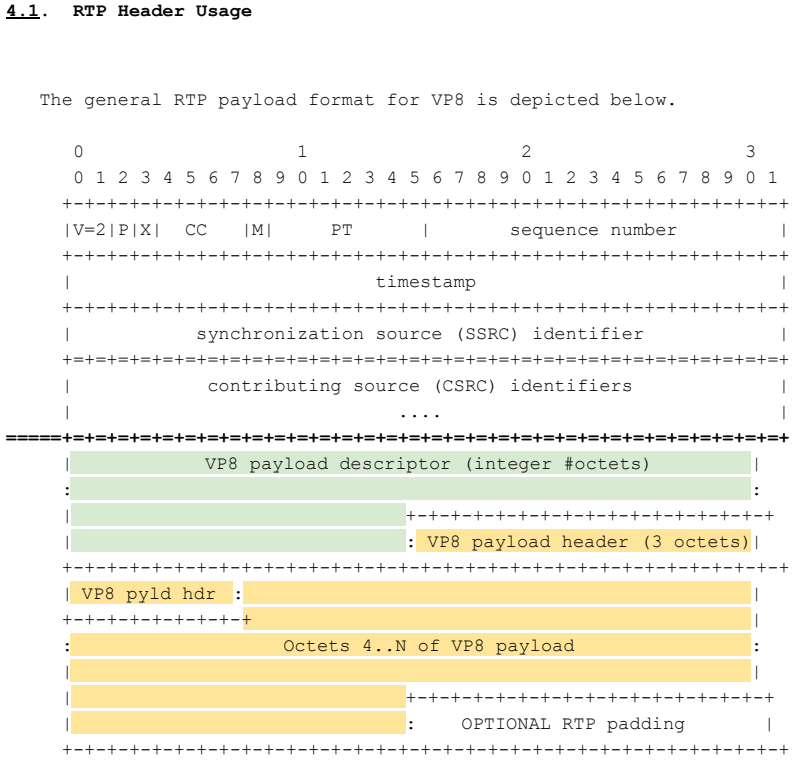
Video Payloads

Usage with SFrame

Using VP8 with SFrame

- Encrypted frames are sent using standard VP8 RTP packetization.
- As a result, the RTP packet, using VP8 packetization contains the RTP VP8 payload description in “clear” and the RTP VP8 payload header and VP8 payload, encrypted.
- SFU can use the VP8 payload descriptor as normal, but I-frame bit is in the VP8 payload header which is encrypted.
- This issue is currently bypassed by skipping the first bytes of the payload when encrypting the video frame.

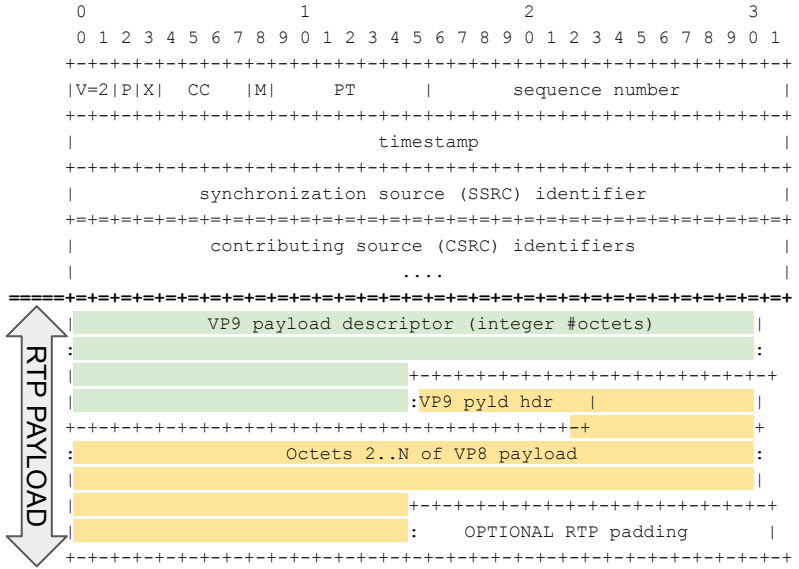


Using VP9 with SFrame

- As in VP8 case, the VP9 payload description is available (after DTLS/SRTP decryption) in clear by the SFU that is able to operate normally.
- But unlike VP8, all info required for SFU operation is in the VP9 Payload Descriptor, which is not encrypted by SFrame.
- When using spatial scalability, each layer frame must be encoded independently by SFrame as the SFU may discard any of them and the rest of the frame must still be decryptable.

4.1. RTP Header Usage

The general RTP payload format for VP9 is depicted below.



Using H264 with SFrame

- SFrame will encrypt the NAL headers while the H264 packetization will replace them with an un-encrypted version, so upon reception, the frame will not be decryptable.
- To solve that, SFrame must work with NAL units, instead of with with frames, increasing overhead.
- Only NAL data is encrypted.

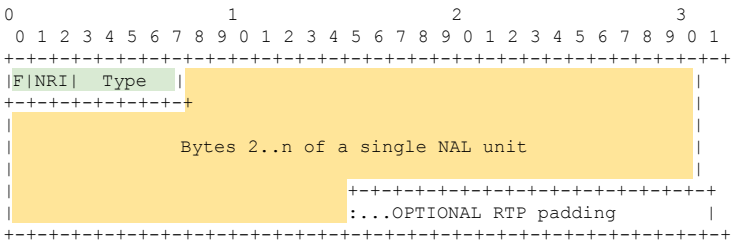


Figure 2. RTP payload format for single NAL unit packet

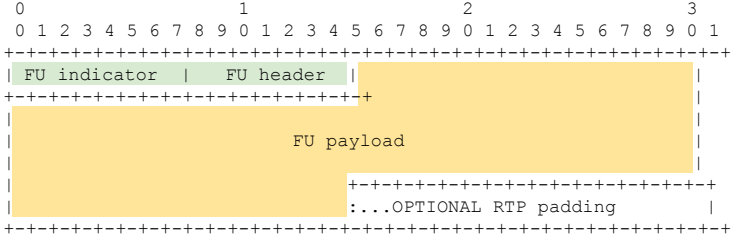


Figure 14. RTP payload format for FU-A

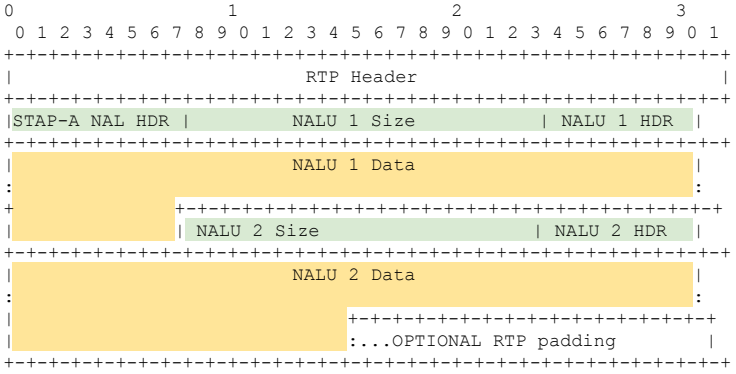
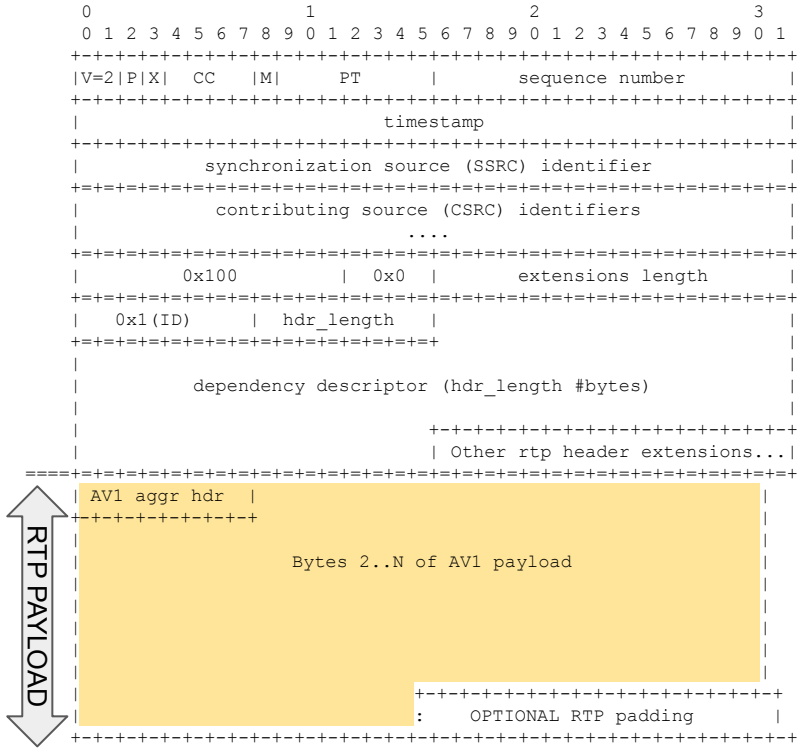


Figure 7. An example of an RTP packet including an STAP-A containing two single-time aggregation units

Using AV1 with SFrame

- AV1 RTP packetization is designed to support e2ee.
- All frame payload is encrypted by SFrame.
- Frame metadata required by the SFU is sent in a new RTP header extension, the Dependency Descriptor
- As in VP9, each spatial frame must be feed. independently to SFrame for encryption.
- SFUs does not need to process RTP payload.
- DD can’t be authenticated e2e, as it must be modifiable by the SFU to signal layer changes.
- DD is encrypted hoop by hoop (RFC 6904 or cryptex).
- DD is codec agnostic, can be applied to any codec.
- DD also provides packetization information, can be use as raw packetizer.
- DD can be reused by non-RTP protocols like QUIC datagrams.



Conclusions

- Not all video codecs support SFrame easily.
- Each video codec requires different processing by SFrame if used with their standard RTP packetization.
- Having a different solution per video codec will require extra specification effort, will make implementation harder and will create huge problems in interoperability.
- SFUs requires access to frame metadata for detecting frame type and performing layer selection.
- If RTP, Frame metadata is better carried on an header extension than inside the payload.
- It would be preferably to have a transport protocol agnostic solution, to complement the protocol agnostic SFrame.