PAYLOAD
Internet-Draft
Intended status: Standards Track
Expires: May 8, 2019

M. Zanaty
Cisco
V. Singh
callstats.io
A. Begen
Networked Media
G. Mandyam
Qualcomm Inc.
November 4, 2018

RTP Payload Format for Flexible Forward Error Correction (FEC) draft-ietf-payload-flexible-fec-scheme-10

Abstract

This document defines new RTP payload formats for the Forward Error Correction (FEC) packets that are generated by the non-interleaved and interleaved parity codes from source media encapsulated in RTP. These parity codes are systematic codes, where a number of FEC repair packets are generated from a set of source packets from one or more source RTP streams. These FEC repair packets are sent in a redundancy RTP stream separate from the source RTP stream(s) that carries the source packets. RTP source packets that were lost in transmission can be reconstructed using the source and repair packets that were received. The non-interleaved and interleaved parity codes which are defined in this specification offer a good protection against random and bursty packet losses, respectively, at a cost of decent complexity. The RTP payload formats that are defined in this document address the scalability issues experienced with the earlier specifications including RFC 2733, RFC 5109 and SMPTE 2022-1, and offer several improvements. Due to these changes, the new payload formats are not backward compatible with the earlier specifications, but endpoints that do not implement this specification can still work by simply ignoring the FEC repair packets.

Status of This Memo

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

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF). Note that other groups may also distribute working documents as Internet-Drafts. The list of current Internet-Drafts is at https://datatracker.ietf.org/drafts/current/.

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."

This Internet-Draft will expire on May 8, 2019.

Table of Contents

1. Introduction

This document defines new RTP payload formats for the Forward Error Correction (FEC) that is generated by the non-interleaved and interleaved parity codes from a source media encapsulated in RTP [RFC3550]. The type of the source media protected by these parity codes can be audio, video, text or application. The FEC data are generated according to the media type parameters, which are communicated out-of-band (e.g., in SDP). Furthermore, the associations or relationships between the source and repair RTP streams may be communicated in-band or out-of-band. The in-band mechanism is advantageous when the endpoint is adapting the FEC parameters. The out-of-band mechanism may be preferable when the FEC parameters are fixed.

The Redunadncy RTP Stream [RFC7656] repair packets proposed in this document protect the Source RTP Stream packets that belong to the same RTP session.

1.1. Parity Codes

Both the non-interleaved and interleaved parity codes use the eXclusive OR (XOR) operation to generate the repair packets. The following steps take place:

1. The sender determines a set of source packets to be protected by FEC based on the media type parameters.

2. The sender applies the XOR operation on the source packets to generate the required number of repair packets.

3. The sender sends the repair packet(s) along with the source packets, in different RTP streams, to the receiver(s). The repair packets may be sent proactively or on-demand based on RTCP feedback messages such as NACK [RFC4585].

At the receiver side, if all of the source packets are successfully received, there is no need for FEC recovery and the repair packets are discarded. However, if there are missing source packets, the repair packets can be used to recover the missing information. Figure 1 and Figure 2 describe example block diagrams for the systematic parity FEC encoder and decoder, respectively.

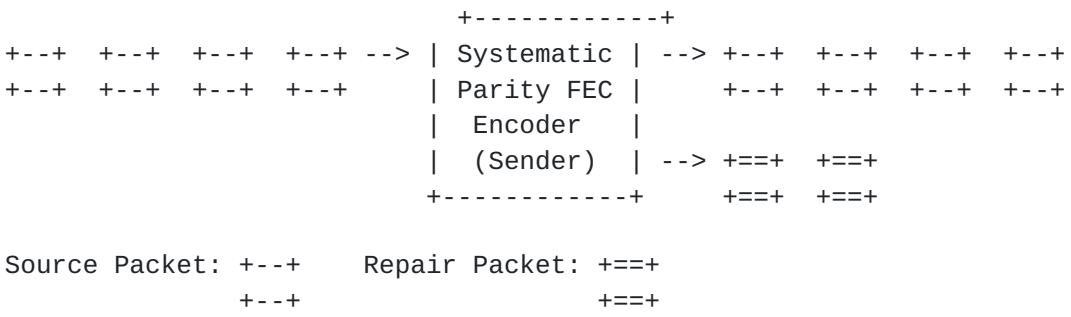

Figure 1: Block diagram for systematic parity FEC encoder

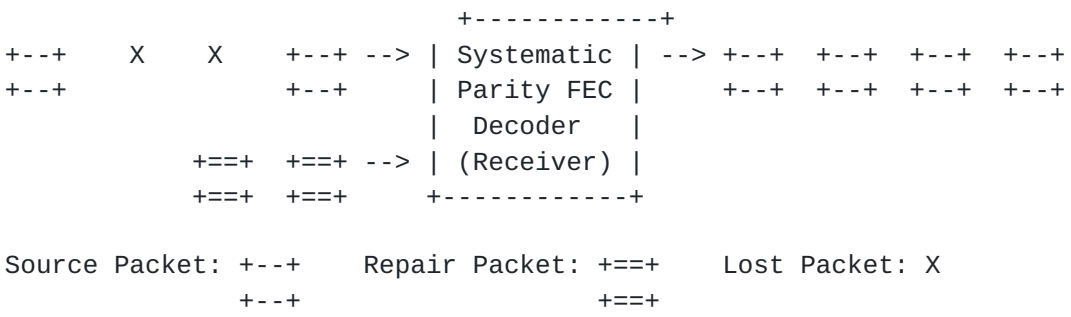

Figure 2: Block diagram for systematic parity FEC decoder

In Figure 2, it is clear that the FEC repair packets have to be received by the endpoint within a certain amount of time for the FEC recovery process to be useful. The repair window is defined as the time that spans a FEC block, which consists of the source packets and the corresponding repair packets. At the receiver side, the FEC decoder SHOULD buffer source and repair packets at least for the

duration of the repair window, to allow all the repair packets to arrive. The FEC decoder can start decoding the already received packets sooner; however, it should not register a FEC decoding failure until it waits at least for the duration of the repair window.

1.1.1. 1-D Non-interleaved (Row) FEC Protection

Consider a group of D x L source packets that have sequence numbers starting from 1 running to D x L, and a repair packet is generated by applying the XOR operation to every L consecutive packets as sketched in Figure 3. This process is referred to as 1-D non-interleaved FEC protection. As a result of this process, D repair packets are generated, which are referred to as non-interleaved (or row) FEC repair packets.

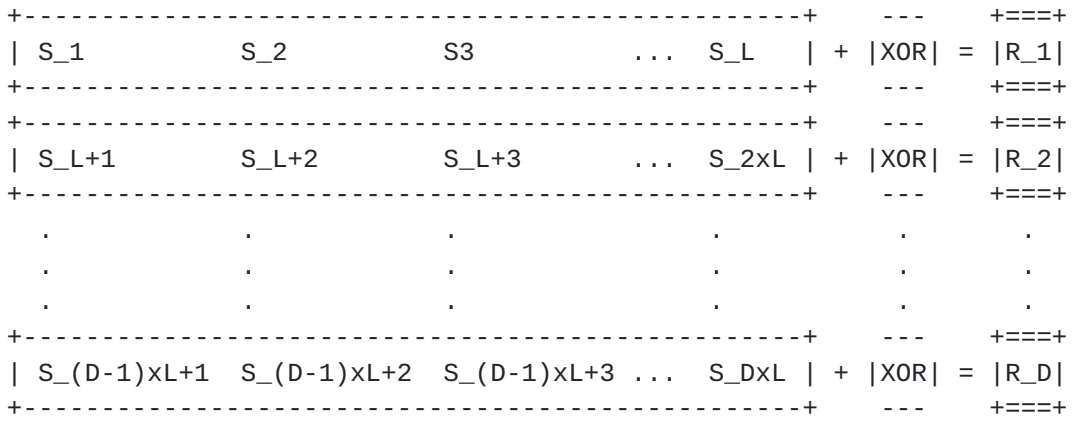

Figure 3: Generating non-interleaved (row) FEC repair packets

1.1.2. 1-D Interleaved (Column) FEC Protection

If the XOR operation is applied to the group of the source packets whose sequence numbers are L apart from each other, as sketched in Figure 4. In this case the endpoint generates L repair packets. This process is referred to as 1-D interleaved FEC protection, and the resulting L repair packets are referred to as interleaved (or column) FEC repair packets.

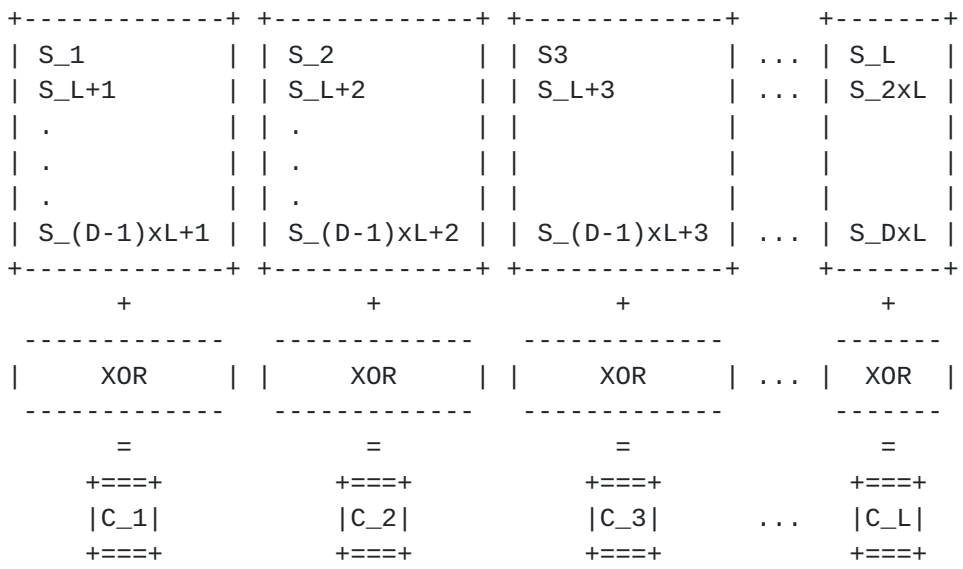

Figure 4: Generating interleaved (column) FEC repair packets

1.1.3. Use Cases for 1-D FEC Protection

A sender may generate one non-interleaved repair packet out of L consecutive source packets or one interleaved repair packet out of D non-consecutive source packets. Regardless of whether the repair packet is a non-interleaved or an interleaved one, it can provide a full recovery of the missing information if there is only one packet missing among the corresponding source packets. This implies that 1-D non-interleaved FEC protection performs better when the source packets are randomly lost. However, if the packet losses occur in bursts, 1-D interleaved FEC protection performs better provided that L is chosen large enough, i.e., L-packet duration is not shorter than the observed burst duration. If the sender generates non-interleaved FEC repair packets and a burst loss hits the source packets, the repair operation fails. This is illustrated in Figure 5.

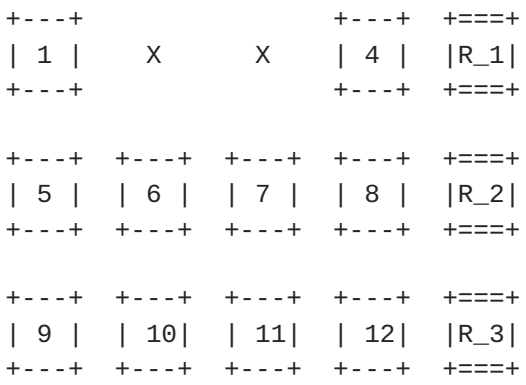

Figure 5: Example scenario where 1-D non-interleaved FEC protection fails error recovery (Burst Loss)

The sender may generate interleaved FEC repair packets to combat with the bursty packet losses. However, two or more random packet losses may hit the source and repair packets in the same column. In that case, the repair operation fails as well. This is illustrated in Figure 6. Note that it is possible that two burst losses may occur back-to-back, in which case interleaved FEC repair packets may still fail to recover the lost data.

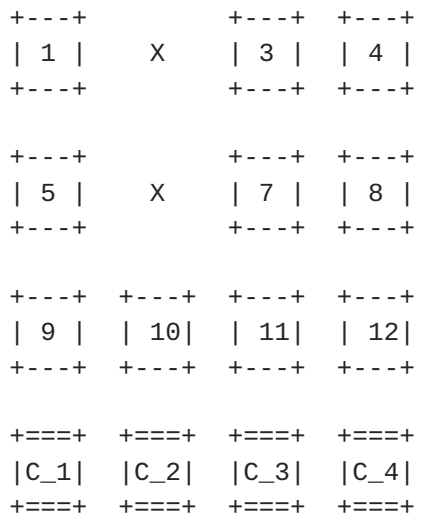

Figure 6: Example scenario where 1-D interleaved FEC protection fails error recovery (Periodic Loss)

1.1.4. 2-D (Row and Column) FEC Protection

In networks where the source packets are lost both randomly and in bursts, the sender ought to generate both non-interleaved and interleaved FEC repair packets. This type of FEC protection is known as 2-D parity FEC protection. At the expense of generating more FEC repair packets, thus increasing the FEC overhead, 2-D FEC provides superior protection against mixed loss patterns. However, it is still possible for 2-D parity FEC protection to fail to recover all of the lost source packets if a particular loss pattern occurs. An example scenario is illustrated in Figure 7.

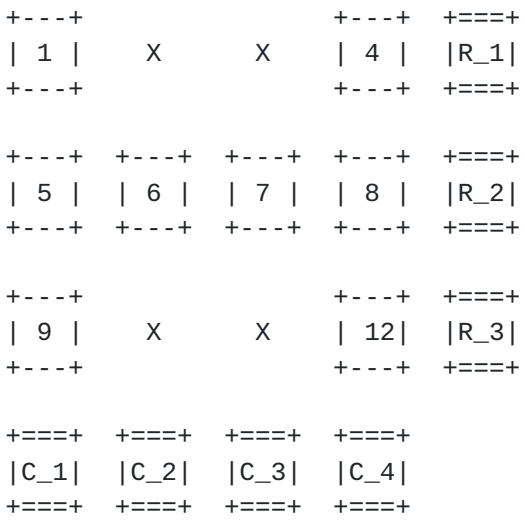

Figure 7: Example scenario #1 where 2-D parity FEC protection fails error recovery

2-D parity FEC protection also fails when at least two rows are missing a source and the FEC packet and the missing source packets (in at least two rows) are aligned in the same column. An example loss pattern is sketched in Figure 8. Similarly, 2-D parity FEC protection cannot repair all missing source packets when at least two columns are missing a source and the FEC packet and the missing source packets (in at least two columns) are aligned in the same row.

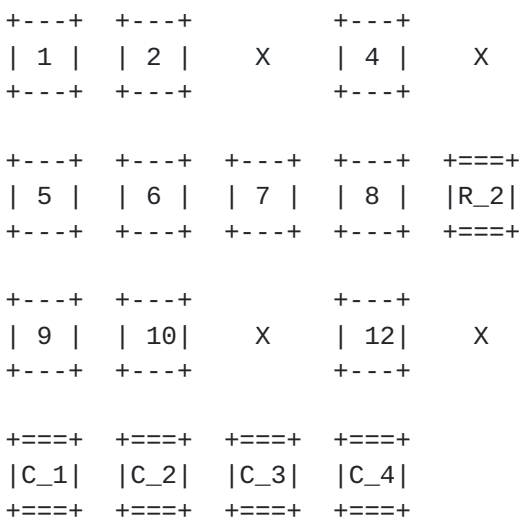

Figure 8: Example scenario #2 where 2-D parity FEC protection fails error recovery

1.1.5. FEC Overhead Considerations

The overhead is defined as the ratio of the number of bytes belonging to the repair packets to the number of bytes belonging to the protected source packets.

Generally, repair packets are larger in size compared to the source packets. Also, not all the source packets are necessarily equal in size. However, assuming that each repair packet carries an equal number of bytes carried by a source packet, the overhead for different FEC protection methods can be computed as follows:

- 1-D Non-interleaved FEC Protection: Overhead = 1/L
- 1-D Interleaved FEC Protection: Overhead = 1/D
- 2-D Parity FEC Protection: Overhead = 1/L + 1/D

where L and D are the number of columns and rows in the source block, respectively.

2. Requirements Notation

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 [RFC2119].

3. Definitions and Notations

3.1. Definitions

This document uses a number of definitions from [RFC6363].

1-D Non-interleaved Row FEC: A protection scheme that operates on consecutive source packets in the source block, able to recover a single lost source packet per row of the source block.

1-D Interleaved Column FEC: A protection scheme that operates on interleaved source packets in the source block, able to recover a single lost source packet per column of the source block.

2-D FEC: A protection scheme that combines row and column FEC.

Source Block: A set of source packets that are protected by a set of 1-D or 2-D FEC repair packets.

FEC Block: A source block and its corresponding FEC repair packets.

Repair Window: The time that spans a FEC block, which consists of the source packets and the corresponding FEC repair packets.

XOR Parity Codes: A FEC code which uses the eXclusive OR (XOR) parity operation to encode a set of source packets to form a FEC repair packet.

3.2. Notations

L: Number of columns of the source block (length of each row).

D: Number of rows of the source block (depth of each column).

bitmask: A 15-bit, 46-bit, or 110-bit mask indicating which source packets are protected by a FEC repair packet. If the bit i in the mask is set to 1, the source packet number N + i is protected by this FEC repair packet, where N is the sequence number base indicated in the FEC repair packet. The most significant bit of the mask corresponds to i=0. The least signficant bit of the mask corresponds to i=14 in the 15-bit mask, i=45 in the 46-bit mask, or i=109 in the 110-bit mask.

4. Packet Formats

This section describes the formats of the source packets and defines the formats of the FEC repair packets.

4.1. Source Packets

The source packets contain the information that identifies the source block and the position within the source block occupied by the packet. Since the source packets that are carried within an RTP stream already contain unique sequence numbers in their RTP headers [RFC3550], the source packets can be identified in a straightforward manner and there is no need to append additional field(s). The primary advantage of not modifying the source packets in any way is that it provides backward compatibility for the receivers that do not support FEC at all. In multicast scenarios, this backward compatibility becomes quite useful as it allows the non-FEC-capable and FEC-capable receivers to receive and interpret the same source packets sent in the same multicast session.

The source packets are transmitted as usual without altering them. They are used along with the FEC repair packets to recover any missing source packets, making this scheme a systematic code.

The source packets are full RTP packets with optional CSRC list, RTP header extension, and padding. If any of these optional elements are present in the source RTP packet, and that source packet is lost, they are recovered by the FEC repair operation, which recovers the full source RTP packet including these optional elements.

4.2. FEC Repair Packets

The FEC repair packets MUST contain information that identifies the source block they pertain to and the relationship between the contained repair packets and the original source block. For this purpose, the RTP header of the repair packets is used, as well as another header within the RTP payload, called the FEC header, as shown in Figure 9.

Note that all the source stream packets that are protected by a particular FEC packet need to be in the same RTP session.

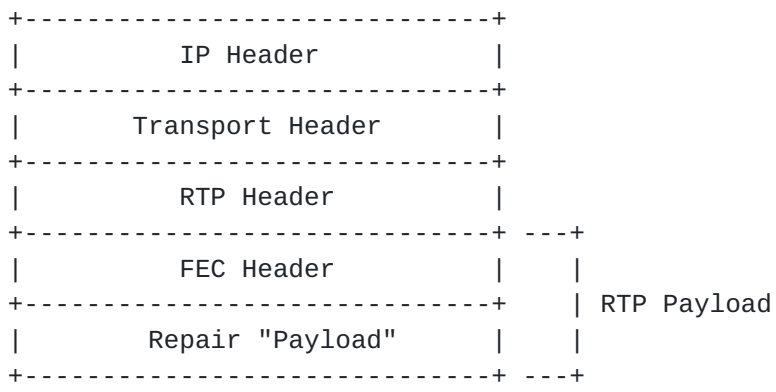

Figure 9: Format of FEC repair packets

Repair "Payload", which follows the FEC Header, includes repair of everything following the fixed 12-byte RTP header of the source packet, including any CSRC list and header extensions if present.

4.2.1. RTP Header of FEC Repair Packets

The RTP header is formatted according to [RFC3550] with some further clarifications listed below:

Version (V) 2 bits: This MUST be set to 2 (binary 10), as this specification requires all source RTP packets and all FEC repair packets to use RTP version 2. The reason for this restriction is the first 2 bits of the FEC header contain other information (R and F bits) rather than recovering the RTP version field.

Padding (P) bit: Source packets can have optional RTP padding, which can be recovered. FEC repaire packets can have optional RTP padding, which is independent of the RTP padding of the source pakcets.

Extension (X) bit: Source packets can have optional RTP header extensions, which can be recovered. FEC repair packets can have optional RTP header extensions, which are independent of the RTP header extensions of the source packets.

CSRC Count (CC) 4 bits, and CSRC List (CSRC_i) 32 bits each: Source packets can have an optional CSRC list and count, which can be recovered. FEC repair packets MUST use the CSRC list and count to specify the SSRC(s) of the source RTP stream(s) protected by this FEC repair packet.

Marker (M) bit: This bit is not used for this payload type, and SHALL be set to 0 by senders, and SHALL be ignored by receivers.

Payload Type: The (dynamic) payload type for the FEC repair packets is determined through out-of-band means. Note that this document registers new payload formats for the repair packets (Refer to Section 5 for details). According to [RFC3550], an RTP receiver that cannot recognize a payload type must discard it. This provides backward compatibility. If a non-FEC-capable receiver receives a repair packet, it will not recognize the payload type, and hence, will discard the repair packet.

Sequence Number (SN): The sequence number has the standard definition. It MUST be one higher than the sequence number in the previously transmitted repair packet. The initial value of the sequence number SHOULD be random (unpredictable, based on [RFC3550]).

Timestamp (TS): The timestamp SHALL be set to a time corresponding to the repair packet's transmission time. Note that the timestamp value has no use in the actual FEC protection process and is usually useful for jitter calculations.

Synchronization Source (SSRC): The SSRC value for each repair stream SHALL be randomly assigned as suggested by [RFC3550]. This allows the sender to multiplex the source and repair RTP streams in the same RTP session, or multiplex multiple repair streams in an RTP session. The repair streams' SSRC's CNAME SHOULD be identical to the CNAME of the source RTP stream(s) that this repair stream protects. In cases when the repair stream covers packets from multiple source RTP streams with different CNAME values, any of these CNAME values MAY be used.

In some networks, the RTP Source, which produces the source packets and the FEC Source, which generates the repair packets from the source packets may not be the same host. In such scenarios, using the same CNAME for the source and repair RTP streams means that the RTP Source and the FEC Source MUST share the same CNAME (for this specific source-repair stream association). A common CNAME may be produced based on an algorithm that is known both to the RTP and FEC Source [RFC7022]. This usage is compliant with [RFC3550].

Note that due to the randomness of the SSRC assignments, there is a possibility of SSRC collision. In such cases, the collisions MUST be resolved as described in [RFC3550].

4.2.2. FEC Header of FEC Repair Packets

The format of the FEC header has 3 variants, depending on the values in the first 2 bits (R and F bits) as shown in Figure 10.

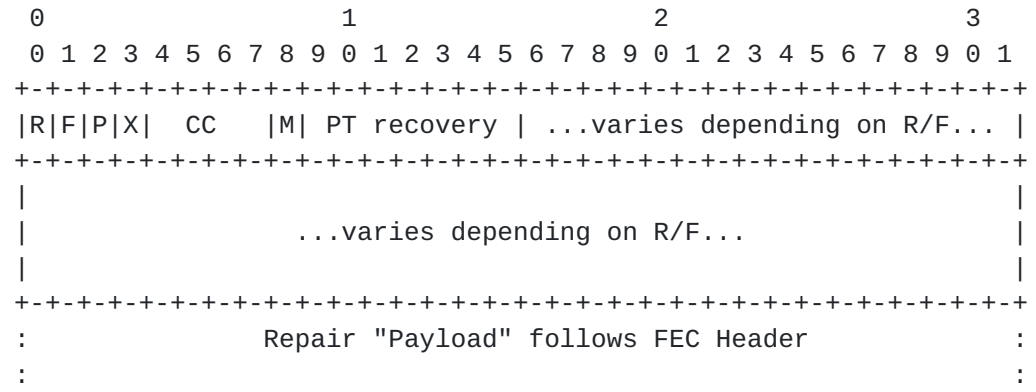

Figure 10: FEC Header

Repair "Payload", which follows the FEC Header, includes repair of everything following the fixed 12-byte RTP header of the source packet, including any CSRC list and header extensions if present.

R	F	FEC Header variant
0	0	Flexible FEC Mask fields indicate source packets
0	1	Fixed FEC L/D (cols/rows) fields indicate source packets
1	0	Retransmission of a single source packet
1	1	Invalid, MUST NOT send, MUST ignore if received

Figure 11: R and F bit values for FEC Header variants

The first variant, when R=0 and F=0, has a mask to signal protected source packets, as shown in Figure 12.

The second variant, when R=0 and F=1, has a number of columns (L) and rows (D) to signal protected source packets, as shown in Figure 13.

The final variant, when R=1 and F=0, is a retransmission format as shown in Figure 15.

No variant uses R=1 and F=1, which is invalid, and MUST NOT be sent by senders, and MUST be ignored by receivers.

The FEC header for all variants consists of the following common fields:

- The R bit MUST be set to 1 to indicate a retransmission packet, and MUST be set to 0 for FEC repair packets.

- The F bit indicates the type of FEC repair packets, as shown in Figure 11, when the R bit is 0. The F bit MUST be set to 0 when the R bit is 1 for retransmission packets.

- The P, X, CC, M and PT recovery fields are used to determine the corresponding fields of the recovered packets.

4.2.2.1. FEC Header with Flexible Mask

When R=0 and F=0, the FEC Header includes flexible mask fields.

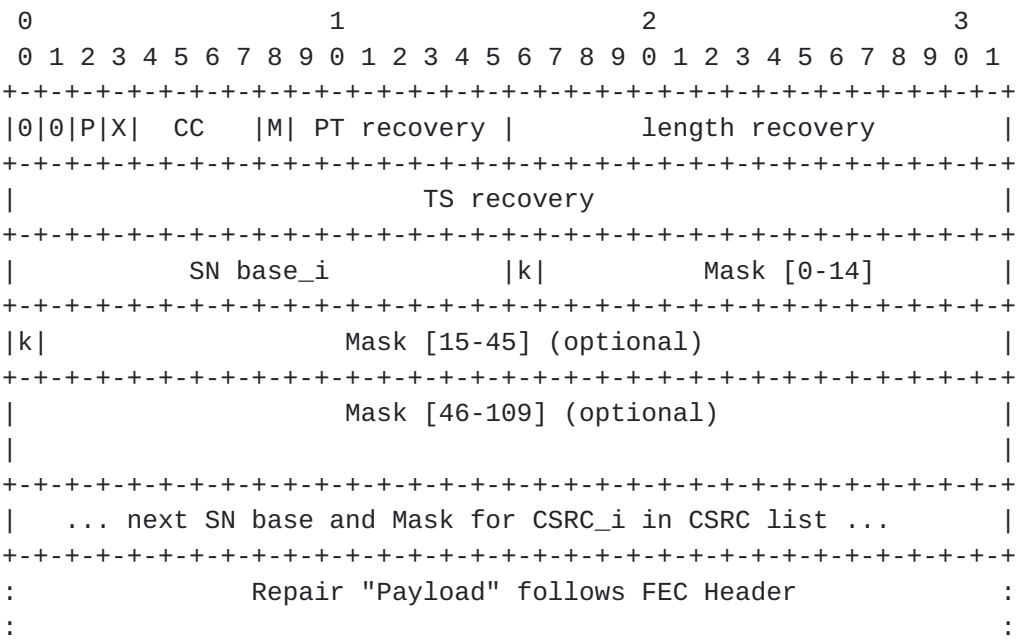

Figure 12: FEC Header for F=0

- The Length recovery (16 bits) field is used to determine the length of the recovered packets. This length includes all octets following the fixed 12-byte RTP header of source packets, including CSRC list and optional header extension(s) if present. It excludes the fixed 12-byte RTP header of source packets.

- The TS recovery (32 bits) field is used to determine the timestamp of the recovered packets.

- The CSRC_i (32 bits) field in the RTP Header (not FEC Header) describes the SSRC of the source packets protected by this particular FEC packet. If a FEC packet protects multiple SSRCs (indicated by the CSRC Count > 1 in the RTP Header), there will be multiple blocks of data containing the SN base and Mask fields.

- The SN base_i (16 bits) field indicates the lowest sequence number, taking wrap around into account, of the source packets for a particular SSRC (indicated in CSRC_i) protected by this repair packet.

- The Mask fields indicate a bitmask of which source packets are protected by this FEC repair packet, where bit j of the mask set to 1 indicates that the source packet with sequence number (SN base_i + j) is protected by this FEC repair packet, where j=0 is the most significant bit in the mask.

- The k-bit in the bitmasks indicates if the mask is 15, 46, or 110 bits. k=1 denotes that another mask follows, and k=0 denotes that it is the last block of mask.

- Repair "Payload", which follows the FEC Header, includes repair of everything following the fixed 12-byte RTP header of the source packet, including any CSRC list and header extensions if present.

4.2.2.2. FEC Header with Fixed L Columns and D Rows

When R=0 and F=1, the FEC Header includes L and D fields for fixed columns and rows. The other fields are the same as the prior section. As in the previous section, the CSRC_i (32 bits) field in the RTP Header (not FEC Header) describes the SSRC of the source packets protected by this particular FEC packet. If there are multiple SSRC's protected by the FEC packet, then there will be multiple blocks of data containing an SN base along with L and D fields.

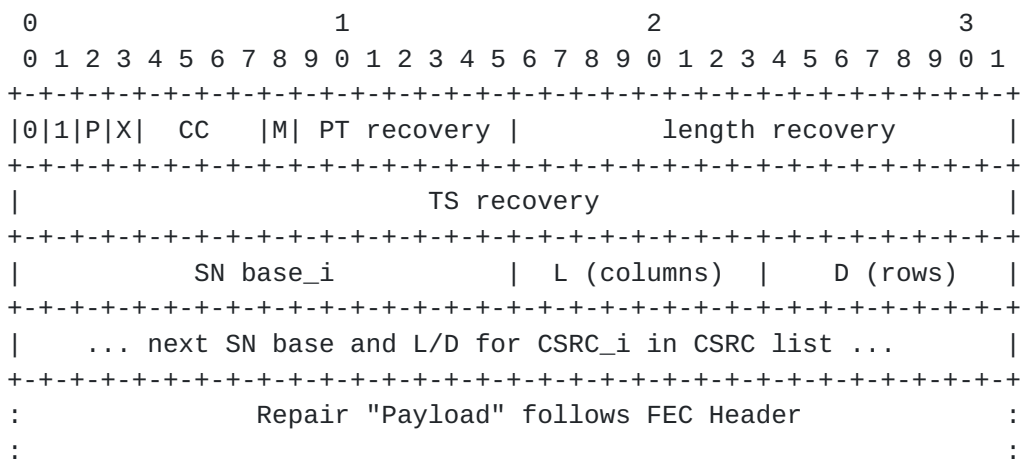

Figure 13: FEC Header for F=1

Consequently, the following conditions occur for L and D values:

If L=0, D=0, use the optional payload format parameters for L and D.

If L>0, D=0, indicates Row FEC, and no column FEC will follow.
Hence, FEC = SN, SN+1, SN+2, ... , SN+(L-1), SN+L.

If L>0, D=1, indicates Row FEC, and column FEC will follow.
Hence, FEC = SN, SN+1, SN+2, ... , SN+(L-1), SN+L will be produced for each row.
Then FEC = SN, SN+L, SN+2L, ..., SN+(D-1)L will be produced for each column.
After all row FEC's have been sent, then the column FEC's will be sent.

If L>0, D>1, indicates column FEC of every L packet in a group of D packets starting at SN base.
Hence, FEC = SN+(Lx0), SN+(Lx1), ... , SN+(LxD).

Figure 14: Interpreting the L and D field values

It should be noted that the flexible mask-based approach may be inefficient for protecting a large number of source packets, or impossible to signal if larger than the largest mask size. In such cases, the fixed columns and rows variant may be more useful.

4.2.2.3. FEC Header for Retransmissions

When R=1 and F=0, the FEC packet is a retransmission of a single source packet. Note that the layout of this retransmission packet is different from other FEC repair packets. The sequence number (SN base_i) replaces the length recovery in the FEC header, since the length is already known for a single packet. There are no L, D or Mask fields, since only a single packet is retransmitted, identified by the sequence number in the FEC header. The source packet SSRC is included in the FEC header for retransmissions, not in the RTP header CSRC list as in the FEC header variants with R=0. When performing retransmissions of single source packets, a unique repair packet stream (SSRC) MUST be used for each source packet stream (SSRC) to enable efficient handling after the first initial repair packet on each SSRC.

This FEC header layout is identical to the source RTP (version 2) packet, starting with its RTP header, where the retransmission "payload" is everything following the fixed 12-byte RTP header of the source packet, including CSRC list and extensions if present. Therefore, the only operation needed for sending retransmissions is to prepend a new RTP header to the source packet.

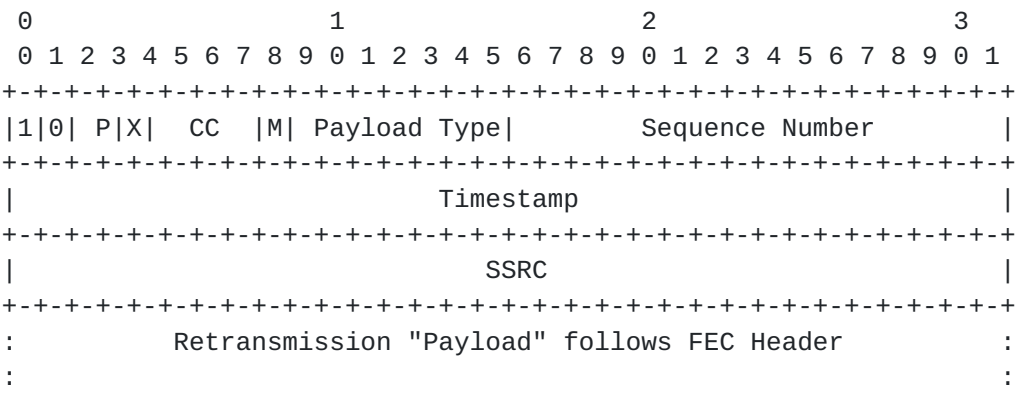

Figure 15: FEC Header for Retransmission

5. Payload Format Parameters

This section provides the media subtype registration for the non-interleaved and interleaved parity FEC. The parameters that are required to configure the FEC encoding and decoding operations are also defined in this section. If no specific FEC code is specified in the subtype, then the FEC code defaults to the parity code defined in this specification.

5.1. Media Type Registration - Parity Codes

This registration is done using the template defined in [RFC6838] and following the guidance provided in [RFC3555].

Note to the RFC Editor: In the following sections, please replace "XXXX" with the number of this document prior to publication as an RFC.

5.1.1. Registration of audio/flexfec

Type name: audio

Subtype name: flexfec

Required parameters:

- rate: The RTP timestamp (clock) rate. The rate SHALL be larger than 1000 Hz to provide sufficient resolution to RTCP operations. However, it is RECOMMENDED to select the rate that matches the rate of the protected source RTP stream.

- repair-window: The time that spans the source packets and the corresponding repair packets. The size of the repair window is specified in microseconds.

Optional parameters:

- L: indicates the number of columns of the source block that are protected by this FEC block and it applies to all the source SSRCs. L is a positive integer.

- D: indicates the number of rows of the source block that are protected by this FEC block and it applies to all the source SSRCs. D is a positive integer.

- ToP: indicates the type of protection applied by the sender: 0 for 1-D interleaved FEC protection, 1 for 1-D non-interleaved FEC protection, 2 for 2-D parity FEC protection, and 3 for retransmission. There can only be one value listed for ToP.

Note that both L and D in the optional parameters should follow the value pairings stated in Section 4.2.2.2 if included.

Encoding considerations: This media type is framed (See Section 4.8 in the template document [RFC6838]) and contains binary data.

Security considerations: See Section 9 of [RFCXXXX].

Interoperability considerations: None.

Published specification: [RFCXXXX].

Applications that use this media type: Multimedia applications that want to improve resiliency against packet loss by sending redundant data in addition to the source media.

Fragment identifier considerations: None.

Additional information: None.

Person & email address to contact for further information: Varun Singh <varun@callstats.io> and IETF Audio/Video Transport Payloads Working Group.

Intended usage: COMMON.

Restriction on usage: This media type depends on RTP framing, and hence, is only defined for transport via RTP [RFC3550].

Author: Varun Singh <varun@callstats.io>.

Change controller: IETF Audio/Video Transport Working Group delegated from the IESG.

Provisional registration? (standards tree only): Yes.

5.1.2. Registration of video/flexfec

Type name: video

Subtype name: flexfec

Required parameters:

- rate: The RTP timestamp (clock) rate. The rate SHALL be larger than 1000 Hz to provide sufficient resolution to RTCP operations. However, it is RECOMMENDED to select the rate that matches the rate of the protected source RTP stream.

- repair-window: The time that spans the source packets and the corresponding repair packets. The size of the repair window is specified in microseconds.

Optional parameters:

- L: indicates the number of columns of the source block that are protected by this FEC block and it applies to all the source SSRCs. L is a positive integer.

- D: indicates the number of rows of the source block that are protected by this FEC block and it applies to all the source SSRCs. D is a positive integer.

- ToP: indicates the type of protection applied by the sender: 0 for 1-D interleaved FEC protection, 1 for 1-D non-interleaved FEC protection, 2 for 2-D parity FEC protection, and 3 for retransmission. There can only be one value listed for ToP.

Note that both L and D in the optional parameters should follow the value pairings stated in Section 4.2.2.2 if included.

Encoding considerations: This media type is framed (See Section 4.8 in the template document [RFC6838]) and contains binary data.

Security considerations: See Section 9 of [RFCXXXX].

Interoperability considerations: None.

Published specification: [RFCXXXX].

Applications that use this media type: Multimedia applications that want to improve resiliency against packet loss by sending redundant data in addition to the source media.

Fragment identifier considerations: None.

Additional information: None.

Person & email address to contact for further information: Varun Singh <varun@callstats.io> and IETF Audio/Video Transport Payloads Working Group.

Intended usage: COMMON.

Restriction on usage: This media type depends on RTP framing, and hence, is only defined for transport via RTP [RFC3550].

Author: Varun Singh <varun@callstats.io>.

Change controller: IETF Audio/Video Transport Working Group delegated from the IESG.

Provisional registration? (standards tree only): Yes.

5.1.3. Registration of text/flexfec

Type name: text

Subtype name: flexfec

Required parameters:

- rate: The RTP timestamp (clock) rate. The rate SHALL be larger than 1000 Hz to provide sufficient resolution to RTCP operations. However, it is RECOMMENDED to select the rate that matches the rate of the protected source RTP stream.

- repair-window: The time that spans the source packets and the corresponding repair packets. The size of the repair window is specified in microseconds.

Optional parameters:

- L: indicates the number of columns of the source block that are protected by this FEC block and it applies to all the source SSRCs. L is a positive integer.

- D: indicates the number of rows of the source block that are protected by this FEC block and it applies to all the source SSRCs. D is a positive integer.

- ToP: indicates the type of protection applied by the sender: 0 for 1-D interleaved FEC protection, 1 for 1-D non-interleaved FEC protection, 2 for 2-D parity FEC protection, and 3 for retransmission. There can only be one value listed for ToP.

Note that both L and D in the optional parameters should follow the value pairings stated in Section 4.2.2.2 if included.

Encoding considerations: This media type is framed (See Section 4.8 in the template document [RFC6838]) and contains binary data.

Security considerations: See Section 9 of [RFCXXXX].

Interoperability considerations: None.

Published specification: [RFCXXXX].

Applications that use this media type: Multimedia applications that want to improve resiliency against packet loss by sending redundant data in addition to the source media.

Fragment identifier considerations: None.

Additional information: None.

Person & email address to contact for further information: Varun Singh <vvarun@callstats.io> and IETF Audio/Video Transport Payloads Working Group.

Intended usage: COMMON.

Restriction on usage: This media type depends on RTP framing, and hence, is only defined for transport via RTP [RFC3550].

Author: Varun Singh <varun@callstats.io>.

Change controller: IETF Audio/Video Transport Working Group delegated from the IESG.

Provisional registration? (standards tree only): Yes.

5.1.4. Registration of application/flexfec

Type name: application

Subtype name: flexfec

Required parameters:

- rate: The RTP timestamp (clock) rate. The rate SHALL be larger than 1000 Hz to provide sufficient resolution to RTCP operations. However, it is RECOMMENDED to select the rate that matches the rate of the protected source RTP stream.

- repair-window: The time that spans the source packets and the corresponding repair packets. The size of the repair window is specified in microseconds.

Optional parameters:

- L: indicates the number of columns of the source block that are protected by this FEC block and it applies to all the source SSRCs. L is a positive integer.

- D: indicates the number of rows of the source block that are protected by this FEC block and it applies to all the source SSRCs. D is a positive integer.

- ToP: indicates the type of protection applied by the sender: 0 for 1-D interleaved FEC protection, 1 for 1-D non-interleaved FEC protection, 2 for 2-D parity FEC protection, and 3 for retransmission. There can only be one value listed for ToP.

Note that both L and D in the optional parameters should follow the value pairings stated in Section 4.2.2.2 if included.

Encoding considerations: This media type is framed (See Section 4.8 in the template document [RFC6838]) and contains binary data.

Security considerations: See Section 9 of [RFCXXXX].

Interoperability considerations: None.

Published specification: [RFCXXXX].

Applications that use this media type: Multimedia applications that want to improve resiliency against packet loss by sending redundant data in addition to the source media.

Fragment identifier considerations: None.

Additional information: None.

Person & email address to contact for further information: Varun Singh <varun@callstats.io> and IETF Audio/Video Transport Payloads Working Group.

Intended usage: COMMON.

Restriction on usage: This media type depends on RTP framing, and hence, is only defined for transport via RTP [RFC3550].

Author: Varun Singh <varun@callstats.io>.

Change controller: IETF Audio/Video Transport Working Group delegated from the IESG.

Provisional registration? (standards tree only): Yes.

5.2. Mapping to SDP Parameters

Applications that are using RTP transport commonly use Session Description Protocol (SDP) [RFC4566] to describe their RTP sessions. The information that is used to specify the media types in an RTP session has specific mappings to the fields in an SDP description. This section provides these mappings for the media subtypes

registered by this document. Note that if an application does not use SDP to describe the RTP sessions, an appropriate mapping must be defined and used to specify the media types and their parameters for the control/description protocol employed by the application.

The mapping of the media type specification for "non-interleaved-parityfec" and "interleaved-parityfec" and their parameters in SDP is as follows:

- The media type (e.g., "application") goes into the "m=" line as the media name.

- The media subtype goes into the "a=rtpmap" line as the encoding name. The RTP clock rate parameter ("rate") also goes into the "a=rtpmap" line as the clock rate.

- The remaining required payload-format-specific parameters go into the "a=fmtp" line by copying them directly from the media type string as a semicolon-separated list of parameter=value pairs.

SDP examples are provided in Section 7.1.

5.2.1. Offer-Answer Model Considerations

When offering 1-D interleaved parity FEC over RTP using SDP in an Offer/Answer model [RFC3264], the following considerations apply:

- Each combination of the L and D parameters produces a different FEC data and is not compatible with any other combination. A sender application may desire to offer multiple offers with different sets of L and D values as long as the parameter values are valid. The receiver SHOULD normally choose the offer that has a sufficient amount of interleaving. If multiple such offers exist, the receiver may choose the offer that has the lowest overhead or the one that requires the smallest amount of buffering. The selection depends on the application requirements.

- The value for the repair-window parameter depends on the L and D values and cannot be chosen arbitrarily. More specifically, L and D values determine the lower limit for the repair-window size. The upper limit of the repair-window size does not depend on the L and D values.

- Although combinations with the same L and D values but with different repair-window sizes produce the same FEC data, such combinations are still considered different offers. The size of the repair-window is related to the maximum delay between the transmission of a source packet and the associated repair packet.

This directly impacts the buffering requirement on the receiver side and the receiver must consider this when choosing an offer.

- Any unknown option in the offer MUST be ignored and deleted from the answer. If FEC is not desired by the receiver, it can be deleted from the answer.

5.2.2. Declarative Considerations

In declarative usage, like SDP in the Real-time Streaming Protocol (RTSP) [RFC2326] or the Session Announcement Protocol (SAP) [RFC2974], the following considerations apply:

- The payload format configuration parameters are all declarative and a participant MUST use the configuration that is provided for the session.

- More than one configuration may be provided (if desired) by declaring multiple RTP payload types. In that case, the receivers should choose the repair stream that is best for them.

6. Protection and Recovery Procedures - Parity Codes

This section provides a complete specification of the 1-D and 2-D parity codes and their RTP payload formats. It does not apply to the single packet retransmission format (R=1 in the FEC Header).

6.1. Overview

The following sections specify the steps involved in generating the repair packets and reconstructing the missing source packets from the repair packets.

6.2. Repair Packet Construction

The RTP Header of a repair packet is formed based on the guidelines given in Section 4.2.

The FEC Header and Repair "Payload" of repair packets are formed by applying the XOR operation on the bit strings that are generated from the individual source packets protected by this particular repair packet. The set of the source packets that are associated with a given repair packet can be computed by the formula given in Section 6.3.1.

The bit string is formed for each source packet by concatenating the following fields together in the order specified:

- The first 16 bits of the RTP header (16 bits).

- Unsigned network-ordered 16-bit representation of the source packet length in bytes minus 12 (for the fixed RTP header), i.e., the sum of the lengths of all the following if present: the CSRC list, extension header, RTP payload and RTP padding (16 bits).

- The timestamp of the RTP header (32 bits).

- All octets after the fixed 12-byte RTP header. (Note the SSRC field is skipped.)

The FEC bit string is generated by applying the parity operation on the bit strings produced from the source packets. The FEC header is generated from the FEC bit string as follows:

- The first (most significant) 2 bits in the FEC bit string, which contain the RTP version field, are skipped. The R and F bits in the FEC header are set to the appropriate value, i.e., it depends on the chosen format variant. As a consequence of overwriting the RTP version field with the R and F bits, this payload format only supports RTP version 2.

- The next bit in the FEC bit string is written into the P recovery bit in the FEC header.

- The next bit in the FEC bit string is written into the X recovery bit in the FEC header.

- The next 4 bits of the FEC bit string are written into the CC recovery field in the FEC header.

- The next bit is written into the M recovery bit in the FEC header.

- The next 7 bits of the FEC bit string are written into the PT recovery field in the FEC header.

- The next 16 bits are written into the length recovery field in the FEC header.

- The next 32 bits of the FEC bit string are written into the TS recovery field in the FEC header.

- The lowest Sequence Number of the source packets protected by this repair packet is written into the Sequence Number Base field in the FEC header. This needs to be repeated for each SSRC that has packets included in the source block.

- Depending on the chosen FEC header variant, the mask(s) are set when F=0, or the L and D values are set when F=1. This needs to be repeated for each SSRC that has packets included in the source block.

- The rest of the FEC bit string, which contains everything after the fixed 12-byte RTP header of the source packet, is written into the Repair "Payload" following the FEC header, where "Payload" refers to everything after the fixed 12-byte RTP header, including extensions, CSRC list, true payloads, and padding.

If the lengths of the source packets are not equal, each shorter packet MUST be padded to the length of the longest packet by adding octet 0's at the end.

Due to this possible padding and mandatory FEC header, a repair packet has a larger size than the source packets it protects. This may cause problems if the resulting repair packet size exceeds the Maximum Transmission Unit (MTU) size of the path over which the repair stream is sent.

6.3. Source Packet Reconstruction

This section describes the recovery procedures that are required to reconstruct the missing source packets. The recovery process has two steps. In the first step, the FEC decoder determines which source and repair packets should be used in order to recover a missing packet. In the second step, the decoder recovers the missing packet, which consists of an RTP header and RTP payload.

The following describes the RECOMMENDED algorithms for the first and second steps. Based on the implementation, different algorithms MAY be adopted. However, the end result MUST be identical to the one produced by the algorithms described below.

Note that the same algorithms are used by the 1-D parity codes, regardless of whether the FEC protection is applied over a column or a row. The 2-D parity codes, on the other hand, usually require multiple iterations of the procedures described here. This iterative decoding algorithm is further explained in Section 6.3.4.

6.3.1. Associating the Source and Repair Packets

Before associating source and repair packets, the receiver must know in which RTP sessions the source and repair respectively are being sent. After this is established by the reciever the first step is associating the source and repair packets. This association can be via flexible bitmasks, or fixed L and D offsets which can be in the

FEC header or signaled in SDP in optional payload format parameters when L=D=0 in the FEC header.

6.3.1.1. Using Bitmasks

To use flexible bitmasks, the first two FEC header bits MUST have R=0 and F=0. A 15-bit, 46-bit, or 110-bit mask indicates which source packets are protected by a FEC repair packet. If the bit i in the mask is set to 1, the source packet number N + i is protected by this FEC repair packet, where N is the sequence number base indicated in the FEC header. The most significant bit of the mask corresponds to i=0. The least signficant bit of the mask corresponds to i=14 in the 15-bit mask, i=45 in the 46-bit mask, or i=109 in the 110-bit mask.

The bitmasks are able to represent arbitrary protection patterns, for example, 1-D interleaved, 1-D non-interleaved, 2-D, staircase.

6.3.1.2. Using L and D Offsets

Denote the set of the source packets associated with repair packet p* by set T(p*). Note that in a source block whose size is L columns by D rows, set T includes D source packets plus one repair packet for the FEC protection applied over a column, and L source packets plus one repair packet for the FEC protection applied over a row. Recall that 1-D interleaved and non-interleaved FEC protection can fully recover the missing information if there is only one source packet missing per column or row in set T. If there are more than one source packets missing per column or row in set T, 1-D FEC protection may fail to recover all the missing information.

When value of L is non-zero, the 8-bit fields indicate the offset of packets protected by an interleaved (D>0) or non-interleaved (D=0) FEC packet. Using a combination of interleaved and non-interleaved FEC repair packets can form 2-D protection patterns.

Mathematically, for any received repair packet, p*, the sequence numbers of the source packets that are protected by this repair packet are determined as follows, where p*_snb is the sequence number base in the FEC header:

```
When D = 0:
  p*_snb, p*_snb+1,..., p*_snb+L
When D > 0:
  p*_snb, p*_snb+(Lx1), p*_snb+(Lx2),..., p*_snb+(LxD)
```

6.3.1.3. Signaled in SDP

If the endpoint relies entirely on out-of-band signaling (R=0, F=1, L=0, D=0 in the FEC header), then this information may be inferred from the media type parameters specified in the SDP description. Furthermore, the payload type field in the RTP header assists the receiver to distinguish an interleaved or non-interleaved FEC packet.

Mathematically, for any received repair packet, p*, the sequence numbers of the source packets that are protected by this repair packet are determined as follows:

$$p^*\_snb + i * X\_1 \text{ (modulo 65536)}$$

where p*_snb denotes the value in the SN base field of p*'s FEC header, X_1 is set to L and 1 for the interleaved and non-interleaved FEC repair packets, respectively, and

$$0 <= i < X\_2$$

where X_2 is set to D and L for the interleaved and non-interleaved FEC repair packets, respectively.

6.3.2. Recovering the RTP Header

For a given set T, the procedure for the recovery of the RTP header of the missing packet, whose sequence number is denoted by SEQNUM, is as follows:

1. For each of the source packets that are successfully received in T, compute the 80-bit string by concatenating the first 64 bits of their RTP header and the unsigned network-ordered 16-bit representation of their length in bytes minus 12.

2. For the repair packet in T, compute the FEC bit string from the first 80 bits of the FEC header.

3. Calculate the recovered bit string as the XOR of the bit strings generated from all source packets in T and the FEC bit string generated from the repair packet in T.

4. Create a new packet with the standard 12-byte RTP header and no payload.

5. Set the version of the new packet to 2. Skip the first 2 bits in the recovered bit string.

6. Set the Padding bit in the new packet to the next bit in the recovered bit string.

7. Set the Extension bit in the new packet to the next bit in the recovered bit string.

8. Set the CC field to the next 4 bits in the recovered bit string.

9. Set the Marker bit in the new packet to the next bit in the recovered bit string.

10. Set the Payload type in the new packet to the next 7 bits in the recovered bit string.

11. Set the SN field in the new packet to SEQNUM. Skip the next 16 bits in the recovered bit string.

12. Set the TS field in the new packet to the next 32 bits in the recovered bit string.

13. Take the next 16 bits of the recovered bit string and set the new variable Y to whatever unsigned integer this represents (assuming network order). Convert Y to host order. Y represents the length of the new packet in bytes minus 12 (for the fixed RTP header), i.e., the sum of the lengths of all the following if present: the CSRC list, header extension, RTP payload and RTP padding.

14. Set the SSRC of the new packet to the SSRC of the missing source RTP stream.

This procedure recovers the header of an RTP packet up to (and including) the SSRC field.

6.3.3. Recovering the RTP Payload

Following the recovery of the RTP header, the procedure for the recovery of the RTP "payload" is as follows, where "payload" refers to everything following the fixed 12-byte RTP header, including extensions, CSRC list, true payload and padding.

1. Append Y bytes to the new packet.

2. For each of the source packets that are successfully received in T, compute the bit string from the Y octets of data starting with the 13th octet of the packet. If any of the bit strings generated from the source packets has a length shorter than Y, pad them to that length. The padding of octet 0 MUST be added at

the end of the bit string. Note that the information of the first 8 octets are protected by the FEC header.

3. For the repair packet in T, compute the FEC bit string from the repair packet payload, i.e., the Y octets of data following the FEC header. Note that the FEC header may be different sizes depending on the variant and bitmask size.

4. Calculate the recovered bit string as the XOR of the bit strings generated from all source packets in T and the FEC bit string generated from the repair packet in T.

5. Append the recovered bit string (Y octets) to the new packet generated in Section 6.3.2.

6.3.4. Iterative Decoding Algorithm for the 2-D Parity FEC Protection

In 2-D parity FEC protection, the sender generates both non-interleaved and interleaved FEC repair packets to combat with the mixed loss patterns (random and bursty). At the receiver side, these FEC packets are used iteratively to overcome the shortcomings of the 1-D non-interleaved/interleaved FEC protection and improve the chances of full error recovery.

The iterative decoding algorithm runs as follows:

1. Set num_recovered_until_this_iteration to zero

2. Set num_recovered_so_far to zero

3. Recover as many source packets as possible by using the non-interleaved FEC repair packets as outlined in Section 6.3.2 and Section 6.3.3, and increase the value of num_recovered_so_far by the number of recovered source packets.

4. Recover as many source packets as possible by using the interleaved FEC repair packets as outlined in Section 6.3.2 and Section 6.3.3, and increase the value of num_recovered_so_far by the number of recovered source packets.

5. If num_recovered_so_far > num_recovered_until_this_iteration
   ---num_recovered_until_this_iteration = num_recovered_so_far
   ---Go to step 3
   Else
   ---Terminate

The algorithm terminates either when all missing source packets are fully recovered or when there are still remaining missing source

packets but the FEC repair packets are not able to recover any more source packets. For the example scenarios when the 2-D parity FEC protection fails full recovery, refer to Section 1.1.4. Upon termination, variable num_recovered_so_far has a value equal to the total number of recovered source packets.

Example:

Suppose that the receiver experienced the loss pattern sketched in Figure 16.

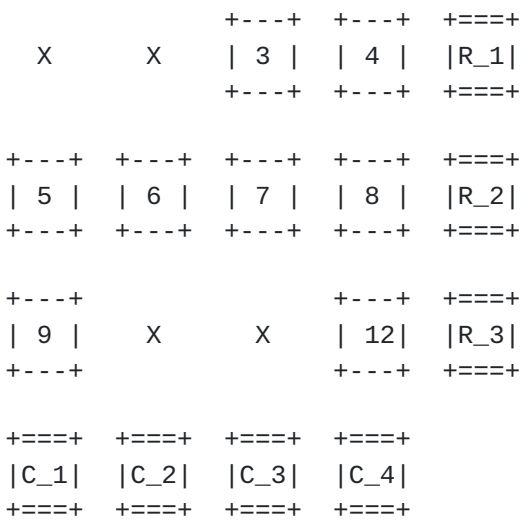

Figure 16: Example loss pattern for the iterative decoding algorithm

The receiver executes the iterative decoding algorithm and recovers source packets #1 and #11 in the first iteration. The resulting pattern is sketched in Figure 17.

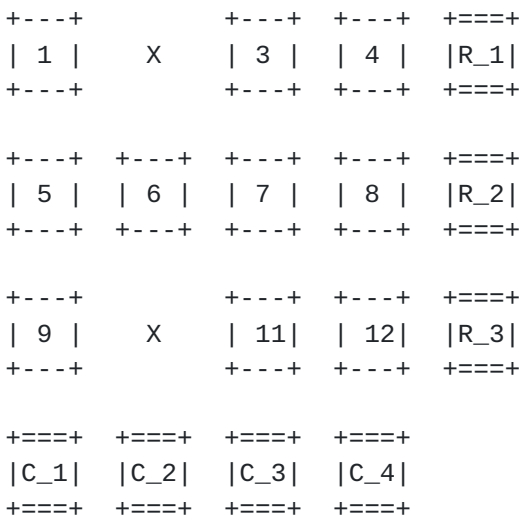

Figure 17: The resulting pattern after the first iteration

Since the if condition holds true, the receiver runs a new iteration. In the second iteration, source packets #2 and #10 are recovered, resulting in a full recovery as sketched in Figure 18.

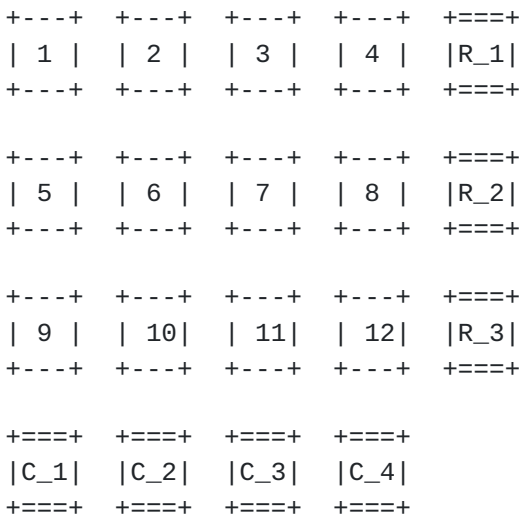

Figure 18: The resulting pattern after the second iteration

7. Signaling Requirements

Out-of-band signaling should be designed to enable the receiver to identify the RTP streams associated with source packets and repair packets, respectively. At a minimum, the signaling must be designed to allow the receiver to

- Determine whether one or more source RTP streams will be sent.
- Determine whether one or more repair RTP streams will be sent.
- Associate the appropriate SSRC's to both source and repair streams.
- Clearly identify which SSRC's are associated with each source block.
- Clearly identify which repair packets correspond to which source blocks.
- Make use of repair packets to recover source data associated with specific SSRC's.

This section provides several Sesssion Description Protocol (SDP) examples to demonstrate how these requirements can be met.

7.1. SDP Examples

This section provides two SDP [RFC4566] examples. The examples use the FEC grouping semantics defined in [RFC5956].

7.1.1. Example SDP for Flexible FEC Protection with in-band SSRC mapping

In this example, we have one source video stream and one FEC repair stream. The source and repair streams are multiplexed on different SSRCs. The repair window is set to 200 ms.

```
v=0
o=mo 1122334455 1122334466 IN IP4 fec.example.com
s=FlexFEC minimal SDP signalling Example
t=0 0
m=video 30000 RTP/AVP 96 98
c=IN IP4 143.163.151.157
a=rtpmap:96 VP8/90000
a=rtpmap:98 flexfec/90000
a=fmtp:98; repair-window=200ms
```

7.1.2. Example SDP for Flex FEC Protection with explicit signalling in the SDP

This example shows one source video stream (ssrc:1234) and one FEC repair streams (ssrc:2345). One FEC group is formed with the "a=ssrc-group:FEC-FR 1234 2345" line. The source and repair streams are multiplexed on different SSRCs. The repair window is set to 200 ms.

```
v=0
o=ali 1122334455 1122334466 IN IP4 fec.example.com
s=2-D Parity FEC with no in band signalling Example
t=0 0
m=video 30000 RTP/AVP 100 110
c=IN IP4 233.252.0.1/127
a=rtpmap:100 MP2T/90000
a=rtpmap:110 flexfec/90000
a=fmtp:110 L:5; D:10; ToP:2; repair-window:200000
a=ssrc:1234
a=ssrc:2345
a=ssrc-group:FEC-FR 1234 2345
```

8. Congestion Control Considerations

FEC is an effective approach to provide applications resiliency against packet losses. However, in networks where the congestion is a major contributor to the packet loss, the potential impacts of using FEC MUST be considered carefully before injecting the repair streams into the network. In particular, in bandwidth-limited networks, FEC repair streams may consume a significant part of the available bandwidth and consequently may congest the network. In such cases, the applications MUST NOT arbitrarily increase the amount of FEC protection since doing so may lead to a congestion collapse. If desired, stronger FEC protection MAY be applied only after the source rate has been reduced.

In a network-friendly implementation, an application SHOULD NOT send/receive FEC repair streams if it knows that sending/receiving those FEC repair streams would not help at all in recovering the missing packets. It is RECOMMENDED that the amount and type (row, column, or both) of FEC protection is adjusted dynamically based on the packet loss rate and burst loss length observed by the applications.

In multicast scenarios, it may be difficult to optimize the FEC protection per receiver. If there is a large variation among the levels of FEC protection needed by different receivers, it is

RECOMMENDED that the sender offers multiple repair streams with different levels of FEC protection and the receivers join the corresponding multicast sessions to receive the repair stream(s) that is best for them.

9. Security Considerations

RTP packets using the payload format defined in this specification are subject to the security considerations discussed in the RTP specification [RFC3550] and in any applicable RTP profile. The main security considerations for the RTP packet carrying the RTP payload format defined within this memo are confidentiality, integrity and source authenticity. Confidentiality is achieved by encrypting the RTP payload. Integrity of the RTP packets is achieved through a suitable cryptographic integrity protection mechanism. Such a cryptographic system may also allow the authentication of the source of the payload. A suitable security mechanism for this RTP payload format should provide confidentiality, integrity protection, and at least source authentication capable of determining if an RTP packet is from a member of the RTP session.

Note that the appropriate mechanism to provide security to RTP and payloads following this memo may vary. It is dependent on the application, transport and signaling protocol employed. Therefore, a single mechanism is not sufficient, although if suitable, using the Secure Real-time Transport Protocol (SRTP) [RFC3711] is recommended. Other mechanisms that may be used are IPsec [RFC4301] and Transport Layer Security (TLS) [RFC5246] (RTP over TCP); other alternatives may exist.

Given that FLEX FEC enables the protection of multiple source streams, there exists the possibility that multiple source buffers may be created that may not be used. In addition, the interaction between a FLEX FEC implementation and higher-layer applications may be affected by non-uniform processing requirements of the FEC scheme.

10. IANA Considerations

New media subtypes are subject to IANA registration. For the registration of the payload formats and their parameters introduced in this document, refer to Section 5.

11. Acknowledgments

Some parts of this document are borrowed from [RFC5109]. Thus, the author would like to thank the editor of [RFC5109] and those who contributed to [RFC5109].

Thanks to Stephen Botzko , Bernard Aboba , Rasmus Brandt , Brian Baldino , Roni Even , Stefan Holmer , Jonathan Lennox , and Magnus Westerlund for providing valuable feedback on earlier versions of this draft.

12. References

12.1. Normative References

[RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, DOI 10.17487/RFC2119, March 1997, <https://www.rfc-editor.org/info/rfc2119>.

[RFC3264] Rosenberg, J. and H. Schulzrinne, "An Offer/Answer Model with Session Description Protocol (SDP)", RFC 3264, DOI 10.17487/RFC3264, June 2002, <https://www.rfc-editor.org/info/rfc3264>.

[RFC3550] Schulzrinne, H., Casner, S., Frederick, R., and V. Jacobson, "RTP: A Transport Protocol for Real-Time Applications", STD 64, RFC 3550, DOI 10.17487/RFC3550, July 2003, <https://www.rfc-editor.org/info/rfc3550>.

[RFC3555] Casner, S. and P. Hoschka, "MIME Type Registration of RTP Payload Formats", RFC 3555, DOI 10.17487/RFC3555, July 2003, <https://www.rfc-editor.org/info/rfc3555>.

[RFC4566] Handley, M., Jacobson, V., and C. Perkins, "SDP: Session Description Protocol", RFC 4566, DOI 10.17487/RFC4566, July 2006, <https://www.rfc-editor.org/info/rfc4566>.

[RFC5956] Begen, A., "Forward Error Correction Grouping Semantics in the Session Description Protocol", RFC 5956, DOI 10.17487/RFC5956, September 2010, <https://www.rfc-editor.org/info/rfc5956>.

[RFC6363] Watson, M., Begen, A., and V. Roca, "Forward Error Correction (FEC) Framework", RFC 6363, DOI 10.17487/RFC6363, October 2011, <https://www.rfc-editor.org/info/rfc6363>.

[RFC6709] Carpenter, B., Aboba, B., Ed., and S. Cheshire, "Design Considerations for Protocol Extensions", RFC 6709, DOI 10.17487/RFC6709, September 2012, <https://www.rfc-editor.org/info/rfc6709>.

[RFC6838] Freed, N., Klensin, J., and T. Hansen, "Media Type Specifications and Registration Procedures", BCP 13, RFC 6838, DOI 10.17487/RFC6838, January 2013, <https://www.rfc-editor.org/info/rfc6838>.

[RFC7022] Begen, A., Perkins, C., Wing, D., and E. Rescorla, "Guidelines for Choosing RTP Control Protocol (RTCP) Canonical Names (CNAMEs)", RFC 7022, DOI 10.17487/RFC7022, September 2013, <https://www.rfc-editor.org/info/rfc7022>.

12.2. Informative References

[RFC2326] Schulzrinne, H., Rao, A., and R. Lanphier, "Real Time Streaming Protocol (RTSP)", RFC 2326, DOI 10.17487/RFC2326, April 1998, <https://www.rfc-editor.org/info/rfc2326>.

[RFC2733] Rosenberg, J. and H. Schulzrinne, "An RTP Payload Format for Generic Forward Error Correction", RFC 2733, DOI 10.17487/RFC2733, December 1999, <https://www.rfc-editor.org/info/rfc2733>.

[RFC2974] Handley, M., Perkins, C., and E. Whelan, "Session Announcement Protocol", RFC 2974, DOI 10.17487/RFC2974, October 2000, <https://www.rfc-editor.org/info/rfc2974>.

[RFC3711] Baugher, M., McGrew, D., Naslund, M., Carrara, E., and K. Norrman, "The Secure Real-time Transport Protocol (SRTP)", RFC 3711, DOI 10.17487/RFC3711, March 2004, <https://www.rfc-editor.org/info/rfc3711>.

[RFC4301] Kent, S. and K. Seo, "Security Architecture for the Internet Protocol", RFC 4301, DOI 10.17487/RFC4301, December 2005, <https://www.rfc-editor.org/info/rfc4301>.

[RFC4585] Ott, J., Wenger, S., Sato, N., Burmeister, C., and J. Rey, "Extended RTP Profile for Real-time Transport Control Protocol (RTCP)-Based Feedback (RTP/AVPF)", RFC 4585, DOI 10.17487/RFC4585, July 2006, <https://www.rfc-editor.org/info/rfc4585>.

[RFC5109] Li, A., Ed., "RTP Payload Format for Generic Forward Error Correction", RFC 5109, DOI 10.17487/RFC5109, December 2007, <https://www.rfc-editor.org/info/rfc5109>.

[RFC5246] Dierks, T. and E. Rescorla, "The Transport Layer Security (TLS) Protocol Version 1.2", RFC 5246, DOI 10.17487/RFC5246, August 2008, <https://www.rfc-editor.org/info/rfc5246>.

[RFC7656] Lennox, J., Gross, K., Nandakumar, S., Salgueiro, G., and B. Burman, Ed., "A Taxonomy of Semantics and Mechanisms for Real-Time Transport Protocol (RTP) Sources", RFC 7656, DOI 10.17487/RFC7656, November 2015, <https://www.rfc-editor.org/info/rfc7656>.

[SMPTE2022-1] SMPTE 2022-1-2007, "Forward Error Correction for Real-Time Video/Audio Transport over IP Networks", 2007.

Authors' Addresses

Mo Zanaty
Cisco
Raleigh, NC
USA

Email: mzanaty@cisco.com

Varun Singh
CALLSTATS I/O Oy
Runeberginkatu 4c A 4
Helsinki 00100
Finland

Email: varun.singh@iki.fi
URI: http://www.callstats.io/

Ali Begen
Networked Media
Konya
Turkey

Email: ali.begen@networked.media

Giridhar Mandyam
Qualcomm Inc.
5775 Morehouse Drive
San Diego, CA 92121
USA

Phone: +1 858 651 7200
Email: mandyam@qti.qualcomm.com