Network Working Group   C. Perkins
Internet-Draft   University of Glasgow
Expires: July 14, 2004   January 14, 2004

                 RTP Retransmission Using Reactive FEC
                 draft-perkins-avt-rtp-reactive-fec-00.txt

Status of this Memo

Copyright Notice

Abstract

   This memo describes how the RTP Payload Format for Generic Forward
   Error Correction can be combined with the Extended RTP Profile for
   RTCP-based Feedback to provide a solution for retransmission of lost
   RTP packets. Such a retransmission format is expected to be useful to
   improve the reliability of real-time applications with relaxed delay
   bounds, for example non-interactive streaming audio/video.

1. **Introduction**

   The quality of real-time audio/video services on the Internet is
   often less than might be desired. In large part, this is due to
   packet loss. As discussed in [6], several techniques may be used to
   correct for the effects of packet loss. These include forward error

correction (FEC) and retransmission. Several FEC solutions exist
[5],[2] and can be used with RTP-based applications. This memo
specifes a retransmission format that can be used with RTP.

The use of retransmissions in RTP as a repair method for streaming
media is appropriate in those scenarios with relaxed delay bounds and
where full reliability is not a requirement. More specifically, RTP
retransmission allows to trade-off reliability vs. delay, i.e. the
endpoints may give up retransmitting a lost packet after a given
buffering time has elapsed. Unlike TCP there is thus no head-of-line
blocking caused by RTP retransmissions. The implementer should be
aware that in cases where full reliability is required or higher
delay and jitter can be tolerated, TCP or other transport options are
more appropriate, and should be considered.

2. Terminology

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT",
"SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this
document are to be interpreted as described in RFC 2119 [1].

3. Requirements

The RTP retransmission scheme defined in this document is designed to
fulfil the following set of requirements:

1. It MUST NOT break general RTP and RTCP mechanisms.

2. It MUST be suitable for unicast and small multicast groups.

3. It MUST work with RTP mixers and translators.

4. It MUST work with all known payload types.

5. It MUST NOT prevent the use of multiple payload types in a
    session.

6. In order to support the largest variety of payload formats, the
    receiver MUST be able to determine how many and which RTP packets
    were lost as a result of a gap in received RTP sequence numbers.

4. Requesting Retransmission

In order to perform retransmission of RTP data packets it is
necessary for a receiver to send a request to the sender, indicating
the packets which should be retransmitted. The extended RTP profile
for RTCP-based feedback, denoted RTP/AVPF, [4] allows receivers to

request retransmission of lost packets using NACK messages. Such
retransmission requests MAY be sent in both unicast and small group
multicast sessions.

Retransmission requests SHOULD be sent according to the timing rules
defined in [4]. These rules may limit the number of packet loss
events that can be repaired. The receivers SHOULD NOT send a
retransmission request as soon as they detect a missing sequence
number, but SHOULD instead add some extra delay to compensate for
packet reordering. This extra delay may, for example, be based on
past observations of the packet reordering.

The receiver should generally assess whether the retransmitted packet
would still be useful at the time it is received. The timestamp of
the missing packet can be estimated from the timestamps of packets
preceding and/or following the sequence number gap caused by the
missing packet in the original stream. In most cases, some form of
linear estimate of the timestamp is good enough.

To increase the robustness to the loss of a NACK packet or of a
retransmitted data packet, a receiver may send a new NACK for the
same packet. This is referred to as multiple retransmissions. Before
sending a new NACK for a missing packet, the receiver SHOULD rely on
a timer to be reasonably sure that the previous retransmission
attempt has failed in order to avoid unnecessary retransmissions.

5. Retransmission of RTP Packets Using Reactive FEC

The RTP payload format defined in [2] allows a sender associate an
RTP session containing parity FEC packets with a standard RTP
session. The amount of FEC data generated may be varied according to
the amount of packet loss observed, and there is no need to signal
changes in the FEC pattern to receivers out-of-band (the information
needed to decode the FEC is included within each packet).

A sender using the FEC format defined in [2] MAY generate FEC packets
continually, according to some pattern. This will allow receivers to
repair a subset of packet losses without having to contact the
sender. Alternatively a sender MAY generate FEC packets only when
requested to do so by a receiver, for example on receipt of NACK
packets sent using the RTP/AVPF profile. Generation of FEC packets on
demand is termed Reactive FEC.

When using reactive FEC to repair a single packet loss, the parity
FEC packets MAY be generated with an "SN base" which is numerically
close to that of the lost packet, and with a single bit of the "mask"
field set (see Section 6.2 of [2]). Since only a single packet is
selected, the protection operation becomes equal to a bit copy, and

the FEC payload is thus equal to the original payload. A receiver of
FEC packets generated in this manner can use them to repair the
original media stream. The FEC packets are standalone, and do not
require receipt of other packets in the stream to decode.

Alternatively, if multiple retransmission requests are received, the
sender MAY generate FEC packets using the parity operation on some
set of data packets. This may be useful when sending to a small
multicast group, since a single FEC packet can be used to repair
multiple losses. Reactive FEC packets generated in this manner can
only be decoded by receivers which understand the parity FEC
operation, and which receive the appropriate subset of packets. This
adds some complexity, compared to the single repair case, but saves
bandwidth due to the reduced number of repair packets sent.

In all cases, the resulting FEC packet MUST be sent as a separate
stream, using the mechanisms defined in [2].

6. Congestion Control

RTP retransmission poses a risk of increasing network congestion. In
a best-effort environment, packet loss is caused by congestion.
Reacting to loss by retransmission of older data without decreasing
the rate of the original stream will further increase congestion.
Implementations MUST follow the recommendations below in order to use
retransmission safely.

The RTP profile under which the retransmission scheme is used MUST
define an appropriate congestion control mechanism for different
environments. Following the rules under the profile, an application
can determine its acceptable bitrate and packet rate in order to be
fair to other TCP or RTP flows.

If an RTP application uses retransmission, the acceptable packet rate
and bitrate includes both the original and retransmitted data. This
guarantees that an application using retransmission achieves the same
fairness as one that does not. Such a rule translates into the
following actions:

o If enhanced service is used, the total bitrate and packet rate
   MUST NOT exceed that of the requested service. A receiver SHOULD
   monitor the network to ensure that the requested service is
   delivered.

o In best-effort environments, senders MUST NOT send retransmission
   packets without reducing the rate of the original stream (e.g. by
   encoding the data at a lower rate) to compensate for the extra
   retransmission data.

These congestion control mechanisms should keep the packet loss rate within acceptable parameters. Packet loss is considered acceptable if a TCP flow across the same network path and experiencing the same network conditions would achieve, on a reasonable timescale, average throughput that is not less than the one the RTP flow achieves. If the packet loss rate exceeds an acceptable level, the sender MUST reduce its sending rate or, if this is not possible, stop transmission.

It is to be noted that RTP retransmission is not appropriate for sessions where reliability is more important than timeliness. In these cases, a protocol such as TCP/IP SHOULD be used.

7. Signalling using SDP

Retransmission using reactive FEC may be signalled using SDP as in the following example:

```
v=0
o=hamming 2890844526 2890842807 IN IP4 10.0.42.7
s=FEC Seminar
c=IN IP4 10.6.1.4
t=0 0
m=audio 49170 RTP/AVPF 0 98
a=rtpmap:98 parityfec/8000
a=fmtp:98 49172 IN IP4 10.6.1.4
a=rtcp-fb:0 nack
```

Note that use of RTP/AVPF with nack feedback for the original media.

8. IANA Considerations

No IANA actions are necessary.

9. Security Considerations

The security considerations of [2], [3] and [4] apply.

10. Acknowledgements

Parts of this memo are derived from an earlier retransmission format developed by Jose Rey, David Leon, Akihiro Miyazaki, Viktor Varsa and Rolf Hakenberg. Thanks are due to John Lazzaro and Jonathan Rosenberg for their comments on an early version of this draft.

Normative References

[1] Bradner, S., "Key words for use in RFCs to Indicate Requirement

Levels", [BCP 14](), [RFC 2119](), March 1997.

[2]   Rosenberg, J. and H. Schulzrinne, "An RTP Payload Format for
      Generic Forward Error Correction", [RFC 2733](), December 1999.

[3]   Schulzrinne, H., Casner, S., Frederick, R. and V. Jacobson,
      "RTP: A Transport Protocol for Real-Time Applications", [RFC
      3550](), July 2003.

[4]   Ott, J. and S. Wenger, "Extended RTP Profile for RTCP-based
      Feedback(RTP/AVPF)", [draft-ietf-avt-rtcp-feedback-07]() (work in
      progress), June 2003.

Informative References

[5]   Perkins, C., Kouvelas, I., Hodson, O., Hardman, V., Handley, M.,
      Bolot, J., Vega-Garcia, A. and S. Fosse-Parisis, "RTP Payload
      for Redundant Audio Data", [RFC 2198](), September 1997.

[6]   Perkins, C. and O. Hodson, "Options for Repair of Streaming
      Media", [RFC 2354](), June 1998.

Author's Address

   Colin Perkins
   University of Glasgow
   Department of Computing Science
   17 Lilybank Gardens
   Glasgow G12 8QQ
   United Kingdom

   EMail: csp@csperkins.org

HEREIN WILL NOT INFRINGE ANY RIGHTS OR ANY IMPLIED WARRANTIES OF
MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE.

Acknowledgment