Colin Perkins
University College London

RTP Interoperability Statement
draft-ietf-avt-rtp-interop-00.txt

Status of this memo

This document is an Internet-Draft and is in full conformance with all
provisions of Section 10 of RFC2026.

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/ietf/1id-abstracts.txt

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

Distribution of this document is unlimited.

Comments are solicited and should be addressed to the author and/or the
IETF Audio/Video Transport working group's mailing list at rem-conf@es.net.

Abstract

   It is required to demonstrate interoperability of RTP implementations
   in order to move the RTP specification to draft standard. This memo
   outlines those features to be tested, as the first stage of an
   interoperability statement.

1 Introduction

The Internet standards process [1] places a number of requirements
on a standards track protocol specification. In particular, when
advancing a protocol from proposed standard to draft standard it
is necessary to demonstrate at least two indpendent and interoperable
implementations, from different code bases, of all options and features
of that protocol. Further, in cases where one or more options or
features have not been demonstrated in at least two interoperable

implementations, the specification may advance to the draft standard
level only if those options or features are removed. The Real-time
Transport Protocol, RTP, was originally specified in RFC1889 as a
proposed standard [2]. The revision of this specification for draft
standard status is well underway, so it has become necessary to conduct
such an interoperability demonstration.

This memo describes the set of features and options of the RTP
specification as a basis for this demonstration. Due to the nature of RTP
there are necessarily two types of test feature described: those which
directly affect the interoperability of implementations at a "bits on the
wire level", and those which affect scalability and safety of the protocol
but do not directly affect interoperability. Two related memos [4,5]
describe a testing framework which may aid with interoperability testing.

This memo is for information only and does not specify a standard
of any kind.

2 Features and options required to demonstrate interoperability

In order to demonstrate interoperability it is required to produce
a statement of interoperability for each feature noted below. Such
a statement of interoperability should note the pair of implementations
tested, and a pass/fail statement for each feature. It is not expected
that every implementation will implement every feature, but each feature
needs to be demonstrated by some pair of applications.

Note that some of these tests depend on the particular profile used,
or upon options in that profile. For example, it will be necessary
to test audio and video applications operating under [3] separately.

  1. Interoperable exchange of data packets using the basic RTP header
     with no extension, padding or CSRC list

  2. Interoperable exchange of data packets which use padding

  3. Interoperable exchange of data packets which use a header extension.
     There are three possibilities here: a) if both implementations
     use a header extension in the same manner, it should be verified
     that the receiver correctly receives the information contained
     in the extension header; b) If the sender uses a header extension
     and the receiver does not, it should be verified that the receiver
     ignores the extension; c) If neither implementation implements
     an extended header, this test is considered a failure.

  4. Interoperable exchange of data packets using the marker bit as
     specified in the profile.

  5. Interoperable exchange of data packets using the payload type
     field to differentiate multiple payload formats according to
     a profile definition.

  6. Interoperable exchange of data packets containing a CSRC list

  7. Interoperable exchange of sender report packets when the receiver
     of the sender reports is not also a sender (ie: sender reports
     which only contain sender info, with no report blocks).

  8. Interoperable exchange of sender report packets when the receiver
     of the sender reports is also a sender (ie: sender reports
     which contain one or more report blocks).

  9. Interoperable exchange of receiver report packets.

 10. Interoperable exchange of receiver report packets when not receiving
     data (ie: the empty receiver report which hs to be sent first
     in each compound RTCP packet when no-participants are transmitting
     data).

 11. Interoperable exchange of source description packets containing
     a CNAME item.

 12. Interoperable exchange of source description packets containing
     a NAME item.

 13. Interoperable exchange of source description packets containing
     a EMAIL item.

 14. Interoperable exchange of source description packets containing
     a PHONE item.

 15. Interoperable exchange of source description packets containing
     a LOC item.

 16. Interoperable exchange of source description packets containing
     a TOOL item.

 17. Interoperable exchange of source description packets containing
     a NOTE item.

 18. Interoperable exchange of source description packets containing
     a PRIV item.

 19. Interoperable exchange of BYE packets.

 20. Interoperable exchange of BYE packets containing multiple SSRCs.

 21. Interoperable exchange of BYE packets containing the optional

reason for leaving text.

  22. Interoperable exchange of application defined RTCP packets. As
      with the RTP header extension this test takes two forms: if
      both implementations implement the same application defined packet
      it should be verified that those packets can be interoperably

      exchanged. If only one implementation uses application defined
      packets, it should be verified that the other implementation
      can receive compound RTCP packets containing an APP packet whilst
      ignoring the APP packet. If neither implementation implements
      APP packets this test is considered a failure.

  23. Interoperable exchange of encrypted RTP packets using DES encryption
      in CBC mode.

  24. Interoperable exchange of encrypted RTCP packets using DES encryption
      in CBC mode.

[3] Features and options relating to scalability

In addition to the basic interoperability tests, RTP includes a number
of features relating to scaling of the protocol to large groups.
Since these features are those which have undergone the greatest
change in the update of the RTP specification, it is considered important
to demonstrate their correct implementation. However, since these
changes do not affect the bits-on-the-wire behaviour of the protocol,
it is not possible to perform a traditional interoperability test.
As an alternative we require that multiple independent implementations
complete the following demonstrations.

  1. Demonstrate correct implementation of basic RTCP transmission
     rules: periodic transmission of RTCP packets at the minimum
     (5 second) interval and randomisation of the transmission interval.

  2. Demonstrate correct implementation of the RTCP step join backoff
     algorithm as a receiver.

  3. Demonstrate correct implementation of the RTCP step join backoff
     algorithm as a sender.

  4. Demonstrate correct steady state scaling of the RTCP interval
     acording to the group size.

  5. Demonstrate correct steady state scaling of the RTCP interval
     acording to the group size with compensation for the number of
     senders.

6. Demonstrate correct implementation of the RTCP reverse reconsideration algorithm.

7. Demonstrate correct implementation of the RTCP BYE reconsideration algorithm.

8. Demonstrate correct implementation of the RTCP member timeout algorithm.

9. Demonstrate random choice of SSRC.

10. Demonstrate correct implementation of the SSRC collision detection algorithm.

4 Author's Address

Colin Perkins
Department of Computer Science
University College London
Gower Street
London WC1E 6BT
United Kingdom

Email: c.perkins@cs.ucl.ac.uk

5 References

[1] S. Bradner, ``The Internet Standards Process -- Revision 3'',
    RFC2026, Internet Engineering Task Force, October 1996.

[2] H. Schulzrinne, S. Casner, R. Frederick and V. Jacobson, ``RTP:
    A Transport Protocol to Real-Time Applications'', RFC1889, Internet
    Engineering Task Force, January 1996.

[3] H. Schulzrinne, ``RTP Profile for Audio and Video Conferences
    with Minimal Control'', draft-ietf-avt-profile-new-05.txt, February
    1999.

[4] C. Perkins, ``RTP Testing Strategies'', draft-ietf-avt-rtptest-00.txt,
    June 1999.

[5] J. Rosenberg, ``Conformance tests for RTP Scalability Algorithms'',
    draft-ietf-avt-rtcptest-01.txt, June 1999.