INTERNATIONAL STANDARD

ISO/IEC 23001-7

First edition
2012-02-01

Information technology — MPEG systems technologies —

Part 7: Common encryption in ISO base media file format files

Technologies de l'information — Technologies des systèmes MPEG —

Partie 7: Cryptage commun des fichiers au format de fichier de médias de la base ISO

Reference number
ISO/IEC 23001-7:2012(E)

ISO copyright office
Case postale 56 • CH-1211 Geneva 20
Tel. + 41 22 749 01 11
Fax + 41 22 749 09 47
E-mail copyright@iso.org
Web www.iso.org

Published in Switzerland

Contents

Page

Foreword

ISO (the International Organization for Standardization) and IEC (the International Electrotechnical Commission) form the specialized system for worldwide standardization. National bodies that are members of ISO or IEC participate in the development of International Standards through technical committees established by the respective organization to deal with particular fields of technical activity. ISO and IEC technical committees collaborate in fields of mutual interest. Other international organizations, governmental and non-governmental, in liaison with ISO and IEC, also take part in the work. In the field of information technology, ISO and IEC have established a joint technical committee, ISO/IEC JTC 1.

International Standards are drafted in accordance with the rules given in the ISO/IEC Directives, Part 2.

The main task of the joint technical committee is to prepare International Standards. Draft International Standards adopted by the joint technical committee are circulated to national bodies for voting. Publication as an International Standard requires approval by at least 75 % of the national bodies casting a vote.

Attention is drawn to the possibility that some of the elements of this document may be the subject of patent rights. ISO and IEC shall not be held responsible for identifying any or all such patent rights.

ISO/IEC 23001-7 was prepared by Joint Technical Committee ISO/IEC JTC 1, *Information technology*, Subcommittee SC 29, *Coding of audio, picture, multimedia and hypermedia information*.

ISO/IEC 23001 consists of the following parts, under the general title *Information technology — MPEG systems technologies*:

- *Part 1: Binary MPEG format for XML*
- *Part 2: Fragment request units*
- *Part 3: XML IPMP messages*
- *Part 4: Codec configuration representation*
- *Part 5: Bitstream Syntax Description Language (BSDL)*
- *Part 7: Common encryption in ISO base media file format files*

Introduction

The Common Encryption (‘cenc’) protection scheme specifies standard encryption and key mapping methods that can be utilized by one or more digital rights and key management systems [digital-rights management (DRM systems)] to enable decryption of the same file using different DRM systems. The scheme operates by defining a common format for the encryption related metadata necessary to decrypt the protected streams, yet leaves the details of rights mappings, key acquisition and storage, DRM compliance rules, etc., up to the DRM system or systems supporting the ‘cenc’ scheme. For instance, DRM systems supporting the ‘cenc’ protection scheme must support identifying the decryption key via ‘cenc’ key identifier (KID) but how the DRM system locates the identified decryption key is left to a DRM-specific method. DRM specific information such as licenses or rights and license/rights acquisition information can be stored in an ISO Base Media file using a Protection System Specific Header box (‘pssh’), using one for each DRM system applied. DRM licenses/rights need not be stored in the file in order to look up a key using KID values stored in the file and decrypt media samples using the encryption parameters stored in each track.

Information technology — MPEG systems technologies —

Part 7:
Common encryption in ISO base media file format files

1 Scope

This part of ISO/IEC 23001 specifies a common encryption format for use in any file format based on ISO/IEC 14496-12, the ISO base media file format.

2 Normative references

The following referenced documents are indispensable for the application of this document. For dated references, only the edition cited applies. For undated references, the latest edition of the referenced document (including any amendments) applies.

ISO/IEC 14496-10, *Information technology — Coding of audio-visual objects — Part 10: Advanced Video Coding*

ISO/IEC 14496-12, *Information technology — Coding of audio-visual objects — Part 12: ISO base media file format*

ISO/IEC 14496-15, *Information technology — Coding of audio-visual objects — Part 15: Advanced Video Coding (AVC) file format*

Advanced Encryption Standard, Federal Information Processing Standards Publication 197, FIPS-197, http://www.nist.gov/

Recommendation of Block Cipher Modes of Operation, NIST, NIST Special Publication 800-38A, http://www.nist.gov/

3 Definitions

3.1 Terms and definitions

For the purposes of this document, the following terms and definitions apply.

3.1.1
ISO Base Media File
name of a file conforming to the file format described in ISO/IEC 14496-12 in which the techniques in ISO/IEC 23001-7 may be used

NOTE Adapted from ISO/IEC 14496-12, definition 3.1.8.

3.2 Abbreviated terms

For the purposes of this document, the following abbreviated terms apply.

AES	Advanced Encryption Standard as specified in Federal Information Processing Standards Publication 197, FIPS-197
AES-CTR	AES Counter Mode as specified in *Recommendation of Block Cipher Modes of Operation*, NIST, NIST Special Publication 800-38A
AVC	Advanced Video Compression as specified in ISO/IEC 14496-10
ISOAVC	An ISO Base Media File containing AVC media tracks as specified in ISO/IEC 14496-15
NAL	Network Abstraction Layer as specified in ISO/IEC 14496-10

4 Scheme Signalling

Scheme signaling shall conform to ISO/IEC 14496-12. As defined in ISO/IEC 14496-12, the sample entry is transformed and a Protection Scheme Information Box (‘sinf’) is added to the standard sample entry in the Sample Description Box to denote that a stream is encrypted. The Protection Scheme Information Box shall contain a Scheme Type Box (‘schm’) so that the scheme is identifiable. The Scheme Type Box has the following additional constraints:

- The scheme_type field is set to a value of ‘cenc’ (Common Encryption).
- The scheme_version field is set to 0x00010000 (Major version 1, Minor version 0).

The Protection Scheme Information Box shall also contain a Scheme Information Box (‘schi’). The Scheme Information Box has the following additional constraint:

- The Scheme Information Box shall contain a Track Encryption Box (‘tenc’), describing the default encryption parameters for the track.

5 Overview of Encryption Metadata

The encryption metadata defined by the ‘cenc’ Common Encryption Scheme can be categorized as follows:

- Protection System Specific Data – this data is opaque to the ‘cenc’ Common Encryption Scheme. This gives protection systems a place to store their own data using a common mechanism. This data is contained in the ProtectionSystemSpecificHeaderBox described in 8.1.
- Common encryption information for a media track – this includes default values for the key identifier (KID), initialization vector size, and encryption flag. This data is contained in the TrackEncryptionBox described in 8.2.
- Common encryption information for groups of media samples – this includes overrides to the track level defaults for key identifier (KID), initialization vector size, and encryption flag. This allows groups of samples within the track to use different keys, a mix of clear and encrypted content, etc. This data is contained in a SampleGroupDescriptionBox ('sgpd') that is referenced by a SampleToGroupBox (‘sbgp’). See 6 for further details.
- Encryption information for individual media samples – this includes initialization vectors and, if required, sub sample encryption data. This data is sample auxiliary information, referenced by using a SampleAuxiliaryInformationSizesBox (‘saiz’) and a SampleAuxiliaryInformationOffsetsBox ('saio'). See 7 for further details.

6 Encryption Parameters shared by groups of samples

Each sample in a protected track shall be associated with an `IsEncrypted` flag, `IV_Size`, and `KID`. This can be accomplished by (a) relying on the default values in the `TrackEncryptionBox` (see 8.2), or (b) specifying the parameters by sample group, or (c) using a combination of these two techniques.

When specifying the parameters by sample group, the `SampleToGroupBox` in the sample table or track fragment specifies which samples use which sample group description from the `SampleGroupDescriptionBox`. The format of the sample group description is based on the handler type for the track.

Tracks with a handler type of ‘vide’ shall use the sample group description structure, `CencSampleEncryptionInformationVideoGroupEntry`, which has the following syntax:

```
aligned(8) class CencSampleEncryptionInformationVideoGroupEntry
   extends VisualSampleGroupEntry( ‘seig’ )
{
   unsigned int(24)      IsEncrypted;
   unsigned int(8)       IV_size;
   unsigned int(8)[16]   KID;
}
```

Similarly, tracks with a handler type of ‘soun’ shall use the sample group description structure, `CencSampleEncryptionInformationAudioGroupEntry`, which has the following syntax:

```
aligned(8) class CencSampleEncryptionInformationAudioGroupEntry
   extends AudioSampleGroupEntry( ‘seig’ )
{
   unsigned int(24)      IsEncrypted;
   unsigned int(8)       IV_size;
   unsigned int(8)[16]   KID;
}
```

NOTE Groups with identical structure should be defined if protection of other media types is needed.

These structures use a common semantic for their fields as follows:

`IsEncrypted` is the flag which indicates the encryption state of the samples in the sample group. See the `IsEncrypted` field in 9.2 for further details.

`IV_size` is the Initialization Vector size in bytes for samples in the sample group. See the `IV_size` field in 9.2 for further details.

`KID` is the key identifier used for samples in the sample group. See the `KID` field in 9.2 for further details.

In order to facilitate the addition of future optional fields, clients shall ignore additional bytes after the fields defined in the `CencSampleEncryption` group entry structures.

7 Common Encryption Sample Auxiliary Information

Each encrypted sample in a protected track shall have an Initialization Vector associated with it. Further, each encrypted sample in protected AVC video tracks shall conform to ISO/IEC 14496-10 and ISO/IEC 14496-15 and shall use the subsample encryption scheme specified in 9.6.2, which requires subsample encryption data. Both initialization vectors and subsample encryption data are provided as Sample Auxiliary Information with `aux_info_type` equal to ‘cenc’ and `aux_info_type_parameter` equal to 0. For tracks protected using the ‘cenc’ scheme, the default value for `aux_info_type` is equal to ‘cenc’ and the default value for the `aux_info_type_parameter` is 0 so content may be created omitting these optional fields. Storage of sample auxiliary information shall conform to ISO/IEC 14496-12.

The format of the sample auxiliary information for samples with this type shall be:

```
aligned(8) class CencSampleAuxiliaryDataFormat
{
   unsigned int(IV_size*8) InitializationVector;
   if ( sample_info_size > IV_size )
   {
      unsigned int(16) subsample_count;
      {
         unsigned int(16) BytesOfClearData;
         unsigned int(32) BytesOfEncryptedData;
      } [ subsample_count ]
   }
}
```

Where:

InitializationVector is the initialization vector for the sample. See the InitializationVector field in 9.2 for further details.

subsample_count is the count of subsamples for this sample. See the subsample_count field in 9.2 for further details.

BytesOfClearData is the number of bytes of clear data in this subsample. See the BytesofClearData field in 9.2 for further details.

BytesOfEncryptedData is the number of bytes of encrypted data in this subsample. See the BytesofEncryptedData field in 9.2 for further details.

If sub-sample encryption is not used (sample_info_size equals IV_size), then the entire sample is encrypted (see 9.5 for further details). In this case, all auxiliary information will have the same size and hence the default_sample_info_size of the SampleAuxiliaryInformationSizes box will be equal to the IV_Size of the initialization vectors.

Note, however, that even if subsample encryption is used, the size of the sample auxiliary information may be the same for all of the samples (if all of the samples have the same number of subsamples) and the default_sample_info_size used.

8 Box Definitions

8.1 Protection System Specific Header Box

8.1.1 Definition

Box Type:	`pssh’
Container:	Movie (‘moov’) or Movie Fragment (‘moof’)
Mandatory:	No
Quantity:	Zero or more

This box contains information needed by a Content Protection System to play back the content. The data format is specified by the system identified by the ‘pssh’ parameter SystemID, and is considered opaque for the purposes of this specification.

The data encapsulated in the Data field may be read by the identified Content Protection System to enable decryption key acquisition and decryption of media data. For license/rights-based systems, the header information may include data such as the URL of license server(s) or rights issuer(s) used, embedded licenses/rights, and/or other protection system specific metadata.

A single file may be constructed to be playable by multiple key and digital rights management (DRM) systems, by including one Protection System Specific Header box for each system supported. Readers that process such presentations shall match the SystemID field in this box to the SystemID(s) of the DRM System(s) they support, and select the matching Protection System Specific Header box(es) for retrieval of Protection System Specific information interpreted or created by that DRM system.

8.1.2 Syntax

```
aligned(8) class ProtectionSystemSpecificHeaderBox extends FullBox('pssh',
version=0, flags=0)
{
   unsigned int(8)[16]         SystemID;
   unsigned int(32)            DataSize;
   unsigned int(8)[DataSize]   Data;
}
```

8.1.3 Semantics

`SystemID` specifies a UUID that uniquely identifies the content protection system that this header belongs to.
`DataSize` specifies the size in bytes of the Data member.
`Data` holds the content protection system specific data.

8.2 Track Encryption Box

8.2.1 Definition

Box Type:	`tenc'
Container:	Scheme Information Box ('schi')
Mandatory:	No (Yes, for encrypted tracks)
Quantity:	Zero or one

The `TrackEncryptionBox` contains default values for the `IsEncrypted` flag, `IV_size`, and `KID` for the entire track These values are used as the encryption parameters for the samples in this track unless over-ridden by the sample group description associated with a group of samples. For files with only one key per track, this box allows the basic encryption parameters to be specified once per track instead of being repeated per sample.

8.2.2 Syntax

```
aligned(8) class TrackEncryptionBox extends FullBox('tenc', version=0, flags=0)
{
   unsigned int(24)     default_IsEncrypted;
   unsigned int(8)      default_IV_size;
   unsigned int(8)[16]  default_KID;
}
```

8.2.3 Semantics

`default_IsEncypted` is the encryption flag which indicates the default encryption state of the samples in the track. See the `IsEncrypted` field in 9.2 for further details.
`default_IV_size` is the default Initialization Vector size in bytes. See the IV_size field in 9.2 for further details.
`default_KID` is the default key identifier used for samples in this track. See the KID field in 9.2 for further details.

9 Encryption of Media Data

9.1 Encryption Schemes

Media data using the 'cenc' Protection Scheme shall use the *Advanced Encryption Standard*, Federal Information Processing Standards Publication 197, FIPS-197 published by the United States National Institute of Standards and Technology (NIST) using 128-bit keys in Counter Mode (AES-CTR), as specified in *Recommendation of Block Cipher Modes of Operation*, NIST, NIST Special Publication 800-38A. The scheme defines two elementary stream encryption formats, full sample encryption and subsample encryption. Full sample encryption is where the entire sample is encrypted as a single encryption unit whereas subsample encryption is where the sample is broken into smaller units each containing a clear area and an encrypted area. Encrypted AVC Video Tracks shall follow the subsample encryption scheme specified in 9.6.2 which defines a NAL unit based encryption scheme to allow access to NAL units and unencrypted NAL unit headers in an encrypted AVC stream. All other types of tracks shall follow the scheme specified in 9.5, which defines a simple sample-based encryption scheme.

9.2 Field semantics

Within the sample groups and sample auxiliary information used by the common encryption scheme, the fields have the following semantics:

`IsEncrypted` is the identifier of the encryption state of the samples in the track or group of samples. This flag takes the following values:

- 0x0: Not encrypted
- 0x1: Encrypted using AES 128-bit in CTR mode
- 0x000002 – 0xFFFFFF: Reserved

`IV_size` is the size in bytes of the InitializationVector field. Supported values:

- 0 – If the IsEncrypted flag is 0x0 (Not Encrypted).
- 8 – Specifies 64-bit initialization vectors.
- 16 – Specifies 128-bit initialization vectors.

`KID` is a key identifier that uniquely identifies the key needed to decrypt the associated samples. This allows the identification of multiple encryption keys per file or track. Unencrypted samples in an encrypted track shall be identified by having an `IsEncrypted` flag of 0x0, an `IV_size` of 0x0, and a `KID` value of 0x0.

`InitializationVector` specifies the initialization vector (IV) needed for decryption of a sample. For an `IsEncrypted` flag of 0x0, no initialization vectors are needed and the auxiliary information should have a size of 0, i.e. not be present.

For an `IsEncrypted` flag of 0x1 (AES-CTR), if the `IV_size` field is 16 then `InitializationVector` specifies the entire 128-bit IV value used as the counter value. If the `IV_size` field is 8, then its value is copied to bytes 0 to 7 of the counter value and bytes 8 to 15 of the counter value are set to zero. The `IV_size` field shall not be 0 when the `IsEncrypted` flag is 0x1 (AES-CTR).

For an `IsEncrypted` flag of 0x1 (AES-CTR), counter values shall be unique per `KID`. If an `IV_size` of 8 is used, then the `InitializationVector` values for a given `KID` shall be unique for each sample in all tracks and samples shall be less than 2^{64} blocks in length. If an `IV_size` of 16 is used, then initialization vectors shall have large enough numeric differences to prevent duplicate counter values for any encrypted block using the same `KID`.

`subsample_count` specifies the number of subsample encryption entries present for this sample. If present this field shall be greater than 0.

`BytesOfClearData` specifies the number of bytes of clear data at the beginning of this subsample encryption entry. (Note: this value may be zero if no clear bytes exist for this entry.)

`BytesOfEncryptedData` specifies the number of bytes of encrypted data following the clear data. (Note: this value may be zero if no encrypted bytes exist for this entry.) The subsample encryption entries shall not include an entry with a zero value in both the `BytesOfClearData` field and in the `BytesOfEncryptedData` field unless the sample is zero bytes in length. The total length of all `BytesOfClearData` and `BytesOfEncryptedData` for a sample shall equal the length of the sample. Further, it is recommended that the subsample encryption entries be as compactly represented as possible. For example, instead of two entries with {15 clear, 0 encrypted}, {17 clear, 500 encrypted} use one entry of {32 clear, 500 encrypted}

9.3 Initialization Vectors

The initialization vector (IV) values for each sample are located in the Sample Auxiliary Information associated with the encrypted samples. See 9.2 for details on how initialization vectors are formed and stored.

It is recommended that applications applying encryption randomly generate the initialization vector for the first sample in the track using a cryptographically sound random number generator.

- For 64-bit (8-byte) IV_Sizes, initialization vectors for subsequent samples can be created by incrementing the initialization vector of the previous sample. Using a random starting value introduces entropy into the initialization vector values and incrementing for each sample processed ensures that each IV value is unique. The 64-bit initialization vector should be allowed to roll over from the maximum value (0xFFFFFFFFFFFFFFFF) to the minimum value (0x0) if the random starting position is close to the maximum value.
- For 128-bit (16-byte) IV_Sizes, initialization vectors for subsequent samples should be created by adding the block count of the previous sample to the initialization vector of the previous sample. Using a random starting value introduces entropy into the initialization vector values and incrementing by the block count of the previous sample ensures that each IV value is unique. Even though the block counter portion of the counter (bytes 8 to 15) is treated as an unsigned 64-bit number by the client as described in 9.4, it is recommended that the initialization vector is treated as a 128-bit number when calculating the next initialization vector from the previous one.

9.4 Counter Operation

AES-CTR mode is a block cipher that can encrypt arbitrary length data without need for padding. It operates by encrypting a counter block with the AES algorithm and then XOR-ing the output of AES with the data to encrypt or decrypt. The counter block used is constructed as described in 9.2. Of the 16 byte counter block, bytes 8 to 15 (i.e. the least significant bytes) are used as a simple 64 bit unsigned integer that is incremented by one for each subsequent block of sample data processed and is kept in network byte order. Note that if this integer reaches the maximum value (0xFFFFFFFFFFFFFFFF) in the case where a 128-bit (16-byte) IV_size is used, then incrementing it resets the block counter to zero (bytes 8 to 15) without affecting the other 64 bits of the counter (i.e. bytes 0 to 7).

9.5 Full Sample Encryption

In full sample encryption, the entire sample is encrypted. Figure 1 shows sample-based encryption using AES-CTR mode.

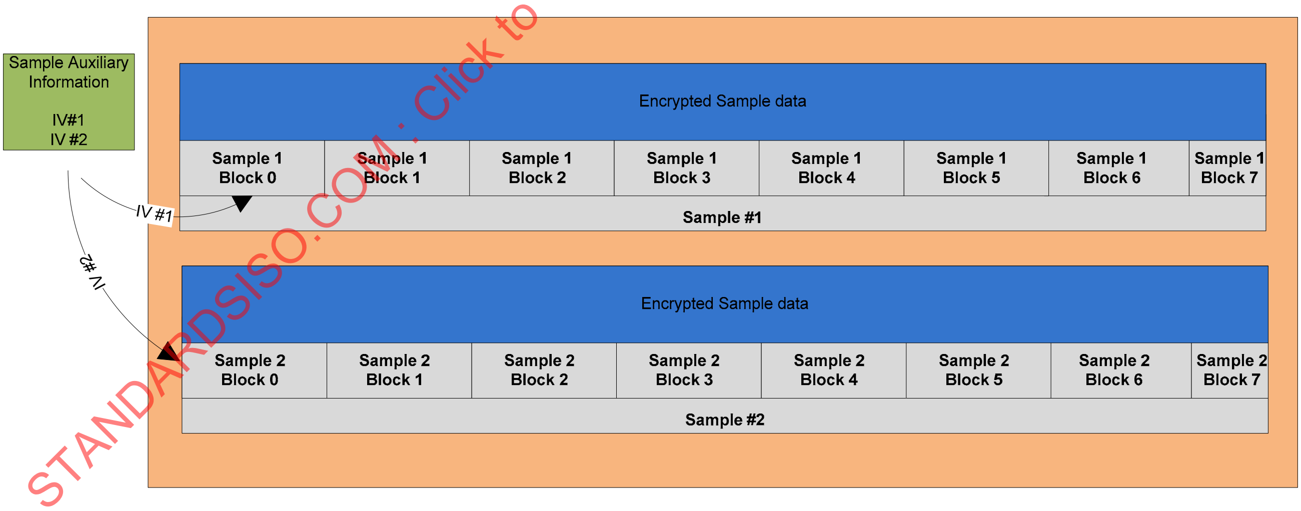

Figure 1 — Sample-based encryption for AES-CTR

Note that AES-CTR mode is a block cipher mode that acts like a stream cipher. Blocks are shown to illustrate the underlying cipher blocks used in generating the stream cipher (this is why Block 7 is shown as only partially used, as the unused bytes of the stream cipher are discarded during the encryption process).

9.6 Subsample Encryption

9.6.1 Definition

In subsample encryption, the sample is divided into one or more subsamples. Each subsample may have an unencrypted part followed by an encrypted part. The total length of all of the subsamples (BytesOfClearData + BytesOfEncryptedData for each subsample) that make up a sample shall be equal to the size of the sample itself.

The encrypted regions of a sample are treated as a logically contiguous block, even though they are broken up by areas of clear data. In other words, the block counter is not arbitrarily incremented between NAL units. Figure 2 shows Subsample based encryption using AES-CTR.

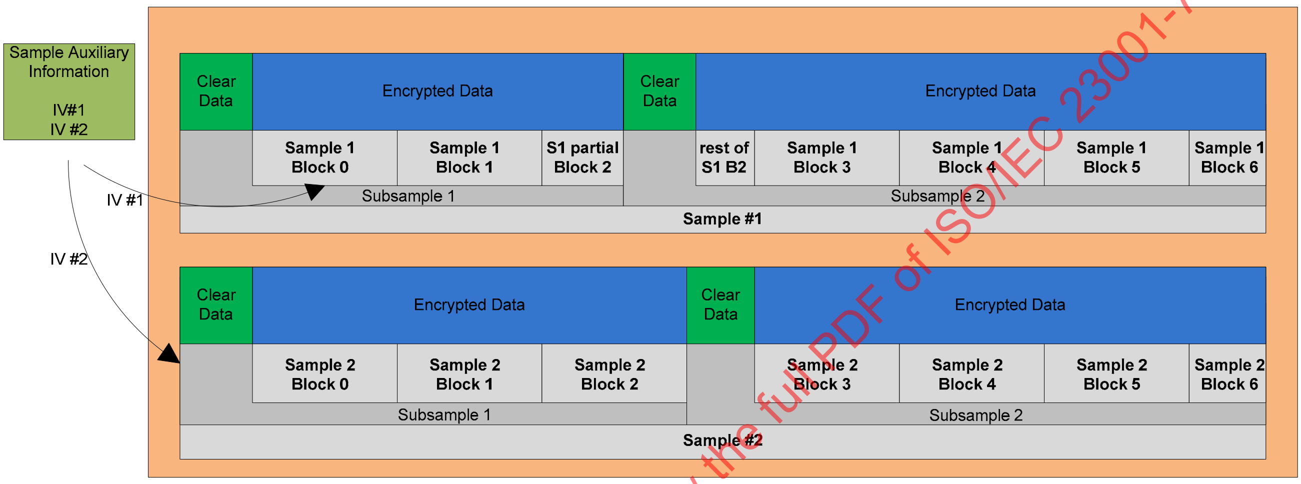

Figure 2 — Subsample-based encryption scheme for AES-CTR with IVs shown

Note that AES-CTR mode is a block cipher mode that acts like a stream cipher. Blocks are shown to illustrate the underlying blocks used in generating the stream cipher. This is why Block 6 in both Sample #1 and Sample #2 are not shown as full 16 byte blocks, the unused bytes of the stream cipher are discarded during the encryption process. Also note that Block 2 of Sample #1 is used to encrypt the end of the first subsample and the beginning of the second subsample.

9.6.2 Encryption of AVC tracks

Encrypted AVC tracks shall use subsample encryption as specified in the following Subclauses.

9.6.2.1 Structure of AVC video tracks

AVC specifies the building blocks of the AVC elementary stream to be Network Abstraction Layer (NAL) units. These units can be used to build AVC elementary streams for various different applications. ISO/IEC 14496-15 shall be used to define how the AVC elementary stream data is to be laid out in an ISO base media file format container. In the ISOAVC layout, the container level samples are composed of multiple NAL units, each separated by a Length field stating the length of the NAL. Figure 3 shows an AVC video sample distributed over several NAL units.