DASH Content Protection using Microsoft PlayReady

Implementing Content Protection for Live and On-Demand Profiles of Dynamic Adaptive Streaming over HTTP (ISO/IEC 23009-1) using Common Encryption (ISO/IEC 23001-7) and Microsoft PlayReady

October 8, 2014

Version 1.2

Abstract:

The ISO/IEC 23009-1 ISO Base Media File Format On-Demand and Live DASH Profiles can be used with the ISO/IEC 23001-7, “Common Encryption in ISO base media file format files” specification. Microsoft PlayReady supports both ISO/IEC 23001-7 and ISO/IEC 23009-1. This specification details how to create an ISO/IEC 23009-1 Media Presentation Description file signaling the use of Microsoft PlayReady for ISO/IEC 14496-12 media representations for both Live and On-Demand adaptive streaming scenarios.

Legal Notice

Contents

Tables

DASH CONTENT PROTECTION USING MICROSOFT PLAYREADY

VERSION 1.2

OCTOBER 8, 2014

1 INTRODUCTION

The MPEG's Dynamic Adaptive Streaming over HTTP standard [DASH] specifies formats for the delivery of media content from HTTP servers to HTTP clients. In DASH the presentation of media content is described by a Media Presentation Description (MPD) file. The MPD provides resource identifiers for Segments along with context for these resources within a Media Presentation.

In DASH a Media Presentation consists of a time sequence of Periods ([DASH], section 5.3.2). Within a Period, media content is arranged into a set of interchangeable encoded versions called Adaptation Sets ([DASH], section 5.3.3). Each Adaptation Set consists of Representations ([DASH], section 5.3.5) - deliverable encoded versions of the media content components.

A ContentProtection Descriptor element may be associated with an Adaptation Set or a Representation, to indicate the encryption scheme, and one or more ContentProtection Descriptor elements may be added to enable DRM license acquisition ([DASH], section 5.8.4.1).

1.1 SCOPE

How to use Microsoft PlayReady as the Content Protection scheme in an ISO/IEC 23009-1 DASH Media Presentation Description file.

1.2 CONVENTIONS

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in [RFC2119]. That is:

- "MUST", "REQUIRED" and "SHALL" mean that the definition is an absolute requirement of the specification.
- "MUST NOT" and "SHALL NOT" mean that the definition is an absolute prohibition of the specification.

- "SHOULD" and "RECOMMENDED" mean that there may be valid reasons to ignore a particular item, but the full implications must be understood and carefully weighed before choosing a different course.
- "SHOULD NOT" and "NOT RECOMMENDED" mean that there may be valid reasons when the particular behavior is acceptable, but the full implications should be understood and the case carefully weighed before implementing any behavior described with this label.
- "MAY" and "OPTIONAL" means the item is truly optional.

1.3 TERMINOLOGY, ABBREVIATIONS AND ACRONYMS

1.3.1 TERMINOLOGY

Adaptation Set	In DASH, a set of interchangeable encoded versions of one or several media content components.
Content Protection (CP)	The process of securing a Protected Resource subsequent to its delivery to a Client device.
Embedded License	A License stored in the PlayReady header Object.
Embedded License Store (ELS)	A record in the PlayReady header Object for storing Embedded Licenses.
Globally Unique Identifier (GUID)	A unique reference number, represented as a hyphen separated 32-character hexadecimal string, and usually stored as a 128-bit integer.
Initialization Segment	A DASH Segment containing metadata necessary to present the media streams encapsulated in Media Segments; in the case of ISO Media, a file header.
Key Identifier (KID)	A UUID which uniquely identifies a key protecting content, licenses or other sensitive information; in the case of PlayReady, stored as a GUID.
Key Rotation	Periodic changes to the encryption key associated with media. Typically this means
Leaf License	A license whose content key is encrypted using a content key in a Root License.
License	A PlayReady data structure that includes policies and an encrypted content key.
License Acquisition URL (LAURL)	The License Acquisition PlayReady web service URL.

License Chain	A License Chain consists of a Root License and a Leaf License. A Leaf License may have multiple Root Licenses and a Root License may have multiple Leaf Licenses. A License Chain exists for each pair.
Live Profile	The ISO Base media file format live profile (see section 8.4 of [DASH]). The Live Profile is optimized for live encoding, where each movie fragment may be requested immediately after it is encoded using a template generated URL.
Media Presentation	Collection of metadata and media data that can be downloaded and rendered as a media presentation, as defined in ISO/IEC 23009-1.
Media Presentation Description (MPD)	Formal XML document description of a Media Presentation defined in ISO/IEC 23009-1.
Media Segment	A DASH Segment that complies with a media format and enables playback, perhaps combined with other Media Segments and/or an Initialization Segment.
Movie box ('moov')	In the ISO Base Media File Format, the box whose sub-boxes define the metadata for a media presentation [ISOBFF].
Movie Fragment box ('moof')	In the ISO Base Media File Format, the Movie Fragment box extends the media presentation in time [ISOBFF], and is contained in a DASH Media Segment.
On Demand Profile	The ISO Base media file format On Demand profile (see section 8.3 of [DASH]). The On Demand Profile provides basic support for On-Demand content. Each Representation is provided as a single Segment, Subsegments are aligned across an Adaptation Set's Representations, and Subsegments begin with a Stream Access Point corresponding to a movie fragment.
Period	Interval of a Media Presentation.
PlayReady header Object (PRO)	A binary object containing a variable number of records. These records contain information related to licenses and license acquisition.
Protection System Specific Header box ('pssh')	In the ISO Base Media File Format, the Protection System Specific Header box contains metadata needed by a specific Content Protection system to acquire a license and decrypt the media content [ISOBFF].

Representation	One of the media content component alternative choices during a defined Period, e.g. an ISO Media file. It is described by an MPD Representation element ([DASH], section 5.3.5).
Rights Management Header	A record in the PlayReady header Object containing metadata needed to decrypt the media content, including a Key ID and License Acquisition URLs (see [PRHEADER]).
Root License	A License whose content key is used to encrypt a content key in a Leaf License
Segment	In DASH, an element in an MPD that references a media resource with an HTTP-URL and optional byte range.
Segment Index	Time range to byte range index mapping within a Media Segment separate from the MPD, defined as an ISO Media 'sidx' box.
Stream Access Point (SAP)	The position in a Representation which enables Media Segment playback using only the Representation data from that position forward.
Subsegment	In DASH, this is a unit within a Media Segment indexed by a Segment Index. A movie fragment is addressed as a Subsegment in DASH On Demand Profile, but a Segment in DASH Live Profile.
Track Encryption box	In the ISO Base Media File Format, the Track Encryption box ('tenc') describes the default encryption parameters for a track [CENC], [ISOBFF].
UUID	A mathematically unique identifier represented as a number or string as specified in [X.667]
Video On Demand (VOD)	System enabling the End-user to select and watch video content on demand. Both the DASH Live Profile and DASH On Demand Profile can be used for VOD presentation.

1.3.2 ABBREVIATIONS AND ACRONYMS

CP	Content Protection
DASH	Dynamic Adaptive Streaming over HTTP
ELS	Embedded License Store
GUID	Globally Unique Identifier
KID	Key Identifier
LAURL	License Acquisition URL

MPD	Media Presentation Description
PRO	PlayReady header Object
SAP	Stream Access Point
UUID	Universally Unique Identifier
VOD	Video On Demand

1.4 REFERENCES

1.4.1 NORMATIVE REFERENCES

[CENC] *ISO/IEC FDIS 23001-7:2014, Second Edition, "Information technology – MPEG systems technologies – Part 7: Common encryption in ISO base media file format files".*

[DASH] *ISO/IEC 23009-1:2014, Second Edition, "Information technology — Dynamic adaptive streaming over HTTP (DASH) — Part 1: Media presentation description and segment formats", http://standards.iso.org/ittf/PubliclyAvailableStandards/c065274_ISO_IEC_23009-1_2014.zip*

[EME] *"Encrypted Media Extensions", RFC Working Draft http://www.w3.org/TR/encrypted-media/*

[PRHEADER] *"Microsoft PlayReady Header Object",* http://www.microsoft.com/playready/documents/

[RFC2119] "*Key words for use in RFCs to Indicate Requirement Levels*", S. Bradner, March 1997, http://www.ietf.org/rfc/rfc2119

[RFC3629] *"UTF-8, a transformation format of ISO 10646"*, F. Yergeau, November 2003, http://tools.ietf.org/html/rfc3629

[RFC4122] *"A Universally Unique IDentifier (UUID) URN Namespace", P. Leach, M. Mealling, R. Salz, July 2005, http://www.ietf.org/rfc/rfc4122.txt*

[X.667] *"Information technology – Open Systems Interconnection – Procedures for the operation of OSI Registration Authorities: Generation and registration of Universally Unique Identifiers (UUIDs) and their use as ASN.1 object identifier components" Series X ITU-T Recommendation http://www.itu.int/rec/T-REC-X.667-201210-I/en*

1.4.2 INFORMATIONAL REFERENCES

[CPSID] *DASH Industry Forum, "Protection System-specific Identifiers", http://dashif.org/identifiers/content-protection/*

[DASHIF] *DASH Industry Forum, "Guidelines for Implementation: DASH-AVC/264 Interoperability Points", August 15, 2013, Version 2.0,* http://dashif.org/w/2013/08/DASH-AVC-264-v2.00-hd-mca.pdf

[ISOBFF] *ISO/IEC 14496-12, Fourth Edition (Corrected version 2012-09-15), "Information technology – Coding of audio-visual objects – Part 12: ISO Base Media File Format",* http://standards.iso.org/ittf/PubliclyAvailableStandards/c061988_ISO_IEC_14496-12_2012.zip

1.5 CHANGE HISTORY

Version 1.1 18-Feb-13	Clarify PlayReady SystemID representation. The PlayReady SystemID is 9a04f079-9840-4286-ab92-e65be0885f95. Big endian representation = {0x9A, 0x04, 0xF0, 0x79, 0x98, 0x40, 0x42, 0x86, 0xAB, 0x92, 0xE6, 0x5B, 0xE0, 0x88, 0x5F, 0x95}. Little endian representation = {0x79, 0xF0, 0x04, 0x9A, 0x40, 0x98, 0x86, 0x42, 0xAB, 0x92, 0xE6, 0x5B, 0xE0, 0x88, 0x5F, 0x95}.
Version 1.2 8-Oct-14	1) Changes for CENC 2nd edition; 2) changes relating to the DASH MPD ContentProtection Descriptor elements; 3) changes in the Terminology for terms addition and clarifications; 4) changes in normative and informative references to refer the latest version; 5) clarify the KID representation and endianness in the ISOBFF boxes, PRO, and PlayReady license.

2 PLAYREADY DASH CONTENT PROTECTION SCHEME

Microsoft PlayReady supports the new ISO/IEC 23009-1 [DASH] and ISO/IEC 23001-7 [CENC] standards. This specification details how to create a DASH Media Presentation Description file signaling the use of Microsoft PlayReady for ISO Base Media File Format media representations, for both On Demand ([DASH], section 8.3) and Live ([DASH], section 8.4) adaptive streaming scenarios.

The four scenarios which are the focus of this specification are VOD or live presentations of media:

1. encrypted with a single key
2. where some content is encrypted and some content is in the clear
3. with Key Rotation without Embedded Leaf Licenses
4. with Key Rotation with Embedded Leaf Licenses

2.1 DASH CONTENTPROTECTION DESCRIPTOR ELEMENTS

DASH defines two types of ContentProtection Descriptor elements for ISO Media ([DASH] sections 5.3.7.2-Table 9, 5.8.5.2, and 5.8.4.1):

1. With @schemeIdUri=”urn:mpeg:dash:mp4protection:2011” @value=”<scheme>”
2. With @schemeIdUri=”urn:uuid:xxxxxxxx-xxxx-xxxx-xxxx-xxxxxxxxxxxx” @value=”DRMNAME version”

The first descriptor type indicates the four character code (“4CC”) of the encryption scheme that is contained in the Protection Scheme Information Box (‘schi’) in each encrypted ISO Media track. This descriptor type SHALL always be present so players are made aware that the content is encrypted, and can determine if they can decrypt the encryption scheme before attempting to download and play it.

The second descriptor type indicates the UUID string for a particular DRM system that can provide a license and decryption key for the associated Adaptation Set. Application specifications, such those by DASH Industry Forum, DVB, and DECE constrain all Representations in an Adaptation Set to share the same key(s) and license to enable seamless adaptive switching, which is accomplished by restricting ContentProtection Descriptors from the Representation level of the MPD. The UUID string is equal to the SystemID field specified by the ISO Media Protection System Specific Information Box (‘pssh’) that MAY be present in the Movie Box (‘moov’) of a file or DASH Initialization Segment.

SystemID values for DRM systems are registered at: http://dashif.org/identifiers/content-protection/

Each AdaptationSet element MAY list multiple DRM Descriptors to indicate that licenses are available for multiple DRM systems. Each DRM system can specify elements and attributes in their own namespace and make them optional or required in Descriptors using their SystemID.

The second edition of Common Encryption ([CENC] section 11) specifies an optional cenc:pssh element that can be used by all DRM ContentProtection Descriptors for improved interoperability. It also specifies a cenc:default_KID attribute recommended for inclusion in the mp4protection Descriptor to identify the license required in one place for all systems. The element cenc:pssh contains a complete ‘pssh’ box structure, so is processed identically by the Encrypted Media Extension (EME) API in web browsers [EME].

DRM systems can provide license acquisition information in:

1. A cenc:pssh element in a ContentProtection Descriptor in the MPD
2. A ‘pssh’ box in the ‘moov’ box of a file or Initialization Segment
3. Both of the above (in which case, the cenc:pssh element takes precedence)

Depending on workflow, it may be more efficient to embed license acquisition information in a ‘pssh’ box in every file in an Adaptation Set during encoding or packaging; or it may be more efficient to embed a cenc:pssh element in an MPD at the time a streaming presentation is offered. For live streaming, it is better to use cenc:pssh in the MPD to enable early license acquisition, rather than trigger many simultaneous license requests at the time the first Initialization Segment and ‘pssh’ box is simultaneously delivered to possibly millions of viewers.

License acquisition information in the MPD allows different streaming services to use different license servers, change them over time, etc. without creating and managing separate media files.

In the case of PlayReady, a PlayReady Header Object (PRO) [PRHEADER] can be contained in a cenc:pssh element, an mspr:pro element, or a 'pssh' box to enable license acquisition. The mspr:pro element is defined by Microsoft PlayReady, and includes only the PRO [PRHEADER] information, not the box structure included in 'pssh' and cenc:pssh. Including both mspr:pro and cenc:pssh will enable old players including a player based on Silverlight, and new players including web pages using script to play protected DASH presentations on HTML5 browsers.

2.1.1 CORRECT PRO IN INITIALIZATION SEGMENT OR MEDIA CONTENT

If there is an Initialization Segment containing the correct PRO, or if the media content includes a PlayReady 'pssh' box with the correct PRO, then the following ContentProtection Descriptor element with PlayReady Content Protection System-specific identifier [CPSID] SHOULD be used in an AdaptationSet element in an MPD to indicate the availability of a PlayReady license:

```
<ContentProtection schemeIdUri="urn:uuid:9a04f079-9840-4286-ab92-
e65be0885f95" value="MSPR 2.0"/>
```

PlayReady supports the Common Encryption [CENC] standard[1]. When the license acquisition metadata is stored in the Initialization Segment or the media content 'pssh', the PlayReady ContentProtection Descriptor element SHOULD be present.

The following ContentProtection Descriptor element SHALL be present in each protected Adaptation Set, and a single instance indicates the encryption scheme for all DRMs that support the 'cenc' scheme.

```
<ContentProtection schemeIdUri="urn:mpeg:dash:mp4protection:2011"
value="cenc"/>
```

2.1.2 INCLUDING A PLAYREADY HEADER OBJECT IN THE MPD

There are multiple situations where the PlayReady header Object (PRO) [PRHEADER] may need to be included in the PlayReady ContentProtection Descriptor element. For example:

- A presentation where the PlayReady 'pssh' box is absent
- A presentation where the PRO found in the PlayReady 'pssh' box needs to be overridden

To identify PlayReady as the Content Protection Scheme and include the PRO in the ContentProtection Descriptor element, it is recommended to use both cenc:pssh syntax according to [CENC] and mspr:pro syntax[2] for backward compatibility as follows:

```
<ContentProtection schemeIdUri="urn:uuid:9a04f079-9840-4286-ab92-
e65be0885f95" value="MSPR 2.0">
   <cenc:pssh>
```

[1] The 'pssh' box includes a SystemID, a UUID [X.667] that uniquely identifies the content protection system. The PlayReady SystemID is 9a04f079-9840-4286-ab92-e65be0885f95.

[2] Throughout this specification 'mspr' is 'urn:microsoft:playready', defined by xmlns:mspr="urn:microsoft:playready".

```
        <!-- base64-encoded PlayReady 'pssh' complete box -->
    </cenc:pssh>
    <mspr:pro>
        <!-- base64-encoded PlayReady object -->
    </mspr:pro>
</ContentProtection>
```

The following rule MUST be followed when including a PlayReady 'pssh' box and a PRO in the PlayReady ContentProtection Descriptor element:

- If cenc:pssh or mspr:pro is included in the PlayReady ContentProtection Descriptor element:
    - If a PRO is included in a Protection System Specific Header ('pssh') box in the **media content**, then a KID base64-encoded string value (a KID tag value of [PRHEADER]) in a PRO which is included in both cenc:pssh or mspr:pro SHALL be equivalent to the KID base64-encoded string value in that PRO which is included in 'pssh' box.
    - If there is an **Initialization Segment** for the Representation which contains a PlayReady 'pssh', then a KID base64-encoded string value in a PRO which is included in both cenc:pssh or mspr:pro for that Representation must SHALL be equivalent to the KID base64-encoded string value in that PRO which is included in 'pssh' box of the Initialization Segment. Note that an Initialization Segment may be a copy of the file header of a stored media content file contained in an HTTP response body, but it may also be dynamically generated and never stored as a file.
- If the media content contains a PlayReady 'pssh' box with a PRO containing a LAURL, and that LAURL differs from the LAURL in the PRO included in the ContentProtection Descriptor element, the ContentProtection Descriptor element LAURL SHALL take precedence.

2.1.3 INCLUDING TRACK ENCRYPTION BOX FIELDS IN THE MPD

Common Encryption indicates the key used to encrypt each media sample (or not used) with key identifier(s) (KID) in each movie fragment. Some information, such as the default_KID in the Track Encryption Box ('tenc') is useful to signal in the MPD to determine what license is required.

There may be Media Presentation Periods which are unencrypted, followed by Periods which are encrypted. Media files and streams may include both encrypted and unencrypted sections. In addition, the Key Identifier (KID) may change from Period to Period, or from section to section. Key changes in sections of a track are signaled using ISO Media sample group and sample group description boxes in each movie fragment. See [CENC] for more information on sample groups and access to Initialization Vectors and subsample encryption ranges using 'saio' and 'saiz' boxes pointing to Sample Auxiliary Information.

The default settings for Common Encryption information are encoded in each track's Track Encryption box ('tenc') (see [ISOBFF] and [CENC]) stored in a Track Box ('trak') in an ISO file header and Initialization Segment. The default Key Identifier (KID) in the Track Encryption Box (default_KID field) SHOULD also be communicated in the ContentProtection Descriptor element associated with the Media Presentation Adaptation Set. The cenc:default_KID may be used to identify a license that can decrypt the Media Segments referenced by the parent AdaptationSet element.

To include the default Key Identifier (KID) in the Common Encryption ContentProtection Descriptor element, use the following cenc:default_KID attribute specified in [CENC]:

```
<ContentProtection schemeIdUri="urn:mpeg:dash:mp4protection:2011"
value="cenc" cenc:default_KID="da9b5994-600c-2ad0-f96d-
f12725780978"/>
```

The default_KID field in 'tenc' is a big endian array of 16 bytes, and is defined above to be stored in the cenc:default_KID attribute in Common Encryption ContentProtection Descriptor element as a UUID string.

When a ContentProtection Descriptor element refers to several tracks, and these use different default Key Identifiers in different 'tenc' boxes, the cenc:default_KID attribute SHALL store a space-delimited list of those different default_KID values.

The cenc:default_KID attribute MAY also be contained in the PlayReady ContentProtection Descriptor element.

Table 1 below lists the fields specified in the ISO Media Track Encryption Box ('tenc') (see section 9.2, [CENC]). The elements below **Table 1** where defined in the "mspr" namespace for the first edition of Common Encryption (mspr:IsEncrypted, mspr:IV_size, and mspr:kid), but are deprecated and functionally replaced by cenc:default_KID specified in the second edition of Common Encryption [CENC]. The IV_size and IsEncrypted fields in the Track Encryption Box ('tenc') are used during decryption, but are not needed in MPD ContentProtection Descriptor elements.

Table 1 – Track Encryption Box Fields

Element	Default	Description
default_IsEncrypted	1	Flag indicating the encryption status of the samples in the sample group. Allowed values are 0 (not encrypted) and 1 (encrypted).
default_IV_size	8	The size in bytes of the InitializationVector field. Supported values are 0, 8 and 16. If default_IsEncrypted =1, default_IV_size MUST NOT be set to 0. Since not all PlayReady enabled players support 16 byte Initialization Vectors, it is RECOMMENDED that only an default_IV_size of 8 be used for encrypted content.

Element	Default	Description
default_KID	*None*	16 Byte Key identifier that uniquely identifies the key needed to decrypt the associated samples. The key identifier is treated as UUID according to [CENC] (Please note that unlike the KID in PlayReady Header Object [PRHEADER], 'tenc' default_KID is stored as a 16 byte array containing a big endian byte ordered, 128-bit integer equivalent to binary (section 6.2) and number (section 6.3) UUID representations specified in [X.667]).

To identify PlayReady as the Content Protection Scheme and include Track Encryption Box fields in the MPD, use the following syntax[3]:

```
<ContentProtection schemeIdUri="urn:uuid:9a04f079-9840-4286-ab92-
e65be0885f95" value="MSPR 2.0">
  <mspr:IsEncrypted>1</mspr:IsEncrypted>
  <mspr:IV_size>8</mspr:IV_size>
  <mspr:kid>2ptZlGAMKtD5bfEnJXgJeA==</mspr:kid>
</ContentProtection>
```

The following rules MUST be followed when including cenc:default_KID attribute in the Common Encryption ContentProtection Descriptor element and Track Encryption Box fields in the PlayReady ContentProtection Descriptor element:

- The KID string in cenc:default_KID attribute in the Common Encryption ContentProtection Descriptor element SHALL be equivalent to the default_KID UUID in the Track Encryption box for that track.
- The mspr:IsEncrypted, mspr:IV_size, and/or mspr:kid value in the PlayReady ContentProtection Descriptor element SHALL be equivalent those found in default_IsEncrypted, default_IV_size, or default_KID fields in the Track Encryption box for that track.[4]
- If mspr:pro and cenc:pssh is included in the PlayReady ContentProtection Descriptor element (see section 2.1.2) with Track Encryption Box fields:
    - The KID embedded in PRO in mspr:pro and cenc:pssh elements SHALL be equivalent to the default_KID in the Track Encryption Box of that track.
    - The KID in the cenc:default_KID attribute in the Common Encryption ContentProtection Descriptor element and mspr:kid value in the PlayReady

[3] KID value in mspr:kid is a base64-encoded little endian GUID interpretation of the 'tenc' default_KID byte array defined to store a big endian UUID.. All fields from the Track Encryption Box may be copied to the MPD, but only default_KID is necessary to expose for license acquisition.
[4] Note that the format of the default_KID in the Track Encryption Box is different than the format of the KID value embedded in PRO in cenc:pssh, mspr:pro, and mspr:kid. See section 2.2.5, [CENC], and [PRHEADER] for details.

ContentProtection Descriptor elementShall be equivalent to the default_KID in the Track Encryption Box of that track.[5]

2.2 IMPLEMENTATION RECOMMENDATIONS AND REQUIREMENTS

The PlayReady header Object (PRO) [PRHEADER] MAY be included in the encoded media Protection System Specific Header box ('pssh') [ISOBFF], the Initialization Segment or encoded in the MPD itself.

A 'pssh' box may be inserted in the Movie box ('moov') or the Movie Fragment box ('moof'). For example, a 'pssh' box may be inserted in the 'moov' box to enable the use of Initialization Segments ([DASH], section 5.3.9.5.2). A 'pssh' box may be inserted in each 'moof' box to convey Leaf Licenses indexed by KID for key rotation.

Sample Group and Sample Auxiliary Information SHALL be stored within any movie fragment that references it. A Sample Group Description Box ('sgdp') SHALL be present in each 'moof' box when a Sample To Group Box ('sbgp') is present. Sample Auxiliary Information Offset Box ('saio') and Sample Auxiliary Information Size Box ('saiz') SHALL be present in each movie fragment of every track containing a Track Encryption Box ('tenc'), with valid pointers to Sample Auxiliary Information.

2.2.1 GENERAL

The PlayReady ContentProtection Descriptor element may be associated with Adaptation Sets or their Representations, but for seamless adaptive bitrate switching and conformance with DASH application specifications, the PlayReady ContentProtection Descriptor element SHALL be contained in an AdaptationSet element rather than Representation elements. This insures that a single license and decryptor configuration can be used for all Segments in an Adaptation Set.

It is RECOMMENDED to include the PlayReady ContentProtection Descriptor and, REQUIRED to include the DASH MP4 ContentProtection Descriptor with a value of "cenc" in the MPD.

This enables DRMs which are "cenc" capable and have license acquisition information in an application, or in the MPD, or in a 'pssh' in the initialization segment to identify that the content is Common Encrypted, identify the license required (using default_KID), acquire a license, and decrypt the media.

It is RECOMMENDED to include the @value attribute with name and version "MSPR 2.0" in addition to the UUID in the PlayReady ContentProtection Descriptor for human recognition. For example:

```
<ContentProtection schemeIdUri="urn:mpeg:dash:mp4protection:2011"
value="cenc" cenc:default_KID="da9b5994-600c-2ad0-f96d-
f12725780978"/>
```

[5] Note that default_KID in the Track Encryption Box is interpreted as a binary or numeric UUID stored as a 16 byte big endian byte array, as a UUID hyphenated string in cenc:default_KID, and as a base64 string encoded from little endian byte order GUID representation in PRO and mspr:kid; but all must be alternative representations of the same UUID. See section 2.2.5.

```
<ContentProtection schemeIdUri="urn:uuid:9a04f079-9840-4286-ab92-
e65be0885f95" value="MSPR 2.0"/>
```

2.2.2 PRECEDENCE OF PRO LOCATION

When a client application finds a PRO in the MPD, it SHALL take precedence over a PRO contained in the Initialization Segment (see additional Initialization Segment PRO requirements in 2.2.3).

When a client application finds a PRO in the MPD, the Rights Management Header contained in that PRO SHALL take precedence over the Rights Management Header in a PRO contained in a 'pssh' box in the 'moov' box of the media content.

When a client application finds a PRO in the Initialization Segment, it SHALL take precedence over a PRO contained in a 'pssh' box in the 'moov' box header of the media content (see additional Initialization Segment PRO requirements in 2.2.3).

2.2.3 WHERE TO INCLUDE THE PRO

The PRO SHALL be present in either MPD or Initialization Segment. It MAY be present in both. In the MPD, it SHOULD be present in both an mspr:pro element and a cenc:pssh element in the PlayReady Content Protection Descriptor (mspr:pro for legacy players).

A PRO in the Initialization Segment or the MPD MAY include a Rights Management Header.

Whether using an Initialization Segment or not, it is RECOMMENDED that the MPD include the correct PRO, so that the Rights Management Header information can be acquired without downloading the Initialization Segment.

2.2.4 WHAT TO INCLUDE IN THE MPD PRO

The PRO may include the Rights Management Header and/or an Embedded License Store (ELS).

It is RECOMMENDED that the MPD PRO include the Rights Management Header.

It is NOT RECOMMENDED to include an ELS unless it is needed as part of a DRM Domain or a License Chain scheme.

2.2.5 KID BYTE ORDER

The Key identifiers (KIDs) in PlayReady are stored in a byte array formatted as a GUID (DWORD, WORD, WORD, 8-BYTE array) in little endian byte order, which is then base64 encoded for storage as a string in the PRO. Both PlayReady license servers and PlayReady clients in a PlayReady ecosystem expect that the KID in the PRO and PlayReady license is a little-endian byte order representation of [GUID].

The equivalent KID can be represented as a UUID string ([X.667] section 6.4) or byte array containing a UUID in big endian byte order binary (section 6.2) or number (section 6.3) as specified in [X.667]. Common Encryption [CENC] and DASH use these representations in the cenc:default_KID attribute (6.4 string), Track Encryption Box ('tenc'), and 'seig' Sample Group Description Box ('sgpd') (6.2 binary or 6.3 number byte array).

As a result, unless there is a change the client must convert the endianness of the KIDs byte array in order to match it to the PlayReady license.

To convert the KIDs in cenc:default_KID attribute and [ISOBFF] boxes to PlayReady KID, use the following sample code:

```
// Create a PlayReady GUID from the KID value in ISOBFF box.
// Since the PlayReady Server always runs on an Intel processor,
// this will be a little endian representation.
// e.g. KID in Track Encryption Box is:
//      {f81d4fae-7dec-11d0-a765-00a0c91e6bf6}
byte[] tencKidBytes = new byte[] {
   0xf8, 0x1d, 0x4f, 0xae,
   0x7d, 0xec,
   0x11, 0xd0,
   0xa7, 0x65,
   0x00, 0xa0, 0xc9, 0x1e, 0x6b, 0xf6
};
Byte[] prKidBytes = new byte[16];

// Swap the endianness of the GUID value:
// - Reverse bytes 0 to 3,
// - swap bytes 4 and 5,
// - swap bytes 6 and 7, and
// - copy bytes 8-15 as-is without swapping
ConvertEndianness(tencKidBytes, prKidBytes);
Guid prKid = new Guid(prKidBytes);

void ConvertEndianness(byte[] original, byte[] guidBytes)
{
    System.Array.Copy(original, guidBytes, 16);
    Swap(ref guidBytes, 0, 3);
    Swap(ref guidBytes, 1, 2);
    Swap(ref guidBytes, 4, 5);
    Swap(ref guidBytes, 6, 7);
}

void Swap(ref byte[] bytes, int pos1, int pos2)
{
    byte temp = bytes[pos1];
    bytes[pos1] = bytes[pos2];
    bytes[pos2] = temp;
}
```

Table 2 illustrates an example of KID representation.

Table 2 – KID representation example

KID Parameter	Type	Representation
KID	UUID BE Hex Number	f81d4fae7dec11d0a76500a0c91e6bf6 Section 6.3 of [X.667]
cenc:default_KID attribute	UUID Hex String with hyphens	"f81d4fae-7dec-11d0-a765-00a0c91e6bf6" Section 6.4 of [X.667]
KID in ISOBFF boxes	UUID BE Byte Array	Hex representation is { 0xf8, 0x1d, 0x4f, 0xae, 0x7d, 0xec, 0x11, 0xd0, 0xa7, 0x65, 0x00, 0xa0, 0xc9, 0x1e, 0x6b, 0xf6 } Section 6.2 of [X.667]
KID in PRO	Base64 String of GUID LE Byte Array	"rk8d+Ox90BGnZQCgyR5r9g==" (Hex representation of the data before Base64 encoding is { 0xae, 0x4f, 0x1d, 0xf8, 0xec, 0x7d, 0xd0, 0x11, 0xa7, 0x65, 0x00, 0xa0, 0xc9, 0x1e, 0x6b, 0xf6 })
mspr:kid	Base64 String of default_KID Byte Array in 'tenc' box	"+B1Prn3sEdCnZQCgyR5r9g==" (Hex representation of the data before Base64 encoding is { 0xf8, 0x1d, 0x4f, 0xae, 0x7d, 0xec, 0x11, 0xd0, 0xa7, 0x65, 0x00, 0xa0, 0xc9, 0x1e, 0x6b, 0xf6 })
KID in PlayReady license	GUID LE Byte Array	Hex representation is { 0xae, 0x4f, 0x1d, 0xf8, 0xec, 0x7d, 0xd0, 0x11, 0xa7, 0x65, 0x00, 0xa0, 0xc9, 0x1e, 0x6b, 0xf6 }

3 MEDIA PRESENTATION DESCRIPTION EXAMPLE

3.1 CORRECT PRO IN INITIALIZATION SEGMENT OR MEDIA CONTENT

See section 2.1.1 above.

```
<?xml version="1.0" encoding="utf-8"?>
<MPD
  xmlns="urn:mpeg:DASH:schema:MPD:2011"
  xmlns:cenc="urn:mpeg:cenc:2013"
  minBufferTime="PT2.00S"
  profiles="urn:mpeg:dash:profile:isoff-live:2011"
  type="static">
  <Period>
    <AdaptationSet mimeType="audio/mp4">
      <ContentProtection
      schemeIdUri="urn:mpeg:dash:mp4protection:2011" value="cenc"
      cenc:default_KID="0b630844-cb17-496a-9700-3702e1d23ee2"/>
       <ContentProtection schemeIdUri="urn:uuid:9a04f079-9840-
4286-ab92-e65be0885f95" value="MSPR 2.0">
      </ContentProtection>
      <Representation bandwidth="134878" id="audio">
        <SegmentList duration="4000" timescale="1000">
          <Initialization sourceURL="audio/init.mp4"/>
          <SegmentURL media="audio/seg-0000.m4f"/>
          <SegmentURL media="audio/seg-0001.m4f"/>
          <SegmentURL media="audio/seg-0002.m4f"/>
        </SegmentList>
      </Representation>
    </AdaptationSet>
  </Period>
</MPD>
```

3.2 INCLUDING A PLAYREADY HEADER OBJECT IN THE MPD

See section 2.1.2 above.

```
<?xml version="1.0" encoding="utf-8"?>
<MPD
  xmlns="urn:mpeg:DASH:schema:MPD:2011"
  xmlns:cenc="urn:mpeg:cenc:2013"
  xmlns:mspr="urn:microsoft:playready"
  minBufferTime="PT4.00S"
  profiles="urn:mpeg:dash:profile:isoff-live:2011"
  type="static">
  <Period>
    <AdaptationSet mimeType="audio/mp4">
      <ContentProtection
      schemeIdUri="urn:mpeg:dash:mp4protection:2011" value="cenc"
      cenc:default_KID="0b630844-cb17-496a-9700-3702e1d23ee2"/>
```

```
      <ContentProtection schemeIdUri="urn:uuid:9a04f079-9840-
4286-ab92-e65be0885f95" value="MSPR 2.0">
        <cenc:pssh>AAAAAJoE8HmYQEKGq5LmW+CIX5UAAALq6gIAAAEAAQDgAj
        wAVwBSAE0ASABFAEEARABFAFIAIAB4AG0AbABuAHMAPQAiAGgAdAB0AHA
        AOgAvAC8AcwBjAGgAZQBtAGEAcwAuAG0AaQBjAHIAbwBzAG8AZgB0AC4A
        YwBvAG0ALwBEAFIATQAvADIAMAAwADcALwAwADMALwBQAGwAYQB5AFIAZ
        QBhAGQAeQBIAGUAYQBkAGUAcgAiACAAdgBlAHIAcwBpAG8AbgA9ACIANA
        AuADAALgAwAC4AMAAiAD4APABEAEEAVABBAD4APABQAFIATwBUAEUAQwB
        UAEkATgBGAE8APgA8AEsARQBZAEwARQBOAD4AMQA2ADwALwBLAEUAWQBM
        AEUATgA+ADwAQQBMAEcASQBEAD4AQQBFAFMAQwBUAFIAPAAvAEEATABHA
        EkARAA+ADwALwBQAFIATwBUAEUAQwBUAEkATgBGAE8APgA8AEsASQBEAD
        4AUgBBAGgAagBDAHgAZgBMAGEAawBtAFgAQQBEAGMAQwA0AGQASQArADQ
        AZwA9AD0APAAvAEsASQBEAD4APABDAEgARQBDAEsAUwBVAE0APgBxAGgA
        SwBXAEgASgBhAEwAMAAxAEkAPQA8AC8AQwBIAEUAQwBLAFMAVQBNAD4AP
        ABMAEEAXwBVAFIATAA+AGgAdAB0AHAAOgAvAC8AcABsAGEAeQByAGUAYQ
        BkAHkALgBkAHkAbgBkAG4AcwAuAG8AcgBnAC8AYwBvAG4AdABvAHMAbwB
        zAHMAcAByAC8AcgBpAGcAaAB0AHMAbQBhAG4AYQBnAGUAcgAuAGEAcwBt
        AHgAPAAvAEwAQQBfAFUAUgBMAD4APABEAFMAXwBJAEQAPgBpAEsARwBsA
        FcARwA0AEQAWABVAHEANAB3AGIAVwBnAFIATgBMAFIASgBnAD0APQA8AC
        8ARABTAF8ASQBEAD4APAAvAEQAQQBUAEEAPgA8AC8AVwBSAE0ASABFAEE
        ARABFAFIAPgA=</cenc:pssh>
        <mspr:pro>6gIAAAEAAQDgAjwAVwBSAE0ASABFAEEARABFAFIAIAB4AG0
        AbABuAHMAPQAiAGgAdAB0AHAAOgAvAC8AcwBjAGgAZQBtAGEAcwAuAG0A
        aQBjAHIAbwBzAG8AZgB0AC4AYwBvAG0ALwBEAFIATQAvADIAMAAwADcAL
        wAwADMALwBQAGwAYQB5AFIAZQBhAGQAeQBIAGUAYQBkAGUAcgAiACAAdg
        BlAHIAcwBpAG8AbgA9ACIANAAuADAALgAwAC4AMAAiAD4APABEAEEAVAB
        BAD4APABQAFIATwBUAEUAQwBUAEkATgBGAE8APgA8AEsARQBZAEwARQBO
        AD4AMQA2ADwALwBLAEUAWQBMAEUATgA+ADwAQQBMAEcASQBEAD4AQQBFA
        FMAQwBUAFIAPAAvAEEATABHAEkARAA+ADwALwBQAFIATwBUAEUAQwBUAE
        kATgBGAE8APgA8AEsASQBEAD4AUgBBAGgAagBDAHgAZgBMAGEAawBtAFg
        AQQBEAGMAQwA0AGQASQArADQAZwA9AD0APAAvAEsASQBEAD4APABDAEgA
        RQBDAEsAUwBVAE0APgBxAGgASwBXAEgASgBhAEwAMAAxAEkAPQA8AC8AQ
        wBIAEUAQwBLAFMAVQBNAD4APABMAEEAXwBVAFIATAA+AGgAdAB0AHAAOg
        AvAC8AcABsAGEAeQByAGUAYQBkAHkALgBkAHkAbgBkAG4AcwAuAG8AcgB
        nAC8AYwBvAG4AdABvAHMAbwBzAHMAcAByAC8AcgBpAGcAaAB0AHMAbQBh
        AG4AYQBnAGUAcgAuAGEAcwBtAHgAPAAvAEwAQQBfAFUAUgBMAD4APABEA
        FMAXwBJAEQAPgBpAEsARwBsAFcARwA0AEQAWABVAHEANAB3AGIAVwBnAF
        IATgBMAFIASgBnAD0APQA8AC8ARABTAF8ASQBEAD4APAAvAEQAQQBUAEE
        APgA8AC8AVwBSAE0ASABFAEEARABFAFIAPgA=</mspr:pro>
      </ContentProtection>
      <Representation bandwidth="134878" id="audio">
        <SegmentList duration="4000" timescale="1000">
          <Initialization sourceURL="audio/init.mp4"/>
          <SegmentURL media="audio/seg-0000.m4f"/>
          <SegmentURL media="audio/seg-0001.m4f"/>
          <SegmentURL media="audio/seg-0002.m4f"/>
        </SegmentList>
      </Representation>
    </AdaptationSet>
  </Period>
</MPD>
```