         RTP Payload Format for Raptor Forward Error Correction (FEC)

Abstract

   This document specifies an RTP payload format for the Forward Error
   Correction (FEC) repair data produced by the Raptor FEC Schemes.
   Raptor FEC Schemes are specified for use with the IETF FEC Framework
   that supports the transport of repair data over both UDP and RTP.
   This document specifies the payload format that is required for the
   use of RTP to carry Raptor repair flows.

Status of This Memo

   This is an Internet Standards Track document.

   This document is a product of the Internet Engineering Task Force
   (IETF).  It represents the consensus of the IETF community.  It has
   received public review and has been approved for publication by
   the Internet Engineering Steering Group (IESG).  Further
   information on Internet Standards is available in Section 2 of
   RFC 5741.

   Information about the current status of this document, any
   errata, and how to provide feedback on it may be obtained at
   http://www.rfc-editor.org/info/rfc6682.

Copyright Notice

Table of Contents

1.  Introduction

   The FEC Framework [RFC6363] defines a general framework for the use
   of Forward Error Correction in association with arbitrary packet
   flows, including flows over UDP and RTP [RFC3550].  Forward Error
   Correction operates by generating redundant data packets ("repair
   data") that can be sent independently from the original flow.  At a
   receiver, the original flow can be reconstructed provided a
   sufficient set of redundant data packets and possibly original data
   packets are received.

   The FEC Framework provides for independence between application
   protocols and FEC codes.  The use of a particular FEC code within the
   framework is defined by means of a FEC Scheme, which may then be used
   with any application protocol compliant to the framework.

   Repair data flows may be sent directly over a transport protocol,
   such as UDP, or they may be encapsulated within specialized
   transports for multimedia, such as RTP.

   This document defines the RTP payload format for the Raptor FEC
   Schemes defined in [RFC6681].

2.  Conventions, Definitions, and Acronyms

   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.  Media Format Background

   The Raptor and RaptorQ codes are efficient block-based fountain
   codes, meaning that from any group of source packets (or 'source
   block'), one can generate an arbitrary number of repair packets.  The
   Raptor and RaptorQ codes have the property that the original group of
   source symbols can be recovered with a very high probability from any
   set of symbols (source and repair) only slightly greater in number
   than the original number of source symbols.  The RaptorQ code
   additionally has the property that the probability that the original
   group of source symbols can be recovered from a set of symbols
   (source and repair) equal in number to the original number of source
   symbols is in many cases also very high.

   [RFC6681] defines six FEC Schemes for the use of the Raptor and
   RaptorQ codes with arbitrary packet flows.  The first two schemes are
   fully applicable to arbitrary packet flows (using Raptor and RaptorQ
   respectively).  The third and fourth schemes are slightly optimized
   versions of the first two schemes, which are applicable in

applications with relatively small block sizes.  The fifth and sixth
schemes are variants of the third and fourth schemes, which are
applicable to a single source flow that already has some kind of
identifiable sequence number.  The presence of a sequence number in
the source flow allows for backwards-compatible operation (the source
flows do not need to be modified in order to apply FEC).  In this
case, in the language of the FEC Framework, there is no need for an
explicit FEC Source Payload ID; therefore, it is not included in the
packets.

This document specifies the payload format for RTP repair flows and
can be used with any of the FEC Schemes defined in [RFC6681].

4.  Payload Format for FEC Repair Packets

4.1.  RTP Header Usage

Header fields SHALL be set according to the rules of [RFC3550].  In
addition, the following rules and definitions apply for the RTP
headers used with FEC repair packets:

o  Marker bit: The marker bit SHALL be set to 1 for the last
   protection RTP packet sent for each source block, and otherwise
   set to 0.

o  Payload Type (PT): The payload type codes SHALL be assigned
   dynamically through non-RTP means.  If the Session Description
   Protocol (SDP) is used for signaling, the rules in Section 7
   apply.

o  Timestamp: This field contains the time at which the packet is
   transmitted.  The time SHOULD be as close as possible to the
   packet's actual time of transmission.  The timestamp value has no
   use in the actual FEC protection process.  However,
   implementations SHOULD supply a value that can be used for packet-
   arrival timing or jitter calculations.  The timestamp rate is
   specified using the "rate" media type parameter defined in Section
   6.  The operator SHALL select a "rate" larger than 1000 Hz to
   provide sufficient resolution to the Real-Time Transport Control
   Protocol (RTCP) operations, and the operator SHOULD select the
   rate that matches the rate of the protected source RTP stream.

o  Synchronization Source (SSRC): The SSRC values MUST be set
   according to [RFC3550].  The SSRC value of the RTP repair flow
   MUST be different from the SSRC value of the protected source
   flow.

4.2.  Payload Header

   There is no payload header in this payload format.

4.3.  Payload Data

   Procedures and data formats for the use of Raptor Forward Error
   Correction in a FECFRAME context are fully defined in [RFC6363] and
   [RFC6681] and are not duplicated here.  The procedures of those
   documents apply in order to generate repair data streams to be
   carried by the payload formats defined in this document.

   The RTP Payload SHALL contain a Repair FEC Payload ID as defined in
   [RFC6363] and [RFC6681].

5.  Congestion Control Considerations

   See [RFC6363].

6.  Media Types

6.1.  Registration of the 'application/raptorfec' Media Type

   This RTP payload format is identified using the
   'application/raptorfec' media type that is registered in accordance
   with [RFC4855] and uses the template of [RFC4288].

6.1.1.  Media Type Definition

   Type name: application

   Subtype name: raptorfec

   Required parameters:

   o   rate: The RTP timestamp (clock) rate.  The RTP timestamp (clock)
       rate is specified in Hz and the format is unsigned integer.

   o   raptor-scheme-id: The value of this parameter is the FEC Scheme ID
       for the specific Raptor FEC Scheme that will be used as defined in
       [RFC6681].

   o   Kmax: The value of this parameter is the FEC Framework
       Configuration Information element, Maximum Source Block Length
       (MSBL), as defined in [RFC6681], encoded as a unsigned integer.
       For specific requirements for this value, refer to [RFC6681].

   o  T: The value of this parameter is the FEC Framework Configuration
      Information element, encoding symbol size, as defined in
      [RFC6681], encoded as a unsigned integer.  For specific
      requirements for this value, refer to [RFC6681].

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

   Optional parameters:

   o  P: The value of this parameter is the FEC Framework Configuration
      Information element, Payload ID Format, as defined in [RFC6681].
      The default value of this parameter (when it does not appear
      explicitly) is 'A'.

   Encoding considerations: This media type is framed and binary; see
   Section 4.8 in [RFC4288]

   Security considerations: Please see the security considerations in
   [RFC6363].

   Interoperability considerations:

   Published specification: [RFC6681]

   Applications that use this media type: Real-time multimedia
   applications like video streaming, audio streaming, and video
   conferencing.

   Additional information:

   Magic number(s): <none defined>

   File extension(s): <none defined>

   Macintosh file type code(s): <none defined>

   Person & email address to contact for further information:
   Thomas Stockhammer, stockhammer@nomor.de

   Intended usage: COMMON

   Restrictions on usage: This media type depends on RTP framing, and
   hence is only defined for transfer via RTP [RFC3550].  Transport
   within other framing protocols is not defined at this time.

   Author: Thomas Stockhammer, Nomor Research

Change controller: IETF PAYLOAD working group delegated from the
IESG.

6.2.  Registration of the 'video/raptorfec' Media Type

This RTP payload format is identified using the 'video/raptorfec'
media type that is registered in accordance with [RFC4855] and uses
the template of [RFC4288].

6.2.1.  Media Type Definition

Type name: video

Subtype name: raptorfec

Required parameters:

o  rate: The RTP timestamp (clock) rate.  The RTP timestamp (clock)
   rate is specified in Hz and the format is unsigned integer.

o  raptor-scheme-id: The value of this parameter is the FEC Scheme ID
   for the specific Raptor FEC Scheme that will be used as defined in
   [RFC6681].

o  Kmax: The value of this parameter is the FEC Framework
   Configuration Information element, MSBL, as defined in [RFC6681],
   encoded as a unsigned integer.  For specific requirements for this
   value, refer to [RFC6681].

o  T: The value of this parameter is the FEC Framework Configuration
   Information element, encoding symbol size, as defined in
   [RFC6681], encoded as a unsigned integer.  For specific
   requirements for this value, refer to [RFC6681].

o  repair-window: The maximum time that spans the source packets and
   the corresponding repair packets.  The size of the repair window
   is specified in microseconds, and the format is unsigned integer.

Optional parameters:

o  P: The value of this parameter is the FEC Framework Configuration
   Information element, Payload ID Format, as defined in [RFC6681].
   The default value of this parameter (when it does not appear
   explicitly) is 'A'.

Encoding considerations: This media type is framed and binary; see
Section 4.8 in [RFC4288].

Security considerations: Please see the security considerations in
[RFC6363].

Interoperability considerations:

Published specification: [RFC6681]

Applications that use this media type: Real-time multimedia
applications like video streaming, audio streaming, and video
conferencing.

Additional information:

Magic number(s): <none defined>

File extension(s): <none defined>

Macintosh file type code(s): <none defined>

Person & email address to contact for further information:
Thomas Stockhammer, stockhammer@nomor.de

Intended usage: COMMON

Restrictions on usage: This media type depends on RTP framing, and
hence is only defined for transfer via RTP [RFC3550].  Transport
within other framing protocols is not defined at this time.

Author: Thomas Stockhammer, Nomor Research.

Change controller: IETF PAYLOAD working group delegated from the
IESG.

6.3.  Registration of the 'audio/raptorfec' Media Type

   This RTP payload format is identified using the 'audio/raptorfec'
   media type that is registered in accordance with [RFC4855] and uses
   the template of [RFC4288].

6.3.1.  Media Type Definition

   Type name: audio

   Subtype name: raptorfec

   Required parameters:

   o  rate: The RTP timestamp (clock) rate.  The RTP timestamp (clock)
      rate is specified in Hz and the format is unsigned integer.

   o  raptor-scheme-id: The value of this parameter is the FEC Scheme ID
      for the specific Raptor FEC Scheme that will be used as defined in
      [RFC6681].

   o  Kmax: The value of this parameter is the FEC Framework
      Configuration Information element, MSBL, as defined in [RFC6681],
      encoded as a unsigned integer.  For specific requirements for this
      value, refer to [RFC6681].

   o  T: The value of this parameter is the FEC Framework Configuration
      Information element, encoding symbol size, as defined in
      [RFC6681], encoded as a unsigned integer.  For specific
      requirements for this value, refer to [RFC6681].

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

   Optional parameters:

   o  P: The value of this parameter is the FEC Framework Configuration
      Information element, Payload ID Format, as defined in [RFC6681].
      The default value of this parameter (when it does not appear
      explicitly) is 'A'.

   Encoding considerations: This media type is framed and binary; see
   Section 4.8 in [RFC4288].

   Security considerations: Please see the security considerations in
   [RFC6363].

   Interoperability considerations:

   Published specification: [RFC6681]

   Applications that use this media type: Real-time multimedia
   applications like video streaming, audio streaming, and video
   conferencing.

   Additional information:

   Magic number(s): <none defined>

   File extension(s): <none defined>

   Macintosh file type code(s): <none defined>

   Person & email address to contact for further information:
   Thomas Stockhammer, stockhammer@nomor.de

   Intended usage: COMMON

   Restrictions on usage: This media type depends on RTP framing, and
   hence is only defined for transfer via RTP [RFC3550].  Transport
   within other framing protocols is not defined at this time.

   Author: Thomas Stockhammer, Nomor Research.

   Change controller: IETF PAYLOAD working group delegated from the
   IESG.

6.4.  Registration of the 'text/raptorfec' Media Type

   This RTP payload format is identified using the 'text/raptorfec'
   media type that is registered in accordance with [RFC4855] and uses
   the template of [RFC4288].

6.4.1.  Media Type Definition

   Type name: text

   Subtype name: raptorfec

   Required parameters:

   o  rate: The RTP timestamp (clock) rate.  The RTP timestamp (clock)
      rate is specified in Hz and the format is unsigned integer.

   o  raptor-scheme-id: The value of this parameter is the FEC Scheme ID
      for the specific Raptor FEC Scheme that will be used as defined in
      [RFC6681].

   o  Kmax: The value of this parameter is the FEC Framework
      Configuration Information element, MSBL, as defined in [RFC6681],
      encoded as a unsigned integer.  For specific requirements for this
      value, refer to [RFC6681].

   o  T: The value of this parameter is the FEC Framework Configuration
      Information element, encoding symbol size, as defined in
      [RFC6681], encoded as a unsigned integer.  For specific
      requirements for this value, refer to [RFC6681].

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

Optional parameters:

o  P: The value of this parameter is the FEC Framework Configuration
   Information element, Payload ID Format, as defined in [RFC6681].
   The default value of this parameter (when it does not appear
   explicitly) is 'A'.

Encoding considerations: This media type is framed and binary; see
Section 4.8 in [RFC4288].

Security considerations: Please see the security considerations in
[RFC6363].

Interoperability considerations:

Published specification: [RFC6681]

Applications that use this media type: Real-time multimedia
applications like video streaming, audio streaming, and video
conferencing.

Additional information:

Magic number(s): <none defined>

File extension(s): <none defined>

Macintosh file type code(s): <none defined>

Person & email address to contact for further information:
Thomas Stockhammer, stockhammer@nomor.de

Intended usage: COMMON

Restrictions on usage: This media type depends on RTP framing, and
hence is only defined for transfer via RTP [RFC3550].  Transport
within other framing protocols is not defined at this time.

Author: Thomas Stockhammer, Nomor Research.

Change controller: IETF PAYLOAD working group delegated from the
IESG.

7.  Mapping to the Session Description Protocol (SDP)

   Applications that are using RTP transport commonly use the 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.
   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 above defined payload format media type and its
   parameters SHALL be done according to Section 3 of [RFC4855],
   following the suggestion therein regarding the mapping of payload-
   format-specific parameters into the "a=fmtp" field.

   When the RTP payload formats defined in this document are used, the
   media type parameters defined above MUST use the media types in this
   document and MUST NOT use those specified in [RFC6364].

8.  Offer/Answer Considerations

   When offering Raptor FEC over RTP using SDP in an Offer/Answer model
   [RFC3264], the following considerations apply:

   o  Each combination of the Kmax and T parameters produces different
      FEC data and is not compatible with any other combination.  A
      sender application MAY desire to provide multiple offers with
      different sets of Kmax and T values, which is possible as long as
      the parameter values are valid.  The receiver SHOULD normally
      choose the offer with the largest value of the product of Kmax and
      T that it supports.

   o  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.

   o  When the P parameter is not present, the receiver MUST use FEC
      Payload ID Format A.  In an answer that selects an offer in which
      the P parameter was omitted, the P parameter MUST either be
      omitted, or included with value "A".

9.  Declarative SDP 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:

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

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

10.  Repair Flow Generation and Recovery Procedures

10.1.  Overview

   This document only specifies repair flow construction when the repair
   packets are delivered with RTP.  Source packet construction is
   covered in [RFC6681].  This section provides an overview on how to
   generate a repair flow, including the repair packets and how to
   reconstruct missing source packets from a set of available source and
   repair packets.  Detailed algorithms for the generation of Raptor and
   RaptorQ symbols are provided in [RFC5053] and [RFC6330],
   respectively.

   As per the FEC Framework document [RFC6363], the FEC Framework
   Configuration Information includes, among others, the identification
   of the repair flow(s) and the source flow(s).  Methods to convey FEC
   Framework Configuration Information are provided in [FEC-SIG].
   Specifically, the reader is referred to the SDP elements document
   [RFC6364], which describes the usage of the 'SDP' encoding format as
   an example encoding format for FEC Framework Configuration
   Information.

   For the generation of a repair flow:

   o  repair packets SHALL be constructed according to Section 10.2, and

   o  RTCP SHALL be used according to Section 10.3.

   For the reconstruction of a source packet of a source RTP session at
   the receiver, based on the availability of a source RTP session and a
   repair RTP session, the procedures in Section 10.4 may be used.

10.2.  Repair Packet Construction

   The construction of the repair packet is fully specified in Section
   4.  A repair packet is constructed by the concatenation of

   o  an RTP header as specified in Section 4.1, and

   o  payload data as defined in Section 4.3.

   Repair Packet Construction may make use of the Sender Operation for
   RTP repair flows as specified in see [RFC6363], Section 4.2.

10.3.  Usage of RTCP

   RTCP SHALL be used according to [RFC3550].  If the repair RTP session
   is sent in a separate RTP session, the two sessions MUST be
   associated using RTCP CNAME (Canonical Name).

10.4.  Source Packet Reconstruction

   Source Packet Reconstruction may make use of the receiver operation
   for the case of RTP repair flows as specified in [RFC6363], Section
   4.3.  Depending on the FEC Scheme using the ones defined in
   [RFC6681], the appropriate source blocks are formed.  If enough data
   for decoding any or all of the missing source payloads in the source
   block has been received, the respective FEC decoding procedures are
   applied.

   In case the FEC Scheme uses Raptor codes as defined in [RFC5053],
   then the Example FEC Decoder, as specified in [RFC5053], Section 5.5,
   may be used.

   In case the FEC Scheme uses RaptorQ codes as defined in [RFC6330],
   then the Example FEC Decoder, as specified in [RFC6330], Section 5.4,
   may be used.

11.  Session Description Protocol (SDP) Example

   This section provides an SDP [RFC4566] example.  Assume we have one
   source video stream (mid:S1) and one FEC repair stream (mid:R1).  The
   'group' attribute and the FEC grouping semantics defined in [RFC5888]
   and [RFC5956], respectively, are used to associate source and repair
   flows.  We form one FEC group with the "a=group:FEC S1 R1" line.  The
   source and repair streams are sent to the same port on different
   multicast groups.  The repair window is set to 200 ms.

```
v=0
o=ali 1122334455 1122334466 IN IP4 fec.example.com
s=Raptor RTP FEC Example
t=0 0
a=group:FEC-FR S1 R1
m=video 30000 RTP/AVP 100
c=IN IP4 233.252.0.1/127
a=rtpmap:100 MP2T/90000
a=fec-source-flow: id=0
a=mid:S1
m=application 30000 RTP/AVP 110
c=IN IP4 233.252.0.2/127
a=rtpmap:110 raptorfec/90000
a=fmtp:110 raptor-scheme-id=1; Kmax=8192; T=128;
        P=A; repair-window=200000
a=mid:R1
```

12.  IANA Considerations

   IANA has registered 'application/raptorfec' as specified in Section
   6.1.1, 'video/raptorfec' as specified in Section 6.2.1,
   'audio/raptorfec' as specified in Section 6.3.1, and 'text/raptorfec'
   as specified in Section 6.4.1.  The media type has also been added to
   the IANA registry for "RTP Payload Format media types"
   (http://www.iana.org/assignments/rtp-parameters).

13.  Security Considerations

   Security Considerations related to the use of the FEC Framework are
   addressed in [RFC6363].  These considerations apply in full to users
   of the RTP payload formats defined in this document, since these are
   defined in terms of the FEC Framework.

   No further security considerations related specifically to the Raptor
   FEC Schemes defined in [RFC6681] have been identified.

   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 can 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 exist.

14.  References

14.1.  Normative References

   [RFC2119]  Bradner, S., "Key words for use in RFCs to Indicate
              Requirement Levels", BCP 14, RFC 2119, March 1997.

   [RFC3550]  Schulzrinne, H., Casner, S., Frederick, R., and V.
              Jacobson, "RTP: A Transport Protocol for Real-Time
              Applications", STD 64, RFC 3550, July 2003.

   [RFC4288]  Freed, N. and J. Klensin, "Media Type Specifications and
              Registration Procedures", BCP 13, RFC 4288, December 2005.

   [RFC4855]  Casner, S., "Media Type Registration of RTP Payload
              Formats", RFC 4855, February 2007.

   [RFC6363]  Watson, M., Begen, A., and V. Roca, "Forward Error
              Correction (FEC) Framework", RFC 6363, October 2011.

   [RFC6364]  Begen, A., "Session Description Protocol Elements for the
              Forward Error Correction (FEC) Framework", RFC 6364,
              October 2011.

   [RFC6681]  Watson, M., Stockhammer, T., and M. Luby, "Raptor Forward
              Error Correction (FEC) Schemes for FECFRAME", RFC 6681,
              August 2012.

   [RFC4566]  Handley, M., Jacobson, V., and C. Perkins, "SDP: Session
              Description Protocol", RFC 4566, July 2006.

   [RFC3264]  Rosenberg, J. and H. Schulzrinne, "An Offer/Answer Model
              with Session Description Protocol (SDP)", RFC 3264, June
              2002.

   [RFC3711]  Baugher, M., McGrew, D., Naslund, M., Carrara, E., and K.
              Norrman, "The Secure Real-time Transport Protocol (SRTP)",
              RFC 3711, March 2004.

   [RFC4301]  Kent, S. and K. Seo, "Security Architecture for the
              Internet Protocol", RFC 4301, December 2005.

   [RFC5246]  Dierks, T. and E. Rescorla, "The Transport Layer Security
              (TLS) Protocol Version 1.2", RFC 5246, August 2008.

   [RFC5053]  Luby, M., Shokrollahi, A., Watson, M., and T. Stockhammer,
              "Raptor Forward Error Correction Scheme for Object
              Delivery", RFC 5053, October 2007.

   [RFC6330]  Luby, M., Shokrollahi, A., Watson, M., Stockhammer, T.,
              and L. Minder, "RaptorQ Forward Error Correction Scheme
              for Object Delivery", RFC 6330, August 2011.

14.2.  Informative References

   [RFC2326]  Schulzrinne, H., Rao, A., and R. Lanphier, "Real Time
              Streaming Protocol (RTSP)", RFC 2326, April 1998.

   [RFC2974]  Handley, M., Perkins, C., and E. Whelan, "Session
              Announcement Protocol", RFC 2974, October 2000.

   [RFC5888]  Camarillo, G. and H. Schulzrinne, "The Session Description
              Protocol (SDP) Grouping Framework", RFC 5888, June 2010.

   [RFC5956]  Begen, A., "Forward Error Correction Grouping Semantics in
              the Session Description Protocol", RFC 5956, September
              2010.

   [FEC-SIG]  Asati, R., "Methods to convey FEC Framework Configuration
              Information", Work in Progress, February 2012.

Authors' Addresses

    Mark Watson
    Netflix
    100 Winchester Circle
    Los Gatos, CA 95032
    United States

    EMail: watsonm@netflix.com


    Thomas Stockhammer
    Nomor Research
    Brecherspitzstrasse 8
    Munich 81541
    Germany

    EMail: stockhammer@nomor.de


    Michael Luby
    Qualcomm Research Berkeley
    2030 Addison Street
    Berkeley, CA 94704
    United States

    EMail: luby@qualcomm.com