FECFRAME Working Group
Internet Draft
Intended status: Standards Track
Expires: May 2009

Rajiv Asati
Cisco Systems

November 1, 2008

Methods to convey FEC Framework Configuration Information
draft-ietf-fecframe-config-signaling-01.txt

Status of this Memo

By submitting this Internet-Draft, each author represents that any applicable patent or other IPR claims of which he or she is aware have been or will be disclosed, and any of which he or she becomes aware will be disclosed, in accordance with Section 6 of BCP 79.

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF), its areas, and its working groups. Note that other groups may also distribute working documents as Internet-Drafts.

Internet-Drafts are draft documents valid for a maximum of six months and may be updated, replaced, or obsoleted by other documents at any time. It is inappropriate to use Internet-Drafts as reference material or to cite them other than as "work in progress."

The list of current Internet-Drafts can be accessed at http://www.ietf.org/ietf/1id-abstracts.txt

The list of Internet-Draft Shadow Directories can be accessed at http://www.ietf.org/shadow.html

This Internet-Draft will expire on May 1, 2007.

Copyright Notice

Copyright (C) The IETF Trust (2008).

Abstract

FEC Framework document [FECARCH] defines the FEC Framework Configuration Information necessary for the FEC framework operation. This document describes how to use existing signaling protocols to determine and dynamically communicate the Configuration information between sender(s) and receiver(s).

Conventions used in this document

In examples, "C:" and "S:" indicate lines sent by the client and server respectively.

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].

Table of Contents

1. Introduction

FEC Framework document [FECARCH] defines the FEC Framework Configuration Information that governs the overall FEC framework operation common to any FEC scheme. This information MUST be available at both sender and reciever(s). This document describes how to use various signaling protocols to determine and communicate the Configuration information between sender and receiver(s). The configuration information may be encoded in any compatible format such as SDP [RFC4566], XML etc. A signaling protocol could be

utilized by any FEC scheme and/or any Content Delivery Protocol (CDP).

This document doesn't describe any FEC scheme specifics information (for example, how are source blocks are constructed) or any sender or receiver side operation for a particular FEC scheme (for example, whether the receiver makes use of one or more repair flows that are received) etc. Such FEC scheme specifics should be covered in separate document(s). This document doesn't mandate a particular encoding format for the configuration information either.

<What is CDP>

Content Delivery Protocol (CDP) is defined as a complete (suite of) specification which, through the use of FEC Framework, is able to make use of FEC scheme(s) to provide FEC capabilities. In other words, CDP makes use of common building blocks (including signaling protocol) as defined in the FEC Framework document [FECARCH].

This document is structured such that Section 2 describes the terms used in this document, section 3 describes the FEC Framework configuration information, section 4 describes how to use signaling protocol for the multicast application, section 5 describes the signaling protocol for the unicast application, and section 6 describes security consideration.

Copyright (C) The IETF Trust (2008). This version of this MIB module is part of RFC XXXX; see the RFC itself for full legal notices.

2. Terminology/Abbreviations

This document makes use of the terms/abbreviations defined in the FEC Framework document [FECARCH]. Additionally, it defines

- Media Sender - Node performing the Media encoding and producing the original media flow to the 'FEC Sender'
- Media Receiver - Node performing the Media decoding;
- FEC Sender - Node performing the FEC encoding on the original stream to produce the FEC stream

- FEC Receiver - Node performing the FEC decoding, as needed, and providing the original media flow to the Media receiver.
- Sender - Same as FEC Sender
- Receiver - Same as FEC Receiver
- (Media) Stream - A single media instance i.e. an audio stream or a video stream.

This document deliberately refers to the 'FEC Sender' and 'FEC Receiver' as the 'Sender' and 'Receiver' respectively.

3. FEC Framework Configuration Information

The FEC Framework [FECARCH] defines a minimum set of information that MUST be communicated between the sender and receiver(s) for a proper operation of an FEC scheme. This information is referred to as "FEC Framework Configuration Information". This is the information that the FEC Framework needs in order to apply FEC protection to the transport flows.

A single instance of the FEC Framework provides FEC protection for all packets of a specified set of source packet flows, by means of one or more packet flows consisting of repair packets. As per the FEC Framework document [FECARCH], the FEC Framework Configuation Information includes, for each instance of the FEC Framework:

1. Identification of Source Flow(s)
2. Identification of the repair flow(s)
3. Identification of FEC Scheme
4. Length of Source FEC payload ID
5. FEC Scheme Specific Information (FSSI)

FSSI basically provides an opaque container to encode FEC scheme specific configuration information such as buffer size, decoding wait-time etc. Please refer to the FEC Framework document [FECARCH] for more details.

The usage of signaling protocols described in this document requires that the application layer responsible for the FEC Framework instance i.e. FEC scheme provide the value for each of the configuration information parameter (listed above) encoded as per the chosen encoding format. Failure to receive the complete information, the signaling protocol module must return an error for the OAM purposes and optionally convey to the application layer. Please refer to the figure 1 of the FEC Framework document [FECARCH] for further illustration.

This document does not make any assumption that the 'FEC sender and receiver' functionality and the 'Media Source/Receiver' functionality are implemented on the single device, though it may be the case.

3.1. Encoding Format

The FEC Framework configuration information (listed above in section 3) may be encoded in any format such as SDP, XML etc. as chosen or prefered by a particular FEC Framework instance i.e. FEC Scheme. The selection of such encoding format or syntax is independent of the signaling protocol and beyond the scope of this document.

Whatever encoding format is selected for a particular FEC framework instance, it must be known by the signaling protocol. This is to provide a mean (e.g. a field such as Payload Type) in the signaling protocol message(s) to convey the chosen encoding format for the configuration information so that the Payload i.e. configuration information can be correctly parsed as per the semantics of the chosen encoding format at the receiver. Please note that the encoding format is not a negotiated parameter, but rather a property of a particular FEC Framework instance i.e. FEC scheme and/or its implementation.

Additionally, the encoding format for each FEC Framework configuration parameter must be defined in terms of a sequence of octets that can be embedded within the payload of the signaling protocol message(s). The length of the encoding format MUST either be fixed, or derived by examining the encoded octets themselves. For example, the initial octets may include some kind of length indication.

Each instance of the FEC Framework muse use a single encoding format to describe e.g. encode all of the configuration information associated with that instance. The signaling protocol may not validate the encoded information, though it may validate the syntax or length of the encoded information.

The reader may refer to the SDP elements document [FECSDP], which describes the usage of 'SDP' encoding format as an example encoding format for FEC framework configuration information.

4. Signaling Protocol Usage

FEC Framework [FECARCH] requires certain FEC Framework Configuration Information to be available to both sender and receiver(s). This configuration information is almost always formulated at the sender (or on behalf of a sender),the receiver(s) somehow must get this configuration information. While one may envision a static method to populate the configuration information at both sender and receiver(s), it would require the knowledge of every receiver in advance and that is something not always feasible. Hence, there is a desire to describe using methods i.e. signaling protocol to convey the configuration information between sender and one or more receivers.

It is important to note that there may be either only one receiver needing the FEC Framework configuration information to FEC protect a "unicasted multimedia stream" (such as Video On Demand stream), or one or more receivers needing the FEC Framework configuration information to FEC protect a "multicasted multimedia stream" (such as Broadcast TV or IPTV). While the unicasted stream requires the identification of the receiver at the sender, the multicasted stream doesn't necessarily require the identification of the receiver at the sender.

Such diversity necessitates describing using at least two types of signaling protocols - one to deliver the configuration information to many receivers via multicasting (described in section 4.1), and the other to deliver the configuration information to one and only one receiver via unicasting (described in section 4.2).

Figure 1 below illustrates a sample topology showing the FEC sender and FEC receiver (that may or may not be the Media Sender and Media Receiver respectively) such that FEC_Sender1 is serving FEC_Reciver11,12,13 via the multicast signaling protocol, whereas the

FEC_Sender2 is serving only FEC_Reciever2 via the unicast signaling protocol.

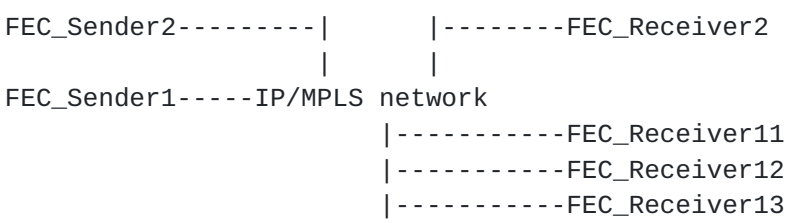

Figure 1 Topology using Sender and Receiver

The rest of the section continues to use the terms 'Sender' and 'Receiver' to refer to the 'FEC Sender' and 'FEC Receiver' respectively.

4.1. Signaling Protocol for Multicasting

This specification describes using Session Announcement Protocol (SAP) version 2 [RFC2974] as the signaling protocol to multicast the configuration information from one sender to many receivers. The apparent advantage is that the server doesn't need to maintain any state for any receiver using SAP.

At the high level, a sender, acting as the SAP announcer, signals the FEC Framework Configuration Information for each FEC Framework instance available at the sender, using the SAP message(s). The configuration information, encoded in a suitable format as per the section 3.1, is carried in the Payload of the SAP message(s). A receiver, acting as the SAP listener, listens on a well known UDP port and at least one well known multicast group IP address. This enables the receiver to receive the SAP message(s) and obtains the FEC Framework Configuration Information for each FEC Framework Instance.

Using the configuration information, the receiver becomes aware of available FEC protection options, and may subscribe to one or more multicast trees to receive the FEC streams using out-of-band multicasting techniques such as PIM [RFC4601]. This, however, is outside the scope of this document.

SAP message is carried over UDP over IP. The destination UDP port must be 9875 and source UDP port may be any available number. The SAP message(s) SHOULD contain an authentication header and MAY be subjected to the cryptography. One cryptography method suggested by this specification is the usage of Group Cryptography as specified in GDOI [RFC3547].

Figure 2 below illustrates the SAP packet format (it is reprinted from the RFC2974) -

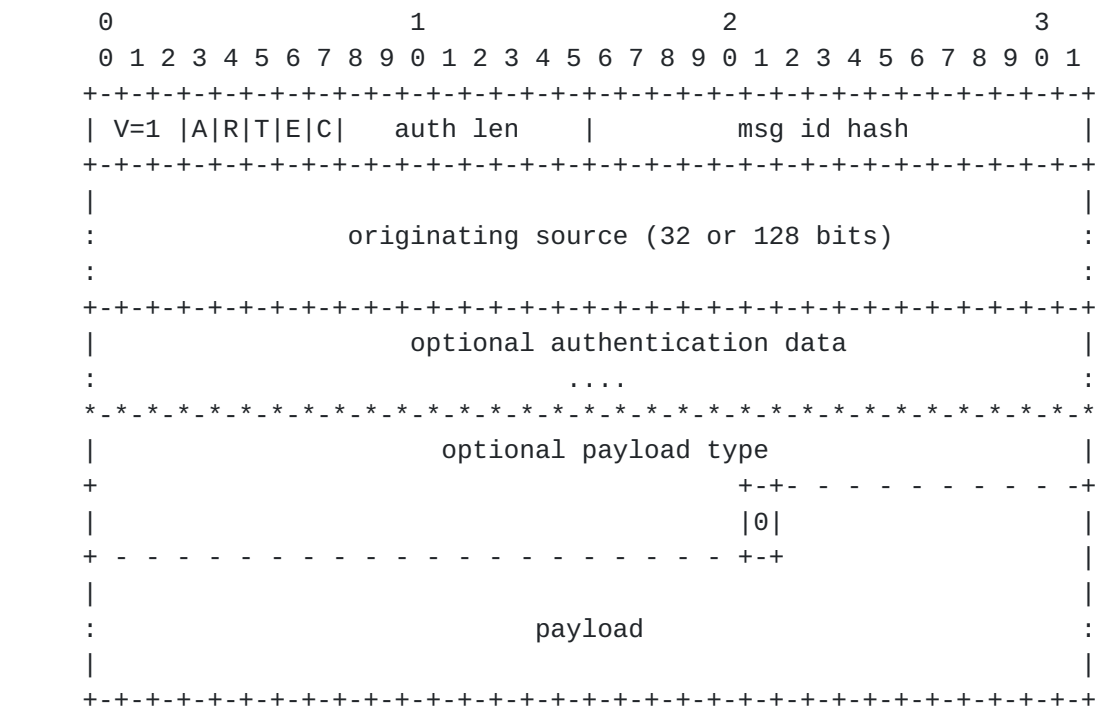

Figure 2 SAP Message format

While the RFC2974 includes explanation for each field, it is worth discussing the 'Payload' and 'Payload Type' field. This field must be used to carry the FEC Framework configuration information. Subsequently, the optional 'Payload Type' field, which is a MIME content type specifier, must describe the encoding format used to encode the Payload. For example, the 'Payload Type' field may be application/sdp if the FEC framework configuration information is encoded in SDP format and carried in the SAP payload. Similarly, it would be application/xml if the FEC framework configuration information was encoded in XML format.

4.1.1. Sender Procedure

The sender signals the FEC framework configuration for each FEC framework instance in a periodic SAP announcement message. The SAP announcement message is sent to a well known multicast IP address and port. The announcement is multicasted with the same scope as the session being announced.

The SAP module at the sender obtains the FEC Framework configuration information per Instance from the 'FEC Framework' module and places that in the SAP payload accordingly. A single SAP (announcement) message may carry the FEC Framework Configuration Information for each FEC Framework Instance. This is the preferred method, though the other method may be to aggregate more than one SAP (announcement) messages in a single UDP datagram as long as the resulting UDP datagram length is less than the IP MTU of the outgoing interface.

The IP packet carrying the SAP message must be sent to destination IP address of either 239.16.33.254 (if IPv4 administrative scope 239.0.0.0-239.255.255.255 is selected) or 224.2.127.254 (if IPv4 global scope 224.0.1.0-238.255.255.255 is selected) or FF0X:0:0:0:0:0:2:7FFE (if IPv6 is selected, where X is the 4-bit scope value) with UDP destination port 9875. The default IP TTL value should be 255, though the implementation should allow to set it to any other value. The IP DSCP field may be set to any value that indicates a desired QoS treatment in the IP network.

The IP packet carrying the SAP message must be sent with source IP address that is reachable by the receiver. The sender may assign the same IP address in the "originating source" field of the SAP message, as the one used in the source IP address of the IP packet.

Furthermore, the FEC Framework Configuration Information must NOT include any of the reserved multicast group IP addresses for the FEC streams (i.e. source or repair flows), though it may use the same IP address as the 'originating source' address to identify the FEC streams (i.e. source or repair flows). Please refer to IANA assignments for multicast addresses.

The sender must periodically send the 'SAP announcement' message. This is required so that the receiver doesn't purge the cached entry(s) from the database and doesn't trigger the deletion of FEC Framework configuration information. While the time interval between repetitions of an announcement can be calculated as per the very sophisticated but complex formula explained in RFC2974, the preferred

and simpler mean is to let the user specify the time interval from the range of 1-200 seconds with suggested default being 60 seconds. The time interval must be chosen to ensure that SAP announcement message is sent out before the corresponding multicast routing entry (S,G) or (*,G) on the router doesn't time out. It is worth noting that the default time-out period for the multicast routing entry is 210 seconds, per the PIM specification [RFC4601], though it may depend on the implementation. The SAP implementation should provide the flexibility to the operator to choose the complex method over the simpler method of determining the SAP announcement time interval. Additionally, the 'time interval' should be signaled within the FEC Framework configuration Information.

The sender may choose to delete the announced FEC framework configuration information by sending a 'SAP deletion' message. This may be used if the sender no longer desires to send any FEC streams. If the sender needs to modify the announced FEC Framework configuration Information for one or more FEC instances, then the sender must send a new announcement message with a different 'Message Identifier Hash' value as per the rules described in section 5 of RFC2974. Such announcement message should be sent immediately (without having to wait for the time-interval) to ensure that the modifications are received by the receiver as soon as possible. The sender must send the SAP deletion message to delete the previous SAP announcement message (i.e. with the previous 'Message Identifier Hash' value).

4.1.2. Receiver Procedure

The receiver must listen on UDP port 9875 for packets arriving with IP destination address of either 239.16.33.254 (if IPv4 administrative scope is selected) or 224.2.127.254 (if IPv4 global scope is selected) or FF0X:0:0:0:0:0:2:7FFE (if IPv6 is selected, where X is the 4-bit scope value).

The receiver, upon receiving a SAP announcement message, creates an entry, if it doesn't already exists, in a local database and passes the FEC Framework configuration information from the SAP Payload field to the 'FEC Framework' module. When the same annoucement (please see section 5 of RFC2974) is received the next time, the timer of the corresponding entry should be reset to the three times the time-interval value that is signaled by the sender or one hour, whichever is greater.

Editor's Note: Unfortunately, SAP doesn't allow the time-interval to be signaled in the SAP header. Hence, the time-interval should be specified in the FEC Framework Configuration Information. For example, the usage of "r=" (repeat time) field in SDP.

The receiver, upon receiving a SAP delete message, must delete the matching SAP entry in its database. This should result in the receiver no longer using the relevant FEC framework configuration information for every instance, and should no longer subscribe to any related FEC streams.

4.2. Signaling Protocol for Unicasting

This document describes leveraging any signaling protocol that is already used by the unicast application, for exchanging the FEC Framework configuration Information between two nodes.

For example, a multimedia (VoD) client may send a request via unicasting for a particular content to the multimedia (VoD) server, which may offer various options such as encodings, bitrates, transport etc. for the content. The client selects the suitable options and answers to the server, paving the way for the content to be unicasted on the chosen transport from server to the client. This offer/answer signaling, described in [RFC3264], is commonly utilized by many application protocols such as SIP, RTSP etc.

The fact that two nodes desiring unicast communication almost always rely on an application to first exchange the application related parameters via the signaling protocol, it is logical to enhance such signaling protocol(s) to (a) convey the desire for the FEC protection and (b) subsequently also exchange FEC parameters i.e. FEC Framework Configuration information. This enables the node acting as the offerer to offer 'FEC Framework Configuration Information' for each of available FEC instances, and the node acting as the answerer conveying the chosen FEC Framework instance(s) to the offerer. The usage of FEC framework instance i.e. FEC scheme is explained the FEC Framework document [FECARCH].

While enhancing anapplication's signaling protocol to exchange FEC parameters is one method (briefly explained above), another method would be to have a unicast based generic protocol that could be used by two nodes independent of the application's signaling protocol. The latter method is under investigation and may be covered by a separate document.

The remainder of this section provides example signaling protocols and explains how they can be used to exchange FEC Framework Configuration information.

4.2.1. SIP

SIP [RFC3261] is an application-level signaling protocol to create, modify, and terminate multimedia sessions with one or more participants. SIP also enables the participants to discover one another and to agree on a characterization of a multimedia session they would like to share. SIP runs on either TCP or UDP or SCTP transport, and uses SDP as the encoding format to describe multmedia session attributes.

SIP already uses offer/answer model with SDP, described in [RFC3264], to exchange the information between two nodes to establish unicast sessions between them. This document extends the usage of this model for exchanging the FEC Framework Configuration Information, explained in section 3. Any SDP specific enhancements to accommodate the FEC Framework are covered in the SDP Elements specification [FECSDP].

4.2.2. RSTP

RTSP [RFC2326] is an application-level signaling protocol for control over the delivery of data with real-time properties. RTSP provides an extensible framework to enable controlled, on-demand delivery of real-time data, such as audio and video. RTSP runs on either TCP or UDP transports.

RTSP already provides an ability to extend the existing method with new parameters. This specification suggests requesting for the FEC protection options by including "FEC Protection Required" in the "Require:" header of SETUP (method) request message. The node receiving such request either responds with "200 OK" message that includes offers i.e. available FEC options (e.g. FEC Framework Configuration Information for each Instnace) or "551 Option not supported" message. A sample of related message exchange is shown below -

```
Node1->Node2:  SETUP < ... > RTSP/1.0
               CSeq: 1
               Transport: <omitted for simplicity>
               Require: FEC Protections Required

Node2->Node1:  RTSP/1.0 200 OK
               CSeq: 1
               Transport: <omitted for simplicity>
```

The requesting node (Node1) may then send either the SETUP message without using the Require: header, if the remote node didn't support the "FEC protection", or a new SETUP message to request the selected FEC protection streams.

4.2.3. DSM-CC

DSM-CC is a predominant suite of protocols including the signaling protocol used for the video control plane in Cable/MSO networks that have offered video services for decades. DSM-CC is standardised in MPEG-2 ISO/IEC 13818-6 (part 6 of the MPEG-2 standard), not within the IETF yet, hence, DSM-CC related enhancements aren't covered in this document. The same is applicable to Session Setup protocol (SSP) and Lightweight Stream Control Protocol (LSCP) that are derived from DSM-CC, as well.

5. Security Considerations

There is no additional security consideration other than what's already covered in RFC2974 for SAP, RFC2326 for RTSP, RFC3261 for SIP etc.

6. IANA Considerations

None.

7. Conclusions

TBD.

8. Acknowledgments

Thanks to Colin Perkins for pointing out the issue with the time-interval for the SAP messages. Additionally, thanks to Mark Watson, Ali Begen and Ulas Kozat for solidifying this proposal.

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

9. References

9.1. Normative References

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

[FECARCH] Watson, M., "Forward Error Correction (FEC) Framework", draft-ietf-fecframe-framework-03 (work in progress), October 2008.

[FECSDP] Begen, A., "SDP Elements for FEC Framework", draft-ietf-fecframe-sdp-elements-01 (work in progress), July 14 2008.

9.2. Informative References

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

[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.

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

[RFC3261] Handley, M., Schulzrinne, H., Schooler, E. and J. Rosenberg, "SIP: Session Initiation Protocol", RFC 3261, June 2002.

[RFC4601] Fenner, etc., "Protocol Independent Multicast - Sparse Mode (PIM-SM) : Protocol Specification", RFC 4601, August 2006.

[RFC3547] Baugher, etc., "The Group Domain of Interpretation", RFC 3547, July 2003.

Author's Addresses

Rajiv Asati
Cisco Systems,
7025-6 Kit Creek Rd, RTP, NC, 27709-4987
Email: rajiva@cisco.com

Intellectual Property Statement

The IETF takes no position regarding the validity or scope of any Intellectual Property Rights or other rights that might be claimed to pertain to the implementation or use of the technology described in this document or the extent to which any license under such rights might or might not be available; nor does it represent that it has made any independent effort to identify any such rights. Information on the procedures with respect to rights in RFC documents can be found in BCP 78 and BCP 79.

Copies of IPR disclosures made to the IETF Secretariat and any assurances of licenses to be made available, or the result of an attempt made to obtain a general license or permission for the use of such proprietary rights by implementers or users of this specification can be obtained from the IETF on-line IPR repository at http://www.ietf.org/ipr.

The IETF invites any interested party to bring to its attention any copyrights, patents or patent applications, or other proprietary rights that may cover technology that may be required to implement this standard. Please address the information to the IETF at ietf-ipr@ietf.org.

Disclaimer of Validity

This document and the information contained herein are provided on an "AS IS" basis and THE CONTRIBUTOR, THE ORGANIZATION HE/SHE REPRESENTS OR IS SPONSORED BY (IF ANY), THE INTERNET SOCIETY, THE IETF TRUST AND THE INTERNET ENGINEERING TASK FORCE DISCLAIM ALL WARRANTIES, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO ANY WARRANTY THAT THE USE OF THE INFORMATION HEREIN WILL NOT INFRINGE ANY RIGHTS OR ANY IMPLIED WARRANTIES OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE.

Copyright Statement

Copyright (C) The IETF Trust (2008).

This document is subject to the rights, licenses and restrictions contained in BCP 78, and except as set forth therein, the authors retain all their rights.

Acknowledgment

Funding for the RFC Editor function is currently provided by the Internet Society.