AVTCORE WG
IETF 110

Virtual Meeting

Thursday, March 11, 2020

04:00 - 06:00 Pacific Time

Session I, Room 1

Mailing list: avtcore@ietf.org
Jabber Room: avtcore@jabber.ietf.org
MeetEcho link: https://gce.conf.meetecho.com/conference/?group=avtcore

IETF 110 Meeting Tips

https://www.ietf.org/how/meetings/110

https://datatracker.ietf.org/meeting/agenda

This session is being recorded

- **IETF 110 registration and a datatracker login required to attend**
- **No need to manually fill in blue sheets, it's automatic.**
- **Join the session Jabber room via IETF Datatracker Meeting agenda**
- **Please use headphones when speaking to avoid echo.**
- **Please state your full name before speaking.**

IETF 110 Meeting Tips

https://www.ietf.org/how/meetings/110

https://datatracker.ietf.org/meeting/agenda

This session is being recorded

- **Enter the queue with**  **, leave with** 
- **When you are called on, you need to enable your audio to be heard.**
- **Audio is enabled by unmuting**  **and disabled by muting** 
- **Video can also be enabled, but it is separate from audio.**
- **Video is encouraged to help comprehension but not required.**

Note Well

This is a reminder of IETF policies in effect on various topics such as patents or code of conduct. It is only meant to point you in the right direction. Exceptions may apply. The IETF's patent policy and the definition of an IETF "contribution" and "participation" are set forth in BCP 79; please read it carefully.

As a reminder:

- By participating in the IETF, you agree to follow IETF processes and policies.
- If you are aware that any IETF contribution is covered by patents or patent applications that are owned or controlled by you or your sponsor, you must disclose that fact, or not participate in the discussion.
- As a participant in or attendee to any IETF activity you acknowledge that written, audio, video, and photographic records of meetings may be made public.
- Personal information that you provide to IETF will be handled in accordance with the IETF Privacy Statement.
- As a participant or attendee, you agree to work respectfully with other participants; please contact the ombudsteam (https://www.ietf.org/contact/ombudsteam/) if you have questions or concerns about this.

Definitive information is in the documents listed below and other IETF BCPs. For advice, please talk to WG chairs or ADs:

- BCP 9 (Internet Standards Process)
- BCP 25 (Working Group processes)
- BCP 25 (Anti-Harassment Procedures)
- BCP 54 (Code of Conduct)
- BCP 78 (Copyright)
- BCP 79 (Patents, Participation)
- https://www.ietf.org/privacy-policy/(Privacy Policy)

About this meeting

- Agenda: https://datatracker.ietf.org/doc/agenda-110-avtcore/
- CodiMD (for notes): https://codimd.ietf.org/notes-ietf-110-avtcore
- Jabber Room: avtcore@jabber.ietf.org
- Secretariat: mtd@jabber.ietf.org
- WG Chairs: Jonathan Lennox & Bernard Aboba
- Jabber Scribe:
- Note takers:

Agenda

1. Note Well, Note Takers, Agenda Bashing, Draft status - (Chairs, 10 min)
2. RTP Payload Format for VP9 Video (Jonathan Lennox, 10 min)
   https://datatracker.ietf.org/doc/html/draft-ietf-payload-vp9
3. RTP Payload Format for ISO/IEC 21122 (JPEGXS) (Tim Bruylants, 5 min)
   https://datatracker.ietf.org/doc/html/draft-ietf-payload-rtp-jpegxs
4. Completely Encrypting RTP Header Extensions and Contributing Sources (Cryptex) (Justin Uberti, 5 min)
   https://datatracker.ietf.org/doc/html/draft-ietf-avtcore-cryptex
5. RTP Payload for EVC (Stefan Wenger, 10 min)
   https://datatracker.ietf.org/doc/html/draft-ietf-avtcore-rtp-evc
6. RTP Payload for VVC (Shuai Zhao, 10 min)
   https://datatracker.ietf.org/doc/html/draft-ietf-avtcore-rtp-vvc
7. SFrame RTP encapsulation (Dr. Alexandre Gouaillard, Sergio Garcia Murillo, Youenn Fablet, 60 min)
   https://datatracker.ietf.org/doc/html/draft-gouaillard-avtcore-codec-agn-rtp-payload
8. Wrapup and Next Steps (Chairs, 10 min)

Draft status

- Published
  - RFC 8817: was draft-ietf-payload-tsvcis
  - RFC 8852: was draft-ietf-avtext-rid
  - RFC 8860: was draft-ietf-avtcore-multi-media-rtp-session
  - RFC 8861: was draft-ietf-avtcore-rtp-multi-stream-optimisation
  - RFC 8872: was draft-ietf-avtcore-multiplex-guidelines
  - RFC 8888: was draft-ietf-avtcore-cc-feedback-message

Draft Status (2)

- Publication Requested
  - draft-ietf-avtcore-multi-party-rtt-mix
- Completed WGLC
  - draft-ietf-payload-vp9
  - draft-ietf-avtext-framemarking (3rd WGLC)
  - draft-ietf-payload-rtp-jpegxs (2nd WGLC)
- Expired
  - draft-ietf-payload-tetra (expired January 27, 2020)
- Adopted
  - draft-ietf-avtcore-rtp-evc
  - draft-ietf-avtcore-rfc7983bis
  - draft-ietf-avtcore-cryptex

draft-ietf-avtext-framemarking

- Confirmation of consensus posted to the list on 02 February 2021: Archive (ietf.org)
  - Proposal to publish as Experimental. Implies that support in future codec documents is optional.
  - Move Section 5.4 from draft-ietf-payload-vp9-10 into the document.
- 4 responses received, no objections to proposed consensus.
  - Suggestion to document VP8 PictureId issues.
    - Question about where this should be done (Spencer).
  - Question on whether VVC/EVC are required to support it.

draft-ietf-payload-tetra

- Chairs e-mailed draft authors; no response.
- Proposal: drop milestone.
  - It can be revived if interest returns.
- Any objections?

RTP Payload Format for VP9 Video

https://datatracker.ietf.org/doc/html/draft-ietf-payload-vp9

J. Lennox

draft-ietf-payload-vp9

- draft-ietf-payload-vp9-11 published.
- BCP 78/79 obligations acknowledged by authors.
- IPR disclosures filed (next slide)
- SDP questions from Christer: Archive (ietf.org)
- Proposal: Publication request to be sent once -12 hits the archive.
  - Any objections?

draft-ietf-payload-vp9

IPR Disclosures Discussion

Document IPR search results for draft-ietf-payload-vp9

Total number of IPR disclosures found: **4**.

Date	ID	Statement
2020-11-23	4497	Nokia Technologies Oy's Statement about IPR related to draft-ietf-payload-vp9 (Updates ID#: 4496)
2020-11-23	4496	Nokia Technologies Oy's Statement about IPR related to draft-ietf-payload-vp9
2020-11-20	4491	Nokia Technologies Oy's Statement about IPR related to draft-ietf-payload-vp9
2015-05-04	2593	Vidyo, Inc.'s Statement about IPR related to draft-uberti-payload-vp9

SDP Questions

- Comment from Christer on media type parameters: profile-id, max-fr, max-fs.
- Jonathan's proposal:
  - profile-id MUST remain unchanged in answers and updated offers
    - If you want to negotiate a new profile-id, use a new PT number
  - max-fr and max-fs are receiver capabilities and are declarative: MAY be added or changed in answers or updated offers
- Need rules on how they're processed.
  - Question: Sender SHOULD NOT or MUST NOT exceed max-fr and max-fs?
- How do implementations (browsers) handle this today?
  - Do they notice updated max-fr and max-fs?
  - What do they do if they receive a stream violating the max-fr/max-fs they sent?

RTP Payload Format for ISO/IEC 21122 (JPEGXS)

https://datatracker.ietf.org/doc/html/draft-ietf-payload-rtp-jpegxs

T. Bruylants

draft-ietf-payload-rtp-jpegxs

- WGLC announcement on 08 February 2021: Archive (ietf.org)
- Concluded February 23, 2021.
- 5 Responses (3 positive, no objections, 2 questions)
  - SDP questions from Christer Holmberg: Archive (ietf.org)
  - SDP question from Inge Hillestad: Archive (ietf.org)

draft-ietf-payload-rtp-jpegxs

- SDP questions from Christer Holmberg: Archive (ietf.org)
  - Section 6.2.4 (Offer/Answer Considerations) is rewritten
  - SDP optional vs mandatory parameters: We believe this is clear in the text already. All parameters are optional, except for *rate* and *transmode*
  - Wording of “SDP object” is replaced by “SDP media description”
- SDP question from Inge Hillestad: Archive (ietf.org)
  - “interlaced” parameter changed back to “interlace”
- Clock rate of 90000 is now mandatory
- draft-09 (submitted): addresses all questions

Completely Encrypting RTP Header Extensions and Contributing Sources (Cryptex)

https://datatracker.ietf.org/doc/html/draft-ietf-avtcore-cryptex

J. Uberti

New Since -00

- WG document
- Changed SDP negotiation attribute to "a=cryptex"
- Fixed references

Next Steps

- Test vectors
- WGLC

RTP Payload Format for EVC

Shuai Zhao, Stephan Wenger (Tencent)

Youngkwon Lim (Samsung Electronics)

https://datatracker.ietf.org/doc/html/draft-ietf-avtcore-rtp-evc

EVC Payload Format Design Principles

- Inherited structure from HEVC/VVC
- Payload structure design mostly follows VVC design:
- No support for PAyload Content Information (PACI) packets akin RFC 7798
- No support for MRST, MRMT, only SRST
- No DOND-based signaling for interleaving support
- No support of Framemarking
- No support of SLI and RPSI FB message

Recap Since IETF 109

- Adopted as WG draft after IETF 108
- WG accepted Youngkwon Lim (Samsung) as co-author
- Current version: draft-ietf-avtcore-rtp-evc-01
  - mostly focus on refining EVC codec specification (Thanks Youngkwon for your detailed comments)

TODO

- Resolve existing editor's notes and placeholders
- Section 7 SDP parameters:
- Will closely follow the VVC payload format because of the similarity
- SDP offer/answer
- IANA considerations
- We aim for WGLC in summer 2021

RTP Payload Format for VVC

Shuai Zhao, Stephan Wenger (Tencent)

Yago Sanchez (Fraunhofer HHI)

Ye-Kui Wang (Bytedance Inc)

https://datatracker.ietf.org/doc/html/draft-ietf-avtcore-rtp-vvc

Update Summary Since IETF 109

- 3 revisions: -06, -07 & -08
- draft-ietf-avtcore-rtp-vvc-08
- Mostly focus on resolving 24 editors' notes

draft-ietf-avtcore-rtp-vvc-07

- Resolved 6 editors' note:
  - Removed support of sprop-opi,
    - Remove editor-note #5, #6, #17, updated editor-note #7
  - Updated text for recv-ols-id
  - Removed editor-note #9, #15, #16 based on confirmed value in recent VVC spec
- Email discussion was initiated and received no comments, therefore implemented as proposed

draft-ietf-avtcore-rtp-vvc-08

(Review is needed!!)

- Authors agreed on the following resolutions for 13 Editor's notes below:
  - #1: Framemarking is no longer supported, therefore removed.
  - #2: VVC spec is mature, nothing to update, therefore removed
  - #3: Since the concept of DONL and sprop-max-don-diff have not ever changed since AVC/HEVC payload format, therefore removed.
  - #4: Optional parameter list will be updated in editor-note 20, therefore removed
  - #7: decided not to support opi, therefore removed
  - #8, #10, #12 #13 removed based on text update for profile-id
  - #18: no longer support max-lps, therefore removed
  - #22 and 23: no long support SLI and RPSI, therefore removed
  - #24: proposed text for FU R bit (editor's note will be removed if no comments received)

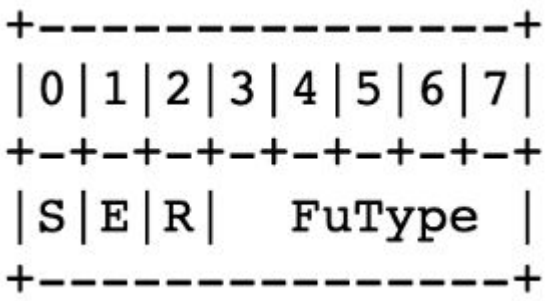
draft-ietf-avtcore-rtp-vvc-08 (cont)

- Removed informative note under ‘M’ bit
- Removed support for the following SDP optional parameters:
  - max-br, max-lps, max-cpb, max-dpb, max-tr, max-tc
  - Removed Framemarking (section 9) and its reference

draft-ietf-avtcore-rtp-vvc-08 (cont)

- Proposed text for the “Reserved” ‘R’ bit in Fragmentation Unit Header:

  “**When set to 1, the R bit indicates the last NAL unit of a coded picture, i.e., the last byte of the FU payload is also the last byte of the coded picture. When the FU payload is not the last fragment of a coded picture, the R bit MUST be set to 0."**



Next Step

- Continue to work on 4 remaining editor's notes
- Complete SDP O/A section and IANA Considerations
- WGLC expected June 2021 the latest

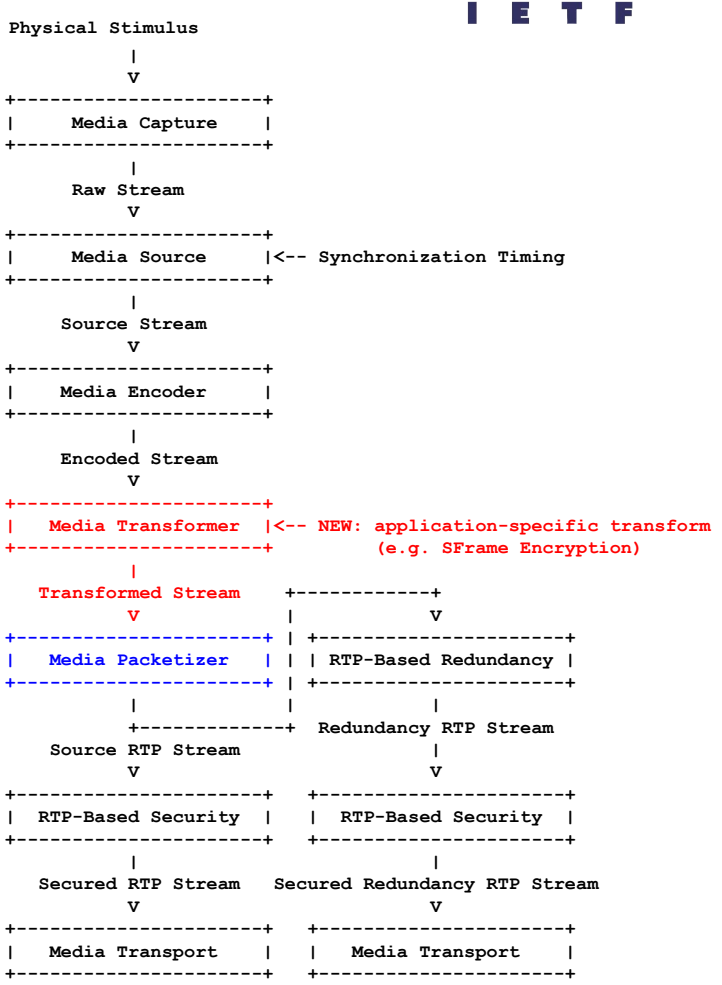
Codec agnostic RTP payload format

Sergio Garcia Murillo, Youenn Fablet

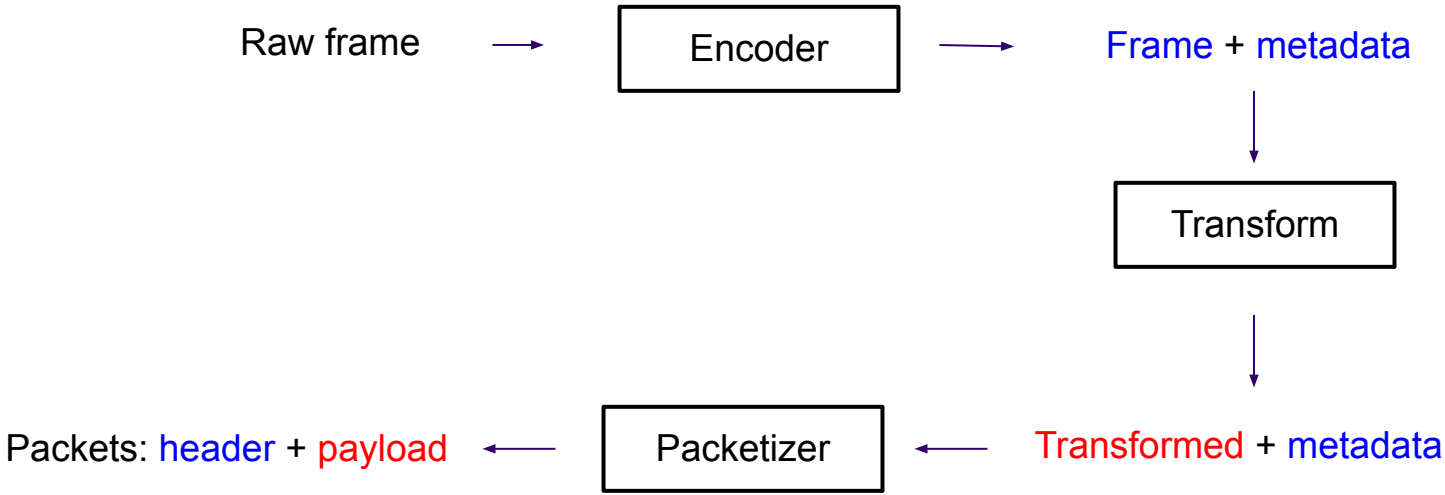
https://www.ietf.org/archive/id/draft-gouaillard-avtcore-codec-agn-rtp-payload-00.txt

Why a codec agnostic packetizer?

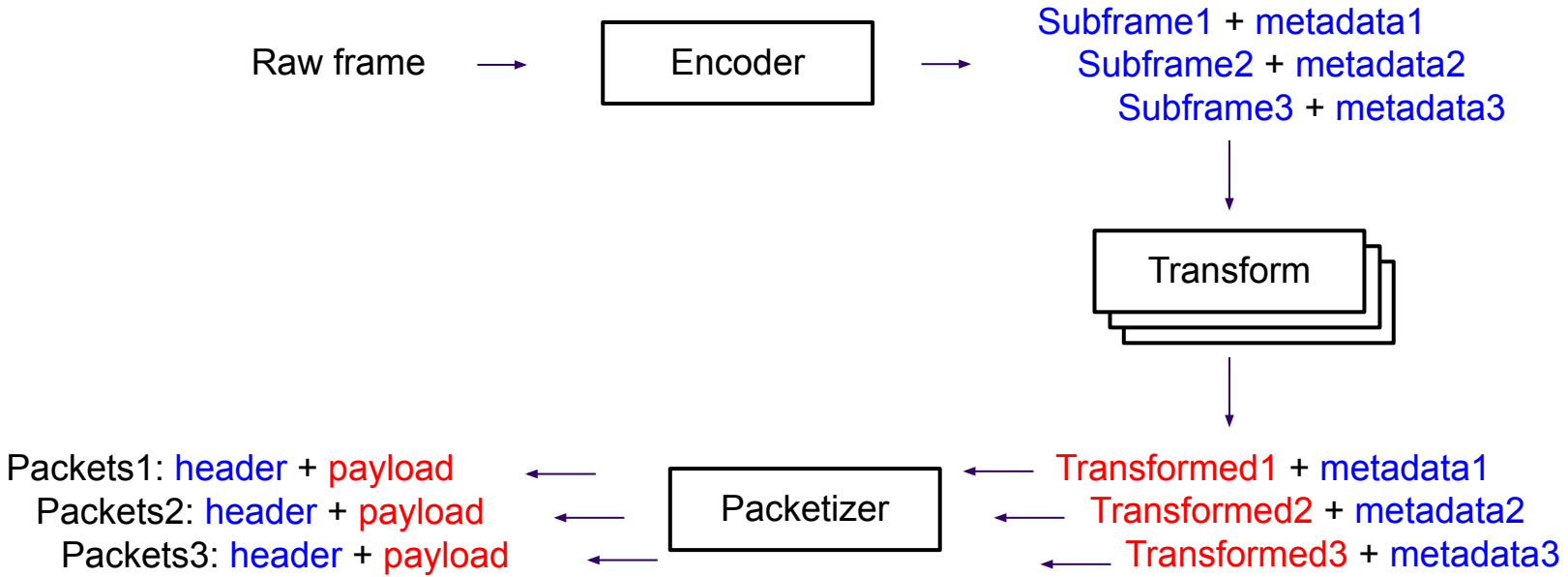
- SFrame End-to-end encryption is implemented by inserting application-specific Media Transformers between Media Encoder and Media Packetizer.
- Current Media Packetizer formats do not support this Transformed stream input and a per-codec “hack” is required: https://datatracker.ietf.org/meeting/109/materials/slides-109-sframe-video-payloads-usage-with-sframe-00
- The generic packetization is intended to solve this problem by defining a new Media Packetizer that works with media content in a codec-agnostic way.



How does it work with H.264/VP8?

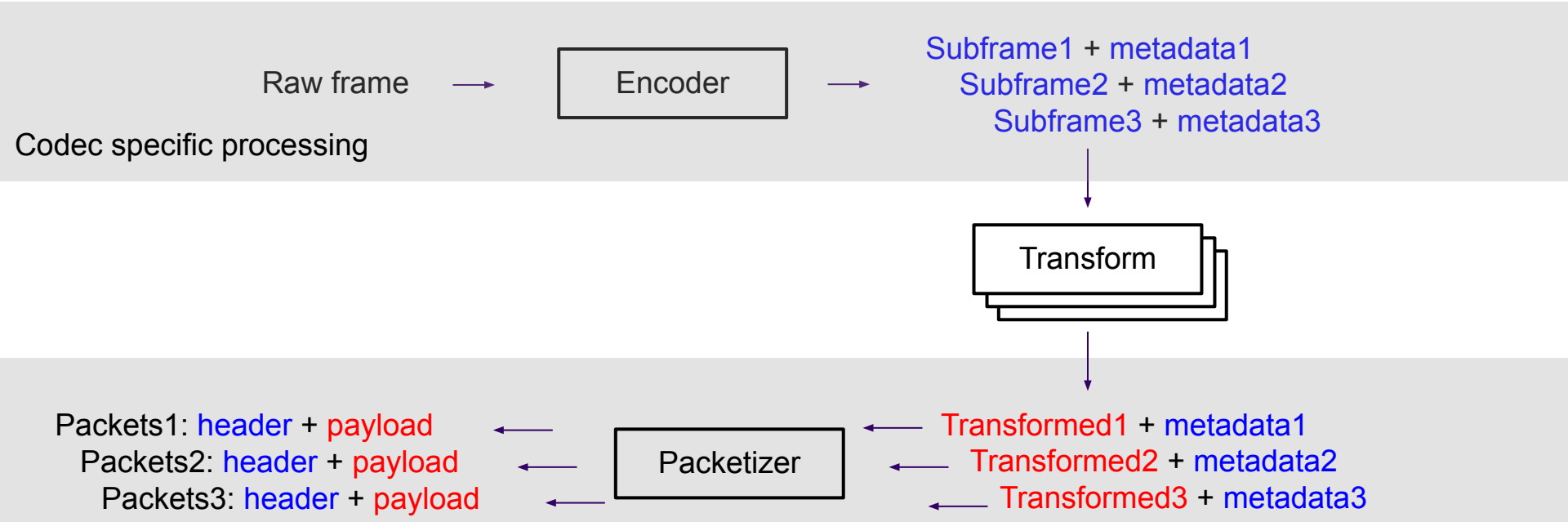


How does it work with VP9/SVC?



In the draft, a subframe is simply called an (encoded) frame

How does it work with VP9/SVC?



Codec agnostic processing

At least for payload, header generation from metadata may be codec-specific

Codec agnostic RTP payload format requirements

MUST NOT DO

- This generic packetization must not change how the mapping between one or several encoded or dependant streams are mapped to the RTP streams or how the synchronization sources(s) (SSRC) are assigned.
- Modify the behavior of the redundancy or recovery mechanism. That is, RTX, RED, ULPFEC, FLEX FEC must work with the codec agnostic RTP out of the box without any modifications.
- Must not interfere with BWE or CC, RMCAT RTCP FB, transport wide CC or REMB must work without any modifications.

MUST DO

- Simulcast must work.
- Support Single RTP stream on a Single media Transport (SRST) when Scalable Video Coding (SVC) is in use.
- Negotiation is done via SDP.
- Allow SFUs to perform layer selection and packet forwarding.

SFU processing rules

- SFUs should rely on RTP header information for packet selection, but not on RTP payload.
- SFUs should not need to rewrite RTP payloads.
  - Rewriting RTP header is fine.
- Metadata exposed as RTP header information to SFU should not create privacy issues.

RTP Packetization

- When the packetizer receives a frame from the application, it MUST fragment the frame content in multiple RTP packets to ensure packets do not exceed the network MTU.
- The content of the frame will be treated as a binary blob by the packetizer, so the decision about the boundaries of each fragment is decided arbitrarily by the packetizer.
- The marker bit of each RTP packet in a frame MUST be set according to the audio and video profiles specified in RFC3551.
- In the case of a video codec supporting spatial scalability, each spatial layer MUST be split in its own frame by the application before passing it to the packetizer.
- The spatial layer frames are sent in ascending order, with the same RTP timestamp, and only the last RTP packet of the last spatial layer frame will have the marker bit set to 1.

SDP negotiation

```
a=rtpmap:96 vp9/90000
a=rtpmap:97 vp8/90000
a=rtpmap:98 generic/90000
a=rtpmap:99 rtx/90000
a=fmtp:99 apt=96
a=rtpmap:100 rtx/90000
a=fmtp:100 apt=97
a=rtpmap:101 rtx/90000
a=fmtp:101 apt=98
```

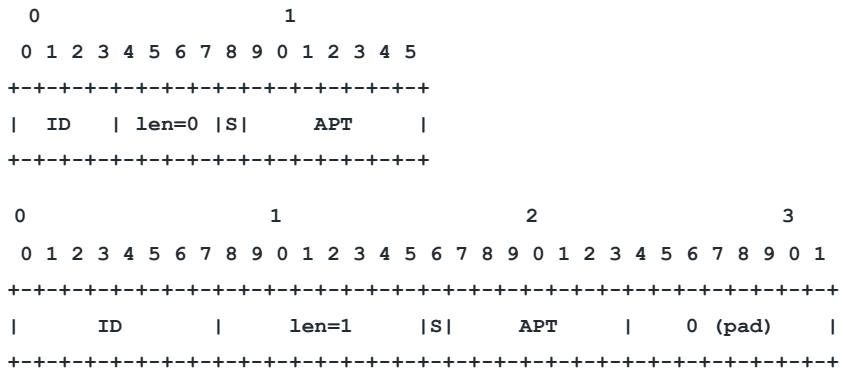
- Generic packetizer negotiation relies on negotiation of the standard packetizer for the each codec.
- Negotiate a generic payload type for all codecs in order to avoid duplication or triplication the number of payload types in use.
- Requires sending actual codec payload type, as a RTP header extension or as a prefix to the payload, which will cause minor network overhead.
- Requires negotiating different payload types for each clock rate for audio.

Payload Type Multiplexing

- In order to reduce the number of payload type in the SDP exchange, a single payload type is used all negotiated media formats.
- That requires to identify the original payload type code of the negotiated media format, called the associated payload type (APT).
- The APT value is the payload type code of the associated format passed to the generic Media Packetizer before any transformation is applied.

Payload Type Multiplexing (cont)

- The APT value is sent in a dedicated header extension:



- The APT value is the associated payload type value.
  - The S bit indicates if the media stream can be forwarded safely starting from this RTP packet. It will be set to 1 on the first RTP packet of an intra video frame and in all RTP audio packets.
  - Receivers MUST be ready to receive RTP packets with different associated payload types in the same way they would receive different payload type codes on the RTP packets.
- The URI for declaring this header extension in an extmap attribute is `urn:ietf:params:rtp-hdrext:associated-payload-type`

Frame metadata

Potential metadata of interest

- At which packet SFU can switch (SVC and simulcast).
- Resolution and more generally stream 'quality': frame rate, bit rate...
- Codec specific information like profile/levels.
- Recovery mechanism required in case of loss (none, RTX, LRR/PLI).
- Opus TOC to know frame length (recording scenarios).

Potential solutions

- Use/extend frame marking.
- Use AV1 Dependency Descriptor (current proposed solution).
- Design a new RTP header extension, potentially complemented with either frame marking or AV1 Dependency Descriptor.

How does it work with redundancy?

RTX: no change required

- RTX will allow sending missing packets, both header and payload.

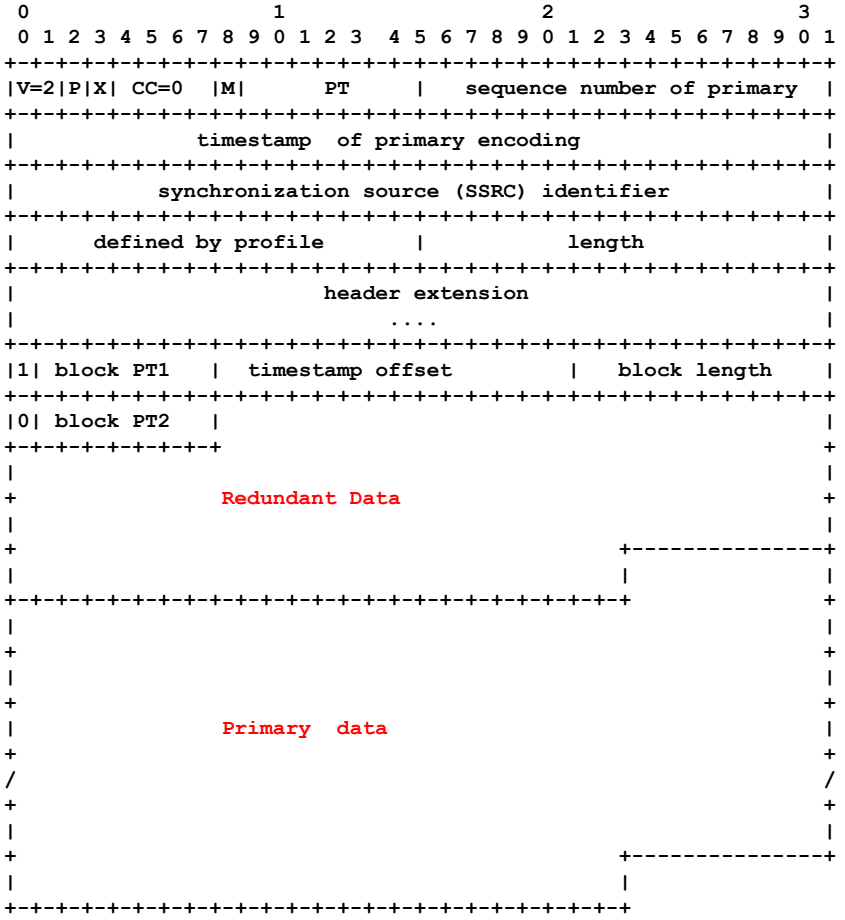
FEC: no change required

- FEC will allow protecting some packets more than others.
- But FEC heuristics might change in terms of granularity.
  - Not possible to protect specifically SPS/PPS NALUs.
  - Possible to protect packets belonging to a low-resolution variant of a key frame.
  - An application can fine tune FEC granularity.
- By tuning the encoder generation of subframes.
  - Trade-off between transform overhead and FEC granularity.

How does it work with redundancy? (cont)

RED

- Primary and redundant data can be transformed data:
  - Transform happens before RED concatenates the data blocks.
  - No change required to RED but potential overhead of the transform.
- Receiver needs to be able to identify that redundant Data and primary are transformed:
  - Using PT1/PT2.
  - APT RTP header extension is used for both PT1/PT2, so if it is transformed data, both must be of same codec.



Thank you

Special thanks to:

The Secretariat, WG Participants & ADs

AVTCORE WG
IETF 110

Virtual Meeting

Thursday, March 11, 2020

04:00 - 06:00 Pacific Time

Session I, Room 1

Mailing list: avtcore@ietf.org
Jabber Room: avtcore@jabber.ietf.org
MeetEcho link: https://gce.conf.meetecho.com/conference/?group=avtcore