# 点播

# 视频播放控制

# 产品文档

腾讯云

# 文档目录

视频播放控制
视频加密
防盗链
防盗链综述
Referer 防盗链
Key 防盗链

# 视频播放控制
# 视频加密

最近更新时间：2018-09-05 17:37:48

数字版权管理（Digital Rights Management，DRM），对于在线教育、行业培训等领域都是十分重要的。视频文件的泄露，有可能造成十分严重的经济损失。

传统的视频服务提供商大多是通过各种各样的防盗链机制来实现视频内容保护。该方案的基本原理是：APP服务端给客户派发专属的视频URL，CDN服务对请求URL、请求IP、HTTP头等参数进行校验，如果校验通过，则返回正常的视频数据；否则返回403错误码。

但是，对于需要付费观看的视频，一旦恶意用户通过一次付费行为拿到了合法的防盗链播放URL，其便可以将视频完整下载到本地，进而实现二次分发。因此，防盗链方案对于视频版权保护是远远不够的。要进一步提升视频内容的保护程度，就不能仅仅在视频的分发环节做文章，而是必须对视频数据本身的加密。对视频数据加密之后，即使恶意用户把视频下载到本地，视频本身也是被加密的，这样就提高了恶意用户将视频内容二次分发的门槛。

点播视频加密DEMO参见这里，登录用户名：test，密码：111111。

## 腾讯云点播视频加密方案

### 加密算法

点播系统目前支持HTTP Live Streaming中规定的加密方案，该方案的安全级别可以达到：

1. 使用AES-128对视频内容本身进行加密；
2. 支持对单个视频文件使用多个密钥进行加密，避免单个密钥泄露导致整个文件泄密。

如果您需要定制私有化的视频加密方案，可以与腾讯云客服联系。

### 播放器适配性

腾讯云点播视频加密方案能够支持所有HLS播放器。

## 术语介绍

### 密钥管理服务（Key Management Service，简称KMS）

一项安全管理服务，主要负责数据密钥的生产、加密、解密等工作。例如腾讯云的密钥管理服务。

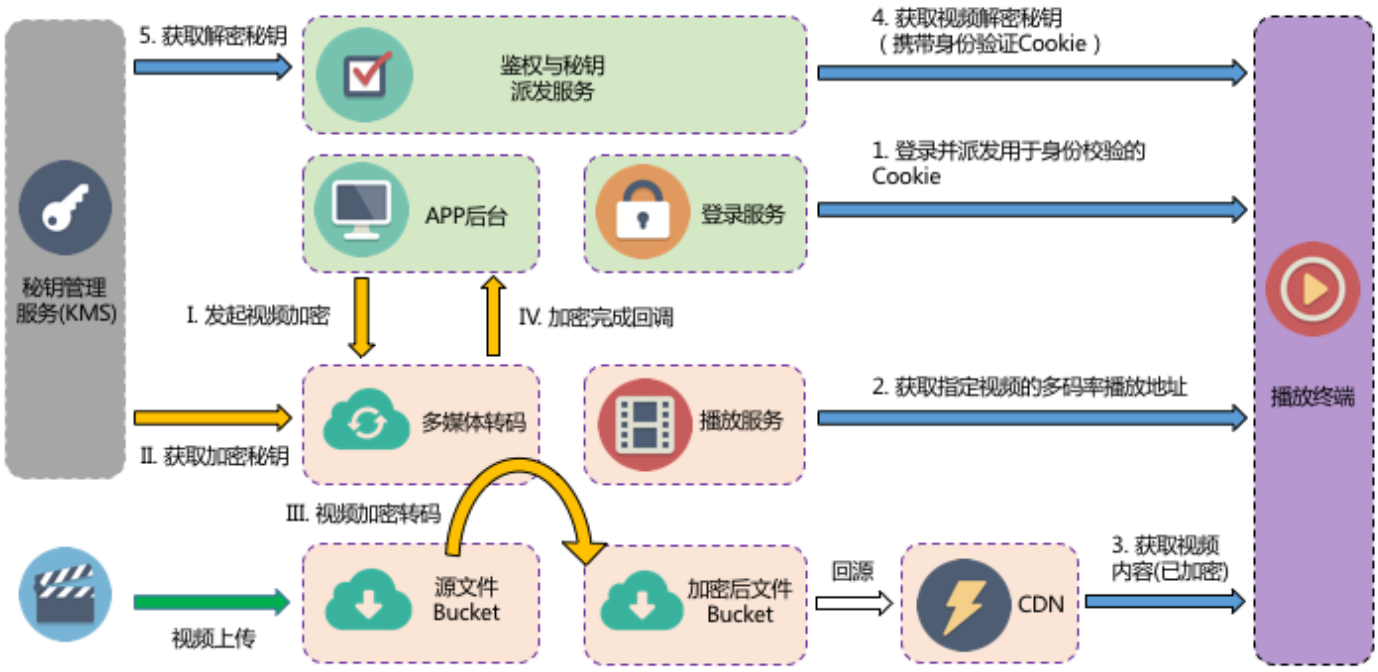
### 数据密钥（Data Key，简称DK）

由KMS系统生成的，用于对称加解密的密钥。

### 加密后的数据密钥（Encrypted Data Key，简称EDK）

经过KMS系统加密之后的DK，可以用于公开分发。要通过EDK换取DK，必须调用KMS的解密接口。

## 整体架构




视频加密过程是通过转码操作来实现的，不会产生新的FileID。与一般转码场景相比，视频加密转码的主要区别在于：

1. 加密转码，转出来的视频是经过加密的；
2. 加密转码完成之后，如果通过点播播放器来播放视频，源文件播放地址是不会被获取到的。

## 准备工作

### 建立密钥管理服务（KMS）

密钥管理服务主要用于管理视频密钥。视频加密过程需要与KMS系统进行交互的步骤包括：

1. 生成用于视频加密的数据密钥，即架构图中第II步。这一步将返回DK和EDK。在后续环节中，能够接触到DK的角色包括：点播转码服务、APP后台、经过合法身份校验的最终用户。EDK可以分发给任意用户，但通过EDK获取DK这一步，必须由APP后台进行身份校验。
2. 根据EDK获取DK来进行数据播放，即架构图中的第4步。APP后台在通过用户的身份验证之后，需要调用KMS相关接口，使用EDK去获取DK，即架构图中的第5步。

腾讯云点播系统支持以下三种密钥管理服务：

1. 腾讯云点播内置KMS服务：为最大限度地降低开发者的接入成本，腾讯云点播服务内部集成了KMS服务，并且提供了最简单的调用接口。如果使用点播内置KMS服务，则在整个视频加密方案中，APP后台与KMS服务唯一需要交互的地方在于获取解密密钥（架构图中的第5步）。
2. 腾讯云KMS服务（即将支持）：在开通腾讯云KMS服务之后，您可以将根据某个主密钥来生成数据密钥的权限授权给点播服务，点播系统便可融入您在云端的密钥管理体系中。
3. 自建KMS服务（即将支持）：如果您已经自建了KMS服务，您可以在每次加密操作中自行指定DK和EDK，从而达到最灵活的密钥控制级别。

### 搭建鉴权与密钥派发服务

对于已经加密的视频，只有经过APP后台认证过的客户端才能得到DK。因此，最终客户获取密钥的行为必须要有APP后台参与鉴权。该服务的主要业务逻辑是：

1. 对于客户端携带EDK换取DK的请求（即架构图中第4步），对请求方进行身份认证；
2. 如果身份认证通过，则去KMS系统获取对应的DK（即架构图中第5步），并返回给客户端。

建议：

1. 由于EDK所对应的的DK总是固定的，故而APP后台可以缓存（甚至永久保存）EDK和DK之间的对应关系，以降低调用KMS系统的次数（即减少架构图中第5步的调用次数）；
2. APP后台给客户端的应答，可以增加HTTP缓存控制参数（例如Cache-Control），以降低客户端到APP后台获取DK的次数（即减少架构图中第4步的调用次数）。

### 配置视频加密模板

为确保点播后台能够进行正确的加密操作，您需要配置视频加密模板。详情参见视频加密参数模板。

# 业务流程

# 视频上传

可以通过服务端上传、客户端上传、控制台上传、录制上传、URL转拉上传等方式来将已有视频文件上传到点播平台。

# 视频加密

视频加密主要分为以下四个步骤：

### I. APP后台发起视频加密

目前您可以通过ProcessFile接口发起视频加密，目前只支持对HLS文件进行加密。

如下示例的含义是：

1. 对视频文件进行转码，转码目标输出模板为210、220、230、240；禁止从较低码率转为较高码率；
2. 转码过程使用加密模板10进行加密；
3. 事件通知模式为：待整个事件执行完毕之后发起一次事件通知。

```
https://vod.api.qcloud.com/v2/index.php?Action=ProcessFile
&transcode.definition.0=210
&transcode.definition.1=220
&transcode.definition.3=230
&transcode.definition.4=240
&transcode.drm.definition=10
&notifyMode=Finish
&COMMON_PARAMS
```

### II. 点播平台获取加密密钥

点播平台根据调用方指定的加密参数模板，读取密钥获取方式、最终用户获取解密密钥的URL（假定为 `https://getkey.example.com` ），然后从指定KMS系统中获取视频加密密钥DK、EDK。

### III. 点播平台发起视频加密转码

点播转码平台在进行视频加密时，不仅会依照指定的加密算法和密钥对目标输出文件进行加密，而且会将获取解密密钥的URL写入视频文件中。例如，对于HLS，该URL会被写入到m3u8文件的EXT-X-KEY标签中。但在写入之前，转码平台会在该URL的QueryString中增加三个参数：

1. fileID：被加密文件的ID；
2. keySource：KMS的类型，为以下三种之一：
   i. VodBuildInKMS：腾讯云点播内置KMS；
   ii. QCloudKMS：腾讯云KMS系统（暂不支持）；
   iii. PrivateKMS：用于自有KMS系统（暂不支持）。
3. edk：即DK对应的EDK。

在增加上述参数之后，写入转码目标视频文件的URL可能为

```
https://getkey.example.com?fileId=123456&keySource=VodBuildInKMS&edk=abcdef
```

该URL也是客户端最终在视频播放过程中获取解密密钥时访问的URL。

### IV. 点播平台发起加密完成回调

包含加密操作的任务流状态发生变化（或者执行完毕）之后，点播平台将发起任务流状态变更通知。

## 媒资管理

视频加密操作完成之后，可以通过GetVideoInfo接口获取视频的加密信息。
GetVideoInfo接口会返回该视频Id所有转码规格的视频播放地址，包括源文件的播放地址，由于源文件是没有加密处理的，APP服务端可以过滤掉源文件的播放地址，只提供加密视频的播放地址给客户端。GetVideoInfo获取到的源文件definition参数是0，可以根据这个值来过滤源文件的视频播放地址。

## 视频播放综述

只有经过合法身份认证的客户才应当得到视频解密密钥。因此在播放过程中，如何对用户的身份信息校验就成为关键因素。

播放过程中，播放器会访问m3u8文件中EXT-X-KEY标签所标识的URL以获取密钥，播放器需要在这一步中携带观看者的身份认证信息。此时有两种方式可以将这一信息传递给APP鉴权服务：

1. 将用户身份信息通过参数的方式追加到URL中，带给APP的鉴权服务；该方案适用于所有的HLS播放器。具体方案参见视频播放方案1：通过QueryString传递身份认证信息。
2. 将用户身份信息通过Cookie带给APP的鉴权服务；该方案安全性更高，但其仅适用于在访问EXT-X-KEY标签所标识的URL时会携带Cookie的播放器。具体方案参见视频播放方案2：通过Cookie传递身份认证信息。

## 视频播放方案1：通过QueryString传递身份认证信息

该方案适用于任意支持HLS的播放器。

### 1. 登录并派发用于身份校验的Token

只有经过合法身份认证的客户才应当得到视频解密密钥。因此在视频播放之前，客户端必须进行登录操作，并由APP服务端给客户端派发包含身份认证信息的签名，我们称其为Token。

### 2. 获取包含Key防盗链签名的多码率播放地址

加密转码API ProcessFile的回调通知或者GetVideoInfo API都可以获取到加密视频的多码率播放地址。

在拿到多码率播放地址之后，客户端需要将用户身份信息添加到播放地址中。对于任意播放URL，增加用户身份信息的方法是：在URL中的**文件名**之前增加 voddrm.token.<Token> 。

例如，假定用户身份信息标识为ABC123；某一码率的播放地址为

```
http://example.vod2.myqcloud.com/path/to/a/video.m3u8
```

则最终URL为

```
http://example.vod2.myqcloud.com/path/to/a/voddrm.token.ABC123.video.m3u8
```

### 3. 获取视频内容(已加密)

当播放器访问已经按照上一步所述流程携带用户身份信息的URL时，点播后台会自动将Token信息以QueryString的方式附加到原始m3u8文件EXT-X-KEY标签所标识的URL中。

例如，假定某一码率的已加密视频URL为

```
http://example.vod2.myqcloud.com/path/to/a/video.m3u8
```

该文件中，EXT-X-KEY标签所标识的获取视频解密密钥的URL为

```
https://getkey.example.com?fileId=123456&keySource=VodBuildInKMS&edk=abcdef
```

则当播放器访问携带Token信息的播放地址，即

```
http://example.vod2.myqcloud.com/path/to/a/voddrm.token.ABC123.video.m3u8
```

其中EXT-X-KEY标签所标识的获取视频解密密钥的URL会被替换为

```
https://getkey.example.com?fileId=123456&keySource=VodBuildInKMS&edk=abcdef&token=ABC123
```

此时，播放器获取解密密钥dk时便会带上第1步派发的Token。

### 4/5. 获取视频解密密钥（携带身份验证Cookie）

当播放器获取到视频索引文件（m3u8文件）之后，会在播放视频文件之前自动发起第4步。

APP后台在收到客户端的请求之后，首先对QueryString中的Token进行校验。如果用户身份非法，则直接拒绝请求。如果用户身份合法，则根据URL中携带的fileId、keySource、edk等参数，到KMS系统中获取DK，并返回给客户端。

以上步骤均完成之后，客户端便拿到了视频解密密钥，从而可以进行正常的视频解密与播放。

# 视频播放方案2：通过Cookie传递身份认证信息

该方案仅适用于iOS/PC平台的H5/Flash播放器；在该平台下，播放器在访问EXT-X-KEY标签所标识的URL时会带上Cookie。

注意：实际测试发现，Android平台的H5播放器在访问EXT-X-KEY标签所标识的URL时不会携带Cookie，所以安卓平台目前只能使用方案1。

### 1. 登录并派发用于身份校验的Cookie

只有经过合法身份认证的客户才应当得到视频解密密钥。因此在视频播放之前，客户端必须进行登录操作，并由APP服务端给客户端派发签名。例如，客户端通过 `login.example.com` 进行账号密码登录，APP后台在通过身份认证之后，给客户端下发 `example.com` 域的cookie来标识用户身份。

### 2. 获取指定视频的多码率播放地址

腾讯云点播Web端视频播放器提供了多码率播放能力，即可以根据FileID获取一个视频对应的多码率播放地址。如果您使用了其他播放器，则必须自行获取多码率播放地址。

### 3. 获取视频内容(已加密)

当开始播放视频时，视频播放器会自动发起这一步。

视频播放器开始播放视频时，会向点播CDN边缘节点请求视频数据文件。对于HLS格式的视频，播放器会根据m3u8文件中的EXT-X-KEY标签来获取视频解密密钥。例如，假设EXT-X-KEY标签中获取视频解密密钥的URL为

```
https://getkey.example.com?fileId=123456&keySource=VodBuildInKMS&edk=abcdef
```

则当播放器获取解密密钥dk时，会带上第1步由APP后台派发的 `example.com` 域的cookie。

### 4/5. 获取视频解密密钥（携带身份验证Cookie）

当播放器获取到视频索引文件（m3u8文件）之后，会在播放视频文件之前自动发起第4步。

APP后台在收到客户端的请求之后，首先对cookie中的身份认证标识进行校验。如果用户身份非法，则直接拒绝请求。如果用户身份合法，则根据URL中携带的fileId、keySource、edk等参数，到KMS系统中获取DK，并返回给客户端。

以上步骤均完成之后，客户端便拿到了视频解密密钥，从而可以进行正常的视频解密与播放。

# FAQ

### 1. 加密HLS与普通HLS有什么差异？

根据HLS文档规范，HLS加密是对媒体文件(TS文件)进行加密，m3u8文件描述了播放器如何解密TS文件的方法。加密HLS的m3u8文件里包含了 #EXT-X-KEY 标签，该参数包含 METHOD 和 URI 属性。 METHOD 属性描述了加密的算法，如 AES-128 。 URI 属性描述了获取解密密钥的地址， 播放器访问这个URI就可以获取到解密的密钥数据。如URI为

```
http://www.test.com/getdk?fileId=123&edk=14cf
```

播放器解析该m3u8文件时就会向这个URI发起http请求，从返回包里获取到密钥数据。

### 2. 开通点播加密功能需要提供哪些信息？

开通点播加密功能需要提供getkeyurl，即 #EXT-X-KEY 标签中的 URI 属性。APP服务端需要部署客户端播放加密视频时获取密钥数据的http服务。点播服务在加密视频时，会把加密视频的m3u8文件 #EXT-X-KEY 标签的 URI 属性设置为getkeyurl。为了获取密钥和管理方便，我们会在getkeyurl后面附加三个参数fileId、edk、keySource。fileId即视频Id，edk是加密后的密钥， keySource是密钥来源，使用点播内置KMS系统的加密文件，keySource为VodBuildInKMS。播放器发起获取解密密钥请求时，APP服务端收到的http请求的QueryString就会包含fileId, edk等参数，APP服务端可以根据fileId和edk等参数来返回对应的dk给播放器。

### 3. 播放器播放加密视频时从哪里获取解密密钥？

播放器播放加密视频时根据m3u8文件里的#EXT-X-KEY URI发起获取密钥的请求，即APP提供给点播的getkeyurl地址。注意，播放器不是向点播服务器发起获取密钥的请求。APP服务器在调用点播ProcessFile API进行加密时，加密完成后再调用获取视频解密密钥 API获取到密钥后，需要将密钥保存起来，当播放器请求密钥时，根据播放器的请求参数来返回对应的密钥。

### 4. APP服务端如何返回密钥给播放器？

播放器向APP服务端发起获取密钥请求时，APP服务端需要将对应的密钥数据返回给播放器，返回的密钥数据为16字节的二进制数据。通过获取视频解密密钥 API获取到密钥为Base64编码的字符串，返回给播放器时需要将这个字符串转换为二进制数据。
例如，dkData为Base64编码的密钥数据

java

```
import java.util.Base64;
byte[] dkBin = Base64.getDecoder().decode(dkData ) ;
```

php

```
$dkBin = base64_decode($dkData);
```

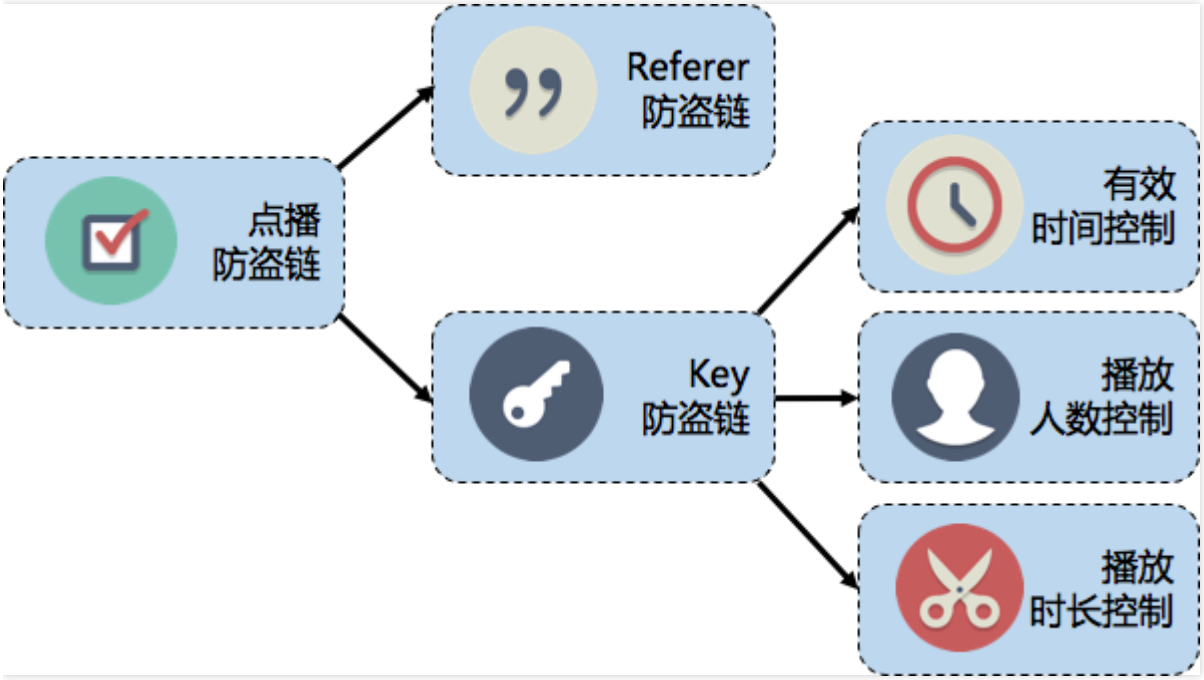
# 防盗链
# 防盗链综述

最近更新时间：2018-09-26 10:03:53

## 简介

为支持视频播放的权限控制，腾讯云点播推出了防盗链的解决方案。开通防盗链后，腾讯云 CDN 节点将对播放请求中的关键信息进行检查，并对通过检查的请求返回视频数据。

## 类型和能力




腾讯云点播防盗链支持 Referer 防盗链和 Key 防盗链两种类型。

### Referer 防盗链

基于 HTTP 协议支持的 Referer 机制，通过播放请求的 Header 中携带的 referer 字段识别请求的来源。开发者可以设置一批域名为黑名单或者白名单，CDN 节点将依照名单中的域名做鉴权，允许或拒绝播放请求。

有关 Referer 防盗链，更多详情请参见 Referer 防盗链介绍。

### Key 防盗链

允许开发者将视频的播放控制参数以 QueryString 的形式拼接在视频 URL 中。CDN 节点将检查 URL 中的播放控制参数，并依据参数控制视频的播放。目前，Key 防盗链通过“过期时间参数”、“允许播放的 IP 数量参数”和“试看时间参数”，支持“防盗链有效时间控制”、“防盗链播放人数控制”和“视频播放时长控制”。

**防盗链有效时间控制**

在视频 URL 中指定过期时间。如果请求的视频 URL 已经过期，则视频无法播放。通过这种方式，可以为视频 URL 设置有效时间，防范他人将视频 URL 转移到其他站点后长期使用。

**防盗链播放人数控制**

在视频 URL 中指定链接最多能供多少人播放。不在同一内网的播放终端，它们的公网 IP 一般是不同的。通过限制一个 URL 允许最多能被多少公网 IP 播放，就能够限制同一个 URL 可以播放的人数。这样，可以防范他人将视频 URL 转移到其他站点后，无限制地分发给任意多的人数观看。

**视频允许播放时长控制**

在视频 URL 中指定试看时长（例如仅允许播放视频的前5分钟）。通过这种方式，可以实现对未付费用户的试看功能。

有关 Key 防盗链，更多详情请参见 Key 防盗链介绍。

# Referer 防盗链

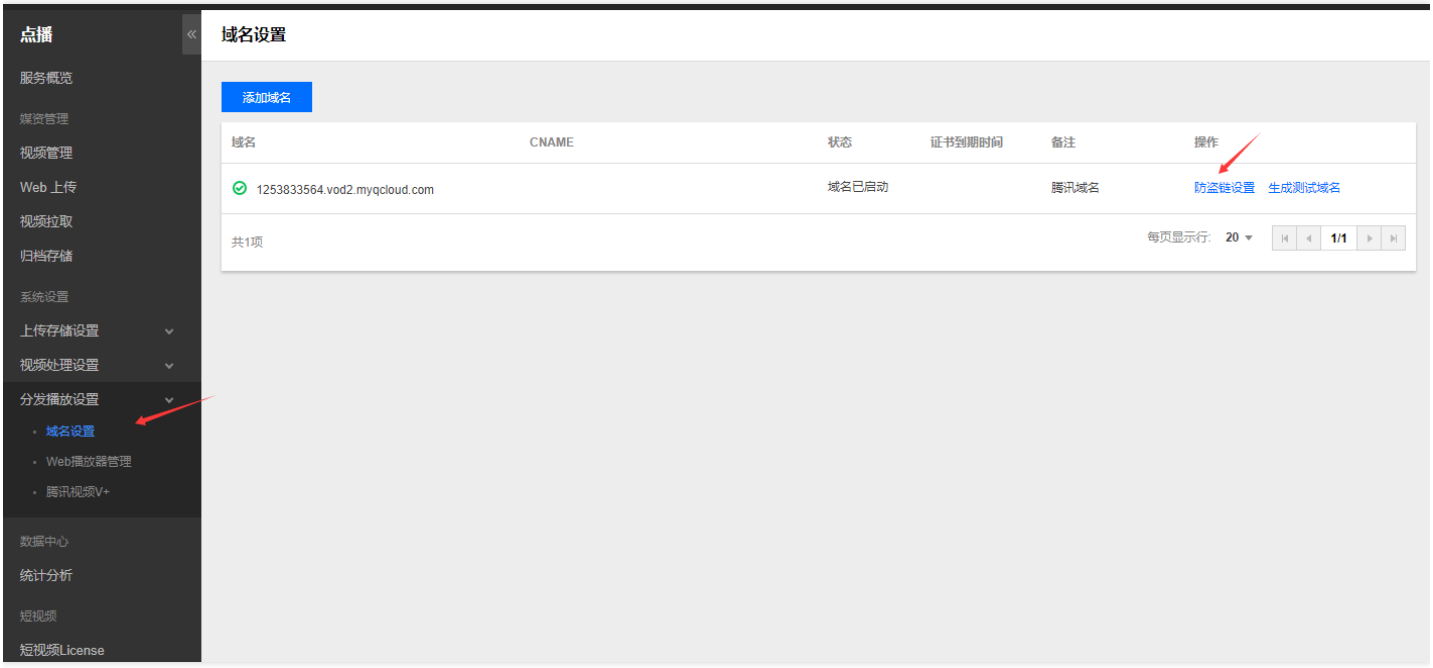
最近更新时间：2018-07-31 10:55:25

## 功能介绍

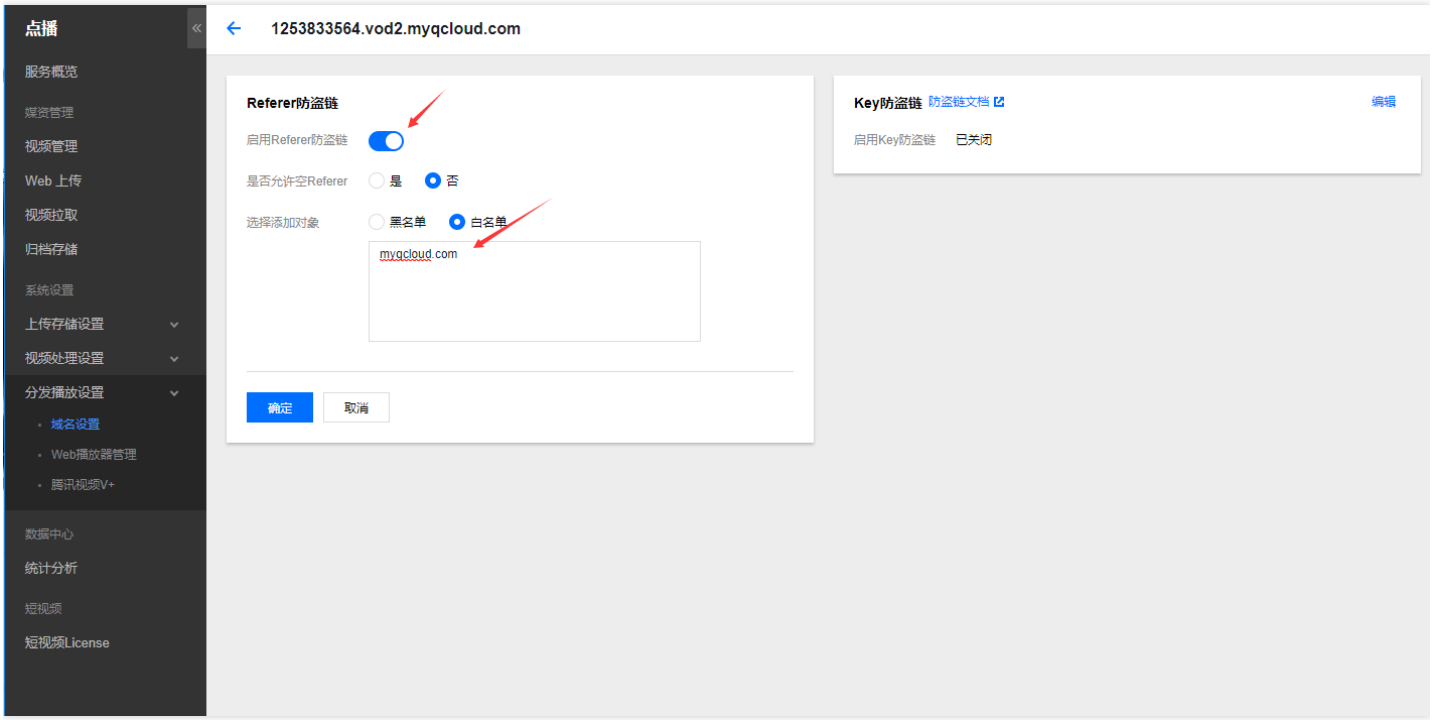
- 基于 HTTP 协议支持的 referer 机制，通过 HTTP 头部中携带的 referer 字段识别请求的来源。开发者可以通过配置 referer 黑白名单，对视频请求来源进行识别和鉴权 。
- 支持黑名单和白名单两种模式。当视频播放请求到达 CDN 节点后，节点将依据用户配置的 Referer 黑白名单对请求来源鉴权。对于符合规则的请求，CDN 将返回视频数据；否则，将返回403响应码，拒绝播放请求。

## 配置向导

腾讯云点播【控制台】> 【分发播放设置】>【域名设置】>【防盗链设置】。




启用 Referer 防盗链 -> 选择是否允许空 Referer -> 选择黑名单或白名单并添加对象 -> 确定。




保存配置后，大概需要5分钟使所有 CDN 节点生效该配置。

## 注意事项

- 该功能为可选项，默认不启用。
- 开启功能后，选择并填写黑名单或白名单，黑名单和白名单互斥，同一时间仅支持一种模式。
- 支持选择是否允许空 referer 请求视频（即是否允许在浏览器中直接输入视频 URL 播放视频）。
- 黑名单或白名单中的域名最多支持 10 条（最少1条），每一行一条记录。
- 域名前不要带协议名（ `http://` 和 `https://` ），域名为前缀匹配（填写 `abc.com` ，则 `abc.com/123` 和 `abc.com.cn` 也会匹配），且支持通配符（ `*.abc.com` ）。

# Key 防盗链

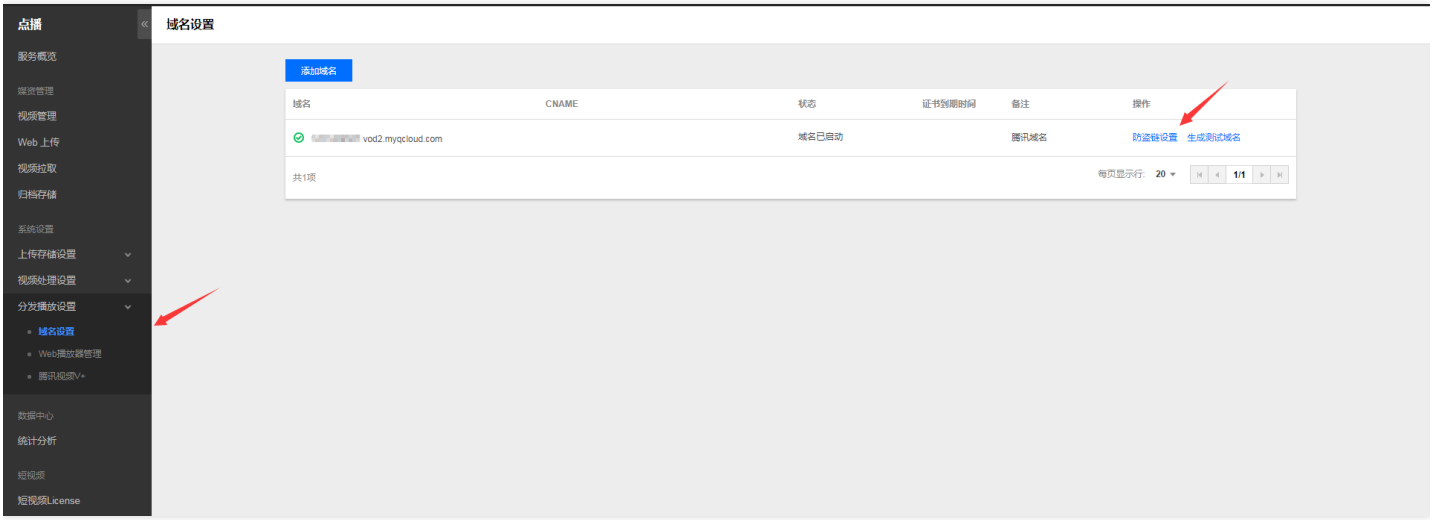
最近更新时间：2018-09-26 10:04:54

## 功能介绍

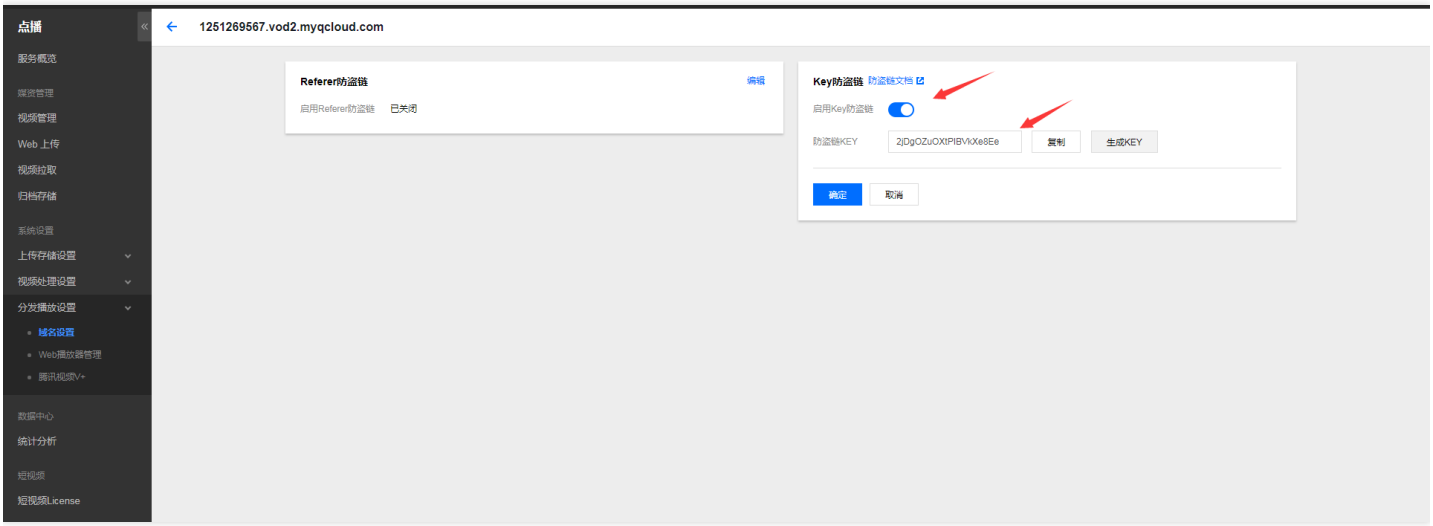
- 支持在视频 URL 中指定过期时间，他人获取后无法长期使用。
- 支持在视频 URL 中指定最大允许播放 IP 数，他人获取后不能无限制地分发给更多人观看。
- 支持在视频 URL 中指定试看时长，实现试看功能。
- 开发者使用密钥（KEY）对视频 URL 签名，并在 URL 中带上签名结果。只要用户密钥不泄露，其他用户无法伪造视频 URL。
- CDN 节点检查视频 URL 中的参数和签名，对视频播放请求进行控制。如果请求检查不通过，将返回403响应码。

## 配置向导

腾讯云点播【控制台】>【分发播放设置】>【域名设置】>【防盗链设置】。




启用 Key 防盗链 -> 生成 KEY -> 确定。




保存配置后，大概需要 5 分钟使所有 CDN 节点生效该配置。

# 防盗链 URL 生成方式

- 开发者在腾讯云点播中的视频均存在**视频原始 URL**。未开启防盗链时，使用视频原始 URL 即可播放视频。
- 开启 Key 防盗链后，视频原始 URL 不再能播放，此时需要构造视频的**防盗链 URL**。

视频的防盗链 URL 的生成规则是，在原始 URL 后以 QueryString 的方式加入防盗链参数，形如：

```
http://example.vod2.myqcloud.com/dir1/dir2/myVideo.mp4?t=[t]&exper=[exper]&rlimit=[rlimit]&us=[us]&s
```

QueryString 中的防盗链参数必须按照 `t` ， `exper` ， `rlimit` ， `us` ， `sign` 的顺序拼接，下面详细介绍防盗链 URL 中各个参数的含义和取值方法。

### 防盗链参数

| 参数名 | 必选 | 说明 |
| --- | --- | --- |
| KEY | 是 | 开启 Key 防盗链时填写的密钥。必须由大小写字母（a-Z）或者数字（0-9）组成，长度在8-20个字符之间。建议在控制台中单击“生成 KEY”按钮生成。 |
| Dir | 是 | 视频原始 URL 的 PATH 中除去文件名的那部分路径。假设原始 URL 为 `http://example.vod2.myqcloud.com/dir1/dir2/myVideo.mp4` ，则播放路径为 `/dir1/dir2/` 。 |
| t | 是 | 播放地址的过期时间戳，以 Unix 时间的16进制小写形式表示。过期后该 URL 将不再有效，返回 403 响应码。过期时间戳不要过短，使视频有足够时间完整播放。 |

| 参数名 | 必选 | 说明 |
| --- | --- | --- |
| exper | 否 | 试看时长，单位为秒，以十进制表示。不填或者填0表示不试看（即返回完整视频）。试看时长不要超过视频原始时长，否则可能导致播放失败。 |
| rlimit | 否 | 最多允许多少个不同 IP 的终端播放，以十进制表示，不填表示不做限制。当限制 URL 只能被1个人播放时，建议 rlimit 不要严格限制成1（比如可以设置成3），因为移动端断网后重连 IP 可能改变。 |
| us | 否 | 链接标识，用于随机化一个防盗链 URL，增强链接的唯一性。建议每次生成防盗链 URL 时，指定一个随机的 us 值。 |
| sign | 是 | 防盗链签名，以32个字符长的16进制数表示，用于校验防盗链 URL 的合法性。签名校验失败将返回 403 响应码。下面将介绍签名计算公式。 |

### 签名计算公式

```
sign = md5(KEY + Dir + t + exper + rlimit + us)
```

公式中的 `+` 代表字符串拼接，选填参数可以为空字符串。

## 防盗链 URL 生成示例

- 假设某个开发者在腾讯云点播中有一个视频，视频的原始播放 URL 是 `http://example.vod2.myqcloud.com/dir1/dir2/myVideo.mp4` 。
- 该开发者开通了 Key 防盗链，生成的密钥是 `24FEQmTzro4V5u3D5epW` 。
- 希望为这个视频生成防盗链 URL，URL 的过期时间是2018年1月31日20:00整（Unix时间为1517400000）。
- 如果要生成一个试看 URL，希望试看时长为视频的前5分钟（视频原始时长大于5分钟）。
- 如果要限制 URL 可播放的 IP 数，希望允许最多3个不同 IP 的终端可以播放该 URL。
- 生成防盗链 URL 时生成了一个随机字符串 `72d4cd1101` 。

以此假设为例，下面分别就“视频播放地址有效时间控制”，“视频播放地址允许最多播放 IP 数”和“视频允许播放时长控制”两个场景，介绍如何生成防盗链 URL。

### 示例1：视频播放地址有效时间控制

#### 第一步：确定防盗链参数

首先，确定除了签名 `sign` 以外的防盗链参数。

| 参数名 | 取值 | 说明 |
| --- | --- | --- |

| 参数名 | 取值 | 说明 |
| --- | --- | --- |
| KEY | 24FEQmTzro4V5u3D5epW | 开发者开通 Key 防盗链时选择的密钥 |
| Dir | /dir1/dir2/ | 原始播放 URL 的 PATH 中除去 myVideo.mp4 的剩余部分 |
| t | 5a71afc0 | 过期时间戳1517400000的十六进制表示结果 |
| us | 72d4cd1101 | 生成的随机字符串 |

**第二步：计算签名**

根据签名计算公式，获得签名结果。

```
sign = md5("24FEQmTzro4V5u3D5epW/dir1/dir2/5a71afc072d4cd1101") = "3d8488faeb37d52d6bf63b63c1b171c3"
```

**第三步：生成防盗链 URL**

将防盗链参数拼接到视频原始 URL 的 QueryString 中，就得到了视频防盗链 URL：

```
http://example.vod2.myqcloud.com/dir1/dir2/myVideo.mp4?t=5a71afc0&us=72d4cd1101&sign=3d8488faeb37d52
```

### 示例2：视频播放地址最多可播放 IP 数

**第一步：确定防盗链参数**

首先，确定除了签名 sign 以外的防盗链参数。

| 参数名 | 取值 | 说明 |
| --- | --- | --- |
| KEY | 24FEQmTzro4V5u3D5epW | 开发者开通 Key 防盗链时选择的密钥 |
| Dir | /dir1/dir2/ | 原始播放 URL 的 PATH 中除去 myVideo.mp4 的剩余部分 |
| t | 5a71afc0 | 过期时间戳1517400000的十六进制表示结果 |
| rlimit | 3 | 限制最多允许3个不同的 IP 播放 URL |
| us | 72d4cd1101 | 生成的随机字符串 |

**第二步：计算签名**

根据签名计算公式，获得签名结果。

```
sign = md5("24FEQmTzro4V5u3D5epW/dir1/dir2/5a71afc0372d4cd1101") = "c5214f0d5961b13acd558b4957c4dfc5
```

**第三步：生成防盗链 URL**

将防盗链参数拼接到视频原始 URL 的 QueryString 中，就得到了视频防盗链 URL：

```
http://example.vod2.myqcloud.com/dir1/dir2/myVideo.mp4?t=5a71afc0&rlimit=3&us=72d4cd1101&sign=c5214f
```

### 示例3：视频允许播放时长控制

**第一步：确定防盗链参数**

首先，确定除了签名 `sign` 以外的防盗链参数。

| 参数名 | 取值 | 说明 |
| --- | --- | --- |
| KEY | `24FEQmTzro4V5u3D5epW` | 开发者开通 Key 防盗链时选择的密钥 |
| Dir | `/dir1/dir2/` | 原始播放 URL 的 PATH 中除去 `myVideo.mp4` 的剩余部分 |
| t | `5a71afc0` | 过期时间戳1517400000的十六进制表示结果 |
| exper | `300` | 试看前5分钟即300秒 |
| us | `72d4cd1101` | 生成的随机字符串 |

**第二步：计算签名**

根据签名计算公式，获得签名结果。

```
sign = md5(”24FEQmTzro4V5u3D5epW/dir1/dir2/5a71afc030072d4cd1101”) = ”547d98c4b91e81b5ea55c95cef6322
```

**第三步：生成防盗链 URL**

将防盗链参数拼接到视频原始 URL 的 QueryString 中，就得到了视频防盗链 URL：

```
http://example.vod2.myqcloud.com/dir1/dir2/myVideo.mp4?t=5a71afc0&exper=300&us=72d4cd1101&sign=547d9
```

## Key 防盗链生成和校验工具

点播为开发者提供了 Key 防盗链 URL 的生成工具和校验工具，开发者可以使用该工具快捷准确地生成和校验符合要求的防盗链 URL。

- Key 防盗链视频播放地址生成工具
- Key 防盗链视频播放地址校验工具

# 注意事项

- 该功能为可选项，默认不启用。
- 启用该功能后，视频原始 URL 将不再能直接播放，需要按规则生成合法的防盗链 URL。
- 密钥 KEY 必须由大小写字母（a-Z）或者数字（0-9）组成，长度在8-20个字符之间。
- 若防盗链 URL 过期，或者签名不能通过，将无法播放视频，并返回403响应码。
- 防盗链 URL 中 QueryString 中各参数必须按照 `t` ， `exper` ， `rlimit` ， `us` ， `sign` 的顺序出现，如果顺序不正确将无法播放视频。
- 考虑到机器之间可能存在时间差，防盗链 URL 的实际过期时间一般比指定的过期时间长5分钟，即额外给出300秒的容差时间。
- 如果使用试看功能，需确保试看时长不大于视频时长，否则将导致视频无法播放。
- 试看对视频的格式有较严格的要求（仅支持H264，视频元信息在视频文件的头部等），不符合格式要求的原始视频使用试看功能将产生异常。建议使用点播转码功能进行转码，对转码后视频设置试看（转码后的格式均符合试看格式要求）。