmission interval R_f influences the retransmission times at most linearly and we set it to 0.5 seconds. Figure 8 shows that CMR greatly reduce the response time of the system when we compare this alternative with Figure 6. For $p_l = 10^{-3}$ E[RED] is still negligible. CMR reduce the influence of the refresh interval R and

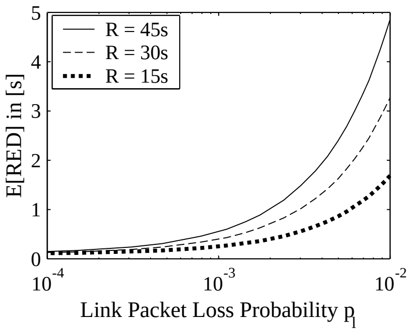

Fig. 6. The impact of R on E[RED] ($n = 10$, EC, no CMR).

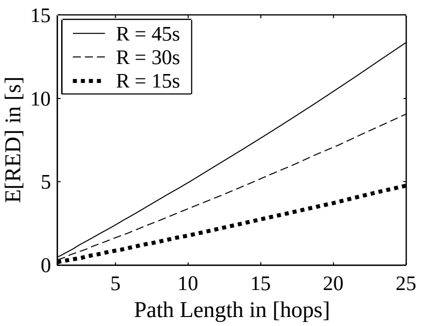

Fig. 7. The impact of R on E[RED] ($p_l = 10^{-2}$, EC, no CMR).

yield shorter RED. Even a single retransmission ($R_l = 2$) reduces E[RED] from about 3 to 0.5 seconds for $p_l = 10^{-2}$. For large loss rates the effect of the rapid retry limit R_l can be well observed. In Figure 9 ($p_l = 10^{-2}$) we observe that E[RED] increases like in Figure 7 about linearly with the length of the reservation path. However, the absolute delay for CMR ($R_l = 3$) is less than 10% compared to the result without CMR. Therefore, CMR is even more important for long paths since the absolute saved delay is larger. In the following experiments, we set $R_l = 3$.

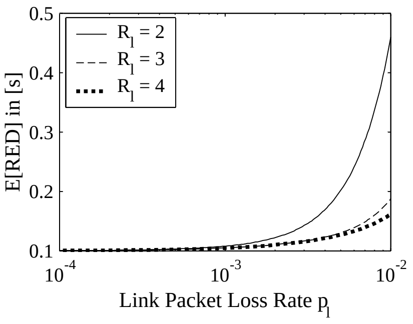

Fig. 8. The impact of R_l on E[RED] ($n = 10$, $R = 30s$, EC, CMR).

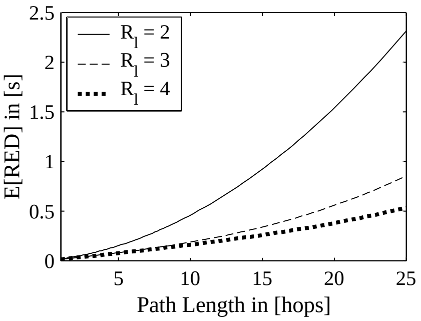

Fig. 9. The impact of R_l on E[RED] ($p_l = 10^{-2}$, $R = 30s$, EC, CMR).

So far, we have considered only EC: When a control message is lost on the way from sender to receiver, the next one is triggered again at the initial node. With CC, the nodes are more active: The last node, that has received an initial PATH or RESV control messages, generates a refresh messages by itself by latest after R time, i. e. the remaining distance to the receiver is shorter than with EC. However, Figure 10 shows

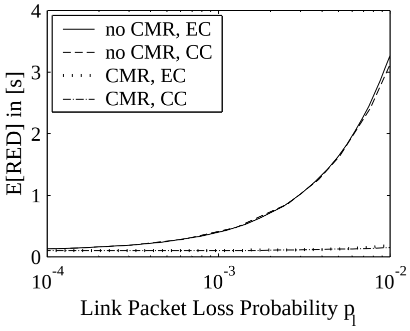

Fig. 10. The impact of the control and CMR option on E[RED] ($n = 10$, $R = 30s$, $R_l = 3$).

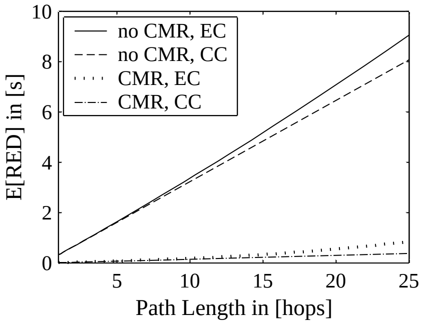

Fig. 11. The impact of the control and CMR option on E[RED] ($p_l = 10^{-2}$, $R = 30s$, $R_l = 3$).

that this has no impact: Without CMR, CC is hardly better than EC. In contrast, the effect of CMR is evident: With CMR, E[RED] stays very small while without CMR, E[RED] rises notably. In case of CC we have fewer losses between CMR peers (link loss probabilities) than for EC (end-to-end loss probabilities). Therefore, we can see a difference for high loss rates also between EC and CC with CMR. In case of CMR, E[RED] stays below one second (cf. Figure 11) which means that most of the delay is produced by the link transmission delay ($t_l = 10$ milliseconds) and that the delay due to the refresh interval R is minimized.

3.3 Reservation Teardown Delay

Reserved but unused resources can not be allocated to other connections until their reservation is torn down. The resources are not profitable for that time, so it is important to keep the RTD small. The RTD increases if a TEARDOWN message is lost on the way from the sender to receiver. Due to the soft state concept, an intermediate node tears down the reservation when its cleanup timer expires after L time. If only the endpoints control the session, all nodes time out after L since the terminating endpoint refrains from sending refresh messages. If the reservation is under CC (like in RSVP), the second router is not refreshed by the sender but it keeps on generating refresh messages autonomously in periodic intervals. When it times out after L time, its TEARDOWN message can be lost as well. This leads in the worst case to a maximum RTD of $n \cdot L$ for a reservation length of n links.

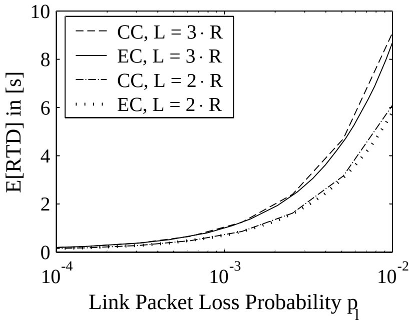

Fig. 12. The impact of L and the control option on E[RTD] ($n = 10$, $R = 30s$, no CMR).

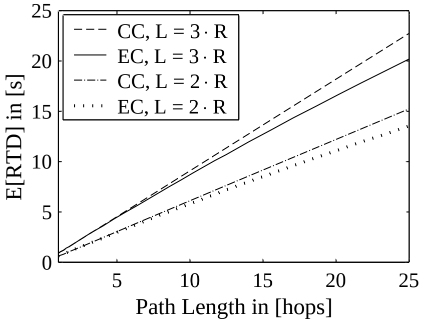

Fig. 13. The impact of L and the control option on E[RTD] ($p_l = 10^{-2}$, $R = 30s$, no CMR).

However, Figure 12 shows that the control option has no impact on E[RTD]. The influence of L dominates E[RTD] for high link loss probabilities by a linear law. The same phenomenon can be observed in Figure 13. Here, it is visible that CC is clearly worse than EC. But the value L for the expiration timer has still greater impact. In the following, we set $L = 3 \cdot R$.

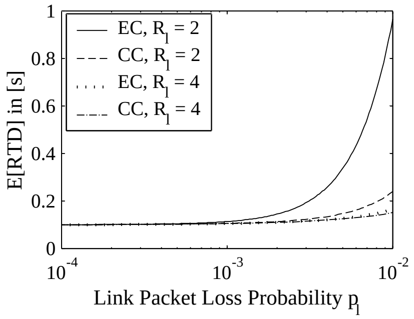

Fig. 14. The impact of the CMR parameter R_l and the control option on E[RTD] ($n = 10$, $R = 30s$, $L = 3$, CMR).

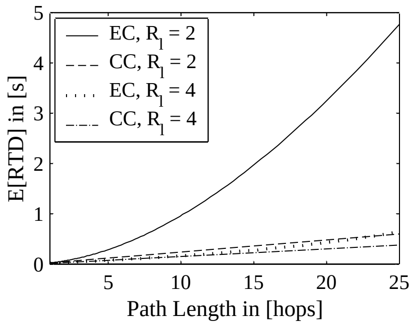

Fig. 15. The impact of the CMR parameter R_l and the control option on E[RTD] ($p_l = 10^{-2}$, $R = 30s$, $L = 3$, CMR).

We investigate the influence of the CMR and its parameters on E[RTD]. With CMR, up to R_l TEARDOWN messages are sent repeatedly until an acknowledgement returns. A comparison of Figure 14 and Figure 12 shows that this reduces E[RTD] notably. Especially CC profits from CMR: Their E[RTD] is now shorter than for EC. A rapid retry limit of $R_l = 2$ already suffices to keep E[RTD] small, however EC with $R_l = 4$

exhibits an excellent performance as well. Figure 15 shows that a single retransmission reduces E[RTD] for EC to 25% compared to without CMR (cf. Figure 13). But this combination performs still quite poorly in contrast to the other configurations.

3.4 Concluding Remarks

If packet loss rates are high, CMR are a powerful method to reduce the response time in sigaling. CC is an alternative to EC to improve the session stability in that case. In our investigations CMR turned out to be more effective than CC to reduce E[RED] and E[RTD]. The combination of both techniques is possible [6] and leads to optimum results. However, both extensions increase the software complexity of existing RSVP solutions which may have drawbacks that are not respected in this study.

4 Summary

In this paper, we gave an overview of several resource reservation protocols. We presented their basic operations with focus on their information passing concepts.

We explained RSVP and various extensions that reduce the number of refresh messages, make it more robust against packet losses and more scalable for the use in transit networks. Apart from RSVP, light-weight protocols as Boomerang and YESSIR were presented. BGRP is intended for reservation aggregation between autonomous systems. All these approaches derive the possible traffic intensity and the remaining capacity from the stored flow information. Measurement based admission control (MBAC) substitutes this by an estimation of the current network load. We also presented the principle of stateless reservation protocols which is a special MBAC approach.

In RSVP every node supporting a reservation is actively involved in keeping the reservation alive (common control). This is unlike in Boomerang or YESSIR (endpoint control) where only sender and receiver trigger refresh messages. Recently, an option for RSVP control message retransmission (CMR) was created to make RSVP more responsive in networking scenarios with high packet loss probabilities.

We investigated these protocol concepts that are basic features for general signaling protocols. They have an influence on the reservation establishment delay (RED) and on the reservation teardown delay (RTD). Their impact depends both on the packet loss probability of a single link as well as on the number of hops in the reservation path. For small loss probabilities ($< 10^{-3}$), however, their effect is negligible. In networking scenarios with high loss probabilities like wireless networks, the performance gain by CMR is considerable whereas the alternative endpoint or common control plays only a marginal role.

References

1. Chiueh, T., Neogi, A., Stirpe, P.: Performance Analysis of an RSVP-Capable Router. In: 4th Real-Time Technology and Applications Symposium. (1998)
2. Karsten, M., Schmitt, J., Steinmetz, R.: Implementation and Evaluation of the KOM RSVP Engine. In: Infocom, IEEE (2001) 1290–1299

3. Fehér, G., Németh, K., Czslényi, I.: Performance Evaluation Framework for IP Resource Reservation Signalling. In: 8^{th} IFIP Workshop of Performance Modelling and Evaluation of ATM & IP Networks. (2000)
4. Pan, P., Schulzrinne, H.: Processing Overhead Studies in Resource Reservation Protocols. In: 17^{th} International Teletraffic Congress. (2001)
5. Eramo, V., Mocci, U., Fratini, M., Listanti, M.: Reliability Evaluation of RSVP. In: 8^{th} International Telecommunication Network Planning Symposium (Networks 98). (1998)
6. Berger, L., Gan, D.H., Swallow, G., Pan, P., Tommasi, F., Molendini, S.: RFC2961: RSVP Refresh Overhead Reduction Extensions. http://www.ietf.org/rfc/rfc2961.txt (2001)
7. Schulzrinne, H., Rosenberg, J.: Internet Telephony: Architecture and Protocols - an IETF Perspective. Computer Networks **31** (1999) 237–255
8. Braden, B., et al.: RFC2205: Resource ReSerVation Protocol (RSVP) - Version 1 Functional Specification. ftp://ftp.isi.edu/in-notes/rfc2205.txt (1997)
9. Wroclawski, J.: RFC2210: The Use of RSVP with IETF Integrated Services. ftp://ftp.isi.edu/in-notes/rfc2210.txt (1997)
10. Pan, P., Schulzrinne, H.: Staged Refresh Timers for RSVP. In: 2^{nd} Global Internet Conference. (1997)
11. Baker, F., Iturralde, C., Le Faucheur, F., Davie, B.: Rfc3175: Aggregation of RSVP for IPv4 and IPv6 Reservations. http://www.ietf.org/rfc/rfc3175.txt (2001)
12. Rosen, E.C., Viswanathan, A., Callon, R.: RFC3031: Multiprotocol Label Switching Architecture. http://www.ietf.org/rfc/rfc3031.txt (2001)
13. Menth, M.: A Scalable Protocol Architecture for End-to-End Signaling and Resource Reservation in IP Networks. In: 17^{th} International Teletraffic Congress. (2001) 211–222
14. Menth, M., Hauck, N.: A Graph-Theoretical Concept for LSP Hierarchies. Technical Report, No. 287, University of Würzburg, Institute of Computer Science (2001)
15. Fehér, G., Németh, K., Maliosz, M., Czslényi, I., Bergkvist, J., Ahlard, D., Engborg, T.: Boomerang - A Simple Protocol for Resource Reservation in IP Networks. In: IEEE Workshop on QoS Support for Real-Time Internet Applications, Vancouver, Canada (1999)
16. Pan, P., Schulzrinne, H.: YESSIR: A Simple Reservation Mechanism for the Internet. Computer Communication Review **29** (1999)
17. Schulzrinne, H., Casner, S., Frederick, R., Jacobson, V.: RFC1889: RTP - A Transport Protocol for Real-Time Applications. ftp://ftp.isi.edu/in-notes/rfc1889.txt (1996)
18. Pan, P., Schulzrinne, H.: BGRP: A Tree-Based Aggregation Protocol for Inter-domain Reservations. Journal of Communications and Networks **2** (2000) 157–167
19. Breslau, L., Knightly, E.W., Shenker, S., Zhang, H.: Endpoint Admission Control: Architectural Issues and Performance. In: ACM SIGCOMM. (2000)
20. Breslau, L., Jamin, S., Shenker, S.: Comments on the Performance of Measurement-Based Admission Control Algorithms. In: Infocom. (2000) 1233–1242
21. Más, I., Karlsson, G.: PBAC: Probe-Based Admission Control. In: 2^{nd} International Workshop on Quality of future Internet Services (QofIS2001). (2001)
22. Stoica, I., Zhang, H.: Providing Guaranteed Services Without Per Flow Management. Computer Communication Review **29** (1999)
23. Almesberger, W., Ferrari, T., Le Boudec, J.Y.: SRP: a Scalable Resource Reservation for the Internet. In: IFIP 6^{th} International Workshop on Quality of Service (IWQoS'98). (1998)
24. Eriksson, A., Ghermann, C.: Robust and Secure Light-weight Resource Reservation of Unicast IP Traffic. In: IFIP 6^{th} International Workshop on Quality of Service (IWQoS'98). (1998)