INTERNET-DRAFT
November 20, 2006
Category: Experimental
Expiration: April 23, 2007

Do J. Byun
John Border
Roderick Ragland

Header Compression over Unidirectional Lightweight Encryption (ULE)
draft-byun-ipdvb-ule-header-comp-01.txt

Status of This Memo

Copyright Notice

Intellectual Property Right

Abstract

Multi-Protocol Encapsulation (MPE) is widely deployed in DVB-S and DVB-S2 networks [DVB-S2]. Replacing MPE with Unidirectional Lightweight Encryption (ULE) has been proposed to gain flexibility and reduce overhead. This paper introduces a signaling method for sending header-compressed unicast packets over satellite networks using ULE, taking advantage of ULE's increased flexibility.

Ed. Note: This is a quick first draft to get the discussion going.

Table of Contents

1. Introduction

Header compression is a mechanism that compresses the header fields that do not change or change in predictable ways. RFC 3095 defines "Robust Header Compression (ROHC)" as a standard for compressing RTP/UDP/IP, UDP/IP and ESP/IP headers. [RFC3095]. There could be other proprietary compression schemes besides ROHC.

To support header compression, the link-layer has to be flexible enough to indicate whether the payload is header-compressed or not. Such indication has been difficult with MPE due to its limited flexibility in its header format.

Unidirectional Lightweight Encryption has been proposed to overcome this shortcoming of MPE and there had been numerous proposals to standardize one as the link-layer protocol of DVB-S2 [GSE]. This document describes how ULE is used to support header compression over ISO MPEG-2 transport streams [ISO-MPEG2, RFC4259] for peer-to-peer traffic.

2. Terminology

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 RFC 2119.

DVB
: Digital Video Broadcast. A framework and set of associated standards published by the European Telecommunications Standards Institute (ETSI) for the transmission of video, audio, and data using the ISO MPEG-2 Standard [ISO-MPEG2].

MAC
: Medium Access Control [IEEE-802.3]. A link-layer protocol defined by the IEEE 802.3 standard (or by Ethernet v2 [DIX]).

MPE
Multiprotocol Encapsulation [ETSI-DAT, ATSC-DAT, ATSC-DATG]. A scheme that encapsulates PDUs, forming a DSM-CC Table Section. Each Section is sent in a series of TS Packets using a single TS Logical Channel.

MPEG-2
A set of standards specified by the Motion Picture Experts Group (MPEG) and standardized by the International Standards Organisation (ISO/IEC 13818-1) [ISO-MPEG2], and ITU-T (in H.222 [ITU-H222]).

PSI
Program Specific Information [ISO-MPEG2]. Tables used to convey information about the service carried in a TS Multiplex. The information is carried in one of four specifically identified Table Sections defined by MPEG-2 [ISO-MPEG2]. See also SI Table.

PDU
Protocol Data Unit. Examples of a PDU include Ethernet frames, IPv4 or IPv6 datagrams, and other network packets.

Receiver
Equipment that processes the signal from a TS Multiplex and performs filtering and forwarding of encapsulated PDUs to the network-layer service (or bridging module when operating at the link layer).

SI Table
Service Information Table [ISO-MPEG2]. In this document, this term describes a table that is defined by another standards body to convey information about the services carried in a TS Multiplex. A Table may consist of one or more Table Sections; however, all sections of a particular SI Table must be carried over a single TS Logical Channel [ISO-MPEG2].

SNDU
SubNetwork Data Unit. An encapsulated PDU sent as an MPEG-2 Payload Unit.

TS
Transport stream (TS) is a format specified in MPEG-2 Part 1, Systems (ISO/IEC standard 13818-1). Its design goal is to allow multiplexing of digital video and audio and to synchronize the output. Transport stream offers features for error correction for transportation over unreliable media, and is used in broadcast applications such as DVB and ATSC.

ULE Stream
An MPEG-2 TS Logical Channel that carries only ULE encapsulated

PDUs. ULE Streams may be identified by definition of a
stream_type in SI/PSI [ISO-MPEG2].

3. Signaling Method

Header compression shall be indicated by the EtherType field of the ULE header. When this this field is set to header compression type, the payload is header-compressed. The actual type of header compression is determined during the context establishment between the two peers (one compressor and one decompressor). Therefore, the method by which the payload is compressed and decompressed is part of the compression context. Moreover, compression context control messages can also be header-compressed but their context will be different from the one for the actual user data.

```
Decompressor                                                Compressor

   |                                                              |
   |  <----- (EtherType=IPv4) Uncompressed Payload        ------  |
   |  ------ (EtherType=IPv4) Compression Request         ----->  |
   |  <----- (EtherType=IPv4) Compression Ack             ------  |
   |  <----- (EtherType=Comp) Compressed Header Payload   ------  |
   |  <----- (EtherType=Comp) Compressed Header Payload   ------  |
Something Bad Happens
   |  ------ (EtherType=IPv4) Compression Error           ----->  |
   |  <----- (EtherType=IPv4) Uncompressed Header Payload ------  |
   |  <----- (EtherType=IPv4) Uncompressed Header Payload ------  |
```

Figure 1: Header compression example

Figure 1 illustrates an example where control messages (that signal and synchronize peers to compress/decompress) are not header-compressed but the user data messages are. When EtherType is set to 'Comp' whose hex value is TBD, the ULE payload contains header-compressed user data messages.

The EtherType of TBD will be a newly registered IANA EtherType number that indicates a compression algorithm that is agreed by both the sender and receiver. In other words, it could be any proprietary header compression algorithm as long as the receiver knows how to decompress it. EtherType of 0x876B (TCP/IP Compression [RFC1144]) was intentionally not used because it is currently defined to imply a specific header-compression algorithm.

3.1. SNDU Format

This section describes the SNDU format of the MPEG-2 PDU with ULE where headers for PDU are compressed.

```
< ---------------------------- SNDU ---------------------------- >
+-+-------------------------------------------------------+--------+
|D| Length | Type | Dest Address* |           PDU          | CRC-32 |
+-+-------------------------------------------------------+--------+
```

Figure 2: SNDU Encapsulation (* optional Destination Address)

The definition of all of the fields in Figure 2 stays the same as the definition in [RFC4326]. The 16-bit type field will have a new EtherType to indicate the PDU is header-compressed with an algorithm that both sender and received agreed on. The hex value for this type is TBD. Note that the new header-compressed PDU EtherType does not indicate a specific header-compression algorithm. It is the sender and receiver's responsibility to make sure the algorithm is synchronized.

Ed. Note: This is one of the main points we want to discuss on the mailing list.

3.2. Header Compression Algorithm

In order to use the proposed EtherType to indicate the PDU is header-compressed, both the sender and receiver have to agree with the compression algorithm. This is not an issue because such synchronization is always required in peer-to-peer header compression anyway.

```
  Incapable                                                Incompatible
 Decompressor                                               Compressor

    |                                                              |
    |  <----- (EtherType=IPv4) Uncompressed Header Payload ------  |
    |                                                     Waiting for
    |                                                     Comp Request
    |  <----- (EtherType=IPv4) Uncompressed Header Payload ------  |
    |                                                     Waiting for
    |                                                     Comp Request
    |  <----- (EtherType=IPv4) Uncompressed Header Payload ------  |
    |                                                     Waiting for
    |                                                     Comp Request
    |                                                              |
```

Figure 3: Incapable decompressor

Figure 3 illustrates a scenario where the decompressor (receiver) is not capable of decompressing the packets that the compressor (sender) sent. The decompressor does not send any compression request to the compressor and the compressor continues to send uncompressed headers to the decompressor with non-header-compression EtherType.

```
 Incompatible                                           Incompatible
 Decompressor                                            Compressor

   |                                                          |
   |  <----- (EtherType=IPv4) Uncompressed Header Payload ------  |
   |  ------ (EtherType=IPv4) Compression Request         ----->  |
   |                                                        detects
   |                                                     incompatible
   |                                                     decompressor
   |  <----- (EtherType=IPv4) Uncompressed Header Payload ------  |
   |  <----- (EtherType=IPv4) Uncompressed Header Payload ------  |
   |  <----- (EtherType=IPv4) Uncompressed Header Payload ------  |
   |                                                          |
```

Figure 4: Incompatible compressor and decompressor

Figure 4 illustrates a scenario where the compressor is not compatible with the decompressor and therefore it continues to send uncompressed headers to the decompressor with non-header-compression EtherType.

Specifics of a header compression algorithm may differ widely. They include the way header-compression is initiated and synchronized. For example, compression request messages can be initiated by the compressor instead of decompressor. Regardless of the algorithm, the header-compression indication method proposed here signals the decompressor that the payload is header-compressed with the algorithm that it agreed to use.

3.3. Multicast and Broadcast Traffic

Since out of band synchronization is also assumed for multicast and broadcast, the proposed header-compressed PDU signaling scheme supports multicast and broadcast as well.

4. Summary

This document defines a mechanism to signal the receiver that the payload is header-compressed using ULE as the link layer. The mechansim is compatible with any peer-to-peer header compression algorithm. It uses a newly proposed EtherType to indicate that the payload is header-compressed. The EtherType has the value of TBD which is not tied to a specific header compression algorithm.

The proposed method to indicate header-compressed payload is not for multicast and broadcast as there is no gaurantee that the receivers are compatible decompressors.

5. Acknowledgements

TBD

6. Security Considerations

The proposed header compression signaling method does not introduce any additional security concerns.

7. IANA Considerations

A new EtherType number will be proposed to the IANA EtherType registry. This number will be used to indicate that the ULE PDU is header-compressed as described in this document.

8. References:

8.1. Normative References

[ISO-MPEG2]   IS 13818-1, "Information technology -- Generic coding of moving pictures and associated audio information -- Part 1: Systems", International Standards Organization (ISO), 2000.

[RFC1144]   Jacobson, V., "Compressing TCP/IP Headers for Low-Speed Serial Links", 1990.

[RFC3095]   Bormann, C., Burmeister, C., Degermark, M., et al, "Robust Header Compression (ROHC): Framework and four profiles: RTP, UDP, ESP, and uncompressed", RFC 3095, 2001.

[RFC4326]   Fairhurst, G., Collini-Nocker, B., "Unidirectional Lightweight Encapsulation (ULE) for Transmission of IP Datagrams over an MPEG-2 Transport Stream (TS)", RFC 4326, 2005.

[DVB-S2]   Digital Video Broadcasting (DVB); Second generation framing structure, channel coding and modulation systems for Broadcasting, Interactive Services, News Gathering and other broadband satellite applications, ETSI EN 302 307 V1.1.1, 2005.

8.2. Informative References

[GSE]   Fairhurst, G., "A Network-Layer Interface To The Second Generation Standard For DVB Over Satellite", Work in Progress, September 2006.

[DIX] Digital Equipment Corp, Intel Corp, Xerox Corp, "Ethernet Local Area Network Specification" Version 2.0, November 1982.

[ITU-H222] H.222.0, "Information technology - Generic coding of moving pictures and associated audio information: Systems", International Telecommunication Union, (ITU-T), 1995.

Authors' Addresses

Do J. Byun
Hughes Network Systems
11717 Exploration Lane
Germantown, MD, 20876
USA

EMail: dbyun@hns.com

John Border
Hughes Network Systems
11717 Exploration Lane
Germantown, MD, 20876
USA

EMail: border@hns.com

Roderick Ragland
Hughes Network Systems
11717 Exploration Lane
Germantown, MD, 20876
USA

EMail: rragland@hns.com

Comments are solicited and should be addressed to the authors.