Payload Working Group
Internet-Draft
Intended status: Standards Track
Expires: April 30, 2015

J. Uberti
S. Holmer
M. Flodman
Google
J. Lennox
Vidyo

October 27, 2014

RTP Payload Format for VP9 Video
draft-uberti-payload-vp9-00

Abstract

This memo describes an RTP payload format for the VP9 video codec. The payload format has wide applicability, as it supports applications from low bit-rate peer-to-peer usage, to high bit-rate video conferences. It includes provisions for temporal and spatial scalability.

Status of This Memo

This Internet-Draft is submitted in full conformance with the provisions of BCP 78 and BCP 79.

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF). Note that other groups may also distribute working documents as Internet-Drafts. The list of current Internet-Drafts is at http://datatracker.ietf.org/drafts/current/.

Internet-Drafts are draft documents valid for a maximum of six months and may be updated, replaced, or obsoleted by other documents at any time. It is inappropriate to use Internet-Drafts as reference material or to cite them other than as "work in progress."

This Internet-Draft will expire on April 30, 2015.

Table of Contents

1. Introduction 2
2. Conventions, Definitions and Acronyms 2
3. Media Format Description 3
4. Payload Format 4
4.1. RTP Header Usage 4
4.2. VP9 Payload Description 6
4.2.1. Scalability Structure (SS): 8
4.2.2. Scalability Structure Update (SU): 9
4.3. VP9 Payload Header 10
4.4. Frame Fragmentation 10
4.5. Examples of VP9 RTP Stream 10
5. Using VP9 with RPSI and SLI Feedback 10
5.1. RPSI 10
5.2. SLI 11
5.3. Example 11
6. Layer Intra Request 13
7. Payload Format Parameters 14
7.1. Media Type Definition 14
7.2. SDP Parameters 15
7.2.1. Mapping of Media Subtype Parameters to SDP 16
7.2.2. Offer/Answer Considerations 16
8. Security Considerations 16
9. Congestion Control 17
10. IANA Considerations 17
11. References 17
Authors' Addresses 18

1. Introduction

This memo describes an RTP payload specification applicable to the transmission of video streams encoded using the VP9 video codec [I-D.grange-vp9-bitstream]. The format described in this document can be used both in peer-to-peer and video conferencing applications.

TODO: VP9 description. Please see [I-D.grange-vp9-bitstream].

2. Conventions, Definitions and Acronyms

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 [RFC2119].

3. Media Format Description

The VP9 codec can maintain up to eight reference frames, of which up to three can be referenced or updated by any new frame.

VP9 also allows a reference frame to be resampled and used as a reference for another frame of a different resolution. This allows internal resolution changes without requiring the use of keyframes.

These features together enable an encoder to implement various forms of coarse-grained scalability, including temporal, spatial, and quality scalability modes, as well as combinations of these, without the need for explicit spatially scalabile encoding modes.

This payload format specification defines how such scalability modes can be encoded and communicated. In this payload, three separate types of layers are defined: temporal, spatial, and quality.

Temporal layers define different frame rates of video; spatial and quality layers define different, dependent representations of a single picture. Spatial layers allow a picture to be encoded at different resolutions, whereas quality layers allow a picture to be encoded at the same resolution but at different bitrates (and thus with different amounts of coding error).

Layers are designed (and MUST be encoded) such that if any layer, and all higher layers, are removed from the bitstream along any of the three dimensions, the remaining bitstream is still correctly decodable.

For terminology, this document uses the term "frame" to refer to a single encoded VP9 image, and "picture" to refer to all the representations of frames at a single instant in time. A picture thus can consist of multiple frames, encoding different spatial and/or quality layers.

[Editor’s Note: Are separate spatial and quality layers necessary and useful? We could simplify by only defining a single sequence of frames within a picture.

Two modes of describing layer information are possible: "non-flexible mode" and "flexible mode". An encoder can freely switch between the two as appropriate.

In non-flexible mode, an SS message, which defines the layer hierarchy, is sent in the beginning of the stream together with the key frame. Each packet will have a picture id and reference indices, which in conjunction with the SS and the RTP sequence number can be

used to determine if the packet is decodable or not. An SU message can be sent by the sending client, or an MCU, to notify the receiver about what subset of the SS it will actually be receiving.

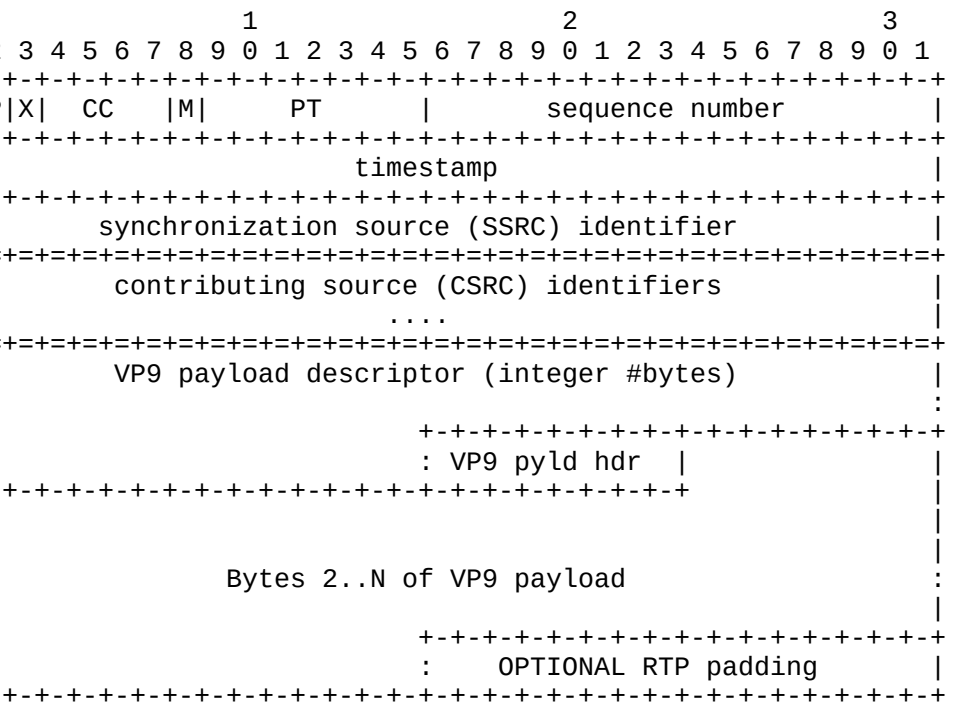
In the flexible mode each packet contains 1-4 reference indices, which identifies all frames referenced by the frame transmitted in the current packet. This enables a receiver to identify if a frame is decodable or not and helps it understand the layer structure so that it can drop packets as it sees fit. Since this is signaled in each packet it makes it possible to have more flexible layer hierarchies and patterns which are changing dynamically.

4. Payload Format

This section describes how the encoded VP9 bitstream is encapsulated in RTP. To handle network losses usage of RTP/AVPF [RFC4585] is RECOMMENDED. All integer fields in the specifications are encoded as unsigned integers in network octet order.

4.1. RTP Header Usage

The general RTP payload format for VP9 is depicted below.



The VP9 payload descriptor and VP9 payload header will be described in the next section. OPTIONAL RTP padding MUST NOT be included unless the P bit is set.

Figure 1

Marker bit (M): MUST be set for the final packet of each encoded frame. This enables a decoder to finish decoding the frame, where it otherwise may need to wait for the next packet to explicitly know that the frame is complete. Note that, if spatial or quality scalability is in use, more frames from the same picture may follow; see the description of the E bit below.

Timestamp: The RTP timestamp indicates the time when the frame was sampled, at a clock rate of 90 kHz. If a picture is encoded with multiple frames, all of the frames of the picture have the same timestamp.

Sequence number: The sequence numbers are monotonically increasing in order of the encoded bitstream.

The remaining RTP header fields are used as specified in [RFC3550].

4.2. VP9 Payload Description

The first octets after the RTP header are the VP9 payload descriptor, with the following structure.

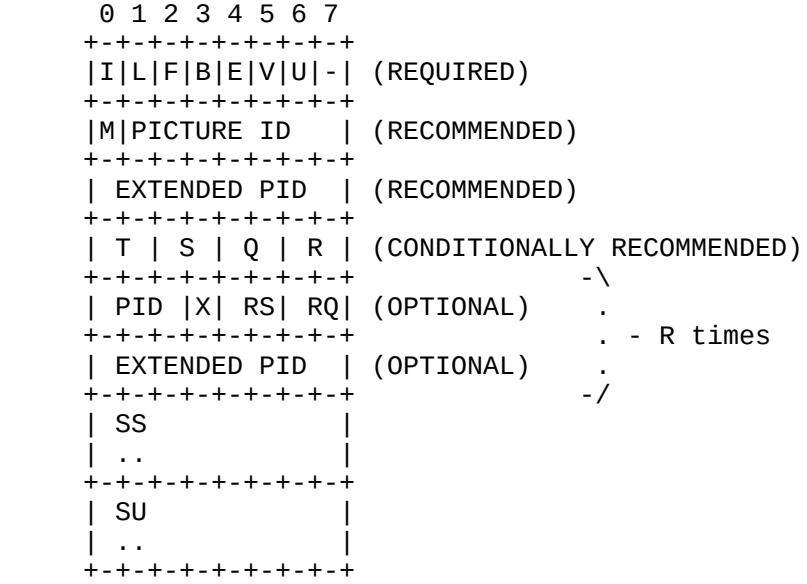


Figure 2

I: PictureID present. When set to one, the OPTIONAL PictureID MUST be present after the mandatory first octet and specified as below. Otherwise, PictureID MUST NOT be present.

L: Layer indices present. When set to one, the octets following the first octet and the extended Picture ID (if present) are as described by "Layer indices" below.

F: Reference indices present. When set to one, the octets following the first octet and the extended Picture ID (if present) are as described by "Reference indices" below. This MUST only be set if L is also 1; if L is 0 then this MUST be set to zero and ignored by receivers.

B: Start of VP9 frame. MUST be set to 1 if the first payload octet of the RTP packet is the beginning of a new VP9 frame, and MUST

NOT be 1 otherwise. Note that this frame might not be the first frame of the picture.

E: End of picture. MUST be set to 1 for the final RTP packet of a VP9 picture, and 0 otherwise. Unless spatial or quality scalability is in use for this picture, this will have the same value as the marker bit in the RTP header.

V: Scalability Structure (SS) present. When set to one, the OPTIONAL Scalability Structure MUST be present in the payload descriptor. Otherwise, the Scalability Structure MUST NOT be present.

U: Scalability Structure Update (SU) present. When set to one, the OPTIONAL Scalability Structure Update MUST be present in the payload descriptor. Otherwise, the Scalability Structure Update MUST NOT be present.

-: Bit reserved for future use. MUST be set to zero and MUST be ignored by the receiver.

After the extension bit field follow the extension data fields that are enabled.

M: The most significant bit of the first octet is an extension flag. The field MUST be present if the I bit is equal to one. If set the PictureID field MUST contain 16 bits else it MUST contain 8 bits including this MSB, see PictureID.

PictureID: 8 or 16 bits including the M bit. This is a running index of the frames. The field MUST be present if the I bit is equal to one. The 7 following bits carry (parts of) the PictureID. If the extension flag is one, the PictureID continues in the next octet forming a 15 bit index, where the 8 bits in the second octet are the least significant bits of the PictureID. If the extension flag is zero, there is no extension, and the PictureID is the 7 remaining bits of the first (and only) octet. The sender may choose 7 or 15 bits index. The PictureID SHOULD start on a random number, and MUST wrap after reaching the maximum ID. The receiver MUST NOT assume that the number of bits in PictureID stay the same through the session.

Layer indices: This byte is optional, but recommended whenever encoding with layers. T, S and Q are 2-bit indices for temporal, spatial, and quality layers, respectively. S and Q start at zero for each picture, and increment consecutively (with Q incrementing before S). These can help MCUs measure bitrates per layer and can help them make a quick decision on whether to relay a packet or not. They can also help receivers determine what layers they are

currently decoding. If "F" is set in the initial octet, R is 2 bits representing the number of reference fields this frame refers to. R MAY be zero, indicating a keyframe. The layer indices field will be followed by R reference indices. If "F" is not set, R MUST be set to zero and ignored by receivers.

Reference indices: These bytes are optional, but recommended when encoding with layers in the flexible mode. They are also recommended in the non-flexible mode when sending frames which are out of sync with the pattern signaled with the SS, for instance when encoding a layer synchronization frame in response to a LIR.

PID: The relative Picture ID referred to by this frame. I.e., PID=3 on a packet containing the frame with Picture ID 112 means that the frame refers back to the frame with picture ID 109. This calculation is done modulo the size of the Picture ID field, i.e. either 7 or 15 bits. For most layer structures a 3-bit relative Picture ID will be enough; however, the X bit can be used to refer to pictures with Picture IDs more than 7 previously.

RS and RQ: The spatial and quality layer IDs of the frame referred to by this frame, in the picture identified by the relative Picture ID.

X: 1 if this layer index has an extended relative Picture ID.

These 1-2 bytes are repeated R times, defined by the two R bits in the layer indices field.

4.2.1. Scalability Structure (SS):

The Scalability Structure data describes the pattern of scalable frames that will be used in a scalable stream. If the VP9 payload header's "V" bit is set, the scalability structure (SS) is present in the position indicated in Figure 2.

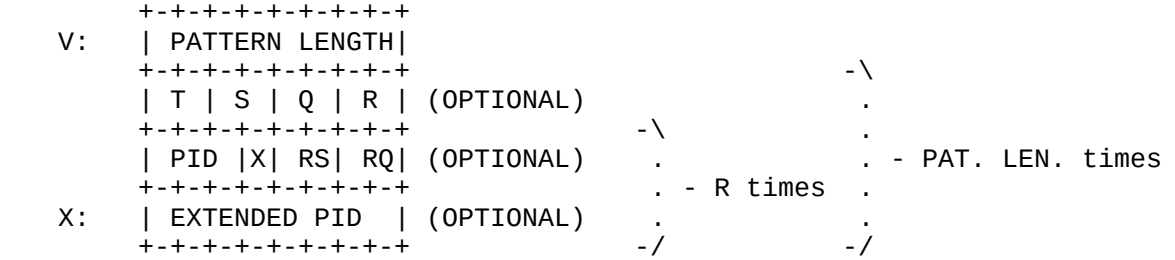


Figure 3

The scalability structure allows the structure of the VP9 stream to be predeclared, rather than indicating it on the fly with every frame as with the layer indices.

Its structure consists of a sequence of frames, encoded as with the layer indices. It begins with PATTERN LENGTH, indicating the number of frames in the pattern; it is then followed by that many instances of data encoded using the same semantics as the layer indices.

TODO: add frame resolution information.

In a scalable stream sent with a fixed pattern, the scalability structure SHOULD be included in the first packet of every keyframe picture, and also in the first packet of the first picture in which the scalability structure changes. If a SS is included in a picture with TID not equal to 0, it MUST also be repeated in the first packet the first frame with a lower TID, until TID equals 0.

If PATTERN LENGTH is 0, it indicates that no fixed scalability information is present going forward in the bitstream. An SS with a PATTERN LENGTH of 0 allows a bitstream to be changed from non-flexible to flexible mode.

4.2.2. Scalability Structure Update (SU):

TODO

4.3. VP9 Payload Header

TODO: need to describe VP9 payload header.

4.4. Frame Fragmentation

VP9 frames are fragmented into packets, in RTP sequence number order, beginning with a packet with the B bit set, and ending with a packet with the RTP marker bit set. There is no mechanism for finer-grained access to parts of a VP9 frame.

4.5. Examples of VP9 RTP Stream

TODO

5. Using VP9 with RPSI and SLI Feedback

The VP9 payload descriptor defined in Section 4.2 above contains an optional PictureID parameter. One use of this parameter is included to enable use of reference picture selection index (RPSI) and slice loss indication (SLI), both defined in [RFC4585].

5.1. RPSI

TODO: Update to indicate which frame within the picture.

The reference picture selection index is a payload-specific feedback message defined within the RTCP-based feedback format. The RPSI message is generated by a receiver and can be used in two ways. Either it can signal a preferred reference picture when a loss has been detected by the decoder -- preferably then a reference that the decoder knows is perfect -- or, it can be used as positive feedback information to acknowledge correct decoding of certain reference pictures. The positive feedback method is useful for VP9 used as unicast. The use of RPSI for VP9 is preferably combined with a special update pattern of the codec’s two special reference frames -- the golden frame and the altref frame -- in which they are updated in an alternating leapfrog fashion. When a receiver has received and correctly decoded a golden or altref frame, and that frame had a PictureID in the payload descriptor, the receiver can acknowledge this simply by sending an RPSI message back to the sender. The message body (i.e., the "native RPSI bit string" in [RFC4585]) is simply the PictureID of the received frame.

5.2. SLI

TODO: Update to indicate which frame within the picture.

The slice loss indication is another payload-specific feedback message defined within the RTCP-based feedback format. The SLI message is generated by the receiver when a loss or corruption is detected in a frame. The format of the SLI message is as follows [RFC4585]:

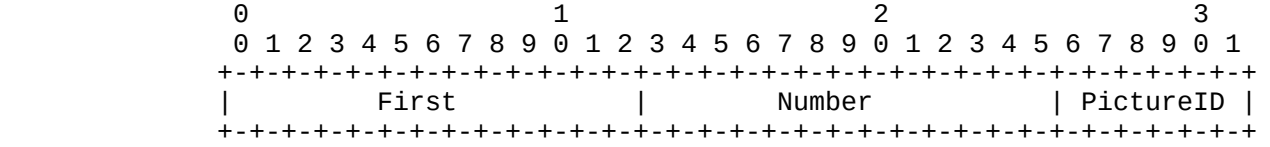


Figure 4

Here, First is the macroblock address (in scan order) of the first lost block and Number is the number of lost blocks. PictureID is the six least significant bits of the codec-specific picture identifier in which the loss or corruption has occurred. For VP9, this codec-specific identifier is naturally the PictureID of the current frame, as read from the payload descriptor. If the payload descriptor of the current frame does not have a PictureID, the receiver MAY send the last received PictureID+1 in the SLI message. The receiver MAY set the First parameter to 0, and the Number parameter to the total number of macroblocks per frame, even though only parts of the frame is corrupted. When the sender receives an SLI message, it can make use of the knowledge from the latest received RPSI message. Knowing that the last golden or altref frame was successfully received, it can encode the next frame with reference to that established reference.

5.3. Example

TODO: this example is copied from the VP8 payload format specification, and has not been updated for VP9. It may be incorrect.

The use of RPSI and SLI is best illustrated in an example. In this example, the encoder may not update the altref frame until the last sent golden frame has been acknowledged with an RPSI message. If an update is not received within some time, a new golden frame update is sent instead. Once the new golden frame is established and acknowledged, the same rule applies when updating the altref frame.

Event	Sender	Receiver	Established reference
1000	Send golden frame PictureID = 0		
		Receive and decode golden frame	
1001		Send RPSI(0)	
1002	Receive RPSI(0)		golden
...	(sending regular frames)		
1100	Send altref frame PictureID = 100		
		Altref corrupted or lost	golden
1101		Send SLI(100)	golden
1102	Receive SLI(100)		
1103	Send frame with reference to golden		
		Receive and decode frame (decoder state restored)	golden
...	(sending regular frames)		
1200	Send altref frame PictureID = 200		
		Receive and decode altref frame	golden
1201		Send RPSI(200)	
1202	Receive RPSI(200)		altref
...	(sending regular		

	frames)			
1300	Send golden frame PictureID = 300			
		Receive and decode golden frame	altref	
1301		Send RPSI(300)	altref	
1302	RPSI lost			
1400	Send golden frame PictureID = 400			
		Receive and decode golden frame	altref	
1401		Send RPSI(400)		
1402	Receive RPSI(400)		golden	

Table 1: Example signaling between sender and receiver

Note that the scheme is robust to loss of the feedback messages. If the RPSI is lost, the sender will try to update the golden (or altref) again after a while, without releasing the established reference. Also, if an SLI is lost, the receiver can keep sending SLI messages at any interval allowed by the RTCP sending timing restrictions as specified in [RFC4585], as long as the picture is corrupted.

6. Layer Intra Request

Editor's Note: The message described in this section is applicable to other codecs beyond just VP9. In the future it will be likely be split out into another document.

TODO: details of how this is encoded in RTCP.

A synchronization frame can be requested by sending a LIR, which is an RTCP feedback message asking the encoder to encode a frame which makes it possible to upgrade to a higher layer. The LIR message contains two tuples, {T1,S1,Q1} and {T2,S2,Q2}, where the first tuple is the currently highest layer the decoder can decode, while the second tuple is the layer the decoder wants to upgrade to.

Identification of an upgrade frame can be derived from the reference IDs of each frame by backtracking the dependency chain until reaching a point where only decodable frames are being referenced. Therefore it's recommended both for both the flexible and the non-flexible mode that, when upgrade frames are being encoded in response to a LIR, those packets should contain layer indices and the reference fields so that the decoder or an MCU can make this derivation.

Example:

LIR {1,1,0}, {1,2,1} is sent by an MCU when it is currently relaying {1,1,0} to a receiver and which wants to upgrade to {1,2,1}. In response the encoder should encode the next frames in layers {1,1,1} and {1,2,1} by only referring to frames in {1,1,0}, {1,0,0} or {0,0,0}.

In the non-flexible mode, periodic upgrade frames can be defined by the layer structure of the SS, thus periodic upgrade frames can be automatically identified by the picture ID.

7. Payload Format Parameters

This payload format has two required parameters.

7.1. Media Type Definition

This registration is done using the template defined in [RFC6838] and following [RFC4855].

Type name: video

Subtype name: VP9

Required parameters:
These parameters MUST be used to signal the capabilities of a receiver implementation. These parameters MUST NOT be used for any other purpose.

max-fr: The value of max-fr is an integer indicating the maximum frame rate in units of frames per second that the decoder is capable of decoding.

max-fs: The value of max-fs is an integer indicating the maximum frame size in units of macroblocks that the decoder is capable of decoding.

The decoder is capable of decoding this frame size as long as the width and height of the frame in macroblocks are less than

int(sqrt(max-fs * 8)) - for instance, a max-fs of 1200 (capable of supporting 640x480 resolution) will support widths and heights up to 1552 pixels (97 macroblocks).

Optional parameters: none

Encoding considerations:
This media type is framed in RTP and contains binary data; see Section 4.8 of [RFC6838].

Security considerations: See Section 8 of RFC xxxx.
[RFC Editor: Upon publication as an RFC, please replace "XXXX" with the number assigned to this document and remove this note.]

Interoperability considerations: None.

Published specification: VP9 bitstream format
[I-D.grange-vp9-bitstream] and RFC XXXX.
[RFC Editor: Upon publication as an RFC, please replace "XXXX" with the number assigned to this document and remove this note.]

Applications which use this media type:
For example: Video over IP, video conferencing.

Additional information: None.

Person & email address to contact for further information:
TODO [Pick a contact]

Intended usage: COMMON

Restrictions on usage:
This media type depends on RTP framing, and hence is only defined for transfer via RTP [RFC3550].

Author: TODO [Pick a contact]

Change controller:
IETF Payload Working Group delegated from the IESG.

7.2. SDP Parameters

The receiver MUST ignore any fmtp parameter unspecified in this memo.

7.2.1. Mapping of Media Subtype Parameters to SDP

The media type video/VP9 string is mapped to fields in the Session Description Protocol (SDP) [RFC4566] as follows:

- The media name in the "m=" line of SDP MUST be video.
- The encoding name in the "a=rtpmap" line of SDP MUST be VP9 (the media subtype).
- The clock rate in the "a=rtpmap" line MUST be 90000.
- The parameters "max-fs", and "max-fr", MUST be included in the "a=fmtp" line of SDP. These parameters are expressed as a media subtype string, in the form of a semicolon separated list of parameter=value pairs.

7.2.1.1. Example

An example of media representation in SDP is as follows:

```
m=video 49170 RTP/AVPF 98
a=rtpmap:98 VP9/90000
a=fmtp:98 max-fr=30; max-fs=3600;
```

7.2.2. Offer/Answer Considerations

TODO: Update this for VP9

8. Security Considerations

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. The main security considerations for the RTP packet carrying the RTP payload format defined within this memo are confidentiality, integrity and source authenticity. Confidentiality is achieved by encryption of the RTP payload. Integrity of the RTP packets through suitable cryptographic integrity protection mechanism. Cryptographic system may also allow the authentication of the source of the payload. A suitable security mechanism for this RTP payload format should provide confidentiality, integrity protection and at least source authentication capable of determining if an RTP packet is from a member of the RTP session or not. Note that the appropriate mechanism to provide security to RTP and payloads following this memo may vary. It is dependent on the application, the transport, and the signaling protocol employed. Therefore a single mechanism is not sufficient, although if suitable the usage of SRTP [RFC3711] is

recommended. This RTP payload format and its media decoder do not exhibit any significant non-uniformity in the receiver-side computational complexity for packet processing, and thus are unlikely to pose a denial-of-service threat due to the receipt of pathological data. Nor does the RTP payload format contain any active content.

9. Congestion Control

Congestion control for RTP SHALL be used in accordance with RFC 3550 [RFC3550], and with any applicable RTP profile; e.g., RFC 3551 [RFC3551]. The congestion control mechanism can, in a real-time encoding scenario, adapt the transmission rate by instructing the encoder to encode at a certain target rate. Media aware network elements MAY use the information in the VP9 payload descriptor in Section 4.2 to identify non-reference frames and discard them in order to reduce network congestion. Note that discarding of non-reference frames cannot be done if the stream is encrypted (because the non-reference marker is encrypted).

10. IANA Considerations

The IANA is requested to register the following values:
- Media type registration as described in Section 7.1.

11. References

[I-D.grange-vp9-bitstream]
Grange, A. and H. Alvestrand, "A VP9 Bitstream Overview", draft-grange-vp9-bitstream-00 (work in progress), February 2013.

[RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, March 1997.

[RFC3550] Schulzrinne, H., Casner, S., Frederick, R., and V. Jacobson, "RTP: A Transport Protocol for Real-Time Applications", STD 64, RFC 3550, July 2003.

[RFC3551] Schulzrinne, H. and S. Casner, "RTP Profile for Audio and Video Conferences with Minimal Control", STD 65, RFC 3551, July 2003.

[RFC3711] Baugher, M., McGrew, D., Naslund, M., Carrara, E., and K. Norrman, "The Secure Real-time Transport Protocol (SRTP)", RFC 3711, March 2004.

[RFC4566] Handley, M., Jacobson, V., and C. Perkins, "SDP: Session Description Protocol", RFC 4566, July 2006.

[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, July 2006.

[RFC4855] Casner, S., "Media Type Registration of RTP Payload Formats", RFC 4855, February 2007.

[RFC6838] Freed, N., Klensin, J., and T. Hansen, "Media Type Specifications and Registration Procedures", BCP 13, RFC 6838, January 2013.

Authors' Addresses

Justin Uberti
Google, Inc.
747 6th Street South
Kirkland, WA 98033
USA

Email: justin@uberti.name

Stefan Holmer
Google, Inc.
Kungsbron 2
Stockholm 111 22
Sweden

Magnus Flodman
Google, Inc.
Kungsbron 2
Stockholm 111 22
Sweden

Jonathan Lennox
Vidyo, Inc.
433 Hackensack Avenue
Seventh Floor
Hackensack, NJ 07601
US

Email: jonathan@vidyo.com