ETSI TS 104 002 V1.1.1 (2023-08)

Publicly Available Specification (PAS);
DASH-IF Forensic A/B Watermarking
An interoperable watermarking integration schema

DASH Industry Forum EBU

CAUTION

The present document has been submitted to ETSI as a PAS produced by DASH-IF and approved by the Joint Technical Committee (JTC) Broadcast of the European Broadcasting Union (EBU), Comité Européen de Normalisation ELECtrotechnique (CENELEC) and the European Telecommunications Standards Institute (ETSI).

DASH-IF is owner of the copyright of the document DASH-IF CPIX and/or had all relevant rights and had assigned said rights to ETSI on an "as is basis". Consequently, to the fullest extent permitted by law, ETSI disclaims all warranties whether express, implied, statutory or otherwise including but not limited to merchantability, non-infringement of any intellectual property rights of third parties. No warranty is given about the accuracy and the completeness of the content of the present document.

Reference

DTS/JTC-118

Keywords

broadband, CDN, DASH, DRM, internet, PAS

ETSI

650 Route des Lucioles
F-06921 Sophia Antipolis Cedex - FRANCE

Tel.: +33 4 92 94 42 00 Fax: +33 4 93 65 47 16

Siret N° 348 623 562 00017 - APE 7112B
Association à but non lucratif enregistrée à la
Sous-Préfecture de Grasse (06) N° w061004871

Contents

Intellectual Property Rights

Essential patents

IPRs essential or potentially essential to normative deliverables may have been declared to ETSI. The declarations pertaining to these essential IPRs, if any, are publicly available for **ETSI members and non-members**, and can be found in ETSI SR 000 314: *"Intellectual Property Rights (IPRs); Essential, or potentially Essential, IPRs notified to ETSI in respect of ETSI standards"*, which is available from the ETSI Secretariat. Latest updates are available on the ETSI Web server (https://ipr.etsi.org/).

Pursuant to the ETSI Directives including the ETSI IPR Policy, no investigation regarding the essentiality of IPRs, including IPR searches, has been carried out by ETSI. No guarantee can be given as to the existence of other IPRs not referenced in ETSI SR 000 314 (or the updates on the ETSI Web server) which are, or may be, or may become, essential to the present document.

Trademarks

The present document may include trademarks and/or tradenames which are asserted and/or registered by their owners. ETSI claims no ownership of these except for any which are indicated as being the property of ETSI, and conveys no right to use or reproduce any trademark and/or tradename. Mention of those trademarks in the present document does not constitute an endorsement by ETSI of products, services or organizations associated with those trademarks.

DECT™, **PLUGTESTS™**, **UMTS™** and the ETSI logo are trademarks of ETSI registered for the benefit of its Members. **3GPP™** and **LTE™** are trademarks of ETSI registered for the benefit of its Members and of the 3GPP Organizational Partners. **oneM2M™** logo is a trademark of ETSI registered for the benefit of its Members and of the oneM2M Partners. **GSM**® and the GSM logo are trademarks registered and owned by the GSM Association.

Foreword

This Technical Specification (TS) has been produced by Joint Technical Committee (JTC) Broadcast of the European Broadcasting Union (EBU), Comité Européen de Normalisation ELECtrotechnique (CENELEC) and the European Telecommunications Standards Institute (ETSI).

The present document had initially been prepared by DASH-IF (http://dashif.org) and was sent to ETSI under the PAS agreement.

Comments on the present document may be provided at https://github.com/Dash-Industry-Forum/Watermarking/issues.

NOTE: The EBU/ETSI JTC Broadcast was established in 1990 to co-ordinate the drafting of standards in the specific field of broadcasting and related fields. Since 1995 the JTC Broadcast became a tripartite body by including in the Memorandum of Understanding also CENELEC, which is responsible for the standardization of radio and television receivers. The EBU is a professional association of broadcasting organizations whose work includes the co-ordination of its members' activities in the technical, legal, programme-making and programme-exchange domains. The EBU has active members in about 60 countries in the European broadcasting area; its headquarters is in Geneva.

European Broadcasting Union
CH-1218 GRAND SACONNEX (Geneva)
Switzerland
Tel: +41 22 717 21 11
Fax: +41 22 717 24 81

Modal verbs terminology

In the present document "**shall**", "**shall not**", "**should**", "**should not**", "**may**", "**need not**", "**will**", "**will not**", "**can**" and "**cannot**" are to be interpreted as described in clause 3.2 of the ETSI Drafting Rules (Verbal forms for the expression of provisions).

"**must**" and "**must not**" are **NOT** allowed in ETSI deliverables except when used in direct citation.

Executive summary

The present document describes proposed architecture and API for supporting forensic watermarking for Over The Top (OTT) on content that is delivered in an Adaptive Bit Rate (ABR) format. To the possible extend, the proposed solutions do not make assumptions on the ABR technology that is being used, it can be for example, DASH or HLS.

While digital watermarking can be used for different use cases, the present document focuses on forensic use cases. In this context, it is used to define the origin of content leakage. the watermarking technology modifies media content in a robust and invisible way in order to encode a unique identifier, e.g. a unique session ID. The embedded watermark provides means to identify where the media content, that has been redistributed without authorization, is coming from. In other words, the watermark is used to forensically trace the origin of content leakage.

1 Scope

The present document specifies DASH-IF Forensic A/B Watermarking.

2 References

2.1 Normative references

References are either specific (identified by date of publication and/or edition number or version number) or non-specific. For specific references, only the cited version applies. For non-specific references, the latest version of the referenced document (including any amendments) applies.

Referenced documents which are not found to be publicly available in the expected location might be found at https://docbox.etsi.org/Reference.

NOTE: While any hyperlinks included in this clause were valid at the time of publication, ETSI cannot guarantee their long-term validity.

The following referenced documents are necessary for the application of the present document.

[1] ISO/IEC 23009-1:2022: "Information technology -- Dynamic adaptive streaming over HTTP (DASH) -- Part 1: Media presentation description and segment formats".

[2] ISO/IEC 13818-1:2019: "Information technology -- Generic coding of moving pictures and associated audio information -- Part 1: Systems".

[3] IETF Internet Draft draft-pantos-hls-rfc8216bis-12: "HTTP Live Streaming 2nd Edition", R. Pantos.

[4] IETF RFC 8949: "Concise Binary Object Representation (CBOR)", C. Bormann, P. Hoffman, December 2020.

[5] IETF RFC 8610: Concise Data Definition Language (CDDL): A Notational Convention to Express Concise Binary Object Representation (CBOR) and JSON Data Structures", H. Birkholz, C. Vigano, C. Bormann, June 2019.".

[6] IETF RFC 8392: "CBOR Web Token (CWT)", M. Jones, E. Wahlstroem, S. Erdtman, H. Tschofenig. May 2018.

[7] IETF RFC 4648: "The Base16, Base32, and Base64 Data Encodings", S. Josefsson, October 2006.

[8] UHD Forum: "Watermarking API for Encoder Integration, version 1.0.1", March 2021.

[9] IEEE Std 1003.1™ 2018 Edition, The Open Group Base Specifications Issue 7, 31 January 2018.

[10] DASH-IF registry of watermarking technology vendors IDs.

[11] IETF RFC 9053: "CBOR Object Signing and Encryption (COSE): Initial Algorithms", J. Schaad, August 2022.

[12] IANA: "CBOR Web Token (CWT) Claims".

2.2 Informative references

References are either specific (identified by date of publication and/or edition number or version number) or non-specific. For specific references, only the cited version applies. For non-specific references, the latest version of the referenced document (including any amendments) applies.

NOTE: While any hyperlinks included in this clause were valid at the time of publication, ETSI cannot guarantee their long term validity.

The following referenced documents are not necessary for the application of the present document but they assist the user with regard to a particular subject area.

[i.1] DASH-IF Live Media Ingest Protocol.

[i.2] Web Sequence Diagram.

3 Definition of terms, symbols and abbreviations

3.1 Terms

For the purposes of the present document, the following terms apply:

client-driven watermarking: action of watermarking content when the user device is performing some actions allowing it to make unique requests for content

NOTE: The user device embeds a watermarking agent that is integrated with the application.

client-side watermarking: action of watermarking when the user device is the sole responsible for doing the actual watermarking of content

NOTE: The user device embeds a watermarking agent that is integrated with the audio-visual rendering engine.

server-driven watermarking: action of watermarking content when the user device is not performing any other operation than conveying information such as tokens, between servers that are responsible for doing the actual watermarking of content that is delivered to the user device

sequencing: action of returning a Variant of a segment when it is requested, based on a watermark token

NOTE: Typically, this action is performed on a CDN edge server and is thus referred to as "edge sequencing".

variant: alternative representation of a given segment of a multimedia asset

NOTE: Typically, a Variant is a pre-watermarked version of the segment.

WaterMark (WM) pattern: series of A/B decisions for every segment that is unique per user device

3.2 Symbols

Void.

3.3 Abbreviations

For the purposes of the present document, the following abbreviations apply:

ABR	Adaptive Bit Rate
AES	Advanced Encryption Standard
AF	Adaptation Field
API	Application Programming Interface
AVC	Advanced Video Codec
CBOR	Concise Binary Object Representation
CDDL	Concise Data Definition Language
CDN	Content Delivery Network
CMAF	Common Media Application Format
COSE	CBOR Object Signing and Encryption
CPU	Central Processing Unit
CWT	CBOR Web Token
DAI	Dynamic Ad Insertion
DASH	Dynamic Adaptive Streaming over HTTP
DRM	Digital Rights Management

ECDH	Elliptic Curve Diffie-Hellman
HEVC	High Efficiency Video Coding
HLS	HTTP Live Streaming
HMAC	keyed-Hashing for Message AuthentiCation
HTTP	Hypertext Transfer Protocol
IANA	Internet Assigned Numbers Authority
IOP	InterOPerability
IP	Internet Protocol
ISOBMFF	ISO Base Media File Format
JITP	Just In Time Packager
JSON	JavaScript Object Notation
JWT	JSON Web Token
MPD	Media Presentation Description
NAL	Network Abstraction Layer
OTT	Over The Top
RIST	Reliable Internet Stream Transport
RTMP	Real-Time Messaging Protocol
RTP	Real Time Protocol
SEI	Supplemental Enhancement Information
SRT	Secure Reliable Transport
TS	Transport Stream
TV	TeleVision
UDP	User Datagram Protocol
UHD	Ultra-High Definition
URI	Uniform Resource Identifier
URL	Uniform Resource Locator
UUID	Universally Unique IDentifier
VOD	Video On Demand
WM	WaterMark
WMID	WaterMark IDentifier
WMT	WaterMark Token

4 OTT Watermarking Using Variants

The objective of forensic watermarking is to deliver a unique version of a media asset to the different users consuming the asset. This is somewhat in opposition with media delivery mechanisms that aim at delivering the same asset to all users for efficiency purposes. As a result, in the broadcast era, a typical approach was to perform the watermarking operation at the very last step of the media delivery pipeline, within the end user device e.g. a set-top box. This solution has the virtue of leaving the whole media delivery pipeline unaltered but raises security and interoperability challenges when a large variety of devices owned and operated by the end user shall be supported. This is for instance the case with Over The Top (OTT) media delivery where content is consumed on mobile phones, tablets, laptops, connected TVs, etc. As a result, new forensic watermarking solutions have gained momentum that do not perform security-sensitive and complex operations in the end user realm. While such approaches require minimal changes in the end-user devices, they do mandate the media delivery pipeline to be modified accordingly.

A notable example of such network-side watermarking solutions is OTT watermarking using Variants for Adaptive Bit Rate (ABR) content. In this case, the content is delivered by segments. The baseline idea is then to generate pre-watermarked Variants of each segment and to modify the delivery protocol so that each end user receives a unique sequence of Variants depending on a watermark pattern that has been assigned to the end user. The semantic of this pattern is context dependent and can be, for instance, a device identifier, an account identifier, a session identifier, etc. Figure 1 illustrates a particular case of this strategy, coined as A/B watermarking, where there are two Variants generated for each segment, each Variant containing a watermark that either encodes the information '0' or '1'. As a result, the watermarking system will require the transmission of a sequence of Variants as long as the length of the pattern to successfully recover the whole unique identifier.

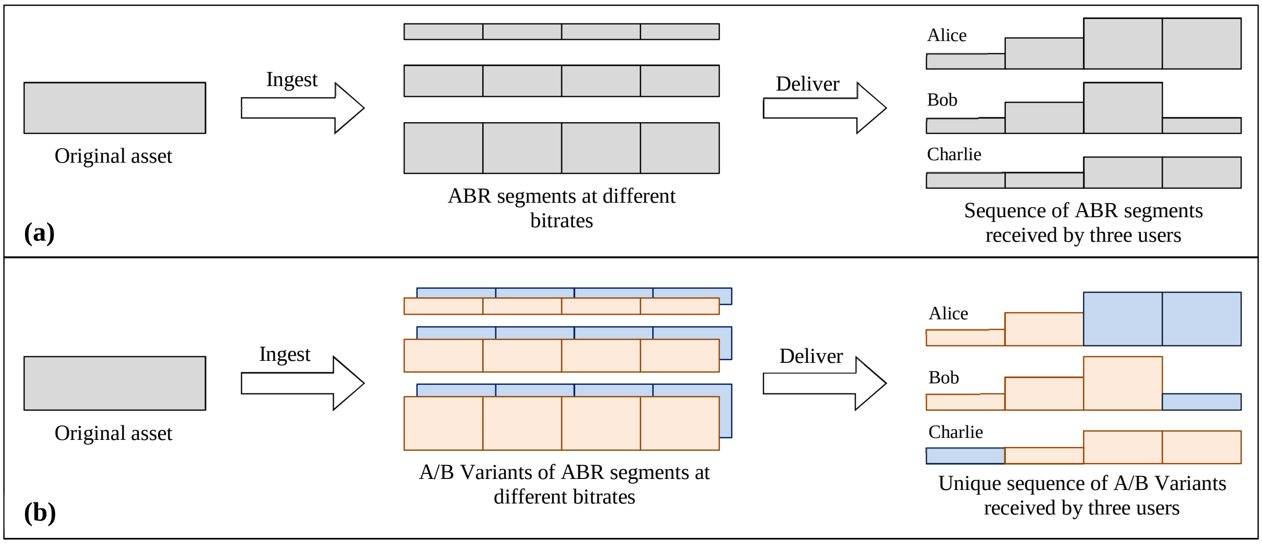

Figure 1: A/B watermarking concept with (a) ABR content delivery and (b) A/B Variants delivery

When using Variants, the serialization process essentially boils down to delivering a unique sequence of Variants to each individual end user. There are two main families of methods to achieve this:

1) Server-driven methods, wherein the client does perform no operation related to watermarking. It simply fetches and forwards a token to the CDN that is responsible for delivering a unique sequence of Variants.

2) Client-driven methods, wherein the client is responsible for the serialization operation. For instance, it relies on some session-based digital object to tamper the URI ABR segments and thereby directly query a unique sequence of Variants from the CDN.

The present document is describing the server-driven methods. Client-driven methods are not part of the present document.

5 Server-Driven Architecture and Workflows

5.1 Introduction

In the server-driven architecture, the device is unaware that content it consumes is watermarked. The device only exchanges a token with servers allowing these servers, usually CDN edges, to make the decision on which A or B Variant it delivers to the device. In the present document, an end-to-end system is presented. It includes the definition of watermarking metadata that limits the need for naming conventions by allowing the encoder to send this metadata all the way to the edge through origins to enable the sequencing of bits. The following goes through the functional architecture and describes the workflows when preparing content and when consuming content.

In the following, it is assumed that the edge is a CDN edge. There are optional architectures, but this does impact the overall functional architecture and workflows. It is also assumed that multi-track content (audio and video multiplexed in one segment) is out of the scope of the present document. In addition, all the workflows are only examples of possible implementations.

5.2 Functional Architecture

Figure 2 shows the simplified high-level functional architecture and the different interaction between the components that are involved in the flows when a device consumes watermarked content. Note that this also shows that content is encrypted, as watermarking will likely be added for premium content that is also encrypted and protected by a DRM system.

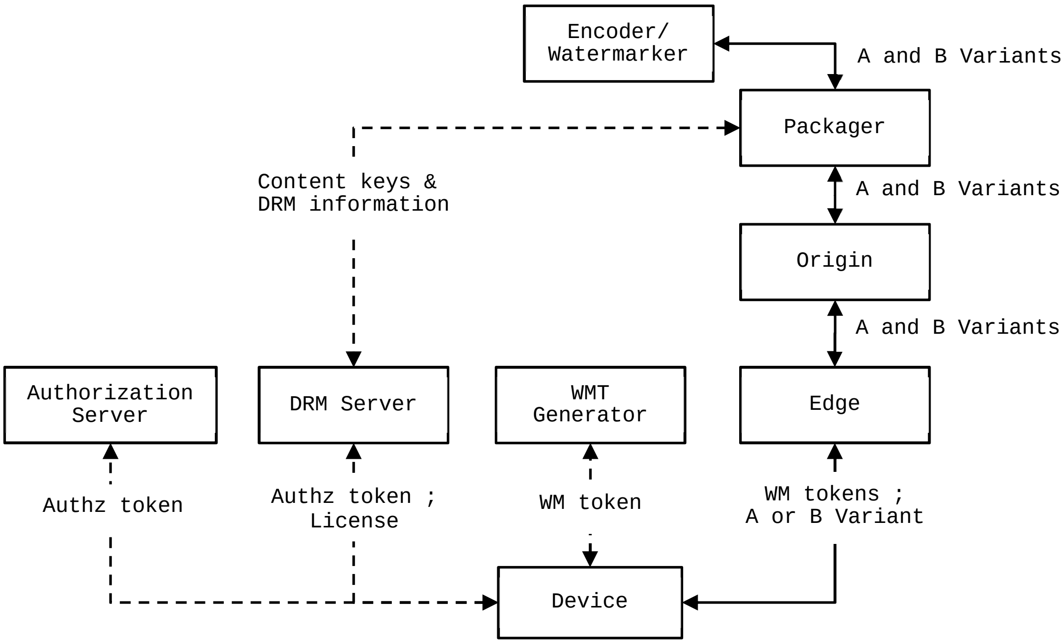

Figure 2: Functional architecture

To consume content, a device needs, at minimum, to have an authorization token (for getting a DRM license) and a WM token that contains a WM pattern, a series of A or B decisions. The device is responsible for obtaining the required data before requesting segments to the CDN.

5.3 System Configuration

Enabling or disabling the edge sequencing logic is set through the configuration to the edge. As an example, this can be useful for a service of live sporting events where only premium events require watermarking enforcement. Other moments of the day do not require it. In this case, content is still watermarked but the edge is only configured to sequence during the limited period of time of the premium event. When sequencing is disabled, the edge shall consume segments on the endpoint for Variant A. If this endpoint is not working properly, the origin shall deliver any available Variant on this endpoint.

NOTE 1: When enabling watermarking, all devices that do not have a WM token will receive an error when requesting content, hence they are then forced to request such token.

NOTE 2: As an example, enabling and disabling sequencing can be done with an API enable (true/false).

Watermarked objects names shall include a pattern that the CDN can match to differentiate these objects from non-watermarked objects (initialization segments, subtitles, trickplay images). As an example, for a DASH manifest located at https://edge.hostname/path/to/endpoint/index.mpd that references video segments as:

```
<SegmentTemplate timescale="60000" media="video_segment_$RepresentationID$_$Time$.mp4"
initialization="video_init_$RepresentationID$.mp4" startNumber="10967120"
presentationTimeOffset="903486496960">
```

The pattern for the differentiation of these objects from non-watermarked objects is **`video_segment_`**.

One of the following identification schemes, referred as `variantId` in the present document, shall be used for the identification of the Variants:

- A lower-case letter beginning with 'a'. Variants are then 'a', 'b' and so on.
- A number beginning with 0. Variants are then 0, 1 and so on.

When addressing content, `variantId` shall be translated into `variantPath` as follows:

- `variantPath` = `${variantId}` followed by '/' with the exception, that if `${variantId}` is 'a' or '0' then `${variantPath}` may be empty.

5.4 WM Token

A WM token provides a WM pattern which is unique (for example per streaming session or per user). This pattern allows the sequencing of A/B Variants.

Two tokenization schemes are defined in the present document. The first, named direct, embeds the WM pattern in the token and can be opened and interpreted by an edge irrespective of the underlying WM technology and provider. The second, named indirect, requires integration of a WM technology provider's edge sequencing software at the edge.

The following are requirements on the WM token:

- The token shall be a CWT token, the basic structural requirements are defined in IETF RFC 8392 [6].
- The token shall be with integer keys in "deterministically encoded CBOR" as specified in IETF RFC 8949 [4], clause 4.2.
- Recipients shall process claims listed in IETF RFC 8392 [6], clause 3.1 when they are present. `exp` and `iat` shall be present.
- The token shall include either a WM pattern (direct mode) or data for deriving the WM pattern (indirect mode). Absence of a `wmpattern` claim implies that the token is in indirect mode.
- Recipients shall support direct mode and may support indirect mode.
- The token shall be signed as described in clause 7 of IETF RFC 8392 [6]. Recipients shall support the HMAC 256/256 (kty number 5) and ES256 (kty number -7) algorithms.
- The token shall be base64url-encoded as described in clause 5 of IETF RFC 4648 [7].

The following claims are defined and Table 1 provides the integer claim keys:

```
wmtoken = {
  wmver-label ^ => wmver-value,
  wmvnd-label ^ => wmvnd-value,
  wmpatlen-label ^ => wmpatlen-value,
  ? wmsegduration-label ^ => wmsegduration-value,
  wmtoken-direct // wmtoken-indirect,
  * wmext-label => any
}

wmver-value = uint .size 1
wmvnd-value = uint .size 1
wmpatlen-value = uint .size 2
wmsegduration-value = [(wmtimeticks : uint, wmtimescale : uint)]
wmext-label = int

; direct mode
wmtoken-direct = {
  wmpattern-label ^ => wmpattern-value
}
wmpattern-value = COSE_Encrypt0 // COSE_Encrypt // bytes

; indirect mode
wmtoken-indirect = {
  wmid-label ^ => wmid-value
  wmopid-label ^ => wmopid-value
  wmkeyver-label ^ => wmkeyver-value
}
wmid-value = text
wmopid-value = uint
wmkeyver-value = uint
```

Table 1: Integer Claim key values for the WM token

Claim label	Integer key
`wmver-label`	300
`wmvnd-label`	301
`wmpatlen-label`	302
`wmsegduration-label`	303
`wmpattern-label`	304
`wmid-label`	305
`wmopid-label`	306
`wmkeyver-label`	307

`wmver`
The version of the WM Token. Recipients shall support this claim. The present document describes version 1.

`wmvnd`
The WM technology vendor. Recipients shall support this claim. This provides the context for the key material needed for signature verification. In the direct mode, it also provides the context for the key material needed for decrypting `wmpattern` if needed. In the indirect mode, it identifies the vendor specific core to use. A list of WM technology vendor identifiers is available at [10].

`wmpatlen`
The length in bits of the WM pattern. Recipients shall support this claim.

`wmpattern`
The WM pattern. Recipients shall support this claim in direct mode. It is recommended to encrypt the pattern. Recipients shall support ECDH-SS+A128KW (key type -32) as defined in IETF RFC 9053 [11].

`wmsegduration`
The nominal duration of a segment. This claim is optional. Recipients may support this claim. When `WMPaceInfo` data is not available, this may allow the edge to define the index to be considered in the WM pattern. If `WMPaceInfo` is available, this claim shall be ignored. The array contains exactly 2 values. The first value is a duration in time ticks where its base unit is defined by the second value. The second value is the scale in number of time ticks per second. As an example, [60'000, 10'000] means that the segments are 60'000 ticks long while the scale is 10'000 ticks per second, `wmsegduration` is then equal to 6 seconds.

`wmid`
Used as input to derive the WM pattern for indirect mode. Recipients shall support this claim in indirect mode. The derivation algorithm is not defined in the present document and is vendor specific.

`wmopid`
Used as additional input to derive the WM pattern for indirect mode. Recipients shall support this claim in indirect mode.

`wmkeyver`
The key to use for derivation of the WM pattern in indirect mode. Recipients shall support this claim in indirect mode.

Once the WM pattern is obtained from the token (either directly, decrypted or calculated), the CDN edge shall enforce big-endian convention to address a single bit in it when using the value of `position` (defined in clause 5.5.2).

The following is an example with a WM pattern equal to `0x0A0B0C0D`.

Byte	0	1	2	3
bit offset	01234567	01234567	01234567	01234567
binary	00001010	00001011	00001100	00001101
hex	0A	0B	0C	0D

For a value of `position` equal to 3, the bit to consider is highlighted in green (equal to 0). This is not any other bit, especially, those highlighted in red.