

Cloud Streaming Services

機能の実践

製品ドキュメント



Tencent Cloud

Copyright Notice

©2013-2023 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：

機能の実践

Push and Playback

CSSプッシュ

CSS再生

自身でのCSS URLの接合

SRTプロトコルのプッシュ

遅延再生

ライブストリーミング機能

CSSパッケージ化とトランスコーディング

CSSレコーディング

CSSレコーディング

ライブストリーミングのタイムシフト

CSSスクリーンキャプチャ

CSSポルノ検出

ライブストリーミングAV1コーデックの実現

クラウドミクスストリーミング

Live Streaming Security

ビデオライブストリーミング再生のセキュリティ

ホットリンク防止の計算

海外のCSSサービス

海外のCSSサービス紹介

HttpDNSスケジューリング

コールバックによるイベントメッセージ通知

イベント通知の受信方法

プッシュ切断イベント通知

レコーディングイベント通知

スクリーンキャプチャイベント通知

ポルノ検出イベント通知

プッシュ異常イベント通知

プルリツイートイベント通知

一般的なサードパーティ製ツールガイド

OBSプッシュ

VCLプッシュ

機能の実践

Push and Playback

CSSプッシュ

最終更新日：2022-06-10 16:07:14

CSSのサービスは本質的に放送プロセスの一環であり、テレビ局のライブストリーミングプログラムがケーブルネットワークを介して何百万もの世帯に送信されるといった行程と類似しています。このプロセスを完成するには、CSSには収集とプッシュのためのデバイス（カメラに類似）、CSSサービス（テレビ局のケーブルネットワークに類似）および再生デバイス（テレビに類似）が必要です。収集とプッシュのデバイスおよび再生デバイスをスマートフォン、PC、Padなどのスマート端末およびWebブラウザとすることができ、また、弊社は対応するデバイスのプッシュソフトウェアのDemoも提供しています。

準備作業

1. [Tencent CSSサービス](#)をアクティブ化します。
2. [【Domain Management】](#)を選択して、[【ドメイン名の追加】](#)を選択し、ICP登録したプッシュドメイン名を追加します。詳細については、[独自のドメイン名の追加](#)をご参照ください。

説明：

CSSはデフォルトのプッシュドメイン名を提供します。形式は `xxx.livepush.myqcloud.com` ですが、実際の業務でこのドメイン名をプッシュドメイン名として使用することはお勧めしません。

プッシュアドレスの取得

CSSコンソールの[【CSSツールボックス】](#) > [【アドレスジェネレーター】](#)に移動し、プッシュアドレスを発行します。設定は次のとおりです。

- 発行タイプの選択：[プッシュドメイン名](#)。
- ドメイン名管理で追加したプッシュドメイン名を選択します。
- 同一ドメイン名の複数Appを区別するために使用されるアドレスパスであるAppNameを入力します。デフォルト値はliveです。
- カスタマイズされたストリーム名StreamNameを入力します（例：`liveteststream`）。
- アドレス期限を選択します（例：`2019-10-18 23:59:59`）。

- 【アドレスの発行】を選択します。

注意：

- ライブストリーミングのセキュリティを保護するために、システムが自動的にプッシュ認証を有効にします。【Domain Management】で変更するプッシュドメイン名を選択し、右側の【管理】を選択して、ドメイン名詳細情報ページの【プッシュ設定】に移動し、認証設定情報をカスタマイズすることもできます。プッシュアドレスの形式は次のとおりです。

```
rtmp://domain/AppName/StreamName?
```

```
txSecret=Md5(key+StreamName+hex(time))&txTime=hex(time)
```

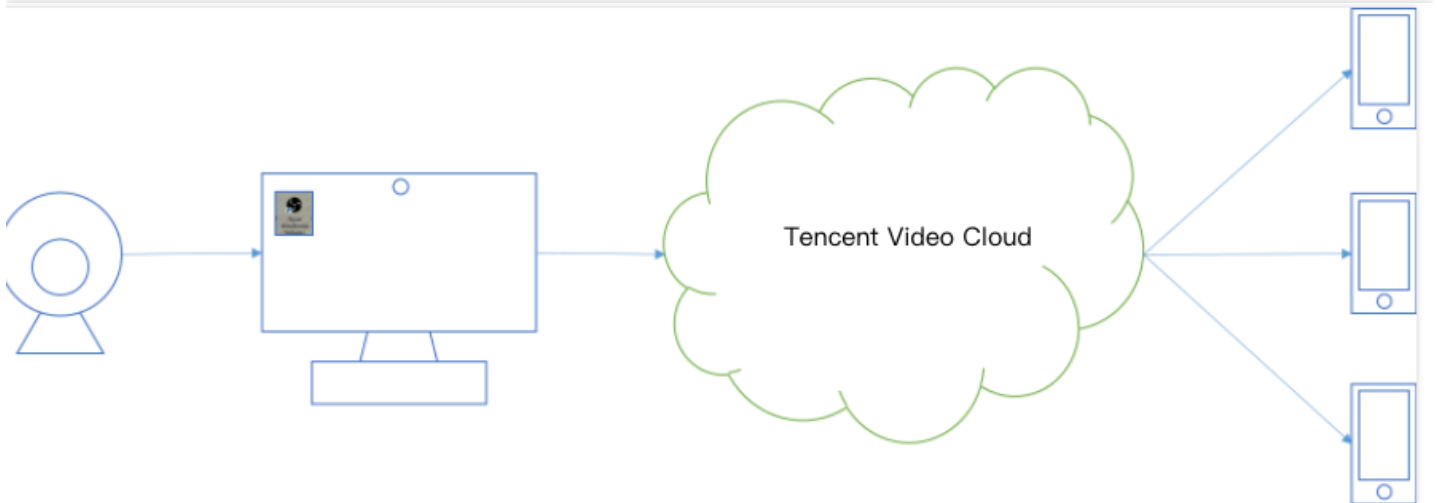
- 上記方法以外にも、CSSコンソールの【Domain Management】でプッシュドメイン名を選択して、【管理】を選択し、【プッシュ設定】を選択して、プッシュアドレスの期限切れ時間とカスタマイズされたストリーム名StreamNameを入力し、【プッシュアドレスの発行】を選択すると、プッシュアドレスを発行できます。
- 長期的に有効なプッシュアドレスが必要な場合は、【Domain Management】に移動し、プッシュドメイン名を選択したのち【管理】を選択。【プッシュ設定】をまた選択後、【プッシュアドレスサンプルコード】にありますサンプルコードから発行します。具体的なクエリー方式については、[プッシュサンプルコードのクエリー方法](#)をご参照ください。

CSSプッシュ

業務のシナリオに応じて次の方式でCSSプッシュを実現できます。

シーン1：PCプッシュ

PC (Windows/Mac) でプッシュする場合は、実際の状況に応じて [OBS](#) または [XSplit](#) のインストールを選択して、プッシュすることができます。OBSはWindows、Mac、Linuxなどのシステムをサポートする無償オープンソースのビデオレコーディングおよびビデオリアルタイムストリームソフトウェアです。XSplitの使用は有償ですが、XSplitにはゲームライブストリーミング用の個別のインストールパッケージがあります。ゲームライブストリーミングでない場合は、BroadCasterの使用をお勧めします。

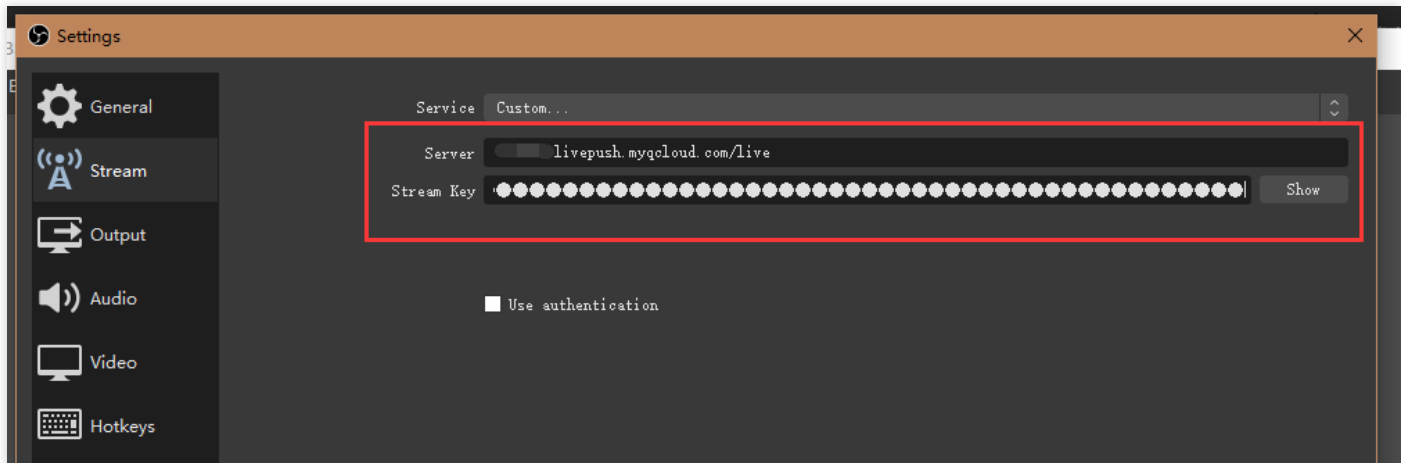


ここではOBSプッシュのインストールを例示します。操作ステップは次の説明のとおりです。準備の完了したプッシュアドレスが次のとおりであると仮定します。

```
rtmp://3891.livepush.myqcloud.com/live/3891_test?bizid=3891&txSecret=xxx&txTime=58540F7F
```

1. [OBS公式サイト](#) に移動し、プッシュツールをインストールします。
2. OBSを開き、下部ツールバーの【コントロール】>【設定】を選択し、ボタンを押して設定インターフェースに移動します。
3. 【プッシュ】を選択し、プッシュ設定ページに移動して、次の設定を行います。
4. サービスタイプ：カスタムを選択します。
5. サーバーに `rtmp://3891.livepush.myqcloud.com/live/` のようにプッシュアドレスの前半部分を入力します。
6. ストリームキーに `3891_test?bizid=3891&txSecret=xxx&txTime=58540F7F` のようにプッシュアドレスの後半部分を入力します。

7. 右下隅の【OK】を選択します。



8. ツールバーの【コントロール】>【プッシュ開始】を選択すれば、プッシュテストを実行できます。OBS操作の詳細については、[OBSプッシュ](#)をご参照ください。

シーン2：Webプッシュ

1. CSSコンソールにログインします。
2. 【支援ツール】>【Webプッシュ】を選択します。
3. Webプッシュのページで次の設定を行います。
4. プッシュドメイン名を選択します。
5. 同一ドメイン名の複数Appを区別するために使用されるアドレスパスのAppNameを入力します。デフォルト値はliveです。
6. カスタマイズされたStreamNameを入力します（例：liveteststream）。
7. 期限を選択します（例： 2019-10-30 23:59:59 ）。
8. 【プッシュ開始】を選択し、カメラの承認を完了すると、プッシュを開始できます

注意：

Webプッシュ機能は、デバイスにカメラがインストールされている必要があり、かつブラウザがカメラ許可を呼び出すFlashプラグインをサポートする必要があります。

Select push domain *

Select a push domain. If there is no available domain name, please [Add Domain](#)


AppName *

Use "live" by default. Only letters, digits, and symbols are supported.

StreamName *

Only support letters, digits, and symbols.

Expiration Time

Push address expiration time is the setting time

[Start Push](#)[Stop Push](#)

シーン3：モバイルプッシュ

1. スマートフォンで2次元コードをスキャンし、モバイルビデオクラウドツールキットをインストールすることで体験できます。
2. ツールキットを開き、【モバイルライブストリーミング】>【カメラプッシュ】を選択します。
3. [プッシュアドレス](#)を手動入力するか、または2次元コードをスキャンして入力します。
4. 左下隅のスタートボタンを選択すれば、プッシュを開始することができます。

説明：

事前にプッシュアドレスを準備していない場合は、カメラプッシュページでプッシュアドレス右側の【NEW】を選択すれば、システムが自動的にプッシュアドレスを入力し、対応する再生アドレスが提供されます。この再生アドレスを介してCSSプッシュ機能を確認できます。

シーン4：ライブストリーミングSDKプッシュ

CSSプッシュ機能を既存のAppに統合するのみの使用は、以下の手順に従うことで利用可能です。

1. モバイルライブストリーミングSDK 開発パッケージをダウンロードします。
2. ドッキングドキュメント（iOS & Androidを参照してアクセスを完了します。

ライブストリーミングSDKはモバイル端末ライブストリーミングソリューションの集合であり、無償のソースコードの形式で表示されます。CSS、VOD、IM、COS等のサービスを組み合わせて利用し、最適なライブストリーミングソリューションを構築します。

よくあるご質問

- [CSS再生はどのように実現するのですか。](#)
- [プッシュURLはどのようにスプライスするのですか。](#)
- [ホットリンク防止はどのように計算するのですか。](#)

CSS再生

最終更新日：2022-08-03 16:25:42

準備作業

1. [Tencent CSSサービス](#)をアクティブ化し。
2. [CSSコンソール](#)にアクセスし、プッシュアドレスを取得して、CSSプッシュを実現します。具体的な操作については、[CSSプッシュ](#)をご参照ください。
3. [Domain Management](#)を選択して、【ドメイン名の追加】を選択し、ICP登録済みのドメイン名を入力して、【再生ドメイン名】タイプを選択し、【保存】をクリックしてください。
4. [ドメイン名サービスコンソール](#)にログインし、追加済みの再生ドメイン名にCNAMEの設定を行います。具体的な操作については、[ドメイン名CNAME設定](#)をご参照ください。

再生アドレスの取得

CSSコンソールの【CSSツールボックス】>【アドレスジェネレーター】に移動して、再生アドレスを取得し、このページで次の設定を行います。

- 生成タイプ：[再生ドメイン名](#)を選択します。
- ドメイン名管理に追加済みの再生ドメイン名を選択します。
- プッシュアドレスと同一のStreamNameを入力します。対応するストリームを再生するためには、再生アドレスのStreamNameはプッシュアドレスのStreamNameと一致させる必要があります。
- 選択アドレスの期限を選択します（例：2019-12-13 23:59:59）。
- 【アドレスの発行】を選択します。

The screenshot shows the 'Address Generator' interface with the following fields and values:

- Domain Type:** Playback domain (selected). A dropdown menu shows a list of domain names.
- AppName:** live
- StreamName:** liveteststream
- Expiration Time:** 2020-05-11 23:59:59 (with a calendar icon)

Below the fields, there are two buttons: 'Generate Address' and 'Address Resolution Sample'.

説明：

上記方法以外にも、CSSコンソールの【Domain Management】で再生ドメイン名を選択して、【管理】をクリックし、【再生設定】を選択し、再生アドレスの期限を選択して、プッシュアドレスと同一のStreamNameを入力し、再生アドレスの発行】を選択するとアドレスの発行が可能です。

CSS再生

始めにCSSプッシュを実施する必要があり、プッシュが正常に実行された後、再生アドレスを介してライブストリーミング画面を確認することができます。業務のシナリオに基づき、次の方式でライブストリーミングテストを実施することができます。

シーン1：PCでの再生

VLC、FFmpeg および TCPlayerDemo などのツールを使用して再生することができます。

シーン2：モバイルでの再生

1. Tencent Video Cloud Demo をダウンロードし、インストールします。
2. 【モバイルライブストリーミング】>【CSSプル】を開き、選択します。
3. 入力ボックスに再生アドレスを入力するか、または再生アドレスの2次元コードをスキャンして入力します。
4. 左下隅の再生ボタンをクリックし、再生して視聴します。

説明：

Appでプッシュ/再生を実施する場合はモバイルライブストリーミングSDKを統合し、CSSサービスと連携して使用することができます。モバイルライブストリーミングSDKはRTMP、HTTP-FLV、HLS再生プロトコルをサポートしています。

シーン3：Webでの再生

Tencent Cloudの強力なバックエンド機能とAI技術に基づき、CSSとVODの強力な再生機能を提供する、プレーヤーSDKのTCPlayerを選択して再生することをお勧めします。Player+は、Tencent Video CSS、VODサービスと深く融合し、スムーズで安定した再生パフォーマンスを実現し、広告の配置、データモニタリングなどの機能を包括しています。

注意：

現在、市販で販売されている大多数のスマホブラウザはHTTP-FLV再生をサポートしていないため、Tencent Cloudは、Web再生時のプロトコル選択に際し、PCブラウザでは、HTTP-FLVプロトコルを使用

してのライブストリーミング再生、スマホブラウザではHLSを使用してのCSSストリーム再生を推奨しています。

よくあるご質問

- どのような再生プロトコルがサポートされているのですか。
- 再生アドレスはどのような構成ですか。
- 再生トランスコードはどのように使用するのですか。
- タイムシフト視聴はどのように使用するのですか。
- HTTPS再生はどのように使用するのですか。
- 海外アクセラレーションノードをどのように再生に使用するのですか。
- 再生ホットリンク保護はどのようにアクティブ化するのですか。

自身でのCSS URLの接合

最終更新日：：2022-01-24 14:18:39

注意事項

トランスコードテンプレートの作成を行い、再生ドメイン名バインドを行ってから、トランスコーディング設定後のLVBストリームは、再生アドレスのStreamNameを `StreamName_トランスコードテンプレート名` に結合する必要があります。詳細は再生設定をご参照ください。

前提条件

- Tencent Cloudアカウントを登録済みで、Tencent LVBサービスを有効にしていること。
- 独自のドメイン名があること。
- 【LVBコンソール】>【Domain Management】で、プッシュ/再生ドメイン名の追加、およびCNAMEに成功していること。操作の詳細は独自のドメイン名の追加をご参照ください。

プッシュURLのsprays

実際のサービスを使用中で、ライブストリーミンググループが多い場合、キャスターごとに手動でプッシュと再生のURLを作成することはできませんが、サーバーでプッシュと再生のアドレスをspraysすることができます。Tencent Cloud標準仕様に準拠するURLであれば、プッシュに使用でき、4つの部分で構成される標準的なプッシュURLは次のとおりです。

`rtmp://domain/AppName/StreamName?txSecret=Md5(key+StreamName+hex(time))&txTime=hex(time)`

Domain name Application name Stream ID Authentication key

- **Domain**
プッシュドメイン名には、Tencent Cloud LVBの提供するデフォルトのプッシュドメインあるいは自ら登録してCNAMEが正常に設定された独自プッシュドメイン名を使用します。
- **AppName**
LVBアプリケーション名のデフォルトはliveですが、カスタマイズすることも可能です。

- **StreamName (ストリームID)**

カスタムストリーム名、各ライブストリーミングのストリームの一意IDには、ランダムな数字または数字とアルファベットを組み合わせたの使用をお勧めします。

- **認証Key (オプション)**

txSecretとtxTimeの両部分を含みます。

す：`txSecret=Md5(key+StreamName+hex(time))&txTime=hex(time)`。

プッシュ認証が有効な場合は、プッシュに使用されるURLには認証Keyが含まれている必要があります。プッシュ認証が有効でない場合は、プッシュアドレスに「?」とそれに続く内容を含める必要はありません。

- **txTime (アドレスの有効期限)**

これはURLの有効期限を意味し、UNIXの16進数のタイムスタンプをサポートしています。

説明：

例えば、`5867D600` は、2017年1月1日0:00:00に期限が切れることを意味し、クライアントは、通常、txTimeは現在の時間から24時間後に期限切れとなるように設定されます。キャスターはライブストリーミング中にネットワークが途切れた場合にプッシュを再開できるように、期限切れまでの時間は長すぎず、また短すぎないように設定する必要があります。期限切れまでの時間が短すぎると、プッシュURL期限によって切れてしまうため、プッシュを再開することができません。

- **txSecret (ホットリンク防止署名)**

攻撃者がバックグラウンドを偽造してプッシュURLを生成することを防止するには、[ベストプラクティス-ホットリンク防止の計算](#)をご参照ください。

再生URL のsprays

次に例示するとおり、再生アドレスは、主に再生プレフィックス、再生ドメイン名 (domain)、アプリケーション名 (AppName)、ストリーム名 (StreamName)、再生プロトコルサフィックス、認証パラメータ、およびその他のカスタムパラメータで構成されます。

```
webrtc://domain/AppName/StreamName?txSecret=Md5(key+StreamName+hex(time))&txTime=hex(time)
http://domain/AppName/StreamName.flv?txSecret=Md5(key+StreamName+hex(time))&txTime=hex(time)
rtmp://domain/AppName/StreamName?txSecret=Md5(key+StreamName+hex(time))&txTime=hex(time)
http://domain/AppName/StreamName.m3u8?txSecret=Md5(key+StreamName+hex(time))&txTime=hex(time)
```

- **再生プレフィックス**

| 再生プロトコル | 再生プレフィックス | 備考 |
|-----------|--|---|
| RTMP | <code>rtmp://</code> | 非推奨。インスタントブロードキャストのパフォーマンス不良。高度な同時実行性をサポートせず |
| HTTP-FLV | <code>http://</code> または <code>https://</code> | 推奨。インスタントブロードキャストのパフォーマンス良好。極めて高度な同時実行性をサポート。 |
| HLS(m3u8) | <code>http://</code> または <code>https://</code> | スマートフォンとMac safariブラウザに推奨される再生プロトコル。 |

- **Domain**

再生ドメイン名。自ら登録し、CNAMEが正常に設定された独自の再生ドメイン名。

- **AppName**

ライブストリーミングのアプリケーション名であり、ライブストリームメディアファイルの格納パスを参照するために使用されます。デフォルトはliveで、カスタマイズできます。

- **StreamName (ストリーム名)**

カスタムストリーム名、各ライブストリーミングのストリームの一意IDには、ランダムな数字または数字とアルファベットの組み合わせを使用することをお勧めします。

- **認証パラメータ (オプション)**

txSecretとtxTimeの両部分を含みます

す：`txSecret=Md5(key+StreamName+hex(time))&txTime=hex(time)`。

再生認証が有効な場合は、再生に使用されるURLには認証Keyが含まれている必要があります。再生認証が有効ではない場合、再生アドレスに「?」とそれに続く内容を含める必要はありません。

- **txTime (アドレスの有効期限)**：このURLの有効期限が切れる時期を意味し、フォーマットはUNIXの16進数のタイムスタンプをサポートしています。
- **txSecret (ホットリンク防止署名)**：攻撃者がバックグラウンドを偽造して再生URLを生成することを防止するために使用します。計算方法は[ベストプラクティス-ホットリンク防止の計算](#)をご参照ください。

プッシュサンプルコードの表示

【LVBコンソール】>【Domain Management】で、事前に設定されたプッシュドメイン名を選択すると、【管理】>【プッシュ設定】ページの下半分にホットリンク防止アドレスの生成方法を示す【プッシュアドレスサンプルコード】(PHPとJavaの両バージョン)が表示されます。操作についての詳細は[プッシュ設定](#)をご参照ください。

SRTプロトコルのプッシュ

最終更新日：2021-07-21 19:35:42

TS over SRTプッシュは、**SRTプロトコル**を介してオーディオビデオデータを含むtsストリームを直接伝送し、ダウンストリームは既存のライブストリーミングを多重化しています。TS over SRTは、HaivisionのハードウェアおよびOBSのプッシュ形式規格に準拠しています。

このモードでは、SRTサーバーが負荷（TS）を解析し、RTMPプロトコルにカプセル化して、バックエンドのRTMPサーバーにプッシュします。

注意：

アップストリームにSRTプッシュプロトコルを選択しても、コストは増加しません。

アップストリームのラグ率比較

SRTプッシュを使用すると、次の品質比較図に示すように、ラグ率の改善がはっきりと示されます。

プッシュのパケット損失率比較

ダウンストリーム側では、SRTプッシュの適用後、アップストリームの品質の最適化により、ダウンストリームのスムーズさも向上します。以下は、闘魚Appでの実際の比較結果です。

- AndroidプラットフォームのSRTプッシュのパフォーマンステストデータ（テストプラットフォーム—MI9）：
- iOSプラットフォームのSRTプッシュのパフォーマンステストデータ（テストプラットフォーム—iphone XR）：

アンチパケット損失比較

伝送品質の指標では、QUICと比較しました。SRTは、より正確で高速な再送制御とCSSストリームメディアセッション用のPacingメカニズムにより、同じパケット損失率でもアプリケーション層のパケット損失を低減します。パケット損失率が50%の場合でも、SRTはQUICよりも安定した伝送を保証します。

QUICのアップストリームと比較すると、プッシュ端末の同一リンク、同一ライブストリーミングファイルの場合、5分ごとにパケット損失率が5%増加しました。SRTのプッシュのほうがフレームレートが安定していること

が、以下の図からわかります。

CSSプッシュ

アクセス方法

CSSプッシュはSRTプロトコルをサポートしますが、**9000ポート**を使用してプッシュする必要があります。プッシュアドレスはCSSコンソールの【[アドレスジェネレーター](#)】の中の [プッシュアドレスの生成](#) で生成し、以下のルールで接続することができます。

Tencent Cloud SRTプッシュURL：

```
srt://${rtmp-push-domain}:9000?streamid=#!::h=${rtmp-push-domain},r=${app}/${stream},txSecret=${txSecret},txTime=${txTime}
```

注意：

`${app}` は表示内容が変更可能であることを示しており、実際には `$`、`{`、`}` の3つの記号を入力する必要はありません。

実現方法

SRTサーバーはTSをRTMPにカプセル化して、`${rtmp-push-domain}`ドメイン名 にプッシュします。

OBSプッシュコードの入力事例：

注意：

SRTプロトコルを使用してプッシュする場合、OBSバージョンを25.0未満にすることはできません。

ライブストリーミングをプルする

正常なプル再生のフローに従って操作してください。具体的には、[CSS再生](#)をご参照ください。

遅延再生

最終更新日：：2022-05-07 11:29:09

遅延再生は、ユーザー側にプルする際の再生を遅延させる機能です。ユースケースは主に重要なライブストリーミングイベントに対してのもので、ライブストリーミング中に突発的な状況が発生することを回避し、対応する処理を事前にコントロールします。また、直接パラメータで設定できます。

注意事項

遅延再生は現在2つの方法で実現できます。

- [遅延再生インターフェース](#)を呼び出して遅延再生機能を実現します。
- [プッシュアドレス](#)の後ろにtxDelayTimeパラメータを追加すると、遅延再生機能をすぐに実現できます。詳細は、[プッシュ設定](#)をご参照ください。

説明：

インターフェースを呼び出すと、設定キャッシュが関連し、有効化時間の制御が難しいため、現在インターフェースを使用する方法は推奨していません。すぐに実現するには、プッシュアドレスの後ろにパラメータを直接追加する方法をお勧めします。

準備作業

1. [Tencent CSSサービス](#)を有効にします。
2. CSSコンソールにログインし、[【ドメイン名管理】](#)を選択して、[【ドメイン名の追加】](#)をクリックし、ICP登録したプルストリーミングドメイン名を追加します。詳細については、[独自のドメイン名の追加](#)をご参照ください。

プッシュ設定

1. CSSコンソールの[【CSSツールボックス】](#) > [【アドレスジェネレーター】](#)に移動し、プッシュアドレスを発行します。

Domain Type * **Push Domain**

If you select push domain, a push address will be generated; and if you select playback domain, a playback address will be generated. If there is no available domain, please [Add Domain](#)

AppName *

Use "live" by default. Only letters, digits, and symbols are supported.

StreamName *

Only support letters, digits, and symbols.

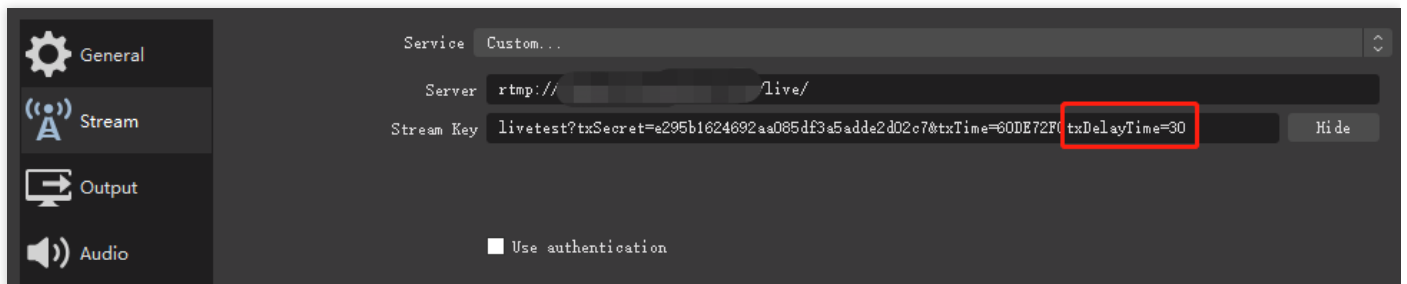
Expiration Time **UTC+8**

The expiration time of playback address is the setting timestamp plus the playback authentication expiration time, and the push address expiration time is the setting time.

Generation Result (Generate the following address according to the above settings)

| | |
|------------------|--|
| Type | Push Domain |
| Expiration Time | 2021-07-02 09:59:12 (UTC+8) reference documentation |
| Push Address | rtmp://[redacted]/live/livetest?txSecret=e295b1624692aa085df3a5adde2d02c7&txTime=60DE72F0 <input type="button" value="🔗"/> |
| OBS Push Address | rtmp://[redacted]/live/ <input type="button" value="🔗"/> |
| OBS Push Name | livetest?txSecret=e295b1624692aa085df3a5adde2d02c7&txTime=60DE72F0 <input type="button" value="🔗"/> |

2. プッシュアドレスの後ろに `txDelayTime` パラメータを追加し、OBSでプッシュを行います。具体的なプッシュ操作については、[OBSプッシュ](#)をご参照ください。



注意：

`txDelayTime`、すなわち遅延再生時間を入力します。単位：秒、上限：600秒、Integer型。

遅延再生

1. CSSコンソールの【CSSツールボックス】>【アドレスジェネレーター】に移動し、対応するプストリーミングアドレスを発行します。設定は次のとおりです。
2. VLC、FFmpegなどのツールを使用して再生を行うことができます。具体的な再生の詳細は、[CSS再生](#)をご参照ください。

![]

以上の再生比較から、再生画面の遅延は34秒であることがわかります。しかし、設定した遅延再生時間は30秒です。プッシュアドレスにtxDelayTimeパラメータを追加する方法で、遅延再生効果を実現できます。

ライブストリーミング機能

CSSパッケージ化とトランスコーディング

最終更新日：：2022-12-21 09:23:53

CSSカプセル化トランスコード機能

CSSカプセル化トランスコード機能とは、ライブストリーミングからプッシュされたオリジナルストリーム（通常はRTMPプロトコルを使用してクラウドにプッシュ）が視聴者にプッシュされる前にクラウドでさまざまなカプセル化形式のビデオストリームに変換されるプロセスです。

サポートする出力カプセル化トランスコード形式

- RTMP
- FLV
- HLS
- DASH
- HDS
- TSストリーム

サポートする出力メディア

- オーディオのみの出力：ビデオメディアを削除し、オーディオメディアのみを出力します。カプセル化形式は上述のとおりです。
- ビデオのみの出力：オーディオメディアを削除し、ビデオメディアのみを出力します。カプセル化形式は上述のとおりです。

サポートするメディア暗号化スキーム

- **Fairplay**
HLSカプセル化はApple fairplay drmソリューションをサポートしています。
- **Widevine**
DASHカプセル化はGoogle widevine drmソリューションをサポートしています。
- **HLSユニバーサルAes-128暗号化**
HLSカプセル化は、ユニバーサルなAes-128を使用した暗号化スキームをサポートしています。

CSSトランスコード機能

CSSトランスコード機能（ビデオトランスコーディングとオーディオトランスコーディングを含みます）とは、ライブストリーミングからリアルタイムでプッシュされたオリジナルストリームを視聴者にプッシュする前に、クラウドでさまざまなコーデック、解像度、ビットレートのトランスコードストリームに変換する機能です。各種インターネット環境、各種端末デバイスなど、各種シナリオでの再生ニーズを満たします。

一般的なユースケースの例

- 元のビデオストリームは、さまざまな解像度のストリームにトランスコードできます。ユーザーは、ネットワーク状態に応じてさまざまなビットレートのビデオストリームを選択し、スムーズに再生できます。
- 著作権表示とマーケティングのためにカスタマイズしたウォーターマークをオリジナルビデオストリームに挿入し、ビデオに独自のロゴを追加することができます。
- ビデオストリームを、元のエンコード形式がH264であるビデオストリームなど、さらに高い圧縮率のビデオコーデック形式に変換します。例えば、視聴する人数が多い場合は、H264のビデオストリームをさらに圧縮率の高いH265のビデオストリームに変換することによって、帯域幅とコストを節約できます。
- 特殊な端末の再生ニーズに対応するため、オリジナルビデオストリームを異なるコーデックに変換します。例えば、ある特殊なシナリオで、パフォーマンスの問題により、H264ビデオをデコードしてもリアルタイム再生できない場合は、元のH264ストリームをMpegコーデックのビデオに変換する必要があります。本手順を踏むことで、各種端末でリアルタイムのデコードと再生が利用できます。

ビデオトランスコードパラメータ

| パラメータタイプ | 説明 |
|-----------------|--|
| ビデオコーデック | 次のビデオコーデック形式をサポート： <ul style="list-style-type: none"> H264 H265 |
| ビデオプロファイル | 次の3種のプロファイルをサポート： <ul style="list-style-type: none"> Baseline Main High |
| ビデオコーデックビットレート | <ul style="list-style-type: none"> サポートするビデオ出力ビットレートの範囲：50kbps - 10Mbps。 指定された出力ビットレートが入力オリジナルビットレートより大きい場合、出力ビットレートはオリジナルビットレートを維持します。例えば、指定された出力ビットレートが3000kbpsで、オリジナル入力ストリームのビットレートが2000kbpsしかない場合、出力ビットレートは2000kbpsを維持します。 |
| ビデオコーデックフレームレート | <ul style="list-style-type: none"> サポートするビデオ出力フレームレート範囲：1fps - 60fps。 指定された出力フレームレートが入力オリジナルフレームレートより大きい場合、出力フレームレートはオリジナルフレームレートを維持します。例えば、指定された出力フレームレートが30fpsであり、オリジナル入力ストリームのフレームレートが20fpsしかない場合、出力フレームレートは20fpsを維持します。 |

| | |
|---------------|---|
| ビデオ解像度 | <ul style="list-style-type: none"> サポートする幅の範囲：128～4096。 サポートする高さの範囲：128～4096。 高さは幅に比例してスケーリングされます。 幅は高さに比例してスケーリングされます。 |
| ビデオGOP長さ | サポートするビデオGOPの長さの範囲：1秒～10秒、通常、2秒～4秒を推奨。 |
| ビデオビットレート制御方法 | <p>次の2種のビットレート制御方法をサポート：</p> <ul style="list-style-type: none"> 固定ビットレート（CBR）。 動的ビットレート（VBR）。 |
| ビデオ画面の回転 | <p>元のビデオ画面の時計回りに3つの角度での回転をサポート：</p> <ul style="list-style-type: none"> 時計回りに90度回転。 時計回りに180度回転。 時計回りに270度回転。 |

オーディオトランスコードパラメータのご紹介

| パラメータタイプ | 説明 |
|------------------|--|
| オーディオコーデック | <p>次のコーデックをサポート：</p> <ul style="list-style-type: none"> AAC-LC AAC-HE AAC-HEV2 |
| オーディオサンプルレート | <p>次の一般的なサンプルレートをサポートします。一般的なサンプルレートは48000と44100です。</p> <ul style="list-style-type: none"> 96000 64000 48000 44100 32000 24000 16000 12000 8000 |
| オーディオコーデックビットレート | <p>サポートするオーディオビットレート範囲：20kbps～192kbps。一般的なオーディオビットレートは次のとおりです。</p> <ul style="list-style-type: none"> 48kbps 64kbps 128kbps |
| オーディオサウンドチャンネル | <p>オーディオは次のサウンドチャンネルをサポート：</p> <ul style="list-style-type: none"> シングルサウンドチャンネル ダブルサウンドチャンネル |

ビデオトランスコーディングに使用される一般的なプリセットテンプレート

| 解像度 | テンプレート名 | ビデオ解像度 | ビデオビットレート | ビデオフレームレート | ビデオコーデック |
|-----|---------|-----------------|-----------|------------|----------|
| LD | 550 | 比例ズーム * 540 | 500kbps | 23 | H264 |
| SD | 900 | 比例ズーム * 720 | 1000kbps | 25 | H264 |
| HD | 2000 | 比例ズーム * 1080 | 2000kbps | 25 | H264 |

高速高画質トランスコーディング機能

Tencent Video Cloudで長年にわたって蓄積されたオーディオ/ビデオコーデック技術、インテリジェントなシナリオ認識、動的コーデック技術、CTU/ライン/フレームの3レベルのビットレートによる正確な制御モデルに基づき、ライブストリーミング、オンデマンドなどの業界向けに、より低いビットレート（平均で50%以上削減）でより高解像度のストリーミングメディアサービスを提供します。

ユースケース

CSSのプッシュビットレートが高く、画面が劣悪な場合は、インテリジェントでダイナミックなエンコード技術と正確なビットレート制御モデルを介して、高解像度と低ビットレートを実現し、視聴者には同等の画質を保証できます。

機能のメリット

今日の各種ビデオプラットフォームユーザーのビデオソースの解像度と視聴体験に対する要求がますます高まっている現状を受け、現在のライブストリーミングサービスは、1080P、ビットレート3Mbps - 10Mbpsが主なストリームの設定となりつつあり、帯域幅のコストがビデオプラットフォームコストの大部分を占めています。ビデオのビットレートを下げることで、帯域幅のコストを効果的に削減することができます。

事例：

一標準ライブストリーミングでビットレートが3Mbps、ライブストリーミングの時間は4時間、視聴者数200人、コーデックはH.264を使用し、解像度1080Pで高速高画質トランスコーディングを使用しない場合、[CSS価格計算機](#)を使用して、発生した帯域幅コストを計算すると、127.08米ドルになります。

- 高速高画質トランスコーディングを使用しビットレートを下げた後に、発生する帯域幅コストは、約 $127.08 \times (100\% - 30\%) = 88.956$ 米ドルとなります。
- 高速高画質トランスコーディングを使用して発生する費用は、 $0.0443 \times 240 = 10.632$ 米ドル（標準価格、いかなる値引きも含まず）です。
- コスト合計： $88.956 + 10.632 = 99.588$ 米ドル。

以上より、高速高画質トランスコーディングは、ユーザーにより良い視聴体験を提供しつつ、プラットフォームの帯域幅コストを効率的に削減します。

主要パラメータ

高速高画質トランスコーディングと標準的なCSSトランスコードのパラメータ設定方法は基本的に同じです。[ビデオトランスコーディングパラメータ](#)をご参照ください。

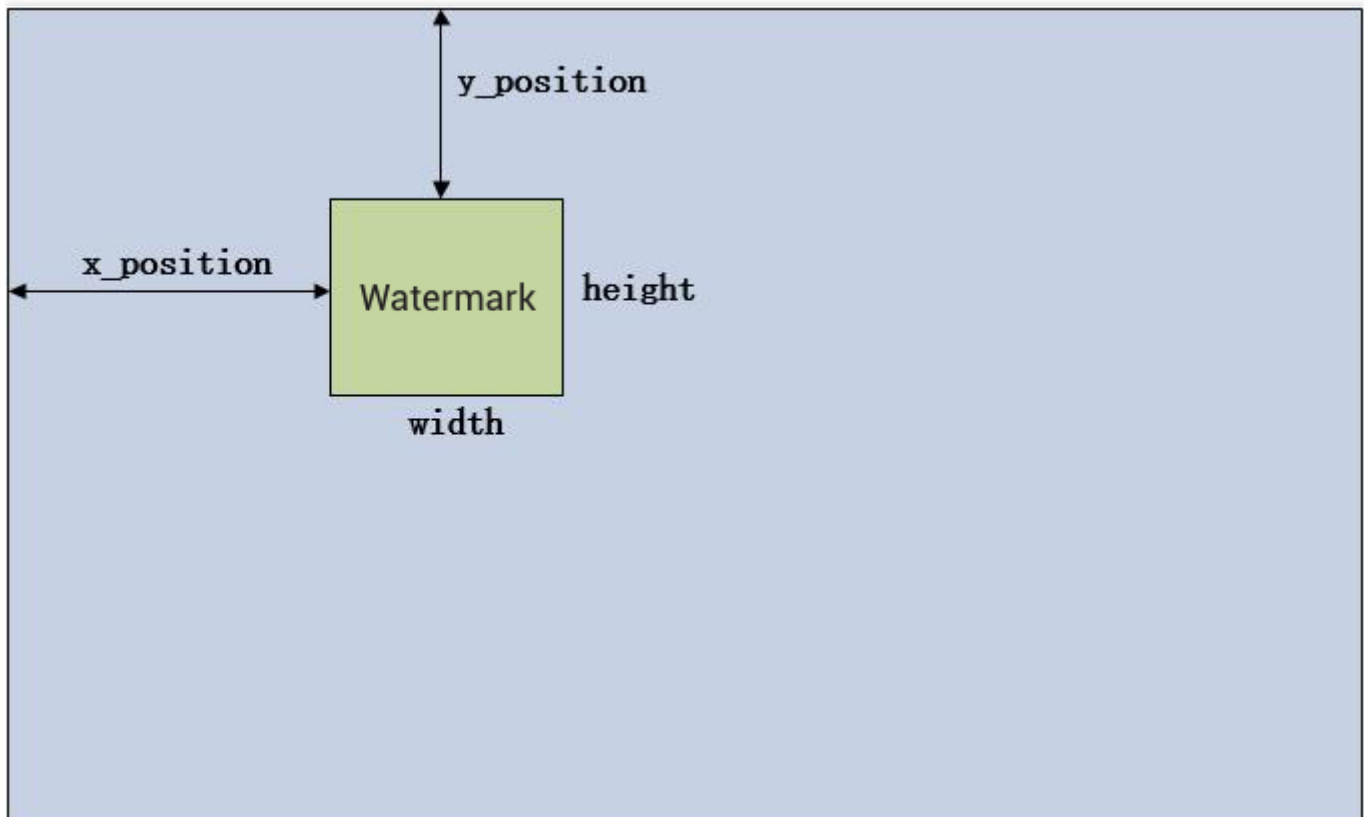
CSSウォーターマーク追加機能のご紹介

ビデオストリームウォーターマーク追加機能とは、オリジナルストリームをベースにして、各種のロゴや画像をビデオ画面に追加し、著作権を表示し、ビデオに独自のロゴを追加するためにプリセットされた機能であり、本機能を利用することでプロモーション効果を発揮することが可能です。

ウォーターマーク関連パラメータ

ウォーターマークの主要パラメータにはサイズと位置があり、それらはパラメータXPosition、YPosition、WidthとHeightによって調整されます。各パラメータの定義は次のとおりです。

- **XPosition** : X軸オフセット。ウォーターマークの左端からビデオの左端までの距離のパーセンテージを示します。
- **YPosition** : Y軸オフセット。ウォーターマークの上端からビデオの上端までの距離のパーセンテージを示します。
- **Width** : ウォーターマークの幅。ライブストリーミングオリジナル画面の幅に占めるパーセンテージ。
- **Height** : ウォーターマークの高さ。ライブストリーミングオリジナル画面の高さに占めるパーセンテージ。

**注意：**

ストリームが複数のビットレートトランスコードを有効にする場合（つまり、オリジナルストリームが複数の解像度のストリームに変換される場合）は、ウォーターマークを追加する必要があります。[CSSコンソール](#) または [API](#) でX軸、Y軸の位置をパーセンテージで設定することができ、システムが自動的にウォーターマークの位置を調整します。

ウォーターマークのパラメータ例

ビデオ出力画面が1920×1080、ウォーターマークのサイズが320×240である場合、パーセンテージの計算方式に XPosition = 5、YPosition = 5、Width=10を使用します。

ビデオ出力画面の解像度に基づき計算されたウォーターマークの絶対位置とサイズは次のとおりです。

```
XPosition_pixel = 1920 * 5% = 96
YPosition_pixel = 1080 * 5% = 54
Width_pixel = 1920 * 10% = 192
Height_pixel = 192 * 240 / 320 = 144
```

以上より、ウォーターマークの位置はビデオ画面左端から96ピクセル、ビデオ出力画面上端から54ピクセル離れた位置となり、ウォーターマークのサイズは192ピクセル * 144ピクセルとなります。

使用法の概要

ウォーターマークを追加するには、[CSSコンソール](#)と[サーバーAPI](#)の2つの方法があります。業務ニーズに応じて対応する使用方式を選択します。

CSSコンソール

1. 【機能設定】 > 【[CSSウォーターマーク](#)】に進み、ウォーターマーク設定テンプレートを追加します。ウォーターマーク関連パラメータを設定し、さらに対応するウォーターマークテンプレートIDを発行します。具体的な操作については、[CSSウォーターマーク](#)をご参照ください。
2. 【[Domain Management](#)】で操作が必要なプッシュドメイン名に対し、【管理】 > 【[テンプレート設定](#)】を選択し、このドメイン名とウォーターマークテンプレートをバインドします。具体的な操作については、[ウォーターマーク設定](#)をご参照ください。

APIの呼び出し

1 [AddLiveWatermark](#) ウォーターマーク追加のインターフェースを呼び出してウォーターマーク名とパラメータ情報を設定します。

2 [CreateLiveWatermarkRule](#) を呼び出してウォーターマークルールを作成し、プッシュドメイン名パラメータ DomainNameとWatermarkId（第1ステップに戻ります）を設定します。AppNameはプッシュと再生アドレス内のAppNameと一致させる必要があり、デフォルトは `live` です。

注意：

ウォーターマーク追加機能を使用すると、標準のトランスコード料金が発生します。

トランスコードパラメータの設定使用方法

使用法の概要

トランスコードパラメータの設定は、[CSSコンソール](#)と[サーバーAPI](#)の2つの方法があります。いずれの方式を使用しても、主にウォーターマークテンプレート、トランスコードテンプレート、トランスコードルールの関連操作が必要です。

CSSコンソール

1. 【機能設定】 > 【[CSSトランスコード](#)】に進み、トランスコード設定テンプレートを追加します。[標準トランスコーディング](#)、[高速高画質トランスコーディング](#)、[オーディオのみのトランスコーディング](#)のテンプレートの追加をサポートしています。
2. ニーズに応じて対応するトランスコードタイプを作成し、トランスコード関連パラメータを設定します。システムのデフォルトパラメータを使用すると、対応するトランスコードテンプレートIDが生成されます。

3. **【Domain Management】** で操作が必要なプルストリームドメイン名を探し、**【管理】** > **【テンプレート設定】** を選択し、このドメイン名とトランスコードテンプレートをバインドします。具体的な操作については、[トランスコーディング設定](#)をご参照ください。

APIの呼び出し

1. **CreateLiveTranscodeTemplate** トランスコードテンプレート作成のインターフェースを呼び出し、必要なトランスコードタイプとパラメータ情報を設定します。
2. **CreateLiveTranscodeRule** トランスコード作成ルールを呼び出し、プルドメイン名DomainNameとTemplateId (第1ステップに戻ります) のパラメータを設定します。AppNameおよびStreamNameに空文字列を入力します。本操作は1つのドメイン名ですべてのプルトランスコードが共通化されたことを意味します。またトランスコードテンプレートを異なるストリーム名とバインドして、特定のライブストリーミングのトランスコードを有効にすることもできます。
3. 各トランスコードテンプレートには対応する**唯一のトランスコードテンプレート名**があり、トランスコードテンプレート名はトランスコードストリームを再生するための一意の識別子として使用されます。トランスコードテンプレート名を再生プルアドレスのストリームIDの末尾に追加して、各種トランスコードテンプレートに対応するトランスコードストリームをプルすることができます。

注意：

トランスコードルールについての操作は、主に特定のドメイン名または特定のストリームに対して特定のトランスコードテンプレートを有効にするために使用されます。トランスコードルールが作成された場合のみ、対応する再生ドメイン名が対応するトランスコードテンプレートをプルできます。トランスコードルールが作成されていない場合は、直接スプライスされたトランスコードテンプレート名を利用してもプルストリーミングアドレスは無効になります。

使用方法の例

再生アドレス = 再生ドメイン名 + 再生パス + ストリームID名_トランスコードテンプレート名 + 認証文字列

プッシュでは、ストリームIDは1234_testであり、次の3つのアドレスを介してウォーターマークの挿入された各種ストリームが再生可能です。

- **オリジナルストリーム**： `http://liveplay.tcloud.com/live/1234_test.flv?認証文字列`
- **SDトランスコードストリーム（ウォーターマーク挿入）**： `http://liveplay.tcloud.com/live/1234_test_sd.flv?認証文字列`
- **HDトランスコードストリーム（ウォーターマーク挿入）**： `http://liveplay.tcloud.com/live/1234_test_hd.flv?認証文字列`

注意：

ウォーターマークが挿入されたストリームを再生する場合は、対応するプッシュドメイン名に作成されたウォーターマークテンプレートをバインドする必要があります。

インターフェースの使用

- **コンソールでトランスコードテンプレートを管理：**

コンソールではトランスコードテンプレートのクエリー、追加、変更、削除をサポートします。

- **サーバーAPIでトランスコードテンプレートを管理：**

| 機能モジュール | APIインターフェース |
|-------------|-------------------------------------|
| CSSトランスコード | トランスコードテンプレートの作成 |
| | トランスコードテンプレート設定の変更 |
| | 単一トランスコードテンプレートの取得 |
| | トランスコードテンプレートリストの取得 |
| | トランスコードテンプレートの削除 |
| | トランスコードルールの作成 |
| | トランスコードルールリストの取得 |
| | トランスコードルールの削除 |
| CSSウォーターマーク | ウォーターマークの追加 |
| | ウォーターマークの更新 |
| | ウォーターマークの削除 |
| | ウォーターマークリストのクエリー |

CSSレコーディング

CSSレコーディング

最終更新日：：2021-08-23 17:43:08

- CSSレコーディングは、ライブストリーミングオリジナルストリームのオーディオおよびビデオのパッケージ化（オーディオ、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）で生成されたファイルをVODプラットフォームにストレージするサービスです。

注意事項

- [レコーディングタスク作成](#) と [レコーディングテンプレート作成の設定](#) の2種類の方法でレコーディングを開始できます。同一のライブストリーミングストリームで、レコーディングテンプレートの設定と同時にレコーディングタスクを作成すると、レコーディングが繰り返されます。
- プッシュ開始後、レコーディングタスクの開始までに一時的なディレイがあるため、プッシュ時間が短すぎると、レコーディングファイルを生成できなくなります。レコーディングファイルの品質を保証するためにも、レコーディング時のプッシュ時間を10秒より長くすることをお勧めします。

レコーディングのストレージ

CSSレコーディングは、ファイルをVODプラットフォームにストレージします。CSSレコーディングサービスを使用したい場合は、まず [VODサービス](#) をアクティブにします。

説明：

生成されるレコーディングファイルの命名ルールを理解したい場合は、[レコーディングテンプレートパラメータ-VodFileName](#) をご参照ください。

レコーディング形式

レコーディングファイル形式のサポート：FLV/HLS/MP4/AAC、そのうちAACはオーディオレコーディング。

レコーディングユースケース

| ユースケース | 説明 |
|-----------------------------------|---|
| プッシュドメイン名、ストリーム名に応じたマルチレベルレコーディング | プッシュドメイン名、ストリーム名のレベル別にレコーディングの要否を設定できます。 |
| 指定時間帯ごとのレコーディング | APIの呼び出しを介して、レコーディングの開始および終了時間を制御し、指定の時間内にレコーディングを実施できます。 |
| 精彩なビデオレコーディング | プッシュ中の精彩な画面は、APIを呼び出し、リアルタイムでレコーディングできます。 |
| 純粋なオーディオレコーディング | プッシュが純粋なオーディオである場合、AACの純粋なオーディオレコーディングを設定できます。 |

指定プッシュドメイン名におけるライブストリームのレコーディングについて

レコーディングパラメータをテンプレート形式で管理し、各種業務シナリオのレコーディング設定テンプレートを作成します。また設定テンプレートを使用して各種プッシュドメイン名、ストリーム名とのバインドを行い、レコーディング設定を柔軟に管理します。

オンデマンドサービスをアクティブ化した後、特定のプッシュドメイン名のライブストリーミングストリームのレコーディングが必要となった場合は、2つの実現方法があります。

CSSコンソール

- i. 【機能設定】 > 【CSSレコーディング】に入り、レコーディング設定テンプレートを追加します。
- ii. 【Domain Management】にプッシュドメイン名を追加し、【管理】をクリックしてこのドメイン名とレコーディングテンプレートをバインドします。操作の詳細は[レコーディング設定](#)をご参照ください。

APIの呼び出し

- i. [CreateLiveRecordTemplate](#)を呼び出し、FlvParamなど、1つ以上のレコーディング形式を設定します。
- ii. [CreateLiveRecordRule](#)を呼び出し、パラメータプッシュドメイン名DomainNameとTemplateId（第1ステップに戻ります）を設定します。AppNameおよびStreamNameに空の文字列を入力します。本操作は本ドメイン名ですべてのプッシュが共通化されることを意味します。

同様に、レコーディングテンプレートを異なるストリーム名とバインドすることによって、一部のCSSストリームのレコーディング開始の効果を実現します。

さらに同一のレコーディングテンプレートを各種の異なるプッシュドメイン名、ストリーム名に連結させることができるため、同一のCSSストリームが同時に複数のレコーディングテンプレートにバインドされている場

合がありますが、最終的に優先度が最も高いテンプレートのみとバインドします。テンプレートのバインドルールには下表に示す優先度があります（煩雑な工程を含むシナリオでのみ使用されます）。

| DomainName | StreamName | 優先度 |
|------------|------------|-----|
| ✓ | ✓ | 0 |
| 空 | ✓ | 1 |
| ✓ | 空 | 2 |
| 空 | そら | 3 |

そのうち：【空】はワイルドカード、【✓】は正確なマッチング、【0】は最も優先度が高いです。優先度の高いテンプレートにマッチすると、マッチングを停止し、この優先度のテンプレートに戻ります。

同一プッシュドメイン名の特定のストリームのレコーディングの無効化

あるプッシュドメイン名にレコーディングを設定済みであるが、このドメイン名に業務上の理由でレコーディングの必要がない一部のプッシュが存在する場合は、次の操作を行ってください。

i. [CreateLiveRecordTemplate](#) を呼び出し、いずれのレコーディング形式も指定しません。

```
https://live.tencentcloudapi.com/?Action=CreateLiveRecordTemplate
&TemplateName=norecord
&Description=test
&<パブリックリクエストパラメータ>
```

ii. [CSSコンソール](#) または Tencent Cloud API [CreateLiveRecordRule](#) を介して、DomainName と StreamName パラメータを設定すると、上述のレコーディングテンプレートと指定プッシュドメイン名、ストリーム名のバインドが行われます。

注意：

上述のスキームは一部のプッシュのレコーディングが不要な場合のみに対応します。操作する必要があるライブストリーミングのストリーム数が多い場合は、主に次の点を考慮しつつ、別のプッシュドメイン名を使用して個別に管理することをお勧めします。

- レコーディングテンプレートかレコーディングルールかを問わず、最大数に制限があります（50）。
- プッシュドメイン名の管理制度は柔軟性が高いため、ドメイン上の変化に対してレコーディングテンプレートとルールを変更する必要性はありません。

指定時間帯のレコーディング

一部のプッシュにつき、指定の開始時間にレコーディングを開始し、指定の終了時間にレコーディングを終了したい場合は、API方式を介してレコーディングを指定することができます。レコーディングテンプレートの設定方式とは異なり、具体的なレコーディングパラメータはAPIを介して指定する必要があります。通常この方式は、いずれのレコーディング方式も設定されていない場合に使用されます。

APIの呼び出し

レコーディングタスク作成APIの具体的な説明については、[CreateRecordTask](#)をご参照ください。

レコーディング例

- 最もシンプルなケースでは、指定されたStreamName、DomainName、AppNameと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=mytest.live.push.com
&StreamName=livetest
&StartTime=1597017600
&EndTime=1597024800
&TemplateId=0
&<パブリックリクエストパラメータ>
```

- 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。

i. [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=templat
&Description=test
&Mp4Param.Enable=1
&Mp4Param.RecordInterval=3600
```

```
&Mp4Param.StorageTime=0
<<パブリックリクエストパラメータ>
```

- 出力例：*

```
{
  "Response": {
    "RequestId": "839d12da-95a9-43b2-a9a0-03366d01b532",
    "TemplateId": 17016
  }
}
```

ii. `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
&AppName=live
&DomainName=mytest.live.push.com
&StartTime=1597017600
&EndTime=1597024800
&TemplateId=17016
<<パブリックリクエストパラメータ>
```

説明：

- 同一のCSSストリームについては、定時タスク間および定時タスクとその他形式のレコーディングタスクの間に衝突などの問題はありません。つまり、作成した複数の定時タスクを指定する時間範囲で重複させることができます。またレコーディング設定を有効にした後、APIを再度呼び出し、レコーディングタスクを作成できます。
- このほか、事前に（例：1時間前、または早朝に当日のタスクを作成するなど）レコーディングタスクを作成し、イベント時間よりやや早い時間をタスク開始時間として指定することをお勧めします。

精彩なビデオレコーディング

プッシュ中に、精彩なビデオ画面を見つけ、その場でレコーディングを開始し、後で使用するために精彩なビデオクリップを作成したい場合は、APIを呼び出して、精彩なビデオレコーディングを指定できます。

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

```
&AppName=live
&DomainName=mytest.live.push.com
&EndTime=1597024800
&<パブリックリクエストパラメータ>
```

精彩なビデオレコーディングについて、説明すべき点は次のとおりです。

- プッシュ中にレコーディングを作成する必要があります。
- `APIStopRecordTask`を呼び出し、事前にタスクを終了することができます。
- 海外のプッシュをサポートします。

ミクスストリーミングのレコーディング

ミクスストリーミングサービスについては、始めに[クラウドミクスストリーミング](#)をご参照ください。

CSSクラウドミクスストリーミングサービスを使用するシナリオでは、レコーディング側がミクスストリーミングパラメータ`OutputStreamType`（出力ストリームタイプ）に従ってミクスストリーミングを2つのタイプに分類します。

- `OutputStreamType` が `0` であるのは、出力ストリームが入力ストリームリストに存在すること、つまり、新しいストリームが生成されていないことを意味します。
- `OutputStreamType` が `1` であるのは、出力ストリームが入力ストリームリストに存在しないこと、つまり、新しいストリームが生成されたことを意味します。

プッシュAとBがあり、ミキシング後の出力ストリームがCであると仮定した場合について説明します。

- `OutputStreamType` が `0` であり、かつCストリームがAストリーム（ストリーム名は同一ですが、ミクスストリーミング後の画面です）と仮定した場合について、レコーディング設定を有効にすると、デフォルトでAストリーム（ミクスストリーミング画面）、Bストリームのレコーディングファイルが作成されます。同一のストリームIDが再利用されるため、Aストリームのオリジナルプッシュはレコーディングされないことにご注意ください。
- `OutputStreamType` が `1` である場合、レコーディング設定を有効にした後、デフォルトでAストリーム、BストリームおよびCストリーム（ミクスストリーミング画面）のレコーディングファイルが作成されます。

ミクスストリーミング画面のみをレコーディングしたい場合は、[APICreateRecordTask](#)を介して実現できます。OutputStreamType 1 のミクスストリーミングタイプについては、上述のAPIを呼び出し、StreamTypeパラメータを 1 に設定する必要があることに注意してください。

注意：

ミクスストリーミングのレコーディングは、レコーディングファイルエラーが発生し、通常の視聴と再生に影響を与えるため、中国内地（大陸）とグローバル/中国香港・マカオ・台湾のライブミクスストリーミングをサポートしていません。

##自動sprayレコーディング（マルチストリームレコーディング）

ネットワークジッターのためにプッシュが複数回中断されると、レコーディングファイルが複数生成されてライブストリームの連続再生に不都合が生じます。本問題を解決するために、CSSレコーディングでは中断した複数のプッシュレコーディングファイルを自動sprayレコーディングすることで短時間で一つのファイルに統合します。

その原理は、HLSレコーディング形式で、HLSの**#EXT-X-DISCONTINUITY** タグを使用して複数プッシュのオーディオおよびビデオデータをセグメント化することです。このタグの機能は、オーディオおよびビデオデータのタイムスタンプ、ビデオコーデック、オーディオコーデック、サンプルレートなどの情報が異なる場合があることを示します。プレーヤーは、連続再生のためにデコーダーを更新する必要があります。したがって、この機能を使用する場合は、プレーヤーは**#EXT-X-DISCONTINUITY** タグをサポートする必要があります。iOSネイティブプレーヤー（またはSafari直接再生）、AndroidのExoPlayer、Web端末のhls.jsプレーヤーはいずれもこのタグをサポートしていますが、VLCなどのプレーヤーはこのタグをサポートしていません。

この機能を使用した後、自動sprayの中断時間（設定可能時間は最長30分です。つまり最長30分中断したプッシュを1つのファイルにsprayできます）を設定し、最終の正常なプッシュが終了した後、中断前後のファイルが自動的にsprayされて、HLSレコーディングファイルが生成されます。

自動sprayレコーディングは、現在、HLS形式をサポートしており、[CSSレコーディング](#)で連続レコーディングタイムアウト時間を設定できます。

注意：

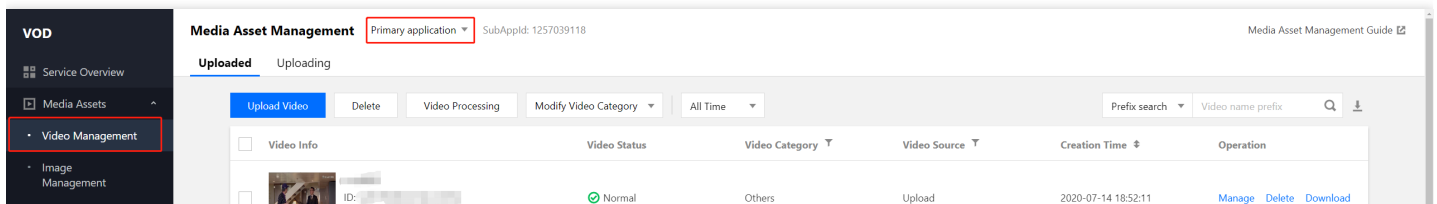
- 自動sprayでは、オーディオデータなしのCSSストリームをサポートしていません。
- ビデオマージ機能は、VODのビデオ合成インターフェースを呼び出すことで実現します。関連ドキュメントの説明については、[ビデオの合成処理](#)をご参照ください。
- HLS連続レコーディング機能を設定すると、途中でプッシュが中断されてもコールバックしません。デフォルトではストリーミングを継続し、最終的なファイル生成のみコールバックします。

レコーディングファイルの取得

レコーディングファイルが生成されると、自動的にVODシステムに保存されます。レコーディングファイルを取得するには、次のような方法があります。

VODコンソール

VODコンソールにログインし、**非管理者**ページで【メディアリソース管理】>【ビデオ管理】を選択すれば、生成されたレコーディングファイルを全て閲覧することが可能です。



レコーディングイベント通知

コンソールまたはAPIの呼び出しによって、レコーディングコールバックアドレスを設定すると、レコーディングファイルの生成後に、当該コールバックアドレスにメッセージ形式で通知されます。メッセージを受信した後は、レコーディングの[コールバックイベントメッセージ通知](#)に従って、業務を処理することができます。

イベント通知メカニズムは、効率的かつ信頼性が高く、即時性もあります。コールバックメソッドを使用してレコーディングファイルを取得することをお勧めします。

VOD APIクエリー

使用方法の詳細についてはVODのAPI [SearchMedia](#) インターフェースを参照してレコーディングファイルをフィルタリングして確認してください。

注意：

CSS APIを介して [レコーディングタスク作成](#) する際は、レコーディングコールバックはユーザーのプッシュURLの [stream_param](#) パラメータは返されませんが、その他のレコーディング方式で作成した場合は返されますのでご注意ください。

設定更新に関する注意事項

レコーディング設定が更新されたら、再プッシュし、設定を検証することをお勧めします。有効性の設定におけるルールは次の通りです。

- 設定が有効になるまでの時間はデフォルトで10分です。
- 設定は、CSSプッシュ開始時に有効になり、レコーディング中に更新されません。
- プッシュ継続時間が長時間にわたるシナリオ（監視カメラなど）については、有効にするためにプッシュを停止してから再起動する必要があります。

ライブストリーミングのタイムシフト

最終更新日：2023-03-13 15:00:00

CSSはタイムシフトを全面的にアップグレードしました。コンソールでタイムシフトのテンプレートを作成することで、新しいバージョンのCSSタイムシフト機能を簡単にアクティブ化できるようになりました。組立ルールに従って、CSSタイムシフトの再生アドレスを組み立てれば、ライブストリーミング中に、ライブストリーミングの内容をプレイバックできます。また、CSSタイムシフト機能はAPI3.0に取り込まれています。詳細については、[新しいバージョンのCSSタイムシフトに関するインターフェース](#)をご参照ください。本書では、新しいバージョンのCSSタイムシフトの仕組みと再生のリクエスト方法を説明します。

注意

- 現在、新しいバージョンのCSSタイムシフトは、3万人の同時視聴をサポートします。タイムシフトでの再生に対して、これ以上の同時視聴者数を要求する場合は、[チケットを提出](#)してご連絡ください。
- 再生ドメイン名に再生認証と有効期限が設定されている場合、認証の有効期限が切れると、タイムシフトの再生アドレスが無効になります。
- 既存の、VODドメイン名でタイムシフトを取得する方法は、チケットを介して評価する必要があります。関連ドキュメントについては、[古いバージョンのCSSタイムシフト](#)をご参照ください。タイムシフトをより良く楽しむために、現在のCSSタイムシフトソリューションを使用することをお勧めします。

タイムシフトの仕組み

CSSタイムシフトはライブストリーミング中に、メディアストリームをTSに変換して保存すると同時に、クラウド側でTSとライブストリームのリアルタイムとの対応関係を作成することで、ライブストリーミングをプレイバックする機能を実装します。この機能は、テレビ局の再生や試合のハイライト再生などによく使用されます。クライアントでHLSプロトコルを通して配信し、M3U8リクエストのパラメータで再生する時間帯(パラメータの詳細は、[再生リクエスト](#)を参照)を指定します。

再生リクエスト

CSSタイムシフトの再生アドレスの形式は `http://domain/appname/stream.m3u8` です。以下の2種類のタイムシフト再生をサポートします。

- 指定した時間帯の再生：試合のハイライトシーンを再生することに使用できます。視聴内容は指定した時間まで再生されます。

- 現在の時間に対するオフセットの再生:ライブストリーミングを遅延させるシーンに使用できます。視聴内容はライブストリーミングが終了するまで再生されます。

指定した時間帯を再生するパラメータ

| フィールド名 | 意味 | 必須か | 例 |
|-------------|--|-----|---------------------------------------|
| txTimeshift | 値がonの場合、新しいバージョンのCSSタイムシフトを有効にする | はい | txTimeshift=on |
| tsStart | タイムシフトの開始時間 | はい | tsStart=20121010010101 |
| tsEnd | タイムシフトの終了時間 | はい | tsEnd=20121010010102 |
| tsFormat | <ul style="list-style-type: none"> tsStartとtsEndの形式。値の形式は <code>{timeformat}_{unit}_{zone}</code> とする timeformatの値： <ul style="list-style-type: none"> UNIX - UNIXタイムスタンプ。UNIXを選択した場合、後続のzoneを省略できる human - 人間が分かる時間 20121010010101 unit : s ms 単位はsとms。 zone : タイムゾーンは東側と西側に分かれている。 <ul style="list-style-type: none"> 東側の値の範囲は1~12、 西側の値の範囲は -12~-1とする。 | はい | tsFormat=unix_s tsFormat=human_s_8 |
| tsCodecname | トランスコーディングストリームは、テンプレート名を指定する必要がある。オリジナルストリームとウォーターマークストリームには、このフィールドがない | いいえ | tsCodecname=hd |

リクエスト例1 (U形式の時間)

```
http://example.domain.com/live/stream.m3u8?txTimeshift=on&tsFormat=unix_s&tsStart=1675302995&tsEnd=1675303025&tsCodecname=test
```

リクエスト例2 (human形式の時間)

```
http://example.domain.com/live/stream.m3u8?txTimeshift=on&tsFormat=unix_s_8&tsStart=20230202095635&tsEnd=20230202095705&tsCodecname=test
```

現在の時間に対するオフセットを再生するパラメータ

| フィールド名 | 意味 | 必須か | 例 |
|-------------|-------------------------------------|-----|--------------------------------|
| txTimeshift | 値がonの場合、新しいバージョンのCSSタイムシフトを有効にする | はい | txTimeshift=on |
| tsDelay | 現在の時間より前の秒数 | はい | tsDelay=30 現在の時間より30秒前の内容を再生する |
| tsCodecname | トランスコーディングストリームは、テンプレート名を指定する必要がある。 | いいえ | tsCodecname=2000 |

リクエスト例

```
http://example.domain.com/live/stream.m3u8?txTimeshift=on&tsDelay=30&tsCodecname=test
```

タイムシフトを認証するパラメータ

タイムシフトはCSS認証のパラメータと一致します。詳細については、[参考ドキュメント](#)をご参照ください（公式サイトで生成したHLSアドレスの有効期限は1日で、1日が経つと、再生成する必要があります）。

タイムシフトインデックスの検索

コンソールのCSSタイムシフトインデックスの情報で、ある時間帯におけるタイムシフトストリームリストを検索できます。詳細をクリックすると、ストリームの詳細情報を個別に確認できます。

TencentCloud APIでもタイムシフトストリームリストと個別のストリームの詳細を確認できます。詳細については、以下のドキュメントをご参照ください：

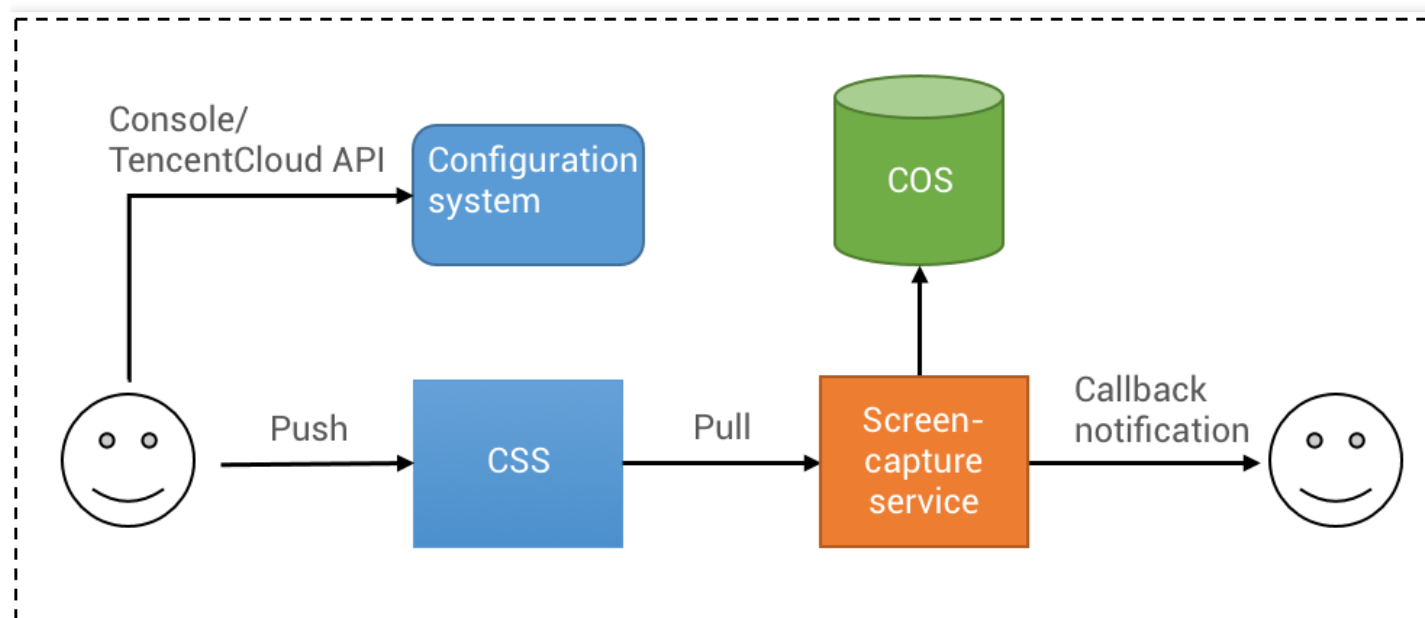
- [CSS タイムシフトリストの確認-APIドキュメント-ドキュメントセンター-Tencent Cloud](#)
- [CSS タイムシフトストリームレコーディング詳細の確認-APIドキュメント-ドキュメントセンター-Tencent Cloud](#)

CSSスクリーンキャプチャ

最終更新日：2023-07-10 10:51:20

CSSスクリーンキャプチャとは、ご自身の好きなタイミングでリアルタイムのライブストリーミングの画像を切り取り、画像を生成することを指します。通知のコールバックによってスクリーンキャプチャ情報を取得でき、またスクリーンキャプチャのデータは、CSSポルノ検出、ライブルームのサムネイルなどの多様なシナリオに応用できます。

CSSスクリーンキャプチャ全体の流れ



全体の流れ：

1. コンソールで、またはTencent Cloud APIを直接呼び出し、CSSスクリーンキャプチャ機能を設定します。
- CSSプッシュを実行します。
 - スクリーンキャプチャサービスは、設定に基づいた上でスクリーンキャプチャデータを生成し、Cloud Object Storageシステムに保存します。
 - 生成されたスクリーンキャプチャの関連情報はコールバックで通知されます。

CSSスクリーンキャプチャの設定

スクリーンキャプチャの設定方式

- [CSS API](#)

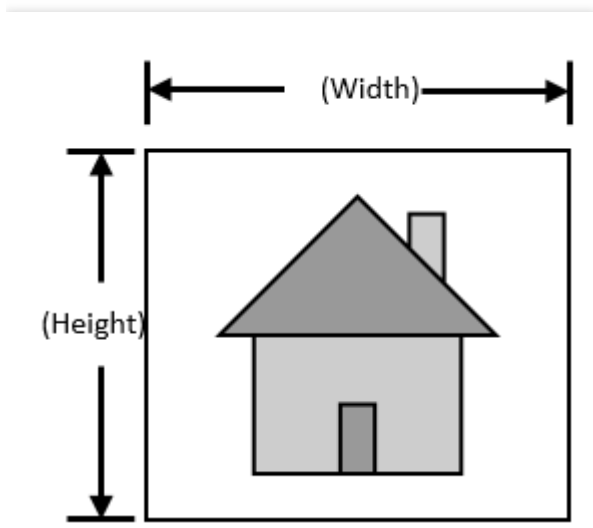
- [【CSSコンソール】](#) > [【機能テンプレート】](#) > [【Screencapturing and Porn Detection Configuration】](#)、詳細な情報については、[Screencapturing and Porn Detection Configuration](#)をご参照ください。

スクリーンキャプチャ間隔の設定

業務上のニーズに沿って、スクリーンキャプチャの頻度を指定することができます。つまり、スクリーンキャプチャの時間間隔（SnapshotInterval）を5秒～300秒の範囲で設定でき、デフォルトの間隔は10秒です。

スクリーンキャプチャの幅と高さの設定

スクリーンキャプチャサービスは、スクリーンキャプチャの幅（Width）と高さ（Height）の指定をサポートしています。



注意：

特別な幅と高さを指定する必要がない場合は、デフォルトのスクリーンキャプチャの幅と高さ（0に設定）がプッシュビデオ画面の幅と高さになります。以下の詳細設定を読まずに、次のセクションにスキップすることもできます。

まず次の3つの幅と高さのコンセプトをお読みください。

- プッシュの幅と高さとは、ライブストリーミングのビデオ画面の幅と高さであり、このテキストでは（X, Y）とします。
- 設定の幅と高さとは、コンソール/Tencent Cloud APIを介して設定する幅と高さであり、このテキストでは（W, H）とします。

- スクリーンキャプチャの幅と高さとは、スクリーンキャプチャサービスが生成するスクリーンキャプチャの幅と高さであり、このテキストでは (N, M) とします。

スクリーンキャプチャサービスはより詳細な設定も対応しています。

- 設定しない場合、つまりデフォルトの (W, H) = (0, 0) を採用する場合は、スクリーンキャプチャの幅と高さ、プッシュの幅と高さは一致し、(N, M) = (X, Y) となります。
- 幅 W のみを設定する場合は、スクリーンキャプチャの幅 $N = W$ となりますが、スクリーンキャプチャの高さは、同等比率でスケーリングされ、 $M = N / X * Y$ となります。
- 高さ H のみを設定する場合は、スクリーンキャプチャの高さ $M = H$ となりますが、スクリーンキャプチャの幅は、同等比率でスケーリングされ、 $N = M / Y * X$ となります。
- (W, H) を同時に設定する場合は、スクリーンキャプチャの幅と高さ、設定の幅と高さが一致し、(N, M) = (W, H) となります。

幅と高さの自動切換機能を設定する場合は、以下のシナリオに沿って設定を行う必要があります。

- $W < H$ に設定し、かつ W と H がいずれも 0 より大きく、プッシュ時に $X > Y$ である場合、設定の幅は小さくなり、プッシュの幅が広がります。

この時、直接スクリーンキャプチャを撮影すると、画像に湾曲や変形が出現することがあります。これらの問題を避けるため、CSSスクリーンキャプチャサービスは、設定の幅と高さのサイズ関係とCSSプッシュ画面が一致するように、バックエンドでWとHの値を自動的に切り替えます。

CSSスクリーンキャプチャのイベントメッセージ通知

イベントメッセージ通知の設定は、[イベントメッセージ通知](#)をご参照ください。スクリーンキャプチャのコールバック通知は、JSON形式で、HTTP POSTプロトコルを使用し、クライアントが事前に設定済みの受信サーバーに通知されます。

スクリーンキャプチャコールバック関連フィールド

| フィールド名 | タイプ | 説明 |
|-------------|--------|-------------------------------------|
| event_type | int | コールバック情報タイプ。スクリーンキャプチャコールバックを200に固定 |
| stream_id | string | CSSストリーム名 |
| channel_id | string | CSSストリーム名と同じ |
| create_time | int64 | スクリーンキャプチャ生成のUnixタイムスタンプ |

| フィールド名 | タイプ | 説明 |
|--------------|--------|---|
| file_size | int | スクリーンキャプチャファイルサイズ。単位：byte |
| width | int | スクリーンキャプチャの幅。単位：ピクセル |
| height | int | スクリーンキャプチャの高さ。単位：ピクセル |
| pic_url | string | スクリーンキャプチャファイルパス /path/name.jpg。詳細は以下のテキスト 一部フィールドの説明 を参照 |
| pic_full_url | string | スクリーンキャプチャの完全なURL。詳細は以下のテキスト 一部フィールドの説明 を参照 |
| sign | string | コールバック署名。詳細は イベントメッセージ通知 を参照 |
| t | int64 | コールバック署名の期限切れUnixタイムスタンプ、詳細は イベントメッセージ通知 を参照 |

一部フィールドの説明

- `pic_url` 説明：
- path：年-月-日
- name：CSSストリーム名-screenshot-時-分-秒-幅x高さ.jpg

例：

```
/2018-12-17/stream_name-screenshot-19-06-59-640x352.jpg
```

このフィールドはカスタムCOS CDNドメイン名のスプライスに使用することができ、CDNドメイン名が不要である場合は、`pic_full_url`を直接使用することができます。

- `pic_full_url` 説明：
- http://COSドメイン名+pic_url

例：

```
http://testbucket-1234567890.cos.region.myqcloud.com/2018-12-17/stream_name-screenshot-19-06-59-640x352.jpg
```

スクリーンキャプチャコールバックの例

```
{
  "event_type": 200,
```

```
"stream_id": "stream_name",  
  
"channel_id": "stream_name",  
  
"create_time": 1545030273,  
  
"file_size": 7520,  
  
"width": 640,  
  
"height": 352,  
  
"pic_url": "/2018-12-17/stream_name-screenshot-19-06-59-640x352.jpg",  
  
"pic_full_url": "http://testbucket-1234567890.cos.region.myqcloud.com/2018-12-17/s  
tream_name-screenshot-19-06-59-640x352.jpg",  
  
"sign": "ca3e25e5dc17a6f9909a9ae7281e300d",  
  
"t": 1545030873  
  
}
```

CSSポルノ検出

最終更新日：2023-02-27 15:47:12

CSSのポルノ検出をアクティブ化するには、まずCSSスクリーンキャプチャ機能をアクティブにする必要があります。CSSコンソールまたはCSSスクリーンキャプチャポルノ検出APIを使用しアクティブにすることができます。本書では、CSSスクリーンキャプチャポルノ検出APIを使用し、CSSポルノ検出機能を実装する方法を説明します。

注意事項

COS Bucketへのアクセス権限がパブリック読取りで、Bucketにポルノ、政治及びその他の禁令違反に関するスクリーンキャプチャが存在する場合、COS Bucketからこれらの画像を削除し、COS Bucketが禁止され利用に支障が出ることを回避することをお勧めします。

CSSポルノ検出のアクティブ化

CSSポルノ検出は、CSSスクリーンキャプチャに基づくため、CSSポルノ検出のアクティブ化には、CSSスクリーンキャプチャ機能のアクティブ化が必要です。具体的な操作は次のとおりです。

1. ポルノ検出機能付きCSSスクリーンキャプチャテンプレートの作成

[CreateLiveSnapshotTemplate](#)を呼び出し、PornFlag = 1を設定することによって、ポルノ検出機能を備えたCSSスクリーンキャプチャテンプレートを作成します。

2. ポルノ検出機能付きCSSスクリーンキャプチャルールの作成

[CreateLiveSnapshotRule](#)を呼び出し、ポルノ検出機能付きCSSスクリーンキャプチャルールを作成し、ステップ1で作成したCSSスクリーンキャプチャテンプレートIDをポルノ検出が必要なライブストリーミングAppId、DomainName、AppName、StreamNameに関連付けます。

3. CSSを開始し、ポルノ検出を開始

ポルノ検出機能付きCSSスクリーンキャプチャルールを作成した後、新たに開始するCSSではポルノ検出機能が有効になります。現在ライブストリーミング中のストリームに対して、CSSのポルノ検出を有効にするには、ライブストリーミングを停止し再開する必要があります。

CSSポルノ検出結果の取得

CSSポルノ検出機能をアクティブにした後、ポルノ検出コールバックテンプレートで登録したコールバックドメイン名を設定すれば、CSSバックエンドからポルノ検出結果がコールバックされます。

注意：

デフォルトでは疑わしい結果に対してのみコールバックを行い、通常の結果にはコールバックを行いません。

1. CSSポルノ検出コールバックテンプレートの作成

[CreateLiveCallbackTemplate](#)を呼び出し、`PornCensorshipNotifyUrl`パラメータにご利用のドメイン名を設定し、CSSポルノ検出コールバックテンプレートを作成します。

2. CSSポルノ検出コールバックルールの作成

[CreateLiveSnapshotRule](#)を呼び出し、ポルノ検出機能付きCSSスクリーンキャプチャルールを作成し、前のステップで作成したCSSスクリーンキャプチャテンプレートIDをポルノ検出が必要なライブストリーミングAppId、DomainName、AppNameに関連付けます。

3. ポルノ検出結果の取得

CSSのバックエンドは、HTTP POSTリクエストを介して登録されたドメイン名にポルノ検出結果を送信します。ポルノ検出結果はJSON形式でHTTP Bodyに保存されます。`type`フィールドのみでライブストリーミングにポルノ関連のコンテンツがあるかを判断できます。

注意：

画像の `type` を使用してポルノ画像かどうかを判断することを推奨しますが、検出システムの判断が100%正しいことを保証できないため、ポルノ疑い有無の認識が正確でない場合もございます。したがって、目視による再確認を行うことをお勧めします。

完全なプロトコルは次のとおりです：

| パラメータ | 入力必須の有無 | データタイプ | 説明 |
|-----------|---------|--------|---------|
| streamId | オプション | String | ストリーム名 |
| channelId | オプション | string | チャンネルID |
| img | 入力必須 | string | 警告画像リンク |

| パラメータ | 入力必須の有無 | データタイプ | 説明 |
|---------------|---------|-----------------------|---|
| type | 入力必須 | Array | 検出結果で優先度が最も高い悪意のあるタグに対応する分類値のことです。具体的な意味については、パラメータlabelによって返される補足説明をご参照ください |
| score | 入力必須 | Array | typeに対応する評価点数 |
| ocrMsg | オプション | string | 画像のOCR認識情報（該当する場合） |
| suggestion | 入力必須 | string | 推奨値。選択可能な値： <ul style="list-style-type: none"> • Block：ブロック • Review：レビュー待ち • Pass：正常 |
| label | 入力必須 | string | このフィールドは、検出結果(LabelResults)において優先度が最も高い悪意のあるタグを返すために使用され、モデルが推奨するレビュー結果を表します。業務上のニーズに応じて、不正タイプと推奨値を処理することをお勧めします |
| subLabel | 入力必須 | string | このフィールドは、検出結果でヒットした最も優先度が高い悪意のあるタグのサブタグ名を返すために使用されます。例えば、ポルノ--性行為。サブタグがヒットされなかった場合は、空の文字列が返されます |
| labelResults | オプション | Array of LabelResult | このフィールドは、分類モデルによってヒットした悪意のあるタグの詳細な識別結果を返すために使用されます。これには、ポルノや広告など好ましくない、安全ではない、または不適切なコンテンツタイプの識別結果が含まれます 注意：このフィールドは、有効な値を取得できなかったことを意味するnullを返す場合があります |
| objectResults | オプション | Array of ObjectResult | このフィールドは、オブジェクト検出モデルの詳細な検出結果を返すために使用されます。これには、エンティティ、広告ロゴ、2次元コードなどのコンテンツでヒットしたタグ名、タグスコア、座標情報、シーンの識別結果、推奨操作といったコンテンツの識別情報が含まれます。詳細な戻り値の情報については、対応するデータ構造(ObjectResults)の説明をご参照ください 注意：このフィールドは、有効な値を取得できなかったことを意味するnullを返す場合があります |

| パラメータ | 入力必須の有無 | データタイプ | 説明 |
|----------------|---------|------------------------------------|--|
| ocrResults | オプション | Array of OcrResult | このフィールドは、OCRテキスト識別の詳細な検出結果を返すために使用されます。これには、テキスト座標情報、テキスト識別結果、推奨操作といったコンテンツの識別情報が含まれます。詳細な戻り値の情報については、対応するデータ構造 (ObjectResults)の説明をご参照ください 注意：このフィールドは、有効な値を取得できなかったことを意味するnullを返す場合があります |
| libResults | オプション | Array of LibResult | リスクライブラリのレビュー結果 |
| screenshotTime | 入力必須 | Number | スクリーンキャプチャ時間 |
| sendTime | 入力必須 | Number | リクエスト送信時間、UNIXタイムスタンプ |
| stream_param | オプション | String | プッシュパラメータ |
| app | オプション | String | プッシュドメイン名 |
| appid | オプション | Number | 業務ID |
| appname | オプション | String | プッシュpathパス |

LabelResult

分類モデルのヒット結果。

| 名称 | タイプ | 説明 |
|------------|----------|--|
| Scene | String | 広告、ポルノ、有害なコンテンツのシーンなど、モデルによって識別されたシーンの結果を返します。 |
| Suggestion | String | 現在の悪意のあるタグに対する後続操作の提案を返します。判定結果が得られた後の戻り値はシステムが推奨する後続操作を表示します。業務上のニーズに応じて、各不正のタイプと推奨値に対処することをお勧めします。戻り値： <ul style="list-style-type: none"> • Block：ブロックを推奨 • Review：手動レビューを推奨 • Pass：承認を推奨 |
| label | String | このフィールドは、検出結果に対応する悪意のあるタグを返すために使用されます |
| | SubLabel | String |

| 名称 | タイプ | 説明 |
|---------|--|-----------------------|
| Score | Integer | このタグモデルでヒットしたスコア |
| Details | Array of LabelDetailItem | 分類モデルでヒットしたサブタグの詳細な結果 |

LabelDetailItem

分類モデルでヒットしたサブタグの結果。

| 名称 | タイプ | 説明 |
|-------|---------|----------------------------|
| Id | Integer | 番号 |
| Name | String | サブタグ名 |
| Score | Integer | サブタグスコア。値の範囲は0ポイント～100ポイント |

ObjectResult

エンティティの検出結果の詳細。

| 名称 | タイプ | 説明 |
|------------|-----------------|---|
| Scene | String | 2次元コード、ロゴ、画像OCRのシーンなど、エンティティによって識別されたエンティティシーンの結果を返します。 |
| Suggestion | String | 現在の悪意のあるタグに対する後続操作の提案を返します。判定結果が得られた後の戻り値は、システムが推奨する後続操作を表します。業務上のニーズに応じて、それぞれの不正タイプと推奨値を処理することをお勧めします。戻り値： <ul style="list-style-type: none"> • Block：ブロックを推奨 • Review：手動レビューを推奨 • Pass：承認を推奨 |
| label | String | このフィールドは、検出結果に対応する悪意のあるタグを返すために使用されます |
| SubLabel | String | サブタグ名 |
| Score | Integer | 該当するシーンモデルでヒットするサブタグのスコア。値の範囲は0ポイント～100ポイント |
| Names | Array of String | エンティティ名リスト |

| 名称 | タイプ | 説明 |
|---------|---------------------------------------|---------------|
| Details | Array of ObjectDetail | エンティティ検出結果の詳細 |

ObjectDetail

エンティティ検出結果の詳細で、検出シーンがエンティティ、広告ロゴ、2次元コードである場合、モデル検出対象フレームのタグ名、タグ値、タグスコアおよび検出フレームの位置情報を表します。

| 名称 | タイプ | 説明 |
|----------|--------------------------|---|
| Id | Integer | このパラメータは、識別や区別を容易にするために、識別されたオブジェクトのIDを返すために使用されます |
| Name | String | このパラメータは、ヒットしたエンティティタグを返すために使用されます |
| Value | String | このパラメータは、対応するエンティティタグに対応する値またはコンテンツを返すために使用されます。例えば、タグが2次元コード(QrCode)の場合、このフィールドは識別された2次元コードに対応するURLアドレスになります |
| Score | Integer | このパラメータは、対応するエンティティタグのスコアを返すために使用され、値は0~100です。例えば、QrCode 99は、対応する識別コンテンツが2次元コードシーンタグにヒットする確率が非常に高いことを意味します |
| Location | Location | このフィールドは、エンティティ検出フレームの座標位置（左上隅のxy座標、長さ、幅、回転角度）を返すために使用され、エンティティの関連情報の素早い特定を容易にします |

Location

座標。

| 名称 | タイプ | 説明 |
|--------|-------|-------------|
| X | Float | 左上隅の横座標 |
| Y | Float | 左上隅の縦座標 |
| Width | Float | 幅 |
| Height | Float | 高さ |
| Rotate | Float | 検出フレームの回転角度 |

OcrResult

OCR検出結果の詳細です。

| 名称 | タイプ | 説明 |
|------------|--|---|
| Scene | String | 識別シーンを表します。デフォルト値はOCR（画像OCR認識）です。 |
| Suggestion | String | 優先度が最も高い悪意のあるタグに対応する後続操作の提案を返します。判定結果が得られた後の戻り値は、システムが推奨する後続操作を表します。業務上のニーズに応じて、それぞれの不正タイプと推奨値を処理することをお勧めします。戻り値： <ul style="list-style-type: none"> • Block：ブロックを推奨 • Review：手動レビューを推奨 • Pass：承認を推奨 |
| label | String | このフィールドは、検出結果に対応する悪意のあるタグを返すために使用されます |
| SubLabel | String | サブタグ名 |
| Score | Integer | 該当するシーンモデルでヒットするサブタグのスコア。値の範囲は0ポイント～100ポイント |
| Text | String | テキストコンテンツ |
| Details | Array of OcrTextDetail | OCR 結果の詳細 |

OcrTextDetail

OCRテキスト結果の詳細です。

| 名称 | タイプ | 説明 |
|----------|--------------------------|---|
| Text | String | OCRで識別されたテキストコンテンツを返します（OCRで識別できるテキストの上限は** 5000バイト**です）。 |
| label | String | このフィールドは、検出結果に対応する悪意のあるタグを返すために使用されます |
| Keywords | Array of String | このタグでヒットしたキーワード |
| Score | Integer | このタグモデルでヒットするスコア。値の範囲は0ポイント～100ポイント |
| Location | Location | OCR テキスト座標位置 |

LibResult

ブラック/ホワイトライブラリ結果の詳細です。

| 名称 | タイプ | 説明 |
|------------|---------------------------------------|---|
| Scene | String | モデルのシーン識別結果を表します。デフォルト値はSimilarです。 |
| Suggestion | String | 後続操作の提案を返します。判定結果が得られた後の戻り値は、システムが推奨する後続操作を表します。業務上のニーズに応じて、それぞれの不正タイプと推奨値を処理することをお勧めします。戻り値： <ul style="list-style-type: none"> • Block：ブロックを推奨 • Review：手動レビューを推奨 • Pass：承認を推奨 |
| label | String | このフィールドは、検出結果に対応する悪意のあるタグを返すために使用されます |
| SubLabel | String | サブタグ名 |
| Score | Integer | 画像検索モデル認識スコア。値の範囲は0ポイント～100ポイント |
| Details | Array of ObjectDetail | ブラック/ホワイトライブラリ結果の詳細 |

LibDetail

カスタマイズコーパス/ブラック/ホワイトライブラリの詳細です。

| 名称 | タイプ | 説明 |
|---------|---------|---------------------------------------|
| Id | Integer | 番号 |
| ImageId | String | 画像ID |
| label | String | このフィールドは、検出結果に対応する悪意のあるタグを返すために使用されます |
| Tag | String | カスタムタグ |
| Score | Integer | モデル認識スコア。値の範囲は0ポイント～100ポイント |

コールバックメッセージの例

```
{
  "ocrMsg": "",
  "type": [1],
  "score": 99,
  "screenshotTime": 1610640000,
  "level": 0,
```

```
"img": "http://1.1.1.1/download/porn/test.jpg",
"abductionRisk": [],
"faceDetails": [],
"sendTime": 1615859827,
"suggestion": "Block",
"label": "Porn",
"subLabel": "PornHigh",
"labelResults": [{
  "HitFlag": 0,
  "Scene": "Illegal",
  "Suggestion": "Pass",
  "Label": "Normal",
  "SubLabel": "",
  "Score": 0,
  "Details": []
}, {
  "HitFlag": 1,
  "Scene": "Porn",
  "Suggestion": "Block",
  "Label": "Porn",
  "SubLabel": "PornHigh",
  "Score": 99,
  "Details": [{
    "Id": 0,
    "Name": "PornHigh",
    "Score": 99
  }, {
    "Id": 1,
    "Name": "WomenChest",
    "Score": 99
  }]
}, {
  "HitFlag": 0,
  "Scene": "Sexy",
  "Suggestion": "Pass",
  "Label": "Normal",
  "SubLabel": "",
  "Score": 0,
  "Details": []
}, {
  "HitFlag": 0,
  "Scene": "Terror",
  "Suggestion": "Pass",
  "Label": "Normal",
  "SubLabel": "",
  "Score": 0,
  "Details": []
}
```

```
    }],  
    "objectResults": [{  
      "HitFlag": 0,  
      "Scene": "QrCode",  
      "Suggestion": "Pass",  
      "Label": "Normal",  
      "SubLabel": "",  
      "Score": 0,  
      "Names": [],  
      "Details": []  
    }, {  
      "HitFlag": 0,  
      "Scene": "MapRecognition",  
      "Suggestion": "Pass",  
      "Label": "Normal",  
      "SubLabel": "",  
      "Score": 0,  
      "Names": [],  
      "Details": []  
    }, {  
      "HitFlag": 0,  
      "Scene": "PolityFace",  
      "Suggestion": "Pass",  
      "Label": "Normal",  
      "SubLabel": "",  
      "Score": 0,  
      "Names": [],  
      "Details": []  
    }],  
    "ocrResults": [{  
      "HitFlag": 0,  
      "Scene": "OCR",  
      "Suggestion": "Pass",  
      "Label": "Normal",  
      "SubLabel": "",  
      "Score": 0,  
      "Text": "",  
      "Details": []  
    }],  
    "streamId": "teststream",  
    "channelId": "teststream",  
    "stream_param": "txSecret=40f38f69f574fd51126c421a3d96c374&txTime=5DEBEC80",  
    "app": "5000.myqcloud.com",  
    "appname": "live",  
    "appid":10000,  
    "event_type": 317,  
    "sign": "ac920c3e66*****78cf1b5de2c63",
```

```
"t": 1615860427  
}
```

CSSポルノ検出の無効化

CSSポルノ検出を無効にするには、スクリーンキャプチャルールを削除するか、スクリーンキャプチャテンプレートを変更する必要があります。削除と変更は新たなCSSに対してのみ有効です。すでに開始されているCSSに対して、ポルノ検出を無効にする場合、ライブストリーミングを停止し、再開する必要があります。

1. ポルノ検出スクリーンキャプチャルールの削除による無効化

[DeleteLiveSnapshotRule](#)を呼び出し、テンプレートIDでDomainName、AppName、StreamNameに対応するCSSスクリーンキャプチャルールを削除します。

2. ポルノ検出スクリーンキャプチャテンプレートの削除または変更による無効化

[ModifyLiveSnapshotTemplate](#)を呼び出し、CSSテンプレートのポルノ検出タグを0に変更します。

ライブストリーミングAV1コーデックの実現

最終更新日：2022-10-14 16:26:04

AV1はオープンソースで、著作権フリーのビデオ圧縮形式です。前世代のH.265[HEVC]コーデックに比べて、同じ画質でのビットレートをさらに30%以上も低下させることができます。これは同じ帯域幅でより高画質な画像を伝送でき、それによって帯域幅コストを削減する効果を達成することを意味します。ここでは、どのようにしてビデオをAV1形式のビデオにトランスコードして再生するかを知ることができます。

AV1の使用

前提条件

- [Tencent Cloudアカウントの登録](#)を行っていること。
- Tencent Cloud CSSサービスをアクティブ化し、[プッシュ&再生ドメイン名](#)を追加済みであること。

ステップ1：トランスコードテンプレートの作成

1. CSSコンソールにログインし、[機能設定](#) > [CSSトランスコード](#)と進みます。
2. [トランスコードテンプレートの作成](#)をクリックして、トランスコードタイプは標準トランスコーディングまたは高速高画質トランスコーディングを選択し、高度な設定を開きます。

3. コーデック方式から**AV1**を選択します。

Live Transcoding

Transcoding Type Standard Transcoding Top Speed Codec Transcoding Audio-only Transcoding

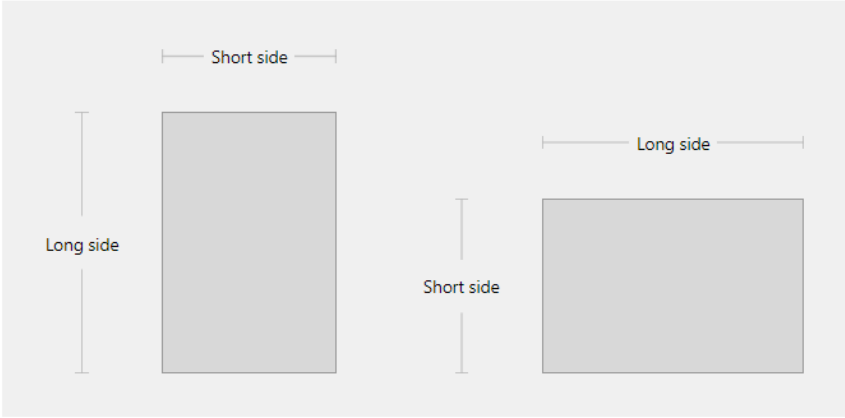
Template Name *
Supports only letters or a combination of letters and digits

Template Description
Only supports letters, digits, underscores, and dashes

Video quality Smooth SD HD

Video Bitrate *

Video Resolution *
Enter a multiple of 2. The other side will be scaled proportionally according to resolution.



DRM encryption
Supports Widevine, FairPlay, and NormalAES DRM encryption for HLS playback. For FairPlay DRM, you need to upload the certificate you obtain from Apple to your player. [Obtaining a FairPlay certificate](#)
Before you enable DRM encryption, please go to [DRM Management](#) to configure the key information.

Codec Original codec H.264 H.265 AV1

4. 入力完了したら、**保存**をクリックして終了します。

ステップ2：ドメイン名のバインド

トランスコードテンプレートを選択し、**ドメイン名のバインド**をクリックして、バインドが必要なトランスコードテンプレートおよび**再生ドメイン名**を選択し、**OK**をクリックするとバインドが成功します。

Bind Domain Name ✕

Domain name binding takes effect in about 10 minutes after the configuration. Please add "_transcoding template name" after the StreamName to generate a new URL for playing back the transcoded stream.

Transcoding Template

Playback Domain Delete

[Add](#)

ステップ3：再生アドレスの発行

アドレスジェネレーターをクリックし、ステップ2でバインドした再生ドメイン名とステップ1のトランスコードテンプレートを選択し、再生アドレスを発行します。

ステップ4：AV1ビデオの再生

AV1をサポートしているプレーヤーによって、再生ステップ3で発行したアドレスに従って再生すれば完了です。プレーヤーの選択では、AV1をサポートしているプレーヤーを選択することができ、独自のプレーヤーを改造することもできます。

• AV1をサポートしているプレーヤー

◦ Appクライアント

- [ExoPlayer](#) AV1対応で、libgav1を使用しています
- [ijkplayer](#) FFmpegはバージョンが古いため、FFmpegをアップグレードしてdav1dに統合することができません

◦ Web端末

- [dash.js](#) サポート済みです（デコードはブラウザに依存し、Chromeをサポートしています）
- [shaka-player](#) サポート済みです（デコードはブラウザに依存し、Chromeをサポートしています）

◦ PC端末

VLC PC版は、AV1 in FLV、HEVC in FLVをサポートしており、必要に応じて[Windows版](#) & [MacOS版](#)をダウンロードできます

- 独自のプレーヤーの改造

プレーヤーにAV1形式のビデオを再生する機能がない場合は、[お問い合わせ](#) からプレーヤーを改造できます。

クラウドミクスストリーミング

最終更新日：2021-10-09 11:48:15

CSSサービスはライブミクスストリーミング機能を提供します。設定済みのミクスストリーミングレイアウトに基づき、各入力ソースの複数のストリームを統合して同期時に新しいストリームとみなし、ライブストリーミングのインタラクティブ効果を実現します。また、CSSライブミクスストリーミング機能はAPI 3.0が接続されています。具体的には、[ライブミクスストリーミングインターフェース](#)をご参照ください。このテキストでは、例を挙げて、各種シナリオでライブストリーミングのミクスストリーミングを実現する方法を説明します。

注意事項

- クラウドミクスストリーミング機能を使用すると、標準トランスコード料金が発生します。料金に関する説明については、[トランスコード料金](#)をご参照ください。
- ミクスストリーミングのクリッピング機能を使用する場合は、クリッピングパラメータの値をソースストリームのパラメータより大きくすることはできません。



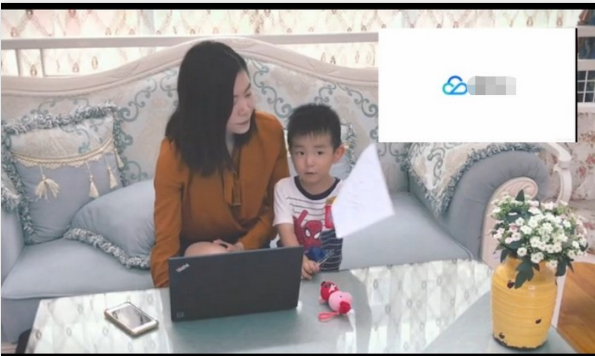
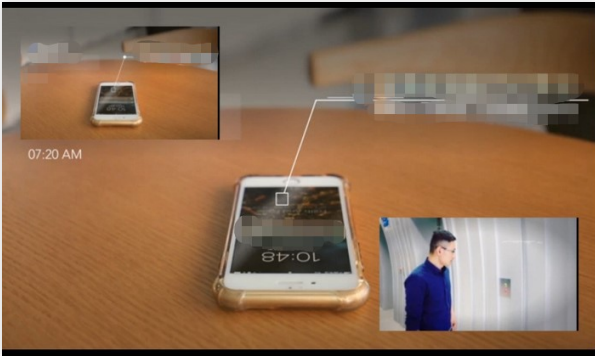
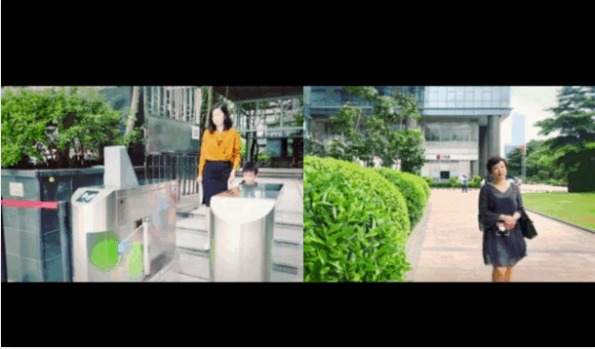
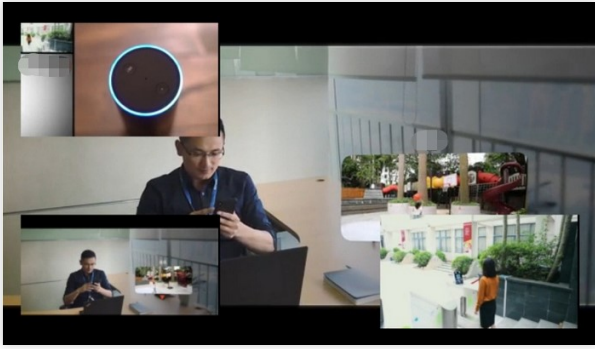
機能サポート

- 最大**16**のストリームのミクスストリーミングを同時にサポートします。
- 5**種類の入力ソースタイプ（オーディオおよびビデオ、純粋なオーディオ、純粋なビデオ、画像、キャンバス）のミキシングをサポートします。
- ミキシング後のストリームを新ストリームとして出力することをサポートします。
- クリッピング、ウォーターマーク機能をサポートします。
- テンプレートの設定をサポートします。
- ミクスストリーミングのレコーディングをサポートします。
- 自動ミクスストリーミングをサポートします。
- リアルタイムでミクスストリーミングの種類と位置の切り替えをサポートします。
- ミクスストリーミングの開始とキャンセルの切り替えをシームレスに実行できます。

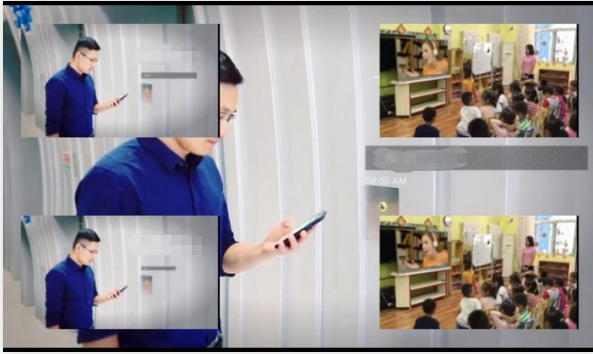
一般的なレイアウトテンプレート

一般的なテンプレートには、10、30、40、310、390、410、510および610があります。これら8つのテンプレートを使用する場合、入力ストリームに位置および長さおよび幅のパラメータを入力する必要はなく、**オリジナル画面と同等比率のスケーリング**になります。ただし、テンプレートIDが必要です。

一般的なレイアウトテンプレート：

| | |
|---|--|
| <p>テンプレート10</p> | <p>テンプレート30</p> |
|  |  |
| <p>テンプレート40</p> | <p>テンプレート310</p> |
|  |  |
| <p>テンプレート390</p> | <p>テンプレート410</p> |
|  |  |
| <p>テンプレート510</p> | <p>テンプレート610</p> |

テンプレート10



テンプレート30



ミクスストリーミングの作成

パラメータの説明

より詳細な説明をご覧になる場合は、[ライブミクスストリーミング](#)をご参照ください。

シナリオ1：ミクスストリーミングの申請-20テンプレートの使用

ミクスストリーミングを使用して、テンプレートミクスストリーミングをプリセットします。

入力例

```
https://live.tencentcloudapi.com/?Action=CreateCommonMixStream
&MixStreamSessionId=test_room
&MixStreamTemplateId=20
&OutputParams.OutputStreamName=test_stream1
&InputStreamList.0.InputStreamName=test_stream1
&InputStreamList.0.LayoutParams.ImageLayer=1
&InputStreamList.1.InputStreamName=test_stream2
&InputStreamList.1.LayoutParams.ImageLayer=2
&<パブリックリクエストパラメータ>
```

出力例

```
{
  "Response": {
    "RequestId": "e8fa8015-0892-40d5-95c4-12a4bc06ed31"
  }
}
```

キャスターのマイク接続のミクスストリーミング効果



シナリオ2：ミクスストリーミングの申請-390テンプレートの使用

ミクスストリーミングを使用して、テンプレートミクスストリーミングをプリセットします。

入力例

```
https://live.tencentcloudapi.com/?Action=CreateCommonMixStream
&MixStreamSessionId=test_room
&MixStreamTemplateId=390
&OutputParams.OutputStreamName=test_stream2
&InputStreamList.0.InputStreamName=test_stream1
&InputStreamList.0.LayoutParams.ImageLayer=1
&InputStreamList.0.LayoutParams.InputType=3
&InputStreamList.0.LayoutParams.ImageWidth=1920 (キャンバスの幅)
&InputStreamList.0.LayoutParams.ImageHeight=1080 (キャンバスの高さ)
&InputStreamList.0.LayoutParams.Color=0x000000
&InputStreamList.1.InputStreamName=test_stream2
&InputStreamList.1.LayoutParams.ImageLayer=2
&InputStreamList.2.InputStreamName=test_stream3
&InputStreamList.2.LayoutParams.ImageLayer=3
&<パブリックリクエストパラメータ>
```


出力例

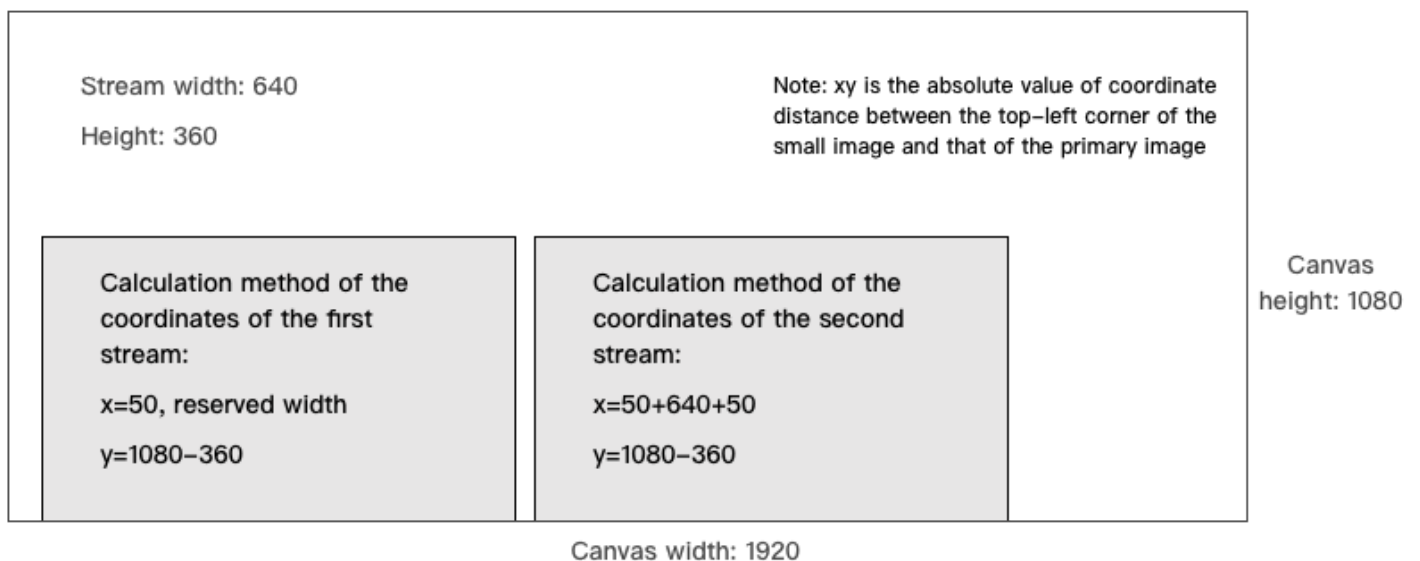
```
{
  "Response": {
    "RequestId": "9d8d5837-2273-4936-8661-781aeab9bc9c"
  }
}
```

キャストPKのミクスストリーミング効果



シナリオ3：カスタマイズされたミクスストリーミングの例

カスタマイズされたレイアウトを使用します。そのうち、位置パラメータLocationXとLocationYは画面左上隅に対応する背景画面左上隅の絶対ピクセル距離です。



入力例

```
https://live.tencentcloudapi.com/?Action=CreateCommonMixStream
&MixStreamSessionId=test_room
&OutputParams.OutputStreamName=test_stream2
&InputStreamList.0.InputStreamName=test_stream1
&InputStreamList.0.LayoutParams.ImageLayer=1
&InputStreamList.0.LayoutParams.InputType=3
&InputStreamList.0.LayoutParams.ImageWidth = 1920
&InputStreamList.0.LayoutParams.ImageHeight= 1080
&InputStreamList.0.LayoutParams.Color=0x000000
&InputStreamList.1.InputStreamName=test_stream2
&InputStreamList.1.LayoutParams.ImageLayer=2
&InputStreamList.1.LayoutParams.ImageWidth = 640
&InputStreamList.1.LayoutParams.ImageHeight= 360
&InputStreamList.1.LayoutParams.LocationX= 50
&InputStreamList.1.LayoutParams.LocationY= 720
&InputStreamList.2.InputStreamName=test_stream3
&InputStreamList.2.LayoutParams.ImageLayer=3
&InputStreamList.2.LayoutParams.ImageWidth = 640
&InputStreamList.2.LayoutParams.ImageHeight= 360
&InputStreamList.2.LayoutParams.LocationX= 740
&InputStreamList.2.LayoutParams.LocationY= 720
&<パブリックリクエストパラメータ>
```

出力例

```
{
  "Response": {
    "RequestId": "8c443359-ba07-4b81-add8-a6ff54f9bf54"
  }
}
```

カスタマイズされたミクスストリーミングの効果



ミクスストリーミングのキャンセル

パラメータの説明

具体的には、[一般的なミクスストリーミングのキャンセル](#)をご参照ください。

シナリオの例

session idに基づき、ミクスストリーミングをキャンセルします。

入力例

```
https://live.tencentcloudapi.com/?Action=CancelCommonMixStream
&MixStreamSessionId=test_room
```

出力例

```
{
  "Response": {
    "RequestId": "3c140219-cfe9-470e-b241-907877d6fb03"
  }
}
```

注意：

- ミクスストリーミングを申請後、5秒以上経ってから、ミクスストリーミングをキャンセルしてください。
- ミクスストリーミングをキャンセルした後、30秒経たなければ、同一のsession idでミクスストリーミングを申請することができません。

エラーコード

クラウドミクスストリーミングAPI3.0は大部分の一般的なエラーコードを [API3.0 エラーコード](#) のスタイルに変換されていますが、一部のエラーコードをカバーしきれていない可能性があります。これらエラーコードは `InvalidParameter` でエラー表示され、`Message` 中に `err_code [$code],msg [$message]` の形式で提示されます。具体的なcodeに対応する原因は次のとおりです。

| エラーコード | 原因 | 原因調査の推奨 |
|--------|---------------|---|
| -1 | 入力解析パラメータエラー | <ul style="list-style-type: none"> • リクエストボディ <code>body json</code> 形式が正しいかどうかをチェックします。 • <code>InputStreamList</code> が空かどうかをチェックします。 |
| -2 | 入力パラメータエラー | 画面パラメータがオーバーフローしていないかチェックします。 |
| -3 | ストリーム数のエラー | 入力ストリーム数が <code>[1, 16]</code> の範囲内にあるかチェックします。 |
| -4 | ストリームパラメータエラー | <ul style="list-style-type: none"> • 入力出力の長さや幅が <code>(0, 3000)</code> の範囲内にあるかどうかをチェックします。 • 入力ストリーム数が <code>(0, 16]</code> の範囲内にあるかどうかをチェックします。 • 入力ストリームが <code>LayoutParams</code> を伴っているかどうかをチェックします。 • <code>InputType</code> が (有効数値: 0, 2, 3, 4, 5) をサポートしているかどうかをチェックします。 • ストリームIDの長さが <code>(1, 80)</code> を満たしているかどうかをチェックします。 |
| -11 | レイヤーエラー | <ul style="list-style-type: none"> • レイヤー数と入力ストリーム数が一致しているかどうかをチェックします。 • レイヤーIDが重複していないかチェックします。 • レイヤーIDが <code>(0, 16]</code> の間にあるかどうかをチェックします。 |

| エラーコード | 原因 | 原因調査の推奨 |
|--------|---|--|
| -20 | 入力パラメータとインターフェースがマッチしません | <ul style="list-style-type: none"> 入力ストリーム数がテンプレートIDとマッチしているかどうかをチェックします。 カラーパラメータが正しいかどうかをチェックします。 |
| -21 | ミクスストリーミング入力ストリーム数のエラー | 入力ストリームの本数が2本以上であるかどうかをチェックします。 |
| -28 | 背景の長さや幅の取得に失敗しました | <ul style="list-style-type: none"> キャンバスを設定する場合は、キャンバスの長さや幅が設定されているかどうかをチェックします。 バックグラウンドストリームが存在しているかどうかをチェックします（プッシュ後、5秒経ってから、再びミクスストリーミングする必要があります）。 |
| -29 | クリッピングパラメータのエラー | クリッピング位置がストリームの長さや幅を超過していないかチェックします。 |
| -33 | ウォーターマーク画像IDエラー | 入力画像IDが設定されているかどうかをチェックします。 |
| -34 | ウォーターマーク画像URLの取得に失敗しました | 画像が正常にアップロードされているかどうか、URLが生成済みかどうかをチェックします。 |
| -111 | OutputStreamName パラメータと OutputStreamType がマッチしません | <ul style="list-style-type: none"> OutputStreamTypeが0である場合、OutputStreamNameはInputStreamListに表示される必要があります。 OutputStreamTypeが1である場合、OutputStreamNameはInputStreamListに含まないでおく必要があります。 |
| -300 | 出力ストリームIDがすでに使用されています | 現在の出力ストリームが別のミクスストリーミングの出力ストリームとなっていないかどうかをチェックします。 |
| -505 | 入力ストリームがuploadで見つかりません | プッシュの成功から5秒後に、ミクスストリーミングを開始します。再生できるかどうかをチェックします。 |

| エラーコード | 原因 | 原因調査の推奨 |
|----------------------------|---|---|
| -507 | ストリームの長さや幅パラメータのクエリーに失敗 | <ul style="list-style-type: none"> キャンバスの幅、高さが設定されているかどうかをチェックします。 プッシュが正常に実施されたかどうかをチェックし、プッシュから5秒後にミクスストリーミングを再開することを推奨します。 |
| -508 | 出力ストリームIDエラー | 同様のMixStreamSessionIdに異なる出力ストリームIDを使用する状況が存在するかどうかをチェックします。 |
| -10031 | ミクスストリーミングの起動に失敗 | プッシュから5秒後に再びミクスストリーミングすることを推奨します。 |
| -30300 -31001 -31002 | ミクスストリーミングキャンセル時のsessionidが存在しません | MixStreamSessionIdが存在するかどうかをチェックします。 |
| -31003 | 出力ストリームIDとsessionの出力ストリームIDがマッチしません | ミクスストリーミングキャンセル時に入力した出力ストリームIDをチェックします。 |
| -31004 | 出力ストリームのビットレートが不正です | 出力ストリームのビットレートが[1, 50000]の間であるかどうかをチェックします。 |
| その他 | その他エラーについては、 でカスタマーサービス に連絡し、テクニカルサポートを要請してください | - |

よくあるご質問

- ミクスストリーミング中に入力ストリームの自動ズームを維持し、かつ黒枠がない状態にするにはどうすればよいですか。
- プッシュ後にミクスストリーミングした場合に、-505エラーコードが返されるのはなぜですか。
- ミクスストリーミングの申請後、ミクスストリーミングをキャンセルしない状態を継続した場合、どうなりますか。
- ミクスストリーミングのアシスタントキャスター画面が予想される位置とは異なることがあるのはなぜですか。

説明：

クラウドミクスストリーミングに関連するその他の質問については、[クラウドミクスストリーミング関連](#)をご参照ください。

Live Streaming Security

ビデオライブストリーミング再生のセキュリティ

最終更新日：2022-12-23 15:11:47

用語の説明

- **クライアント**：Tencent Cloud CSS製品を使用する使用者側で、通常はTencent Cloudアカウントの所有者です。
- **ユーザー**：Tencent Cloudクライアント製品の端末使用者で、通常は視聴者です。

ライブストリーミングの安全なユースケース

| ユースケース | 説明 |
|-----------------|---|
| ライブストリーミングの盗用防止 | ライブストリーミング再生リンクはURLです。ホットリンク防止対策がない再生URLは、取得さえすれば誰でも再生することができます。さらに再生URLは一定のルールがあることが多く、例えば プロトコル:\\再生ドメイン名\AppName\ストリームID 方式をスプレディングしたものが 있습니다。不正アクセス防止リンクがない場合、ルールを推測してルールに応じて組み合わせてすれば、誰でもトラバース再生し、すべてのストリームを視聴することができます。一つ目にはコンテンツを秘密にできず、二つ目には盗用によって発生したトラフィックは自身で支払う必要があります。 |
| 視聴者のフィルタリング | ライブストリーミングコンテンツは視聴者が自由に視聴することを許可するものではなく、次の条件を満たすユーザー層の視聴のみを承認するものです。例えば会員、ログインしたユーザー、購入が完了したユーザーなどです。 |
| コンテンツの著作権制限 | ライブストリーミングコンテンツの著作権側は明確な制限があり、信頼性の高い安全な環境で再生する必要があります。例えばディズニー映画、中国のテレビ局のライブストリーミングなどです。 |
| コンテンツセキュリティ | ライブストリーミングコンテンツは一定の機密性またはプライバシーがあり、特定の人だけが視聴できます。 |

以上のライブストリーミングシーンにはいずれもライブストリーミングのセキュリティが必要ですが、ここでは簡単なものから複雑なものまでTencent Cloud CSSのセキュリティソリューションをそれぞれご紹介します。全体として、ライブストリーミングのセキュリティには2つの面で保証があります。一つは再生URLで、再生URLの使

用および配信を制限します。もう一つはコンテンツで、ライブストリーミングのコンテンツを暗号化します。前者は実現が簡単で、プレーヤーのサポートは必要なく、どのプレーヤーでも可能です。後者はプレーヤーのサポートが必要で、さらにハードウェアおよびOSのサポートが必要です。

ここでは一般的なライブストリーミング再生のセキュリティソリューションと特徴、および実施方法をご紹介します。

Referer認証

- **適用ケース**：ライブストリーミングのコンテンツセキュリティをあまり気にせず、再生の敷居を少し高くしたいだけであれば、ホットリンク防止のためのRefererを使用することができます。
- **実施方法**：CSSコンソール > ドメイン名管理 > 再生ドメイン名の選択 > アクセス制御 > Refererホットリンク防止をオンにしてReferer内容を入力します。
- **優位性**：簡単に実施でき、設定を制御すれば完了します。
- **存在する問題**：再生リクエスト中のRefererは修正、偽造できるため、Refererの制限内容が知られると、再生リクエスト中のRefererが修正され、サーバーのReferer制限が回避されてしまいます。

IPブラックリスト/ホワイトリスト

- **適用ケース**：再生を許可するまたは許可しないユーザーのパブリックネットワークの再生IPが明確であれば、IPブラックリストによって制限できます
- **実施方法**：Referer認証と同様に、アクセス制御内でIPブラックリスト/ホワイトリストを書き込みます。
- **優位性**：簡単に実施でき、コンソールで設定し、制限ポリシーが厳格で、偽造できません。
- **存在する問題**：再生を許可するまたは許可しないユーザーのパブリックネットワーク IPを容易に知ることができず、さらに変更される可能性があります。

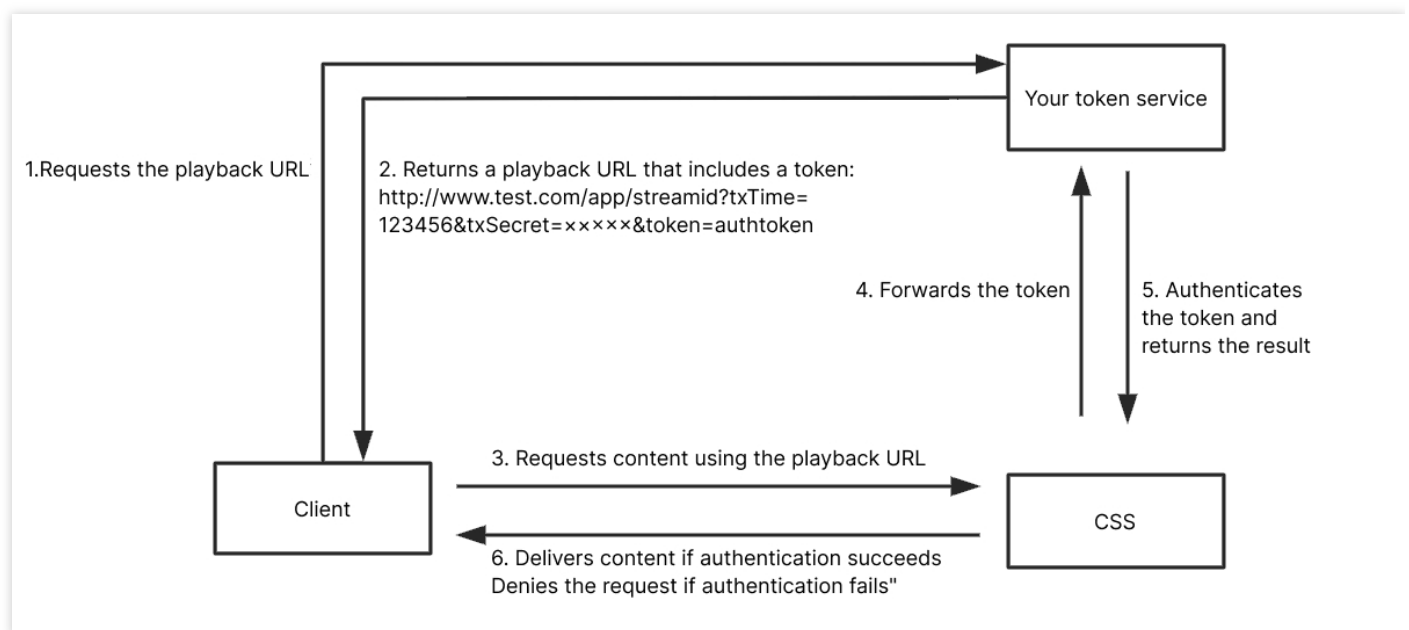
再生のホットリンク防止

- **適用ケース**：ライブストリーミングに特別なセキュリティ要件がない場合、盗用防止のみが必要です。
- **実施方法**：Referer認証と同様に、CSSコンソールでKey認証を有効化し、キーを書き込みます。このようにして生成された再生URLはtxTimeとtxSecretを含み、バックエンドでKeyによってtxSecretを復号して再生URLの合法性を検証します。具体的な原理と方法は[ホットリンク防止の計算](#)をご参照ください。
- **優位性**：簡単に実施でき、コンソールを開いて設定します。再生URLは前のドキュメントに従ってtxTimeとtxSecretを生成すれば完了です。
- **存在する問題**：有効期間内に、再生URLを取得した視聴者は再生することができます。再生URLは簡単に配布されるため、盗用される可能性があります。有効期間を長く設定すると、盗用の可能性が大きくなります。有効期間が短い場合は、URLを取得する時に期限切れになっている可能性があります。再生プロセスで誤って切断された場合、改めて再生する時は新しい再生URLを取得する必要があります。ネットワークが不安定な再生ユーザーは再生URLを頻繁に取得しなければならない可能性があります。一般的に再生URLの有効期間は24時間に設定することをお勧めします。

再生ホットリンク防止とリモート認証の組み合わせ

視聴者が不合理なチャンネルから再生URLを取得し、盗用することを防止するため、再生ホットリンク防止をベースに、クライアントによるカスタムtoken検証方式を追加し、安全性が高いホットリンク防止メカニズムを提供します。

- **ユースケース**：ライブストリーミング再生は、再生を許可するユーザー層を正確に制御する必要があります。例えば会員、ログインした視聴者などです。
- **実施方法**：Tencent Cloudで再生ホットリンク防止の検証に合格した後、クライアントの業務サーバーインターフェースを呼び出し、再生リクエストをユーザーの業務サーバーに転送します。業務サーバーはtoken情報を抽出して検証し、検証に合格したら検証結果をTencent Cloud CSSに返し、それによって再生を許可するかどうかを決定します。クライアント自身の検証を追加し、クライアント自身で再生を許可するかを制御することができます。具体的な方法は次のとおりです：



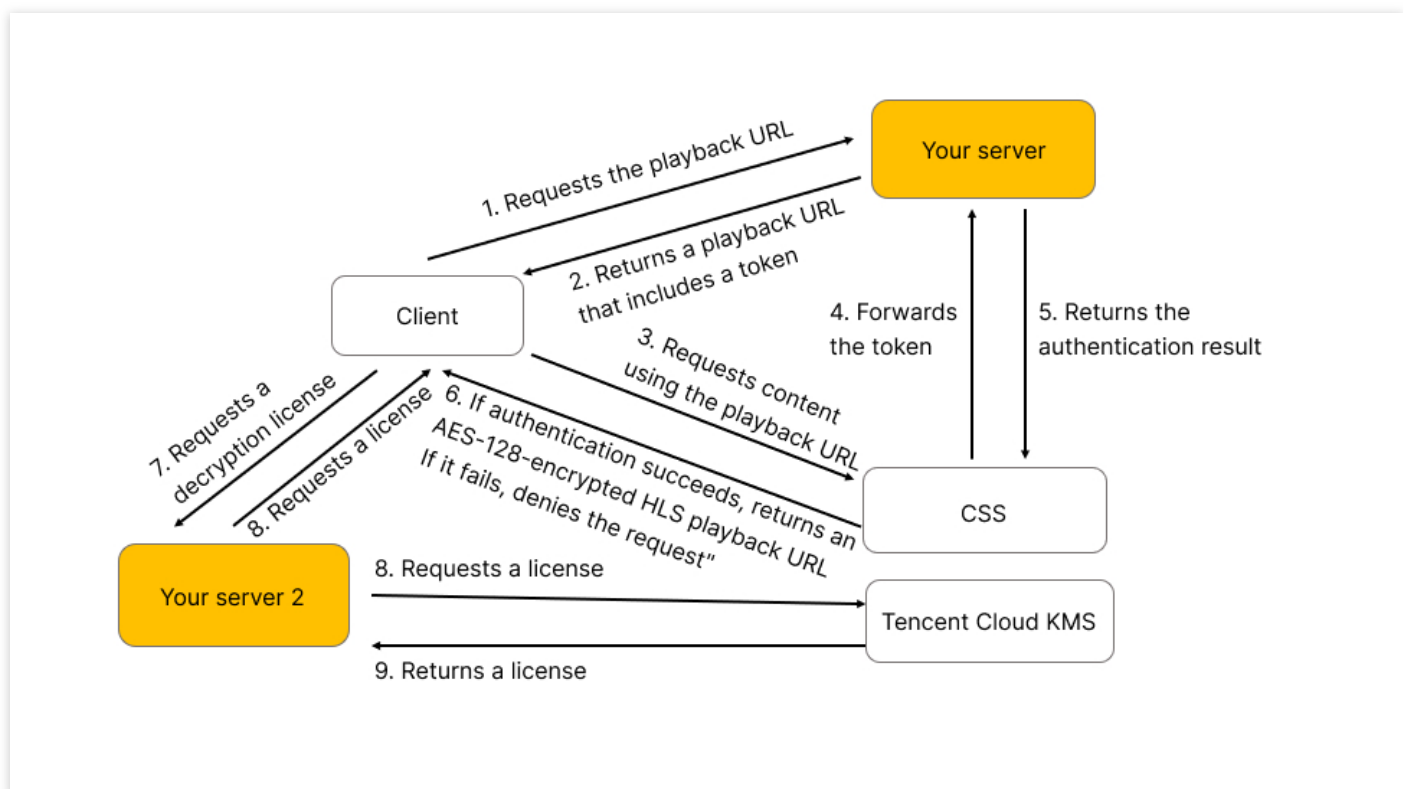
- i. クライアントサーバー（tokenサービス）に再生リンクを申請します。
- ii. クライアントサーバーはリクエストの合法性に応じて再生URLを発行します。再生URLにはTencent Cloudのホットリンク防止txSecretとクライアント自身が配布したtokenが含まれます。Tencent Cloudホットリンク防止は、例えば5分のように短く設定することができます。
- iii. Tencent Cloudホットリンク防止およびビジネスtokenを含む再生リンクを使用してTencent Cloud CSSをリクエストします
- iv. Tencent Cloud CSSはTencent Cloudホットリンク防止を検証し、合格するとクライアントtokenを含む再生URLを、クライアントのtokenサービスに転送します。
- v. クライアントのtokenサービスの検証に成功すると、HTTP statusによって検証結果が返信されます。200はOKで、その他はNGです。
- vi. Tencent Cloud CSSはtokenの検証結果に応じてライブストリーミングのコンテンツリソースをクライアントに返すかどうかを決定します。

- **実施方法**：再生ホットリンク防止を有効化してリモート認証を有効化します。具体的な実施プロセスについてはTencent Cloudの営業部門にご相談いただくか、チケットを提出してTencent Cloud CSSにお問い合わせください。
- **ソリューションの優位性**：クライアントはどのようなユーザーに再生を許可するかを自身で柔軟に制御することができます。
- **存在する問題**：クライアント自身でtoken配布サービスを開発する必要があります。ライブストリーミングコンテンツは暗号化されておらず、視聴者は合法的な再生リンクによってライブストリーミングコンテンツを記録、またはリアルタイムにライブストリーミングコンテンツを転送することができます。ハッカーがネットワークによってライブストリーミングコンテンツを捕捉する可能性があります。

HLS AES128によるストリーミングコンテンツの暗号化、再生ホットリンク防止を組み合わせたリモート認証

再生ホットリンク防止にリモート認証を組み合わせることをベースに、AES128暗号化アルゴリズムを使用してHLSのTSビデオフレームを暗号化します。

- **ユースケース**：HLSプロトコルを使用して再生し、ストリーミングメディアコンテンツが著作権、機密またはプライバシーに関わることで、ネットワーク上でハッカーから捕捉されないようにします。
- **実施方法**：HLS基準がサポートするAES128を使用してTSビデオストリームのビデオフレームを暗号化します。具体的なフローは次のとおりです：



- **実施方法**：再生ホットリンク防止を有効化してリモート認証を有効化します。具体的な実施プロセスについてはTencent Cloudの営業部門にご相談いただくか、チケットを提出してTencent Cloud CSSにお問い合わせください。

さい。

- **ソリューションの優位性**：暗号化方式はHLSの標準的方式で、HLSプロトコルをサポートしているプレーヤーは当然サポートしています。キーアクセスは成熟した製品によってサポートされ、再生の復号はプレーヤー内に内蔵されています。再生ホットリンク防止とリモート認証は組み合わせて使用することができ、安全性が高く、方式が成熟しており、統合が簡単です。
- **存在する問題**：HLS方式の再生シーンしかサポートしていない。

業界のDRMスキーム

主にAppleのFairplay DRM、GoogleのWidevine DRM、中国のChinaDRMなどのような国外および国内の認証済みDRMをサポートしていることを指します。その基本原理はオーソリティサーバーにDRM証明書を申請し、証明書によってID情報の管理および暗号化/復号フローにおけるキーおよびキーの使用制限を管理します。

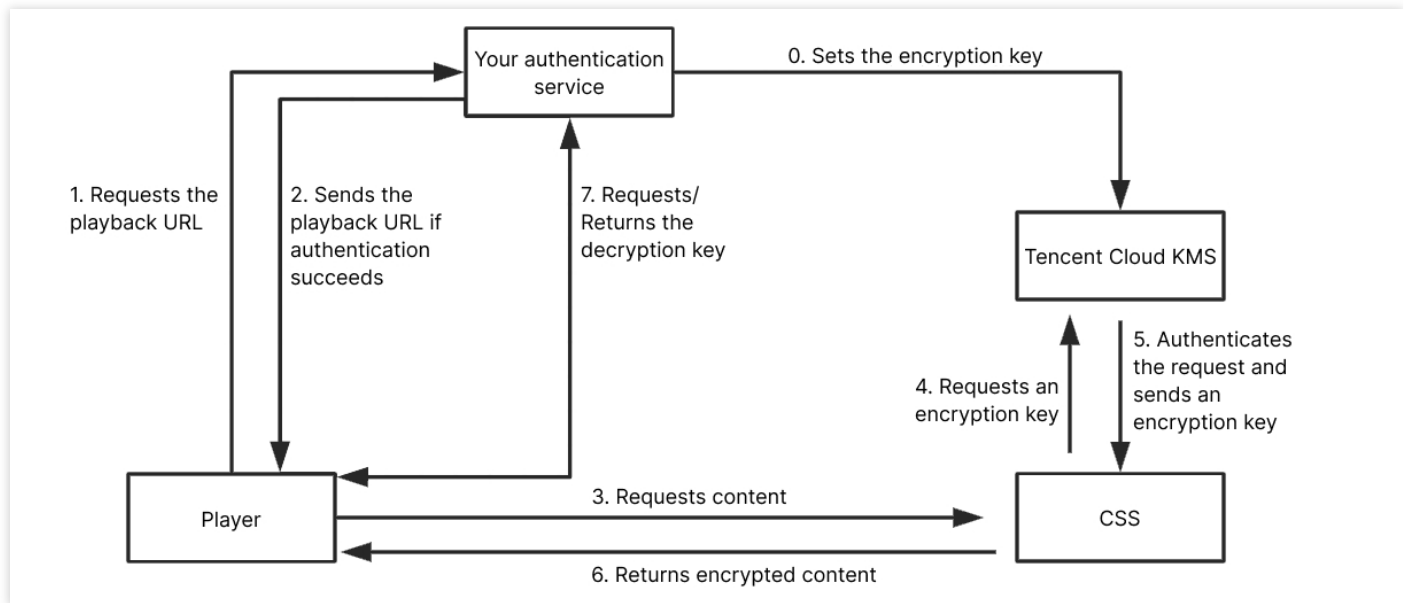
- **ユースケース**：著作権者には明確な制限性があり、承認された標準的DRMスキームの使用が求められます。
- **実施方法**：現在Tencent CloudはAppleのFairplay DRM、GoogleのWidevine DRMの2種類のDRMスキームのみをサポートしています。その他のDRMスキームは開発中です。
- **実施方法**：具体的な実施プロセスについてはTencent Cloudの営業部門にご相談いただくか、チケットを提出してTencent Cloud CSSにお問い合わせください。
- **ソリューションの優位性**：メディアの復号はOSまたはハードウェアによって完了し、セキュリティレベルが高いです。権限制御が柔軟で、例えば視聴者にビデオの最初の10分間だけ視聴を許可するなどの制御が可能です。
- **存在する問題**：指定のプロトコル、HLSまたはDashのみをサポートしています。実施するのが複雑で、基準に応じて証明書を申請する必要があるため、またプレーヤーの開発、DRMコンポーネントへの統合が必要です。主要なブラウザのサポートは不完全で、例えばFairplayはHLSプロトコルを使用する必要があり、かつiOSおよびmacOSプラットフォームの再生のみをサポートしています。WidevineはFirefoxなどの非Chromiumカーネルブラウザの再生をサポートしていますが、別途CDMモジュールをダウンロードする必要があり、全体的にスムーズではありません。ブラックボックス部分が多く（ライセンスのリクエストプロセスと暗号化/復号プロセス）、デバッグと互換性の面で手間がかかります。

Tencent Cloudが自社開発したストリーミングコンテンツの暗号化方式

大部分のプライバシーライブストリーミングまたはコンテンツセキュリティが必要なライブストリーミングはハードウェアレベルのセキュリティ、および複雑な証明書配布による検証プロセスが必要ありません。さらに国内のライブストリーミングでは、FLVライブストリーミング方式も流行しています。FLVに対する安全なライブストリーミングのソリューションが求められています。

- **ユースケース**：FLVプロトコルを使用して再生する状態で、ストリーミングコンテンツを暗号化し、ネットワーク上でハッカーから捕捉されないようにします。
- **実施方法**：Tencent Cloud CSSが自社開発したストリーミングの暗号化方式です。クライアントはストリーミングに暗号化するキーを設定し、Tencent Cloud CSSはキーを読み取って指定のストリーミングを暗号化します。ユーザーが再生する時は、クライアントがまずユーザーを認証し、その後クライアント端末に復号キーを配布

し、ビデオストリームを復号します。プロセスはHLS AES128と類似しています。
 自社開発の暗号化/復号のフローは次のとおりです：



- **実施方法**：具体的な実施プロセスについてはTencent Cloudの営業部門にご相談いただくか、チケットを提出してTencent Cloud CSSにお問い合わせください。
- **ソリューションの優位性**：すべてのプロセスが制御可能で、キーと暗号化/復号は製品とツールによってサポートされています。Tencent CloudはPlayer+を提供し、統合しやすく、ソリューションが成熟しています。
- **存在する問題**：SDKを統合する必要があり、クライアントの自社開発のプレーヤーしかサポートしていません。Web端末およびブラウザは再生できません。

ライブストリーミング再生のセキュリティ上の特徴および実施方法の長所と短所の比較

| セキュリティソリューション | セキュリティソリューション | 特徴 | 存在する問題 | 実施の複雑度 |
|---------------|---------------|----|--------|--------|
|---------------|---------------|----|--------|--------|

| セキュリティソリューション | セキュリティソリューション | 特徴 | 存在する問題 | 実施の複雑度 |
|-------------------|-------------------------|-------------------------|--|--------|
| Referer認証 | HTTPリクエストのRefererを認証します | 簡単に実施でき、設定するとすぐに有効化されます | Referer内容を偽造できます | 非常に簡単 |
| IPブラックリスト/ホワイトリスト | アクセスするIPを認証します | 簡単に実施でき、設定するとすぐに有効化されます | 視聴者のIPは簡単に取得できず、頻繁に変更される可能性があります | 非常に簡単 |
| 再生ホットリンク防止 | 再生URLを暗号化し、URLを認証します | 簡単に実施でき、設定するとすぐに有効化されます | ホットリンク防止の有効期間を制御するのは難しく、長すぎるとリンク不正アクセスされやすく、短すぎると頻繁に取得する必要があります。 | 非常に簡単 |

| セキュリティソリューション | セキュリティソリューション | 特徴 | 存在する問題 | 実施の複雑度 |
|-----------------------|---|--|--------|--------|
| 再生ホットリンク防止+リ モート認証 | 再生URLを暗号化し、URLを 認証し、クライアント自身で の認証プロセスを提供します | 実施 しや す く、 不正 取得 が困 難で あ り、 token 認証 の開 発に つ い てク ライ アン トが 協 力 す る 必 要 が あ り ま す | - | 簡単 |

| セキュリティソリューション | セキュリティソリューション | 特徴 | 存在する問題 | 実施の複雑度 |
|---------------|--|--|---|--------|
| HLS暗号化 | 再生URLを暗号化すると同時に、ストリーミングコンテンツを暗号化します | 実施しやすく、クライアントはtoken認証の開発を組み合わせる必要があります | HLSプロトコルのみをサポート、FLVは使用できません | 簡単 |
| 業界のDRMスキーム | コンテンツを暗号化し、再生URLも暗号化でき、ソフトウェアおよびハードウェアレベルのセキュリティ保障をサポートします | やや複雑で、プレーヤーの開発が必要であり、互換性の問題も若干多くなっています | 実施が複雑で、互換性の問題が多く、iOSおよびAndroidは別々にサポートする必要があります | 複雑 |

| セキュリティソリューション | セキュリティソリューション | 特徴 | 存在する問題 | 実施の複雑度 |
|---------------|-----------------|----------------------------------|--|--------|
| 自社開発の暗号化方式 | コンテンツと再生URLの暗号化 | やや複雑で、プレイヤーの開発が必要ですが、全フローが制御可能です | プライベートソリューションは、Tencent Cloudの通信プロトコルに準拠する必要があります。ブラウザ未対応です | 複雑 |

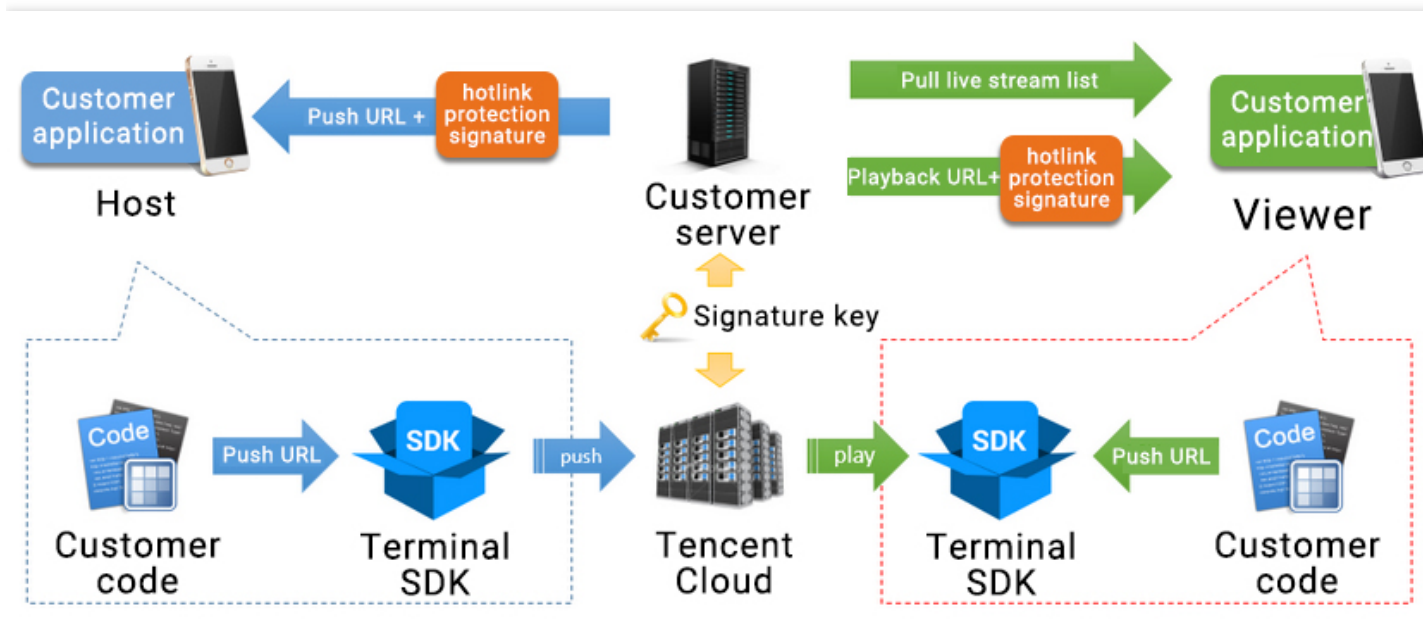
ホットリンク防止の計算

最終更新日：：2023-04-25 14:42:34

リンク不正アクセス防止は、プッシュURLと再生URLのtxSecretフィールドを参照します。攻撃側がバックエンドを偽造してプッシュURLを作成したり、利益のために再生アドレスを不正に利用したりすることを防ぐ機能があります。

セキュリティ原理

下図に示すとおり、攻撃者がサーバーを偽造してプッシュおよび再生URLを生成するのを防止するために、CSS管理コンソールでリンク不正アクセス防止暗号化KEYを設定し、攻撃者が有効なプッシュと再生URLを容易に生成できないようにします。



計算プロセス

手順1：認証キー

まず、公式サイトでコンソールで暗号化キーを設定する必要があります。この暗号化キーはサーバーでリンク不正アクセス防止署名を生成するために使用されます。Tencent Cloudは同じキーを保有しているため、生成されたリンク不正アクセス防止署名を復号化して確認できます。

暗号化キーは、プッシュリンク不正アクセス防止Keyまたは再生リンク不正アクセス防止Keyに区分され、前者はプッシュリンク不正アクセス防止URLの作成に使用され、後者は再生リンク不正アクセス防止URLの生成に使用されます。**CSSコンソール**>**ドメイン名管理**に移動し、対応するドメイン名または**管理**をクリックし、**プッシュ設定**を選択すると、下図のようなプッシュリンク不正アクセス防止KEYが自動的に設定されます。

Generate Push URL



Push Hotlink Protection Key ⓘ [Redacted]

Push Callback URL [Redacted]

Push Domain [Redacted]

Expiry Time 2020-05-11 [Calendar icon] 23:59:59 StreamName livestestream **Generate Push URL**

Stream URL `rtmp://[Redacted]/live/liveteststream?txSecret=[Redacted]&txTime=5EB9767F` [Copy icon] [QR icon]

再生リンク不正アクセス防止Keyの関連情報については、[再生リンク不正アクセス防止はどのように有効化するのはですか?](<https://www.tencentcloud.com/document/product/267/35598>)をご参照ください。

手順2：txTimeの生成

署名の平文部分はtxTime（時間単位：秒）であり、このリンクの有効期限を意味します。例えば、現在の時刻が 2018-12-29 11:13:45 であり、新たに生成するURLを3時間後に無効にしたい場合は、txTimeを 2018-12-29 14:13:45 と設定することができます。

ただし、このように長い時間文字列をURLに含めることは困難であるため、実際の使用においては、2018-12-29 14:13:45 をUNIXタイムスタンプ1546064025（別のプログラミング言語で時間関数を直接呼び出して変換処理することもできます）に変換し、その後、16進数文字列に変換することで更に文字数を短縮します。つまり、txTime=1546064025（10進数）=5C271099（16進数）であり、10進数の使用もサポートされています。

注意：

- 実際の最終終了時間は `txTime + 認証有効期間` であり、認証有効期間の変更はURL発行に影響されません。
- txTime期限は長すぎず、短すぎない時間で設定することを推奨します。
 - 期限切れまでの時間が短すぎると、カスタマーがライブストリーミング中にネットワークが途切れた場合に、プッシュURLの期限切れによってプッシュを再開することができません。
 - 期限切れまでの時間が長すぎると、プッシュが盗用されるリスクがあります。

海外のCSSサービス

海外のCSSサービス紹介

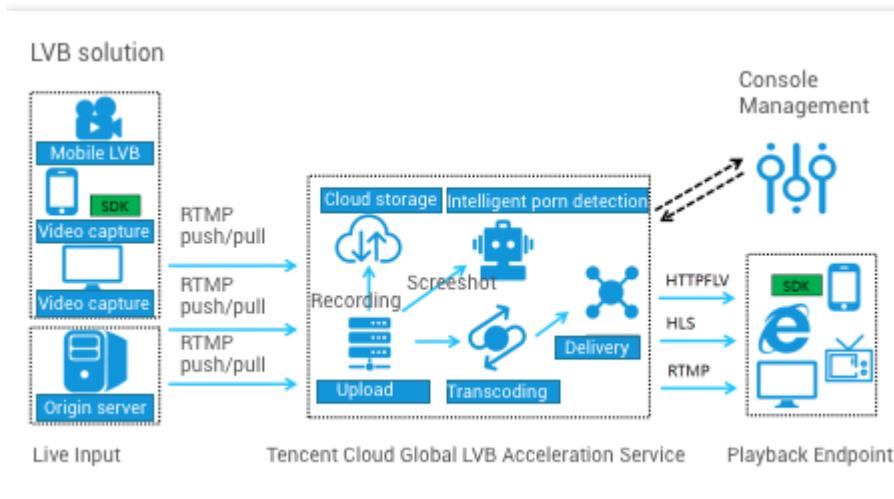
最終更新日：：2022-05-17 11:05:47

ここ数年来、オーディオビデオ基盤技術の成熟にともない、ライブストリーミング業界が世界的に爆発的な成長を見せています。中国のインターネットプラットフォームの海外進出も珍しくなくなり、中国のサービス系製品が海外に進出したことにより、ライブストリーミング業界にも海外にチャレンジするための確かな土台が築かれています。中国国内ですでに突出した影響力をもつビデオライブストリーミング企業は、競争の優位性を高めるため、製品のグローバル化発展計画を実施し、また一部の小規模なライブストリーミングプラットフォームは、国内での生き残りが厳しいため、別の手段として海外に活路を見出そうとしています。これと同時に、海外のライブストリーミングプラットフォーム市場でも激しい争いが繰り広げられ、YouTube、Periscope、Facebookの三大巨頭が市場を攻略していますが、まだ数多くの開発されていないブルーオーシャン市場が、中小規模ライブストリーミングプラットフォームにも残されています。これらに基づき、Tencent Cloudでは、海外ライブストリーミング分野で、引き続きリソースのストックを拡大し、絶えずそのアクセラレーション性能を最適化して、ライブストリーミングプラットフォームの海外市場開拓を支援していきます。



全体的なライブストリーミングサービスには、プッシュと視聴の外にも、認証、トランスコーディング、スクリーンキャプチャ、レコーディング、コールバック、ポルノ検出、DRMなどの機能が含まれるべきです。下図は

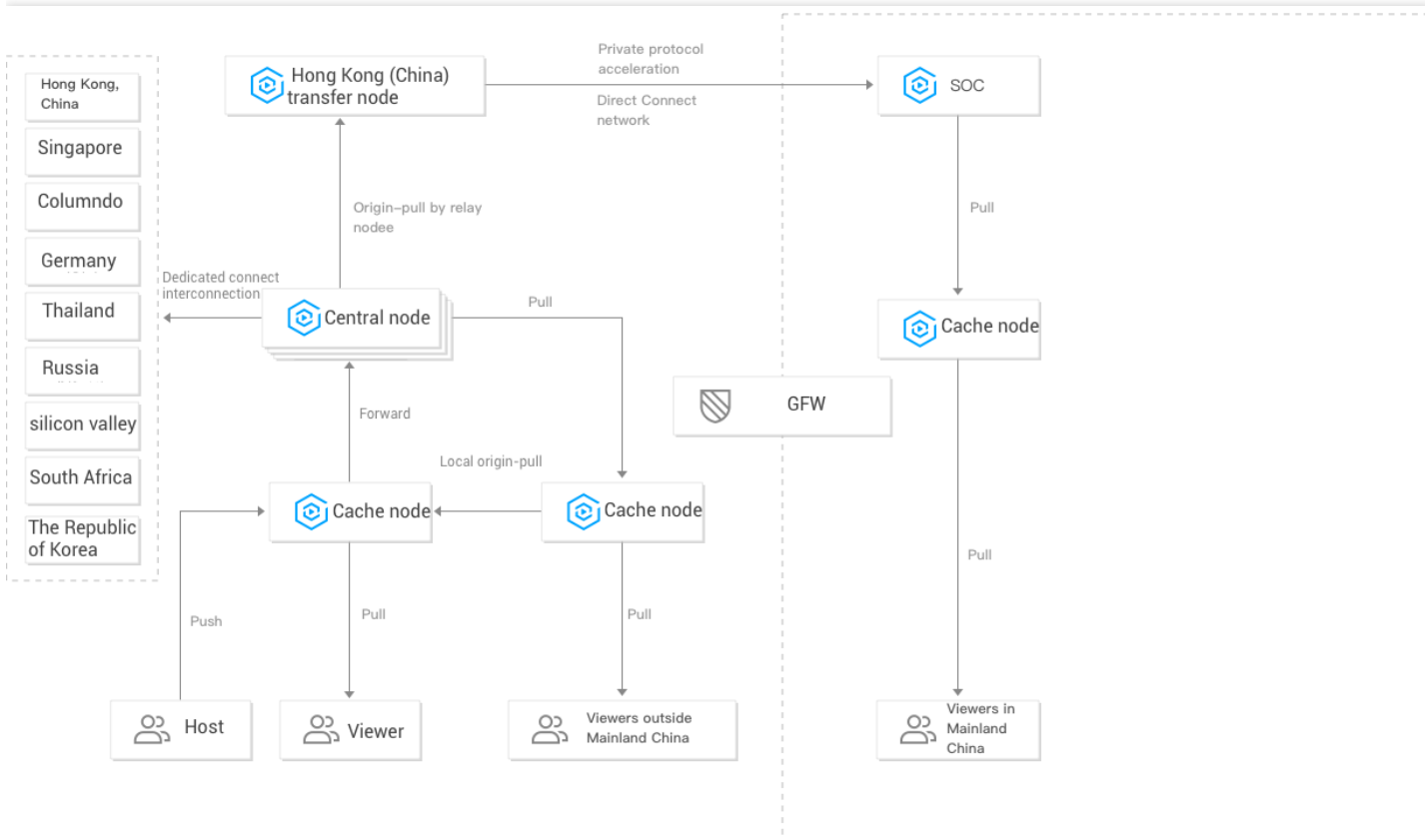
Tencent Cloudの海外ライブストリーミングソリューションの基本機能のモジュールです。



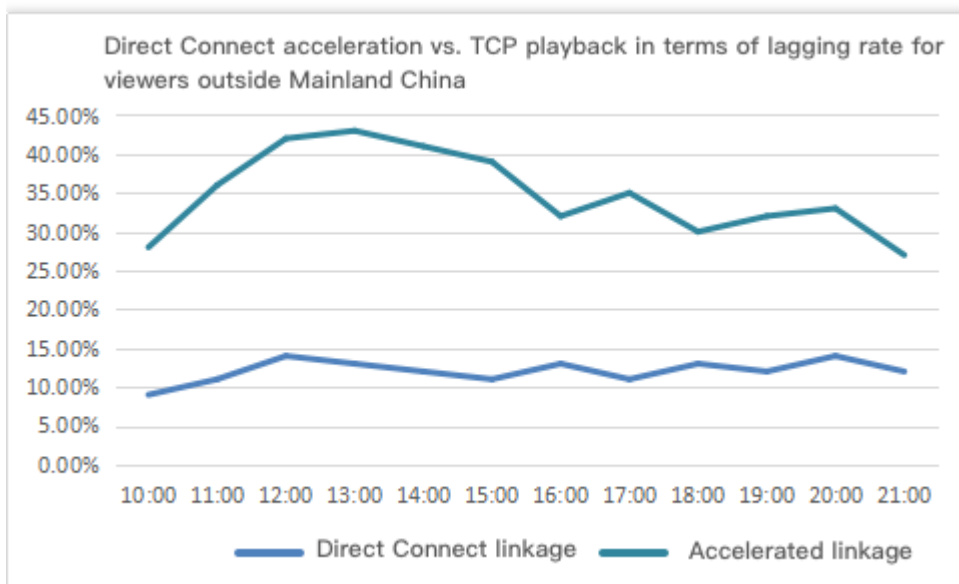
基本機能については、海外のライブストリーミングと中国のライブストリーミングのニーズは一致していますが、海外にはより多くの課題が存在します。これは主に海外のエリアの広さ、海外の国々の内部ネットワークの煩雑さ、国を跨ぐネットワークの品質のバラツキなどに起因します。ディレイを軽減し、ラグを減少させ、安定した信頼できるサービスを提供するため、Tencent Cloudでは、海外のライブストリーミングシナリオに対して、フレームワーク、ネットワーク、セキュリティ、リソース等の分野に特化して最適化を行っています。

多数のセンターの展開

Tencent Cloudは、中国香港、タイ、シンガポール、ドイツ、トロント、シリコンバレー、ロシア、南アフリカ、韓国などに、多数のセントラルデータセンターをすでに建設し、セントラルデータセンターを建設する国のリージョンのカバー範囲を徐々に拡大させています。セントラルデータセンターには、海外のライブストリーミングに必要な全てのモジュールが含まれ、同時にグローバルユーザーにサービスを提供するため、いかなるデータセンターの異常にも切り替えできることを保証し、すべてのフレームワークの分散化を要求しています。国を跨ぐネットワークの品質と安定性がライブストリーミングのレイテンシーとラグに与える影響を考慮し、海外に建設するセントラルノードは専用線のインターネットを利用し、中国国内と海外は、中国香港センターの中継ノードの専用線のインターネットを利用しています。全体構成（略図）は以下のとおりです。

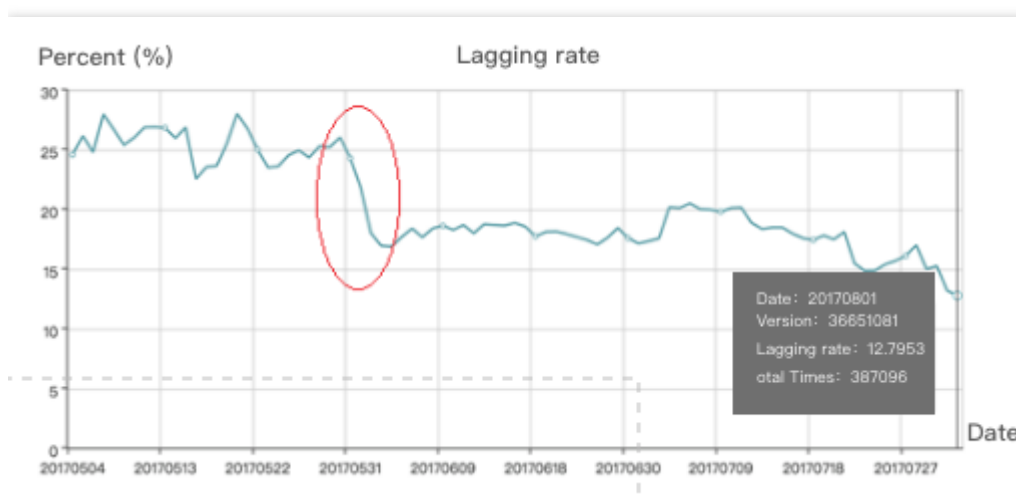


海外セントラルノードのアクセラレーション効果を直感的に表現できるように、中国国内のユーザーがアメリカのキャストのビデオを視聴するケースを対象に統計と比較を行っています。効果は下図のようになっており、専用線アクセラレーションのラグ率が低く、ネットワークも安定していることが明らかにわかります。



エッジロケーションのアクセラレーション

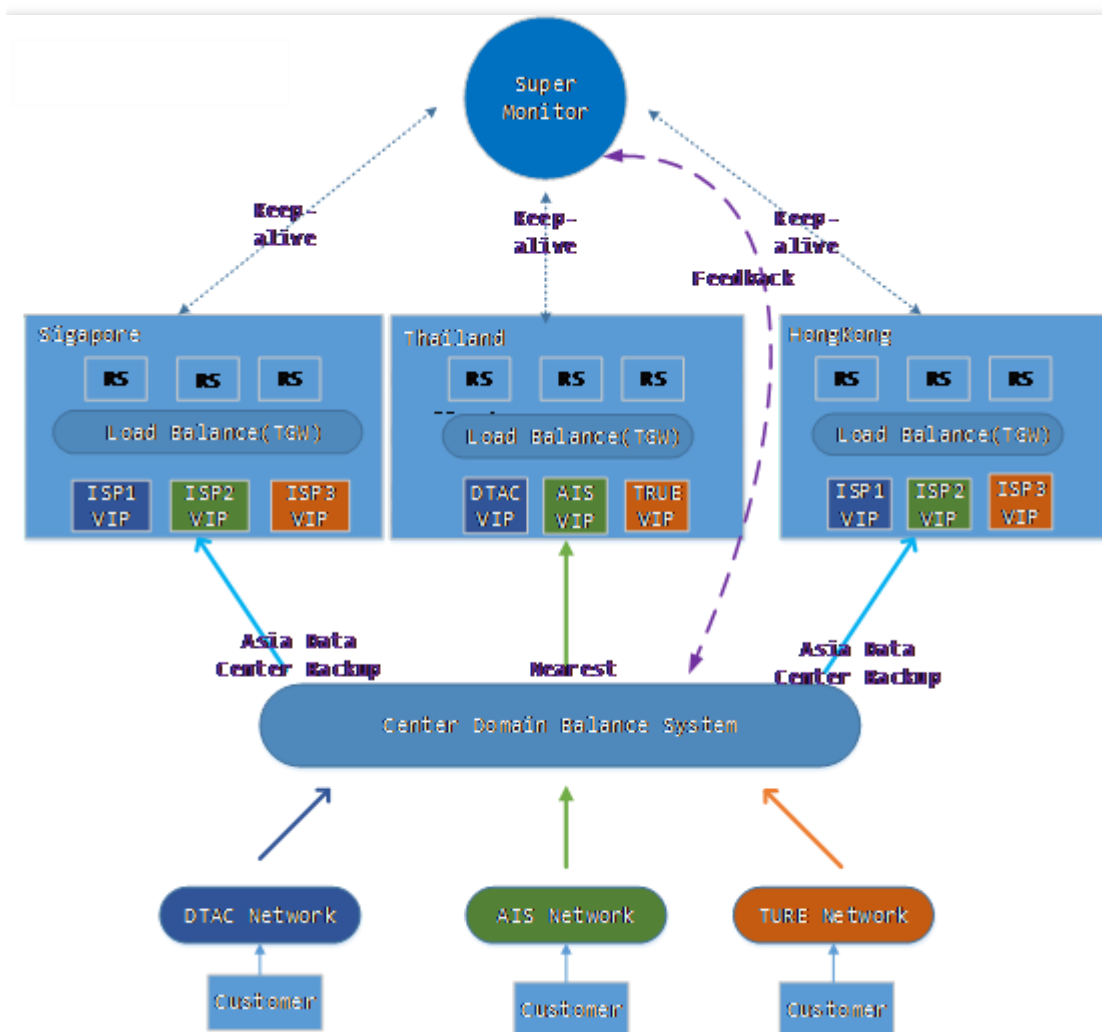
セントラルノードによって当該地区のユーザーのニーズはしっかりとカバーできますが、センターリージョン以外にいる数多くの顧客のニーズも見過ごすことはできません。条件的な制約により、これらの地域の国にはセントラルノードが建設されておらず、アクセラレーションノードを増やす必要があります。通常、このような地域の国を跨ぐネットワークの品質はかなり劣り、クロスリージョンのプルのラグ率も高くなります。これらの地域をエッジロケーションと呼んでいますが、マレーシア、インドネシア、中東、インド、アフリカ、南米などの国が該当します。これらのエッジロケーションでは、Tencent Cloudは、サービスのモジュールに対して優先度の取捨選択を行い、優先的にローカルの合法的ユーザーの視聴を保護し、当該エリアのデータが国を跨ぐ必要なく完成できるように保証しています。その他モジュールのサービスについては、エッジサーバーからセントラルノードにプッシュ転送し、最終的にセントラルノードで完成させます。下図より、エッジロケーションでローカルのノードアクセラレーションを立ち上げた後、ラグ率が明らかに低減し、業界内のその他メーカーのアクセラレーションレベルを遥かに上回っていることがわかります。



最適なアクセスと障害復旧切り替え

海外のほとんどの国が実は中国大陆と類似しており、1つの地区に複数のキャリアが存在し、複数のキャリア間のアクセスは帯域幅とリソースの影響を受け、一定の制限があります。例えば、タイのDTAC、AIS、TURE、中国台湾の中華電信、台湾モバイル、so-net、インドネシアのTelkomsel、XL、INDOSATなどです。これらのキャリアのアクセスにおけるユーザーエクスペリエンスを向上させるためには、スケジューリングシステムによって各キャリアが可能な限り同一ISPにアクセスする問題を解決する必要があり、同時に私たちが当該エリアに建設するアクセラレーションノードも可能な限りBGPにアクセスし、現地の関連キャリアとのpeering linkを行います。タイを例に説明しましょう。3つの異なるキャリア（DTAC、AIS、TURE）のユーザーを対象に、センターのスケジューリングシステムが膨大な量の海外のIPおよびキャリアを記録し、かつユーザーのIPに応じて最も近いCDNノードに自動スケジューリングします。その識別の精度は**99.5%**以上にもなります。同時にデータセンター異常時の切り替えもサポートし、ノードを監視して、あるリージョンに異常が発生したことを検出すると、システ

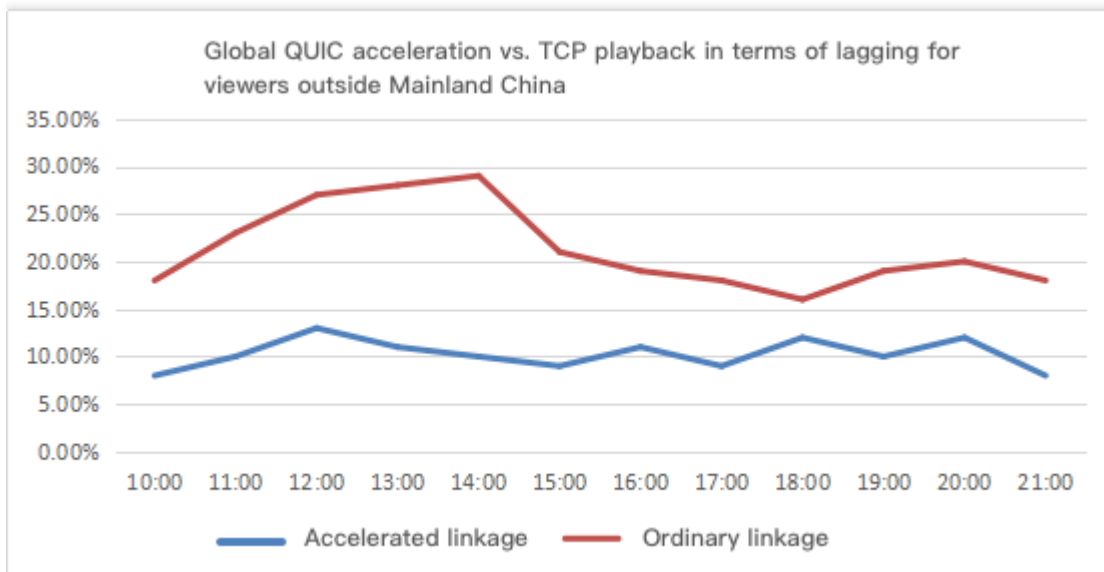
ムが自動的に最適なデータセンターを選択して切り替えを行います。



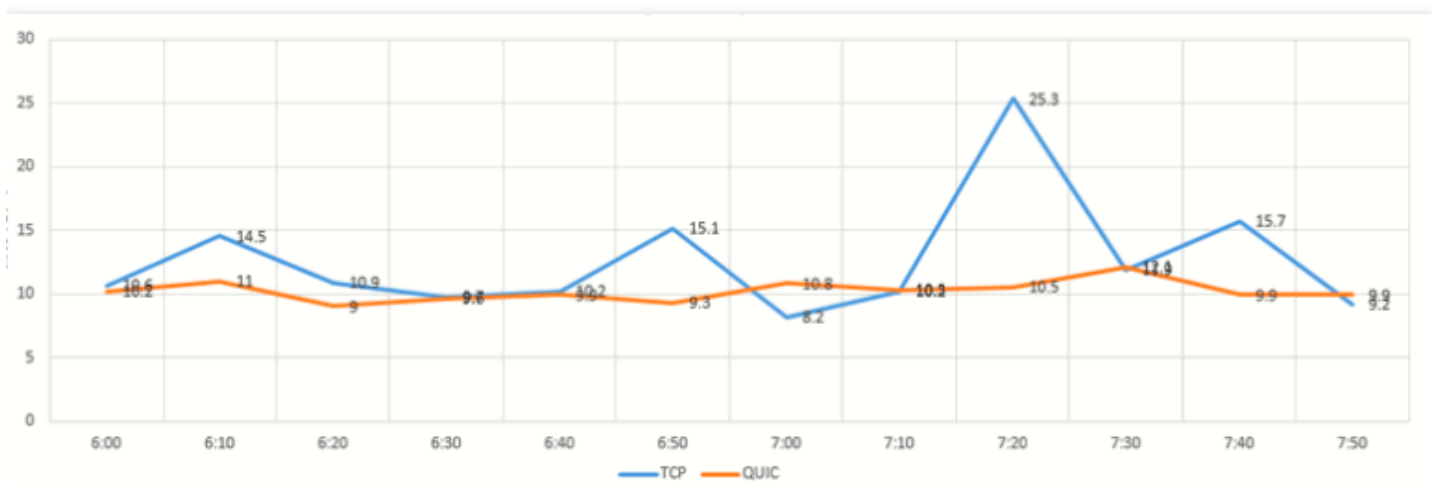
ネットワーク伝送の最適化

海外のクロスリージョン外部ネットワークで伝送する時は、従来のTCP伝送ではトランスポートのレイテンシーを保証できません。海外では伝送距離が長い為、インターナショナルなエクスポートの帯域幅に制限があり、ネットワーク品質の変動もかなり頻繁に起こります。TCPはこのようなシナリオにおいて、そのプロトコルの特性から、アップデート/最適化の周期が長い、高パケットロスシナリオでのパフォーマンス不良などの問題が存在します。Tencent CloudはQUIC（UDPによる信頼性の高いデータ伝送を実現）を採用し海外ネットワーク伝送の最適化を行っています。QUICによって上位層のデータの代理アクセラレーションを行い、アプリケーション層で実現するため、パラメータの調整または輻輳制御のアルゴリズムの調整で速やかに効果があり、アルゴリズム・パラメータを調整して効果的に高レイテンシーと高パケットロスシナリオに対処することができています。同時にHoLブロッキングの問題を回避でき、RTTの時間の消耗を低減できます。実際のデータで計算して、従来のTCPと

比較すると、最適化プランの接続時間は平均40%減、ラグ率も平均20%削減できています。



下図はアラブ首長国連邦のキャスターがアラブ首長国連邦のアクセラレーションノードにプッシュしたもので、グローバルユーザーがこのキャスターを視聴するラグ率を抜き取り、比較しています。比較データから見ると、QUICアクセラレーションのストリーミングラグ率は非常に安定していることがわかります。



莫大なリソースのストック

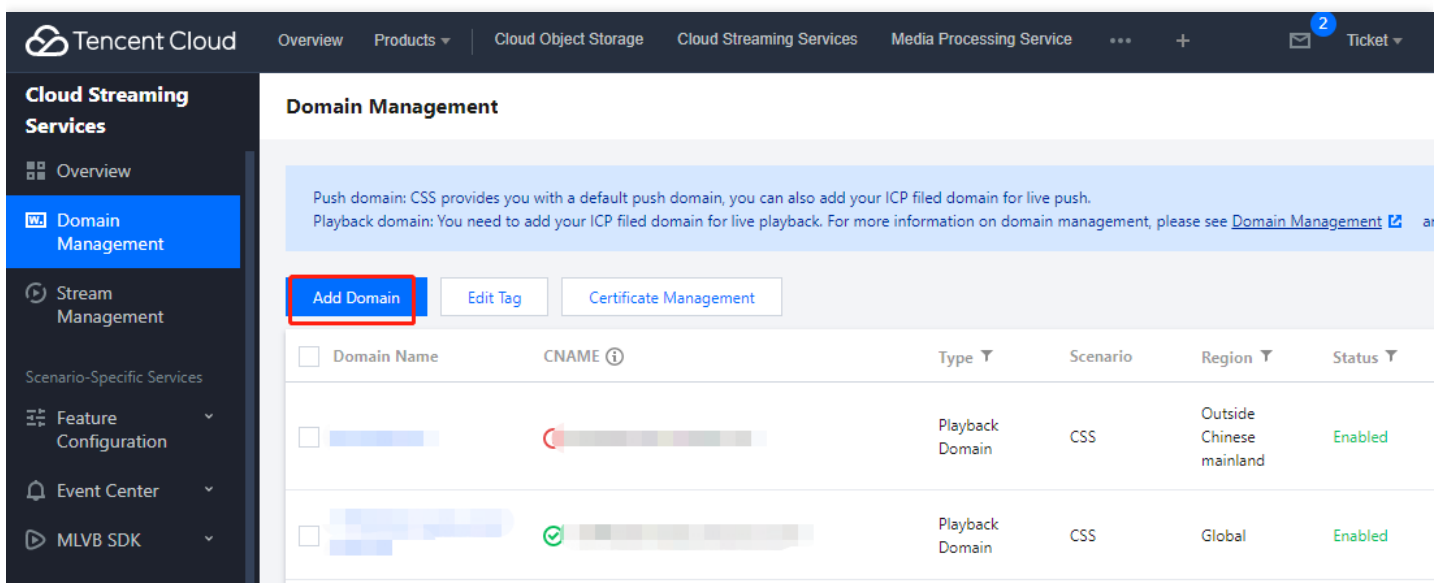
ライブストリーミングにおいて、結合技術のフレームワークとソリューション以外に、より重要な点は、リソースのストックです。海外に基盤を支える該当するリソースがなければ、すべての技術は机上の空論です。文章の冒頭のTencent Cloudグローバルノード分布図の中でも言及しましたが、**Tencent Cloudの海外進出戦略と長期的な海外での投資のおかげで、世界の50余りの国と地域に1300強の伝送ノードを建設し、帯域幅の総ストック量は100Tを超えています。50社を超える全世界のキャリアと提携し、Global Content Deliveryのノードも200+となっています。**同時にTencent Cloudは同じ地区で複数のキャリアと提携し、各地点に少なくとも3つのマルチサイトを設け、サービスの安定性と信頼性を保証しています。

アクティブ化の方式

海外CSSサービスは、[CSSコンソール](#)から直接アクティブ化が可能です

- Tencent Cloudアカウントを未登録の場合は、Tencent Cloudアカウントの登録を行っていただく必要があります。登録の操作については、[Tencent Cloudアカウントの登録](#)ドキュメントをご参照のうえ、[CSSサービスアクティブ化の申請](#)を行ってください。
- Tencent Cloudアカウントをお持ちで、CSSサービスのアクティブ化を申請済みの場合は、以下の操作手順を直接行うことができます。

CSSコンソールに入り、左側ナビゲーションバーから【[Domain Management](#)】選択し、その後【ドメイン名の追加】をクリックします。



The screenshot shows the 'Domain Management' page in the Tencent Cloud console. The left sidebar contains navigation options: Overview, Domain Management (selected), Stream Management, and Scenario-Specific Services (Feature Configuration, Event Center, MLVB SDK). The main content area has a header 'Domain Management' and a blue informational banner. Below the banner are three buttons: 'Add Domain' (highlighted with a red box), 'Edit Tag', and 'Certificate Management'. A table lists domain configurations:

| <input type="checkbox"/> | Domain Name | CNAME ⓘ | Type ▾ | Scenario | Region ▾ | Status ▾ |
|--------------------------|-------------|------------|-----------------|----------|--------------------------|----------|
| <input type="checkbox"/> | [Redacted] | [Redacted] | Playback Domain | CSS | Outside Chinese mainland | Enabled |
| <input type="checkbox"/> | [Redacted] | [Redacted] | Playback Domain | CSS | Global | Enabled |

ポップアップしたドメイン名追加のボックスの中でタイプ【再生ドメイン名】を選択し、さらに対応する【アクセラレーションリージョン】を選択します。最後にアクセラレーションが必要な【ドメイン名】を入力すれば完了です。

HttpDNSスケジューリング

最終更新日：2022-10-14 16:26:05

ソリューションの背景

CSSの海外のプッシュと再生のスケジューリングは、デフォルトではドメイン名のDNS解決を利用してスケジューリングを行います。これは最も一般的で、最も簡単なアクセス方式です。国内外のネットワーク環境は複雑なため、ドメイン名解決エラーまたはトラフィックのクロスネットワークの問題が普遍的に生じています。CSSではHTTPDNSソリューションを使用し、ライブストリーミングのスケジューリングを最適化することをお勧めします。

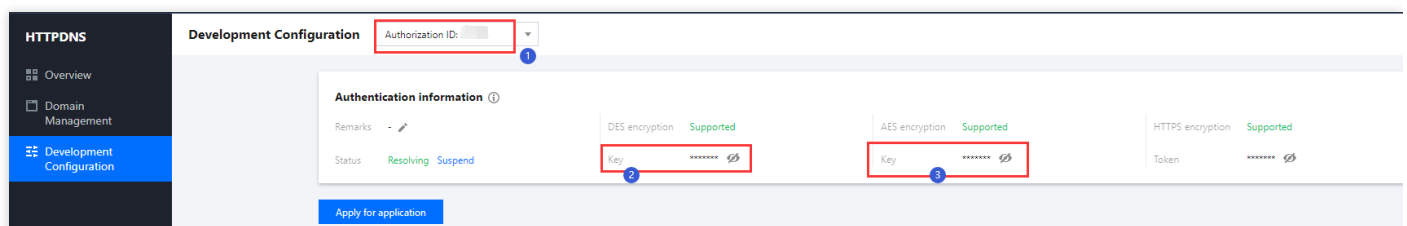
キャリアのLocalDNSの出口では、認証DNSのターゲットIPアドレスに基づきNATを行うか、または解決リクエストをその他DNSサーバーに転送するため、認証DNSがキャリアのLocalDNS IPを正確に識別できなくなる状況が起き、ドメイン名解決エラーやトラフィックのクロスネットワークなどの問題が引き起こされます。Tencent Cloud HTTPDNSは、世界最先端のDNSクラスター技術を備え、マルチキャリアとカスタマイズパスに対応し、スケジューリングを最適化することができます。

説明：

ここではどのようにHTTPDNSスケジューリングソリューションをTencent Cloudの国内および海外のCSSプッシュおよび再生のスケジューリングアクセラレーションに使用するかを説明します。HTTPDNSインタフェースについては [モバイルHTTPDNSの解決](#) をご参照ください。

事前準備

1. HTTPDNSサービスをアクティブ化するには、Tencent Cloudのモバイル解決HTTPDNSコンソールのサービス [アクティブ化の手順](#) をご参照ください。
2. [開発設定ページ](#) に進み、承認ID、DESキーの認証情報を確認します。



HTTPDNSを使用したアップストリームプッシュのスケジューリング

上りアクセスポイントIPのリクエスト

HTTPDNSリクエスト： `http://119.29.29.98/d?dn={$push_domain DES暗号化文字列}&ip={$ip DES暗号化文字列}&id=$id`、HTTP Getリクエストです。パラメータの意味は次のとおりです。

- `push_domain` プッシュドメイン名を表し、このフィールドはDESによって暗号化する必要があり、キー情報は [HTTPDNS開発設定ページ](#) から取得します。具体的な内容は [DESによる暗号化および復号の説明](#) をご参照ください。
- `IP` フィールドは、リクエスト側のパブリックネットワークの `egress IP` を表し、このIPに、最終的にスケジューリングするアクセスポイントIPが存在するリージョンとキャリアが示されます。このフィールドも同様にDESによって暗号化する必要があります。
- `id` フィールドはユーザー認証IDを表し、各ユーザーを一意に識別します。

アクセスポイントIPの復号

HTTPDNSによって取得したデータはDES暗号文です。DESによって復号し、`server_ip`を取得する必要があります。具体的な内容は [DESによる暗号化および復号の説明](#) をご参照ください。

アップストリームプッシュURLの結合

ここでの `server_ip` はリクエストした上りアクセスポイントIPから取得できるIPとなり、したがって結合したプッシュURLは次のとおりになります。 `rtmp://server_ip/live/streamname?`

`txTime=xxx&txSecret=xxx&txHost=domain`、最も重要なことは、既存のプッシュパラメータに、業務のプッシュドメイン名を意味するフィールドである `txHost` を新たに追加することです。

HTTPDNSを使用したダウンストリーム再生のスケジューリング

下りアクセスポイントIPのリクエスト

HTTPDNSリクエスト： `http://119.29.29.98/d?dn={$domain DES暗号化文字列}&ip={$ip DES暗号化文字列}&id=$id`、HTTP Getリクエストです。パラメータの意味は次のとおりです。

| フィールド | 意味 |
|--------------------------|--|
| <code>push_domain</code> | 再生ドメイン名です。このフィールドはDESによって暗号化する必要があり、キー情報は HTTPDNS開発設定ページ によって取得します。具体的な内容は DESによる暗号化および復号の説明 をご参照ください。 |

| フィールド | 意味 |
|-------|---|
| ip | リクエスト側のパブリックネットワークのegress IPを表し、このIPに、最終的にスケジューリングするアクセスポイントIPが存在するリージョンとキャリアが示されます。このフィールドも同様にDESによって暗号化する必要があります。 |
| id | ユーザー認証IDで、各ユーザーを一意に識別します。 |

アクセスポイントIPの復号

HTTPDNSによって取得したデータはDES暗号文です。DESによって復号し、server_ipを取得する必要があります。具体的な内容は[DESによる暗号化および復号の説明](#)をご参照ください。

ダウンストリーム再生URLの結合

- **HTTP**：FLVおよびHLSの再生プロトコルが含まれ、ここでのserver_ipは、リクエストした下りアクセスポイントIPの中から取得できるIPとなり、play_domainは再生ドメイン名を表します。したがってHTTPの再生URLの結合は次のようになります。

```
http://server_ip/play_domain/live/streamname.flv?xxxxxxxxxx
http://server_ip/play_domain/live/ streamname.m3u8?xxxxxxxxxx
http://server_ip/play_domain/live/ streamname -123.ts?xxxxxxxxxx
```

- **HTTPS**：FLVおよびHLSの再生プロトコルが含まれ、ここでのserver_ipは、リクエストした下りアクセスポイントIPの中から取得できるIPとなり、play_domainは再生ドメイン名を表します。HTTPSの結合ルールはプレーヤーのロジックに依存し、TCPで作成し、接続したターゲットIPをHTTPDNSスケジューリングのserver_ipにすることが要求されます。具体的にリクエストする再生URLに必要とされるのは通常の再生リクエストです。

```
https://server_ip/play_domain/live/ streamname.flv?xxxxxxxxxx
https://server_ip/play_domain/live/ streamname.m3u8?xxxxxxxxxx
https://server_ip/play_domain/live/ streamname -123.ts?xxxxxxxxxx
```

- **RTMP**：ここでのserver_ipは、リクエストした下りアクセスポイントIPの中から取得できるIPとなり、play_domainは再生ドメイン名を表します。したがってRTMPの再生URLは以下のように結合します。

```
rtmp://server_ip/play_domain/live/ streamname?xxxxxxxxxx
```

説明：

HTTPDNSリクエストは、HTTPDNSアクセスがタイムアウトした、返された結果が非IP形式であった、または空で返されたなど、低確率で異常が生じます。このような低確率な異常イベントが発生した場合は、LocalDNSにアクセスしてドメイン名の解決を行ってください。

コールバックによるイベントメッセージ通知

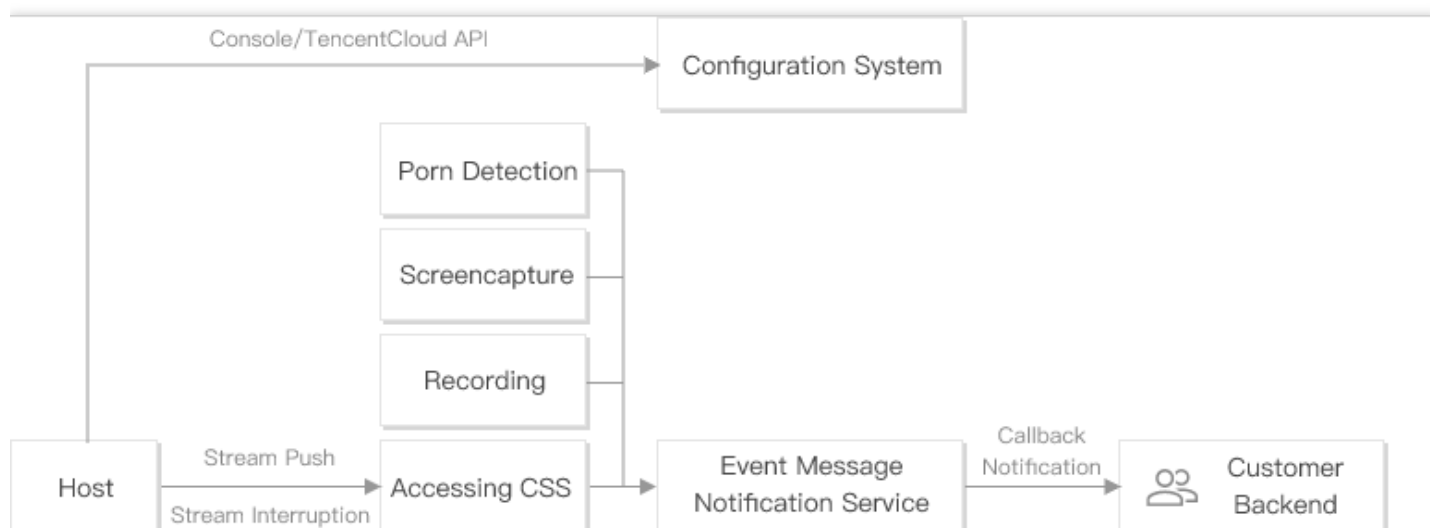
イベント通知の受信方法

最終更新日：：2022-02-14 16:04:29

ライブストリーミングの過程でドメイン名をテンプレートにバインドしたイベントがトリガーされると、Tencent Cloudが自動的にリクエストを顧客のサーバーに送信し、顧客サーバーがリクエストに応答します。検証を通過した後、顧客側ではCSSイベントコールバック情報が含まれたJSONデータパックを取得できます。

現在CSSのイベントトリガーのメッセージ通知でサポートされているイベントには、CSSプッシュ、CSSストリーム切断、CSSレコーディング、CSSスクリーンキャプチャ、CSSポルノ検出の通知メッセージがあります。

全体フロー



フロー説明：

1. キャスターが、コンソールまたはTencentCloud APIを直接呼び出して、イベントメッセージ通知のURLおよびレコーディング、スクリーンキャプチャなどの関連機能を設定します。
2. キャスターがCSSプッシュを切断します。
3. CSSサービス内部にイベントが発生すると、メッセージがイベントメッセージ通知サービス経由でお客様のバックグラウンドに一括でコールバックされます。

イベントメッセージ通知のプロトコル

ネットワークプロトコル

- リクエスト：HTTP POSTリクエスト。パケットの中身はJSON。各メッセージの具体的なパケットの内容は後述をご参照ください。
- 応答：HTTP STATUS CODE = 200、サーバーは応答パケットの具体的な内容を無視します。プロトコルとの親和性のため、クライアントの応答内容に JSON : `{"code":0}` を付けることをお勧めします

通知の信頼性

イベント通知サービスはリトライ機能を備えています。スクリーンキャプチャイベント通知は2分間隔で5回までリトライします。その他、CSSプッシュ、CSSストリーム切断、CSSレコーディング、ライブストリーミングポルノ検出イベント通知は1分間隔で12回までリトライします。

リトライとお客様のサーバーやネットワーク帯域幅との衝突が起きるのを避けるため、正常なリターンパケットを維持してください。リトライがトリガーされる条件は以下となります。

- 長時間（20 秒）リターンパケットの応答がない場合。
- HTTP STATUSの応答が200でない場合。

コールバックイベントの設定方式

コールバック設定は、主に2種類の方式で実現します。1つは、[CSSコンソール](#)を利用し、う1つは [サーバーAPI](#)の呼び出しによるものです。

説明：

CSSのイベントメッセージ通知のコールバックURLでは、プッシュイベント、ストリーム切断イベント、レコーディングイベント、スクリーンキャプチャイベント、ポルノ検出イベントの設定に対して単独のコールバックURLをサポートしています。

CSSコンソール

1. CSSコンソールの**イベントセンター>[CSSコールバック](#)**に進み、コールバックテンプレートを作成します。具体的な操作については、[コールバックテンプレートの作成](#)をご参照ください。
2. [ドメイン名管理](#)で操作が必要なプッシュドメイン名を探し、[管理>テンプレート設定](#)をクリックして、このドメイン名とコールバックテンプレートに関連付けます。具体的な操作については、[コールバック設定](#)をご参照ください。

サーバーAPI

1. [CreateLiveCallbackTemplate](#) を呼び出して、コールバックテンプレートのインターフェースを作成し、必要なコールバックパラメータ情報を設定します。
2. [CreateLiveCallbackRule](#) を呼び出して、コールバックルールを作成し、パラメータのプッシュドメイン名 `DomainName`、`TemplateId` を設定します（ステップ1を繰り返す）。プッシュおよび再生アドレスのものと一致する `AppName` を入力すると、一部のライブストリーミングでコールバック起動の効果が実現されます。

コールバック情報パラメータの説明

コールバックテンプレートとドメイン名のバインドが成功した後、ライブストリーミングの過程でテンプレートのイベントがトリガーされると、Tencent Cloudは自動的にコールバック情報が含まれたJSONパッケージを顧客のサーバーに送信します。コールバック情報の具体的なパラメータ説明は下記のとおりです。

- [プッシュイベントメッセージ通知](#)
- [ストリーム切断イベントメッセージ通知](#)
- [レコーディングイベントメッセージ通知](#)
- [スクリーンキャプチャイベントメッセージ通知](#)
- [ポルノ検出イベントメッセージ通知](#)

プッシュ切断イベント通知

最終更新日：2022-05-27 11:54:41

プッシュ切断コールバックは、CSSプッシュの成功やCSSプッシュの中断といったCSSストリームのステータス情報をプッシュするために使用します。コールバックテンプレートの中でプッシュコールバックやストリーム切断コールバックメッセージの受信サーバーアドレスを設定し、このテンプレートとプッシュドメイン名をバインドしてください。対応するプッシュアドレスを発行し、CSSプッシュを開始するとTencent Cloud CSSのバックエンドがプッシュ結果をお客様が設定した受信サーバーにコールバックします。

このドキュメントでは、主にプッシュ切断コールバックイベントがトリガーされた後、Tencent Cloud CSSがユーザーに送信するコールバックメッセージ通知のフィールドについてご説明します。

注意事項

このドキュメントを読む前に、Tencent Cloud CSSによるコールバック機能の設定方法とコールバックメッセージの受信方法についてご理解下さると幸いです。詳細については、[イベント通知の受信方法](#)をご参照ください。

プッシュ切断イベントのパラメータ説明

イベントタイプパラメータ

| イベントタイプ | フィールド設定値の説明 |
|------------|----------------|
| CSSプッシュ | event_type = 1 |
| CSSストリーム切断 | event_type = 0 |

コールバック共通パラメータ

| フィールド名 | タイプ | 説明 |
|--------|--------|--|
| t | int64 | 期限切れ時間、イベント通知サイン期限のUNIXタイムスタンプ。 <ul style="list-style-type: none"> Tencent Cloudからのメッセージ通知のデフォルトの期限切れ時間は10分です。メッセージ通知中のt値の示す時間が期限に達した場合、この通知を無効と判断し、ネットワークのリプレイアタックを防止できます。 tの形式は10進数UNIXタイムスタンプとなり、即ち1970年1月1日（UTC/GMTの真夜中）から経過した秒数となります。 |
| sign | string | イベント通知セキュリティサイン sign = MD5 (key + t)。 説明：Tencent Cloudが、暗号化keyとtで文字列を結合した後、MD5でsignの値を |

算出し、それを通知メッセージに入れます。お客様のバックエンドサーバーは、通知メッセージの受信後、同じアルゴリズムに基づきsignが正しいか確認し、さらにメッセージが確実にTencent Cloudバックエンドから来たものかを確認することができます。

説明：

keyは、**イベントセンター>CSSコールバック**の中のコールバックキーとなり、主に認証に使用します。お客様のデータ情報のセキュリティを保護するために、入力することをお勧めします。

| | |
|-------------------------|--|
| Callback Key | Please enter a callback key (composed of uppercase and lc |
| Push Callback | Please enter a push callback URL (header: http, https, etc.) |
| Interruption Callback | Please enter an interruption callback URL (header: http, htt |
| Recording Callback | Please enter a recording callback URL (header: http, https, |
| Screencapture Callback | Please enter a screencapture callback URL (header: http, ht |
| Porn Detection Callback | Please enter a porn detection callback URL (header: http, h |

コールバックメッセージのパラメータ

| フィールド名 | タイプ | 説明 |
|------------|--------|--------------|
| appid | int | ユーザー APPID |
| app | string | プッシュドメイン名 |
| appname | string | プッシュパス |
| stream_id | string | CSSストリーム名 |
| channel_id | string | CSSストリーム名と同じ |

| フィールド名 | タイプ | 説明 |
|---------------|--------|--|
| event_time | int64 | イベントメッセージ生成のUNIXタイムスタンプ |
| sequence | string | メッセージシリアルナンバー。1回のプッシュのアクションを表します。1回のプッシュのアクションで同じシリアルナンバーのプッシュとストリーム切断メッセージが生成されます |
| node | string | CSSアクセスポイントのIP |
| user_ip | string | ユーザーのプッシュIP |
| stream_param | string | ユーザープッシュURLに付帯するパラメータ |
| push_duration | string | ストリーム切断イベント通知プッシュの時間（長さ）。単位：ミリ秒 |
| errcode | int | プッシュ切断エラーコード |
| errmsg | string | プッシュ切断エラーの説明 |
| set_id | int | 国内外のプッシュかどうかを判断します。1-6は国内、7-200は海外です |
| width | int | ビデオ幅。最初のプッシュコールバック時にビデオヘッダー情報が欠落している場合は、0になっている可能性があります |
| height | int | ビデオ高さ。最初のプッシュコールバック時にビデオヘッダー情報が欠落している場合は、0になっている可能性があります |

ストリーム切断の原因

ストリーム切断原因の詳細については、[ストリーム切断記録](#)をご参照ください。

コールバックメッセージの例

```
{
  "app": "test.domain.com",

  "appid": 12345678,

  "appname": "live",

  "channel_id": "test_stream",

  "errcode": 0,

  "errmsg": "ok",
```

```
"event_time":1545115790,  
"event_type":1,  
"set_id":2,  
"node":"100.121.160.92",  
"sequence":"6674468118806626493",  
"stream_id":"test_stream",  
"stream_param":"stream_param=test",  
"user_ip":"119.29.94.245",  
"width": 0,  
"height": 0,  
"sign":"ca3e25e5dc17a6f9909a9ae7281e300d",  
"t":1545030873  
}
```

レコーディングイベント通知

最終更新日：2022-12-30 17:01:47

CSSレコーディングでは、プッシュドメイン名にバインドしたレコーディングテンプレートに基づきライブストリーミング画面をリアルタイムでレコーディングし、生成した対応するレコーディングファイルをVODの中に保存します。またレコーディングのコールバックを、レコーディングファイル情報をプッシュ送信するために利用し、これには主にレコーディングの開始時間、終了時間、生成したレコーディングファイルのID、レコーディングファイルサイズ、ファイルダウンロードアドレスが含まれます。お客様はコールバックテンプレートの中でレコーディングのコールバックメッセージを受信するサーバーアドレスを設定し、このテンプレートとプッシュドメイン名をバインドする必要があります。これにより、ライブストリーミングがレコーディングイベントをトリガーすると、Tencent Cloud CSSのバックエンドがレコーディングファイル情報をお客様の設定した受信サーバーにコールバックします。

ここでは、主にレコーディングコールバックイベントがトリガーされた後、Tencent Cloudバックエンドがユーザーに送信するコールバックメッセージ通知のフィールドについてご説明します。

注意事項

- このドキュメントを読む前に、Tencent Cloud CSSによるコールバック機能の設定方法とコールバックメッセージの受信方法についてご理解いただければ幸いです。詳細については、[イベント通知の受信方法](#)をご参照ください。
- レコーディングしたビデオファイルはデフォルトでVODコンソールに保存されますので、オンデマンド業務が支払い延滞によりサービス停止にならないよう注意してください。
- APIを介してレコーディングタスク作成を実行する時は、レコーディングのコールバックはユーザーのプッシュURLに含まれるstream_paramパラメータをリターンしません。その他のレコーディング方式ではリターンされます。
- HLS連続レコーディング機能を設定すると、途中でプッシュが中断されてもコールバックしません。デフォルトではストリーミングを継続し、最終的なファイル生成のみコールバックします。

レコーディングイベントパラメータの説明

イベントタイプパラメータ

| イベントタイプ | フィールド設定値の説明 |
|------------|------------------|
| CSSレコーディング | event_type = 100 |

コールバック共通パラメータ

| フィールド名 | タイプ | 説明 |
|--------|--------|---|
| t | int64 | <p>期限切れ時間、イベント通知サイン期限のUNIXタイムスタンプ。</p> <ul style="list-style-type: none"> Tencent Cloudからのメッセージ通知のデフォルトの期限切れ時間は10分です。メッセージ通知中のt値が示す時間が期限に達した場合、この通知を無効と判断され、ネットワークのリプレイアタックを防止します。 tの形式は10進数UNIXタイムスタンプとなり、1970年1月1日（UTC/GMT 真夜中）から経過した秒数となります。 |
| sign | string | <p>イベント通知セキュリティサイン $sign = MD5(key + t)$。</p> <p>説明：Tencent Cloudが、暗号化keyとtで文字列を結合した後、MD5でsignの値を算出し、それを通知メッセージに入れます。お客様のバックエンドサーバーは、通知メッセージの受信後、同じアルゴリズムに基づきsignが正確か確認し、さらにメッセージが確実にTencent Cloudバックエンドから来たものかを確認することができます。</p> |

説明：

keyは、**イベントセンター>CSSコールバック**の中のコールバックキーとなり、主に認証に使用します。お客様のデータ、情報のセキュリティ保護のため、入力することをお勧めします。

| | |
|-------------------------|---|
| Callback Key | <input type="text" value="Please enter a callback key (composed of uppercase and lc)"/> |
| Push Callback | <input type="text" value="Please enter a push callback URL (header: http, https, etc.)"/> |
| Interruption Callback | <input type="text" value="Please enter an interruption callback URL (header: http, htt"/> |
| Recording Callback | <input type="text" value="Please enter a recording callback URL (header: http, https,"/> |
| Screencapture Callback | <input type="text" value="Please enter a screencapture callback URL (header: http, ht"/> |
| Porn Detection Callback | <input type="text" value="Please enter a porn detection callback URL (header: http, h"/> |

コールバックメッセージのパラメータ

| フィールド名 | タイプ | 説明 |
|------------------|--------|--|
| appid | int | ユーザー APPID |
| app | string | プッシュドメイン名 |
| appname | string | プッシュパス |
| stream_id | string | CSSストリーム名 |
| channel_id | string | CSSストリーム名と同じ |
| file_id | string | VOD file ID、 VODプラットフォーム で1つのVODビデオファイルを一意的に特定することができます |
| record_file_id | string | VODファイルID |
| file_format | string | FLV, HLS, MP4, AAC |
| task_id | string | レコーディングタスクID。APIで作成したレコーディングタスクに対してのみ有効であり、すなわち CreateRecordTask によって返されるタスクID |
| start_time | int64 | レコーディングタスクによるファイル書き込み開始時間。この値をレコーディングコンテンツの開始時間にすることはできません。レコーディングコンテンツの開始時間 = end_time - duration |
| end_time | int64 | レコーディングタスクによるファイル書き込み終了時刻 |
| start_time_usec | int | レコーディングタスクでファイルへの書き込みを開始する時間（マイクロ秒部分） |
| end_time_usec | int | レコーディングタスクでファイルへの書き込みを終了する時間（マイクロ秒部分） |
| duration | int64 | レコーディングファイルの長さ。単位：秒 |
| file_size | uint64 | レコーディングファイルサイズ。単位：byte |
| stream_param | string | ユーザープッシュURLに付帯するパラメータ（カスタマイズ） |
| video_url | string | レコーディングファイルのダウンロードURL |
| media_start_time | int | レコーディングでプルを開始するときに受信した最初のフレームpts（必ずしもファイルの最初のフレームptsではありません） |
| record_bps | int | レコーディングでトランスコードのプルから対応するビットレート（Kbps） |

| フィールド名 | タイプ | 説明 |
|--------------|----------------------------------|--|
| callback_ext | string、 jsonオブ ジェクト 文字列 | jsonオブジェクトに複数のフィールドが含まれます。うち video_codecはプッシュビデオcodec名です resolutionはプッシュビデオの解像度です 以上はいずれもレコーディング コールバック拡張フィールドであり、ビ ジネスの参考となります。ビジネスロジックがこれらのフィールドに強く 依存することはお勧めしません。 |

コールバックメッセージの例

```
{
  "event_type": 100,

  "appid":12345678,

  "app": "yourapp",

  "callback_ext": "{\"video_codec\":\"h264\",\"resolution\":\"640x480\"}",

  "appname": "yourappname",

  "stream_id":"stream_test",

  "channel_id":"stream_test",

  "file_id":"1234567890",

  "record_file_id": "1234567890",

  "file_format":"hls",

  "task_id":"UpTbk5RSVhRQ*****0xTS1NTQ1t1RVRLU1JAWW9EUb",

  "start_time":1642089445,

  "end_time":1642089598,

  "start_time_usec": 316441,

  "end_time_usec": 618577,

  "duration":154,

  "file_size":277941079,
```

```
"stream_param": "stream_param=test",  
  
"video_url": "http://12345678.vod2.myqcloud.com/xxxx/yyyy/zzzz.m3u8",  
  
"media_start_time": 135802,  
  
"record_bps": 0,  
  
"sign": "ca3e25e*****09a9ae7281e300d",  
  
"t": 1545030873  
}
```


スクリーンキャプチャイベント通知

最終更新日：2022-05-05 15:20:46

ライブブロードキャストのスクリーンキャプチャは、一定の時間間隔でリアルタイムのライブブロードキャストストリーミングの画像をキャプチャし、画像を生成してCOSに保存することです。スクリーンキャプチャのコールバックは、スクリーンキャプチャをプッシュして、主に、画像生成時間、画像サイズ、パス、ダウンロード可能なアドレスなどを含む画像ファイル情報を保存することに使用します。コールバックテンプレートでスクリーンキャプチャコールバックメッセージの受信サーバーアドレスを設定し、そのテンプレートとプッシュドメイン名をバインドする必要があります。ライブブロードキャストストリーミングがスクリーンキャプチャイベントをトリガーすると、Tencent Cloud Live Video Broadcasting (LVB)がバックエンドで、お客様が設定した受信サーバーにスクリーンキャプチャ情報をコールバックします。

このドキュメントでは、主にスクリーンキャプチャコールバックイベントをトリガーすると、LVBがユーザーに送信するコールバックメッセージ通知のフィールドについて説明します。

注意事項

- このドキュメントを読む前に、LVBによるコールバック機能の構成方法とコールバックメッセージの受信方法について理解してください。詳細については、[イベント通知の受信方法](#)をご参照ください。
- スクリーンキャプチャコールバックイベントがトリガーされた後に取得したスクリーンキャプチャ情報は、ポルノ検出のライブブロードキャスト、ルームサムネイルのライブブロードキャストなどのさまざまなシナリオに適用できます。

スクリーンキャプチャイベントのパラメータ説明

イベントタイプパラメータ

| イベントタイプ | フィールド設定値の説明 |
|------------------------|------------------|
| スクリーンキャプチャのライブブロードキャスト | event_type = 200 |

共通パラメータのコールバック

| フィールド名 | タイプ | 説明 |
|--------|-------|--|
| t | int64 | 期限切れ時間、イベント通知サイン期限のUNIXタイムスタンプ。 <ul style="list-style-type: none">Tencent Cloudからのメッセージ通知のデフォルトの期限切れ時間は10分です。メッセージ通知のt値が示す時間は期限切れの場合、この通知を無効と判 |

| | | |
|------|--------|---|
| | | <p>断され、ネットワークのリプレーアタックを防止できます。</p> <ul style="list-style-type: none"> tの形式は10進数UNIXタイムスタンプとなり、1970年1月1日（UTC/GMT真夜中）から経過した秒数となります。 |
| sign | string | <p>イベント通知セキュリティサイン $sign = MD5(key + t)$。</p> <p>説明：Tencent Cloudが、暗号化keyとtで文字列を結合した後、MD5でsignの値を算出し、それを通知メッセージに入れます。お客様のバックエンドサーバーは、通知メッセージを受信すると、同じアルゴリズムに基づいてsignが正確さを確認でき、さらにメッセージが確実にTencent Cloudバックエンドからきたものかを確認できます。</p> |

説明：

keyは、イベントセンター>[CSSコールバック](#)の中のコールバックキーとなり、主に認証に使用します。お客様のデータ情報のセキュリティを保護するために、入力することをお勧めします。

Callback Key

Please enter a callback key (composed of uppercase and lc

Push Callback

Please enter a push callback URL (header: http, https, etc.)

Interruption Callback

Please enter an interruption callback URL (header: http, htt

Recording Callback

Please enter a recording callback URL (header: http, https,

Screenshot Callback

Please enter a screenshot callback URL (header: http, ht

Porn Detection Callback

Please enter a porn detection callback URL (header: http, h

メッセージパラメータのコールバック

| フィールド名 | タイプ | 説明 |
|--------|--------|-----------|
| app | string | プッシュドメイン名 |

| フィールド名 | タイプ | 説明 |
|--------------|--------|---|
| appname | string | プッシュパス |
| stream_param | string | ユーザープッシュURLに付加するパラメータ |
| stream_id | string | ライブブロードキャストストリーミング名 |
| channel_id | string | ライブブロードキャストストリーミング名と同じ |
| create_time | int64 | スクリーンキャプチャがUNIXタイムスタンプを生成する |
| file_size | int | スクリーンキャプチャのファイルサイズ。単位：byte |
| width | int | スクリーンキャプチャの幅。単位：ピクセル |
| height | int | スクリーンキャプチャの高さ。単位：ピクセル |
| pic_url | string | スクリーンキャプチャのファイルパス <code>/path/name.jpg</code> |
| pic_full_url | string | スクリーンキャプチャのダウンロードURL |

コールバックメッセージの例

```
{
  "app": "test.app",
  "appname": "live",
  "channel_id": "your_channelid",
  "create_time": 1622599925,
  "event_type": 200,
  "file_size": 30670,
  "height": 720,
  "pic_full_url": "http://your.cos.region.myqcloud.com/channelid/channelid-screenshot-10-12-05-1280x720.jpg",
  "pic_url": "/channelid/channelid-screenshot-10-12-05-1280x720.jpg",
  "sign": "ca3e25e5dc17a6f9909a9ae7281e300d",
  "stream_id": "your_streamid",
```

```
"stream_param": "txSecret=ca3e25e5dc17a6f9909a9ae7281e300d&txTime=60B83800",  
"t": 1622600525,  
  "width": 1280  
}
```

ポルノ検出イベント通知

最終更新日：2022-01-05 11:00:24

CSSポルノ検出は、スクリーンキャプチャポルノ検出テンプレートで設定されたルールに従ってCSSストリームのスクリーンキャプチャを行い、画像を生成してCOSに保存するとともに、Pornfoundの識別機能によって不正な内容のある画像を識別します。ポルノ検出コールバックはライブストリーミングのポルノ検出画像情報のプッシュに用いられ、これには問題画像が属するタイプ、レベル評価、スクリーンキャプチャの時間などが含まれます。コールバックテンプレートの中でポルノ検出コールバックメッセージの受信サーバーアドレスを設定し、当該テンプレートとプッシュドメイン名を関連付ける必要があります。ライブストリーミングがポルノ検出イベントをトリガーした後、Tencent Cloud CSSのバックエンドがポルノ関連画像情報をお客様が設定した受信サーバーにコールバックします。

ここでは、主にポルノ検出コールバックイベントがトリガーされた後、Tencent Cloudバックエンドがユーザーに送信するコールバックメッセージ通知のフィールドについてご説明します。

注意事項

- このドキュメントを読む前に、Tencent Cloud CSSによるコールバック機能の設定方法とコールバックメッセージの受信方法についてご理解下さると幸いです。詳細については、[イベント通知の受信方法](#)をご参照ください。
- CSSポルノ検出は、デフォルトでは疑わしい結果に対してのみコールバックを行い、通常の結果にはコールバックを行いません。
- 画像の `type` を利用してポルノ画像に対する評定を行う場合、検出システムの判定を100%の精度にすることはできないため、時折判定が誤っていることもあります。よって、実際のユースケースに応じて、人による二次確認を実施することをお勧めします。

スクリーンキャプチャイベントのパラメータ説明

イベントタイプパラメータ

| イベントタイプ | フィールド設定値の説明 |
|----------|-------------------------------|
| CSSポルノ検出 | <code>event_type = 317</code> |

コールバック共通パラメータ

| フィールド名 | タイプ | 説明 |
|--------|-----|----|
|--------|-----|----|

| | | |
|------|--------|---|
| t | int64 | <p>期限切れ時間、イベント通知サイン期限のUNIXタイムスタンプ。</p> <ul style="list-style-type: none"> • Tencent Cloudからのメッセージ通知のデフォルトの期限切れ時間は10分です。メッセージ通知中のt値が示す時間が期限に達した場合、この通知を無効と判断され、ネットワークのリプレイアタックを防止します。 • tの形式は10進数UNIXタイムスタンプとなり、1970年1月1日（UTC/GMT 真夜中）から経過した秒数となります。 |
| sign | string | <p>イベント通知セキュリティサイン $sign = MD5(key + t)$。</p> <p>説明：Tencent Cloudが、暗号化 key と t で文字列を結合した後、MD5でsignの値を算出し、それを通知メッセージに入れます。お客様のバックエンドサーバーは、通知メッセージの受信後、同じアルゴリズムに基づきsignが正確か確認し、さらにメッセージが確実にTencent Cloudバックエンドから来たものかを確認することができます。</p> |

説明：

keyは、イベントセンター>[CSSコールバック](#)の中のコールバックキーとなり、主に認証に使用します。お客様のデータ、情報のセキュリティ保護のため、入力することをお勧めします。

Callback Key

Please enter a callback key (composed of uppercase and lc

Push Callback

Please enter a push callback URL (header: http, https, etc.)

Interruption Callback

Please enter an interruption callback URL (header: http, htt

Recording Callback

Please enter a recording callback URL (header: http, https,

Screenshot Callback

Please enter a screenshot callback URL (header: http, ht

Porn Detection Callback

Please enter a porn detection callback URL (header: http, h

コールバックメッセージのパラメータ

| パラメータ | 入力必須の有無 | データタイプ | 説明 |
|--------------|---------|----------------------|--|
| streamId | オプション | String | ストリーム名 |
| channelId | オプション | string | チャンネルID |
| img | 入力必須 | string | 予備警告画像リンク |
| type | 入力必須 | Array | 検出結果で優先度が最も高い悪意のあるタグに対応する分類値のことです。具体的な意味については、パラメータlabelによって返される補足テキストの説明を参照することができます |
| score | 入力必須 | Array | type 対応する評点 |
| ocrMsg | オプション | string | 画像のOCR認識情報（該当する場合） |
| suggestion | 入力必須 | string | 推奨値。値はオプション： <ul style="list-style-type: none"> • Block：ブロック • Review：レビュー待ち • Pass：正常 |
| label | 入力必須 | string | このフィールドは、検出結果(LabelResults)において対応する優先度が最も高い悪意のあるタグを返すために使用され、モデルが推奨するレビュー結果を表します。業務上のニーズに応じて、それぞれの不正タイプと推奨値を処理することをお勧めします |
| subLabel | 入力必須 | string | このフィールドは、検出結果でヒットした最も優先度が高い悪意のあるタグの下にあるサブタグの名前を返すために使用されます。例えば、ポルノ--性行為で、サブタグがヒットしない場合は、空の文字列が返されます |
| labelResults | オプション | Array of LabelResult | このフィールドは、分類モデルによってヒットした悪意のあるタグの詳細な識別結果を返すために使用されます。これには、ポルノや広告といった好ましくない、安全ではない、または不適切なコンテンツタイプの識別結果が含まれます 注意：このフィールドは、有効な値が取得できなかったことを意味するnullを返す場合があります |

| パラメータ | 入力必須の有無 | データタイプ | 説明 |
|----------------|---------|-----------------------|--|
| objectResults | オプション | Array of ObjectResult | このフィールドは、オブジェクト検出モデルの詳細な検出結果を返すために使用されます。これには、エンティティ、広告ロゴ、2次元コードなどのコンテンツでヒットしたタグ名、タグスコア、座標情報、シーン識別結果、推奨される操作といったコンテンツ識別情報が含まれます。詳細な戻り値の情報については、対応するデータ構造(ObjectResults)の説明をご参照ください 注意：このフィールドは、有効な値が取得できなかったことを意味するnullを返す場合があります |
| ocrResults | オプション | Array of OcrResult | このフィールドは、OCRテキスト識別の詳細な検出結果を返すために使用されます。これには、テキスト座標情報、テキスト識別結果、推奨される操作といったコンテンツ識別情報が含まれます。詳細な戻り値の情報については、対応するデータ構造(ObjectResults)の説明をご参照ください 注意：このフィールドは、有効な値が取得できなかったことを意味するnullを返す場合があります |
| libResults | オプション | Array of LibResult | リスクライブラリのレビュー結果 |
| screenshotTime | 入力必須 | Number | スクリーンキャプチャ時間 |
| sendTime | 入力必須 | Number | リクエスト送信時間、UNIXタイムスタンプ |
| stream_param | オプション | String | プッシュパラメータ |
| app | オプション | String | プッシュドメイン名 |
| appid | オプション | Number | 業務ID |
| appname | オプション | String | プッシュpathパス |

LabelResult

分類モデルのヒット結果。

| 名称 | タイプ | 説明 |
|-------|--------|--|
| Scene | String | 広告、ポルノ、有害なコンテンツのシーンなど、モデルによって識別されたシーンの結果を返します。 |

| 名称 | タイプ | 説明 |
|------------|--|--|
| Suggestion | String | 現在の悪意のあるタグに対する後続操作の提案を返します。判定結果が得られた後の戻り値はシステムが推奨する後続操作を表示します。業務上のニーズに応じて、各不正のタイプと推奨値に対処することをお勧めします。戻り値： <ul style="list-style-type: none"> • Block：ブロックを推奨 • Review：手動レビューを推奨 • Pass：承認を推奨 |
| label | String | このフィールドは、検出結果に対応する悪意のあるタグを返すために使用されます |
| SubLabel | String | サブタグ名 |
| Score | Integer | このタグモデルでヒットするスコア |
| Details | Array of LabelDetailItem | 分類モデルでヒットするサブタグの詳細な結果 |

LabelDetailItem

分類モデルでヒットするサブタグの結果。

| 名称 | タイプ | 説明 |
|-------|---------|----------------------------|
| Id | Integer | 番号 |
| Name | String | サブタグ名 |
| Score | Integer | サブタグスコア。値の範囲は0ポイント～100ポイント |

ObjectResult

エンティティの検出結果の詳細。

| 名称 | タイプ | 説明 |
|-------|--------|---|
| Scene | String | 2次元コード、ロゴ、画像OCRのシーンなど、エンティティによって識別されたエンティティシーンの結果を返します。 |

| 名称 | タイプ | 説明 |
|------------|---------------------------------------|---|
| Suggestion | String | 現在の悪意のあるタグに対する後続操作の提案を返します。判定結果が得られた後の戻り値は、システムが推奨する後続操作を表します。業務上のニーズに応じて、それぞれの不正タイプと推奨値を処理することをお勧めします。戻り値： <ul style="list-style-type: none"> • Block：ブロックを推奨 • Review：手動レビューを推奨 • Pass：承認を推奨 |
| label | String | このフィールドは、検出結果に対応する悪意のあるタグを返すために使用されます |
| SubLabel | String | サブタグ名 |
| Score | Integer | 該当するシーンモデルでヒットするサブタグのスコア。値の範囲は0ポイント～100ポイント |
| Names | Array of String | エンティティ名リスト |
| Details | Array of ObjectDetail | エンティティ検出結果の詳細 |

ObjectDetail

エンティティ検出結果の詳細で、検出シーンがエンティティ、広告ロゴ、2次元コードである場合、モデル検出対象フレームのタグ名、タグ値、タグスコアおよび検出フレームの位置情報を表します。

| 名称 | タイプ | 説明 |
|-------|---------|---|
| Id | Integer | このパラメータは、識別や区別を容易にするために、識別されたオブジェクトのIDを返すために使用されます |
| Name | String | このパラメータは、ヒットのエンティティタグを返すために使用されます |
| Value | String | このパラメータは、対応するエンティティタグに対応する値またはコンテンツを返すために使用されます。例えば、タグが2次元コード(QrCode)の場合、このフィールドは識別された2次元コードに対応するURLアドレスになります |
| Score | Integer | このパラメータは、対応するエンティティタグのスコアを返すために使用され、値は0～100です。例えば、QrCode 99は、対応する識別コンテンツが2次元コードシーntagにヒットする確率が非常に高いことを意味します |

| 名称 | タイプ | 説明 |
|----------|----------|---|
| Location | Location | このフィールドは、エンティティ検出フレームの座標位置（左上隅のxy座標、長さ と幅、回転角度）を返すために使用され、エンティティの関連情報の素早い配置を 容易にします |

Location

座標。

| 名称 | タイプ | 説明 |
|--------|-------|-------------|
| X | Float | 左上隅の横軸 |
| Y | Float | 左上隅の縦軸 |
| Width | Float | 幅 |
| Height | Float | 高さ |
| Rotate | Float | 検出フレームの回転角度 |

OcrResult

OCR検出結果の詳細です。

| 名称 | タイプ | 説明 |
|------------|---------|---|
| Scene | String | 識別シーンを表します。デフォルト値はOCR（画像OCR認識）です。 |
| Suggestion | String | 優先度が最も高い悪意のあるタグに対応する後続操作の提案を返します。判 定結果が得られた後の戻り値は、システムが推奨する後続操作を表します。 業務上のニーズに応じて、それぞれの不正タイプと推奨値を処理すること をお勧めします。戻り値： <ul style="list-style-type: none"> • Block：ブロックを推奨 • Review：手動レビューを推奨 • Pass：承認を推奨 |
| label | String | このフィールドは、検出結果に対応する悪意のあるタグを返すために使用さ れます |
| SubLabel | String | サブタグ名 |
| Score | Integer | 該当するシーンモデルでヒットするサブタグのスコア。値の範囲は0ポイン ト～100ポイント |
| Text | String | テキストコンテンツ |

| 名称 | タイプ | 説明 |
|---------|--|-----------|
| Details | Array of OcrTextDetail | OCR 結果の詳細 |

OcrTextDetail

OCRテキスト結果の詳細です。

| 名称 | タイプ | 説明 |
|----------|--------------------------|--|
| Text | String | OCRで識別されたテキストコンテンツを返します（OCRテキスト識別の上限は**5000バイト以内**です）。 |
| label | String | このフィールドは、検出結果に対応する悪意のあるタグを返すために使用されます |
| Keywords | Array of String | このタグでヒットしたキーワード |
| Score | Integer | 該当するタグモデルでヒットするスコア。値の範囲は0ポイント～100ポイント |
| Location | Location | OCR テキスト座標位置 |

LibResult

ブラック/ホワイトライブラリ結果の詳細です。

| 名称 | タイプ | 説明 |
|------------|---------|---|
| Scene | String | モデルのシーン識別結果を表します。デフォルト値はSimilarです。 |
| Suggestion | String | 後続操作の提案を返します。判定結果が得られた後の戻り値は、システムが推奨する後続操作を表します。業務上のニーズに応じて、それぞれの不正タイプと推奨値を処理することをお勧めします。戻り値： <ul style="list-style-type: none"> • Block：ブロックを推奨 • Review：手動レビューを推奨 • Pass：承認を推奨 |
| label | String | このフィールドは、検出結果に対応する悪意のあるタグを返すために使用されます |
| SubLabel | String | サブタグ名 |
| Score | Integer | 画像検索モデル認識スコア。値の範囲は0ポイント～100ポイント |

| 名称 | タイプ | 説明 |
|---------|---------------------------------------|---------------------|
| Details | Array of ObjectDetail | ブラック/ホワイトライブラリ結果の詳細 |

LibDetail

カスタマイズコーパス/ブラック/ホワイトライブラリの詳細です。

| 名称 | タイプ | 説明 |
|---------|---------|---------------------------------------|
| Id | Integer | 番号 |
| Imageld | String | 画像ID |
| label | String | このフィールドは、検出結果に対応する悪意のあるタグを返すために使用されます |
| Tag | String | カスタムタグ |
| Score | Integer | モデル認識スコア。値の範囲は0ポイント～100ポイント |

コールバックメッセージの例

```
{
  "ocrMsg": "",
  "type": [1],
  "score": 99,
  "screenshotTime": 1610640000,
  "level": 0,
  "img": "http://1.1.1.1/download/porn/test.jpg",
  "abductionRisk": [],
  "faceDetails": [],
  "sendTime": 1615859827,
  "suggestion": "Block",
  "label": "Porn",
  "subLabel": "PornHigh",
  "labelResults": [{
    "HitFlag": 0,
    "Scene": "Illegal",
    "Suggestion": "Pass",
    "Label": "Normal",
    "SubLabel": "",
    "Score": 0,
    "Details": []
  }, {
    "HitFlag": 1,
```

```
"Scene": "Porn",
"Suggestion": "Block",
"Label": "Porn",
"SubLabel": "PornHigh",
"Score": 99,
"Details": [{
  "Id": 0,
  "Name": "PornHigh",
  "Score": 99
}, {
  "Id": 1,
  "Name": "WomenChest",
  "Score": 99
}]
}, {
  "HitFlag": 0,
  "Scene": "Sexy",
  "Suggestion": "Pass",
  "Label": "Normal",
  "SubLabel": "",
  "Score": 0,
  "Details": []
}, {
  "HitFlag": 0,
  "Scene": "Terror",
  "Suggestion": "Pass",
  "Label": "Normal",
  "SubLabel": "",
  "Score": 0,
  "Details": []
}],
"objectResults": [{
  "HitFlag": 0,
  "Scene": "QrCode",
  "Suggestion": "Pass",
  "Label": "Normal",
  "SubLabel": "",
  "Score": 0,
  "Names": [],
  "Details": []
}, {
  "HitFlag": 0,
  "Scene": "MapRecognition",
  "Suggestion": "Pass",
  "Label": "Normal",
  "SubLabel": "",
  "Score": 0,
```

```
"Names": [],
"Details": []
}, {
"HitFlag": 0,
"Scene": "PolityFace",
"Suggestion": "Pass",
"Label": "Normal",
"SubLabel": "",
"Score": 0,
"Names": [],
"Details": []
}],
"ocrResults": [{
"HitFlag": 0,
"Scene": "OCR",
"Suggestion": "Pass",
"Label": "Normal",
"SubLabel": "",
"Score": 0,
"Text": "",
"Details": []
}],
"streamId": "teststream",
"channelId": "teststream",
"stream_param": "txSecret=40f38f69f574fd51126c421a3d96c374&txTime=5DEBEC80",
"app": "5000.myqcloud.com",
"appname": "live",
"appid": 10000,
"event_type": 317,
"sign": "ac920c3e66*****78cf1b5de2c63",
"t": 1615860427
}
```

プッシュ異常イベント通知

最終更新日：2023-03-13 15:46:49

プッシュ異常イベントのコールバックは、主にプッシュ異常に関する具体的な情報をコールバックするために使用されます。お客様はプッシュ異常イベントのコールバックでコールバックアドレスを設定する必要があります。Tencent CloudのCSSバックグラウンドでは、設定した受信サーバーにタイプの結果がコールバックされます。ここでは、プッシュ異常イベントのコールバックをトリガーした後、Tencent Cloud CSSがユーザーに送信するコールバックメッセージ通知フィールドについてご説明します。

注意事項

このドキュメントを読む前に、Tencent Cloud CSSによるコールバック機能の設定方法とコールバックメッセージの受信方法についてご理解下さると幸いです。詳細については、[イベント通知の受信方法](#)をご参照ください。

プッシュ異常イベントパラメータの説明

イベントタイプパラメータ

| イベントタイプ | フィールド設定値の説明 |
|------------|------------------|
| プッシュ異常イベント | event_type = 321 |

コールバック共通パラメータ

| パラメータ | タイプ | 意味 |
|-----------------|--------|----------------------------|
| appid | int | ユーザー APPID |
| stream_id | string | CSSストリーム名 |
| data_time | int | プッシュイベントコールバック時間（単位ms） |
| report_interval | int | 異常イベントが発生した場合の上位送信間隔（単位ms） |
| abnormal_event | json | 詳細な異常イベントグループ |

abnormal_event内のパラメータ説明

| パラメータ | タイプ | 意味 |
|-------|-----|----|
|-------|-----|----|

| | | |
|--------------|--------|--|
| type | int | 異常イベントタイプ |
| count | int | 異常イベントの単位時間内（上位送信間隔内）の発生回数 |
| detail | json | <ul style="list-style-type: none"> desc：異常イベントの記述 occur_time：異常イベントの発生時間 |
| type_desc_cn | string | 異常イベントの日本語記述 |
| type_desc_en | string | 異常イベントの英語記述 |

異常イベントのタイプ

| タイプ | 意味 |
|-----|-----------------------------------|
| 1 | ビデオのタイムスタンプのロールバック |
| 2 | オーディオのタイムスタンプのロールバック |
| 3 | ビデオのタイムスタンプがいきなり大きくなった（1秒を超える） |
| 4 | オーディオのタイムスタンプがいきなり大きくなった（1秒を超える） |
| 5 | chunk sizeが大きすぎる（8192を超える） |
| 6 | 2フレームのビデオフレームの到達時間が長すぎる（3sを超える） |
| 7 | 2フレームのオーディオフレームの到達時間が長すぎる（3sを超える） |
| 8 | ビデオエンコーディングタイプが変化した |
| 9 | オーディオエンコーディングタイプが変化した |
| 10 | ビデオフレームが到着する前にcodecヘッダーがない |
| 11 | オーディオフレームが到着する前にcodecヘッダーがない |

注意：

- 現在のプッシュ異常イベントのコールバックでは、上位送信の間隔ごとに発生した異常イベントをコールバックするため、個々のイベントを設定することができません。異常イベントがない場合、コールバックは行われません。
- プッシュ異常イベントのコールバックでは、現在のプッシュ異常が統計されるが、他の処理は行われません。

コールバックメッセージの例

```
{
  "abnormal_event": [
    {
      "count": 2,
      "detail": [
        {
          "desc": "video frame arrive interval too long, interval=3046(msec)",
          "occur_time": 1670588070569
        },
        {
          "desc": "video frame arrive interval too long, interval=2953(msec)",
          "occur_time": 1670588073522
        }
      ],
      "type": 6,
      "type_desc_cn": "ビデオフレームの到達時間差が1000msを超えています",
      "type_desc_en": "video frame arrive interval bigger than 1000(ms)"
    },
    {
      "count": 2,
      "detail": [
        {
          "desc": "audio frame arrive interval too long, interval=3009(msec)",
          "occur_time": 1670588070532
        },
        {
          "desc": "audio frame arrive interval too long, interval=2917(msec)",
          "occur_time": 1670588073486
        }
      ],
      "type": 7,
      "type_desc_cn": "オーディオフレームの到達時間差が1000msを超えています",
      "type_desc_en": "audio frame arrive interval bigger than 1000(ms)"
    }
  ],
  "appid": 0,
  "data_time": 1670588074971,
  "domain": "xxxx.xxxxx.xxxx.xxxx",
  "event_type": 321,
  "interface": "general_callback",
  "path": "xxxx",
  "report_interval": 5000,
  "sequence": "000000000000000000",
  "stream_id": "xxxxxx",
  "stream_param": "txSecret=f5828cd4a8a09109304b060172fb3960&txTime=665982e4",
}
```

```
"timeout":5000  
}
```

プルリツイートイベント通知

最終更新日：：2022-01-13 14:53:20

プルリツイートコールバックは主にプルリツイートタスクのステータス情報をコールバックするために使用されます。プルリツイートタスクでコールバックアドレスを設定する必要があり、Tencent Cloud CSSのバックグラウンドがタイプの結果をお客様の設定した受信サーバーにコールバックします。

ここでは、主にプッシュ切断コールバックイベントがトリガーされた後、Tencent Cloud CSSがユーザーに送信するコールバックメッセージ通知のフィールドについてご説明します。

注意事項

このドキュメントに目を通す前に、Tencent Cloud CSSではコールバック機能をどのように設定し、コールバックメッセージをどのように受信するのかを理解している必要があります。具体的な内容については、[イベント通知の受信方法](#)をご参照ください。

プルリツイートイベントパラメータの説明

イベントタイプパラメータ

| イベントタイプ | フィールド設定値の説明 |
|---------|------------------|
| プルリツイート | event_type = 314 |

コールバック共通パラメータ

| パラメータ | タイプ | 意味 |
|----------------|--------|----------------|
| appid | int | ユーザー APPID |
| callback_event | string | コールバックイベントのタイプ |
| source_urls | string | プルソース URL |
| to_url | string | プッシュターゲット URL |
| stream_id | string | CSSストリーム名 |
| task_id | string | タスクID |

| パラメータ | タイプ | 意味 |
|------------------|--------|--------------------|
| <code>msg</code> | string | イベントごとの詳細なコールバック情報 |

msg 内パラメータの説明

| パラメータ | タイプ | 意味 |
|------------------------------|---------|-------------------------|
| <code>task_start_time</code> | int | タスク開始時間、ミリ秒タイムスタンプ |
| <code>url</code> | String | 現在プルされているソースURL |
| <code>index</code> | string | VODファイルが配置されるリストのインデックス |
| <code>duration</code> | int | VODファイルの継続時間、秒 |
| <code>task_end_time</code> | int | タスク終了時間、ミリ秒タイムスタンプ |
| <code>code</code> | string | タスク終了エラーコード |
| <code>message</code> | String? | タスク終了エラー情報。 |

コールバックメッセージの例

TaskStart - タスク開始のコールバック

```
{
  "appid": 4,
  "callback_event": "TaskStart",
  "event_type": 314,
  "interface": "general_callback",
  "msg": "{\"task_start_time\":0}",
  "product_name": "pullpush",
  "source_urls": "[\"http://yourURL.cn/live/normal_230753472*****21162358-upload-45eb/playlist.m3u8\"]\n",
  "stream_id": "",
  "task_id": "118148",
```

```
"to_url": "rtmp://5000.livepush.myqcloud.com/live/testvod"
}
```

VodSourceFileStart - VODファイル開始のコールバック

```
{
  "appid": 4,

  "callback_event": "VodSourceFileStart",

  "callback_url": "http://you.callback.url",

  "event_type": 314,

  "interface": "general_callback",

  "msg": "{ \"url\": \"http://remit-tx-ugcpub.douyucdn2.cn/live/normal_466247620****
*3100448-upload-216b/playlist.m3u8\", \"index\": 0, \"duration\": 14920 }",

  "product_name": "pullpush",

  "source_urls": "[ \"http://yourURL.cn/live/normal_466247620*****3100448-upload-21
6b/playlist.m3u8\" ]\n",

  "stream_id": "",

  "task_id": "118145",

  "to_url": "rtmp://5000.livepush.myqcloud.com/live/testvod"
}
```

VodSourceFileFinish - VODファイル終了のコールバック

```
{
  "appid": 4,

  "callback_event": "VodSourceFileFinish",

  "callback_url": "http://you.callback.url",

  "event_type": 314,

  "interface": "general_callback",

  "msg": "{ \"url\": \"http://yourURL.cn/live/normal_466247620*****3100448-upload-21
```

```
6b/playlist.m3u8\", \"index\":0, \"duration\":14920}\",  
  
"product_name": "pullpush",  
  
"source_urls": "[\"http://yourURL.cn/live/normal_466247620*****3100448-upload-21  
6b/playlist.m3u8\"]\n",  
  
"stream_id": "",  
  
"task_id": "118145",  
  
"to_url": "rtmp://5000.livepush.myqcloud.com/live/testvod"  
}
```

TaskExit - タスク終了のコールバック

```
{  
"appid": 4,  
  
"callback_event": "TaskExit",  
  
"event_type": 314,  
  
"interface": "general_callback",  
  
"msg": "{ \"message\": \"write packet error.\", \"code\":-22, \"task_exit_time\":  
0 }",  
  
"product_name": "pullpush",  
  
"source_urls": "[\"http://yourURL.cn/live/normal_230753472*****21162358-upload-4  
\"]\n"  
}
```

注意：

- VODビデオプリリツイート設定のコールバック順序は次のとおりです：「TaskStart-タスク開始のコールバック」 > 「VodSourceFileStart-VODファイル開始のコールバック」 > 「VodSourceFileFinish-VODファイル終了のコールバック」。
- 「TaskStart-タスク開始のコールバック」と「VodSourceFileStart-VODファイル開始のコールバック」の間には**2s以内**の間隔があります。
- プリリツイートのコールバック設定はプリリツイートタスクで設定します。

一般的なサードパーティ製ツールガイド

OBSプッシュ

最終更新日：：2022-05-05 15:17:24

ソリューションの背景

Open Broadcaster Software（略称 OBS）は、オープンソースプログラムのサードパーティ製ライブストリーミングメディアコンテンツ制作ソフトウェアで、使いやすく、ユーザーに無償で提供されています。OS X、Windows、LinuxなどのOSをサポートし、多様なライブブロードキャストのシナリオに適応でき、ライブブロードキャスト行為のほとんどのオペレーションニーズに応えることが可能です。最新版のソフトウェアは、[OBS公式サイト](#)でダウンロードできます。

準備作業

- [OBS](#)ツールをインストールします。
- [CSSサービスの有効化](#)が実行され、登録されたドメイン名を準備し、[再生ドメイン名の追加](#)が実行されます（システムが提供するデフォルトのプッシュドメイン名を使用するか、カスタムドメイン名を追加してプッシュすることもできます）

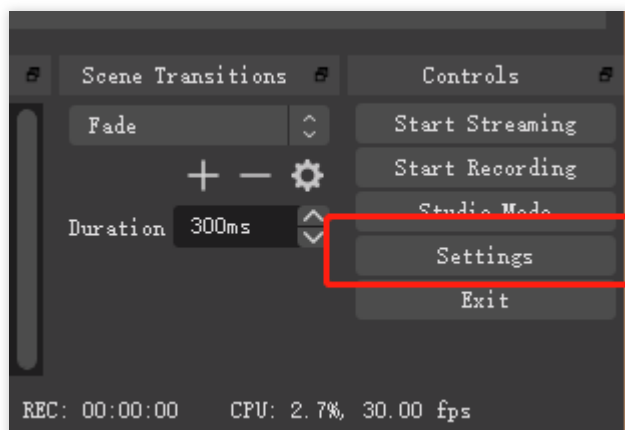
プッシュアドレスの取得

1. CSSコンソールにログインし、[アドレスジェネレータ](#)を選択し、以下のとおり設定します：
 - i. [プッシュドメイン名](#)または[再生ドメイン名](#)の生成タイプを選択します。
 - ii. ドメイン管理下で追加済みのドメイン名を選択します。
 - iii. AppNameは、同一ドメイン名の複数Appを区別するために使用されるアドレスパスです。デフォルト値はliveです。
 - iv. カスタマイズされたストリーム名StreamNameを入力します（例：`live`）。
 - v. アドレスの期限切れ時間を選択します（例：`2020-06-09 23:59:59`）。
2. [アドレスを生成する](#)をクリックして、OBSプッシュアドレスを取得できます。

OBSオンラインプッシュ

手順1：プッシュアドレスの設定

1. OBSを開いて、1番下のツールバーの**コントロール**>**設定**ボタンから設定画面に入ることができます。

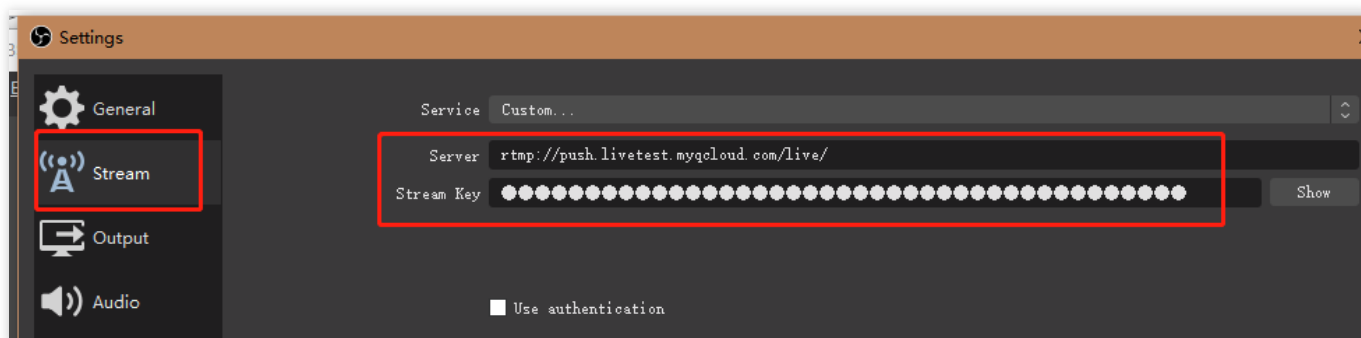


2. **プッシュ**をクリックしてプッシュ設定タブに進み、サービスタイプにカスタムストリームメディアサーバーを選択します。

3. サーバーとストリームパスワードに取得した**プッシュアドレス**を記入します。

- サーバー：「OBSプッシュアドレス」である `rtmp://domain/AppName/` とマッピングします。
- ストリームパスワード：「OBSプッシュ名」である `StreamName?`

`txSecret=xxxxx&txTime=5C1E5F7F` とマッピングします。



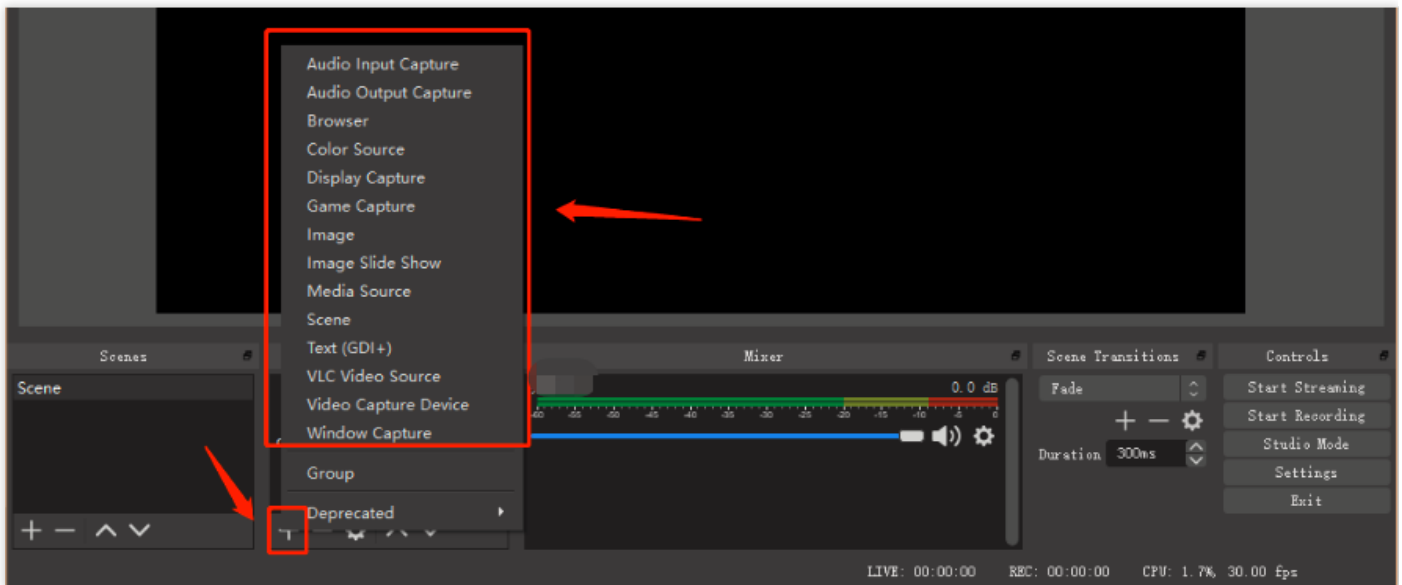
4. **確定**をクリックして設定情報を保存します。

手順2：プッシュガイドの設定

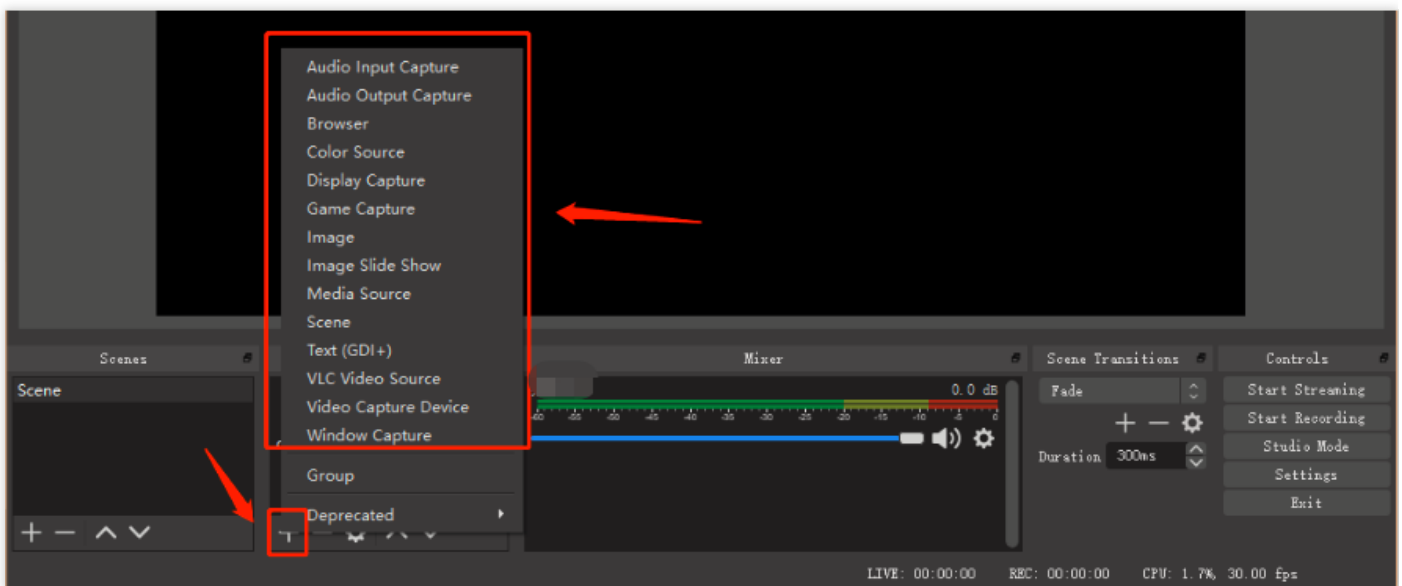
説明：

ビットレートと録画などの設定が必要である場合は、1番上のツールバーをクリックして、**ツール**>**ウィザード**の**自動設定**を選択し、OBSのガイドに従ってプッシュを設定できます。

1. 1番下のツールバーのソースタグを確認します。



2. +ボタンをクリックして、必要に応じて入力ソースを選択します（例：ディスプレイのキャプチャ）。



よく使用するライブブロードキャスト入力ソースの説明

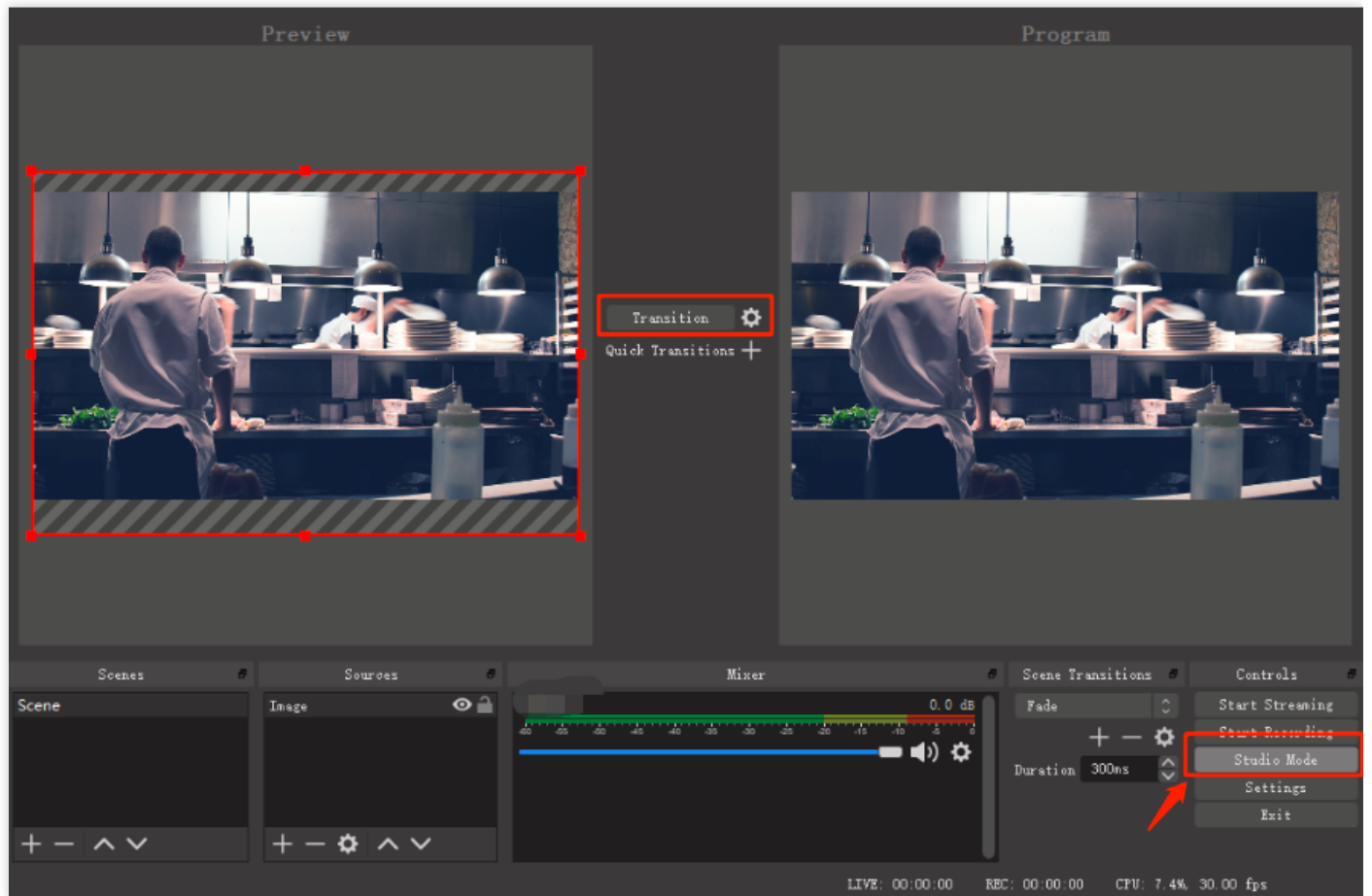
| 入力ソース | 説明 |
|------------|--------------------------|
| 画像 | 単一画像のライブブロードキャストに適しています。 |
| 画像のスライドショー | 複数の画像をループまたは順番に再生できます。 |

| 入力ソース | 説明 |
|--------------|---|
| シーン | 各種の大きなライブブロードキャスト効果を実現します。この時、別のシーンをソース元として現在のシーンに追加すれば、シーン全体の挿入を実現できます。 |
| メディアソース | ローカルのビデオをアップロードできます。またローカルのVODビデオファイルに対し、ライブブロードキャスト化の処理を行うことができます。 |
| テキスト | ライブブロードキャストウィンドウにリアルタイムでテキストを追加します。 |
| ディスプレイキャプチャ | デスクトップの操作をリアルタイムかつ動的にキャプチャし、PCデスクトップ上のすべての操作をライブブロードキャストすることができます。 |
| ゲームキャプチャ | 指定されたソースからのゲームのライブブロードキャストを許可します。大小ゲームのライブブロードキャストに適しています。 |
| ウィンドウキャプチャ | 選択したウィンドウに応じてリアルタイムにキャプチャを行うことができます。ライブブロードキャストは現在のウィンドウのコンテンツのみを表示し、その他ウィンドウはライブストリーミングのキャプチャを行いません。 |
| カラーソース | このソースを使用して1つのカラーブロックを自分のシーンの中に追加し、背景色とすることができます。このカラーブロックは透明度を調整でき、画面全体を透明の色彩にすることができます。 |
| ビデオキャプチャデバイス | ビデオをリアルタイムかつ動的にキャプチャするデバイスです。撮影された画面をリアルタイムにライブブロードキャストできます。 |
| オーディオ入力キャプチャ | オーディオのライブブロードキャストイベントに使用されます（オーディオ入力デバイス）。 |
| オーディオ出力キャプチャ | オーディオのライブブロードキャストイベントに使用されます（オーディオ出力デバイス）。 |

手順3：スタジオモードをオンにする

スタジオモードでは、現在のライブブロードキャストコンテンツをリアルタイムに編集し、かつ編集プロセスの中でトランジションアニメーションによるシーン切り替えが可能です。これによりユーザーエクスペリエンスへの影響が極めて少ない状態で、ライブブロードキャストコンテンツの編集を行うことができます。

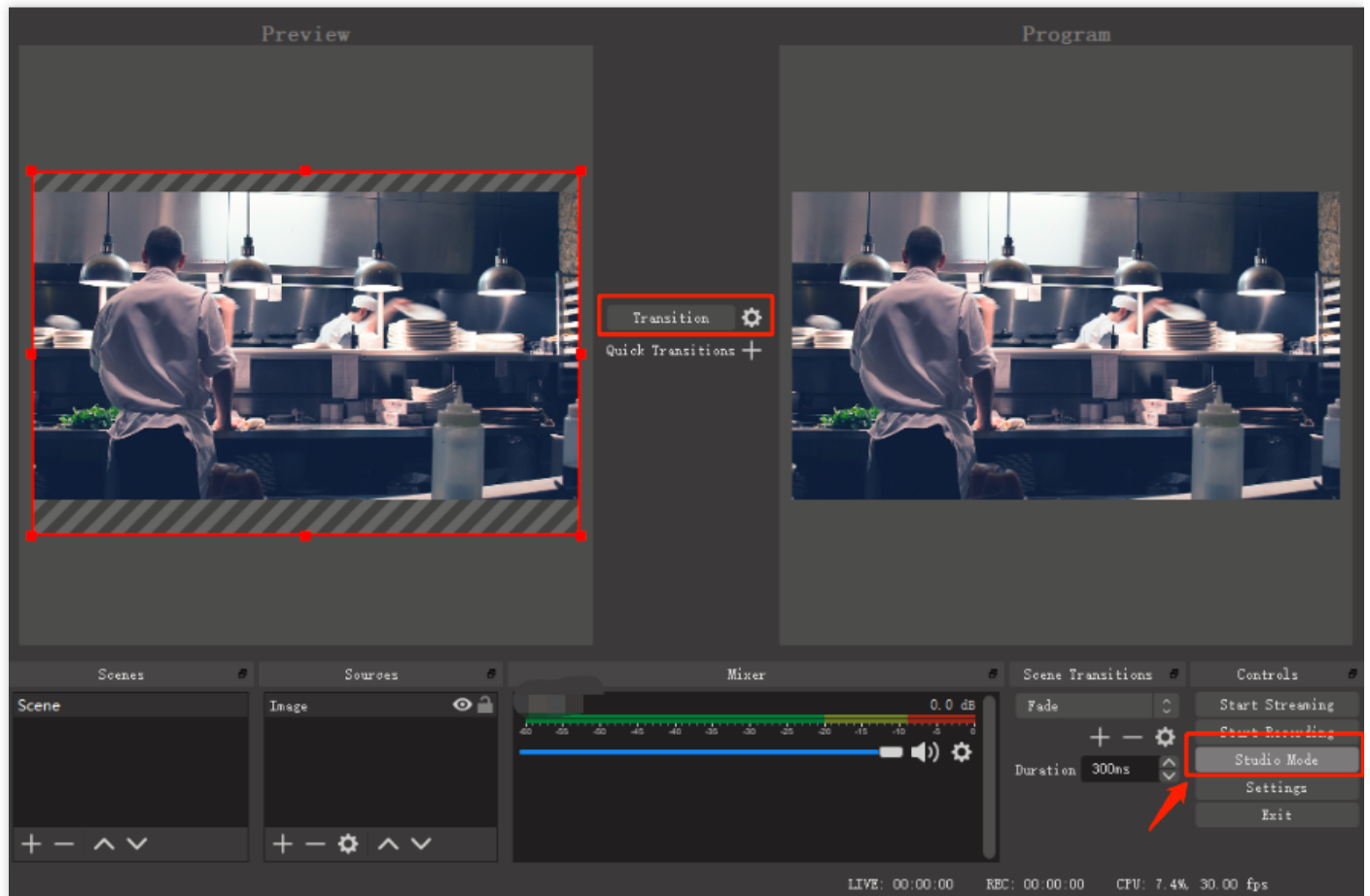
1. 1番下のツールバーの**コンソール**タブを確認して、**スタジオモード**をクリックします。
2. **トランジションエフェクト**をクリックして、プレビューウィンドウのライブブロードキャスト画面を出力ウィンドウに移動できます。



手順4：ライブブロードキャストプッシュ

1. 1番下のツールバーのコンソールを確認します。

2. **プッシュを開始する**をクリックして、ビデオストリームを設定されたプッシュアドレスにプッシュできます。



説明：



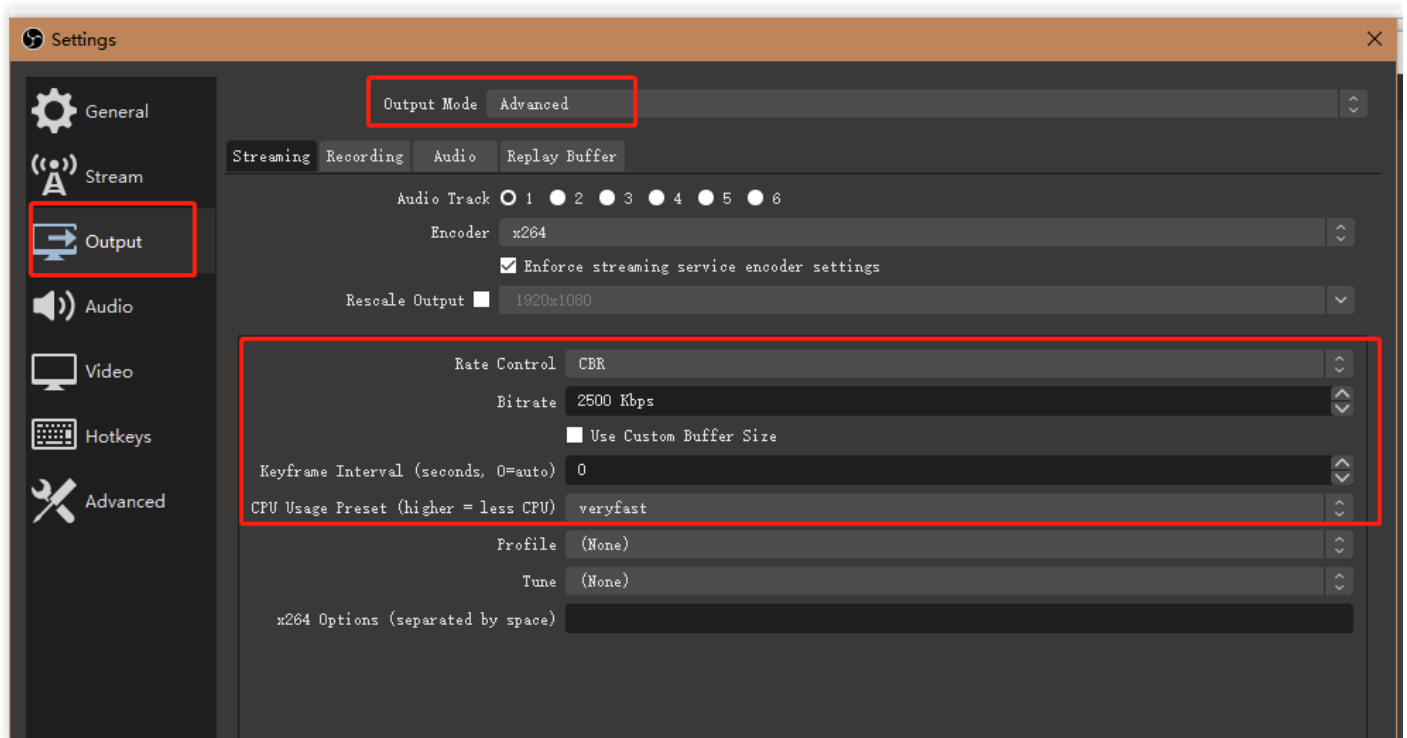
- 1番下に 緑色のランプが表示されると、プッシュに成功したことを示します。
- プッシュを停止するには、**プッシュの停止** > **プッシュの停止**をクリックします。

その他のプッシュ設定

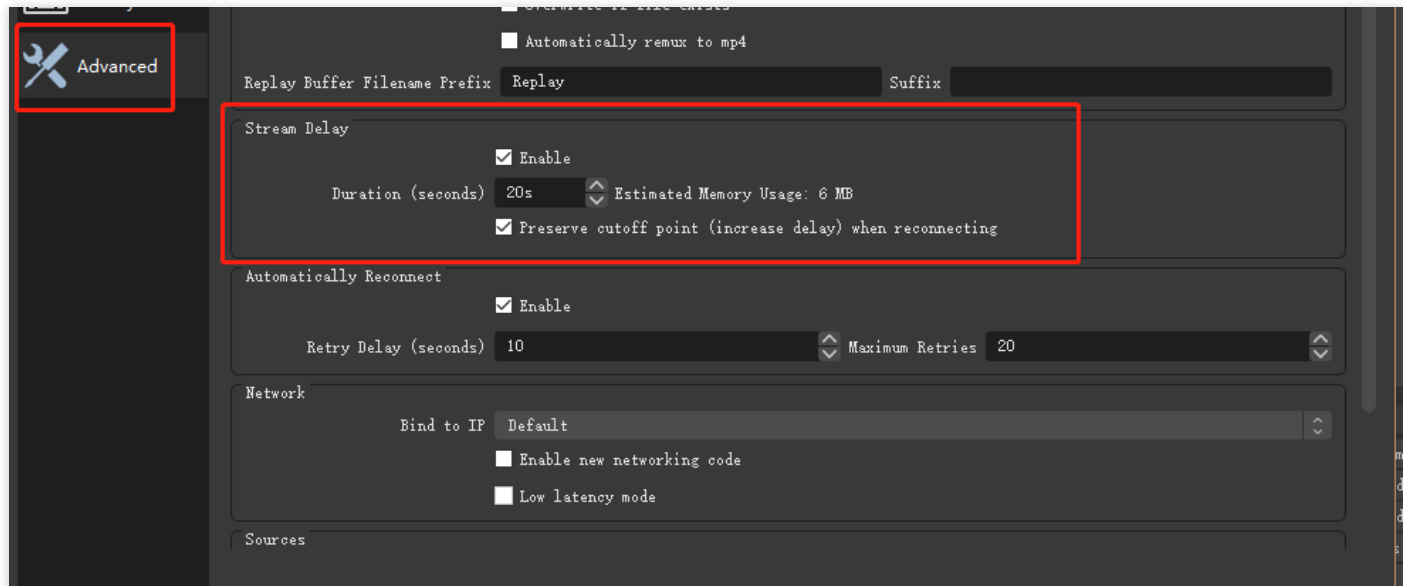
- ライブブロードキャストのディレイに影響する関連設定

1. **コンソール** > **設定** > **出力**を実行します。

2. 出力モードを詳細として選択すると、キーフレーム間隔などを設定できます。設定方法は下図のとおりです：



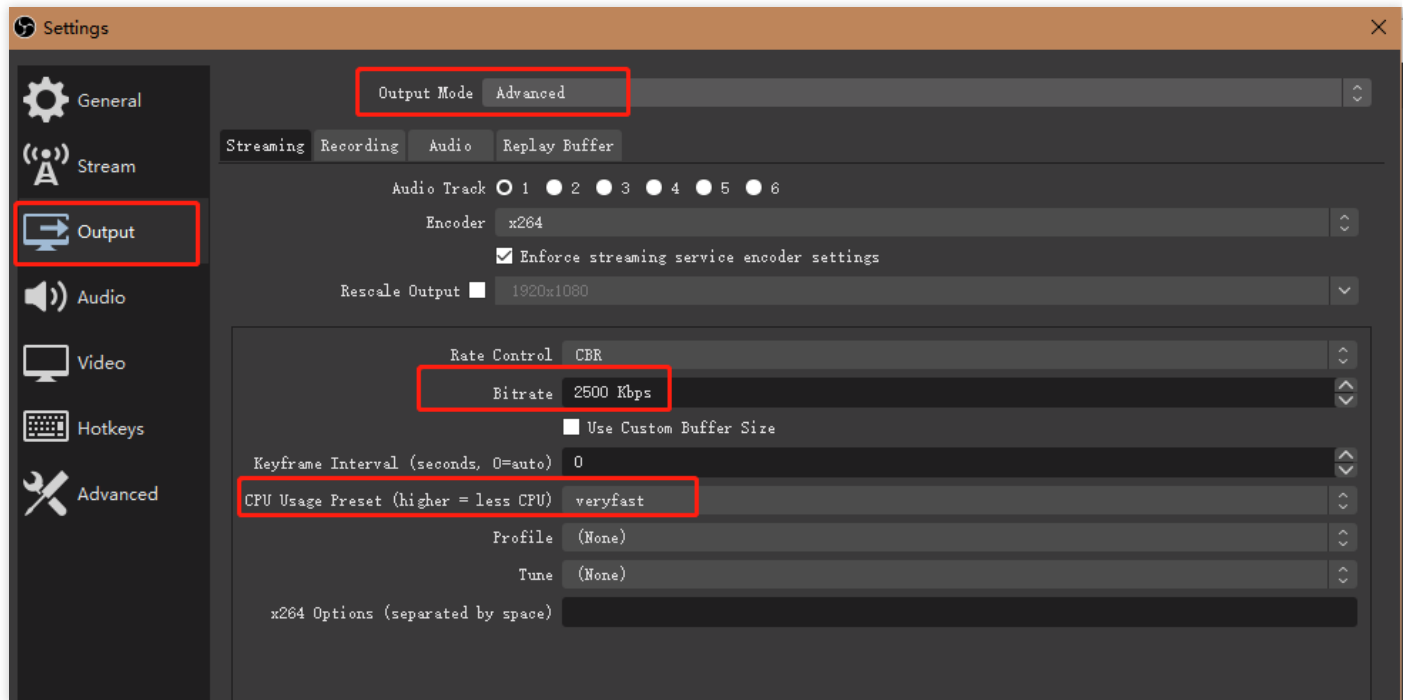
3. 同時に、左側のナビゲーションバーで詳細を選択すると、ストリームディレイを設定できます。



- ローカルのライブブロードキャストのレコーディングの関連設定
ライブブロードキャストの際にローカルレコーディングのニーズがある場合は、この時にローカルにレコーディングのバックアップを行う方法を選択できます。設定方法は次のとおりです：

1. コンソール>設定>出力をクリックして、出力設定タブに進みます。

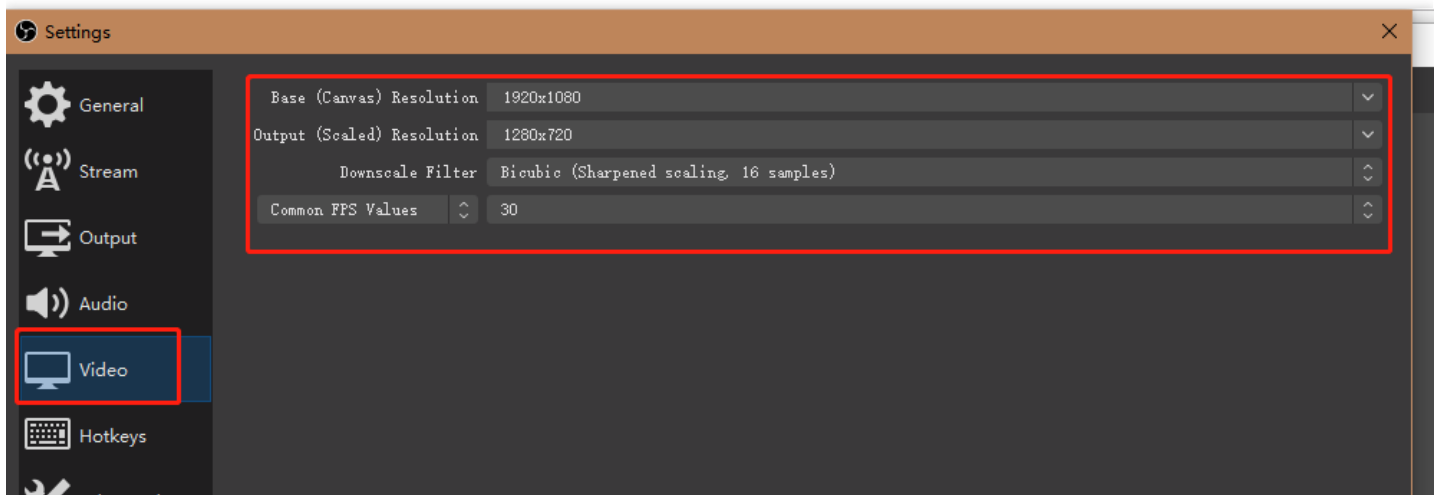
2. 録画設定バーで対応する設定を行い、**確定**をクリックしてレコーディングファイルをローカルに保存できます。



3. **ビデオ**をクリックして、ビデオ解像度とフレームレート情報を設定します。下図のとおりです：

説明：

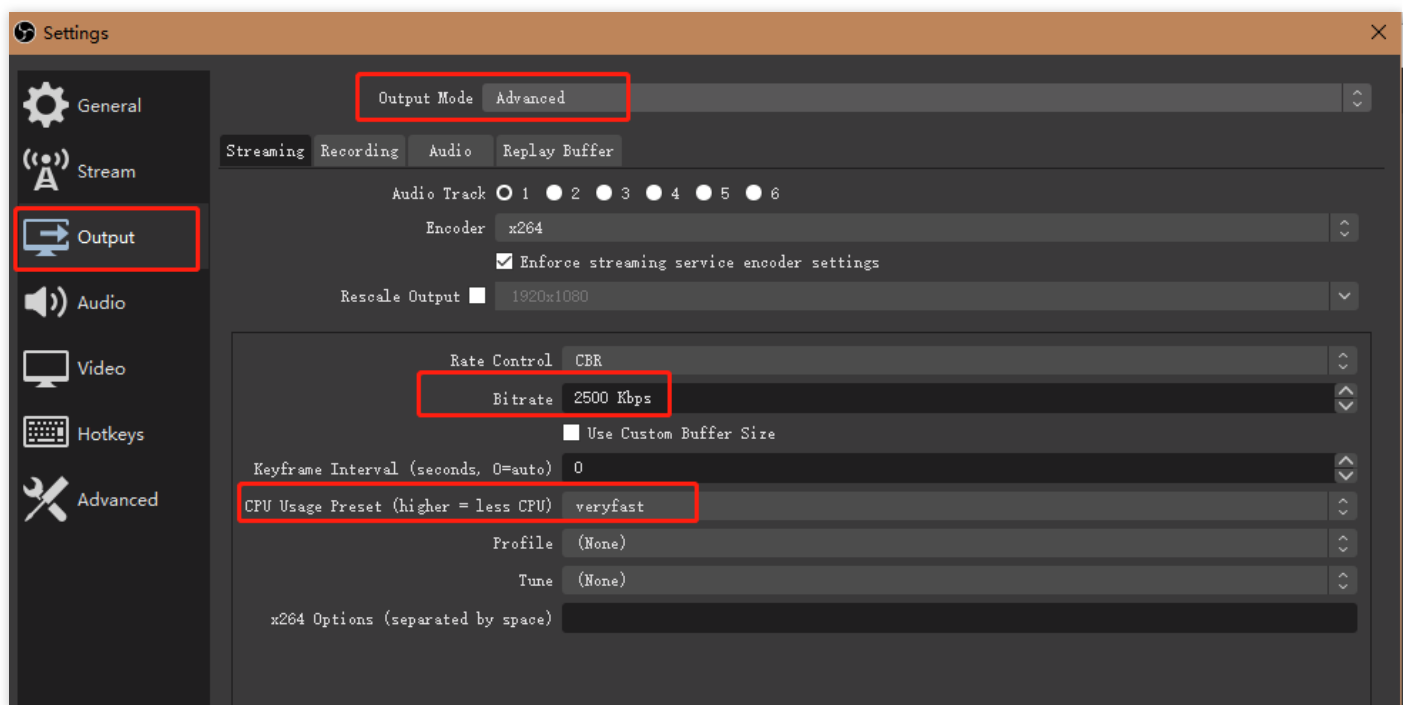
解像度は視聴者が見る画面の精細度を決定し、解像度が高いほど画面は鮮明になります。FPSはビデオフレームレートで、ビデオ視聴のなめらかさを制御します。スタンダードのビデオフレームレートとしては24フレーム - 30フレームがあり、16フレーム未満の画面は見るとラグ感があります。またゲームはフレームレートに求める基準が高いため、通常は30フレーム未満のゲームでは画面の連続性が失われます。



プッシュトランスコーディングの関連設定

プッシュプロセス中にビデオのビットレートを変更する必要がある場合、具体的な操作は次のとおりです：

1. 1番下の **コントロール>設定** ボタンから設定画面に入ります。
2. **出力** タブをクリックして、出力モードに「シンプル」を選択します。
3. ビデオビットレートを記入し、**確定** をクリックします。



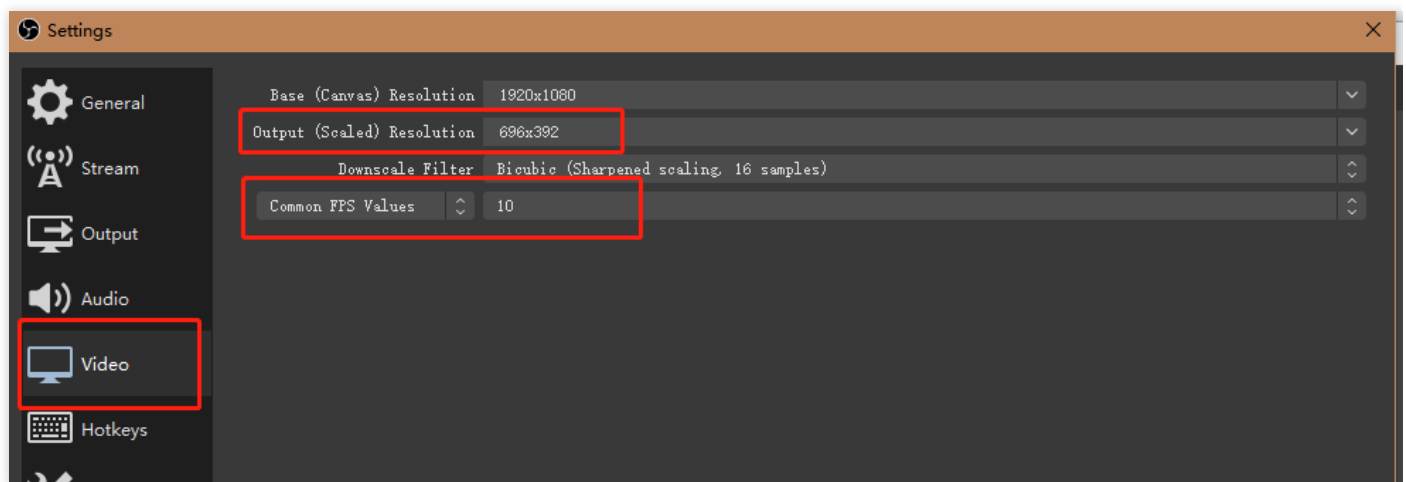
その他の操作の説明

純音声プッシュ

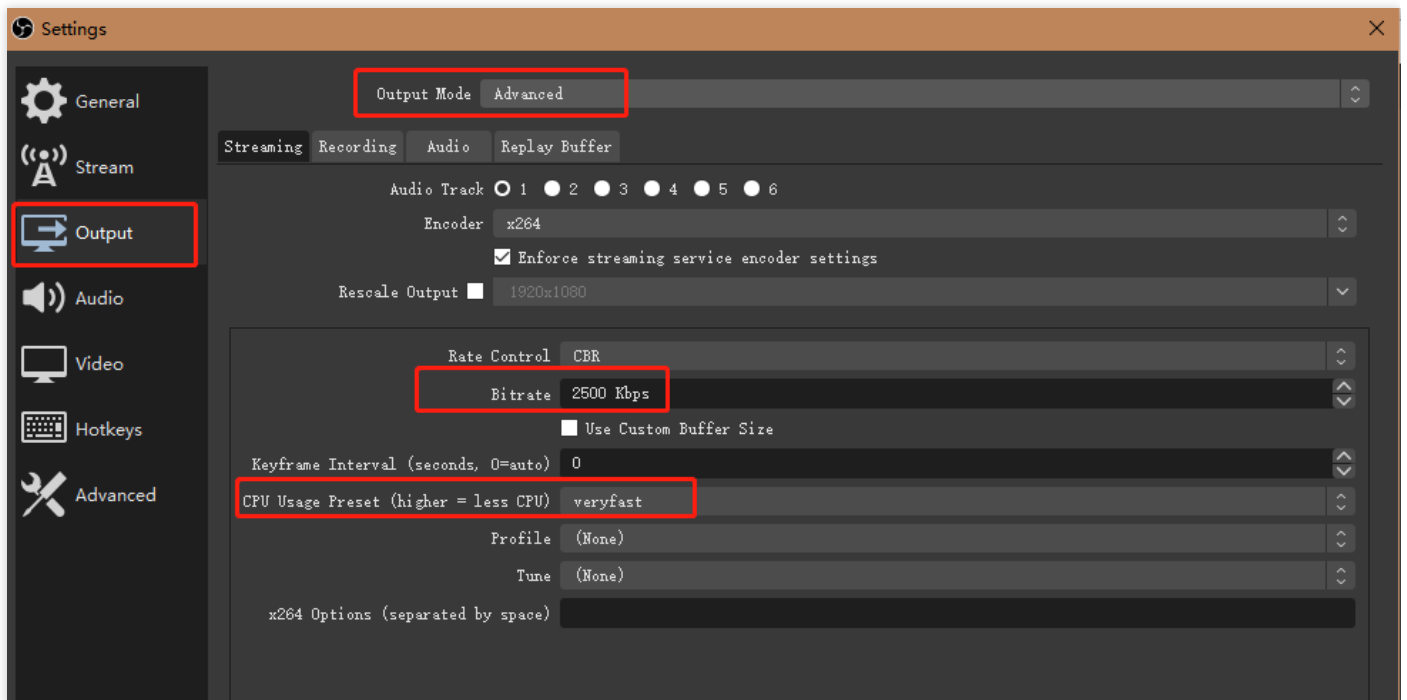
OBS公式フォーラムの回答によれば、現在OBS Studio 23.2.1およびそれ以前のバージョンでは純音声プッシュをサポートしていません。

近似する音声プッシュを実現するために、以下の手順を参考に設定を行うことができます。本方法は主に、静的キャンバス（黒画面または画像）を使用して、ビデオを代替します。帯域幅を下げたい場合は、ビデオのフレームレートとビットレートを下げ、これにより純音声プッシュの状態に近づけることができます。ただし、ライブストリーミングの中には依然としてビデオデータがあります。

1. **プッシュガイドの設定**を参考にして、**オーディオ入力キャプチャ**の入力ソースを入力ソース元として追加します。ビデオ入力ソースまたは画像をビデオ入力ソース元として追加することはできません。
2. **コンソール>設定>ビデオ**ページに進みます。
3. **出力（スケーリング）解像度**を最小に、**共通FPS値（フレームレート）**を最小に選択し、**確定****をクリックして設定を保存します。



4. **出力**をクリックして出力設定ページに進みます。下図のとおり設定します。ここで、**ビットレート**を最小に設定する必要があります。**確定**をクリックして設定を保存します。



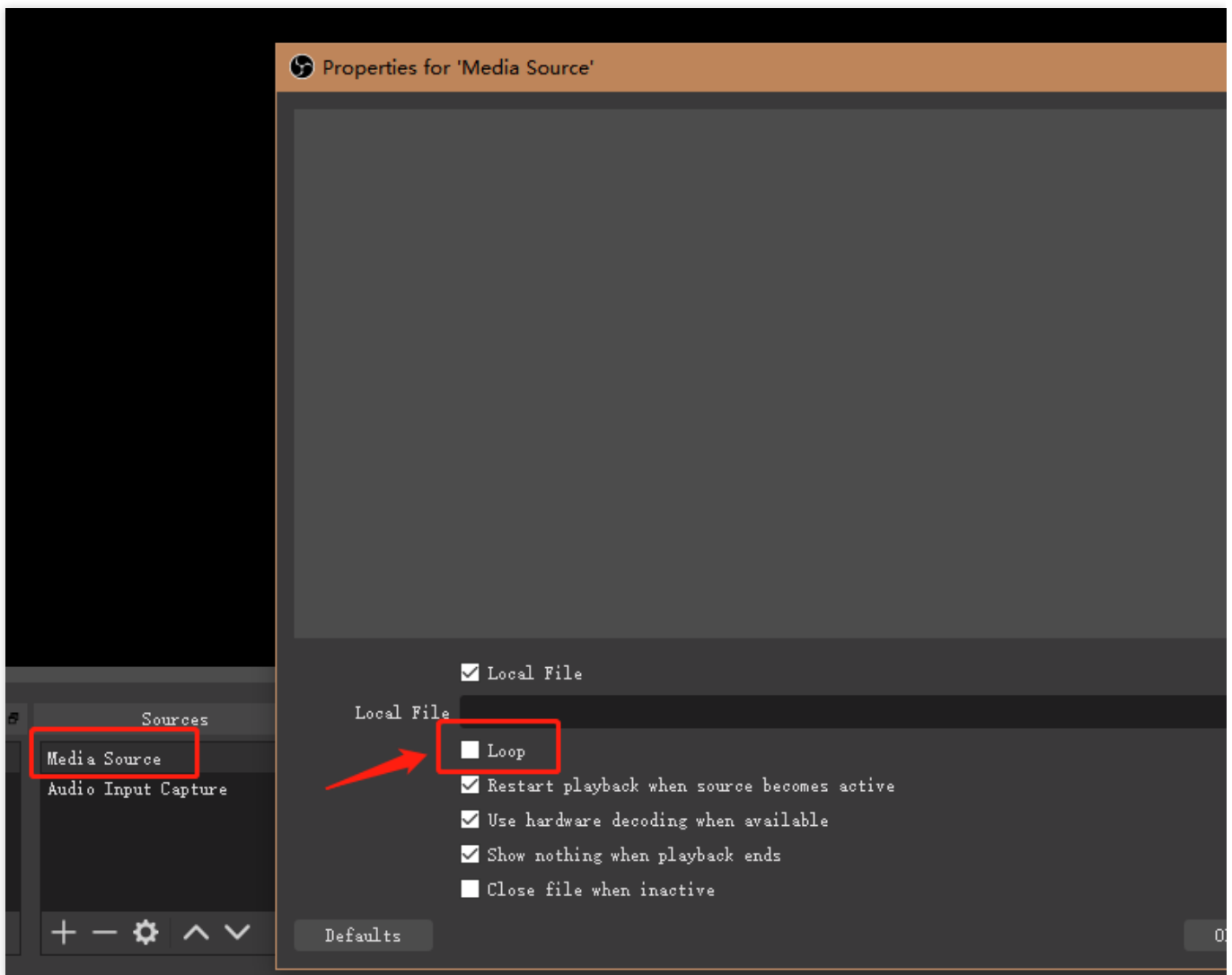
5. [OBSプッシュオンラインプッシュ](#)の中の手順を参考にしてライブブロードキャストプッシュを開始します。そうすると音声聞こえ、ビデオのコンテンツは黒画面または画像となります。これと同時に、ビデオのビットレートは最小となり、使用される帯域幅も低下します。

ビデオの繰り返し再生

• 単独のファイルのループ再生設定方法

- i. 1番下のツールバーのソースタブでは、**+をクリックしてメディアソースを追加し、ポップアップのローカルファイルからループ再生を必要とするビデオファイルを選択し、ループにチェックを入れて、確定**をク

リックします。



- ii. [OBSプッシュ設定](#) の手順にしたがい、[プッシュページ](#)のサーバーとストリームキーの項目を設定します。プッシュが開始されると単独のファイルの繰り返し再生が可能です。

プッシュが完了すると、次の方法でプルして再生し、正常にプッシュされたかどうかを確認できます：

- PC端末では、[VLCプレーヤー](#)を使用したプルをサポートします。
- モバイル端末では、[MLVB SDK](#)を使用して再生を実行することをサポートします。

説明：

MLVB SDKはCSSサービス (LVB) のモバイルシナリオでの拡張です。MLVBは、主にクラウド接続向けのライブブロードキャスト (LVB) サービスと比較して、RTMP SDKに基づく「迅速な統合ソリューション」を提供するだけでなく、標準ライブブロードキャスト (LVB)、ライブイベントブロードキャスト (LEB)、クラウドオンデマンド (VOD)、インスタントメッセージング (IM)、クラウドオブ

ジェクトストレージ (COS) などのマルチクラウドサービスを統合した「オールインワンソリューション」を提供しています。

その中で、**ライブイベントブロードキャスト**(Live Event Broadcasting、LEB) (超低遅延ライブブロードキャスト) は、超低遅延の再生シナリオにおける標準ライブブロードキャストプロトコルの拡張です。従来のライブブロードキャストプロトコルよりもディレーが低く、視聴者にミリ秒レベルのライブブロードキャスト視聴体験を提供します。eラーニング、スポーツイベントのライブブロードキャスト、オンライン質疑応答など、より高いディレーパフォーマンスを必要とする特定のシナリオのニーズを満たすことができます。

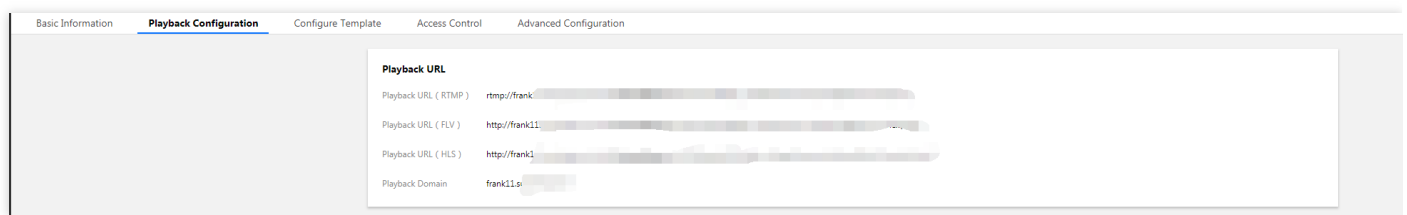
VCL プッシュ

最終更新日：：2021-07-27 20:31:03

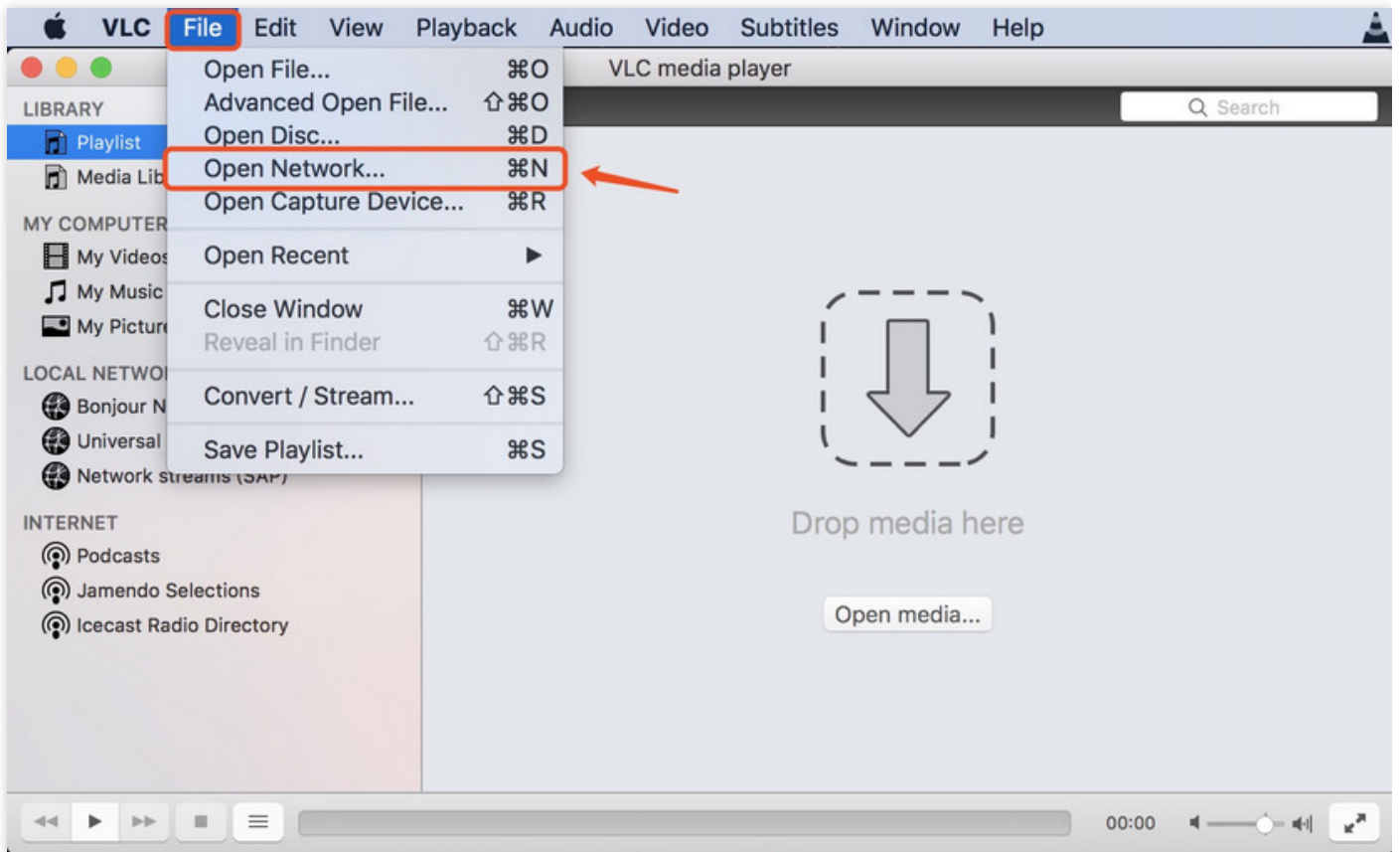
VLC media player は、フリーで提供されている、オープンリソース、マルチプラットフォーム対応のマルチメディアプレーヤーおよびフレームワークであり、ほとんどのマルチメディアファイル、DVD、音声CD、VCD、各タイプのストリーミングプロトコルの再生に利用できます。OS X、Windows、Linux、iOS、Android、Chrome OS等のOSをサポートし、RTMP、FLV、M3U8などのあらゆる一般的なライブストリーミング形式に適用することが可能です。最新版のソフトウェアは、[VLC公式サイトでダウンロード](#) できます。

VLCビデオ再生手順

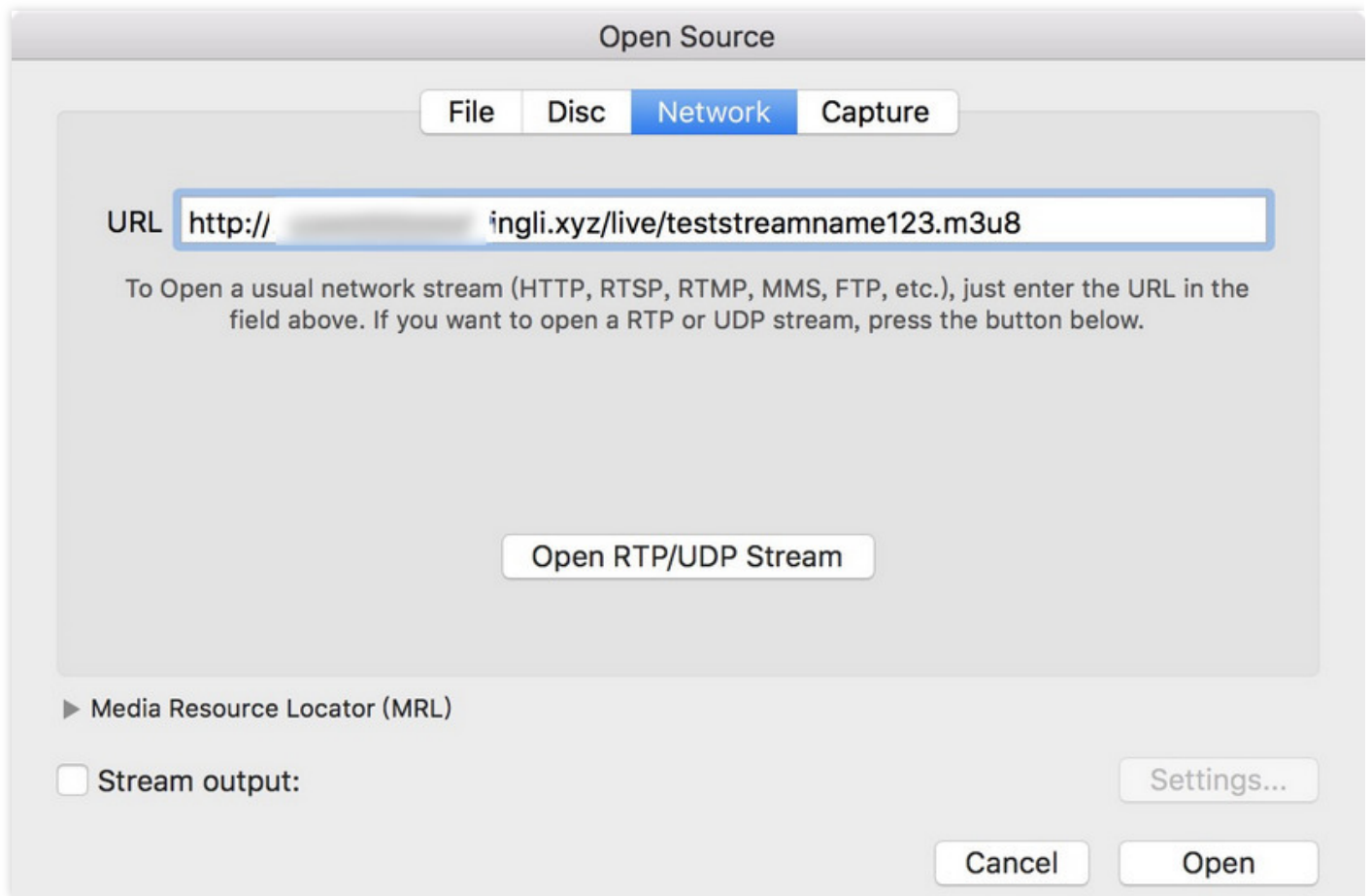
1. [ビデオストリーミングの再生アドレス](#)を取得し、[Domain Management](#) で、自身のニーズに合わせて、ICP登録済みの再生ドメイン名を選択し、再生設定で表示される形式に従って再生アドレスのドッキングを行います。



2. VLCメイン画面を開き、**【File】 > 【OpenNetwork】** を選択します（中国語版は**【メディア】 > 【ネットワークストリームを開く】**）。



3. ポップアップしたダイアログボックスにライブストリーミングの再生アドレスを入力します。



4. 入力完了後、【Open】をクリックして再生を確認します。再生アドレスによって正常にプルされている場合は、再生のボックスが直接ポップアップします。

