FECFRAME Working Group
Internet-Draft
Intended status: Informational
Expires: June 5, 2008

M. Watson
Digital Fountain

December 3, 2007

FECFRAME requirements
draft-ietf-fecframe-req-02

Status of this Memo

Copyright Notice

Abstract

This document defines requirements for a "FEC Framework" to be defined by the IETF FECFRAME working group. The object of this group is primarily to develop specifications for using forward error correction (FEC) codes with applications in the Internet to provide protection against packet loss.

Table of Contents

1. Introduction

This document defines requirements for a "FEC Framework" to be defined by the IETF FECFRAME working group. The purpose of this working group is defined in the working group charter as follows:

> "The object of this group is to develop specifications for using forward error correction (FEC) codes with applications in the Internet to provide protection against packet loss. The group will develop a protocol framework for application of FEC codes to arbitrary packet flows over unreliable transport protocols over both IP multicast and unicast."

This document defines requirements for this protocol framework. Both 'essential' ('SHALL') and 'non-essential' ('SHOULD') requirements are considered.

A 'protocol framework' is a partial specification of a protocol, along with a formal description of the missing aspects which are required to form a full protocol specification - i.e. a protocol framework is a protocol with 'holes' and a detailed description of the 'shape' of those holes. Protocol frameworks provide for maximum commonality between different complete protocols which provide similar functions and therefore simplify implementation and understanding of a set of alternative protocols which perform similar functions. In this case, support for different complete protocols is valuable for two reasons. Firstly because there exist many different forward error correction codes, with different properties in terms of error correction capability, computational complexity, flexibility and intellectual property rights. Secondly, there are many applications which could benefit from the use of forward error correction. The FEC framework therefore replaces the "full mesh" of application/FEC code combinations with a single general approach which specifies how any FEC code meeting the FEC code requirments defined in the framework can be used with any application meeting the application requirements defined in the framework.

The FEC protocol framework must therefore define as much as possible of a protocol for providing forward error correction for arbitrary packet flows over unreliable transport, without defining a particular FEC code or assuming a particular application. Furthermore, the protocol framework specification will define a clear interface between the specified parts and the unspecified, FEC-code-specific and application-specific, parts. For this purpose, the building block techniques applied in the Reliable Multicast (RMT) working group will be re-used, specifically the FEC Building Block [RFC5052]

The term "Forward Error Correction" refers here to application/

transport layer techniques for recovering lost packets of data. More accurately, the term "Forward Erasure Correction" should be used. In many contexts the term "Application Layer FEC (AL-FEC)" is also used, although the mechanisms considered here could be considered as either application or transport layer (the important point being that they are end-to-end).

Generally, an "FEC Code" is defined in terms of the operations required to construct encoded data from source data (at an encoder) and to reconstruct source data from encoded data (at a decoder). In order to apply an FEC Code to arbitrary packet flows, additional elements are required such as protocol elements to identify encoded data within packets, pre-processing of source data (e.g. segmentation and/or addition of FEC-related indications into the source data). Therefore, in order to adapt an FEC Code for use in the context of the FEC Framework, additional FEC-code specific specification is required. Following the approach of the FEC Building Block, this specification is known as an "FEC Scheme". The FEC Framework will define the requirements that FEC Schemes must meet for use with the framework.

Generally, it is required to add forward error correction to existing applications, for example media streaming applications. In this case, the application protocols must be extended to support this. The FEC framework will describe the requirements that application protocols must meet in order to be used with the FEC framework.

2. Terminology

'FEC' Forward Erasure Correction.

'AL-FEC' Application Layer Forward Erasure Correction

'FEC Framework' The protocol framework which is to be defined by FECFRAME and for which this document provides requirements.

'Source data flow' The packet flow or flows to which FEC protection is to be applied.

'Repair data flow' The packet flow or flows carrying forward error correction data

'Source protocol' A protocol used for the source data flow being protected - e.g. RTP.

'Transport protocol' The protocol used for transport of the source data flow being protected - e.g. UDP, DCCP.

'Control protocol' Application layer protocols used to establish and

modify the source data flow being protected - e.g. RTSP.

'FEC Code' An algorithm for encoding data such that the encoded dats flow is resiliant to data loss or corruption.

'FEC Scheme' A specification which defines the additional protocol aspects required to use a particular FEC code with the FEC framework, or (in the context of RMT), with the RMT FEC Building Block.

'Source Block' the group of source data packets which are to be FEC protected as a single block

'Protection amount' The relative increase in data sent due to the use of FEC.

3. Motivation

One approach to the problem addressed in this document would be to arrange the source packet flows into a sequence of 'objects' and then apply FEC protection using the mechanisms defined by the RMT working group for object transport. This section describes the motivation for following a separate approach, although one that draws heavily on the RMT work.

FEC Schemes defined according to the RMT FEC Building Block [RFC5052] envisage objects with a finite size. Mapping arbitrary flows to this environment one would need to consider the flows as a sequence of such objects (also known as Source Blocks). For each object, the RMT FEC Schemes expect FEC Object Transmission Information to be communicated with the object. In many cases some or all of this

information will be the same for every block. Thus there is some advantage in explicitly introducing the concept of a flow (or bundle of flows) for which some or all of the FEC Object Transmission Information can be the same for every source block. As well as reducing overhead, it is advantageous to be able to inform the receiver that these parameters won't change during the lifetime of the flow or flows.

A second issue is that FEC Schemes in RMT generally also include recommendations for parameter settings, which are based on single-object delivery. Recommendations for protection of packet flows may be different from these for a variety of reasons. There is a need, therefore, for FEC-Scheme specific specification material which is specific to the case of arbitrary packet flows and different from the recommendations for single-object delivery. One of the key aspects of the FEC Framework contemplated here is that it provides a context for such material, in the form of an explicit description of the requirements that FEC Schemes must meet in order to be used with this framework.

A third issue is the question of how source data from a packet flow or flows is formatted into data blocks that an 'object-based' FEC Scheme could process. RMT FEC Schemes expect an object which is just a sequence of bytes. We therefore would need to build such an object out of a sequence of potentially variable-length source packets. There are several ways this could be done and different FEC Schemes may require different approaches. Again, the framework contemplated here provides a context for the definition of these mechanisms through the concept of FEC Schemes which are adapted for use with this framework. The RMT work then envisages that both source packets and repair packets consist of symbols which are extracted from or generated from (respectively) this source block. In the case of FEC protection of arbitrary packet flows it is desirable to support cases

where the source packets are transmitted unchanged, thereby providing backwards compatibility. This is not compatible with in the RMT approach.

As a result of the considerations above, this document describes requirements for an FEC Framework for arbitrary packet flows which is independent of the RMT FEC Building Block, although we draw heavily on the concepts developed there. FEC Schemes defined for use with

this FEC Framework are distinct from FEC Schemes defined for object delivery in the context of the RMT FEC Building Block. However, it is expected that in many cases the task of generalising an RMT FEC Scheme into one which can be used with both the RMT protocols and this FEC Framework will be a simple one.

4. Essential requirements

Req-10: The FEC Framework shall support a wide range of FEC codes, using the abstractions of the FEC Building Block defined in RMT [RFC5052] (including short and long block FEC codes, systematic and non-systematic codes). Specifically, the FEC Framework shall define the requirements that FEC code specifications shall meet in order to be used with the framework, re-using, as far as possible, the FEC code specification approach and requirements from the FEC Building Block and specifying any further requirements that must be met for the FEC Framework.

Req-20: The FEC Framework shall support a wide range of application protocols, using the abstractions of the FEC Building Block [RFC5052]. Specifically, the FEC Framework shall define the requirements that application protocol specifications shall meet in order to be used with the framework, re-using, as far as possible, the Content Delivery Protocol specification approach and requirements from the FEC Building Block and specifying any further requirements that must be met for the FEC Framework.

Req-30: The FEC Framework shall support variable source block sizes, including real-time variation of source block size between blocks of a given source data flow.

Req-35: The FEC Framework shall support variable protection amounts, including dynamic variation of protection amount between blocks within a given source data flow.

Req-40: The FEC Framework shall be independent of the source protocols (provided that source protocol uses one of the supported transport protocols).

Req-50: The FEC Framework shall place minimal requirements on the application protocols.

Req-60: The FEC Framework shall support variable source data flow rates.

Req-70: The FEC Framework shall support variable source data flow packet sizes.

Req-80: The FEC Framework shall provide support of combined protection of multiple source data flows.

Req-90: The FEC Framework shall provide support of multiple transport protocols for the source data protocols (UDP, DCCP, others ?).

Req-100: The FEC Framework shall provide support for definition of backwards-compatible FEC protocols (i.e. where the source packets are not modified in any way).

Req-110: The FEC Framework shall provide support for different source data protocols (RTP, MIKEY, others ?).

Req-120: The FEC Framework shall shall address the security issues, if any, associated with the use of FEC.

5. Non-essential requirements

The FEC Framework should be constructed such that the FEC streaming protocol defined by 3GPP in TS26.346 is a valid protocol according to the FEC Framework.

6. Security Considerations

This document defines requirements for the work of the FECFRAME working group and includes a requirement that the security implications of the use of FEC, if any, should be addressed in that work.

7. References

[RFC5052] Watson, M., Luby, M., and L. Vicisano, "Forward Error Correction (FEC) Building Block", RFC 5052, August 2007.

Author's Address

Mark Watson
Digital Fountain
39141 Civic Center Dr.
Suite 300
Fremont, CA 94538
US

Email: mark@digitalfountain.com

Full Copyright Statement

Intellectual Property

Acknowledgment

Funding for the RFC Editor function is provided by the IETF Administrative Support Activity (IASA).