

Internet Engineering Task Force
Internet Draft
[draft-ietf-avt-aggregation-00.txt](#)
May 6, 1998
Expires: November 6, 1998

AVT Working Group
J.Rosenberg, H.Schulzrinne
Bell Labs/Columbia U.

An RTP Payload Format for User Multiplexing

STATUS OF THIS MEMO

This document is an Internet-Draft. 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''.

To learn the current status of any Internet-Draft, please check the ``1id-abstracts.txt'' listing contained in the Internet-Drafts Shadow Directories on ftp.is.co.za (Africa), nic.nordu.net (Europe), munnari.oz.au (Pacific Rim), ds.internic.net (US East Coast), or ftp.isi.edu (US West Coast).

Distribution of this document is unlimited.

ABSTRACT

This memo describes an RTP payload format for multiplexing data from multiple users into a single RTP packet. Such multiplexing is especially useful for transporting voice data between Internet telephony gateways. It causes significant reductions in header overheads and improves scalability.

1 Introduction

Internet telephone gateways (ITGs) allow a public switched telephony user (PSTN) user to contact another PSTN user, with the long distance portion of the call routed over the Internet. Such a scenario is depicted in Figure 1.

size of the payloads to increase efficiency. However, if multiplexing is performed, the packet payload increases. This would allow smaller packetization delays to be used as the number of multiplexed users increases.

Yet another benefit is the reduction in interrupt processing at the receiving ITG. Whenever a packet arrives at the gateway, the operating system must perform a context switch into the kernel and process the packet. Without aggregation, the frequency of these interrupts increases linearly with the number of users. However, with aggregation, the packet rate does not increase as more users are added, and thus the interrupt rate stays constant. This improves scalability.

The main drawback to multiplexing is the increase in store-and-forward delays. These delays are often most problematic in end systems, which are typically connected via dialup modems. In this application, ITGs are likely connected to the Internet via higher rate connections, such as a T1. Assuming 24 users are multiplexed into a packet, using the same codec and packetization delays as above, store and forward delays are only 3.7 ms, which are relatively low compared to typical queuing and propagation delays.

This draft describes an RTP payload format for supporting multiplexing of many users into a single RTP session. It is based on an earlier expired Internet Draft [4]

2 Payload Format

The format of RTP packets with multiplexed users is given in Figure 2.

Figure 2: Packet format for multiplexing

All fields of the RTP header except the timestamp, marker bit and SSRC maintain their current definition. Payload Type: The payload type field designates the RTP packet as a multiplexed payload. The payload type value is chosen dynamically and the binding to this format is conveyed via non-RTP means, such as SDP [1]. Timestamp: This protocol requires that all multiplexed streams in one packet have the same clock rate (i.e., sampling rate for audio) and generate media frames at integer multiples of a common frame duration. It is possible, for example, that a set of users generates a packet every 10 ms, while others generate packets at intervals of 20 and 30 ms, but all frame generation instants must be multiples of this 10 ms interval. (We expect this to be the common case for ITGs. Otherwise, each media frame would require a timestamp offset or an independent

Figure 3: Multiplexing header

Each media data frame that is being multiplexed is associated with a header, 16 bits (optionally, 32) in length. These headers indicate, among other things, the length of the media data for user *i* present in the packet (call this length *L_i*). To determine the number of users present, user headers are read until the sum of the lengths *L_i* encoded in them is equal to the remaining RTP packet length.

The format of each user header is shown in Fig. 4:

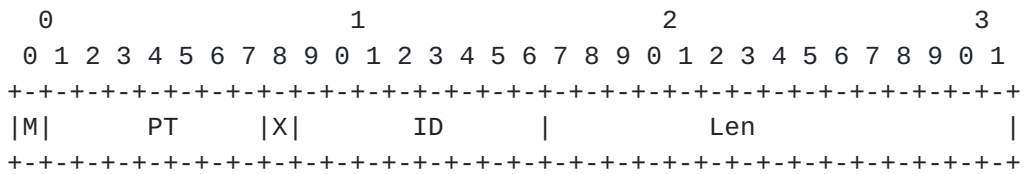


Figure 4: Format of user header

The fields have the following meaning: M: The marker bit has the same definition as in [5], but applies only to the particular media frame. PT: This is the payload type corresponding to the media frame. It is anticipated that this field will also generally be sufficient to determine the length of the data for the user. For example, a PT value of 26 might indicate that the payload is 10 bytes of G.729 compressed speech. The method by which the payload types are bound to particular codec types and length values is outside the scope of this document. It is envisioned, however, that signaling protocols such as SIP [6] together with a session description protocol such as SDP or H.323 could be used for this purpose. L: The length flag; if set to one, the 16-bit length field is present. ID: The ID field is a 7-bit key used to identify the user to the remote end system. It associates the media data with the user identifiers contained in signaling protocols. For example, one gateway may set up a call through another, indicating that user 38 is calling PSTN number (800) 555-1212. In order to know which user data corresponds to user 38, the ID value in the header would be set to 38. The value of zero is reserved (see below). Length: In cases where the payload type is not sufficient to determine the length of the user media data, a 16-bit length field can be included. This field contains the length of the user media data, in bytes. It is only present if the L bit has a value of one.

The user header fields appear immediately after the RTP header, independent of their size. If the last user header does not end on a 32-bit boundary, an additional user header with a distinguished user ID

of zero is added. The other fields in that padding header MUST be zero.

The user media frames follow after the multiplexing header, packed without any padding or headers between. The first user data field contains the data for the user defined by the first user header field, and so on.

Note that not every RTP packet has to contain media data frames from all active users. For example, if the packetization interval for a particular user is twice that of another user, only every other packet will contain media frames from both users.

3 Multiple Packets

In some cases, it may not be possible to multiplex all users together into the same packet. This could be because the timestamps are not all the same, or because the transmitter wants to restrict the packet size. We define a set of users whose data are placed into the same packet as a user group. Each user group is sent in a separate packet.

To distinguish packets from different user groups, the SSRC field is used. The SSRC field is always the same across two packets from the same user group, and always different between two packets in different user groups. Packets from different user groups have different sequence number spaces as well. Each user group can essentially be considered a different virtual user, and it is for this reason that we use the SSRC field to identify them.

Note that the users are only identified by their ID field, not by the SSRC value of their user group. This allows a user to migrate from user group to user group. Such migration may be necessary if the user chooses to change media coder, which would affect its timestamp frequency. The size of the ID field imposes a limitation of 127 users in a multiplexed RTP session. Additional RTP sessions would need to be opened if this number is not sufficient. Users may not migrate from session to session, since ID's are only guaranteed unique within a single RTP session.

4 Impact on QoS

In this section, we discuss the impact of the multiplexing protocol on end to end QoS.

4.1 Loss

At first glance, it may seem that multiplexing linearly increases the impact of packet loss on per user loss rates (assuming Bernoulli

packet losses). However, this is not the case. The fact that there are more users per packet is exactly offset by the decrease in the number of packets transmitted, yielding no difference between multiplexing and non-multiplexing. Mathematically, one can consider a stream of packets as a Bernoulli process X_i . In the case of multiplexing, media frames from a particular user are present in each packet X_i , whereas without, media frames are only present in some subset, X_{Nj} , $j=0..inf$. However, the average value of the Bernoulli process and its subsampled version are identical, so the observed loss rate is unchanged.

In reality, losses aren't Bernoulli. However, multiplexing is likely to reduce losses on the Internet, for several reasons. First, the improved efficiencies mean the overall bitrate for the stream is less. This has the effect of helping reduce congestion, which causes losses in the first place. Secondly, many routers drop packets not because of link congestion and buffer overflow, but because of processing overload. A burst of small packets can overwhelm the processors on a typical router, causing loss. Thus, a critical characteristic of a router is the number of packets per second it can process. The multiplexing protocol has the advantage of keeping the packet rates low, which can help reduce process-based losses in routers. Finally, many routers use packet based queues, not byte based. Thus, N packets, each of size $1/N$, is N times as much buffer resource occupancy of 1 packet of size 1. Multiplexing therefore helps improve buffer usage as well.

Put together, we expect multiplexing to therefore improve end to end observed loss rates.

4.2 Delay

One of the costs of multiplexing is delay - not packetization delay (which can be effectively reduced by multiplexing) but store and forward delays. These delays are directly proportional to link bandwidths. For modem speeds, the increased delays would be unacceptable. However, multiplexed voice is likely to be generated by gateways with much higher link speeds (T1, for example). Assuming all links are T1, the delay for storing and forwarding 20 multiplexed users over 5 hops is 2.2ms (assuming again G.729 with 30 ms packetization delays), which is far less than other delays in the system.

Users of the multiplexing protocol with concerns about delays can always opt to use as few users per user group as they feel comfortable with.

4.3 Jitter

The amount of jitter introduced by the multiplexing protocol depends entirely on its usage. A system which uses a common payload type and packetization delay among all users in a user group will suffer no additional jitter through multiplexing.

However, schemes which involve users changing user groups and payload types, and which involve mixing together different frame sizes per packet, may result in additional jitter. Once again, it is up to the administrator to make the appropriate tradeoff.

5 Security Considerations

There are no security considerations beyond those addressed in RTP itself. The multiplexing protocol can make use of whatever encryption and authentication schemes are present in RTP, SIP, H.323 or other relevant protocols.

6 Open Issues

There are a few open issues:

1. The multiplexing gain is based entirely on the assumption of synchronized packet generation among some group of users. It is possible to achieve gains without this assumption by introducing timestamp offsets in the user header. The result is an increase in jitter and header overheads, and for this reason we have not taken this route. However, how valid is our assumption of synchronization for gateways?
2. Should the length field be made mandatory?
3. What support is needed for this in H.323 and SIP?
4. How does this relate to the MPEG4 Flexmux encapsulation?

7 Conclusion

This document has specified an RTP payload format allowing multiple user media frames to reside in an RTP packet. This multiplexing is very useful for ITG-to-ITG communications, where it can reduce packet header overhead and improve gateway scalability.

8 Full Copyright Statement

Copyright (C) The Internet Society (1998). All Rights Reserved.

This document and translations of it may be copied and furnished to others, and derivative works that comment on or otherwise explain it or assist in its implementation may be prepared, copied, published and

distributed, in whole or in part, without restriction of any kind,

J.Rosenberg, H.Schulzrinne

[Page 8]

provided that the above copyright notice and this paragraph are included on all such copies and derivative works.

However, this document itself may not be modified in any way, such as by removing the copyright notice or references to the Internet Society or other Internet organizations, except as needed for the purpose of developing Internet standards in which case the procedures for copyrights defined in the Internet Standards process must be followed, or as required to translate it into languages other than English.

The limited permissions granted above are perpetual and will not be revoked by the Internet Society or its successors or assigns.

This document and the information contained herein is provided on an "AS IS" basis and THE INTERNET SOCIETY AND THE INTERNET ENGINEERING TASK FORCE DISCLAIMS 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."

9 Authors' Addresses

Jonathan Rosenberg
Rm. 4C-526
Bell Laboratories, Lucent Technologies
101 Crawfords Corner Rd.
Holmdel, NJ 07733
electronic mail: jdrosen@bell-labs.com

Henning Schulzrinne
Dept. of Computer Science
Columbia University
1214 Amsterdam Avenue
New York, NY 10027
USA
electronic mail: schulzrinne@cs.columbia.edu

10 Bibliography

- [1] M. Handley and V. Jacobson, SDP: session description protocol, Request for Comments (Proposed Standard) [2327](#), Internet Engineering Task Force, Apr. 1998.
- [2] H. Schulzrinne, S. Casner, R. Frederick, and V. Jacobson, RTP: a transport protocol for real-time applications, Request for Comments (Proposed Standard) [1889](#), Internet Engineering Task Force, Jan. 1996.

- [3] International Telecommunication Union, Coding of speech at 8 kbit/s using conjugate-structure algebraic-code-excited linear-prediction, Recommendation G.729, Telecommunication Standardization Sector of ITU, Geneva, Switzerland, Mar. 1996.
- [4] J. Rosenberg and H. Schulzrinne, Issues and options for an aggregation service within RTP, Internet Draft, Internet Engineering Task Force, Nov. 1996. Work in progress.
- [5] H. Schulzrinne, RTP profile for audio and video conferences with minimal control, Request for Comments (Proposed Standard) [1890](#), Internet Engineering Task Force, Jan. 1996.
- [6] M. Handley, H. Schulzrinne, and E. Schooler, SIP: Session initiation protocol, Internet Draft, Internet Engineering Task Force, Mar. 1998. Work in progress.