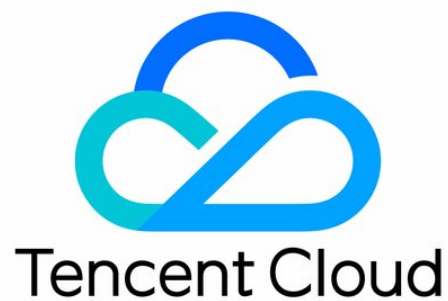


Instant Messaging

Getting Started with Demos

Product Documentation



Copyright Notice

©2013-2019 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

 Tencent Cloud

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

Getting Started with Demos

- Free Demos





- Demo Quick Start

- Quick TWebLive Run

Getting Started with Demos

Free Demos

Last updated : 2021-01-04 14:56:22

Android Code Scanning via Browser	iOS	Mini Program	Live-Streaming E-commerce Scenarios on Mini Programs	Web
				<div data-bbox="1098 752 1426 804" style="background-color: #007bff; color: white; padding: 5px; text-align: center;">Basic Web Chat Scenarios</div> <div data-bbox="1035 853 1490 904" style="background-color: #007bff; color: white; padding: 5px; text-align: center;">Web Live-streaming Interaction Scenarios</div>

Note :

For more information on the Demo, see [Quick Demo Operation](#).

Demo Quick Start

Last updated : 2021-01-05 19:23:37

This document introduces how to quickly run through the IM demo.

Step 1: Create an App

1. Log in to the [IM console](#).

Note :

If you already have an app, record its SDKAppID and [obtain key information](#).

A Tencent Cloud account can create a maximum of 100 IM apps. If you want to create more apps, [disable and delete](#) an unwanted app first and then create a new one. **Once an app (along with its SDKAppID) is deleted, the service it provides and all its data are lost. Proceed with caution.**

2. Click **+Add App**.
3. In the **Create App** dialog box, enter your app name, and click **OK**.
After creation, you can see the status, service version, SDKAppID, creation time, and expiry time of the new app on the overview page of the console. Record the SDKAppID.

Step 2: Obtain Key Information

1. Click the target app card to enter the basic configuration page of the app.
2. In the **Basic Information** area, click **Display Key**, and then copy and save the key information.

Note :

Please store the key information properly to prevent leakage.

Step 3: Download and Configure the Demo Source Code

1. Download the IM Demo project. For more information about the specific download address, see [SDK Download](#).

Note :

To respect the copyright of emoji design, the downloaded Demo project does not contain sliced images of major emoji elements. You can use your local emoji packs to configure the code. Unauthorized use of the emoji packs in the IM Demo may infringe on design copyrights.

2. Open the project in the terminal directory and find the `GenerateTestUserSig` file.

Platform	Relative Path to File
Android	Android/app/src/main/java/com/tencent/qcloud/tim/demo/signature/GenerateTestUserSig.java
iOS	iOS/TUIKitDemo/TUIKitDemo/Debug/GenerateTestUserSig.h
Mac	Mac/TUIKitDemo/TUIKitDemo/Debug/GenerateTestUserSig.h
Windows	cross-platform/Windows/IMApp/IMApp/GenerateTestUserSig.h
Web (general)	H5/dist/debug/GenerateTestUserSig.js
Mini Program	WXMini/dist/wx/debug/GenerateTestUserSig.js

3. Set relevant parameters in the `GenerateTestUserSig` file:

Note :

In this document, an Android project is opened by using Android Studio as an example.

- SDKAPPID: set it to the actual SDKAppID obtained in [Step 1](#).
- SECRETKEY: enter the actual key information obtained in [step 2](#).

Note :

In this document, the method to obtain UserSig is to configure a SECRETKEY in the client code. In this method, the SECRETKEY is vulnerable to decompilation and reverse engineering. Once your SECRETKEY is leaked, attackers can steal your Tencent Cloud traffic. Therefore, **this method is only suitable for locally running a demo project and feature debugging.**

The correct `UserSig` distribution method is to integrate the computing code of `UserSig` into your server and provide an app-oriented API. When `UserSig` is needed, your app can send a request to the business server to obtain the dynamic `UserSig`. For more information, see [How to Generate UserSig on the Server](#).

Step 4: Compile and Run the Demo

Use the IDE on each end to compile and run the demo. For more information, see the `README.md` file in the corresponding directory of the demo project cloned in [Step 3](#).

The compilation and running of the iOS and Mac Demo require the integration of pods. The detailed steps are as follows:

1. Run the following command on the terminal to check the pod version:

```
pod --version
```

If the system indicates that no pod exists or that the pod version is earlier than 1.7.5, run the following command to install the latest pod.

```
// Change gem sources.
gem sources --remove https://rubygems.org/
gem sources --add https://gems.ruby-china.com/
// Install pods.
sudo gem install cocoapods -n /usr/local/bin
// If multiple Xcodes are installed, run the following command to choose an Xcode version (usually the latest Xcode version should be chosen):
sudo xcode-select -switch /Applications/Xcode.app/Contents/Developer
// Update the local pod library.
pod setup
```

2. Run the following commands on the terminal to install dependent libraries:

```
//iOS
cd iOS/TUIKitDemo
pod install
//Mac
cd Mac/TUIKitDemo
pod install
```

If installation fails, run the following commands to update the local CocoaPods repository list:

```
pod repo update
```

3. Compile and run the demo:

- iOS: go to the iOS/TUIKitDemo folder and open `TUIKitDemo.xcworkspace` to compile and run the demo.
- Mac: go to the Mac/TUIKitDemo folder, and open `TUIKitDemo.xcworkspace` to compile and run the demo.

Enabling Advanced Features

- [Enabling Video Calls](#)
- [Enabling Group Livestreaming](#)
- [Enabling Live Rooms](#)

Reference

- [Pricing](#)

Quick TWebLive Run

Last updated : 2021-01-04 14:57:07

This document describes how to quickly run the experience demo of the Tencent Cloud Web ILVB component.

Prerequisites

You have [signed up](#) for a Tencent Cloud account and completed [identity verification](#).

Directions

Step 1: create an application

1. Log in to the [Instant Messaging console](#).
2. Click **Add Application**.
3. In the **Create Application** dialog box, enter an application name and click **OK**.
After the application is created, you can view the status, service version, `SDKAppID`, creation time, and expiration time of the new application on the overview page of the console. Record the `SDKAppID`.

Step 2: obtain the key information and activate the TRTC service

1. Click the target application card to go to its basic configuration page.
2. In the **Basic Information** area, click **Display Key**, and then copy and save the key information.

Note :

Please store the key information properly to prevent leakage.

3. Activate the TRTC service.

Step 3: download and configure the demo source code

1. Download the demo project of the Tencent Cloud Web ILVB component from the [download address](#).
2. Open the `TWebLive/dist/debug/GenerateTestUserSig.js` file and set relevant parameters:
 - SDKAPPID: set it to the actual SDKAppID obtained in Step 1.
 - SECRETKEY: enter the actual key obtained in Step 2.

Note :

- The method of local UserSig calculation is used for local development debugging only. Do not directly publish it to your online systems. Once your `SECRETKEY` is disclosed, attackers can use your Tencent Cloud traffic without authorization.

- The correct `UserSig` distribution method is to integrate the computing code of the `UserSig` into your server and provide an app-oriented API. When `UserSig` is required, your app can send a request to the business server to obtain the dynamic `UserSig`. For more information, see [How to Generate UserSig on the Server](#).

Step 4. Run the demo

Open the `index.html` file in the `dist` directory with Chrome to run the demo.

Note :

- Generally, the demo needs to be deployed on your server and then accessed through `https://domain name/xxx`. You can also build a server locally and access the demo through `localhost:port`.
- Currently, the desktop version of Chrome offers more comprehensive support for the features of the TRTC SDK for desktop browsers. Therefore, Chrome is recommended for the demo.

WebRTC needs to use a camera and mic to capture audio and video. During the trial, you may receive relevant prompts from Chrome and you should click **Allow**.

Environment Requirements

- Please use the latest version of Chrome.
- TWebLive uses the following ports for data transfer, which should be added to the allowlist of the firewall. After configuring these ports, use the [official demo](#) to check whether the ports work.
 - TCP port: 8687
 - UDP ports: 8000, 8080, 8800, 843, 443, and 16285
 - Domain name: `qcloud.rtc.qq.com`

FAQs

1. Only public and private key information can be obtained when I try to view the encryption key. How do I get the encryption key?

Starting from TRTC SDK v6.6 (August 2019), the new signature algorithm HMAC-SHA256 is used. For applications created before this version, you need to upgrade the signature algorithm before you can get a new encryption key.

2. What should I do if the client error "RtcError: no valid ice candidate found" occurs?

This indicates that the TRTC desktop browser SDK failed to pass Session Traversal Utilities for NAT (STUN). Please check your firewall policy based on the environment requirements.

3. What should I do if the client error "RtcError: ICE/DTLS Transport connection failed" or "RtcError: DTLS Transport connection timeout" occurs?

This indicates that the TRTC desktop browser SDK failed to build a media transfer channel. Please check your firewall policy based on the environment requirements.

4. What should I do if a 10006 error occurs?

If "Join room failed result: 10006 error: service is suspended, if charge is overdue, renew it" is displayed, please check whether your TRTC application service is available.

Log in to the [TRTC console](#), click the application you created, click **Account Info**, and you can view the service status in the account info tab.

References

- [TWebLive API Guide](#)
- [Online Demo](#)