Network Working Group
Internet-Draft
Intended status: Standards Track
Expires: June 6, 2020

S. Zhao
S. Wenger
Tencent

December 04, 2019

RTP Payload Format for Essential Video Coding (EVC)
draft-zhao-avtcore-rtp-evc-00

Abstract

This memo describes an RTP payload format for the video coding standard ISO/IEC International Standard 23094-1, also known as Essential Video Coding (EVC) and developed by ISO/IEC JTC1/SC29/WG11. The RTP payload format allows for packetization of one or more Network Abstraction Layer (NAL) units in each RTP packet payload as well as fragmentation of a NAL unit into multiple RTP packets. The payload format has wide applicability in videoconferencing, Internet video streaming, and high-bitrate entertainment-quality video, among others.

Status of This Memo

Copyright Notice

Table of Contents

1. Introduction

The EVC specification, which will be formally designatited (once approved) as ISO/IEC International Standard 23094-1 [EVC], is planned for ratification in early 2020. A draft that’s currently in the approval process of ISO/IEC can be found as [EVC] (Essential Video Coding). One goal of MPEG is to keep [EVC]’s baseline essentially royalty free by agreement among the key contributors, whereas more advanced profiles follow a reasonable and non-dscriminatory licensing policy. Both baseline and higher profiles of [EVC] are reported to provide coding efficiency gains over H.265 and H.264 under certain configurations.

This memo describes an RTP payload format for [EVC]. It shares its basic design with the NAL unit-based RTP payload formats of [VVC], [RFC7798], [RFC6184] and [RFC6190]. With respect to design philosophy, security, congestion control, and overall implementation complexity, it has similar properties to those earlier payload format specifications. This is a conscious choice, as at least RFC 6184 is widely deployed and generally known in the relevant implementer communities. Certain mechanisms known from [RFC6190] were incorporated as EVC supports temporal scalability. [EVC] does not offer higher forms of scalability.

1.1. Overview of the EVC Codec

EVC, H.265 and H.266 share a similar hybrid video codec design. In this memo, we provide a very brief overview of those features of EVC that are, in some form, addressed by the payload format specified herein. Implementers have to read, understand, and apply the ISO/IEC specifications pertaining to EVC to arrive at interoperable, well-performing implementations. The EVC standard has a baseline profile and on top of that, a main profile, the latter including more advanced features. A "toolset" syntax element allows encoders to mark a bitstream as to what of the many independent coding tools are exercised in the bitstream, in a spirit similar to the general_constraint_flags of H.266.

Conceptually, All [EVC], HEVC and [VVC] include a Video Coding Layer (VCL), which is often used to refer to the coding-tool features, and a Network Abstraction Layer (NAL), which is often used to refer to the systems and transport interface aspects of the codecs.

1.1.1. Coding-Tool Features (informative)

Coding blocks and transform structure

[EVC] uses a traditional quad-tree coding structure, which divides the encoded image into blocks of up to 128x128 luma samples, which can be recursively divided into smaller blocks. The main profile adds two advanced coding structure tools: Binary Ternary Tree (BTT) that allows non-square coding units and segmentation that changes the processing order of the segmentation unit from traditional left-scanning order processing to right-scanning order processing Unit Coding Order (SUCO). In the main profile, the picture can be divided into rectangular tiles, and these tiles can be independently encoded and/or decoded in parallel.

When predicting a data block using intra prediction or inter prediction, the remaining data is usually added to the prediction block. The residual data is added to the prediction block. The resildual data is obtained by applying an inverse quantization process and an inverse transform. [EVC] includes integer discrete cosine transform (DCT2) and scalar quantization. For the main profile, Improved Quantization and Transform (IQT) uses a different mapping/clipping function for quantization. An inverse zig-zag scanning order is used for coefficient coding. Advanced Coefficient Coding (ADCC) in the main profile can code coefficient values more efficiently, for example, indicated by the last non-zero coefficient. In main profile, Adaptive Transformation Selection (ATS) is also available and can be applied to integer versions of DST7 or DCT8, and not just DCT2.

Entropy coding

[EVC] uses a similar binary arithmetic coding mechanism as H.264. The mechanism includes a binarization step and a probability update defined by a lookup table. In the main profile, the derivation process of syntax elements based on adjacent blocks makes the context modeling and initialization process more efficient.

In-loop filtering

The baseline profile of [EVC] uses the deblocking filter defined in H.263 Annex J. In the main profile, compared to the deblocking filter in the baseline profile, an Advanced Deblocking Filter (ADDB) can be used, which can further reduce artifacts. The main profile also defines two additional in-loop filters that can be used to improve the quality of decoded pictures before output and/or for inter prediction. A Walsh-Hadamard Transform Domain Filter (HTDF) is applied to the luma samples before deblocking, and the scanning process is used to determine 4 adjacent samples for filtering. An adaptive Loop Filter (ALF) allows to send signals of up to 25 different filters for the luma components, and the best filter can be selected through the classification process for each 4x4 block. The filter parameters of the ALF filter are signaled in the Adaptation Parameter Set (APS).

Inter-prediction

The basis of [EVC] inter prediction is motion compensation using interpolation filters with a quarter sample resolution. In baseline profile, a motion vector signal is transmitted using one of three spatially neighboring motion vectors and a temporally collocated motion vector as a predictor. The motion vector difference may be signaled relative to the selected predictor, but for the case where no motion vector difference is signaled and there is no remaining data in the block, there is a specific mode called a skip mode. The main profile includes six additional tools to provide improved inter prediction. With advanced Motion Interpolation and Signaling (AMIS), adjacent blocks can be conceptually merged to indicate that they use the same motion, but more advanced schemes can also be used to create predictions from the basic model list of candidate predictors. The Merge with Motion Vector Difference (MMVD) tool uses a process similar to the concept of merging neighboring blocks, but also allows the use of expressions that include a starting point, motion amplitude, and direction of motion to send a motion vector signal.

Using Advanced Motion Vector Prediction (AMVP), candidate motion vector predictions for the block can be derived from its neighboring blocks in the same picture and collocated blocks in the reference picture. The Adaptive Motion Vector Resolution (AMVR) tool provides a way to reduce the accuracy of a motion vector from a quarter sample to half sample, full sample, double sample, or quad sample, which provides the efficiency advantage, such as when sending large motion vector differences. The main profile also includes the Decoder-side Motion Vector Refinement (DMVR), which uses a bilateral template matching process to refine the motion vectors in a bidirectional fashion.

Intra prediction and intra-coding

Intra prediction in [EVC] is performed on adjacent samples of coding units in a partitioned structure. For the baseline profile, all coding units are square, and there are five different prediction modes: DC (mean value of the neighborhood), horizontal, vertical, and two different diagonal directions. In the main profile, intra prediction can be applied to any rectangular coding unit, and there are 28 additional direction modes available in the so-called Enhanced Intra Prediction Directions (EIPD). In the main profile, an encoder can also use Intra Block Copy (IBC), where a previously decoded sample blocks of the same picture is used as a predictor. A displacement vector in integer sample precision is signaled to indicate where the prediction block in the current picture is used for this mode.

Decoded picture buffer management

In the previous technology, decoded pictures can be stored in a decoded picture buffer (Decoded Picture Buffer, DPB) for predicting pictures that follow them in decoding order. In the baseline profile, the management of the DPB (i.e. the process of adding and deleting reference pictures) is controlled by the information in the SPS. For the main profile, if an Reference Picture List (RPL) scheme is used, DPB management can be controlled by information that is signaled at the picture level.

1.1.2. Systems and Transport Interfaces

[EVC] inherited the basic systems and transport interfaces designs from H.264 and H.265. These include the NAL-unit-based syntax structure, the hierarchical syntax and data unit structure and the Supplemental Enhancement Information (SEI) message mechanism. The hierarchical syntax and data unit structure consists of a sequence-level parameter set (SPS), two picture-level parameter sets (PPS and APS, each of which can apply to oen or more pictures), slice-level header parameters, and lower-level parameters.

Below described are a number of key components that influenced the Network Abstraction Layer design of EVC as well as this memo.

Sequence parameter set

The Sequence Parameter Set (SPS) contains syntax elements pertaining to a coded video sequence (CVS), which is a group of pictures, starting with a random access point, and followed by pictures that may depend on each other and the random access point picture. In MPGEG-2, the equivalent of a CVS was a Group of Pictures (GOP), which normally started with an I frame and was followed by P and B frames. While more complex in its options of random access points, EVC retains this basic concept. In many TV-like applications, a CVS contains a few hundred milliseconds to a few seconds of video. In video conferencing (without switching MCUs involved), a CVS can be as long in duration as the whole session.

Picture and Adaptation parameter set

The Picture Parameter Set and the Adaptation Parameter Set (PPS and APS, respectively) carry information pertaining to a single picture. The PPS contains information that is likely to stay constant from picture to picture-at least for pictures for a certain type-whereas the APS contains information, such as adaptive loop filter coefficients, that are likely to change from picture to picture.

Profile, level and toolsets

Profiles and levels follow the same design considerations ask known form H.264, H.265, and in fact video codecs as old as MPEG-1 visual. A profile defines a set of tools (not to confuse with the "toolset" discussed below) that a decoder compliant with this profile has to support. In [EVC], profiles are defined in Annex A. Formally, they are defined as a set of constraints that a bitstream needs to conform to. In [EVC], the baseline profile is much more severely constraint than main profile, reducing implementation complexity. Levels relate to bitstream complexity in dimensions such as maximum sample decoding rate, maximum picture size, etc parameters that are directly related to computational complexity.

Profiles and levels are signaled in the highest parameter set available, the SPS.

[EVC] contains another mechanism related to the use of coding tools, known as the toolset syntax element. This syntax element, also located in the SPS, is a bitmask that allows encoders to indicate which coding tools they are using, within the menu of profiles offered by the profile that is also signaled. No decoder conformance point is associated with the toolset, but a bitstream that were using a coding tool that is indicated as not used in the toolset syntax element would obviously be non-compliant. While MPEG specifically rules out the use of the toolset syntax element as a conformance point, walled garden implementations could do so without incurring the interoperability problems MPEG fears, and create bitstreams and decoders that do not support one or more given tools. That, in turn, may be useful to mitigate certain patent related risks.

Bitstream and elementary stream

Above the Coded Video Sequence (CVS), [EVC] defines a video bitstream that can be used in the MPEG systems context as an elementary stream. For the purpose of this memo, this is not relevant.

Random access support

At this point, the authors believe [EVC] supports only clean random access. WG input is solicited.

Temporal scalability support

[EVC] includes support for temporal scalability through the generalized reference picture selection approach known since H.264/SVC. Up to six temporal layers are supported. The temporal layer is signaled in the NAL unit header (which co-serves as athe payload header in this memo), in the nuh_temporal_id field.

Reference picture management

TBD

SEI Message

[EVC] inherits many of H.265’s SEI Messages, occasionally with changes in syntax and/or semantics making them applicable to EVC.

1.1.3. Parallel Processing Support (informative)

Placeholder

1.1.4. NAL Unit Header

EVC maintains the NAL unit concept of H.265 with different parameter options. EVC also uses a two-byte NAL unit header, as shown in Figure 1. The payload of a NAL unit refers to the NAL unit excluding the NAL unit header.

```
+---------------+---------------+
|0|1|2|3|4|5|6|7|0|1|2|3|4|5|6|7|
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|F|   Type    | TID | Reserve |E|
+-------------+-----------------+
```

The Structure of the EVC NAL Unit Header

Figure 1

The semantics of the fields in the NAL unit header are as specified in [EVC] and described briefly below for convenience. In addition to the name and size of each field, the corresponding syntax element name in [EVC] is also provided.

F: 1 bit

forbidden_zero_bit. Required to be zero in [EVC]. Note that the inclusion of this bit in the NAL unit header was to enable transport of EVC video over MPEG-2 transport systems (avoidance of start code emulations) [MPEG2S]. In the context of this memo,the value 1 may be used to indicate a syntax violation, e.g., for a

NAL unit resulted from aggregating a number of fragmented units of a NAL unit but missing the last fragment, as described in Section xxx. (section # placeholder)

Type: 6 bits

nal_unit_type_plus1. This field specifies the NAL unit type as defined in Table 7-1 of [EVC]. If the most significant bit of this field of a NAL unit is equal to 0 (i.e., the value of this field is less than 32), the NAL unit is a VCL NAL unit. Otherwise, the NAL unit is a non-VCL NAL unit. For a reference of all currently defined NAL unit types and their semantics, please refer to Section 7.3.1.2 in [EVC].

TID: 3 bits

nuh_temporal_id. This field specifies the temporal identifier of the NAL unit plus 1. The value of TemporalId is equal to TID minus 1. A TID value of 0 is illegal to ensure that there is at least one bit in the NAL unit header equal to 1, so to enable independent considerations of start code emulations in the NAL unit header and in the NAL unit payload data.

Reserve: 5 bits

nuh_reserved_zero_5bits. This field shall be equal to the version of the [EVC] specification. Values of nuh_reserved_zero_5bits greater than 0 are reserved for future use by ISO/IEC. Decoders conforming to a profile specified in [EVC] Annex A shall ignore (i.e., remove from the bitstream and discard) all NAL units with values of nuh_reserved_zero_5bits greater than 0.

E: 1 bit

nuh_extension_flag. This field shall be equal the version of the [EVC] specification. Value of nuh_extesion_flag equal to 1 is reserved for future use by ISO/IEC. Decoders conforming to a profile specified in Annex A shall ignore (i.e., remove from the bitstream and discard) all NAL units with values of nuh_extension_flag equal to 1.

1.2. Overview of the Payload Format

This payload format defines the following processes required for transport of [EVC] coded data over RTP [RFC3550]:

- Usage of RTP header with this payload format

- Packetization of EVC coded NAL units into RTP packets using three types of payload structures: a single NAL unit packet, aggregation packet, and fragment unit

- Transmission of EVC NAL units of the same bitstream within a single RTP stream.

- Media type parameters to be used with the Session Description Protocol (SDP) [RFC4566]

2. Conventions

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 BCP 14 [RFC2119]. In this document, the above key words will convey that interpretation only when in ALL CAPS. Lowercase uses of these words are not to be interpreted as carrying the significance described in [RFC2119]. This specification uses the notion of setting and clearing a bit when bit fields are handled. Setting a bit is the same as assigning that bit the value of 1 (On). Clearing a bit is the same as assigning that bit the value of 0 (Off).

3. Definitions and Abbreviations

3.1. Definitions

This document uses the terms and definitions of EVC. Section 3.1.1 lists relevant definitions from EVC for convenience. Section 3.1.2

3.1.1. Definitions from the EVC Specification

PlaceHolder

3.1.2. Definitions Specific to This Memo

PlaceHolder

4. RTP Payload Format

4.1. RTP Header Usage

The format of the RTP header is specified in [RFC3550] (reprinted as Figure 2 for convenience). This payload format uses the fields of the header in a manner consistent with that specification.

The RTP payload (and the settings for some RTP header bits) for aggregation packets and fragmentation units are specified in Section 4.3.2 and Section 4.3.3, respectively.

```
  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
 +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
 |V=2|P|X|  CC   |M|     PT      |       sequence number         |
 +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
 |                           timestamp                           |
 +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
 |           synchronization source (SSRC) identifier            |
 +=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+
 |            contributing source (CSRC) identifiers             |
 |                             ....                              |
 +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

RTP Header According to {{RFC3550}}

Figure 2

The RTP header information to be set according to this RTP payload format is set as follows:

Marker bit (M): 1 bit

Set for the last packet of the access unit, carried in the current RTP stream. This is in line with the normal use of the M bit in video formats to allow an efficient playout buffer handling.

> Informative note: The content of a NAL unit does not tell whether or not the NAL unit is the last NAL unit, in decoding order, of an access unit. An RTP sender implementation may obtain this information from the video encoder

Payload Type (PT): 7 bits

The assignment of an RTP payload type for this new payload format is outside the scope of this document and will not be specified here. The assignment of a payload type has to be performed either through the profile used or in a dynamic way.

Sequence Number (SN): 16 bits

Set and used in accordance with [RFC3550].

Timestamp: 32 bits

The RTP timestamp is set to the sampling timestamp of the content. A 90 kHz clock rate MUST be used. If the NAL unit has no timing properties of its own (e.g., parameter sets or certain SEI NAL units), the RTP timestamp MUST be set to the RTP timestamp of the coded picture of the access unit in which the NAL unit (according to Annex D of [EVC]) is included. Receivers MUST use the RTP timestamp for the display process, even when the bitstream contains picture timing SEI messages or decoding unit information SEI messages as specified in [EVC].

Synchronization source (SSRC): 32 bits

Used to identify the source of the RTP packets. When using SRST, by definition a single SSRC is used for all parts of a single bitstream.

4.2. Payload Header Usage

The first two bytes of the payload of an RTP packet are referred to as the payload header. The payload header consists of the same fields (F, TID, Reserve and E) as the NAL unit header as shown in Section 1.1.4, irrespective of the type of the payload structure.

The TID value indicates (among other things) the relative importance of an RTP packet, for example, because NAL units belonging to higher temporal sub-layers are not used for the decoding of lower temporal sub-layers. A lower value of TID indicates a higher importance. More-important NAL units MAY be better protected against transmission losses than less-important NAL units.

4.3. Payload Structures

Three different types of RTP packet payload structures are specified. A receiver can identify the type of an RTP packet payload through the Type field in the payload header.

The Three different payload structures are as follows:

- Single NAL unit packet: Contains a single NAL unit in the payload, and the NAL unit header of the NAL unit also serves as the payload header. This payload structure is specified in Section 4.3.1.

- Aggregation Packet (AP): Contains more than one NAL unit within one access unit. This payload structure is specified in Section 4.3.2.

- o Fragmentation Unit (FU): Contains a subset of a single NAL unit. This payload structure is specified in Section 4.3.3.

4.3.1. Single NAL Unit Packets

A single NAL unit packet contains exactly one NAL unit, and consists of a payload header (denoted as PayloadHdr), a conditional 16-bit DONL field (in network byte order), and the NAL unit payload data (the NAL unit excluding its NAL unit header) of the contained NAL unit, as shown in Figure 3.

```
      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
     +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
     |           PayloadHdr          |      DONL (conditional)       |
     +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
     |                                                               |
     |                   NAL unit payload data                       |
     |                                                               |
     |                               +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
     |                               :...OPTIONAL RTP padding        |
     +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

The Structure of a Single NAL Unit Packet

Figure 3

The DONL field, when present, specifies the value of the 16 least significant bits of the decoding order number of the contained NAL unit. If sprop-max-don-diff is greater than 0 for any of the RTP streams, the DONL field MUST be present, and the variable DON for the contained NAL unit is derived as equal to the value of the DONL field. Otherwise sprop-max-don-diff is equal to 0 for all the RTP streams), the DONL field MUST NOT be present.

4.3.2. Aggregation Packets (APs)

Aggregation Packets (APs) are introduced to enable the reduction of packetization overhead for small NAL units, such as most of the non-VCL NAL units, which are often only a few octets in size.

An AP aggregates NAL units within one access unit. Each NAL unit to be carried in an AP is encapsulated in an aggregation unit. NAL units aggregated in one AP are in NAL unit decoding order.

An AP consists of a payload header (denoted as PayloadHdr) followed by two or more aggregation units, as shown in Figure 4.

```
     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
    +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
    |    PayloadHdr (Type=56)       |                               |
    +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+                               |
    |                                                               |
    |             two or more aggregation units                     |
    |                                                               |
    |                               +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
    |                               :...OPTIONAL RTP padding        |
    +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

The Structure of an Aggregation Packet

Figure 4

The fields in the payload header are set as follows. The F bit MUST be equal to 0 if the F bit of each aggregated NAL unit is equal to zero; otherwise, it MUST be equal to 1. The Type field MUST be equal to 56.

The value of TID MUST be the lowest value of TID of all the aggregated NAL units. The value of Reserve and E Must match the version of [EVC] specification.

> Informative note: All VCL NAL units in an AP have the same TID value since they belong to the same access unit. However, an AP may contain non-VCL NAL units for which the TID value in the NAL unit header may be different than the TID value of the VCL NAL units in the same AP.

An AP MUST carry at least two aggregation units and can carry as many aggregation units as necessary; however, the total amount of data in an AP obviously MUST fit into an IP packet, and the size SHOULD be chosen so that the resulting IP packet is smaller than the MTU size so to avoid IP layer fragmentation. An AP MUST NOT contain FUs specified in Section 4.3.3. APs MUST NOT be nested; i.e., an AP must not contain another AP.

The first aggregation unit in an AP consists of a conditional 16-bit DONL field (in network byte order) followed by a 16-bit unsigned size information (in network byte order) that indicates the size of the NAL unit in bytes (excluding these two octets, but including the NAL unit header), followed by the NAL unit itself, including its NAL unit header, as shown in Figure 5.

```
     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
    +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
    |               :       DONL (conditional)      |   NALU size   |
    +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
    |   NALU size   |                                               |
    +-+-+-+-+-+-+-+-+         NAL unit                              |
    |                                                               |
    |                               +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
    |                               :
    +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

The Structure of the First Aggregation Unit in an AP

Figure 5

The DONL field, when present, specifies the value of the 16 least significant bits of the decoding order number of the aggregated NAL unit.

If sprop-max-don-diff is greater than 0 for any of the RTP streams, the DONL field MUST be present in an aggregation unit that is the first aggregation unit in an AP, and the variable DON for the aggregated NAL unit is derived as equal to the value of the DONL field. Otherwise (sprop-max-don-diff is equal to 0 for all the RTP streams), the DONL field MUST NOT be present in an aggregation unit that is the first aggregation unit in an AP.

An aggregation unit that is not the first aggregation unit in an AP consists of a conditional 8-bit DOND field followed by a 16-bit unsigned size information (in network byte order) that indicates the size of the NAL unit in bytes (excluding these two octets, but including the NAL unit header), followed by the NAL unit itself, including its NAL unit header, as shown in Figure 6.

```
       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
      +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
                      : DOND (cond)   |          NALU size            |
      +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
      |                                                               |
      |                       NAL unit                                |
      |                               +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
      |                               :
      +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

The Structure of an Aggregation Unit That Is Not
the First Aggregation Unit in an AP

Figure 6

When present, the DOND field plus 1 specifies the difference between the decoding order number values of the current aggregated NAL unit and the preceding aggregated NAL unit in the same AP.

If sprop-max-don-diff is greater than 0 for any of the RTP streams, the DOND field MUST be present in an aggregation unit that is not the first aggregation unit in an AP, and the variable DON for the aggregated NAL unit is derived as equal to the DON of the preceding aggregated NAL unit in the same AP plus the value of the DOND field plus 1 modulo 65536. Otherwise (sprop-max-don-diff is equal to 0 for all the RTP streams), the DOND field MUST NOT be present in an aggregation unit that is not the first aggregation unit in an AP, and in this case the transmission order and decoding order of NAL units carried in the AP are the same as the order the NAL units appear in the AP.

Figure 7 presents an example of an AP that contains two aggregation units, labeled as 1 and 2 in Figure 7, without the DONL and DOND fields being present.

```
       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
      +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
      |                          RTP Header                           |
      +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
      |   PayloadHdr (Type=56)        |         NALU 1 Size           |
      +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
      |          NALU 1 HDR           |                               |
      +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+         NALU 1 Data           |
      |                   . . .                                       |
      |                                                               |
      +               +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
      |  . . .        | NALU 2 Size                   | NALU 2 HDR    |
      +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
      | NALU 2 HDR    |                                               |
      +-+-+-+-+-+-+-+-+              NALU 2 Data                      |
      |                   . . .                                       |
      |                               +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
      |                               :...OPTIONAL RTP padding        |
      +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

An Example of an AP Packet Containing Two Aggregation
Units without the DONL and DOND Fields

Figure 7

Figure 8 presents an example of an AP that contains two aggregation units, labeled as 1 and 2 in the figure, with the DONL and DOND fields being present.

```
       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
      +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
      |                          RTP Header                           |
      +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
      |   PayloadHdr (Type=56)        |        NALU 1 DONL            |
      +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
      |          NALU 1 Size          |            NALU 1 HDR         |
      +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
      |                                                               |
      |                 NALU 1 Data   . . .                           |
      |                                                               |
      +     . . .     +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
      |               |  NALU 2 DOND  |          NALU 2 Size          |
      +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
      |          NALU 2 HDR           |                               |
      +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+          NALU 2 Data          |
      |                                                               |
      |        . . .                  +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
      |                               :...OPTIONAL RTP padding        |
      +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

An Example of an AP Containing Two Aggregation Units
with the DONL and DOND Fields

Figure 8

4.3.3. Fragmentation Units

Fragmentation Units (FUs) are introduced to enable fragmenting a single NAL unit into multiple RTP packets, possibly without cooperation or knowledge of the EVC [EVC] encoder. A fragment of a NAL unit consists of an integer number of consecutive octets of that NAL unit. Fragments of the same NAL unit MUST be sent in consecutive order with ascending RTP sequence numbers (with no other RTP packets within the same RTP stream being sent between the first and last fragment).

When a NAL unit is fragmented and conveyed within FUs, it is referred to as a fragmented NAL unit. APs MUST NOT be fragmented. FUs MUST NOT be nested; i.e., an FU must not contain a subset of another FU.

The RTP timestamp of an RTP packet carrying an FU is set to the NALU-time of the fragmented NAL unit.

An FU consists of a payload header (denoted as PayloadHdr), an FU header of one octet, a conditional 16-bit DONL field (in network byte order), and an FU payload, as shown in Figure 9.

```
     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
    +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
    |    PayloadHdr (Type=57)       |   FU header   | DONL (cond)   |
    +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-|
    | DONL (cond)   |                                               |
    |-+-+-+-+-+-+-+-+                                               |
    |                         FU payload                            |
    |                                                               |
    |                               +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
    |                               :...OPTIONAL RTP padding        |
    +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

The Structure of an FU

Figure 9

The fields in the payload header are set as follows. The Type field MUST be equal to 57. The fields F, TID, Reserve and E MUST be equal to the fields F, TID, Reserve and E, respectively, of the fragmented NAL unit.

The FU header consists of an S bit, an E bit, and a 6-bit FuType field, as shown in Figure 10.

```
                        +---------------+
                        |0|1|2|3|4|5|6|7|
                        +-+-+-+-+-+-+-+-+
                        |S|E|  FuType   |
                        +---------------+
```

The Structure of FU Header

Figure 10

The semantics of the FU header fields are as follows:

S: 1 bit

When set to 1, the S bit indicates the start of a fragmented NAL unit, i.e., the first byte of the FU payload is also the first byte of the payload of the fragmented NAL unit. When the FU payload is not the start of the fragmented NAL unit payload, the S bit MUST be set to 0.

E: 1 bit

When set to 1, the E bit indicates the end of a fragmented NAL unit, i.e., the last byte of the payload is also the last byte of the fragmented NAL unit. When the FU payload is not the last fragment of a fragmented NAL unit, the E bit MUST be set to 0.

FuType: 6 bits

The field FuType MUST be equal to the field Type of the fragmented NAL unit.

The DONL field, when present, specifies the value of the 16 least significant bits of the decoding order number of the fragmented NAL unit.

If sprop-max-don-diff is greater than 0 for any of the RTP streams, and the S bit is equal to 1, the DONL field MUST be present in the FU, and the variable DON for the fragmented NAL unit is derived as equal to the value of the DONL field. Otherwise (sprop-max-don-diff is equal to 0 for all the RTP streams, or the S bit is equal to 0), the DONL field MUST NOT be present in the FU.

A non-fragmented NAL unit MUST NOT be transmitted in one FU; i.e., the Start bit and End bit must not both be set to 1 in the same FU header.

The FU payload consists of fragments of the payload of the fragmented NAL unit so that if the FU payloads of consecutive FUs, starting with an FU with the S bit equal to 1 and ending with an FU with the E bit equal to 1, are sequentially concatenated, the payload of the fragmented NAL unit can be reconstructed. The NAL unit header of the fragmented NAL unit is not included as such in the FU payload, but rather the information of the NAL unit header of the fragmented NAL unit is conveyed in F, TID, Reserve and E fields of the FU payload headers of the FUs and the FuType field of the FU header of the FUs. An FU payload MUST NOT be empty.

If an FU is lost, the receiver SHOULD discard all following fragmentation units in transmission order corresponding to the same fragmented NAL unit, unless the decoder in the receiver is known to gracefully handle incomplete NAL units.

A receiver in an endpoint or in a MANE MAY aggregate the first n-1 fragments of a NAL unit to an (incomplete) NAL unit, even if fragment n of that NAL unit is not received. In this case, the forbidden_zero_bit of the NAL unit MUST be set to 1 to indicate a syntax violation.

4.4. Decoding Order Number

For each NAL unit, the variable AbsDon is derived, representing the decoding order number that is indicative of the NAL unit decoding order.

Let NAL unit n be the n-th NAL unit in transmission order within an RTP stream.

If sprop-max-don-diff is equal to 0 for all the RTP streams carrying the HEVC bitstream, AbsDon[n], the value of AbsDon for NAL unit n, is derived as equal to n.

Otherwise (sprop-max-don-diff is greater than 0 for any of the RTP streams), AbsDon[n] is derived as follows, where DON[n] is the value of the variable DON for NAL unit n:

- o If n is equal to 0 (i.e., NAL unit n is the very first NAL unit in transmission order), AbsDon[0] is set equal to DON[0].

- o Otherwise (n is greater than 0), the following applies for derivation of AbsDon[n]:

```
If DON[n] == DON[n-1],
   AbsDon[n] = AbsDon[n-1]

If (DON[n] > DON[n-1] and DON[n] - DON[n-1] < 32768),
   AbsDon[n] = AbsDon[n-1] + DON[n] - DON[n-1]

If (DON[n] < DON[n-1] and DON[n-1] - DON[n] >= 32768),
   AbsDon[n] = AbsDon[n-1] + 65536 - DON[n-1] + DON[n]

If (DON[n] > DON[n-1] and DON[n] - DON[n-1] >= 32768),
   AbsDon[n] = AbsDon[n-1] - (DON[n-1] + 65536 -
   DON[n])

If (DON[n] < DON[n-1] and DON[n-1] - DON[n] < 32768),
   AbsDon[n] = AbsDon[n-1] - (DON[n-1] - DON[n])
```

For any two NAL units m and n, the following applies:

- o AbsDon[n] greater than AbsDon[m] indicates that NAL unit n follows NAL unit m in NAL unit decoding order.

- o When AbsDon[n] is equal to AbsDon[m], the NAL unit decoding order of the two NAL units can be in either order.

- AbsDon[n] less than AbsDon[m] indicates that NAL unit n precedes NAL unit m in decoding order.

Informative note: When two consecutive NAL units in the NAL unit decoding order have different values of AbsDon, the absolute difference between the two AbsDon values may be greater than or equal to 1.

Informative note: There are multiple reasons to allow for the absolute difference of the values of AbsDon for two consecutive NAL units in the NAL unit decoding order to be greater than one. An increment by one is not required, as at the time of associating values of AbsDon to NAL units, it may not be known whether all NAL units are to be delivered to the receiver. For example, a gateway may not forward VCL NAL units of higher sub-layers or some SEI NAL units when there is congestion in the network. In another example, the first intra-coded picture of a pre-encoded clip is transmitted in advance to ensure that it is readily available in the receiver, and when transmitting the first intra-coded picture, the originator does not exactly know how many NAL units will be encoded before the first intra-coded picture of the pre-encoded clip follows in decoding order. Thus, the values of AbsDon for the NAL units of the first intra-coded picture of the pre-encoded clip have to be estimated when they are transmitted, and gaps in values of AbsDon may occur.

5. Packetization Rules

The following packetization rules apply:

- If sprop-max-don-diff is greater than 0 for any of the RTP streams, the transmission order of NAL units carried in the RTP stream MAY be different than the NAL unit decoding order and the NAL unit output order.

- A NAL unit of a small size SHOULD be encapsulated in an aggregation packet together with one or more other NAL units in order to avoid the unnecessary packetization overhead for small NAL units. For example, non-VCL NAL units such as access unit delimiters, parameter sets, or SEI NAL units are typically small and can often be aggregated with VCL NAL units without violating MTU size constraints.

- Each non-VCL NAL unit SHOULD, when possible from an MTU size match viewpoint, be encapsulated in an aggregation packet together with its associated VCL NAL unit, as typically a non-VCL NAL unit would be meaningless without the associated VCL NAL unit being available.

- For carrying exactly one NAL unit in an RTP packet, a single NAL unit packet MUST be used.

6. De-packetization Process

The general concept behind de-packetization is to get the NAL units out of the RTP packets in an RTP stream and pass them to the decoder in the NAL unit decoding order.

The de-packetization process is implementation dependent. Therefore, the following description should be seen as an example of a suitable implementation. Other schemes may be used as well, as long as the output for the same input is the same as the process described below. The output is the same when the set of output NAL units and their order are both identical. Optimizations relative to the described algorithms are possible.

All normal RTP mechanisms related to buffer management apply. In particular, duplicated or outdated RTP packets (as indicated by the RTP sequences number and the RTP timestamp) are removed. To determine the exact time for decoding, factors such as a possible intentional delay to allow for proper inter-stream synchronization must be factored in.

NAL units with NAL unit type values in the range of 0 to 55, inclusive, may be passed to the decoder. NAL-unit-like structures with NAL unit type values in the range of 56 to 63, inclusive, MUST NOT be passed to the decoder.

The receiver includes a receiver buffer, which is used to compensate for transmission delay jitter within individual RTP streams and across RTP streams, to reorder NAL units from transmission order to the NAL unit decoding order. In this section, the receiver operation is described under the assumption that there is no transmission delay jitter within an RTP stream. To make a difference from a practical receiver buffer that is also used for compensation of transmission delay jitter, the receiver buffer is hereafter called the de-packetization buffer in this section. Receivers should also prepare for transmission delay jitter; that is, either reserve separate buffers for transmission delay jitter buffering and de-packetization buffering or use a receiver buffer for both transmission delay jitter and de-packetization. Moreover, receivers should take transmission delay jitter into account in the buffering operation, e.g., by additional initial buffering before starting of decoding and playback.

When sprop-max-don-diff is equal to 0 for the received RTP stream, the de-packetization buffer size is zero bytes, and the process described in the remainder of this paragraph applies. The NAL units carried in the RTP stream are directly passed to the decoder in their transmission order, which is identical to their decoding order. When there are several NAL units of the same RTP stream with the same NTP timestamp, the order to pass them to the decoder is their transmission order.

> Informative note: The mapping between RTP and NTP timestamps is conveyed in RTCP SR packets. In addition, the mechanisms for faster media timestamp synchronization discussed in [RFC6051] may be used to speed up the acquisition of the RTP-to-wall-clock mapping.

When sprop-max-don-diff is greater than 0 for the received RTP stream the process described in the remainder of this section applies.

There are two buffering states in the receiver: initial buffering and buffering while playing. Initial buffering starts when the reception is initialized. After initial buffering, decoding and playback are started, and the buffering-while-playing mode is used.

Regardless of the buffering state, the receiver stores incoming NAL units, in reception order, into the de-packetization buffer. NAL units carried in RTP packets are stored in the de-packetization buffer individually, and the value of AbsDon is calculated and stored for each NAL unit.

Initial buffering lasts until condition A (the difference between the greatest and smallest AbsDon values of the NAL units in the de-packetization buffer is greater than or equal to the value of sprop-max-don-diff) or condition B (the number of NAL units in the de-packetization buffer is greater than the value of sprop-depack-buf-nalus) is true.

After initial buffering, whenever condition A or condition B is true, the following operation is repeatedly applied until both condition A and condition B become false:

- o The NAL unit in the de-packetization buffer with the smallest value of AbsDon is removed from the de-packetization buffer and passed to the decoder.

When no more NAL units are flowing into the de-packetization buffer, all NAL units remaining in the de-packetization buffer are removed from the buffer and passed to the decoder in the order of increasing AbsDon values.

7. Payload Format Parameters

Placeholder

8. Use with Feedback Messages

Placeholder

8.1. Picture Loss Indication (PLI)

Placeholder

8.2. Slice Loss Indication (SLI)

Placeholder

8.3. Reference Picture Selection Indication (RPSI)

Placeholder

8.4. Full Intra Request (FIR)

Placeholder

9. Use With Framemarking

Placeholder

10. Security Considerations

The scope of this Security Considerations section is limited to the payload format itself and to one feature of [EVC] that may pose a particularly serious security risk if implemented naively. The payload format, in isolation, does not form a complete system. Implementers are advised to read and understand relevant security-related documents, especially those pertaining to RTP (see the Security Considerations section in [RFC3550]), and the security of the call-control stack chosen (that may make use of the media type registration of this memo). Implementers should also consider known security vulnerabilities of video coding and decoding implementations in general and avoid those.

Within this RTP payload format, neither the various media-plane-based mechanisms, nor the signaling part of this memo, seems to pose a security risk beyond those common to all RTP-based systems.

RTP packets using the payload format defined in this specification are subject to the security considerations discussed in the RTP specification [RFC3550] , and in any applicable RTP profile such as RTP/AVP [RFC3551] , RTP/AVPF [RFC4585] , RTP/SAVP [RFC3711], or RTP/SAVPF [RFC5124] . However, as "Securing the RTP Framework: Why RTP Does Not Mandate a Single Media Security Solution" [RFC7202] discusses, it is not an RTP payload format’s responsibility to discuss or mandate what solutions are used to meet the basic security goals like confidentiality, integrity and source authenticity for RTP in general. This responsibility lays on anyone using RTP in an application. They can find guidance on available security mechanisms and important considerations in "Options for Securing RTP Sessions" [RFC7201]. Applications SHOULD use one or more appropriate strong security mechanisms. The rest of this section discusses the security impacting properties of the payload format itself.

Because the data compression used with this payload format is applied end-to-end, any encryption needs to be performed after compression. A potential denial-of-service threat exists for data encodings using compression techniques that have non-uniform receiver-end computational load. The attacker can inject pathological datagrams into the bitstream that are complex to decode and that cause the receiver to be overloaded. EVC is particularly vulnerable to such attacks, as it is extremely simple to generate datagrams containing NAL units that affect the decoding process of many future NAL units. Therefore, the usage of data origin authentication and data integrity protection of at least the RTP packet is RECOMMENDED, for example, with SRTP [RFC3711].

End-to-end security with authentication, integrity, or confidentiality protection will prevent a MANE from performing media-aware operations other than discarding complete packets. In the case of confidentiality protection, it will even be prevented from discarding packets in a media-aware way. To be allowed to perform such operations, a MANE is required to be a trusted entity that is included in the security context establishment.

11. Congestion Control

Congestion control for RTP SHALL be used in accordance with RTP [RFC3550] and with any applicable RTP profile, e.g., AVP [RFC3551]. If best-effort service is being used, an additional requirement is that users of this payload format MUST monitor packet loss to ensure that the packet loss rate is within an acceptable range. Packet loss is considered acceptable if a TCP flow across the same network path, and experiencing the same network conditions, would achieve an average throughput, measured on a reasonable timescale, that is not less than all RTP streams combined is achieving. This condition can be satisfied by implementing congestion-control mechanisms to adapt the transmission rate, the number of layers subscribed for a layered multicast session, or by arranging for a receiver to leave the session if the loss rate is unacceptably high.

The bitrate adaptation necessary for obeying the congestion control principle is easily achievable when real-time encoding is used, for example, by adequately tuning the quantization parameter. However, when pre-encoded content is being transmitted, bandwidth adaptation requires the pre-coded bitstream to be tailored for such adaptivity. The key mechanism available in [EVC] is temporal scalability. A media sender can remove NAL units belonging to higher temporal sub-layers (i.e., those NAL. units with a high value of TID) until the sending bitrate drops to an acceptable range.

Above mechanisms generally work within a defined profile and level and, therefore, no renegotiation of the channel is required. Only when non-downgradable parameters (such as profile) are required to be changed does it become necessary to terminate and restart the RTP stream(s). This may be accomplished by using different RTP payload types.

MANEs MAY remove certain unusable packets from the RTP stream when that RTP stream was damaged due to previous packet losses. This can help reduce the network load in certain special cases. For example, MANES can remove those FUs where the leading FUs belonging to the same NAL unit have been lost or those dependent slice segments when the leading slice segments belonging to the same slice have been lost, because the trailing FUs or dependent slice segments are meaningless to most decoders. MANES can also remove higher temporal scalable layers if the outbound transmission (from the MANE’s viewpoint) experiences congestion.

12. IANA Considertaions

Placeholder

13. Acknowledgements

Large parts of this specification share text with the RTP payload format for HEVC [RFC7798]. We thank the authors of that specification for their excellent work.

14. References

14.1. Normative References

[ISO23090-3]
Bradner, S, ., "Versatile Video Coding", DOI 10.17487/RFC2119,, March 1997, <https://www.rfc-editor.org/info/rfc2119>.

[ISO23094-1]
N/A, ., "Essential Video Coding", 2020.

[RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, DOI 10.17487/RFC2119, March 1997, <https://www.rfc-editor.org/info/rfc2119>.

[RFC3550] Schulzrinne, H., Casner, S., Frederick, R., and V. Jacobson, "RTP: A Transport Protocol for Real-Time Applications", STD 64, RFC 3550, DOI 10.17487/RFC3550, July 2003, <https://www.rfc-editor.org/info/rfc3550>.

[RFC3551] Schulzrinne, H. and S. Casner, "RTP Profile for Audio and Video Conferences with Minimal Control", STD 65, RFC 3551, DOI 10.17487/RFC3551, July 2003, <https://www.rfc-editor.org/info/rfc3551>.

[RFC3711] Baugher, M., McGrew, D., Naslund, M., Carrara, E., and K. Norrman, "The Secure Real-time Transport Protocol (SRTP)", RFC 3711, DOI 10.17487/RFC3711, March 2004, <https://www.rfc-editor.org/info/rfc3711>.

[RFC4566] Handley, M., Jacobson, V., and C. Perkins, "SDP: Session Description Protocol", RFC 4566, DOI 10.17487/RFC4566, July 2006, <https://www.rfc-editor.org/info/rfc4566>.

[RFC4585] Ott, J., Wenger, S., Sato, N., Burmeister, C., and J. Rey, "Extended RTP Profile for Real-time Transport Control Protocol (RTCP)-Based Feedback (RTP/AVPF)", RFC 4585, DOI 10.17487/RFC4585, July 2006, <https://www.rfc-editor.org/info/rfc4585>.

[RFC5104] Wenger, S., Chandra, U., Westerlund, M., and B. Burman, "Codec Control Messages in the RTP Audio-Visual Profile with Feedback (AVPF)", RFC 5104, DOI 10.17487/RFC5104, February 2008, <https://www.rfc-editor.org/info/rfc5104>.

[RFC5124] Ott, J. and E. Carrara, "Extended Secure RTP Profile for Real-time Transport Control Protocol (RTCP)-Based Feedback (RTP/SAVPF)", RFC 5124, DOI 10.17487/RFC5124, February 2008, <https://www.rfc-editor.org/info/rfc5124>.

14.2. Informative References

[CABAC] Sole, J, . and . et al, "Transform coefficient coding in HEVC, IEEE Transactions on Circuts and Systems for Video Technology", DOI 10.1109/TCSVT.2012.2223055, December 2012.

[EVC] "w18774_Text_DIS_23094-1_draft_final_v3.docx", 2019.

[Girod99] Girod, B, . and . et al, "Feedback-based error control for mobile video transmission, Proceedings of the IEEE", DOI 110.1109/5.790632, October 1999.

[MPEG2S] IS0/IEC, ., "Information technology - Generic coding ofmoving pictures and associated audio information - Part 1:Systems, ISO International Standard 13818-1", 2013.

[RFC6051] Perkins, C. and T. Schierl, "Rapid Synchronisation of RTP Flows", RFC 6051, DOI 10.17487/RFC6051, November 2010, <https://www.rfc-editor.org/info/rfc6051>.

[RFC6184] Wang, Y., Even, R., Kristensen, T., and R. Jesup, "RTP Payload Format for H.264 Video", RFC 6184, DOI 10.17487/RFC6184, May 2011, <https://www.rfc-editor.org/info/rfc6184>.

[RFC6190] Wenger, S., Wang, Y., Schierl, T., and A. Eleftheriadis, "RTP Payload Format for Scalable Video Coding", RFC 6190, DOI 10.17487/RFC6190, May 2011, <https://www.rfc-editor.org/info/rfc6190>.

[RFC7201] Westerlund, M. and C. Perkins, "Options for Securing RTP Sessions", RFC 7201, DOI 10.17487/RFC7201, April 2014, <https://www.rfc-editor.org/info/rfc7201>.

[RFC7202] Perkins, C. and M. Westerlund, "Securing the RTP Framework: Why RTP Does Not Mandate a Single Media Security Solution", RFC 7202, DOI 10.17487/RFC7202, April 2014, <https://www.rfc-editor.org/info/rfc7202>.

[RFC7798] Wang, Y., Sanchez, Y., Schierl, T., Wenger, S., and M. Hannuksela, "RTP Payload Format for High Efficiency Video Coding (HEVC)", RFC 7798, DOI 10.17487/RFC7798, March 2016, <https://www.rfc-editor.org/info/rfc7798>.

[VVC] "JVET-P2001-vE-draft-7.docx", 2019.

[Wang05] Wang, YK, ., Zhu, C, ., and . Li, H, "Error resilient video coding using flexible reference fames", Visual Communications and Image Processing 2005 (VCIP 2005) , July 2005.

Appendix A. Change History

Authors’ Addresses

Shuai Zhao
Tencent
2747 Park Blvd
Palo Alto 94588
USA

Email: shuaiizhao@tencent.com

Stephan Wenger
Tencent
2747 Park Blvd
Palo Alto 94588

Email: stewe@stewe.org