Additional UltraGrid Packet Formats for LDGM and Encryption (draft)

Petr Holub, Martin Pulec

February 18, 2014

This documents is related to and complements *CESNET Technical Report 1/2012 4K Video and Audio Packet Format for UltraGrid*[1]. In addition to audio and video formats, this document defines UltraGrid formats for LDGM, encryption and its combinations.

1 LDGM

LDGM payloads and headers are encapsulated directly in RTP packet. RTP payload type is set to 22 for video and 23 for audio.

1.1 Payload header

Figure 1 shows structure of LDGM packet header. Content of first 3 words is exactly the same as with the regular audio and video packets, for its description please refer to [1]. Remaining fields hold properties of LDGM and have the following semantics:

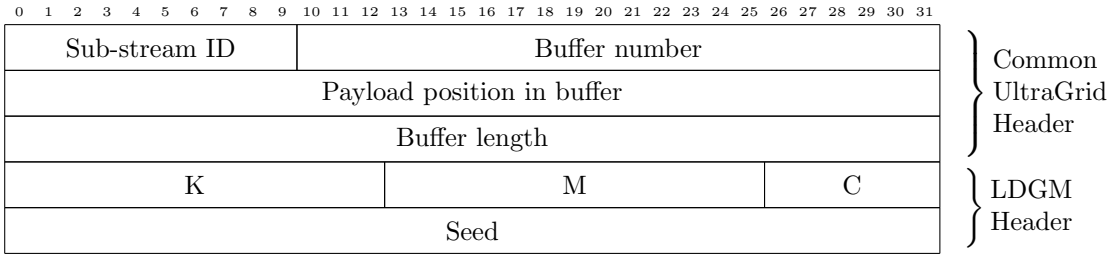

Figure 1: LDGM payload header

k: 13 bits
width of LDGM parity matrix

m: 13 bits
height of LDGM parity matrix

c: 6 bits
number of ones per row of matrix

seed: 32 bits
specifies a seed of PRNG algorithm as defined in [2]. This generator will then be used for parity matrix generation on both ends.

1.2 Payload body

Body consists of split LDGM encoded buffer. Additionally, first packet contains some additional headers as defined in figure 2.

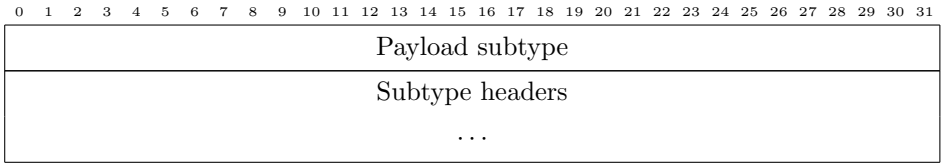

Figure 2: Additional LDGM header

Payload subtype: 32 bits
contains RTP payload type which would be sent with original data if it weren't accompanied by LDGM.

Subtype headers: variable
contains subtype headers (audio and video according to [1], AES as defined in section 2 etc.)

2 Encryption

Encrypted audio and video packets are transmitted inside a RTP packet. Payload type field of RTP is 24 for video and 25 for audio data.

Actual RTP payload headers are very similar to original audio and video payload headers. Figure 3 shows AES video header. Similarly formed is also audio header. The original video header is kept as it was originally defined. After the video header, one more word was added, that describes encryption.

Crypto type: 8 bits
contains type of encryption used, currently defined is:
0 - none
1 - AES128 counter mode

Authentication type: 8 bits
tells which message integrity algorithm was used:
0 - none

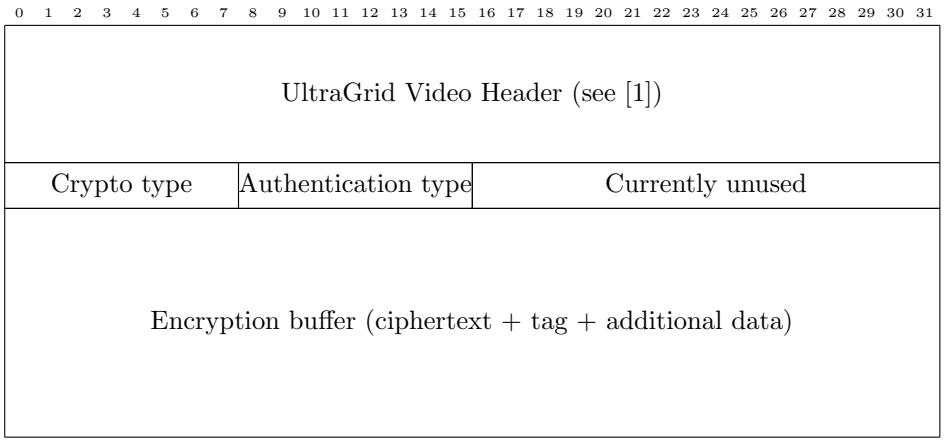

Figure 3: AES video payload header

Buffer length and **position in buffer** in the video header refers to encryption buffer not plaintext video or audio buffer. Encrypted/authenticated data are considered opaque. Crypto module is obliged to provide following pieces of information from the buffer:

- whether or not is the tag sent with data valid
- plaintext data and its length

Encryption buffer must contain all other required data which are required to be passed along with the ciphertext, for instance IV, counter value and the authentication tag.

Authentication mechanism should be able to authenticate also payload headers sent along with data.

3 LDGM and Encryption

Combination of LDGM and encryption on application layer can be realized in two ways. One possibility is to form LDGM packets and encrypt them individually. The second way is, in contrary, to encrypt whole frame buffer, generate parity data and packetize the resulting LDGM buffer. Schematically:

- LDGM then encrypt
    1. generate parity
    2. packetize
    3. encrypt
- encrypt then LDGM

1. encrypt
2. generate LDGM
3. packetize

Both approaches are roughly equivalent in case we either reconstruct whole package or drop it at all. Even when we are not able to reconstruct whole frame, parts can be decompressed in both cases.

From the perspective of performance is more advantageous the second approach, because we are not encrypting parity data. Also we are sending with every datagram only LDGM header.

On the other hand, encoding whole buffer forces also encryption to be on an application layer which differs from handling not-FEC-protected buffers.

LDGM protected data in combination with encryption are transmitted in UltraGrid in the following fashion:

1. whole frame buffer is LDGM encoded
2. LDGM buffer is packetized, individual packets are encrypted, encryption and LDGM headers are prepended

References

[1] Petr Holub, Miloš Liška, and Martin Pulec. *4K Video and Audio Packet Format for UltraGrid*. Tech. rep. 2012. URL: `http://www.sitola.cz/files/4K-packet-format.pdf` (visited on 07/02/2013).

[2] V. Roca, C. Neumann, and D. Furodet. *Low Density Parity Check (LDPC) Staircase and Triangle Forward Error Correction (FEC) Schemes*. RFC 5170. Internet Engineering Task Force, July 2008. URL: `http://tools.ietf.org/html/rfc5170` (visited on 07/02/2013).