INTERNATIONAL STANDARD

ISO/IEC 15444-8

First edition
2007-04-15

AMENDMENT 1
2008-12-15

Information technology — JPEG 2000 image coding system: Secure JPEG 2000

AMENDMENT 1: File format security

Technologies de l'information — Système de codage d'images JPEG 2000: JPEG 2000 sécurisé

AMENDEMENT 1: Sécurité de format de fichier

Reference number
ISO/IEC 15444-8:2007/Amd.1:2008(E)

PDF disclaimer

This PDF file may contain embedded typefaces. In accordance with Adobe's licensing policy, this file may be printed or viewed but shall not be edited unless the typefaces which are embedded are licensed to and installed on the computer performing the editing. In downloading this file, parties accept therein the responsibility of not infringing Adobe's licensing policy. The ISO Central Secretariat accepts no liability in this area.

Adobe is a trademark of Adobe Systems Incorporated.

Details of the software products used to create this PDF file can be found in the General Info relative to the file; the PDF-creation parameters were optimized for printing. Every care has been taken to ensure that the file is suitable for use by ISO member bodies. In the unlikely event that a problem relating to it is found, please inform the Central Secretariat at the address given below.

COPYRIGHT PROTECTED DOCUMENT

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 by ISO in 2009

Published in Switzerland

CONTENTS

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.

Amendment 1 to ISO/IEC 15444-8:2007 was prepared by Joint Technical Committee ISO/IEC JTC 1, *Information technology*, Subcommittee SC 29, *Coding of audio, picture, multimedia and hypermedia information* in collaboration with ITU-T. The identical text is published as ITU-T Rec. T.807 (05/2006)/Amd.1(E).

INTERNATIONAL STANDARD ISO/IEC 15444-8
RECOMMENDATION ITU-T T.807

Information technology – JPEG 2000 image coding system: Secure JPEG 2000

Amendment 1

File format security

1) Clause 2: Normative references

Add the following references:

- Recommendation ITU-T T.803 (2002) | ISO/IEC 15444-4:2004, *Information technology – JPEG 2000 image coding system: Conformance testing*.
- ISO/IEC 13818-11:2004, *Information technology – Generic coding of moving pictures and associated audio information – Part 11: IPMP on MPEG-2 systems*.
- ISO/IEC 15444-6:2003, *Information technology – JPEG 2000 image coding system – Part 6: Compound image file format*.
- ISO/IEC 15444-12:2005, *Information technology – JPEG 2000 image coding system – Part 12: ISO base media file format (technically identical to ISO/IEC 14496-12)*.

2) Clause 3: Terms and definitions

a) *Rewrite the first paragraph as follows (with the changes underlined):*

For the purposes of this Recommendation | International Standard, the following definitions apply. The definitions defined in ITU-T Rec. T.800 | ISO/IEC 15444-1 clause 3 and ISO/IEC 15444-12:2005 clause 3 apply to this Recommendation | International Standard.

b) *Add the following terms and definitions:*

Normal decoder

Standard decoder is a process to decode a codestream that is fully compliant with the normative part of coding standard. Its behaviour is not defined if it tries to decode a non-compliant codestream.

Adaptive-format decoder

Adaptive-format decoder is a process to decode a codestream which is not fully compliant with the normative part of the coding standard. It shall reconstruct the media (possibly with low quality or resolution) even if the codestream has missing packets or inconsistent packet headers. For example, an adaptive-format decoder is able to understand a simply-transcoded codestream, such as the one that has its highest resolution packets removed.

Elementary Stream (ES)

Elementary streaming contains a sequence of samples, where each sample could be a video frame or a contiguous section of audio data. A sample in ES contains media data, ByteData structure, pointer structure, container structure, or any mixture of the above.

Self-Contained ES

Self-contained ES contains only media data, whose format is not defined in this amendment. The self-contained ES could be stored in MDAT box co-located with the file format specified in this amendment, or be stored in a separate file whose format is not specified by this amendment.

Composed ES

Composed ES may contain a mixture of ByteData, pointer and container structures, that is, its samples are composed with data from other elementary streams. A composed ES can either copy (using ByteData structure) or reference (using pointer) data from other ESes.

Scalable Composed ES

Scalable composed ES is made up of samples that may not be decodable by themselves. It may need to be combined with other scalable composed ESes to form a fully decodable codestream. Scalable composed ES is designed to support scalability, i.e., to make media data "thinable". For example, for a motion JPEG 2000 codestream where each picture has three layers, it can be divided into 3 scalable composed ESes: the first one consists of all layer 0 data, the second one consists of all layer 1 data and the third one consists of all layer 2 data.

Decodable Composed ES

Decodable composed ES is made up of samples that are decodable by themselves. It is designed for simple adaptation where the adaptor just needs to retrieve data pointed by pointer structure and remove the wrapper to form a fully scalable codestream. For example, for a motion JPEG 2000 codestream where each picture has three layers, it can form 3 decodable composed ESes: the first one consists of layer 0 data, the second one consists of layer 0 and layer 1 data and the third one consists of layer 0, 1 and 2 data.

Adaptor/transcoder

Adaptor/transcoder is a process to transform media data to lower scalability level, like lower resolution or lower quality or bit-rate, by removing portions of the file. The adaptor/transcoder can transform media data based on the information specified in this amendment. An adaptor/transcoder shall update byte offset values in file format parameters that are impacted by the process.

Secure adaptor/transcoder

Secure adaptor/transcoder is a process to transform encrypted or authenticated media data without necessity to decrypt or regenerate the MAC or signature. Thus, end-to-end security remains for the transcoded media data.

JPEG 2000-aware adaptor/transcoder

JPEG 2000-aware adaptor/transcoder combines one or more scalable composed ESes to form a fully decodable media codestream. It should have the capability to generate the headers and markers of media codestream and modify the packet index, such that the adapted codestream can be decoded by a normal decoder. It may also add empty packets to replace the removed ones, or it may insert POC marker.

Simple adaptor/transcoder

Simple adaptor/transcoder is able to transform data based on information specified by this amendment. It may not be capable of generating media headers or modifying packet indices. It simply retrieves data pointed by pointer structure and removes the wrappers, and the resulting codestream can be decoded by adaptive-format decoder, which can cope with missing packets and inconsistent headers.

Authentication adaptor/transcoder

An authentication adaptor/transcoder removes data that is not verifiable with the available media data and authentication data. For example, in a streaming system, some media packets may be lost during transmission. A file format receiver may reconstruct the received data to the best of its ability based on the available data. Then, an authentication adaptor/transcoder can determine which data can be verified, and then remove the packets that are not verified. The resulting file only contains the decodable, verified data.

Container

Container structure is used to wrap a sample in a composed ES. It might contain any number of ByteData or pointer structures, but is not allowed to contain another container structure.

Pointer

Pointer structure is used to reference a data segment in another ES. It must be contained inside a container structure.

ByteData

ByteData structure is used to wrap a data segment which is physically located in a composed ES. It must be contained inside a container structure.

4CC Code

4CC code is a 32-bit identifier, normally 4 printable characters. A 4CC code can be used to indicate the file type, the type of file format box, type of a file format track, type of a file format sample description and type of file format track reference. A 4CC code must be registered with a registration authority.

3) Annex E: File format security

Create a new annex and add the following text:

Annex E

File Format Security

(This annex forms an integral part of this Recommendation | International Standard)

E.1 Scope

This annex specifies JPSEC file format derived from the ISO base file format and modifications to JPEG family file format (including JP2, JPX and JPM) for protection and secure adaptation of scalable pictures, which is possibly encrypted and/or authenticated by the owner. The pictures could be either static pictures or time-sequenced pictures. In particular, this annex provides functionality to do the following:

- To store coded media data corresponding to different scalability levels. Elementary stream (ES) is used for this purpose. There are three types of ESes, self-contained ES, scalable composed ES and decodable composed ES.
- To define tracks describing the characteristics of the coded media data stored in ES. For example, the track should be able to indicate scalability level (resolution, layer, region, etc.) and the rate-distortion hints of the coded media data in order to facilitate easy and secure adaptation.
- To define new file format boxes to signal protection tools and parameters applied to coded media data or metadata. The protection tools can be applied to either static JPEG 2000 pictures or time-sequenced JPEG 2000 pictures.
- The protection tools defined in this amendment can be applied to JPEG family file formats including JP2, JPX and JPM and ISO-derived file formats such as MJ2 for motion JPEG.

E.2 Introduction

E.2.1 Security protection at file format level

This annex describes a JPSEC file format derived from the ISO base file format and modifications to JPEG family file format, to add security protection to JPEG 2000 pictures at the file format level. The protection applied at the file format level can be classified into two types: item-based protection and sample-based protection, both structures are defined by the ISO base file format. The item-based protection is designed to protect any byte ranges (including coded media data and metadata) while the sample-based protection is designed to protect time-sequenced media including JPEG 2000 pictures.

When the security tools applied change the data length, it shall update all pointers and length fields in all boxes, to ensure correct parsing by the reader.

E.2.2 Item-based protection

This annex describes two item-based protection schemes in the ISO base file format, by leveraging the syntax and structures specified by the JPSEC standard. Specifically, it describes schemes for decryption and authentication. Each item in the ItemLocationBox is protected by one or more protection schemes in the ItemProtectionBox. When multiple schemes are used (or chained together), the order in which they are applied may be significant and thus must be specified. This annex also specifies how such operations should be chained together. In addition, the

ItemDescriptionBox and ItemCorrespondingBox are added into the ISO base file format to allow the flexible processing properties that are provided by JPSEC. Specifically, the ItemDescriptionBox allows media-dependent metadata (such as resolution, quality layer, spatial region, and color space component) to be associated with different portions of the file. These descriptions can be provided regardless of whether protection is applied. When used with scalable coded pictures, this allows the file to be scaled down or transcoded without parsing or decoding the media data. In cases where protection is applied, this provides the benefit of enabling transcoding without requiring decryption.

E.2.3 Sample-based protection of scalable media

For time-sequenced pictures, this annex adds syntaxes to facilitate scalability at the file format level, including scalable composed elementary stream (ES), decodable composed ES, pointer structure, container structure and ByteData structure. The scalable coded pictures can be divided (either physically or virtually) into elementary streams at different scalability level, such that the adaptor/transcoder can "thin" media data with low complexity.

Figure E.1 gives an overview of the file format specified by this annex and also shows how the specified FF is used to adapt the media data.

Given a sequence of JPEG 2000 pictures (also referred to as *Self-contained ES*), there are two approaches to construct the file format. In the first approach, the MDAT box contains one or more *Scalable Composed ESes*, each of which corresponds to one scalability level of the media data, e.g., a resolution or a layer. The scalable composed ES must be stored in MDAT box that is co-located with the file format. The self-contained ES can be located in either MDAT box in the same file, or a different file whose format is not specified in this amendment. The scalable composed ES may not be decodable by itself, it may need to be combined with other scalable composed ESes to generate fully decodable JPEG 2000 pictures. In the second approach, the MDAT box contains one or more *Decodable Composed ESes*, and each ES constitutes fully decodable JPEG 2000 pictures by itself. Similarly, decodable composed ESes must be stored in MDAT box co-located with the file format, and the self-contained ES can be stored in either MDAT box in the same file, or a different file whose format is not specified by this annex.

Each scalable composed ES or decodable composed ES must be described by at least one track. The characteristics of the ES (like resolution, layer, and region) are indicated in SampleEntryBox inside each track.

To generate a fully decodable JPEG 2000 codestream from scalable composed ESes, a JPEG 2000-aware adaptor should have the capability to dynamically generate the image headers (based on the number of resolutions, layers and region in the adapted codestream), to insert empty packets or to insert POC markers as needed to make the resulting codestream decodable by any standard decoder. However, if a simple adaptor is used, the resulting codestream may have an inconsistent image header and there may be a missing packet, which require a JPEG 2000 adaptive-format decoder.

As a decodable composed ES is decodable by itself, a simple adaptor is sufficient to generate fully compliant JPEG 2000 pictures.

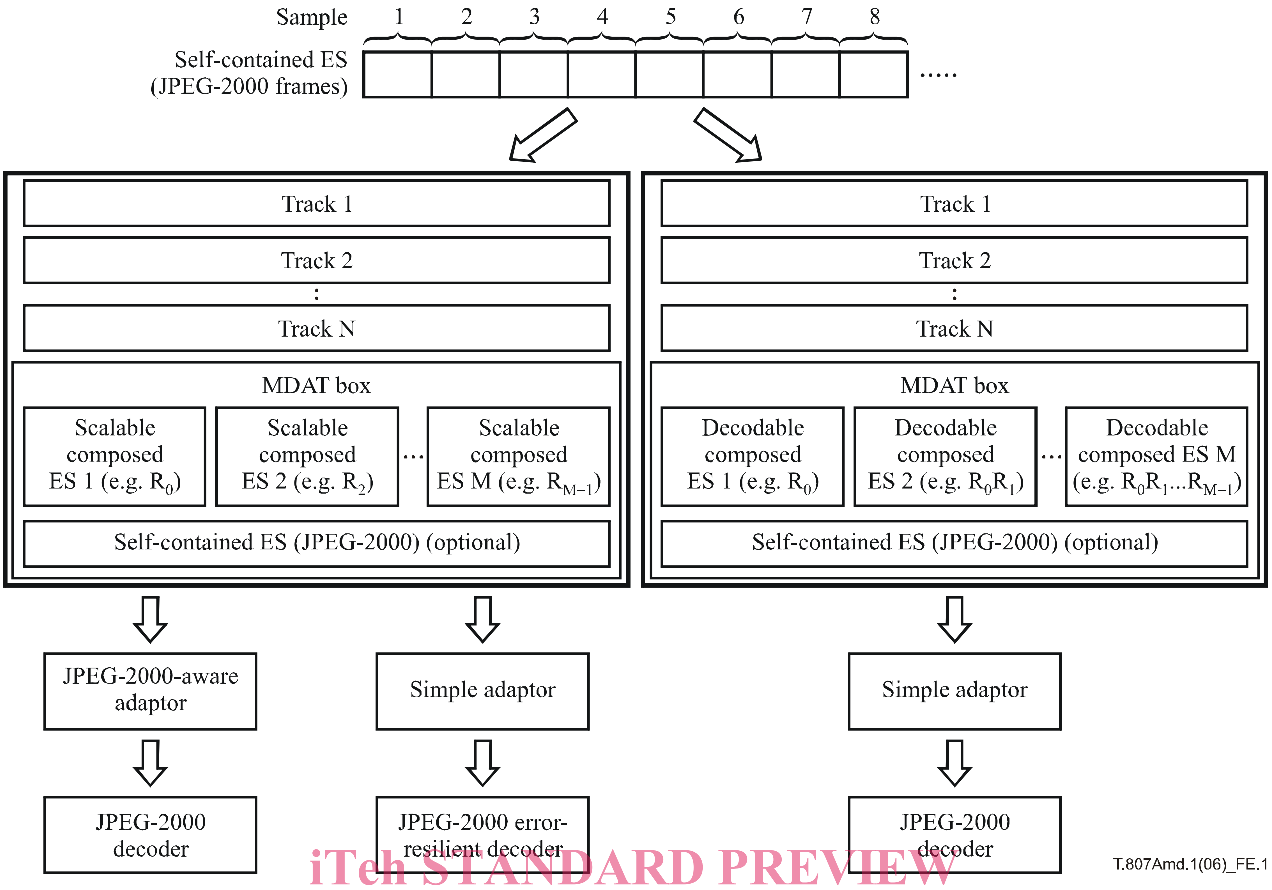
Sample
1
2
3
4
5
6
7
8
Self-contained ES (JPEG-2000 frames)
.....
Track 1
Track 2
:
Track N
MDAT box
Scalable composed ES 1 (e.g. R_0)
Scalable composed ES 2 (e.g. R_2)
...
Scalable composed ES M (e.g. R_{M-1})
Self-contained ES (JPEG-2000) (optional)
Track 1
Track 2
:
Track N
MDAT box
Decodable composed ES 1 (e.g. R_0)
Decodable composed ES 2 (e.g. R_0R_1)
...
Decodable composed ES M (e.g. $R_0R_1...R_{M-1}$)
Self-contained ES (JPEG-2000) (optional)
JPEG-2000-aware adaptor
Simple adaptor
Simple adaptor
JPEG-2000 decoder
JPEG-2000 error-resilient decoder
JPEG-2000 decoder
T.807Amd.1(06)_FE.1

Figure E.1 – System diagram for time-sequenced scalable media

Each elementary stream is described by at least one media track, and its characteristics are described in SampleDescriptionEntry or SampleGroupEntryBox within the track. It is possible that a single elementary stream is described by multiple tracks, each of which may describe different aspects of the elementary stream.

The sample-based protection can be applied to all samples or a group of samples in a scalable composed ES or decodable composed ES. If protection is applied to all samples, a ProtectionSchemeInfoBox signalling the parameters of the protection tool is added to the SampleDescriptionBox, which is then encapsulated as described in E.5.2. In addition, if protection is applied to a group of samples, a ProtectionSchemeInfoBox is added to their SampleGroupEntryBox, which is then encapsulated as described in E.5.4.

E.3 Extension to ISO base media file format

E.3.1 Overview

This subclause documents technical extensions (additional box types) to the ISO based media file format, which could be used for protection, adaptation, or secure adaptation of scalable coded pictures. However, the added box types could be used for other purposes as well. In particular, this subclause defines ProtectionSchemeInfoBox for the decryption tool and authentication tool, ItemDescriptionBox, ScalableSampleDescriptionEntry, ScalableSampleGroupEntry, and Generic Protected Box. All other boxes defined in ISO/IEC 15444-12 are still used as is.

E.3.2 Incorporate JPSEC codestream into ISO-driven file format

A JPSEC codestream can be placed as a payload in the 'mdat' box of the ISO base file format. In the Sample Description Box ('stsd'), the 'codingname' of the corresponding Sample Entry is defined to be 'jpsc', which is a registered identifier for JPSEC decoder. In this case, the security service is provided by JPSEC at codestream.

E.3.3 Protected file format brand

Files conforming to this Recommendation | International Standard may use 'ffsc' as the major brand in the File Type Compatibility Box.

Files conforming to this Recommendation | International Standard, i.e., containing protection or authentication information may use 'ffsc' as a compatible brand in the File Type Compatibility Box.

There are uses of this Recommendation | International Standard which are compatible with JP2, JPX, MJ2, and JPM files. A typical use of this Recommendation | International Standard will leave the major brand of a file unchanged, but add boxes and thus add 'ffsc' as a compatible brand.

Thus brands including *'isom'* , *'iso2'*, *'jp2\040*, *'jpx\040'* and *'jpm\040'* should be compatible.

The *'ffsc'* compatible brand indicates the use of new boxes and new tools corresponding to the protection methods in JPSEC.

A file that has been protected, to the extent that an application intending to process the JP2, JPX, JPM, or other file type content will be unable to do so without using protection tools, may use the 'ffsc' major brand as the file type; such a protected file must not use a major brand for which it is no longer conformant.

E.3.4 Summary of boxes used

The ISO base media file format defines two structures to describe a presentation: the logical structure and media sequence structure. The logical structure uses the ItemLocationBox ('iloc') to describe an item which is the byte range or a series of byte ranges for a particular file, either a local file or a remote file. The media sequence structure uses the SampleGroupDescriptionBox ('spgd') or SampleDescriptionBox ('stsd') to describe the samples, which could be a frame of video, a time-contiguous series of video frames, or a time-contiguous compressed section audio.

Accordingly, the protection in the ISO base media file format level is classified into item-based protection and sample-based protection, as described in E.5.2 and E.5.4, respectively.

Several boxes are used from ISO/IEC 15444-12, these are marked as "Existing" in Table E.1. Boxes defined in this Recommendation | International Standard are listed as "New" in Table E.1. The definitions for these boxes depend on the definitions of Box and FullBox from ISO/IEC 15444-12, which are repeated for convenience in E.7.

Table E.1 – List of existing and new boxes

Box names						**Status**	**Remarks**
meta						Existing	Metadata
	iloc					Existing	Item location
	iproc					Existing	Item protection
		sinf				Existing	Protection scheme information box
			frma			Existing	Original format box
			schm			Existing	Scheme type box
			schi			Existing	Scheme information box
				gran		New	Granularity box
				vall		New	Value List box
				bcip		New	Block cipher box
					keyt	New	Key template box
				scip		New	Stream cipher box
					keyt	New	Key template box
				auth		New	Authentication box
					keyt	New	Key template box
	iinf					Existing	Item information box
	ides					New	Item description box
		dest				New	Description type box
		desd				New	Description data box
		vide				New	Visual item description entry
		j2ke				New	JPEG 2000 item description entry
	icor					New	Item correspondence box
…	…	…	…	…	…	…	…
stbl						Existing	Sample table box
	stsd					Existing	Sample description box

Table E.1 – List of existing and new boxes

Box names						Status	Remarks
		ScalableSampleDescriptionEntry				New	Scalable sample description entry
	sbgp					Existing	Sample to group box
	sgpd					Existing	Sample group box
		ScalableSampleGroupEntry				New	Scalable sample group entry
gprt						New	Generic Protected box

E.3.5 Decryption scheme

The Decryption protection scheme is identified in SchemeTypeBox as follows:

```
scheme_type="decr"
scheme_version=0
scheme_uri=null
```

For the Decryption protection scheme, the structure of SchemeInfoBox is as follows:

```
aligned(8) class GranularityBox extends Box('gran') {
unsigned int(8) granularity;
}
```

Semantics:

granularity is used for item-based protection. For item-based protection, 0 indicates that the processing unit is the entire item and 1 indicates that the processing unit is one extent within an item. For sample-based protection, 0 indicates that the processing unit is all samples in the track or sample group and 1 indicates that the processing unit is one sample.

```
aligned(8) class ValueListBox extends Box('vall') {
      unsigned int(8) value_size;
      unsigned int(16) value_count;
      unsigned int(16) count[value_count];
      unsigned char (value_size) value[value_count];
}
```

Semantics:

value_size is the size in bytes of each value in the array.
value_count is the number of (count, value) pairs in the array. For item-based protection, the (count, value) pairs are used to map each value to count processing units. For sample-based protection, the (count, value) pairs are used to map each value to count samples. For instance, the value[0] corresponds to the first count[0] sample or units, and the value[1] corresponds to the next count[1] samples or units, and so on.

```
aligned(8) class KeyTemplateBox extends Box('keyt') {
      unsigned int(16) key_size;
      unsigned int(8) key_info;
      GranularityBox GL;            //optional
      ValueListBox    VL;
}
```

Semantics:

key_size is the size of key in bits.

key_info indicates the meaning of the values in the ValueListBox. 1 means the values are X.509 certificate; 2 means the values are URIs for certificate or secret keys.

GL is a GranularityBox.

VL is a ValueListBox, containing a list of values, whose meaning is defined by the key_info field.

```
aligned(8) class BlockCipherBox extends Box('bcip') {
unsigned int(16) cipher_id;
bit (6) cipher_mode;
bit (2) padding_mode;
unsigned int(8) block_size;
KeyTemplateBox KT;
}
```

Semantics:

cipher_id identifies which block cipher algorithm is used for protection. Values are defined in Table 25.

cipher_mode could be ECB, CBC, CFB, OFB or CTR. Values are defined in Table 29.

padding_mode is ciphertext stealing or PKCS#7-padding. Values are defined in Table 30.

block_size is size of block for block cipher.

KT is a KeyTemplateBox, holding all the key information used by the block cipher.

```
aligned(8) class StreamCipherBox extends Box('scip') {
unsigned int(8)  cipher_type;
unsigned int(16) cipher_id;
KeyTemplateBox KT;
}
```

Semantics:

cipher_type indicates the type of cipher used. It has values cipher_type = STRE for stream cipher or cipher_type = ASYM for asymmetric cipher.

cipher_id identifies the stream cipher algorithm used for the protection. If cipher_type = STRE, see Table 26; if cipher_type = ASYM, see Table 27.

KT is a KeyTemplateBox, holding all the key information used by the stream cipher.

```
aligned(8) class SchemeInfomationBox extends Box('schi', cipher_id) {
unsigned int(8) MetaOrMedia;
unsigned int(8) HeaderProtected;
      BlockCipherBox(); or    StreamCipherBox();
      GranularityBox GL;
      ValueListBox   VL;
}
```

Semantics:

MetaOrMedia is to indicate whether the protected data segment corresponds to media data segment or meta data boxes. (0 for media data and 1 for meta data boxes).

HeaderProtected is to indicate whether the protection is applied to the box content only (value 0) or applied to the whole box including its header (value 1).

The SchemeInformationBox can contain a BlockCipherBox, or a StreamCipherBox, which are containers for the parameters of the cipher algorithms. These boxes can only contain particular cipher_id values.

GL is a GranularityBox, holding information about the processing unit. This field is optional for sample-based protection and required for item-based protection.

VL is a ValueListBox. For block cipher, this box may be empty. For stream cipher, this box contains all the initial vectors.