GY

中华人民共和国广播电视和网络视听行业标准

GY/T 333—2020

视音频内容分发数字版权管理　有线数字电视数字版权管理系统集成

Digital rights management for video audio content distribution—
Digital rights management system integration for digital cable television

2020－11－09 发布　　2020－11－09 实施

国家广播电视总局　　发布

目　　次

前　言

本文件按照GB/T 1.1—2020《标准化工作导则　第1部分：标准化文件的结构和起草规则》的规定起草。

请注意本文件的某些内容可能涉及专利。本文件发布机构不承担识别这些专利的责任。

本文件由全国广播电影电视标准化技术委员会（SAC/TC 239）归口。

本文件起草单位：国家广播电视总局广播电视科学研究院、中央广播电视总台、华数数字电视传媒集团有限公司、上海海思技术有限公司、阿里巴巴（中国）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、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、北京安视网信息技术有限公司、中国传媒大学、英特尔（中国）有限公司。

本文件主要起草人：丁文华、郭沛宇、王磊、陈靓、邵淇锋、戴金晶、梁志坚、吴迪、赵鹏、郑黎方、赵云辉、马吉伟、刘琦、汪沛、张晶、田雪冰、刘好伟、张鹏、林卫国、隋爱娜、尚文倩、周菁、曹建香、梅雪莲、张智军、沈阳、姜涛。

视音频内容分发数字版权管理　有线数字电视数字版权管理系统集成

1　范围

本文件规定了有线数字电视DRM系统集成框架、内容加密、密钥管理、内容授权及终端集成等核心机制与接口协议。

本文件适用于有线数字电视DRM系统集成部署与实施。

2　规范性引用文件

下下列文件中的内容通过文中的规范性引用而构成本文件必不可少的条款。其中，注日期的引用文件，仅该日期对应的版本适用于本文件；不注日期的引用文件，其最新版本（包括所有的修改单）适用于本文件。

GY/T 277—2019　视音频内容分发数字版权管理技术规范

ISO/IEC 23000-19:2018　信息技术　多媒体应用格式（MPEG-A）　第19部分：分片媒体的通用媒体应用格式（CMAF）（Information technology—Multimedia application format (MPEG-A)—Part 19: Common media application format(CMAF)for segmented media）

3　缩略语

下列缩略语适用于本文件。

API　应用程序接口（Application Programming Interface）

CBC　密码分组链接（Cipher Block Chain）

CEI　内容加密信息（Content Encryption Information）

ChinaDRM　中国数字版权管理（China Digital Rights Management）

DASH　用HTTP协议传输的动态自适应流媒体协议（Dynamic Adaptive Streaming over HTTP）

DRM　数字版权管理（Digital Rights Management）

FTP　文件传输协议（File Transfer Protocol）

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）

ID　唯一标识（Identifier）

JSON　JS对象简谱（JavaScript Object Notation）

MPD　媒体展现描述（Media Presentation Description）

NFS　网络文件系统（Network File System）

OTP　一次性可编程（One Time Programmable）

PGP　优良保密协议（Pretty Good Privacy）

PMT　节目映射表（Program Mapping Table）

REE　富执行环境（Rich Execution Environment）

TA　可信应用（Trusted Application）
TEE　可信执行环境（Trusted Execution Environment）
TS　传送流（Transport Stream）
URI　通用资源标识符（Uniform Resource Identifier）
URL　统一资源定位符（Uniform Resource Locator）
uimsbf　无符号整数，高有效位优先（unsigned integer,most significant bit first）

4　集成框架

有线数字电视DRM系统用于保护IP双向有线数字电视直播和点播内容版权，确保数字电视内容通过双向IP有线网络分发到终端播放、输出全流程的安全。有线数字电视DRM系统应符合GY/T 277—2019的相关规定，包括DRM服务端系统和DRM客户端。DRM服务端系统应包括直播加密、点播加密、直播密钥管理、点播密钥管理、密钥网关、内容授权等核心子系统，通过直播加密、点播加密、内容授权与有线数字电视运营系统的协同实现有线数字电视DRM系统的服务端集成，直播加密后的直播内容通过直播内容分发发送到机顶盒等智能终端，点播加密后的内容通过点播内容分发发送到机顶盒等智能终端，通过在机顶盒等智能终端中集成DRM客户端功能，实现IP双向有线数字电视直播和点播内容版权的端到端保护。有线数字电视DRM系统集成框架如图1所示。

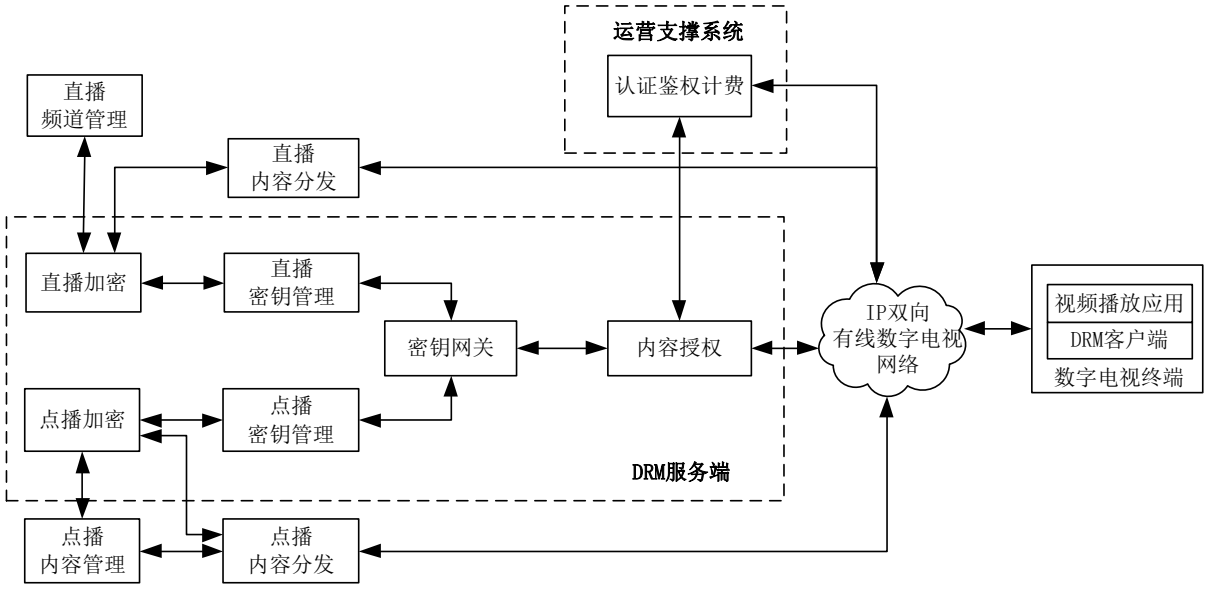

图1　有线数字电视 DRM 系统集成框架

直播加密和点播加密从对应的直播密钥管理和点播密钥管理请求内容加密密钥进行内容加密；直播密钥管理和点播密钥管理将内容加密密钥同步到密钥网关；密钥网关为内容授权提供全部直播频道和点播内容的内容加密密钥查询。

终端视频播放应用调用DRM客户端生成内容授权许可证请求，从内容授权申请内容授权许可证，根据获得的内容授权许可证进行内容的解密播放。

运营系统的频道管理实现对直播加密的配置，点播内容管理实现对点播加密的配置。内容授权接收到内容授权申请时，从运营系统查询该申请的认证鉴权计费结果，依据认证鉴权计费返回的结果判断是否生成内容授权许可证给终端。

5 授权机制

5.1 直播授权机制

直播加密按照配置的加密模式和密钥更新频率进行直播内容加密。直播内容加密密钥来自于密钥管理。

密钥管理应为直播加密生成当前内容加密密钥和下一内容加密密钥，在发送给直播加密的同时，同步到密钥网关。

直播加密按照配置的密钥更新频率进行密钥的更新，当内容加密密钥到期时，切换到下一内容加密密钥，并从密钥管理申请新的内容加密密钥；密钥管理每次都将当前加密密钥、下一加密密钥发送给直播加密，并同步直播加密密钥到密钥网关。

终端设备根据直播频道标识申请直播内容授权许可证，内容授权接收到申请后从认证鉴权计费查询设备播放权限，依据直播频道标识从密钥网关查询直播加密密钥，如果终端设备具备播放权限，则将该频道的当前内容加密密钥和下一内容加密密钥封装到直播内容授权许可证中发送给终端。

终端设备根据内容授权许可证进行直播内容解密播放，在直播播放过程中发现CEI信息中发生密钥更新时，应检查本地是否存储有相应的内容加密密钥，如未发现本地有相应密钥，则启动新的直播内容授权许可申请。

5.2 点播授权机制

点播内容管理维护点播内容加密模式、加密内容唯一标识、加密内容URL等信息，通过向内容加密系统下达加密任务实现内容的加密。

内容加密完成后，通过点播内容分发系统进行分发。

内容加密密钥都同步到密钥网关，内容授权在接收到终端点播内容授权许可证请求后，通过认证鉴权计费系统查询终端的播放许可，从密钥网关查询内容加密密钥，封装成内容授权许可证发给终端。

终端设备的DRM客户端通过内容授权许可证按照密钥使用规则进行内容的解密播放。

6 内容加密

6.1 内容加密方法

6.1.1 直播加密方法

直播加密包括本地加密和第三方加密两种。第三方加密的情况下，第三方机构的密钥管理将直播频道内容加密密钥同步到运营机构的密钥网关。

直播加密从密钥管理申请直播内容加密密钥进行直播频道内容加密。在实际部署时，直播加密可作为独立的设备部署，也可做为直播编转码设备中的一个功能模块进行部署。

直播内容加密应按照GY/T 277—2019中6.2的方法对直播内容基本码流进行加密，在基本码流的扩展数据中增加内容加密信息CEI，在数字电视传输流的PMT表中增加ChinaDRM描述子。CEI语法见GY/T 277—2019中表1，ChinaDRM描述子见GY/T 277—2019中表3，ChinaDRM描述子中的DRM_data_bytes应包含直播频道标识。

直播加密应配置直播频道标识、直播加密模式、直播密钥更新频率、密钥管理URL等配置信息。直播加密应按8.1.2规定的直播加密密钥请求接口从密钥管理请求直播内容加密密钥。

直播加密应具备加密控制接口，通过加密控制接口控制节目是否加密。当接收到不加密的控制消息时，应将ChinaDRM描述子中的视频加密模式变更为NONE，将CEI中的encryption_flag设置为0，即后续内容不加密。终端设备在解密过程中，发现ChinaDRM描述子中的视频加密模式为NONE将不检查更新内容授权许可证，发现CEI中的encryption_flag设置为0，将后续数据不解密直接送到解码模块。

6.1.2 点播加密方法

点播采用TS分发内容时，内容加密应按照GY/T 277—2019中6.2的方法对基本码流加密。

点播采用HLS分发内容时，内容加密封装采用TS文件格式，支持H.264、H.265、AVS+、AVS2等视频编码。内容加密可采用全加密模式或部分加密模式，内容加密算法应采用SM4算法，加密模式应采用CBC模式。加密内容的基本码流中包含CEI数据，CEI数据中包含内容加密密钥唯一标识和初始向量。M3U8文件中包含#EXT-X-KEY，其METHOD属性应为SM4-CBC或SAMPLE-SM4，VIDEOFORMAT应为实际的编码内容格式，URI中包含该内容授权许可证获取URL。

点播采用DASH分发内容时，MPD文件的规定见GY/T 277—2019中6.3.2，内容加密应遵循GY/T 277—2019中6.3.3的规定。

点播采用ISO/IEC 23000-19:2018分发内容时，内容加密应遵循GY/T 277—2019中6.3.3的规定。

6.2 内容使用规则

内容使用规则设置在频道管理或点播内容管理中，频道管理通过设置直播加密将内容使用规则设置到直播加密中，由直播加密在请求直播加密密钥时发送给直播密钥管理，点播内容管理在向点播内容加密发送加密任务时需要携带内容使用规则，由点播加密在请求点播内容加密密钥时发送给点播密钥管理。

密钥管理通过密钥同步消息将内容使用规则同步到密钥网关；客户端申请内容授权许可证时，密钥网关将内容使用规则封装在内容授权的密钥查询响应消息中发送给内容授权，由内容授权将其转换为密钥使用规则封装到内容授权许可证中发送给DRM客户端。

内容使用规则包括输出规则和客户端安全等级规则等，编码规定见GY/T 277—2019中表15、表16的密钥使用规则，语法格式见表1。

表1 内容使用规则语法格式

字段	比特数	类型	描述
KeyRulesNum	8		密钥使用规则数量
for(i=0;i<KeyRulesNum;i++)			
{			
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
}			

6.3 点播加密接口

6.3.1 概述

点播内容管理通过点播加密接口与点播内容加密通信，实现对点播内容的加密。点播加密接口包括：添加加密任务、查询任务状态、删除加密任务等，通信协议采用HTTP/HTTPS协议，POST（JSON）接口。

6.3.2 添加加密任务

添加加密任务由内容管理系统发起，URI示例为：https://<IP>:<Port>/add_vod_enc_task。添加加密任务的JSON消息数据见表2。

表2 添加加密任务的 JSON 消息数据

JSON 键	值类型	必选/可选
contentID	string	必选
cotentEncMode	string	必选
contentRules	base64_string	必选
fileFormat	string	必选
videoFormat	string	必选
audioFormat	string	必选
sourceFilePath	string	必选
desFilePath	string	必选
responseURL	string	可选

contentID：内容唯一标识。

contentEncMode：视频加密模式，“SM4-CBC”为全加密模式，“SAMPLE-SM4-CBC”为部分加密模式。

contentRules：内容使用规则，按照表1编码，采用BASE64编码格式编码传输，见表1。

fileFormat：文件封装格式，包括：“TS”、“MP4”、“HLS”、“DASH”等。

videoFormat：视频编码格式，包括：“H264”、“H265”、“AVS+”、“AVS2”等。

audioFormat：音频编码格式。

sourceFilePath、desFilePath：待加密内容路径和加密后内容路径，应支持HTTP、FTP、NFS等路径格式。

responseURL：内容加密任务反馈加密结果消息的URI，示例为：https://<IP>:<Port>/response_addtask。

添加加密任务的JSON消息示例如下：

```
{
      "contentID":"string",
      "contentEncMode":"SM4-CBC",
      "contentRules":"base64_string",
      "fileFormat":"string",
      "videoFormat":"H264",
      "audioFormat":"AC3",
      "sourceFilePath":"string",
      "desFilePath":"string",
      "responseURL":"string"
}
```

添加加密任务的内容加密系统响应消息见表3。

表3　添加加密任务的内容加密系统响应消息

消息内容项	消息内容项描述
响应状态码	200

响应数据	{
	"code":"000",//返回代码,000 为成功,其他待定义
	"details":"error message"
	}

code 字段定义	000　正常情况 101　参数错误 301　系统异常

6.3.3　查询任务状态

查询加密任务由内容管理系统发起，URI示例为：https://<IP>:<Port>/req_vod_enc_task。查询加密任务的JSON消息数据见表4。

表4　查询加密任务的 JSON 消息数据

JSON 键	值类型	必选/可选
contentID	string	可选
reqMode	string	必选
page	number	必选
pageSize	number	必选

contentID：内容唯一标识，可选，如无该字段则默认为获取所有的加密任务状态信息。

reqMode：指定要获取哪些任务的状态，包括：加密中“encrpting”、排队中“waiting”、已完成等待反馈“finished”、失败等待反馈“failed”、全部“all”。

page：标识当前要获取哪一页的结果。

pageSize：标识每页有多少条查询结果。

查询加密任务的JSON消息示例如下：

```
{
    "contentID":"string",
    "reqMode":"string",
    "page":0,
    "pageSize":20
}
```

查询加密任务的响应状态码为200，响应消息见表5。

表5　查询加密任务状态响应消息

JSON 键	值类型	必选/可选
code	string	必选
details	string	必选
count	number	必选
startNum	number	必选
page	number	必选
pageSize	number	必选
values	string	必选

code：返回状态代码，000正常、101参数异常、301系统异常。

details：详细的状态描述信息。

count：查询到的任务总数量。

startNum：当前页起始任务序号。

page：当前页码。

pageSize：当前页数据数量。

values：查询到的结果数据，数据描述见表6。

表6　查询结果数据数据描述

JSON 键	值类型	必选/可选
contentID	string	必选
sourceFilePath	string	必选
desFilePath	string	必选
responseURL	string	必选
status	string	必选
process	number	必选

contentID：内容管理系统中内容唯一标识。

sourceFilePath、desFilePath：待加密内容路径和加密后内容路径，应支持HTTP、FTP、NFS等路径格式。

responseURL：内容加密任务反馈加密结果消息的URI，示例为：https://<IP>:<Port>/response_addtask。

status：任务状态。包括：加密中“encrpting”、排队中“waiting”、已完成等待反馈“finished”、失败等待反馈“failed”、全部“all”。

process：任务完成的百分比。

查询任务状态消息示例如下：

```
{
    "code":"000",
    "count":100,
    "startNum":20,
    "page":1,
```

```
    "pageSize":20,
    "values":[
        {
            "contentID":"string",
            "sourceFilePath":"string",
            "desFilePath":"string",
            "responseURL":"string",
            "status":"string",
            "process":90
        },
        …]
}
```

6.3.4 删除加密任务

删除加密任务由内容管理系统发起，URI示例为：https://<IP>:<Port>/del_vod_enc_task。删除加密任务的JSON消息数据见7表。

表7 删除加密任务的 JSON 消息数据

JSON 键	值类型	必选/可选
contentID	string	必选
type	string	必选

contentID：内容管理系统中内容唯一标识。

type：默认为“del”，删除该任务；如果为“notified”表示加密机中处于finish/failed状态的任务收到notify通知，则认为媒资管理平台已收到加密任务的反馈，将会删除该任务信息。

删除加密任务的响应消息见表8。

表8 删除加密任务的响应消息

消息内容项	消息内容项描述
响应状态码	200

响应数据	{
	"code":"000",//错误代码,000 为成功,其他待定义
	"details":"error message"
	}

Code	101 参数异常 201 媒资不存在 202 操作失败（删除失败） 301 系统异常

7 密钥管理

7.1 直播密钥管理

7.1.1 概述

直播密钥管理为直播加密生成直播加密密钥，并负责将直播加密密钥同步到密钥网关。直播密钥管理通过直播内容加密密钥请求协议接收直播加密的密钥请求，为直播加密生成直播加密密钥，并同步到密钥网关，完成密钥同步后将直播内容加密密钥发送给直播加密。

7.1.2 直播内容加密密钥请求协议

7.1.2.1 直播内容加密密钥请求

直播内容加密密钥请求消息由直播加密发起，密钥管理验证并响应，直播内容加密密钥请求消息包括：版本号、直播加密唯一标识、随机数、直播频道唯一标识、直播加密模式、密钥更新方式、密钥更新频率、直播加密证书链、数字签名。直播内容加密密钥请求消息见表9。

表9 直播内容加密密钥请求消息

JSON 键	值类型	必选/可选
type	string	必选
version	string	必选
liveEncID	base64_string	必选
nonce	base64_string	必选
selectedAlgorithm	string	必选
contentID	base64_string	必选
contentEncMode	string	必选
cekUpdateMode	string	必选
cekUpdateFreq	string	必选
contentRules	base64_string	必选
extensions	对象数组	可选
certificateChain	base64_string 数组	必选
signature	base64_string	必选

type：消息类型，固定为“liveKeyRequest”。

version：当前消息协议版本号。当前版本Version默认为“1.0”。

liveEncID：直播加密唯一标识。

nonce：消息发送方产生的nonce，应由随机数生成器生成。

selectedAlgorithm：固定为“KMSProfile1”。

contentID：直播频道标识符。

contentEncMode：直播内容加密模式。“SM4-CBC”为全加密模式，“SAMPLE-SM4-CBC”为部分加密模式。

cekUpdateMode:直播内容加密密钥的更新方式,周期“period”或者固定每日某个时间点“fixed”。

cekUpdateFreq：直播内容加密密钥更新频率。如果更新方式为周期，使用十进制字符串，单位为秒（s），比如“60”表示60s；如果变换方式为fixed,使用“12:00:00”,表示12点0分0秒。

contentRules：内容使用规则，见表1。

extensions：可选的厂商自定义扩展信息，本文件不做规定。

certificateChain：直播加密证书链，该证书链不包括根证书。

signature：消息的签名。

直播内容加密密钥请求消息编码格式如下：

```
{
      "type":"liveKeyRequest",
      "version":"1.0",
      "liveEncID ":"base64_string",
      "nonce":"base64_string",
      "selectedAlgorithm":"string",
      "contentID":" base64_string",
      "contentEncMode":"string",
      "contentEncAlgorithm":"string",
      "cekUpdateMode":"string",
      "cekUpdateFreq":" string",
      "contentRules":"base64_string",
      "extensions":{…},
      "certificateChain":["base64_string","base64_string",…],
      "signature":"base64_string"
}
```

7.1.2.2 直播内容加密密钥响应

直播内容加密密钥响应消息包括：版本号、密钥管理唯一标识、随机数、直播频道标识、状态信息、会话密钥、当前密钥ID、加密后的当前密钥、下一密钥ID、加密后的下一密钥、密钥管理证书链、数字签名。直播内容加密密钥响应消息见表10。

表10 直播内容加密密钥响应消息

JSON 键	值类型	必选/可选
type	string	必选
version	string	必选
kmsID	base64_string	必选
nonce	base64_string	必选
status	string	必选
selectedAlgorithm	string	status=“success”时必选
cekInfo	对象	status=“success”时必选
contentID	base64_string	status=“success”时必选
sessionKeyID	base64_string	status=“success”时必选
encSessionKey	base64_string	status=“success”时必选
encCEKs	对象数组	status=“success”时必选
cekID	base64_string	status=“success”时必选
encCEK	base64_string	status=“success”时必选
extensions	对象数组	可选
certificateChain	base64_string 数组	必选
signature	base64_string	必选

type：消息类型，固定为“liveKeyResponse”。
version：当前消息协议版本号。当前版本Version默认为“1.0”。
kmsID：密钥管理唯一标识。
nonce：消息发送方产生的nonce，应与直播内容加密密钥请求消息的nonce一致。
status：反馈的状态信息，包括：请求成功、直播加密证书不合法、未知错误等，见表11。
selectedAlgorithm：固定为“KMSProfile1”。

表11　直播内容加密密钥响应状态信息

状态值	状态描述
success	查询成功
deviceCertInvalid	DRM 客户端证书不合法
liveEncCertInvalid	内容授权证书不合法
signatureInvalid	密钥同步请求消息数字签名不正确
unknownError	未知错误

cekInfo：内容加密密钥对象。
contentID：直播频道标识符。
sessionKeyID：会话密钥标识符。
encSessionKey：会话密钥为直播加密公钥加密的随机密钥。
encCEKs：内容加密密钥数组，包括了当前密钥和下一密钥。
cekID：内容加密密钥标识符。
encCEK：内容加密密钥为会话密钥加密的内容加密密钥。
extensions：可选的厂商自定义扩展信息，本文件不做规定。
certificateChain：密钥管理证书链，该证书链不包括根证书。
signature：消息的签名。
直播内容加密密钥响应消息编码格式如下：

```
{
    "type":" liveKeyResponse ",
    "version":"1.0",
    "kmsID":"base64_string",
    "nonce":"base64_string",
    "status":"string",
    "selectedAlgorithm":"string",
    "cekInfo":
        {
          "contentID":"base64_string",
          "sessionKeyID":"base64_string",
          "encSessionKey":"base64_string",
          "encCEKs":[
                {
                    "cekID":"base64_string",
```

```
                    "encCEK":"base64_string"
                },…],
            },
        "extensions":{…},
        "certificateChain":["base64_string","base64_string",…],
        "signature":"base64_string"
}
```

7.1.3 直播内容加密密钥同步协议

直播内容加密密钥同步协议见GY/T 277—2019中9.2。

7.2 点播密钥管理

7.2.1 概述

点播内容加密密钥申请由内容加密系统发起向密钥管理系统申请内容加密密钥,密钥管理系统为内容加密系统生成符合要求的内容加密密钥,并将内容加密密钥同步到密钥网关，同步成功后将内容加密密钥返回给内容加密系统。内容加密成功后通过点播内容加密处理消息通知密钥管理；如果内容加密失败，需要从密钥管理重新申请内容加密密钥时，密钥管理将其保存的该内容加密密钥返回给内容加密系统。如果内容管理系统通知内容加密系统某个内容需要删除，内容加密系统需要通过点播内容加密处理消息通知密钥管理系统删除该内容加密密钥,密钥管理系统接收到处理申请后应通知密钥网关删除该内容加密密钥，同时在本地删除该密钥，将处理结果返回给内容加密系统。

7.2.2 点播内容加密密钥申请

7.2.2.1 概述

点播内容加密密钥申请由内容加密系统发起向密钥管理系统申请内容加密密钥,密钥管理系统为内容加密系统生成符合要求的内容加密密钥返回给内容加密系统。点播内容加密密钥申请包括点播内容加密密钥请求消息、点播内容加密密钥响应消息。

7.2.2.2 点播内容加密密钥请求

点播内容加密密钥请求消息由内容加密系统发起，密钥管理验证并响应，点播内容加密密钥请求消息包括：版本号、内容加密系统唯一标识、随机数、内容唯一标识、内容加密模式、内容加密系统证书链、数字签名。点播内容加密密钥请求消息见表12。

表12 点播内容加密密钥请求消息

JSON 键	值类型	必选/可选
type	string	必选
version	string	必选
contentEncID	base64_string	必选
nonce	base64_string	必选
selectedAlgorithm	string	必选
contentID	base64_string	必选
contentEncMode	string	必选

表 12（续）

JSON 键	值类型	必选/可选
contentRules	base64_string	必选
extensions	对象数组	
certificateChain	base64_string 数组	必选
signature	base64_string	必选

type：消息类型，固定为“cekRequest”。

version：当前消息协议版本号。当前版本Version默认为“1.0”。

contentEncID：内容加密系统唯一标识。

nonce：消息发送方产生的nonce，应由随机数生成器生成。

selectedAlgorithm：固定为“KMSProfile1”。

contentID：内容唯一标识。

contentEncMode：内容加密模式。“SM4-CBC”为全加密模式，“SAMPLE-SM4-CBC”为部分加密模式。

contentRules：内容使用规则，见表1。

extensions：可选的厂商自定义扩展信息，本文件不做规定。

certificateChain：内容加密系统证书链，该证书链不包括根证书。

signature：消息的签名。

点播内容加密密钥请求消息编码格式如下：

```
{
      "type":"cekRequest",
      "version":"1.0",
      "contentEncID ":"base64_string",
      "nonce":"base64_string",
      "selectedAlgorithm":"string",
      "contentID":" base64_string",
      "contentEncMode":"string",
      "contentRules":"string",
      "extensions":{…},
      "certificateChain":["base64_string","base64_string",…],
      "signature":"base64_string"
}
```

7.2.2.3 点播内容加密密钥响应

点播内容加密密钥响应消息包括：版本号、密钥管理系统唯一标识、随机数、内容唯一标识、状态信息、会话密钥、内容加密密钥唯一标识、加密后的内容加密密钥、密钥管理系统证书链、数字签名。点播内容加密密钥响应消息见表13。

表13　点播内容加密密钥响应消息

JSON 键	值类型	必选/可选
type	string	必选
version	string	必选
kmsID	base64_string	必选
nonce	base64_string	必选
status	string	必选
selectedAlgorithm	string	status=“success”时必选
cekInfo	对象	status=“success”时必选
contentID	base64_string	status=“success”时必选
sessionKeyID	base64_string	status=“success”时必选
encSessionKey	base64_string	status=“success”时必选
encCEKs	对象数组	status=“success”时必选
cekID	base64_string	status=“success”时必选
encCEK	base64_string	status=“success”时必选
certificateChain	base64_string 数组	必选
signature	base64_string	必选

type：消息类型，固定为“cekResponse”。
version：当前消息协议版本号。当前版本Version默认为“1.0”。
kmsID：密钥管理唯一标识。
nonce：消息发送方产生的nonce，应与点播内容加密密钥请求消息的nonce一致。
status：反馈的状态信息，包括：请求成功、证书不合法、签名错误、未知错误等，见表14。
selectedAlgorithm：固定为“KMSProfile1”。

表14　点播内容加密密钥响应状态信息

状态值	状态描述
success	请求成功
doNotSupportSelectedAlgorithm	不支持请求的算法
contentIDInvalid	内容 ID 重复
contentEncCertInvalid	内容加密系统证书不合法
signatureInvalid	消息数字签名不正确
unknownError	未知错误

cekInfo：内容加密密钥对象。
contentID：内容唯一标识。
sessionKeyID：会话密钥唯一标识。
encSessionKey：会话密钥为内容加密系统公钥加密的随机密钥。
encCEKs：内容加密密钥数组。
cekID：内容加密密钥标识符。
encCEK：内容加密密钥为会话密钥加密的内容加密密钥。

certificateChain：密钥管理证书链，该证书链不包括根证书。

signature：消息的签名。

点播内容加密密钥响应消息编码格式如下：

```
{
      "type":" cekResponse ",
      "version":"1.0",
      "kmsID":"base64_string",
      "nonce":"base64_string",
      "status":"string",
      "selectedAlgorithm":"string",
      "cekInfos":
          {
            "contentID":"base64_string",
            "sessionKeyID":"base64_string",
            "encSessionKey":"base64_string",
            "encCEKs":[
                  {
                      "cekID":"base64_string",
                      "encCEK":"base64_string"
                  },…],
          },
      "certificateChain":["base64_string","base64_string",…],
      "signature":"base64_string"
}
```

7.2.3 点播内容加密处理

7.2.3.1 概述

点播内容加密处理是指内容加密系统通知密钥管理系统内容加密已完成或内容加密密钥需要删除等。点播内容加密处理包括点播内容加密密钥处理请求和点播内容加密密钥处理响应。

7.2.3.2 点播内容加密处理请求

点播内容加密处理请求消息由内容加密系统发送到密钥管理系统，包括：版本号、内容加密系统唯一标识、随机数、内容唯一标识、内容加密模式、内容加密系统证书链、数字签名。点播内容加密处理请求消息见表15。

表15 点播内容加密处理请求消息

JSON 键	值类型	必选/可选
type	string	必选
version	string	必选
contentEncID	base64_string	必选
nonce	base64_string	必选

表 15（续）

JSON 键	值类型	必选/可选
selectedAlgorithm	string	必选
contentID	base64_string	必选
processType	string	必选
certificateChain	base64_string 数组	必选
signature	base64_string	必选

type：消息类型，固定为“cekProcessRequest”。

version：当前消息协议版本号。当前版本Version默认为“1.0”。

contentEncID：内容加密系统唯一标识。

nonce：消息发送方产生的nonce，应由随机数生成器生成。

selectedAlgorithm：固定为“KMSProfile1”。

contentID：内容唯一标识。

processType：操作类型，包括内容加密完成、内容已删除，内容加密完成用“success”表示，内容已删除用“delete”表示。

certificateChain：内容加密系统证书链，该证书链不包括根证书。

signature：消息的签名。

点播内容加密请求消息编码格式如下：

```
{
      "type":"cekProcessRequest",
      "version":"1.0",
      "contentEncID ":"base64_string",
      "nonce":"base64_string",
      "selectedAlgorithm":"string",
      "contentID":" base64_string",
      "processType":"string",
      "certificateChain":["base64_string","base64_string",…],
      "signature":"base64_string"
}
```

7.2.3.3 点播内容加密处理响应

点播内容加密处理响应消息包括：版本号、密钥管理系统唯一标识、随机数、内容唯一标识、状态信息、密钥管理系统证书链、数字签名。点播内容加密处理响应消息见表16。

表16 点播内容加密处理响应消息

JSON 键	值类型	必选/可选
type	string	必选
version	string	必选
kmsID	base64_string	必选

表16（续）

JSON键	值类型	必选/可选
Nonce	base64_string	必选
status	string	必选
certificateChain	base64_string数组	必选
signature	base64_string	必选

type：消息类型，固定为“cekProcessResponse”。

version：当前消息协议版本号。当前版本Version默认为“1.0”。

kmsID：密钥管理唯一标识。

nonce：消息发送方产生的nonce，应与点播内容加密处理请求消息的nonce一致。

status：反馈的状态信息，包括：请求成功、证书不合法、签名错误、未知错误等，见表17。

selectedAlgorithm：固定为“KMSProfile1”。

表17　点播内容加密处理响应状态信息

状态值	状态描述
success	请求成功
doNotSupportSelectedAlgorithm	不支持请求的算法
contentEncCertInvalid	内容加密系统证书不合法
signatureInvalid	消息数字签名不正确
unknownError	未知错误

certificateChain：密钥管理证书链，该证书链不包括根证书。

signature：消息的签名。

点播内容加密密钥响应消息编码格式如下：

```
{
    "type":" cekProcessResponse ",
    "version":"1.0",
    "kmsID":"base64_string",
    "nonce":"base64_string",
    "status":"string",
    "certificateChain":["base64_string","base64_string",…],
    "signature":"base64_string"
}
```

7.2.4　点播内容加密密钥同步

内容加密系统通知密钥管理系统内容加密完成时，密钥管理系统将内容加密密钥同步到密钥网关，密钥管理与密钥网关之间的密钥同步接口见GY/T 277—2019中9.2。

当内容管理删除上线内容时，即密钥管理系统接收到内容加密密钥删除请求时，应向密钥网关发送密钥删除消息，通知密钥网关删除内容加密密钥；该删除消息采用内容加密系统与密钥管理系统之间发送的点播内容加密处理消息。

8 内容授权

8.1 概述

内容授权负责接收DRM客户端内容授权请求消息，从认证鉴权计费查询DRM客户端授权信息，从密钥网关请求直播频道内容加密密钥，封装成直播频道内容授权许可证发送给DRM客户端。

内容授权与DRM客户端的交互机制见GY/T 277—2019中第8章。内容授权与密钥网关的交互机制见GY/T 277—2019中9.3。内容授权与认证鉴权计费之间通过授权同步和授权查询的方式进行终端授权信息交互。

8.2 直播频道授权

8.2.1 直播频道授权同步

内容授权与运营支撑系统的直播频道授权同步包括：授权同步请求和授权同步响应。授权同步请求由运营支撑系统发送给内容授权，授权同步响应由内容授权返回给运营支撑系统。

授权同步请求消息包括：版本号、运营支撑唯一标识、随机数、授权信息、运营支撑证书链、数字签名。直播频道授权同步请求消息见表18。

表18 直播频道授权同步请求消息

JSON 键	值类型	必选/可选
type	string	必选
version	string	必选
aaaID	base64_string	必选
nonce	base64_string	必选
selectedAlgorithm	string	必选
channelAuthInfos	base64_string	必选
channelID	base64_string	必选
channelRules	base64_string	必选
certificateChain	base64_string 数组	必选
signature	base64_string	必选

type：消息类型，固定为“channelRightsSync”。

version：当前消息协议版本号。当前版本Version默认为“1.0”。

aaaID：运营支撑唯一标识。

nonce：消息发送方产生的nonce，应由随机数生成器生成。

selectedAlgorithm：固定为“KMSProfile1”。

channelAuthInfos：直播频道授权信息，包括直播频道唯一标识、直播频道授权信息。

channelID：直播频道唯一标识。

channelRules：直播频道授权信息，见表1。

certificateChain：运营支撑证书链，该证书链不包括根证书。

signature：消息的签名。

直播频道授权同步请求消息编码格式如下：

```
{
    "type":"channelRightsSync",
    "version":"1.0",
    "aaaID":"base64_string",
    "nonce":"base64_string",
    "selectedAlgorithm":"string",
    "channelAuthInfos":[
        {
            "channelID":"base64_string",
            "channelRules":"base64_string"
        },…],
    "certificateChain":["base64_string","base64_string",…],
    "signature":"base64_string"
}
```

直播频道授权同步响应消息包括：版本号、内容授权唯一标识、随机数、状态信息、内容授权证书链、数字签名。直播频道授权同步响应消息见表19。

表19　直播频道授权同步响应消息

JSON 键	值类型	必选/可选
type	string	必选
version	string	必选
drmServerID	base64_string	必选
Nonce	base64_string	必选
status	string	必选
certificateChain	base64_string 数组	必选
signature	base64_string	必选

type：消息类型，固定为“channelRightsSyncResponse”。

version：当前消息协议版本号。当前版本Version默认为“1.0”。

drmServerID：内容授权唯一标识。

nonce：消息发送方产生的nonce，应与直播频道授权同步请求消息的nonce一致。

status：反馈的状态信息，包括：请求成功、运营支撑证书不合法、消息数字签名不正确、未知错误等，见表20。

表20　直播频道授权同步响应状态信息

状态值	状态描述
success	请求成功
aaaCertInvalid	运营支撑证书不合法
signatureInvalid	消息数字签名不正确
unknownError	未知错误

certificateChain：内容授权证书链，该证书链不包括根证书。

signature：消息的签名。

直播频道授权同步响应消息编码格式如下：

```
{
      "type":"channelRightsSyncResponse ",
      "version":"1.0",
      "drmServerID":"base64_string",
      "nonce":"base64_string",
      "status":"string",
      "certificateChain":["base64_string","base64_string",…],
      "signature":"base64_string"
}
```

8.2.2　直播频道授权查询

直播频道授权查询是内容授权从运营支撑查询指定的DRM客户端的直播频道授权信息，包括直播频道授权查询请求和直播频道授权查询响应，直播频道授权查询请求由内容授权发送给运营支撑系统，直播频道授权查询响应由运营支撑系统返回给内容授权。

直播频道授权查询请求包括：版本号、DRM服务端唯一标识、随机数、直播频道唯一标识、DRM客户端唯一标识、内容授权证书链、数字签名。直播频道授权查询请求消息见表21。

表21　直播频道授权查询请求消息

JSON 键	值类型	必选/可选
type	String	必选
version	String	必选
drmServerID	base64_string	必选
nonce	base64_string	必选
selectedAlgorithm	string	必选
drmClientID	base64_string	必选
channelIDs	对象数组	可选
certificateChain	base64_string 数组	必选
signature	base64_string	必选

type：消息类型，固定为“channelRightsRequest”。

version：当前消息协议版本号。当前版本Version默认为“1.0”。

drmServerID：内容授权的唯一标识。

nonce：消息发送方产生的nonce，应由随机数生成器生成。

drmClientID：DRM客户端唯一标识。

channelIDs：直播频道唯一标识符对象数组，该参数不存在时表示查询该DRM客户端对应的所有频道的授权信息。

selectedAlgorithm：固定为“KMSProfile1”。

certificateChain：内容授权证书链，该证书链不包括根证书。

signature：消息的签名。

直播频道授权查询请求消息编码格式如下：

```
{
      "type":" channelRightsRequest ",
      "version":"1.0",
      "drmServerID":"base64_string",
      "nonce":"base64_string",
      "selectedAlgorithm":"string",
      "drmClientID":"base64_string",
      "channelIDs":["base64_string",…],
      "certificateChain":["base64_string","base64_string",…],
      "signature":"base64_string"
}
```

直播频道授权查询响应消息包括：版本号、运营支撑唯一标识、随机数、状态信息、DRM客户端唯一标识、授权信息、运营支撑证书链、数字签名等。直播频道授权查询响应消息见表22。

表22　直播频道授权查询响应消息

JSON 键	值类型	必选/可选
type	string	必选
version	string	必选
aaaID	base64_string	必选
nonce	base64_string	必选
status	string	必选
selectedAlgorithm	string	必选
drmClientID	base64_string	status=“success”时必选
authInfos	base64_string 数组	status=“success”时必选
channelID	base64_string	status=“success”时必选
channelRules	base64_string	status=“success”时必选
certificateChain	base64_string 数组	必选
signature	base64_string	必选

type：消息类型，固定为“channelRightsResponse”。

version：当前消息协议版本号。当前版本Version默认为“1.0”。

aaaID：运营支撑唯一标识。

nonce：消息发送方产生的nonce，应与直播频道授权查询请求消息的nonce一致。

status：反馈的状态信息，包括：查询成功、找不到该DRM客户端ID、找不到直播频道ID、内容授权证书不合法、未知错误等，见表23。

selectedAlgorithm：固定为“KMSProfile1”。

表23　直播频道授权查询响应状态信息

状态值	状态描述
success	查询成功
doNotSupportSelectedAlgorithm	不支持请求的算法
contentIDInvalid	找不到该内容 ID
deviceIDInvalid	DRM 客户端 ID 不合法
drmServerCertInvalid	内容授权证书不合法
signatureInvalid	数字签名不正确
unknownError	未知错误

drmClientID：DRM客户端唯一标识。

authInfos：认证授权信息。

channelID：直播频道唯一标识。

channelRules：直播频道授权信息，按照密钥使用规则的方式封装，见表1，KeyRuleType见GY/T 277—2019中表14。

certificateChain：运营支撑证书链，该证书链不包括根证书。

signature：消息的签名。

直播频道授权查询响应消息编码格式如下：

```
{
      "type":" channelRightsResponse ",
      "version":"1.0",
      "aaaID":"base64_string",
      "nonce":"base64_string",
      "status":"string",
      "selectedAlgorithm":"string",
      "drmClientID":"base64_string",
      "authInfos":[
          {
              "channelID":"base64_string",
              "channelRules":"base64_string"
          },…],
      "certificateChain":["base64_string","base64_string",…],
      "signature":"base64_string"
}
```

8.3　点播内容授权

点播内容授权通过点播内容授权查询协议从运营支撑查询指定的DRM客户端的点播内容授权信息，包括点播内容授权查询请求和点播内容授权查询响应，点播内容授权查询请求由内容授权发送给运营支撑系统，点播内容授权查询响应由运营支撑系统返回给内容授权。

点播内容授权查询请求包括：版本号、DRM服务端唯一标识、随机数、内容唯一标识、DRM客户端唯一标识、内容授权证书链、数字签名。点播内容授权查询请求消息见表24。

表24　点播内容授权查询请求消息

JSON 键	值类型	必选/可选
type	String	必选
version	String	必选
drmServerID	base64_string	必选
nonce	base64_string	必选
selectedAlgorithm	string	必选
drmClientID	base64_string	必选
contentIDs	对象数组	可选
certificateChain	base64_string 数组	必选
signature	base64_string	必选

type：消息类型，固定为“contentRightsRequest”。

version：当前消息协议版本号。当前版本Version默认为“1.0”。

drmServerID：内容授权的唯一标识。

nonce：消息发送方产生的nonce，应由随机数生成器生成。

drmClientID：DRM客户端唯一标识。

contentIDs：点播内容唯一标识符对象数组。

selectedAlgorithm：固定为“KMSProfile1”。

certificateChain：内容授权证书链，该证书链不包括根证书。

signature：消息的签名。

直播频道授权查询请求消息编码格式如下：

```
{
      "type":" contentRightsRequest ",
      "version":"1.0",
      "drmServerID":"base64_string",
      "nonce":"base64_string",
      "selectedAlgorithm":"string",
      "drmClientID":"base64_string",
      "contentIDs":["base64_string",…],
      "certificateChain":["base64_string","base64_string",…],
      "signature":"base64_string"
}
```

点播内容授权查询响应消息包括：版本号、运营支撑唯一标识、随机数、状态信息、DRM客户端唯一标识、授权信息、运营支撑证书链、数字签名等。点播内容授权查询响应消息见表25。

表25　点播内容授权查询响应消息

JSON 键	值类型	必选/可选
type	string	必选
version	string	必选
aaaID	base64_string	必选
nonce	base64_string	必选
status	string	必选
selectedAlgorithm	string	必选
drmClientID	base64_string	status=“success”时必选
authInfos	base64_string 数组	status=“success”时必选
contentID	base64_string	status=“success”时必选
contentRules	base64_string	status=“success”时必选
certificateChain	base64_string 数组	必选
signature	base64_string	必选

type：消息类型，固定为“contentRightsResponse”。

version：当前消息协议版本号。当前版本Version默认为“1.0”。

aaaID：运营支撑唯一标识。

nonce：消息发送方产生的nonce，应与直播频道授权查询请求消息的nonce一致。

status：反馈的状态信息，包括：查询成功、不支持请求的算法、找不到该内容ID、DRM客户端ID不合法、内容授权证书不合法、数字签名不正确、以及未知错误等，见表26。

selectedAlgorithm：固定为“KMSProfile1”。

表26　点播内容授权查询响应状态信息

状态值	状态描述
success	查询成功
doNotSupportSelectedAlgorithm	不支持请求的算法
contentIDInvalid	找不到该内容 ID
deviceIDInvalid	DRM 客户端 ID 不合法
drmServerCertInvalid	内容授权证书不合法
signatureInvalid	数字签名不正确
unknownError	未知错误

drmClientID：DRM客户端唯一标识。

authInfos：认证授权信息。

contentID：点播内容唯一标识。

contentRules：点播内容授权信息，见表1。

certificateChain：运营支撑证书链，该证书链不包括根证书。

signature：消息的签名。

直播频道授权查询响应消息编码格式如下：

```
{
      "type":" channelRightsResponse ",
      "version":"1.0",
      "aaaID":"base64_string",
      "nonce":"base64_string",
      "status":"string",
      "selectedAlgorithm":"string",
      "drmClientID":"base64_string",
      "authInfos":[
          {
              "contentID":"base64_string",
              "contentRules":"base64_string"
          },…],
      "certificateChain":["base64_string","base64_string",…],
      "signature":"base64_string"
}
```

9 终端集成

9.1 终端集成方法

DRM客户端在终端设备的集成主要是DRM客户端与播放应用的集成。

DRM客户端运行环境应由运行在TEE中的安全操作系统提供，DRM客户端运行环境接口见 GY/T 277—2019附录D，DRM客户端功能库通过调用DRM客户端运行环境接口实现DRM客户端核心功能，DRM客户端TA通过DRM客户端功能接口调用DRM客户端功能库的基础功能，DRM客户端功能接口见GY/T 277—2019附录C。

DRM客户端TA通过TEE通信机制为播放应用提供DRM客户端功能调用，播放应用调用的DRM客户端功能应符合GY/T 277—2019附录C定义的接口。

9.2 DRM 客户端证书置入方法

DRM客户端证书和私钥由证书管理系统签发，证书管理系统签发的DRM客户端证书和私钥采用PGP方式安全的分发给运营商或终端设备制造商，并由运营商或终端设备制造商在终端设备生产时在产线安全置入终端设备中。

在产线烧写时，DRM客户端私钥应采用随机传输密钥加密后，与DRM客户端证书链、加密后的传输密钥、以及其他参数发送到终端设备，终端设备的TEE环境中应包含由芯片厂商提供的DRM_KEY_TA，产线烧写工具与DRM_KEY_TA进行交互，实现DRM客户端私钥、DRM客户端证书链等的安全存储。终端设备制造商应采用附录A中规定的接口实现DRM客户端证书及私钥的置入。

DRM_KEY_TA将DRM客户端证书及私钥的存储和使用分开，DRM_TA只能通过接口读取和使用DRM_KEY_TA存储的DRM客户端证书及私钥，但无法获取到私钥数据，同时DRM_KEY_TA可支持多个DRM_TA访问，从而保证在同一系统内部可支持DRM_TA的增加及更新，DRM客户端私钥及证书的置入整体业务流程图如图2所示。

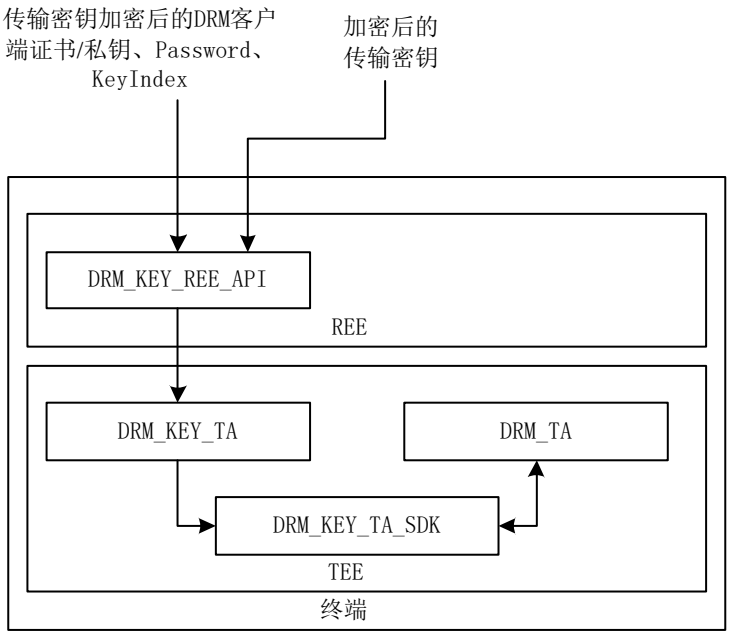

图2　DRM 客户端私钥及证书置入整体业务流程图

传输密钥由运营商指定，用于加密DRM客户端证书及私钥、Password。DRM_KEY_TA应由芯片厂商实现,芯片厂商可针对传输密钥提供层级加密技术,以增强安全性,传输密钥需发送到芯片厂商进行加密，该加密流程可使用层级加密技术等由一个已经预留在OTP存储中密钥进行安全保护。

芯片厂商应将加密后的传输密钥提供运营商,运营商将加密后的传输密钥,加密的DRM客户端证书及私钥、Password、KeyIndex等交给终端设备制造商。

终端设备制造商应通过A.1的接口将加密的传输密钥、加密的DRM客户端证书及私钥、Password、KeyIndex等写入到终端设备TEE的DRM_KEY_TA。

DRM_KEY_TA完成解密后，对DRM客户端证书及私钥使用Password再次加密，与KeyIndex配对存入安全存储区域。

DRM_TA在运行过程中应通过A.2规定的接口使用DRM客户端证书和私钥。

9.3　DRM 客户端升级改造

DRM客户端支持基于TEE等的硬件安全运行环境，则DRM客户端升级改造时应在终端集成硬件安全及以上的等级的DRM客户端。

DRM客户端不具备硬件安全运行环境，应基于白盒密码等软件的方式实现DRM客户端运行环境，集成软件安全等级DRM客户端。在该情况下，DRM客户端运行环境接口API是DRM客户端运行环境提供的运行时库（run-time library），DRM客户端运行环境接口API见GY/T 277—2019中附录D。DRM客户端功能库通过调用DRM客户端运行环境接口API实现DRM客户端核心功能，DRM客户端功能库通过DRM客户端功能接口为DRM客户端提供基础功能，其功能接口API见GY/T 277—2019中附录C。

升级改造的DRM客户端应通过在线方式置入DRM客户端证书和密钥，应采用A.2定义的接口扩展。

附 录 A
（规范性）
DRM 客户端证书置入接口 API

A.1 DRM客户端证书离线烧写接口

离线烧写接口是终端设备制造商产线烧写时需在终端设备侧调用的烧写接口。

原 型： int CDRM_Key_Provision(unsigned char* pTransportKey, int s32TransportKeyLen, unsigned char* pInfor, int s32Inforlen, unsigned char* pPrivateKey, int s32privateKeyLen, unsigned char* pCert, int s32CertLen)

参数：pTransportKey—输入参数，芯片厂商协助加密后的TransportKey；

s32TransportKeyLen—输入参数，TransportKey的长度；

pInfor—输入参数，passwd信息；

s32Inforlen—输入参数，信息的长度；

pPrivateKey—输入参数，加密后的私钥数据地址；

s32privateKeyLen—输入参数，加密后的私钥长度；

pCert—输入参数，加密的证书数据地址；

s32CertLen—输入参数，加密的证书数据地址。

返回：int，0表示成功，其他表示失败。

备注:调用该离线烧写接口时,在DRM客户端安全运行环境中存储的DRM客户端密钥的Index默认为0。

为支持离线烧写时终端设备制造商校验烧写流程，DRM客户端烧写需提供校验接口。

原型：int CDRM_Key_IsValid(unsigned char* pInfor, int s32Inforlen, int keyIndex)

参数：pInfor—输入参数，passwd信息，运营商提供；

s32Inforlen—输入参数，信息的长度；

int—输入参数，keyIndex。

返回：int，0表示成功，其他表示失败。

A.2 DRM客户端运行环境接口扩展

A.2.1 概述

DRM客户端运行环境接口扩展是指对GY/T 277—2019中附录D的扩展，该扩展用于DRM客户端密钥及证书的安全置入，以及DRM SDK对DRM客户端证书的读取和利用DRM客户端私钥进行密码运算等。

A.2.2 CDRM_Key_Insert_KeySlot

该接口用于在线分发DRM客户端密钥及证书时DRM客户端密钥和证书的安全存储。

原型：int CDRM_Key_Insert_KeySlot(unsigned char* pTransportKey, int s32TransportKeyLen, int* keyIndex, unsigned char* pPasswd, int s32Passlen, unsigned char* pPrivateKey, int s32privateKeyLen, unsigned char* pCert, int s32CertLen)

参数：pTransportKey—输入参数，芯片厂商协助加密后的TransportKey；

s32TransportKeyLen—输入参数，TransportKey的长度；

keyIndex—输入参数，新的keyIndex，如果输入是小于0，则认为没有输入keyIndex，则生成新的keyIndex返回；如果输入大于0，尝试更新或插入，更新时校验password，如校验失败，则更新失败，插入不验证password；输入0，返回失败；更新时password不能更新；

pPasswd—输入参数，由ChinaDRM TA随机生成的passwd数据；

s32Passlen—输入参数，passwd的长度；

pPrivateKey—输入参数，ChinaDRM的私钥，明文输入，需要用transportkey加密；

s32privateKeyLen—输入参数，ChinaDRM私钥的长度；

pCert—输入参数，ChinaDRM的证书，明文输入，需要用transportkey加密；

s32CertLen—输入参数，ChinaDRM证书的长度。

返回：int，0表示成功，其他表示失败。

A.2.3 CDRM_Key_GetNewKeyIndex

该接口用于DRM TA申请一个新的存储私钥用的KeyIndex。

原型：int CDRM_Key_GetNewKeyIndex (unsigned int* pNewKeyIndex)

参数：pNewKeyIndex—输出参数，生成的新的keyindex。

返回：int，0表示成功，其他表示失败。

A.2.4 CDRM_Key_GetCert

该接口用于DRM TA获取证书数据。

原型：int CDRM_Key_GetCert (unsigned int keyIndex, unsigned char* pPasswd, int s32Passlen, unsigned char* pCert, int* pCertLen)

参数：keyIndex—输入参数，该参数需要ChinaDRM TA存储在安全存储中，需要使用时传入；

pPasswd—输入参数，该参数需要ChinaDRM TA存储在安全存储中，需要使用时传入；

s32Passlen—输入参数，Passwd的长度；

pCert—输出参数，证书的数据；

pCertLen—输出参数，证书的长度。

返回：int，0表示成功，其他表示失败。

A.2.5 CDRM_Key_PrivateKey_Signature

该接口用于DRM TA使用私钥进行签名，该算法只支持国密。

原型：int CDRM_Key_PrivateKey_Signature (unsigned int keyIndex, CDRM_KEY_Algorithm alg, unsigned char* pPasswd, int s32Passlen, unsigned char* pInputBuffer, int len, unsigned char* pOutPutBuffer, int* pBufferLen)

参数：keyIndex—输入参数，该参数需要ChinaDRM TA存储在安全存储中，需要使用时传入；

alg—输入参数，算法参数，现在保留；

pPasswd—输入参数，该参数需要ChinaDRM TA存储在安全存储中，需要使用时传入；

s32Passlen—输入参数，Passwd的长度；

pInputBuffer—输入参数，传入需要签名的数据；

len—输入参数，签名数据的长度；

pOutPutBuffer—输出参数，签名的数据地址；

pBufferLen—输出参数，签名数据的长度。

返回：int，0表示成功，其他表示失败。

A.2.6 CDRM_Key_PrivateKey_Decrypt

该接口用于DRM TA使用私钥进行解密，该算法现在只支持国密。

原型：int CDRM_Key_PrivateKey_Decrypt(unsigned int keyIndex, CDRM_KEY_Algorithm alg, unsigned char* pPasswd, int s32Passlen, unsigned char* pInputBuffer, int inLen, unsigned char* pOutPutBuffer, int* pBufferLen)

参数：keyIndex—输入参数，该参数需要ChinaDRM TA存储在安全存储中，需要使用时传入；

alg—输入参数，算法参数，现在保留；

pPasswd—输入参数，该参数需要ChinaDRM TA存储在安全存储中，需要使用时传入；

s32Passlen—输入参数，Passwd的长度；

pInputBuffer—输入参数，传入需要解密的数据；

len—输入参数，解密数据的长度；

pOutPutBuffer—输出参数，解密后数据的地址；

pBufferLen—输出参数，解密后数据的长度。

返回：int，0表示成功，其他表示失败。