Network Working Group   M. Luby
Request for Comments: 3452   Digital Fountain
Category: Experimental   L. Vicisano
                                                               Cisco
                                                           J. Gemmell
                                                           Microsoft
                                                            L. Rizzo
                                                          Univ. Pisa
                                                          M. Handley
                                                                ICIR
                                                         J. Crowcroft
                                                      Cambridge Univ.
                                                       December 2002

Forward Error Correction (FEC) Building Block

Status of this Memo

Copyright Notice

Abstract

   This document generally describes how to use Forward Error Correction
   (FEC) codes to efficiently provide and/or augment reliability for
   data transport. The primary focus of this document is the
   application of FEC codes to one-to-many reliable data transport using
   IP multicast. This document describes what information is needed to
   identify a specific FEC code, what information needs to be
   communicated out-of-band to use the FEC code, and what information is
   needed in data packets to identify the encoding symbols they carry.
   The procedures for specifying FEC codes and registering them with the
   Internet Assigned Numbers Authority (IANA) are also described. This
   document should be read in conjunction with and uses the terminology
   of the companion document titled, "The Use of Forward Error
   Correction (FEC) in Reliable Multicast".

Table of Contents

1. Introduction

    This document describes how to use Forward Error Correction (FEC)
    codes to provide support for reliable delivery of content using IP
    multicast. This document should be read in conjunction with and uses
    the terminology of the companion document [4], which describes the
    use of FEC codes within the context of reliable IP multicast
    transport and provides an introduction to some commonly used FEC
    codes.

    This document describes a building block as defined in RFC 3048 [9].
    This document is a product of the IETF RMT WG and follows the general
    guidelines provided in RFC 3269 [3].

    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 [2].

    Statement of Intent

       This memo contains part of the definitions necessary to fully
       specify a Reliable Multicast Transport protocol in accordance with
       RFC 2357. As per RFC 2357, the use of any reliable multicast
       protocol in the Internet requires an adequate congestion control
       scheme.

While waiting for such a scheme to be available, or for an existing scheme to be proven adequate, the Reliable Multicast Transport working group (RMT) publishes this Request for Comments in the "Experimental" category.

It is the intent of RMT to re-submit this specification as an IETF Proposed Standard as soon as the above condition is met.

2. Rationale

FEC codes are a valuable basic component of any transport protocol that is to provide reliable delivery of content. Using FEC codes is valuable in the context of IP multicast and reliable delivery because FEC encoding symbols can be useful to all receivers for reconstructing content even when the receivers have received different encoding symbols. Furthermore, FEC codes can ameliorate or even eliminate the need for feedback from receivers to senders to request retransmission of lost packets.

The goal of the FEC building block is to describe functionality directly related to FEC codes that is common to all reliable content delivery IP multicast protocols, and to leave out any additional functionality that is specific to particular protocols. The primary functionality described in this document that is common to all such protocols that use FEC codes are FEC encoding symbols for an object that is included in packets that flow from a sender to receivers. This document for example does not describe how receivers may request transmission of particular encoding symbols for an object. This is because although there are protocols where requests for transmission are of use, there are also protocols that do not require such requests.

The companion document [4] should be consulted for a full explanation of the benefits of using FEC codes for reliable content delivery using IP multicast. FEC codes are also useful in the context of unicast, and thus the scope and applicability of this document is not limited to IP multicast.

3. Functionality

This section describes FEC information that is either to be sent out-of-band or in packets. The FEC information is associated with transmission of data about a particular object. There are three classes of packets that may contain FEC information: data packets, session-control packets and feedback packets. They generally contain different kinds of FEC information. Note that some protocols may not use session-control or feedback packets.

Data packets may sometimes serve as session-control packets as well;
both data and session-control packets generally travel downstream
from the sender towards receivers and are sent to a multicast channel
or to a specific receiver using unicast.

As a general rule, feedback packets travel upstream from receivers to
the sender. Sometimes, however, they might be sent to a multicast
channel or to another receiver or to some intermediate node or
neighboring router that provides recovery services.

This document specifies the FEC information that must be carried in
data packets and the other FEC information that must be communicated
either out-of-band or in data packets. This document does not
specify out-of-band methods nor does it specify the way out-of-band
FEC information is associated with FEC information carried in data
packets. These methods must be specified in a complete protocol
instantiation that uses the FEC building block. FEC information is
classified as follows:

1) FEC Encoding ID

   Identifies the FEC encoder being used and allows receivers to
   select the appropriate FEC decoder. The value of the FEC Encoding
   ID MUST be the same for all transmission of data related to a
   particular object, but MAY vary across different transmissions of
   data about different objects, even if transmitted to the same set
   of multicast channels and/or using a single upper-layer session.
   The FEC Encoding ID is subject to IANA registration.

2) FEC Instance ID

   Provides a more specific identification of the FEC encoder being
   used for an Under-Specified FEC scheme. This value is not used
   for Fully-Specified FEC schemes. (See Section 3.1 for the
   definition of Under-Specified and Fully-Specified FEC schemes.)
   The FEC Instance ID is scoped by the FEC Encoding ID, and is
   subject to IANA registration.

3) FEC Payload ID

   Identifies the encoding symbol(s) in the payload of the packet.
   The types and lengths of the fields in the FEC Payload ID, i.e.,
   the format of the FEC Payload ID, are determined by the FEC
   Encoding ID. The full specification of each field MUST be
   uniquely determined by the FEC Encoding ID for Fully-Specified FEC
   schemes, and MUST be uniquely determined by the combination of the
   FEC Encoding ID and the FEC Instance ID for Under-Specified FEC
   schemes. As an example, for the Under-Specified FEC scheme with

FEC Encoding ID 129 defined in Section 5.1, the fields in the FEC
Payload ID are a 32-bit Source Block Number followed by a 32-bit
Encoding Symbol ID, where the full specification of both of these
fields depends on the FEC Instance ID.

4) FEC Object Transmission Information

This is information regarding the encoding of a specific object
needed by the FEC decoder. As an example, for the Under-Specified
FEC scheme with FEC Encoding ID 129 defined in Section 5.1, this
information might include the lengths of the different source
blocks that make up the object and the overall object length.
This might also include specific parameters of the FEC encoder.

The FEC Encoding ID, FEC Instance ID (for Under-Specified FEC
schemes) and the FEC Object Transmission Information can be sent to a
receiver within the data packet headers, within session control
packets, or by some other means. In any case, the means for
communicating this to a receiver is outside the scope of this
document. The FEC Payload ID MUST be included in the data packet
header fields, as it provides a description of the encoding symbols
contained in the packet.

3.1. FEC Encoding ID and FEC Instance ID

The FEC Encoding ID is a numeric index that identifies a specific FEC
scheme OR a class of encoding schemes that share the same FEC Payload
ID format.

An FEC scheme is a Fully-Specified FEC scheme if the encoding scheme
is formally and fully specified, in a way that independent
implementors can implement both encoder and decoder from a
specification that is an IETF RFC. The FEC Encoding ID uniquely
identifies a Fully-Specified FEC scheme. Companion documents of this
specification may specify Fully-Specified FEC schemes and associate
them with FEC Encoding ID values.

These documents MUST also specify a format for the FEC Payload ID and
specify the information in the FEC Object Transmission Information.

It is possible that a FEC scheme may not be a Fully-Specified FEC
scheme, because either a specification is simply not available or a
party exists that owns the encoding scheme and is not willing to
disclose the algorithm or specification. We refer to such an FEC
encoding schemes as an Under-Specified FEC scheme. The following
holds for an Under-Specified FEC scheme:

o The fields and their formats of the FEC Payload ID and the specific
  information in the FEC Object Transmission Information MUST be
  defined for the Under-Specified FEC scheme.

o A value for the FEC Encoding ID MUST be reserved and associated
  with the fields and their formats of the FEC Payload ID and the
  specific information in the FEC Object Transmission Information.
  An already reserved FEC Encoding ID value MUST be reused if the
  associated FEC Payload ID has the same fields and formats and the
  FEC Object Transmission Information has same information as the
  ones needed for the new Under-Specified FEC scheme.

o A value for the FEC Instance ID MUST be reserved.

An Under-Specified FEC scheme is fully identified by the tuple (FEC
Encoding ID, FEC Instance ID). The tuple MUST identify a single
scheme that has at least one implementation. The party that owns
this tuple MUST be able to provide information on how to obtain the
Under-Specified FEC scheme identified by the tuple, e.g., a pointer
to a publicly available reference-implementation or the name and
contacts of a company that sells it, either separately or embedded in
another product.

Different Under-Specified FEC schemes that share the same FEC
Encoding ID -- but have different FEC Instance IDs -- also share the
same fields and corresponding formats of the FEC Payload ID and
specify the same information in the FEC Object Transmission
Information.

This specification reserves the range 0-127 for the values of FEC
Encoding IDs for Fully-Specified FEC schemes and the range 128-255
for the values of Under-Specified FEC schemes.

3.2. FEC Payload ID and FEC Object Transmission Information

A document that specifies an FEC scheme and reserves a value of FEC
Encoding ID MUST define the fields and their packet formats for the
FEC Payload ID and specify the information in the FEC Object
Transmission Information according to the needs of the encoding
scheme. This applies to documents that reserve values of FEC
Encoding IDs for both Fully-Specified and Under-Specified FEC
schemes.

The specification of the fields and their packet formats for the FEC
Payload ID MUST specify the meaning of the fields and their format
down to the level of specific bits. The total length of all the

fields in the FEC Payload ID MUST have a length that is a multiple of
a 4-byte word. This requirement facilitates the alignment of packet
fields in protocol instantiations.

4. Applicability Statement

The FEC building block applies to creating and sending encoding
symbols for objects that are to be reliably transported using IP
multicast or unicast. The FEC building block does not provide higher
level session support. Thus, for example, many objects may be
transmitted within the same session, in which case a higher level
building block may carry a unique Transport Object ID (TOI) for each
object in the session to allow the receiver to demultiplex packets
within the session based on the TOI within each packet. As another
example, a receiver may subscribe to more than one session at a time.

In this case a higher level building block may carry a unique
Transport Session ID (TSI) for each session to allow the receiver to
demultiplex packets based on the TSI within each packet.

Other building blocks may supply direct support for carrying out-of-
band information directly relevant to the FEC building block to
receivers. For example, the length of the object is part of the FEC
Object Transmission Information that may in some cases be
communicated out-of-band to receivers, and one mechanism for
providing this to receivers is within the context of another building
block that provides this information.

Some protocols may use FEC codes as a mechanism for repairing the
loss of packets. Within the context of FEC repair schemes, feedback
packets are (optionally) used to request FEC retransmission. The
FEC-related information present in feedback packets usually contains
an FEC Block ID that defines the block that is being repaired, and
the number of Repair Symbols requested. Although this is the most
common case, variants are possible in which the receivers provide
more specific information about the Repair Symbols requested (e.g.,
an index range or a list of symbols accepted). It is also possible
to include multiple requests in a single feedback packet. This
document does not provide any detail about feedback schemes used in
combination with FEC nor the format of FEC information in feedback
packets. If feedback packets are used in a complete protocol
instantiation, these details must be provided in the protocol
instantiation specification.

The FEC building block does not provide any support for congestion
control. Any complete protocol MUST provide congestion control that
conforms to RFC 2357 [5], and thus this MUST be provided by another
building block when the FEC building block is used in a protocol.

A more complete description of the applicability of FEC codes can be found in the companion document [4].
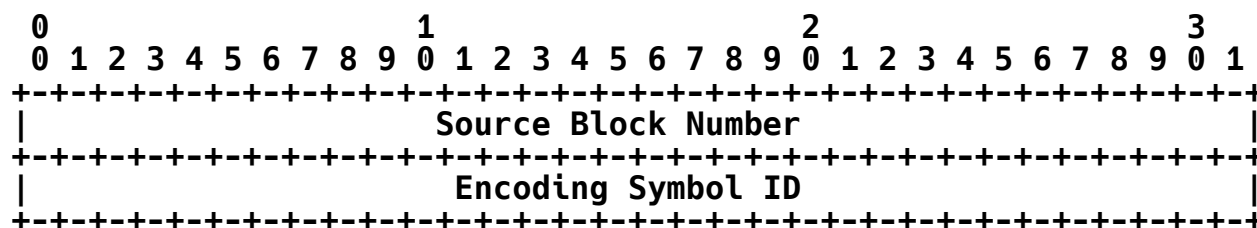
5. Packet Header Fields

This section specifies the FEC Encoding ID, the associated FEC Payload ID format, and the specific information in the FEC Object Transmission Information for a number of known Under-Specified FEC schemes. Under-Specified FEC schemes that use the same FEC Payload ID fields, formats, and specific information in the FEC Object Transmission Information (as for one of the FEC Encoding IDs specified in this section) MUST use the corresponding FEC Encoding ID. Other FEC Encoding IDs may be specified for other Under-Specified FEC schemes in companion documents.

5.1. Small Block, Large Block and Expandable FEC Codes

This subsection reserves the FEC Encoding ID value 128 for the Under-Specified FEC schemes described in [4] that are called Small Block FEC codes, Large Block FEC codes and Expandable FEC codes.

The FEC Payload ID is composed of a Source Block Number and an Encoding Symbol ID structured as follows:

```
 0                   1                   2                   3
 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                      Source Block Number                      |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                       Encoding Symbol ID                      |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

The Source Block Number identifies from which source block of the object the encoding symbol(s) in the payload are generated. These blocks are numbered consecutively from 0 to N-1, where N is the number of source blocks in the object.

The Encoding Symbol ID identifies which specific encoding symbol(s) generated from the source block are carried in the packet payload. The exact details of the correspondence between Encoding Symbol IDs and the encoding symbol(s) in the packet payload are dependent on the particular encoding algorithm used as identified by the FEC Encoding ID and by the FEC Instance ID, and these details may be proprietary.

The FEC Object Transmission Information has the following specific information:

o The FEC Encoding ID 128.

o The FEC Instance ID associated with the FEC Encoding ID 128 to be
  used.

o The total length of the object in bytes.

o The number of source blocks that the object is partitioned into,
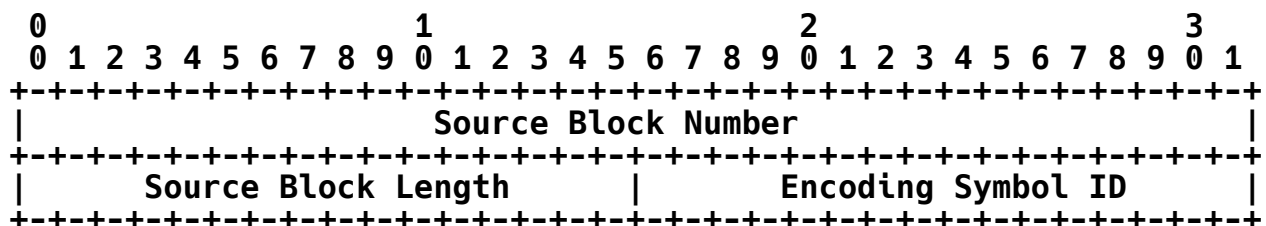 and the length of each source block in bytes.

To understand how this out-of-band information is communicated, one
must look outside the scope of this document. One example may be
that the source block lengths may be derived by a fixed algorithm
from the object length. Another example may be that all source
blocks are the same length and this is what is passed out-of-band to
the receiver. A third example could be that the full sized source
block length is provided and this is the length used for all but the
last source block, which is calculated based on the full source block
length and the object length.

5.2. Small Block Systematic FEC Codes

This subsection reserves the FEC Encoding ID value 129 for the
Under-Specified FEC schemes described in [4] that are called Small
Block Systematic FEC codes. For Small Block Systematic FEC codes,
each source block is of length at most 65536 source symbols.

Although these codes can generally be accommodated by the FEC
Encoding ID described in Section 5.1, a specific FEC Encoding ID is
defined for Small Block Systematic FEC codes to allow more
flexibility and to retain header compactness. The small source block
length and small expansion factor that often characterize systematic
codes may require the data source to frequently change the source
block length. To allow the dynamic variation of the source block
length and to communicate it to the receivers with low overhead, the
block length is included in the FEC Payload ID.

The FEC Payload ID is composed of the Source Block Number, Source
Block Length and the Encoding Symbol ID:

```
 0                   1                   2                   3
 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                     Source Block Number                       |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|       Source Block Length         |        Encoding Symbol ID     |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

The Source Block Number identifies from which source block of the
object the encoding symbol(s) in the payload are generated. These
blocks are numbered consecutively from 0 to N-1, where N is the
number of source blocks in the object.

The Source Block Length is the length in units of source symbols of
the source block identified by the Source Block Number.

The Encoding Symbol ID identifies which specific encoding symbol(s)
generated from the source block are carried in the packet payload.
Each encoding symbol is either an original source symbol or a
redundant symbol generated by the encoder. The exact details of the
correspondence between Encoding Symbol IDs and the encoding symbol(s)
in the packet payload are dependent on the particular encoding
algorithm used as identified by the FEC Encoding ID and by the FEC
Instance ID, and these details may be proprietary.

The FEC Object Transmission Information has the following specific
information:

o The FEC Encoding ID 129.

o The FEC Instance ID associated with the FEC Encoding ID 129 to be
  used.

o The total length of the object in bytes.

o The maximum number of encoding symbols that can be generated for
  any source block. This field is provided for example to allow
  receivers to preallocate buffer space that is suitable for decoding
  to recover any source block.

o For each source block, the length in bytes of encoding symbols for
  the source block.

How this out-of-band information is communicated is outside the scope
of this document. As an example the length in bytes of encoding
symbols for each source block may be the same for all source blocks.
As another example, the encoding symbol length may be the same for
all source blocks of a given object and this length is communicated
for each object. As a third example, it may be that there is a
threshold value I, and for all source blocks consisting of less than
I source symbols, the encoding symbol length is one fixed number of
bytes, but for all source blocks consisting of I or more source
symbols, the encoding symbol length is a different fixed number of
bytes.

Note that each encoding symbol, i.e., each source symbol and
redundant symbol, must be the same length for a given source block,
and this implies that each source block length is a multiple of its
encoding symbol length. If the original source block length is not a
multiple of the encoding symbol length, it is up to the sending
application to appropriately pad the original source block to form
the source block to be encoded, and to communicate this padding to
the receiving application. The form of this padding, if used, and
how it is communicated to the receiving application, is outside the
scope of this document, and must be handled at the application level.

6. Requirements from other building blocks

The FEC building block does not provide any support for congestion
control. Any complete protocol MUST provide congestion control that
conforms to RFC 2357 [5], and thus this MUST be provided by another
building block when the FEC building block is used in a protocol.

There are no other specific requirements from other building blocks
for the use of this FEC building block. However, any protocol that
uses the FEC building block will inevitably use other building blocks
for example to provide support for sending higher level session
information within data packets containing FEC encoding symbols.

7. Security Considerations

Data delivery can be subject to denial-of-service attacks by
attackers which send corrupted packets that are accepted as
legitimate by receivers. This is particularly a concern for
multicast delivery because a corrupted packet may be injected into
the session close to the root of the multicast tree, in which case
the corrupted packet will arrive to many receivers. This is
particularly a concern for the FEC building block because the use of
even one corrupted packet containing encoding data may result in the
decoding of an object that is completely corrupted and unusable. It
is thus RECOMMENDED that the decoded objects be checked for integrity
before delivering objects to an application. For example, an MD5
hash [8] of an object may be appended before transmission, and the
MD5 hash is computed and checked after the object is decoded but
before it is delivered to an application. Moreover, in order to
obtain strong cryptographic integrity protection a digital signature
verifiable by the receiver SHOULD be computed on top of such a hash
value. It is also RECOMMENDED that a packet authentication protocol
such as TESLA [7] be used to detect and discard corrupted packets
upon arrival. Furthermore, it is RECOMMENDED that Reverse Path
Forwarding checks be enabled in all network routers and switches

along the path from the sender to receivers to limit the possibility
of a bad agent successfully injecting a corrupted packet into the
multicast tree data path.

Another security concern is that some FEC information may be obtained
by receivers out-of-band in a session description, and if the session
description is forged or corrupted then the receivers will not use
the correct protocol for decoding content from received packets. To
avoid these problems, it is RECOMMENDED that measures be taken to
prevent receivers from accepting incorrect session descriptions,
e.g., by using source authentication to ensure that receivers only
accept legitimate session descriptions from authorized senders.

8.   IANA Considerations

Values of FEC Encoding IDs and FEC Instance IDs are subject to IANA
registration. FEC Encoding IDs and FEC Instance IDs are
hierarchical: FEC Encoding IDs scope ranges of FEC Instance IDs.
Only FEC Encoding IDs that correspond to Under-Specified FEC schemes
scope a corresponding set of FEC Instance IDs.

The FEC Encoding ID is a numeric non-negative index. In this
document, the range of values for FEC Encoding IDs is 0 to 255.
Values from 0 to 127 are reserved for Fully-Specified FEC schemes and
Values from 128 to 255 are reserved for Under-Specified FEC schemes,
as described in more detail in Section 3.1. This specification
already assigns the values 128 and 129, as described in Section 5.

Each FEC Encoding ID assigned to an Under-Specified FEC scheme scopes
an independent range of FEC Instance IDs (i.e., the same value of FEC
Instance ID can be reused for different FEC Encoding IDs). An FEC
Instance ID is a numeric non-negative index.

8.1. Explicit IANA Assignment Guidelines

This document defines a name-space for FEC Encoding IDs named:

                         ietf:rmt:fec:encoding

IANA has established and manages the new registry for the
"ietf:rmt:fec:encoding" name-space. The values that can be assigned
within the "ietf:rmt:fec:encoding" name-space are numeric indexes in
the range [0, 255], boundaries included. Assignment requests are
granted on a "Specification Required" basis as defined in RFC 2434
[6]: An IETF RFC MUST exist and specify the FEC Payload ID fields and
formats as well as the FEC Object Transmission Information for the
value of "ietf:rmt:fec:encoding" (FEC Encoding ID) being assigned by
IANA (see Section 3.1 for more details). Note that the values 128

and 129 of "ietf:rmt:fec:encoding" are already assigned by this
document as described in Section 5.

This document also defines a name-space for FEC Instance IDs named:

                    ietf:rmt:fec:encoding:instance

The "ietf:rmt:fec:encoding:instance" name-space is a sub-name-space
associated with the "ietf:rmt:fec:encoding" name-space. Each value
of "ietf:rmt:fec:encoding" assigned in the range [128, 255] has a
separate "ietf:rmt:fec:encoding:instance" sub-name-space that it
scopes. Values of "ietf:rmt:fec:encoding" in the range [0, 127] do
not scope a "ietf:rmt:fec:encoding:instance" sub-name-space.

The values that can be assigned within each
"ietf:rmt:fec:encoding:instance" sub-name-space are non-negative
numeric indices. Assignment requests are granted on a "First Come
First Served" basis as defined in RFC 2434 [6]. The same value of
"ietf:rmt:fec:encoding:instance" can be assigned within multiple
distinct sub-name-spaces, i.e., the same value of
"ietf:rmt:fec:encoding:instance" can be used for multiple values of
"ietf:rmt:fec:encoding".

Requestors of "ietf:rmt:fec:encoding:instance" assignments MUST
provide the following information:

o The value of "ietf:rmt:fec:encoding" that scopes the
  "ietf:rmt:fec:encoding:instance" sub-name-space. This must be in
  the range [128, 255].

o Point of contact information

o A pointer to publicly accessible documentation describing the
  Under-Specified FEC scheme, associated with the value of
  "ietf:rmt:fec:encoding:instance" assigned, and a way to obtain it
  (e.g., a pointer to a publicly available reference-implementation
  or the name and contacts of a company that sells it, either
  separately or embedded in a product).

It is the responsibility of the requestor to keep all the above
information up to date.

9. Intellectual Property Disclosure

The IETF has been notified of intellectual property rights claimed in
regard to some or all of the specification contained in this
document. For more information consult the online list of claimed
rights.

10. Acknowledgments

Brian Adamson contributed to this document by shaping Section 5.2 and providing general feedback. We also wish to thank Vincent Roca, Justin Chapweske and Roger Kermode for their extensive comments.

11. References

[1] Bradner, S., "The Internet Standards Process -- Revision 3", BCP 9, RFC 2026, October 1996.

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

[3] Kermode, R. and L. Vicisano, "Author Guidelines for Reliable Multicast Transport (RMT) Building Blocks and Protocol Instantiation documents", RFC 3269, April 2002.

[4] Luby, M., Vicisano, L., Gemmell, J., Rizzo, L., Handley, M. and J. Crowcroft, "The Use of Forward Error Correction (FEC) in Reliable Multicast", RFC 3453, December 2002.

[5] Mankin, A., Romanow, A., Bradner, S. and V. Paxson, "IETF Criteria for Evaluating Reliable Multicast Transport and Application Protocols", RFC 2357, June 1998.

[6] Narten, T. and H. Alvestrand, "Guidelines for Writing an IANA Considerations Section in RFCs", BCP 26, RFC 2434, October 1998.

[7] Perrig, A., Canetti, R., Song, D. and J. Tygar, "Efficient and Secure Source Authentication for Multicast", Network and Distributed System Security Symposium, NDSS 2001, pp. 35-46, February 2001.

[8] Rivest, R., "The MD5 Message-Digest Algorithm", RFC 1321, April 1992.

[9] Whetten, B., Vicisano, L., Kermode, R., Handley, M., Floyd, S. and M. Luby, "Reliable Multicast Transport Building Blocks for One-to-Many Bulk-Data Transfer", RFC 3048, January 2001.

12. Authors' Addresses

Michael Luby
Digital Fountain, Inc.
39141 Civic Center Drive
Suite 300
Fremont, CA 94538

EMail: luby@digitalfountain.com

Lorenzo Vicisano
Cisco Systems, Inc.
170 West Tasman Dr.,
San Jose, CA, USA, 95134

EMail: lorenzo@cisco.com

Jim Gemmell
Microsoft Research
455 Market St. #1690
San Francisco, CA, 94105

EMail: jgemmell@microsoft.com

Luigi Rizzo
Dip. di Ing. dell'Informazione
Universita` di Pisa
via Diotisalvi 2, 56126 Pisa, Italy

EMail: luigi@iet.unipi.it

Mark Handley
ICSI Center for Internet Research
1947 Center St.
Berkeley CA, USA, 94704

EMail: mjh@icir.org

Jon Crowcroft
Marconi Professor of Communications Systems
University of Cambridge
Computer Laboratory
William Gates Building
J J Thomson Avenue
Cambridge
CB3 0FD

EMail: Jon.Crowcroft@cl.cam.ac.uk

13. Full Copyright Statement

Acknowledgement