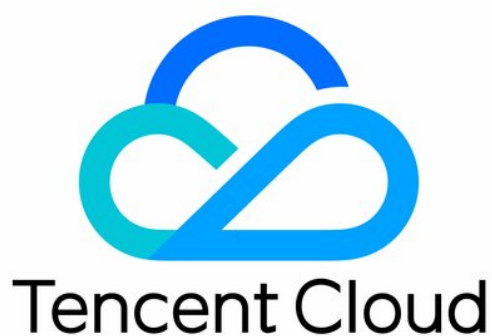


Video on Demand

ベストプラクティス

製品ドキュメント



Copyright Notice

©2013-2024 Tencent Cloud. All rights reserved.

Copyright in this document is exclusively owned by Tencent Cloud. You must not reproduce, modify, copy or distribute in any way, in whole or in part, the contents of this document without Tencent Cloud's the prior written consent.

Trademark Notice



All trademarks associated with Tencent Cloud and its services are owned by Tencent Cloud Computing (Beijing) Company Limited and its affiliated companies. Trademarks of third parties referred to in this document are owned by their respective proprietors.

Service Statement

This document is intended to provide users with general information about Tencent Cloud's products and services only and does not form part of Tencent Cloud's terms and conditions. Tencent Cloud's products or services are subject to change. Specific products and services and the standards applicable to them are exclusively provided for in Tencent Cloud's applicable terms and conditions.

カタログ：

ベストプラクティス

メディアのアップロード

ネットワークビデオをVODにプルする方法

サーバーからビデオをアップロードする方法

How to Upload Videos over the Web

アップロード品質最適化の実践

オンデマンドメディアファイルのインテリジェントコールド化方法

ビデオ処理

ビデオをトランスコードする方法

メディアの内容審査を実行する方法

ビデオに複雑なウォーターマークを追加する方法

配信と再生

Keyホットリンク防止を使用する方法

悪意あるビデオホスティング問題と防止

イベント通知の受信方法

オリジンサーバーのマイグレーションツール

Live Recording

ライブストリーミングビデオをVODにレコーディングして処理する方法

ライブストリーミング中のキャッチアップ機能の実装方法

VODビデオから疑似ライブストリーミング効果を生み出す方法

Video on DemandおよびWordPressを使用したウェブサイトの構築方法

カスタムオリジンサーバーback-to-originの方法

ベストプラクティス メディアのアップロード ネットワークビデオをVODにプルする方法

最終更新日：：2023-10-26 17:26:52

ご使用にあたっての注意事項

内容紹介

ここではネットワークのビデオ（URL形式での提供）をVODにプルする方法をご紹介します。

費用

ここで提供するコードは無償のオープンソースですが、使用過程において以下の費用が発生することがあります。

- Tencent CloudのCloud Virtual Machine（CVM）インスタンスの購入は、APIのリクエストスクリプトの実行に使用します。詳細は [CVM料金](#) をご参照ください。
- VODのストレージ容量はアップロードをプルしたビデオの保存に使用します。詳細は [ストレージ料金](#) をご参照ください。

制限

VODが提供するURLからのプル機能には以下の制限があります。

- URLはビデオファイルを直接ポイントする必要がありますが、ビデオWebサイト画面へのリンクにすることはできません。
- URLにホットリンク防止のタイムスタンプが付いている場合は、ホットリンク防止の制限（有効期間、アクセス数など）に十分な余裕があることを確認してください。余裕がない場合はアクセスが失敗する恐れがあります。
- Refererホットリンク防止のURLを有効にすることをサポートしません。
- DASH（MPDファイルタイプ）をサポートしません。
- プルするオブジェクトがHLS（M3U8ファイルタイプ）である場合、Media Segment（通常はTSファイルタイプ）のURI要件はパラメータのない相対パスでなければなりません。

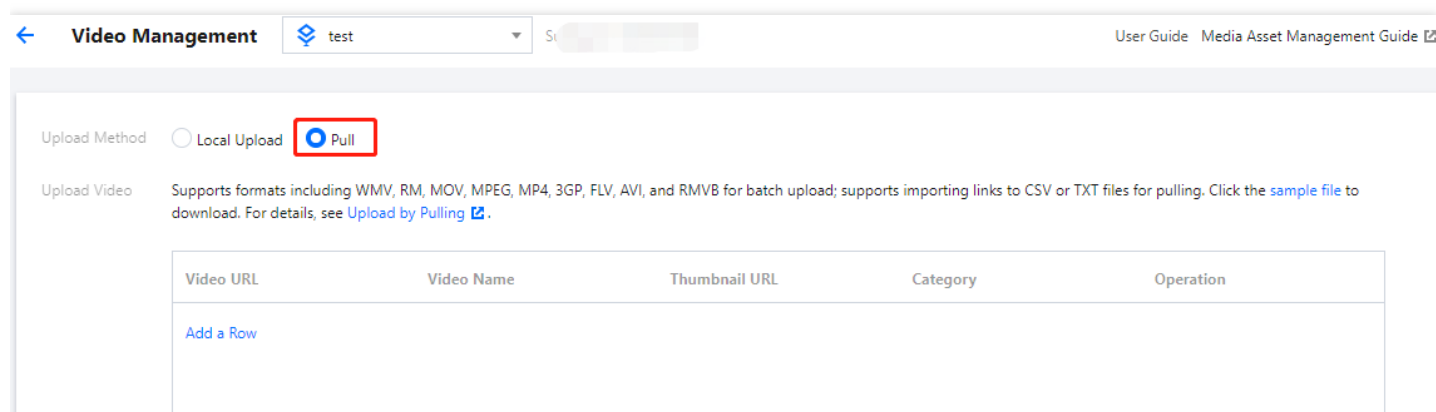
コンソールでのプルによるアップロード

手順1：VODのアクティブ化

[クイックスタート - 手順1](#) を参照してVODサービスをアクティブにします。

手順2：プルタスクの作成

VODコンソールの [アップロード画面](#) にアクセスして、アップロード方式は【ビデオのプル】を選択してから【行の追加】をクリックして、プルするビデオのURLに入力します（ここでは [テストビデオURL](#) を例にして、それ以外の項目はオプションです。開発者は必要に応じて入力入できます）。最後に左下隅の【ビデオのプル】をクリックします：



The screenshot shows the 'Video Management' interface. At the top, there is a navigation bar with 'Video Management', a user profile 'test', and links for 'User Guide' and 'Media Asset Management Guide'. Below the navigation bar, the 'Upload Method' section has two radio buttons: 'Local Upload' and 'Pull'. The 'Pull' radio button is selected and highlighted with a red box. Below this, the 'Upload Video' section contains text explaining supported formats and a link to 'Upload by Pulling'. At the bottom, there is a table with columns: 'Video URL', 'Video Name', 'Thumbnail URL', 'Category', and 'Operation'. A blue link 'Add a Row' is located below the table.

説明：

ビデオのプルに要する時間は、ビデオファイルのサイズに正比例します。長時間待つことのないように、テストには小さめのビデオ（例えば数10MB以下）を選択することをお勧めします。

手順3：プル結果の表示

1～2分間待つと（ビデオファイルのサイズによって異なります）、[メディア資産管理画面](#) でプルが完了したビデオが表示されます。

← **Create Policy** test Su

User Guide Cold Storage

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

Target Storage Class: STANDARD_JA

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_JA storage policy will be triggered.

Media File Type

Confirm Cancel

説明：

プルのプロセス中に、ブラウザがメディア資産管理画面で動かなくなった場合は、画面をリフレッシュしない限り、プルが完了したビデオは表示されません。

Tencent Cloud APIを呼び出してプルによるアップロード

手順1：Tencent Cloud CVMの準備

Tencent Cloud APIのリクエストは、1台のTencent Cloud CVM上で実行させる必要があります。要件は次のとおりです。

- リージョン：任意。
- モデル：公式サイトは最低構成（1コア1GB）であればOKです。
- パブリックネットワーク：パブリックIPを有する必要があります。帯域幅は1Mbps以上。
- OS：公式パブリックイメージ `Ubuntu Server 16.04.1 LTS 64ビット` または `Ubuntu Server 18.04.1 LTS 64ビット`。

CVMの購入方法は [操作ガイド - インスタンス作成](#) をご参照ください。システムの再インストール方法は [操作ガイド - システム再インストール](#) をご参照ください。

注意：

上述の条件に適合するTencent Cloud CVMがない場合は、その他のパブリックネットワークアクセスを備えたLinux（CentOS、Debianなど）またはMac機器でスクリプトを実行することもできます。ただし、OSの違いによってスクリプトの特定のコマンドを修正する必要があります。具体的な修正方式については、開発者自身で検索してください。

手順2：APIキーの取得

Tencent Cloud APIのリクエストにはAPIキー（SecretIdおよびSecretKey）が必要です。まだキーを作成していない場合は、[キードキュメントの作成](#)を参照して、新しいAPIキーを作成してください。キーを作成済の場合は、[キードキュメントの表示](#)を参照してAPIキーを取得してください。

手順3：VODのアクティブ化

[クイックスタート - 手順1](#)を参照してVODサービスをアクティブにします。

手順4：プルタスクの起動

[手順1](#)で準備したCVM（ログイン方法の詳細は[操作ガイド - Linuxにログイン](#)をご参照ください）にログインして、リモートターミナルで以下のコマンドを入力して実行します。

```
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
```

説明：

コマンドの中のSECRET_IDおよびSECRET_KEYに[手順2](#)で取得したコンテンツに割り当ててください。

このコマンドでは、GithubからDemoソースコードをダウンロードして、スクリプトのインストールを自動的に実行します。インストールのプロセスには数分間必要になり（CVMのネットワーク状況次第）、その間、リモートターミナルは以下に示す情報を出力します。

```
[2020-07-15 17:40:13]pip3のインストール開始。
[2020-07-15 17:40:39]pip3のインストール成功。
[2020-07-15 17:40:39]Tencent Cloud API Python SDKのインストール開始。
[2020-07-15 17:40:42]Tencent Cloud API Python SDKのインストール完了。
[2020-07-15 17:40:42]APIパラメータの設定開始。
[2020-07-15 17:40:42]APIパラメータの設定完了。
```

`pull_upload.py` スクリプトを実行してトランスコードを開始：

```
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
```

説明：

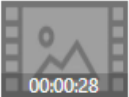

コマンドのURLを、プルする実際のビデオアドレスに置換してください。

このコマンドは、指定したURLに **PullUpload** リクエストを開始して、以下の応答のような内容を出力します。

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

手順5：プル結果の表示

1~2分間待つと（ビデオファイルのサイズによって異なります）、[メディア資産管理画面](#) でプルが完了したビデオが表示されます。

<input type="checkbox"/>	Video Info	Video Status	Video Category ▼	Video Source ▼
<input type="checkbox"/>	 API-PullUpload ID: 5285890806967135776	 Normal	Others	Upload

説明：

プルのプロセス中に、ブラウザがメディア資産管理画面で動かなくなった場合は、画面をリフレッシュしない限り、プルが完了したビデオは表示されません。

サーバーからビデオをアップロードする方法

最終更新日：2023-10-26 17:26:52

ご使用にあたっての注意事項

内容紹介

ここでは、ビデオファイルをローカルのサーバーからVODにアップロードする方法をご紹介します。

費用

ここで提供するコードは無償のオープンソースですが、使用過程において以下の費用が発生することがあります。

- Tencent CloudのCloud Virtual Machine (CVM) インスタンスの購入は、アップロードスクリプトの実行に使用します。詳細は [CVM料金](#) をご参照ください。
- VODのストレージ容量はアップロードをプルしたビデオの保存に使用します。詳細は [ストレージ料金](#) をご参照ください。

CVMのビデオをVODにアップロード

手順1：Tencent Cloud CVMの準備

アップロードスクリプトは、1台のTencent Cloud CVM上で実行させる必要があります。要件は次のとおりです。

- リージョン：任意。
- モデル：公式サイトは最低構成（1コア1GB）であればOKです。
- パブリックネットワーク：パブリックIPを有する必要があります。帯域幅は1Mbps以上。
- OS：公式パブリックイメージ `Ubuntu Server 16.04.1 LTS 64ビット` または `Ubuntu Server 18.04.1 LTS 64ビット`。

CVMの購入方法は [操作ガイド - インスタンス作成](#) をご参照ください。システムの再インストール方法は [操作ガイド - システム再インストール](#) をご参照ください。

注意：

上述の条件に適合するTencent Cloud CVMがない場合は、その他のパブリックネットワークアクセスを備えたLinux（CentOS、Debianなど）またはMac機器でスクリプトを実行することもできます。ただし、OSの違いによってスクリプトの特定のコマンドを修正する必要があります。具体的な修正方式については、開発者自身で検索してください。

手順2：VODのアクティブ化

[クイックスタート - 手順1](#) を参照してVODサービスをアクティブにします。

手順3：APIキーの取得

ビデオのアップロードのリクエストにはAPIキー（SecretIdおよびSecretKey）が必要です。まだキーを作成していない場合は、[キードキュメントの作成](#) を参照して、新しいAPIキーを作成してください。キーを作成済の場合は、[キードキュメントの表示](#) を参照してAPIキーを取得してください。

手順4：コードのダウンロードおよびSDKのインストール

[手順1](#) で準備したCVM（ログイン方法の詳細は [操作ガイド - Linuxにログイン](#) をご参照ください）にログインして、リモートターミナルで以下のコマンドを入力して実行します。

```
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
```

説明：

コマンドの中のSECRET_IDおよびSECRET_KEYに [手順3](#) で取得したコンテンツに割り当ててください。

このコマンドでは、GithubからDemoソースコードをダウンロードして、スクリプトのインストールを自動的に実行します。インストールのプロセスには数分間必要になり（CVMのネットワーク状況次第）、その間、リモートターミナルは類似した以下の情報を出力します。

```
[2020-06-23 19:56:31]pip3のインストール開始。
[2020-06-23 19:56:34]pip3のインストール成功。
[2020-06-23 19:56:34]VOD PythonアップロードSDKのインストールを開始。
[2020-06-23 19:56:36]VOD PythonアップロードSDKのインストール完了。
[2020-06-23 19:56:36]SDKパラメータの設定開始。
[2020-06-23 19:56:36]SDKパラメータの設定完了。
```

手順5：ビデオのアップロード

アップロードの開始前に、CVMでビデオファイルおよびカバーピクチャ（オプション）を準備しておく必要があります。ビデオをCVMにアップロードするのに不都合がある場合は、リモートターミナルで以下のコマンドを実行して、テストビデオおよびテストカバーをCVMにダウンロードすることができます。

```
ubuntu@VM-69-2-ubuntu:~$ wget http://1400329073.vod2.myqcloud.com/d62d88a7vodtran  
scq1400329073/7a9b2b565285890804459281865/v.f100010.mp4 -O ~/vod-server-demo/serv  
er_upload/tencent_cloud.mp4; wget http://1400329073.vod2.myqcloud.com/ff439affvod  
cq1400329073/8aa658d15285890804459940822/5285890804459940825.jpg -O ~/vod-server-  
demo/server_upload/tencent_cloud.jpg
```

`server_upload.py` スクリプトを実行してトランスコードを開始：

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

説明：

コマンドのビデオパスおよびカバー画像のパスを実際のファイルパスに置換してください。そのうち、カ
バー画像のパスのパラメータがオプションで、空のままにすると、アップロード後のビデオにカバーは付
きません。

このコマンドは、`tencent_cloud.mp4` ビデオを VOD にアップロードすると同時に、アップロードした
`tencent_cloud.jpg` 画像をそのカバーにします。アップロードの完了後、リモートターミナルは以下のような情報を
出力します。

```
{"CoverUrl": "http://1400329073.vod2.myqcloud.com/ff439affvodcq1400329073/8aa658d  
15285890804459940822/5285890804459940825.jpg", "FileId": "5285890804459940822",  
"MediaUrl": "http://1400329073.vod2.myqcloud.com/ff439affvodcq1400329073/8aa658d1  
5285890804459940822/f0.mp4", "RequestId": "84a7fb42-9f05-4acd-9cc8-843690b188ce"}
```

説明：

自身のビデオを使用してテストを行う場合は、CVM 帯域幅が不足のためにアップロード時間がかかりすぎ
ないように、小さめのビデオファイル（例えば数 MB）を使用することをお勧めします。

手順6：結果の表示

コンソールの [ビデオ管理](#) 画面で、アップロードしたビデオファイルおよびカバーを見ることができます。

コードの解釈

1. `main()` はスクリプトのエントリー。
2. `parse_conf_file()` を呼び出して、`config.json` ファイルから設定情報を読み取ります。設定項目の説明は以下のとおりです。

フィールド	データ型	機能
<code>secret_id</code>	String	APIキー
<code>secret_key</code>	String	APIキー
<code>procedure</code>	String	タスクフロー名。ビデオのアップロード完了後、そのタスクフローは自動的にトリガーされます。デフォルトは空です。
<code>subappid</code>	String	ビデオを VODサブアプリケーション にアップロードしますか

説明：

このDemoは、`procedure` および `subappid` の2つのアップロードパラメータのみをサポートしています。すべての機能については、[PythonアップロードSDKインターフェースの記述](#)をご参照ください。

3. コマンドラインパラメータから、アップロードするビデオファイルのローカルパスおよびカバー画像パス（カバーがある場合）を取得してから、`upload_media()` を呼び出してアップロードを開始します。

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

# アップロードの開始
rsp = upload_media(configuration, video_path, cover_path)
```

4. `upload_media()` で、Python SDKが提供する方法を使用して、1個のアップロードインスタンス `client` を構築します。その後、`req` にアップロードパラメータを設定して、最後にアップロードを開始します。

```
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()` の最初のパラメータ (`"ap-guangzhou"`) は、アップロードされたインスタンスのアクセスリージョンです。ビデオのアップロードされたビデオのストレージリージョンではありません。そのパラメータは `"ap-guangzhou"` に固定するだけです。アップロードしたビデオのストレージリージョンを指定する場合は、`req.StorageRegion` パラメータを設定してください。

その他の機能

VOD サーバーからアップロードする SDK は、その他の機能もサポートしています。ビデオ名、分類、期限切れなどを設定する場合の詳細情報は、対応する言語の SDK 開発ガイドをご参照ください。

- [Java](#)
- [C#](#)
- [PHP](#)
- [Python](#)
- [Golang](#)

How to Upload Videos over the Web

最終更新日： : 2022-12-16 15:12:27

Overview

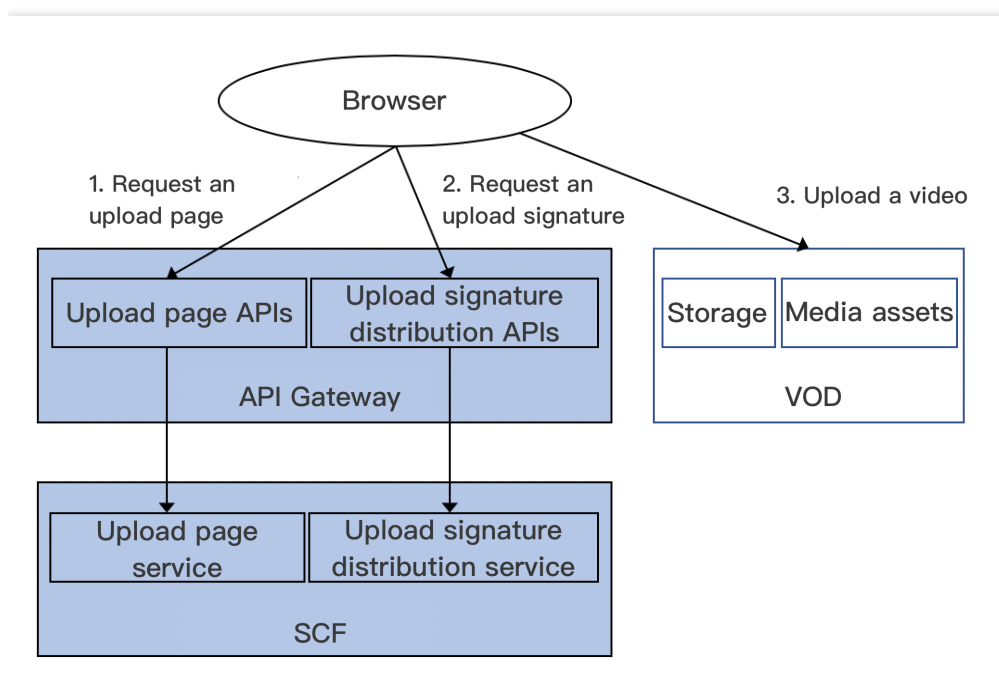
Demo features

This demo shows you how to upload videos to VOD through the webpage. It builds two HTTP services based on SCF:

- The first service is used to receive a request from the browser to get the [signature for upload from client](#), calculate the signature, and return it.
- The second service uses VOD's [web upload SDK](#) to implement a page. You can access the page in a browser to upload local videos to VOD.

Architecture and process

The system mainly involves four components: browser, API Gateway, SCF, and VOD. Here, API Gateway and SCF are the deployment objects of this demo as shown below:



The main business process is as follows:

1. The browser requests SCF for an upload page.
2. You select a local video and click upload on the upload page, and the browser requests SCF for an upload signature.

3. The browser uses the upload signature to initiate an upload request to VOD and displays the upload result on the upload page after completion.

Note :

The SCF code in the demo is developed based on Python 3.6. SCF also supports other programming languages such as Python 2.7, Node.js, Go, PHP, and Java for your choice as needed. For more information, please see [Development Guide](#).

Fees

The VOD web upload demo (including the webpage code and service backend code) provided in this document is open-source and free of charge, but it may incur the following fees during service building and use:

- Fees for purchasing a Tencent Cloud CVM instance to run the service deployment script. For more information, please see [Instance Billing Modes](#).
- Fees for using the upload page and signature distribution service provided by SCF. For more information, please see [Billing Mode](#) and [Free Tier](#).
- Fees for using Tencent Cloud API Gateway to provide public network APIs for SCF. For more information, please see [Billing Overview](#).
- Fees for VOD storage of uploaded videos. For more information, please see [Pay-as-You-Go \(Postpaid Daily Billing Cycle\)](#).
- Fees for VOD traffic consumed by video playback. For more information, please see [Pay-as-You-Go \(Postpaid Daily Billing Cycle\)](#).

Quick Deployment of Web Upload Demo

The web upload demo is deployed on SCF with a service entry provided by API Gateway. To make it easier for you to build services, we provide a quick deployment script as detailed below.

Step 1. Prepare a CVM instance

The deployment script needs to be executed on a CVM instance meeting the following requirements:

- Region: not limited.
- Model: the minimum official configuration (1 CPU core and 1 GB memory) is sufficient.
- Public network: a public IP is required, and the bandwidth should be at least 1 Mbps.
- Operating system: official public image `Ubuntu Server 16.04.1 LTS 64-bit` or `Ubuntu Server 18.04.1 LTS 64-bit` .

For detailed directions on how to purchase a CVM instance and reinstall the system, please see [Operation Guide - Creating Instances via CVM Purchase Page](#) and [Operation Guide - Reinstalling System](#), respectively.

Note :

- **The web upload demo itself does not depend on CVM but only uses CVM to run the deployment script.**
- If you do not have a CVM instance satisfying the above conditions, you can also run the script on another Linux (such as CentOS or Debian) or macOS server with public network access, but you need to modify certain commands in the deployment script based on the operating system. Please search for the specific modification method by yourself.

Step 2. Activate VOD

Please activate the VOD service as instructed in [Getting Started - Step 1](#).

Step 3. Get the API key and APPID

Your API key (i.e., `SecretId` and `SecretKey`) and `APPID` are required for deploying and running the web upload demo service.

- If you have not created an API key yet, please generate one as instructed in [Root Account Access Key](#). If you have already created a key, please get it as instructed in the same document.
- You can view the `APPID` on the [Account Information](#) page in the console.

Step 4. Deploy the service backend and webpage

Log in to the [CVM instance prepared in step 1](#) as instructed in [Logging In to Linux Instance in Standard Login Method](#) and enter and run the following command on the remote terminal:

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

Note :

Please assign the corresponding values obtained in [step 3](#) to `SECRET_ID` , `SECRET_KEY` , and `APPID` in the command.

This command will download the demo source code from GitHub and automatically run the installation script. The installation process will take several minutes (subject to the CVM network conditions), during which the remote terminal will print the following information:

```
[2020-04-25 23:03:20] Start installing pip3.
[2020-04-25 23:03:23] pip3 is successfully installed.
[2020-04-25 23:03:23] Start installing Tencent Cloud SCF.
[2020-04-25 23:03:26] SCF is successfully installed.
[2020-04-25 23:03:26] Start configuring SCF.
[2020-04-25 23:03:28] SCF configuration is completed.
[2020-04-25 23:03:28] Start deploying the VOD client upload client signature distribution service.
[2020-04-25 23:03:40] The deployment of the VOD client upload signature distribution service is completed.
[2020-04-25 23:03:44] Start deploying the VOD web upload page.
[2020-04-25 23:03:53] The deployment of the VOD web upload page is completed.
[2020-04-25 23:03:53] Please access the following address in your browser to use the demo: https://service-xxxxxxx-125xxxxxx.gz.apigw.tencentcs.com/release/web_upload_html
```

Copy the address of the webpage in the output log (which is `https://service-xxxxxxx-125xxxxxx.gz.apigw.tencentcs.com/release/web_upload_html` in this example).

Note :

If the following warning is displayed in the output log, it is generally because the CVM instance cannot immediately parse the service domain name deployed just now. You can ignore this warning.

```
>[2020-04-25 17:18:44] Warning: the client upload signature distribution service failed the test.
>
```

Step 5. Use the web upload demo

1. Access the address copied in [step 4](#) in a browser to start using the web upload demo.
2. Perform video upload operations on this page:
 - i. Select a local video file (MP4 format is recommended).
 - ii. (Optional) Select a local cover image (in JPG or PNG format).
 - iii. (Optional) Enter the video name.
 - iv. Click **Start Upload** to upload the video.

3. After the upload is completed, the VOD media IDs (i.e., `fileId`) and URLs of the uploaded video and cover will be displayed at the bottom of the page.

You can view the uploaded video in the [VOD console](#).

Note :

You can try out other features on the upload page as prompted.

System Design Description

API protocol and test

Both the **upload page** and **upload signature distribution** functions use API Gateway to provide APIs. The specific API protocol is as detailed below:

Service	Function Name	API Form	Response Content
Upload page	web_upload_html	HTTP GET	HTML page
Upload signature distribution	ugc_upload_sign	HTTP POST	Upload signature

Upload page

You can access the [SCF service list](#) to view the details of the upload page service:

Note :

- The two SCF functions used by the demo are deployed under the namespace `vod_demo` in the Guangzhou region.
- You need to select the corresponding region and namespace in the console to view the deployed SCF functions.

Click the function name, select **Trigger Management** on the left, and **Access Path** on the right is the URL of the upload page. Click **API Service Name** to redirect to the corresponding API Gateway page.

To test the service, directly access the page URL in a browser to check whether the upload page is displayed normally.

Upload signature distribution

You can access the [SCF service list](#) to view the details of the upload signature distribution service in the same way as detailed in [Upload page](#).

Click the function name, select **Trigger Management** on the left, and **Access Path** on the right is the URL of the service. Click **API Service Name** to redirect to the corresponding API Gateway page.

To test the service, manually send an HTTP request and run the following command on a Linux or macOS device with public network access (please modify the service URL according to the actual situation):

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

If the service is normal, an upload signature will be returned. Below is a sample signature:

```
VYapc9EYdoZLzGx0CglRW4N6kuhzZWNyZXRJZD1BS0lEZk5xMz16dG5tYW1tVzBMOXFvZERia25hUjdZa0xPM1UmY3VycmVudFRpbWVtdGFtcD0xNTg4NTg4MDIzJmV4cGlyZVRpbWU9MTU4ODU4ODYyMyZyYW5kb209MTUwNzc4JmNsYXNzSWQ9MCZvbWVUaW1lVmFsaWQ9MCZ2b2RTdWJBcHBjZD0w
```

You can also use third-party tools such as Postman to send HTTP requests. Please search for specific usage on the internet.

Upload page service code interpretation

1. `main_handler()` is the entry function.
2. Read the content of the `web_upload.html` file, which is the upload page content.

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

3. Read configuration items from `config.json`, which refer to the content that you cannot predict when you write the SCF service and need to determine during the deployment process. The content is written into `config.json` in real time by the deployment script before deploying the upload page service.

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

4. Call `render_template` and modify the upload page content according to the configuration information obtained in the previous step. The configuration items are expressed in the format of `"variable name":`

"value" in the `config.json` file or in the format of `{variable name}` in the `web_upload.html` file. When modifying them, please replace them with the specific values as detailed below.

```
def render_template(html, keys):
    """Replace the variables (in the format of `${variable name}`) in HTML with specific content."""
    for key, value in keys.items():
        html = html.replace("${" + key + "}", value)
    return html
```

Variable	Description	Value Type	Value Source
UGC_UPLOAD_SIGN_SERVER	Upload signature distribution service URL	String	Output by SCF CLI after the deployment of the upload signature distribution service is completed.

5. Return the modified content of the upload page. For the formats and descriptions of the returned data, please see [\[Overview of API Gateway Trigger\]\(https://www.tencentcloud.com/document/product/583/12513\)](https://www.tencentcloud.com/document/product/583/12513).

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

Upload signature distribution service code interpretation

- `main_handler()` is the entry function.
- Call `parse_conf_file()` and read the configuration information from the `config.json` file. The configuration items are as described below (for specific parameters, please see [Signature for Upload from Client](#)):

Field	Data Type	Description
secret_id	String	API key
secret_key	String	API key
sign_expire_time	Integer	Signature validity period in seconds
class_id	Integer	Category ID of the uploaded video. 0 indicates the default category
otp	Integer	Whether the signature is one-time

Field	Data Type	Description
subappid	Integer	Whether to upload to a VOD subapplication

- Call `parse_source_context()` to parse the `sourceContext` field in the request body, which can be passed through to the event notification receipt service during [video upload completion event notification](#) (not used in this demo).

Note :

This field is optional during the upload process. If you don't need this feature, you can ignore this part of the code.

- Call the `generate_sign()` function to calculate the signature. For more information, please see [Signature for Upload from Client](#).
- Return the signature. For the formats and descriptions of the returned data, please see [Overview of API Gateway Trigger](#).

```
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')
}
```

アップロード品質最適化の実践

最終更新日：2023-10-26 17:26:52

概要

クライアントからのアップロードか、サーバーからのアップロードかを問わず、ファイル転送プロセスにおいては下記のいくつかのよくある品質問題が生じることがあります。

1. ファイルのアップロードがこんなにも遅い理由は？
2. アップロードの速度を上げる方法は？
3. アップロード成功率が低い点についての解決方法は？
4. モバイル端末では弱いネットワーク環境により、アップロードの失敗がよく発生しますが、その対処方法は？

アップロードの品質を判断する指標：アップロードの速度および成功率です。

- アップロードの速度がとても遅いと、往々にしてユーザーの直観的印象を悪くすることがあります。例えば、ユーザーが50Mのビデオを1個アップロードして、30分経ってもアップロードが完了しないと、ユーザーが我慢できなくなり、顧客を失う可能性が高くなります。
- アップロードの成功率が高いか低いかは、サービス品質の拠り所となります。ネット事情によって最初のアップロードに失敗すると、ユーザーが再度アップロードを始める可能性は低くなり、ユーザーからのクレームという直接的な結果を被ることになるため、アップロードの成功率を保証することは最も基本的な要件になります。

ここでは、VODを基本としたアップロードのシナリオを、原因分析および解決案の両面に重点を置いてご紹介します。ご自身の業務シナリオを基に比較して、適切なプランを選択してアップロードの品質を高めることができます。

アップロードの品質に影響する原因

ネットワーク帯域幅

ネットワーク帯域幅は、時間単位内に転送できるデータ量を示します。帯域幅が広がるほど、時間単位内にアップロードするデータ量も多くなり、アップロード速度が速くなることを示します。アップロードはエンドツーエンドの行為であるため、エンドツーエンドの帯域幅はアップロードの品質に影響します。そして、VODのバックエンドサーバーは現在帯域幅が十分にあるために、アップロードの品質は多くの場合ユーザー側の帯域幅に依存します。

ユーザーとストレージセンターとの距離

アップロードされたファイルは最終的にVODに保存する必要があります。保存する場所はストレージセンターになります。ユーザーはVODをアクティブにすると、VODはデフォルトで重慶リージョンをストレージセンターとしてアサインします。ユーザーとストレージセンターとの距離はネットワークのリンクの長さに影響します。

例えば、北京から重慶にアップロードする場合、同じファイルを成都から重慶にアップロードするのと比較すると、途中で経由するリンクはさらに長くなり、距離が長くなるにつれて影響要因も多くなります。最終的にアップロード速度は遅くなってしまいます。長いリンク転送中で現れるネットワークジッター、パケットロスなどの問題のために、アップロード成功率にも影響を与えることとなります。リンクが短い場合でもこれらの問題を避けることは不可能です。ただ長いリンクに比較すれば、出現する確率はかなり低下します。ユーザーとストレージセンターとの距離を短くすることは、アップロードの品質を高めるキーポイントの1つです。

弱いネットワーク環境

弱いネットワーク環境とは、遅延とパケットロスの頻度が高いネットワークの状態のことを指し、一般には「ネットアクセスが遅い」ということです。例えば、エレベータや地下鉄の中など、弱いネットワーク事情は現実生活の中でもよく現れます。その主な原因は受信状態が良好でないことです。影響を受ける環境ではデータパケットの転送が遅かったり失敗したりします。こうしたシナリオではクライアントからのアップロードが極めて多数であり、特に現在はモバイルインターネットが盛んな時代であることから、弱いネットワークは多くの開発者を困らせています。このことも、成功率の指標を高めるうえで最も難しい課題です。

対処方法

同時アップロード

ネットワーク帯域幅が不足するシナリオについて、直接的な方法は帯域幅をさらに広げる申請を行うことですが、私たちが直面している問題は帯域幅ネットワークに制限があることである場合、帯域幅をいかに十分に活用してアップロードするか、これは私たちが解決すべきことです。同時アップロードは2つのカテゴリに分けられます。

- ファイル別、複数のファイルを同時にアップロード。
- パート別、1個のファイルを複数のパートに分けて同時にアップロード。

前者でも、後者でも、対応する同時実行数を調整することで帯域幅の使用率を向上させることができます。

ファイル同時アップロード

ファイルの同時アップロードは、複数のプロセスまたはスレッドを使用して、同時にアップロード操作を開始することです。現在のVODは関連するSDKパッケージを提供していません。ユーザーは特定の言語を参照することで実現することができます。以下ではVODの [Java SDK](#) を基に1個の簡単な例を示します。

```
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 {
// 同時実行数
Integer threadNumber = 20;
// アップロード待ちのファイルパスのリスト
List<String> filePathList = new ArrayList<String>();

// アップロード待ちのファイルパスの追加
filePathList.add("/data/path1.mp4");
filePathList.add("/data/path2.mp4");
filePathList.add("/data/path3.mp4");

// スレッドプールの作成
ExecutorService pool = Executors.newFixedThreadPool(threadNumber);
// アップロードClientの作成
VodUploadClient client = new VodUploadClient("your secretId", "your secretKey");

// 同時アップロード
for (String path : filePathList) {
// アップロードタスクの送信
pool.submit(new UploadThread(client, path));
}
}

// スレッドのアップロード
class UploadThread implements Runnable {
// Clientのアップロード
private VodUploadClient uploadClient;
// ファイルパス
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 {
    // アップロードの実行
    VodUploadResponse response = uploadClient.upload("ap-guangzhou", request);
    System.out.println(response.getFileId());
} catch (Exception e) {
    e.printStackTrace();
}
}
```

パートの同時アップロード

パート同時アップロードは、大きなファイルに対応するもので、複数のパートに分割すると同時にアップロードするものです。パート同時アップロードのメリットは、1個のファイルのアップロードを素早く完了させられることで、VODが提供するSDKはファイルのサイズを基に、通常アップロードとパートアップロードを自動的に選択します。ユーザーは、パートアップロードの各手順に気を遣う必要もなく、パートでのアップロードを実現できます。こうしてファイルの同時パート数は、`ConcurrentUploadNumber` パラメータによって指定します。具体的なユースケースは対応するSDKをご参照ください。現在このパラメータをサポートしているSDKは以下のとおりです。

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

近くでのアップロード

近くでのアップロードは、アップロードする人の所在位置を検知して、アップロードする人から最も近いストレージセンターをアサインしてアップロードできることを指します。例えば、成都のユーザーは、上海リージョンではなく重慶リージョンにアサインされてアップロードします。

近くでのアップロード機能の最大のメリットは、アップロードする人からサーバーまでの転送距離を短くしたことです。そのメリットは以下のとおりです。

- 転送距離を短縮し、アップロード速度を向上。
- 安定性を高め、成功率を保証。

VODは近くでのアップロード機能をサポートしています。以下の2点だけ簡単に確認することが必要です。

• 複数のストレージリージョンのアクティブ化

VODがデフォルトで提供するストレージリージョンは**シンガポール**です。近くでのアップロード機能を十分に活用したい場合は、近くで保存したいリージョンをアクティブにする必要があります。コンソールでその他のストレージリージョンを自分で追加することができます。詳細は [アップロードストレージ設定](#)をご参照ください。複数リージョンでのストレージを有効にした後で、ユーザーがアップロードするときは、VODはIPでユー

ザーのリージョンを識別し、パスを通じて、アップロード用にアクティブになったリージョンリストを基に、ユーザーの最も近くのリージョンをインテリジェントにアサインします。

• スケジューリングが正しいかどうかを検証

重慶と上海のストレージリージョンをアクティブにしている場合、ユーザーが成都でアップロードを開始すると、理論上では近くのスケジューリングによって、重慶にアップロードされます。スケジューリングが適切か確認する方法は、アップロードの完了時に返された `FileId` を取得し、[メディア詳細情報インターフェース](#) を通じて返された基本情報 (`basicInfo`) を確認します。そこにはアップロードされたメディアファイルのストレージリージョンを示す `StorageRegion` フィールドが含まれます。

途中でプロキシまたは転送を経由し、VODのIPによるリージョン識別が正しくない場合、ファイルアップロード用のストレージリージョンを強制的に指定できます。対象のリージョンにアップロードするための具体的な方法については以下をご参照ください。

- [クライアントからのアップロードガイド](#)
- [サーバーからのアップロードガイド](#)

アップロード事前検出

アップロードの事前検出は、ネットワーク接続障害、タイムアウト、DNSの乗っ取りなどのネットワークエラーのシナリオを最適化するための手段です。これは、弱いネットワークでのアップロードについてVODが提供する有効な軽減措置です。最適化ポリシーには以下の数点があります。

- HTTPDNSを使用してドメイン名を解決してバックエンドアドレスを取得し、DNSの乗っ取りを防止。
- 複数のリージョンの接続性およびアップロード速度を検出して、最適なアップロードリージョンを取得。
- Tencent Cloudのアクセラレーションネットワークは、信頼性のある安定した転送コネクションを提供。

アップロードの事前検出機能は現在、クライアントからのアップロードに適用されています。接続方式は簡単です。具体的な使用方法については、SDKの `事前アップロード` の説明をご参照ください。

- [AndroidのアップロードSDK](#)
- [iOSのアップロードSDK](#)

オンデマンドメディアファイルのインテリジェントコールド化方法

最終更新日：：2023-10-26 17:26:52

インテリジェントコールド化は、オンデマンドのメディア資産管理機能に依存しています。当社は、VODサービスを使用する開発者のストレージコストを削減するために、メディアライフサイクル管理機能をリリースしました。開発者は、自分のビジネスの特徴に応じてユーザーのストレージコストを効果的に削減し、オンデマンドファイルのストレージタイプを所定のポリシーに従って、標準ストレージから低頻度ストレージ、アーカイブストレージ、ディープアーカイブストレージに引き下げ、自分のメディアファイルをよりフレキシブルに管理できるようになります。

シナリオ

- **ライブeコマース**：ネットワークライブストリーミングのサービスプロバイダは、中国文化部の「[インターネット取引監督管理弁法](#)」の要件に基づいて、オンライン取引におけるライブストリーミングビデオの保存期間を、ライブストリーミングが終了した日から3年以上としています。ライブストリーミングビデオはVODに保存され、ストレージタイプは標準ストレージになります。ビデオの一部は再試聴されず（またはほとんど再試聴されず）、関連部門による審査にのみ用いられます。メディア資産のインテリジェントコールド化機能は、お客様がストレージコストを効果的に削減するお手伝いをします。
- **低頻度アクセスメディアのコールド化**：ビデオポータル、ストリーミングメディア、UGCコンテンツ管理プラットフォームや、ユーザーが頻繁にアクセスや視聴をしないメディアの場合、直接オフラインにすることはできず、ストレージコストも高くなります。VODのメディア資産のインテリジェントコールド化機能は、アクセス回数を通じてメディアファイルのコールド化を行い、ストレージコストを効果的に削減するとともに、低頻度の視聴の要件を満たすこともできます。
- **メディアアーカイブ**：ニュースメディア・ラジオ・テレビ業界の場合、メディアファイルは高い適時性を持っています。通常、このようなメディアは長期間履歴データとして保存され、後日必要になったときに再検索して視聴します。視聴の適時性の要件は高くなります。このようなシーンでは、メディアをアーカイブストレージまたはディープアーカイブストレージに変更して、ユーザーのストレージコストを削減することができます。

前提条件

1. Tencent Cloudアカウントを[登録](#)して[ログイン](#)し、アカウントの実名認証を完了していること。
2. Tencent Cloud VODサービスがアクティブ化されました。まだアクティブ化されていない場合は、[VODサービス](#)に移動してアクティブ化してください。

3. ポリシーの作成を行います。詳細については、[メディア資産コールド化](#)をご参照ください。

利用説明

ユーザーは、メディア資産機能の理解を深め、使いやすくするため、[ストレージタイプ](#)、[データの取得および取得モード](#)、[ポリシー管理](#)といった概念を理解する必要があります。

ストレージタイプ

ユーザーのメディアファイルがVODで保存される際のストレージタイプです。VODは、標準ストレージ、低頻度ストレージ、アーカイブストレージ、ディープアーカイブストレージといったストレージタイプを提供します。その特性を次の表に示します。

ストレージタイプ	標準ストレージ	低頻度ストレージ	アーカイブストレージ	ディープアーカイブストレージ
VODデフォルト	はい	いいえ	いいえ	いいえ
ストレージコスト	高い	中程度	低い	極めて低い
アクセス性能	高い	低い	アクセスサポートなし	アクセスサポートなし
データ取得料金	なし	なし	あり	あり
サポートリージョン	すべて	すべて	すべて	北京、上海、重慶

標準ストレージは、VODのデフォルトのストレージタイプです。CSSレコーディングやさまざまなアップロード方法、MPSタスクによって発行されたファイルはすべて標準ストレージです。

属性	高から低への対比
ストレージコスト	標準ストレージ>低頻度ストレージ>アーカイブストレージ>ディープアーカイブストレージ
アクセスパフォーマンス	標準ストレージ>低頻度ストレージ

説明：

- アーカイブストレージとディープアーカイブストレージは直接アクセスをサポートしていないため、まず取得操作を行う必要があります。VODは、ユーザーによる標準ストレージへの取得のみをサポートします。

- アクセスパフォーマンスは、ビデオを視聴する際の**最初の画面時間**、**ラグ率**などの指標に影響を与えるため、**頻繁にアクセスするオンラインビジネスの場合、ユーザーがストレージタイプを変更することはお勧めしません。**

VODは、さまざまなストレージタイプをサポートして変更を行います。許可されるストレージタイプの変更は次のとおりです。

ソースストレージタイプ	ターゲットストレージタイプ
標準ストレージ	低頻度ストレージ、アーカイブストレージ、ディープアーカイブストレージ
低頻度ストレージ	標準ストレージ、アーカイブストレージ、ディープアーカイブストレージ
アーカイブストレージ	標準ストレージ
ディープアーカイブストレージ	標準ストレージ

説明：

-アーカイブストレージとディープアーカイブストレージを標準ストレージ以外の他のストレージタイプに変更する場合は、まず標準ストレージに変更してから、最終的なターゲットストレージタイプに変更する必要があります。

-ストレージタイプの変更は、FileIdを粒度とします。すなわち、オリジナルファイルとMPSファイルのストレージタイプは同じであり、FileId下の一部ファイルに対するストレージタイプの個別指定はサポートされていません。

データの取得および取得モード

データの取得

メディアをアーカイブストレージまたはディープアーカイブストレージに変更した場合、再生・MPSの開始などを含む、直接アクセスができなくなります。アクセスする必要がある場合は、上記で紹介したストレージタイプの変更操作を使用することができます。ただし、開発者が短時間でアクセスする必要があり、アクセスが終了した後も、メディアがアーカイブストレージまたはディープアーカイブストレージのステータスを維持できるように希望する場合は、上記の方法は適していません。このような場合、オンデマンドでデータの取得機能を提供しています。

VODは、アーカイブストレージとディープアーカイブストレージという2つの取得操作をサポートしています。取得には、永続的な取得と指定期間中の取得があります。

アーカイブストレージ/ ディープアーカイブストレージの操作	ストレージタイプの変更	取得
適時性	非同期	非同期
モード	超高速/標準/バッチ	超高速/標準/バッチ
有効期間	永続的	指定期間
レプリカの作成	いいえ	はい

1. 取得した後、VODは、ストレージタイプが標準ストレージであるメディアのレプリカを発行します。
2. 取得したレプリカには有効期間（単位は日数）があります。有効期間中はメディアへのアクセスが許可されます。有効期間が終了すると、レプリカは自動的にクリーンアップされ、それ以降はメディアへのアクセスは許可されなくなります。
3. レプリカの有効期間中は、対応する標準ストレージ料金が追加で発生します。
4. レプリカの有効期間中は取得操作を再度開始することはできません。

取得モード

アーカイブストレージおよびディープアーカイブストレージに対するストレージタイプの変更や取得操作には、複数のモードがあります。それぞれのモードの最終的な効果は同じですが、速度と料金（取得料金）が異なります。

取得モード	アーカイブストレージ取得時の消費時間	ディープアーカイブストレージ取得時の消費時間
超高速モード	5分	サポートなし
標準モード	5時間	24時間
バッチモード	12時間	48時間

説明：

- 1つのメディア（1つのFileIdで表されるもの）のストレージには多くのファイル（オリジナルファイル、トランスコードファイル、スクリーンキャプチャファイルなど）が含まれ、各ファイルの実際の解凍完了時間は前後する場合があります。VODは各ファイルの具体的な解凍完了状況を管理するのではなく、

メディアを一体とみなして、解凍にかかる可能性のある最大時間によってメディアの解凍完了時間を一括計算しています。この時間より前に、メディア内のすべてのファイルが実際には解凍完了した場合でも、メディアは未解凍状態と表示され、アクセスは許可されません。

- メディアが解凍済みと表示される時点は実際に解凍された時点より遅くなる場合があるため、メディアの解凍レプリカの利用可能時間は予想より短くなる場合があります。解凍後のレプリカの利用可能時間を十分に確保するため、ディープアーカイブストレージを解凍する際は有効期間をもう1日追加しておくことをお勧めします。

ポリシー管理

VODは、開発者が大量のメディアのライフサイクルを一元管理しやすくするため、メディア資産情報とメディアファイル再生統計データを組み合わせて、ポリシーベースのインテリジェント管理システムを開発しました。

VODバックエンドは、管理タスクを毎日実行し、指定されたポリシーを満たすメディアのストレージタイプを変更します。

ポリシー機能

フィルタリングの組み合わせ

ポリシーによって、開発者は一連の条件を指定し、すべての条件を**同時に**満たすメディアのストレージタイプを変更することができます。具体的なフィルタリング条件は次のとおりです。

- ビデオメディア、オーディオメディア、画像メディアなど、メディアの種類をフィルタリング条件とします。
- メディアの作成日をフィルタリング条件とします。次のいくつかの形式があります。
 - 指定された日付より前に作成されたファイル。
 - 指定された日付より後に作成されたファイル。
 - 指定された2つの日付の間に作成されたファイル。
 - 指定された日数より前に作成されたファイル（現在の日付に応じて動的に変更されます）。
- 作成時間は無制限です（すべてのメディアがこの条件を満たします）。
- メディアのカテゴリIDをフィルタリング条件とします。複数のカテゴリIDを同時に指定することも、制限を設定しないこともできます（すべてのメディアがこの条件を満たします）。
- メディアのソースタイプをフィルタリング条件とします。これには、CSSレコーディング、アップロード、ビデオ編集、その他などが含まれます。複数のソースタイプを同時に指定することも、制限を設定しないこともできます（すべてのメディアがこの条件を満たします）。
- メディアの最近の再生時間をフィルタリング条件とします。過去数日間に数回未満しか再生されていないメディアをフィルタリングします。また、制限を設定しないこともできます（すべてのメディアがこの条件を満たします）。

ポリシーの無効化

ポリシーが作成されると自動的に有効になり、開発者は必要に応じてポリシーを無効化することができます。無効化されたポリシーは、再び有効にされるまで、翌日から管理タスクにおいて無視されます。

説明：

ポリシーが無効化された後、そのポリシーがその当日の管理タスクで有効になる場合も、有効にならない場合もあります。

制限

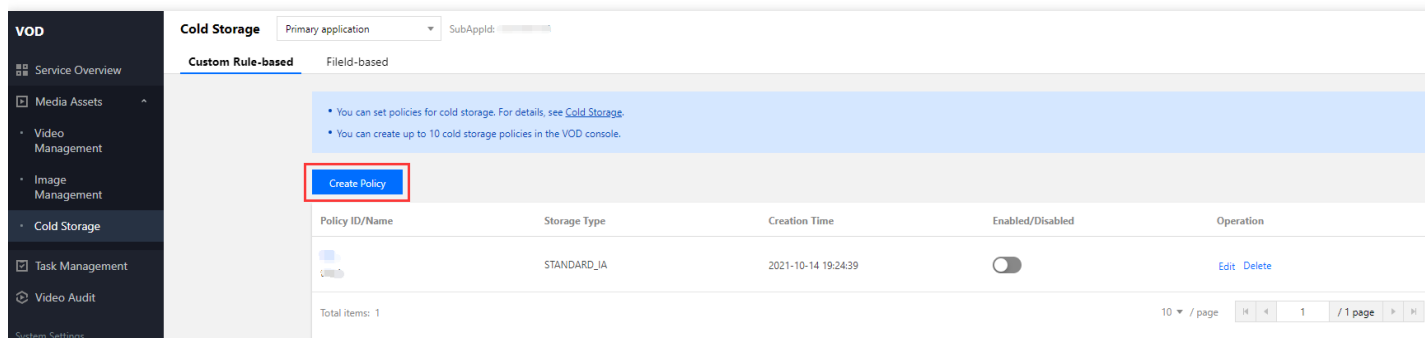
最大10件のポリシーを設定できます。

ポリシーの優先度を指定することはできません。1つのメディアが複数のポリシーに同時にヒットした場合、優先度はターゲットストレージタイプに応じて自動的に決定されます。高から低への優先度としては、ディープアーカイブストレージ>アーカイブストレージ>低頻度ストレージです。

VODによるインテリジェントコールド化ポリシーの設定

手順1：コールド化ポリシーの作成

VODコンソール（非管理者）の左側ナビゲーションバーで、【メディア資産管理】>【メディア資産コールド化】と選択し、【インテリジェントコールド化ポリシー】をクリックします。



The screenshot shows the VOD console interface. On the left is a navigation menu with 'Cold Storage' selected. The main area displays 'Cold Storage' settings with a 'Create Policy' button highlighted in a red box. Below the button is a table with one policy entry.

Policy ID/Name	Storage Type	Creation Time	Enabled/Disabled	Operation
	STANDARD_JA	2021-10-14 19:24:39	<input type="checkbox"/>	Edit Delete

Total items: 1

手順2：コールド化ポリシーの設定

← **Create Policy** test Su

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

Target Storage Class: STANDARD_JA

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_JA storage policy will be triggered.

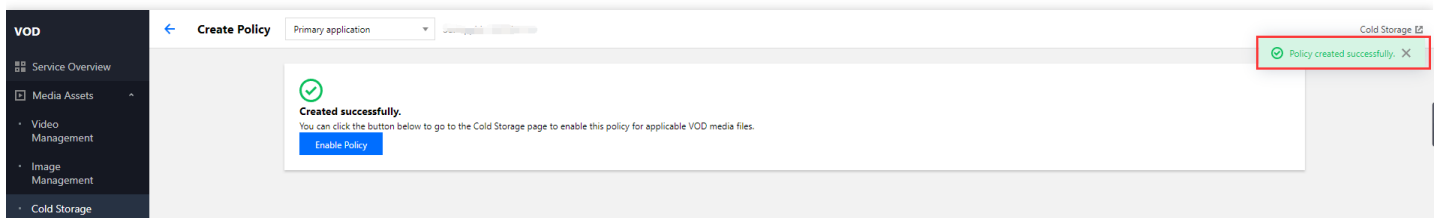
Media File Type

Confirm **Cancel**

ユーザーのニーズに応じてコールド化ポリシーを設定します。

例：図に示すように、30日以内にメディアのビデオ再生回数が300回未満の場合、低頻度のコールド化ポリシーがトリガーされます。VODは、すべてのルールを満たすメディアを選択してコールド化を行います。

手順3：設定の有効化の確認



ビデオ処理

ビデオをトランスコードする方法

最終更新日：：2023-10-26 17:26:52

ご使用にあたっての注意事項

内容紹介

ここでは開発者にVODでビデオをトランスコードする方法およびトランスコードした後の出力結果を取得する方法をご紹介します。

料金

ここで提供するコードは無償のオープンソースですが、使用過程において以下の料金が発生することがあります。Tencent Cloud APIのリクエストスクリプト実行に使用するため、Tencent CloudのCloud Virtual Machine (CVM)を購入します。詳細については、[CVM料金](#)をご参照ください。

VODのストレージはアップロードしたビデオを保存するために消費されます。詳細については、[ストレージ料金](#)および[ストレージリソースパック](#)をご参照ください。

VODトランスコード時間はビデオのトランスコードに使用されます。詳しくは、[トランスコード料金](#)および[リソースパックのトランスコード](#)をご参照ください。

VODのトラフィックはビデオを再生するために消費されます。詳細については、[トラフィック課金](#)および[トラフィックリソースパック](#)をご参照ください。

パラメータ紹介

Tencent Cloud VODビデオトランスコードは現在、以下のビデオフォーマットをサポートしています。

パラメータ	タイプ	詳細説明
入力形式	コンテナ形式	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。
	ビデオコーデック形式	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。
出力形式	コンテナ形式	ビデオ：FLV、MP4、HLS (m3u8+ts)。
		オーディオ：MP3、MP4、OGG、FLAC、m4a。

	画像：GIF、WEBP
ビデオ コーデック 形式	H.264/AVC、H.265/HEVC、AV1。

トランスコードの目標仕様には、エンコード形式、解像度、ビットレートなどのパラメータがあります。VODではトランスコードテンプレートを使用してトランスコードパラメータのグループを表します。トランスコードテンプレートによって、以下のトランスコード関連パラメータを指定することができます。 [ビデオ処理の概要](#)をご参照ください。

分類	パラメータ	説明
ビデオ コーデック	コーデック (Codec)	H.264、H.265およびAV1コーデックをサポート。
	ビットレート (Bitrate)	サポートするビデオビットレートの範囲：10kbps～35Mbps。
	フレームレート (Frame Rate)	サポートするフレームレートの範囲：1fps～60fps。一般的なフレームレートは24fps、25fps、30fps。
	解像度 (Resolution)	サポートする幅の範囲：128px～4096px。 サポートする高さの範囲：128px～4096px。
	GOP 長さ	サポートするGOP長さの範囲：1秒～10秒。
	プロファイル (Profile)	ビデオコーデックがH.264の時は、Baseline、Main、Highのプロファイルをサポートします。 ビデオコーデックがH.265の時は、Mainのプロファイルをサポートします。
	カラースペース (Color Space)	YUV420Pをサポートします。

説明：

コーデック：特定の圧縮技術によって、任意のビデオフォーマットのファイルを、別のビデオフォーマットファイルに変換する方法を指します。H265はH.264に比べてさらに先進的なコーデックトランスコードを採用し、元の画質を失うことなく、ビットレートを大幅に削減し、再生帯域幅を削減できます。

ビットレート：エンコーダが1秒間に発生させるデータの大きさで、単位はkbpsとし、800kbpsの場合、エンコーダが1秒間に800kbのデータを生成することを表わします。

フレームレート (FPS)：1秒間に何枚のフレームが必要かを指します。

解像度：単位インチの中に含まれる画素のドット数。

GOP：通常は2つのフレームの間隔を指します。

通常のトランスコードについては、その明瞭度により、使用に推奨されるビットレート、解像度および設定区間は下表に示すとおりです。

解像度	推奨ビットレート	推奨解像度	解像度の間隔
SD	600	640x480	標準SD（短辺 ≤ 480px）
HD	2000	1280x720	高精細度HD（短辺 ≤ 720px）
FHD	4000	1920x1080	フル高精細度FHD（短辺 ≤ 1080px）
2K	6000	2560x1440	2K（短辺 ≤ 1440px）
4K	8000	3840x2160	4K（短辺 ≤ 2160px）

Tencent Cloud VOD特有の超高速HD（TESHD）は画質の修復と増強、コンテンツアダプティブパラメータの選択、V265エンコーダなど一連のビデオ処理ソリューションを統合しています。ビデオにさらに小さく、さらに明瞭なトランスコード方法を提供し、ネットワークリソースの低消費を保証すると同時に、さらに優れた視覚体験をユーザーにもたらしめます。VODも各種の解像度をプリセットしました。具体的なパラメータは次のとおりです。

解像度	推奨ビットレート	推奨解像度	解像度の間隔
SD	350または設定しない	640x480	標準SD（短辺 ≤ 480px）
HD	1350または設定しない	1280x720	高精細度HD（短辺 ≤ 720px）
FHD	2700または設定しない	1920x1080	フル高精細度FHD（短辺 ≤ 1080px）
2K	3500または設定しない	2560x1440	2K（短辺 ≤ 1440px）
4K	7500または設定しない	3840x2160	4K（短辺 ≤ 2160px）

説明：

設定しない場合、超高速HD（TESHD）はビデオソースのインテリジェント分析に基づき、ビデオの最低ビットレートを設定します。

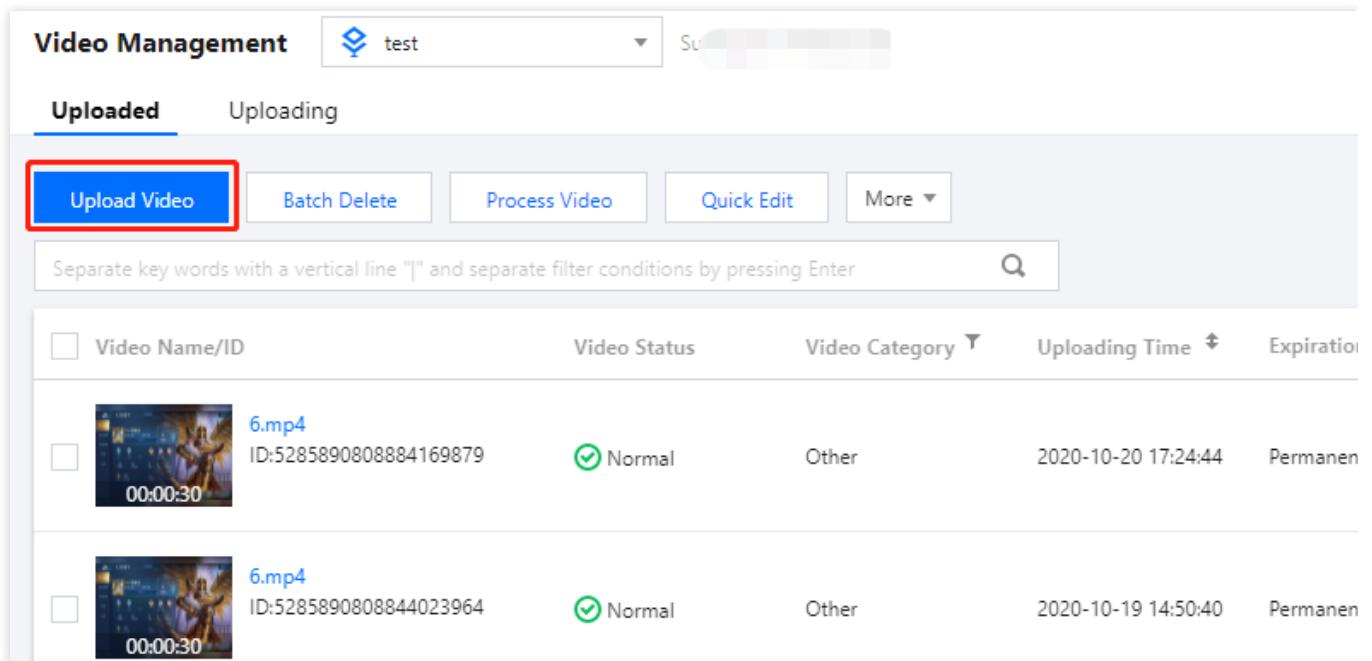
コンソールでのトランスコードの開始

ステップ1：VODをアクティブ化する

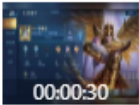
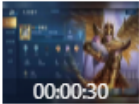
[クイックスタート - ステップ1](#)を参考に、VODサービスをアクティブにします。

ステップ2：ビデオのアップロード

[クイックスタート - ステップ2](#)を参考にして、テストビデオを1つアップロードします。 [ここ](#)をクリックし、このDemoが使用するテストビデオを確認します。対応するFileIdは、243791581135666182です。下図に示すとおりです。



The screenshot shows the 'Video Management' interface. At the top, there's a dropdown menu set to 'test' and a user profile icon. Below that, there are two tabs: 'Uploaded' (selected) and 'Uploading'. A toolbar contains buttons for 'Upload Video' (highlighted with a red box), 'Batch Delete', 'Process Video', 'Quick Edit', and 'More'. A search bar is present with the placeholder text 'Separate key words with a vertical line "|" and separate filter conditions by pressing Enter'. Below the search bar is a table with the following columns: 'Video Name/ID', 'Video Status', 'Video Category', 'Uploading Time', and 'Expiration'. Two video entries are listed:

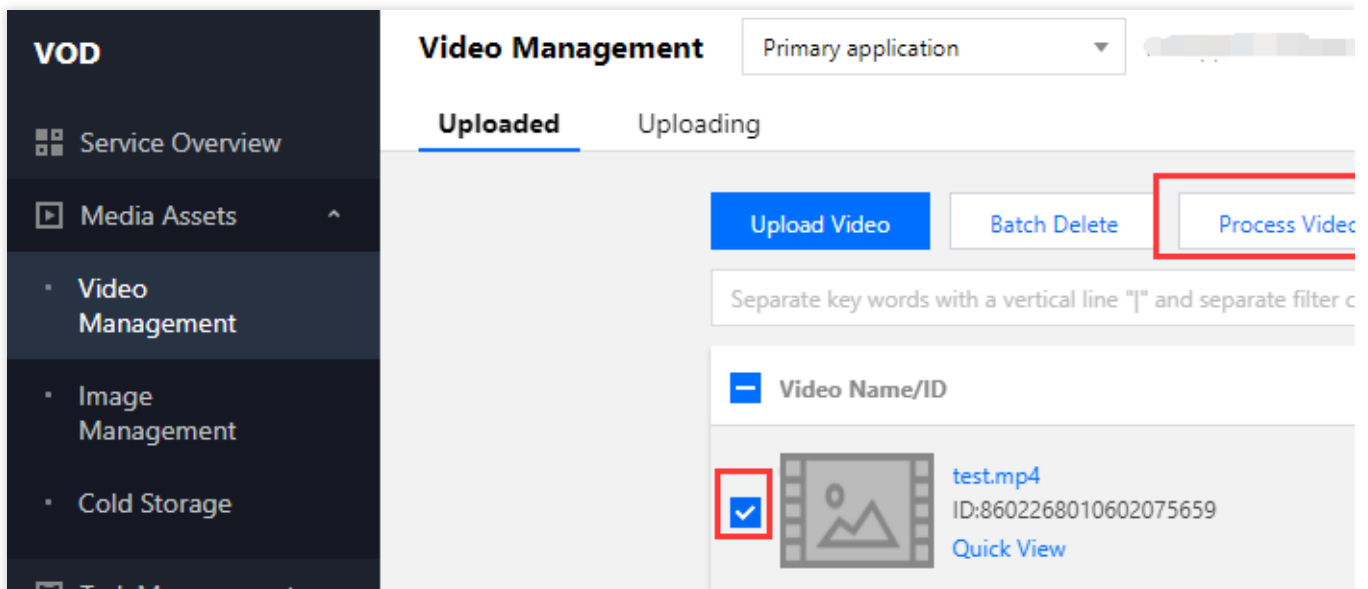
<input type="checkbox"/>	Video Name/ID	Video Status	Video Category	Uploading Time	Expiration
<input type="checkbox"/>	 6.mp4 ID:5285890808884169879	✔ Normal	Other	2020-10-20 17:24:44	Permanen
<input type="checkbox"/>	 6.mp4 ID:5285890808844023964	✔ Normal	Other	2020-10-19 14:50:40	Permanen

説明：

トランスコードに時間がかかりすぎないように、短めのビデオファイルを使用してテストすることをお勧めします（例えば数十秒間のビデオ）。

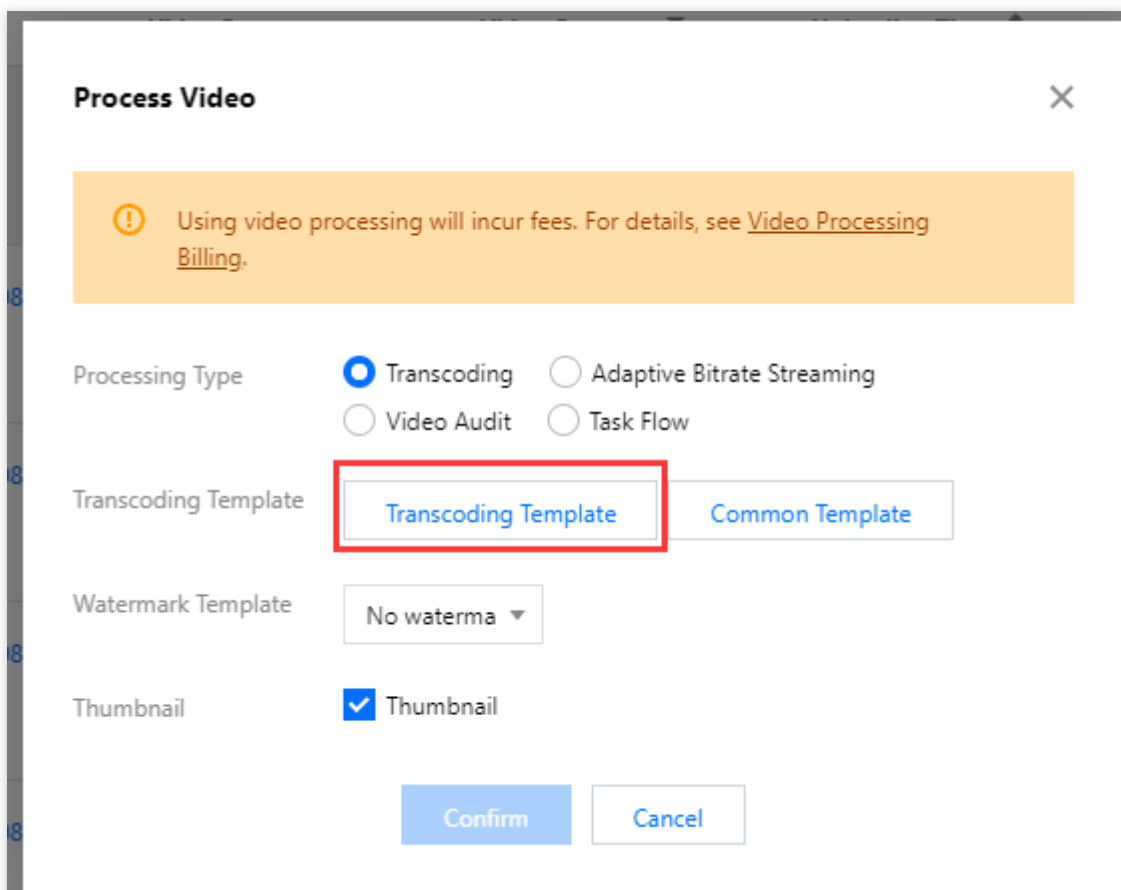
ステップ3：トランスコードの開始

1. コンソールの [オーディオビデオ管理](#)画面でアップロードされたテストビデオにチェックを入れた後、**ビデオ処理**をクリックします。



The screenshot shows the 'Video Management' interface. On the left is a navigation menu with 'Video Management' selected. The main area has tabs for 'Uploaded' and 'Uploading'. At the top right, there are buttons for 'Upload Video', 'Batch Delete', and 'Process Video'. Below these is a search bar and a table of video assets. One asset, 'test.mp4', is selected with a blue checkmark in a red box. The 'Process Video' button is also highlighted with a red box.

2. ポップアップウィンドウで処理タイプを「トランスコード」に選択してから、トランスコードテンプレートをクリックします。



The 'Process Video' dialog box is shown. It contains a warning message: 'Using video processing will incur fees. For details, see [Video Processing Billing](#).' Below this, there are radio buttons for 'Processing Type': 'Transcoding' (selected), 'Adaptive Bitrate Streaming', 'Video Audit', and 'Task Flow'. Under 'Transcoding Template', there are two buttons: 'Transcoding Template' (highlighted with a red box) and 'Common Template'. The 'Watermark Template' is set to 'No watermark'. The 'Thumbnail' checkbox is checked. At the bottom are 'Confirm' and 'Cancel' buttons.

3. 必要なトランスコードテンプレートを選択してからOKをクリックします。このDemoでは、プリセットテンプレートのSTD-H264-MP4-360P（テンプレートID 100010）およびSTD-H264-MP4-540P（テンプレートID 100020）を例にしています。カスタマイズしたトランスコードテンプレートを使用する場合は、[テンプレート設定ドキュメント](#)をご参照ください。

Transcoding Template

Select transcoding template

Enter a template name/ID Q

Template Name/ID	Resolution (px)
<input checked="" type="checkbox"/> STD-H264-MP4-360P 100010 General Transcoding	Width (auto-scaled) x Height (360)
<input checked="" type="checkbox"/> STD-H264-MP4-540P 100020 General Transcoding	Width (auto-scaled) x Height (540)
<input type="checkbox"/> STD-H264-MP4-720P 100030 General Transcoding	Width (auto-scaled) x Height (720)
<input type="checkbox"/> STD-H264-MP4-1080P 100040 General Transcoding	Width (auto-scaled) x Height (1080)

Selected (2)

Template Nam...	Container For...	Te
STD-H264-MP4-360P 100010	mp4	Pr
STD-H264-MP4-540P 100020	mp4	Pr

Confirm
Cancel

4. **OK**をクリックしてトランスコードを開始します。

Process Video

Using video processing will incur fees. For details, see [Video Processing Billing](#).

Processing Type Transcoding Adaptive Bitrate Streaming
 Video Audit Task Flow

Transcoding Template Transcoding Template2 Common Template

Watermark Template No waterma

Thumbnail Thumbnail

Confirm Cancel

5. 「タスクセンター」画面でテストビデオのステータスが「完了」になっていれば、オーディオビデオトランスコーディングが成功したことを表します。

ステップ4：トランスコード結果の確認

1. コンソールの [アプリケーション管理](#) > [メディア資産管理](#) > [オーディオビデオ管理](#) 画面でテストビデオのステータスが「正常」に変わるのを待ちます。これはトランスコーディングが完了したことを示します。テストビデオの右側の [管理](#) をクリックしてビデオ管理画面に進みます。

VOD

- Service Overview
- Media Assets
 - Video Management
 - Image Management
 - Cold Storage
- Task Management

Video Management

Primary application S...

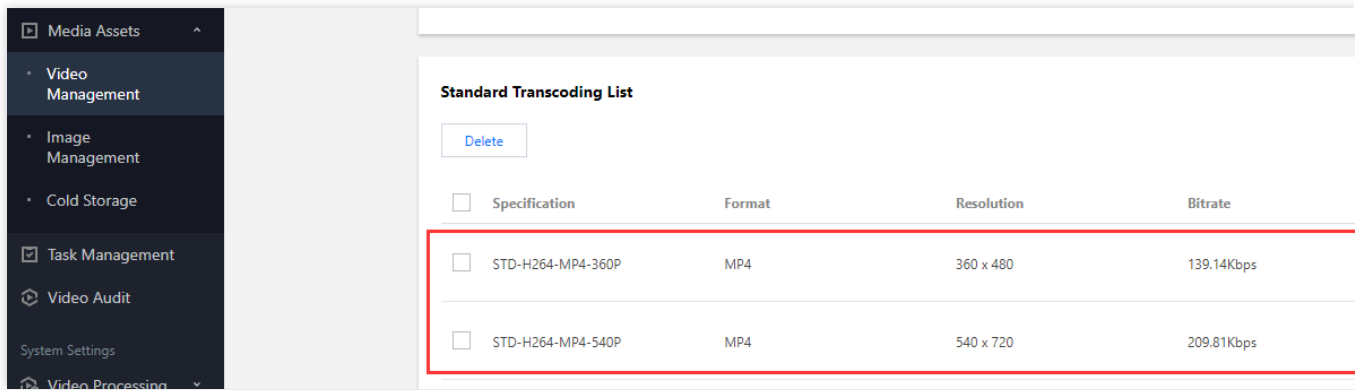
Uploaded Uploading

Upload Video Batch Delete Process Video Quick Edit More

Separate key words with a vertical line "|" and separate filter conditions by pressing Enter

Video Name/ID	Video Status	Video Category	Uploading Time
<input checked="" type="checkbox"/> test.mp4 ID:8602268010602075659 Quick View	Normal	Other	2021-10-17 15:42:56

2. 「基本情報」タブ画面の「標準トランスコーディングリスト」で STD-H264-MP4-360P と STD-H264-MP4-540P の2つの仕様で出力されます。開発者は右側の [プレビュー](#) をクリックしてビデオを直接視聴できます。さらに [アドレスコピー](#) をクリックしてトランスコードされたビデオのURLをコピーして、その他のチャンネルによって視聴者に公開することもできます。



Tencent Cloud APIを呼び出してトランスコードを開始

ステップ1：Tencent Cloud CVMの準備

Tencent Cloud APIのリクエストスクリプトは、1台のTencent Cloud CVM上で実行させる必要があります。要件は次のとおりです。

リージョン：任意。

モデル：公式サイトは最低構成（1コア1GB）であれば問題ありません。

パブリックネットワーク：パブリックIPを有する必要があります。帯域幅は1Mbps以上。

OS：公式パブリックイメージ `Ubuntu Server 16.04.1 LTS 64ビット` または `Ubuntu Server 18.04.1 LTS 64ビット`。

CVMの購入方法は[操作ガイド - インスタンス作成](#)をご参照ください。システムの再インストール方法は[操作ガイド - システム再インストール](#)をご参照ください。

注意：

上述の条件に適合するTencent Cloud CVMがない場合は、その他のパブリックネットワークアクセスを備えたLinux（CentOS、Debianなど）またはMac機器でスクリプトを実行することもできます。ただし、OSの違いによってスクリプトの特定のコマンドを修正する必要があります。具体的な修正方式については、開発者自身で検索してください。

ステップ2：APIキーの取得

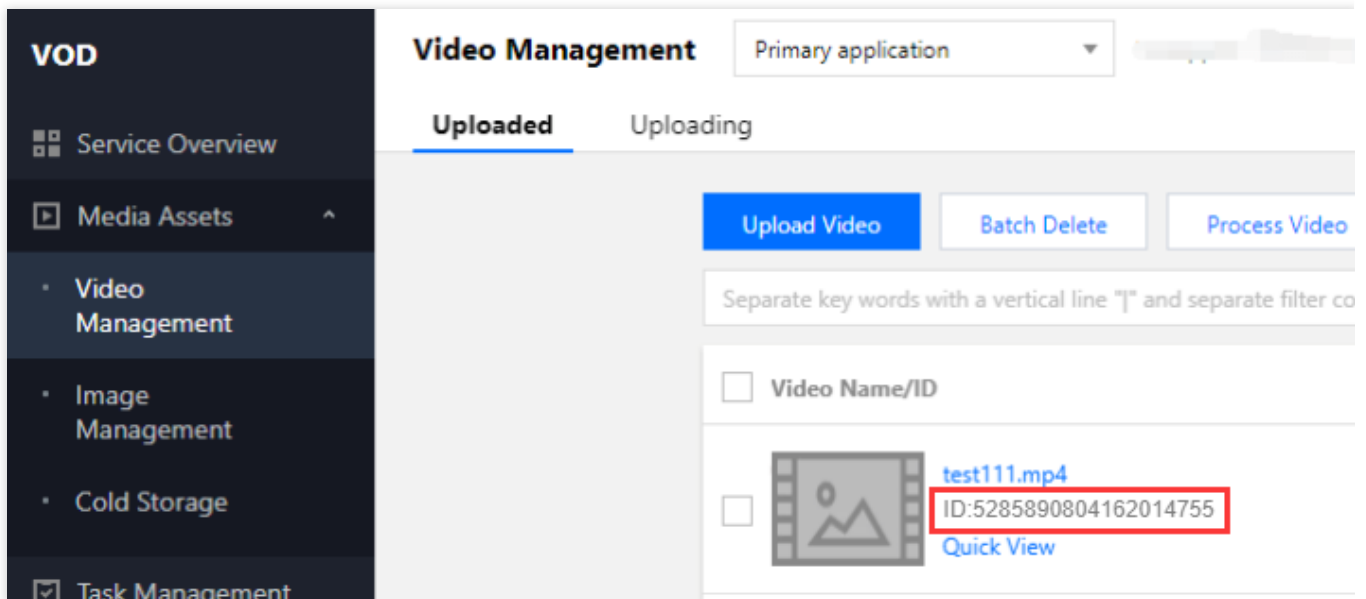
Tencent Cloud APIのリクエストにはAPIキー（SecretIdおよびSecretKey）が必要です。まだキーを作成していない場合は、[キー作成ドキュメント](#)を参照して、新しいAPIキーを作成してください。キーを作成済みの場合は、[キー表示ドキュメント](#)を参照してAPIキーを取得してください。

ステップ3：VODをアクティブ化する

[クイックスタート - ステップ1](#)を参考に、VODサービスをアクティブにします。

ステップ4：ビデオのアップロード

クイックスタート - ステップ2 を参考にして、テストビデオを1つアップロードします。 [ここ](#) をクリックし、この Demo が使用するテストビデオを確認します。対応する FileId は、243791581340253754 です。下図に示すとおりです。

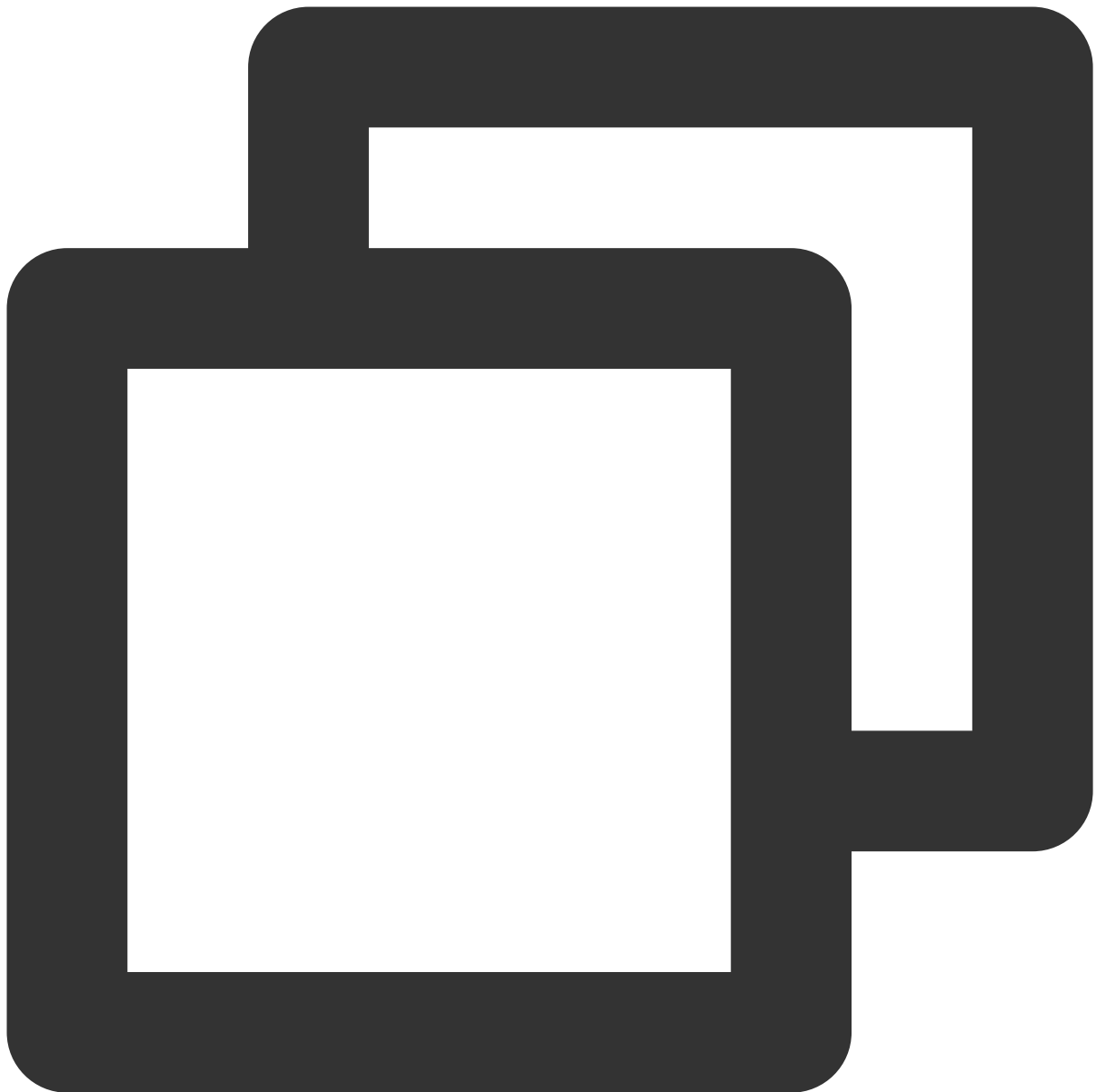


説明：

トランスコードに時間がかかりすぎないように、短めのビデオファイルを使用してテストすることをお勧めします（例えば数十秒間のビデオ）。

ステップ5：トランスコードの開始

1. [ステップ1](#)の中で準備したCVM（ログイン方法の詳細は [操作ガイド - Linuxにログイン](#)）をご参照ください）にログインして、リモートターミナルで以下のコマンドを入力して実行します。

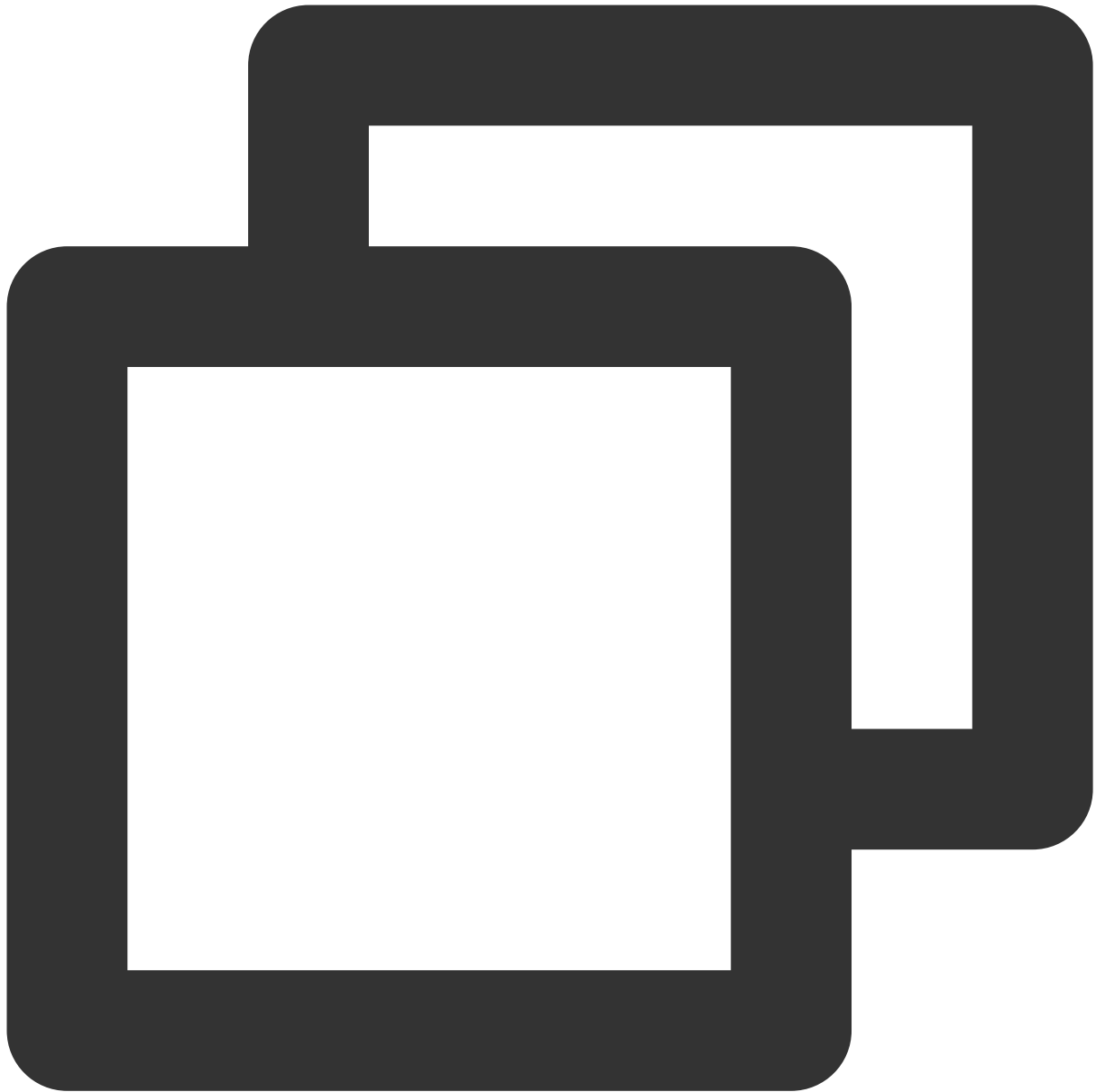


```
ubuntu@VM-69-2-ubuntu:~$ export SECRET_ID=AKxxxxxxxxxxxxxxxxxxxxxxxxxxxx; export SECRET_
```

説明：

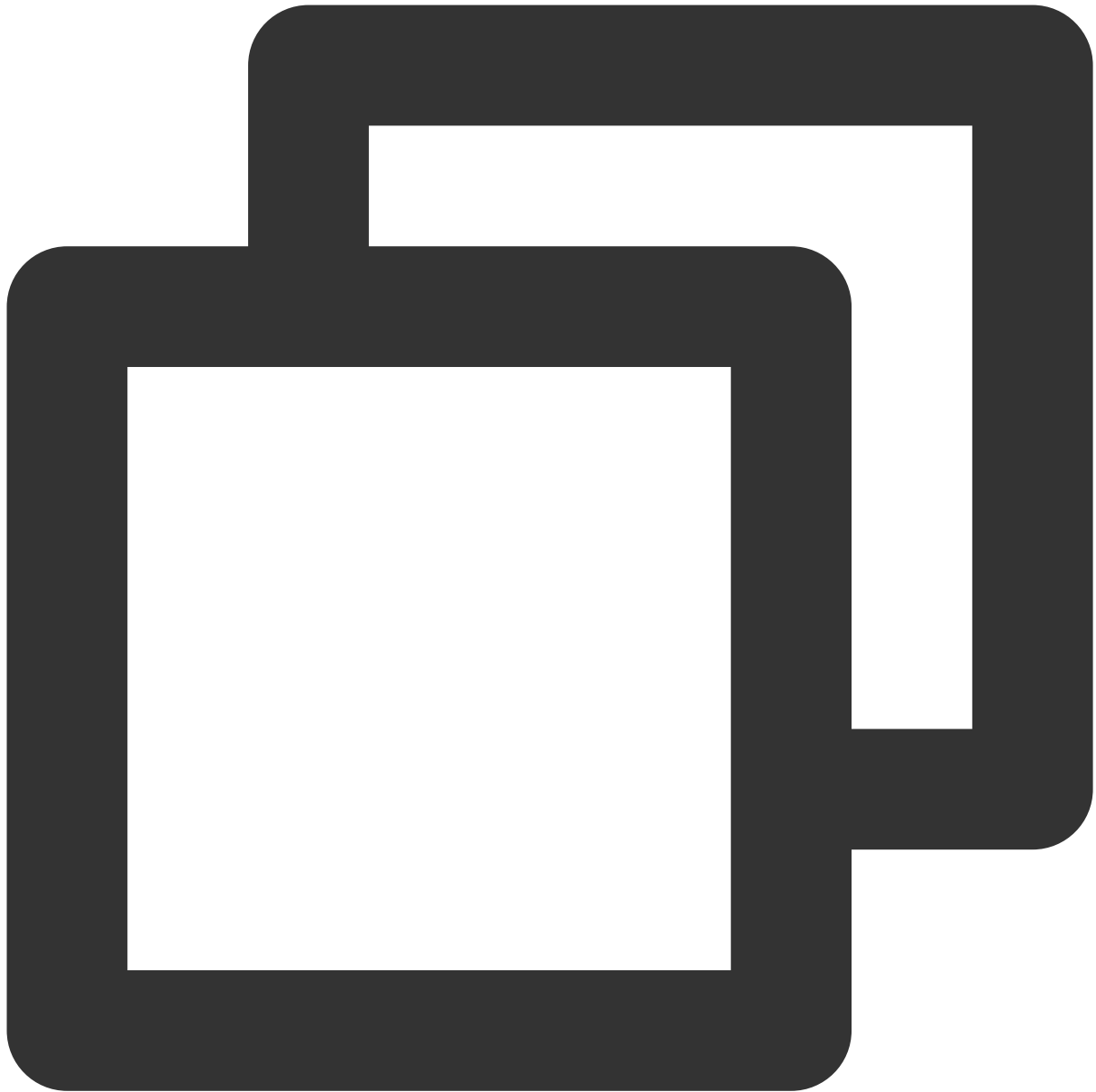
コマンドのSECRET_IDおよびSECRET_KEYを[ステップ2](#)で取得したコンテンツに割り当ててください。

2. このコマンドでは、GithubからDemoソースコードをダウンロードして、インストールスクリプトを自動的に実行します。インストールのプロセスには数分間必要になり（CVMのネットワーク状況次第）、その間、リモートターミナルは以下に示す情報を出力します。



```
[2020-06-15 20:39:56]pip3のインストール開始。
[2020-06-15 20:40:06]pip3のインストール成功。
[2020-06-15 20:40:06]Tencent Cloud API Python SDKのインストール開始。
[2020-06-15 20:40:07]Tencent Cloud API Python SDKのインストール完了。
[2020-06-15 20:40:07]APIパラメータの設定開始。
[2020-06-15 20:40:07]APIパラメータの設定完了。
```

3. `process_media.py` スクリプトを実行してトランスコードを開始：

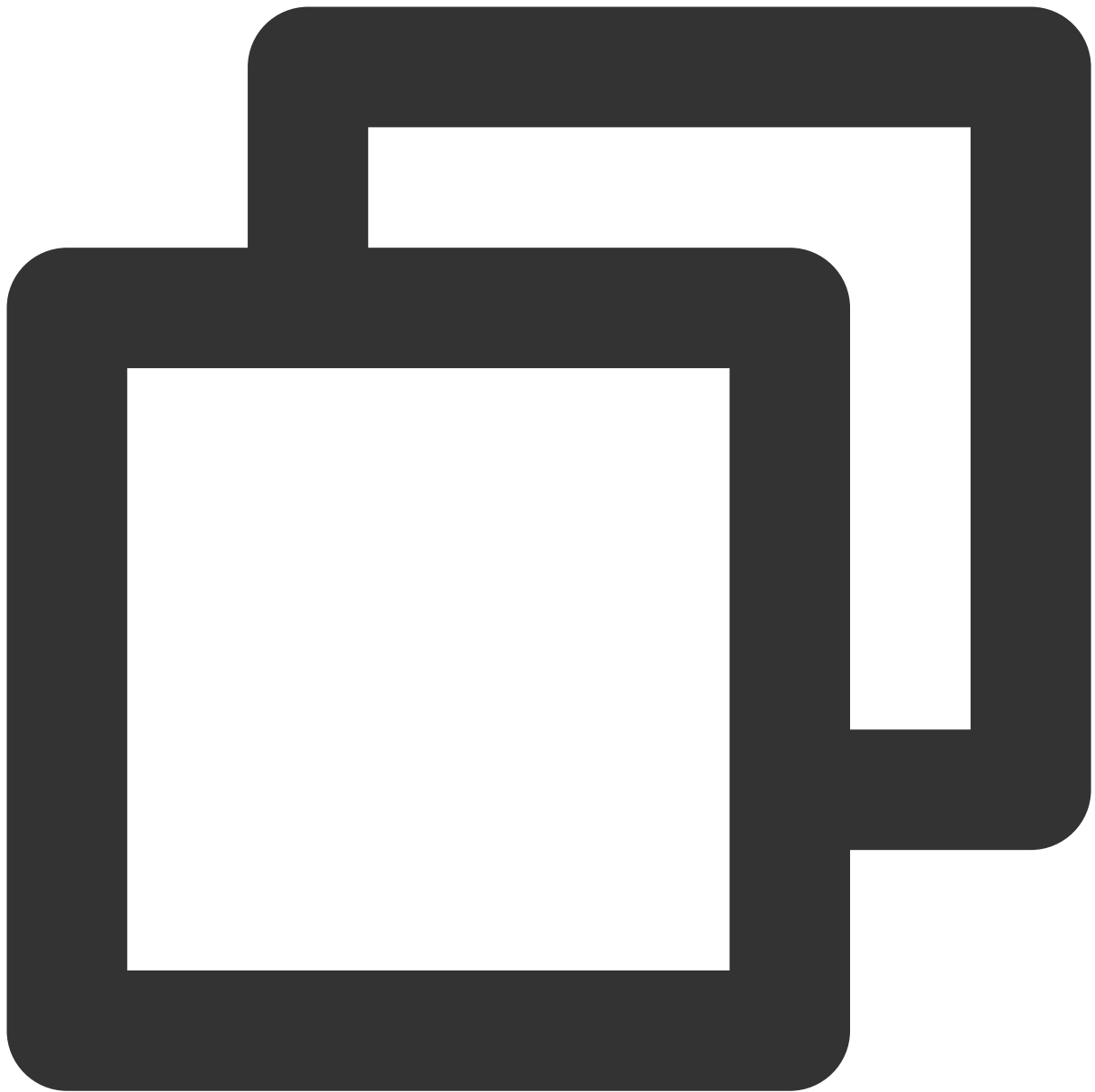


```
ubuntu@VM-69-2-ubuntu:~$ cd ~/vod-server-demo/transcode_api/; python3 process_media
```

説明：

コマンドの243791581340253754を[ステップ4](#)で取得して実際のFileIdに置換します。

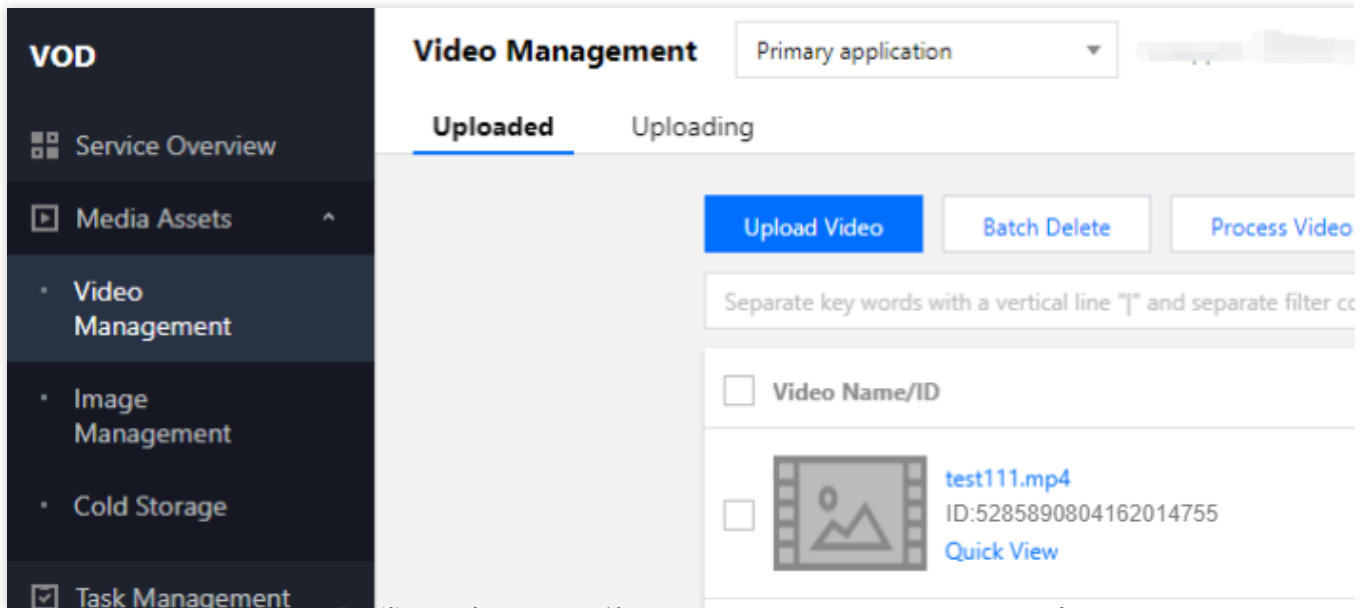
4. このコマンドは、243791581340253754のビデオに [ProcessMedia](#) リクエストを開始して、[VOD プリセットトランスコードテンプレート 100010](#) および [100020](#) の2つの仕様に基づいてトランスコードし、リクエストの応答内容を出力します。



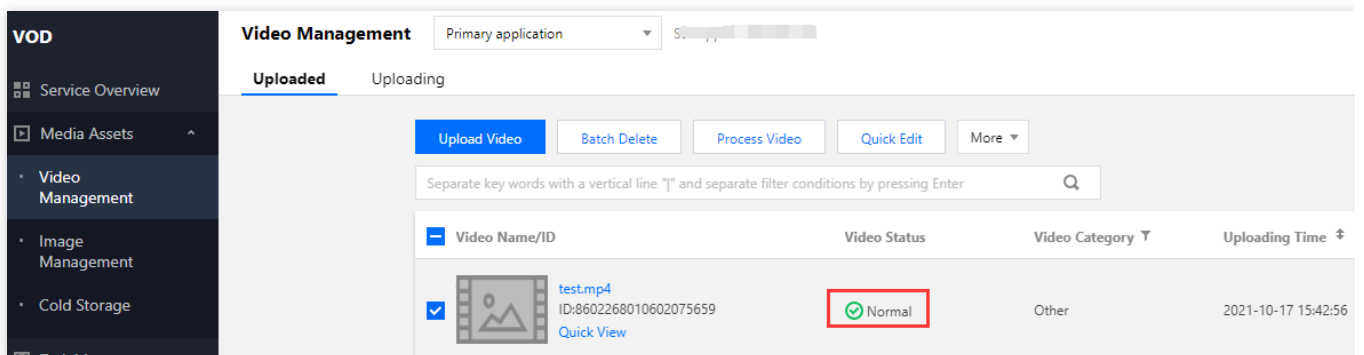
```
{"TaskId": "1400329073-procedurev2-f6bf6f01612369b6db30f2224792a2aft0", "RequestId"}
```

ステップ6：トランスコード結果の確認

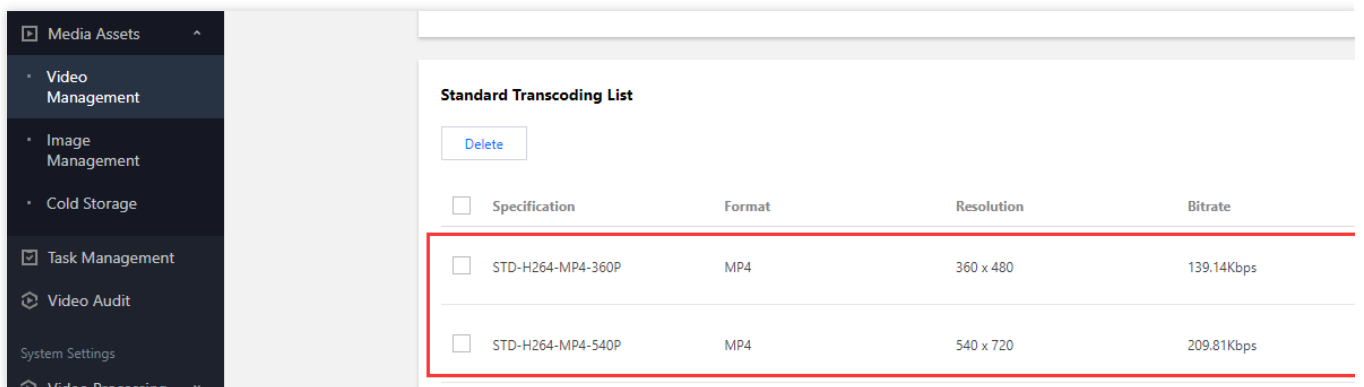
1. 「タスクセンター」画面でテストビデオのステータスが「完了」になっていれば、オーディオビデオトランスコーディングが成功したことを表します。



2. テストビデオのステータスが「正常」になるのを待ちます。これはトランスコードが完了したことを示します。テストビデオの右側の**管理**をクリックすると、ビデオ管理画面に進みます。



3. 「基本情報」タブ画面の「標準トランスコーディングリスト」で、対応する仕様のビデオが出力されます。開発者は右側の**プレビュー**をクリックしてビデオを直接視聴できます。さらに**アドレスコピー**をクリックしてトランスコードされたビデオのURLをコピーして、その他のチャンネルによって視聴者に公開することもできます。



ビデオアップロード後の自動トランスコード（タスクフロー）

VODは、コンソールからのアップロード、サーバーからのアップロード、クライアントからのアップロード、URLからのプルアップロードなどの複数の種類のビデオアップロード方法があります（詳細は[メディアアップロードの概要](#)をご参照ください）。アップロード方式はすべて指定した1個の[タスクフロー](#)をサポートしていて、アップロードの完了後にトランスコードを自動的にトリガーします。

ビデオアップロード後の自動トランスコード（イベント通知）

VODバックエンドは、ビデオアップロードの完了後およびトランスコードタスクの完了後に、[イベントの通知](#)リクエストを開始します。イベント通知システムを使用して、新しくアップロードされたビデオのトランスコードを開始し、イベント通知によってトランスコード結果を自動的に取得できます（ここに表示した方法は、コンソールでトランスコード結果を手動で表示するものです）。イベント通知の使用方法は、ベストプラクティスのドキュメント単独の[イベント通知の受信](#)に詳細な説明があります。

メディアの内容審査を実行する方法

最終更新日：2023-10-26 17:26:52

メディア制作事業にはコンテンツの制御不可能性が存在しますが、コンプライアンスに反するコンテンツが存在する場合、関連する法的リスクやブランド毀損が生じる可能性があります。VODでは事業の健全性とサービスの安定性を確保するため、[オーディオビデオ審査](#)および[画像審査](#)機能を提供しています。

オーディオビデオコンテンツ審査

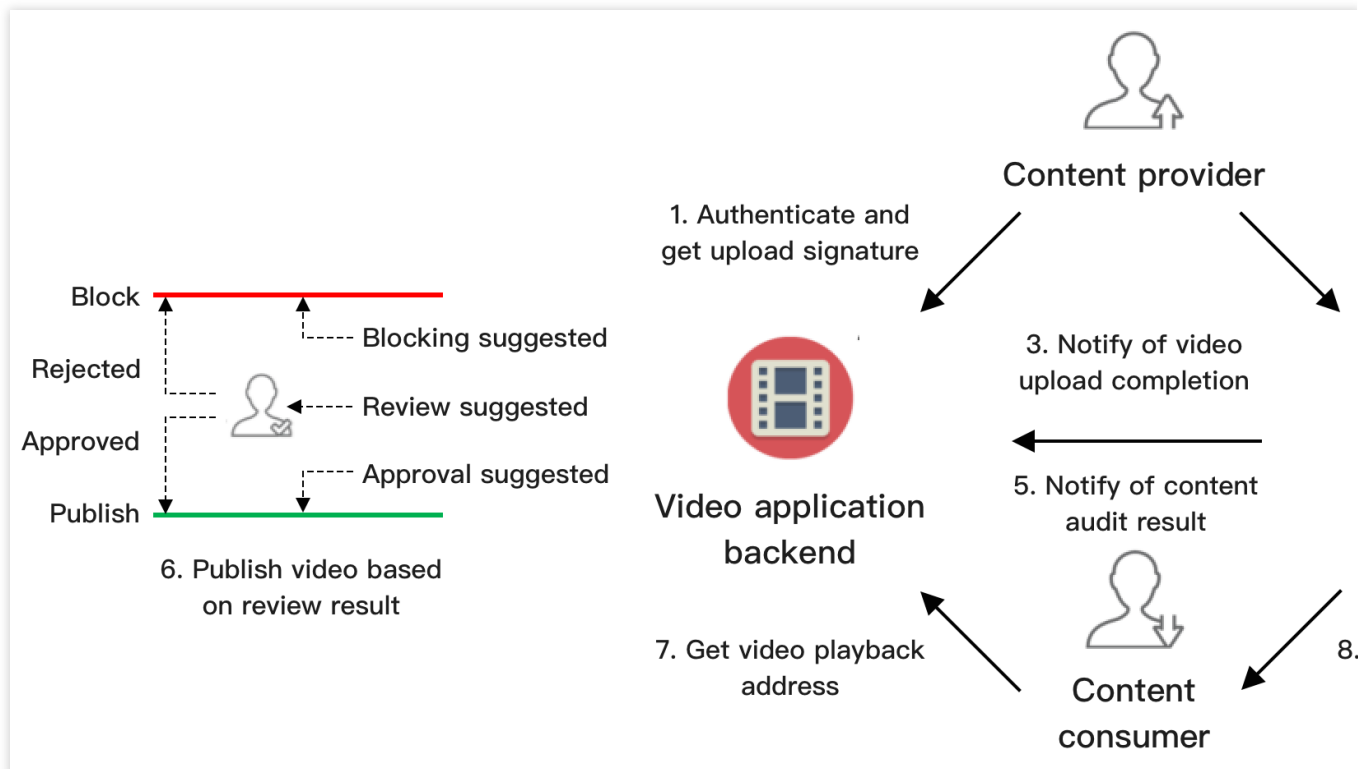
VODは[オーディオビデオコンテンツ審査](#)機能を提供して、オーディオビデオコンテンツの審査を開始し、認識結果に審査の推奨事項を反映することができます（再審査や承認の推奨）。開発者はその結果に基づいて、オーディオビデオのコンテンツを処理することができます。

オーディオビデオコンテンツ審査を開始するには、次の2つの方法があります。

1. VOD [コンソール](#)から操作します。
2. サーバーAPI [オーディオビデオコンテンツ審査](#)経由で呼び出します。

サーバーAPIオーディオビデオコンテンツ審査の呼び出し

たとえば、特定のビデオアプリはVODのオーディオとビデオの審査サービスにアクセスするとします。サーバー側のAPI呼び出しプロセスは次のとおりです：



1. Appバックエンドがコンテンツプロバイダを認証し、認証に通った後 **クライアントからのアップロード署名** ビデオを配布します。
 2. コンテンツプロバイダは共有するコンテンツをVODにアップロードします。
 3. VODは、アップロードしたビデオのFileIdおよび再生URLなどの **関連情報** をAppバックエンドに通知します。
 4. VODは署名のアップロード時に `procedure` パラメータで設定されたオーディオビデオ審査タスク(`ReviewAudioVideoTask`)を実行します。
 5. VODは**オーディオビデオ審査の完了**によって、Appバックエンドに審査結果を通知します。
審査結果が「block」の場合、非常に疑わしいので、そのままブロックすることをお勧めします。
審査結果が「pass」の場合、疑わしさは低いため、そのまま合格させることをお勧めします。
審査結果が「review」の場合、疑わしさが高いため、手動で再審査を行うことをお勧めします。
 6. Appバックエンドは、「承認の提案」のビデオ、および「再審査の提案」、並びに手動で再点検を行って合格したビデオを公開します。
 7. コンテンツ消費者は、Appバックエンドからビデオ再生URLをリクエストします。
 8. コンテンツ消費者は、再生URLを介してVODからビデオをアクセラレーション再生します。
- それらのうち、ステップ4~6の手順により、コンテンツ消費者がステップ7で取得したビデオが、審査検証を経てコンプライアンスに沿ったビデオであるということが保証されます。

注意：

ここで紹介したフローは「審査後に公開」モード（審査に合格したビデオのみ公開）になります。必要があれば、「公開後に審査」モード（ビデオアップロード完了後に公開して、審査で不適切なコンテンツが発見されればビデオを削除）を採用することもできます。

画像審査

VODが提供する[画像審査](#)機能では、画像コンテンツに対する審査を開始し、審査結果の中でアドバイス（再審査を推奨、承認を推奨）を行うことができます。オーディオビデオコンテンツの審査と異なり、画像審査は同期処理プロセスであり、タスクの開始後すぐに結果を取得できます。ユーザー側はアドバイスに沿って画像を公開するかどうかを決定できます。

審査結果が「**block**」の場合、非常に疑わしいので、そのまま画像をブロックすることをお勧めします。

審査結果が「**pass**」の場合、疑わしさは低いため、そのまま合格させることをお勧めします。

審査結果が「**review**」の場合、疑わしさが高いため、手動で再審査を行うことをお勧めします。

画像審査を開始する方法は2つあります。

1. [VODコンソール](#)から操作します。
2. サーバーAPI[画像審査](#)により呼び出します。

料金関連

具体的な料金については、[購入ガイド](#)をご参照ください。

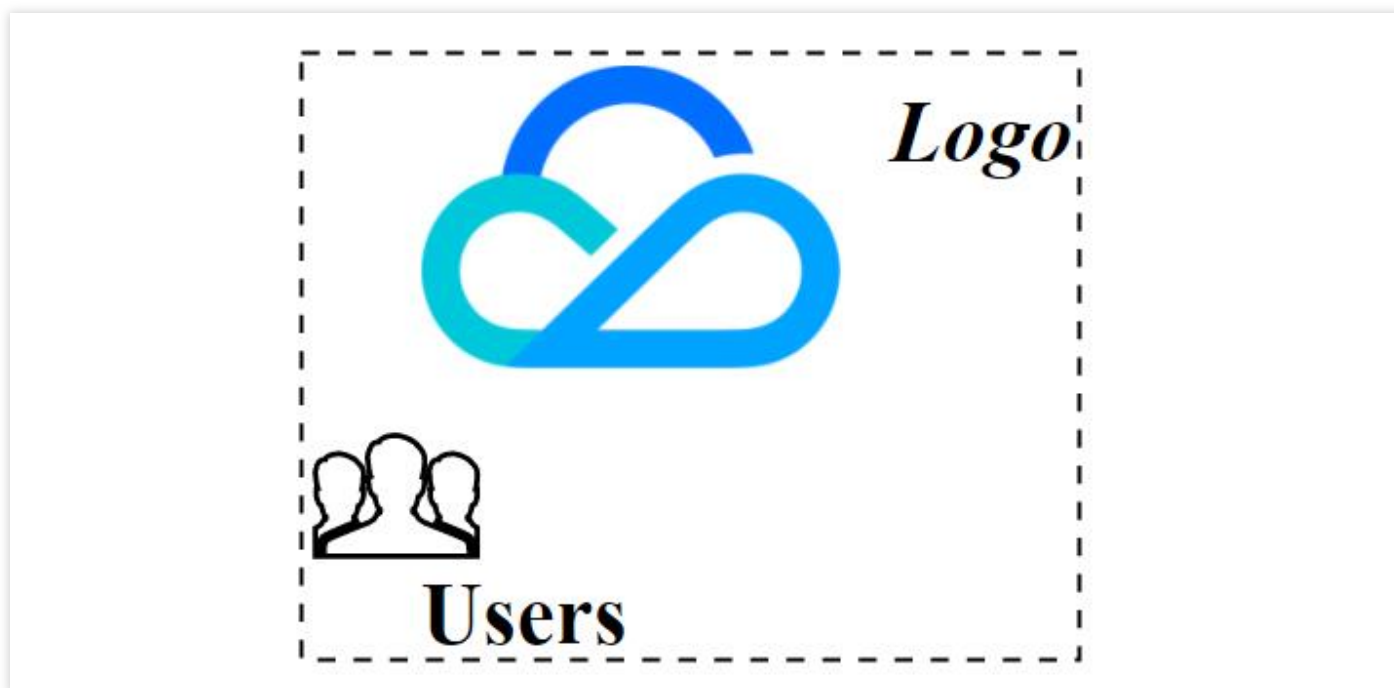
ビデオに複雑なウォーターマークを追加する方法

最終更新日： : 2023-10-26 17:26:52

VOD [ウォーターマーク](#) が提供する画像ウォーターマークと文字ウォーターマークは簡単なウォーターマークの要件を実現できますが、複雑なウォーターマークの要件（例えば、画像と文字が混在するウォーターマーク、フィルター等）については、対応する難度は高くなります。このため、VODがSVG(Scalable Vector Graphics)ウォーターマークを導入することで使用者はウォーターマークコンテンツを便利かつ柔軟にカスタマイズでき、画像や文字コンテンツを自由に配置し、画像作成、フィルターの追加、グラデーション等の特殊効果に対応することができます。

SVGの紹介

- [SVG](#)(Scalable Vector Graphics) はXMLをベースにしたマークアップ言語です。2次元ベクトル画像を記述するために用いられるオープンネットワーク標準で、CSS、DOM、JavaScript等の各種ネットワーク標準に広く応用されています。
- VODはSVGの幅と高さに制限がないだけでなく、完全な要素を含むことができる最小の矩形を自動計算し、これをSVGウォーターマークの元のサイズ（幅と高さ）とします。例：最小の矩形を計算して下図の点線で囲まれた部分を取得します。



説明：

- オンライン上には多くのWebベースの無料のSVGオンラインデバッグツールがあり、お客様はこのツールを使用して満足のいく図形を作成し、完成後XMLにエクスポートすることができます。
- 詳細な設定（要素の整列属性、フォントサイズ等）については、手動でXMLを変更できます。最終的な効果を確認してXMLコードを取得します。

SVGウォーターマークの使用フロー

手順1：SVGウォーターマーク効果のデバッグ

1. SVGデバッグツールを使用して（簡単な[オンラインHTML WYSIWYG編集ツール](#)、またはより豊富な[オンラインSVGエディタ](#)を操作して）、お気に入りの図形をデバッグし、WYSIWYG全行程を操作します。完了後、SVGを `.html` 拡張子ファイルで保存すると、ブラウザを開いて効果を表示することができます。

説明：

異なる要素間で整列の要件がある場合、各要素の整列クラス属性を柔軟に活用し、要素の内容を変更して有効かどうか検証することをお勧めします（例：画像と文字を混在させて、テキストの長さを増減し、画像とテキストが常に整列の要件を満たしていることを確認する）。完全なチュートリアルは以下をご参照ください。[SVGチュートリアル](#)。具体的な事例はテキスト下の[事例](#)を参照することができます。

手順2：SVGウォーターマークテンプレートの作成

[ウォーターマークテンプレートの作成] (<https://www.tencentcloud.com/document/product/266/34163>) インターフェースを呼び出し、ウォーターマーク位置、幅と高さ等の情報を指定します。作成完了後、ウォーターマークテンプレートIDを取得します。

手順3：SVGビデオウォーターマークの追加

ビデオ処理タスクの開始、およびタスク結果を取得する方法については、[ビデオ処理タスクシステム](#)をご参照ください。

[ビデオ処理](#) インターフェースで説明します。[事例2 ウォーターマーク付きのトランスコードタスクの開始](#)

で、`WatermarkSet.N.Definition` は手順2で取得したテンプレートIDを使用します。また、SVGウォーターマークを使用しているため、手順1で取得したXMLをパラメータ `WatermarkSet.N.SvgContent` に渡す必要があります。

事例

以下では、具体的な事例を挙げ複雑なウォーターマークの追加方法をご紹介します。可変テキストの文字と画像が混在するウォーターマークを実現します。

要件の説明

特定の顧客はビデオにウォーターマークを追加する必要があります。要件は以下のとおりです。

- ウォーターマークはブランドLogoとログインユーザーのアカウントID（テキスト）で構成されます。
- ウォーターマークをビデオの右上隅に設定し、ビデオの右枠線からのウォーターマーク始点のオフセットサイズはビデオ幅の2%、ビデオの上部枠線からのオフセットサイズはビデオの高さの5%とします。
- ウォーターマークの幅はビデオの短辺の30%、高さは元の図などの比例に応じて拡大縮小します。
- 文字はブランドLogoを上詰め右側に揃えます。
- 文字フォントはゴシック体、背景色は白、ガウシアンぼかしの影、影の色を黒に設定します。ブランドLogoを拡大縮小しない場合の文字サイズは50ピクセルです。

手順1：SVGウォーターマーク効果のデバッグ

SVGのXMLソースコードは以下のとおりです。

```
<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,iVBORw0KGgoAAAANSUhEUgAAQAAAAC5CAYAAADHwOFvAAAAAXNSR0IArs4c6QAAARnQU1BAACxjwv8YQUAAAAJcEhZcwAADsMAAA7DAcdvqGQAACHaSURBVHhe7Z0JmF1FlccLZ2QQEEQWZVUwI4oDqAPIOCOOiKg4IyOLyoCiomFJv+4EQUdRZmAGcQnIGFDW5N330gkRCAQIm6GBJH1fd5oQsUnS796XPYQsJGSBJCR97vxP3YJJ0pWk03239975f9/vu90S+9atqlN7naNENaDmYHc1hPZXQ40j1FA6VTXSRaAGlaOR4An8PA0sBCTVYxBsHyL8+6V4VvAs4TkBzZvw/Dm4AH/7BPx+mGoK3qfODf7GvF0kEiWqS4P9YPAnwBC/B408DUwCc2GkG+2GHSX0Bt41EzyG32/C8xXwjBoc7G1SjXKJi1Vz8Lcw+E/D+JrABOBrQ7QaaArkaBXoQppGgu+jYfqwSblIJNplNQfvUpcFe6sm+ir4PYyrAsNabzW+zIFpRCOtW8+deP4XOE5dGOxhvkwkEm1XP6D3wti/AK05Hc9FvY2raulUueA6NGqfVOcGu5uvFY1ESgW7qWE0CEb/UzAdhr/JYkA1Ao8MaJJq0GsXB5oMEInqVI10IhgNVgIMnW1GU4PkgAfmzF/Bs/DTG6IRHWgC2gvVPozUf1btSHYDKSuoLeAg7z4lBocvNvkkkhUY2o091ANdBYq+mQxfBu0Aflyv97e5EVQkahmFB7OeTrs7WYVX9iC11UuGIWG4KMm90SiK1QQ7KYrcY7GwvATOKBTa9Ba5N31mBYcYHJUJkOSDaP3Y7h/LSrwKnlFvpMjspoSP8Tz78zuSsSZVTc6zcF/4qK29mrIgv9R2+N0kOqIfiYyWmRKG06kt6LSnoLKiufgrNXZGFg5OgV1UQXy26BKDviXj9HJ8P4Z1krrRAD9LC6nA43JSASpStE2svR5WCNvaIK8UG+Gkpfk2vJonTER1lzwTg8a/jobsbhpfvGLA/BJEoMQ2hY1H5/mqt1EKY6ENVNAFlsr8pHZEOJvFws5G+AZZaK6OQIjRDNdDx+nKVSBS5+HhqAzWioq21V0AhfbRXpC9KIYCKVmHP/98y368KXldN9HVTciLRAMW+7hppRDj
```

XtFY4IWvk9NHRwbJDIBqYcrQPjN8Bm3tVMiHj0Bv60JA0AqJ+i f3YNQXjUJGq1FEH+xCkbsC3EO8CPIUZ
ohrou/j5DAv/hv/+Q3AFfh6OJ19imgIWg+rMgxy9iedgU6IiUR/FF09y9MdeFSqTaCec7Kq7DPJgCHq+z
+gLScPoPYo9CvdHfMKRj9zyFGgYHYq/y05M/gvPJ8By/Fwlv5uRTm7YxMeAQe8KjZ898WZ4zq+NfiHSOB
49+sVqKH3CpD4ZcaPQeJyGd9+IdEwDGT8JidFQA10gjYBox+IK0kTXosJktXd7HUB/INL4Ld0rZ0HcYPL
+eyM1g+kgm+s1+mo2pjk10XbVSBeCjDnv0L39HFTgn+mAG/0d0ichjmUQej+6D2ywf0+a0DJwnEmtSLSF
hmJIm6VoO6GzzDb09ufpXrbaxI1VeD36Ffv3pUSOPHUZfcikUiSCLsMcWsfWs1SYxOEhNLWds2vi3vuld
LhpCLJzfLqJnlJXBfuaFIrqWhzpnkfpWStK0rCv/DDg5z4mdbWhcG31aHzX3Wgm019f4QXeBjRK4likzs
UVs5G3+9Le5+b7BTRcXR3sZ1JWm+JD0U10Cr61XRuhNS+SQk+xLjQpE9W1htL59sqRIBwnYBidbFUH+L
zCY3B1TDA5b3yI0ly+t7A0SZVoroSx7RPdYFKLzjepH0J1qs4G1COOnrnTaI8W9dlUJfigyw5mmipDMnA
c/0GcWelxScWOQR6WmsDoUORZpMaUe1LO/G8yloZYocI7/4zkICYWypcJDWP+bPCnm8xE/p0/BeTGlFNI
4f+qVQ0vb13J95fWYv8UYrvMORotj3/4oZK+iCTqIYVrvo/Ya8AcaJXnJv14pdoxxpGg9AIPLMuwDchRT
WsRh1iK1mvPm87p+DGR9Q3cTzARnq8V17GDd8X4AZIVIO6LPggKlWYwTty9IZqoItMCKs7osHBviiVJ63
5GifsByHL9y1E/ZCO3hNchwqV3IEfHSIMxi8r/f2Xjr2Q9G6N9q1wukmBqCbUQEeiYJfZCzwGwmnG1TLs
j0BhI9DaK4/jJEeTZL2mZqS3/e61FnQs6Ou7N5qXi6LQFzi+5ehFe37HRAOdZ94uqmoNoY/CINdbCzkW6
D70/BKqKmpxaPBGWmLP8xjIOV+kHGtB2iGmpYDjoVMNpYPNm0VRK0f/jvJMzmdDjr5p3iyqSoXn/VdZCz
dy0DtXzEBRfApDsV+DvE7I7RhNlyvDVStUFr7zbS3YqKFNqJjnmxeL4hSHZG+kB+z1EDF8hqMh+A/z1F
Vic/b52iRtWCjJkctsuKfoC7XXoYWWssianL0mIwCq1FNNNRaoFHdn/vkck/yte5yUwFaJ0aSp82bxVV
hX5Me6HgZtgLNEowRJSFonTEvbIO22Yrl6ihETLCqybl6Asg/1N/HJijGj321orCLd7F1rKJkhzeMYT2N
28VZV65YJS1IKPlddUUFNK8UZSKgt3QAPwEBhq/f0H2VyCqAnFL3UgrYUYKXSLeaMoTbE7r0aaaS+jCG
FX4qIqUCNdEH+PQCvwjgPNG0Vpky4115004Dva/AZEd4p4sCquhGhafh5axrpJTxn4/m4eZso0+J5ua3
AoyRHN+jhpygb4nWYxu3dFdDOWHz8m6fAH8BQcKZqoJPQq39cBy65NNgvdAxqKVM+d8BXk+VyUBWoMTgC
hRuvp98cvSghpTKoRvo2YLdrG1BGJfAr/Hy2vgnKwV9kL780xAUe9/CfTxdK75898TYdh3hjYxfVo/SK8
D1Wo40KDiTBvuxFILHGxC1/A71sNdZIoCfEw49IleUNo2NhoPEF1+CpBR8/FY1EGVSOLrUabmTQMn3EWC
QSZU16/1+0G25E50h08zKRSJQpsRvnuCPKsDcakUiUQf2YDrIabVTw3r9sL41EGRX7cbcZblTk6EH1eQk
WIRJ1Uw3B1VbdjYoc5cybRCJRZtTa+reqq2tvddWmMerKTT2qef0aNXztRvXh1YG6d8Vm1bJ0tRr76ho1
fvHqrbj/lbX4b2tUftkGdfdrGBP1dY+64c116ppNG9QVmwPV1LPFaULagOc/mjeKqk7Bbuo+eo+6i96vH
PoHlaezVIEuBzeDPJikivQ8/tts/Ny9FQ49h+dToIj/H//7K/G/nQ20x+/7qzuCPbWjUlFCasUwvLNyhC
qVT1dt5Wbleo+C2er50ZuU6weqFAHPziH1yML1quVVUr9bsx4Nw2JVXPrPamJZH9Ui9jYR6HRbqFLYbT
3gmlglSqqQY8MwoOG4EVQQGMwFL+fqFqCA6RBiFqdne9W7f4Jyi1fBQnThcG/gmfPVkYbJ2HDSlKVvBJ
+/h80QP+i2hbIzbCsquAHwxjPzxOPJ/fm6+2GGxMObcQ7PTwfabEBQdIY9FdB8C41ZdbRMPbrYHwz8Xxj
K6NMFW89KCNNv1Jt/rFq3Dg5GpyWHNoLRvcNwEb3Gp49VuNMGodIOcHrmFq0Ik0Xy8igr2qdfQCG9ufD0
FrBpq0NL5NsBlPVVO87qn2m+IxLQuOCv4FRHQ1DuwmGtriX8WUSTBcK5KhR9AU1IdjTfIlIi1vGUvkw9K
g/A93bGFj14PqzVXvlatXhHW6+TBSluJ4U6GQYfQHPiofzCeHQZvA80n+euiPY13xZHWvys/updgy1S/4
iGD/1MqqqA9/g+gv1AmXrXdk4FJXy9CkYzRiAntRiWNUGNwQF+gv4IUY0dRiAdPj8GH71chjMqzCcGjD8
beDGzPXQEPiXqCmz3mu+WrrLQo8/JvgwjOReGMybVkoQeqqH3/YCnl+vj4aAV/Td7rNg+H+xGk5t0q7aK
qfpLUxR3zQOw+Nwa22Z3XBqjHBEMEGNrmVnM1PKH4HhF9AzvmUxklpnHb77D6qt+1CTGyKbwnn+52AQk3
XvaDOWWibcyfgZnjV07Zy39Nr9/8TceLHFMOqHcFowDz9/TbaELBp070HFb4YBvGE1jrqBGz6aqtC9q15
T/YPQ69/byxjqm8zuFU9X5a4AixuDEfRcTD+KXaDqFdoHfJkiJpYraHn2ionocbhpfe4t8A0WPBvw2
NWXWCSa361cFugAsshtBnRNUgxbUyOCDJreqRG3eN1HJV1grv7Aly9BQnm9yrb7Eq9754CYY/wZr5Re2p
FO1BB8zOZdhjevaHfP9q2D89bjQ1z9cn/NquGpdurfJxdrXaDoEhj8BvRtZKrtghZaDL2Z3/agLFdgt/x
6VmY/H2iu7sH1c/3HV2XWEyc3aFV/J5Vt01kou7BC+X+DQd/Vx6Eypc/Geqs0fGy5wWSq30Ddc72X1gl+
7vgccOhXMs1ZuoY/QG2oUXZydkQAbf4mNXxb7IsH1lmIa9Y2a2irksF1F+haM/zV7pRZ2Cb56zFeOU68j
fMy15N/XqxILA8P1N2I6dbXqqoHjoeyxmT3w6Pvy1sos9A/OzyJdkt50oHXuHhjy3waSc85RVyBfXX+Ee
rpSvbfG2BWxQ9ejaai/U32JwIem6Lx0RgIl/xdA5vzxgkbAe0QfQKo2sfEX6HY0AJvt1VeIBIdWq5H0RZ
PrCYhbm3b/Q1TOanDYURu43l9VqXyMKYHsq0D7qDyNR+WUbb5E4EtTlFD96JjzeVTIlDaKKsSH681X7XN
OMaWQXTl0EJhor6hCbDhBp7qLPmBKISZ1LDoc89KXRu0K7h6TWIE0vkcnmMxWrkZP1+LHnTo/+NdH65f
+A+D6WANsP+9TOEt1acsm5vfZUokW+Ijqw49Y62gmULfw18I+NzhefwWXAn4CnKIQ/8DbsPPPJLpAMvtf

ytLUF7dGtfdAb7LX/IesFfMFNE6bxZqr1yi+rwz1LT5n9Ee+NhN947Wxxhh55PzthLz7Fd719Vu3cdDM
zF392Av5nNbU12kOqWm/Bt2WoE7qMjUQFLIIPDfn3DbhmMeDSeP0BD9TE1PkAd6YOx8BYm+//nOAB8aYm
dfIaNXhL8vWytb4R+BS7Tz1Qiv8lrsFbINHdZQP1OpOmnMNPRL4KOrn7KIwcGvH3+TLTm1u9Oyu43q9V
W1s23JOz8Ts0y1opU8UYfQudoY04SvFdhtB3we/wrICM7HTQmugdiw/+zgyXRYu92Co7o1RbuVLqpxAo
A691emfileB3umv3SYtabMZacprv4ppihefHPZvZ6uMKcGnDfP0CzyT8c3fEuynish3MA9/Fvnx1jvNiY
KpTWRORTgIRqky0VIBk8P1eMehonrnfYIV11ocgmxq+VMYFYxDWrK09cnbhe+qztkHmJQmq9BZp2evhG1
AK8DPwcEmhcmKtz7z9GUYX5s9fQnhYDRSpJ+YVA1EaD3b/ItRydI57MPG5nqTYHgn6L1Y2uJ5d6nyOYxC
JqeWJ1a8l1Rp3pEm1cmoQJ9GRVtorYBJw05D83QHDPDQeOa/uyieHnCUiod85FM6ayIOpj9FOtakqJ9iH
3a8om6tdLGzDAzOZEw+TpPrXY4hOicps6U9DTxVKp+M1MVvADzHzITxs3HRdKTLsyZ12RJviRbpNjzT8n
kwTjdG/RbfU7dXtvhgblt5VY1pfuTJhUZFXqaa6xSO/TSHdWRgOL1VTvy7HOe/loehZu9Dm0CwKzoY0
sy+KRa4G+mU6e8XoEnW5Ssovq6D4KFWrdNhUsbnjI/wfVWUXn33kbkWMy8gUe+zclCy9WtnnnxtIItASn
oUIttVe2JOHIQDCqqnG1jrLgcGaprA3QtF2vC3zQhBfdbBUsnrW39bC6W1Xq/hqMb4H92xKGz0a0z8mpr
q7obhPei2E2O6SwVrIEcfSQ/3iTquoShwMr0FikP8F1AbwrT2eZFPRRbd6nUYlWWytXHLjeOrzzh5lY6B
uI2ud+Et/T2ev70mEjpgNXmJT1X9x7F0mrqEgpB+nQFflxPA8zKatOjQv2RgPwR5DgISKMPPEYGe5gU7ER
c4HwN1V6pYoDDbVcuqpkw261dH1Tt/qP2b02c002q+inUhrb6NirrsnvFShCHW1CR9zEJq25xDIQi3aMb
Ndu3Rg69hfw727x9Jwoj+Cy1VKbo0dt85UtjXbKRQ3xCz/V/r4fitu+Om9Dt+ADPB6BMwq2s1OPy6cW+m
6vXP/52FF6XHmP/5jigp/o2CtAXZyyVKmrCSDk3m7fWnjoDvjtXhZ7e2L4/VrxqvXF/kcn5gaZHU0UUj
Z+vX1GP43vgkvK4hFNgdqt3x45KEuHPmvevB3x0VfXm22vVBHjeo/WfORcNqRS+Rw0qsmtp/AnyIHkK7u
YcoLBqCyr7RUPiULjb9LpqWX16eP4zoQCo9DtOx5t1+acjgqUwC04b56eatSLpnafggzvVj0vIoJHVNz
D8SFGBsBR+VN7+CKQbu6urDqF4X7JBhk6DA1gTynpSjjHdQP7drbUrmiZaNqn3OueWP9qGPO0TSDkt+R
IC+ulxQM5b0/wIIG394Dz7dW218hJXj5bNh1Iu4oQt9D8S/KMjrOlZNnr8fhqrLe1euiHH9P9X8sG570k
erI75YpXt+/041Y0b/jf+04N3o+dP33MtON3jLsR7FF5gKNNeal1Hi0MPmjduoVMFncNeabbq73umqvfNS
8sT7Fzkrcc75Mfdjw+Fq///qG5v9FRs/e79JO0afox1s7GSRqsb10PetemRmptArvsRyfdv17rJUsWn5Z
c1t+/RGf0ct5NyA/1m+TP31Hb6H6Nw/oshQfpS3Qb1Ah0rmx9jb6nDydaFJVv+JtOu1VyZJHkYEp3mj6p
nmjEZ+9j3/1f5GaXAex7/qq8Lj1xWgIdt3j0NtbqLzV2F/x1lqBbrRXkiShv6IB+AeTK1GBzGgb7HkVFT
TSvM1omo713//eqC+wCyvp/XvL9c9C3vTdy7I+54+efyBRg9hbJEPDQcwVbWdQGwkyZFIlyOVHhTvs+RU
V1KVG4D3vy01utFa2qHC9VarD11beJm4U07zP6q1RW95tzUbV5v10QD0/DzPz1EclSNmHhC0ER5tUibZU
ni5BIxDftMyhNdrBqZY+rOKNsVS2KNnOyqPoHb3gDUJD+aIl70LC68Y/QXn1f2+cHWM6VLBWiiQJ3XGn4
7arGhTuCMR7OKhI3wlFvirvgwbgpV4VLip4vsq+7EU7194m9P7cOw+16/NrBnQwhoeWecz9Uo/WQ1PVaD
rEpEq0PRXpXnv+RQXdGr7ohVmHoJLFRhjb0f+Y1SnYsP0COyCh/fGH/lmgE5vvgZvA+G/yBI14+9w+f
epfvk4p0JvIqvsY6T5PQoeweH1PtbbQR4j1mPknUV83V7siHoxF4BfxoQFEl2QmFQ3+2VoKk0KMOFRvt
gyZVop1pgp6urbHmZxSErtP3x7Czfelvo40Q1280nyTaffFcv30mCmgA4gLORJw+Gq9HiaJdk0NP2vMzC
ugtNSb4sFLt1V9ZDTc2q2KW3KHmFcfobWkf8nkUla02HHkkrSjda83TyKDP8RHg+6yGGwWut1Q9Xz7QfI
4okd1Nh8DwOKBlesYfDvsnSM8/ADn0FWveRoVD3+UGYKrVeKPA9d0BnVMX7bpG0aEo2BesBZ4YfMaA7la
3Ss8/IPEQPd6F26vRAPiVXoYbHWP19F+CKtKHUGFcS0EnjI7SIw3/QHUXvR95ucCexxFQPFvQAHjxBf1
0/d+YTxHFrdDvfMo9P4Oep+pIvPWq8OBWjMeCaQyPAOzGGwWu92PzKaI4FRp//HfJd4h23nm79PwRKryq
HedOwP3xNgAl77vmU0RxiW/SFajbXsAJwfnU65XA4pDj7LkOT9Z8zwanol7BPAT8xmiOMQXOor0sqVgE
4R6UEn/u3rCdFWZ4mwAeL3IarhRIQ1AfMrTP8H4EvImuz30sP8G6fljldQAol5i11lph+rigCEOXSS9f8
yq7gZgzo/MZ4iiUpFOQcGLG5ufnYc6NES2eGMW5y9f4rKVQRTwAqPVCpDu8Z8iigKcaz3LBh/gXL14b
M/ZbHbNodareUQBUWMLmCki+zGGwFt3u3mU0QDEnqCUFR1GN46a0Emhg7YcZkyf0J6iN6LBqDLXhZRwL4
BS/7MXoYbFW3eY3UbAyAy8TAw+A8UWLqX+TLUmBN8T4b9CeoeOhBGGt9aT5F+rZTrTbIabzS8JM5ABiLd
838D1WCFtQCTwqG1SIN4dUpaHDcwzgtdefoxe6XlIBU2440A7031wnxx/9QfcU9bpItgeBh2WwovKXSQU
Bi/9PzJK0/fsZZJVLQE53ID8HO78UbEFP8M8zmiVoqdgYTgn7bb7hUYgZxjUiVKWnkaYS+XiGgJ/pHXAM
7rZbRR4vo3ms8R9UW8ZsLhudPv+ZeDU02qRGMig6bYyiYS9ILuYUp1zP2U1XCjwvWmm88R7Uxh1NgGIMZ
f7xpDh8NA37KWTxQ4NB0dzb6YAlQ+gLn6UqvxRoK3WT378t+bzxJtT806Tt/PUTBpu/B6BcP+U0yqRgnJ
oUZr+USFQ4+EO3Tssc14/MKpKnIgaAdic/S87Ha90f888X4MyAeCRZjPAAUcp15G+T6I+yGGx8DRhIF
NtaFp+lZ+NPPzb/XHC8SZUoTXE56N0XSz1FAY8yOe7AO3IRf1gNNzo2qPY5XzZvE70tHol16VYUSNo9v

y/GnxXxwS/6tb2cIiJc49kiKGtp3jHopVdZDDdCKg+qzs7+B7WsNbHnHIduxzZQkjHUVmM1PHdmxE5des
pmK6uo4AhN7G3oHZUxPC950+yGGxGu94aa6n3WvLG+xcZfoBEoiLRDdf1FjaZPmFSJUhd6/wJdaS2rSKF
fmRduoVLLJqvhRkmb94Bqba3v++Mcnpt7/rSNv0AzkiYjTapEWZBDB4FX7OUVffQW3vEl88Yt1FH+vApD
UNuNNxI8Uu1zv2LeWH/iCL0OPYRCSHerr0DT9D6zKEPSc/9b7OUVJbRIDbc5bu3q2h1GGmeMgLD5UXVW9
jVvrR+N1xF6/wRS3ucPOLU+OMKkSpQVtdBJMM74r3sX6Y/mjRaV/N9uY6xx0IORwC/r6nIJx8YrBOPw70
lVIIlCkyQ2fwYVRnCeBc+ziGmhZ5u3WjTNPxbGCQO1Gm50uP5q1dZ9mnlrbas12A+F+2S6PT/e7dDT+n6
5KFsKj39fn0jn4NDsHcdt4G26Nv8Zq9FGjet7anL3UebNtalxOkLvc9bCSBKHn1A6Frwoc8rTmTD+9dZy
ixz6mXnrDsTBPfYpREYbPU+rJ2fsZd5cWyrQwWASSpts/0QdY06UPYUOP1611lvk0AoM//twJ2eqf5Dun
e0GGwPlu1RXjfmV596Wo65YCyJB8jQGo5D6W3CtBvFaDG/F2sotFsJrFib6JNe/KrFRgH5PeXjN3BXgvf
UCzbQXQkLo9QYUuE01ObqqdoWn/aZayy4O2KXbKDrJvL0PaltwKBqBBVaDjYdNYLjQXFzdUWXz9FFk9ov
WQkgShwpi/BkVG79DU6z1Fhd89XeXPTmX/OYtDDR+9EjAK6hZy6vTiehdNAit+nRrASRF2PPfoQ8cibKn
e+gY1E+Cw37AF804jNwuiw/ruN5iq7HGRTjtmKqmv0YU1A95wQKKFj2sGIrgKTQR4vpZn3UWJQtce9b0
HEdko/16NB9+pRhv1TyBye2FrAV3qvg06pzyxtLGVWeTkAmL7BmfmlLoCL2/3fqGlygTCs+BDEcZxefe7
vQKnCMSUk/xD79uE2GmnMuN5m1V4Zrdy5HzepyZ4KdCLw7ZmfFNr4bxHjz5jYxVueztBhr63llgAO/XL
X5/7byi1/Ccb4htVIk8D1l+P5S9U+M1sHWdiBRoHmWjM+KcL53S/URBJvS1nSaDoOdeN+lm+b1nJLBjQF
dHzApGgA4r3Dkndr0lOBLXC9lardv1G9sGBQuiGp9a2tU8Fye8YnhEMbYPxXDryFF0UipMjbpH9GuYwFa
V/1Xo80/LtJWQSaXj4QBviiy1TCTZ40qVR5QHZWzMUU5JN1LRdr4vwLSjc2v55Ni/KlLx3CgQWAIMBKWi6
28Esahu6KvG3x5h3372Y0yDXr0WQXXvW8Nwg/0WkEQcyBSdqRQSOro5vZAz+/QVbryiZIXn6/gnr5IP8H
Pz6A80BmkfNx7Sxzt3/Egk9oIxVOBnu+G1KcCNjhNvE7h+gvx+8OystyA531qWuUkPUroWrq3PmTUoncP
feKQ4YtP3GBsi01hnL6vImNXWjM9Mwgt0vB9Mf5+isuR8643u+tY/AwP5ScEe5rQ3EeqUfQFTLV+hLy/G
b+34rkUbLCXT9rQOqTR4u0nKs1YsheMbMJWxlCnuN46sAT8Fb9P17SVW9FQTNyGx/BvRuEb79mKp5c4aA
DSjdbDsOvuIt0j9JMCjUEeTrTg4r9NN3hgCf63lOfxu0jowfppq3cJfqtK8I2FAM98xrlrn2QU9auzGlB1
5CMJOcKgluUNGUysnwjjW9DKWwu05+T1qzFti/EK24bMgid/67JjzdT20thlOLFDMoh41epMYv5Bt2MtP
S5CCY53m5ncpt3wpjOXNXsZT7UxavFmN3mzPcEHIDDRHjaSTjUWmIPbx7/pNMJqY3YkniDb+TZbMfoQM4
dBr03bwmZTC7cEcjIfv9NuNqlp4ZvEm1SI9v5BxnOB1NACnGgvMgHjrwfUbleuttXPw1uFzBE+/ukEVe+
wZLgiZAcP+YhZDuPN0oK18EYypunYHXJ/UU0s2quLm7JzmEgQrNBM9/2emxWVQemHQ/6oqeSutxpZFn16
Cnl+MX8g4Dk1To+gjxtIyLrf742gIXtBDA5vRZQHd8y/dgFbVnuGcKAX0CT8qqhHV5ubNrXwAdcAoGFv2
FgfZ+B9ftlEveqTnFzIMrUIDMKx6/T2Uy38Hg7sYDcHrvYwwLVzwxHJe8BPjF7KLQ7PQAHwu/rP9sQsf0
MYxB/1nweatjDFp3jz+GfYlWYw9CBXpttqL5sRXcUveEBjiol6GmQRuJVCPvsbzKXvGC0KasEt3hzpAhv
b341BH91EwyDvB2q0MNE608a/cjJZVhV1C9nBoIermsOQv9KQ1PjPANwrbK+OtBhs1j66Unl/IHhyyq0A
3qnxwRA3M9fspfbW48gDm56utxjsQ2vve9cggdpRgLwBBSBwdvYkdHnWUjdfEWhC3fp07j1dt3q2q5M2L
5PxAW6VHPbRarvMKGUHV588A14ODTc0XbaPdtA8/jgrk+o/AkPt31Zh7fjF+IRPQCnA3hvmn4bmPqeein
YrXCTorR8CYL4FRP6LcMpb812PmhIhFGP16MX0gJhzbqRb0C5fE8W4cGE/ftEait6/0w7jNUqXITpgjP4e
cFGClSffwbc4m9eBamFMLCaHn82uAB2N/GLCr8M9q78KiGMUT6rR5Byt3zmdUW/eFaAhuwkhhvHpoDxt
3nYtCWG0vMEHoJ+yAw6EyeB51rEUvqVn16Uz8frwaGbzP1ExRauKFRA7akKf91Wg6BD8PUMPQGufp3/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>

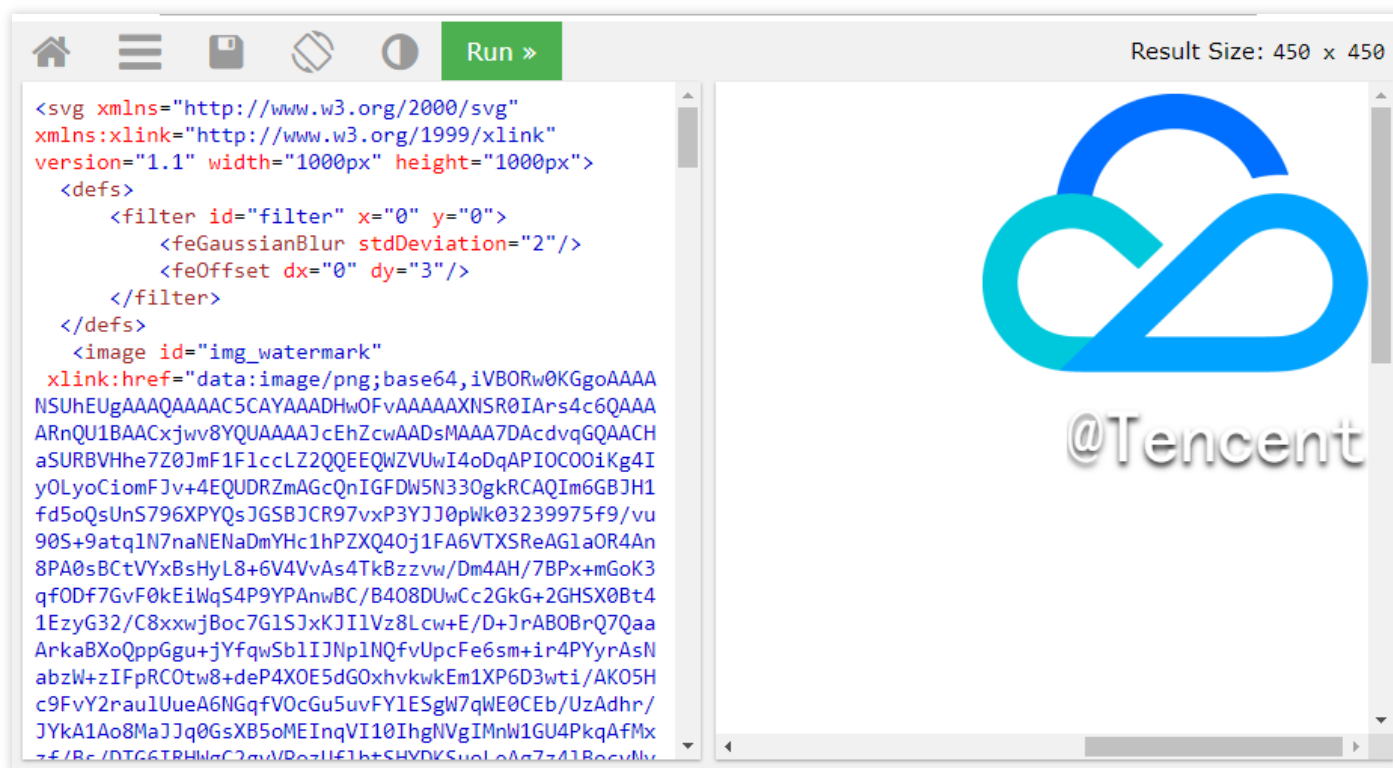
<text id="text_watermark" text-anchor="end" font-family="SimHei" font-style="Regular" font-size="50px"

x="100%" y="245px" fill="white">@Tencent</text>

</svg>

効果の検証：

- 方式1：上記のコードを[オンラインHTML WYSIWYG編集ツール](#) ページ左側のテキスト入力ボックスにコピーし、ページ上のRunボタンをクリックすると、以下のとおり、効果がページ右側に作成されます。



- 方式2：上記のコードを `.html` 拡張子のファイルに保存し、ブラウザで開くと、図2：白背景のウォーターマーク効果を確認できます。



SVGの説明

パラメータ	説明	属性
<code><svg></code>	SVGキャンバスの定義。	<code>width="1000px" height="1000px"</code> : 幅と高さはそれぞれ1000px。ここではSVG内に要素を完全に含むことができればOK。
<code><filter id="filter"></code>	使用するフィルターの定義。	-
<code><feGaussianBlur></code>	ガウシアンぼかしフィルター。	<code>stdDeviation="2"</code>
<code><feOffset></code>	フィルター要素のオフセット。	<code>dx="0" dy="3"</code> : フィルター要素を「下方向」へ3ピクセルオフセットするよう設定。
<code><image id="img_watermark"></code>	ブランドLogo画像。	<ul style="list-style-type: none"> <code>xlink:href="data:image/png;base64,{画像データのbase64}"</code> : ローカル画像を参照。 <code>height="185px" width="256px"</code> : 元の画像の幅と高さ。 <code>x="74.4%" y="0px"</code> : 必要に応じて画像は上部にあるため、y軸のオフセットは0pxに設定し、x値を連続的に**調整**して効果を確認することで、最終的に <code>x="74.4%"</code> が期待どおりであることを確認。

パラメータ	説明	属性
<pre><text id="text_watermark_shadow"></pre>	文字のシャドウ効果の実現。	<ul style="list-style-type: none"> • <code>font-family="SimHei" font-style="Regular" font-size="50px"</code> : 文字のフォントとサイズを設定。 • <code>text-anchor="end"</code> : 文字の整列方法を設定。テキストの文字列の末尾は現在のテキストの初期位置。 • <code>x="100%"</code> : x軸に沿って要素を「右端」にオフセットするよう設定。ブランドLogoの位置（タグ <code></code> の <code>x</code> 属性）を調整するため、<code>text-anchor</code>、<code>x</code> 属性の設定により、テキストの最後の文字をSVGキャンバス「右端」に「固定」。 • <code>style="opacity:1;"</code> : 透明度を不透明に設定。 • <code>fill="black"</code> : 塗りつぶし色を黒に設定。 • <code>filter="url(#filter)"</code> : <code>id</code> をフィルターの <code>filter</code> として使用。 • <code>y="245px"</code> : 要素のy軸オフセットサイズを設定。ブランドLogoの高さが185ピクセルの場合、文字サイズは50ピクセル（文字サイズは高さと同値）のため、<code>y</code> を $185+50=235$ 以上に設定し、調整を使用して245の効果適切であると判断。
<pre><text id="text_watermark"></pre>	文字。	属性を <code><text id="text_watermark_shadow"></code> と比較し、大部分の属性は同じ。違いはフィルターの未使用（ <code>filter</code> 属性がない）、かつ文字色が白（属性 <code>fill="white"</code> ）であること。

- 最終的に効果を検証する際、テキストの長さ（SVG `<text>` タグの内容）の増減により、効果が要件を満たしているか確認することをお勧めします。例： `@Tencent` を `@Tencentは善良な人です` に置き換えます。
- 使用の際、異なるユーザーIDについては、SVG `<text>` タグの内容を置き換えるだけでOKです。

手順2：SVGウォーターマークテンプレートの作成

```
https://vod.tencentcloudapi.com/?Action=CreateWatermarkTemplate
&Type=svg
&Name=テスト
&CoordinateOrigin=TopRight
&XPos=2%
&YPos=5%
&SvgTemplate.Width=30S%
&SvgTemplate.Height=0px
&<パブリックリクエストパラメータ>
```

パラメータの説明：

パラメータ	説明
Type=svg	作成するウォーターマークテンプレートのタイプはSVGです。
Name=テスト	テンプレートの名称。非入力必須パラメータ。
CoordinateOrigin=TopRight	ウォーターマークはビデオ右上隅に設定。
XPos=2%	ビデオの右枠線からのウォーターマーク始点のオフセットサイズはビデオ幅の2%です。
XPos=5%	ビデオの上部枠線からのウォーターマーク始点のオフセットサイズはビデオの高さの5%です。
SvgTemplate.Width=30S%	ウォーターマーク幅はビデオの短辺の30%です。
SvgTemplate.Height=0px	ウォーターマークの高さは元の図などの比例に応じて拡大縮小します。

作成後、SVGウォーターマークテンプレートIDは12345と仮定します。

手順3：SVGビデオウォーターマークの追加

ビデオ処理の開始：

```
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={手順1のXML}
&<パブリックリクエストパラメータ>
```

ビデオにウォーターマーク効果を追加する



付録

対応するフォントリスト

- SimHei,ゴシック体:style=Regular
- Roboto:style=Bold

その他フォントをご要望の際は、[テクニカルサポート](#)までお問い合わせください。

配信と再生

Keyホットリンク防止を使用する方法

最終更新日：：2023-10-26 17:31:32

ご使用にあたっての注意事項

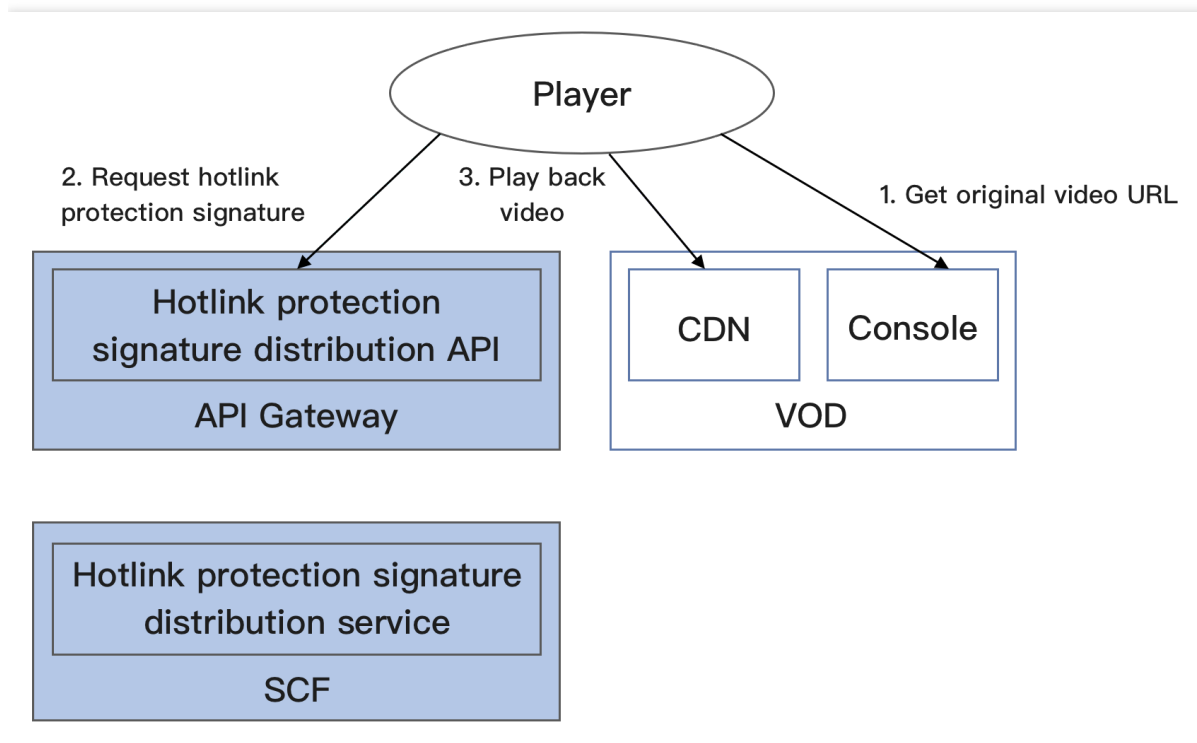
Demo機能紹介

このDemoは、VODKeyホットリンク防止システムの使用方法を提示するもので、コンソールでのKeyホットリンク防止の有効化、ホットリンク防止署名の配布サービスの構築およびホットリンク防止署名を使用してのビデオ再生を含んでいます。

アーキテクチャおよびフロー

Demoは、Serverless Cloud Function（SCF）を基にHTTPサービスを構築して、クライアントからホットリンク防止署名を取得するためのリクエストを受信します。サービスは、リクエストBodyからのVODのビデオオリジナルURLを取得し、ホットリンク防止署名を計算して、ホットリンク防止署名付きのURLをクライアントに返します。

システムは主に4つの構成部分に及びます：開発者、API Gateway、Serverless Cloud Function、VODであり、このうちAPI GatewayおよびServerless Cloud Functionは、下図に示すとおりこのDemoのデプロイオブジェクトです。



具体的な業務フローは次のとおりです。

1. VODコンソールでビデオのオリジナルURLを取得します（実際の本番環境では、プレーヤーがビジネスバックエンドからビデオのURLをリクエストする必要がありますが、ここではフローを簡素化するために、開発者がその業務行為をシミュレーションします）。
2. ビデオのオリジナルURLを使用して、SCFにホットリンク防止署名をリクエストします。
3. ホットリンク防止署名付きのビデオURLを使用して、VOD CDNをリクエストして、ビデオを再生します。

説明：

DemoのSCFコードはPython3.6を使用して開発されており、このほかSCFはPython2.7、Node.js、Golang、PHP、Javaなどの複数のプログラミング言語もサポートしています。開発者は状況に応じて自由に選択できます。詳細は [SCF開発ガイド](#) をご参照ください。

料金

ここで提供するVOD Keyホットリンク防止署名の配布サービスであるDemoは無料のオープンソースですが、構築および使用のプロセスにおいて以下の料金が発生することがあります。

- サービスのデプロイスクリプト実行に使用するため、Tencent CloudのCloud Virtual Machine（CVM）を購入します。詳細については、[CVM料金](#)をご参照ください。
- Serverless Cloud Function（SCF）を使用して署名配布サービスを提供するには、[SCF料金](#) および [SCF無料限度額](#)をご参照ください。
- Tencent Cloud API Gatewayを使用して、SCFのためにパブリックネットワークインターフェースを提供します。詳細は [API Gateway料金](#)をご参照ください。
- VODのストレージはアップロードしたビデオを保存するために消費されます。詳細については、[ストレージ料金](#)および[ストレージリソースパック](#)をご参照ください。
- VODのトラフィックはビデオを再生するために消費されます。詳細については、[トラフィック課金](#)および[トラフィックリソースパック](#)をご参照ください。

Keyホットリンク防止署名の配信サービスのクイックデプロイ

手順1：Tencent Cloud CVMの準備

デプロイスクリプトは、1台のTencent Cloud CVM上で実行させる必要があります。要件は次のとおりです。

- リージョン：任意。
- モデル：公式サイトは最低構成（1コア1GB）であればOKです。
- パブリックネットワーク：パブリックIPを有する必要があります。帯域幅は1Mbps以上。
- OS：公式パブリックイメージ `Ubuntu Server 16.04.1 LTS 64ビット` または `Ubuntu Server 18.04.1 LTS 64ビット`。

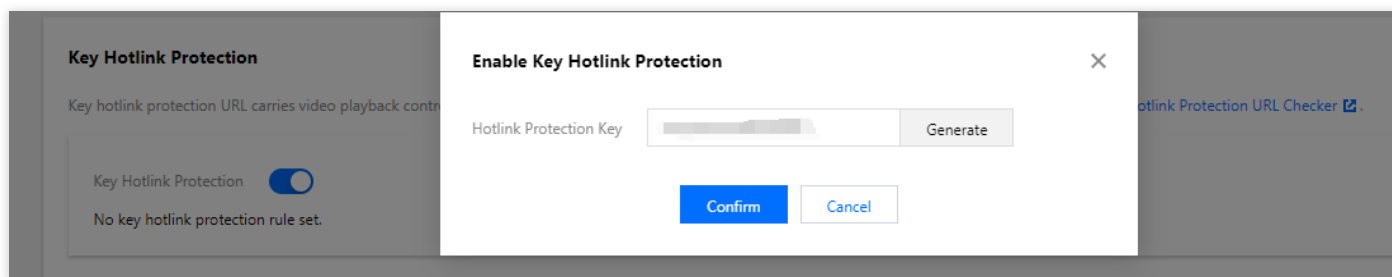
CVMの購入方法は [操作ガイド - インスタンス作成](#) をご参照ください。システムの再インストール方法は [操作ガイド - システム再インストール](#) をご参照ください。

注意：

- Keyホットリンク防止署名配布サービスのDemo自身はCVMに依存していません。CVMを使用してデプロイスクリプトを実行しているだけです。
- 上述の条件に適合するTencent Cloud CVMがない場合は、その他のパブリックネットワークアクセスを備えたLinux（CentOS、Debianなど）またはMac機器でデプロイスクリプトを実行することもできます。ただし、OSの違いによってスクリプトの特定のコマンドを修正する必要があります。具体的な修正方式については、開発者自身で検索してください。

手順2：VODのアクティブ化ならびにKeyホットリンク防止の設定

1. [クイックスタート - 手順1](#) を参照してVODサービスをアクティブにします。
2. アクティブにした後、[ホットリンク防止の設定](#) ドキュメントを参照して、Keyホットリンク防止を有効にし、ホットリンク防止Keyを記録します。



注意：

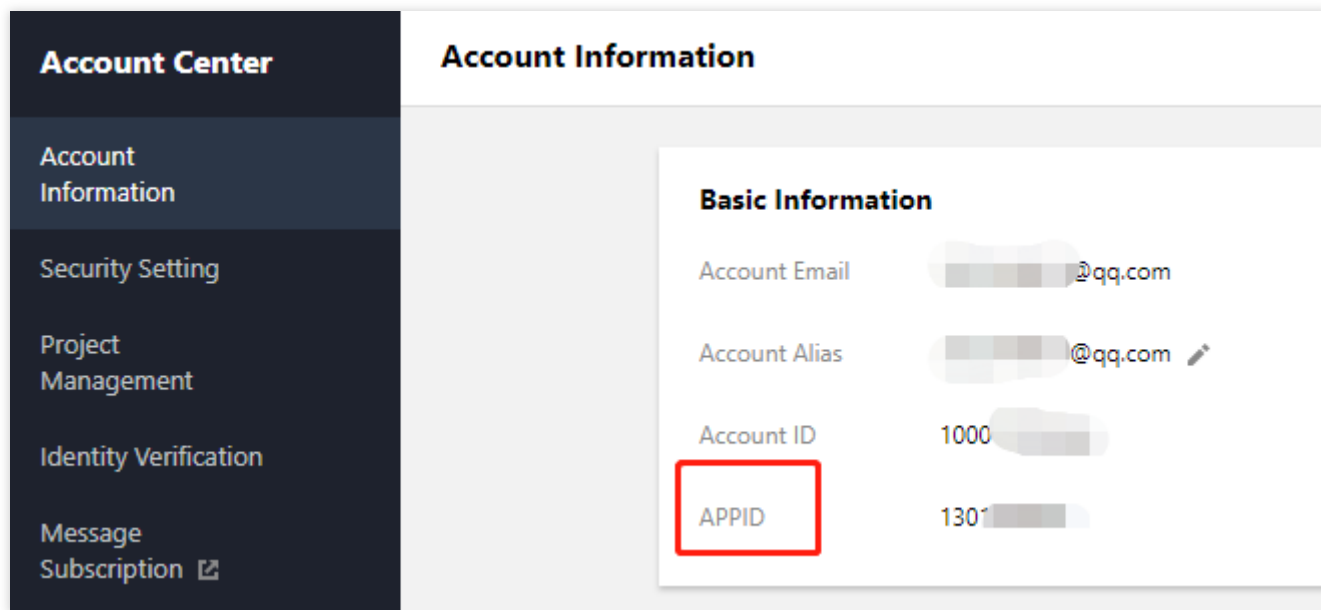
ここではKeyホットリンク防止をアクティブにして、Refererホットリンク防止はアクティブにはしません。Refererホットリンク防止を同時にアクティブにした場合、以下のテスト方法では、Refererホットリンク防止要件に適合しないことから、リクエストが失敗することがあります。

手順3：APIキーおよびAPPIDの取得

Keyホットリンク防止署名配布サービスのDemoのデプロイおよび稼働プロセスは、開発者のAPIキー（SecretIdおよびSecretKey）およびAPPIDまで使用する必要があります。

- まだキーを作成していない場合は、[キー作成ドキュメント](#) を参照して、新しいAPIキーを作成してください。キーを作成済みの場合は、[キー表示ドキュメント](#) を参照してAPIキーを取得してください。

- コンソールの [アカウント情報](#) 画面で下部に示すとおり、APPIDを表示することができます。



手順4：ホットリンク防止署名配布サービスのデプロイ

手順1で準備したCVM（ログイン方法の詳細は [操作ガイド - Linuxにログイン](#) をご参照ください）にログインして、リモートターミナルで以下のコマンドを入力して実行します。

```
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
```

説明：

手順3で取得した内容をコマンドのSECRET_ID、SECRET_KEY、APPIDに割り当ててください。手順2で取得したホットリンク防止KeyをANTI_LEECH_KEYに割り当てます。

このコマンドでは、GithubからDemoソースコードをダウンロードして、インストールスクリプトを自動的に実行します。インストールのプロセスには数分間必要になり（CVMのネットワーク状況次第）、その間、リモートターミナルは以下に示す情報を出力します。

```
[2020-06-04 15:57:10] npmの検査を開始します。
[2020-06-04 15:57:18] npmインストール成功。
[2020-06-04 15:57:18] ServerLessツールインストール開始。
[2020-06-04 15:57:19] ServerLessインストール成功。
```

```
[2020-06-04 15:57:20]VOD Keyホットリンク防止署名配布サービスのデプロイを開始。
[2020-06-04 15:57:30]VOD Keyホットリンク防止署名配布サービスのデプロイを完了。
[2020-06-04 15:57:32]サービスアドレス：https://service-xxxxxxxx-125xxxxxxxx.gz.apigw.tencentcs.com/release/anti_leech_sign
```

出力ログの署名配布サービスアドレス（例：`https://service-xxxxxxxx-125xxxxxxxx.gz.apigw.tencentcs.com/release/callback`）をコピーします。

注意：

出力ログで以下に示す警告が現れた場合は、通常CVMはデプロイしたばかりのサービスドメイン名を直ちに解析できないことから、その警告は無視してもかまいません。

```
> [2020-04-25 17:18:44]警告：Keyホットリンク防止署名配布サービスがテストに失敗しました。
>
```

手順5：Keyホットリンク防止のテスト

ビデオのアップロード - ローカルからのアップロード手順の説明に従って、1個のテストビデオをVODにアップロードします。アップロードの完了後、【クイックビュー】をクリックしてから、右側の【アドレスコピー】をクリックしてそのビデオのURLをコピーします。

The screenshot shows the 'Video Management' interface. On the left, there is a table of uploaded videos. The first row is highlighted, showing a video named 'test111.mp4' with a 'Quick View' button. On the right, the details for 'test111.mp4' are displayed, including its ID, name, category, label, size (589.43KB), duration (00:00:04), and expiration time (Permanent). At the bottom of the details panel, there is a 'Video URL' section with a 'Copy All URLs' button and a table of video specifications. The 'Original Video' row shows the format as MP4 and resolution as 720 x 960, with a 'Copy URL' button highlighted in red.

Video Name/ID	Video Status	Video Category	Upload
test111.mp4 ID: [redacted] Quick View	Normal	Other	2021-10-08
111_2021-08-12-10-04-58_2021-08-12-10-08-46 ID:3701925922273478162 Quick View	Normal	Other	2021-08-12
111_2021-08-12-10-08-22_2021-08-12-10-08-46 ID:3701925922273478109 Quick View	Normal	Other	2021-08-12
111_2021-08-12-10-03-22_2021-08-12-10-08-22 ID:3701925922273478109 Quick View	Normal	Other	2021-08-12
111_2021-08-12-09-59-58_2021-08-12-10-04-58			

Specification	Format	Resolution	Operation
Original Video	MP4	720 x 960	Copy URL Download

CVMのコマンドラインで `curl` コマンドを実行してこのURLへの直接アクセスをテストします。結果はKeyホットリンク防止規則に適合しなかったことから、サーバーにアクセスを拒絶されます。HTTPリターンコードは403になります（テスト時は、コマンドのURLを実際のURLに置換してください。以下同じ）。

```
ubuntu@VM-69-2-ubuntu:~$ curl -I "http://125xxxxxxx.vod2.myqcloud.com/f888c998vodc
cq125xxxxxxx/c849148f528xxxxxxxxxxxxxxxxxxxx/xxxxxxxxxx.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
```

CVMコマンドラインで `curl` コマンドを実行して、手順4でデプロイされたサービスからのホットリンク防止署名付きのURLをリクエストします（`-d` は、POSTメソッドを使用してリクエストを開始し、付帯するパラメータはビデオURLです）。

```
ubuntu@VM-69-2-ubuntu:~$ curl -d 'http://125xxxxxxx.vod2.myqcloud.com/f888c998vodc
cq125xxxxxxx/c849148f528xxxxxxxxxxxxxxxxxxxx/xxxxxxxxxx.mp4' https://service-xxxxxxx
-125xxxxxxx.gz.apigw.tencentcs.com/release/anti_leech_sign; echo
http://125xxxxxxx.vod2.myqcloud.com/f888c998vodcq125xxxxxxx/c849148f528xxxxxxxxxxx
xxxxxx/xxxxxxxxxx.mp4?t=5ed8b8d2&exper=0&rlimit=0&us=455041&sign=fe6394007c2e7aef
39fc70a02e897f69
```

`curl` コマンドを再度使用して、前の手順で得たホットリンク防止署名付きのURLにアクセスすれば、正常にアクセスできます（HTTPリターンコードは200）。

```
ubuntu@VM-69-2-ubuntu:~$ curl -I "http://125xxxxxxx.vod2.myqcloud.com/f888c998vodc
cq125xxxxxxx/c849148f528xxxxxxxxxxxxxxxxxxxx/xxxxxxxxxx.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: NWVkoGiyYmVfZDUyMzYyNjRfYWwMF85YjkyNzA=
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: *
```

説明：

ブラウザでホットリンク防止署名付きのURLにアクセスして、ビデオを再生してホットリンク防止署名を検証できます。ただし、この方法はビデオ形式に要件があり、一般にH.264でエンコードされたMP4ファイルを使用すれば良好な互換性がありますので、このビデオを選択使用することをお勧めします。Postmanなどのサードパーティツールを使用してHTTPリクエストを送信することもできます。使用方法の詳細はご自身でご検索ください。

システム設計の説明

インターフェースプロトコル

Keyホットリンク防止署名配布クラウド関数は、API Gatewayによって対外的なインターフェースを提供します。具体的なインターフェースプロトコルは以下のとおりです。

サービス	クラウド関数名	インターフェース形式	リクエスト内容	返信内容
Keyホットリンク防止署名配布	anti_leech_sign	HTTP POST	ビデオオリジナルURL	ホットリンク防止署名付きURL

署名配布サービスコードの解説

- `main_handler()` はエントリー関数です。
- `parse_conf_file()` を呼び出して、`config.json` ファイルから設定情報を読み取ります。設定項目の説明は以下のとおりです（詳細パラメータは [Keyホットリンク防止署名パラメータ](#) をご参照ください）。

フィールド	データ型	機能
key	String	Keyホットリンク防止キー
t	Integer	署名有効時間、単位：秒。リクエストが処理されているときは、このパラメータとSCFサーバーの現在の時間を足し合わせると、ホットリンク防止パラメータのtになります
exper	Integer	プレビュー時間
rlimit	Integer	署名にアクセスできるクライアントIPの最大数

3. リクエストのBodyから `Dir` パラメータを解析して、ローカルで `t` および `us` のパラメータを生成し、設定ファイルからは `exper` および `rlimit` のパラメータを読み取ります。

```
original_url = event["body"]
parse_result = urlparse(original_url)
directory = path.split(parse_result.path)[0] + '/'
# 署名パラメータ
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
}
```

4. `generate_sign()` を呼び出してホットリンク防止署名を計算します。詳細なアルゴリズムは [ホットリンク防止署名](#)をご参照ください。

5. `QueryString`を生成してオリジナルURLの最後に追加して、URLをホットリンク防止署名と連結します。

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

6. 署名を返します。返されるデータの形式と説明については [クラウド関数の統合レスポンス](#)をご参照ください。

```
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
}
```

悪意あるビデオホスティング問題と防止

最終更新日：：2023-10-26 17:31:32

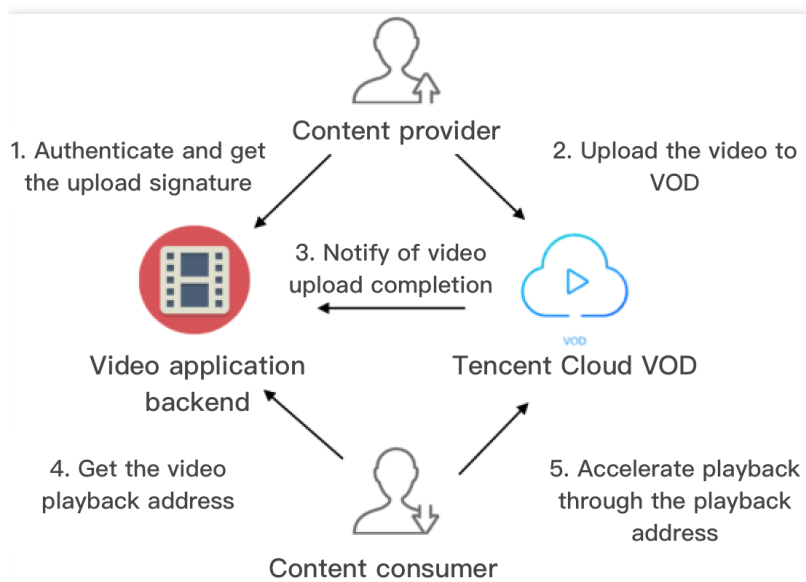
UGC（ユーザーにより制作されたコンテンツ）およびPGC（プロにより制作されたコンテンツ）は、ビデオ業界ではよく見られる2つのシナリオで、ビデオ共有者がビデオコンテンツを自由にアップロードおよび共有することができます。

しかし、サードパーティのビデオプラットフォームでは、Appの普通のユーザーになりすまして、自身のビデオをアップロードして、ビデオの再生URLを自身のプラットフォームに配置することがあります。こうすると、彼らはプラットフォームに「寄生」して、ビデオの保存と再生のアクセラレーションを「無償」で得ることができます。その結果、ビデオプラットフォームは他人のビデオの温床と見なされるため、「悪意のあるビデオホスティング」と呼ばれています。

パラサイトが生成するストレージ、再生の帯域幅、トラフィックのすべての費用は、App開発者が負担する必要があります。これは甚大な経済的損失になります。

悪意あるビデオホスティング問題が発生する原因

UGCおよびPGCプラットフォームの一般的な相互作用



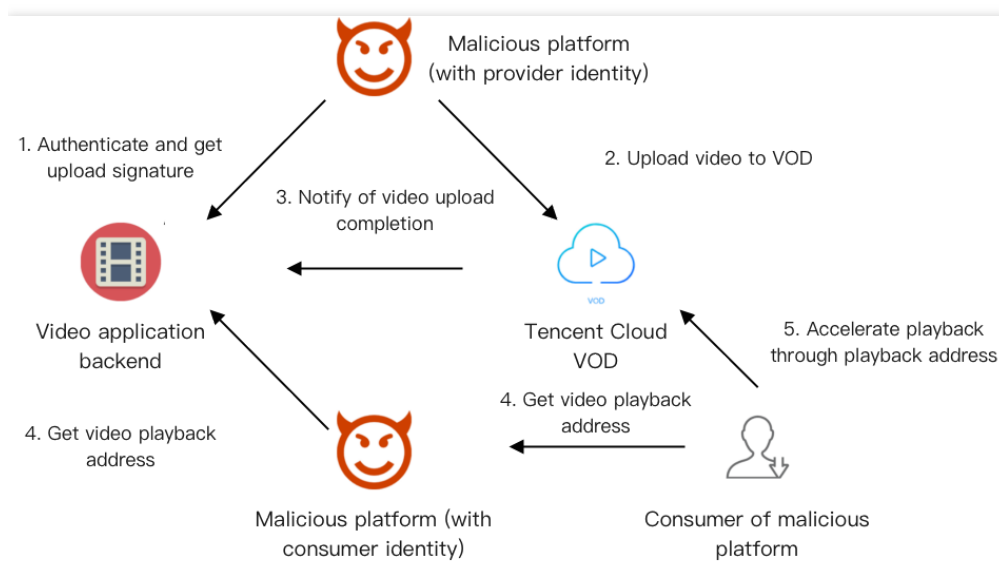
説明：

図中の矢印の向きはネットワークのリクエスト方向を示します。

UGC（またはPGC）のビデオプラットフォームでは、コンテンツプロバイダ、コンテンツ消費者、VODと以下の方法で対話します（手順1～3は [クライアントからのアップロード](#)をご参照ください）。

1. Appバックエンドはコンテンツプロバイダを認証します。認証が通過した後、ビデオアップロード署名を配布します。
2. コンテンツプロバイダは共有するコンテンツをVODにアップロードします。
3. VODは、アップロードされたビデオのfileIdおよび再生URLなどの関連情報をAppバックエンドに通知します。
4. コンテンツ消費者は、Appバックエンドからビデオ再生URLをリクエストします。
5. コンテンツ消費者は、再生URLを介してVODからビデオをアクセラレーション再生します。

悪意のあるユーザーがビデオホスティングを実現する手口



説明：

図中の矢印の向きはネットワークのリクエスト方向を示します。

悪意のあるサードパーティのビデオプラットフォームは、Appプラットフォームの普通のユーザーになります。

- 初めに、ビデオプロバイダとして自身のビデオをVODにアップロードします（ステップ1およびステップ2）。
- 次に、今度は消費者としてAppプラットフォームからビデオの再生アドレスを取得します（ステップ4）。
- 最後に、悪意のあるプラットフォームのユーザーが、これらの再生アドレスを取得して（ステップ4）、VODを通じてこれらのビデオをアクセラレーション再生します（ステップ5）。

問題を引き起こす主要原因

悪意のあるユーザーが獲物を探してビデオホスティングする根本的な目的は、他人のCDN帯域幅リソース（ストレージリソースを使用しながら）を盗用することです。悪意のあるユーザーが付け入る可能性のある主要原因は以下のとおりです。

- ステップ4：悪意のあるプラットフォームは、無制限にAppから素早くビデオの再生URLを取得して、保存し自身の消費ユーザーに配信します。
- ステップ5：悪意のあるプラットフォームの消費ユーザーは、ビデオの再生URLを取得した後、無制限にビデオをアクセラレーション再生できます。

悪意のあるビデオホスティング防止プラン

悪意あるビデオホスティング問題の主な原因に直面してのキーポイントは以下のとおりです。

- ステップ4の**ビデオ再生URLを無制限に取得**の防止。
- ステップ5の**ビデオを無制限にアクセラレーション再生**の防止。

以下、ビデオURLの再生および取得を制限する方法をそれぞれご紹介します。

ビデオURLの再生制限

VODの**Keyホットリンク防止**は、URLから再生できる端末の数を制限する機能を提供し、1個のビデオURLが任意の数のクライアントの端末に配布され再生されるのを防止します。

ビデオ再生URLへの制御を実現するために、コンソールでホットリンク防止を有効にする必要があります。そして、ステップ4では、AppバックエンドでKeyホットリンク防止生成規則に従う必要があります（「ビデオ再生アドレスでの最大再生可能IP数」の例をご参照ください）、ホットリンク防止を生成して、URLの有効時間および再生が許容されるIP数を制限します。

ビデオURLの取得制限

ビデオのアクセラレーション再生を制限するだけでは、ビデオホスティングを防止するには不完全です。ステップ4では、悪意のあるプラットフォームは同一のビデオに対して大量の異なるホットリンク防止URLをリクエストできるため、その後、自身のプラットフォームのユーザーに異なるURLを配布して、再生できるIP数の制限を回避します。

このため、Appのバックエンドはステップ4でユーザーのIDを識別する必要があります。同一ユーザーが指定時間内に同じビデオ再生URLを取得する回数を頻繁に制御して、悪意のあるユーザーが短時間にビデオ再生アドレスを大量に取得できないようにします。

イベント通知の受信方法

最終更新日：：2023-10-26 17:31:32

ご使用にあたっての注意事項

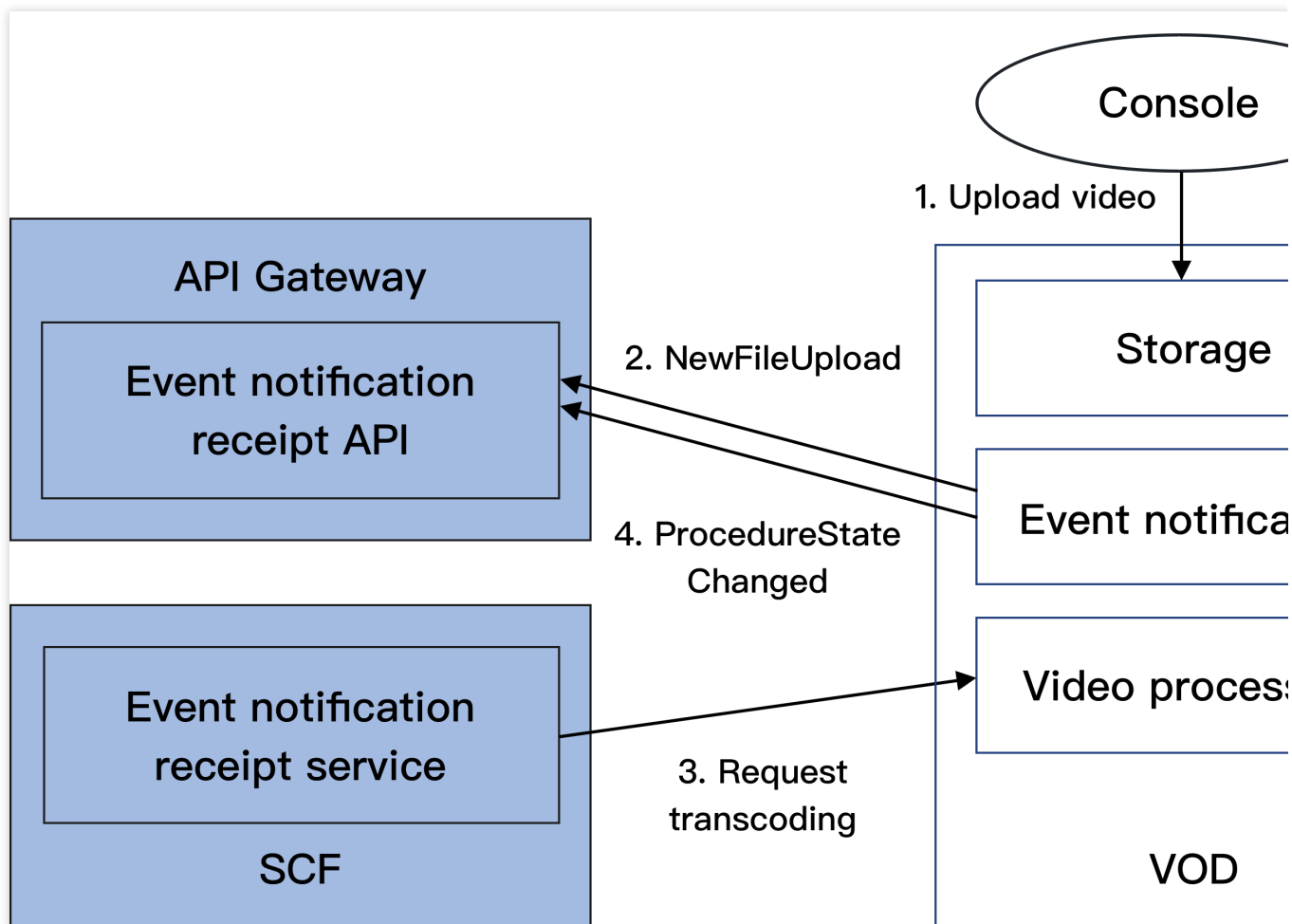
Demo機能紹介

ここではビデオのアップロード、トランスコードのフローを例に、開発者にVODの[イベント通知システム](#)の使用方法について表示します。

アーキテクチャおよびフロー

Demoは、クラウド関数（SCF）を基にHTTPサービスを構築して、VODからのイベント通知リクエストを受信するのに使用します。このサービスは、[NewFileUpload](#)（[ビデオアップロード完了イベント通知](#)）および[ProcedureStateChanged](#)（[タスクフローステータスの変更](#)）を処理することによって、ビデオトランスコードの開始およびトランスコード結果の取得を実現します。

システムは主に4つの構成部分に及びます。コンソール、API Gateway、Serverless Cloud Function、VODであり、このうちAPI GatewayおよびServerless Cloud Functionは、下図に示すとおりこのDemoのデプロイオブジェクトです。



具体的な業務フローは次のとおりです。

1. コンソールでビデオをVODにアップロードします。
2. VODバックエンドはNewFileUploadイベント通知リクエストを起動してDemoに送ります。
3. Demoはイベント通知コンテンツを解析し、VODの [ProcessMedia](#) インターフェースを呼び出して、アップロードされたばかりのビデオのトランスコードを開始します。使用するトランスコードテンプレートは [システムプリセットテンプレート](#) の100010および100020です。
4. VODはトランスコードタスクを完了後、DemoへのProcedureStateChangedイベント通知リクエストを開始します。
5. Demoはイベント通知コンテンツを解析して、トランスコードされた出力ファイルのURLをSCFログに出力します。

説明：

DemoのSCFコードはPython3.6を使用して開発されており、このほかSCFはPython2.7、Node.js、Golang、PHP、Javaなどの複数のプログラミング言語もサポートしています。開発者は状況に応じて自由に選択できます。詳細は [SCF開発ガイド](#) をご参照ください。

料金

ここで提供するVODイベント通知の受信サービスであるDemoは無料のオープンソースですが、構築および使用のプロセスにおいて以下の料金が発生することがあります。

サービスのデプロイスクリプト実行に使用するため、Tencent CloudのCloud Virtual Machine (CVM) を購入します。詳細については、[CVM料金](#)をご参照ください。

SCFを使用して署名配布サービスを提供するには、[SCF料金](#)および[SCF無料利用枠](#)をご参照ください。

Tencent Cloud API Gatewayを使用して、SCFのためにパブリックネットワークインターフェースを提供します。詳細は[API Gateway料金](#)をご参照ください。

VODストレージは、アップロードしたビデオに使用されます。詳しくは、[ストレージ料金](#)および[ストレージリソースパック](#)をご参照ください。

VODトランスコード時間は、ビデオのトランスコードに使用されます。詳しくは、[トランスコード料金](#) および [リソースパックのトランスコード](#)をご参照ください。

本番環境への影響の回避

イベント通知受信サービスDemoの業務ロジックはVODイベント通知システムで使用します。このため、デプロイプロセスでは、開発者がイベント通知アドレスを設定する必要があります。そのアカウントがVODに基づく本番環境をすでに有していると、イベント通知アドレスの変更は業務上の不具合を生じることがあります。**操作前には本番環境に影響しないことを必ず確認し、確定できない場合は、すべて新しいアカウントに交換して Demo をデプロイしてください。**

イベント通知受信サービスのクイックデプロイ

ステップ1：Tencent Cloud CVMの準備

デプロイスクリプトは、1台のTencent Cloud CVM上で実行させる必要があります。要件は次のとおりです。

リージョン：任意。

モデル：公式サイトは最低構成（1コア1GB）であれば問題ありません。

パブリックネットワーク：パブリックIPを有する必要があります。帯域幅は1Mbps以上。

OS：公式パブリックイメージ `Ubuntu Server 16.04.1 LTS 64ビット` または `Ubuntu Server 18.04.1 LTS 64ビット`。

CVMの購入方法は[操作ガイド - インスタンス作成](#)をご参照ください。システムの再インストール方法は[操作ガイド - システム再インストール](#)をご参照ください。

注意：

イベント通知受信サービスDemo自身はCVMに依存していません。CVMを使用してデプロイスクリプトを実行しているだけです。

上述の条件に適合するTencent Cloud CVMがない場合は、その他のパブリックネットワークアクセスを備えたLinux（CentOS、Debianなど）またはMac機器でデプロイスクリプトを実行することもできます。ただし、OSの違いによってスクリプトの特定のコマンドを修正する必要があります。具体的な修正方式については、開発者自身で検索してください。

ステップ2：VODをアクティブ化する

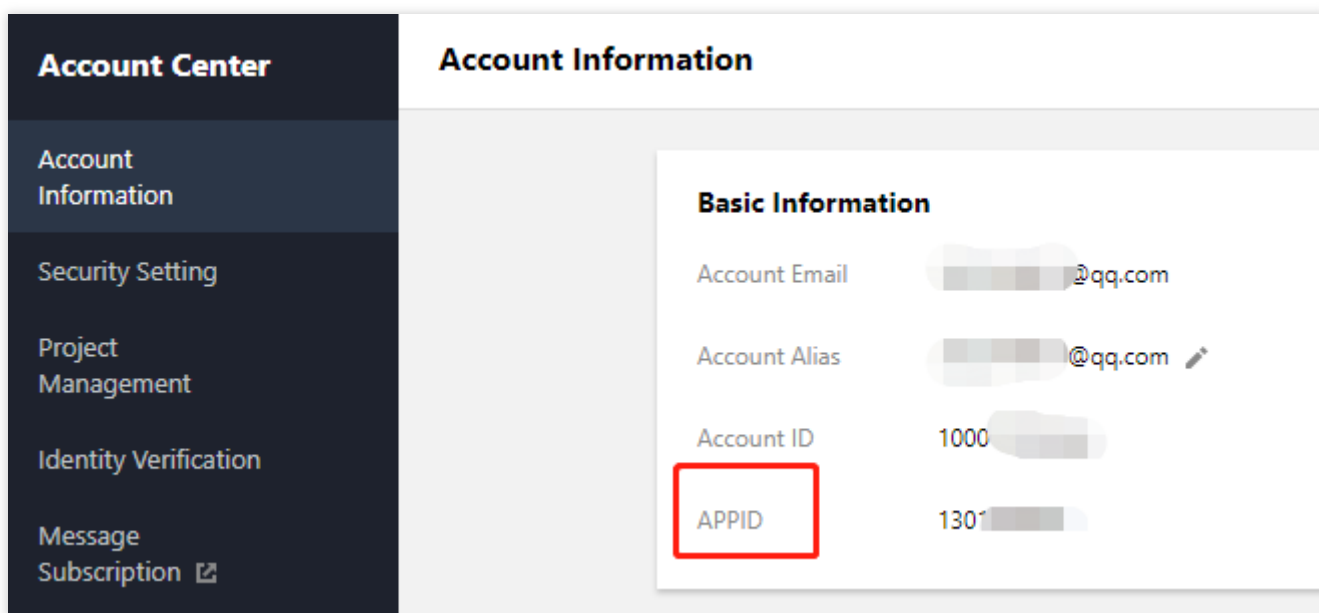
[クイックスタート - ステップ1](#)を参考に、VODサービスをアクティブにします。

ステップ3：APIキーおよびAPPIDの取得

イベント通知受信サービスDemoのデプロイおよび実行では、開発者のAPIキー（すなわちSecretIdおよびSecretKey）およびAPPIDを使用する必要があります。

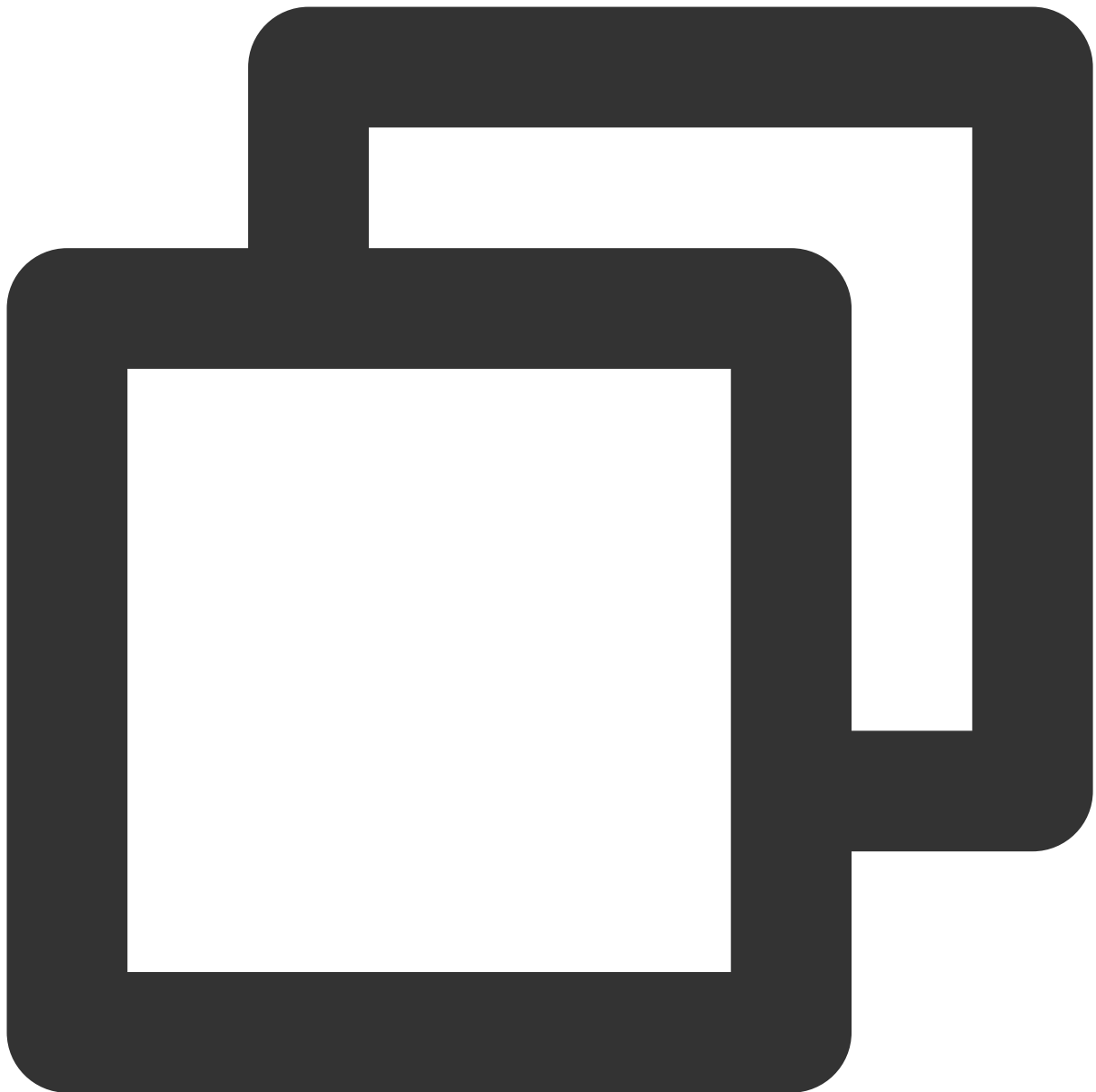
まだキーを作成していない場合は、[キー作成ドキュメント](#)を参照して、新しいAPIキーを作成してください。キーを作成済みの場合は、[キー表示ドキュメント](#)を参照してAPIキーを取得してください。

コンソールの [アカウント情報](#)画面で下部に示すとおり、APPIDを表示することができます。



ステップ4：イベント通知受信サービスのデプロイ

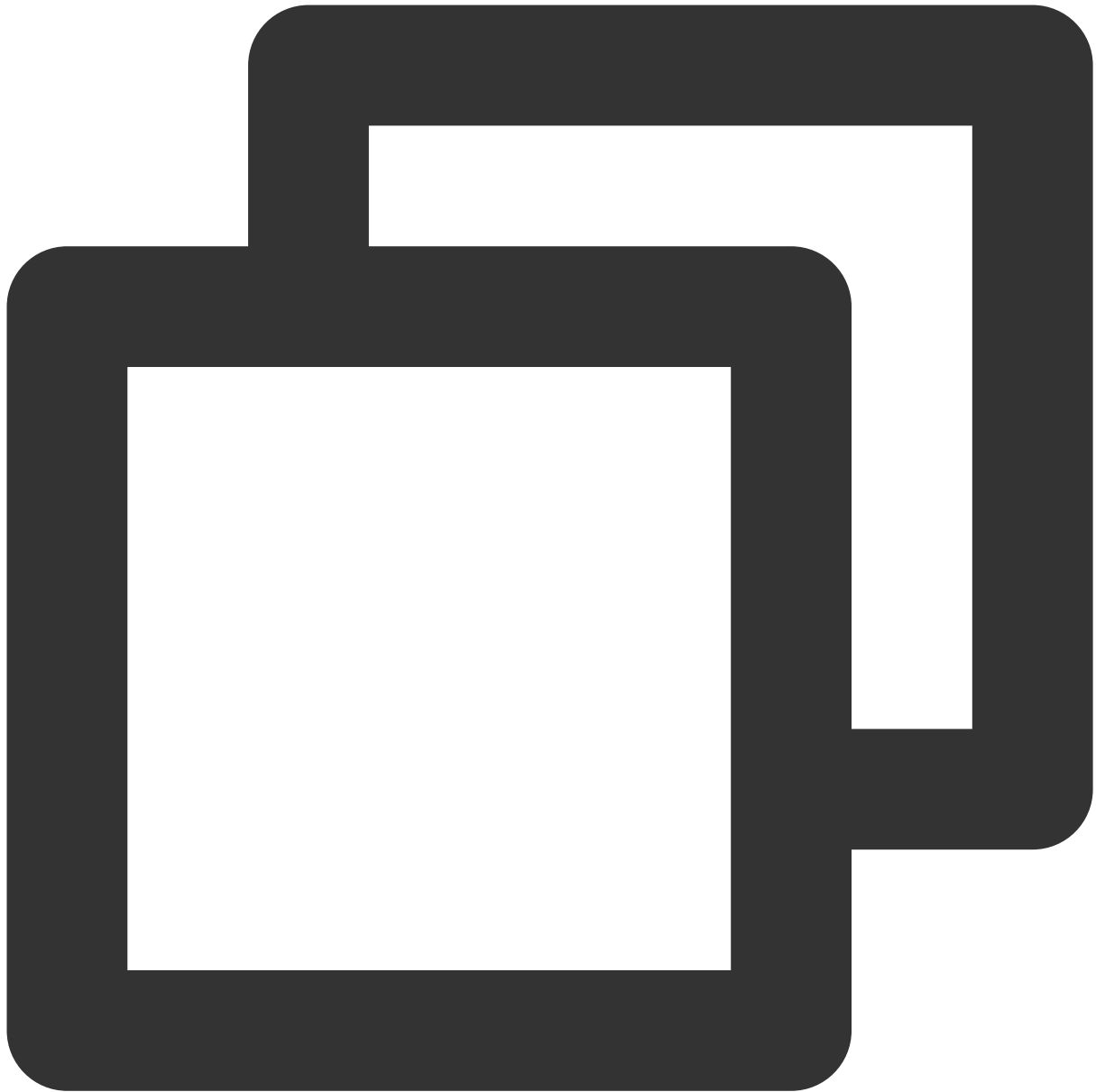
[ステップ1](#)で準備したCVM（ログイン方法の詳細は [操作ガイド - Linuxにログイン](#)をご参照ください）にログインして、リモートターミナルで以下のコマンドを入力して実行します。



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

説明：

コマンドのSECRET_ID、SECRET_KEY、APPIDを [ステップ3](#)で取得したコンテンツに割り当ててください。
このコマンドでは、GithubからDemoソースコードをダウンロードして、インストールスクリプトを自動的に実行します。インストールのプロセスには数分間必要になり（CVMのネットワーク状況次第）、その間、リモートターミナルは以下に示す情報を出力します。

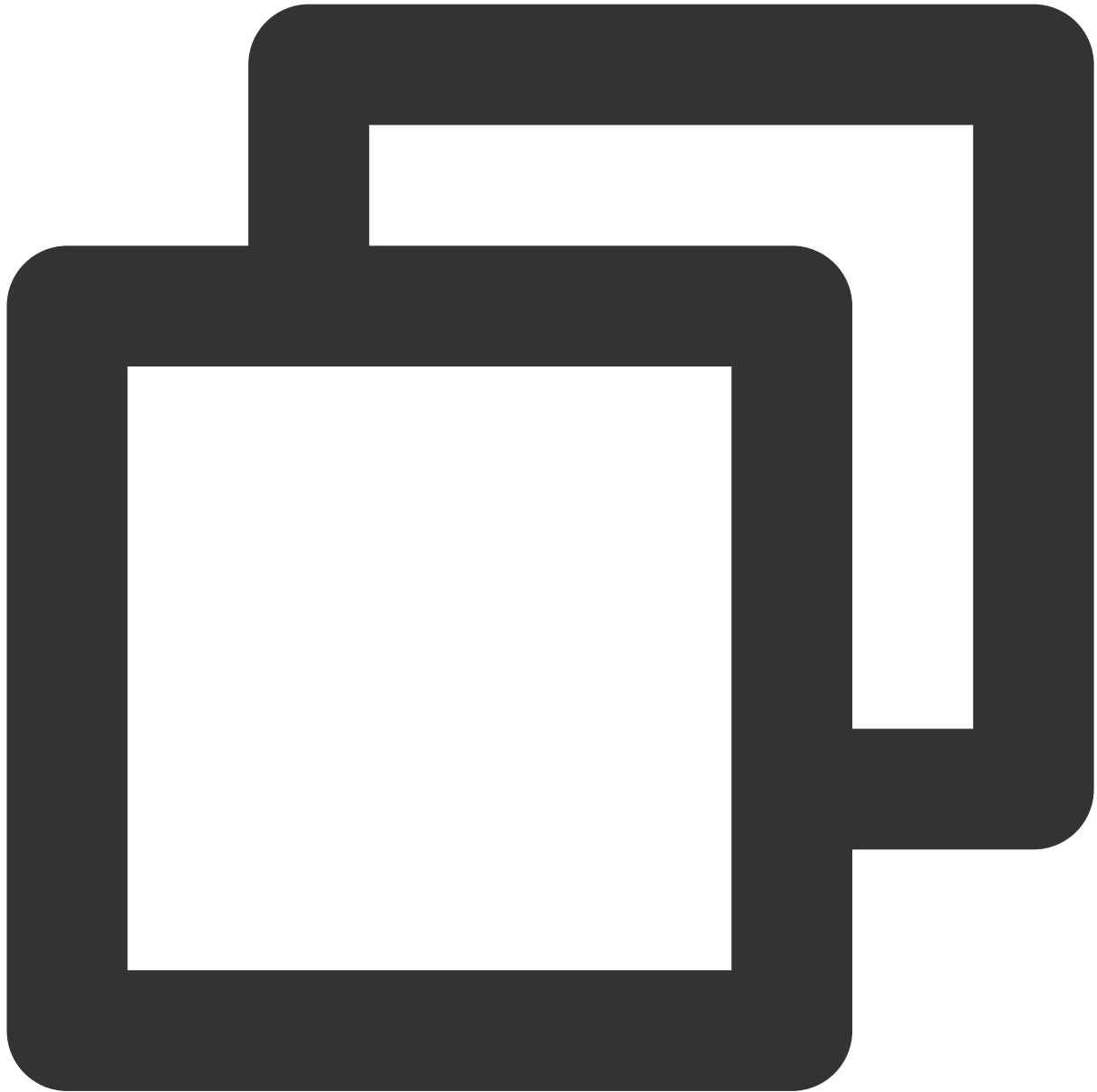


```
[2020-06-05 17:16:08] 検査npmを開始。  
[2020-06-05 17:16:12] npmインストール成功。  
[2020-06-05 17:16:12] ServerLessインストール開始。  
[2020-06-05 17:16:13] serverlessインストール成功。  
[2020-06-05 17:16:14] VODのイベント通知受信サービスのデプロイ開始。  
[2020-06-05 17:16:24] VODのイベント通知受信サービスのデプロイ完了。  
[2020-06-05 17:16:26] サービスアドレス：https://service-xxxxxxxx-125xxxxxxxx.gz.apigw.te
```

出力ログのイベント通知受信サービスアドレス（例の <https://service-xxxxxxxx-125xxxxxxxx.gz.apigw.tencentcs.com/release/callback> ）をコピーします。

注意：

出力ログで以下に示す警告が現れた場合は、通常CVMはデプロイしたばかりのサービスドメイン名を直ちに解析できないことから、その警告は無視してもかまいません。



[2020-04-25 17:18:44] 警告：イベント通知受信サービステストをクリアしていません。

ステップ5：イベント通知アドレスの設定**注意：**

本番環境への影響の回避 の節で記述したとおり、操作前にまずオンライン業務がVODイベント通知に依存していないことを確認してください。

1. VODコンソールにログインし、左側ナビゲーションバーの**アプリケーション管理**をクリックし、アプリケーションリストページに進みます。
2. オーディオビデオを処理したいアプリケーションを見つけ、アプリケーション名をクリックし、アプリケーション管理ページに進みます。
3. 左側ナビゲーションバーの**コールバック設定**をクリックし、次に**設定**をクリックします。コールバックモードは「**ノーマルコールバック**」を選択し、コールバックURLは、**ステップ4**で獲得したイベント通知受信サービスのアドレスを入力します。コールバックイベントのすべてにチェックを入れ、クリックして確定します。下図に示すとおりです。

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 by time points in the video
- Finished screencapturing of image sprite in the video

Confirm **Cancel**

注意：

コンソールに2個のコールバックURL設定（2.0バージョン形式および3.0バージョン形式）が同時にあった場合は、3.0バージョンを設定してください。

ステップ6：Demoテスト

1. [ビデオのアップロード - ローカルからのアップロード手順](#)の説明に従って、1個のテストビデオをVODにアップロードして、アップロード中にデフォルトの「アップロードのみ、ビデオは実施しない」を選択します。アップロードの完了後は、「アップロード済み」タブの画面でそのビデオの状態が「処理中」になっていることが確認できます。このことは、DemoがNewFileUploadイベント通知を受信して、トランスコードリクエストを起動させたことを示しています。
2. ビデオ処理が完了するのを待ってから（ステータスが「正常」に変わります）、[クイックビュー](#)をクリックすれば、画面右側に、そのビデオの2つのトランスコードされたビデオがあることが確認できます。
3. [SCFコンソールログ画面](#)にログインして、SCFのログを表示します。最新のログで、2個のトランスコードされたファイルのURLが出力されていることが確認できます。実際のユースケースでは、開発者はSCFを使用してURLを自身のデータベースに記録するか、その他のチャンネルを通じて視聴者に公開することができます。

説明：

SCFログにはいくらかのディレイが生じることがあります。画面上にログが見当たらない場合は、1～2分間待ってからリセットをクリックして更新してください。

システム設計の説明

インターフェースプロトコル

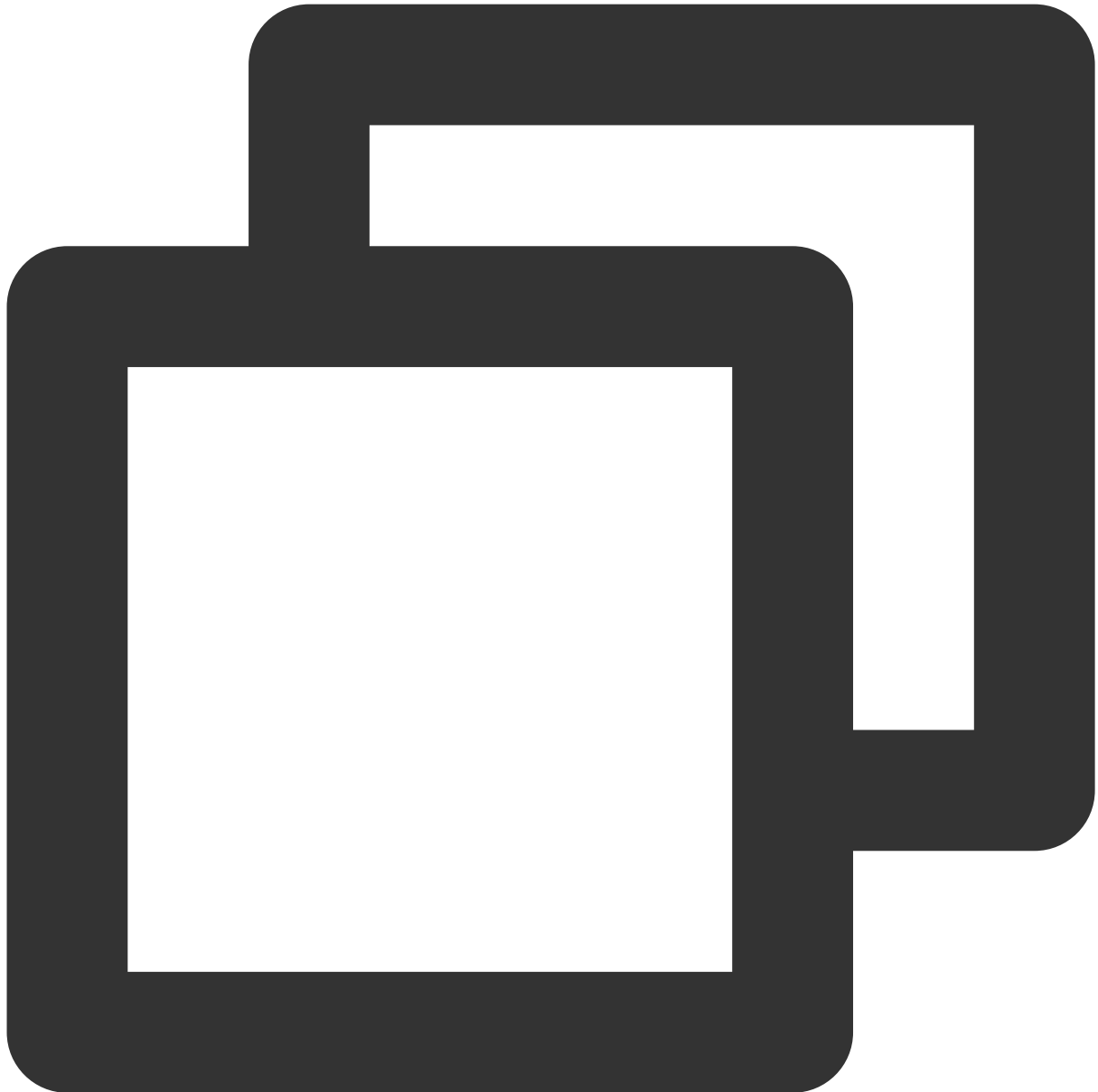
イベント通知受信関数は、API Gatewayによって外部にインターフェースを提供します。具体的なインターフェースプロトコルについては、[ビデオアップロード完了イベント通知](#) および [タスクフローステータスの変更](#)をご参照ください。

イベント通知受信サービスコードの解釈

1. `main_handler()` はエントリー関数です。
2. `parse_conf_file()` を呼び出して、`config.json` ファイルから設定情報を読み取ります。設定項目の説明は以下のとおりです。

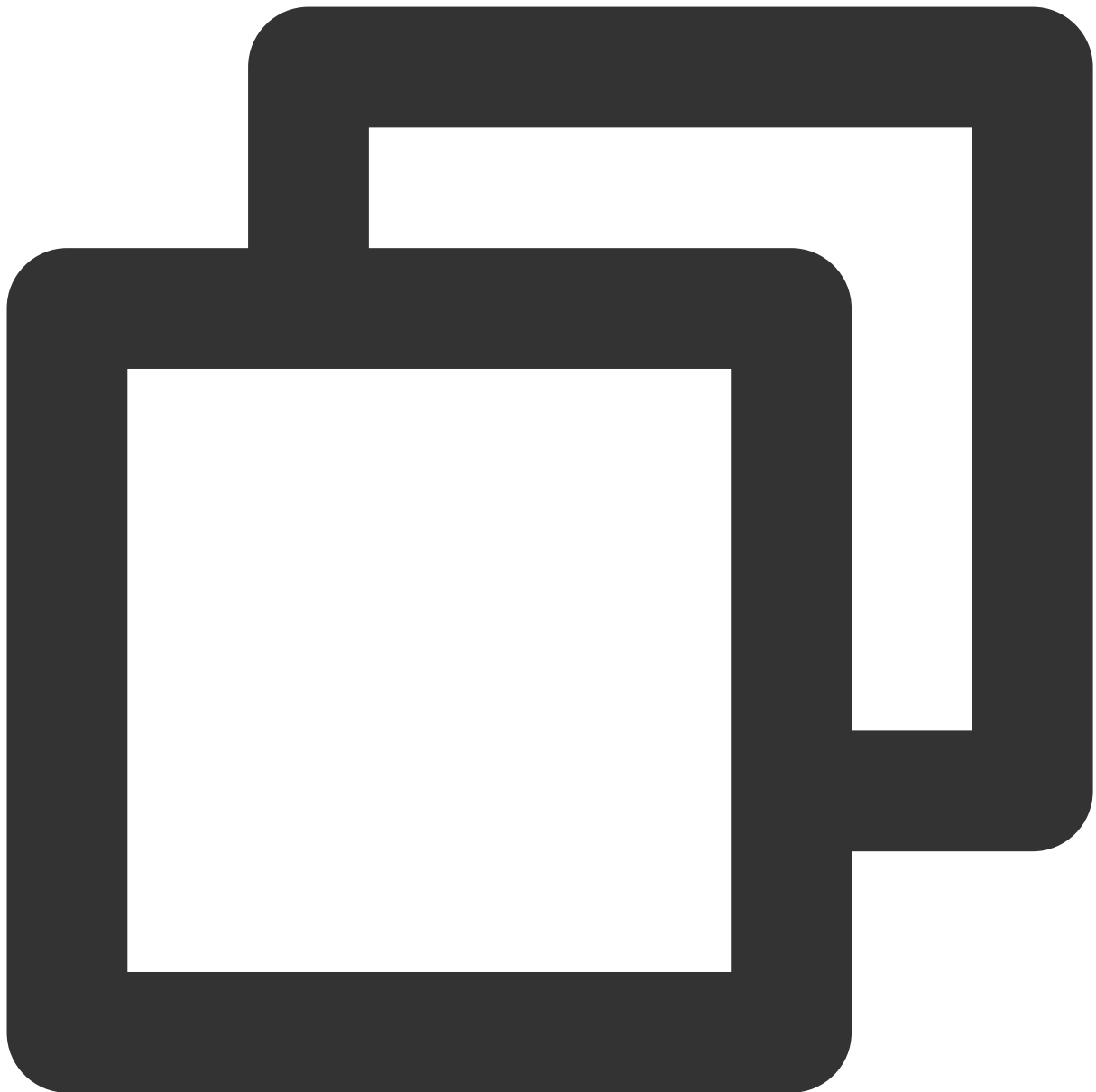
フィールド	データタイプ	機能
<code>secret_id</code>	String	APIキー
<code>secret_key</code>	String	APIキー
<code>region</code>	String	TencentCloud APIリクエストリージョン。VODについて自由に入力できます
<code>definitions</code>	Array of Integer	トランスコードテンプレート
<code>subappid</code>	Integer	イベントの通知が VODサブアプリケーション から来ているかどうか

3. NewFileUploadタイプのイベント通知の場合、 `deal_new_file_event()` を呼び出してリクエストを解析して、その中から新しくアップロードされたビデオのFileIdを取得します。



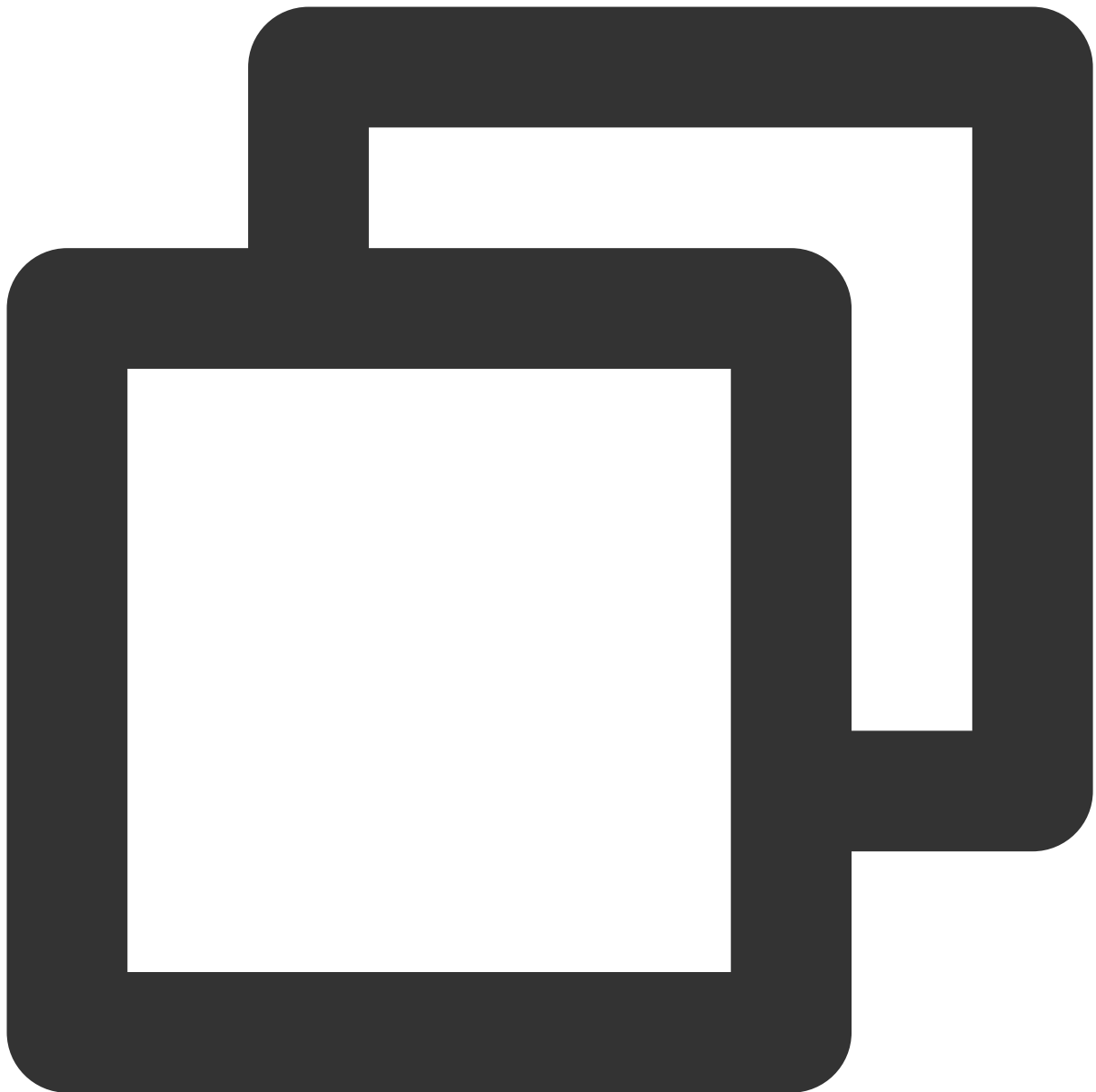
```
if event_type == "NewFileUpload":
    fileid = deal_new_file_event(body)
    if fileid is None:
        return ERR_RETURN
```

4. `trans_media()` を呼び出してトランスコードを開始し、Tencent Cloud APIの応答パッケージをSCFログに出力して、VODのイベント通知サービスに返します。



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

5. `trans_media()` でTencent Cloud APIのSDKを呼び出して `ProcessMedia` リクエストを開始します。

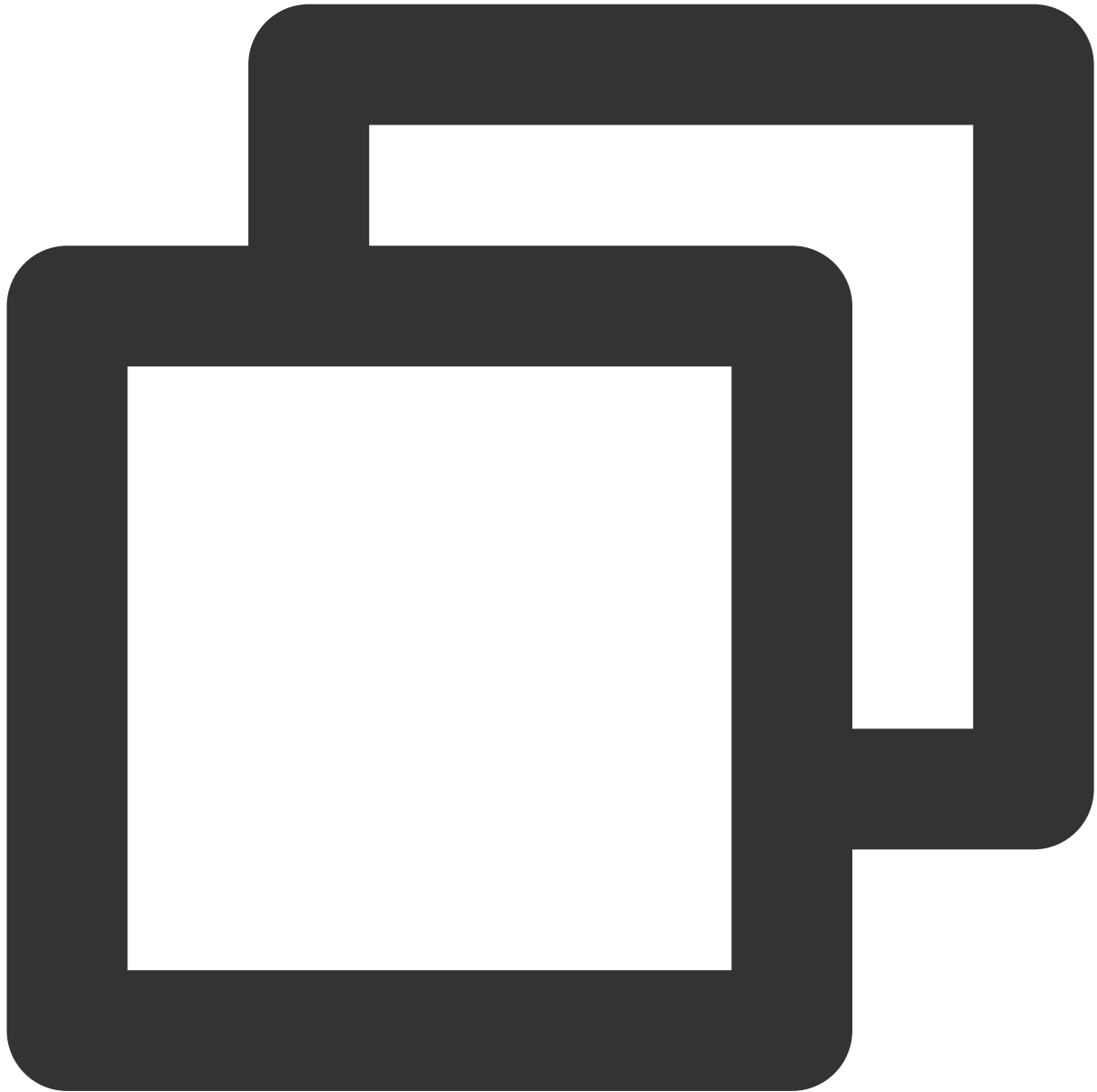


```
cred = credential.Credential(conf["secret_id"], conf["secret_key"])
client = vod_client.VodClient(cred, conf["region"])

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. ProcedureStateChangedタイプのイベント通知の場合、 `deal_procedure_event()` を呼び出してリクエストを解釈して、トランスコード出力ビデオURLを取得しSCFログに出力します。



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

オリジンサーバーのマイグレーションツール

最終更新日：：2023-10-26 17:31:32

概要

VOD Migrate Toolは、データマイグレーション機能を集約化した一体型ツールです。シンプルな設定ファイルを編集することで、ユーザーはソースアドレスのメディアファイルをVODに迅速にマイグレーションできます。

サポートするデータソース

- ローカルフォルダ
- URLリスト
- Tencent Cloud COS
- AWS S3
- Alibaba Cloud OSS
- Qiniu Kodo

使用環境

システム環境

Windows、Linux、macOSシステムをサポートします。

ソフトウェア依存

- Python 2.7/3.4+。
- 最新バージョンのpip。

インストール

Pipによるインストール（推奨）

SDKをpipによってプロジェクトにインストールすることができます。プロジェクト環境にまだpipをインストールしていない場合は、pip公式サイトを参照してインストールしてください。

```
pip install vodmigrate
```

ソースコードパッケージによるインストール

ソースコードダウンロードアドレス：[ここをクリック](#)。

最新コードをダウンロードして解凍後：

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

ユースケース

コマンドの実行：

```
vodmigrate config.toml
```

説明：

マイグレーションが完了すると、結果は設定項目"migrateResultOutputPath"に対応するディレクトリに出力されます。ファイル名：vod_migrate_result.txt。

設定ファイルの説明

設定ファイルは、toml形式（参考：[config_template.toml](#)を採用しています。ファイルがUTF-8でエンコードされていること確認してください）。ファイルの内容は以下のいくつかの部分に分けられます。

1. マイグレーションの種類の設定

typeはマイグレーションのニーズに応じて入力するマイグレーションのタイプを表します。例えば、ローカルデータをVODにマイグレーションする場合は、`[migrateType]` の設定内容は `type=migrateLocal` になります。

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

現在サポートするマイグレーションの種類は以下のとおりです。

migrateType	説明
migrateLocal	ローカルからVODにマイグレーション
migrateUrl	ダウンロードURLからVODにマイグレーション
migrateCos	Tencent Cloud COSからVODにマイグレーション
migrateAws	AWS S3からVODにマイグレーション
migrateAli	Alibaba Cloud OSSからVODにマイグレーション
migrateQiniu	Qiniu KodoからVODにマイグレーション

2. マイグレーションタスクの設定

ユーザーは実際のマイグレーションのニーズに従って関連設定を行います。主にVOD設定とタスク設定の情報関連のマイグレーションになります。

#マイグレーションツールの標準設定

[common]

```
secretId = "SECRETID"
secretKey = "SECRETKEY"
region = 'REGION'
subAppId = 0
concurrency = 5
supportMediaClassification = [ 'video', 'audio', 'image' ]
excludeMediaType = [ ]
migrateDbStoragePath = ''
migrateResultOutputPath = ''
```

名称	説明
secretId	SecretIdはユーザーキーです。 <code>SECRETID</code> を実際のキー情報に置換してください。 CAMコンソール のTencent Cloud API キー画面に進んで取得することができます。
secretKey	SecretKeyはユーザーキーです。 <code>SECRETKEY</code> を実際のキー情報に置換してください。 CAMコンソール のTencent Cloud API キー画面に進んで取得することができます。
region	アクセスポイントリージョンとは、VODサーバーをリクエストするリージョンのことで、ストレージリージョンとは異なります。詳細はサポートする リージョンリスト をご参照ください。

名称	説明
subAppld	VODのサブアプリケーション IDです。ファイルをサブアプリケーションにマイグレーションする場合は、このフィールドにサブアプリケーションIDを入力します。マイグレーションの必要がない場合、入力する必要はありません。
concurrency	同時にマイグレーションされるファイルの数量。最大値50
supportMediaClassification	マイグレーションでサポートされるメディアタイプのリスト。有効値：video（ビデオ）、audio（オーディオ）、image（画像）
excludeMediaType	排除するファイルタイプのリスト
migrateDbStoragePath	マイグレーションされたdbの保存パス。空欄の場合は、現在のディレクトリを意味します。
migrateResultOutputPath	マイグレーション結果の保存パス（1個のマイグレーションレコードは、1行のjson形式の文字列に対応）。空欄の場合は、現在のディレクトリを意味します。

ファイルタイプの説明：

- ビデオ：MP4、TS、FLV、WMV、ASF、RM、RMVB、MPG、MPEG、3GP、MOV、WEBM、MKV、AVI、** HLS、DASHはサポートしない**
- オーディオ：MP3、M4A、FLAC、OGG、WAV
- 画像：JPG、JPEG、PNG、GIF、BMP、TIFF、AI、CDR、EPS

3. データソース情報の設定

[migrateType] のマイグレーションタイプに従って、対応するセクションを設定します。例えば、[migrateType] の設定内容がtype=migrateLocalであれば、ユーザーは [migrateLocal] セクションを設定するだけです。

3.1 ローカルのデータソースmigrateLocalの設定

ローカルからVODにマイグレーションする場合は、この部分の設定を行います。具体的な設定項目および説明は以下のとおりです。

```
# ローカルからVODへのマイグレーションの設定セクション
[migrateLocal]
localPath = ''
excludes = [ ]
```

設定項目	説明
localPath	ローカルパスは、絶対パスの形式である必要があります

設定項目	説明
excludes	排除するディレクトリの絶対パス。localPathのディレクトリにあるファイルはマイグレーションされないことを示します。

3.2 URLリストのデータソースmigrateUrlの設定

指定したURLリストからVODにマイグレーションする場合は、この部分の設定を行います。具体的な設定項目および説明は以下のとおりです。

```
# URLリストのダウンロードからVODにマイグレーションするための設定セクション
```

```
[migrateUrl]
```

```
urllistPath = 'D:\folder\urllist.txt'
```

設定項目	説明
urllistPath	URLリストを保存しているファイルの絶対パス。ファイルの内容は1行に1個の元のURLアドレスを含むURLテキストです。

3.3 COSデータソースmigrateCosの設定

Tencent CloudのCOSからVODにマイグレーションする場合は、この部分の設定を行います。具体的な設定項目および説明は以下のとおりです。

```
# Tencent CloudのCOSからVODへのマイグレーションの設定セクション
```

```
[migrateCos]
```

```
region = 'ap-shanghai'
```

```
bucket = 'examplebucket-1250000000'
```

```
secretId = 'COS_SECRETID'
```

```
secretKey = 'COS_SECRETKEY'
```

```
prefix = ''
```

設定項目	説明
region	BucketのRegion情報については アベイラビリティリージョン をご参照ください
bucket	<bucketname-appid> 形式のBucket名。Bucket名はAPPIDを必ず含める必要があります。 例：examplebucket-1250000000
secretId	Bucketが属するユーザーキーのsecretId。 Tencent Cloud APIキー で表示することができます。
secretKey	Bucketが属するユーザーキーのsecretKey。 Tencent Cloud APIキー で表示することができます。

設定項目	説明
prefix	マイグレーションするパスのプレフィックス。Bucketの全データをマイグレーションする場合は、prefixを空欄にします。

3.4 AWSデータソースmigrateAwsの設定

AWSからVODにマイグレーションする場合は、この部分の設定を行います。具体的な設定項目および説明は以下のとおりです。

```
# AWSからVODへのマイグレーションの設定セクション
[migrateAws]
region = 'ap-northeast-2'
bucket = 'bucket-aws'
accessKeyId = 'AccessKeyId'
accessKeySecret = 'AccessKeySecret'
prefix = ''
```

設定項目	説明
region	AWS COS Region
bucket	AWS COS Bucket名
accessKeyId	AccessKeyIdをユーザーキーに置換
accessKeySecret	AccessKeySecretをユーザーキーに置換
prefix	マイグレーションするパスのプレフィックス。Bucketの全データをマイグレーションする場合は、prefixを空欄にします。

3.5 Alibaba OSSデータソースmigrateAliの設定

Alibaba Cloud OSSからVODにマイグレーションする場合は、この部分の設定を行います。具体的な設定項目及び説明は以下のとおりです。

```
# Alibaba OSSからVODへのマイグレーションの設定セクション
[migrateAli]
bucket = 'bucket-aliyun'
accessKeyId = 'yourAccessKeyId'
accessKeySecret = 'yourAccessKeySecret'
endPoint = 'oss-cn-hangzhou.aliyuncs.com'
prefix = ''
```

設定項目	説明
------	----

設定項目	説明
bucket	Alibaba Cloud OSS Bucket名
accessKeyId	yourAccessKeyIdをユーザーキーに置換
accessKeySecret	yourAccessKeySecretをユーザーキーに置換
endPoint	Alibaba Cloud endpointアドレス
prefix	マイグレーションするパスのプレフィックス。Bucketの全データをマイグレーションする場合は、prefixを空欄にします

3.6 QiniuデータソースmigrateQiniuの設定

QiniuからVODにマイグレーションする場合は、この部分の設定を行います。具体的な設定項目および説明は以下のとおりです。

```
# QiniuからVODへのマイグレーションの設定セクション
[migrateQiniu]
bucket = 'bucket-qiniu'
accessKeyId = 'AccessKey'
accessKeySecret = 'SecretKey'
endPoint = 'www.bkt.clouddn.com'
prefix = ''
```

設定項目	説明
bucket	Qiniu Kodo Bucket名
accessKeyId	AccessKeyをユーザーキーに置換
accessKeySecret	SecretKeyをユーザーキーに置換
endPoint	QiniuダウンロードアドレスはdownloadDomainに対応
prefix	マイグレーションするパスのプレフィックス。Bucketの全データをマイグレーションする場合は、prefixを空欄にします

制限事項

- このツールは1回限りのマイグレーションツールとして設計されています。マイグレーションは**オリジンサーバーのファイルスキャン**、**マイグレーション中**、**マイグレーションの完了**の3段階に分かれます。ファイルス

キャンの完了後は、設定変更が必要な場合は、md5ファイルのチェックでエラーが発生しないように、dbファイルをクリアする必要があります（migrate.dbの削除またはdbストレージパスの修正）。

- マイグレーションするファイルは接尾辞を付けて表示する必要があります。
- HLS/DASHのマイグレーションは現在サポートしていません。
- マイグレーション後は、元のビデオ間のディレクトリ関係は維持できず、各ビデオには独立したFileIdがあり、相互には関連していません。

マイグレーションフローの概要

1. 設定ファイルが読み取られ、セクションがマイグレーションtypeに従って読み取られ、パラメータがチェックされます。
2. オリジンサーバーはマイグレーションタイプに従ってスキャンされ、マイグレーションタスクが生成されます。
3. スキャンが完了すると、マイグレーションが実行され、各タスクの結果および全体の進捗が出力されます。
4. マイグレーションが完了すると、詳細情報が結果ファイルに出力されます。

```
[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 - ██████████ (2).mp4
[2020-06-23 11:03:16,616] add migrate task: E:\cloud\vod_migrate\media\Wildlife - ██████████.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\██████████.jpg
[2020-06-23 11:03:16,627] add migrate task: E:\cloud\vod_migrate\media\██████████.png
[2020-06-23 11:03:17,772] ██████████ : 16.67%, ██████████ : 6, ██████████ : 1, ██████████ : 0
[2020-06-23 11:03:17,852] ██████████ : 33.33%, ██████████ : 6, ██████████ : 2, ██████████ : 0
[2020-06-23 11:03:17,957] ██████████ : 50.00%, ██████████ : 6, ██████████ : 3, ██████████ : 0
[2020-06-23 11:03:18,068] ██████████ : 66.67%, ██████████ : 6, ██████████ : 4, ██████████ : 0
[2020-06-23 11:03:18,218] ██████████ : 83.33%, ██████████ : 6, ██████████ : 5, ██████████ : 0
[2020-06-23 11:03:18,292] ██████████ : 100.00%, ██████████ : 6, ██████████ : 6, ██████████ : 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
```

Live Recording

ライブストリーミングビデオをVODにレコーディングして処理する方法

最終更新日：：2023-10-26 17:31:32

CSSレコーディングのソリューションは、ライブストリーミングのオリジナルストリームをオーディオビデオパッケージングに変換（音声、ビデオデータおよび対応するタイムスタンプなどの情報は変更しない）することで得られるファイルをTencent Cloud VODプラットフォームに保存し、さらにレコーディングファイルに対する二次制作、配信、再生を行うスタンダードソリューションです。詳細については、[CSSレコーディングソリューション](#)をご参照ください。

製品の特徴

Tencent Cloud CSSの機能をベースとして、CSSストリームのコンテンツを素早くレコーディングしてクラウドプラットフォームに保存し、さらに二次制作および配信を行うことができます。

Tencent Cloudの最先端のオーディオビデオAI技術と全世界の膨大なライブストリーミングアクセラレーションノードをベースとして、専門的で安定性の高いCSSプッシュ、トランスコーディング、配信および再生サービスを提供します。超低遅延、超高画質、大規模並列処理（同時アクセス）のニーズに全面的に対応します。

CSSレコーディングサービスをベースとして、お客様のライブストリーミングイベントを各種のユースケースやAppの中でスピーディーに伝播させることができます。

企業のライブストリーミング、eコマースライブストリーミング、eラーニングのライブストリーミングなど多様な業界のシナリオに適用できます。またWeChat Mini Program、Tencent Videoなど、多様な配信方式にも適用できます。

前提条件

Tencent Cloudアカウントを[登録](#)して、[ログイン](#)します。

Tencent Cloud CSSおよびVODサービスがアクティブになっていること。アクティブになっていない場合は、[CSSサービス](#) および[VODサービス](#)にてアクティブ化してください。

実践手順

手順1：レコーディングテンプレートの作成

レコーディング機能を使用するには、まずレコーディングテンプレートを作成する必要があります。CSSレコーディング機能の設定はいずれもレコーディングテンプレートの中に保存します。さまざまな設定のレコーディングテンプレートを作成することで、さまざまな形式、さまざまな長さでのレコーディングファイルなどを実現できます。

コンソールから作成する方法：

1.1 [CSSコンソール](#)に入り、[機能設定](#)>[CSSレコーディング](#)を選択します。

1.2 [テンプレートの作成](#)をクリックし、必要なレコーディングファイルのタイプを選択します（少なくとも1つの形式を選択してください）。その他の設定項目の説明については、[テンプレートの作成](#)をご参照ください。

The screenshot displays the 'Create Recording Template' interface. At the top, there are two buttons: 'Create Recording Template' (highlighted with a red box) and 'Bind Domain Name'. Below these is a list of existing templates: '443', 'test0825', 'test', 'recordchenhui', and 'All_Formats'. The main area is titled 'Recording Configuration' and contains the following fields:

- Template Name ***: A text input field with the placeholder 'Enter a template name'. Below it, a note states 'Only support letters, digits, underscores, and di'.
- Template Description**: A text area with the placeholder 'Please describe template'. Below it, a note states 'Only support letters, digits, underscores, and di'.
- Recording Format ***: Radio buttons for 'HLS', 'FLV', 'MP4' (selected), and 'AAC'. An information icon is next to AAC.
- Audio/Video - MP4**: A dropdown menu showing 'Audio/Video - MP4'.
- Max Recording Time Per File**: A text input field with the value '1-120'.
- Storage Period**: Radio buttons for 'Permanent' (selected) and another option.
- VOD Subapplication/Category**: A text input field with the value 'Primary applicator'.
- Advanced Configuration**: A dropdown menu.

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

1.3 [保存](#)をクリックすれば、テンプレートの作成が完了します。

APIによる作成：

[CreateLiveRecordTemplate](#) インターフェースを呼び出してレコーディングテンプレートを作成します。テンプレートの作成に成功すると対応するテンプレートIDが返ってきます。

ステップ2：レコーディング方法の選択

CSSでは様々なシナリオにもとづき、以下の種類のCSSレコーディング機能を呼び出す方法を提供しています。

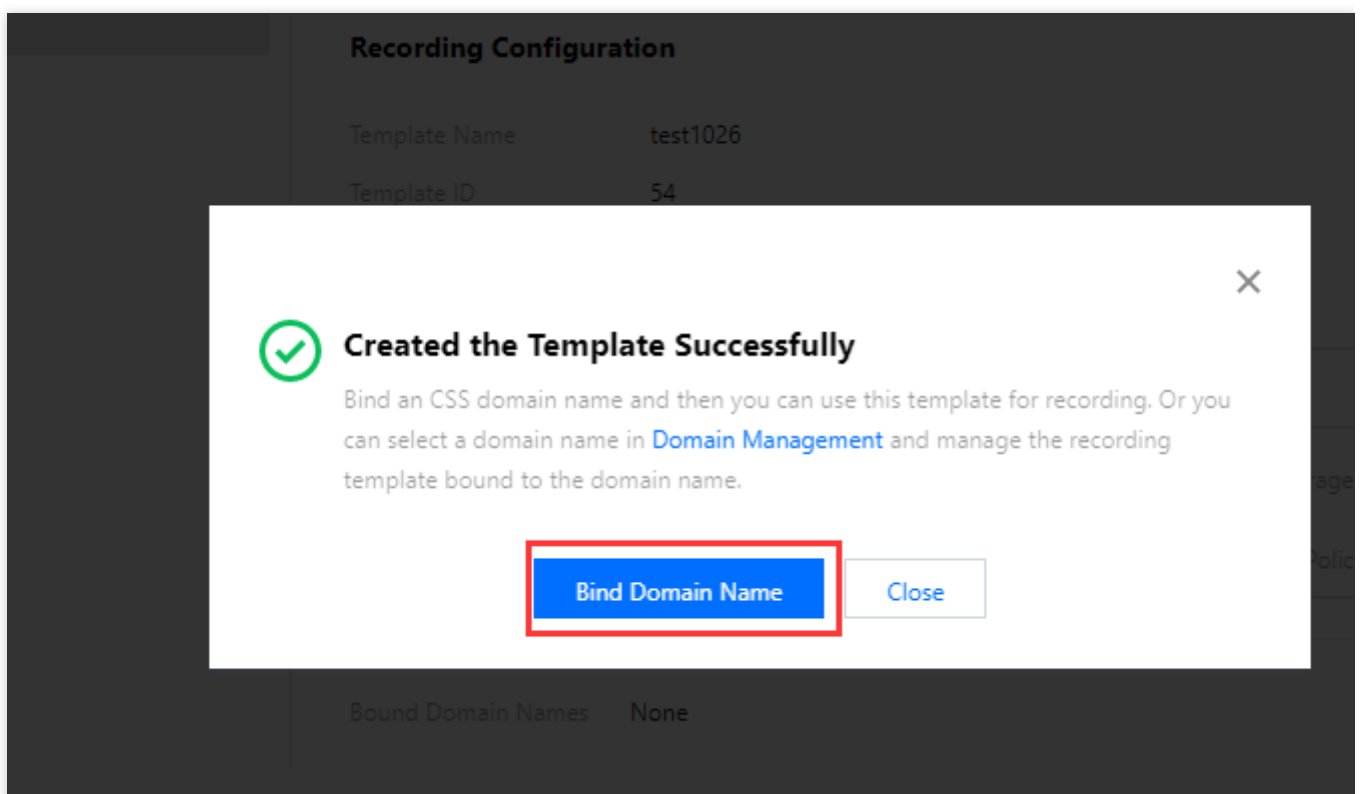
方法1：ドメイン名指定によるグローバルレコーディング

[CSSコンソール](#) またはAPIを呼び出して、CSSレコーディングテンプレートをプッシュドメイン名にバインドします。このドメイン名経由でプッシュした場合のみ自動レコーディングが行われます。

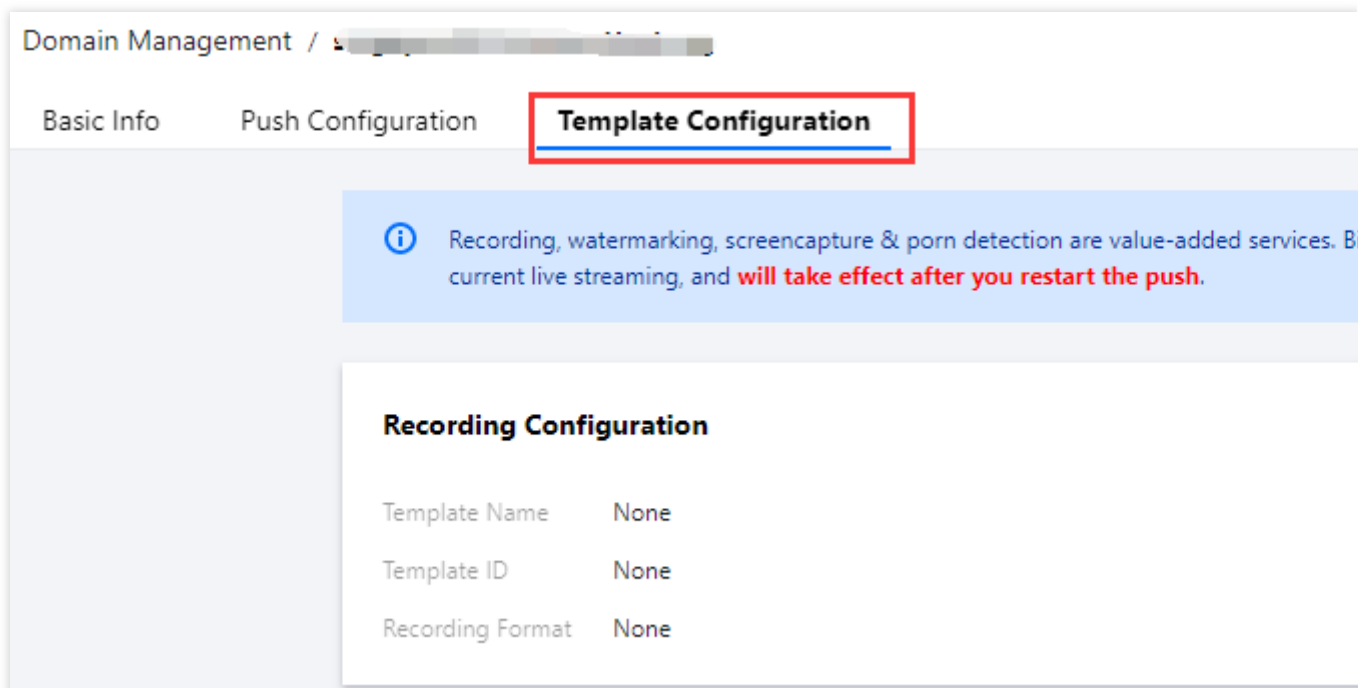
適用ケース：ショーのライブストリーミング、eコマースライブストリーミング、オンライン講座、ビデオ監視などのフルレコーディングシナリオ。

操作フロー：

1.1 レコーディングテンプレートの作成に成功すると、[ドメイン名のバインド](#)を促すポップアップボックスが表示されますので、[ドメイン名のバインドに進む](#)をクリックしてプッシュドメイン名を選択します。



1.2 [ドメイン名管理](#)の中で、お客様の[CSSプッシュドメイン名](#)をクリックすると、プッシュ詳細ページに移動します。その後、[テンプレート設定](#) > [レコーディング設定](#)と選択し、[編集](#)をクリックすれば、プッシュドメイン名をバインドできます。詳細については、[レコーディングテンプレートのバインド](#)をご参照ください。



1.3 [CreateLiveRecordRule](#) インターフェースを介してレコーディングテンプレートのテンプレートIDとプッシュドメイン名を渡せば、レコーディングテンプレートとプッシュドメイン名のバインドが完了します。

方法2：単独ストリームの指定によるレコーディング

APIを介してCSSレコーディングテンプレートを特定のCSSストリームにバインドし、これによりその特定のCSSストリームのレコーディングを実現します。

適用ケース： イベント・展示会・試合のライブストリーミングやマイク接続によるコラボライブなど、単独のイベントのための特殊なレコーディングシナリオ。

操作フロー： [CreateLiveRecordRule](#) インターフェースを介してレコーディングルールを作成し、レコーディングテンプレートのテンプレートIDとバインドしたいドメイン名、パス、ストリーム名StreamName（必ず正確にマッチさせること）を渡せば、レコーディングテンプレートと指定CSSストリームのバインドが完了します。

方法3：指定時間帯によるレコーディング

APIを呼び出すことでレコーディングの開始時間と終了時間を制御でき、指定した時間内にレコーディングタスクをトリガーしてレコーディングを行います。

適用ケース： ニュースやイベントのライブストリーミングなど、ライブストリーミングのフローが比較的明確なレコーディングシナリオ。

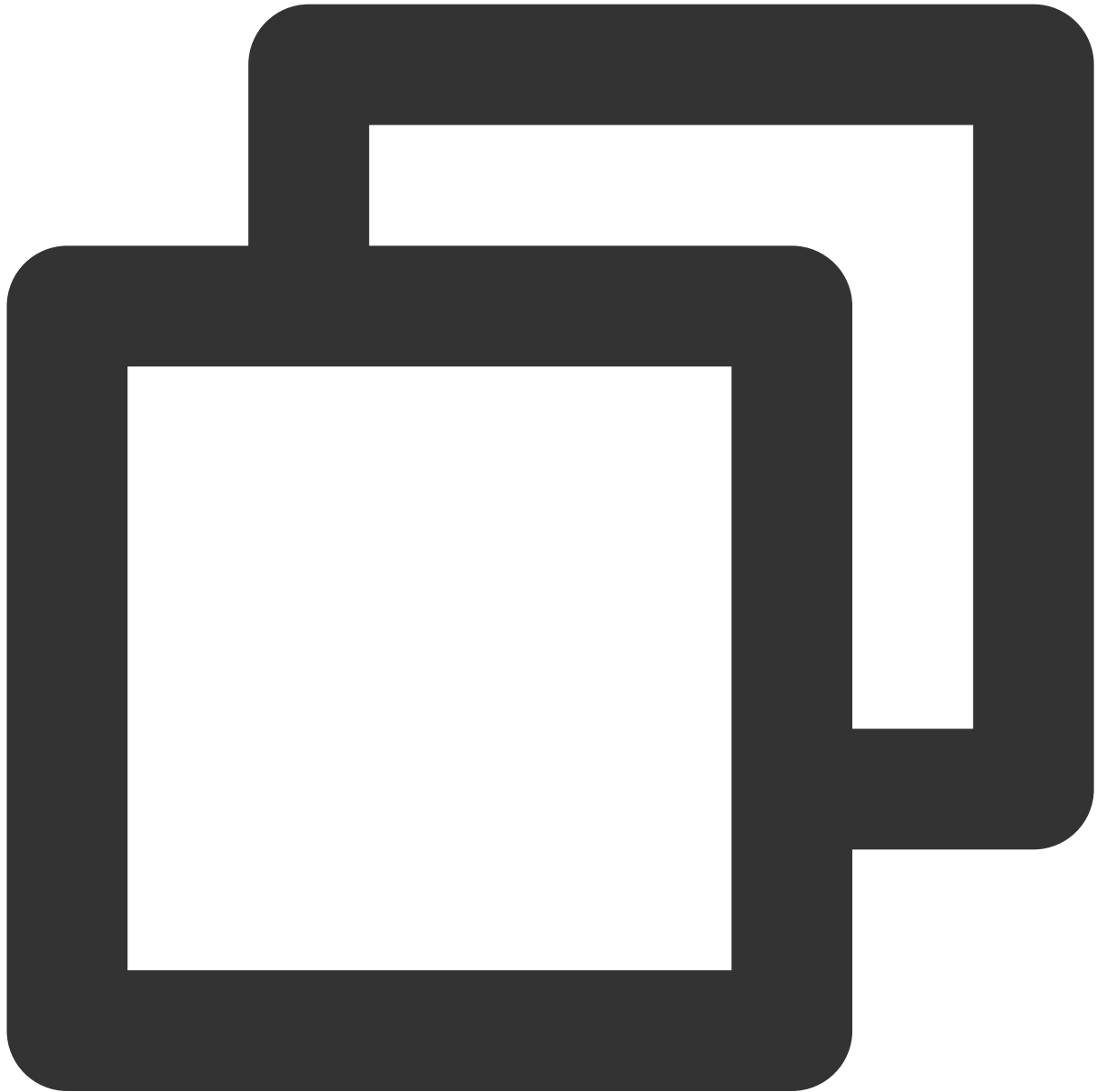
操作フロー： [CreateRecordTask](#) インターフェースを介してレコーディングタスクを作成し、レコーディングテンプレートのテンプレートIDとバインドしたいドメイン名、パス、ストリーム名StreamName（必ず正確にマッチさせること）を指定します。設定ではレコーディングの開始時間および終了時間を設定する必要があり、これにより開始時間にレコーディングがスタートします。

レコーディング例：

1. 最もシンプルなケースでは、指定されたStreamName、DomainName、AppName、EndTimeパラメータを入力するだけで利用可能です。

例：2020年08月10日午前08時から10時までのレコーディングタスクを作成、FLV形式、ビデオレコーディング、セグメント30分、永久保存。

入力例：



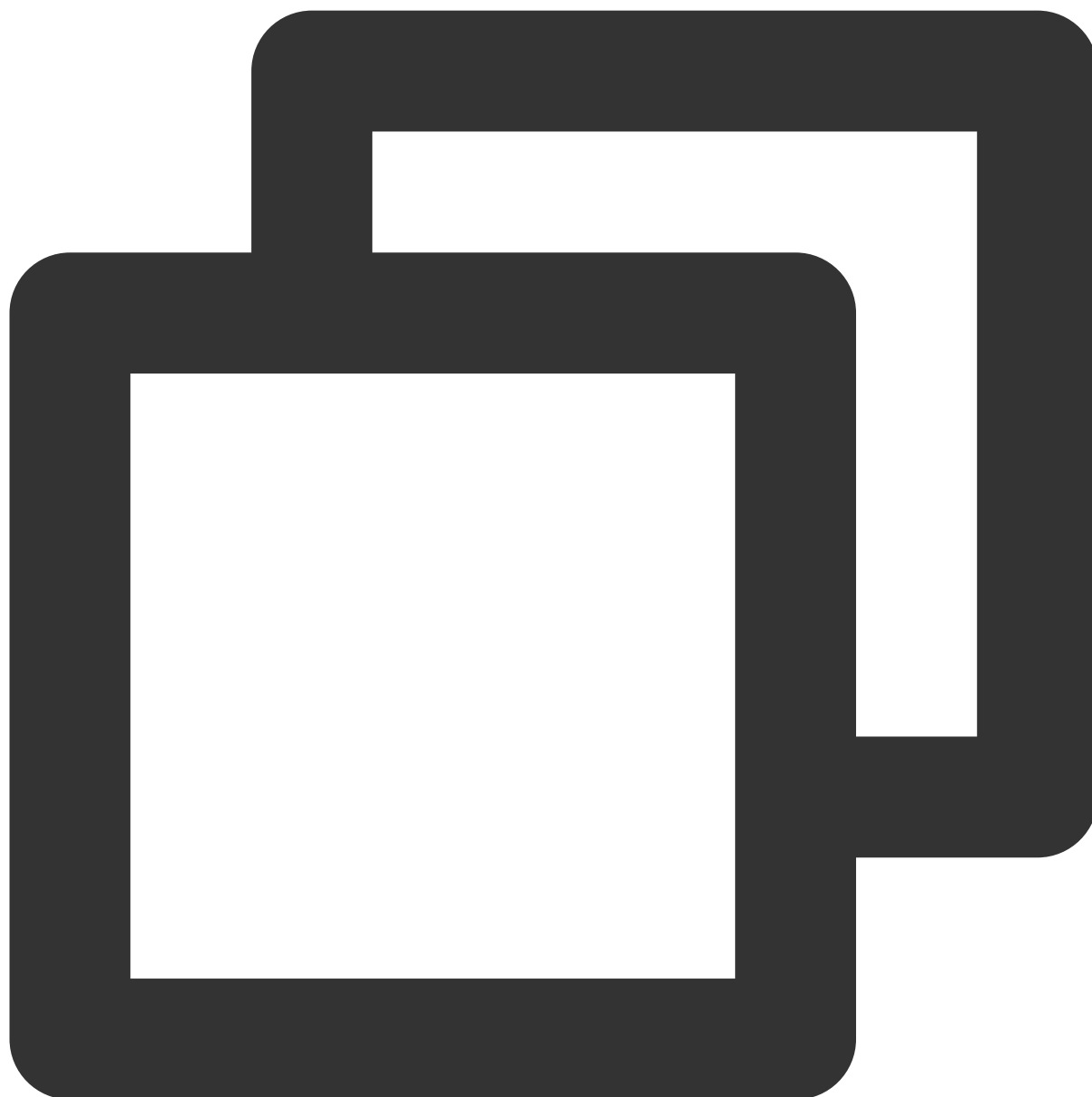
```
https://live.tencentcloudapi.com/?Action=CreateRecordTask&AppName=live&DomainName=m
```

2. さらに具体的なレコーディング形式、レコーディングタイプ、ストレージパラメータなどを指定できます。

例：2020年08月10日午前08時から10時までのレコーディングタスクを作成、MP4形式、セグメント1時間、永久保存。

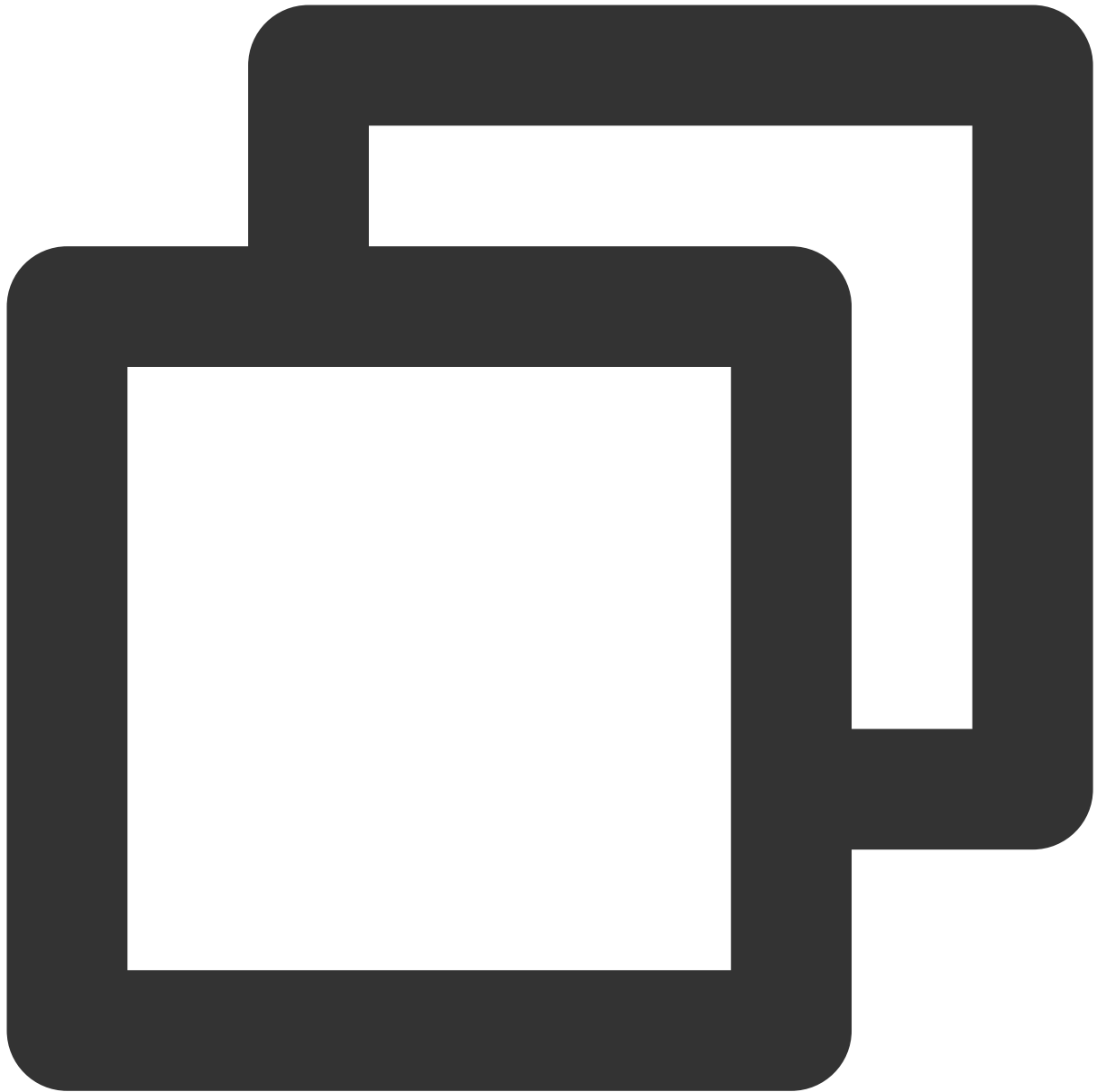
3. `CreateLiveRecordTemplate` を呼び出し、まずレコーディングテンプレートを作成します。

入力例：



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

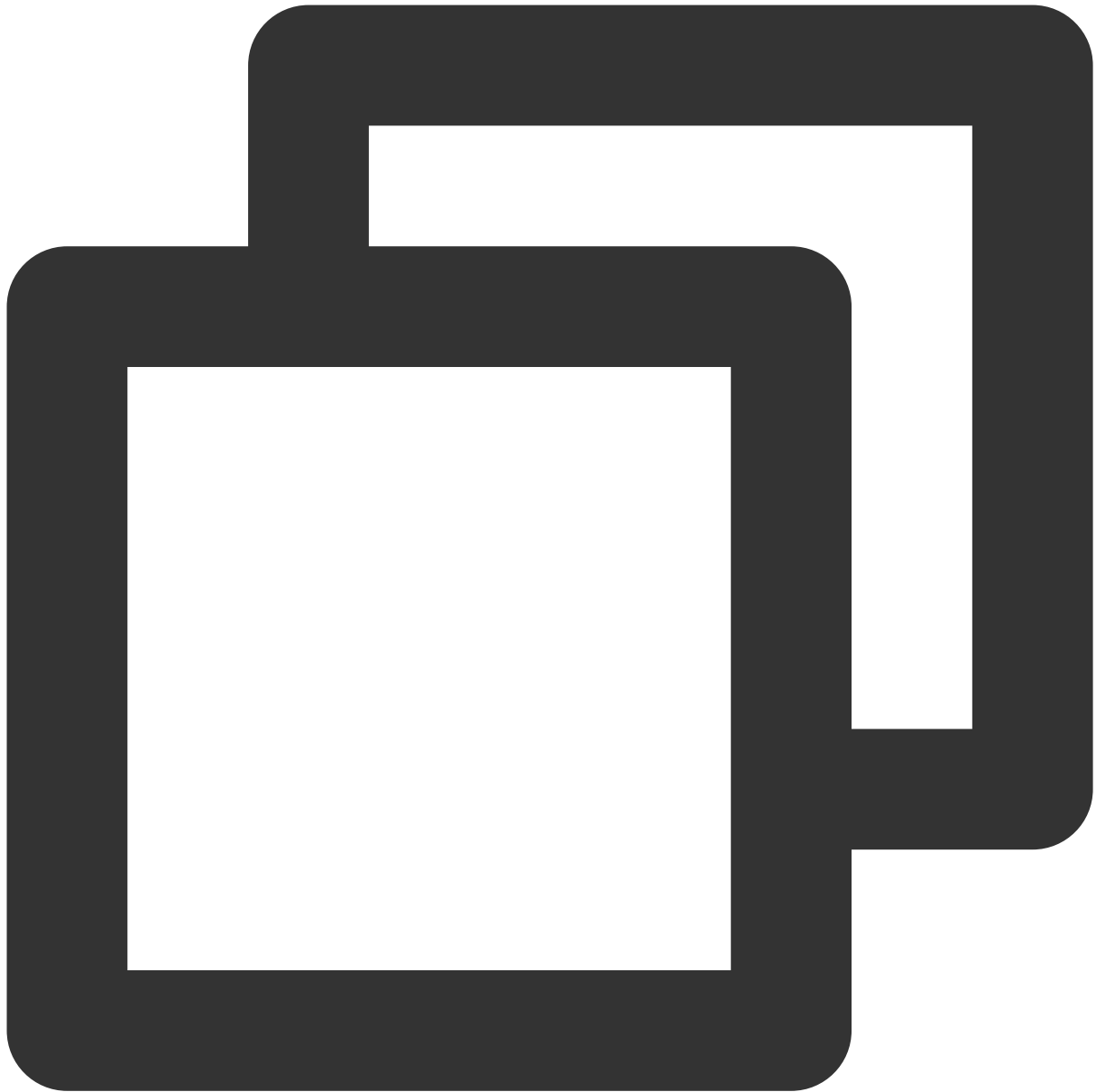
出力例：



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

4. [CreateRecordTask](#)を呼び出し、レコーディングタスクを作成します。

入力例：



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

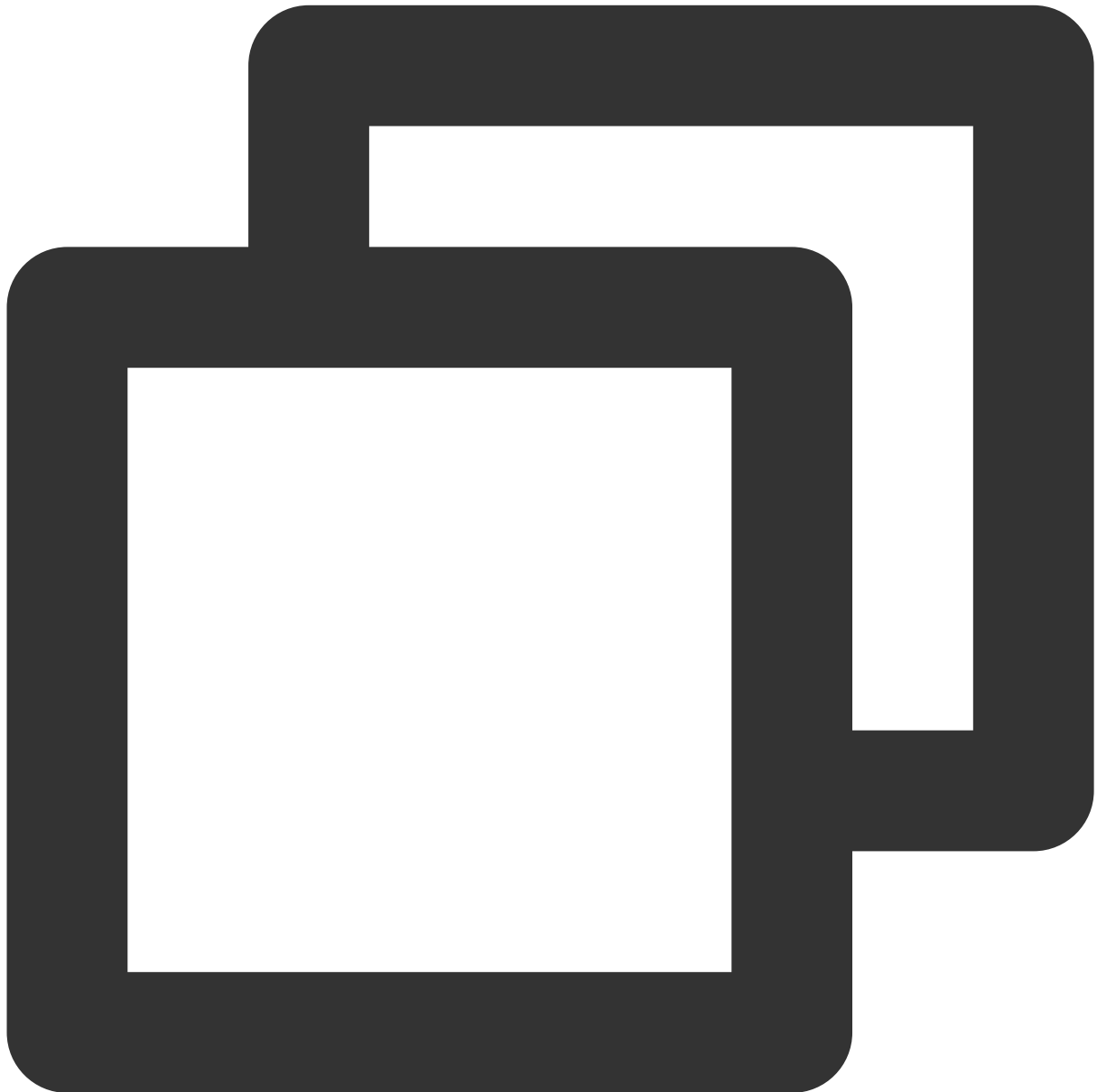
方法4：ハイライトレコーディング（ミクスストリーミングのレコーディングに対応）

ライブストリーミングの最中に素晴らしい場面に遭遇したときに、APIを呼び出してリアルタイムにCSSレコーディングを行うことができます。

適用ケース：試合やゲームのライブストリーミングなど、部分的な場面のレコーディングのみが必要なシナリオ（またはグローバルレコーディング後にトリミングを行う場合）。

操作フロー： [CreateRecordTask](#) インターフェースを介してレコーディングタスクを作成し、レコーディングテンプレートのテンプレートIDとバインドしたいドメイン名、パス、ストリーム名StreamName（必ず正確にマッチさせること）を指定します。設定ではレコーディングの終了時間を設定する必要があります。作成に成功するとレコーディングタスクがスタートします。

レコーディング例：



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

方法5：純音声のレコーディング

プッシュがオーディオのみの場合は、AACのオーディオのみのレコーディングを設定することができます。

適用ケース：音声ライブストリーミング、音声マイク接続などのシナリオ。

操作フロー：レコーディングテンプレートの作成時、レコーディングファイルタイプにACC純音声レコーディングを選択し、対応するプッシュアドレスとのバインドを行います。

注意：

バインディングルールは作成後約5~10分経ってから有効となります。バインディングルールの修正はプッシュ中のライブストリーミングには影響しません。新しくプッシュするCSSストリームに対してのみ有効となります。

ステップ3：CSSプッシュの開始

ステップ2にしたがってレコーディングテンプレートをプッシュドメイン名にバインドすると、そのプッシュアドレスによって対応するプッシュドメイン名が生成され、**CSSプッシュ**が行われます。

ライブストリーミングの終了後、レコーディングによって生成されたファイルは**VOD**プラットフォームに保存されます。

説明：

レコーディングテンプレートの中でレコーディング先にサブアプリケーションを選択すると、対応するサブアプリケーションの下に保存されます。

レコーディングファイルのアドレス情報をコールバックしたい場合は、プッシュ前にコールバックテンプレートを作成して、レコーディングコールバックアドレスを入力してから保存し、さらにコールバックしたいプッシュドメイン名をバインドする必要があります。詳細については、[レコーディングイベント通知](#)をご参照ください。

ステップ4：レコーディングファイルの取得

次の方式によるレコーディングファイルのクエリーと取得をサポートしています。

レコーディングコールバックによる取得：CSSプッシュの前にコールバックテンプレートを設定し（テンプレートには[レコーディングコールバックアドレスの設定](#)が必要です）、レコーディングファイル生成時にコールバックによってファイルをコールバックサーバーに送信します。詳細については、[レコーディングイベント通知](#)をご参照ください。

The screenshot shows the 'Create Callback Template' interface in the Tencent Cloud VOD console. At the top, there are two buttons: 'Create Callback Template' (highlighted with a red box) and 'Bind Domain Name'. Below these is a list of templates, with 'test' selected. The main area is titled 'Callback Configuration' and contains several input 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)
- Screenshot Callback: Enter a screenshot 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.

VODコンソールによる取得：[VODコンソール](#)で確認を行います、具体的な内容は、[ビデオの確認](#)をご参照ください。

VOD APIによる取得：[SearchMedia](#)インターフェースを呼び出してファイルの照会を行います。

ステップ5：レコーディングファイルの処理

方法1：CSSレコーディング + 自動トランスコーディング + ビデオアクセラレーション再生

適用ケース：CSSレコーディング後にレコーディングファイルに対する自動トランスコーディングとビデオアクセラレーションを素早く行い、ユーザーがVODを再生できるようにします。ビデオ二次加工が不要なほとんどのライブストリーミングシナリオに適用できます。

操作フロー：

1.1 CSSプッシュのレコーディング前にレコーディングテンプレートを作成し、**高度な設定**をクリックしてタスクフローを設定します。

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 ⓘ

▼ Audio/Video - HLS

Max Recording Time Per File min

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

Storage Period Permanent Custom

VOD Subapplication/Category

Advanced Configuration ▼

1.2 VODコンソールで事前に作成したタスクフローテンプレートをバインドします。

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 お客様がCSSプッシュを行う場合は、[CSSプッシュ](#)をご参照ください。

1.4 CSSレコーディングが完了したら、オンデマンドのFileIdを取得します。

Callback information ✕

Callback request

```
{"file_id": "BAK_3701925923130574711", "file_size": 15239029, "media_start_time": 4134330, "record_bps_id": "BAK_3701925923130574711", "sign": "ec4a9c9431b3ea34ab273915135491db", "start_time": 1629695838, "start_time_usec": 491893, "stream_id": "LIVE102", "stream_param": {"LIVE102": {"task_id": "1722212987248255541", "video_id": "1258442684_ae428fe56ba14e7
```

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 再生アドレスを取得し再生します。

方法2：CSSレコーディング + 手動トランスコーディング + ビデオアクセラレーション再生

適用ケース：一部のユーザーは、CSSレコーディングのビデオを先にVODに保存だけして、その後のトランスコーディングの操作は行わないことを希望することがあります。この場合は新規作成したレコーディングをVODに保存する時に、その他の操作を追加しないことができます。後からビデオに対してトランスコーディングを行いたい場合は、手動でトランスコーディング操作を起動することができます。また、VODのクラウドトリミング機能と連携させて使用すれば、より素晴らしい効果が得られます。

操作フロー：

- 1.1 お客様がCSSプッシュを行う場合は、[CSSプッシュ](#)をご参照ください。
- 1.2 ファイルが自動的にレコーディングされ、VODに保存されます。
- 1.3 VOD FileIdを取得します。
- 1.4 トランスコードテンプレートを設定するか、タスクフローを手動でトランスコードします。詳細については、[テンプレート設定](#)をご参照ください。

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 Task Dynamic Image Task

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 -> 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 お客様は、ビデオの二次編集を選択することができます。
- 1.6 トランスコーディングおよび処理の完了後、ビデオアドレスを取得してその後の再生を行います。

方法3：CSSレコーディング + アダプティブビットレートストリーミング + ビデオアクセラレーション + プレーヤー

適用ケース：一部のユーザーは、ビデオのセキュリティに対して極めて高い要求があり、一般的なHLS暗号化ではその暗号化に対するニーズを満たすことができませんが、ABSおよびPlayer SDKを組み合わせ使用すれば、ビデオのセキュリティレベルを効果的に高めることができます。eラーニング、企業のトレーニングのような顧客シナリオに非常に適しています。

操作フロー：

- 1.1 お客様がCSSプッシュを行う場合は、[CSSプッシュ](#)をご参照ください。
- 1.2 ファイルが自動的にレコーディングされ、VODに保存されます。

1.3 VOD FileIdを取得します。

1.4 ABSに変換するタスクフローを設定します。詳細については、[タスクフロー設定](#)をご参照ください。

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 プレーヤーの構成を設定し、作成したABSを選択して再生します。

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 FileIdを使用してビデオを再生します。

ライブストリーミング中のキャッチアップ機能の実装方法

最終更新日：：2023-10-26 17:31:32

タイムシフト視聴は、CSSレコーディングの機能に依存し、CSSタイムシフトとVODのアクセラレーション配信を結合させて、ライブストリーミング中にプレイバック視聴する機能を実現しています。ユーザーは、配信開始後のある過去の時間点をリアルタイムに選択してプレイバック視聴することができ、これにより、それ以前のライブストリーミングコンテンツを再生する効果を達成できます。ライブストリーミング中の試合のハイライトをプレイバック視聴するためによく利用されますが、ユーザーは、プログレスバーを直接ドラッグすればプレイバック視聴でき、ライブストリーミングの終了まで待つ必要がありません。プレイバック視聴の最中もCSSストリームは変わらず維持されます。また、プレイバックの途中で切り替え、ライブ映像に戻すことも可能です。

機能特性

プレイバック視聴の相対的なタイムシフト時間（プレイバック視聴の起点と現在の時間の差）を指定することが許されています。

ライブストリームにマルチビットレートを使用してレコーディングしているときは、指定ビットレートのビデオストリームを指定することが許されています。

前提条件

Tencent Cloudアカウントを[登録](#)して、[ログイン](#)します。

Tencent Cloud CSSサービスをアクティブ化し、[プッシュドメイン名](#)を追加済みであること。

注意事項

タイムシフト視聴機能を使用する前提条件は、ターゲットとなるライブストリームで[タイムシフト視聴](#)機能を有効にしていることです。

相対的なタイムシフト時間は、現在最小90秒です。すなわち、プレイバック視聴のコンテンツは、ライブストリーミングのコンテンツに対して90秒以上のディレイがあります。

現在、タイムシフト機能の内部テストに対する個別の課金はありません。CSSタイムシフト機能は**2022年6月1日0時**をもってベータ版テストを終了し、正式に課金を開始します。詳細については[お知らせ](#)をご参照ください。タイムシフト機能を有効にするにはCSSレコーディング機能の使用が必要となり、[CSSレコーディング料金](#)、[VODストレージ料金](#)および[ビデオ再生料金](#)が発生します。

実践手順

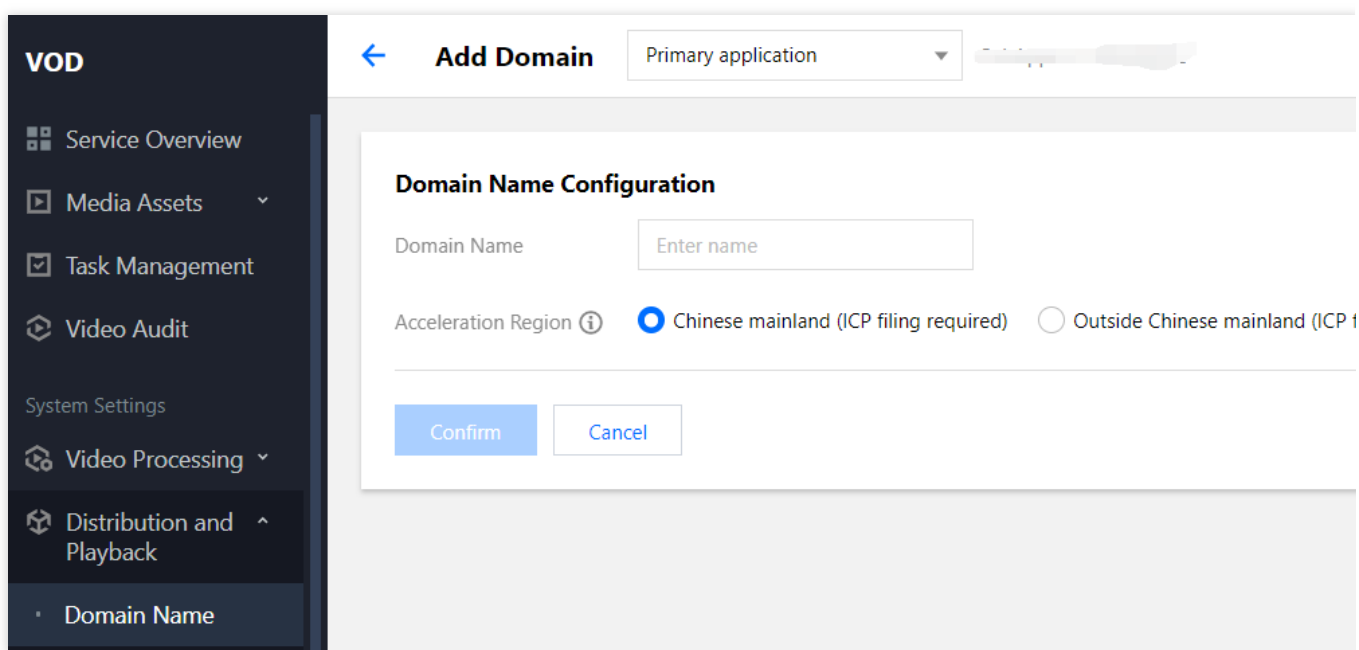
ステップ1：VODサービスのアクティブ化

1. [VODコンソール](#)に移動し、[今すぐアクティブにする](#)をクリックします。
2. サービスプロトコルのアクティブ化にチェックを入れ、**OK**をクリックすると、VODサービスは正常にアクティブ化し、VODコンソールにアクセス可能になります。

ステップ2：タイムシフト再生ドメイン名の追加

タイムシフト再生に使用するVODドメイン名を追加する必要があります、具体的な操作は次のとおりです。

1. [VODコンソール](#)の [アプリケーション管理](#)にて、アプリケーション名をクリックし、アプリケーションリスト画面に進みます。左側ナビゲーションバーの [配信再生管理](#) > [ドメイン名管理](#) をクリックします。
2. [ドメイン名の追加](#) をクリックし、ICP登録済みのVODドメイン名を入力します。具体的な操作については、[配信と再生の設定](#) をご参照ください。



3. 新規追加ドメイン名のCNAME設定操作の完了。

ステップ3：レコーディングテンプレートとの関連付け

1. [CSSコンソール](#)に入り、[機能設定](#) > [CSSレコーディング](#) をクリックします。
2. [レコーディングテンプレート作成](#) をクリックします。具体的な操作方法は、[レコーディングテンプレート設定](#) をご参照ください。

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

Resumption Timeout

This value will affect when to generate a recording file.

Storage Period Permanent Custom

VOD Subapplication/Cate gory

[Advanced Configuration](#) ▼

注意：

レコーディングファイルタイプにHLS形式を選択し、HLSレコーディングを有効にします。

ファイル保存時間をカスタマイズします。この時間は[タイムシフト時間](#)より短くできません。

3. レコーディングテンプレートを設定が必要なプッシュドメイン名と関連付けします。具体的な操作方法は、[レコーディング設定](#)をご参照ください。

ステップ4：タイムシフトサービスのアクティブ化

[チケットを提出](#)し、質問が属する製品に「CSS」を選択して、タイムシフトサービスのアクティブ化申請のリクエストを送信してください。以下のパラメータも提供します。

[ステップ2](#)の中で、すでに追加されているVODタイムシフト再生ドメイン名。

[ステップ3](#)の中のレコーディングテンプレートID。

カスタマイズしたタイムシフト時間 timeshift_dur。単位：秒。

説明：

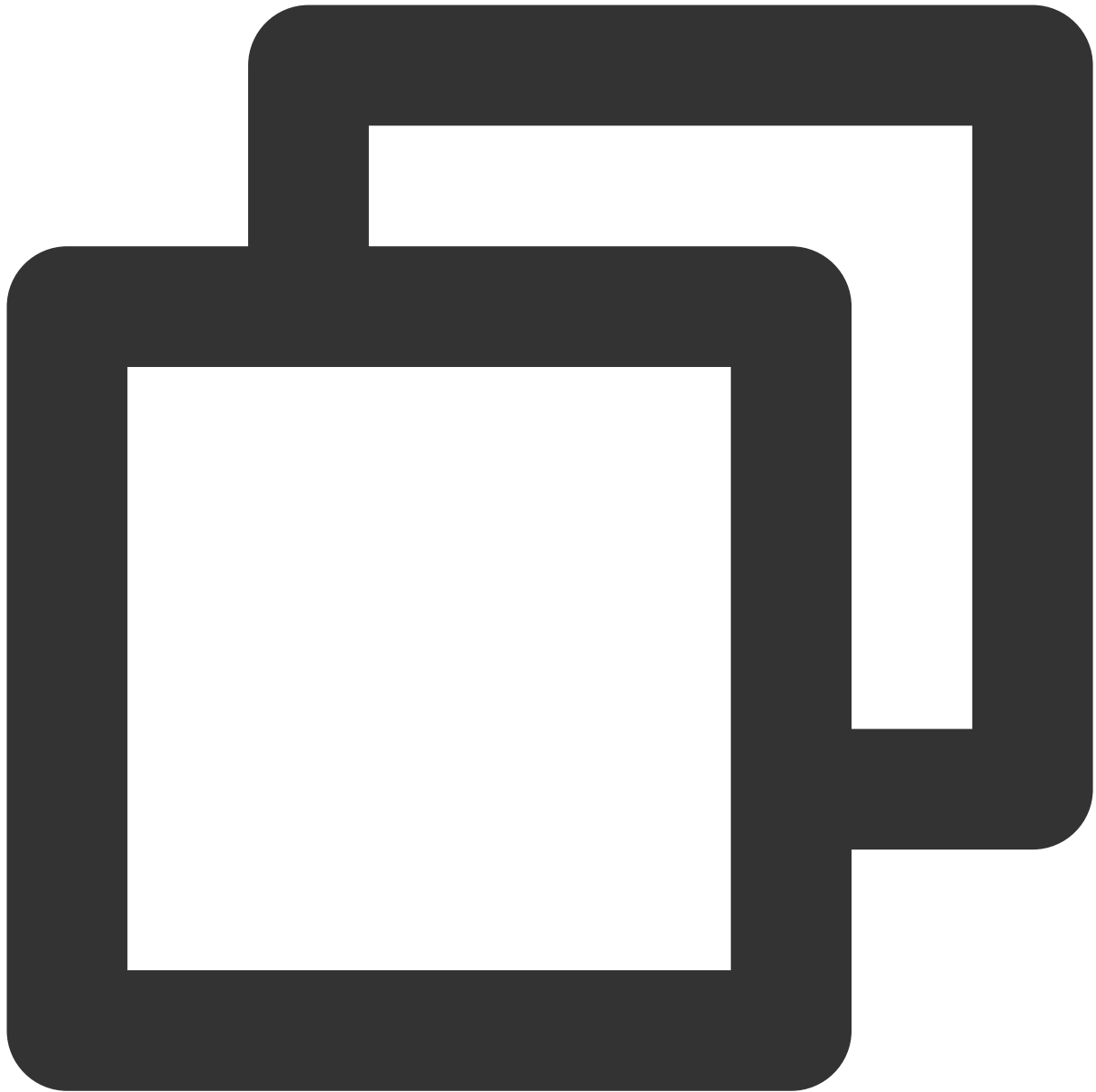
タイムシフト時間：タイムシフトでコンテンツを視聴できる時間（現在、タイムシフトで最長で7日以内のコンテンツを設定できます）。

この項目は絶対的な精度を保証するものではありません。ニーズに応じて設定することを前提とし、時間に余裕を持たせることをお勧めします。

7200（2時間）に設定する場合：目下、2時間前から現在までのタイムシフトされたコンテンツをリクエストできることを意味します（つまり、**delay**の相対的なタイムシフト時間を90秒～2時間とすることができます）。2時間以上前にタイムシフトしたコンテンツは、ライブストリーミングコンテンツが存在しても、HTTP 404が返されません。

再生リクエスト

リクエストURL形式



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

パラメータの説明

パラメータ	説明
[Domain]	登録したタイムシフトサービスアクセスドメイン名。VODコンソールで追加したタイムシフト再生ドメイン名です。
timeshift	固定のフィールド。変更は不要です。

[AppName]	アプリケーション名。アプリケーション名がliveであれば、liveと入力してください。
[StreamName]	ストリーム名。リクエストに対応するストリーム名を入力してください。
timeshift.m3u8	固定のフィールド。変更は不要です。
delay	相対的なタイムシフト時間を意味し、単位は秒。現在、この値は90未満ですが、バックグラウンドで90に調整できます。

事例

現在のタイムシフトドメイン名を `testtimeshift.com` 、タイムシフトアプリケーション名を `live` 、ストリーム名を `SLPUrIFzGPE` とした場合、このアドレスの5分前のライブストリーミングコンテンツをタイムシフトしたい場合は、次のURLをリクエストします。

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

VODビデオから疑似ライブストリーミング 効果を生み出す方法

最終更新日：：2023-10-26 17:31:32

疑似ライブストリーミングは、VODの再生制御能力に依存し、VODファイルに「視聴時間の制限」と「視聴の進行状況の同期」の2種類のアクセス制御の機能を追加することで、VODファイルにライブストリーミングに似た効果をもたらします。ユーザーはまずVODファイルを生成してから、指定するライブストリーミング時間に、VODファイルを使用してライブストリーミングに類似する配信を行うことができ、ライブストリーミングにおけるリスク低減とコスト削減が効果的に行えるようになっています。

機能特性

特徴	説明
低開発コスト	オンデマンドビデオを標準のライブストリーミングに変換して配信することを選択した場合、ユーザーはOBSを使ってビデオをライブストリーミングプラットフォームにプッシュした上で、ライブストリーミングシステムに接続する必要がありますが、高い開発コストがかかってしまいます。これと比較すると、疑似ライブストリーミングはVOD内部で実現でき、ユーザーはトランスコードしてリンク不正アクセス防止機能を有効にするだけで完了です。
不正コンテンツのリスク低減	疑似ライブストリーミング機能を使用することで、ユーザーは事前に自分のVODファイルに対する審査と編集を行うことができ、ライブストリーミング中に抵触する可能性があるコンプライアンス違反のリスクを効果的に回避できます。不正なコンテンツを回避し、ライブストリーミングの品質を向上させます。
シンプルでフレキシブルな作成	ライブストリーミングルームという概念はなく、あらゆるビデオでいつでも、疑似ライブストリーミングを生成できます。 同時実行数の制限はなく、開始時刻を指定して、あらかじめ視聴リンクを配布することができます。

ユースケース

主にビデオの事前レコーディングが完了していて、予定の時刻にユーザーが同期視聴できるようにスケジューリングするケースに用いられます。ユーザーは事前に視聴エントリーを取得することはできますが、開始予定時刻より前にビデオを視聴することはできません。

疑似ライブストリーミングの進行中は早送りできません。eラーニング、ライブストリーミングのパーティやラジオ・テレビなどの業種でよく用いられます。

ご利用説明

使用制限

疑似ライブストリーミングは本質的にはオンデマンド配信であるため、次のような標準ライブストリーミングの機能は備えていません。

「1回限り」の疑似ライブストリーミングに対するデータ統計はサポートしていません。

「1回限り」の疑似ライブストリーミングの場合、開始と終了を認識できません。

進行中の疑似ライブストリーミングに対する一時停止/終了などの操作はサポートしていません。

配信済みの疑似ライブストリーミングのリンクの無効化はサポートしていません。

ビデオコンテンツの動的な変更（リアルタイムのトランスコーディング、ウォーターマークなど）はサポートしていません。

用語の説明

視聴許可時間の制限：ビデオエントリーは、視聴者に事前配布することができます。視聴者は疑似ライブストリーミングの開始前または終了後に視聴することはできません。疑似ライブストリーミングは、「進行」中のときのみ視聴することができます。

視聴進行状況の同期：疑似ライブストリーミングの「進行」中は、すべての視聴者の視聴進行状況が同期されま（分単位でのずれはあります）。

前提条件

Tencent Cloudアカウントを[登録](#)して、[ログイン](#)し、アカウントの実名認証を完了していること。

Tencent Cloud CSSおよびVODサービスがアクティブになっていること。アクティブになっていない場合は、[CSSサービス](#) および[VODサービス](#)にてアクティブ化してください。

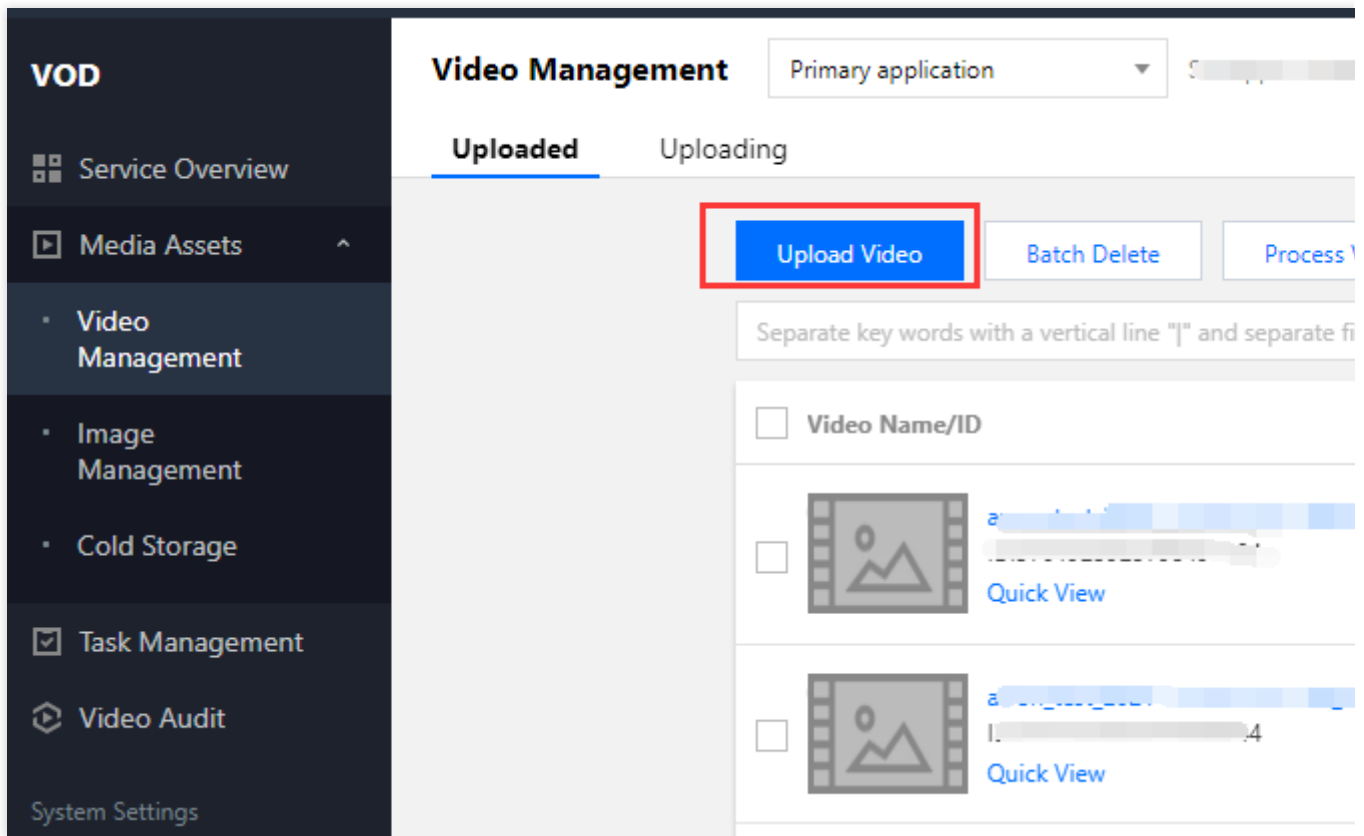
CSSレコーディングを行います。詳細については、[CSSレコーディングのVODへの変換](#)をご参照ください。

実践手順

ステップ1：ビデオをVODにアップロードする

1. [VODコンソール](#)にログインし、左側ナビゲーションバーの[アプリケーション管理](#)をクリックし、アプリケーションリストページに進みます。

- オーディオビデオを処理したいアプリケーションを見つけ、アプリケーション名をクリックし、アプリケーション管理ページに進みます。
- デフォルトでメディア資産管理 > オーディオビデオ管理画面に進み、オーディオビデオをアップロードをクリックします。



業務の状況に応じて適した方法を選択し、ビデオファイルをVODにアップロードすることも可能です。その他のアップロード方式は、[メディアアップロードの概要](#)および[CSSレコーディングのVODへの変換](#)をご参照ください。

ステップ2：ビデオをHLSにトランスコードする

- 疑似ライブストリーミングを使用するためにはHLS形式をベースにする必要があります。[トランスコードのタスクの開始](#)の説明にしたがって、アップロードしたビデオをHLSにトランスコーディングすることができます（具体的な[トランスコードテンプレート](#)は、業務にもとづき選択してください）。
- トランスコーディングが完了したら、コンソールのオーディオビデオ管理画面でHLSのURLを[確認](#)するか、または[イベントの通知](#)を受信する方式でHLSのURLを取得します。

The screenshot shows the 'Video Management' console with the 'Basic Info' tab selected. The video name is partially visible as 'a...'. The status is 'Normal', and it was created on 2021-10-12 11:43:08 and last updated on 2021-11-03 09:46:44. Below the basic info is a 'Standard Transcoding List' with one entry: 'STD-H264-HLS-360P' in 'HLS' format with a resolution of '640 x 360' and a bitrate of '273.17Kbps'. A 'Delete' button is visible above the table.

ステップ3：Keyリンク不正アクセス防止をアクティブにする

1. 疑似ライブストリーミングを使用するには、リンク不正アクセス防止を有効にする必要があります。[VODコンソール](#)にログインして、左側ナビゲーションバーの[アプリケーション管理](#)にてターゲットアプリケーション名をクリックし、[配信と再生の設定 > ドメイン名管理](#)を選択し、ターゲットドメイン名がある行の[設定](#)をクリックしてください。ドメイン名関連の設定画面に入ります。

The screenshot shows the 'Domain Management' console. A table lists domain configurations. One domain is shown with a lock icon, a status of 'Enable', and a CNAME. Below the table, there are links for 'Domain Name Guide' and 'CNAME Guide'.

Domain Name	Status	CNAME
[Redacted] .com	Enable	[Redacted]

2. 「アクセス制御」画面で、Refererリンク不正アクセス防止、Keyリンク不正アクセス防止を有効にします。

test.com test Su

Basic Configuration **Access Control**

Referer Hotlink Protection

The HTTP Referer header is used for source identification and authentication. For details, see [Referer Hotlink Protection](#).

Referer Hotlink Protection

No Referer hotlink protection rule set.

Key Hotlink Protection

Key hotlink protection URL carries video playback control parameters. For details, see [Key Hotlink Protection](#) [Key Hotlink Protection URL Ge](#)

Key Hotlink Protection

No key hotlink protection rule set.

Certificate Management

The domain name for which you configure the certificate must be connected to Tencent Cloud CDN, and the domain name status must be "de applied to all service regions of the selected domain name. For details, see [Billing Documentation](#).

Use Certificate

No certificate is configured.

- 有効化すると以下のようになります。その他の選択項目の説明については、[リンク不正アクセス防止の設定](#)をご参照ください。設定が完了したら、リンク不正アクセス防止KEYの内容を保存してください。これは以下のリンク不正アクセス防止署名の計算に使用します。

The screenshot shows the 'Access Control' configuration page in the Tencent Cloud console. The page is divided into three main sections: 'Referer Hotlink Protection', 'Key Hotlink Protection', and 'Certificate Management'. A modal dialog titled 'Enable Referer Hotlink Protection' is open over the first section. The modal contains a list of instructions, radio buttons for 'Referer Type' (Allowlist selected), a checkbox for 'Allow Empty Referer', a text input field for domain names, and a 'Confirm' button.

Referer Hotlink Protection

The HTTP Referer header is used for source identification and authentication. For details, see [Referer Hotlink Protection](#)

Referer Hotlink Protection

No Referer hotlink protection rule set.

Key Hotlink Protection

Key hotlink protection URL carries video playback content. For details, see [Key Hotlink Protection](#)

Key Hotlink Protection

No key hotlink protection rule set.

Certificate Management

The domain name for which you configure the certificate is the domain name. For details, see [Billing Documentation](#)

Use Certificate

Certificate content Update

Enable Referer Hotlink Protection

- No need to enter "http://". Use "Enter" to separate domain name per line without duplication.
- If "Allow Empty Referer" is not checked and the protection is not enabled.

Referer Type Allowlist Blocklist

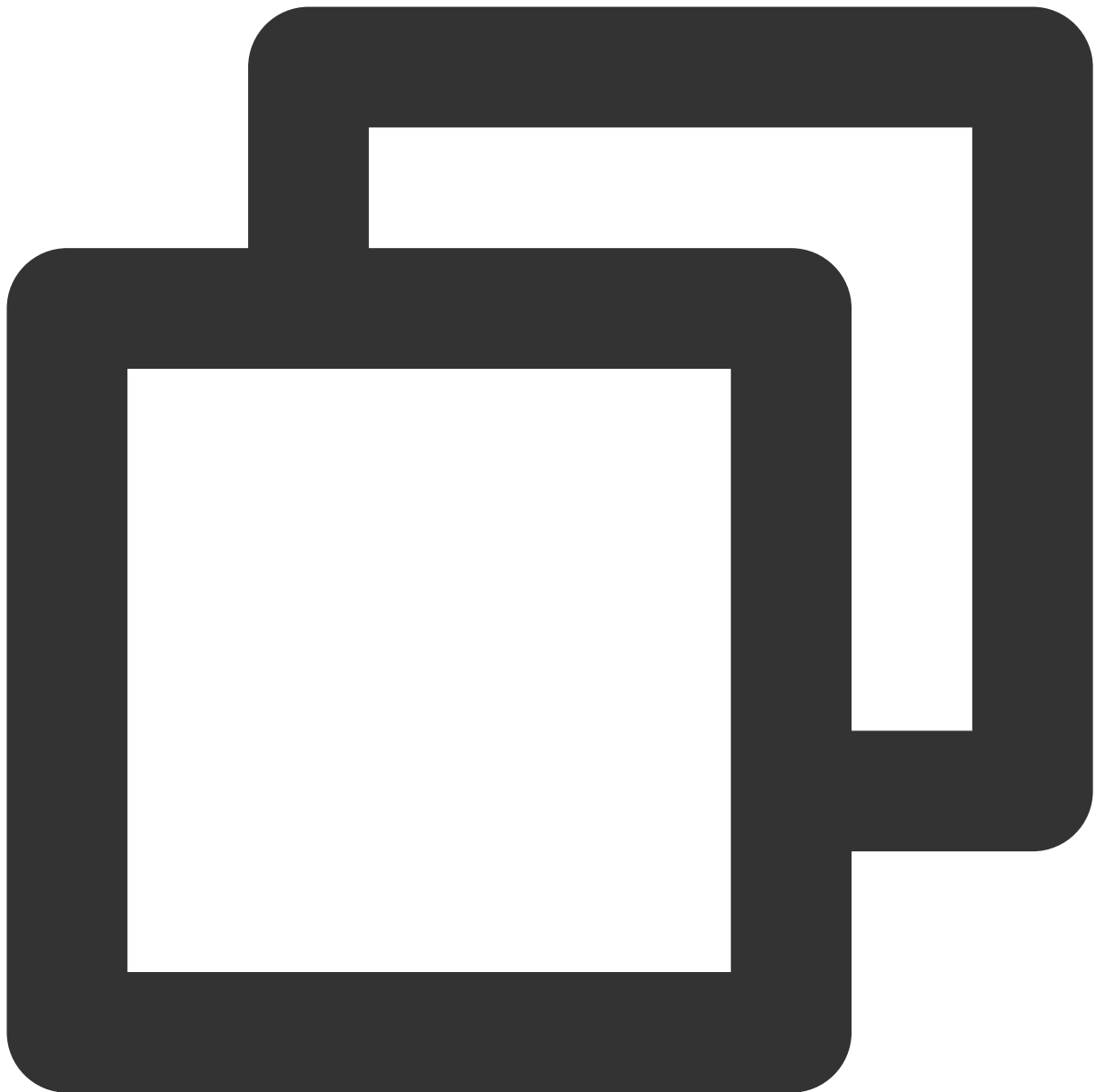
Allow Empty Referer ⓘ

Enter domain names, such as www.test.com, in the beginning, such as *.test.com

Confirm

ステップ4：リンク不正アクセス防止署名の計算

署名の計算式



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

注意：

疑似ライブストリーミングのリンク不正アクセス防止パラメータは、[標準のKeyリンク不正アクセス防止](#)よりもパラメータ（`plive`）が1つ多くなります。リンク不正アクセス防止の署名を計算するときは、`plive` を計算に加える必要があります。

パラメータ名	値	説明

KEY	11111111	開発者がKeyリンク不正アクセス防止をアクティブにするときに選択したキー。
Dir	/dir1/dir2/	元の再生URLのPATHからmyVideo.mp4を除いた残りの部分
t	5a71afc0	有効期限タイムスタンプ1517400000を16進数で表した結果。
plive	5e344f00	疑似ライブストリーミングの開始時間（北京時間）は、UNIXタイムスタンプの形式で表します。例えば1577808000は、2020-01-01 00:00:00の時点を示しています。
exper	0	プレビュー時間の長さ。0は無制限を表します。
rlimit	0	視聴IP数の制限。0は無制限を表します。
us	test	生成されたランダムな文字列。

例えば、ある開発者はVODにビデオがあり、そのHLSのURL（元のビデオのURLではない）の再生アドレスは、`http://1250000000.vod2.myqcloud.com/vodtranscq125000000/12345678/v.f240.m3u8;` です。この場合、現時点で次のような要件があります。

HLSのURLリンク不正アクセス防止KEYは、`11111111` です。

有効期限tは、`5e5a8a80`（2020-03-01 00:00:00）です。

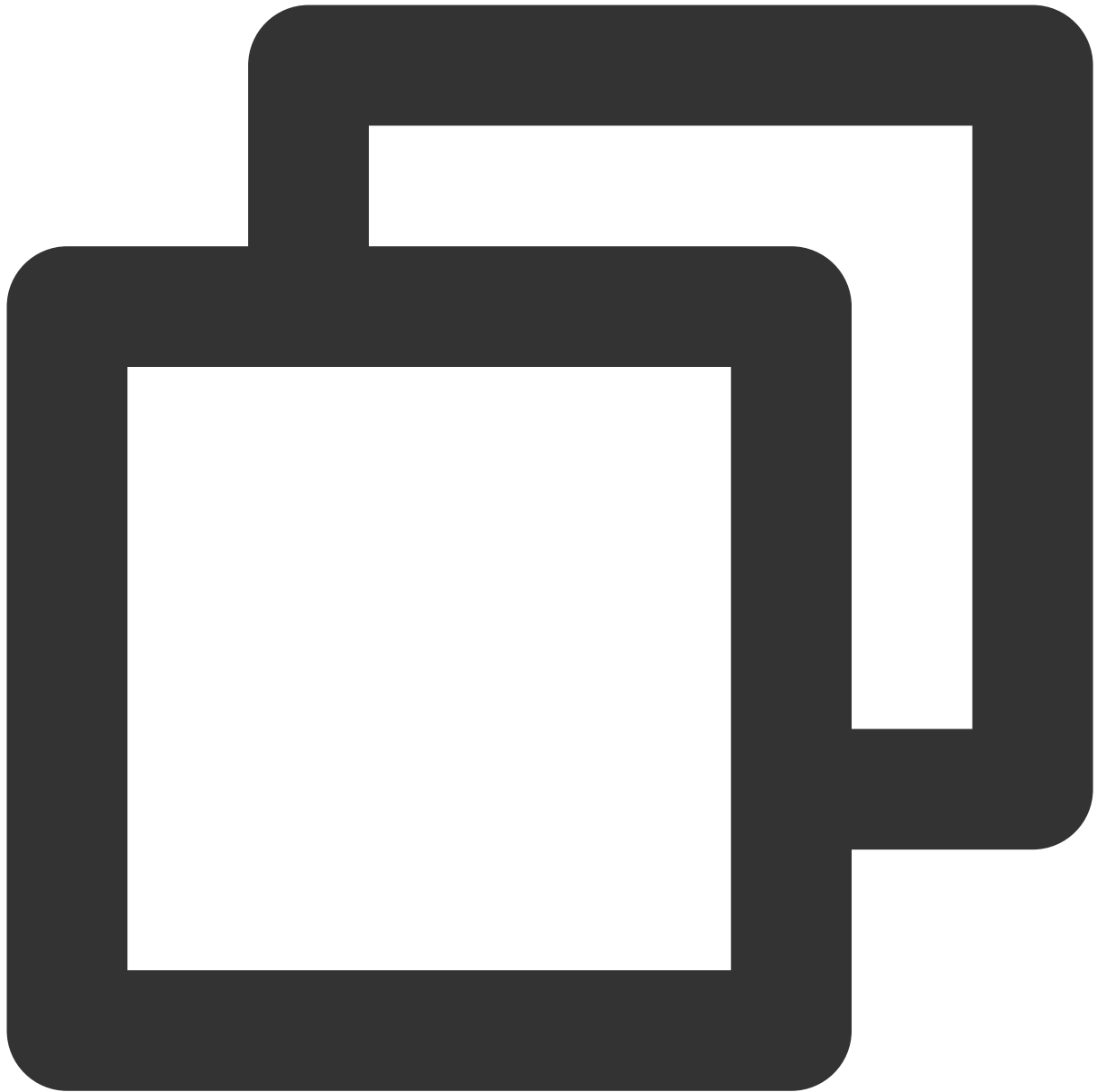
疑似ライブストリーミングの開始時間pliveは、`5e344f00`（2020-02-01 00:00:00）です。

プレビュー時間の長さexperは、`0`（無制限）です。

視聴IP数の制限rlimitは、`0`（無制限）です。

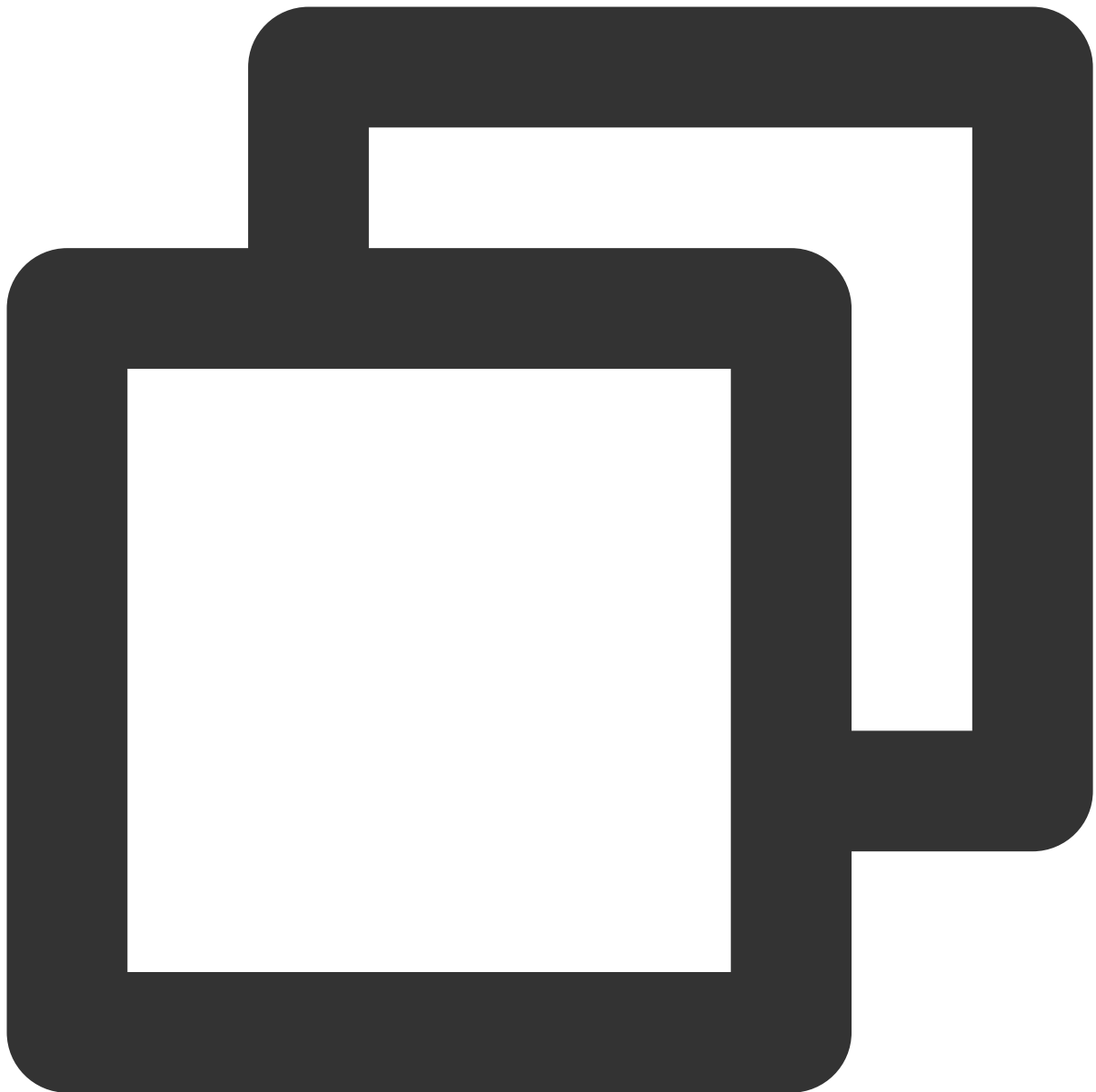
ランダムな文字列usは、`test` です。

1. [署名の計算式](#) に基づいて署名を計算します。



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

2. 計算された署名をHLSのURLのQueryStringにつなげて、最終的に生成されるリンク不正アクセス防止リンクは、次のとおりです。



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

説明：

リンク不正アクセス防止リンクQueryStringにおけるパラメータの順序は、`sign` の計算時と同じである必要があります。すなわち、`t-plive-exper-rlimit-us-sign` となります。

開発者がデバッグを実行しやすいように、[リンク不正アクセス防止署名発行ツールページ](#)を提供しています。ページの内容に応じてパラメータを入力すると、署名計算の中間結果と最終的なリンク不正アクセス防止リンクを表示することができます。

疑似ライブストリーミングの体験

HLS再生をサポートするプレーヤー（例えば、Safariブラウザ、VLC、PotPlayerなど）では、上記のリンク不正アクセス防止リンクにアクセスして、直接体験することができます。

注意：

ChromeブラウザはデフォルトではHLSをサポートしていないため、プラグインをインストールする必要があります。

Video on DemandおよびWordPressを使用したウェブサイトの構築方法

最終更新日： : 2023-10-26 17:31:32

概要

WordPressは有名なオープンソースのブログフレームワーク、そしてコンテンツ管理システムです。開発者のウェブсайт構築へのハードルを大きく引き下げ、ウェブサイト構築の効率を上げることができます。WordPressが提供するデフォルトソリューションを使用すれば、小型のウェブサイトなら数分で実行が可能です。しかし、ネットワークの持続的な運営につれて、画像およびオーディオビデオなどのメディアファイルへのアクセス速度、およびサーバーの帯域幅とストレージ容量の増加により、サイトは次第に負荷に耐えられなくなっていくます。サイトにとって重要な地位を占めるこれらのメディアファイルを、どのようにして適切に扱うのが、サイトを長期運営するにあたり、避けて通れない問題となります。

サイト構築に、なぜVideo on Demand (VOD) を使用すべきなのか

WordPressのデフォルトソリューションを使用して構築したウェブサイトが長期運営に向かない主な理由は、メディアファイルを直接サイトのサーバーに保存すると、以下の問題が発生するためです。

番号	詳細
1	ウェブページから直接サーバー上のメディアファイルにアクセスすると、ビデオラグおよび画像のロードが遅くなる問題がおこりがちです。
2	ウェブサイトからメディアファイルにアクセスすると、サーバーの帯域幅を占めるため、帯域幅がいっぱいになりがちです。
3	メディアファイルのサイズは一般的に大きいため、ウェブサイトのコンテンツが豊富になるにつれ、サーバーのストレージ容量がいっぱいになりやすくなります。
4	サイトのサーバーを移行する際、メディアファイルの移行は困難です。

VODは、オーディオビデオおよび画像などのメディアにストレージを提供します。トランスコーディングおよび配信と再生のアクセラレーション機能などのクラウドサービスでは、ウェブサイトの構築に、以下のサポートを行います。

番号	詳細
1	アップロードおよびストレージです。プラグインあるいはSDKを用いて、ファイルをVODにアップロードすることができます。その後、サーバーのストレージ容量を占有しません。サーバー移行時

	に、ファイルを誤って削除してしまうなどの問題を回避できます。
2	アクセラレーションアクセスです。VODはファイルのCDNアドレスを提供します。オーディオビデオおよび画像へのアクセス時に、ラグ現象を起こしません。サーバーの帯域幅も占有しません。

このほか、VODはアクセス制御、メディア処理およびインテリジェント審査などの機能も提供します。詳細はVODの[機能の概要](#)をご参照ください。

VODのWordPressプラグイン

VODは、[WordPressプラグイン](#)を提供します。このプラグインをインストールし、簡単な設定をするだけでVODサブアプリケーションにバインドし、WordPress上のメディアファイルをVODにホスティングできます。VODが提供するCDNアドレスで、あなたのウェブサイト上のファイルにアクセスできます。現在、プラグインには以下の機能があります。

- VODサブアプリケーションのバインド。
- WordPressの管理端末にビデオファイルをアップロードすれば、自動でVODのサブアプリケーションにアップロードできます。
- 記事あるいはページ上にビデオを挿入する際は、自動でVOD上のビデオアドレスを使用します。

準備作業

1. VODサブアプリケーションの作成

VODサブアプリケーションは、リソース隔離の実装をサポートします。例えば、異なるサイトのメディアファイルをそれぞれ異なるサブアプリケーションに割り当て、単独で管理ができます。作成方法の詳細については[サブアプリケーションのアクティブ化](#)をご参照ください。

2. WordPressウェブサイトの構築

実際の状況に基づいてソリューションを選択し、WordPressウェブサイトを構築します。

- 方法1（推奨）
- 方法2

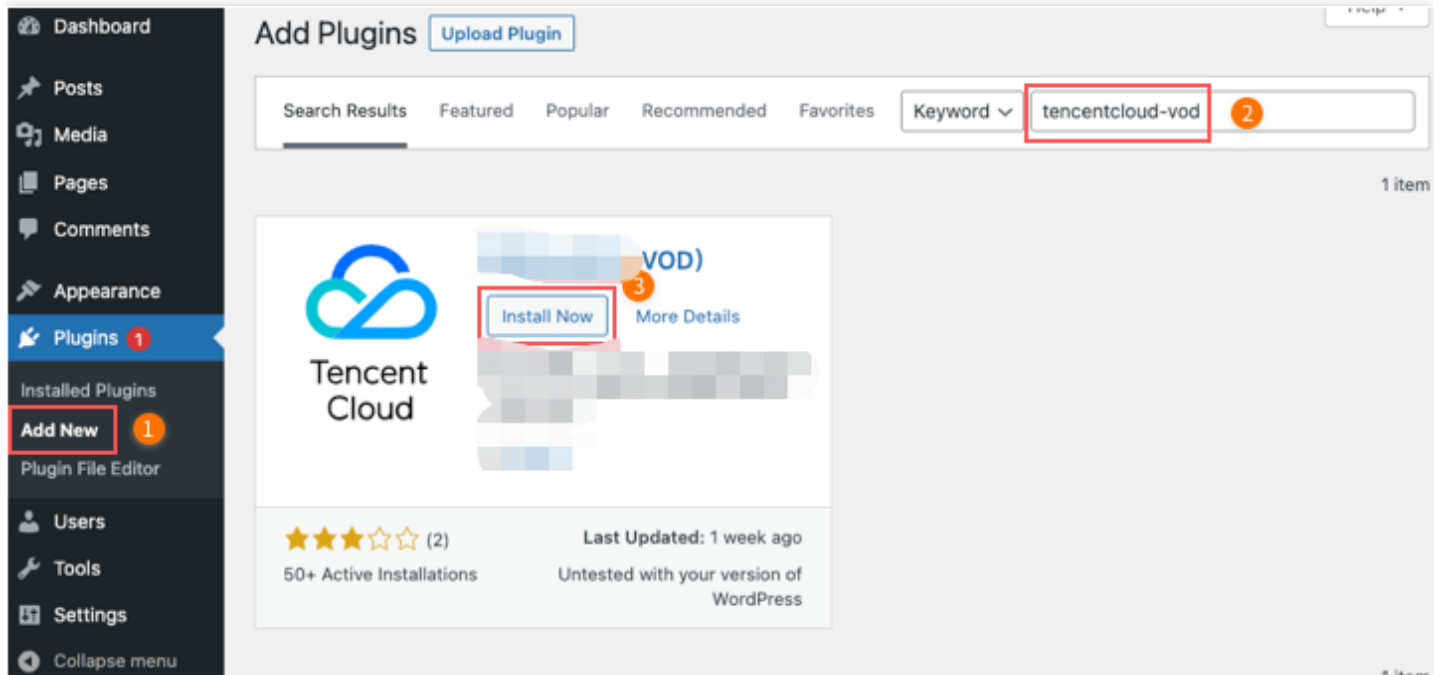
Tencent Cloud LighthouseあるいはCloud Virtual Machine（CVM）のWordPressアプリケーションイメージで迅速にサイトを構築します。

操作手順

ステップ1：プラグインのインストール

- 方式1
- 方式2

オンラインでインストールします（推奨）。検索ボックスで**tencentcloud-vod**を検索し、**今すぐインストール>開始**をクリックします。



ステップ2：プラグインの設定

WordPress左側ナビゲーションバーの****Tencent Cloud設定 > VOD****をクリックすると、VODプラグインの設定ができます。

設定の説明は下表をご参照ください。

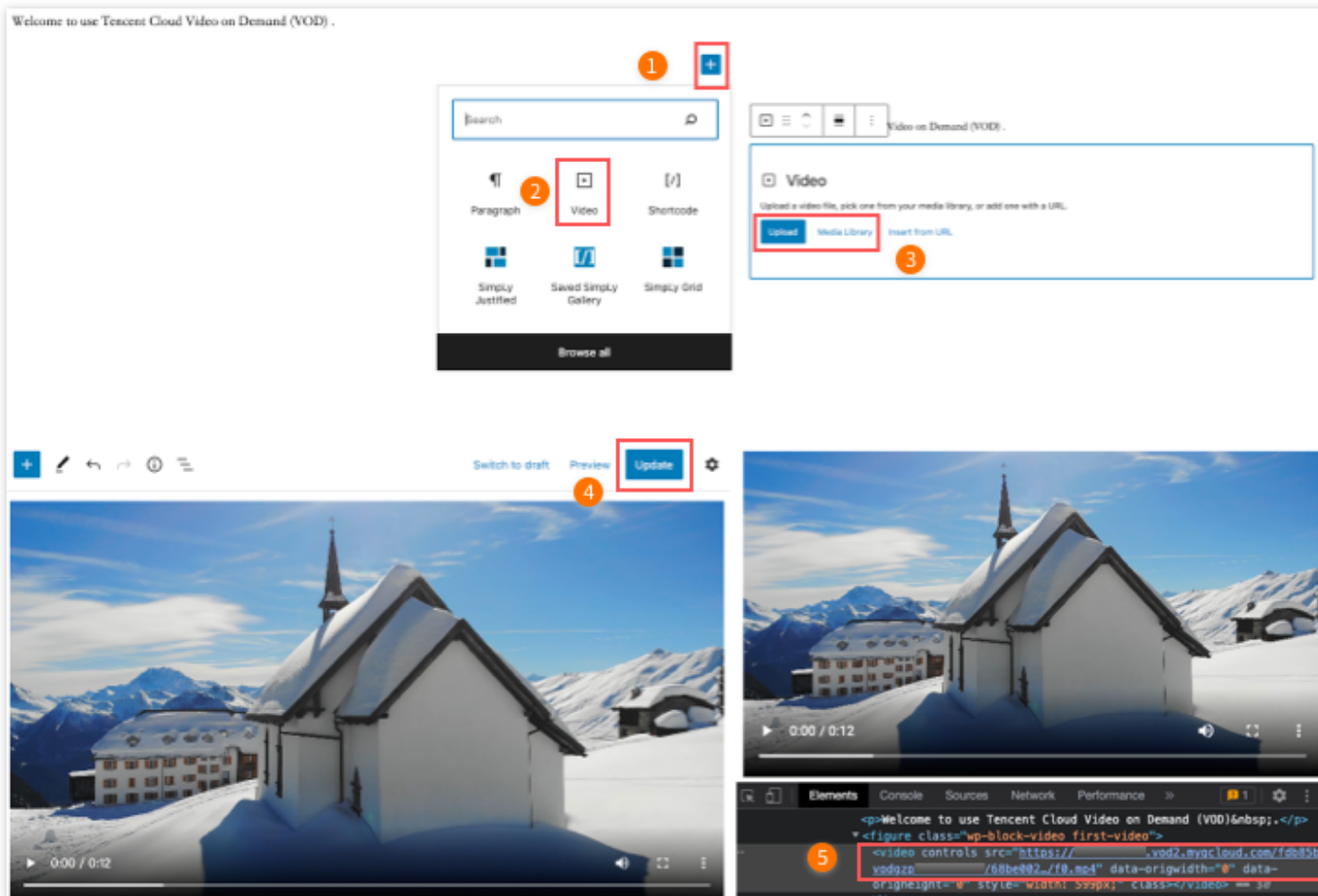
設定項目	説明
カスタマイズキー	このオプションをオンにしない場合、プラグインは Tencent Cloud設定 で設定されたキーを使用します。つまり、その他Tencent Cloudプラグインと同じキーを共用します。オンにする場合は、下記で指定するキーを使用します。
SecretId、SecretKey	アクセスキー情報です。 Tencent Cloud APIキー に進んで作成と取得を行うことができます。
SubAppID	VODのサブアプリケーションIDです。準備中の作成するサブアプリケーションIDを入力すれば完了です。もしくは、VODの アプリケーション管理 からも照会できます。

設定項目	説明
アダプティブビットレートストリーミング変換を開始するかどうか	オンにすると、アップロード時にビデオに対し、アダプティブビットレートストリーミング変換を行います。詳しくはVODの アダプティブビットレートストリーミング をご参照ください。

設定完了後に、**設定を保存**をクリックすれば完了です。

ステップ3：オンデマンドビデオのアップロードと使用

- VODプラグインをオンにして設定を完了した後に、メディア資源データベースおよび記事またはページ編集時にアップロードされたビデオファイルは、自動でVODにアップロードされます。
- 記事編集時のアップロードを例とします。下図に示すとおり、記事に「ビデオブロック」を挿入後、**アップロード**をクリックし、ビデオファイルを選択してアップロードします。次に、記事の右上隅の「プレビュー」をクリックし、記事のプレビューページを確認します。ビデオのURLを確認することで、この時に使用されているファイルが、VOD上のビデオであることがわかります。



- 下図に示すように、左側で使用しているのはサーバー上のビデオで、右側はVODです。再生サーバーのローカルビデオは、帯域幅制限とCDNアクセラレーションサービスが無いために頻繁にラグを起こします。VODを使

用すれば、これらの問題を容易に解決できます。



拡張

HTTPSの使用

VODはデフォルトでHTTPプロトコルを使用します。HTTPSを使用する場合は、VODコンソールにてサブアプリケーションを選択し、その後にシステム設定 > 配信と再生の設定 > デフォルト配信設定を開くと、デフォルト配信プロトコルタイプでHTTPSに変更できます。

カスタムオリジンサーバーback-to-originの方法

最終更新日：：2023-10-26 17:31:32

概要

カスタムオリジンサーバーのback-to-originとは、Video on Demand (VOD) のCDN機能を利用し、カスタムドメイン名を作成してback-to-origin情報を設定することで、ユーザーがサードパーティオリジンサーバーに保存したメディアファイルをVODによってアクセラレーション配信できるようにし、マルチクラウドシナリオにおけるメディア配信ソリューションをご提供するものです。ここではVODのカスタムオリジンサーバーback-to-origin機能の設定および使用方法についてご説明します。

シナリオ

- **オリジンサーバーの移行コストが高い**：ユーザーのメディアが他のサードパーティクラウド上に保存され、そのオリジンサーバーのコンテンツがサードパーティプラットフォームと高度に結びついているためにオリジンサーバーの移行が困難な場合、カスタムback-to-originによって、オリジンサーバーを移行せずにVOD CDNを使用してコンテンツのアクセラレーション配信を行うことができます。
- **高い再生品質が求められる**：ユーザーのコンテンツがネットワーク遅延やラグに対する要件が厳しく、他のサードパーティプラットフォームではニーズを満たせない場合、カスタムback-to-originによってお客様のオーディオビデオ再生をよりスムーズにし、業務サービス品質を保証することができます。
- **複数のクラウドの共存によりリスクを低減**：ユーザーのコンテンツおよび業務が、その信頼性を保障し、業務の障害復旧機能を向上させるために複数のコンテンツアクセラレーションチャンネルを必要とするものである場合は、Tencent Cloud VOD CDNを選択してオーディオビデオアクセラレーション配信を行うことができます。

前提条件

1. Tencent Cloudアカウントを[登録](#)して[ログイン](#)し、アカウントの実名認証を完了していること。
2. Tencent Cloud VODサービスがアクティブ化されました。まだアクティブ化されていない場合は、[VODサービス](#)に移動してアクティブ化してください。

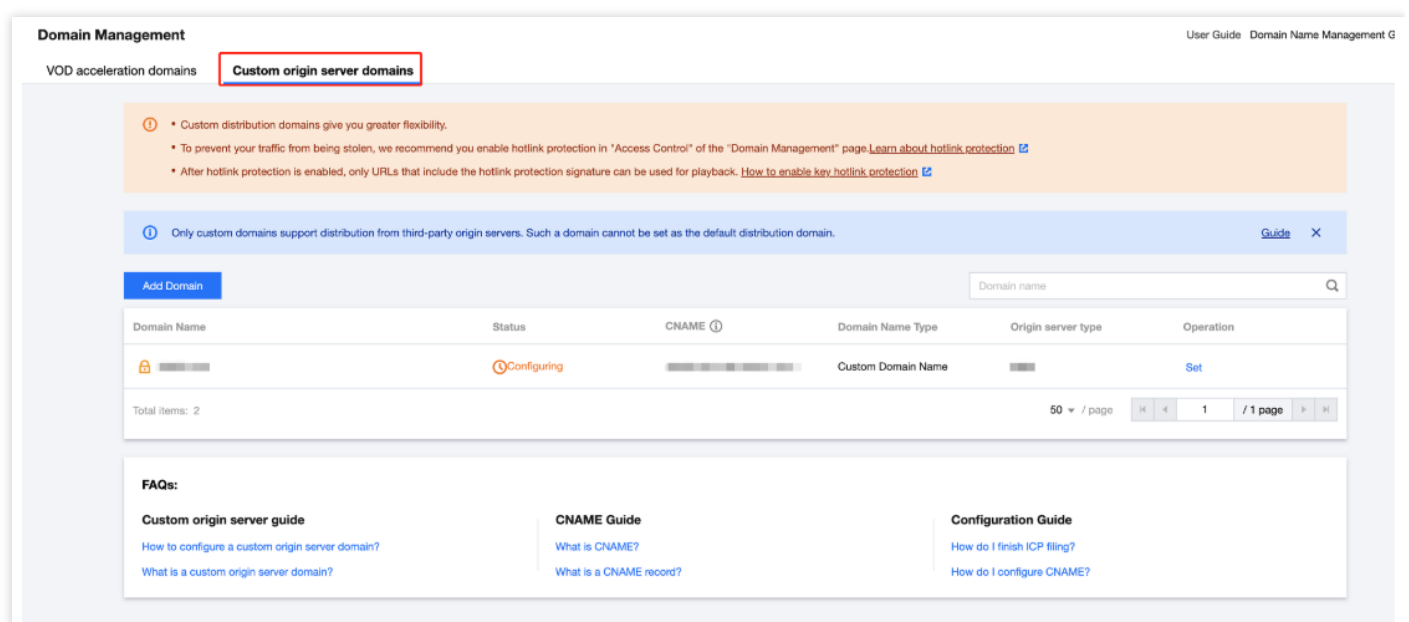
サポートするback-to-originオリジンサーバータイプ

- ユーザー保有オリジンサーバー
- サードパーティCOS

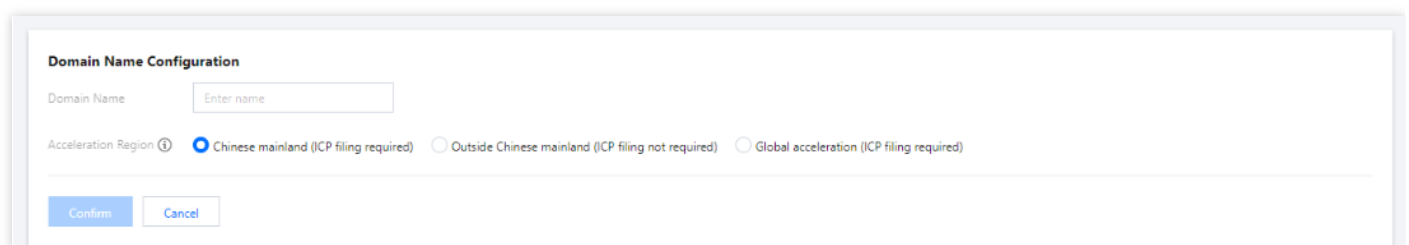
実践手順

ステップ1：カスタムドメイン名の作成とback-to-origin情報の設定

1. カスタムオリジンサーバーback-to-origin機能を有効化した後、VODコンソールにログインし、左側ナビゲーションバーで**配信と再生の設定**>**ドメイン名管理**を選択し、**ドメイン名管理**ページに進みます。
2. 上の**カスタムオリジンサーバーアクセラレーションドメイン名**をクリックし、**カスタムオリジンサーバーアクセラレーションドメイン名**のページに進みます。



3. **ドメイン名の追加**をクリックし、ICP登録済みのドメイン名を入力し、**アクセラレーションリージョン**を選択します。具体的な操作については、**VODアクセラレーションドメイン名**をご参照ください。



4. ユーザーの実際のback-to-originニーズに応じてオリジンサーバーの設定を入力します。現在は**ユーザー保有オリジンサーバー**と**サードパーティCOS**のback-to-origin方式をサポートしています。具体的な設定についての説

明は次のとおりです。

Origin server (i)

Origin server type Self-owned server Third-party storage

Protocol HTTP HTTPS Same as request (i)

Origin server address

Host (i)

ユーザー保有オリジンサーバー

安定して稼働する業務用サーバーをback-to-originオリジンサーバーとし、VOD CDNによってサーバー上のメディアファイルをアクセラレーション配信したい場合は、以下の方法でオリジンサーバーを設定してください。

設定項目	説明
オリジンサーバータイプ	ユーザー保有オリジンサーバーを選択します
back-to-originプロトコル	オリジンサーバーのサポート状況に応じてback-to-originリクエストプロトコルを選択します。HTTP、HTTPSおよびプロトコル追従をサポートしています
オリジンサーバーアドレス	複数のIPオリジンサーバー（コンマで区切る）または1つのドメイン名オリジンサーバーを入力できます
back-to-origin HOST	ユーザー保有オリジンサーバーではback-to-origin HOSTを指定できます。back-to-origin HOSTは、CDNノードのback-to-originの際に、オリジンサーバーでアクセスするサイトのドメイン名/IPの具体的なサイトを指定するために用いられます back-to-origin HOSTを指定していない場合は、デフォルトで現在作成されているアクセラレーションドメイン名となります

説明

- オリジンサーバーアドレスにはVODデフォルトドメイン名は使用できません。
- プロトコル追従では、HTTPアクセスはHTTP back-to-originを使用し、HTTPSアクセスはHTTPS back-to-originを使用することができます（オリジンサーバーがHTTPSアクセスをサポートしていることが必要です）。
- ドメイン名をback-to-originアドレスとして選択できます。このドメイン名は、業務用のアクセラレーションドメイン名と同じものにすることはできません。

サードパーティCOS

サードパーティCOSに保存されたメディアファイルを、VOD CDN機能によってアクセラレーション配信したい場合は、以下の方法でオリジンサーバーを設定してください。

設定項目	説明
オリジンサーバータイプ	サードパーティCOSを選択します
サードパーティCOS	現在サポートしているサードパーティCOSはAlibaba Cloud OSSとAWS S3です
back-to-originプロトコル	HTTP、HTTPSをサポートしています
オリジンサーバーアドレス	有効なバケットアクセスアドレスをオリジンサーバーとして入力します（http://またはhttp://プロトコルヘッダーを含めることはできません）

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 ⓘ

プライベートアクセスのサードパーティCOSをオリジンサーバーとして選択する場合は、有効なアクセスIDおよびkeyを入力してback-to-origin認証を行う必要があります。認証に成功するとプライベートバケットへのアクセスが有効になります。

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

ステップ2：カスタムドメイン名の解決

追加済みのカスタムアクセラレーションドメイン名については、このドメイン名が指定するDNSサービスプロバイダでCNAMEを設定しなければ、ユーザーはアクセラレーションドメイン名によるメディアファイルへのアクセスできません。具体的な操作については、[VODアクセラレーションドメイン名-ドメイン名の解決](#)をご参照ください。

ステップ3：カスタムオリジンサーバーback-to-originの設定結果の確認と変更

1. ドメイン名管理に進み、**カスタムオリジンサーバーアクセラレーションドメイン名**で、作成済みのドメイン名を選択して**設定**をクリックします。

Domain Management User Guide Domain Name Management

VOD acceleration domains **Custom origin server domains**

• Custom distribution domains give you greater flexibility.
• To prevent your traffic from being stolen, we recommend you enable hotlink protection in "Access Control" of the "Domain Management" page. [Learn about hotlink protection](#)
• After hotlink protection is enabled, only URLs that include the hotlink protection signature can be used for playback. [How to enable key hotlink protection](#)

Only custom domains support distribution from third-party origin servers. Such a domain cannot be set as the default distribution domain. [Guide](#) X

Add Domain Domain name

Domain Name	Status	CNAME	Domain Name Type	Origin server type	Operation
com	Configuring		Custom Domain Name		Set

Total items: 2 50 / page 1 / 1 page

FAQs:

- Custom origin server guide**
[How to configure a custom origin server domain?](#)
[What is a custom origin server domain?](#)
- CNAME Guide**
[What is CNAME?](#)
[What is a CNAME record?](#)
- Configuration Guide**
[How do I finish ICP filing?](#)
[How do I configure CNAME?](#)

2. **基本設定**をクリックすると、現在のカスタムドメイン名のback-to-origin設定情報を確認して変更できます。

shishi.com User Guide Domain Name Management

Basic Configuration Access Control

• ICP filing is required if a domain is used to distribute content in the Chinese mainland. [Applying for an ICP filing](#)
• By default, VOD distributes content stored in VOD. If you use a third-party origin server, you may be charged additional fees by your vendor. [Custom origin server domains](#)
• After you add a domain or modify the settings of a domain, it takes about 5-10 minutes for the domain or new settings to take effect. You can view the status, CNAME record, and type of a domain you added in the domain list.
• Each account can add at most 20 domains

Basic Info

Domain Name:

CNAME:

Creation Time: 2023-01-12 18:17:08

Acceleration Region: **Outside Chinese mainland** [Modify](#)

Origin server **Modify**

Origin server type:

Protocol:

Origin server address:

Host: -

Private bucket access:

上記の手順によって操作することで、ユーザーはユーザー保有オリジンサーバーまたはサードパーティCOSに基づくback-to-origin設定を完了し、VOD CDNによってカスタムオリジンサーバー上のメディアファイルを配信できるようになります。具体的な配信のフローについての説明は次のとおりです。

1. ユーザーがカスタムドメイン名（例：`test.com`）を追加してドメイン名解決とback-to-origin設定を完了した後、ブラウザからこのドメイン名下のメディアファイルにアクセスする場合

- (例: `http://www.test.com/test.mp4`)、まずLocal DNSに対しドメイン名解決リクエストを送信します。
- Local DNSがドメイン名 `test.com` を解決する際、すでにCNAME: `www.test.com.cdn.dnsv1.com` が設定されていることを発見し、リクエストをTencent GSLB (Global Server Load Balance。Tencentが自社開発したグローバルロードバランサシステム) に解決します。
 - GSLBは最適なアクセスノードのIPリストを返し、Local DNSはこれに応じた解決IPを取得します。
 - ユーザーは最適なアクセスIPノードを取得します。
 - ユーザーは最適なアクセスIPノードに、`http://www.test.com/test.mp4` に対するアクセスリクエストを送信します。
 - このCDNノード上に`test.mp4`がキャッシュされている場合、ユーザーはデータを取得し、このリクエストを終了します。このCDNノードにこのリソースがキャッシュされていない場合、このノードは設定された**業務オリジンサーバー**に、`test.mp4`に対するリクエストを送信し、リソースを取得した後、デフォルトで現在のノードにキャッシュし、リソースをユーザーに返します。リクエストはこの時点で終了します。

説明:

ユーザーがカスタムオリジンサーバーback-to-origin機能を使用してメディアファイルのアクセラレーション配信および再生を行うと、**ダウンストリームトラフィック料金**と**back-to-originオリジンサーバートラフィック料金**が発生します。このうちダウンストリームトラフィック料金はVODによって課金されます。具体的なルールについては[トラフィック課金](#)および[トラフィックリソースパック](#)をご参照ください。back-to-originオリジンサーバーのトラフィック料金はオリジンサーバーによって課金されます。