Text

Description automatically generatedISO/IEC JTC 1/SC 29/WG 03 N1504

**ISO/IEC JTC 1/SC 29/WG 03  
MPEG Systems   
Convenorship: KATS (Korea, Republic of)**

**Document type:** Output Document

**Title:** Technologies under Consideration for ISO/IEC 23001-7

**Status:** Approved

**Date of document:** 2025-05-01

**Source:** ISO/IEC JTC 1/SC 29/WG 03

**No. of pages:** 9 (with cover page)

**Email of Convenor:** young.L @ samsung . com

**Committee URL:** <https://isotc.iso.org/livelink/livelink/open/jtc1sc29wg3>

**INTERNATIONAL ORGANIZATION FOR STANDARDIZATION**

**ORGANISATION INTERNATIONALE DE NORMALISATION**

**ISO/IEC JTC 1/SC 29/WG 03 MPEG SYSTEMS**

**ISO/IEC JTC 1/SC 29/WG 03 N1504**

**May 2025, Online**

|  |  |
| --- | --- |
| **Title** | **Technologies under Consideration for ISO/IEC 23001-7** |
| **Source** | **WG 03, MPEG Systems** |
| **Status** | **Approved** |
| **Serial Number** | **25040** |

# Signaling of Common Encryption Tools

ISO/IEC 23001-7 relies on ISO/IEC 14496-12 structures (e.g., boxes) for signaling various aspects of encryption. As ISO/IEC 14496-12 evolves and adds versions/flags to boxes used by ISO/IEC 23001-7, the File Format group considers improving the signaling of the exact support required. Experts’ feedback is encouraged either through [GitHub](https://github.com/MPEGGroup/FileFormat/issues) or through MPEG contributions.

# Option 1 defining a new brand

1. The brand 'coen'
   1. Overview

The requirements for files and readers compliant to the brand 'coen' are defined in the following subclause. It is meant to be independent of the actual encryption scheme ('cenc', 'cens', 'cbc1', 'cbcs' …).

* 1. Requirements on files

Files containing the brand 'coen' in the compatible brands array of the FileTypeBox shall conform to the constraints defined in this subclause.

The following boxes are required in a file under the 'coen' brand. The Version column in the following table lists the versions of the boxes allowed by this brand. Other versions of the boxes shall not be present. The Flags column in the following table lists the flags that may be present in files of this brand and are required to be supported by the readers of this brand. If other flags of the boxes have been specified, they may be present but are not required to be supported by the readers of this brand.

NOTE A '-' in the Version and Flags column indicates that the box is a container box or does not support versioning.

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Hierarchy of boxes** | | | | **Version** | **Flags** | **Box description** |
| encv  enca  encs  enct  encu  encf  enc3  encm  encp |  |  |  | - | - | Protected Sample Entry. At least one of these entry needs to be present. |
|  | sinf |  |  | - | - |  |
|  |  | frma |  | - | - |  |
|  |  | schm |  | 0 | 0x000001 |  |
|  |  | schi |  | - | - |  |
|  |  |  | tenc | 0,1 | 0 |  |

* 1. Requirements on readers

Support for the following boxes is required under the 'coen' brand. The Version column in the following table specifies the versions of the boxes that shall be supported by the readers of the 'coen' brand.

NOTE A '-' in the Version and Flags column indicates that the box is a container box or does not support versioning nor flagging.

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Hierarchy of boxes** | | | | **Version** | **Flags** | **Box description** |
| encv  enca  encs  enct  encu  encf  enc3  encm  encp |  |  |  | - | - | Readers may support only some Protected Sample Entry types. Derived specifications may require specific support. |
|  | sinf |  |  | - | - |  |
|  |  | frma |  | - | - |  |
|  |  | schm |  | 0 | 0x000001 |  |
|  |  | schi |  | - | - |  |
|  |  |  | tenc | 0,1 | 0 |  |
| moov  moof |  |  |  | - | - |  |
|  | pssh |  |  | 0,1 | 0 |  |
| sbtl  traf |  |  |  | - | - |  |
|  | sbgp |  |  | 0,1 |  |  |
|  | sgpd |  |  | 0,1,2 | 0 |  |
|  |  | seig |  | - | - |  |
|  | saiz |  |  | 0 | 0x000001 |  |
|  | saio |  |  | 0,1 | 0x000001 |  |
| trak  traf |  |  |  | - | - |  |
|  | senc |  |  | 0 | 0x000002 |  |

Open questions:

* Is one brand sufficient to represent today’s support of “old tools”? For example, some devices only support ‘saio’/’saiz’, others only support ‘senc’. Should we define 2 different brands?
* Similarly, use of ‘seig’ is specific to some use cases (e.g. multi-key support). Should this be in a separate brand?

# Option: General restructuration of protection schemes

From the existing CENC specification **Error! Reference source not found.**, two sets of features may be distinguished:

1. a first set possibly leading to “sub-schemes” and
2. a second set not leading to “sub-schemes” because aligned with (or corresponding to) one or several existing protection schemes.

**Set of features leading to “sub-schemes”**

* **Full sample versus subsample encryption** clearly splits each protection scheme into 2 “sub-schemes”
* **Information about encryption keys:** 
  + Single versus multiple keys per sample
    - Multiple keys inducing presence of ‘seig’ sample group
  + 128 versus 256 bit key length
* The clear trailing (cipher alignement or last cipher block) or not (BytesOfProtectedData = multiple of 16 bytes)

**Set of features aligned with existing protection schemes (and not leading to “sub-schemes”)**

* Pattern encryption (has its dedicated protection schemes)
* Initialization vector per sample or not also corresponds to the protection scheme in use
* CENC SAI does not seem to be discriminant and may not lead to “sub-schemes”. A possibly relevant information may be whether it comes as a senc box or as saiz/saio boxes.

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Protection Scheme | Pattern enc.  (‘tenc’ version) | Per sample Initialization Vector (IV) | **Full or partial**  **sample** | **Allow last cipher block to be partial** | **Single / multi key** | **Key length** | Sample auxiliary information? |
| **‘cenc’**  **CTR**  (10.1) | No  (v0) | Yes | Full (non NAL) | Should avoid (10.1) | Single  (non NAL) | 128 | Yes (9.3) |
| 256 |
| Multi=N.A |  |
| Partial (NAL or as defined by derived spec) | Single | 128 | Yes (9.3) |
| 256 |
| Multi (+non NAL) | 128 |
| 256 |
| **‘cbc1’**  **CBC**  (10.2) | No  (v0) | Yes | Full (non NAL) | Yes in clear (9.4.3) | Single (non NAL) | 128 | Yes  (9.4.3) |
| 256 |
| Multi=N.A |  |
| Partial (NAL or as defined by derived spec) | Forbiden  (10.2) | Single | 128 | Yes  (9.5.1) |
| 256 |
| Multi (+non NAL) | 128 |
| 256 |
| **‘cens’ CTR**  (10.3) | Yes  (v1) | Yes | Partial (NAL or as defined by derived spec) | Forbiden (10.3) | Single | 128 | Yes (9.3) |
| 256 |
| Multi (+non NAL) | 128 |
| 256 |
| Whole block full sample (**non NAL**) | Yes (9.7)  Forbiden (10.3) | Single | 128 | Yes (9.3) |
| 256 |
|  | Multi=N.A. |  |
| **‘cbcs’ CBC**  (10.4) | Yes  (v1) | No | Partial (NAL or other) | Yes (9.5) | Single (+non NAL) | 128 | Yes (10.4) |
| 256 |
| Multi (+non NAL) | 128 |
| 256 |
| Whole block full sample  (**non NAL**) | Yes (9.7) | Single | 128 | No  (SAI size is 0) |
| 256 |
| Multi=N.A |  | Yes (10.4) |
| **‘sve1’ CTR**  (10.5) | No  (v0) | Yes (10.5) | **N.A.** | Yes (9.5) | No restrictions | 128 | Yes |
| 256 |

Table 1: CENC features versus protection schemes

The current text for protection schemes is prone to errors as it duplicates many statements from the rest of the specification, with the additional risk of error propagation in derived specifications.

It would be much cleaner if all tools were clearly identified in the TrackEncryptionBox, and protection schemes only defining the allowed values.

Moreover, it would be cleaner if one protection scheme would not define completely different things, such as full-sample encryption mode ***AND*** sub-sample based encryption ***AND*** pattern-based usage. A reader should know, inspecting only from the protection scheme info and/or the TrackEncryptionBox (either directly or through MIME sub-parameters mapping), the kind of protection involved, rather than having to look at the media type as well.

It might be difficult to change existing definitions of the protection schemes for backward compatibility reasons, but *it would be much cleaner to have a single protection scheme with MIME sub-parameters*. If the current approach were to be kept, it would still be much better to define one protection scheme per functionality, e.g.:

* ‘scbc’: CBC subsample, pattern indicated in `tenc`
* ‘fcbc’: CBC whole-block full sample, pattern indicated in `tenc`
* ‘mcbc`: Multikey CBC, pattern indicated in `tenc`
* ‘…’ possibly as many as rows in Table 1

# Relaxing constraints in Common Encryption

This section discusses the possibility to relax constraints written in the CENC specification, while technically feasible, for different encryption modes.

## Full Sample Encryption

CENC section 9.4, currently has two modes for full sample encryption:

1. one for AES-CTR (section 9.4.2) where all bytes are encrypted (even if not complete),
2. one for AES-CBC (section 9.4.3) where the last bytes are left in the clear if less than a full block (called “whole-block full sample encryption”).

It is **currently not possible to use full sample encryption** (‘cenc’ or ‘cbc1’ protection schemes) **for a NALU-based video**, for example in case the content provider does not want to expose any of the NAL structure of the stream.

The section 9.4.1 states:

“*Full sample encryption may be used for all encrypted media types other than* ***NAL structured video****, which* ***shall use subsample encryption***.”

Moreover, the definitions of the cenc (section 10.1) and cbc1 (section 10.2) protection schemes both state:

“*Encrypted video tracks or items using NAL unit structured video conforming to ISO/IEC 14496-15 shall be protected using subsample encryption specified in* ***Error! Reference source not found.***, …”

The restriction only comes from the definitions of the protection schemes, and most implementations are likely able to support full-sample encryption even for NALU-based video tracks.

It would then be interesting to investigate how these restrictions could be lifted.

## Subsample encryption for non NALU-based media

CENC section 9.5 currently constrains the usage of subsample encryption for non-NALU based formats as follows:

* For cenc, cens and cbc1: can be only if multi-key per sample is used (otherwise no information is available at sub-sample level)
* For cbcs: can be used

For new formats which are not NALU-based, e.g. uncompressed video and images or MPEG-H Audio Stream (MHAS) audio, this implies that, for single-key, only cbcs can be used if part of the media is designed to be in the clear, or if the content provider wants to align internal framing of the sample (e.g. a tile in uncompressed video, an MHAS packet) to encryption boundaries.

This restriction is problematic, and again most implementations can already handle sub-sample decryption independently of the NALU-based structure of the track.

It would then be interesting to investigate how these restrictions could be lifted.

In particular, it is important to note that the usage of subsample for a track is implicit (depends on protection scheme and media type) and never explicitly signaled in the TrackEncryptionBox.

## Pattern encryption

CENC section 9.6 currently defines pattern encryption as a simple tool allowing to protect only N out of K blocks. In theory, the tool could be applied to any protection scheme, but is only allowed on subsample-based encryption modes in the definitions of cens and cbcs protection schemes, consequently mostly for NALU-based media only (except for cbcs scheme).

It would be interesting to investigate how these restrictions could be lifted, especially considering the following statement from section 10.3:

“*Derived specifications may relax this constraint to allow usage of subsample encryption as specified in subclause* ***Error! Reference source not found.****, in which case pattern encryption as specified in* ***Error! Reference source not found.*** *shall be used.*”

# Other comments

## Comments on the ‘cens’ protection scheme

### Comments “cens\_1”

Section 10.3 states, for ‘cens’ AES-CTR, that:

" The BytesOfProtectedData size shall be a multiple of 16 bytes to avoid partial cipher blocks in subsamples”

(while the definition of cenc in 10.2 is considering a “should be a multiple of 16 bytes” …)

We need to understand why the "shall” is used in cens, as this decision is taken by the encryptor but does not impact the decryptor module.

### Comments “cens\_2”

In the definition of the cens (10.3) protection scheme, the following statement:

“The BytesOfProtectedData size shall be a multiple of 16 bytes to avoid partial cipher blocks in subsamples.”

should be rephrased as follows:

“In video tracks using subsample encryption, the BytesOfProtectedData size shall be a multiple of 16 bytes to avoid partial cipher blocks in subsamples.“

This is because “The BytesOfProtectedData size ” is meaningless in full sample encryption.

### Comments “cens\_3”

The ‘cens’ protection scheme states that

* “Constant IVs shall not be used”, and
* “When a single key applies to each sample, encrypted tracks or items not using NAL structured video shall be protected using whole-block full-sample encryption as specified in subclause **Error! Reference source not found.**”

We note that the whole-block full sample encryption (9.7) is using constant IVs.

The text should then be corrected to indicate usage of full sample encryption (9.4).

## On subsample encryption and CENC SAI

Section 9.5.1 (definition of subsample encryption) uses BytesOfClearData and BytesOfProtectedData. However, the CENC specification never explicitly states that CENC SAI shall be present.

This should be clarified.

As another general editorial comment, we suggest the File Format group to consider moving NAL / non NAL specific sections in dedicated Annexes to avoid “polluting” definition of protection schemes with video format.