GY

中华人民共和国广播电影电视行业标准

GY/T 277—2014

互联网电视数字版权管理技术规范

Technical specification of digital rights management for internet television

2014－05－06 发布　　2014－05－06 实施

国家新闻出版广电总局　发布

目　　次

前　言

本标准按照GB/T 1.1-2009给出的规则起草。

本标准由全国广播电影电视标准化技术委员会（SAC/TC 239）归口。

本标准起草单位：中央电视台、清华大学、国家新闻出版广电总局广播科学研究院、英特尔（中国）有限公司、央视国际网络有限公司、北京数字太和科技有限责任公司、北京永新视博数字电视技术有限公司、北京数码视讯科技有限公司、北京江南天安科技有限公司、中国科学院声学研究所（国家网络新媒体工程技术研究中心）、北京中科大洋科技发展股份有限公司、北京捷成世纪科技发展有限公司、数码辰星科技发展（北京）有限公司、UT斯达康（中国）有限公司、飞利浦上海研发中心、深圳国微技术有限公司、索尼（中国）有限公司、北京安视网信息技术有限公司、耐格如信（上海）软件技术服务有限公司、新奥特（北京）视频技术有限责任公司。

本标准主要起草人：丁文华、杨杰、赵黎、田忠、王磊、王兴军、张晶、苏永锋、梅雪莲、郭晓霞、郭沛宇、齐志峰、魏启任、刘璐、梁志坚、梅红兵、陈鹏飞、姜远航、曾学文、王付生、胡杰、李晖、赵于平、邢彩虹、汪诚、屈劲、陈普贵、赵涛、孙鹏、江南、白慧生、吴丽莎、刘毅、张大勇、支春霞。

引　言

本标准的发布机构提请注意，声明符合本标准时，第7章中关于权利描述结构和权利描述编码的规定，可能涉及相关专利的使用，相关专利的名称为“一种数字媒体内容保护方法及装置、服务器、终端”和“一种内容权利封装的方法”；第8章中关于权利获取协议框架、消息、及编码的规定等内容，可能涉及相关专利的使用，相关专利名称为“数字版权管理中数字化作品的权利对象描述和获取的方法”。

本标准的发布机构对于该专利的真实性、有效性和范围无任何立场。

参与本标准的上述所有起草单位已向本标准的发布机构保证，愿意同任何申请人在合理且无歧视的条款和条件下，就专利授权许可进行谈判。上述专利持有人的声明已在本标准的发布机构备案。

下表列出专利权利人的信息：

专利权利人	联系地址	联系人	邮政编码	电话	电子邮箱
广播科学研究院	北京市西城区复兴门外大街2号	李霄	100866	010-86098010	lixiao@abs.ac.cn
北京数字太和科技有限责任公司	北京市西城区西直门内南大安胡同6号中宏集团三层	阮平	100035	010-82357885	p.ruan@unitend.com
清华大学	北京市海淀区清华大学信息技术研究院（FIT楼）3-413	赵黎	100084	010-62796665	zhaoli@tsinghua.edu.cn

请注意除专利许可声明中已经识别出的专利外，本标准的某些内容有可能涉及其他专利。本标准的发布机构不应承担识别这些专利的责任。

互联网电视数字版权管理技术规范

1 范围

本标准规定了互联网电视内容分发数字版权管理的内容封装格式、权利描述与授权、权利获取协议、信任与安全体系的相关技术要求。

本标准适用于互联网电视内容分发过程中的版权保护。

2 规范性引用文件

下列文件对于本标准的应用是必不可少的。凡是注日期的引用文件，仅所注日期的版本适用于本标准。凡是不注日期的引用文件，其最新版本（包括所有的修改单）适用于本标准。

GY/T 260-2012 广播电视数字版权管理数字内容标识

GM/T 0015-2012 基于SM2密码算法的数字证书格式规范

ISO 14496-12 信息技术——音视频对象编码 第12部分：ISO基础媒体文件格式（Information technology—Coding of audio-visual objects-Part 12: ISO base media file format）

ISO 23001-7 信息技术——MPEG系统技术 第7部分：ISO基本媒体文件格式文件通用加密（Information technology—MPEG systems technologies—Part 7: Common encryption in ISO base media file format files）

ISO 23001-7 Amendment1 信息技术——MPEG系统技术 第7部分：ISO基本媒体文件格式文件通用加密 补充1：AES-CBC-128和密钥循环（Information technology—MPEG systems technologies—Part 7:Common encryption in ISO base media file format files AMENDMENT 1: AES-CBC-128 and key Rotation）

ISO 23009-1 信息技术——基于HTTP的动态自适应流媒体（DASH） 第1部分：媒体展现描述与分段格式（Information technology—Dynamic adaptive streaming over HTTP (DASH)—Part 1:Media presentation description and segment formats）

ISO 23009-4 信息技术——基于HTTP的动态自适应流媒体（DASH） 第4部分：分段加密与认证）（Information technology—Dynamic adaptive streaming over HTTP (DASH)—Part 4:Segment encryption and authentication）

ITU-T Recommendation X.509 (1997 E) 信息技术——开放系统互联-目录:认证框架(Information technology—Open systems interconnection-The directory: authentication framework）

RFC 2045 多目标INTERNET邮件扩展 第1部分：Internet消息主体格式（Multipurpose internet mail extensions—Part 1: Format of internet message bodies）

RFC 2459 Internet X.509公钥基础设施：证书和CRL简介（Internet X.509 public key infrastructure certificate and CRL profile）

NIST SP 800-38A 块加密模式的操作方式推荐（Recommendation for Block Cipher Modes of Operation）

RFC 2630 加密消息语法（Cryptographic Message Syntax）

RFC 2396 URI通用语法（URI Generic Syntax）

ECMA 404 JSON数据互交换格式（The JSON data interchange format）

3 术语和定义

下列术语和定义适用于本标准。

3.1

内容提供者 content provider
是指拥有数字媒体内容并提供带版权信息的数字媒体内容的功能实体。

3.2

采样单元 sample
是ISO基础媒体文件格式中一帧独立的视频，或一个时间连续的视频帧序列，或一个时间连续的压缩音频数据段。

3.3

许可证 license
对数字媒体内容访问权限、使用规则和密钥等控制信息的描述。

3.4

设备 device
安装有DRM代理的消费内容的实体。

3.5

DRM 代理 DRM agent
设备中的可信实体，负责执行与DRM内容相关的许可和限制。

3.6

DRM 服务端 DRM server
向DRM代理发送许可证的实体。

3.7

DRM 内容 DRM content
采用DRM技术管理的数字媒体内容。

3.8

明文 plaintext
未加密的信息。

3.9

密文 ciphertext
已加密的信息。

3.10

加密 encryption

为了产生密文，即隐藏数据的信息内容，由密码算法对数据进行（可逆）变换。

3.11

解密 decryption

与加密过程相对应的逆过程，即由密码算法对密文数据进行逆变换。

3.12

密钥 key

控制密码变换操作（例如：加密、解密、密码校验函数计算、签名生成或签名验证）的符号序列。

3.13

数字签名 digital signature

附加在数据单元上的一些数据，或是对数据单元所作的密码变换，用于验证数字信息来源的真实性和数据的完整性。

4 缩略语

下列缩略语适用于本标准。

AES 高级数据加密标准（Advanced Encryption Standard）
CBC 密码块链接（Cipher Block Chain）
ChinaDRM 中国数字版权管理（China Digital Rights Management）
CENC 通用加密（Common Encryption）
CTR 计数器（Counter）
DASH 用HTTP协议传输的动态自适应流媒体协议（Dynamic Adaptive Streaming over HTTP）
DER 区分编码规则（Distinguished Encoding Rules）
DRM 数字版权管理（Digital Rights Management）
DRM-CF DRM内容格式（DRM Content Format）
HLS 基于HTTP的实时流媒体协议（Http Live Streaming）
HTTP 超文本传输协议（Hyper Text Transport Protocol）
ISO 国际标准化组织（International Organization for Standardization）
MPD 媒体展现描述（Media Presentation Description）
OCSP 在线证书状态协议（Online Certificate Status Protocol）
PDCF 打包DRM内容格式（Packetized DRM Content Format）
PKI 公钥基础设施（Public Key Infrastructure）
uimsbf 无符号整数，高有效位优先（unsigned integer, most significant bit first）
URI 通用资源标识符（Uniform Resource Identifier）
URL 统一资源定位符（Uniform Resource Locator）
UTC 协调通用时间（Coordinated Universal Time）

5 概述

本标准面向互联网电视业务应用定义了内容封装格式、权利描述与授权、权利获取协议和信任与安全体系等数字版权管理基础格式、技术机制和方法，其中内容封装格式定义了受保护内容在数字版权保护系统中的基本呈现形式，包括内容标识、加密信息及获取许可证必须的信息；权利描述与授权定义了描述数字版权管理系统权利的方法和向内容使用方进行使用授权的技术机制；权利获取协议定义了DRM服务端和DRM代理进行安全通信和传递许可证的技术方法；信任与安全体系定义了基于PKI系统的信任技术机制，包括内容保密性、身份鉴别和数据完整性。

采用本标准定义的基础格式、技术机制和方法，能够构造一个端到端的互联网电视数字版权管理系统，对互联网电视内容进行有效的数字版权管理。

互联网电视数字版权管理系统的逻辑示意图如图1所示。

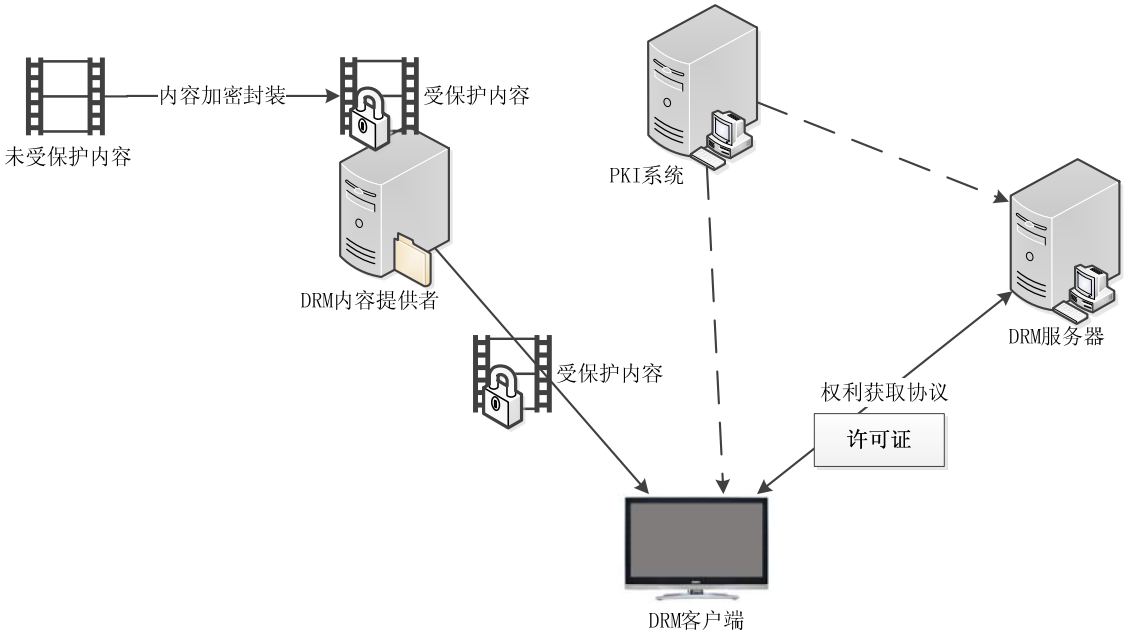

图1　互联网电视数字版权管理系统逻辑示意图

6　内容封装格式

6.1　概述

本章规定互联网电视内容分发数字版权管理技术体系的加密内容封装格式。加密后的内容应包含内容标识及获取许可证所必需的信息，其中内容标识应遵循GY/T 260-2012。根据不同的应用场景，需要定义不同的加密内容封装格式。

本标准既支持流式内容封装格式也支持存储类内容封装格式。5.2定义基于ISO基础媒体文件格式的打包DRM内容格式（PDCF），该打包DRM文件格式适用于DRM内容的存储。5.3和5.4定义的信息用以支持基于HTTP动态自适应流媒体协议（DASH）的内容封装格式、以及基于ISO基础媒体文件格式的通用加密格式（CENC）。

本标准也可支持其他主流格式，如HLS，参见附录A。

6.2　打包 DRM 内容格式

6.2.1　概述

打包DRM内容格式（PDCF）是在ISO基础媒体文件格式基础上定义的加密内容封装格式。PDCF内容格式在ISO 14496-12中规定的ProtectionSchemeInfoBox（‘sinf’）的SchemeInformationBox（‘schi’）中定义ChinaDRMKMSBox（‘cdkm’），ChinaDRMKMSBox规定ChinaDRM指定的结构和参数，如图2所示。PDCF内容格式的基本数据结构由相应的基础文件格式规范定义。

上述文件中的媒体数据是以采样单元为单位存储的，一个或多个采样单元组成一个访问单元，加密操作是对访问单元进行的，在加密过程中，应在每个加密的访问单元之前插入ChinaDRMAUHeader（见5.2.2.4）。

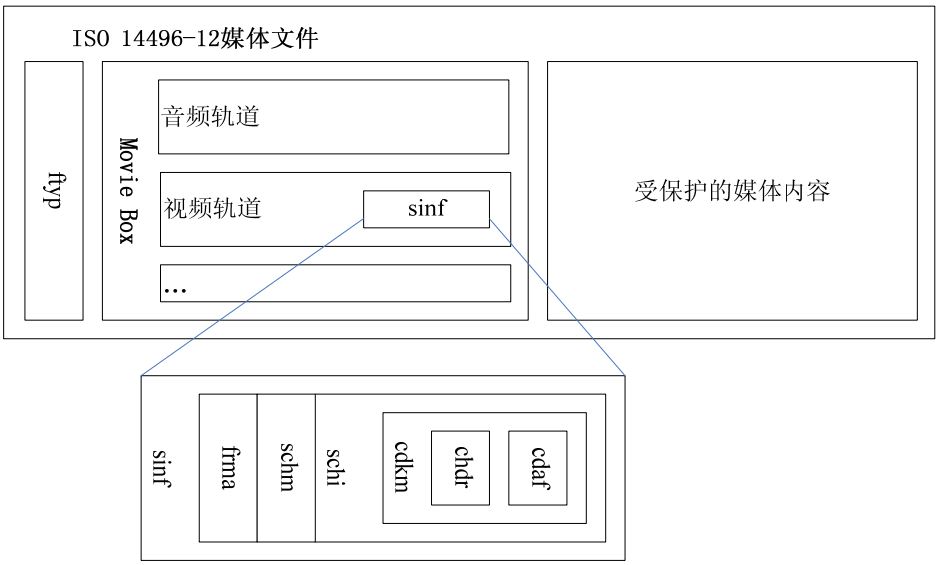

图2　PDCF 结构

6.2.2　打包 DRM 内容格式语法

6.2.2.1　SchemeTypeBox

SchemeTypeBox（‘schm’）的语法见ISO 14496-12。SchemeTypeBox中SchemeType应为‘cdkm’。SchemeVersion为ChinaDRM的版本号，当前为0x00000100，即1.0版本。

在SchemeType为‘cdkm’时，SchemeInformationBox（‘schi’）中应包含一个ChinaDRMKMSBox。

6.2.2.2　SchemeInformationBox

SchemeInformationBox（‘schi’）的语法见ISO 14496-12， SchemeInformationBox应仅包含一个ChinaDRMKMSBox。

6.2.2.3　ChinaDRMKMSBox

6.2.2.3.1　ChinaDRMKMSBox

在PDCF内容格式中，可能有一个或几个ChinaDRMKMSBox实例，如每个受保护的轨道均有一个。ChinaDRMKMSBox 中应包含一个 ChinaDRMCommonHeaders 和一个可选的 ChinaDRMAUFormatBox。ChinaDRMKMSBox的语法如下。

```
aligned(8) class ChinaDRMKMSBox extends FullBox('cdkm', version, 0) {
    ChinaDRMCommonHeaders        Headers;    // Common headers box
    ChinaDRMAUFormatBox          AUFormat;   // optional
```

```
}
```

6.2.2.3.2 ChinaDRMCommonHeaders

ChinaDRMCommonHeaders的语法如下，设备不应修改ChinaDRMCommonHeaders中的任意域。

```
aligned(8) class ChinaDRMCommonHeaders extends FullBox('chdr',version,0)
{
    unsigned int(8)     EncryptionMethod;     // Encryption method
    unsigned int(8)     PaddingScheme;        // Padding type
    unsigned int(64)    PlaintextLength;      // Plaintext content length in bytes
    unsigned int(16)    ContentIDLength;      // Length of ContentID field in bytes
    unsigned int(16)    DRMServerURLLength;   // DRM Server URL field length in bytes
    char                ContentID[];          // Content ID string
    char                DRMServerURL[];       // DRM Server URL string
    Box                 ExtendedHeaders[];    // Extended headers boxes
}
```

Version：定义了内容对象采用China DRM保护内容时盒子的版本号，符合本标准的对象之版本号应为0。

EncryptionMethod：定义了内容的加密方法，该字段的取值见表1。

表1 算法标识符的值

算法标识符	值	语义
NULL	0x00	对象不加密。NULL加密的内容对象不需获取版权对象即可使用。PaddingScheme 域的值应为0。在ChinaDRMAUFormatBox中，SelectiveEncryption和IVLength应均为0。
AES_128_CBC	0x01	NIST SP 800-38A定义的AES对称加密算法。 128比特密钥。 密码块链接模式（CBC）。 密文前缀为128比特初始向量，它包含于ChinaDRMAUHeader中，5.2.2.4中有对ChinaDRMAUHeader结构的详细说明。ChinaDRMAUFormatBox中的初始向量长度字段值必须设置为16，根据RFC2630填充。

表1（续）

算法标识符	值	语义
AES_128_CTR	0x02	NIST SP 800-38A定义的AES对称加密算法。 128比特密钥。 计数器模式（CTR）。计数块长度为128比特。 密文前缀为128比特初始向量，它包含于ChinaDRMAUHeader中，
		5.2.2.4中有对ChinaDRMAUHeader结构的详细说明。 ChinaDRMAUFormatBox中的初始向量长度字段值必须设置为16。 对于每个密码块，计数器增1（对2^{128}取模）。 无填充。

NULL加密的媒体对象有以下特性：

——NULL 加密的媒体对象不受机密性保护；

——NULL 加密的媒体对象一般可在无相关许可证条件下使用；

——NULL 加密的媒体对象不受完整性保护。

PaddingScheme：定义加密数据的填充机制，填充机制域的取值定义见表2。

表2　填充机制域的值

值	语义
0x00	不填充（例如，当采用 NULL 或 CTR 算法时）
0x01	根据 RFC2630 填充

PlaintextLength：定义明文内容的长度。若内容要进行加密，应设定一个明文长度值。当终端得到的内容实际长度同设定的明文长度值不匹配,将会出现错误并且丢弃内容对象。在正在下载的情景下，DRM代理只有在完整内容对象接收完毕后才能对明文长度值进行验证。

ContentIDLength：定义内容标识（ContentID）的长度，内容标识应遵循GY/T 260-2012，长度固定为8。

DRMServerURLLength：定义DRM服务端（DRMServerURL）域所占的字节数。设备应支持至少256字节的DRM服务端URL；不应使用大于256字节的DRM服务端URL。

ContentID：内容标识符应遵循GY/T 260-2012的定义。

DRMServerURL：定义DRM服务端的URL。DRM服务端URL可能用于设备获取DRM内容的许可证。域的长度由DRM服务端URL长度域给出。DRM服务端URL应符合RFC2396标准；在内容对象没有加密时，DRM服务端URL可以为空。

ExtendedHeaders[]:可选的扩展头盒子，本标准不做规定。

6.2.2.3.3　ChinaDRMAUFormatBox

ChinaDRMAUFormatBox用于规定媒体访问单元头部ChinaDRMAUHeader中数据的格式。

```
Aligned (8) class ChinaDRMAUFormatBox extends FullBox('cdaf', 0, 0) {
```

```
    bit (1)           SelectiveEncryption;
    bit (7)           reserved;
    unsigned int (8)  KeyIndicatorLength;
    unsigned int (8)  IVLength;
}
```

SelectiveEncryption：描述选择性加密算法的使用。在本标准中，这个比特应设为1；

IV length：以字节为单位描述初始向量的大小。其长度应与表1中指明的所用算法一致；

KeyIndicatorLength：以字节为单位描述密钥指示器的大小。在本标准中，KeyIndicatorLength的值为0。

6.2.2.4 ChinaDRMAUHeader

ChinaDRMAUHeader是位于每个采样单元数据之前的信息，用于规定每个采样单元的加密保护方式；其格式如下。

```
aligned (8) class ChinaDRMAUHeader {
    bit (1)          EncryptedAU;  // Encryption indicator
    bit (7)          reserved;     // Must be zero
    if (EncryptedAU==1) {
        unsigned int (8 * KeyIndicatorLength)   KeyIndicator;
        unsigned int (8 * IVLength)             IV;
    }
}
```

PDCF访问单元格式头见表3，加密指示器EncryptedAU的值见表4。

表3 PDCF 访问单元格式头

域名	类型	作用
EncryptedAU	bit(1)	访问单元的加密指示器
KeyIndicator	unsigned int(8 * KeyIndicatorLength)	本标准中，KeyIndicator的长度为0
IV	unsigned int(8 * IVLength)	IV数据

表4 加密指示器的值

值	语义
0	访问单元未加密
1	访问单元已加密

当加密PDCF内容时，ChinaDRMAUHeader的信息应加到处理过的访问单元中去。播放设备用头部信息来解密，从而能提取出实际的样本。

6.3 MPEG DASH

ISO 23009-4规定了MPEG DASH的分段加密与认证，MPEG DASH中媒体分块如果是加密的，其加密相关信息通过MPD文件中的Content Protection单元来指定。

基于本标准的互联网电视数字版权管理系统在使用ISO 23009-4 MPD文件描述时，在ContentProtection的@schemeIdUri属性中应指定由ChinaDRM申请的唯一编号；在ContentProtection

的@Value属性中应按“ChinaDRM版本/ChinaDRM方案提供商/扩展信息”的方式赋值，通过“/”进行区分，如“ChinaDRM V1.0/CompanyName/Extended Information”。

6.4 CENC

通用加密格式CENC是基于ISO/IEC 14496-12的一种加密格式，这种加密方式使不同的DRM系统能够解密同一个文件，CENC具体规定见ISO 23001-7及此文档的补充文档ISO 23001-7 Amendment1。

基于本标准的互联网电视数字版权管理系统在使用CENC通用加密格式时，应符合以下要求：

a） 在 ProtectionSystemSpecificHeaderBox （‘PSSH’)中，将 16 字节长度的 SystemID 的前 8 个字节设置为“ChinaDRM”，后 8 个字节保留为 0x00。

b） 在 PSSH 的 Data 部分，应包含获取许可证的 URL。

c） 对 TrackEncryptionBox (‘tenc’)中的 default_IsEncypted 或 SampleGroupDescriptionBox (‘sgpd’)中的 IsEncypted 的设置增加对加密算法 SM4-CBC 的支持，具体定义如下；

——0x0:没有加密。

——0x1:加密。

- 采用 AES_CTR 加密时，保护模式信息盒(‘sinf’)中的模式类型盒(‘schm’)中的模式类型 scheme_type=‘cenc’;
- 采用 AES_CBC 加密时，保护模式信息盒(‘sinf’)中的模式类型盒(‘schm’)中的模式类型 scheme_type=‘cbc1’;
- 采用 SM4_CTR 加密时，保护模式信息盒(‘sinf’)中的模式类型盒(‘schm’)中的模式类型 scheme_type=’sm41’;
- 采用 SM4_CBC 加密时，保护模式信息盒(‘sinf’)中的模式类型盒(‘schm’)中的模式类型 scheme_type=’sm42’。

——0x000002～0xFFFFFF:保留。

7 权利描述与授权

7.1 权利描述结构

数字版权管理的权利描述由内容、被授权对象、权利、密钥、密钥使用规则、计算器和数字签名等逻辑元素构成，如图3所示。

图3 权利描述逻辑结构

权利描述逻辑结构的元素说明如下：

a） 内容

内容是数字化的事物，例如图、文、音频、视频、动漫及其集合（如文件、数据库、软件、可执行代码等）等，一切可以数字化的事物都是内容。

b） 被授权对象

被授权对象是对指定内容相关权利的承载者。被授权对象通过其唯一标识描述。

c） 权利

权利是使用内容的权利，如：播放、存储等。

d） 密钥

密钥是指该许可证中所包含的密钥信息，包括密钥的类型、算法、密钥数据等。

e） 密钥使用规则

密钥使用规则包括密钥索引信息以及起始时间、截止时间、时间段、次数等相关规则信息。在使用密钥时应按照密钥使用规则中包含的规则合理使用密钥。

f） 计算器

计算器是内容许可操作的一种方式，用来描述权利描述单元的逻辑运算方法。支持与、或、非、异或等逻辑运算。许可证中包括计算器单元的情况下，通过内容、被授权对象、密钥、密钥使用规则、权利之间的逻辑运算实现对内容的许可。许可证中不包括计算器单元的情况下，内容、被授权方、密钥、密钥使用规则、权利之间是逻辑“与”的关系。

7.2 权利描述编码

7.2.1 编码方法

权利描述编码由许可证索引单元和一系列的基本单元组成。许可证索引单元描述许可证的版本、许可证ID和基本单元的数量。基本单元包括：内容单元、被授权对象单元、密钥单元、权利单元、密钥使用规则单元、计算器单元以及数字签名单元等。许可证发布时由许可证索引单元和后续的一个或多个基本单元组成，许可证索引单元应为许可证的第一个单元，许可证编码结构如图4所示。

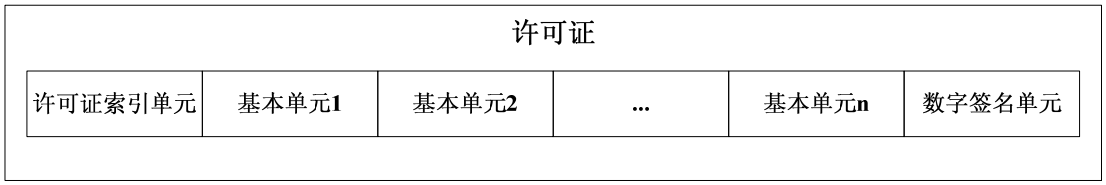

图4 许可证编码结构

许可证中的许可证索引单元和基本单元均由单元标识、长度和数据三个部分组成，单元编码如图5所示。

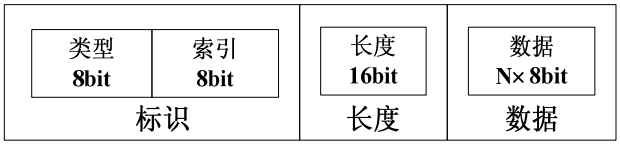

图5 单元编码方法

标识由2个字节构成，包括类型和索引，第1个字节是类型，第2个字节是该单元在许可证中的索引，用于支持许可证的分段传输。单元的索引从0开始，许可证索引单元为许可证的第1个单元，许可证索引单元编码后的索引始终为0；许可证索引单元后的基本单元的索引依次为1、2、3、……，依此类推。

长度是该单元实际数据信息的长度，由两个字节表示。

数据是单元的实际数据。

单元类型规定见表5。

表5　单元类型

类型	编码
许可证索引	0x00
内容	0x01
被授权对象	0x02
密钥	0x03
密钥使用规则	0x04
权利	0x10～0x9F
计算器	0xA0～0xAF
数字签名	0xFF
保留	0x05～0x0F,0xD0～0xDF,0xE0～0xEF

一个许可证由多个独立的不同单元构成，单元的数量和种类由许可证索引单元标识。一个许可证可用一个签名完整传输，同时也可以按单元分别传输。

7.2.2　许可证索引单元

许可证索引单元是许可证的第一个单元，许可证索引单元中包括：版本、许可证编号、基本单元数量。许可证单元数据结构见表6。

表6　许可证索引单元数据结构

字段	比特数	类型	描述
Type	8	uimsbf	0x00
Index	8	uimsbf	0x00
Length	16	uimsbf	数据长度
Version	8	uimsbf	许可证版本
LicenseID	64	uimsbf	许可证 ID
UnitsNumber	8	uimsbf	许可证中基本单元的数量

Version：许可证版本号，当前为1。

LicenseID：许可证的唯一编号。

UnitsNumber：许可索引单元后面包含的基本单元的数量。

7.2.3　内容单元

内容单元的数据结构见表7。

表7　内容单元数据结构

字段	比特数	类型	描述
Type	8	uimsbf	0x01
Index	8	uimsbf	0x01～0xFF
Length	16	uimsbf	数据长度
ContentID	64	uimsbf	内容唯一标识
KeyIdentifierLen	8	uimsbf	密钥标识长度
KeyIdentifier[]	N	uimsbf	密钥标识

ContentID：内容唯一标识，符合GY/T 260-2012的规定。

KeyIdentifierLen、KeyIdentifier：密钥标识用来唯一标识密钥，密钥唯一标识由厂商自定义；本标准只定义密钥标识的长度和密钥标识数据，密钥标识长度用1个字节表示。KeyIdentifierLen、KeyIdentifier可以不限于一个，以支持一个内容对应多个密钥的场景。

7.2.4　被授权对象单元

被授权对象是对指定内容相关权利的承载者。被授权对象单元中包括被授权对象类型和被授权对象ID，被授权对象单元数据结构见表8。

表8　被授权对象单元数据结构

字段	比特数	类型	描述
Type	8	uimsbf	0x02
Index	8	uimsbf	0x01～0xFF
Length	16	uimsbf	数据长度
ObjectType	8	uimsbf	被授权对象类型
ObjectID	8×N	uimsbf	被授权对象唯一标识

ObjectType：被授权对象的类型。

ObjectID：被授权对象唯一标识。

在具体实施中，ObjectType和ObjectID由实施者规定。

7.2.5　密钥单元

密钥单元中包括密钥算法、密钥数据、和密钥附加信息，密钥数据中是否包含密钥附加信息可根据密钥单元的长度判断。密钥附加信息包括：密钥类型、密钥标识、加密该密钥的密钥类型和密钥标识信息。密钥单元数据结构见表9。

表9　密钥单元数据结构

字段	比特数	类型	描述
Type	8	uimsbf	0x03
Index	8	uimsbf	0x01～0xFF
Length	16	uimsbf	数据长度
KeyAlgorithm	8	uimsbf	密钥算法
KeyDataLen	16	uimsbf	密钥数据的长度
KeyData[]	M	uimsbf	密钥数据

表 9（续）

字段	比特数	类型	描述
KeyType	8	uimsbf	密钥类型
KeyIdentifierLen	8	uimsbf	密钥标识长度
KeyIdentifier[]	N	uimsbf	密钥标识
UpperKeyType	8	uimsbf	上层密钥类型
UpperKeyIdentifierLen	8	uimsbf	上层密钥标识长度
UpperKeyIdentifier[]	N	uimsbf	上层密钥标识

KeyAlgorithm：密钥算法，用1个字节标识该密钥所对应的加密算法，密钥算法规定见附录B。

KeyDataLen、KeyData[]：密钥数据是由加密该密钥的密钥加密后的密钥数据。

KeyType：密钥类型，用1个字节标识该密钥的功能，密钥类型规定见表10。

KeyIdentifierLen、KeyIdentifier：密钥标识用来唯一标识该密钥；本标准只定义密钥标识的长度和密钥标识数据；KeyIdentifierLen表示密钥标识的长度，用1个字节表示；KeyIdentifier表示密钥标识。

UpperKeyType：加密该密钥的密钥类型，密钥类型规定见表10。

UpperKeyIdentifierLen、UpperKeyIdentifier：加密该密钥的密钥的唯一标识。

本标准支持对内容不同部分用不同的密钥加密，用不同的密钥ID指向不同的内容类型或部分进行加密。

表10　密钥类型

类型	编码
内容密钥	0x01
业务密钥	0x02
设备密钥	0x03
保留	0x04～0xFF

7.2.6　密钥使用规则单元

密钥使用规则规定被授权对象要按照密钥使用规则合理的使用密钥。密钥使用规则单元中包括：密钥标识、使用规则；密钥使用规则单元数据结构见表11。

表11　密钥使用规则单元数据结构

字段	比特数	类型	描述
Type	8	uimsbf	0x04
Index	8	uimsbf	0x01～0xFF
Length	16	uimsbf	数据长度
KeyType	8	uimsbf	密钥类型
KeyIdentifierLen	8	uimsbf	密钥标识长度
KeyIdentifier[]	N	uimsbf	密钥标识
KeyRulesNum	8		密钥使用规则数量
for(i=0;i<KeyRulesNum;i++)			
{			

表 11（续）

字段	比特数	类型	描述
KeyRuleType	8	uimsbf	密钥使用规则类型
KeyRuleLen	8	uimsbf	密钥使用规则长度
KeyRuleData[]	L	uimsbf	密钥使用规则数据
}			

KeyType：密钥类型，用1个字节标识该密钥的功能，密钥类型规定见表10。

KeyIdentifierLen、KeyIdentifier：密钥标识用来唯一标识该密钥，密钥唯一标识由厂商自定义；本标准只定义密钥标识的长度和密钥标识数据，KeyIdentifierLen表示密钥标识的长度，用1个字节表示；KeyIdentifier表示密钥标识。

KeyRulesNum：密钥使用规则数量，用1个字节表示。

KeyRuleType：密钥使用规则类型，用1个字节表示；本标准中定义的常用密钥使用规则的类型，密钥使用规则类型规定见表12。

KeyRuleLen、KeyRuleData：密钥使用规则数据；KeyRuleLen表示密钥使用规则数据的长度，用1个字节表示，KeyRuleData表示密钥使用规则数据。

表12　密钥使用规则类型

名称	类型
起始时间	0x01
截止时间	0x02
次数	0x03
时间段	0x04
累计时间段	0x05
保留	0x06～0xFF

密钥使用规则规定了密钥在使用时需要遵循的约束规则，密钥使用规则包括：起始时间、截止时间、次数、时间段、累计时间段等。

起始时间：规定在该时间之后允许使用密钥，在该时间之前不允许使用密钥，该时间为UTC时间，用32位长度uimsbf类型数据表示，表示自1970年1月1日0:00:00起到该时间点的秒数；

截止时间：规定在该时间之前允许使用密钥，在该时间之后不允许使用密钥，该时间为UTC时间，用32位长度uimsbf类型数据表示，表示自1970年1月1日0:00:00起到该时间点的秒数；

次数：规定允许使用密钥的次数，用32位长度uimsbf类型数据表示；

时间段：规定从第一次使用密钥之后允许使用密钥的时间范围，用32位长度uimsbf类型数据表示，单位为秒；

累计时间段：规定从第一次使用密钥开始，累计使用密钥的时间段，每次停止使用密钥即停止计时，用32位长度uimsbf类型数据表示，单位为秒；

密钥使用规则不分优先级；密钥使用规则均为可选规则，如果某个密钥没有定义任何使用规则，则对于该密钥的使用无任何限制；密钥使用规则可单独使用也可联合使用；在多规则联合使用时按照“逻辑与”的方式联合各规则，即多规则情况下只要有一项规则不满足就不允许使用密钥。

7.2.7　权利单元

7.2.7.1　权利单元类型

权利单元用来描述授予被授权对象使用内容的权利。权利单元的数据结构见表13。

表13　权利单元数据结构

字段	比特数	类型	描述
Type	8	uimsbf	权利单元类型
Index	8	uimsbf	0x01～0xFF
Length	16	uimsbf	数据长度
RightsData[]	8×N	uimsbf	权利单元数据

Type：权利单元类型，权利单元类型规定见表14。

RightsData[]：权利单元数据，规定权利的具体信息；某些情况下，权利单元可以不包含数据。具体权利单元的权利数据描述见各权利单元的描述。

表14　权利单元类型

类型		规定	描述
播放	播放	0x10	强制
	按次播放	0x11	强制
	按时长播放	0x12	可选
	按时间段播放	0x13	强制
	连接保护播放	0x14	强制
	按质量播放	0x15	可选
	保留	0x16～0x1F	可选
录制	录制	0x20	强制
	按时间段录制	0x21	可选
	按时长录制	0x22	可选
	保留	0x23～0x2F	可选
复制	复制	0x30	可选
	保留	0x31～0x3F	可选
存储	存储	0x40	强制
	保留	0x41～0x4F	可选
转发	转发	0x50	可选
	保留	0x51～0x5F	可选
执行	执行	0x60	可选
	保留	0x61～0x6F	可选
超级权利	超级权利	0x80	可选
其他	按次权利	0x91	可选
	按时长权利	0x92	可选
	按时间段权利	0x93	可选
	连接保护权利	0x94	强制
	保留	0x70～0x7F、0x81～0x8F、0x95～0x9F	可选

7.2.7.2　播放权利

7.2.7.2.1　播放

播放权利单元中的RightsData[]为空。

7.2.7.2.2 按次播放

按次播放权利单元中的RightsData[]为32位长度uimsbf类型的播放次数。

7.2.7.2.3 按时长播放

按时长播放权利单元中的RightsData[]是32位长度的uimsbf类型的播放时长，单位：秒。

7.2.7.2.4 按时间段播放

按时间段播放权利单元中的RightsData[]是32位长度uimsbf类型的播放起始时间和32位长度uimsbf类型的播放截止时间，时间数据为UTC时间，单位：秒。

7.2.7.2.5 按质量播放

按质量播放权利单元中的RightsData[]是8位的播放质量，0：所有质量；1：标清；2：高清；3：超高清。

7.2.7.2.6 连接保护播放

连接保护播放权利单元中的RightsData[]用1个字节表示是否需要连接保护。0：不允许HDMI；1：允许HDMI；2：允许HDMI但必须强制连接保护。

7.2.7.3 录制权利

7.2.7.3.1 录制

录制权利单元中的RightsData[]为空。

7.2.7.3.2 按时长录制

按时长录制权利单元中的RightsData[]为32位uimsbf类型数据，表示允许录制的时长，单位：秒。

7.2.7.3.3 按时间段录制

按时间段录制权利单元中的RightsData[]是32位长度的录制起始时间和32位长度的录制截止时间，时间数据为UTC时间。

7.2.7.4 复制权利

复制权利单元中的RightsData[]为空。

7.2.7.5 存储权利

存储权利单元中的RightsData[]为空。

7.2.7.6 转发权利

转发权利单元中的RightsData[]为空。

7.2.7.7 执行权利

执行权利单元中的RightsData[]为空。

7.2.7.8 超级权利

超级权利，拥有本版本下的所有权利。超级权利类型权利单元中的RightsData[]为空。

7.2.7.9 其他运算权利

7.2.7.9.1 按次权利

按次权利单元的RightsData[]为32位uimsbf类型数据，代表次数。按次权利单元单独使用时，对对象已有的权利按次数做进一步的限制。表示允许执行的次数。

7.2.7.9.2 按时长权利

按时长权利单元的RightsData[]为32位uimsbf类型数据，表示时长，单位：秒。

按时长权利单独使用时，对对象已有的权利按时长做进一步限制。

7.2.7.9.3 按时间段权利

按时间段权利单元的RightsData[]是32位长度uimsbf类型的起始时间和32位长度uimsbf类型的截止时间，时间数据为UTC时间，单位：秒。

按时间段权利单独使用时，对对象已有的权利按时间段做进一步限制。

7.2.7.9.4 连接保护权利

连接保护权利单元的RightsData[]用1个字节表示是否需要连接保护：0：不允许HDMI；1：允许HDMI；2：允许HDMI但必须强制连接保护。

7.2.8 计算器单元

计算器单元描述参与运算的各个基本单元之间的逻辑关系，各基本单元在计算器单元中按照其索引被引用，计算器单元也可以被其他计算器单元引用，从而实现迭代运算。计算器单元数据结构如表15所示。

表15 计算器单元数据结构

字段	比特数	类型	描述
Type	8	uimsbf	0xA0～0xAF
Index	8	uimsbf	0x01～0xFF
Length	16	uimsbf	数据长度
RightsIndexNumber	16	uimsbf	描述符索引号总数量
for(i=0;i< m,i++){			描述符索引号
RightsIndex	8	uimsbf	
}			

RightsIndexNumber：参与运算的各描述符索引的总数量。

RightsIndex：参与运算的各描述符的在许可证中的位置序号，该序号按照各描述符在许可证中的位置顺序依次自动编号，从0开始编号，计算器也参与编号，计算器单元也可以被后续的计算器单元引用，从而实现迭代运算。

计算器单元类型规定见表16。

表16　计算器类型规定

类型	编码
与	0xA0
或	0xA1
非	0xA2
异或	0xA3

7.2.9　数字签名单元

许可证的最后一个单元是数字签名单元，数字签名单元是对其前面所有单元数据进行签名，保证数据的完整性，数字签名单元数据结构见表17。

表17　数字签名单元数据结构

字段	比特数	类型	描述
Type	8	uimsbf	0xFF
Index	8	uimsbf	0x01～0xFF
Length	16	uimsbf	数据长度
Algorithm	8	uimsbf	数字签名算法
CertificateIDLength	8	uimsbf	证书序列号长度
CertificationID	N×8	uimsbf	证书序列号
SignatureLength	16	uimsbf	签名数据的长度
Signature[]	M×8	uimsbf	签名数据

Algorithm：采用的数字签名算法，算法规定见附录B。

CertificateIDLength：证书序列号长度。

CertificationID：签名使用的证书的序列号、唯一编号。

SignatureLength：签名数据的长度。

Signature：签名数据。

8　权利获取协议

8.1　权利获取协议框架

权利获取协议是DRM服务端与DRM代理之间用于交换安全上下文、请求和获取许可证的安全协议。权利获取协议包括：DRM服务端与DRM代理之间交换安全上下文的4-pass安全交互协议和DRM代理请求许可证的2-pass许可证获取协议。

a）　4-pass安全交互协议

安全交互协议用于DRM服务端与DRM代理之间进行完整的安全信息交互。一般情况下，安全交互协议只在第一次交互时执行；但如果需要更新安全信息如DRM代理时间不准确时，需要重新执行安全交互协议。

b）　2-pass许可证获取协议

DRM代理与DRM服务端通过4-pass安全交互协议建立安全上下文后，可向DRM服务端发起许可证获取协议获取相应的许可证。

DRM代理可以根据用户的交互启动权利获取协议，也可以根据其接收到的权利获取协议触发器触发权利获取协议。权利获取协议触发器由DRM服务端或运营系统生成，DRM代理接收到该触发器后应根据触发器的内容启动安全交互协议或许可证获取协议。

8.2 权利获取协议消息

8.2.1 权利获取协议触发器

触发器包括两类：安全交互触发器和许可证获取触发器。

a） 安全交互触发器

安全交互触发器消息包括：触发器ID、触发器版本、DRM服务端ID、DRM服务端URL；安全交互触发器消息描述见表18。

表18 安全交互触发器消息

参数	描述
Version	必选
TriggerID	必选
DRMServerID	必选
DRMServerURL	必选

Version：触发器版本号，当前版本Version默认为“1.0”。

TriggerID：当前触发器唯一标识符。

DRMServerID：当前触发器DRM服务端唯一标识。

DRMServerURL：DRM服务端地址。

b） 许可证获取触发器

许可证获取触发器消息包括：触发器ID、触发器版本、DRM服务端ID、DRM服务端URL、内容标识列表；内容标识列表中包含一个或多个内容标识，内容标识应与“内容封装格式”中规定的内容标识一致。许可证获取触发器消息描述见表19。

表19 许可证获取触发器消息

参数	描述
Version	必选
TriggerID	必选
DRMServerID	必选
DRMServerURL	必选
ContentIDs	必选

Version：触发器版本号，当前版本Version默认为“1.0”。

TriggerID：当前触发器唯一标识符。

DRMServerID：当前触发器DRM服务端唯一标识。

DRMServerURL：DRM服务端地址。

ContentIDs：内容标识列表，其中包含一个或多个内容标识，指示DRM代理可获取许可证的内容。

8.2.2 安全交互协议

8.2.2.1 DRM 代理 Hello

DRM代理向DRM服务端发送DRM代理Hello消息，发起安全上下文交换。DRM代理Hello消息包括消息版本号、设备ID、设备支持的算法和可选的扩展信息，描述见表20。

表20 DeviceHello消息描述

参数	描述
Version	必选
DeviceID	必选
SupportedAlgorithms	可选
Extensions	可选

Version：DRM代理支持的最高协议版本号。当前版本Version默认为“1.0”，升级到高版本后应向下兼容。

DeviceID：DRM代理唯一标识，DRM代理唯一标识是6.2.4中定义的被授权对象唯一标识的一种。

SupportedAlgorithms：DRM代理支持的加密算法，算法规定见附录B。

Extensions：DRM代理Hello消息的扩展信息，扩展信息说明如下：

——CertificateCaching：该扩展指示DRM代理具有保存DRM服务端证书的能力。

8.2.2.2 DRM服务端Hello

DRM服务端Hello消息是安全交互协议的第二条消息，由DRM服务端发送到DRM代理作为DRM代理Hello消息的响应。DRM服务端Hello消息包括：响应状态、会话ID、选择的版本、DRM服务端ID、选择的算法、随机数、可信认证机构、服务端信息、以及可选的扩展等信息。DRM服务端Hello消息见表21。

表21 DRM服务端Hello消息描述

参数	规定	
	Status=”Success”	Status≠”Success”
Status	必选	必选
SessionID	必选	-
SelectedVersion	必选	-
DRMServerID	必选	-
SelectedAlgorithms	可选	-
Nonce	必选	-
TrustedAuthorities	可选	-
ServerInfo	可选	-
Extensions	可选	-

Status：指示DRM代理Hello请求是否被成功处理；如果没有成功，返回相应的错误码。Status的定义见表22。

表22 Status定义

编码	描述
success	成功
abort	DRM服务端拒绝DRM代理的请求
notSupported	DRM服务端不支持当前类型的请求
accessDenied	DRM代理没有被授权访问DRM服务端

表 22（续）

编码	描述
notFound	该状态码仅会出现在许可证响应消息中，表示找不到 DRM 代理请求的许可证对象
malformedRequest	DRM 服务端无法解析 DRM 代理的请求
unsupportedVersion	DRM 服务端不支持当前 DRM 代理所使用的协议版本
invalidCertificateChain	DRM 服务端无法验证 DRM 代理的证书链
signatureError	DRM 服务端无法验证 DRM 代理的签名
deviceTimeError	DRM 代理的时间和 DRM 服务端不一致
notRegistered	DRM 代理没有在 DRM 服务端注册

SessionID：DRM服务端设置的协议会话标识符。

SelectedVersion：DRM服务端选择的协议版本；DRM服务端选择的版本应是DRM代理所支持的版本。

DRMServerID：当前DRM服务端的唯一标识符。

SelectedAlgorithms：定义在权利获取协议交互中的加密算法。

Nonce：DRM服务端发送的随机数。

TrustedAuthorities：DRM服务端可识别的DRM代理的信任锚点列表。该参数是可选的。如果DRM服务端已有DRM代理的证书或者能够校验DRM代理生成的签名，则不发送该参数。如果该参数存在，但是空的，它表示DRM代理可以自由选择任意的DRM代理证书以鉴别自己，否则DRM代理必须选择一个与该信任锚点列表中信任锚点相关的证书链。

ServerInfo：DRM服务端返回的与DRM服务端相关的信息。

Extensions：DRM服务端Hello消息可选扩展信息，具体包含以下可选扩展：

——PeerKeyIdentifier：DRM 服务端保存的 DRM 代理的公钥标识。如果该标识与 DRM 代理 Hello 消息中的 DeviceID 匹配，则表示 DRM 服务端已经保存有 DRM 代理证书链，DRM 代理在后续消息中不需要发送其证书链。如果该标识为空，表示 DRM 服务端已经保存了所有与 DRM 代理 Hello 消息中 DeviceID 列表中匹配的证书信息，DRM 代理不需要再发送其证书链。

——CertificateCaching：该扩展指示 DRM 服务端具有保存 DRM 代理证书的能力。

——DeviceDetails：该扩展指示 DRM 代理在后续消息中应返回 DRM 代理相关信息，如制造商信息等。

8.2.2.3 安全交互请求

安全交互请求消息是DRM代理向DRM服务端发起的安全信息交互请求，该消息是4-pass安全交互协议的第三条消息。安全交互请求消息描述见表23。

表23 安全交互请求消息描述

参数	描述
SessionID	必选
DeviceNonce	必选
RequestTime	必选
CertificateChain	可选
TrustedDRMServerAuthorities	可选
ServerInfo	可选
Extensions	可选
Signature	必选

SessionID：会话标识，该会话标识应是DRM服务端Hello消息中的会话标识，如果该标识与DRM服务端Hello消息中的会话标识不一致，DRM服务端将终止协议执行。

DeviceNonce：DRM代理生成的随机数。

RequestTime：DRM代理当前DRM时间。DRM时间是指一种安全的、不可被用户修改的时间来源，通常是在DRM客户端向DRM服务端进行安全交互时，从DRM服务端获取的时间，将该信息以某种方式保存在本地，并在以后的通信中使用。如果某设备不支持DRM时间，那么RequestTime的值应置为“Undefined”。不支持DRM时间的DRM代理应在该参数中指明。

CertificateChain：DRM代理证书链，该证书链不包括根证书，且DRM代理证书应在证书链的第一个，每个后续证书必须直接证明前一个证书。如果DRM服务端Hello消息中包含了DRM服务端可选信任锚点，则DRM代理的证书链必须是与可选信任锚点相关的证书链。

TrustedDRMSeverAuthorities：DRM代理可识别的信任锚点列表。如果该参数为空，则表明DRM服务端可自由选择证书。

ServerInfo：DRM服务端信息。如果DRM服务端Hello消息中存在该参数，则安全交互请求消息中必须包含该参数，且必须是DRM服务端Hello消息中发送到DRM代理的ServerInfo信息。

Extensions：安全交互请求消息的可选扩展信息。可选扩展信息说明如下：

——PeerKeyIdentifier：DRM 服务端公钥标识。如果该标识与 DRM 服务端 Hello 消息中的 DRMServerID 匹配，或者为空，则 DRM 服务端在安全交互响应消息中不需要发送其证书链。

——NoOCSPResponse：不需要 OCSP 响应。存在该参数，表示 DRM 服务端在安全交互响应消息中不需要包含 OCSP 响应信息。

——OCSPResponderKeyIdentifier：该扩展标识 DRM 代理保存的 OCSP 响应公钥，如果该公钥与 DRM 服务端的 OCSP 响应公钥相同，则 DRM 服务端不需要发送 OCSP 响应器的证书链。

——DeviceDetails：该扩展指示 DRM 代理相关信息，如支持的许可证格式、制造商信息等。

Signature：对消息中除了Signature之外的所有数据进行签名。

8.2.2.4 安全交互响应

DRM服务端发送安全交互响应消息到DRM代理，响应DRM代理的安全交互请求。该消息执行成功，DRM代理应建立该DRM服务端的安全上下文。安全交互响应消息描述见表24。

表24 安全交互响应消息描述

参数	规定	
	Status=”Success”	Status≠”Success”
Status	必选	必选
SessionID	必选	可选
DRMServerURL	必选	-
CertificateChain	可选	-
OCSPResponse	可选	-
Extensions	可选	-
Signature	必选	-

Status：指示消息是否被成功处理；如果失败，则发送一个错误码，Status规定见表22。

SessionID：会话唯一标识。该会话标识与DRM服务端Hello和安全交互请求消息中的会话标识一致。如果不一致，则DRM代理应终止协议的执行。

DRMServerURL：DRM代理获取许可证的URL，该URL符合RFC2396规范。

CertificateChain：DRM服务端证书链，该证书链不包含根证书，DRM服务端证书应排在证书链的第一个，每个后续证书应直接证明前一个证书。如果安全交互请求消息中包含DRM代理可选的信任锚点，则DRM服务端应从其中选择一个作为其证书链的信任锚点。如果安全交互请求消息中包含PeerKeyIdentifier扩展，则DRM服务端可不发送其证书链。

OCSPResponse：DRM服务端证书链的有效OCSP响应集合。如果DRM代理在安全交互请求中发送了NoOCSPResponse扩展，则不需要发送该参数。

Extensions：安全交互响应消息的扩展信息。

Signature：对消息中除了Signature之外的所有数据进行签名。

8.2.3 许可证获取协议

8.2.3.1 许可证请求

DRM代理发送许可证请求消息到DRM服务端请求许可证。许可证请求消息描述见表25。

表25 许可证请求消息描述

参数	规定
DeviceID	必选
DRMServerID	必选
DeviceNonce	必选
RequestTime	必选
ContentIDs	必选
CertificateChain	可选
Extensions	可选
Signature	必选

DeviceID：DRM代理标识。

DRMServerID：DRM服务端标识。

DeviceNonce：DRM代理产生的nonce。

RequestTime：DRM代理的当前DRM时间。

ContentIDs：描述请求的许可证，如果该许可证获取协议是通过许可证获取触发器触发的，则ContentIDs中包含的是触发器中的内容标识信息。

CertificateChain：如果DRM服务端未保存DRM代理的证书信息，则必须发送该信息。

Extensions：许可证请求消息的扩展信息。下列是为许可证请求消息而定义的扩展：

——PeerKeyIdentifier：存储在 DRM 代理的 DRM 服务端公钥标识符。如果该标识符匹配规定的 DRMServerID，或者为空，则表示 DRM 代理已经保存了 DRMServerID 以及对应的 DRM 服务端的证书链，DRM 服务端不需要在它的响应消息中下发其证书链。

——NoOCSPResponse：该扩展表示 DRM 代理已缓存 DRM 服务端的有效 OCSP 响应，DRM 服务端不需要在响应消息中发送 OCSP 响应。

——OCSPResponderKeyIdentifier：该扩展标识 DRM 代理保存的 OCSP 响应公钥，如果该公钥与 DRM 服务端的 OCSP 响应公钥相同，则 DRM 服务端不需要发送 OCSP 响应器的证书链。

Signature：对消息中除了Signature之外的所有数据进行签名。

8.2.3.2 许可证响应

DRM服务端发送许可证响应消息到DRM代理，该消息中包括DRM代理请求的许可证信息；许可证响应消息描述见表26。

表26　许可证响应消息描述

参数	规定	
	2-pass Status=”Success”	2-pass Status≠”Success”
Status	必选	必选
DeviceID	必选	-
DRMServerID	必选	-
DeviceNonce	必选	-
ProtectedLicenses	必选	-
CertificateChain	可选	-
OCSPResponse	可选	-
Extensions	可选	-
Signature	必选	-

Status：指示许可证请求是否成功完成；如果失败，需要发送一个错误码，Status规定见表22。

DeviceID：DRM代理唯一标识，该唯一标识必须与许可证请求消息中的DRM代理唯一标识一致。如果DRM代理接收到的许可证响应消息中的DRM代理标识与当前DRM代理不一致，DRM代理应忽略该消息。

DRMServerID：DRM服务端唯一标识，该唯一标识必须与许可证请求消息中的DRM服务端唯一标识一致。

DeviceNonce：DRM代理发送的随机数，该随机数必须是DRM代理发送许可证请求消息时携带的随机数。

ProtectedLicenses：DRM服务端返回的DRM代理请求的一个或多个许可证。

CertificateChain：DRM服务端证书链。如果许可证请求消息中不包括PeerKeyIdentifier扩展，则该参数必须出现。

OCSPResponse：DRM服务端OCSP响应集合；如果许可证请求消息中包含NoOCSPResponse扩展，则在许可证响应消息中可不发送该参数。

Extensions：许可证响应消息的扩展。

Signature：许可证响应消息的签名。

8.3　权利获取协议编码

8.3.1　权利获取协议编码方法

权利获取协议的消息利用URI的路径、查询信息和http报文的报文体进行传输。URI固定为“http”，表示使用http或https协议进行传输；URI中的域名为DRM服务端的地址，如“ri.example.com”。http报文体中的消息以JSON字符串进行描述，JSON语法遵循ECMA 404。http报文体中的消息样例见表27。

表27　消息样例

```
{
    "object" : {
        "key1" : "string",
        "key2" : true,
        "array1" : ["elem1","elem2" ],
        "array2" : [ {
            "key1" : "value1"
        }, {
            "key2" : "value2"
        } ]
    }
}
```

8.3.2　状态信息

权利获取协议的请求利用http协议post请求方式进行。权利获取协议的响应随着http post请求的响应返回，在http响应的消息体内，权利获取协议响应的状态信息用status字符串表示，任何响应中都应包含status字符串，标识之前的协议请求是否成功。Status字符串名为“status”，字符串中的字符串值列举了所有可能的消息请求成功或错误的信息。当传送或接收一个Status字符串值不等于“success”的消息时，DRM服务端与DRM代理之间将立即结束连接，终止协议；DRM服务端和DRM代理需要删除与本次会话相关的会话标识符、Nonce、密钥、以及其他隐私信息。状态信息的规定见表22。

8.3.3　扩展信息

DRM服务端和DRM代理之间权利获取协议消息的扩展信息用Extensions数组描述，该扩展是可选的。如果消息中存在Extensions数组，则至少应包含一个Extension对象元素，否则Extensions数组不应出现在消息报文中。若该Extension对象元素为关键性的，则需包含名称为“critical”、取值为true的属性；否则默认该对象为非关键性的。Extensions内容见表28。

表28　Extensions

```
"extensions" : [ {
    "peerKeyIdentifier" : {
        "critical" : true,
        "identifier" : ""
    }
}, {
    "deviceDetails" : {
        "critical" : true
    }
} ]
```

8.3.4　许可证数组

许可证数组“protectedLicenses”中包含的是一个或多个许可证，“protectedLicenses”应包含在LicenseResponse消息中发送。ProtectedLicenses块实例见表29。

表29 ProtectedLicenses 实例

```
"protectedLicenses" : [
    {
        " licenseID" : "base64_string",
         "license" : "base64_string"
     },
     {
        " licenseID" : "base64_string",
         "license" : "base64_string"
    },
     …
]
```

在“protectedLicenses”中应至少包含一个许可证。“licenseID”用Base64编码字符串表示许可证唯一标识，“license”用Base64编码字符串表示该内容对应的许可证。Base64编码方法遵循RFC 2045。

8.3.5 随机数字符串

随机数字符串用“nonce”表示，Nonce字符串用于表示权利获取协议消息中的随机非重复的数值。对于每个需要Nonce元素的权利获取协议消息都应随机生成一个Nonce，且该Nonce值只应被使用一次。Nonce字符串值长度不应小于14个字节。

8.3.6 消息编码规定

8.3.6.1 消息参数编码规定

权利获取协议消息及消息参数的编码规定见表30。

表30 消息编码规定

值类型	数据字段	JSON 键	说明
字符串	Type	"type"	消息类型
	Version	"version"	版本信息
	SelectedVersion	"selectedVerison"	选择的版本信息
	Status	"status"	响应消息的状态信息，成功为“Success”
	TriggerID	"triggerID"	触发器唯一标识
	DRMServerID	"drmServerID"	DRM 服务端唯一标识
	DRMServerURL	"drmServerURL"	DRM 服务端地址
	DeviceID	"deviceID"	DRM 代理唯一标识
	SessionID	"sessionID"	会话唯一标识
	Nonce	"nonce"	随机数
	DeviceNonce	"nonce"	随机数
	RequestTime	"requestTime"	以字符串形式存在的UTC时间
	ServerInfo	"serverInfo"	服务器端信息
	Signature	"signature"	消息签名的 Base64 编码字符串

表 30（续）

值类型	数据字段	JSON 键	说明
布尔值	Critical	"critical"	作为对象的属性指示该对象是否为关键性对象
字符串数组	ContentIDs	"contentIDs"	内容唯一标识数组
	Supported Algorithms	"supportedAlgorithms"	支持的算法数组
	SelectedAlgorithms	"selectedAlgorithms"	选择的算法数组
	TrustedAuthorities	"trustAuthorities"	可信证书发放第三方
	TrustedDRMServerAuthorities	"trustAuthorities"	可信证书发放第三方
	CertificateChain	"certificateChain"	证书链表，以字符串数组的形式表示，数组的每个元素都是一个 X.509 格式 DER 编码的 Base64 编码字符串
	OCSPResponse	"ocspResponse"	OCSP 响应集合，每一个数组元素都是一个 Base64 编码的 OCSP 响应数据
对象	ProtectedLicenses	"protectedLicense"	许可证
	CertificateCaching	"certificateCaching"	是否具有证书存储功能
	DeviceDetails	"deviceDetails"	设备信息
	PeerKeyIdentifier	"peerKeyIdentifier"	证书公钥标识
	NoOCSPResponse	"noOcspResponse"	不需要 OCSP 响应
	OCSPResponderKeyIdentifier	"ocspResponderKeyIdentifier"	OCSP 响应公钥标识
对象数组	ProtectedLicenses	"protectedLicenses"	每个数组元素都是一个 ProtectedLicense
	Extensions	"extensions"	消息扩展信息

8.3.6.2 消息样例

安全交互触发器消息编码格式如下：

```
{
    "type":"securityTrigger",
    "version":"1.0",
    "triggerID":"base64_string",
    "drmServerID":"base64_string",
    "drmServerURL":"string"
}
```

许可证获取触发器消息消息编码格式如下：

```
{
    "type":"licenseTrigger",
    "version":"1.0",
    "triggerID":"base64_string",
    "drmServerID":"base64_string",
```

```
 "drmServerURL":"string" ,
 "contentIDs":["base64_string","base64_string",…]
}
```

DRM 代理 Hello 消息编码格式如下：

```
{
     "type":"deviceHello",
    "version":"1.0",
    "deviceID":"base64_string",
    "supportedAlgorithms":["string","string",…],
    "extensions":["string","string",…]
}
```

DRM 服务端 Hello 消息编码格式如下：

```
{
    "type":"drmServerHello",
    "status":"string",
    "sessionID":"base64_string",
    "selectedVersion":"string",
    "drmServerID":"base64_string",
    "selectedAlgorithms":["string","string",…],
    "nonce":"base64_string",
    "trustedAuthorities":["string","string",…],
    "serverInfo":"string",
    "extensions":["string","string",…]
}
```

安全交互请求消息编码格式如下：

```
{
   "type":"securityRequest",
   "sessionID":"base64_string",
   "nonce":"base64_string",
   "requestTime":"string",
   "certificateChain":"base64_string",
   "trustedAuthorities":["string","string",…],
   "serverInfo":"string",
   "extensions":["string","string",…],
   "signature":"base64_string"
}
```

安全交互响应消息编码格式如下：

```
{
     "type":"securityResponse",
     "status":"string",
     "sessionID":"base64_string",
     "drmServerURL":"string" ,
     "certificateChain":"base64_string",
```

```
    "ocspResponse":"base64_string",
    "extensions":["string","string",…],
    "signature":"base64_string"
}
```

许可证获取请求消息编码格式如下：

```
{
    "type":"licenseRequest",
    "deviceID":"base64_string",
    "drmServerID":"base64_string",
    "nonce":"base64_string",
    "requestTime":"string",
    "contentIDs":["base64_string","base64_string",…],
    "certificateChain":" base64_string ",
    "extensions":["string","string",…],
    "signature":"base64_string"
}
```

许可证获取响应消息编码格式如下：

```
{
    "type":"licenseResponse",
    "status":"string",
    "deviceID":"base64_string",
    "drmServerID":"base64_string",
    "nonce":"base64_string",
    "protectedLicenses" : [
    {
        "licenseID" : "base64_string",
        "license" : "base64_string"
    },
    {
        "licenseID" : "base64_string",
        "license" : "base64_string"
    },
    …
    ],
    "certificateChain":"base64_string",
    "ocspResponse":"base64_string",
    "extensions":["string","string",…],
    "signature":"base64_string"
}
```

权利获取协议消息签名方法：对权利获取协议消息进行签名计算时，将“signature”的取值设置为空，即"signature":""；计算出的签名信息以 Base64 字符串形式填回到消息体中。

9 信任与安全体系

9.1 信任模型

9.1.1 概述

互联网电视数字版权管理系统应实现满足以下安全需求的信任与安全体系：

a） 受保护的内容应加密传输，许可证应被安全的分发和管理；

b） 内容只能被已鉴别和已授权的DRM代理按照许可证的要求合理的访问；

c） DRM代理应准确依从于许可证中的权利和密钥使用规则。

互联网电视数字版权管理系统的信任模型基于PKI体系。DRM系统中的各实体包括内容提供者、DRM服务器、DRM代理等都向认证中心申请获得一个数字证书，作为自己身份的凭证。各实体之间的信任关系基于数字证书的有效性。如果DRM代理的证书被DRM服务器验证有效，则DRM服务器信任该DRM代理。相似地，如果DRM服务器的证书被DRM代理验证有效，则DRM代理信任该DRM服务器。

9.1.2 数字证书

数字证书是DRM系统建立信任体系的基础。DRM代理需要与数字证书进行关联。每个DRM代理应至少携带一个数字证书。每个DRM代理必须有唯一标识符，此唯一标识符应当在数字证书的适当字段载入，并且以便于人读取的方式放在设备外部。

9.1.3 证书格式

数字证书应根据其实现的密码算法符合相关的标准。

基于RSA密码算法实现的数字证书，其格式应符合ITU-T Recommendation X.509 (1997 E)和RFC 2459的规定。

基于SM2密码算法实现的数字证书，其格式应符合GM/T 0015-2012的规定。

9.2 安全机制

9.2.1 内容保密性

在本标准中，保密性应通过加密保护敏感数据，敏感数据至少包括受保护内容和内容密钥。

9.2.2 身份鉴别

在本标准中，DRM代理和DRM服务端之间应通过验证对方的数字证书有效性来实现身份鉴别。DRM系统需依靠运营管理平台对DRM代理和终端硬件匹配的合法性进行检查。

9.2.3 数据完整性

在本标准中，数据完整性的检测应通过协议消息和许可证上的数字签名进行验证。

附 录 A
（资料性附录）
基于 HLS 协议的流媒体中增加对 ChinaDRM 支持的说明

HLS中媒体分块如果是加密的，其加密密钥通过#EXT-X-KEY来指定，格式如下：

#EXT-X-KEY:<attribute-list>。

属性包括METHOD、URI、IV、KEYFORMAT、KEYFORMATVERSIONS，属性说明见表A.1。

表A.1 属性说明

属性	说明	是否强制
METHOD	用字符串来标识媒体加密方法：NONE、AES-128、SAMPLE-AES；NONE 表示该内容未加密，这种情况下不允许出现 URI、IV、KEYFORMAT、KEYFORMATVERSIONS 等属性	如果#EXT-X-KEY 存在则该属性必须存在
URI	获取密钥的 URI 字符串	METHOD 非 NONE 情况下必须存在
IV	十六进制整数，代表加密初始向量。	可选
KEYFORMAT	标识密钥在密钥文件中的存储方式。默认是"identity"，密钥文件中的 AES-128 密钥是以二进制方式存储的 16 个字节的密钥	可选，不存在时默认为"identity"
KEYFORMATVERSIONS	由”/”分隔的字符串（如”1/3”），如果同一 KEYFORMAT 有多个版本，则该属性存在，用来区分 KEYFORMAT 的不同版本	可选，KEYFORMAT 有多个版本时存在

URI用来保存获取解密媒体分块的密钥文件。密钥文件的格式用KEYFORMAT来标识。

本附录通过下述方式描述ChinaDRM对HLS的支持：

KEYFORMAT的规定，如果KEYFORMAT=”chinadrm”，表示URI中给出的是获取ChinaDRM许可证的相关信息。ChinaDRM的不同版本在KEYFORMATVERSIONS中指出，不存在时默认为版本1。

示例：

#EXTM3U

#EXT-X-VERSION:3

#EXT-X-MEDIA-SEQUENCE:7794

#EXT-X-TARGETDURATION:15

#EXT-X-KEY:METHOD=AES-128,URI=https://priv.example.com/key.php?r=52,KEYFORMAT=”**chinadrm**”,KEYFORMATVERSIONS=”**1**”

#EXTINF:2.833,

http://media.example.com/fileSequence52-A.ts

#EXTINF:15.0,

http://media.example.com/fileSequence52-B.ts

#EXTINF:13.333,

http://media.example.com/fileSequence52-C.ts

附 录 B
（规范性附录）
密码算法

本标准规定的互联网电视数字版权管理系统应支持国际通用密码算法和国产密码算法。密码算法用1个字节标识，字节的高4位是密码算法的类别编码，低4位是密码算法的算法编号。密码算法规定见表B.1。

表B.1 密码算法规定

算法类别	类别编号	候选算法名称举例	算法编号
散列算法 （HashAlgorithm）	0000	HashAlgorithm:SHA-1	0000
		HashAlgorithm:SHA-256	0001
		HashAlgorithm:SM3-256	0010
		保留	0100～1111
公钥加密算法 （PublicKeyAlgorithm ）	0001	PublicKeyAlgorithm: RSA-1024	0000
		PublicKeyAlgorithm:RSA-2048	0001
		PublicKeyAlgorithm:SM2-256	0010
		保留	0011～1111
分组密码算法 （BlockCipherAlgorithm）	0010	BlockCipherAlgorithm:AES-128-128	0000
		BlockCipherAlgorithm:3DES-64-112	0001
		BlockCipherAlgorithm:SM4-128	0010
		保留	0101～1111
流密码算法 （StreamCipherAlgorithm ）	0011	StreamingCipherAlgorithm:RC4	0000
		保留	0001～1111
签名算法 （SignatureAlgorithm ）	0100	SignatureAlgorithm:RSA-SHA1-1024	0000
		SignatureAlgorithm:RSA-SHA1-2048	0001
		PublicKeyAlgorithm:SM2-256	0010
		保留	0011～1111
保留	0110-1111	保留	0000～1111

参考文献

[1] Internet-Draft: HTTP Live Streaming:draft-pantos-http-live-streaming

[2] Common File Format & Media Formats Specification Version 1.0.5 Digital Entertainment Content Ecosystem (DECE) LLC