Payload
Internet Draft
Intended status: Informational
Expires: August 2011

C. Bergeron, R. Fracchia, B. Gadat
Thales Communications

February 21, 2011

RTP Payload Format for Reed-Solomon FEC
draft-bergeron-payload-rtpfec-rs-00.txt

Status of this Memo

This Internet-Draft is submitted to IETF in full conformance with the provisions of BCP 78 and BCP 79.

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.

This Internet-Draft will expire on August 21, 2011.

Copyright Notice

Copyright (c) 2011 IETF Trust and the persons identified as the document authors. All rights reserved.

This document is subject to BCP 78 and the IETF Trust's Legal Provisions Relating to IETF Documents (http://trustee.ietf.org/license-info) in effect on the date of publication of this document. Please review these documents carefully, as they describe your rights and restrictions with respect to this document. Code Components extracted from this document must include Simplified BSD License text as described in Section 4.e of the Trust Legal Provisions and are provided without warranty as described in the BSD License.

Abstract

This document defines a new RTP payload type for the insertion of Forward Error Correction (FEC) in RTP. This solution is based on Reed-Solomon codes that protect the transmission both from packet losses and bit errors which may affect the communication over wireless links. These codes are systematic, thus being completely

transparent to FEC unaware RTP clients. The new payload defined in this draft and the insertion of RS codes are particularly suited for H.264 video streaming.

Conventions used in this document

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].

Table of Contents

1. Introduction

The introduction of Forward Error Correction (FEC) at high level of the OSI stack can be done in several different ways. In particular, as a direct insertion inside the data stream, i.e., at the application level, or as an insertion after encapsulation in RTP packets or on transport packets (UDP, DCCP, SCTP, ...).

The idea of using FEC for RTP packets is not new, and some proposals have already been presented at the IETF. However, the RTP-FEC approach defined in [RFC2733] presents the drawback of tagging as "RTP FEC" redundancy packets. This implies that an unaware receiver, i.e., a receiver not specifically enhanced by the RTP-FEC feature, would discard all packets.

Work in [RFC6015] has a similar approach to the one presented in this document: the solution presented here proposes an alternative insertion of FEC in RTP, providing correction not only of packet losses but also of bit errors. This scheme, based on Reed-Solomon codes, reduced moreover the overhead associated to the FEC headers.

2. Encoding/decoding at transport layer

This section provides an overview of the encoding and decoding process.

The packet-level encoder forwards the packets received from the upper layer immediately to the lower layers to avoid the introduction of significant extra-delay. Such packets are called source packets. The encoding process starts when either a sufficient amount of data is available.

Regardless the specific (N,K) packet-level code that is employed, encoding starts by filling with the source packets an encoding table, also called the source block, consisting of N=K+M rows, each of T bytes, indexed from 1 to n. Each such row is called a symbol. The generic source block is filled with progressive RTP packets.

At the decoder, source or repair packets may be missing. Moreover, both correct and corrupted packets (either source or repair) are received. Packet losses may be due to erasures along the IPv6 network, errors in the DLL/IP/UDP/RTP headers caused by the transmission across a radio channel or selective drops operated by an intelligent transport network. The possible presence of partially corrupted RTP packets (i.e., packets which are recognized as corrupted, but without errors in the header) is due to the use of transport protocols like UDP-Lite or to a partial CRC at the data link layer like introduced in WiMAX and represents an important issue, since a robust source decoder may be able to exploit them to improve the end-to-end quality. At the decoder, the correctly received (source and repair) packets are filled into a decoding source block, analogous to the source block in the encoder. Decoding consists of recovering the missing T-byte symbols from erasures and, if partially corrupted RTP packets are received, correct the errors affecting the corresponding symbols.

3. Reed-Solomon (RS) codes

Reed-Solomon (RS) codes are non-binary systematic cyclic error-correcting codes invented by I. Reed and G. Solomon [1] and detecting and correcting multiple random symbol errors.

The Reed-Solomon code is an optimum [N, K, N-K+1] code: in other words, it is a linear block code of length N with dimension K and minimum Hamming distance N-K+1. Moreover, the minimum distance has the maximum possible value for a linear code of size (N,K).

The error-correcting ability of a Reed-Solomon code is determined by its minimum distance, or, equivalently, by N - K, the measure of redundancy in the block. If the locations of the error symbols are not known in advance, then a Reed-Solomon code can correct up to (N - K) / 2 erroneous symbols, i.e., it can correct half as many errors as there are redundant symbols added to the block. As an erasure code, it can correct up to N - K known erasures. Moreover, it can detect and correct combinations of errors and erasures: a Reed-Solomon code is able to correct any combination of errors and erasures as long as the relation 2E + L = N - K is satisfied, where E is the number of errors and L is the number of erasures in the block.

For practical uses of Reed-Solomon codes, it is common to fix a finite field F with 2m elements. In this case, each symbol can be represented as an m-bit value. The sender transmits the data points as encoded blocks, and the number of symbols in the encoded block is N = 2m - 1. Thus a Reed-Solomon code operating on 8-bit symbols has N = 28 - 1 = 255 symbols per block. The number K (with K < N) of data symbols in the block is a design parameter.

The above properties of Reed-Solomon codes make them especially well suited to applications where errors occur in bursts. Moreover, being systematic, they allow sending information bits untouched and adding redundancy bits thus being fully transparent for a user unaware of the Forward Error Correction (FEC) feature.

4. Source packets

In the introduction of RS codes in RTP, source packets are RTP packets as defined in [RFC3984]. This guarantees that non-FEC capable receives could interpret the received source packets.

5. Repair packets

For the generation of repair packets, the RTP encoder builds a matrix of K rows where every row corresponds to an RTP source packet. The K source RTP data packets are transmitted to the receiver followed by M = N- K RS packets generated by interleaving 1 byte of each RTP data packet.

The dimension K, i.e., the number of source packets in the matrix, is a parameter of the system.

However, it may happen that, in case of a H.264 video packetization, the Kth packet may be in the middle of a Network Abstraction Layer (NAL). In order to avoid the split of a NAL into two different matrix, additional RTP packets COULD be inserted in the matrix, up to the completion of the NAL transmission. It follows that a number K'=K+k of RTP source packets are inserted in the matrix, with k=NALsize-K.

The matrix size N is instead dimensioned on the maximum RTP packet size Smax plus 2 additional bytes for packet size indication, as depicted in Figure 1.

The rational behind this choice is as follows. RTP packets have a variable size S, which can assume a value between 13 bytes (i.e., an header size of 12 bytes and a payload of one byte) and Smax. At the receiver side, in case of losses, the amount of lost data (i.e., the packet size) is needed for the recovery process: this extra information SHOULD be appended to the RTP source packets while inserting them in the matrix. The packet size S, virtually inserted in the matrix and thus considered in the generation of the RS repair packets, SHELL NOT be really transmitted to the receiver: at the receiver side, the payload length can be recovered from the RS packets in case of losses, since it has been included in the computation of the protection.

This solution has a general validity and it can be applied to any kind of data packet. It has however to be noticed that for H.264 video frames the addition of the 2 Bytes for the payload size MAY be skipped. Indeed, in H.264 video frames there are no more than two consecutive bits equal to zero. If follows that, if there are three or more bits equal to zero in the received packets, those bits are of padding: in that case, the end of the packet can be identified by looking, starting from the end of the packet, for the first bit different from zero.

```
                                     matrix size
                   <--------------------------------------------->
                  ++++++++++++++++++++++++++++++++++++++++++++++++
                  | RTP Header |        Payload            |  S  |

                  ++++++++++++++++++++++++++++++++++++++++++++++++
                  ++++++++++++++++++++++++++++++++++++++++++++++++
                  | RTP Header |        Payload            |  S  | k'

                  ++++++++++++++++++++++++++++++++++++++++++++++++
                  ++++++++++++++++++++++++++++++++++++++++++++++++
                  | RTP Header |        Payload            |  S  |

                  ++++++++++++++++++++++++++++++++++++++++++++++++
                  ++++++++++++++++++++++++++++++++++++++++++++++++
                  | RTP Header |        Payload            |  S  |

                  ++++++++++++++++++++++++++++++++++++++++++++++++
++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++

| RTP Header | EH |            Reed-Solomon                      |
++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++

| RTP Header | EH |            Reed-Solomon                      |  m
++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
| RTP Header | EH |            Reed-Solomon                      |
++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
```

Figure 1 Application of RS codes to H.264 RTP packets

5.1. Information for FEC decoding

The following information MUST be known at the receiver side for a correct decoding:

o The Sequence Number (SBN) of the first RTP Packet to take in consideration, i.e., the first packet of the matrix.

- The parameters of the RS codes, i.e., N and K (we assume to work on GF(256)).

- The size of the payload.

- The size of the matrix.

As explained above:

- The size of the matrix corresponds to Smax+2, and thus it does not require to be transmitted;

- The size of the payload is virtually added in the matrix and thus can be determined by the receiver.

Moreover, N MAY be considered equal to 255 at the decoder: indeed, a RS(55,25) is equivalent to a RS(255,25) where the last 200 packets are considered lost as depicted in Figure 2.

We SHOULD thus avoid transmitting this information while signaling to the receiver only the parameter K and the Sequence Number.

5.2. RTP Extra-Header

The repair packets are characterized by an "Extra-header" of four Bytes following the RTP header and reporting, as depicted in Figure 3:

- the SBN (2B)

- K' (1B only considering a maximum number of rows N of 255).

- An RS code on the extra-header (1B). It has to be noticed that SBN and k' have the same values for all the packets in the same matrix: in case of errors, values can be obtained from the other correct packets of the matrix. One byte of protection is thus enough to offer a good reliability with a minimum overhead.

```
    0                   1                   2                   3
    0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
   |V=2|P|X|  CC   |M|     PT      |       sequence number         |
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
   |                           timestamp                           |
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
   |           synchronization source (SSRC) identifier            |
   +=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+
   |            contributing source (CSRC) identifiers             |
   |                             ....                              |
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

```
|      SBN                    |       K       |   RS                    |

+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

Figure 2 RTP-FEC standard extended header for unique/multiple Reed-Solomon codes/codewords.

5.2.1. Repair packet payload type

The M RS repair packets are characterized by a new payload type. The use of a new payload type (e.g., no. 99) assures the compliance to RTP receivers not supporting FEC. At the receiver side, one depacketisers is used for both data and redundancy packets: if RTP FEC packets are received by a client not supporting the RTP FEC mode, the packets with an unknown payload type (i.e, 99) are simply discarded. An RTP FEC receiver would instead decode the RS packet and correct the eventual errors or losses.

The repair packets are identified by an X field with a value equal to the new Payload Type (PT) in the RTP header.

6. Security Considerations

RTP packets using the format defined in this specification are subject to the security considerations of the RTP specification [RTP3350].

The repair packets as presented in this document do not cause specific additional security issues.

7. IANA Considerations

There are no IANA considerations in this document.

8. Acknowledgments

This document was prepared using 2-Word-v2.0.template.dot.

9. References

9.1. Normative References

[RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", March 1997

[RFC2733] J. Rosenberg, H. Schulzinne, "RTP payload format for generic forward error correction", RFC 2733, December 1999.

[RFC6015] A. Begen, "RTP Payload Format for 1-D Interleaved Parity FEC", RFC 6015, October 2010.

9.2. Informative References

[1] I. Reed, G. Solomon, "Polynomial Codes over Certain Finite Fields", Journal of the Society for Industrial and Applied Mathematics (SIAM) 8 (2): p. 300-304

Author's Addresses

Cyril Bergeron
Thales Communications
160 boulevard de Valmy, 92704 Colombes Cedex

Email: Cyril.Bergeron@fr.thalesgroup.com

Benjamin Gadat
Thales Communications
160 boulevard de Valmy, 92704 Colombes Cedex

Email: Benjamin.gadat@fr.thalesgroup.com

Roberta Fracchia
Thales Communications
160 boulevard de Valmy, 92704 Colombes Cedex

Email: Roberta.fracchia@fr.thalesgroup.com