IBC2025

TRUSTWORTHY AUTHENTICATION MECHANISM FOR VIDEO CODING AT AN ELEMENTARY STREAM LEVEL

J. Pfaff, K. Sühring, T. Hinz, Y. Sanchez, H. Schwarz, D. Marpe, T. Wiegand

Fraunhofer Institute for Telecommunications, Heinrich Hertz Institute, Berlin, Germany

ABSTRACT

With recent advances in AI technology, it is becoming increasingly hard to identify whether video content is authentic or fake. As a consequence, users are losing trust in the information that they consume, and media outlets are challenged to fight modified or completely faked versions of their content. To overcome this problem, the Joint Video Experts Team JVET has recently developed a mechanism for trustworthy signing and authentication of coded video data at an elementary stream level. This method is designed so that it is compatible with key functionalities of the underlying video codecs like random access or scalable coding. Moreover, it can be applied within real time encoding. Thus, it is applicable in many important use cases that are not covered by existing technologies like C2PA. The method also supports a joint authentication of video and other media data such as coded audio data. In this paper, the key features and main technical aspects of this trustworthy authentication mechanism are described.

INTRODUCTION

Multimedia content such as text, images, audio, and video, is integral to modern life. The rapid evolution of artificial intelligence technologies, for example in large language models or image and video generation, has significantly transformed multimedia's generation and production, enhancing productivity and creativity and enabling new formats and user experiences. However, at the same time, this progress has also sparked the creation of sophisticated deep fakes. Such fakes appear in particular in social media, where everybody can easily share information. They often blur the lines between real and fake content and therefore raise significant concerns, for example regarding cybersecurity and copyrights. Since they are increasingly hard to detect, deep fakes are exploited for fraud and deception, posing threats to individual and national security by mimicking identities to steal credentials. Although governments are starting to require marking of AI content by law, this affects only lawful content generators. Malicious players may not label their content at all or remove such marking. This can undermine digital platform trust and safety and have an impact on societies worldwide.

Today, the by far largest portion of multimedia traffic is generated by video data. The reliable and interoperable exchange of video content is supported globally by joint standards of ISO/IEC JTC1/SC 29 and ITU-T. Here, it has been observed that for the video coding standards H.266/VVC, H.265/HEVC or H.264/AVC, no method for trustworthy verification exist that is applicable in a self-contained way at the elementary stream level.

As a consequence, mechanisms for trustworthy authentication and verification of video content have recently been developed by the Joint Video Experts Team JVET for inclusion into these video coding standards [1, 2]. This has been realized by three new supplemental enhancement information (SEI) messages which enable to attach cryptographic signatures to flexible chunks of data of a video stream at the network abstraction layer (NAL) unit level.

In the present paper, the basic functionalities of these mechanisms and the associated SEI messages will be described. The paper is structured as follows. First, the general principle of digitally signing is reviewed. Then, the specific functionalities of the solution developed by JVET are described. Next, the corresponding SEI messages are described. In the end, an example of a bitstream structure equipped with the presented authentication mechanism is presented.

GENERAL PRINCIPLE OF DIGITALLY SIGNING DATA STREAMS

The proposed technical solution aims to allow the verification of video content's trustworthiness, enabling users to confirm the authenticity of content by its creators, such as governments, companies, or news organizations. It is based on the digital signing of data streams. In general terms, this mechanism works as follows.

The content creator, i.e. the encoder, uses a private key to sign the content, while the recipient, i.e. the decoder, uses a corresponding public key to verify the authenticity. The public key, necessary for verification, is not derived directly from the data stream but is obtained through a trusted, independent method, such as a third-party trust center. This process may utilize ITU Recommendation X.509 for the secure retrieval of digital certificates that validate the encoder's identity. Additionally, the encoder and decoder agree on a cryptographic hash function to compute a unique digital signature for a specified byte range within the data stream. Verification occurs when the decoder successfully matches the digital signature with the computed hash value using the public key, establishing the content's credibility.

Thus, the following steps are used to enable digital signing at the encoder and trustworthy authentication at the decoder.

- The encoder, is in possession of a private (and a public) key for a fixed signature algorithm.
- The decoder only possesses the public, but not the private key. The data stream itself may include a pointer to the public key, however, the public key itself may not be obtained solely from the data stream but must be obtained by invoking a trusted and independent method. For example, the data stream can contain information that identifies the encoder as a certain entity.
- Given this information, the decoder can retrieve the digital certificate corresponding to this entity (a public key and all other parameters needed for verification) from a third-party trust center.
- The encoder fixes a cryptographic hash function whose implementation is also supported by the decoder. Then, a unique range of bytes is determined from the data stream for which the cryptographic hash value is to be computed by the decoder using the given hash function.
- Finally, a digital signature that can be regarded as the claimed digital signature of the computed hash value for the given range of bytes is transmitted in the data stream. For checking trustworthiness, the decoder processes this digital signature using the given public key. Then, if the result of this process is found to be the digital signature of the

computed hash-value of the byte range, the decoder can regard this byte range as information that trustworthily belongs to the entity associated to the given public key while in the opposite case, it should regard it as a fake.

The overall process described here is depicted in Figure 1.

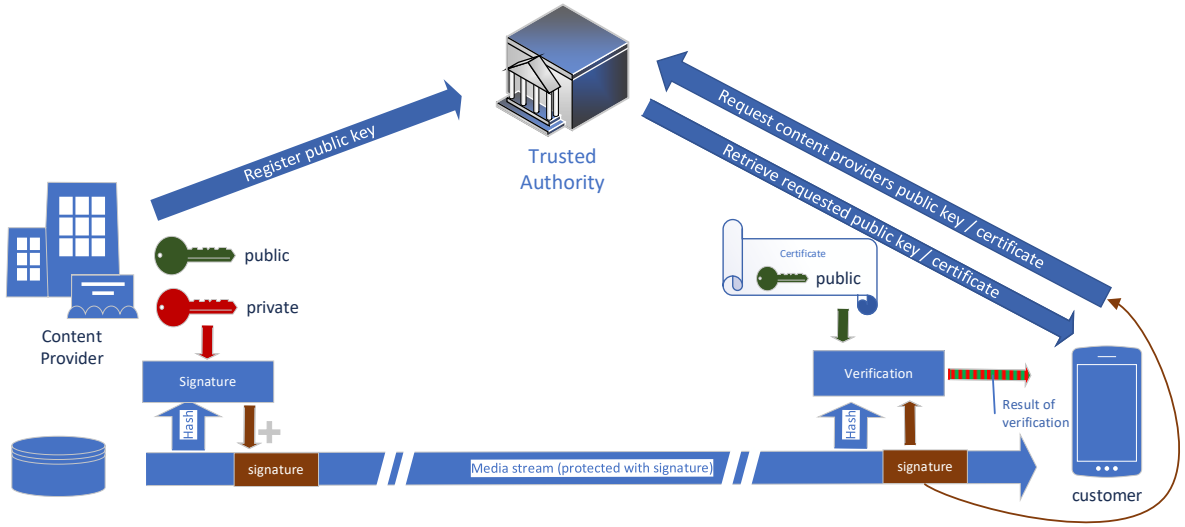

Figure 1: System overview, signature protection for media streams

KEY FUNCTIONALITIES OF THE DESIGNED AUTHENTICATION MECHANISM

Building on the abovementioned principle of digitally signing data streams, JVET developed a technical solution for trustworthy authentication of video data coded using the standards H.266/VVC, H.265/HEVC or H.264/AVC. This solution has several key functionalities specific to various use cases and application scenarios of these standards which shall now be described in more detail.

Random Access capabilities and temporal consistency for authentication

To allow an accessing of the coded data at flexible time instances, typical video data streams are organized into random access segments. At each random access segment, a user can start the decoding process without needing to access the data of previous random access segments. For this reason, it is desirable to allow random access also for the authentication mechanism of coded video data.

However, it is observed that a digital signing mechanism that treats each random access segment of a coded video sequence independent from any other random access segment generates the risk of malicious manipulations: Alterations applied to a data stream via rearrangements or insertions of individually signed segments would not be detectable in the authentication process.

For this reason, in order to guarantee that decoders can verify the continuity between different temporal segments of the coded video data and to prevent attacks by removal, insertion or shuffling data chunks in the protected data stream, in the presented solution, the hash value of a preceding temporal segment is also incorporated in the digital signature of a current temporal segment. Consequently, a joint digital signature of both a current and its previous temporal segment is constructed. This enables users to verify temporal consistency of a current segment with the previous segment. Since the digital signature of the current segment is a joint signature, only entities which are in possession of the private key are able

to combine multiple temporal segments that are verified as trustworthy by the proposed method. The principal of temporal consistency is illustrated in Figure 2 below.

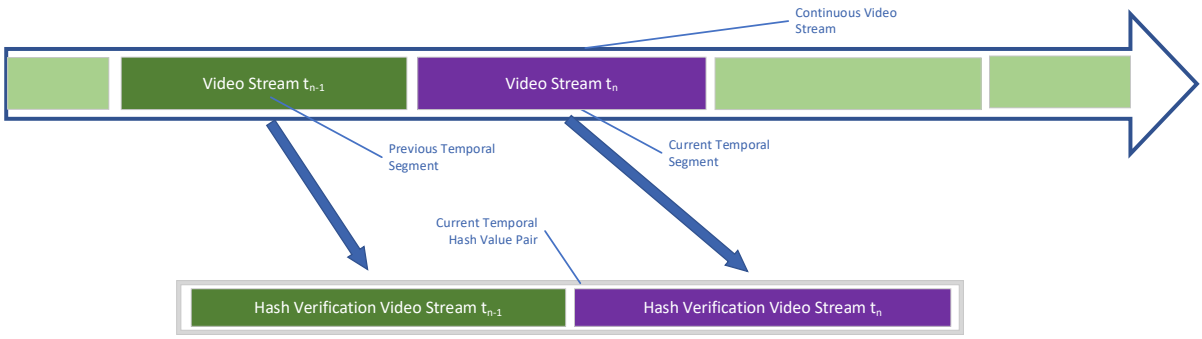

Figure 2: Temporal hash value pair protected by joint signature

It is pointed out that a decoder which decodes the data stream starting at a current random access segment at time instance t_n cannot immediately verify the authenticity of Video Stream t_n since it does not have access to the hash value of Video Stream t_{n-1}. However, as soon as such a decoder also decodes Video Stream t_{n+1}, a successful verification of this stream also implies authenticity of Video Stream t_n, since the hash value of Video Stream t_n is invoked in the verification process for Video Stream t_{n+1}. As a consequence, trustworthy random access is supported for the presented authenticate mechanism.

Besides that, the presented authentication mechanism also supports to add an information to the bitstream which indicates whether a random access segment is an intended first or last segment. This allows detecting removal of information at the beginning and at the end of the bitstream. It also allows the introduction of splicing points, at which different (signed or unsigned) content may be inserted into the bitstream, e.g. when switching to advertising. Therefore, the end of the signed content is indicated before the splicing point and signing is restarted after the spliced content. Such points can be easily detected in pre-encoded bitstreams, e.g. for on-demand ad-insertion in streaming. An example is shown in Figure 3. To ensure that the start- and end-indication is not maliciously modified, it is included into the signed data packet.

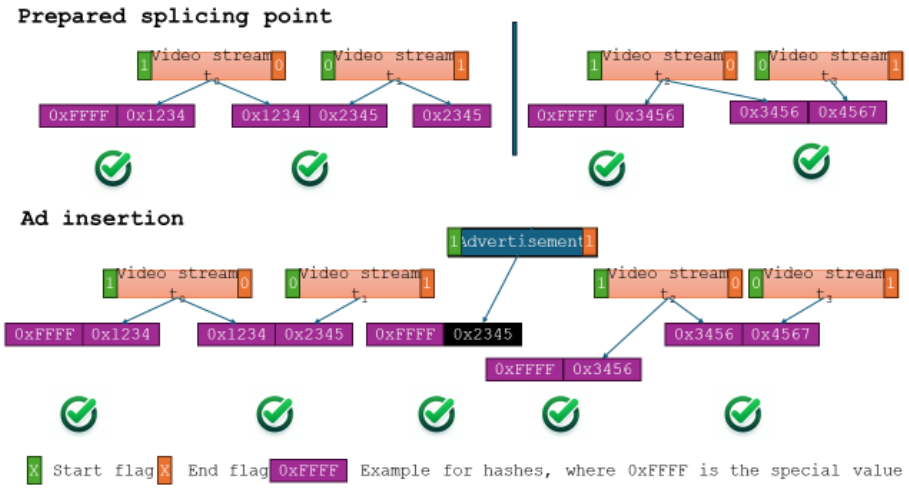

Figure 3: Example for a splicing point

Support for authentication in scalable and general multi-layers scenarios

In many application scenarios, only parts of a single video bitstream might be used by a user for display or further processing. Important examples for this are temporal scalability, where some temporal layers can be dropped, or multi-view coding, where pictures can be displayed for a specific view only.

Consequently, the developed authentication mechanisms support a flexible concept of substreams for trustworthy authentication, where the substreams are related by a flexible ordering concept. Here, each substream can be verified independently from substreams which, in a given ordering, are either of higher order or are unrelated to it with respect to the ordering. However, the verification of a substream of higher order is only possible in combination with the verification of substreams of lower order. Similar to the abovementioned temporally consistent authentication, this prevents malicious manipulations that arise by unintended combinations of individually verified substreams.

Enabling joint trustworthy authentication with other media data

The developed trustworthy authentication mechanism is applicable at the elementary video stream level and can thus be realized in an independent and self-contained way within the respective video coding standards. This is crucial for applications like independent streaming. On the other hand, to prevent forgeries via fake combinations of individually verified media assets, it is highly desirable to allow joint trustworthy authentication mechanisms of video data with other media data, most importantly audio data. This means that, whenever coded video and audio data are to be displayed as a combination, a user should be able to trustworthily verify that this combination was indeed intended by the content generator, while, at the same time, it should still be possible to only verify the individual parts of the combination.

In order to meet this requirement, a dedicated unique identifier can be added to each chunk of signed video data within the context of the developed SEI messages. This unique identifier can be signalled in the bitstream. When present, it is also part of the data covered by the digital signature for the respective chunk of video data. The key point now is that a similar mechanism is also envisioned for other multimedia standards, e.g. for audio coding. Thus, whenever a chunk of coded audio data is intended to be coupled to the respective chunk of coded video data and when this combination is intended to be trustworthily verifiable, the same unique identifier should also be part of the respective chunk of coded audio data and should be protected by the authentication mechanism for these data within the used audio coding standard.

In fact, exactly this proposed concept for a combined authentication mechanism has been picked up by Working Group 6 of MPEG which is also in the process of developing mechanisms for trustworthy authentication of audio data [4, 7]. Consequently, a combined authentication mechanism for video and audio data coded with the widely deployed MPEG standards will be available soon.

In Figure 4, an example of the joint authentication mechanism for video and audio data is displayed. Three elementary streams are given, two audio streams and one video stream. Moreover, a unique identifier UUID is part of each stream. Each stream can be verified in a self-contained way at the elementary-stream level, where the UUID is part of the data protected by the involved authentication mechanisms. However, on top of that, users are also enabled to verify the coherence of some of the different streams, for example the video

stream and one audio stream. To do so, users simply need to check that the values of the UUID present in the respective elementary streams of interest coincide.

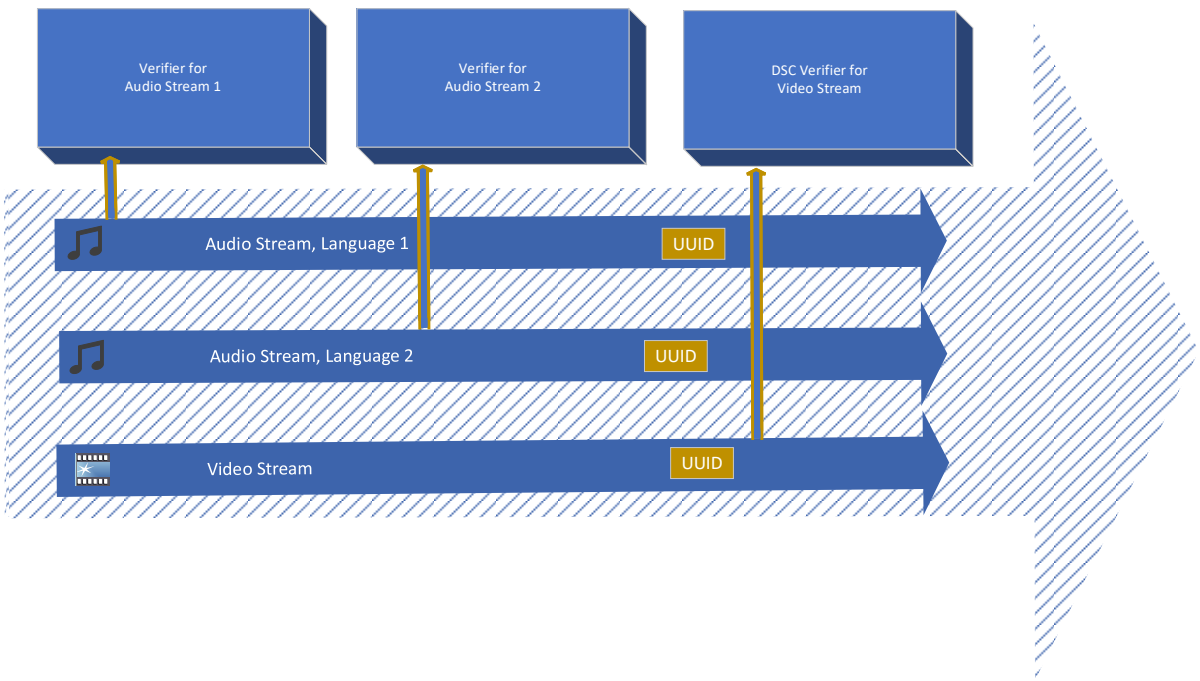

Figure 4: Two elementary audio streams in different languages and one elementary video stream.

DIGITALLY SIGNED CONTENT (DSC) SEI MESSAGES

The above mechanism for trustworthy authentication of coded video data has been achieved by developing three new supplemental enhancement information (SEI) messages for being added to the video coding standards:

- Digitally signed content initialization SEI message
- Digitally signed content selection SEI message
- Digitally signed content verification SEI message

This section summarizes the structure and the content of these messages. More details can be found in the references [1, 2, 3, 5, 6, 8].

Digitally signed content initialization (DSCI) SEI message

The DSCI SEI message indicates the beginning of signed content. It defines the number of used substreams, the dependencies between substreams, the used hash method and a reference to the content creator's public key certificate.

After the DSCI SEI message, all video packets shall be assigned to a substream until a signature is received for each of the substreams.

Digitally signed content selection (DSCS) SEI message

The DSCS SEI message indicates, which substream a picture belongs to.

Different substreams can for instance be used for temporal scalability or multi-view coding. Substreams can be removed while keeping the ability to verify the signature of the remaining bitstream, as long as all dependent substreams are also removed.

If no DSCS SEI message is present, a default association to substreams is used. This can save bits in case that there is either only one substream, or if the substream assignment follows the layer structure of the bitstream.

Digitally signed content verification (DSVC) SEI message

The DSCV SEI message is transmitted for each of the substreams. It contains the actual signature of the substream.

As previously mentioned, the signature spans over the hash of all video data packets of the current substream, as well as the hashes of the substreams, that the current substream depends on and – if available – the hash of the previous substream segment with the same ID.

As additional data, the content UUID, the hash method and start and end flags are included into the signature, so that these data cannot be modified.

Note that there can be multiple signatures within a bitstream, e.g. for supporting different hash methods or from different content creators. In that case, all DSCI, DSCS and DSCV messages that belong to the same signature refer to the same DSC ID value.

EXAMPLE OF A BITSTREAM STRUCTURE USING THE DIGITALLY CONTENT SEI MESSAGES

An example of a bitstream structure that involves the presented authentication mechanism is displayed in Figure 5 below. Here, the case of a bitstream compliant with the H.265/HEVC or the H.266/VVC standard is outlined. For the case of a H.264/AVC compliant bitstream, the overall picture essentially remains the same with the only change that all involved suffix SEI messages would be positioned so that they become prefix SEI messages. The reason for this difference is that H.264/AVC only supports prefix SEI messages, while for H.265/HEVC and H.266/VVC, both prefix and suffix SEI messages are supported.

In the example of Figure 5, the NAL units of the data stream which are digitally signed are grouped into two substreams, called Verification Substream 0 and Verification Substream 1. Here, Substream 0 can be trustworthily verified on its own, while verification of Substream 1 invokes data from Substream 0. This could for example correspond to a case of scalable coding, where Substream 0 is associated to data from a base layer, while Substream 1 is associated to data from an enhancement layer.

The substreams are temporarily interleaved but can be partitioned into consecutive chunks of coded video data, denoted Chunk 0, Chunk 1, Chunk 2 and Chunk 3. Here, Chunk 0 and Chunk 2 belong to Substream 0, while Chunk 1 and Chunk 3 belong to Substream 1. At the end of each chunk, verification of its data can be performed.

The authentication process is initiated with the DSC Init SEI message. The assignment of NAL units from Chunk 0 and Chunk 2 to the default Substream 0 is implicit, while the assignment of NAL units from Chunk 1 and Chunk 3 to Substream 1 is determined by the DSC Select SEI messages. Verification of Substream 0 can be conducted with DSC Verify SEI message A for Chunk 0 or with DSC Verify SEI message C for Chunk 2, where the latter verification process guarantees temporal consistency between Chunk 0 and Chunk 2. Verification of Substream 1 can be conducted with DSC Verify SEI message B for Chunk 1 or with DSC Verify SEI message D for Chunk 3, both guaranteeing consistency between Substream 1 and Substream 0.

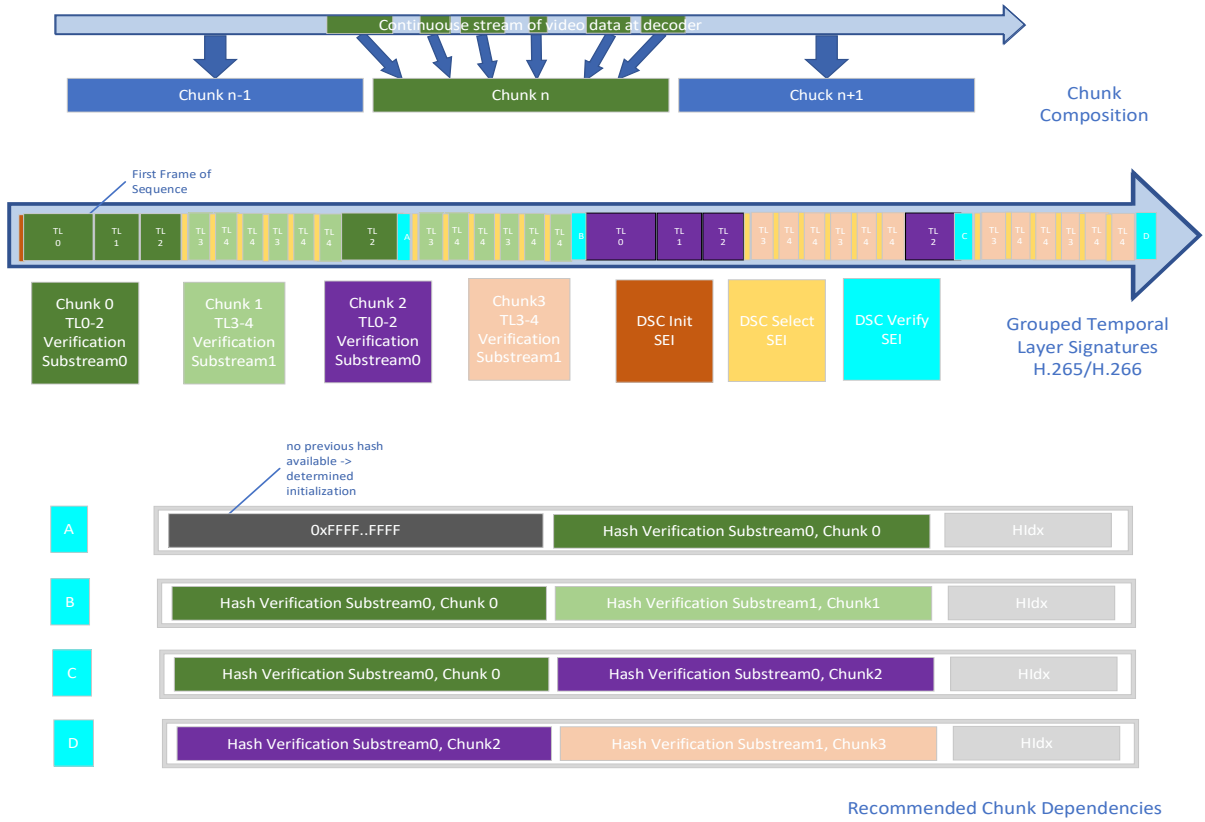

Figure 5: Example of a bit-stream structure invoking the Digitally Signed Content SEI messages

CONCLUSIONS

To meet the increasing challenges of maliciously manipulated content generated with deep fakes, JVET has recently equipped its widely deployed video coding standards with mechanisms for trustworthy content authentication. Here, signing and authentication can be performed at the elementary stream level and can be applied within important use cases such as real time encoding, random access or scalable coding. Also, trustworthy authentication of the combination of video and audio data is supported.

REFERENCES

1. B. Bross et al., Support of additional SEI messages in AVC (Draft 2), Document JVET-AL1017, 38th meeting of JVET (online) 2025

2. J. Boyce et al., Additional SEI messages for VSEI version 4 (Draft 6), Document JVET-AL2005, 38th meeting of JVET (online) 2025

3. Sühring, K. et al., AHG9: Digitally Signed Content Authentication SEI, Document JEVT-AI10127, 35th meeting of JVET, Sapporo, Japan, 2024

4. Fersch, C. et al., Thoughts on Media Authentication for MPEG, Document m69428, 148th meeting of MPEG, Kemer, Turkey, 2024

5. Boyce, J. et al., AHG9: Multilayer digitally signed content authentication SEI messages, Document JVET-AK0287, 37th meeting of JVET, Geneva, Switzerland, 2025

6. Mc Carthy, J., et al., AHG9: On digitally signed content SEI messages, Document JVET-AK0206, 37th meeting of JVET, Geneva, Switzerland, 2025

7. Fersch, C. et al., Media Authenticity in MPEG Audio, Document m71351, 149th meeting of MPEG, Geneva, Switzerland, 2025

8. Sühring, K. et al, On digitally signed content SEI messages, Document JVET-AL0222, 38th meeting of JVET (online), 2025

9. Coalition for Content Provenance and Authenticity, C2PA Technical Specification, https://c2pa.org/specifications/specifications/1.0/specs/_attachments/C2PA_Specification.pdf