

云点播  
最佳实践  
产品文档



腾讯云

**【版权声明】**

©2013-2024 腾讯云版权所有

本文档著作权归腾讯云单独所有，未经腾讯云事先书面许可，任何主体不得以任何形式复制、修改、抄袭、传播全部或部分本文档内容。

**【商标声明】**

及其它腾讯云服务相关的商标均为腾讯云计算（北京）有限责任公司及其关联公司所有。本文档涉及的第三方主体的商标，依法由权利人所有。

**【服务声明】**

本文档意在向客户介绍腾讯云全部或部分产品、服务的当时的整体概况，部分产品、服务的内容可能有所调整。您所购买的腾讯云产品、服务的种类、服务标准等应由您与腾讯云之间的商业合同约定，除非双方另有约定，否则，腾讯云对本文档内容不做任何明示或默示的承诺或保证。

## 文档目录

### 最佳实践

#### 媒体上传

如何将网络媒体拉取到云点播

如何在服务端上传视频

如何通过 Web 上传视频

如何提升媒体上传速度和成功率

如何将点播的媒体文件进行智能降冷

#### 媒体处理

如何对视频进行转码

如何对媒体进行内容审核

如何对视频添加复杂水印

#### 分发播放

如何使用 Key 防盗链

如何防止视频流量和存储空间被盗用

如何接收事件通知

如何进行源站迁移

#### 直播录制

如何将直播视频录制到云点播并进行处理

如何在直播过程中实现回看功能

如何将点播视频转为类直播效果

如何使用云点播和 WordPress 搭建网站

如何进行自定义源站回源

# 最佳实践

## 媒体上传

### 如何将网络媒体拉取到云点播

最近更新时间：2022-12-16 15:02:12

## 使用须知

### 内容介绍

本文档向开发者介绍如何拉取网络视频（以 URL 的形式提供）到云点播（VOD）。

### 费用

本文提供的代码是免费开源的，但在使用的过程中可能会产生以下费用：

- 购买腾讯云云服务器（CVM）用于执行 API 请求脚本，详见 [CVM 计费](#)。
- 消耗 VOD 存储用于存储拉取上传的视频，详见 [存储计费](#)。

### 限制

云点播提供的 URL 拉取功能具有如下限制：

- URL 需要直接指向视频文件，不可以是视频网站页面链接。
- 如果 URL 带有时间戳防盗链，请确保防盗链的限制（有效期、访问次数等）足够宽松，否则可能失败。
- 不支持启用了 Referer 防盗链的 URL。
- 不支持 DASH（MPD 文件类型）。
- 如果拉取的对象是 HLS（M3U8 文件类型），那么 Media Segment（一般是 TS 文件类型）的 URI 要求是相对路径，且不能带参数。

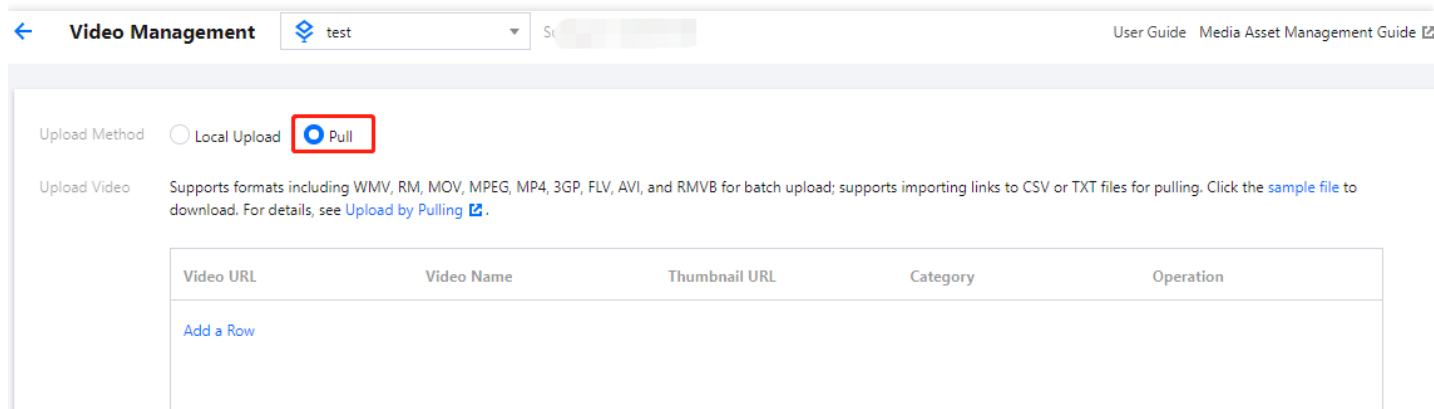
## 在控制台拉取上传

### 步骤1：开通云点播

请参考 [快速入门 - 步骤1](#) 开通云点播服务。

### 步骤2：创建拉取任务

访问云点播控制台的 [上传页面](#)，上传方式选择【视频拉取】，然后单击【添加一行】，填写待拉取视频的 URL（本文以 [测试视频 URL](#) 为例，其余项为选填，开发者可以根据需要进行填写），最后单击左下角的【拉取视频】：



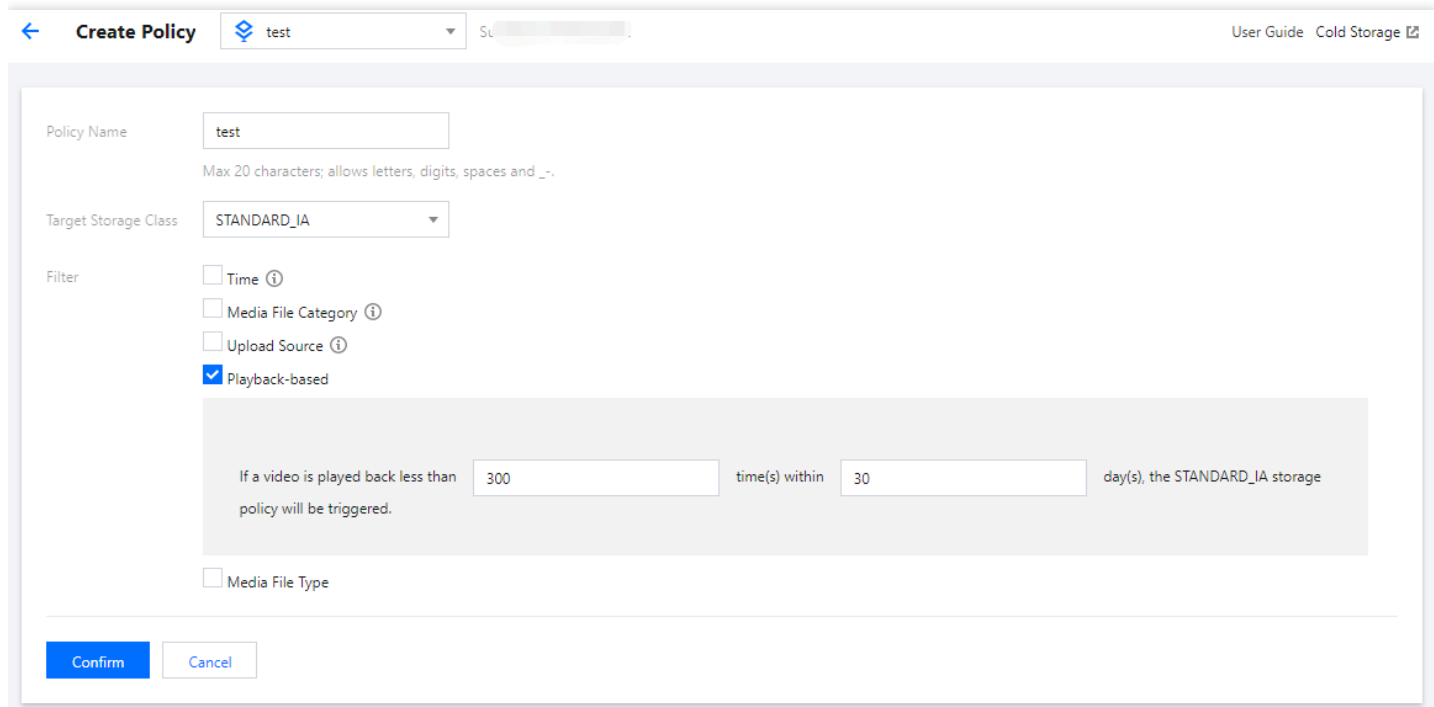
The screenshot shows the 'Video Management' page. Under 'Upload Method', the 'Pull' radio button is selected and highlighted with a red box. Below it, there is a table with columns: Video URL, Video Name, Thumbnail URL, Category, and Operation. A link 'Add a Row' is visible in the first row of the table.

说明：

拉取视频所花费的时间和视频文件的大小成正比。建议开发者选择较小的视频（如几十MB以内）进行测试，避免长时间等待。

### 步骤3：查看拉取结果

等待一两分钟后（根据视频文件的大小有所差别），在 [媒资管理页面](#) 可以看到已经拉取完成的视频：



The screenshot shows the 'Create Policy' page. The 'Policy Name' is 'test'. The 'Target Storage Class' is 'STANDARD\_IA'. Under the 'Filter' section, 'Playback-based' is checked. A configuration box shows: 'If a video is played back less than 300 time(s) within 30 day(s), the STANDARD\_IA storage policy will be triggered.' There are 'Confirm' and 'Cancel' buttons at the bottom.

说明：

如果拉取过程中浏览器一直停留在媒资管理页面，那么需要刷新页面才能看到拉取完成的视频。

## 调用云 API 拉取上传

### 步骤1：准备腾讯云 CVM

云 API 请求脚本需要运行在一台腾讯云 CVM 上，要求如下：

- 地域：任意。
- 机型：官网最低配置（1核1GB）即可。
- 公网：需要拥有公网 IP，带宽1Mbps或以上。
- 操作系统：官网公共镜像 `Ubuntu Server 16.04.1 LTS 64位` 或 `Ubuntu Server 18.04.1 LTS 64位`。

购买 CVM 的方法请参见 [操作指南 - 创建实例](#)。重装系统的方法请参见 [操作指南 - 重装系统](#)。

注意：

如果您没有符合上述条件的腾讯云 CVM，也可以在其它带外网的 Linux（如 CentOS、Debian 等）或 Mac 机器上执行脚本，但需根据操作系统的区别修改脚本中的个别命令，具体修改方式请开发者自行搜索。

### 步骤2：获取 API 密钥

请求云 API 需要使用到开发者的 API 密钥（即 SecretId 和 SecretKey）。如果还未创建过密钥，请参见 [创建密钥文档](#) 生成新的 API 密钥；如果已创建过密钥，请参见 [查看密钥文档](#) 获取 API 密钥。

### 步骤3：开通云点播

请参考 [快速入门 - 步骤1](#) 开通云点播服务。

### 步骤4：发起拉取任务

登录 [步骤1](#) 中准备好的 CVM（登录方法详见 [操作指南 - 登录 Linux](#)），在远程终端输入以下命令并运行：

```
ubuntu@VM-69-2-ubuntu:~$ export SECRET_ID=AKxxxxxxxxxxxxxxxxxxxxxxxxxx; export SECRET_KEY=xxxxxxxxxxxxxxxxxxxxxxxxxx; git clone https://github.com/tencentyun/vod-server-demo.git ~/vod-server-demo; bash ~/vod-server-demo/installer/pull_upload_api_en.sh
```

说明：

请将命令中的 SECRET\_ID 和 SECRET\_KEY 赋值为 [步骤2](#) 中获取到的内容。

该命令将从 Github 下载 Demo 源码并自动执行安装脚本。安装过程需几分钟（具体取决于 CVM 网络状况），期间远程终端会打印如下示例的信息：

```
[2020-07-15 17:40:13]开始安装 pip3。
[2020-07-15 17:40:39]pip3 安装成功。
[2020-07-15 17:40:39]开始安装云 API Python SDK 。
[2020-07-15 17:40:42]云 API Python SDK 安装完成。
[2020-07-15 17:40:42]开始配置 API 参数。
[2020-07-15 17:40:42]API 参数配置完成。
```

执行 pull\_upload.py 脚本发起转码：

```
ubuntu@VM-69-2-ubuntu:~$ cd ~/vod-server-demo/pull_upload_api/; python3 pull_upload.py http://1400329073.vod2.myqcloud.com/ff439affvodcq1400329073/e968a7e55285890804162014755/LKk92603oW0A.mp4 API-PullUpload
```

说明：

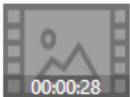
请将命令中的 URL 替换为实际需要拉取的视频地址。

该命令将对指定的 URL 发起 PullUpload 请求，并打印类似如下的应答内容：

```
{"TaskId": "1400329073-PullUpload-4ea60158fc6f8e611bbfa750eb1fd0a9t0", "RequestId": "4e821b4a-9a29-409f-99cb-b703fa184e50"}
```

## 步骤5：查看拉取结果

等待一两分钟后（具体根据视频文件的大小有所差别），在 [媒资管理页面](#) 可以看到已经拉取完成的视频：

<input type="checkbox"/>	Video Info	Video Status	Video Category <span>▾</span>	Video Source <span>▾</span>
<input type="checkbox"/>	 API-PullUpload ID: 5285890806967135776	<span style="color: green;">✔</span> Normal	Others	Upload

说明：

如果拉取过程中浏览器一直停留在媒资管理页面，那么需要刷新页面才能看到拉取完成的视频。



# 如何在服务端上传视频

最近更新时间：2021-05-11 18:37:53

## 使用须知

### 内容介绍

本文档向开发者介绍如何将服务器本地的视频文件上传到云点播（VOD）。

### 费用

本文提供的代码是免费开源的，但在使用的过程中可能会产生以下费用：

- 购买腾讯云云服务器（CVM）用于执行上传脚本，详见 [CVM 计费](#)。
- 消耗 VOD 存储用于存储上传的视频，详见 [存储计费](#)。

## 将 CVM 上的视频上传到 VOD

### 步骤1：准备腾讯云 CVM

上传脚本需要运行在一台腾讯云 CVM 上，要求如下：

- 地域：任意。
- 机型：官网最低配置（1核1GB）即可。
- 公网：需要拥有公网 IP，带宽1Mbps或以上。
- 操作系统：官网公共镜像 `Ubuntu Server 16.04.1 LTS 64位` 或 `Ubuntu Server 18.04.1 LTS 64位`。

购买 CVM 的方法请参见 [操作指南 - 创建实例](#)。重装系统的方法请参见 [操作指南 - 重装系统](#)。

#### ⚠ 注意：

如果您没有符合上述条件的腾讯云 CVM，也可以在其它带外网的 Linux（如 CentOS、Debian 等）或 Mac 机器上执行脚本，但需根据操作系统的区别修改脚本中的个别命令，具体修改方式请开发者自行搜索。

### 步骤2：开通云点播

请参考 [快速入门 - 步骤1](#) 开通云点播服务。

### 步骤3：获取 API 密钥

上传视频需要使用到开发者的 API 密钥（即 SecretId 和 SecretKey）。如果还未创建过密钥，请参见 [创建密钥文档](#) 生成新的 API 密钥；如果已创建过密钥，请参见 [查看密钥文档](#) 获取 API 密钥。

#### 步骤4：下载代码并安装 SDK

登录 [步骤1](#) 中准备好的 CVM（登录方法详见 [操作指南 - 登录 Linux](#)），在远程终端输入以下命令并运行：

```
ubuntu@VM-69-2-ubuntu:~$ export SECRET_ID=AKxxxxxxxxxxxxxxxxxxxxxxxxxx; export SECRET_KEY=xxxxxxxxxxxxxxxxxxxxxxxxxx; git clone https://github.com/tencentyun/vod-server-demo.git ~/vod-server-demo; bash ~/vod-server-demo/installer/server_upload.sh
```

#### 说明：

请将命令中的 SECRET\_ID 和 SECRET\_KEY 赋值为 [步骤3](#) 中获取到的内容。

该命令将从 Github 下载 Demo 源码并自动执行安装脚本。安装过程需几分钟（具体取决于 CVM 网络状况），期间远程终端会打印类似如下的信息：

```
[2020-06-23 19:56:31]开始安装 pip3。
[2020-06-23 19:56:34]pip3 安装成功。
[2020-06-23 19:56:34]开始安装 VOD Python 上传 SDK。
[2020-06-23 19:56:36]VOD Python 上传 SDK 安装完成。
[2020-06-23 19:56:36]开始配置 SDK 参数。
[2020-06-23 19:56:36]SDK 参数配置完成。
```

#### 步骤5：上传视频

发起上传之前，开发者需要在 CVM 上准备好视频文件和封面图片（可选）。如果开发者不方便上传视频到 CVM，可以在远程终端执行以下命令，将测试视频和测试封面下载到 CVM 上：

```
ubuntu@VM-69-2-ubuntu:~$ wget http://1400329073.vod2.myqcloud.com/d62d88a7vodtranscq1400329073/7a9b2b565285890804459281865/v.f100010.mp4 -O ~/vod-server-demo/server_upload/tencent_cloud.mp4; wget http://1400329073.vod2.myqcloud.com/ff439affvodcq1400329073/8aa658d15285890804459940822/5285890804459940825.jpg -O ~/vod-server-demo/server_upload/tencent_cloud.jpg
```

执行 `server_upload.py` 脚本发起上传：

```
ubuntu@VM-69-2-ubuntu:~$ cd ~/vod-server-demo/server_upload/; python3 server_upload.py ./tencent_cloud.mp4 ./tencent_cloud.jpg
```

### ① 说明：

请将命令中的视频路径和封面图片路径替换为实际文件的路径，其中封面图片路径参数是可选的，如果不填则上传后的视频没有封面。

该命令将 `tencent_cloud.mp4` 视频上传到 VOD，同时上传 `tencent_cloud.jpg` 图片作为其封面。上传完成后远程终端会打印出类似如下信息：

```
{"CoverUrl": "http://1400329073.vod2.myqcloud.com/ff439affvodcq1400329073/8aa658d15285890804459940822/5285890804459940825.jpg", "FileId": "5285890804459940822", "MediaUrl": "http://1400329073.vod2.myqcloud.com/ff439affvodcq1400329073/8aa658d15285890804459940822/f0.mp4", "RequestId": "84a7fb42-9f05-4acd-9cc8-843690b188ce"}
```

### ① 说明：

如果开发者使用自己的视频进行测试，建议使用较小的视频文件（例如几个MB），避免因 CVM 带宽不足而需要耗费很长的上传时间。

## 步骤6：查看结果

在控制台 [视频管理](#) 页面上可以看到刚上传的视频文件和封面：

## 代码解读

- `main()` 为脚本入口。
- 调用 `parse_conf_file()`，从 `config.json` 文件中读取配置信息。配置项说明如下：

字段	数据类型	功能
<code>secret_id</code>	String	API 密钥
<code>secret_key</code>	String	API 密钥
<code>procedure</code>	String	任务流名称，即视频上传完成后自动触发该任务流。默认为空
<code>subappid</code>	String	是否将视频上传到 <a href="#">云点播子应用</a>

### ① 说明：

本 Demo 仅支持 `procedure` 和 `subappid` 两个上传参数，完整功能请参考 [Python 上传 SDK 接口描述](#)。

- 从命令行参数中获取待上传视频文件的本地路径，以及封面图片路径（如果有封面），然后调用 `upload_media()` 发起上传：

```
if len(sys.argv) < 2:
    usage()
    return
video_path = sys.argv[1]
cover_path = sys.argv[2] if len(sys.argv) > 2 else ""

# 发起上传
rsp = upload_media(configuration, video_path, cover_path)
```

- 在 `upload_media()` 中，使用 Python SDK 提供的方法构造一个上传实例 `client`，然后在 `req` 中设置上传参数，最后发起上传：

```
client = VodUploadClient(conf["secret_id"], conf["secret_key"])
req = VodUploadRequest()

req.MediaFilePath = video
if cover != "":
    req.CoverFilePath = cover
if conf["procedure"] != "":
    req.Procedure = conf["procedure"]
req.SubAppId = int(conf["subappid"])

rsp = client.upload("ap-guangzhou", req)
return rsp
```

#### ⚠ 注意：

`client.upload()` 中的第一个参数（`"ap-guangzhou"`）是指上传实例的接入地域，不是指视频上传后的存储地域。该参数固定填为 `"ap-guangzhou"` 即可，如果需要指定视频上传后的存储地域，请设置 `req.StorageRegion` 参数。

## 其它功能

---

VOD 服务端上传 SDK 还支持其它特性，如设置视频名称、分类、过期时间等，详情请参考各语言 SDK 开发指南：

- [Java](#)
- [C#](#)
- [PHP](#)
- [Python](#)
- [Node.js](#)
- [Golang](#)

# 如何通过 Web 上传视频

最近更新时间：2022-12-16 15:12:12

## 使用须知

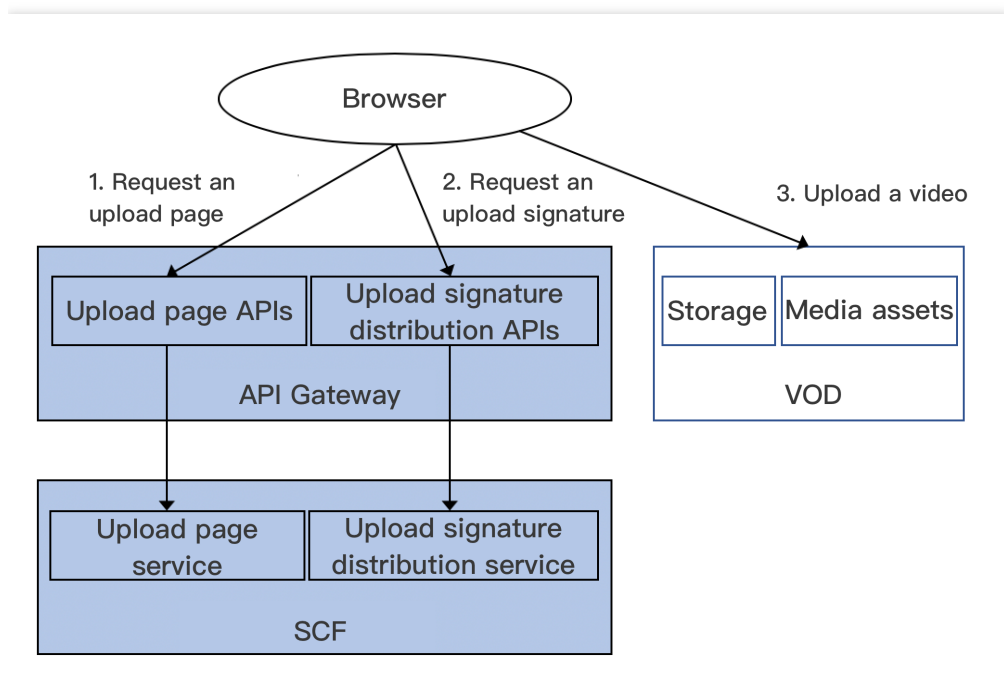
### Demo 功能介绍

本 Demo 向开发者展示如何通过 Web 页面将视频上传到云点播。Demo 基于云函数（SCF）搭建了两个 HTTP 服务：

- 第一个服务用于接收来自浏览器获取 [客户端上传签名](#) 的请求，计算上传签名并返回。
- 第二个服务使用 VOD [Web 上传 SDK](#) 实现一个页面，用户可以通过浏览器访问该页面，并上传本地视频到 VOD。

### 架构和流程

系统主要涉及四个组成部分：浏览器、API 网关、云函数和云点播，其中 API 网关和云函数即是本 Demo 的部署对象，如下图所示：



主要业务流程为：

1. 浏览器向 SCF 请求上传页面。
2. 用户在上传页面进行操作，选中本地视频后点击上传，由浏览器向 SCF 请求上传签名。
3. 浏览器使用上传签名向 VOD 发起上传请求，完成后在上传页面上展示上传结果。

说明：

Demo 中的 SCF 代码使用 Python3.6 进行开发，此外 SCF 还支持 Python2.7、Node.js、Golang、PHP 和 Java 等多种编程语言，开发者可以根据情况自由选择，具体请参考 [SCF 开发指南](#)。

## 费用

本文提供的云点播 Web 上传 Demo 是免费开源的（含 Web 页面代码和业务后台代码），但在搭建和使用的过程中可能会产生以下费用：

- 购买腾讯云云服务器（CVM）用于执行服务部署脚本，详见 [CVM 计费](#)。
- 使用腾讯云云函数（SCF）提供上传页面和签名派发服务，详见 [SCF 计费](#) 和 [SCF 免费额度](#)。
- 使用腾讯云 API 网关为 SCF 提供外网接口，详见 [API 网关计费](#)。
- 消耗云点播（VOD）存储用于存储上传的视频，详见 [存储计费](#)。
- 消耗云点播流量用于播放视频，详见 [流量计费](#)。

## 快速部署 Web 上传 Demo

Web 上传 Demo 部署在 SCF 上，并由 API 网关提供服务入口。为了方便开发者搭建服务，我们提供了快捷部署脚本，使用方法如下。

### 步骤1：准备腾讯云 CVM

部署脚本需要运行在一台腾讯云 CVM 上，要求如下：

- 地域：任意。
- 机型：官网最低配置（1核1GB）即可。
- 公网：需要拥有公网 IP，带宽1Mbps或以上。
- 操作系统：官网公共镜像 `Ubuntu Server 16.04.1 LTS 64位` 或 `Ubuntu Server 18.04.1 LTS 64位`。

购买 CVM 的方法请参见 [操作指南 - 创建实例](#)。重装系统的方法请参见 [操作指南 - 重装系统](#)。

注意：

- **Web 上传 Demo 本身并不依赖于 CVM，仅使用 CVM 来执行部署脚本。**
- 如果您没有符合上述条件的腾讯云 CVM，也可以在其它带外网的 Linux（如 CentOS、Debian 等）或 Mac 机器上执行部署脚本，但需根据操作系统的区别修改脚本中的个别命令，具体修改方式请开发者自行搜索。

## 步骤2：开通云点播

请参考 [快速入门 - 步骤1](#) 开通云点播服务。

## 步骤3：获取 API 密钥和 APPID

Web 上传 Demo 服务的部署和运行过程需要使用到开发者的 API 密钥（即 SecretId 和 SecretKey）和 APPID。

- 如果还未创建过密钥，请参见 [创建密钥文档](#) 生成新的 API 密钥；如果已创建过密钥，请参见 [查看密钥文档](#) 获取 API 密钥。
- 在控制台 [账号信息](#) 页面可以查看 APPID，如下图所示：

## 步骤4：部署业务后台和 Web 页面

登录 [步骤1准备的 CVM](#)（登录方法详见 [操作指南 - 登录 Linux](#)），在远程终端输入以下命令并运行：

```
ubuntu@VM-69-2-ubuntu:~$ export SECRET_ID=AKxxxxxxxxxxxxxxxxxxxxxxxx; export SECRET_KEY=xxxxxxxxxxxxxxxxxxxxxxxx;export APPID=125xxxxxxxx;git clone https://github.com/tencentyun/vod-server-demo.git ~/vod-server-demo; bash ~/vod-server-demo/installer/web_upload_scf_en.sh
```

说明：

请将命令中的 SECRET\_ID、SECRET\_KEY 和 APPID 赋值为 [步骤3](#) 中获取到的内容。

该命令将从 Github 下载 Demo 源码并自动执行安装脚本。安装过程需几分钟（具体取决于 CVM 网络状况），期间远程终端会打印如下示例的信息：

```
[2020-04-25 23:03:20]开始安装 pip3。
[2020-04-25 23:03:23]pip3 安装成功。
[2020-04-25 23:03:23]开始安装腾讯云 SCF 工具。
[2020-04-25 23:03:26]scf 安装成功。
[2020-04-25 23:03:26]开始配置 scf。
[2020-04-25 23:03:28]scf 配置完成。
[2020-04-25 23:03:28]开始部署云点播客户端上传签名派发服务。
[2020-04-25 23:03:40]云点播客户端上传签名派发服务部署完成。
[2020-04-25 23:03:44]开始部署云点播 Web 上传页面。
[2020-04-25 23:03:53]云点播 Web 上传页面部署完成。
[2020-04-25 23:03:53]请在浏览器访问以下地址进行体验：https://service-xxxxxxxx-125xxxxxx.gz.apigw.tencentcs.com/release/web_upload_html
```

复制输出日志中的 Web 页面地址（示例中的 `https://service-xxxxxxxx-125xxxxxx.gz.apigw.tencentcs.com/release/web_upload_html`）。



注意：

如果输出日志中出现如下所示的警告，一般是由于 CVM 无法立即解析刚部署好的服务域名，可尝试忽略该警告。

```
> [2020-04-25 17:18:44] 警告：客户端上传签名派发服务测试不通过。  
>
```

## 步骤5：体验 Web 上传 Demo

1. 在浏览器打开 [步骤4](#) 中复制的地址，即可开始体验 Web 上传 Demo。页面如下图所示：
2. 在该页面进行视频上传操作：
  - i. 选择一个本地视频文件（推荐选择 MP4 文件）。
  - ii. 选择一张本地封面图片（可选，使用 JPG 或者 PNG 格式）。
  - iii. 填写视频名称（可选）。
  - iv. 单击【开始上传】即可上传视频。
3. 上传完成后，页面下方会展示视频和封面的点播媒体 ID（即 fileId）和 URL 等信息。如下图所示：  
您可以在 [云点播控制台](#) 上查看刚上传的视频。如下图所示：

说明：

您可根据页面提示，对上传页面的其它功能进行体验。

## 系统设计说明

### 接口协议及测试

上传页面和上传签名派发两个云函数都通过 API 网关对外提供接口，具体接口协议如下：

服务	云函数名	接口形式	返回内容
上传页面	web_upload_html	HTTP GET	HTML 页面
上传签名派发	ugc_upload_sign	HTTP POST	上传签名

### 上传页面

您可以访问 [SCF 服务列表](#) 来查看上传页面服务的详细信息：

说明：

- Demo 使用的两个 SCF 部署在广州地域，命名空间为 vod\_demo。
- 控制台上需要选择对应地域和命名空间才能看到部署好的云函数。

单击函数名，在左侧选择【触发管理】，右侧【访问路径】即是上传页面的 URL。单击【API服务名】即可跳转到对应的 API 网关页面。如下图所示：

测试该服务的方法为：在浏览器上直接访问页面 URL，正常情况下能看到上传页面。

## 上传签名派发

您可以访问 [SCF 服务列表](#) 来查看上传签名派发服务的详细信息（查看方法同 [上传页面](#)）。

单击函数名，在左侧选择【触发管理】，右侧【访问路径】即是该服务的 URL。单击【API服务名】即可跳转到对应的 API 网关页面。如下图所示：

测试该服务的方法为：选择手动发送 HTTP 请求的方式，在一台有外网的 Linux 或者 Mac 上执行以下命令（请根据实际情况修改服务 URL）：

```
curl -d '' https://service-xxxxxxx-125xxxxxx.gz.apigw.tencentcs.com/release/ugc_upload_sign
```

如果服务正常，则返回上传签名，签名示例如下：

```
VYapc9EYdoZLzGx0CglRW4N6kuhzZWNyZXRJZD1BS0lEZk5xMz16dG5tYW1tVzBMOXFvZERia25hUjdZa0xPM1UmY3VycmVudFRpbWVtdGFtcD0xNTg4NTg4MDIzJmV4cGlyZVRpbWU9MTU4ODU4ODYyMyZyYW5kb209MTUwNzc4JmNsYXNzSWQ9MCZvbWUaW1lVmfSaWQ9MCZ2b2RTdWJBcHBjZD0w
```

您也可以使用 Postman 等第三方工具来发送 HTTP 请求，具体用法请自行搜索。

## 上传页面服务代码解读

1. `main_handler()` 为入口函数。
2. 读取 `web_upload.html` 文件的内容，即上传页面内容。

```
html_file = open(HTML_FILE, encoding='utf-8')
html = html_file.read()
```

3. 从 `config.json` 中读取配置项。配置项是指您在编写 SCF 服务时无法预知，并且需要在部署过程中才能确定的内容。这些内容由部署脚本在部署上传页面服务之前实时写入到 `config.json` 中。

```
conf_file = open(CONF_FILE, encoding='utf-8')
conf = conf_file.read()
conf_json = json.loads(conf)
```

4. 调用 `render_template`，根据上一步得到的配置信息对上传页面内容进行修改。配置项在 `config.json` 文件中以 "变量名": "取值" 的形式来表示；在 `web_upload.html` 文件中以 {变量名} 的形式来表示，**修改时请替换为具体取值**。详情如下：

```
def render_template(html, keys):
    """将 HTML 中的变量（形式为 ${变量名}）替换为具体内容。"""
    for key, value in keys.items():
        html = html.replace("${" + key + "}", value)
    return html
```

变量名	含义	取值类型	取值来源
UGC_UPLOAD_SIGN_SERVER	上传签名派发服务的 URL	String	上传签名派发服务部署完成后，由 SCF 命令行工具 <code>scf</code> 输出

5. 将修改后的上传页面内容返回。返回的数据格式及含义请参见 [云函数集成响应] (<https://www.tencentcloud.com/document/product/583/12513>)。

```
return {
    "isBase64Encoded": False,
    "statusCode": 200,
    "headers": {'Content-Type': 'text/html'},
    "body": html
}
```

## 上传签名派发服务代码解读

- `main_handler()` 为入口函数。
- 调用 `parse_conf_file()`，从 `config.json` 文件中读取配置信息。配置项说明如下（详细参数请参见 [客户端上传签名参数](#)）：

字段	数据类型	功能
<code>secret_id</code>	String	API 密钥
<code>secret_key</code>	String	API 密钥

字段	数据类型	功能
sign_expire_time	Integer	签名有效时间，单位：秒
class_id	Integer	视频上传完成后的分类 ID，0表示默认分类
otp	Integer	签名是否单次有效
subappid	Integer	是否上传到 <a href="#">云点播子应用</a>

3. 调用 `parse_source_context()`，从请求 Body 中解析 `sourceContext` 字段，用于在 [上传完成事件通知](#) 中透传给事件通知接收服务（本 Demo 暂未使用事件通知）。

说明：

在上传过程中该字段是可选的，如果您无需使用该功能，则可以忽略这部分代码。

4. 调用 `generate_sign()` 函数计算签名，详细算法请参见 [客户端上传签名](#)。
5. 返回签名。返回的数据格式及含义请参见 [云函数集成响应](#)。

```
return {
  "isBase64Encoded": False,
  "statusCode": 200,
  "headers": {"Content-Type": "text/plain; charset=utf-8",
  "Access-Control-Allow-Origin": "*",
  "Access-Control-Allow-Methods": "POST,OPTIONS"},
  "body": str(signature, 'utf-8')
}
```

# 如何提升媒体上传速度和成功率

最近更新时间：2021-10-28 14:37:44

## 简介

无论是客户端上传还是服务端上传，在文件传输过程中都会碰到下面几个常见的质量问题：

1. 文件上传为何这么慢？
2. 如何提高上传速度？
3. 上传成功率偏低如何解决？
4. 移动端经常出现弱网问题导致上传失败，如何处理？

衡量上传质量的指标有：上传速度和成功率。

- 上传速度的快慢往往影响着用户最直观的感受。例如，用户上传了一个50M的视频，半个小时还没上传完成，随着用户耐心耗尽，便面临着流失客户的可能性。
- 上传成功率的高低是服务质量的保证，因为网络问题导致第一次上传失败后，用户再次发起上传的可能性会降低，带来的直接结果就是用户投诉，所以保证上传成功率是最基本的要求。

本文将基于云点播上传的场景，从原因分析和解决方案两个方面进行着重介绍，您可以根据自己的业务场景进行对比，选择合适的方案提高上传质量。

## 影响上传质量的原因

### 网络带宽

网络带宽是指在单位时间内能传输的数据量。带宽越大，单位时间内上传的数据量也越大，体现出来就是上传速度越快。上传是端对端的行为，所以两端的带宽对上传质量都有影响，而云点播后台服务器目前带宽充足，所以上传质量往往取决于用户侧的带宽。

### 用户与存储中心的距离

上传的文件最终在云点播都需要进行存储，存储的地方即是存储中心。用户开通云点播后，云点播默认会分配重庆地域作为存储中心。用户与存储中心的距离会影响网络链路的长度。

例如，相同文件，用户在北京上传到重庆，相比在成都上传到重庆，中间经过的链路会更长，影响的因素也会随着距离的增加而变多，最终导致上传速度变慢。因为长链路传输，中间环节出现的网络抖动、丢包等问题，也会影响上传成功率。短链路也不能避免这些问题，只是相比长链路，出现的概率会降低不少。减少用户与存储中心的距离，是提高上传质量的关键一步。

## 弱网

弱网是指一种网络的状态，出现高延迟、高丢包，表现为即我们平时所说的“上网很慢”。弱网问题在现实生活中也很常见，例如在电梯和地铁里，主要原因是所处环境的信号传输不佳，导致数据包传输慢或者失败。这种场景在客户端上传尤为居多，特别是目前移动互联网盛行的时代，弱网问题困扰着许多开发者，这个也是提高成功率指标最难突破的问题。

## 解决方案

### 并发上传

对于网络带宽不足的场景，直接的办法就是申请更大的带宽。如果我们面临的是一个有限的带宽网络，如何充分的利用带宽进行上传，这是需要我们解决的。并发上传分为两个维度：

- 文件级别，多个文件同时进行上传。
- 分片级别，单个文件多个分片同时进行上传。

无论是前者还是后者，都可以通过调整对应的并发数，提高带宽的利用率。

### 文件并发上传

文件并发上传是利用多个进程或者线程，同时发起上传操作。目前云点播没有提供相关的 SDK 包装，用户可以参考具体的语言特性进行实现，下面基于云点播 [Java SDK](#) 提供一个简单的示例：

```
import com.qcloud.vod.VodUploadClient;
import com.qcloud.vod.model.VodUploadRequest;
import com.qcloud.vod.model.VodUploadResponse;
import java.util.ArrayList;
import java.util.List;
import java.util.concurrent.ExecutorService;
import java.util.concurrent.Executors;
public class Main {
public static void main(String[] args) throws Exception {
// 并发数
Integer threadNumber = 20;
// 待上传的文件路径列表
List<String> filePathList = new ArrayList<String>();
// 添加待上传文件路径
filePathList.add("/data/path1.mp4");
filePathList.add("/data/path2.mp4");
filePathList.add("/data/path3.mp4");
// 创建线程池
ExecutorService pool = Executors.newFixedThreadPool(threadNumber);
// 创建上传Client
```

```
VodUploadClient client = new VodUploadClient("your secretId", "your secretKey");
// 并发上传
for (String path : filePathList) {
    // 提交上传任务
    pool.submit(new UploadThread(client, path));
}
// 上传线程
class UploadThread implements Runnable {
    // 上传 Client
    private VodUploadClient uploadClient;
    // 文件路径
    private String filePath;
    public UploadThread(VodUploadClient uploadClient, String filePath) {
        this.uploadClient = uploadClient;
        this.filePath = filePath;
    }
    public void run() {
        VodUploadRequest request = new VodUploadRequest();
        request.setMediaFilePath(filePath);
        try {
            // 执行上传
            VodUploadResponse response = uploadClient.upload("ap-guangzhou", request);
            System.out.println(response.getFileId());
        } catch (Exception e) {
            e.printStackTrace();
        }
    }
}
```

## 分片并发上传

分片并发上传是针对大文件，拆分成多个分片同时进行上传。分片并发上传的优势在于可以快速完成单个文件的上传，云点播提供的 SDK 会根据用户文件的长度，自动选择普通上传以及分片上传，用户不用关心分片上传的每个步骤，即可实现分片上传。而文件的分片并发数通过 `ConcurrentUploadNumber` 参数进行指定，具体的使用示例参考对应 SDK，目前已经支持该参数的 SDK 列表如下：

- [Java SDK](#)
- [Python SDK](#)
- [Go SDK](#)

## 就近上传

就近上传是指能够感知上传者所处的位置，分配离上传者最近的存储中心进行上传。例如，成都的用户会分配重庆地域进行上传，而不会分配到上海地域。

就近上传的能力带来最大的收益是减少了上传者到服务器的传输距离，其优点如下：

- 减少了传输距离，提高了上传速度。
- 提高了稳定性，保证了成功率。

云点播本身支持就近上传的能力，您只需简单确认下面两点：

#### • 开通多个存储地域

云点播默认提供的存储地域设置在**新加坡**，如果要充分利用就近上传的能力，就需要开通希望就近存储的区域列表，可以在控制台上自助添加其他存储地域，详情请参见[上传存储设置](#)。启用多地存储后，当用户上传时，通过IP识别用户所在的地域，云点播根据已经开通的地域列表，智能分配一个最近地域用于上传。

#### • 校验是否正确调度

如果开通了重庆及上海的存储地域，用户在成都发起上传，通过就近调度，理论上应该是上传到重庆。如何确认调度是否合理，可以获取完成上传返回的 `FileId`，通过[获取媒体详细信息接口](#)返回的基础信息（`basicInfo`）进行确认，其中就包含了 `StorageRegion` 字段，代表媒体文件存储地区。

如果上传者中间经过了代理或者转发，导致最后云点播通过IP识别地区有误，也可以通过强制指定存储地域，上传到目标地域中，具体用法请参见：

- [客户端上传指引](#)
- [服务端上传指引](#)

## 预探测上传

预探测上传主要是一种针对各类网络错误场景进行优化的手段，例如网络连接失败、超时、DNS劫持等，是云点播针对弱网上传提出的一种有效缓解方案，优化的策略包括下面几点：

- 使用 HTTPDNS 解析域名获取后端地址，防止 DNS 劫持问题。
- 探测多个地域的连通性及上传速度，获取最优上传地域。
- 接入腾讯云加速网络，提供可靠稳定的传输通道。

预探测上传的能力目前应用在客户端上传中，接入方式简单，具体的用法参考 SDK 中 `预上传` 的说明：

- [Android 上传 SDK](#)
- [iOS 上传 SDK](#)



# 如何将点播的媒体文件进行智能降冷

最近更新时间：2022-05-26 12:13:16

智能降冷依托于点播的媒资管理能力，为了降低开发者使用云点播服务时的存储成本，我们推出了媒体生命周期管理能力。开发者可以根据自身的业务特点，从而达到有效降低用户存储成本，将点播文件的存储类型按照一定的策略从标准存储沉降为低频存储、归档存储和深度归档存储，更灵活的管理自己的媒体文件。

## 使用场景

- **直播电商**：按照文化部《[网络交易监督管理办法](#)》的要求，网络直播服务提供者对网络交易活动的直播视频保存时间自直播结束之日起不少于三年。直播录像存储在云点播中，存储类型为标准存储。部分录像不会（或极少）被回看，仅用于有关部门的审查，媒资智能降冷功能可以有效帮助客户降低存储成本。
- **低频率访问媒体降冷**：对于视频门户、流媒体、UGC 内容管理平台，对于用户不频繁访问和观看的媒体，不能直接下线又带来较高的存储成本，云点播的媒资智能降冷能力可以通过访问次数来将媒体文件进行降冷，有效降低存储成本的同时，满足低频观看的要求。
- **媒体归档**：对于新闻传媒、广播电视行业，部分媒体文件具有很强的时效性，这类媒体一般会作为历史资料被长期保存，在日后需要的时候再检索出来观看，对观看的实时性要求不高。这种场景下可以将媒体转为归档存储或深度归档存储，降低用户存储成本。

## 前提条件

1. [注册](#) 并 [登录](#) 腾讯云账号，并且完成账号实名认证。
2. 已开通腾讯云点播服务。若未开通，请前往开通 [云点播服务](#)。
3. 进行策略创建，详情请参见 [媒资降冷](#)。

## 使用说明

为了便于用户理解和使用媒资降冷能力，用户需要了解 [存储类型](#)、[数据取回及取回模式](#)、[策略管理](#) 的概念。

### 存储类型

用户媒体文件存储至云点播时的存储类型，云点播提供了以下几种存储类型，分别为：标准存储、低频存储、归档存储和深度归档存储。其特性如下表所示：

存储类型	标准存储	低频存储	归档存储	深度归档存储
------	------	------	------	--------

存储类型	标准存储	低频存储	归档存储	深度归档存储
云点播默认	是	否	否	否
存储成本	高	中	低	极低
访问性能	高	低	不支持访问	不支持访问
数据取回费用	无	无	有	有
支持园区	全部	全部	全部	北京、上海、重庆

标准存储是云点播的默认存储类型。不论是直播录制、各种上传方式还是视频处理任务所生成的文件都是标准存储。

属性	由高到低对比
存储成本	标准存储 > 低频存储 > 归档存储 > 深度归档存储
访问性能	标准存储 > 低频存储

说明：

- 归档存储和深度归档存储不支持直接访问，需要先进行取回操作，云点播仅支持用户取回至 **标准存储**。
- 访问性能将影响观看视频时的**首屏时间**、**卡顿率**等指标，故**对于频繁访问的线上业务不建议用户变更存储类型**。

云点播支持不同存储类型进行变更，所允许的存储类型变更方式如下：

源存储类型	目标存储类型
标准存储	低频存储、归档存储、深度归档存储
低频存储	标准存储、归档存储、深度归档存储
归档存储	标准存储
深度归档存储	标准存储

说明：

- 若要将归档存储和深度归档存储变更为除标准存储之外的其它存储类型，需要先变更为标准存储，然后再由标准存储变更为最终的目标存储类型。
- 存储类型的变更以 FileId 为粒度，即原始文件和视频处理文件的存储类型是一致的，不支持对 FileId 下的一部分文件单独指定存储类型。

## 数据取回及取回模式

### 数据取回

媒体变更为归档存储或深度归档存储之后不允许被直接访问，包括播放、发起视频处理等。如果需要访问，那么可以使用上文介绍的变更存储类型操作。但有时候开发者仅需要在短时间内进行访问，访问结束后希望该媒体仍然能保留归档存储或深度归档存储状态，那么上述的方式就不适用。因此点播提供了数据取回能力。

云点播支持对归档存储和深度归档存储的媒体进行两种取回操作：永久取回、指定时长取回。

对归档存储/深度归档存储的操作	变更存储类型	取回
实时性	异步	异步
模式	极速/标准/批量	极速/标准/批量
有效期	永久	指定时长
产生副本	否	是

1. 取回之后，云点播会为该媒体生成一份存储类型为标准存储的副本。
2. 取回副本具备有效期，以天为单位。在有效期之内，允许对该媒体进行访问；在有效期结束后，副本被自动清理，之后不允许对媒体进行访问。
3. 副本有效期内，会额外产生相应的标准存储费用。
4. 副本有效期内，不允许再次发起取回操作。

### 取回模式

对归档存储和深度归档存储变更存储类型和取回操作有多种模式。不同模式的最终效果一致，区别在于速度和费用（即取回费用）：

取回模式	归档存储取回耗时	深度归档存储取回耗时
极速模式	5分钟	不支持
标准模式	5小时	24小时
批量模式	12小时	48小时

说明：

- 一个媒体（由一个 FileId 表示）在存储上可以包含许多个文件，如原始文件、转码文件和截图文件等等，各个文件的实际解冻完成时间可能有先后。云点播不维护各个文件的具体解冻完成情况，而是将媒体作为一个整体，统一以解冻的最大可能耗时来计算媒体的解冻完成时间。在该时间到达之前，即使媒体下的所有文件实际已经解冻完成，媒体仍然被标识为未解冻状态，不允许访问。
- 由于媒体被标识为已解冻的时间点会晚于实际解冻的时间点，因此媒体的解冻副本的可用时长会短于预期想要的时长。为了确保解冻后副本有足够的可用时长，建议在解冻深度归档存储时加多1天的有效期。

## 策略管理

为了方便开发者对大量媒体的生命周期进行统一管理，云点播结合媒资信息和媒体文件播放统计数据，开发了基于策略的智能管理系统。

云点播后台每天执行管理任务，对满足指定策略的媒体进行存储类型变更。

## 策略能力

### 组合筛选

策略允许开发者指定一系列条件，对**同时**符合所有条件的媒体进行存储类型变更。具体筛选条件如下：

- 以媒体的类型作为筛选条件，包括视频媒体、音频媒体和图片媒体。
- 以媒体的创建日期作为筛选条件，有以下几种形式：
  - 指定日期之前创建的文件。
  - 指定日期之后创建的文件。
  - 指定的两个日期之间创建的文件。
  - 指定天数之前创建的文件（根据当前日期动态变化）。
  - 创建时间不限（即所有媒体均满足该条件）。
- 以媒体的分类 ID 作为筛选条件，可以同时指定多个分类 ID，也可以不做限制（即所有媒体均满足该条件）。
- 以媒体的来源类型作为筛选条件，包括直播录制、上传、视频编辑、其它等几类。可以同时指定多种来源类型，也可以不做限制（即所有媒体均满足该条件）。
- 以媒体的近期播放次数作为筛选条件，筛选出最近若干天内播放次数少于若干次的媒体。也可以不做限制（即所有媒体均满足该条件）。

### 停用策略

策略创建完成后自动启用，开发者可以根据需要停用策略。已停用的策略在次日之后的管理任务中被忽略，直到该策略被再次启用。

说明：

停用策略后，该策略在当天的管理任务中有可能生效，也有可能不生效。

## 限制

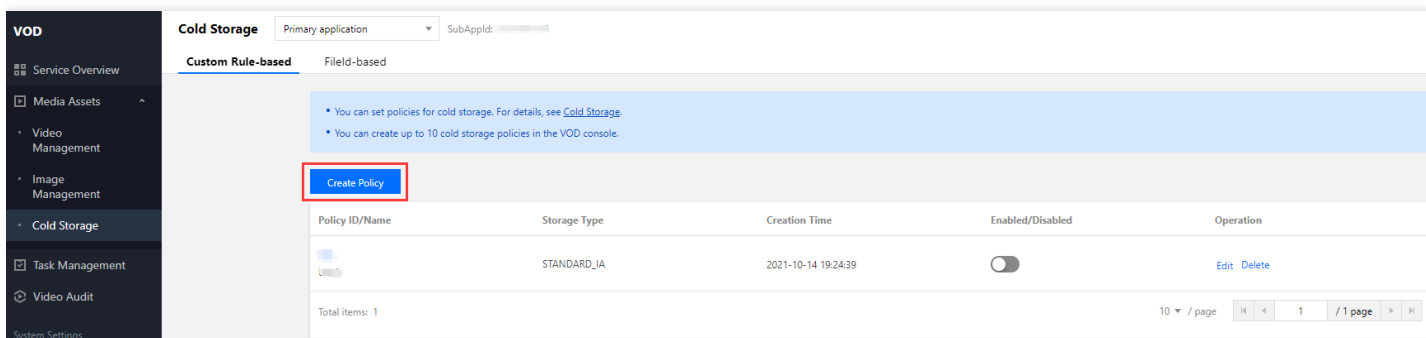
最多配置10条策略。

不允许指定策略优先级。如果一个媒体同时命中多条策略，自动按照目标存储类型来确定优先级。优先级从高到低为：深度归档存储 > 归档存储 > 低频存储。

## 云点播配置智能降冷策略

### 步骤1：创建降冷策略

在 [云点播控制台](#)（非管理员）左侧导航栏，选择【媒资管理】>【媒资降冷】，单击【智能降冷策略】。



### 步骤2：配置降冷策略

← **Create Policy** test Su

Policy Name: test  
Max 20 characters; allows letters, digits, spaces and \_.

Target Storage Class: STANDARD\_IA

Filter:

- Time
- Media File Category
- Upload Source
- Playback-based

If a video is played back less than 300 time(s) within 30 day(s), the STANDARD\_IA storage policy will be triggered.

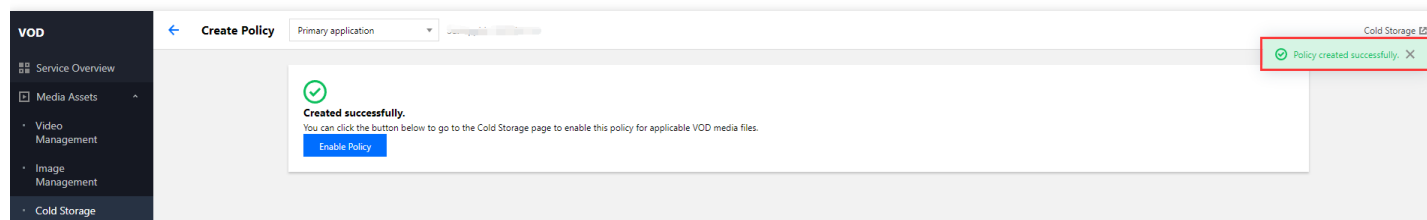
Media File Type

Confirm Cancel

按照用户的需求进行降冷策略的配置。

示例：如图所示则为若媒体在30日内，视频播放次数低于300次，则触发低频降冷策略；云点播会选取满足所有规则的媒体进行降冷。

### 步骤3：确认配置生效



# 媒体处理

## 如何对视频进行转码

最近更新时间：2023-05-15 17:15:06

### 使用须知

#### 内容介绍

本文档向开发者介绍如何对云点播（VOD）中的视频进行转码，以及如何获取转码后的输出结果。

#### 费用

本文提供的代码是免费开源的，但在使用的过程中可能会产生以下费用：

购买腾讯云云服务器（CVM）用于执行云 API 请求脚本，详见 [CVM 计费](#)。

消耗 VOD 存储用于存储上传的视频，详见 [存储计费](#) 和 [存储资源包](#)。

消耗 VOD 转码时长用于对视频进行转码，详见 [转码计费](#) 和 [转码资源包](#)。

消耗 VOD 流量用于播放视频，详见 [流量计费](#) 和 [流量资源包](#)。

#### 参数介绍

腾讯云点播视频转码目前支持以下视频格式：

参数	类型	详细说明
输入格式	封装格式	WMV、RM、MOV、MPEG、MP4、3GP、FLV、AVI、RMVB、TS、ASF、MPG、WEBM、MKV、M3U8、WM、ASX、RAM、MPE、VOB、DAT、MP4V、M4V、F4V、MXF、QT、OGG。
	视频编码格式	AV1、AVS2、H.264/AVC、H.263、H.263+、H.265、MPEG-1、MPEG-2、MPEG-4、MJPEG、VP8、VP9、Quicktime、RealVideo、Windows Media Video。
输出格式	封装格式	视频：FLV、MP4、HLS（m3u8+ts）。
		音频：MP3、MP4、OGG、FLAC、m4a。
	图片：GIF、WEBP。	
视频编码格式	H.264/AVC、H.265/HEVC、AV1。	

转码的目标规格包含了编码格式、分辨率和码率等参数。云点播使用转码模板表示转码参数集合，通过转码模板，可以指定以下转码相关参数，请参见 [视频处理综述](#)。

分类	参数	说明
视频编码	编码方式 (Codec)	支持 H.264、H.265以及 AV1 编码格式。
	码率 (Bitrate)	支持的视频码率范围：10kbps - 35Mbps。
	帧率 (Frame Rate)	支持的帧率范围：1fps - 60fps，常见的有24fps、25fps和30fps。
	分辨率 (Resolution)	支持宽度范围：128px - 4096px。 支持高度范围：128px - 4096px。
	GOP 长度	支持 GOP 长度范围：1秒 - 10秒。
	编码档次 (Profile)	当视频编码方式为 H.264 时，支持 Baseline、Main 和 High 的编码档次。 当视频编码方式为 H.265 时，仅支持 Main 编码档次。
	颜色空间 (Color Space)	支持 YUV420P。

#### 说明：

**编码方式：**指通过特定的压缩技术，将某个视频格式的文件转换成另一种视频格式文件的方式。H.265 相比 H.264 则采用更先进的编码方式转码，在不损失原始画质的情况下显著降低码率，节省播放带宽。

**码率：**编码器每秒编出的数据大小，单位是kbps，例800kbps代表编码器每秒产生800kb的数据。

**帧率 (FPS)：**指每秒钟要多少帧画面。

**分辨率：**单位英寸中所包含的像素点数。

**GOP：**通常指两个I帧的间隔。

对于普通转码，不同清晰度，推荐使用的码率、分辨率及设置区间如下表所示：

清晰度	推荐码率	推荐分辨率	分辨率区间
标清	600	640x480	标清 SD (短边 ≤ 480px)
高清	2000	1280x720	高清 HD (短边 ≤ 720px)
全高清	4000	1920x1080	全高清 FHD (短边 ≤ 1080px)
2K	6000	2560x1440	2K (短边 ≤ 1440px)
4K	8000	3840x2160	4K (短边 ≤ 2160px)

腾讯云点播特有的极速高清集成了画质修复与增强、内容自适应参数选择、V265编码器等一系列视频处理解决方案。提供让视频更小更清晰的转码方式，能够保证网络资源低消耗，同时带给用户更佳视觉体验。点播也预置了各种清晰度，具体参数如下：

清晰度	推荐码率	推荐分辨率	分辨率区间



标清	350或不设	640x480	标清 SD (短边 ≤ 480px)
高清	1350或不设	1280x720	高清 HD (短边 ≤ 720px)
全高清	2700或不设	1920x1080	全高清 FHD (短边 ≤ 1080px)
2K	3500或不设	2560x1440	2K (短边 ≤ 1440px)
4K	7500或不设	3840x2160	4K (短边 ≤ 2160px)

### 说明：

若不设，极速高清会根据视频源智能分析，智能设置视频最低码率。

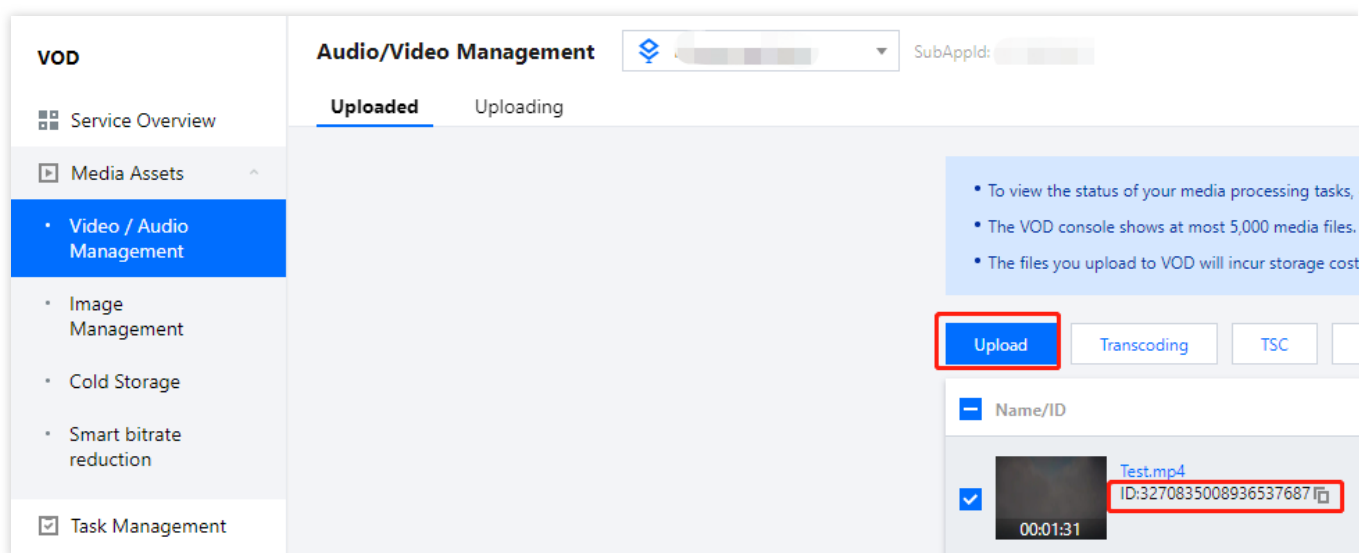
## 在控制台发起转码

### 步骤1：开通云点播

请参考 [快速入门 - 步骤1](#) 开通云点播服务。

### 步骤2：上传视频

参考 [快速入门 - 步骤2](#) 上传一个测试视频。单击 [此处](#) 查看本 Demo 使用的测试视频，对应的 FileId 是 3270835008936537687，如下图所示：

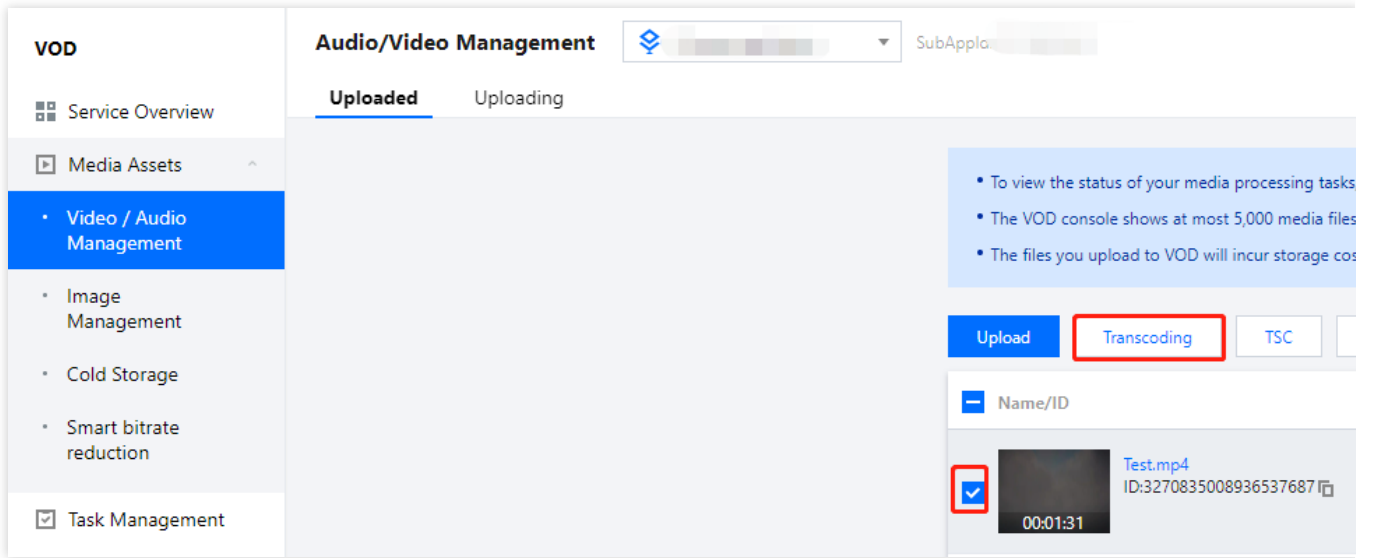


### 说明：

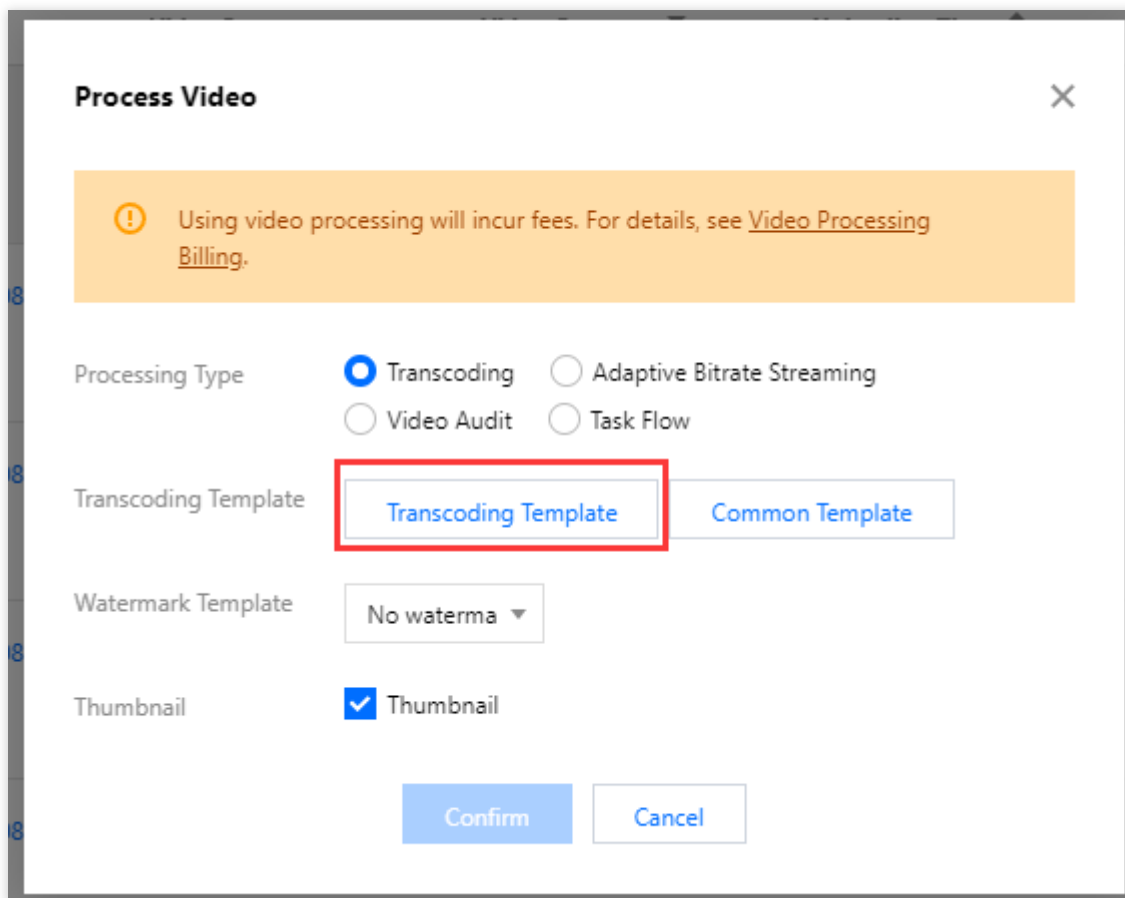
建议使用较短的视频文件进行测试（例如时长为几十秒的视频），避免转码过程耗时太长。

### 步骤3：发起转码

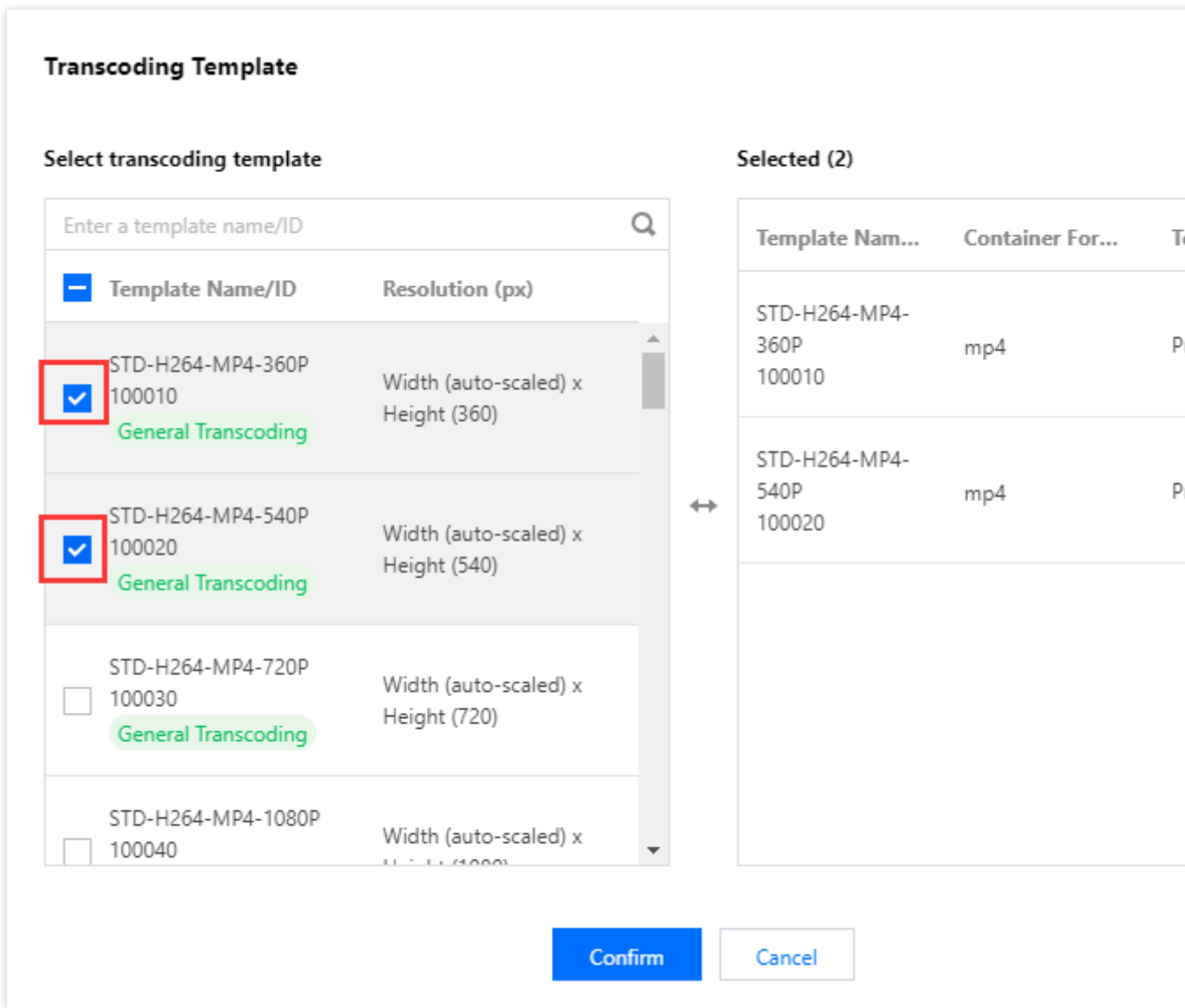
1. 在控制台 [音视频管理](#) 页面勾选新上传的测试视频，然后单击**转码**。



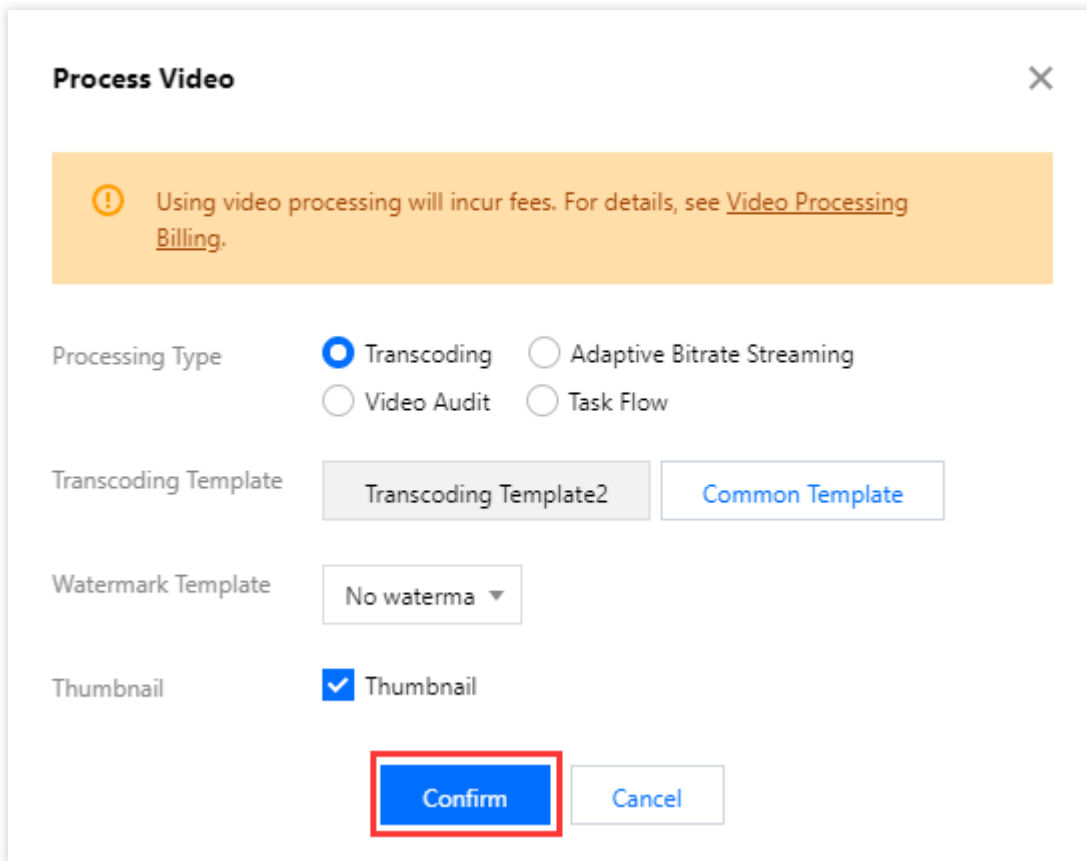
2. 在弹框中，处理类型选择“转码”，然后单击**转码模板**。



3. 选择所需的转码模板，然后单击**确定**。本 Demo 以系统预置模板 STD-H264-MP4-360P（模板 ID 100010）和 STD-H264-MP4-540P（模板 ID 100020）为例，如果开发者需要使用自定义的转码模板，请参考 [模板设置文档](#)。



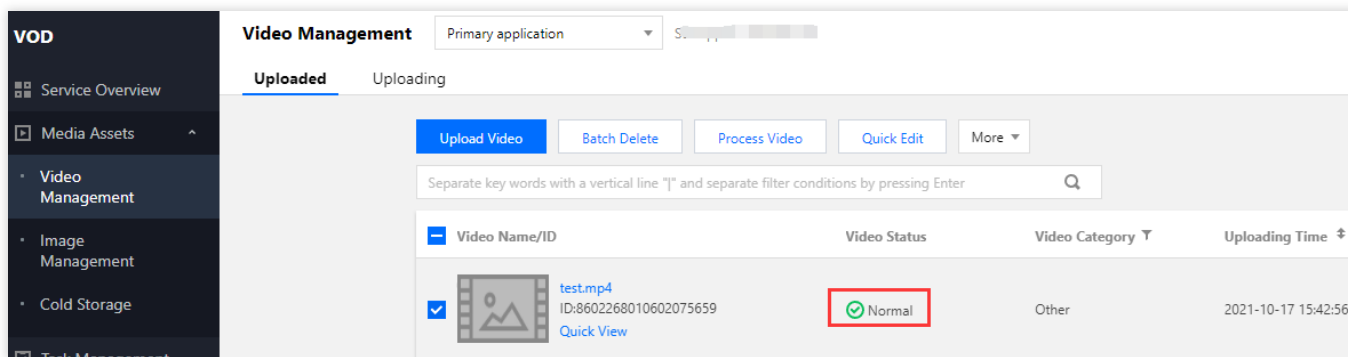
4. 单击**确定**，发起转码。



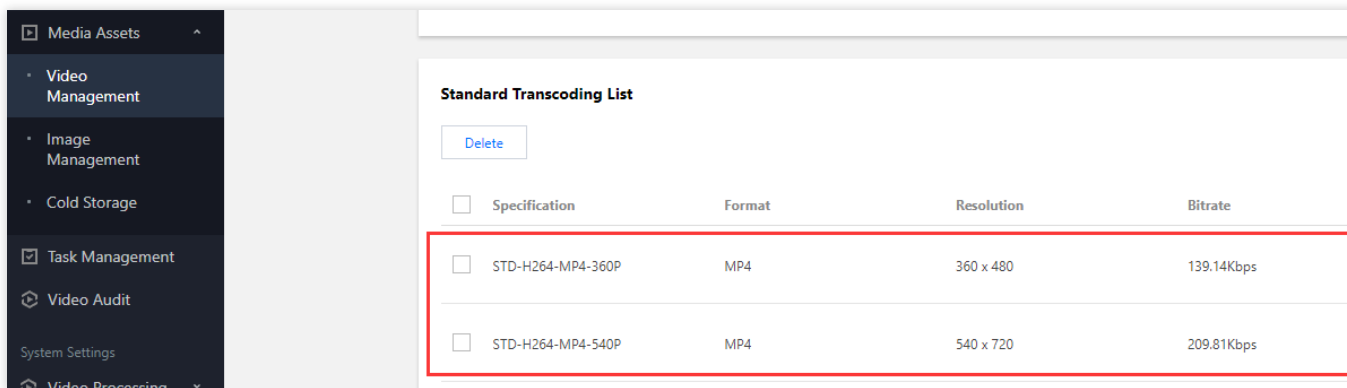
5. 在“任务中心”页面看到测试视频的状态为“已完成”，则表示音视频转码成功。

#### 步骤4：查看转码结果

1. 在控制台 [应用管理](#) > 媒资管理 > [音视频管理](#) 页面等待测试视频的状态变为“正常”，此时表示转码已完成。单击测试视频右侧的 [管理](#)，进入视频管理页面。



2. 在“基本信息”标签页下的“标准转码列表”中，转出了 STD-H264-MP4-360P 和 STD-H264-MP4-540P 两个规格。开发者可以单击右侧的 [预览](#) 直接观看视频，还可以单击 [复制地址](#) 复制转码视频的 URL，然后通过其它渠道发布给观众。



## 调用云 API 发起转码

### 步骤1：准备腾讯云 CVM

云 API 请求脚本需要运行在一台腾讯云 CVM 上，要求如下：

地域：任意。

机型：官网最低配置（1核1GB）即可。

公网：需要拥有公网 IP，带宽1Mbps或以上。

操作系统：官网公共镜像 `Ubuntu Server 16.04.1 LTS 64位` 或 `Ubuntu Server 18.04.1 LTS 64位`。

购买 CVM 的方法请参见 [操作指南 - 创建实例](#)。重装系统的方法请参见 [操作指南 - 重装系统](#)。

#### 注意：

如果您没有符合上述条件的腾讯云 CVM，也可以在其它带外网的 Linux（如 CentOS、Debian 等）或 Mac 机器上执行脚本，但需根据操作系统的区别修改脚本中的个别命令，具体修改方式请开发者自行搜索。

### 步骤2：获取 API 密钥

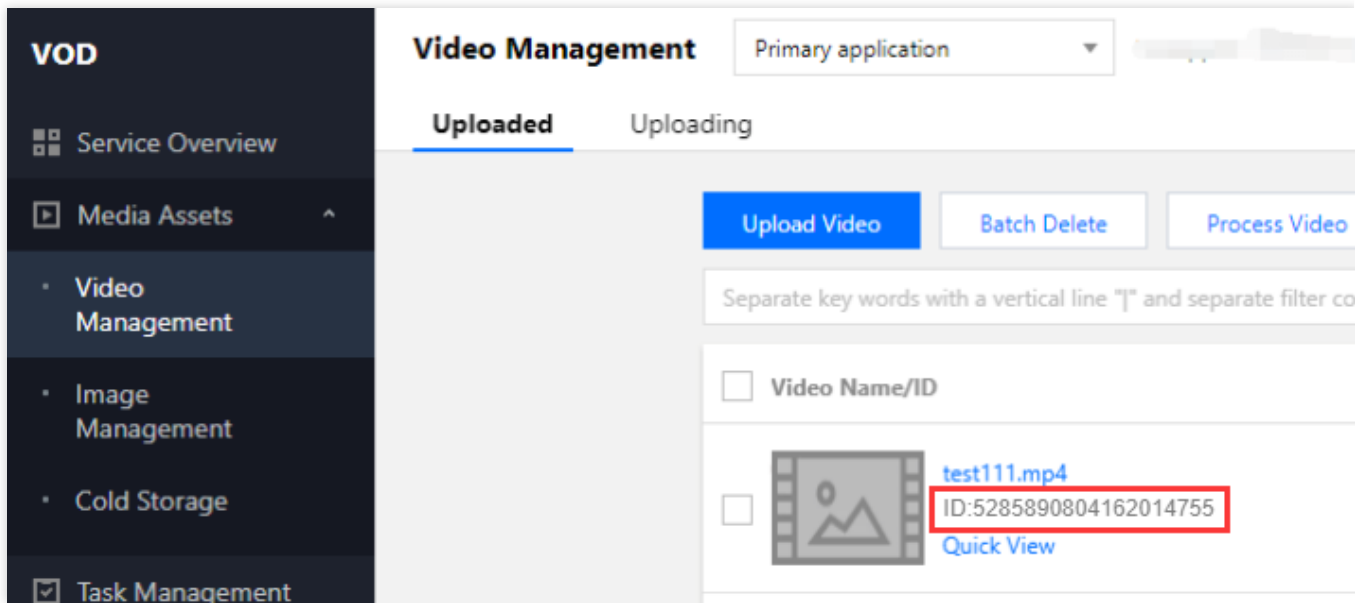
请求云 API 需要使用到开发者的 API 密钥（即 SecretId 和 SecretKey）。如果还未创建过密钥，请参见 [创建密钥文档](#) 生成新的 API 密钥；如果已创建过密钥，请参见 [查看密钥文档](#) 获取 API 密钥。

### 步骤3：开通云点播

请参考 [快速入门 - 步骤1](#) 开通云点播服务。

### 步骤4：上传视频

参考 [快速入门 - 步骤2](#) 上传一个测试视频。单击 [此处](#) 查看本 Demo 使用的测试视频，对应的 FileId 为 243791581340253754，如下图所示：

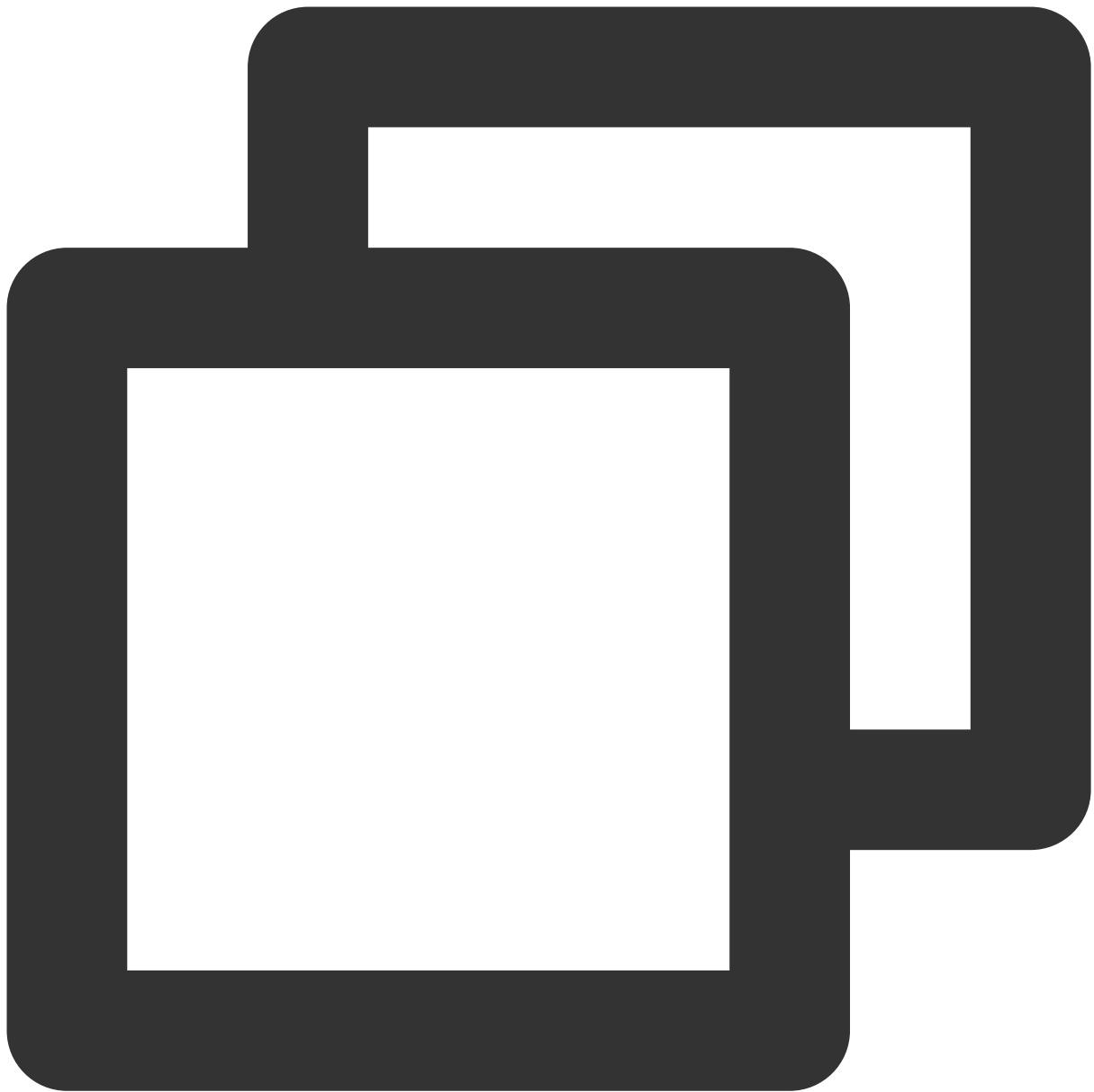


#### 说明：

建议使用较短的视频文件进行测试（例如时长为几十秒的视频），避免转码过程耗时太长。

#### 步骤5：发起转码

1. 登录 [步骤1](#) 中准备好的 CVM（登录方法详见 [操作指南 - 登录 Linux](#)），在远程终端输入以下命令并运行：

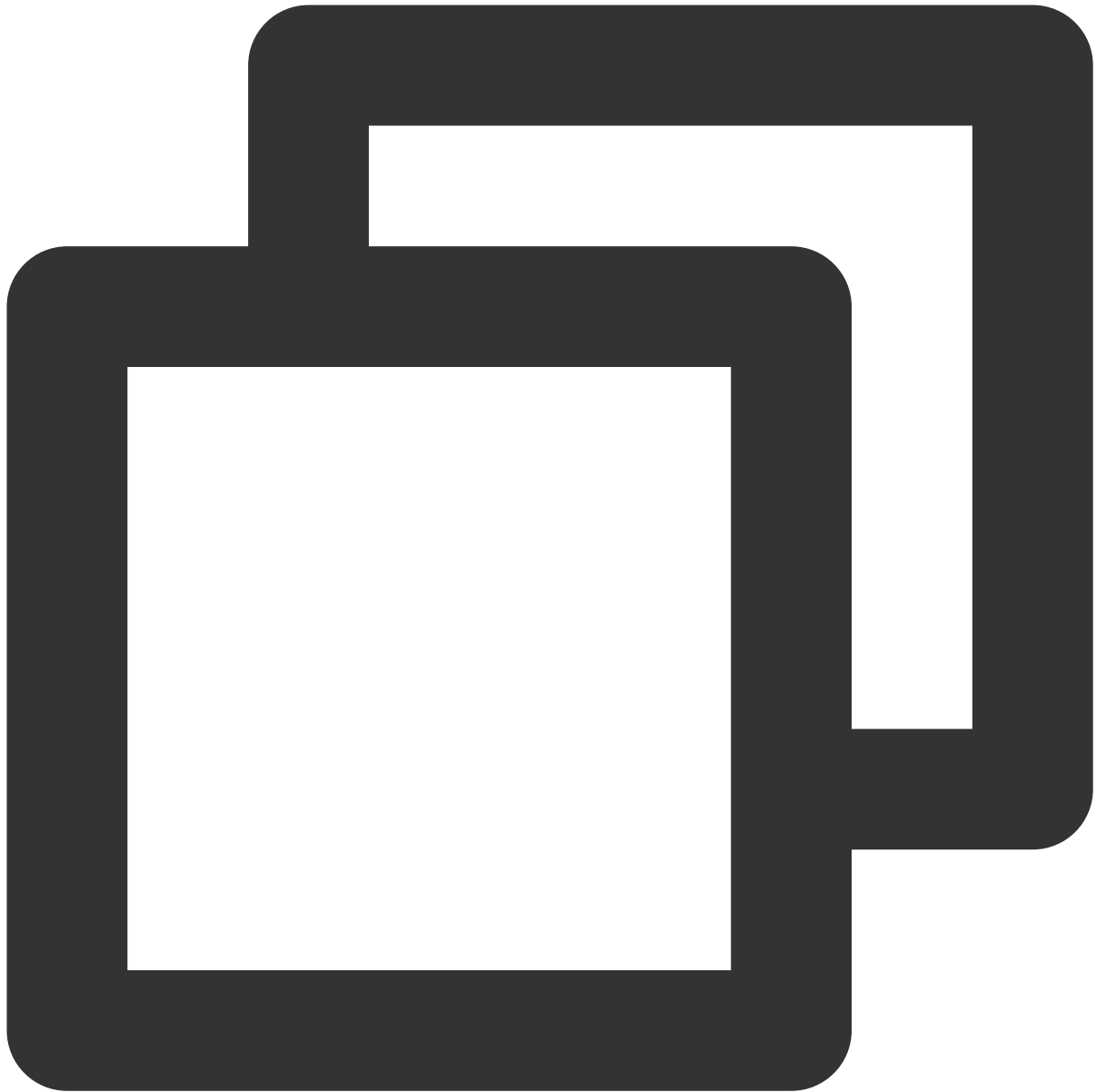


```
ubuntu@VM-69-2-ubuntu:~$ export SECRET_ID=AKxxxxxxxxxxxxxxxxxxxxxxxxxxxx; export SECRET_
```

#### 说明：

请将命令中的 SECRET\_ID 和 SECRET\_KEY 赋值为 [步骤2](#) 中获取到的内容。

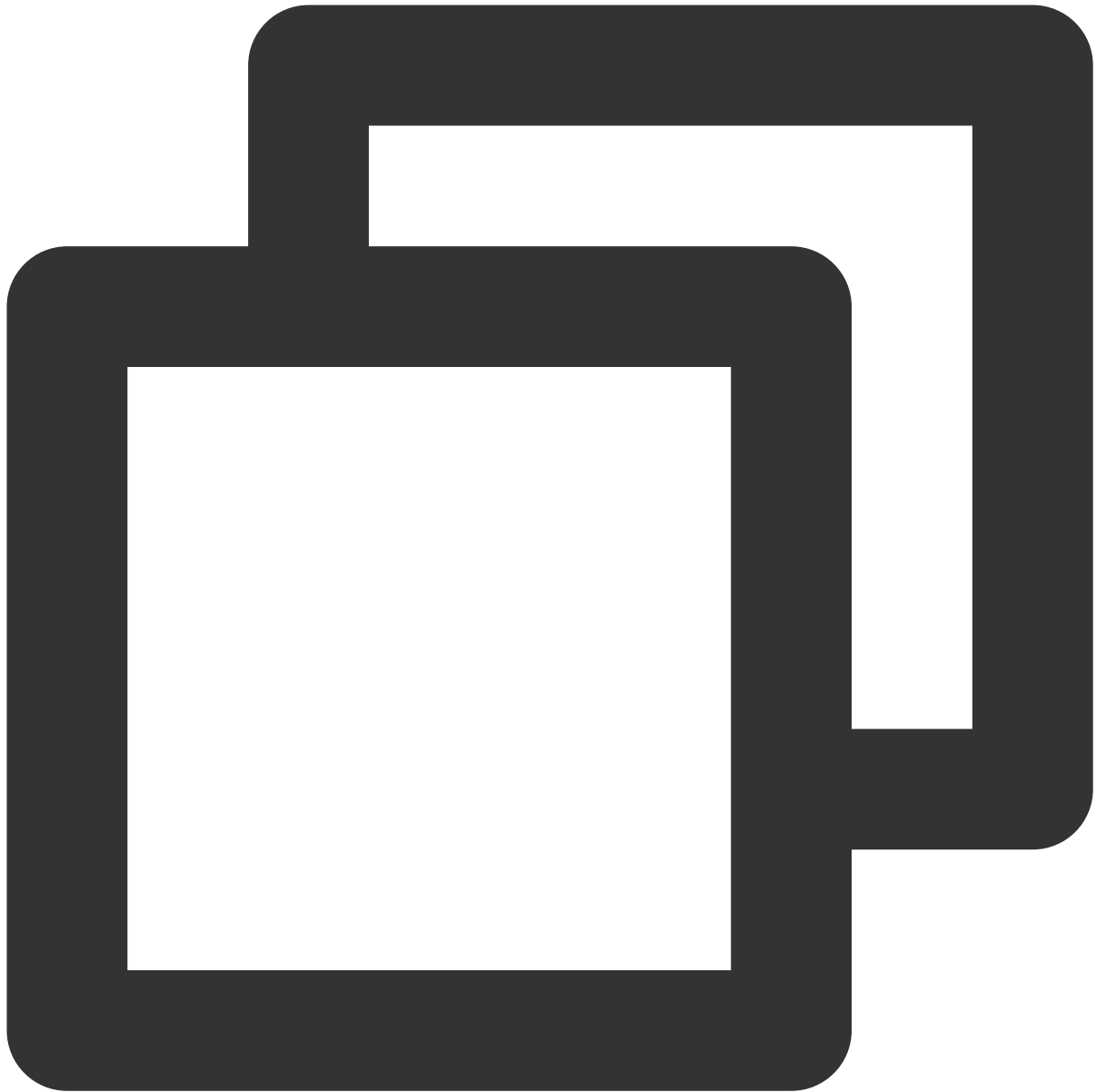
2. 该命令将从 Github 下载 Demo 源码并自动执行安装脚本。安装过程需几分钟（具体取决于 CVM 网络状况），期间远程终端会打印如下示例的信息：



```
[2020-06-15 20:39:56]开始安装 pip3。  
[2020-06-15 20:40:06]pip3 安装成功。  
[2020-06-15 20:40:06]开始安装云 API Python SDK 。  
[2020-06-15 20:40:07]云 API Python SDK 安装完成。  
[2020-06-15 20:40:07]开始配置 API 参数。  
[2020-06-15 20:40:07]API 参数配置完成。
```

3. 执行 `process_media.py` 脚本发起转码：



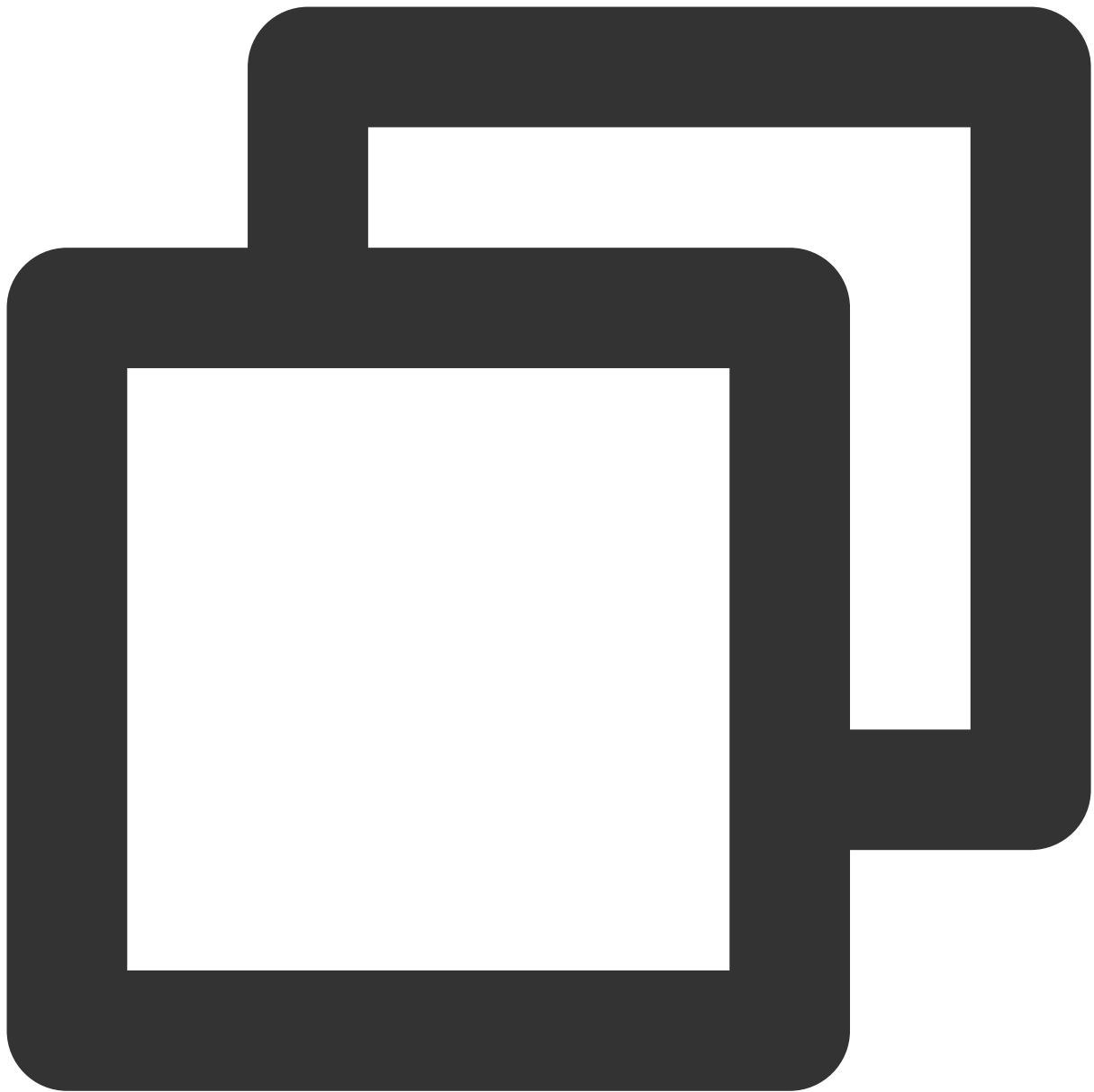


```
ubuntu@VM-69-2-ubuntu:~$ cd ~/vod-server-demo/transcode_api/; python3 process_media
```

**说明：**

请将命令中的243791581340253754替换为 [步骤4](#) 中得到的实际 FileId。

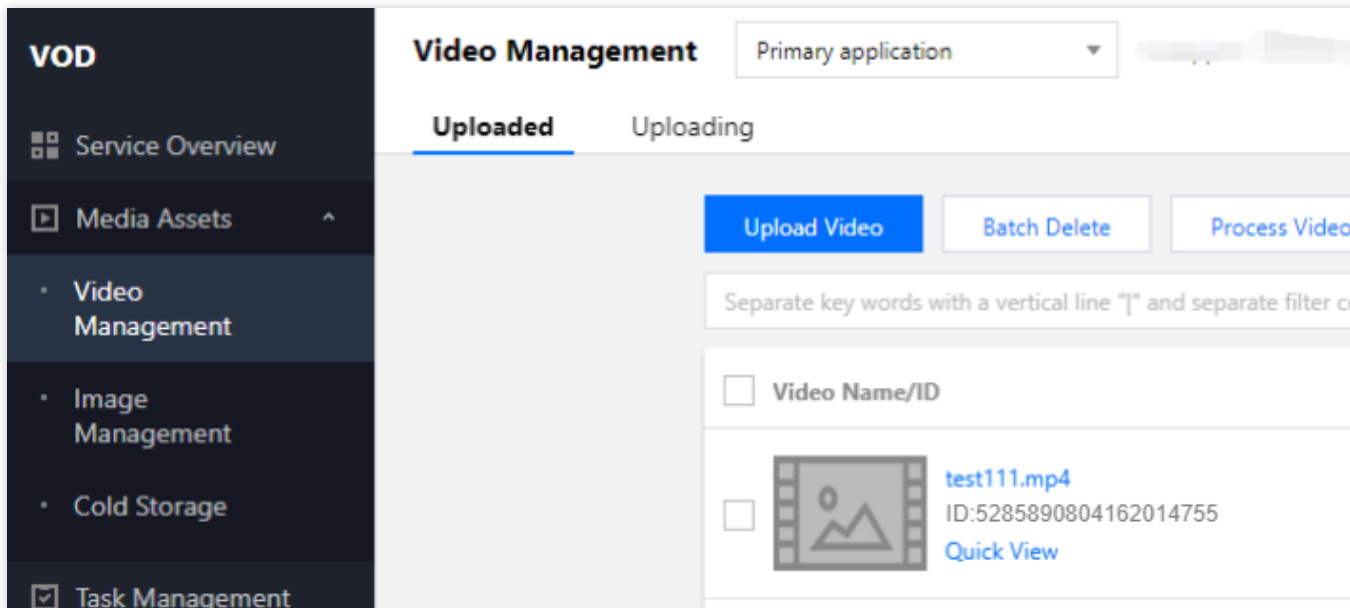
4. 该命令将对243791581340253754这个视频发起 [ProcessMedia](#) 请求，按 VOD [预置转码模板](#) 100010和100020两个规格进行转码，并打印请求的应答内容：



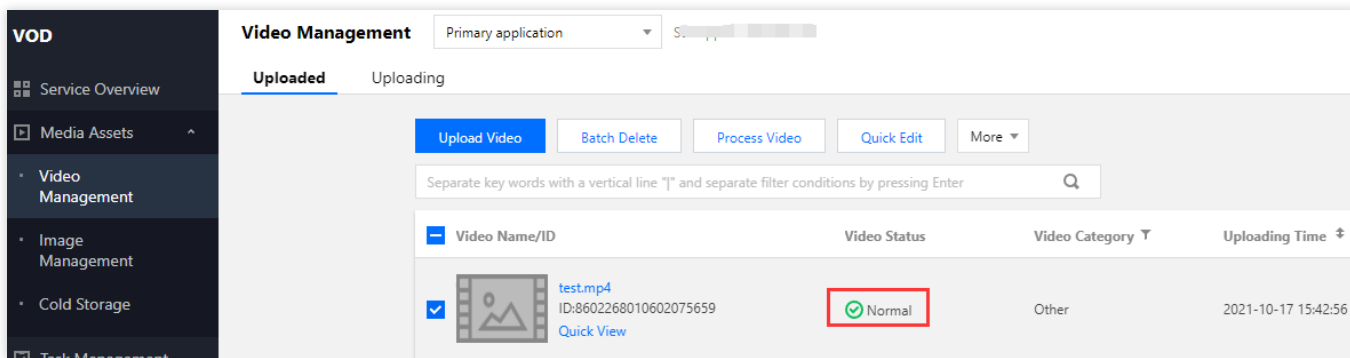
```
{"TaskId": "1400329073-procedurev2-f6bf6f01612369b6db30f2224792a2aft0", "RequestId"}
```

### 步骤6：查看转码结果

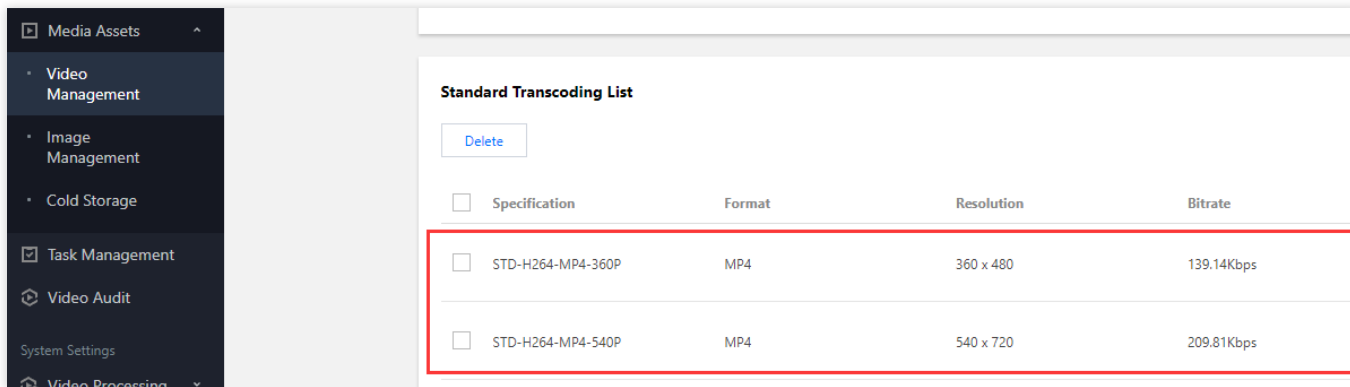
1. 在“任务中心”页面看到测试视频的状态为“已完成”，则表示音视频转码成功。



2. 等待测试视频的状态变为“正常”，此时表示转码已完成。单击测试视频右侧的**管理**，进入视频管理页面。



3. 在“基本信息”标签页下的“标准转码列表”中，转出了对应规格的视频。开发者可以单击右侧的**预览**直接观看视频，还可以单击**复制地址**复制转码视频的 URL，然后通过其它渠道发布给观众。



---

## 上传视频后自动转码（任务流）

VOD 有多种上传视频的方式，包括控制台上传、服务端上传、客户端上传和 URL 拉取上传等（详见 [媒体上传综述](#)）。各种上传方式均支持指定一个 [任务流](#)，在上传完成后自动触发转码。

## 上传视频后自动转码（事件通知）

VOD 后台在视频上传完成和转码任务完成后均会发起 [事件通知](#) 请求。开发者可以利用事件通知机制来对新上传的视频发起转码，也可以通过事件通知来自动获取转码结果（上文展示的方法是人工在控制台查看转码结果）。事件通知的使用方法在单独的最佳实践文档 [接收事件通知](#) 中有详细介绍。

# 如何对媒体进行内容审核

最近更新时间：2023-04-27 15:00:41

媒体生产业务存在内容不可控性，如有存在不合规的内容，可能带来相关法律风险和品牌伤害。为确保业务健康，保障服务稳定，云点播提供了 [音视频审核](#) 以及 [图片审核](#) 能力。

## 音视频内容审核

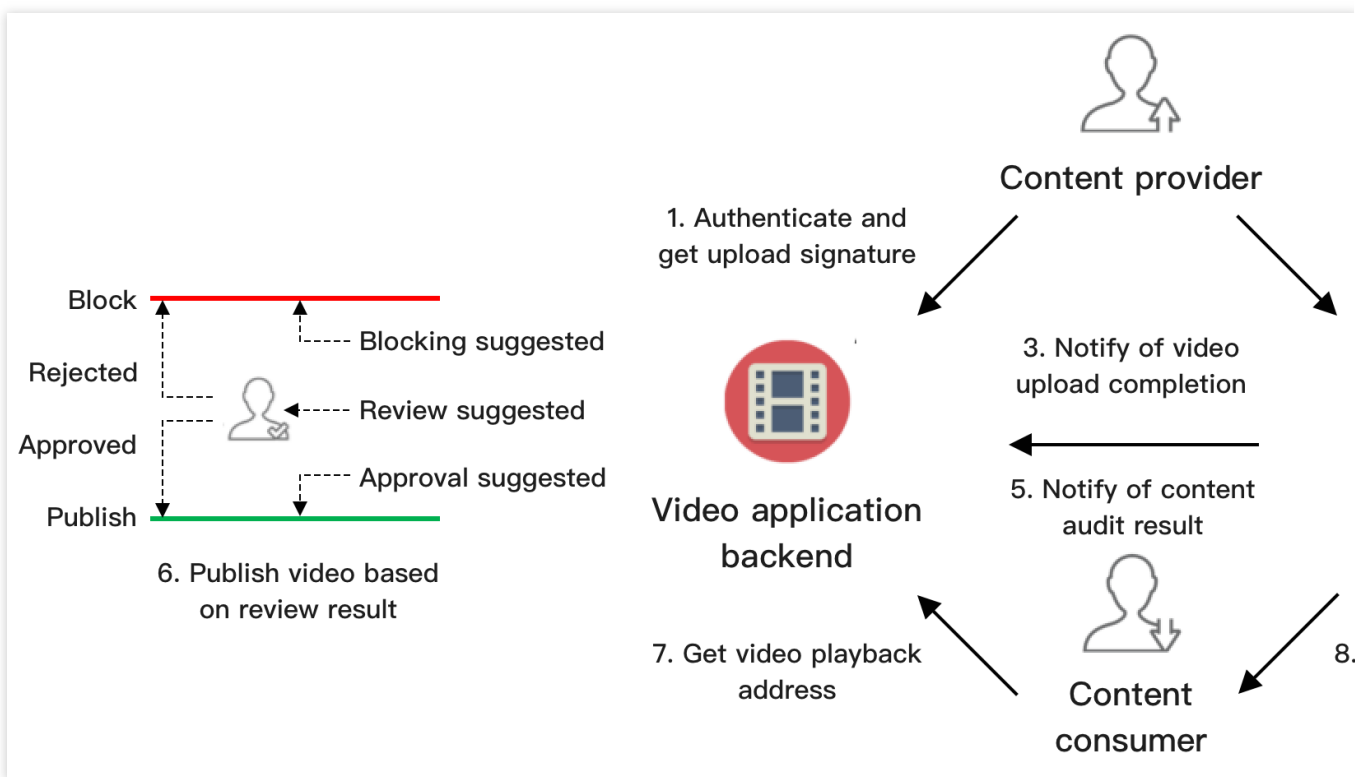
云点播提供的 [音视频内容审核](#) 功能，可对音视频内容发起审核，并在识别结果中给出审核建议（建议复核和建议通过）。开发者可根据结果对音视频内容进行处理。

有两种方式发起音视频内容审核：

1. 通过云点播 [控制台](#) 操作。
2. 通过服务端 API [音视频内容审核](#) 调用。

### 服务端 API 音视频内容审核调用

示例，某视频 App 接入云点播音视频审核服务，服务端 API 调用流程如下：



1. App 后台对内容提供方进行鉴权，鉴权通过后派发视频 [客户端上传签名](#)。
2. 内容提供方执行上传，把分享的内容上传到云点播。
3. 云点播将成功上传的视频 FileId 以及播放 URL 等 [相关信息](#) 通知到 App 后台。

4. 云点播执行上传签名时对 `procedure` 参数配置的音视频审核任务( `ReviewAudioVideoTask` )。
5. 云点播通过 [音视频审核完成](#) 通知 App 后台审核结果。  
审核结果为“block”时，嫌疑度很高，建议直接屏蔽。  
审核结果为“pass”时，嫌疑度不高，建议直接通过。  
审核结果为“review”时，嫌疑度较高，建议人工复核。
6. App 后台发布“建议通过”的视频，以及“建议复核”且经人工复核通过的视频。
7. 内容消费方向 App 后台请求已发布视频的播放 URL。
8. 内容消费方通过播放 URL，从云点播加速播放视频。

其中，上述第4步 - 6步的流程可以保证内容消费方在第7步获取到的视频是经审核验证的合规视频。

#### 注意：

此处介绍的流程属于“先审后发”模式（仅发布审核通过的视频）。如有需要，也可采用“先发后审”模式（视频上传完成后即发布，审核发现不适宜内容后再撤下视频）。

## 图片审核

云点播提供的 [图片审核](#) 功能，可以对图片内容发起审核，并在审核结果中给出建议（建议复核和建议通过）。与音视频内容审核不同，图片审核是同步处理过程，发起任务后将立即获取结果。用户侧可以根据建议来决定是否发布图片：

- 审核结果为“block”时，嫌疑度很高，建议直接拦截图片。
- 审核结果为“pass”时，嫌疑度不高，建议直接通过。
- 审核结果为“review”时，嫌疑度较高，建议人工复核。

有两种方式发起图片审核：

1. 通过云点播 [控制台](#) 操作。
2. 通过服务端 API [图片审核](#) 调用。

## 费用相关

费用的具体单价，请参见 [购买指南](#)。

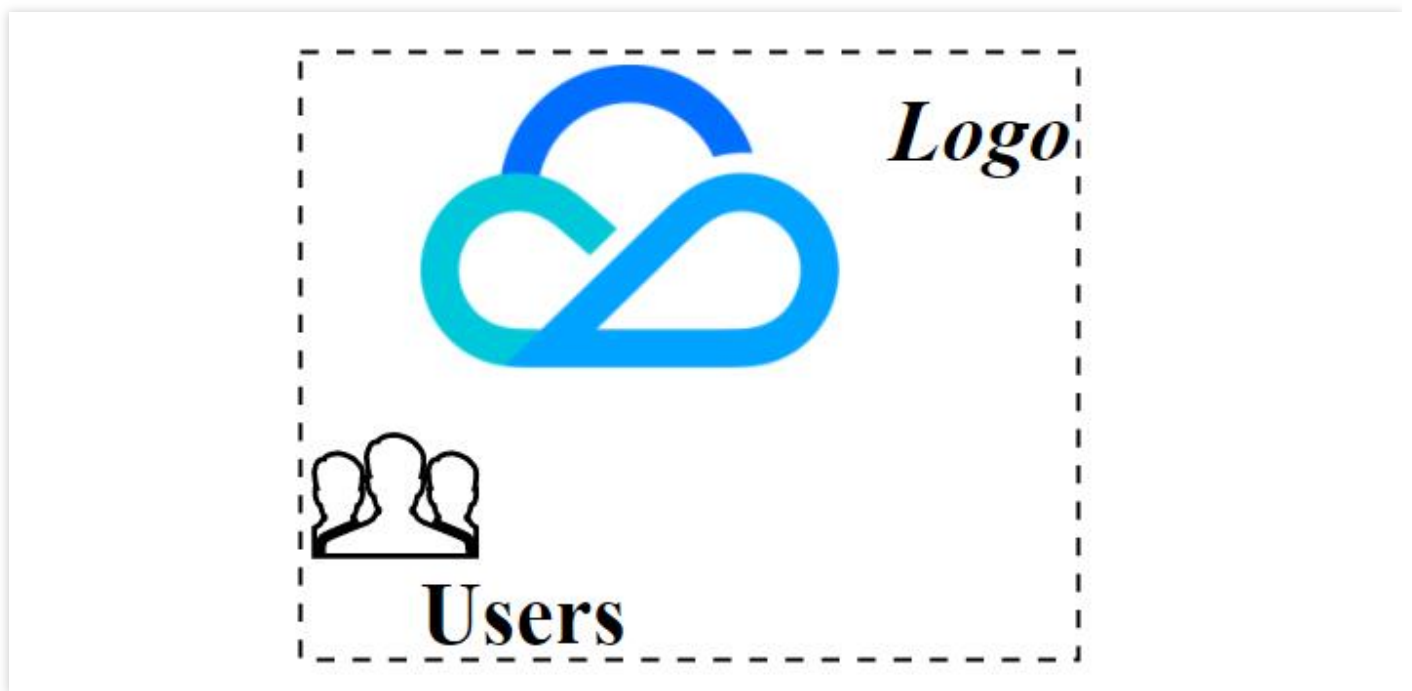
# 如何对视频添加复杂水印

最近更新时间：2021-02-23 14:36:25

云点播 [水印](#) 提供的图片水印和文字水印，可以实现简单水印的需求。但对于复杂水印需求（例如图文混排水印、滤镜等），支持难度大。为此，云点播引入 SVG(Scalable Vector Graphics) 水印，使用者可以方便、灵活地定制水印内容，即可自由排列图文内容，支持绘图、添加滤镜、渐变等特效。

## 介绍 SVG

- **SVG**(Scalable Vector Graphics) 是一种基于 XML 的标记语言，用来描述二维矢量图像的开放网络标准，已经广泛应用于各种网络标准，例如 CSS、DOM、JavaScript 等。
- 云点播不仅对 SVG 宽高没有限制，而且会自动计算出能包含完整元素的最小矩形，并以此作为 SVG 水印原始大小（宽高）。例如，计算最小矩形得出下图虚线包围的部分：



### ① 说明：

- 线上有许多基于 Web 且免费的 SVG 在线调试工具，您可通过此类工具绘制满意的图形，完成后即可导出 XML。
- 对于精细化设置（例如元素对齐属性、字体大小等），可手工修改 XML。确认最终效果，得到 XML 代码。

## 使用 SVG 水印流程

### 步骤1：调试 SVG 水印效果

1. 通过 SVG 调试工具（如简单的[在线 HTML 所见即所得编辑工具](#)或者操作更丰富的[在线 SVG 编辑器](#)），调试满意的图形，操作全程所见即所得。完成后，也可将 SVG 保存为后缀 `.html` 文件后使用浏览器打开查看效果。

#### ① 说明：

若不同元素间有对齐需求，建议灵活运用各元素的对齐类属性，通过改变元素内容验证是否有效（例如图文混排，增减文本长度，查看图片和文字是否始终满足对齐的需求）。完整教程参考[SVG 教程](#)。具体案例可参见本文下面的[示例](#)。

### 步骤2：创建 SVG 水印模板

调用 [创建水印模板](#) 接口，指定水印位置、宽高等信息。创建完成得到水印模板 ID。

### 步骤3：添加 SVG 视频水印

关于如何发起视频处理任务并获取任务结果，请参见 [视频处理任务体系](#)。

以 [视频处理](#) 接口进行说明。在 [示例2 发起带水印的转码任务](#)，`WatermarkSet.N.Definition` 使用步骤2得到的模板 ID，另外，由于我们使用 SVG 水印，所以需要传入步骤1得到的 XML 至参数

`WatermarkSet.N.SvgContent`。

## 示例

下文将通过具体的案例为您介绍如何添加复杂水印，实现文本可变的图文混排的水印。

### 需求描述

某客户需要对视频添加水印，要求如下：

- 水印由品牌 Logo 和登录用户的帐号 ID（文本）组成。
- 水印置于视频右上角，[水印原点](#) 距离视频右边框偏移大小为视频宽度的 2%，距离视频上边框偏移大小为视频高度的 5%。
- 水印宽度为视频短边的 30%，高度按原图等比例缩放。
- 文字向上和向右对齐品牌 Logo。
- 文字字体为黑体，背景颜色为白色，带有高斯模糊阴影，阴影颜色为黑色。品牌 Logo 不缩放时文字大小为 50 像素。

### 步骤1：调试 SVG 水印效果



SVG 的 XML 源码如下：

```
<svg xmlns="http://www.w3.org/2000/svg" xmlns:xlink="http://www.w3.org/1999/xlink" version="1.1" width="1000px" height="1000px">
<defs>
<filter id="filter" x="0" y="0">
<feGaussianBlur stdDeviation="2"/>
<feOffset dx="0" dy="3"/>
</filter>
</defs>

<image id="img_watermark" xlink:href="data:image/png;base64,iVBORw0KGgoAAAANSUHEU
gAAAQAAAAC5CAYAAADHwOFvAAAAAXNSR0IArs4c6QAAARnQU1BAACxjwv8YQUAAAAJcEhZcwAADsMAAA
7DAcdvqGQAACHaSURBVHhe7Z0JmF1FlccLZ2QQEEQWZVUwI4oDqAPIOCOOiKg4IyOLyoCiomFJv+4EQUD
RZmAGcQnIGFDW5N33OgkRCAQIm6GBJH1fd5oQsUnS796XPYQsJGSBJCR97vxP3YJJ0pWk03239975f9/v
u90S+9atq1N7naNENaDmYHc1hPZXQ4Oj1FA6VTXSRaAGlaOR4An8PA0sBCtVYxBsHyL8+6V4VvAs4TkBz
zvw/Dm4AH/7BPx+mGoK3qfODf7GvF0kEiWqS4P9YPAnwBC/B408DUwCc2GkG+2GHSX0Bt41EzyG32/C8x
xwjBoc7G1SJxKJi1Vz8Lcw+E/D+JrABOBrQ7QaaArkaBXoQppGgu+jYfqwSblIJNplNQfvUpcFe6sm+ir
4PYyrAsNabzW+zIFpRCOtW8+deP4XOE5dGOxhvkwkEm1XP6D3wti/AK05Hc9FvY2raulUueA6NGqfVOcG
u5uvFY1ESgW7qWE0CEb/UzAdhr/JYkA1Ao8MaJJq0GsXB5oMEInqVI10IhgNVgIMnW1GU4PkkqAfMxzF/B
s/DTG6IRHWGc2gvVPozUflbtSHYDKSuoLeAg7z4lBocvNvkkkhUY2o09lANdBYq+mQxfBu0Aflyv97e5E
VQkahmFB7OeTrs7WyVX9iC11UuGIWG4Kmm90SiKlQQ7KYrcY7GwvATOKBTa9Ba5N31mBYcYHJUJKoSDaP
3Y7h/LSrwKnlFvpMjspoSP8Tz78zuSsSZVTc6zcF/4qK29mrIgv9R2+N0kOqIfiYyWmRKG06kt6LSnoL
KiufgrNXZGFg5OgV1UQXy26BKDviXj9HJ8P4Z1krrRAD9LC6nA43JSASpSte2svR5WCNvaIK8UG+Gkpfk
2vJonTER11zwTg8a/jobsbhhpfvGLA/BJEoMQ2hY1H5/mqtLEKY6ENVNAFlsr8pHZEOjvFws5G+AZZaK6
OQIjRDNdDx+nKVSBS5+HhqAzWioq21V0AhfbRXpC9KIyCKVmHP/98y368KXldN9HVTciLRAMW+7hppRDj
XtFY4IWvk9NhrwJDIBqYcrQPjN8Bm3tVMiHj0Bv60JA0AqJ+if3YNQXjUJGq1FEH+xCkbsC3EO8CPIUZ
ohrou/j5DAv/hv/+Q3AFfh6OJ19imgIWg+rMgxy9iedgU6IiUR/FF09y9MdeFSqTaCec7Kq7DPJgCHq+z
+gLScPoPYo9CvdHfMKRj9zyFGgYHYq/y05M/gvPJ8By/Fwlv5uRTm7YxMeAqE8KjZ898WZ4zq+NfiHSOB
49+sVqKH3CpD4ZcaPQEJyGd9+IdEwDGT8JidFQA10gjYBox+IK0kTXosJktXd7HUb/INL4Ld0rZ0HcYPL
+eyM1g+kgm+sl+mo2pjk10XbVSBeCjDnv0L39HFTgn+mAG/0d0ichjmUQej+6D2ywf0+a0DJwnEmtSLSF
hmJIm6Vo06GzZdb09ufpXrbaxI1VeD36Ffv3pUSOPHUZfcikUiSCLsMcWsfWs1SYxOEhNLWds2vi3vuld
LhpCLJzfLqJnlJXBfuaFIrqWhzpnkfpWStK0rCv/DDg5z4mdbWhcG3laHzX3WgM0l9f4QXeBjRK4likzs
UVs5G3+9Le5+b7BTRcXR3sZ1JWm+JDOU10Cr61XRuhNS+SQk+xLjQpE9WlhtL59sqRIBwnYBidbFUH+L
zCY3B1TDA5b3yI0ly+t7A0SZVoroSx7RPdYFKLzjeph0J1qs4G1COOnrnTaI8W9dlUJfigy5mmipDMnA
c/0GcWelxScWOQR6WmsDoUORZpMaUe1LO/G8yloZYocI7/4zkICYWypcJDwP+bPCnm8xE/p0/BeTGLFNi
4f+qVQ0vb13J95fWYv8UYrvMORotj3/4oZK+iCTqIYVrvo/Ya8AcaJXnJv14pdoxpGg9AIPLMuwDchRT
WsRh1iKlmvPm87p+DGR9Q3cTzARnq8V17GDd8X4AZIVIO6LPggKlWywTty9IZqoItMCKs7osHBviiVJ63
5GifsByHL9y1E/ZCO3hNchwqV3IEfHSIMxi8r/f2Xjr2Q9G6N9q1wukmBqCbUQEeiYJfZCzwGwmnG1TLs
j0BhI9DaK4/jJEeTZL2mZqS3/e61FnQs6Ou7N5qXi6LQFZi+5ehFe37HRAOdZ94uqmoNoY/CINdbCzkW6
D70/BKqKmpxaPBGWmLP8xjIOV+kHGtB2iGmpYDjoVMNpYPNm0VRK0f/jvJMzmdDjr5p3iyqSoXn/VdZCz
dy0DtzzEBRfApDsV+DvE7I7RhNlyvDVStUFR7zbS3YqKFNqJjnmxeL4hSHZG+kB+zLEDf8hqMh+A/zZlF
Vic/b52iRtWCjKjctsuKfoC7XXoYWWssianL0mIwCqlFNNNRaoFHDN/vkck/yte5yUwFaj0aSp82bxVV
hX5Me6HgZtgLNEowRJSFonTEvbIO22Yrl6ihETLCqybl6Asg/1N/HJijGj321orCLd7F1rKJkhzeMYT2N
28VZV65YJS1IKPlddUUfNK8UZSKgt3QAPwEBhq/f0H2VyCqAnFL3UgrYUYKXSLeaMoTbE7r0aaaS+jCG
FX4qIqUCNdEH+PQCvwjgPNG0Vpky4115004Dva/AZEd4p4sCquhGhafh5axrpJTxn4/m4eZso+J5ua3
```

AoyRHN+jhpygb4nWYxu3dFdDOWHz8m6fAH8BQcKZqoJPQq39cBy65NNgvdAxqKVM+d8BXk+VyUBWoMTgChRuvp98cvSqhpTKoRvo2YLdrG1BGJfAr/Hy2vgnKwV9kL78OxAUe9/CfTxdK75898TYdh3hjYxfVo/SK8D1Wo40KDiTBvuxF1LHGxC1/A71sNdZIoCfEw491LEUNo2NhoPEF1+CpBR8/FY1EGVSOLrUabmTQMn3EWCQSZU16/1+0G25E50h08zKRSJQpsRvnuCPKsDcakUiUQf2YDrIabVTw3r9sL41EGRX7cbcZblTk6EH1eQkWIRJlUw3B1VbdJYoc5cybRCJRZtTa+reqq2tvddWmMerKTT2qef0aNXztRvXHLYG6d8Vm1bJ0tRr76ho1fvHqrbj/lbX4b2tUftkGdfdrGBP1dY+64c116ppNG9QVmwPV1LPFaULagOc/mjeKqk7Bbuo+eo+6i96vHPoHlaezVIEuBzeDPJikivQ8/tts/Ny9FQ49h+dToIj/H//7K/G/nQ2Ox+/7qzucPbWjUlFCasUwvLNyhCqVT1dt5Wbleo+C2er5OZuU6weqFAHPziH1yML1quVVUr9bsx4Nw2JVXPrPamJZHH9Ui9jYR6HRbqFLYbT3gmlglSqqQY8MWOOG4EVQQGMwFL+fQfQCA6RBIFqdne9W7f4Jyi1fBQnthcG/gmfPVkYbJ2HDSlKVvBJ+/h80QP+i2hbIzbCsqUAHwxjPxzOPJ/fm6+2GGxMObcQ7PTwFABeBQdIY9FdB8C41ZdbRMPbrYHwz8XxjK6NMFw89KCNNv1Jt/rFq3Dg5GpyWHNoLRvcNwEb3Gp49VuNMGodIOcHrmFq0Ik0Xy8igr2qdfQCG9ufD0FrBpq0NL5NsBlPVVO87qn2m+IxLQuOCv4FRHQ1DuwmGtriX8WUSTBcK5KhR9AU1IdjtF1l1i1vGUvkw9Kg/A93bGFj14PqzVXvlatXhHW6+TBSluJ4U6GQYfQHPiOfzCeHQZvA80n+euiPY13xZHWvys/updgy1S/4iGD/1MqqqA9/g+gv1AmXrXdk4FJXy9CkYzRiAntRiWNUGNwQF+gv4IUY0dRiAdPJ8GH71chjMqzCcGjD8beDGzPXQEPiXqCmz3mu+WrrLQo8/JvgwjOREGMybVkoQeqgH3/YCnl+vj4aAV/Td7rNg+H+xGk5t0q7aKqfplUxR3zQOw+Nwa22Z3XBqjHBEMEGNrmVnM1PKH4Hhf9AzvmUxklpnHb77D6qt+1CTGyKbwnn+52AQk3XvaDOWWibcyfgZnjV07Zy39Nr9/8TceLHFMOqHcFowDz9/TbaELBp070HFb4YBvGE1jrqBGz6aqtC9q15T/YPQ69/byxjqGm8zuFU9X5a4AixuDEFrCtd+KXaDqFdoHfJkiJpYraHn2ionocebhpfe4t8A0WPBvw2NWXCSa361cFugAsshtBnRNUgxbUyOCDJreqRG3eN1HJV1grv7Aly9BQnm9yrb7Eq9754CYY/wZr5Re2pFO1BB8zOZdhjevaHfP9q2D89bjQ1z9cn/NquGpdurfJxdrXaDoEhj8BvRtZKrtghZaDL2Z3/agLFdgt/x6VmY/H2iu7sh1c/3HV2XWEyc3aFV/J5Vt01kou7BC+X+DQd/Vx6Eypc/Geqs0fGy5wWSq30Ddc72X1gl+7vgccOhXMs1ZuoY/QG2oUXZydkQAbf4mNXxb7IsH1lmIa9Y2a2irksF1F+haM/zV7pRZ2Cb56zFeOU68jfmY15N/XqxILA8P1N2I6dbXqqoHjoeyxmT3w6Pvylsos9A/OzyJdkt50oHXuHhjy3waSc85RVyBfXX+EerpSvbfg2BWXQ9ejaai/U32JwIem6Lx0RgIl/xdA5vzxgkbae0QfQKo2sfEX6HY0AJvtlVeIBIdWq5H0RZPrCYhbm3b/Q1TOanDYURu43l9VqXyMKYHsq0D7qDyNR+WUbb5E4EtTlFD96JjzeVTIldaKKsSH681X7XNOMaWQXTl0EJhor6hCbDhBp7qLPmBKISZ1LDoc89KXRu0K7h6TWIe0vkcnmMxWrkZP1+LHnTo/+NdH65f+A+D6WANsP+9TOEt1acsm5vfZUokW+Ijqw49Y62gmULfw18I+NzhEfwWXAn4CnKIq/8DbsPPPJLPAMvtfytLUF7dGtdAb7LX/IesFfMFNE6bxZqr1yi+rwz1LT5n9Ee+NhN947Wxxhh55PzthLz7Fd719Vu3cdDMzF392Av5nNbU12kOqWm/Bt2WoE7qMjUQFLIIPdfn3DbhmMedSep0BD9TE1PkAd6YoX8BYm+//nOAB8aYmdfIAnxhL8vWytb4R+BS7TzlQiV8lrsFbINHDZQP1OpOmnMNpRL4KOrn7KIwcGvH3+TLTm1u9Oyu43q9VW1s23JOz8Ts0y1opU8UYfQudoY04SvFdhTB3we/wrICM7HTQmugdiZw/+zgyXRYu92Co7o1RbuVLqpXAOA691emfileB3umv3SYtabMZacprv4ppihefHPZvZ6uMKcGndfP0CzyT8c3fEuynish3MA9/Fvnx1jvNiYKpTWRORTgIRqky0VIBk8P1eMehonrnfYIV11ocgmXq+VMYFYxDWrK09cnbHE+qztkHmJQmq9BZp2evhG1AK8DPwcEmhcmKtZ7z9GUYX5s9fQnhYDRSpJ+YVA1EaD3b/ItRyDI57MPG5nqTYHgn6LlY2uJ5d6nyOYxCJqeWJ1a8l1Rp3pEm1cmoQJ9GRVtorYBJw05D83QHDPDQeOa/uyieHnCUiod85FM6ayIOpj9FOtakqJ9iH3a8om6tdLGzDAzOZEw+TpPrXY4hOIcPs6U9DTxVKp+M1MVvADzHzITxs3HRdKTlSyZl2RjvirbpNjzT8nkwTjdG/RbfU7dXtvhgblt5VY1pfuTjHuzFXqaaF6xSO/TSHdWRgOL1VTvy7HOe/loehZu9Dm0CwKzoY0sy+KR4G+mU6e8XoEnW5Ssovq6D4KFWrdNhUsbnjI/wfVWUXn33kbkWMY8gUe+zclCy9WtnnnxtIIItASnoUIttVe2JOHIQDCqqnG1jrLgcGaprA3QtF2vC3zQhBfdbBUsNrw39bc6WlXq/hqMb4H92xKGz0a0z8mprq7obhPei2E2O6SwVrIEcfsQ/3iTquoShwMr0FikP8F1AbwrT2eZFPrrbd6nUYlWwytXHLjeOrzzh5lY6BuI2ud+Et/T2ev70mEjPgNXmJT1X9x7F0mrqEgpb+nQFflxPA8zKatOjQv2RgPwR5DgISKMPYGe5gU7ERc4HwN1V6pYoDDbVcuqpkw261dH1Tt/qP2b02c002q+inUhrb6NirrsnvFShCHWlCR9zEJq25xDIQI3aMbNdu3Rg69hfW727x9Jwoj+Cy1VKbo0dt85UtjXbRKQ3xCz/V/r4fitu+Om9Dt+ADPB6BMwq2sLOPy6cW+m6vXP/52FF6XHmP/5jigp/o2CtAXZyyVKmrCSDk3m7fWnjodVjtxhZ7e2L4/VrwxqvXF/ken5gaZHU0UUjZ+vX1GP43vgkvK4hFNgdqt3x45KEuHPmvevB3x0VfXm22vVBhjeo/WfORcNqRS+Rw0qsmtP/AnyIhkK7uYcoLBqCyr7RUPiULjb9LpQWXl6eP4zoQC09DtOx5t1+acjgqUwC04b56eatSLpnafggZvvj0vIoJHVNzzD8SFGBsBR+VN7+CKQbu6urDqF4X7JBhk6DA1gTynpSjjHdQP7drbUrmiZaNqn3OueWP9qGPO0TDSdkt+R

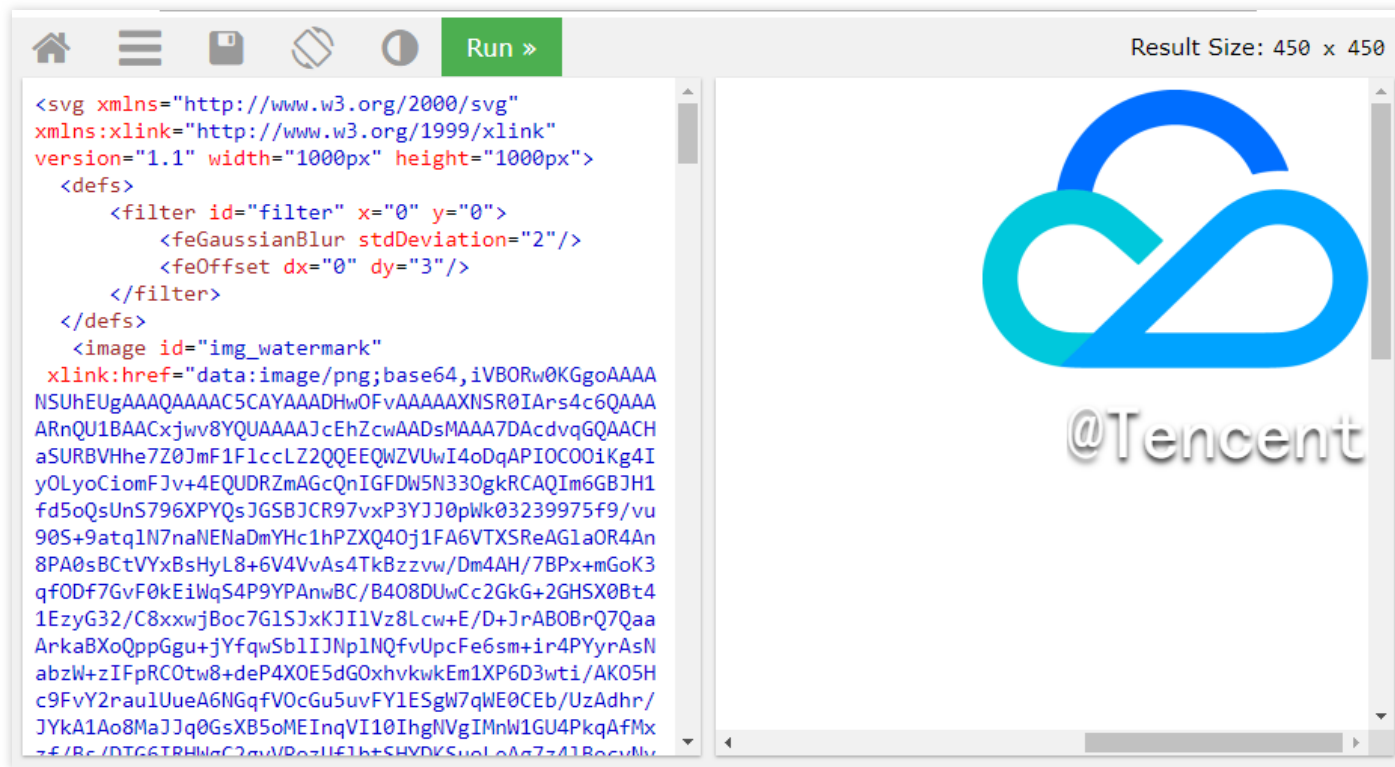
IC+ulxQM5b0/wIIG394Dz7dW218hJXj5bNh1Iu4oQt9D8S/KMjrOlZnNr8fhqrLe1euiHH9P9X8sG570kerI75YpXt+/041Y0b/jf+O4N3o+dP33MtON3jLsR7FF5gKNNeaL1Hi0MPmjduoVMFcNeabbq73umqvfnS8sT7Fzkrcc75MfDjw+Fq///qG5v9FRs/e79J00afox1s7GSRqsbl0PetErMptArvsRyfdv17rJUswN5Zc1t+/RGf0ct5NyA/1m+TP31Hb6H6Nw/oshQfP3S3Qb1Ah0rmx9jb6nDydaFJVv+JtOu1VyZJHkYEp3mj6pnmjEZ+9j3/1f5GaXAx7/qg8Lj1xWgIdt3j0NtbqLzV2F/x1lqBbrRXkiShv6IB+AeTKLGBzG7HkVFTTSvM1omo7l3//eqC+wCyvp/XvL9c9C3vTdy7I+54+efyBRg9hbJEPDQcwVbWdQGwkyZFIlyOVHhTvs+RUV1KVG4D3vy01utFa2qHC9VarD11beJm4U07zP6q1RW95tzUbV5v10QD0/DzPzLEclSNmHHc0ER5tUibZUni5BIxDftMyhNdrBqZY+rOKNsVS2KNnOyqPoHb3gDUJD+aIl70LC68Y/QXn1f2+cHWM6VLBWiiQJ3XGn47arGhTuCMR70KhI3wlfVirvgwbgpV4VLip4vsq+7EU7194m9P7cOw+16/NrBnQwhoewec9Uo/WQ1PvAdrEpEq0PRXpXnv+RQXdGr7ohVmHoJLFRhjb0f+Y1SnYsP0COyCh/fGH/lmgE5vXgZvA+G/yBI14+9w+fepefvk4p0JvIqvsY6T5PQoeweHlPtbbQR4j1mPknUV83V7siHoxF4BfxoQFel2QmFQ3+2VoKk0KMOFrvTgyZVop1pgp6urbHmZxSErtP3x7CzfElvo40Q1280nyTaFFfcv30mCmgA4gLORJw+Gq9HIaJdk0NP2vMzCugtNSb4sFLtLV9ZDTcq2KW3KHmFcf0eBwKf8nkUla02HHkkrSJda83TyKDP8RHg+6yGGwWut1Q9Xz7QfI4oKd1Nh8DwOKBlesYfDvsnSM8/ADn0FWveRoVD3+UGYKrVeKPA9d0BnVMX7bpG0aEo2BesBZ4YfMaA7la3Ss8/IPEQPd6F26vRAPiVXoYbHWP19F+CKtKHUGFcS0EnjI7SIW3/QHUXvR95ucCexxFQpFvQAHjxBf10/d+YTxHFrdDvfMo9P40eP+pIvPWq8OBWjMeCaQyPAOzGGWu92PzKaI4FRp//HfJd4h23nm79PwRKryqHedOwP3xNgAl77vmU0RxiW/SFajbXsAJwfnUH65XA4pDJ7LKoT9Z8zwan017BPAt8xmiOMQXOor0sqVgE4R6UEn/u3rCdFWZ4mwAeL3IarhRIQ1AfMrTP8H4EvImuz30sP8G6fljldQAOl5i111lph+rigCEOXSs9f8yq7gZgzo/MZ4iiUpFOQcG1G5ufnYc6NES2eGMW5y9f4rKVQRTwAqPVCpDU8Z8iigKcaz3LBh/gXL14bM/ZbHbNodareUQBUMLMcki+zGGWft3u3mU0QDEnqCUfR1GN46a0Emhg7YcZkYf0J6iN6LBqDLXhZRwL4BS/7MXoYbFW3eY3UbAyAy8TAw+A8UWLqx+TlUmBN8T4b9CeoOhBGGt9aT5F+rZTrTbIabzS8JM5ABiLd838D1WCFtQCTwqG1SIN4dUpaHDcwzgtdefoxe6XlIBU2440A7031wnxx/9QfcU9bpItgeBh2WwovKXSQUBi/9PzJK0/fsZZJVLQE53ID8H078UbeFP8M8zmiVoqdgYTgn7b7hUYgZxjUiVKWnkaYS+XiGgJ/pHXAM7rZbRR4vo3ms8R9UW8ZsLhudPv+ZeDU02qRgmI6bYyiYS9ILuYUp1zP2U1XCjwvWmm88R7Uxh1NgGIMZf7xpDh8NA37KWTxQ4NB0dzb6YAlQ+gLn6UqvxRoK3WT378t+bzxJtT806Tt/PUTBpu/B6BcP+U0yqRGnJoUZr+USFQ4+EO3Tssc14/MKpKnIgaAdic/S87Ha90f888X4MyAeCRZjPAAUcp15G+T6I+yGGxE8DRhIFntaFp+lZ+NPPzb/XHC8SZUoTXE56N0XSz1FAY8yOe7AO3IRf1gNNzo2qPY5XzZvE70tHoL16VYUSNo9vy/GnxXxwS/6tb2cIiJc49kiKGtp3jHopVdZDDdCKg+qzs7+B7WsnbHnHidusxZQkjHUVmM1PHdmx5deSpmK6uo4AhN7G3oHZUxPC950+yGGxGu94aa6n3WvLG+xcZfoBEoiLRDdf1FjaZPmFSJUhd6/wJdaS2rSKFfmRduoVLLjQvhRkmb94Bqba3v++Mcnpt7/rSNv0AzkiYjTapEWZBDB4FX7OUVfQW3vEl88Yt1FH+vApDUNuNNxI8Uu1zv2LeWH/iCL0OPYRCSHerr0DT9D6zKEPSc/9b7OUVJbRIDbc5bu3q2h1GGmeMgLD5UXVW9jVvrR+N1xF6/wRS3ucPOLU+OMKkSpQvtDBJMM74r3sX6Y/mjRaV/N9uY6xx0IORwC/r6nIjx8YrBOPw70lVIIlCkyQ2fwYVRnCbC+ziGmhZ5u3WjTNPxbGCQO1Gm50uP5q1dZ9mnlrbas12A+F+2S6PT/e7dDT+n65KFsKj39fn0jn4NDsHcdt4G26Nv8Zq9FGjet7anL3UebNtalxOkLvc9bCSBKHNla6Fwrwoc8rTmTD+9dZyixz6mXnrDsTBPfYpREYbPU+rJ2fsZd5cWyrQwWASSpts/0QdY06UPYUOP16111lvk0AoM//twJ2eqf5Dune0GGwPlu1RXjfmV596Wo65YCyJB8jQGo5D6W3CtBvFaDG/F2sotFsJRfib6JNe/KrFRgH5PeXjn3BXgvfUCzbQXQkLo9QYUuE01ObqqdoWn/aZayy402KXbKDrJvL0PaltwKBqBBVaDjYdNYLjqXfzdUWXz9FFk9ovWQkgShwpi/BkVG79DU6z1Fhd89XeXPTmX/OYtDDR+9EjAK6hZy6vTiehdNAit+nRrASRF2PPfoQ8cibKne+gYlE+Cw37AF804jNwuiw/run5iq7HGRTjtmKqmv0YUla95wQKKFj2sGIrgKTQR4vpZn3UWJQtce9b0HEdKo/16NB9+pRhv1TyBye2FrAV3qvgO6pzyxtLGVWeTkAmL7BmfMLoCL2/3fqGlygTCs+BDEcZxefe7vQKnCMSuk/xD79uE2GmnMuN5m1V4Zrdy5HzepyZ4KdCLw7ZmfNnr4bxHjz5jYxVueztBHR63llgAO/XLX5/7byi1/Ccb4htVIk8D1l+P5S9U+M1sHWdiBrOHmWjM+KcL53S/URBJvS1nSaDoOdeN+lM+b1nJLBJqFdHzApGgA4r3DkndrO1OBLXC9lardv1G9sGBQuiGp9a2tU8Fye8YnhEMbYPxXDryFF0UiPmJbpH9GuYwFaV/1Xo80/LtJWQSaXj4QBviy1TCTZ40qVR5QHZZwMUU5JN1LRdr4vWLSjc2v55Ni/KlLx3CgQWAIMBKwi628Esahu6KvG3x5h3372Y0yDXr0WQXXv8Nwg/0WkeQcyBSdqRQSOro5vZAz+/QVbryiZIXn6/gnr5IP8HPz6A80BmkfnX7Sxzt3/Egk9oIxVOBnu+G1KcCNjhnVe7h+gvx+8OYstyA53lqWuUkPUroWrq3PmTUOncPfeKQ4YtP3GBsi01hnL6vImNXWjM9MWgt0vB9Mf5+isuR8643u+tY/AwP5ScEe5rQ3EeqUFQFTLV+hLy/G

```

b+34rkUbLCXT9rQOqTR4u0nKs1YsheMbMJWxlcnuN46sAT8Fb9P17SVW9FQTNyGx/BvRuEb79mKp5c4aA
DSjdbDsOvuIt0j9JMCjUEeTrTg4r9NN3hgCf63lOfxu0joWfpq3cjFqtK8I2FAM98xrlrn2QU9auzG1B1
5CMJOcKgluUNgUysnwjjW9DKWwu05+T1qzFti/EK24bMGid/67JjzdT20thlOLfDMoh41epMYv5Bt2MtP
S5CCY53m5ncpt3wpjOXNXsZT7UxavFmN3mzPcEHIDDRHjaSTjUWmIPbx7/pNMJqY3YkniDb+TzbMfoQM4
dBrO3bwmZTC7cEcjIfv9NuNqlp4ZvEm1SI9v5BxnOB1NACnGgvMgHjrwfUbleuttXPW1uFzBE+/ukEVe+
wZLgizAcP+YhZDuPN0oK18EYypunYHXJ/UU0s2quLm7JzmEgQrNBM9/2eMxWVQemHQ/6oqeSutxpZFnl6
Cnl+MX8g4Dk1To+gjxtIyLrf742gIXtBda5vRZQHd8y/dgFbVnuGcKAX0CT8qqhHV5ubNrXwAdcAoGFv2
FgfZ+B9ftlEveqTnFzIMrUIDMKx6/T2Uy38Hg7sYDcHrvYwwLVzwxHJe8BPjF7KLQ7PQAHwu/rP9sQsf0
MYxB/1nweatjDFp3jZ+GfYlWYw9CBXpttqL5sRxcUveEBjiol6GmQRuJVCpvsbzKXvGC0KasEt3hZpAhv
b341BH91EwyDvB2q0MNE608a/cjJZVhv1C9nBoIermsOQv9KQ1PjPANwrbK+OtBhs1j66Unl/IHhyq0A
3qnxwRA3M9fspfbW48gDm56utxjsQ2vwe9cgqdpRgLwBBSBwdvYkDhNwUjdfEWhC3fp07j1dt3q2q5M2L
5PxAW6VHPbRarvMKGUHV588A14ODTc0XbaPdtA8/jgrk+o/AkPt31Zh7fjF+IRPQCnA3hvmn4bmPqeein
YrXCTorR8CYL4FRP6LcMp8l2PmhIhGP16MX0gJhzbqRb0C5fE8W4cGE/ftEait6/0w7jNUqXITpgjP4e
cFGClSffuwbc4m9eBamfMLCaHn82uAB2N/GLCr8M9q78KiGMUt6rR5Byt3zmdUW/eFaAhuwkhhvHpoDXt
3nYtCWG0vMEHoJ+yAw6EyeB51rEUvqVnl6Uz8frwaGbzP1ExRauKfRA7akKf91Wg6BD8PUMpQGufp3/Dz
t0EDuAJcA363DQUU6v1CheEEo7apAxzB90o8UUEcWxh+G7+fjudnwJH4+WA9nK+pOIxK/R+mdM5ObIwe/
wAAAABJRu5ErkJggg=="
x="74.4%" y="0px" height="185px" width="256px"/>
<text id="text_watermark_shadow" text-anchor="end" font-family="SimHei" font-style="Regular" font-size="50px"
x="100%" y="245px" style="opacity:1;" fill="black" filter="url(#filter)">@Tencent
</text>
<text id="text_watermark" text-anchor="end" font-family="SimHei" font-style="Regular" font-size="50px"
x="100%" y="245px" fill="white">@Tencent</text>
</svg>
    
```

验证效果：

- 方式1：将上面这段代码复制到 [在线 HTML 所见即所得编辑工具](#) 页面左侧的文本输入框，然后单击页面上的 **Run** 按钮，效果在页面右侧生成，如下：



- 方式2：将上面这段代码保存至后缀为 `.html` 的文件，在浏览器打开可看到水印效果为图2：白色背景。

图1：透明背景	图2：白色背景	图3：黑色背景

### SVG 说明

参数	说明	属性
<code>&lt;svg&gt;</code>	定义 SVG	<code>width="1000px" height="1000px"</code> ：宽高均为 1000px，这里只要能够包含 SVG 里完整元素即可。

	画布。	
<code>&lt;filter id="filter"&gt;</code>	定义要用到的滤镜。	-
<code>&lt;feGaussianBlur&gt;</code>	高斯模糊滤镜。	<code>stdDeviation="2"</code>
<code>&lt;feOffset&gt;</code>	滤镜元素偏移。	<code>dx="0" dy="3"</code> : 设置滤镜元素「向下」偏移 3 个像素。
<code>&lt;image id="img_watermark"&gt;</code>	品牌 Logo 图片。	<ul style="list-style-type: none"> <li><code>xlink:href="data:image/png;base64,{图片数据的base64}"</code> : 引用本地图片。</li> <li><code>height="185px" width="256px"</code> : 原始图片的宽高。</li> <li><code>x="74.4%" y="0px"</code> : 根据需求, 图片在上面, 所以 y 轴偏移设置为 0px, 通过不断 <b>调整</b> x 值确认效果, 最终确定 <code>x="74.4%"</code> 符合预期。</li> </ul>
<code>&lt;text id="text_watermark_shadow"&gt;</code>	实现文字阴影效果。	<ul style="list-style-type: none"> <li><code>font-family="SimHei" font-style="Regular" font-size="50px"</code> : 设置字符的字体及大小。</li> <li><code>text-anchor="end"</code> : 设置字符的对齐方式, 文本字符串的末尾即当前文本的初始位置。</li> <li><code>x="100%"</code> : 设置元素沿 x 轴偏移至「最右侧」。通过设置 <code>text-anchor</code>、<code>x</code> 属性, 我们将文字的最后一个字符「固定」在 SVG 画布的「最右侧」, 方便调整品牌 Logo 的位置 (标签 <code>&lt;image id="img_watermark"&gt;</code> 的 <code>x</code> 属性)。</li> <li><code>style="opacity:1;"</code> : 设置透明度为不透明。</li> <li><code>fill="black"</code> : 设置填充颜色为黑色。</li> <li><code>filter="url(#filter)"</code> : 使用 <code>id</code> 为 <code>filter</code> 的滤镜。</li> <li><code>y="245px"</code> : 设置元素 y 轴偏移大小, 由于品牌 Logo 的高度为 185 像素, 文字大小为 50 像素 (文字大小近似为高度), 所以 <code>y</code> 设置为不小于 <code>185+50=235</code>, 通过 <b>调整</b>, 我们认为 245 效果合适。</li> </ul>

<code>&lt;text id="text_watermark"&gt;</code>	文字。	属性与 <code>&lt;text id="text_watermark_shadow"&gt;</code> 相比，大部分属性一样。区别在于没有使用滤镜（无 <code>filter</code> 属性），且文字为白色（属性 <code>fill="white"</code> ）。
---	-----	---

- 在最终验证效果时，建议通过增减文本长度（SVG `<text>` 标签的内容）如将 `@Tencent` 替换成 `@Tencent是个善良的人`，确认效果是否仍然满足要求。
- 使用时，对于不同的用户 ID，只需替换 SVG `<text>` 标签的内容即可。

## 步骤2：创建 SVG 水印模板

```
https://vod.tencentcloudapi.com/?Action=CreateWatermarkTemplate
&Type=svg
&Name=测试
&CoordinateOrigin=TopRight
&XPos=2%
&YPos=5%
&SvgTemplate.Width=30S%
&SvgTemplate.Height=0px
&<公共请求参数>
```

参数说明：

参数	说明
Type=svg	创建水印模板类型为 SVG。
Name=测试	模板名称，非必填参数。
CoordinateOrigin=TopRight	水印置于视频右上角。
XPos=2%	<a href="#">水印原点</a> 距离视频右边框偏移大小为视频宽度的 2%。
YPos=5%	<a href="#">水印原点</a> 距离视频上边框偏移大小为视频高度的 5%。
SvgTemplate.Width=30S%	水印宽度为视频短边的 30%。
SvgTemplate.Height=0px	水印高度按原图等比例缩放。

假设创建后得到 SVG 水印模板 ID 为 12345。

## 步骤3：添加 SVG 视频水印

发起视频处理：

```
https://vod.tencentcloudapi.com/?Action=ProcessMedia
&FileId=5285485487985271487
&MediaProcessTask.TranscodeTaskSet.0.Definition=30
&MediaProcessTask.TranscodeTaskSet.0.WatermarkSet.0.Definition=12345
&MediaProcessTask.TranscodeTaskSet.0.WatermarkSet.0.SvgContent={步骤1 的 XML}
&<公共请求参数>
```

## 视频加水印效果



## 附录

### 支持的字体列表

- SimHei,黑体:style=Regular
- Roboto:style=Bold

如有其他字体需求，请联系 [技术支持](#)。



# 分发播放

## 如何使用 Key 防盗链

最近更新时间：2022-12-16 14:19:32

### 使用须知

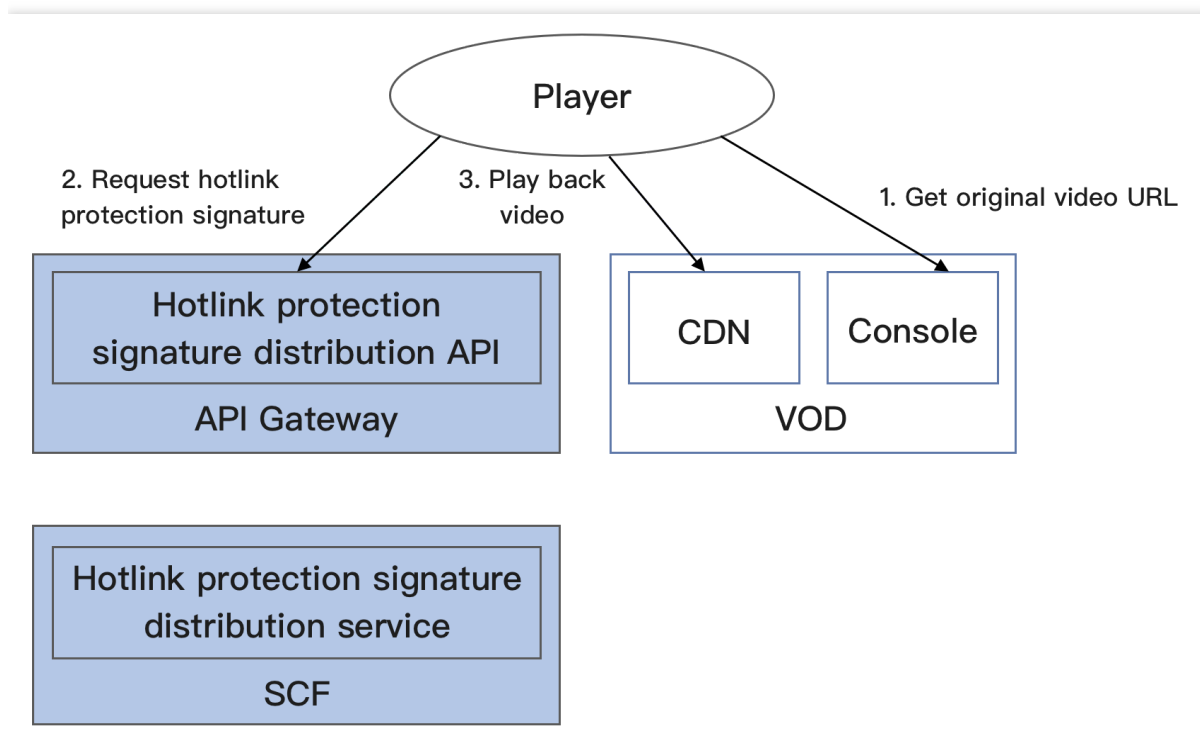
#### Demo 功能介绍

本 Demo 向开发者展示云点播（VOD）[Key 防盗链](#) 机制的使用方法，包括在控制台启用 Key 防盗链、搭建一个防盗链签名派发服务、以及使用防盗链签名播放视频。

#### 架构和流程

Demo 基于云函数（SCF）搭建了一个 HTTP 服务，用于接收来自客户端的获取防盗链签名请求。服务从请求 Body 中获取 VOD 的视频原始 URL，计算防盗链签名，并返回带防盗链签名的 URL 给客户端。

系统主要涉及四个组成部分：开发者、API 网关、云函数和云点播，其中 API 网关和云函数即是本 Demo 的部署对象，如下图所示：



具体业务流程为：

1. 开发者在 VOD 控制台上获取到视频的原始 URL（实际生产环境中，应当是由播放器向业务后台请求视频的 URL，本文为了简化流程由开发者模拟该业务行为）。
2. 开发者使用视频原始 URL 向 SCF 请求防盗链签名。
3. 开发者使用带防盗链签名的视频 URL 请求 VOD CDN，播放视频。

说明：

Demo 中的 SCF 代码使用 Python3.6 进行开发，此外 SCF 还支持 Python2.7、Node.js、Golang、PHP 和 Java 等多种编程语言，开发者可以根据情况自由选择，具体请参考 [SCF 开发指南](#)。

## 费用

本文提供的云点播 Key 防盗链签名派发服务 Demo 是免费开源的，但在搭建和使用的过程中可能会产生以下费用：

- 购买腾讯云云服务器（CVM）用于执行服务部署脚本，详见 [CVM 计费](#)。
- 使用腾讯云云函数（SCF）提供签名派发服务，详见 [SCF 计费](#) 和 [SCF 免费额度](#)。
- 使用腾讯云 API 网关为 SCF 提供外网接口，详见 [API 网关计费](#)。
- 消耗云点播（VOD）存储用于存储上传的视频。

## 快速部署 Key 防盗链签名派发服务

### 步骤1：准备腾讯云 CVM

部署脚本需要运行在一台腾讯云 CVM 上，要求如下：

- 地域：任意。
- 机型：官网最低配置（1核1GB）即可。
- 公网：需要拥有公网 IP，带宽1Mbps或以上。
- 操作系统：官网公共镜像 `Ubuntu Server 16.04.1 LTS 64位` 或 `Ubuntu Server 18.04.1 LTS 64位`。

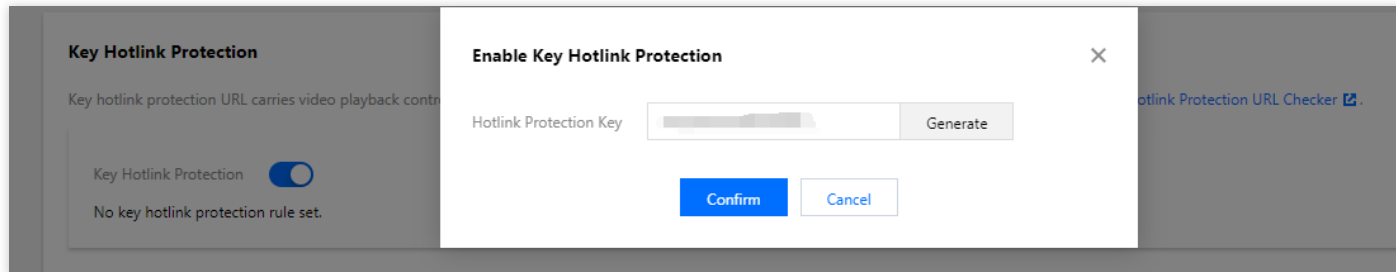
购买 CVM 的方法请参见 [操作指南 - 创建实例](#)。重装系统的方法请参见 [操作指南 - 重装系统](#)。

注意：

- Key 防盗链签名派发服务 Demo 本身并不依赖于 CVM，仅使用 CVM 来执行部署脚本。
- 如果您没有符合上述条件的腾讯云 CVM，也可以在其它带外网的 Linux（如 CentOS、Debian 等）或 Mac 机器上执行部署脚本，但需根据操作系统的区别修改脚本中的个别命令，具体修改方式请开发者自行搜索。

## 步骤2：开通云点播并配置 Key 防盗链

1. 参考 [快速入门 - 步骤1](#) 开通云点播服务。
2. 开通完成后，参考 [设置防盗链](#) 文档启用 Key 防盗链，并记录下防盗链 Key：



注意：

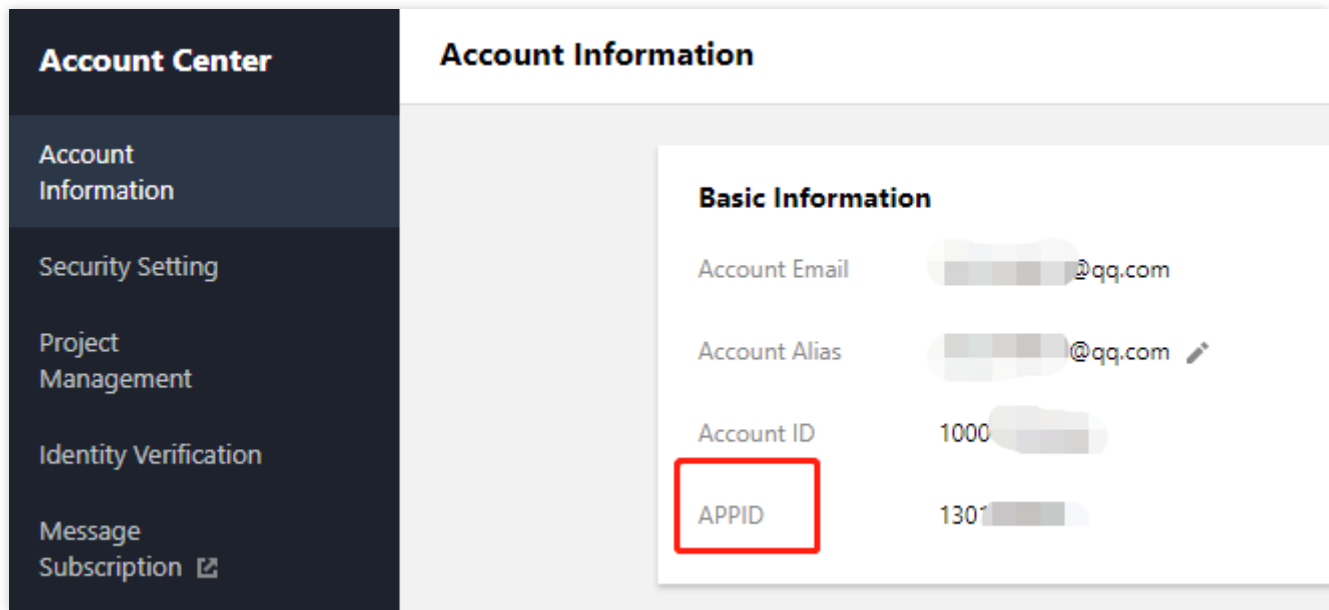
此处是开通 Key 防盗链，而非开通 Referer 防盗链。如果您同步开通了 Referer 防盗链，那么下文的测试方法有可能因为不符合 Referer 防盗链的要求而导致请求失败。

## 步骤3：获取 API 密钥和 APPID

Key 防盗链签名派发服务 Demo 的部署和运行过程需要使用到开发者的 API 密钥（即 SecretId 和 SecretKey）和 APPID。

- 如果还未创建过密钥，请参见 [创建密钥文档](#) 生成新的 API 密钥；如果已创建过密钥，请参见 [查看密钥文档](#) 获取 API 密钥。

- 在控制台 [账号信息](#) 页面可以查看 APPID，如下图所示：



#### 步骤4：部署防盗链签名派发服务

登录 [步骤1](#) 准备的 CVM（登录方法详见 [操作指南 - 登录 Linux](#)），在远程终端输入以下命令并运行：

```
ubuntu@VM-69-2-ubuntu:~$ export SECRET_ID=AKxxxxxxxxxxxxxxxxxxxxxxxxxxxx; export SECRET_KEY=xxxxxxxxxxxxxxxxxxxxxxxxxxxx; export APPID=125xxxxxxx; export ANTI_LEECH_KEY=xxxx; git clone https://github.com/tencentyun/vod-server-demo.git ~/vod-server-demo; bash ~/vod-server-demo/installer/anti_leech_sign_scf_en.sh
```

说明：

请将命令中的 SECRET\_ID、SECRET\_KEY 和 APPID 赋值为 [步骤3](#) 中获取到的内容；将 ANTI\_LEECH\_KEY 赋值为 [步骤2](#) 中获取到的防盗链 Key。

该命令将从 Github 下载 Demo 源码并自动执行安装脚本。安装过程需几分钟（具体取决于 CVM 网络状况），期间远程终端会打印如下示例的信息：

```
[2020-06-04 15:57:10] 开始检查npm。
[2020-06-04 15:57:18] npm 安装成功。
[2020-06-04 15:57:18] 开始安装 ServerLess。
[2020-06-04 15:57:19] serverless 安装成功。
[2020-06-04 15:57:20] 开始部署云点播 key 防盗链签名派发服务。
[2020-06-04 15:57:30] 云点播 key 防盗链签名派发服务部署完成。
```

```
[2020-06-04 15:57:32]服务地址：https://service-xxxxxxxx-125xxxxxxxx.gz.apigw.tencent
tcs.com/release/anti_leech_sign
```

复制输出日志中的签名派发服务地址（示例中的 `https://service-xxxxxxxx-125xxxxxxxx.gz.apigw.tencenttcs.com/release/anti_leech_sign`）。

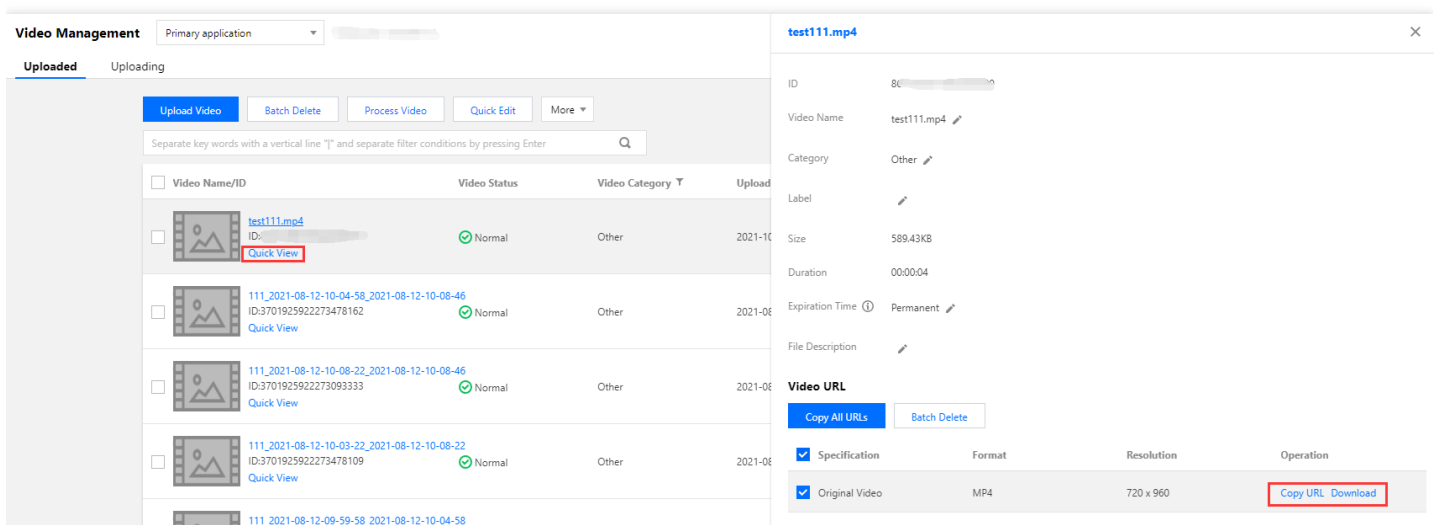
注意：

如果输出日志中出现如下所示的警告，一般是由于 CVM 无法立即解析刚部署好的服务域名，可尝试忽略该警告。

```
> [2020-04-25 17:18:44]警告：Key 防盗链签名派发服务测试不通过。
>
```

### 步骤5：测试 Key 防盗链

按照 [上传视频 - 本地上传步骤](#) 的说明，上传一个测试视频到云点播。上传完成后，单击【快捷查看】，然后单击右侧【复制地址】复制该视频的 URL。



在 CVM 命令行执行 `curl` 命令尝试直接访问该 URL，结果会因不符合 Key 防盗链规则而被服务器拒绝访问，HTTP 返回码为 403（测试时，请将命令中的 URL 替换为实际 URL，下同）：

```
ubuntu@VM-69-2-ubuntu:~$ curl -I "http://125xxxxxxxx.vod2.myqcloud.com/f888c998vod
cq125xxxxxxxx/c849148f528xxxxxxxxxxxxxxxxxxxxxxxx/xxxxxxxx.mp4"
HTTP/1.1 403 Forbidden
Server: NWS_VP
Connection: keep-alive
Date: Thu, 04 Jun 2020 08:27:54 GMT
```

```
Content-Type: text/plain
Content-Length: 14
```

在 CVM 命令行执行 `curl` 命令来请求步骤4中部署的服务，获取带防盗链签名的 URL（`-d` 表示使用 POST 方式发起请求，所带的参数为视频 URL）：

```
ubuntu@VM-69-2-ubuntu:~$ curl -d 'http://125xxxxxxx.vod2.myqcloud.com/f888c998vod
cq125xxxxxxx/c849148f528xxxxxxxxxxxxxxxxxxxx/xxxxxxxxx.mp4' https://service-xxxxxxx
-125xxxxxxx.gz.apigw.tencentcs.com/release/anti_leech_sign; echo
http://125xxxxxxx.vod2.myqcloud.com/f888c998vodcq125xxxxxxx/c849148f528xxxxxxxxxxx
xxxxxx/xxxxxxxxx.mp4?t=5ed8b8d2&exper=0&rlimit=0&us=455041&sign=fe6394007c2e7aef
39fc70a02e897f69
```

再次使用 `curl` 命令访问上一步得到的带防盗链签名的 URL，能够正常访问（HTTP 返回码为200）：

```
ubuntu@VM-69-2-ubuntu:~$ curl -I "http://125xxxxxxx.vod2.myqcloud.com/f888c998vod
cq125xxxxxxx/c849148f528xxxxxxxxxxxxxxxxxxxx/xxxxxxxxx.mp4?t=5ed8b8d2&exper=0&rlimit
=0&us=455041&sign=fe6394007c2e7aef39fc70a02e897f69"
HTTP/1.1 200 OK
Server: tencent-cos
Connection: keep-alive
Date: Thu, 04 Jun 2020 08:37:17 GMT
Last-Modified: Fri, 22 May 2020 15:06:15 GMT
Content-Type: video/mp4
Content-Length: 232952632
Accept-Ranges: bytes
ETag: "1da6be3a0d1da5edae4ff0b1feff02cf-223"
x-cos-hash-crc64ecma: 16209801220610226954
x-cos-request-id: NWVkOGIyYmVfZDUyMzYyNjRfYWMwMF85YjkyNzA=
X-Daa-Tunnel: hop_count=4
X-NWS-LOG-UUID: b404f43e-3c86-4c54-8a78-fb78e4e85cf2 add71e19fb08c6d9dbe1b21a2fb1
57bf
Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: Origin, No-Cache, X-Requested-With, If-Modified-Since,
Pragma, Last-Modified, Cache-Control, Expires, Content-Type, X-Requested-With, Range
Access-Control-Allow-Methods: GET, POST, OPTIONS
Access-Control-Allow-Origin: *
```

说明：

您可以在浏览器中访问带防盗链签名的 URL，通过播放视频的方式来验证防盗链签名。但这种方式对视频格式有要求，一般来说使用 H.264 编码的 MP4 文件具有较好的兼容性，建议选用这类视频；您也可以使用 Postman 等第三方工具来发送 HTTP 请求，具体用法请自行搜索。

## 系统设计说明

### 接口协议

Key 防盗链签名派发云函数通过 API 网关对外提供接口，具体接口协议如下：

服务	云函数名	接口形式	请求内容	返回内容
Key 防盗链签名派发	anti_leech_sign	HTTP POST	视频原始 URL	带防盗链签名 URL

### 签名派发服务代码解读

- `main_handler()` 为入口函数。
- 调用 `parse_conf_file()`，从 `config.json` 文件中读取配置信息。配置项说明如下（详细参数请参见 [Key 防盗链签名参数](#)）：

字段	数据类型	功能
key	String	Key 防盗链密钥
t	Integer	签名有效时间，单位：秒。在处理请求时，该参数与 SCF 服务器的当前时间相加后，才是防盗链参数中的 t
exper	Integer	试看时长
rlimit	Integer	签名最多允许被多少个客户端 IP 访问

- 从请求 Body 中解析出 `Dir` 参数，本地生成 `t` 和 `us` 参数，从配置文件中读取 `exper` 和 `rlimit` 参数：

```
original_url = event["body"]
parse_result = urlparse(original_url)
directory = path.split(parse_result.path)[0] + '/'
# 签名参数
timestamp = int(time.time())
rand = random.randint(0, 999999)
sign_para = {
    "t": hex(timestamp + configuration['t'])[2:],
    "exper": configuration['exper'],
    "rlimit": configuration['rlimit'],
    "us": rand
}
```

- 调用 `generate_sign()` 计算防盗链签名，详细算法请参见 [Key 防盗链签名](#)。
- 生成 `QueryString`，拼接在原始 URL 后组成带防盗链签名的 URL：

```
sign_para["sign"] = signature
query_string = urlencode(sign_para)
new_parse_result = parse_result._replace(query=query_string)
signed_url = urlunparse(new_parse_result)
```

- 返回签名。返回的数据格式及含义请参见 [云函数集成响应](#)。

```
return {
    "isBase64Encoded": False,
    "statusCode": 200,
    "headers": {"Content-Type": "text/plain; charset=utf-8",
                "Access-Control-Allow-Origin": "*",
                "Access-Control-Allow-Methods": "POST,OPTIONS"},
    "body": signed_url
}
```



# 如何防止视频流量和存储空间被盗用

最近更新时间：2021-05-21 11:16:34

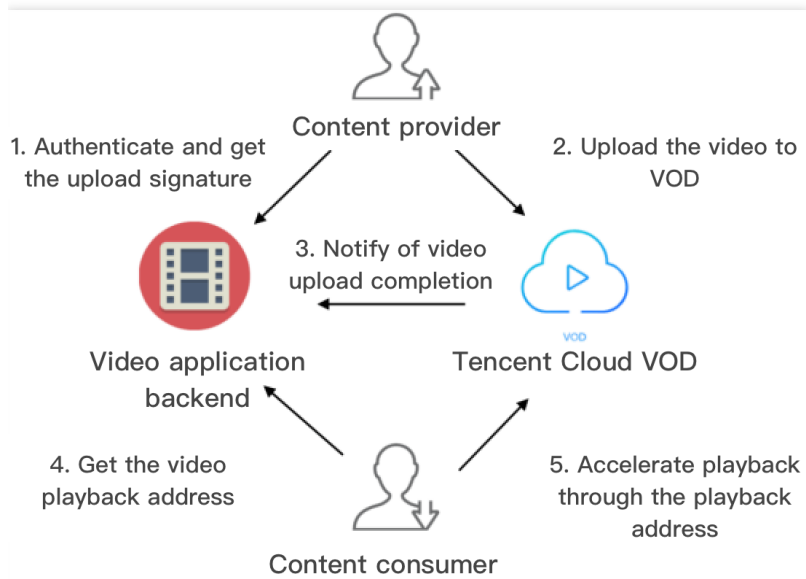
UGC（用户生产内容）和 PGC（专业生产内容）是视频行业中常见的两类场景，可以由视频分享者自由上传和分享视频内容。

然而，第三方的视频平台，可能会冒充 App 的普通用户，上传自有视频，然后将视频的播放 URL 放在自己的平台上播放。这样，他们就能“寄生”于开发者的平台，享受“免费”的视频存储和加速播放。因为开发者的视频平台被当做了他人视频的温床，我们称之为“视频床”。

寄生者产生的所有存储和播放带宽流量上的费用，全部需要 App 开发者来承担，是一项严重的经济损失。

## 视频床问题产生的原因

### UGC 和 PGC 平台的一般交互方式



说明：

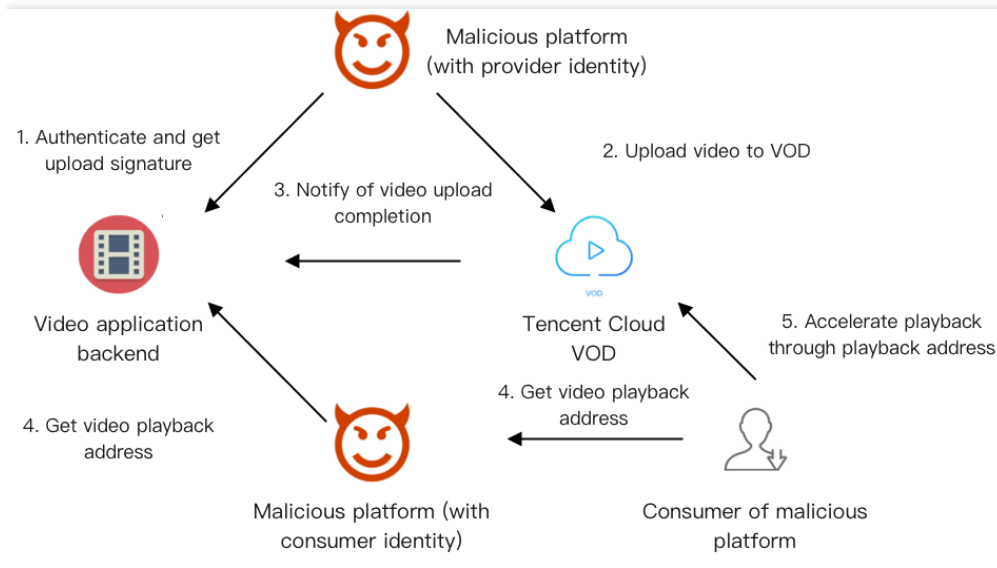
图中箭头方向是指网络请求方向。

UGC（或 PGC）的视频平台，与内容提供方、内容消费方和云点播的交互方式如下（第1 - 3步，可参考 [客户端上传](#)）：

1. App 后台对内容提供方进行鉴权，鉴权通过后派发视频上传签名。

2. 内容提供方执行上传，把分享的内容上传到云点播。
3. 云点播将成功上传的视频 fileId 及播放 URL 等相关信息通知到 App 后台。
4. 内容消费方向 App 后台请求视频的播放 URL。
5. 内容消费方通过播放 URL，从云点播加速播放视频。

### 恶意用户如何实现视频床



说明：

图中箭头方向是指网络请求方向。

恶意的第三方视频平台，会冒充开发者 App 平台的普通用户：

- 首先，以视频提供方的身份，将自有视频上传到云点播中（第1步和第2步）。
- 然后，再以消费者的身份，从 App 平台获取视频的播放地址（第4步）。
- 最后，恶意平台自己的用户，可以获取到这些播放地址（第4步），并通过云点播加速播放这些视频（第5步）。

### 导致问题的核心原因

恶意用户寻找猎物作为视频床的根本目的，是盗用他人的 CDN 带宽资源（附带也占用了存储资源）。恶意用户有机可乘的核心原因在于：

- 第4步：恶意平台能无限制地从 App 快速获取视频的播放 URL，存储并分发给自己的消费用户。
- 第5步：恶意平台的消费用户获取视频的播放 URL 后，能够无限制地加速播放视频。

## 视频床防范方案

面对视频床问题以上的核心原因，关键在于：

- 防止第4步中的**无限制获取视频播放 URL**。
- 防止第5步中的**无限制加速播放视频**。

下面，将分别介绍如何限制视频 URL 的**播放**和**获取**。

### 限制视频 URL 的播放

云点播的 [Key 防盗链](#) 提供了限制 URL 允许播放终端数的能力，防止一个视频 URL 被传播给任意多个客户终端播放。

为了实现对视频播放 URL 的控制，开发者需要在控制台开启防盗链，并且在第4步中，App 后台需要按照 Key 防盗链生成规则（参考“[视频播放地址最多可播放 IP 数](#)”的 [示例](#)）生成防盗链，限制 URL 的有效时间和允许播放的 IP 数。

### 限制视频 URL 的获取

若仅限制视频的加速播放，视频床的防范是不完整的：在第4步中，恶意平台能对同一视频请求大量不同的防盗链 URL，然后为自己平台的用户分发各不相同的 URL，绕开 IP 播放数量的限制。

因此，App 后台需要识别第4步中的用户身份，对同一用户在指定时间内获取某一视频播放 URL 的次数进行频控，防止恶意用户短时间内获取视频的大量播放地址。

# 如何接收事件通知

最近更新时间：2023-05-15 17:15:06

## 使用须知

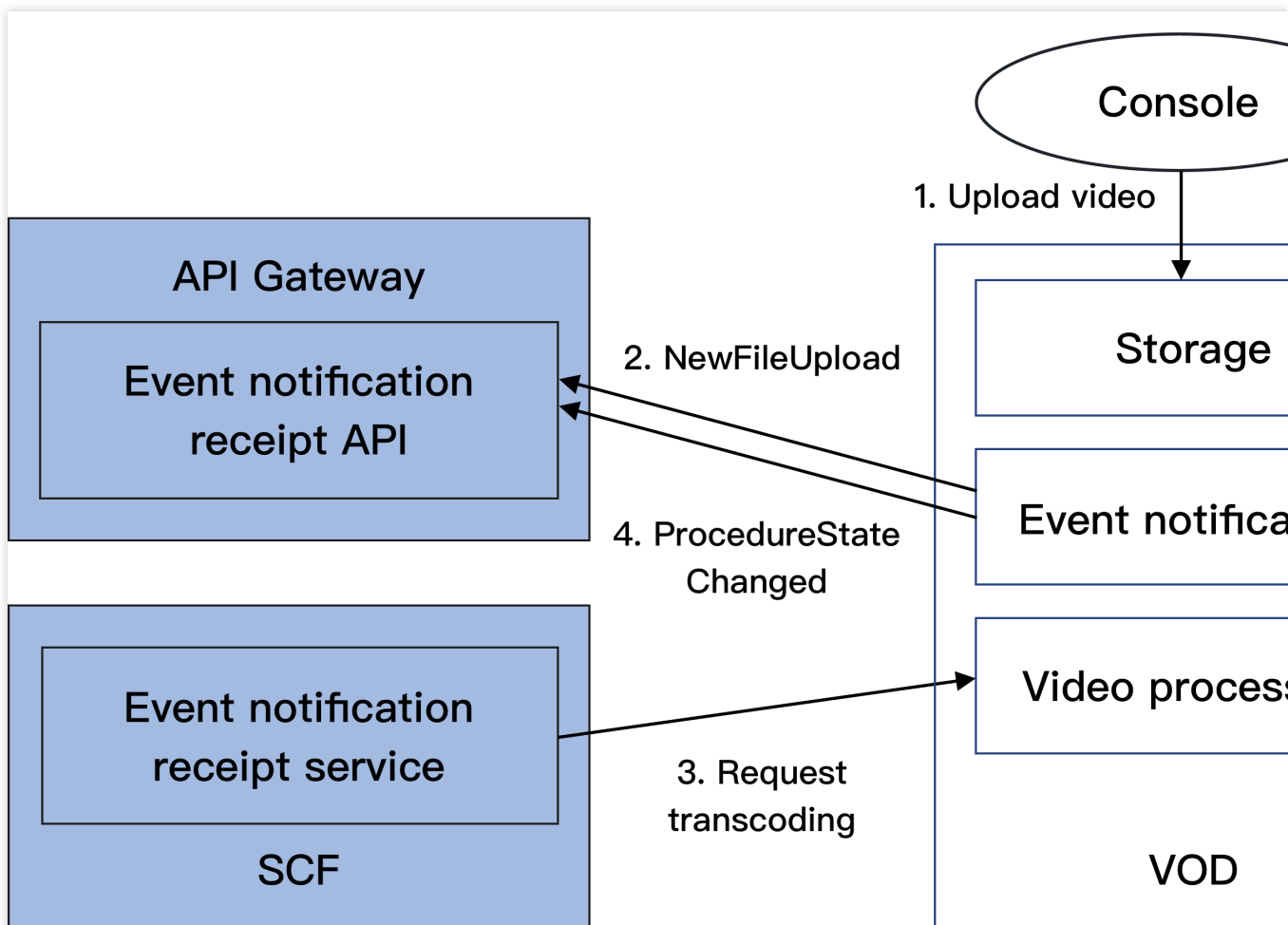
### Demo 功能介绍

本文以一个视频的上传、转码流程为例，向开发者展示云点播（VOD）[事件通知机制](#) 的使用方法。

### 架构和流程

Demo 基于云函数（SCF）搭建了一个 HTTP 服务，用于接收来自 VOD 的事件通知请求。该服务通过对 NewFileUpload（[视频上传完成事件通知](#)）和 ProcedureStateChanged（[任务流状态变更](#)）的处理，实现发起视频转码和获取转码结果。

系统主要涉及四个组成部分：控制台、API 网关、云函数和云点播，其中 API 网关和云函数即是本 Demo 的部署对象，如下图所示：



具体业务流程为：

1. 在控制台上传一个视频到 VOD。
2. VOD 后台发起 NewFileUpload 事件通知请求给 Demo。
3. Demo 解析事件通知内容，调用 VOD 的 [ProcessMedia](#) 接口对刚上传的视频发起转码，使用的转码模板为 [系统预置模板 100010](#)和[100020](#)。
4. VOD 完成转码任务后，发起 ProcedureStateChanged 事件通知请求给 Demo。
5. Demo 解析事件通知内容，将转码输出文件的 URL 打印到 SCF 日志中。

#### 说明：

Demo 中的 SCF 代码使用 Python3.6 进行开发，此外 SCF 还支持 Python2.7、Node.js、Golang、PHP 和 Java 等多种编程语言，开发者可以根据情况自由选择，具体请参考 [SCF 开发指南](#)。

## 费用

本文提供的云点播事件通知接收服务 Demo 是免费开源的，但在搭建和使用的过程中可能会产生以下费用：

购买腾讯云云服务器（CVM）用于执行服务部署脚本，详见 [CVM 计费](#)。

使用 SCF 提供签名派发服务，详见 [SCF 计费](#) 和 [SCF 免费额度](#)。

使用腾讯云 API 网关为 SCF 提供外网接口，详见 [API 网关计费](#)。

消耗 VOD 存储用于存储上传的视频，详见 [存储计费](#) 和 [存储资源包](#)。

消耗 VOD 转码时长用于对视频进行转码，详见 [转码计费](#) 和 [转码资源包](#)。

## 避免影响生产环境

事件通知接收服务 Demo 的业务逻辑使用到 VOD 事件通知机制，因此部署过程中需要开发者配置事件通知地址。如果该账号已有基于 VOD 的生产环境，那么变更事件通知地址可能造成业务异常。**操作前请务必确认不会影响生产环境，如果您无法确定，请更换一个全新账号来部署 Demo。**

# 快速部署事件通知接收服务

## 步骤1：准备腾讯云 CVM

部署脚本需要运行在一台腾讯云 CVM 上，要求如下：

地域：任意。

机型：官网最低配置（1核1GB）即可。

公网：需要拥有公网 IP，带宽1Mbps或以上。

操作系统：官网公共镜像 `Ubuntu Server 16.04.1 LTS 64位` 或 `Ubuntu Server 18.04.1 LTS 64位`。

购买 CVM 的方法请参见 [操作指南 - 创建实例](#)。重装系统的方法请参见 [操作指南 - 重装系统](#)。

#### 注意：

事件通知接收服务 Demo 本身并不依赖于 CVM，仅使用 CVM 来执行部署脚本。

如果您没有符合上述条件的腾讯云 CVM，也可以在其它带外网的 Linux（如 CentOS、Debian 等）或 Mac 机器上执行部署脚本，但需根据操作系统的区别修改脚本中的个别命令，具体修改方式请开发者自行搜索。

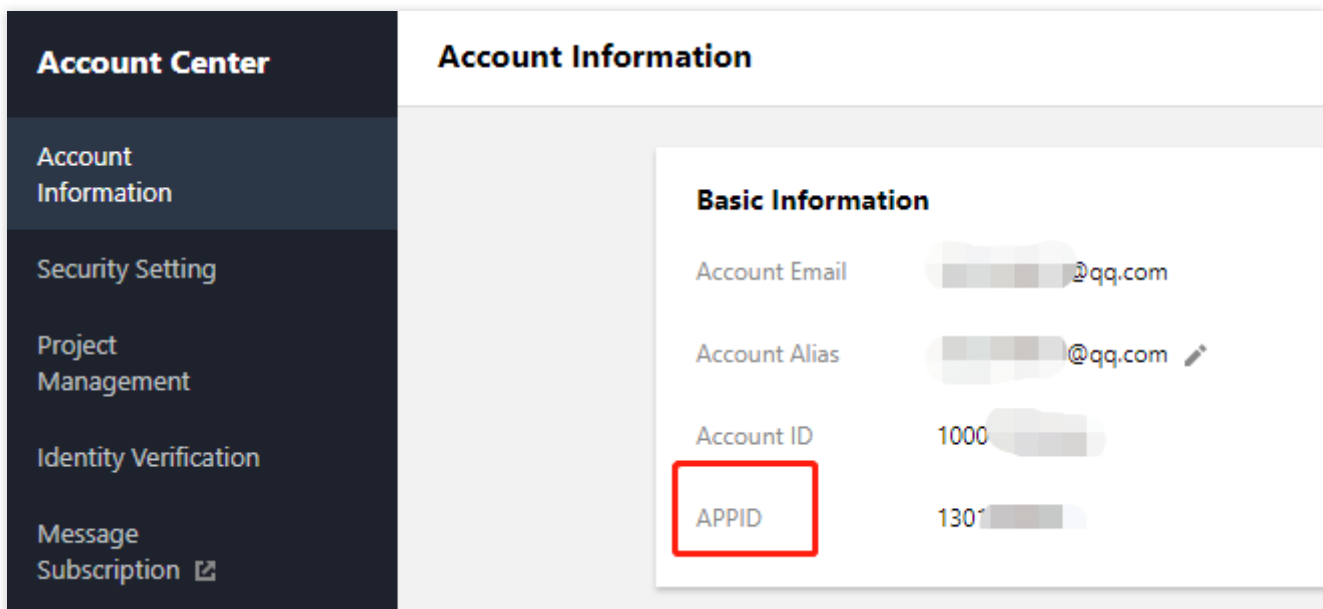
## 步骤2：开通云点播

请参考 [快速入门 - 步骤1](#) 开通云点播服务。

## 步骤3：获取 API 密钥和 APPID

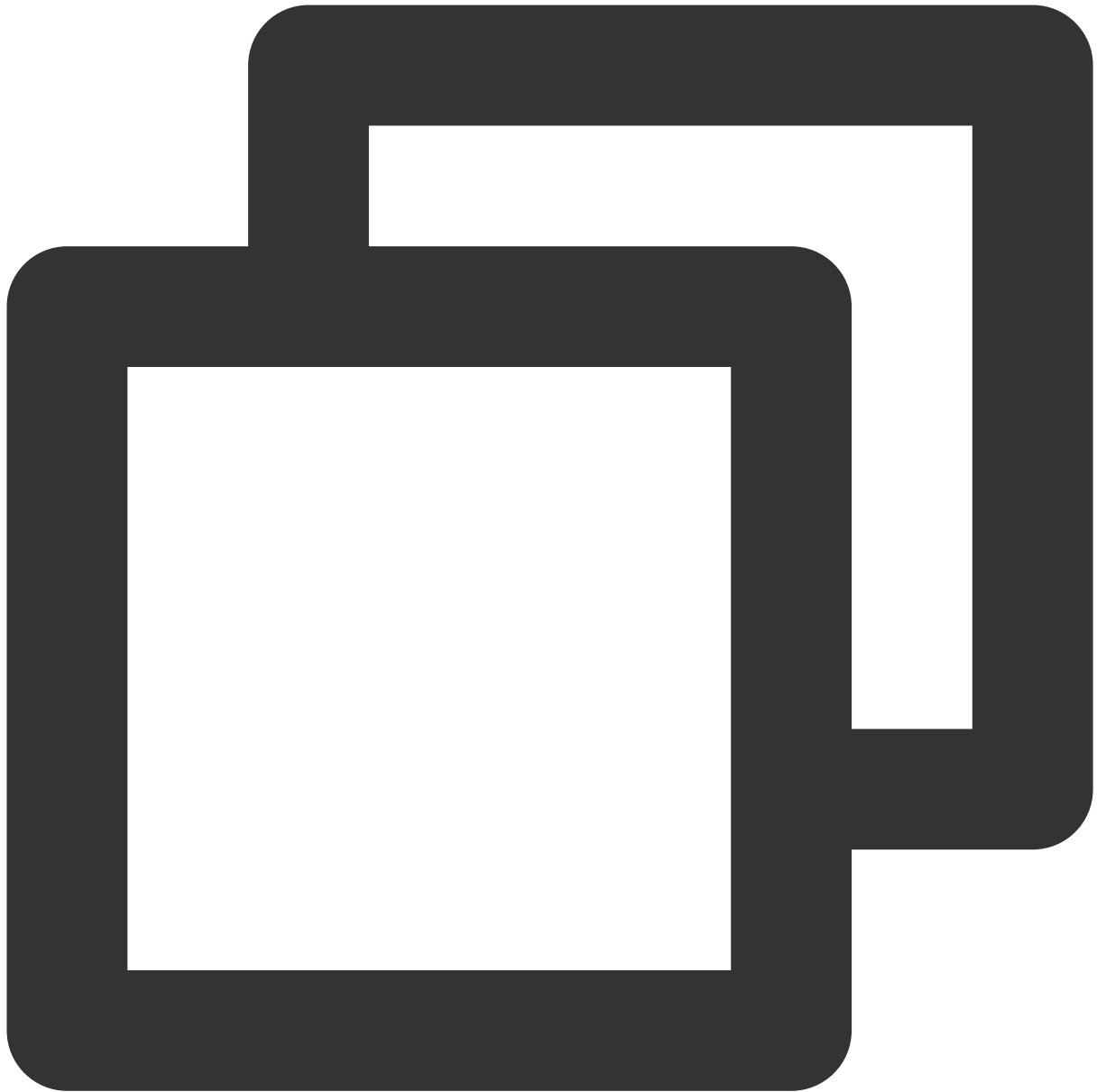
事件通知接收服务 Demo 的部署和运行过程需要使用到开发者的 API 密钥（即 SecretId 和 SecretKey）和 APPID。如果还未创建过密钥，请参见 [创建密钥文档](#) 生成新的 API 密钥；如果已创建过密钥，请参见 [查看密钥文档](#) 获取 API 密钥。

在控制台 [账号信息](#) 页面可以查看 APPID，如下图所示：



## 步骤4：部署事件通知接收服务

登录 [步骤1](#) 准备的 CVM（登录方法详见 [操作指南 - 登录 Linux](#)），在远程终端输入以下命令并运行：

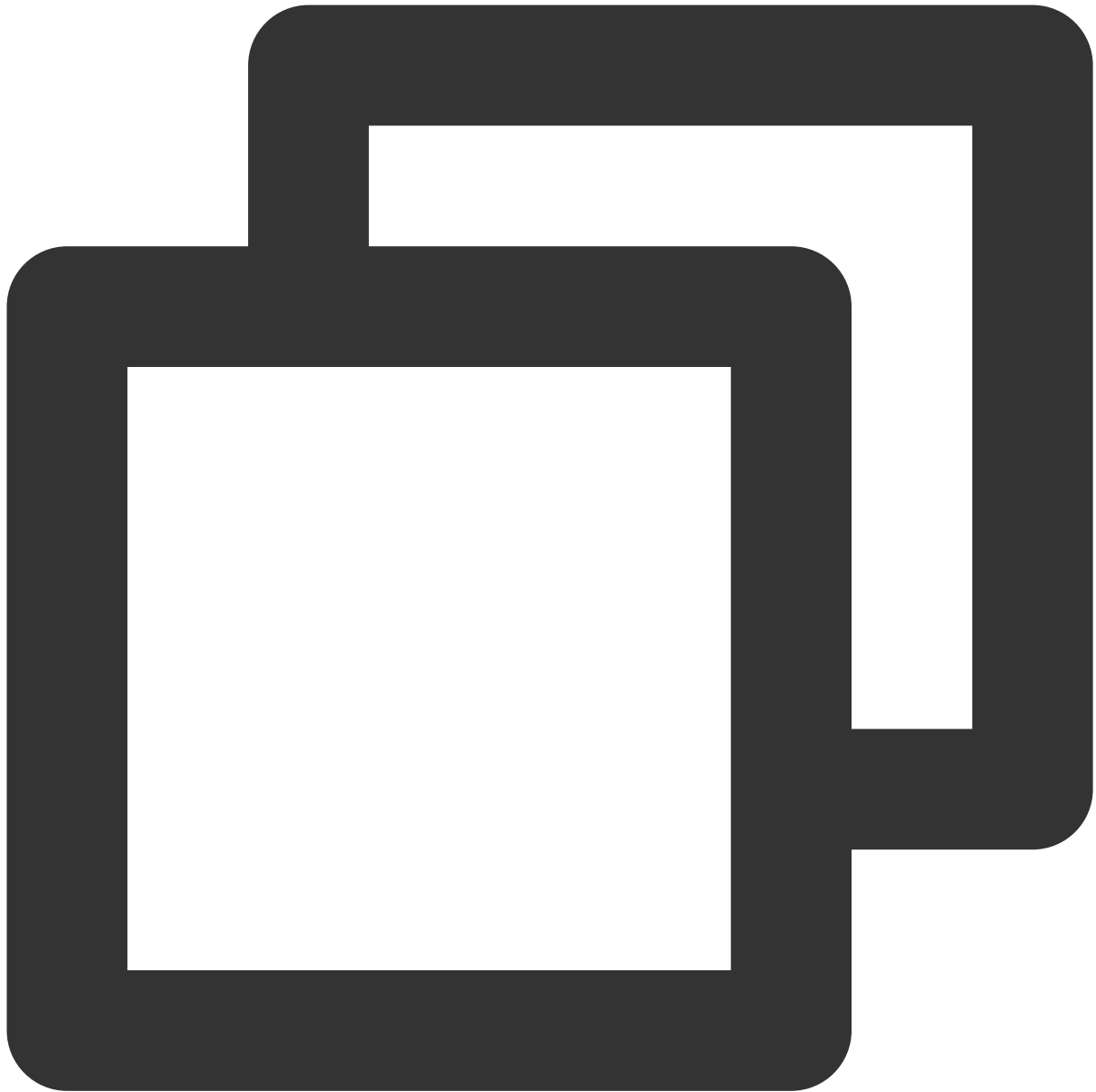


```
ubuntu@VM-69-2-ubuntu:~$ export SECRET_ID=AKxxxxxxxxxxxxxxxxxxxxxxxxxxxx; export SECRET_
```

#### 说明：

请将命令中的 SECRET\_ID、SECRET\_KEY 和 APPID 赋值为 [步骤3](#) 中获取到的内容。

该命令将从 Github 下载 Demo 源码并自动执行安装脚本。安装过程需几分钟（具体取决于 CVM 网络状况），期间远程终端会打印如下示例的信息：



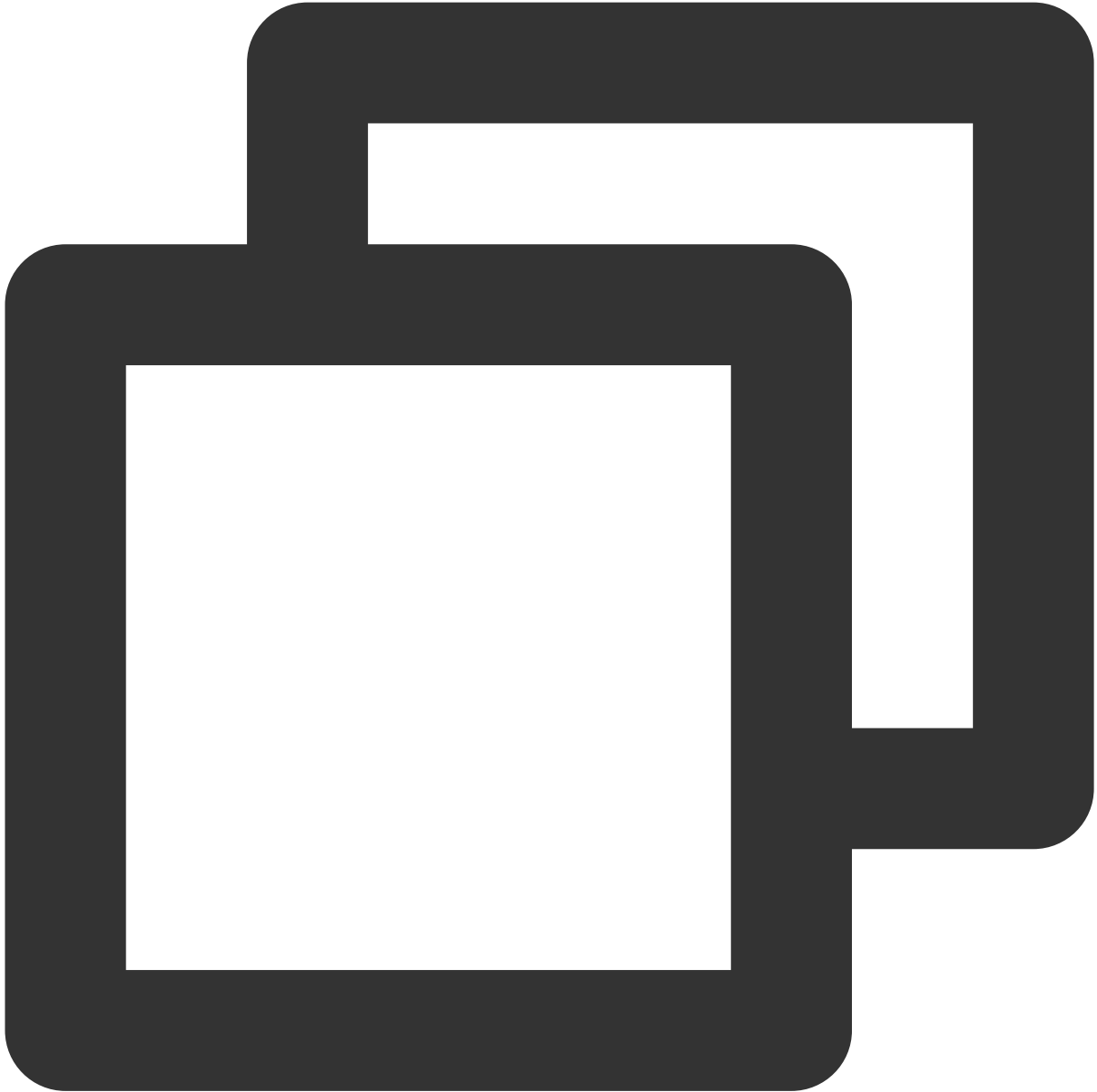
```
[2020-06-05 17:16:08]开始检查npm。  
[2020-06-05 17:16:12]npm 安装成功。  
[2020-06-05 17:16:12]开始安装 ServerLess。  
[2020-06-05 17:16:13]serverless 安装成功。  
[2020-06-05 17:16:14]开始部署云点播事件通知接收服务。  
[2020-06-05 17:16:24]云点播事件通知接收服务部署完成。  
[2020-06-05 17:16:26]服务地址：https://service-xxxxxxxx-125xxxxxxxx.gz.apigw.tencentcs.com/release/callback
```

复制输出日志中的事件通知接收服务地址（示例中的 <https://service-xxxxxxxx-125xxxxxxxx.gz.apigw.tencentcs.com/release/callback>）。



**注意：**

如果输出日志中出现如下所示的警告，一般是由于 CVM 无法立即解析刚部署好的服务域名，可尝试忽略该警告。



[2020-04-25 17:18:44] 警告：事件通知接收服务测试不通过。

**步骤5：配置事件通知地址****注意：**

如 [避免影响生产环境](#) 一节所述，操作之前请先确认您的线上业务不依赖于 VOD 事件通知。

1. 登录 [云点播控制台](#)，单击左侧导航栏**应用管理**，进入应用列表页。

- 找到需要处理音视频文件的应用，单击应用名称进入应用管理页。
- 在左侧导航栏单击**回调设置**后，再单击**设置**，回调模式选择“普通回调”，回调 URL 填写 [步骤4](#) 中获得的事件通知接收服务地址，回调事件全部勾选，然后单击确定。如下图所示：

**Callback Settings** test St

**Set**

Event Notification Method  Normal Callback  Reliable Callback

Callback URL  ✓

Event Notification

- Finished video uploading
- Deleted the video
- Finished video composition
- Task flow status changed
- Finished video editing
- ▲ Legacy Types ⓘ
  - Finished video transcoding
  - Finished screencapturing
  - Finished video splice by time points in the video
  - Finished screencapturing of image sprite in the video

**Confirm** **Cancel**

### 注意：

如果您在控制台同时看到两个回调 URL 设置（2.0版本格式和3.0版本格式），请填写3.0版本。

### 步骤6：测试 Demo

- 按照 [上传视频 - 本地上传步骤](#) 的说明，上传一个测试视频到云点播，注意上传过程选择默认的“只上传，暂不进行视频处理”。上传完成后，在“已上传”标签页可以看到该视频的状态为“处理中”，说明 Demo 接收到了 NewFileUpload 事件通知并发起了转码请求。
- 等待视频处理完成（状态变为“正常”）后，单击**快捷查看**，在页面右侧可以看到该视频有两个转码视频。

3. 登录 [SCF 控制台日志页面](#) 查看 SCF 日志记录，在最新的一条日志中，可以看到两个转码文件的 URL 已经打印出来，在实际应用场景中，开发者可以通过 SCF 将 URL 记录在自己的数据库，或者通过其它渠道发布给观众。

#### 说明：

SCF 日志可能会有些许延迟，如果在页面上没有看到日志，请耐心等待一两分钟，然后单击[重置](#)刷新。

## 系统设计说明

### 接口协议

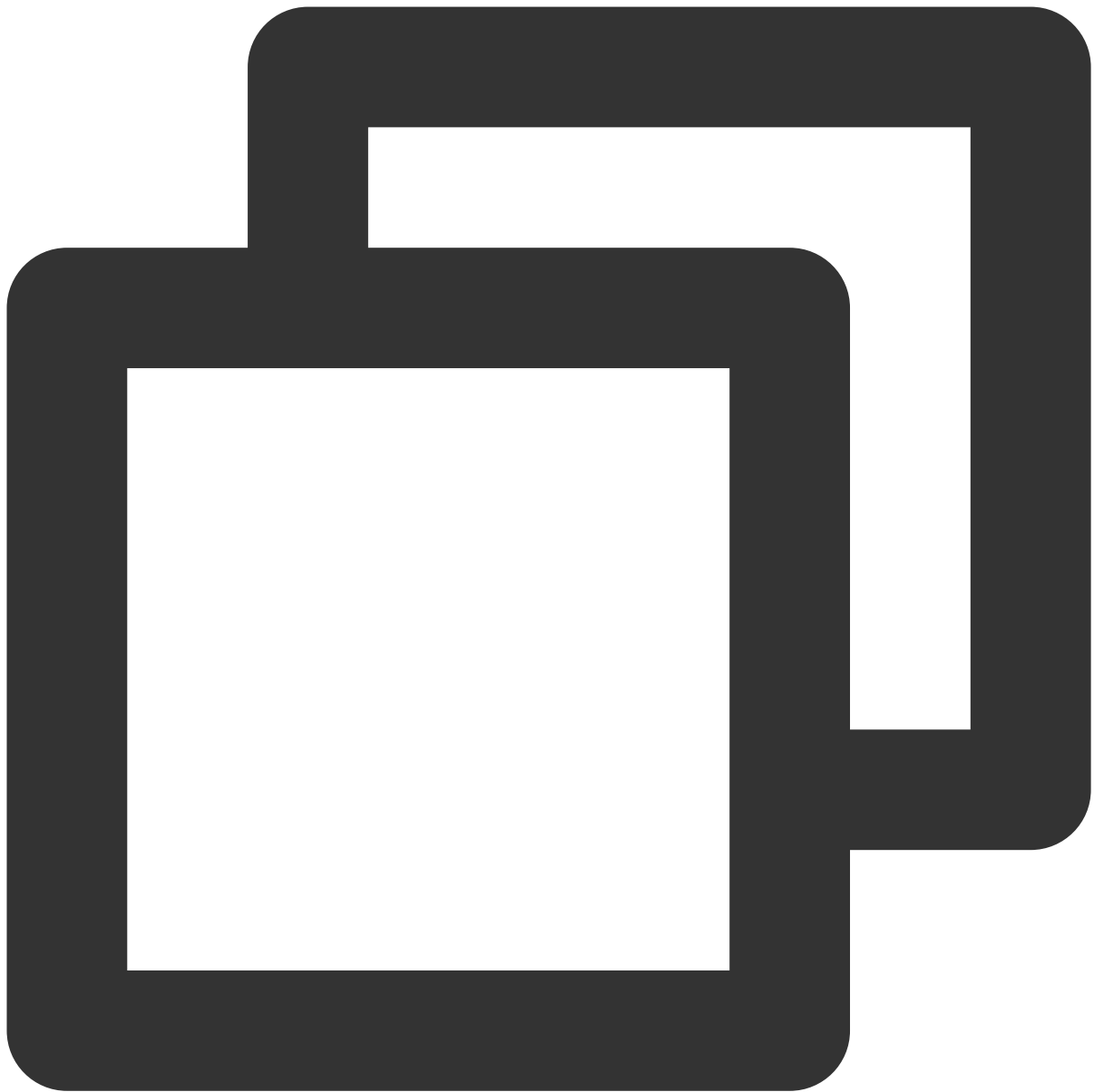
事件通知接收云函数通过 API 网关对外提供接口，具体接口协议请参考文档 [视频上传完成事件通知](#) 和 [任务流状态变更](#)。

### 事件通知接收服务代码解读

1. `main_handler()` 为入口函数。
2. 调用 `parse_conf_file()` ，从 `config.json` 文件中读取配置信息。配置项说明如下：

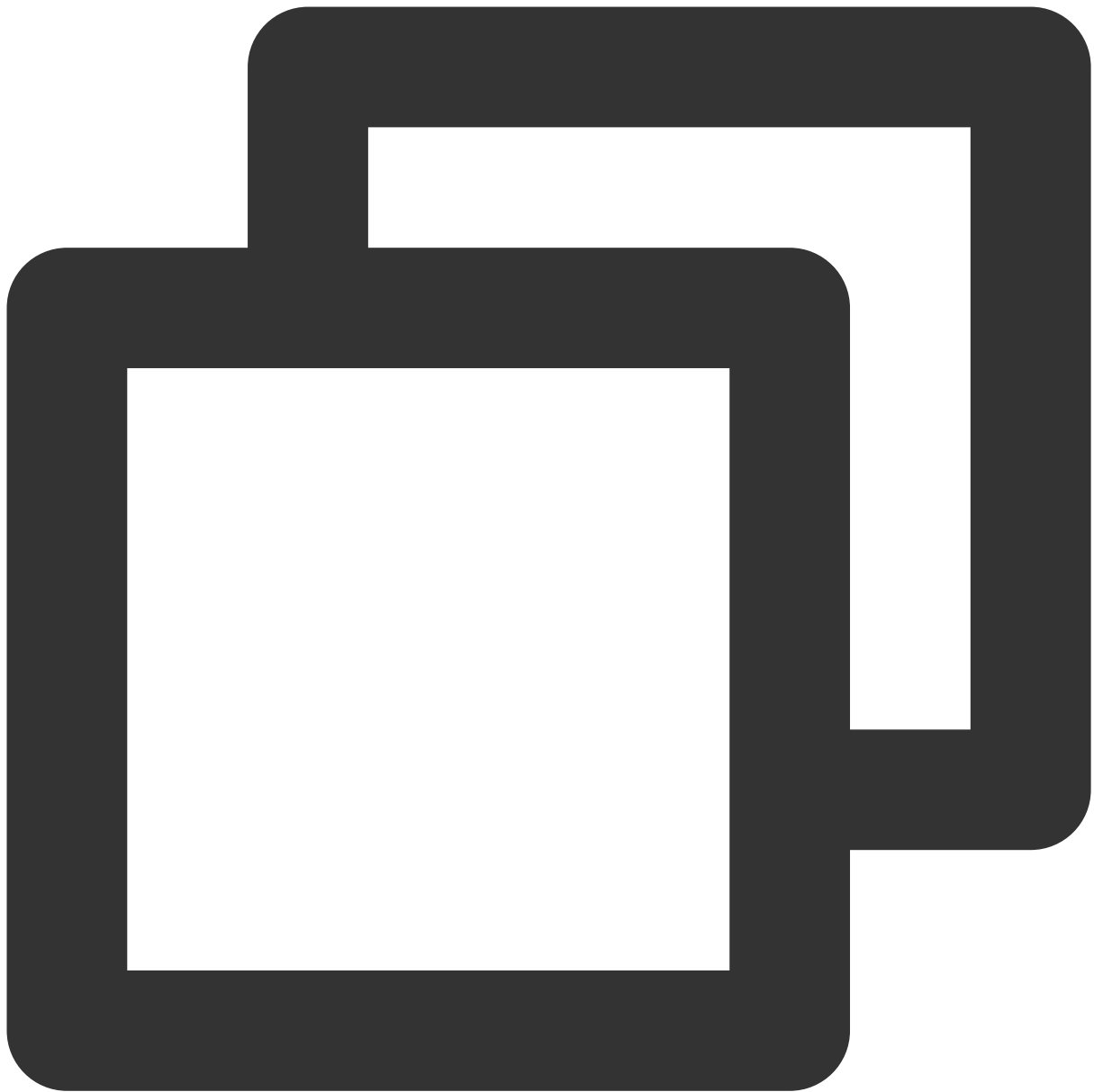
字段	数据类型	功能
<code>secret_id</code>	String	API 密钥
<code>secret_key</code>	String	API 密钥
<code>region</code>	String	云 API 请求地域，对于 VOD 可随意填写
<code>definitions</code>	Array of Integer	转码模板
<code>subappid</code>	Integer	事件通知是否来自 <a href="#">云点播子应用</a>

3. 针对 `NewFileUpload` 类型的事件通知，调用 `deal_new_file_event()` 解析请求，从中取出新上传视频的 `FileId`：



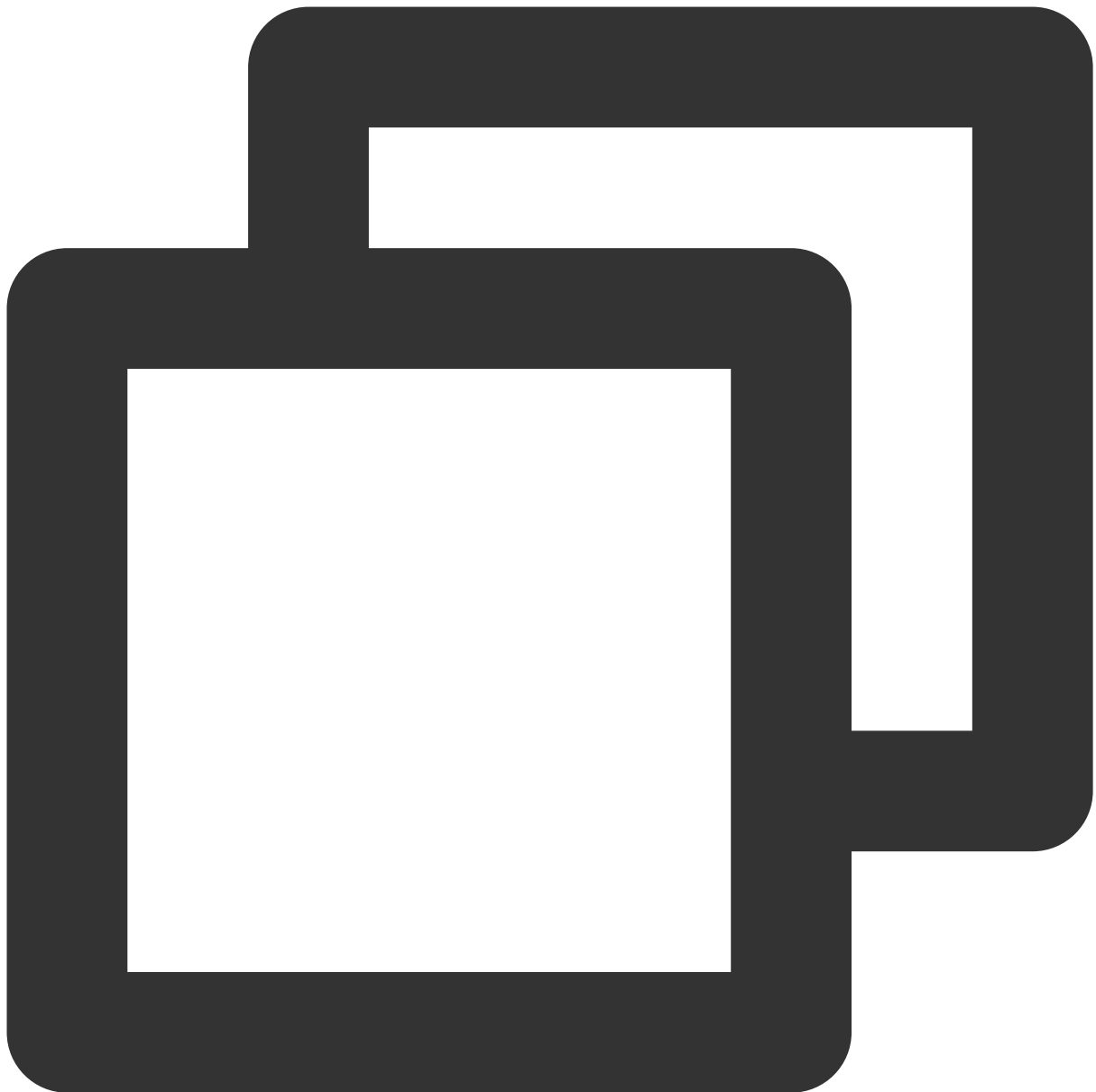
```
if event_type == "NewFileUpload":
    fileid = deal_new_file_event(body)
    if fileid is None:
        return ERR_RETURN
```

4. 调用 `trans_media()` 发起转码，输出云 API 的回包到 SCF 日志，并回包给 VOD 的事件通知服务：



```
rsp = trans_media(configuration, fileid)
if rsp is None:
    return ERR_RETURN
print(rsp)
```

5. 在 `trans_media()` 中，调用云 API SDK 发起 `ProcessMedia` 请求：

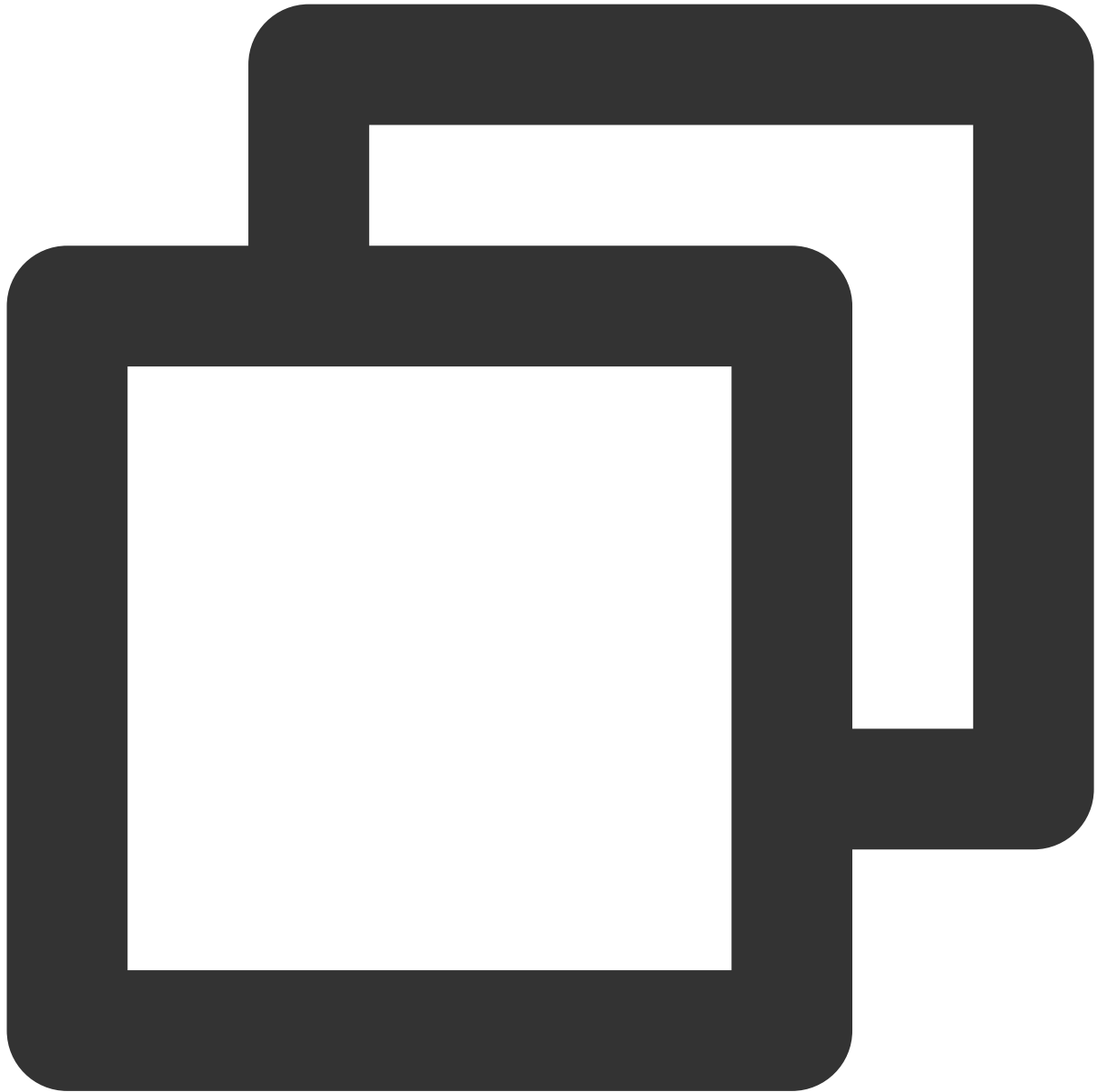


```
cred = credential.Credential(conf["secret_id"], conf["secret_key"])
client = vod_client.VodClient(cred, conf["region"])

method = getattr(models, API_NAME + "Request")
req = method()
req.from_json_string(json.dumps(params))

method = getattr(client, API_NAME)
rsp = method(req)
return rsp
```

6. 针对 ProcedureStateChanged 类型的事件通知，调用 `deal_procedure_event()` 解析请求，从中取出转码输出视频的 URL 并打印到 SCF 日志：



```
elif event_type == "ProcedureStateChanged":
    rsp = deal_procedure_event(body)
    if rsp is None:
        return ERR_RETURN
```

# 如何进行源站迁移

最近更新时间：2021-09-01 09:58:22

## 简介

VOD Migrate Tool 是一个集成了数据迁移功能的一体化工具。通过编写简单的配置文件，用户可以将源地址媒体文件快速迁移至 VOD 中。

## 支持的数据源

- 本地文件夹
- URL 列表
- 腾讯云 COS
- AWS S3
- 阿里云 OSS
- 七牛云 对象存储

## 使用环境

### 系统环境

支持 Windows、Linux 和 macOS 系统。

### 软件依赖

- Python 2.7/3.4+。
- 最新版本的 pip。

## 安装

### 通过 Pip 安装（推荐）

您可以通过 pip 安装方式将 SDK 安装到您的项目中，如果您的项目环境尚未安装 pip，请详细参见 [pip 官网安装](#)。

```
pip install vodmigrate
```



## 通过源码包安装

源码下载地址：[单击此处](#)。

下载最新代码，解压后：

```
git clone https://github.com/tencentyun/vod-migrate.git
cd vod-migrate
python setup.py install
```

## 使用示例

执行命令：

```
vodmigrate config.toml
```

说明：

迁移完成后，结果将输出到配置项"migrateResultOutputPath"对应的目录，文件名为：

vod\_migrate\_result.txt。

## 配置文件说明

配置文件采用 toml 格式（参考：[config\\_template.toml](#)，请确保文件为 UTF-8 编码），内容可以分为以下几部分：

### 1. 配置迁移类型

type 表示迁移类型，用户根据迁移需求填写对应的标识。例如，需要将本地数据迁移至 VOD，

则 [migrateType] 的配置内容是 type=migrateLocal 。

```
[migrateType]
type="migrateLocal"
```

目前支持的迁移类型如下：

migrateType	描述
migrateLocal	从本地迁移至 VOD
migrateUrl	下载 URL 迁移到 VOD

migrateType	描述
migrateCos	从 腾讯云 COS 迁移至 VOD
migrateAws	从 AWS S3 迁移至 VOD
migrateAli	从阿里云 OSS 迁移至 VOD
migrateQiniu	从七牛云存储迁移至 VOD

## 2. 配置迁移任务

用户根据实际的迁移需求进行相关配置，主要包括迁移至 VOD 信息配置及迁移任务相关配置。

```
# 迁移工具公共配置
[common]
secretId = "SECRETID"
secretKey = "SECRETKEY"
region = 'REGION'
subAppId = 0
concurrency = 5
supportMediaClassification = [ 'video', 'audio', 'image' ]
excludeMediaType = [ ]
migrateDbStoragePath = ''
migrateResultOutputPath = ''
```

名称	描述
secretId	用户密钥 SecretId，请将 SECRETID 替换为您的真实密钥信息。可前往 <a href="#">访问管理控制台</a> 中的云 API 密钥页面查看获取
secretKey	用户密钥 SecretKey，请将 SECRETKEY 替换为您的真实密钥信息。可前往 <a href="#">访问管理控制台</a> 中的云 API 密钥页面查看获取
region	接入点地域，即请求到哪个地域的云点播服务器，不同于存储地域，具体参考支持的 <a href="#">地域列表</a>
subAppId	点播子应用 ID。如果需要将文件迁移到子应用，则将该字段填写为子应用 ID；否则无需填写该字段
concurrency	并发迁移文件的数量，最大值50
supportMediaClassification	支持迁移的媒体类型列表：video（视频），audio（音频），image（图像）
excludeMediaType	需要排除的文件类型列表

名称	描述
migrateDbStoragePath	迁移 db 保存路径，为空表示当前目录
migrateResultOutputPath	迁移结果保存路径（一条迁移记录对应一行 json 格式字符串），为空表示当前目录

文件类型说明：

- 视频：MP4、TS、FLV、WMV、ASF、RM、RMVB、MPG、MPEG、3GP、MOV、WEBM、MKV、AVI，不支持 HLS、DASH
- 音频：MP3、M4A、FLAC、OGG、WAV
- 图像：JPG、JPEG、PNG、GIF、BMP、TIFF、AI、CDR、EPS

### 3. 配置数据源信息

根据 [migrateType] 的迁移类型配置相应的分节。例如 [migrateType] 的配置内容是 type=migrateLocal，则用户只需配置 [migrateLocal] 分节即可。

#### 3.1 配置本地数据源 migrateLocal

若从本地迁移至 VOD，则进行该部分配置，具体配置项及说明如下：

```
# 从本地迁移到 VOD 配置分节
[migrateLocal]
localPath = ''
excludes = [ ]
```

配置项	描述
localPath	本地路径，要求格式为绝对路径
excludes	要排除的目录的绝对路径，表示将 localPath 下面某些目录下文件不进行迁移

#### 3.2 配置 URL 列表数据源 migrateUrl

若从指定 URL 列表迁移至 VOD，则进行该部分配置，具体配置项及说明如下：

```
# 从 URL 列表下载迁移到 VOD 配置分节
[migrateUrl]
urllistPath = 'D:\folder\urllist.txt'
```

配置项	描述
-----	----

配置项	描述
urllistPath	存储 URL 列表的文件绝对路径。文件的内容为 URL 文本，一行一条 URL 原始地址

说明：

如需迁移本地大型文件至云点播，建议使用 [pullUpload](#) 接口拉取。

### 3.3 配置 COS 数据源 migrateCos

若从腾讯云 COS 迁移至 VOD，则进行该部分配置，具体配置项及说明如下：

```
# 从腾讯云 COS 迁移至 VOD 配置分节
[migrateCos]
region = 'ap-shanghai'
bucket = 'examplebucket-1250000000'
secretId = 'COS_SECRETID'
secretKey = 'COS_SECRETKEY'
prefix = ''
```

配置项	描述
region	Bucket 的 Region 信息，请参照 <a href="#">可用地域</a>
bucket	Bucket 的名称，命名格式为 <code>&lt;bucketname-appid&gt;</code> ，即 Bucket 名必须包含 APPID，例如 examplebucket-1250000000
secretId	Bucket 隶属的用户密钥 SecretId，可在 <a href="#">云 API 密钥</a> 查看
secretKey	Bucket 隶属的用户密钥 secret_key，可在 <a href="#">云 API 密钥</a> 查看
prefix	要迁移路径的前缀，如果是迁移 Bucket 下所有的数据，则 prefix 为空

### 3.4 配置 AWS 数据源 migrateAws

若从 AWS 迁移至 VOD，则进行该部分配置，具体配置项及说明如下：

```
# 从 AWS 迁移到 VOD 配置分节
[migrateAws]
region = 'ap-northeast-2'
bucket = 'bucket-aws'
accessKeyId = 'AccessKeyId'
accessKeySecret = 'AccessKeySecret'
prefix = ''
```

配置项	描述
region	AWS 对象存储 Region
bucket	AWS 对象存储 Bucket 名称
accessKeyId	将 AccessKeyId 替换为用户的密钥
accessKeySecret	将 AccessKeySecret 替换为用户的密钥
prefix	要迁移的路径的前缀，如果是迁移 Bucket 下所有的数据，则 prefix 为空

### 3.5 配置阿里 OSS 数据源 migrateAli

若从阿里云 OSS 迁移至 VOD，则进行该部分配置，具体配置项及说明如下：

```
# 从阿里 OSS 迁移到 VOD 配置分节
[migrateAli]
bucket = 'bucket-aliyun'
accessKeyId = 'yourAccessKeyId'
accessKeySecret = 'yourAccessKeySecret'
endPoint = 'oss-cn-hangzhou.aliyuncs.com'
prefix = ''
```

配置项	描述
bucket	阿里云 OSS Bucket 名称
accessKeyId	将 yourAccessKeyId 替换为用户的密钥
accessKeySecret	将 yourAccessKeySecret 替换为用户的密钥
endPoint	阿里云 endpoint 地址
prefix	要迁移路径的前缀，如果是迁移 Bucket 下所有的数据，则 prefix 为空

### 3.6 配置七牛数据源 migrateQiniu

若从七牛迁移至 VOD，则进行该部分配置，具体配置项及说明如下：

```
# 从七牛迁移到 VOD 配置分节
[migrateQiniu]
bucket = 'bucket-qiniu'
accessKeyId = 'AccessKey'
accessKeySecret = 'SecretKey'
```

```
endPoint = 'www.bkt.clouddn.com'  
prefix = ''
```

配置项	描述
bucket	七牛对象存储 Bucket 名称
accessKeyId	将 AccessKey 替换为用户的密钥
accessKeySecret	将 SecretKey 替换为用户的密钥
endPoint	七牛下载地址，对应 downloadDomain
prefix	要迁移路径的前缀，如果是迁移 Bucket 下所有的数据，则 prefix 为空

## 限制说明

- 该工具定位为一次性的迁移工具。迁移分为**源站文件扫描**、**迁移中**、**迁移完成**三个阶段。文件扫描完成后，如果配置需变更，这时候需将 db 文件清空（删除 migrate.db 或者修改 db 存储路径），以规避配置文件 md5 校验报错。
- 迁移的文件必须显示的带后缀。
- 暂不支持 HLS/DASH 迁移。
- 迁移后无法维持原视频的目录关系、每个视频都是独立的 FileId，相互无关联。

## 迁移流程介绍

1. 读取配置文件，根据迁移 type，读取相应的配置分节，并执行参数的检查。
2. 根据指定的迁移类型，扫描源站，生成迁移任务。
3. 扫描完成后，执行迁移，并打印每个任务的结果及整体进度。

4. 迁移完成后，将详细信息输出到结果文件。

```
[2020-06-23 11:03:15,077] build tasks
[2020-06-23 11:03:16,611] add migrate task: E:\cloud\vod_migrate\media\fancybg.jpg
[2020-06-23 11:03:16,613] add migrate task: E:\cloud\vod_migrate\media\Wildlife - 副本 (2).mp4
[2020-06-23 11:03:16,616] add migrate task: E:\cloud\vod_migrate\media\Wildlife - 副本.mp4
[2020-06-23 11:03:16,625] add migrate task: E:\cloud\vod_migrate\media\Wildlife.mp4
[2020-06-23 11:03:16,626] add migrate task: E:\cloud\vod_migrate\media\中文.jpg
[2020-06-23 11:03:16,627] add migrate task: E:\cloud\vod_migrate\media\测试.png
[2020-06-23 11:03:17,772] 当前迁移进度: 16.67%, 总量: 6, 成功数: 1, 失败数: 0
[2020-06-23 11:03:17,852] 当前迁移进度: 33.33%, 总量: 6, 成功数: 2, 失败数: 0
[2020-06-23 11:03:17,957] 当前迁移进度: 50.00%, 总量: 6, 成功数: 3, 失败数: 0
[2020-06-23 11:03:18,068] 当前迁移进度: 66.67%, 总量: 6, 成功数: 4, 失败数: 0
[2020-06-23 11:03:18,218] 当前迁移进度: 83.33%, 总量: 6, 成功数: 5, 失败数: 0
[2020-06-23 11:03:18,292] 当前迁移进度: 100.00%, 总量: 6, 成功数: 6, 失败数: 0
[2020-06-23 11:03:19,295] tasks finished
[2020-06-23 11:03:19,454] migrate result: E:\cloud\vod_migrate\vod_migrate_result.txt
```

# 直播录制

## 如何将直播视频录制到云点播并进行处理

最近更新时间：2023-05-15 17:15:06

直播录制方案将直播原始流经过转音视频封装（不修改音频、视频数据以及对应的时间戳等信息）得到的文件存储到腾讯云点播平台，并对录制文件进行二次制作、分发播放的标准解决方案。更多详情可参见 [直播录制解决方案](#)。

### 产品特性

基于腾讯云直播能力，能快速将直播流内容进行录制并存储至云平台并进行二次制作和分发。

基于腾讯云领先的音视频 AI 技术及全球海量直播加速节点，为您提供专业、稳定的直播推流、转码、分发及播放服务，全面满足超低延迟、超高画质、超大并发访问量的要求。

基于直播录制服务，能快速将您的直播活动传播到各种应用场景和 App 中。

适用于多种行业场景，如企业直播、电商直播、教育直播等。适用于多种分发方案，如微信小程序、腾讯视频等。

### 前提条件

[注册](#) 并 [登录](#) 腾讯云账号。

已开通腾讯云直播和云点播服务。若未开通，请前往开通 [云直播服务](#) 和 [云点播服务](#)。

### 实践步骤

#### 步骤1：创建录制模板

使用录制功能需要先创建录制模板，直播录制功能的配置均保存在录制模板中。通过创建不同配置的录制模板，可以实现不同格式、不同录制文件时长等效果。

通过控制台创建：

1.1 进入 [云直播控制台](#)，选择 [功能配置](#) > [直播录制](#)。

1.2 单击 [创建模板](#)，选择需要的录制文件类型（至少选择一种格式）。更多配置项描述请参见 [创建模板](#)。



Create Recording Template    Bind Domain Name

Create Template

443

test0825

test

recordchenhui

All\_Formats

### Recording Configuration

Template Name \*    Enter a template name  
Only support letters, digits, underscores, and d

Template Description    Please describe template  
Only support letters, digits, underscores, and d

Recording Format \*     HLS     FLV     MP4     AAC ⓘ

▼ Audio/Video - MP4

Max Recording Time Per File    1-120

Storage Period     Permanent   

VOD Subapplication/Category    Primary application

Advanced Configuration ▼

Save    Cancel

1.3 单击**保存**即可成功创建模板。

**通过 API 创建：**

调用 `CreateLiveRecordTemplate` 接口创建录制模板，模板创建成功后会返回对应的模板 ID。

## 步骤2：选择录制方案

云直播根据不同的场景，提供了以下几种调用直播录制功能的方案：

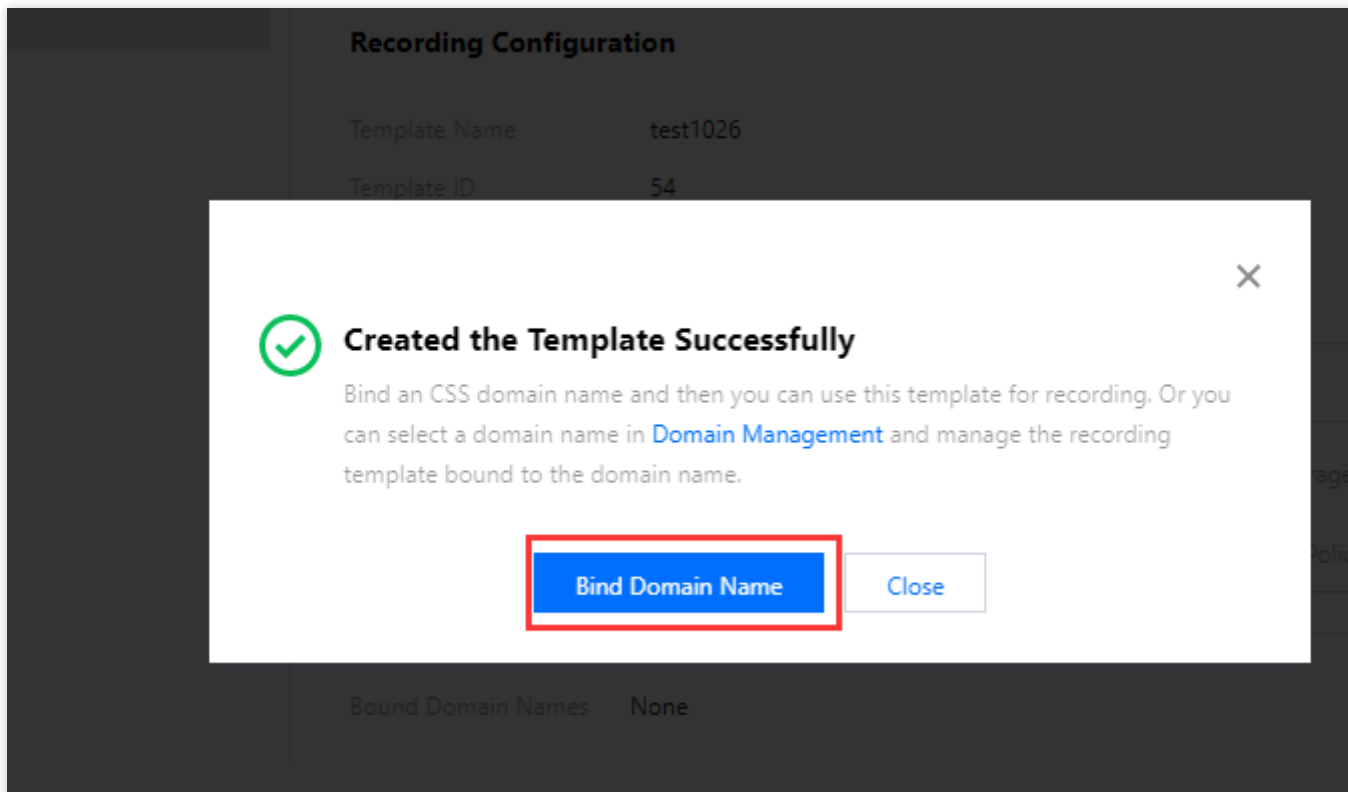
### 方案一：指定域名全局录制

通过 [云直播控制台](#) 或者调用 API，将直播录制模板绑定到推流域名，只要通过该域名推流就会自动进行录制。

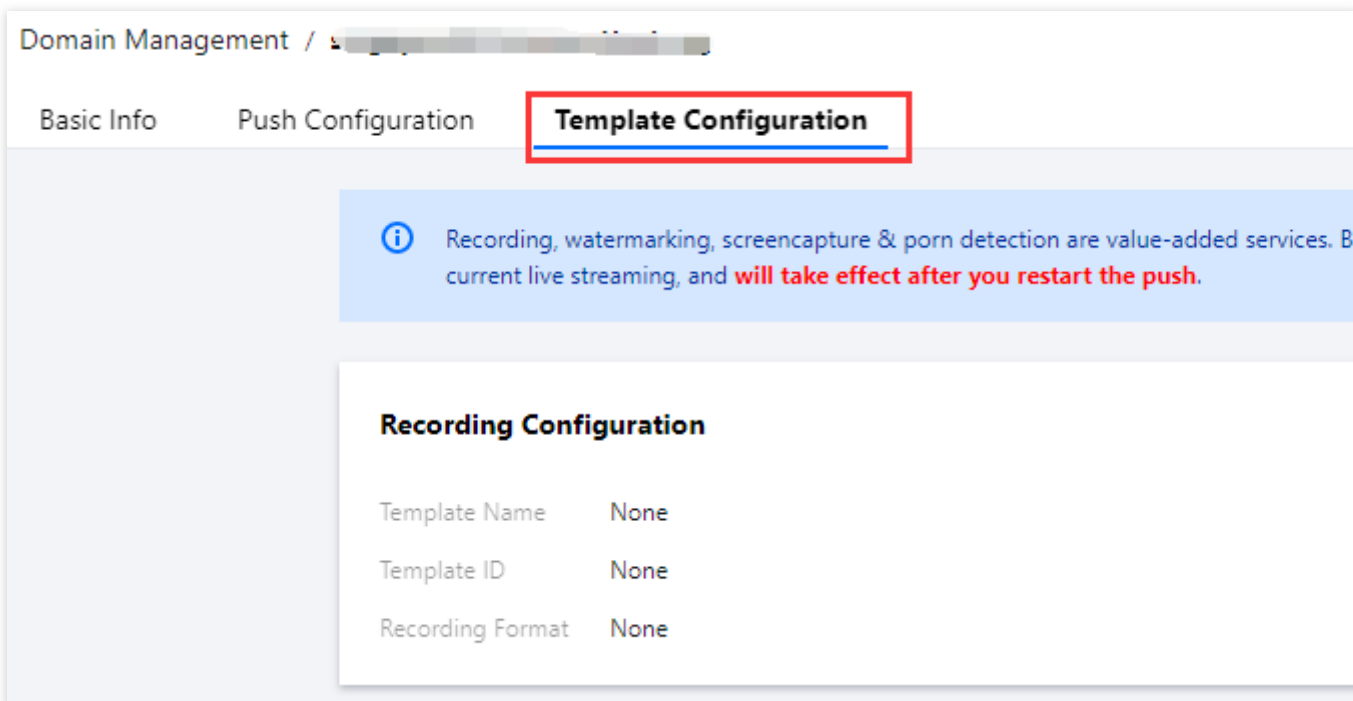
**适用场景：**秀场直播、电商直播、在线课堂、视频监控等全录制场景。

**操作流程：**

1.1 在创建录制模板成功后，将有弹框提醒您 [绑定域名](#)，单击**去绑定域名**并选择推流域名即可。



1.2 在 [域名管理](#) 中，单击您的 [直播推流域名](#) 将会跳转到推流详情页，选择 [模板配置](#) > [录制配置](#)，单击 [编辑](#) 即可绑定您的推流域名。详情请参见 [关联录制模板](#)。



1.3 通过 [CreateLiveRecordRule](#) 接口传入录制模板的模板 ID 和推流域名，即可完成录制模板绑定推流域名。

### 方案二：指定单个流录制

通过 API 将直播录制模板绑定到某个指定的直播流，从而实现录制某个直播流。

**适用场景：**活动直播、展会直播、赛事直播、连麦直播等单个活动特殊录制场景。

**操作流程：**通过 [CreateLiveRecordRule](#) 接口创建录制规则，传入录制模板的模板 ID 和需要绑定的域名、路径和流名称 StreamName（需精准匹配），即可完成录制模板绑定指定直播流。

### 方案三：按指定时间段录制

您可以通过调用 API 控制录制的开始与结束时间，在指定的时间内触发录制任务进行录制。

**适用场景：**新闻直播、活动直播等有直播流程比较明确的录制场景。

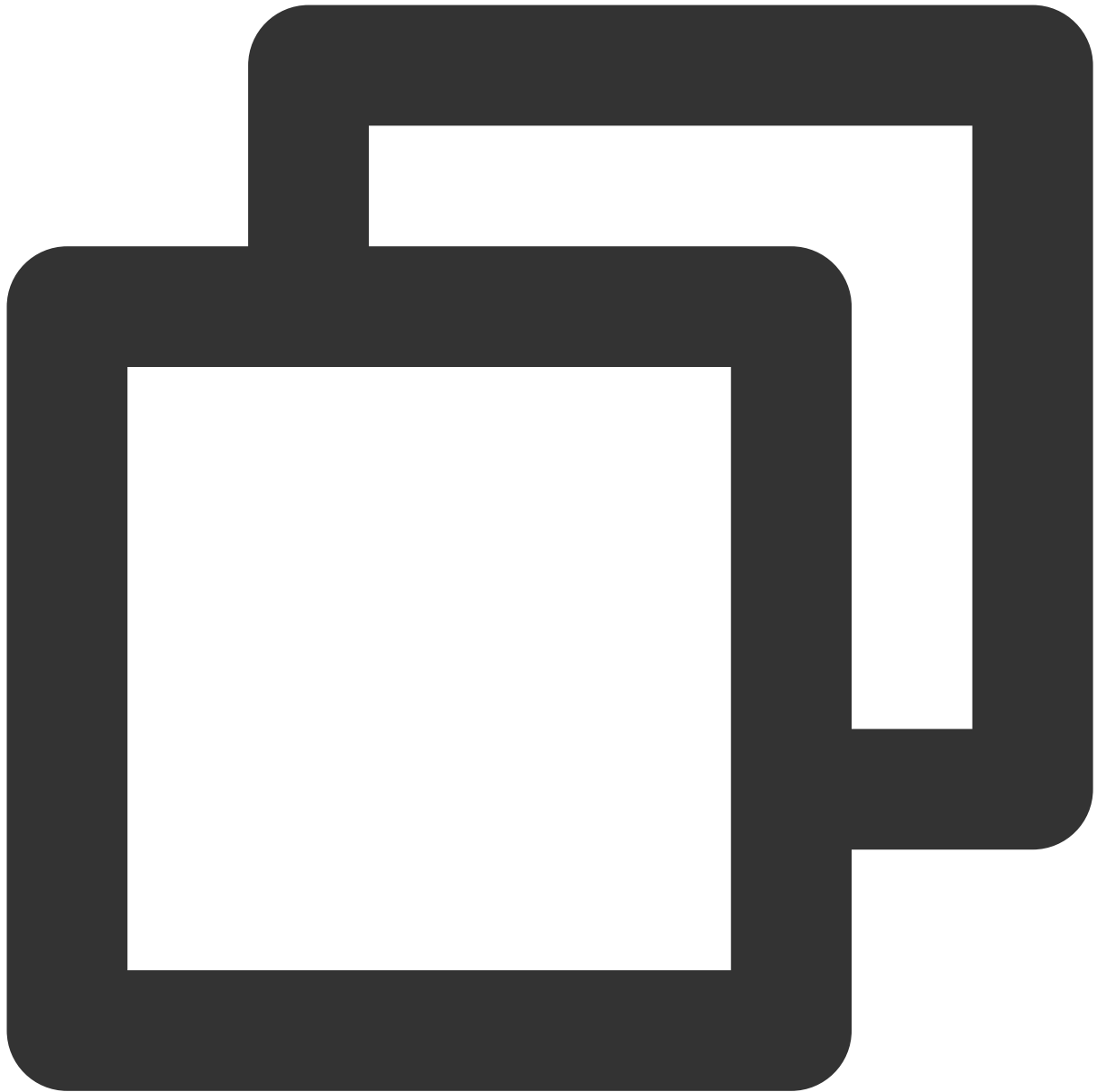
**操作流程：**通过 [CreateRecordTask](#) 接口创建录制任务，指定录制模板的模板 ID 和需要绑定的域名、路径和流名称 StreamName（需精准匹配），设置需要进行录制的开始和结束时间，则到达开始时间时即开始录制。

#### 录制示例：

1. 最简单的情况，只需填写指定的 StreamName、DomainName、AppName 和 EndTime 参数。

例如：创建了2020年08月10日早上08点到10点的录制任务，格式为 FLV，视频录制，分片间隔30分钟，永久存储。

输入示例：



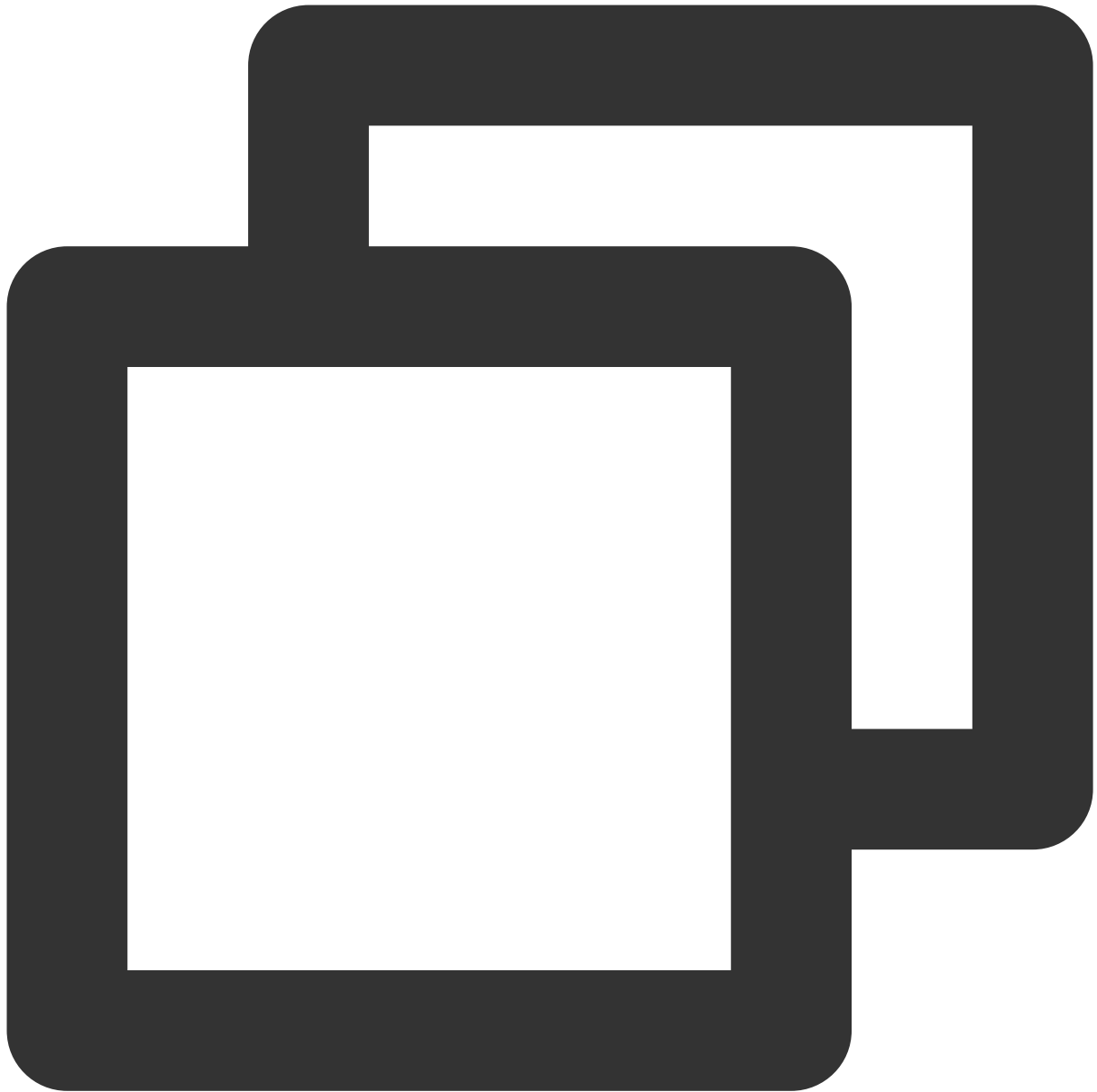
```
https://live.tencentcloudapi.com/?Action=CreateRecordTask&AppName=live&DomainName=m
```

2. 您还可以指定具体录制格式、录制类型以及存储参数等。

例如：创建了2020年08月10日早上08点到10点的录制任务，格式为 MP4，分片间隔1小时，永久存储。

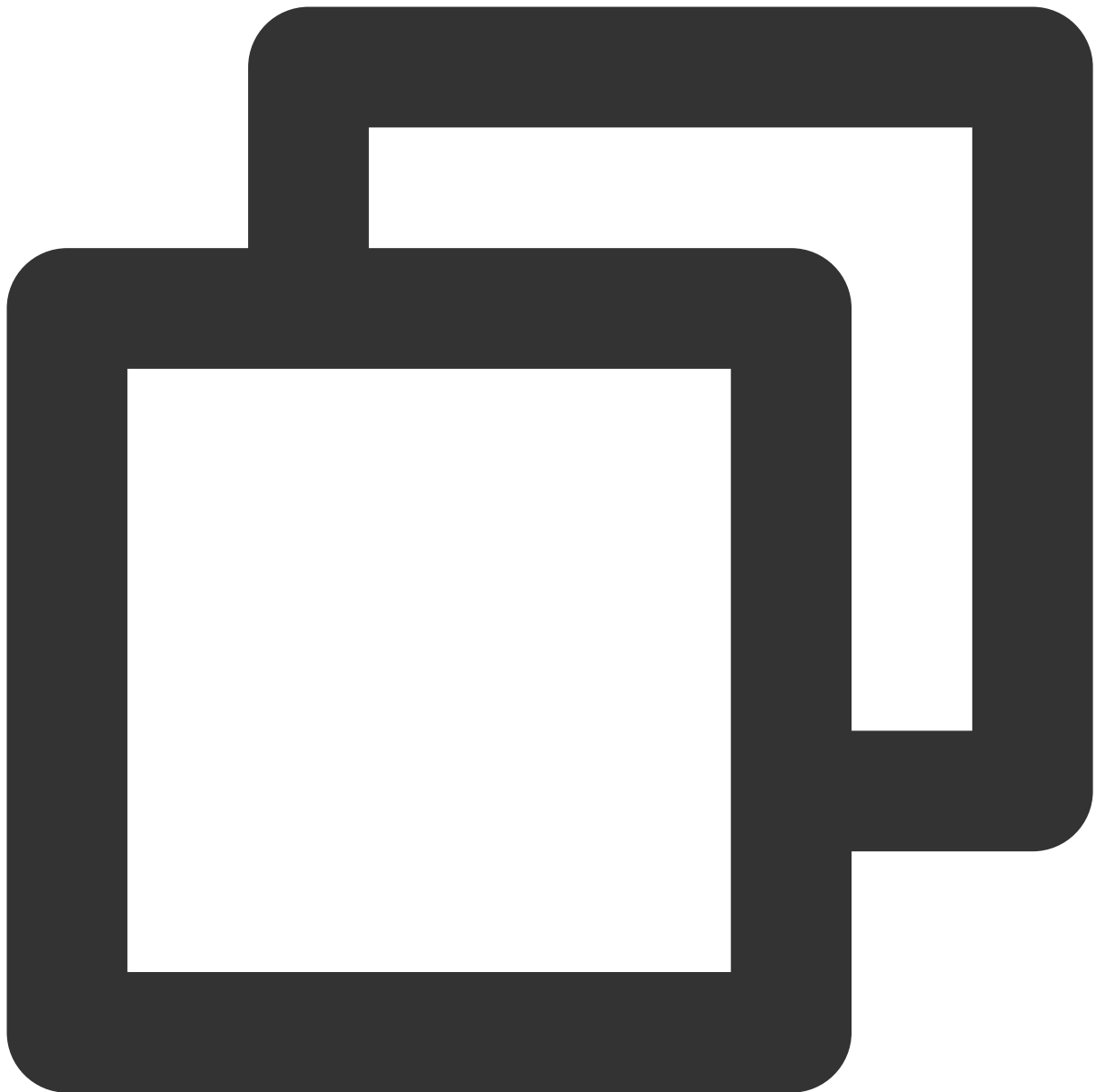
3. 调用 [CreateLiveRecordTemplate](#)，先创建录制模板。

输入示例：



```
https://live.tencentcloudapi.com/?Action=CreateLiveRecordTemplate&TemplateName=temp
```

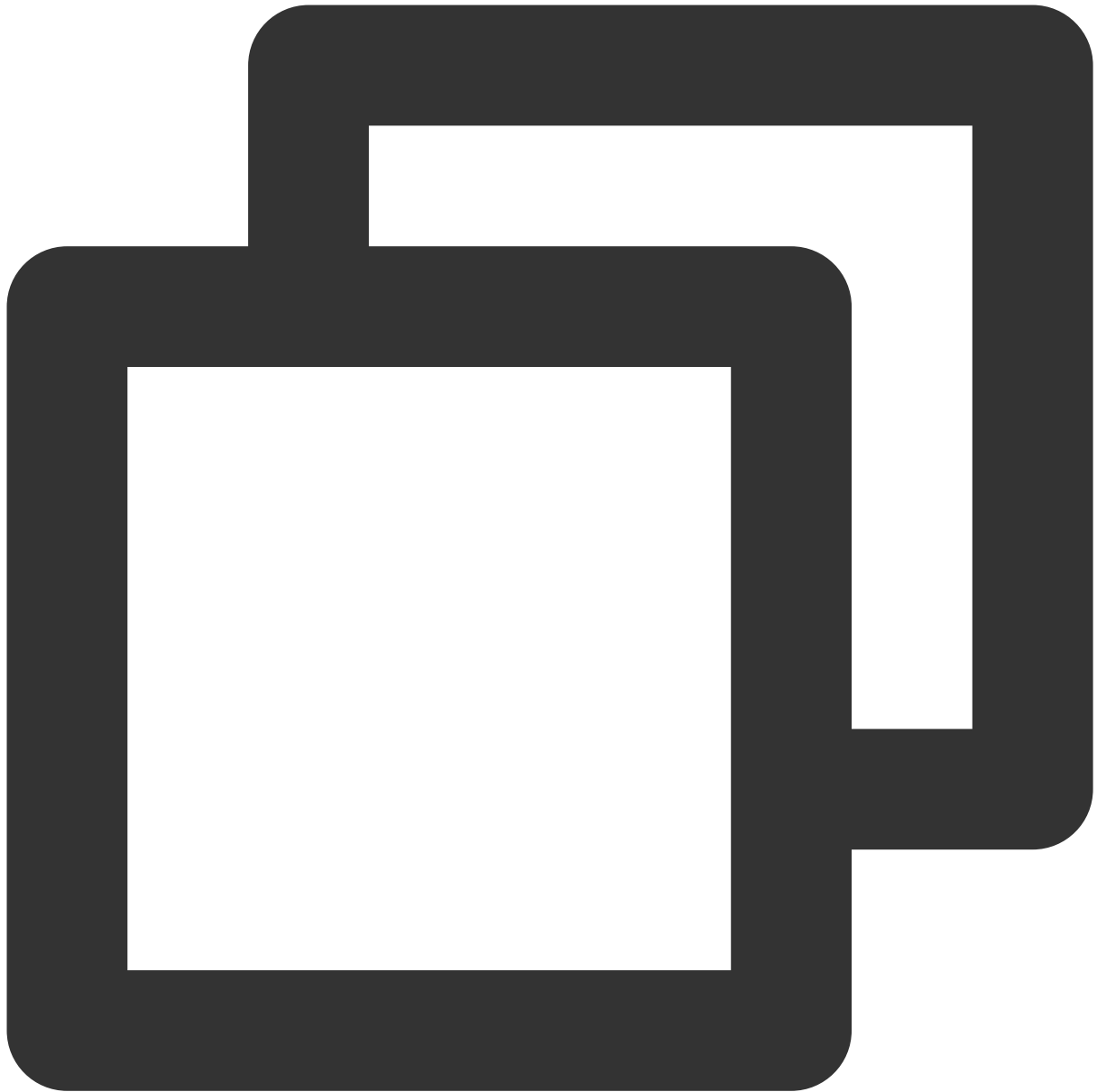
输出示例：



```
{"Response": {"RequestId": "839d12da-95a9-43b2-a9a0-03366d01b532", "TemplateId": 170
```

4. 调用 [CreateRecordTask](#)，创建录制任务。

输入示例：



```
https://live.tencentcloudapi.com/?Action=CreateRecordTask&StreamName=livetest&AppNa
```

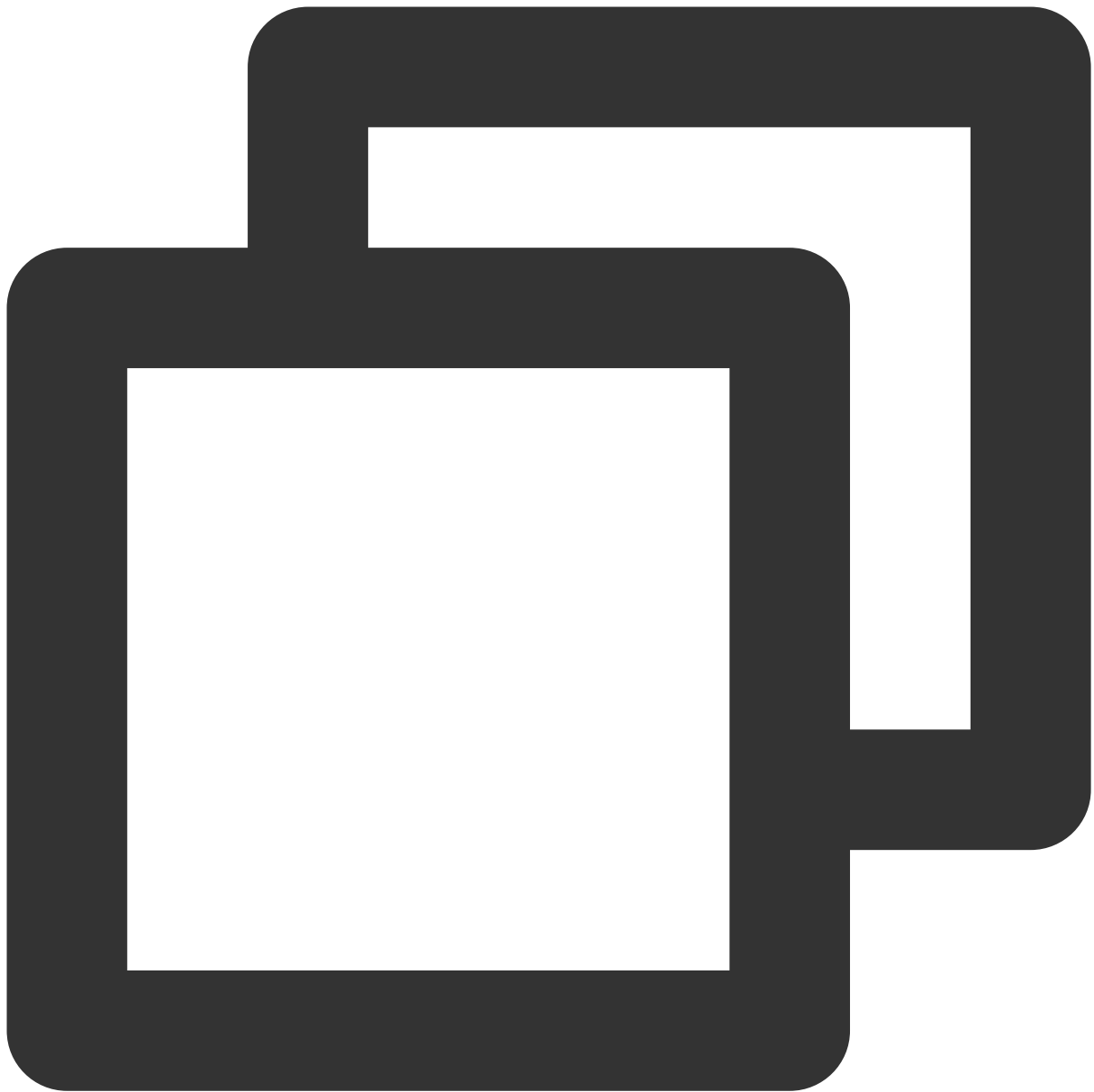
#### 方案四：精彩片段录制（支持录制混流直播）

在直播过程中碰到精彩画面，可以通过调用 API 实时进行直播录制。

**适用场景：**赛事直播、游戏直播等只需要录制部分片段的场景（或者通过全局录制后再进行剪辑）。

**操作流程：**通过 [CreateRecordTask](#) 接口创建录制任务，指定录制模板的模板 ID 和需要绑定的域名、路径和流名称 StreamName（需精准匹配），设置需要进行录制的结束时间，创建成功后即开始录制任务。

**录制示例：**



```
https://live.tencentcloudapi.com/?Action=CreateRecordTask&StreamName=test&AppName=1
```

#### 方案五：录制纯音频

若推流为纯音频，您可以配置 AAC 纯音频录制。

**适用场景：**音频直播、音频连麦等场景。

**操作流程：**创建录制模板时，选择录制文件类型为 ACC 纯音频录制，并关联对应的推流地址即可。

#### 注意：

创建绑定规则后约5分钟 - 10分钟生效，绑定规则的修改不影响正在推流的直播，只对新推的直播流生效。



### 步骤3：开始直播推流

根据 [步骤2](#) 将录制模板绑定推流域名后，通过该推流地址生成对应的推流域名并进行 [直播推流](#)。

当直播结束后，录制生成的文件会存储到 [云点播](#) 平台。

#### 说明：

若录制模板中选择录制至子应用，则存储到对应子应用下。

若需要回调录制文件地址信息，需要在推流前创建回调模板，填写录制回调地址后保存，并绑定需要回调的推流域名，详情请参见 [录制事件通知](#)。

### 步骤4：获取录制文件

支持通过以下方式可查询、获取录制文件：

**通过录制回调获取：**在直播推流前配置好回调模板（模板需要 [配置录制回调地址](#)），生成录制文件时会通过回调将文件发给回调服务器。更多详情请参见 [录制事件通知](#)。

The screenshot shows the 'Create Callback Template' configuration page. At the top, there are two buttons: 'Create Callback Template' (highlighted with a red box) and 'Bind Domain Name'. Below is a list of templates, with 'test' selected. The 'Callback Configuration' section contains the following fields:

- Template Name \*: test
- Template Description: Please describe template
- Callback Key: Enter a callback key (composed of uppercase and lowercase letters, digits, underscores, and dashes)
- Push Callback: Enter a push callback URL (header: http, https, etc)
- Interruption Callback: Enter an interruption callback URL (header: http, https, etc)
- Recording Callback: http:// (highlighted with a red box)
- Screencapture Callback: Enter a screencapture callback URL (header: http, https, etc)
- Porn Detection Callback: Please enter a porn detection callback URL (header: http, https, etc)

At the bottom right, there are 'Save' and 'Cancel' buttons.

通过云点播控制台获取：在 [云点播控制台](#) 进行查询，具体请参见 [查看视频](#)。

通过云点播 API 获取：调用 [SearchMedia](#) 接口进行文件查询。

## 步骤5：处理录制文件

### 方案一：直播录制 + 自动转码 + 视频加速播放


**适用场景：**直播录制后快速将录制文件自动进行转码和视频加速，供用户进行点播播放，适用于大部分不需要视频二次加工的直播场景。


#### 操作流程：


1.1 直播推流录制前先创建录制模板，单击**高级配置**进行任务流的配置。


Create Recording Template
Bind Domain Name

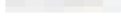
Create Template














### Recording Configuration

Template Name \*   
Only support letters, digits, underscores, and dashes.

Template Description   
Only support letters, digits, underscores, and dashes.

Recording Format \*  HLS  FLV  MP4  AAC (i)

▼ Audio/Video - HLS

Max Recording Time Per File  min

Resumption Timeout  sec  
This value will affect when to generate a recor

Storage Period  Permanent  Custom

VOD Subapplication/Category  Other

Advanced Configuration ▼

Save
Cancel

1.2 绑定预先在云点播控制台创建的任务流模板。

### VOD Task Flow ✕

You can click a task flow to go to the VOD console to add/modify its settings, and [refresh](#) here.

Name	Task Flow T...
<input type="radio"/> LongVideoPreset	Preset
<input type="radio"/> SimpleAesEncryptPreset	Preset

Total items: 2      10 ▾ / page      ⏪ ⏩ 1 / 1 page

After the template is bound with a VOD task flow, the flow will be executed after recording files are generated, which incurs VOD costs. For detailed pricing, see [VOD Billing Overview](#).

Save
Cancel

1.3 客户进行直播推流，具体可参见 [直播推流](#)。

1.4 直播录制完成后，获取点播 FileId。

### Callback information ✕

Callback request

```
{"duration":60,"end_time":1629695887,"end_time_usec":634354,"event_type":100,"file_format":"mp4","file_id":"BAK_3701925923130574711","file_size":15239029,"media_start_time":4184336,"record_ops":0,"record_file_id":"BAK_3701925923130574711","sign":"ec4a9c9431b3ea34ab273915135491db","start_time":1629695838,"start_time_usec":491893,"stream_id":"2102102","stream_params":{"bitrate":102,"task_id":"1722212987248255541","video_id":"1258442684_ee428fe56ba14e7"
```

Response message

```
<head><title>302+Found</title></head>
<body+bgcolor="white">
<center><h1>302+Found</h1></center>
<hr><center>bfe/1.0.8.18</center>
</body>
</html>
```

1.5 获取播放地址进行播放。

## 方案二：直播录制 + 手动转码 + 视频加速播放

**适用场景：**部分用户希望将直播录制的视频仅先存储到点播，且先不进行后续的转码操作，则可在新建录制到点播时，不添加其他操作。如果后续希望对视频进行转码，则可进行手动触发转码操作。同时，可配合点播云剪辑功能进行使用，效果更佳。

### 操作流程：

- 1.1 客户进行直播推流，具体可参见 [直播推流](#)。
- 1.2 文件自动录制到点播。
- 1.3 获取点播 FileId。
- 1.4 配置转码模板，或者任务流进行手动转码，具体可参见 [模板设置](#)。

Task Flow Name   
It should be a combination of letters, numbers, hyphens (-), and underscores (\_) with a length up to 20 chars

Task flow Description   
Up to 15 chars for description

Configuration Item  General Transcoding  TESH D Transcoding  Remux  Adaptive Bitrate Streaming  Screenshot Task  Capture Cover Task  Dynamic Image Task  V  
Select at least one configuration item

**Transcoding Task Configuration**

You can go to "Template Settings - Transcoding Template" to create a transcoding template, and go to "Template Settings -&gt; Watermark" to create a watermark template. After creation, click here to [Purge](#).

Transcoding Template/ID	Encapsulation For...	Video Encoding	Video Bitrate	Resolution	Frame rate	Audio Encoding
The current list is empty						

[Add transcoding template](#)

- 1.5 客户可择需进行视频二次剪辑。
- 1.6 转码和处理完成后获取视频地址进行后续播放。

## 方案三：直播录制 + 自适应码流 + 视频加速 + 播放器

**适用场景：**部分用户对视频安全有极高的诉求，普调的 HLS 加密无法满足加密诉求，通过自适应和播放器 SDK 的组合使用，可以有效完成视频安全升级，对在线教育、企业培训类客户场景适配度很高。

### 操作流程：

- 1.1 客户进行直播推流，具体可参见 [直播推流](#)。
- 1.2 文件自动录制到点播。
- 1.3 获取点播 FileId。
- 1.4 配置任务流出自适应码流，具体可参见 [任务流设置](#)。

Task Flow Name   
 It should be a combination of letters, numbers, hyphens (-), and underscores (\_) with a length up to 20 chars

Task flow Description   
 Up to 15 chars for description

Configuration Item  General Transcoding  TESHD Transcoding  Remux  Adaptive Bitrate Streaming  Screenshot Task  Capture Cover  
 Select at least one configuration item

---

**Adaptive bitrate streaming task configuration**

You can go to "Template Settings - [Adaptive Bitrate Streaming Template](#) to view adaptive bitrate streaming template. Template creation is not supported for the moment.

Adaptive Bitrate Streaming Template/ID	Muxing Type	Substream Count	Switch from Lo
The current list is empty			

[Add Template](#)

1.5 设置播放器配置，选择所创建的用于播放的自适应码流。

**Adaptive bitrate streaming template for playback**

You can go to "Template Settings - [Adaptive Bitrate Streaming Template](#)" to create a Adaptive bitrate streaming template. After creation, click [Purge](#)

Adaptive Bitrate Streaming Template/ID	Encryption Type	Muxing Type	Substream Cou
Adpative-HLS(10)	Not encrypted	HLS	6

1.6 通过 FileId 进行视频的播放。

# 如何在直播过程中实现回看功能

最近更新时间：2023-07-26 16:12:27

时移回看依托直播录制能力，结合直播时移和点播的加速分发，实现在直播过程中进行回看的功能，用户可以实时选择从开播后的某个过往时间点开始进行回看，从而达到播放之前直播内容的效果。常用于赛事直播中的精彩片段回看，用户可以直接拖动进度条，进行回看，而不必等到直播结束，回看过程中直播流保持不变，用户可以在回看过程中切回直播。

## 功能特性

允许指定回看的相对时移时长（即回看的时间起点与当前的时间差）。

当直播流使用多码率录制时，允许指定指定码率的视频流。

## 前提条件

[注册](#) 并 [登录](#) 腾讯云账号。

已开通腾讯云直播服务，并添加 [推流域名](#)。

## 注意事项

使用时移回看功能的前提是目标直播流开启了 [时移回看](#) 功能。

相对时移时长当前最小为90秒，即回看的内容相较直播内容有90秒以上的延迟。

开启时移功能需要使用直播录制功能，会产生 [点播存储费用](#) 和视频播放的 [流量费用](#)。

## 实践步骤

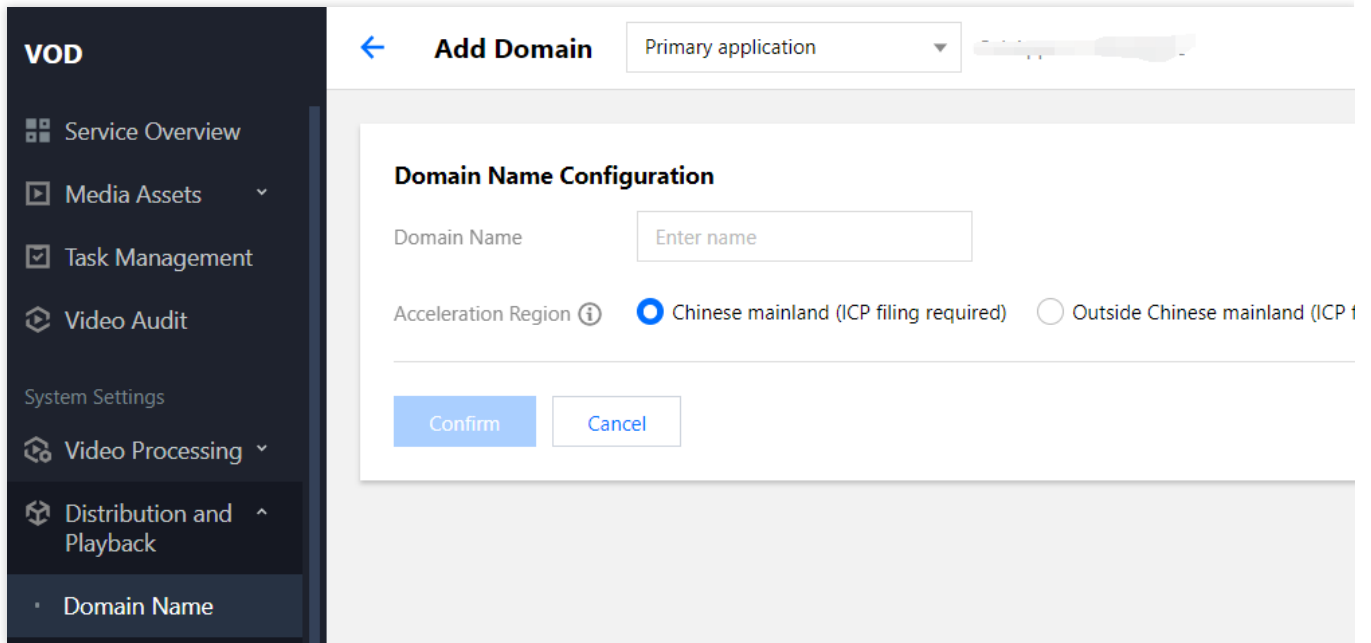
### 步骤1：开通云点播服务

1. 进入 [云点播控制台](#)，单击**立即开通**。
2. 勾选开通服务协议，单击**确定**即可成功开通云点播服务，并访问云点播控制台。

### 步骤2：添加时移播放域名

您需要添加用于时移播放的云点播域名，具体操作如下：

1. 在 [云点播控制台](#) > [应用管理](#)，单击应用名称进入应用列表页后，单击左侧导航栏里[分发播放管理](#) > [域名管理](#)。
2. 单击**添加域名**并填写已备案的云点播域名，具体操作请参见 [分发播放设置](#)。



3. 完成新增域名 CNAME 配置操作。

### 步骤3：关联录制模板

1. 进入 [云直播控制台](#)，单击功能配置 > [直播录制](#)。
2. 单击[创建模板](#)，具体操作请参见 [录制模板配置](#)。



### Recording Configuration

Template Name \*   
Only support letters, digits, underscores, and dashes.

Template Description   
Only support letters, digits, underscores, and dashes.

Recording Format \*  HLS  FLV  MP4  AAC ⓘ

▼ Audio/Video - HLS

Max Recording Time Per File  min

Resumption Timeout  sec  
This value will affect when to generate a recording file.

Storage Period  Permanent  Custom

VOD Subapplication/Cate gory

[Advanced Configuration](#) ▼

#### 注意：

选择录制文件类型为 HLS 格式，开启 HLS 录制。

自定义文件存储时长，且该时长不得短于 [时移时长](#)。

3. 将录制模板关联到所需配置的推流域名，具体操作请参见 [录制配置](#)。

#### 步骤4：开通时移服务

您可 [提交工单](#)，并选择问题所属产品为“云直播”，提出开通时移服务申请要求，并提供以下参数：

[步骤2](#) 中已添加的云点播 时移播放域名。

[步骤3](#) 中的录制模板 ID。

自定义时移时长 timeshift\_dur，单位：秒。

#### 说明：

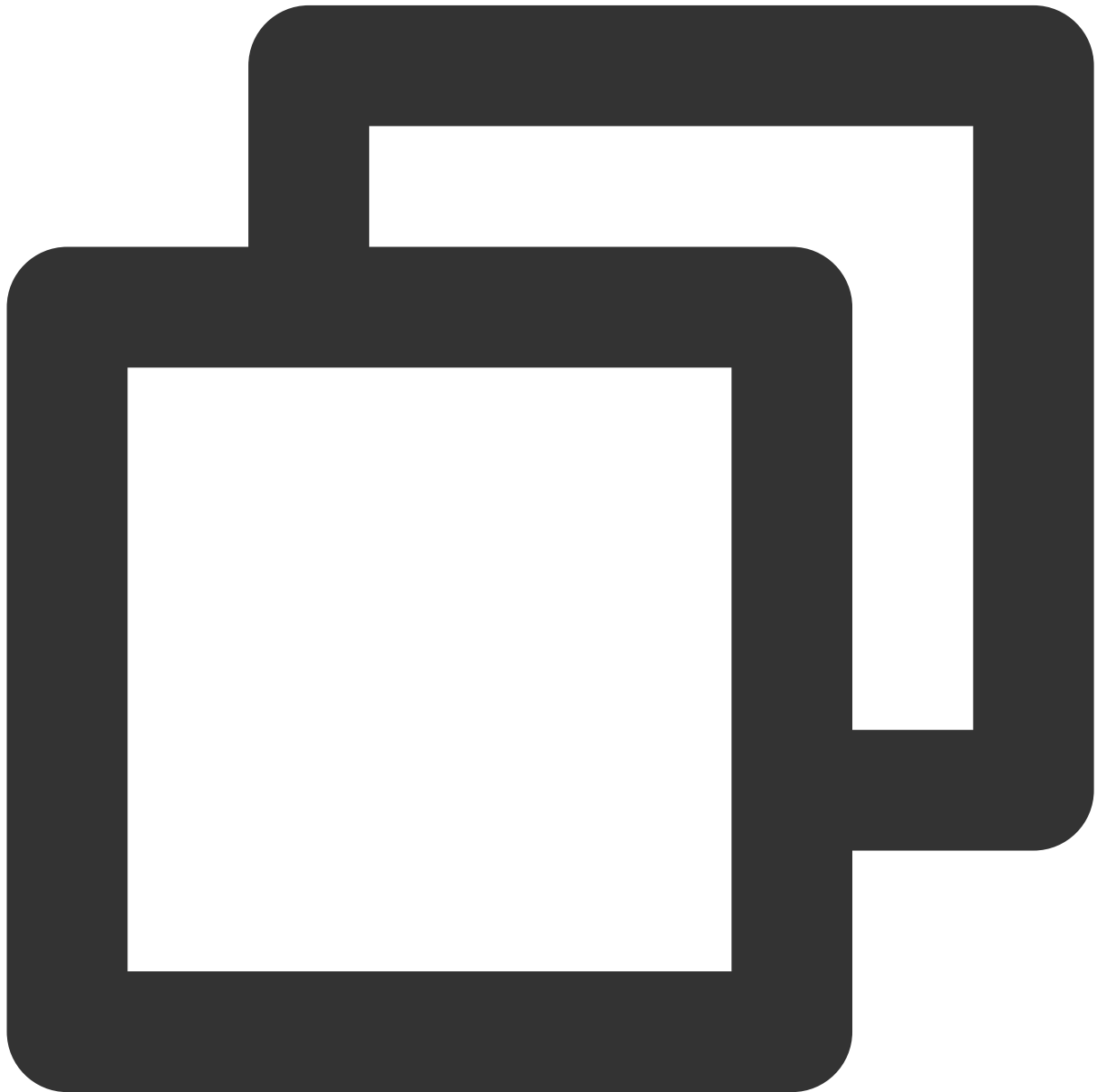
时移时长：指时移可以看多长时间的内容（目前最长可配置时移7天内的内容）。

此项并不保证绝对精准，按需求配置的前提下，建议多加一小段时间。

假配置为7200（2小时）：表示当前可请求2小时前到现在的时移内容（即 `delay` 相对时移时长可为90s - 2小时）；时移超过2小时前的内容，即使有直播内容，也会返回 HTTP 404。

## 播放请求

### 请求 URL 格式



```
http://[Domain]/timeshift/[AppName]/[StreamName]/timeshift.m3u8?delay=xxx
```

## 参数说明

参数	说明
[Domain]	您注册的时移服务访问域名，即您在云点播控制台上添加的时移播放域名。
timeshift	固定字段，无须修改。
[AppName]	应用名称，若您的应用名称是 live，则填 live。
[StreamName]	流名称，填您请求对应的流名称。
timeshift.m3u8	固定字段，无须修改。
delay	表示相对时移时长，单位秒。目前该值小于90，后台可调整成90。

## 示例

假设当前时移域名为 `testtimeshift.com`，时移应用名称为 `live`，流名称为 `SLPurIFzGPE`，若您需要时移该地址5分钟前的直播内容，则请求 URL 如下：

```
http://testtimeshift.com/timeshift/live/SLPurIFzGPE/timeshift.m3u8?delay=300
```

# 如何将点播视频转为类直播效果

最近更新时间：2023-05-15 17:15:06

伪直播依托于点播的播放控制能力，将点播文件增加“限制观看时间”和“同步观看进度”两种访问控制的功能，使点播文件达成类直播效果，用户可以首先生成点播文件，在指定的直播时间使用点播文件进行类直播分发，有效的降低直播的风险与成本。

## 功能特性

特性	说明
开发成本低	如果选择将点播视频转为标准的直播进行分发，那么用户需要将视频经由 OBS 软件推流到直播平台，并对接整套直播系统，开发成本高。相比之下，伪直播都在云点播内部实现，用户只需要启用转码和防盗链功能即可。
违规风险低	使用伪直播能力，用户可以提前对自己的点播文件进行审核和编辑，有效规避直播过程中可能涉及的违规风险，规避不合规内容，提升直播质量。
创建简单灵活	没有直播房间的概念，任何视频都可以随时生成伪直播。 没有并发数限制，可以指定好开播时间并预先分发观看链接。

## 应用场景

主要应用于视频需要提前录制完毕，按照预定的时间安排让用户同步观看。用户可以提前获取到观看入口，但在预定时间点开始之前无法观看到视频。

伪直播进行过程中无法快进，常见于在线教学视频、直播晚会和广电等行业。

## 使用说明

### 使用限制

伪直播本质上是点播，因此并不具备标准直播的一些能力，例如：

不支持对“一场”伪直播进行数据统计。

无法感知“一场”伪直播的开始和结束。

不支持对正在进行的伪直播做暂停/终止等操作。

不支持对已分发出去的伪直播链接进行禁用。

不支持动态改变视频内容（例如实时转码、打水印等）。

## 名词解释

**限制允许观看的时间**：视频入口可以提前分发给观众，在伪直播开始之前或结束之后，观众都无法观看；只有在伪直播“进行”过程中才能够观看。

**同步观看进度**：在伪直播“进行”过程中，所有观众的观看进度都是同步的（存在分钟级别的偏移）。

## 前提条件

注册并登录腾讯云账号，并且完成账号实名认证。

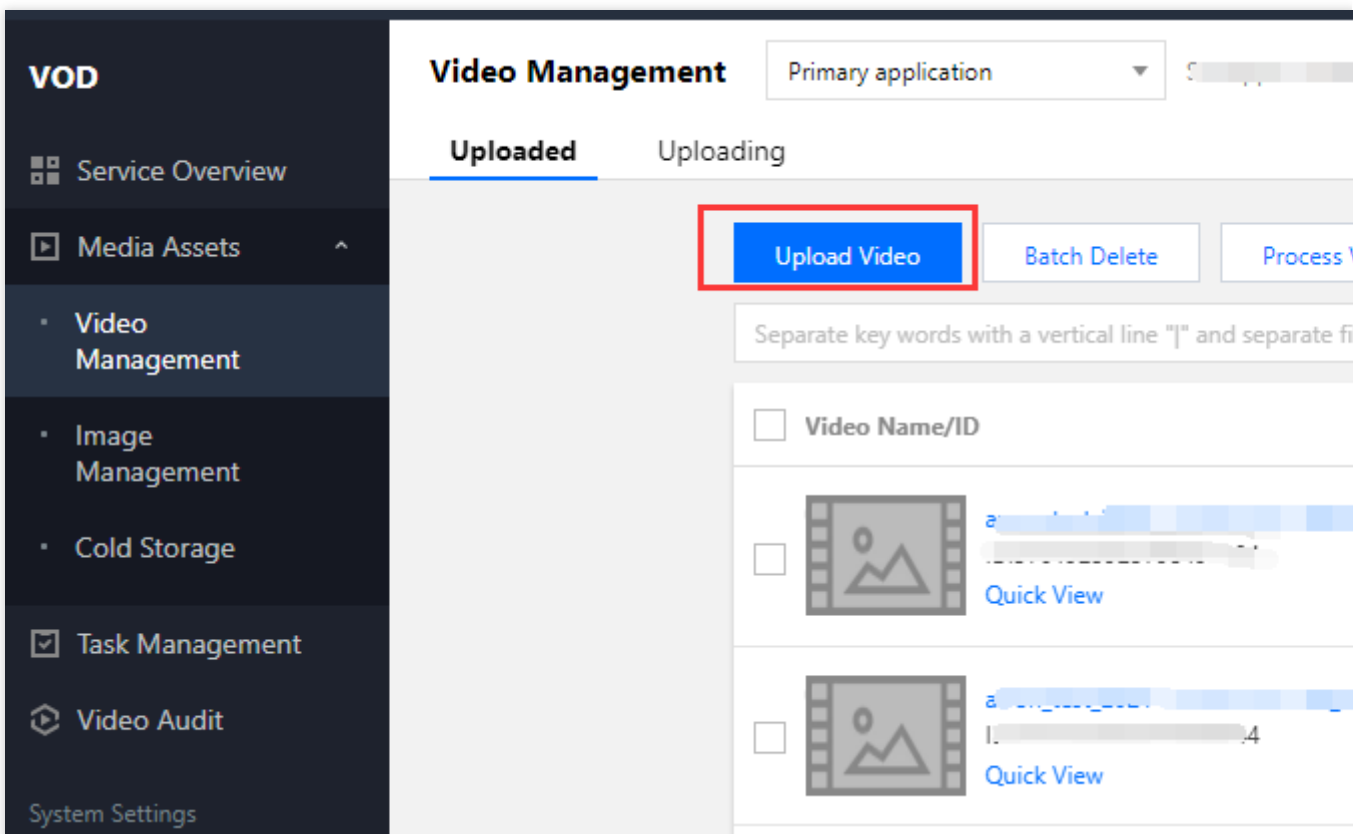
已开通腾讯云直播和云点播服务。若未开通，请前往开通[云直播服务](#)和[云点播服务](#)。

进行直播录制，详情请参见[直播录制转点播](#)。

## 实践步骤

### 步骤1：上传视频到云点播

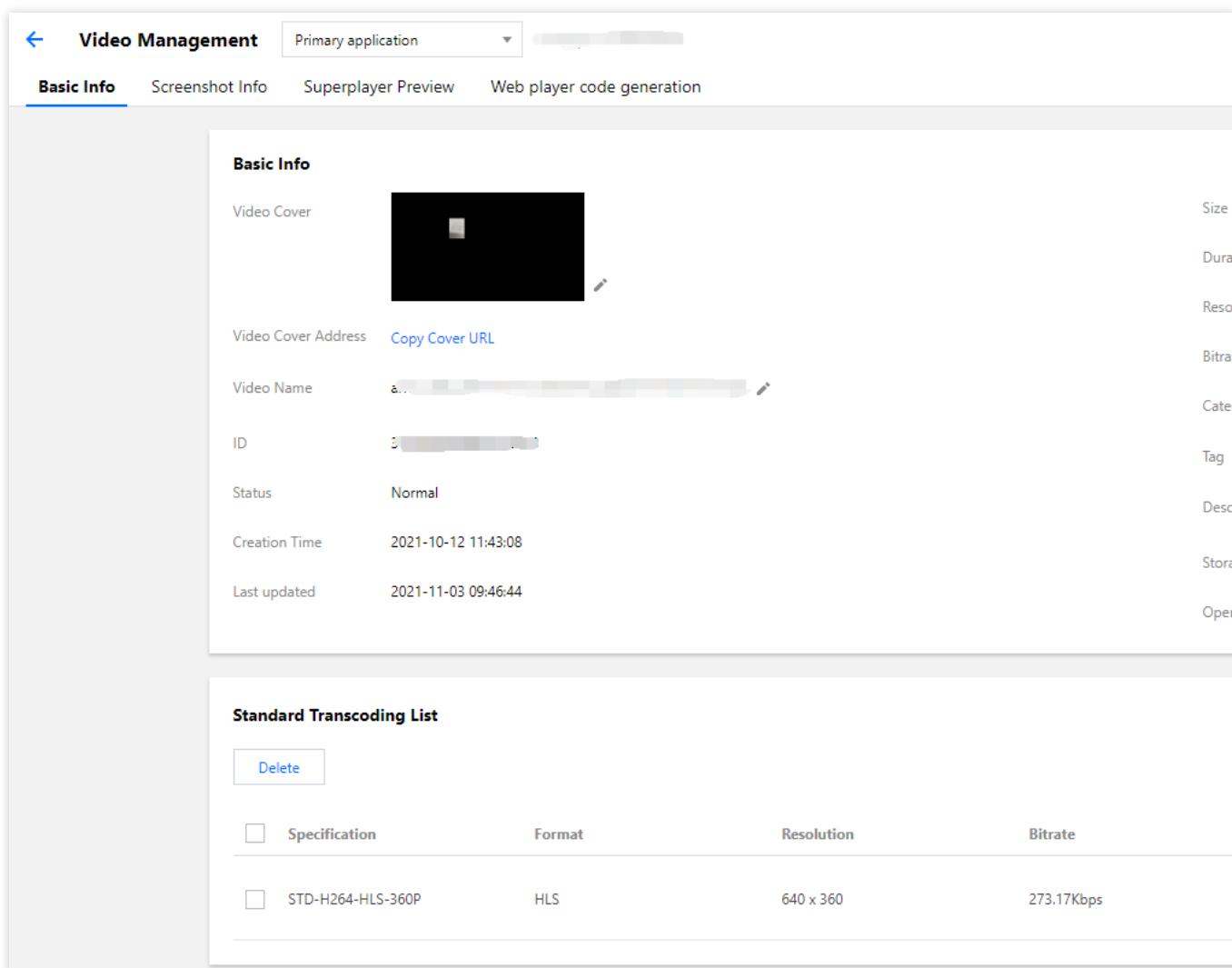
1. 登录[云点播控制台](#)，单击左侧导航栏**应用管理**，进入应用列表页。
2. 找到需要处理音视频文件的应用，单击应用名称进入应用管理页。
3. 默认进入**媒资管理 > 音视频管理**页面，单击**上传音视频**。



您也可根据业务情况选择合适的方法将视频文件上传到云点播，更多上传方式可参见 [媒体上传综述](#) 和 [直播录制转点播](#)。

## 步骤2：将视频转码为 HLS

1. 使用伪直播必须基于 HLS 格式，您可以按照 [转码任务发起](#) 的说明将已上传的视频转码为 HLS（具体的 [转码模板](#) 请根据业务进行选择）。
2. 转码完成后，在控制台 [音视频管理](#) 页面 [查看](#) HLS 的 URL，或者通过接收 [事件通知](#) 的方式获取 HLS 的 URL。



## 步骤3：开启 Key 防盗链

1. 使用伪直播必须开启防盗链，请登录 [云点播控制台](#)，在左侧导航栏的 [应用管理](#) 内单击目标应用名称，选择 [分发播放设置](#) > [域名管理](#)，单击目标域名所在行的 [设置](#)，进入域名相关设置页面。

**VOD**

- Service Overview
- Media Assets
- Task Management
- Video Audit
- System Settings
- Video Processing
- Distribution and Playback
- Domain Name

**Domain Management** Primary application

[Add Domain](#)

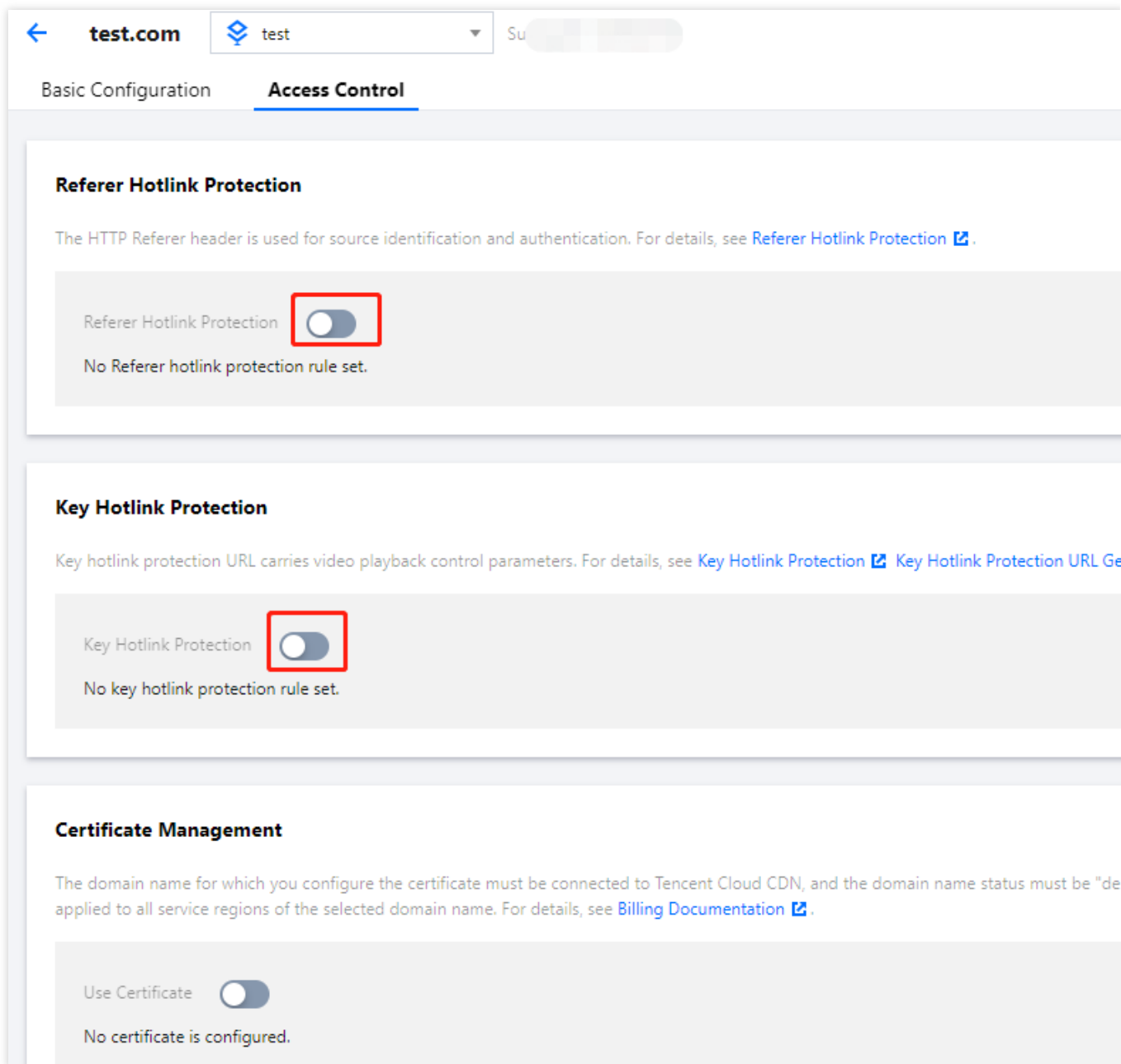
Domain Name	Status	CNAME <sup>①</sup>
[redacted].com <a href="#">Default Distribution Domain</a>	Enable	1 [redacted]

Total items: 1

**FAQs:**

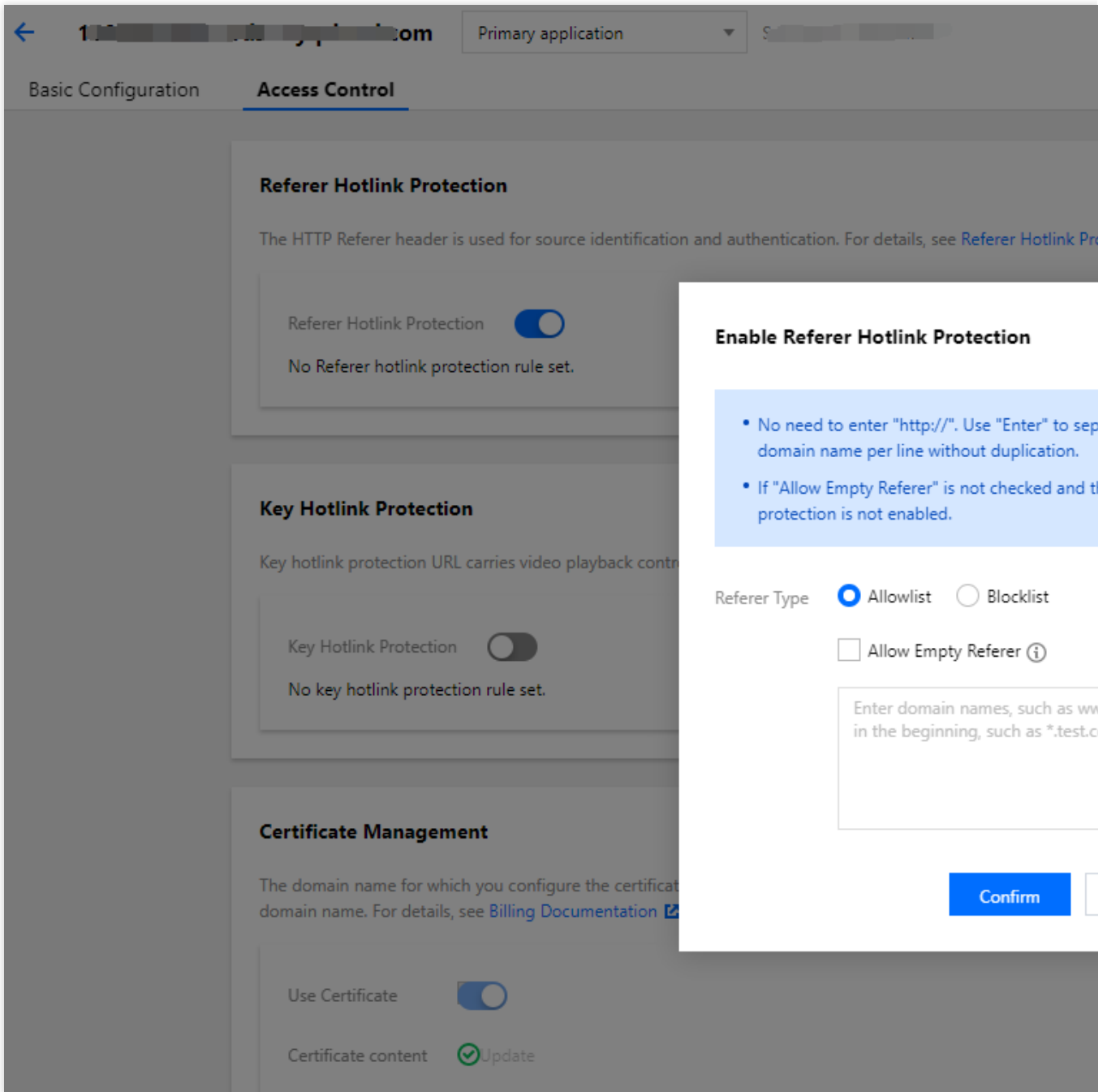
[Domain Name Guide](#) [CNAME Guide](#)

2. 在“访问控制”页面可开启 Referer 防盗链、Key 防盗链。



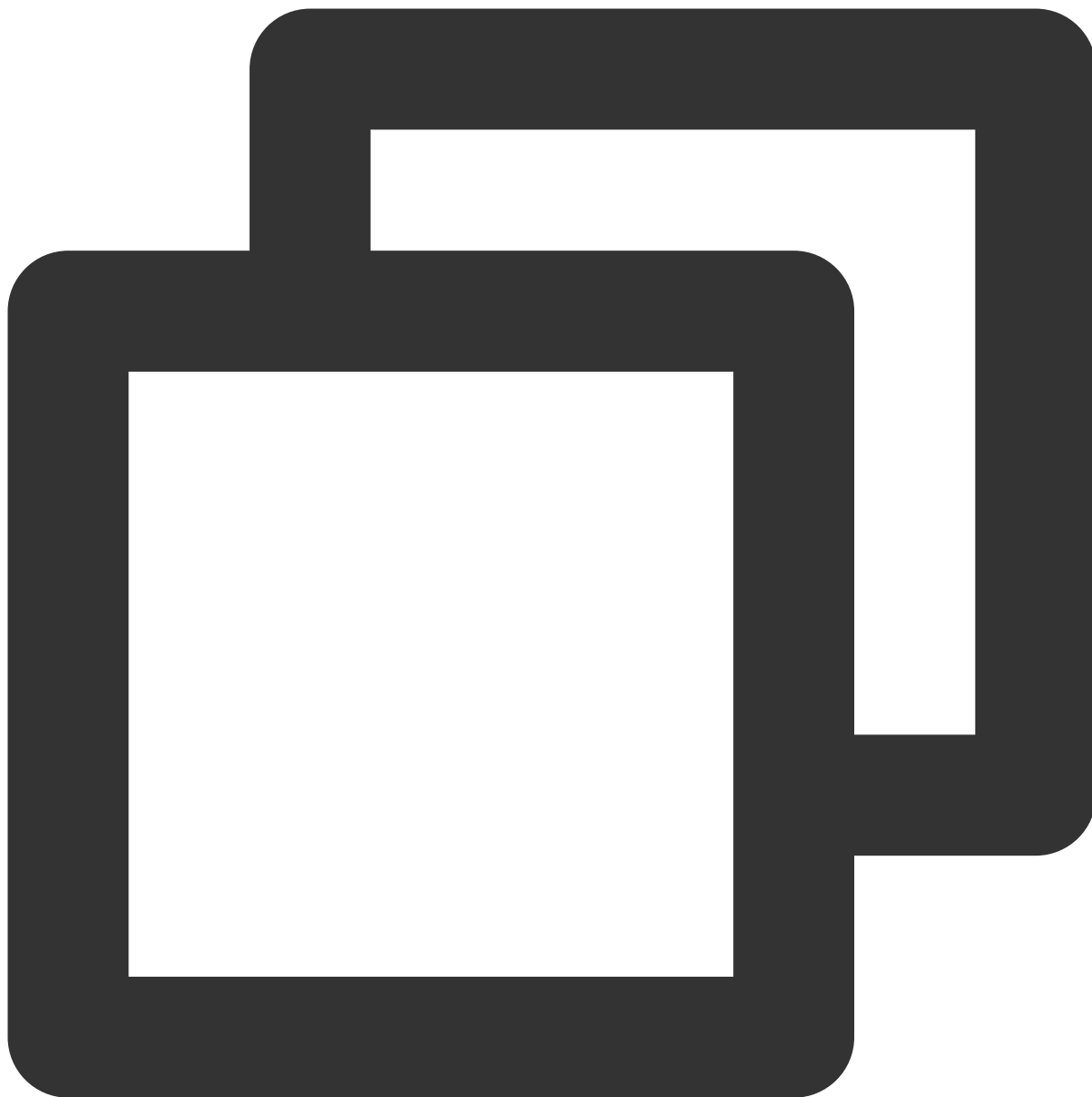
3. 开启后如下图所示。更多选项说明，请参见 [设置防盗链](#)。完成设置后，请保存防盗链 KEY 的内容用于以下的防盗链签名计算。





#### 步骤4：计算防盗链签名

签名计算公式



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

**注意：**

伪直播的防盗链参数比 [标准的 Key 防盗链](#) 多一个参数：`plive`。在计算防盗链签名时，`plive` 需要参与计算。

参数名	取值	说明
KEY	11111111	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。
plive	5e344f00	表示该场伪直播的开始时间（北京时间），以 UNIX 时间戳的形式表示。例如1577808000表示2020-01-01 00:00:00这个时间点。
exper	0	试看时长，0 表示不限制。
rlimit	0	观看 IP 数限制，0 表示不限制。
us	test	生成的随机字符串。

例如，某个开发者在云点播有个视频，其 HLS URL（而非原始视频的 URL）播放地

址：`http://1250000000.vod2.myqcloud.com/vodtranscq125000000/12345678/v.f240.m3u8`；现有如下需求：

HLS URL 防盗链 KEY 为 `11111111`。

有效截止时间 t 为 `5e5a8a80`（即2020-03-01 00:00:00）。

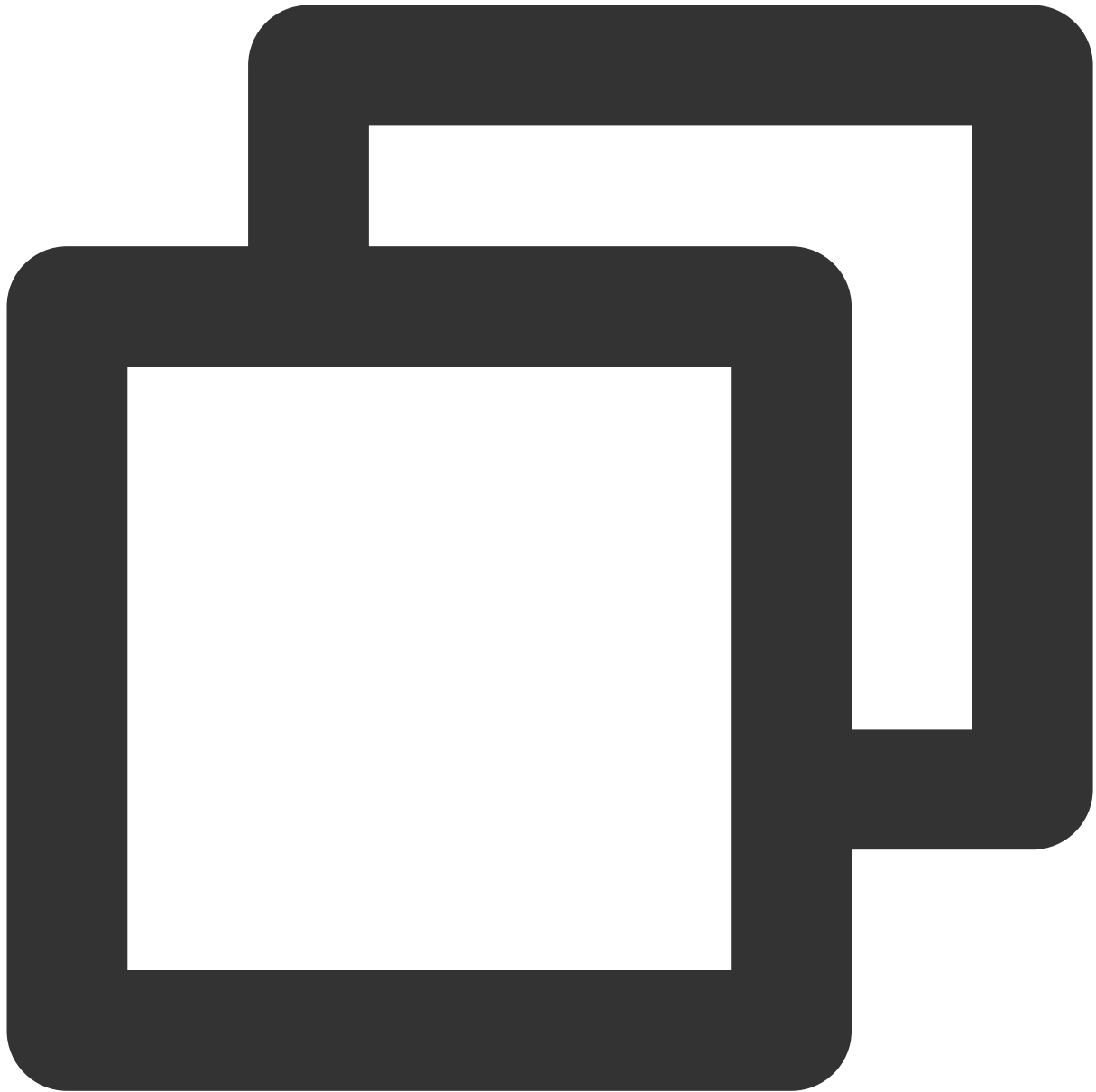
伪直播开始时间 plive 为 `5e344f00`（即2020-02-01 00:00:00）。

试看时长 exper 为 `0`（即不限制）。

观看 IP 数限制 rlimit 为 `0`（即不限制）。

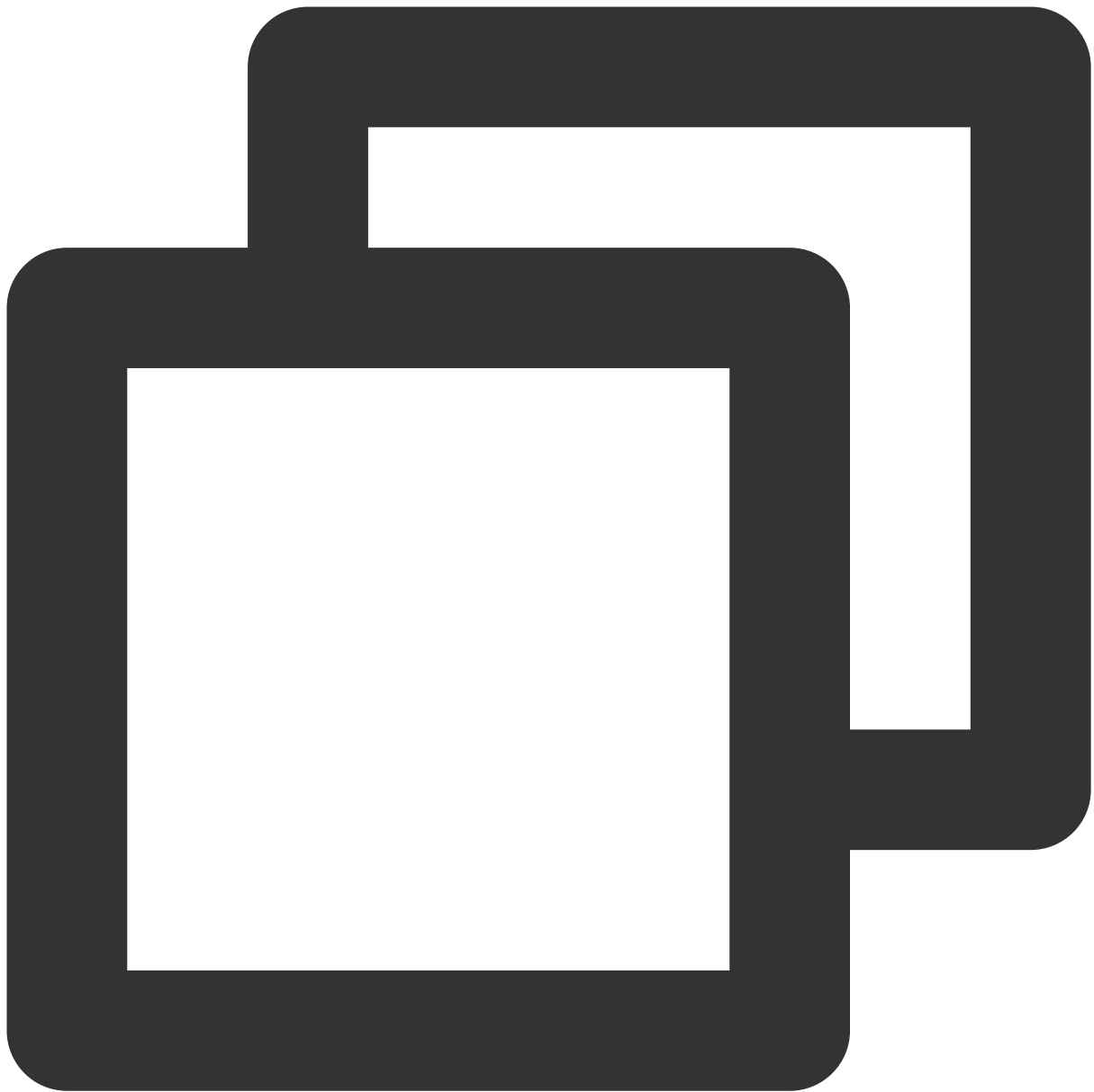
随机字符串 us 为 `test`。

1. 根据 [签名计算公式](#) 来计算签名：



```
sign = md5(111111111/vodtranscq125000000/12345678/5e5a8a805e344f0000test) = 0af5018d
```

2. 将计算后的签名拼接到 HLS URL 的 QueryString 中，最终生成的防盗链链接：



```
http://1250000000.vod2.myqcloud.com/vodtranscq125000000/12345678/v.f240.m3u8?t=5e5a
```

#### 说明：

防盗链链接 QueryString 中参数顺序需要和计算 `sign` 时一样，即：`t-plive-exper-rlimit-us-sign`。为了方便开发者调试，我们提供了 [防盗链签名生成工具页面](#)。按页面内容填写参数后即可查看签名计算的中间结果和最终防盗链链接。

## 体验伪直播

---

在支持 HLS 播放的播放器中（例如 Safari 浏览器、VLC、PotPlayer 等。）访问上述防盗链链接，即可直接体验。

**注意：**

Chrome 浏览器默认不支持 HLS，需要安装插件。

# 如何使用云点播和 WordPress 搭建网站

最近更新时间：2022-11-02 16:27:12

## 简介

WordPress 是一款著名的开源博客框架和内容管理系统，使用它可以极大的降低开发者建站的难度，提高建站效率。使用 WordPress 提供的默认方案可以让一个小型的网站在几分钟的时间就运行起来，但随着网站的持续运营，图片和音视频等媒体文件的访问速度以及它们给服务器带来的带宽和存储量的增涨就会让您的站点开始表现得难堪重负。如何妥善地安置这些网站中举足轻重的媒体文件，是一个站点想要长期运营无法绕开的问题。

## 建站为何要使用云点播

使用 WordPress 默认方案搭建的网站之所以不利于长期运营，主要原因在于媒体文件是直接存放在站点的服务器上，它将带来以下问题：

序号	详情
1	通过网页直接访问服务器上的媒体文件，容易出现视频卡顿和图片加载缓慢的现象。
2	通过网站访问媒体文件会占服务器的带宽，且很容易把带宽占满。
3	媒体文件的体积一般都比较大，随着网站内容不断丰富，很容易把服务器的存储空间占满。
4	站点服务器迁移时，媒体文件迁移困难。

云点播是一款为音视频和图片等媒体提供存储、转码和加速分发播放等能力的云服务，它可为您建站提供以下帮助：

序号	详情
1	上传和存储。您可以使用我们提供的插件或 SDK 将文件上传到云点播，不再占用服务器的存储空间，且避免了服务器迁移、文件误删等问题。
2	加速访问。云点播会为您提供文件的 CDN 地址，访问音视频和图像时不再出现卡顿现象，且不占用服务器带宽。

除此之外，点播还提供访问控制、媒体处理和智能审核等能力，详情请参见云点播的 [功能概述](#)。

## 云点播的 WordPress 插件

云点播提供了 [WordPress 插件](#)，您只需安装该插件，并通过简单的配置关联您的 [云点播子应用](#)，即可将 WordPress 上的媒体文件托管到云点播上，并在您的网站中通过点播提供的 CDN 地址访问您的文件。该插件目前具有以下功

能：

- 绑定云点播子应用。
- 在 WordPress 的管理端上传视频文件，可自动上传至云点播的子应用下。
- 在文章或页面上插入视频时，将自动使用云点播上的视频地址。

## 准备工作

### 1. 创建点播子应用

点播子应用可帮助您实现资源隔离，例如您可将不同站点的媒体文件划分到不同的子应用下，单独管理。创建方法详见 [开通子应用](#)。

### 2. 搭建 WordPress 网站

您可以根据实际情况选择一种方案来搭建 WordPress 网站：

- 方案一（推荐）
- 方案二

使用腾讯云轻量应用服务器或云服务器的 WordPress 应用镜像快速建站。

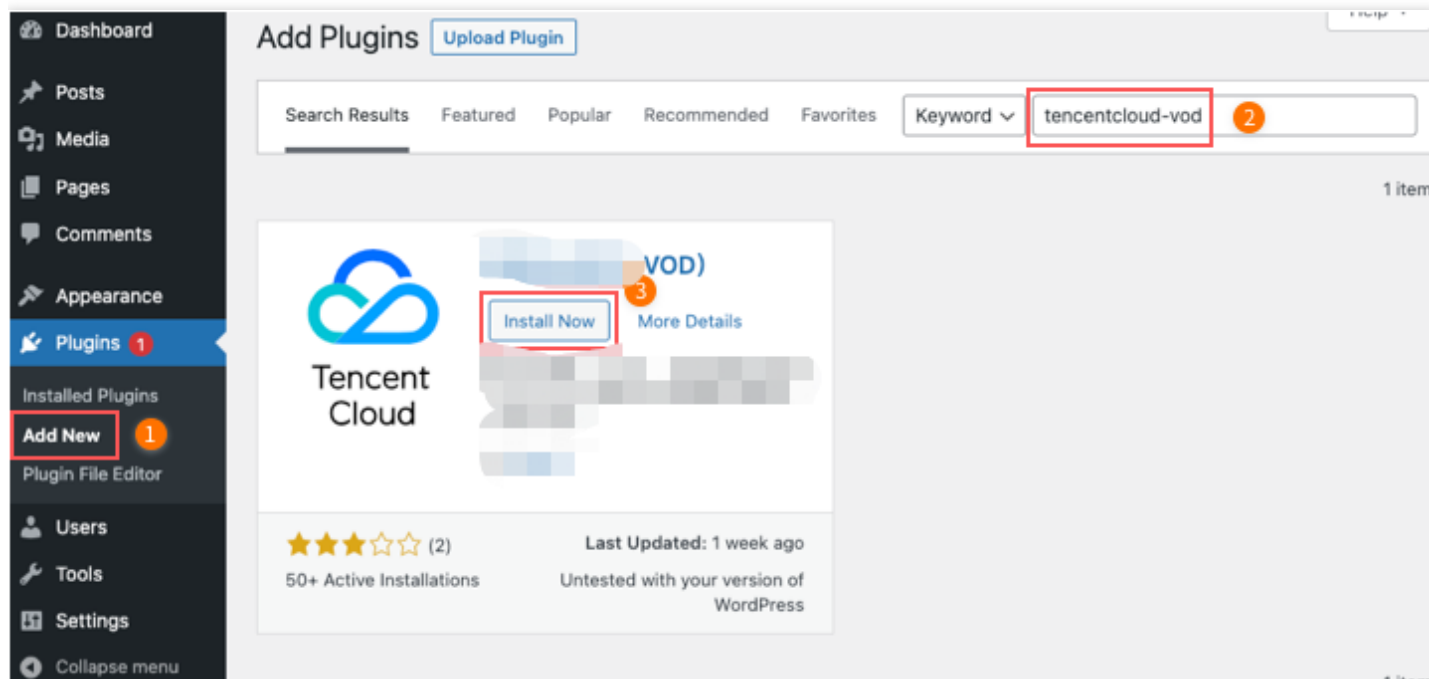
## 操作步骤

### 步骤1：安装插件

- 方式一
- 方式二

在线安装（推荐）。在搜索框中搜索 **tencentcloud-vod**，单击 **立即安装 > 启用**。





## 步骤2：配置插件

单击 WordPress 左侧导航栏 腾讯云设置 > [云点播](#)，可对点播插件进行配置。

配置说明见下表：

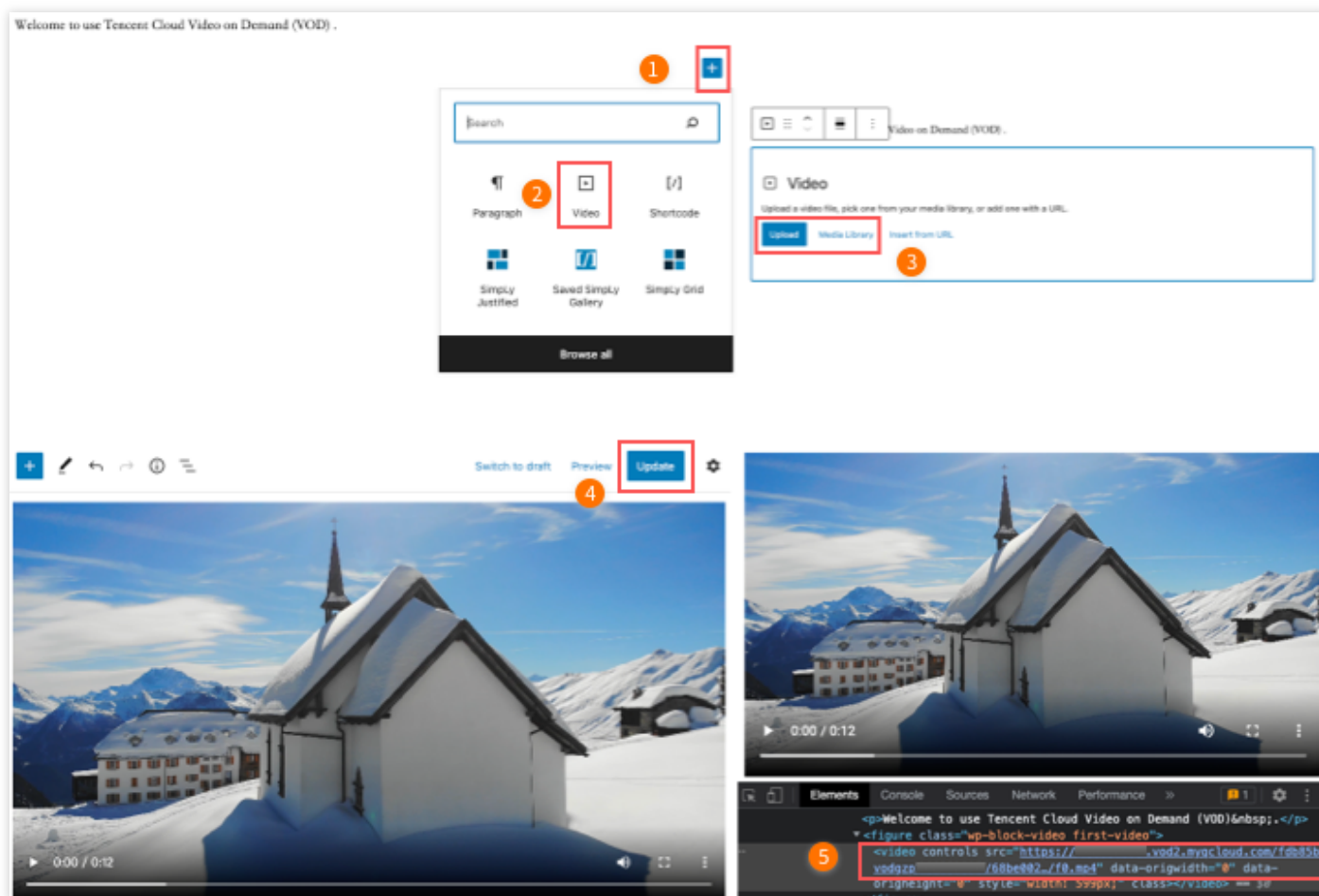
配置项	说明
自定义密钥	不开启该选项时，插件会使用 <a href="#">腾讯云设置</a> 中配置的密钥，即与其它腾讯云插件共用同一个密钥。开启该选项时，则会使用下方指定的密钥。
SecretId、SecretKey	访问密钥信息，可前往 <a href="#">云 API 密钥</a> 中创建和获取。
SubAppID	云点播的子应用 ID，填上准备工作中创建的子应用 ID 即可，也可在点播 <a href="#">应用管理</a> 中查询到。
是否开启转自适应码流	开启后，会在上传时对视频进行转自适应码流。详情见点播 <a href="#">自适应码流</a> 。

配置完成后，单击 **保存设置** 即可。

## 步骤3：上传并使用点播视频

1. 开启并配置好点播插件后，在媒资库和编辑文章或页面时上传的视频文件，就会自动上传到云点播。
2. 以编辑文章时上传为例，如下图所示，在文章中插入“视频区块”，然后单击**上传**，选择视频文件并上传。接着单击文章右上角的**预览**查看文章的预览页面，通过检查视频的网址可以发现，此时使用的已经是云点播上的视频

了。



3. 如下图所示，左侧使用的是服务器上的视频，而右侧使用了云点播。播放服务器上的本地视频会因为带宽限制和没有 CDN 加速服务而频繁卡顿，使用云点播则可以轻松地解决这些问题。



## 扩展

### 使用 HTTPS

---

云点播默认使用 HTTP 协议，若需使用 HTTPS 可在云点播控制台选择该子应用，然后打开 **系统设置 > 分发播放设置 > 默认分发配置**，即可将**默认分发协议类型**修改为 HTTPS。

# 如何进行自定义源站回源

最近更新时间：2023-03-07 15:49:29

## 简介

自定义源站回源依托于云点播 CDN 能力，通过创建自定义域名并配置回源信息，用户可将存储在第三方源站媒体文件借助云点播进行加速分发，提供给用户在多云场景下的媒体分发方案。本文将介绍如何配置和使用云点播自定义源站回源能力。

## 使用场景

- **源站迁移成本高**：用户的媒体存储在其他第三方云上，其源站内容与第三方平台高度耦合难以做源站的迁移，通过自定义回源可以不迁移源站使用点播 CDN 进行内容加速分发。
- **播放质量要求高**：用户的内容对网络时延和卡顿有较高的要求，使用其他第三方平台难以满足用户需求，通过自定义回源可以保证客户的音视频播放更加流畅，保证业务服务质量。
- **多云共存降低风险**：用户的内容和业务需要多个内容加速渠道来保障业务的可靠性，提高业务容灾能力，可以选择腾讯云点播 CDN 进行音视频加速分发。

## 前提条件

1. [注册](#) 并 [登录](#) 腾讯云账号，并且完成账号实名认证。
2. 已开通腾讯云点播服务。若未开通，请前往开通 [云点播服务](#)。

## 支持回源源站类型

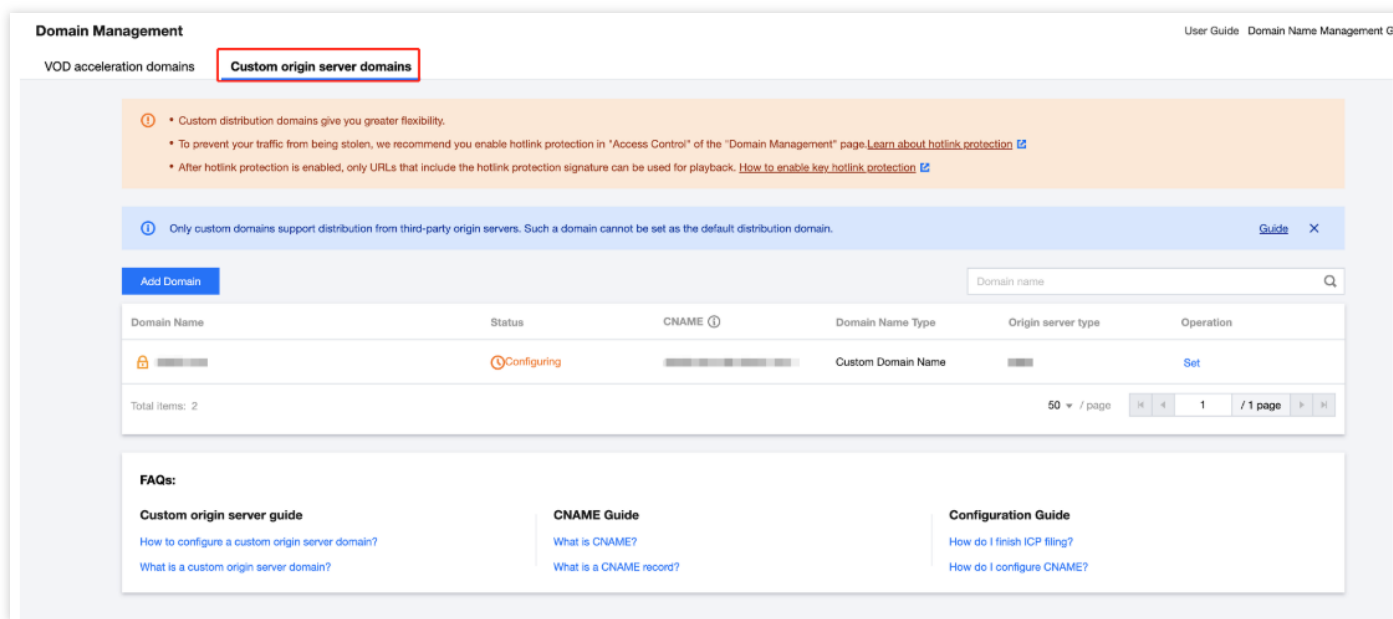
- 自有源站
- 第三方存储对象

## 实践步骤

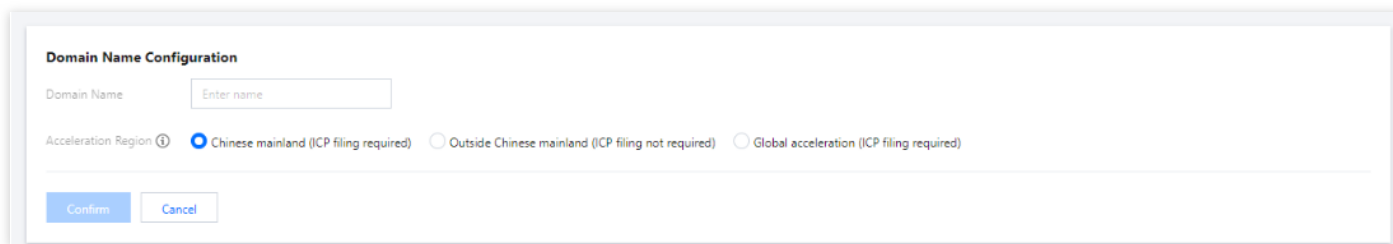
### 步骤1：创建自定义域名并配置回源信息

1. 自定义源站回源功能开启后，登录云点播控制台，选择左侧导航栏中[分发播放设置](#) > [域名管理](#)，进入[域名管理](#)页面。

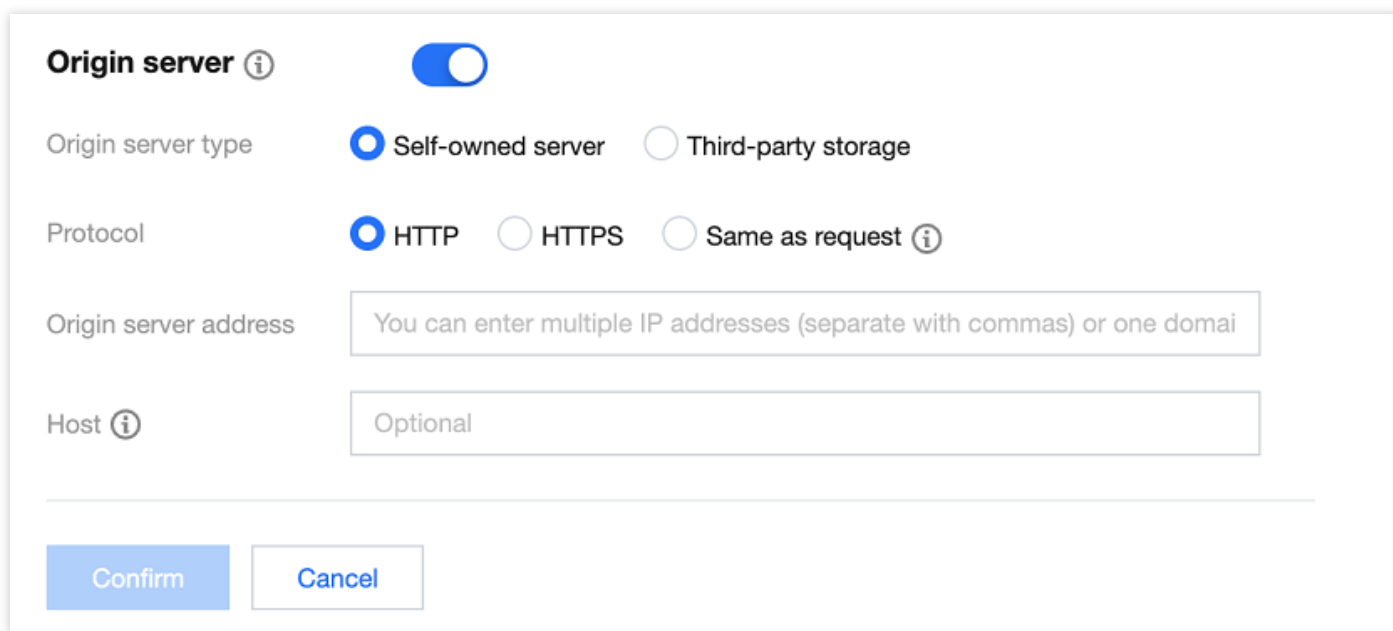
2. 单击顶部自定义源站加速域名，进入自定义源站加速域名页面。



3. 单击添加域名，填写已备案的域名并选择加速区域，具体操作请参见点播加速域名。



4. 根据用户实际回源需求填写源站配置，目前支持自有源和第三方对象存储回源方式，具体配置说明如下：



### 自有源站

若用户希望将稳定运行的业务服务器作为回源源站，借助云点播 CDN 对其上的媒体文件进行加速分发，则请按照以下方式进行源站配置：

配置项	描述
源站类型	选择 <b>自有源</b>
回源协议	根据源站支持情况，选择回源请求协议，支持 HTTP、HTTPS 以及协议跟随
源站地址	支持填入多个 IP 源站（逗号分隔）或一个域名源站
回源 HOST	自有源站可指定回源 HOST，回源 HOST 用于指定 CDN 节点在回源时，在源站访问的站点域名/IP的具体站点 若未指定回源 HOST，默认取当前所创建的加速域名

#### 说明

- 源站地址不支持使用云点播默认域名。
- 协议跟随可实现 HTTP 访问使用 HTTP 回源，HTTPS 访问使用 HTTPS 回源（源站需要支持 HTTPS 访问）。
- 支持选择域名作为回源地址，此域名不可与业务加速域名一致。

### 第三方对象存储

若用户希望将存储在第三方对象存储的媒体文件借助云点播 CDN 能力进行加速分发，则请按照以下方式进行源站配置：

配置项	描述
源站类型	选择 <b>第三方对象存储</b>
第三方对象存储	目前支持的第三方对象存储有阿里云 OSS 和 AWS S3
回源协议	支持 HTTP、HTTPS
源站地址	输入有效的存储桶访问地址作为源站（不可包含 http:// 或 https:// 协议头）

### Modify origin server settings ✕

Origin server type  Self-owned server  Third-party storage

Third-party storage  Qiniu Cloud  Baidu Cloud  Alibaba Cloud  AWS

Protocol  HTTP  HTTPS

Origin server address

Private bucket access [Grant access](#)  
If you use a private bucket as the origin server, you need to grant VOD access to that bucket.

Host ⓘ

若选择私有访问的第三方对象存储桶作为源站，需填写有效访问 ID 和 key 进行回源鉴权，鉴权通过后即开启私有存

储桶访问。请查看 [访问密钥获取指引](#)。

### Authorization ✕

- Make sure you enter a valid access ID and key.
- For security reasons, you cannot view the key after you submit the information. However, you can modify it.
- To protect your account, we recommend you configure permission-based key distribution at your vendor. [How to use a third-party storage service as the origin server?](#)

Access ID

Access key

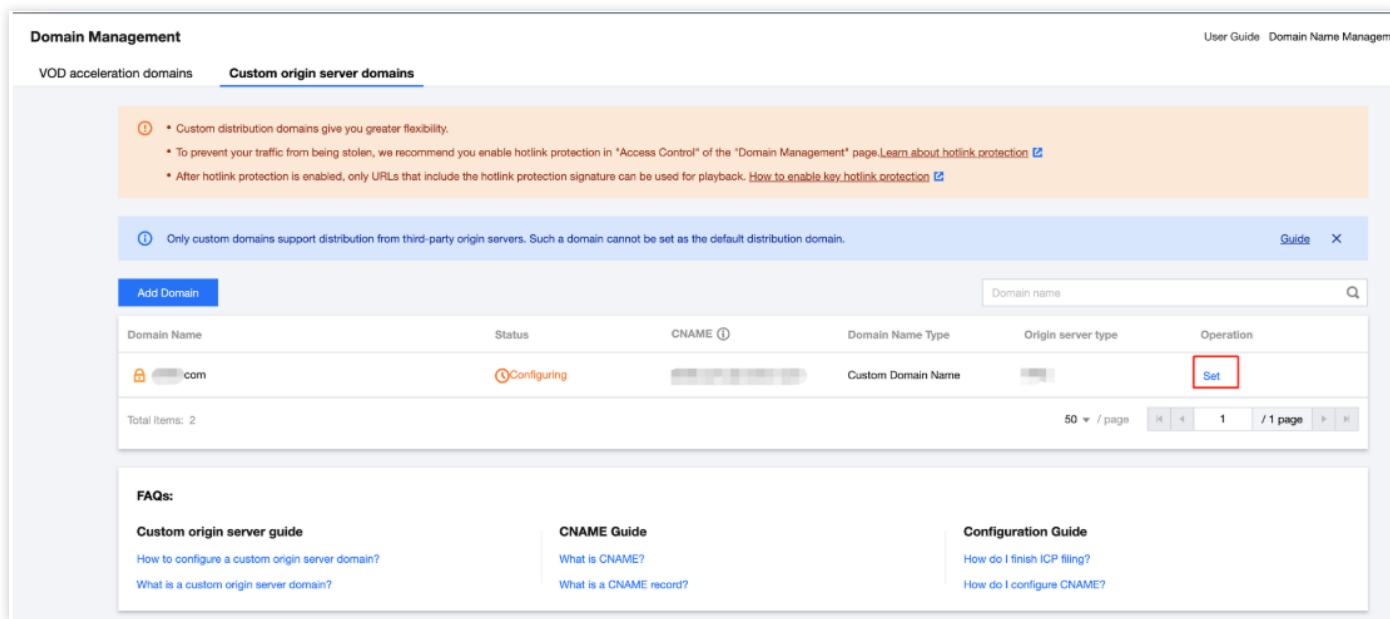
## 步骤2：自定义域名解析

针对已添加的自定义加速域名，您需要在该域名指定的 DNS 服务商配置 CNAME，用户才能通过加速域名访问到您的媒体文件，具体操作请参见 [点播加速域名-域名解析](#)。

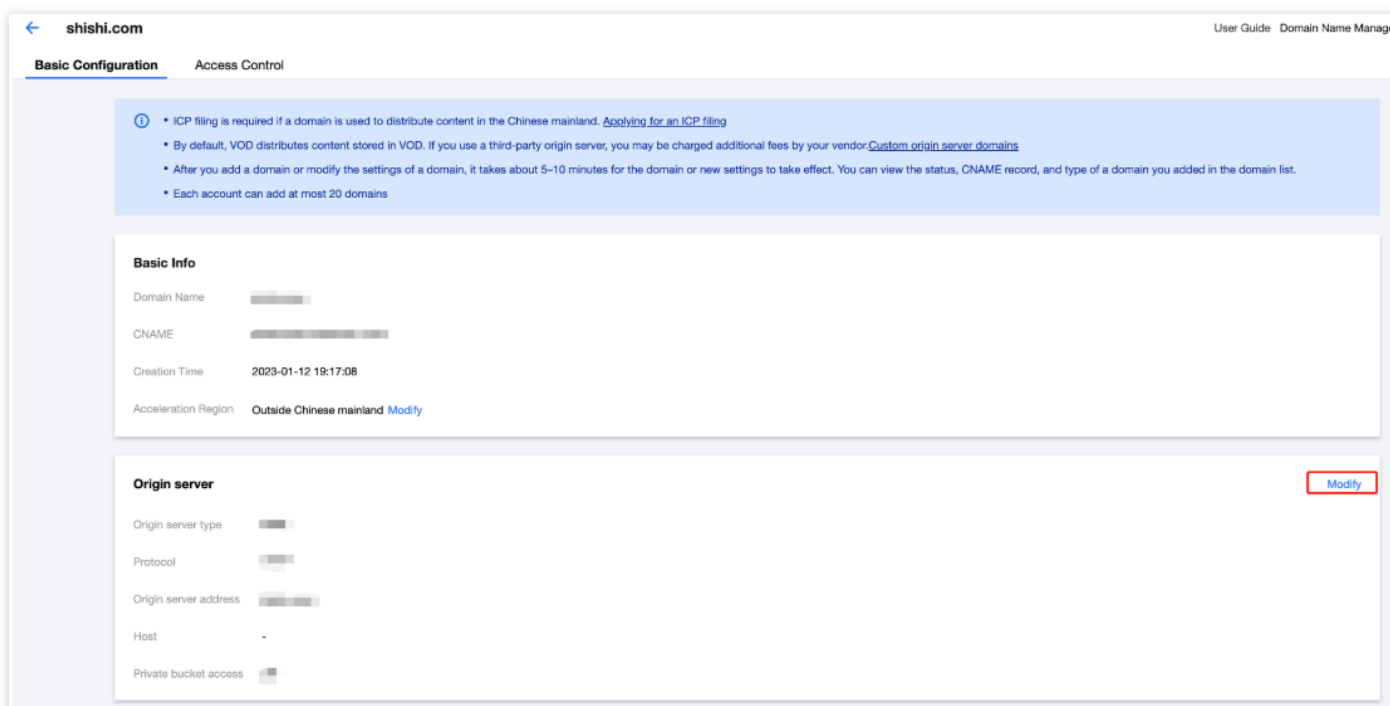
## 步骤3：自定义源站回源配置结果查看与修改



1. 前往域名管理，进入自定义源站加速域名下，选择已创建域名单击**设置**。



2. 单击**基础配置**，可查看当前自定义域名的回源配置信息并进行修改。



通过以上步骤操作，用户便可完成基于自有源站或者第三方对象存储的回源配置，能够通过云点播 CDN 对自定义源站上的媒体文件进行分发。具体分发流程说明如下：

1. 若用户添加自定义域名（例如：`test.com`）并完成域名解析和配置回源后，当通过浏览器访问该域名下的媒体文件时（例如：`http://www.test.com/test.mp4`），首先会向 Local DNS 发起域名解析请求；

2. 当 Local DNS 解析域名 `test.com` 时，发现已经配置有 CNAME：`www.test.com.cdn.dnsv1.com`，将会解析请求到腾讯 GSLB（Global Server Load Balance，即腾讯自主研发的全局负载均衡系统）；
3. GSLB 会返回最佳的接入节点 IP 列表，Local DNS 获取相应的解析 IP；
4. 用户获取到最佳接入 IP 节点；
5. 用户向最佳接入 IP 节点发起对 `http://www.test.com/test.mp4` 的访问请求；
6. 如果该 CDN 节点上缓存有 `test.mp4`，用户将获取数据，结束该请求。如果该 CDN 节点没有缓存该资源，此时该节点会向您配置的**业务源站**发起对 `test.mp4` 的请求，获取资源后，默认会缓存到当前节点中，并将资源返回给用户，此时便结束请求。

说明：

用户使用自定义源站回源功能进行加速分发播放媒体文件时会产生**下行流量费用**和**回源源站流量费用**，其中下行流量费用由云点播收取，具体规则请参见 [流量计费](#) 和 [流量资源包](#)，回源源站的流量费用由源站收取。