

Cloud Streaming Services

Live Video Broadcasting (LVB)

Product Documentation



Copyright Notice

©2013-2024 Tencent Cloud. All rights reserved.

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

Trademark Notice



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

Service Statement

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

Contents

Live Video Broadcasting (LVB)

- Overview

- Use Cases

- Getting Started

- SDK Integration

Live Video Broadcasting (LVB)

Overview

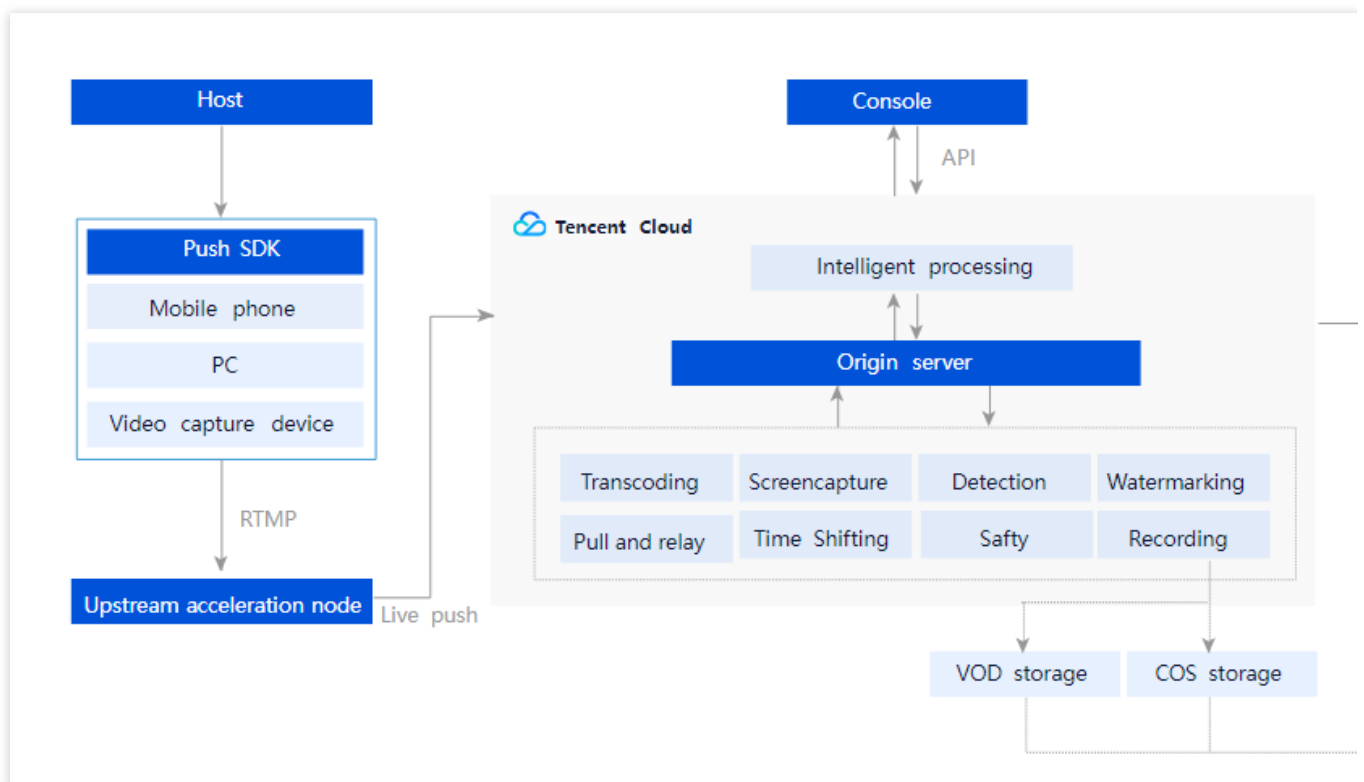
Last updated : 2023-10-30 11:22:07

Leveraging Tencent's suite of audio/video media platforms, global network of cache nodes, and leading AI technologies in audio and video, Live Video Broadcasting (LVB) delivers Tencent Cloud's live streaming capabilities to users. With professional, stable, and agile features such as live push, transcoding, distribution, and playback, LVB provides ultra-low latency, ultra-high image quality, and ultra-high performance even under very high concurrency.

Note

You can use LVB after you activate the CSS service. For more information, see [LVB - Getting Started](#).

Product Architecture



Features

Full suite of features

LVB supports RTMP push, HLS origin servers, and other live streaming sources. It provides a multi-platform live streaming SDK and supports adaptive bitrate streaming, beauty filters, audio processing, instant streaming, stutter reduction, screen recording, and seamless definition switching. Playback is supported on all platforms, and you can also customize your own player.

Global acceleration

Tencent Cloud has over 2,000 CDN cache nodes across all regions of the Chinese mainland, covering not only major internet service providers, but also over a dozen small- and medium-sized ISPs. Outside the mainland, it operates more than 800 cache nodes in over 70 countries and regions across North America, Europe, and Southeast Asia.

Professional video processing

LVB provides cloud-based transcoding and processing services that can handle vast amounts of multimedia data. It can transcode audio and video files to different formats suitable for OTT services or playback on PC and mobile devices. It also offers features such as thumbnail generating, editing, content moderation, and encryption.

Security and hotlink protection

LVB provides all-round protection to prevent unauthorized playback of your videos, including IP blocklists, referrer blocklists, secure HTTPS acceleration, URL encryption, and other live streaming encryption technologies. It also offers a professional digital rights management (DRM) solution to safeguard your video assets.

Intelligent learning

With powerful AI-based recognition technology, LVB can learn and analyze the content of your images or videos. With this technology, LVB can intelligently automate image recognition, speech recognition, content moderation, subtitling, and other video tasks.

Statistics analysis

Based on big data analysis covering scenarios in various industries, LVB enables real-time control of push quality and identification of playback issues. It monitors metrics such as bandwidth, traffic, requests, and concurrent connections in real time and provides detailed statistics to help you scale your business.

Demo

Use our demo to try out LVB features.

iOS	Android



Pricing

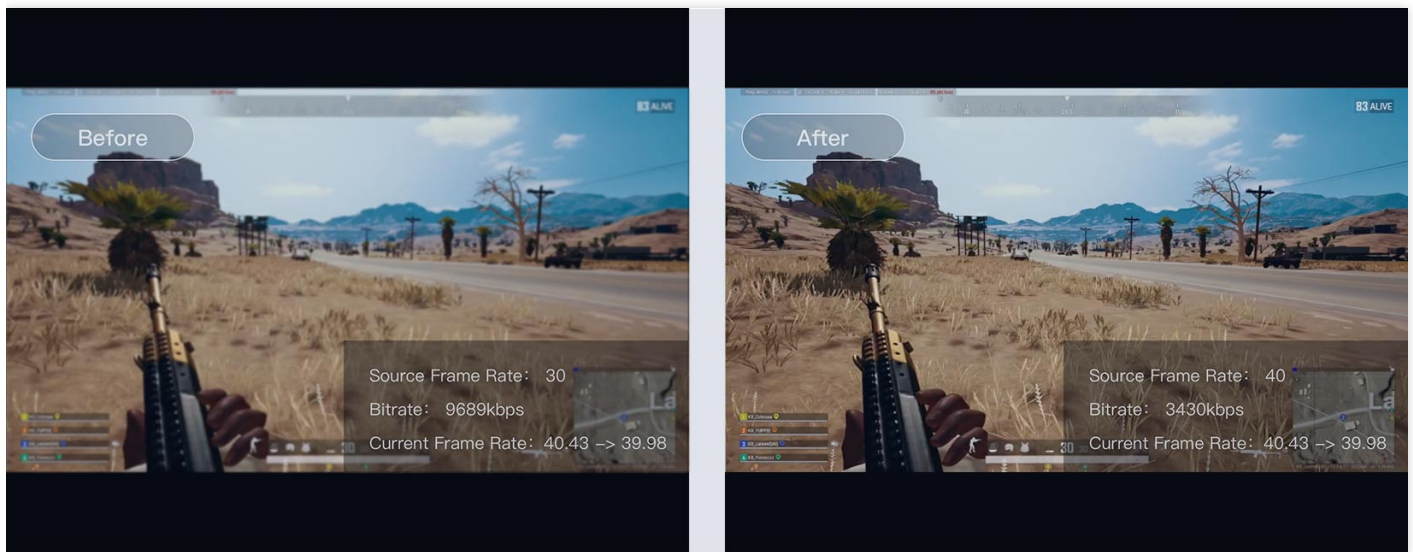
LVB services are billed in daily pay-as-you-go mode based on the traffic/bandwidth consumed for push and playback (upstream and downstream usage). For details, see [Pricing Overview](#).

Use Cases

Last updated : 2021-09-02 10:50:50

Live Game Streaming

LVB provides upstream acceleration, high-quality transcoding, and accelerated delivery services for live game streaming platforms. You can integrate LVB with MLVB SDK to quickly realize smooth live game streaming with low lag and low stutter.



Live Show

LVB features instant streaming, color filters, and manual exposure, which are ideal for live shows. Its out-of-the-box ease of use readily helps boost the buzz around live shows.



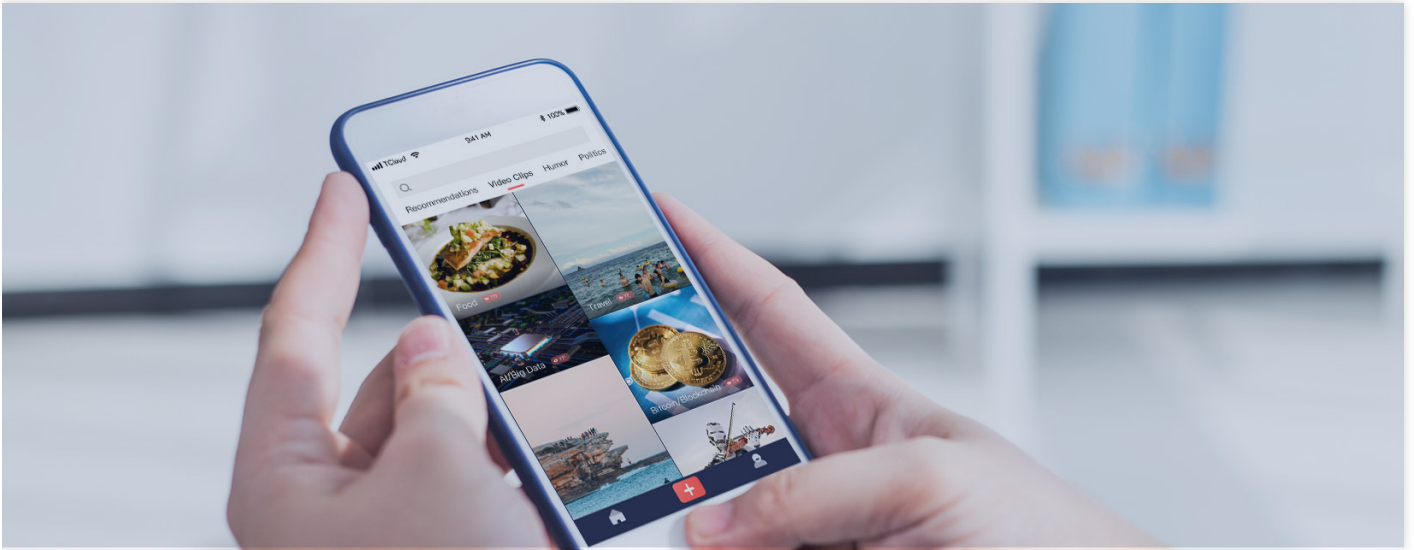
Online Education

LVB provides a one-stop service for online education. Your well-crafted teaching materials can be uploaded and downloaded through the agile console and APIs, and are protected with player passwords and dynamic URL-based hotlink protection. In addition, the open player APIs make it easy to implement various teaching features such as setting playback time limits and disallowing progress bar dragging.



Radio, TV, and New Media

LVB provides a complete set of live video streaming services for the radio, TV, and new media industries. It can be quickly accessed by professional live streaming platforms of news, entertainment and variety shows, etc., helping them easily transform from traditional media to internet-based new media.



Live Shopping

LVB helps with the growth of ecommerce platforms by enabling merchants to display items with more details and assisting consumers in making informed decisions, thus reducing marketing costs and boosting sales.



Getting Started

Last updated : 2023-11-20 16:56:45

This document describes how to get started with CSS. Before trying out CSS, you're advised to read the [Pricing Overview](#) of CSS to get familiar with its **billable items** and **prices**.

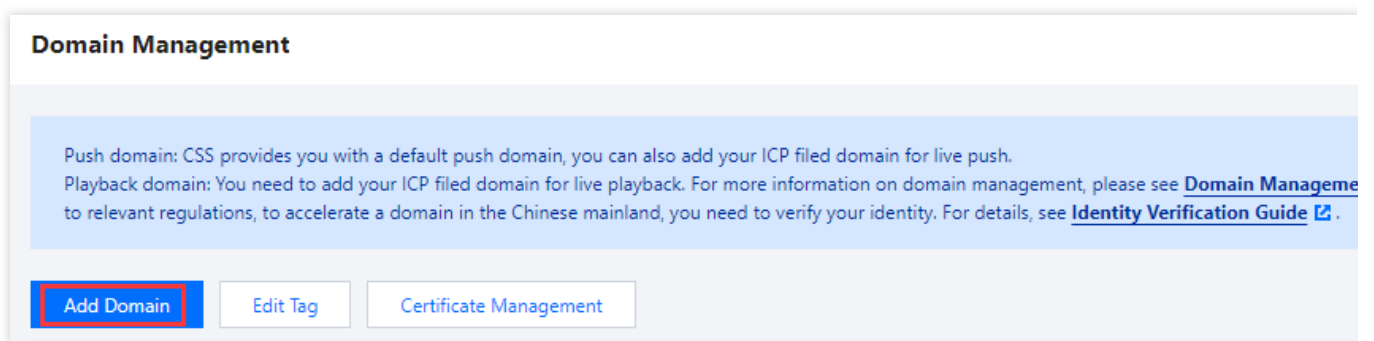
Preparations

1. Sign up for a [Tencent Cloud account](#) and complete [identity verification](#). Unverified users cannot purchase CSS instances in the Chinese mainland.
2. Go to the [CSS service activation page](#), indicate your consent to the "Tencent Cloud Service Terms", and click **Apply for Activation** to activate the CSS service.

Step 1. Add domain names

To use CSS, you should have at least one **push domain name** and one **playback domain name**. You cannot use one domain name for both push and playback.

1. Prepare your own domain names. For domain names in Chinese mainland regions, ICP filing is required.
2. Log in to the CSS console, enter [Domain Management](#), and click **Add Domain**.



2.1 Tags are used to categorize and manage resources from different dimensions. If the existing tags do not meet your requirements, you can also go to the [Tag Console](#) to manage tags uniformly.

2.2 Click **Add domain**.

Add Domain

1 Basic settings > 2 CNAME Configuration

You can add 54 more push or playback domain names.

Type * Push Domain

Domain Name *

Tag (optional) ⓘ Tag Key Tag Value X

+ Add

Add domain

Note

CSS provides a test domain name `xxxx.livepush.myqcloud.com`. You can use it to test live push, but you're not advised to use it as the push domain name for business purposes.

After the domain name is added successfully, you can view its information in the domain name list in **Domain Management**. For how to manage it, see [Domain Management](#).

For more information on live streaming domain names, please see [Basic CSS Features](#).

3. Once your domain name is added, the system will assign it a CNAME domain name (suffixed with `.txlivecdn.com` or `.tlivepush.com`), which cannot be accessed before you complete the CNAME configuration at your DNS service provider. After the CNAME configuration takes effect, you can use CSS. The following example shows you how to add a CNAME record by assuming Tencent Cloud is your DNS service provider:

3.1 Log in to the [Tencent Cloud Domain Name Service console](#).

3.2 Find the target domain name and click **Resolve**.

3.3 On the domain name resolution page, click **Add Record**.

3.4 In the new line, enter the domain name prefix as the host record, select CNAME as the record type, and enter the CNAME domain name as the record value.

3.5 Click **Save**.

Note

After a CNAME record is successfully added, it takes some time for the CNAME configuration to take effect. If the configuration fails, you cannot use CSS.

After the CNAME configuration succeeds, you can see that the status symbol of the CNAME address in **Domain Management** in the CSS console has changed to



If the CNAME configuration failure persists, consult your DNS service provider.

For more information on how to configure with other DNS service providers, please see [Configuring CNAME for Domain Name](#).

Step 2. Get push URL

1. Select **CSS Toolkit** > **Address Generator**.
2. On the **Address Generator** page, configure as follows:
 - 2.1 Select **Push Domain** for **Domain Type**.
 - 2.2 Select a push domain name you added in **Domain Management**.
 - 2.3 Enter an **AppName**. It is `live` by default.
 - 2.4 Enter a custom `StreamName`, such as `liveteststream`.
 - 2.5 You need to choose an encryption type based on your security requirements and performance considerations. The encryption type can be either **MD5** or **SHA256**, with **MD5** being the default option.
 - 2.6 Select the URL expiration time, such as `2023-09-10 19:54:01`.
3. Click **Generate Address** to generate a push address.

Address Generator

URL Type * ☒ Push Address ☐ Playback Address ☐ Push and playback URLs NEW i

Select domain name *

AppName * i

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

StreamName * i

Only supports letters, digits, and symbols

Type ☒ MD5 ☐ SHA256

Expiration Time 📅

The actual expiration time of the playback URL is the timestamp selected plus the validity period

[Generate Address](#)

[Splice manually](#)

[History](#)

Note

In the push URL, `live` is the default `AppName`, `txSecret` is the signature for playing back the stream, and `txTime` is the URL expiration time.

Here is another way to generate a push URL: In [Domain Management](#), find the push domain name you want use to generate a push URL, click **Manage**, select **Push Configuration**, enter an expiration time for the URL and a custom `StreamName`, and click **Generate Push Address**.

Before generating a push URL, you can create [templates](#) and bind them to your push domain. For the prices of CSS value-added services, see [Pricing Overview](#).

Step 3. Push live stream

You can enter the generated push URL into the push software as appropriate for your use cases.

For push on PCs, you're advised to use OBS. For detailed directions, see [Push via OBS](#).

For push on web, you're advised to use [Web Push](#): select a push domain name, enter a custom `StreamName`, select a URL expiration time, turn the camera on, and click **Start Push**.

For push on mobile devices, download and install Tencent Video Cloud Demo, open it, select **Mobile Live Video Broadcasting > Push (Camera)**, enter the push URL into the address box manually or by scanning the QR code,

and then tap the start icon in the bottom-left corner to push the stream.

Note

You can integrate Tencent Cloud [MLVB SDK](#) into customized apps for live push.

Step 4. Get playback URL

1. After push succeeds, select [Stream Management](#) > **Live Streams**, view the status of the push URL, and click **Test** to play back the stream online.
2. Select **CSS Toolkit** > [Address Generator](#) to get a playback address and configure as follows:
 - 2.1 Select **Playback Domain** for **Domain Type**.
 - 2.2 Select a playback domain name you added in **Domain Management**.
 - 2.3 Enter an **AppName**. It is `live` by default.
 - 2.4 Enter the `StreamName` in the push URL to play back the corresponding stream.
 - 2.5 You need to choose an encryption type based on your security requirements and performance considerations. The encryption type can be either **MD5** or **SHA256**, with **MD5** being the default option.
 - 2.6 Select the URL expiration time, such as `2023-09-10 20:08:03`.
 - 2.7 To generate a playback URL for a transcoded stream, please select a transcoding template which must be bound to the domain name in the playback URL. For directions on how to bind a transcoding template, please see [Live Transcoding > Binding Domain Name](#).
 - 2.8 Click **Generate Address** to generate a playback address.

Address Generator

URL Type *

☐ Push Address ☒ Playback Address ☐ Push and playback URLs NEW i

Select domain name *

AppName *

live

i

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

StreamName *

liveteststream

i

Only supports letters, digits, and symbols

Type

☒ MD5 ☐ SHA256

Expiration Time

2023-09-10 20:08:03

Push address expiration time is the setting time

Transcoding Template

Please select

If you select a transcoding template, the generated playback address will be the live streaming a transcoding template to generate the address.

Generate Address

Splice manually

History

3. You can use the following methods to test whether a live stream can be played back normally in different use cases: For playback test on PCs, you're advised to use tools such as [VLC](#). For more information, please see [Live Playback](#). For playback test on mobile devices, you're advised to download and install Tencent Video Cloud Toolkit App, select **Mobile Live Video Broadcasting > LVB Playback**, enter the playback URL into the address box manually or by scanning the QR code, and tap the play icon in the bottom-left corner.

Note

To push/play back a stream on an app, you can integrate the [MLVB SDK](#) into the app to supplement the CSS service. If you encounter any problem during the trial period, please see [FAQs](#).

Step 5.Generate Push and Playback Address Group

1. Go to **CSS Toolkit > Address Generator**.
2. Select the address type as **Push and playback URLs**.

3. Select the **Push Domain Name** and **Playback Domain Name** that you have added to Domain Management.
4. Enter an **AppName**. It is `live` by default.
5. Enter a StreamName, such as `liveteststream`.
6. You need to choose an encryption type based on your security requirements and performance considerations. The encryption type can be either **MD5** or **SHA256**, with **MD5** being the default option.
7. Select the URL expiration time, such as `2023-09-10 20:08:03`.
8. Select an existing transcoding template (optional).
9. Click **Generate Address**.

Address Generator

URL Type *

☐ Push Address ☐ Playback Address ☒ Push and playback URLs NEW ?

Push Domain *

Playback Domain *

AppName *

live ?

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

StreamName *

liveteststream ?

Only supports letters, digits, and symbols

Type

☒ MD5 ☐ SHA256

Expiration Time

2023-09-10 20:08:03

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

Transcoding Template

Please select

If you select a transcoding template, the generated playback address will be the live streaming address after transcoding. If you wan
a transcoding template to generate the address.

Generate

Splice manually

History

Operations

To enable **live recording**, please create a recording template and bind it to a domain name first. For more information, please see [Creating Recording Template](#).

To enable **live transcoding**, please create a transcoding template and bind it to a domain name first. For more information, please see [Creating Transcoding Template](#).

To enable **live watermarking**, please create a watermark template and bind it to a domain name first. For more information, please see [Creating Recording Template](#).

To enable **live screencapture and porn detection**, please create a screencapture and porn detection template and bind it to a domain name first. For more information, please see [Creating Screencapture and Porn Detection Template](#).

To enable **live streaming callbacks**, you can create a callback template and associate it with the domain for configuration. For related documentation, please refer to [Creating a Callback Template](#).

To enable **live streaming standby content**, you can create a standby content template and associate it with the domain for configuration. For related documentation, please refer to [Creating a Standby Content Template](#).

To enable **live stream mixing**, please call the stream mixing API [CreateCommonMixStream](#).

FAQs

[What are the differences between push, live streaming, and video on demand?](#)

[What push protocols are supported?](#)

[What playback protocols are supported?](#)

[What does a playback URL consist of?](#)

[How do I splice multiple live stream URLs?](#)

SDK Integration

Last updated : 2022-06-10 15:35:38

CSS provides LVB as a professional, stable, and fast cloud streaming service for common use cases. You can integrate the MLVB SDK to apps, web, and other platforms to quickly enable live streaming.

Apps

You can integrate the MLVB SDK with apps on iOS or Android clients for live push and playback.

- **Live push on apps:** support capturing camera or mobile phone screens and then pushing the content to the CSS platform using the RTMP protocol.
- **Live playback on apps:** support playback protocols including RTMP, HTTP-FLV, and HLS. You can integrate the MLVB SDK with LVB to quickly build live streaming apps.

Note :

The MLVB SDK uses CSS, IM, TRTC and other services for low-latency audiovisual communication for multiple parties. It offers co-anchoring for interaction between viewers, and other viewers who don't join co-anchoring can also watch the live streaming.

Web

You can use the following ways to achieve live push and playback on your websites:

- **Live push on web:** you can use the standard WebRTC protocol to design and mux streams, and insert code snippets to websites to enable live push. For details, please see [WebRTC Push](#).

Note :

WebRTC push uses the Opus audio codec. If you use a standard live streaming protocol (RTMP, HTTP-FLV, or HLS) for playback, the CSS backend will automatically convert the audio streams to AAC format to ensure normal playback, which will incur audio transcoding fees. For details, please see [Live Transcoding > Audio Transcoding](#).