

# 移动直播 Server端集成 产品文档



腾讯云

**【版权声明】**

©2013-2018 腾讯云版权所有

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

**【商标声明】**

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

**【服务声明】**

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

---

## 文档目录

### Server端集成

入门导读

派发URL ( 防盗链签名 )

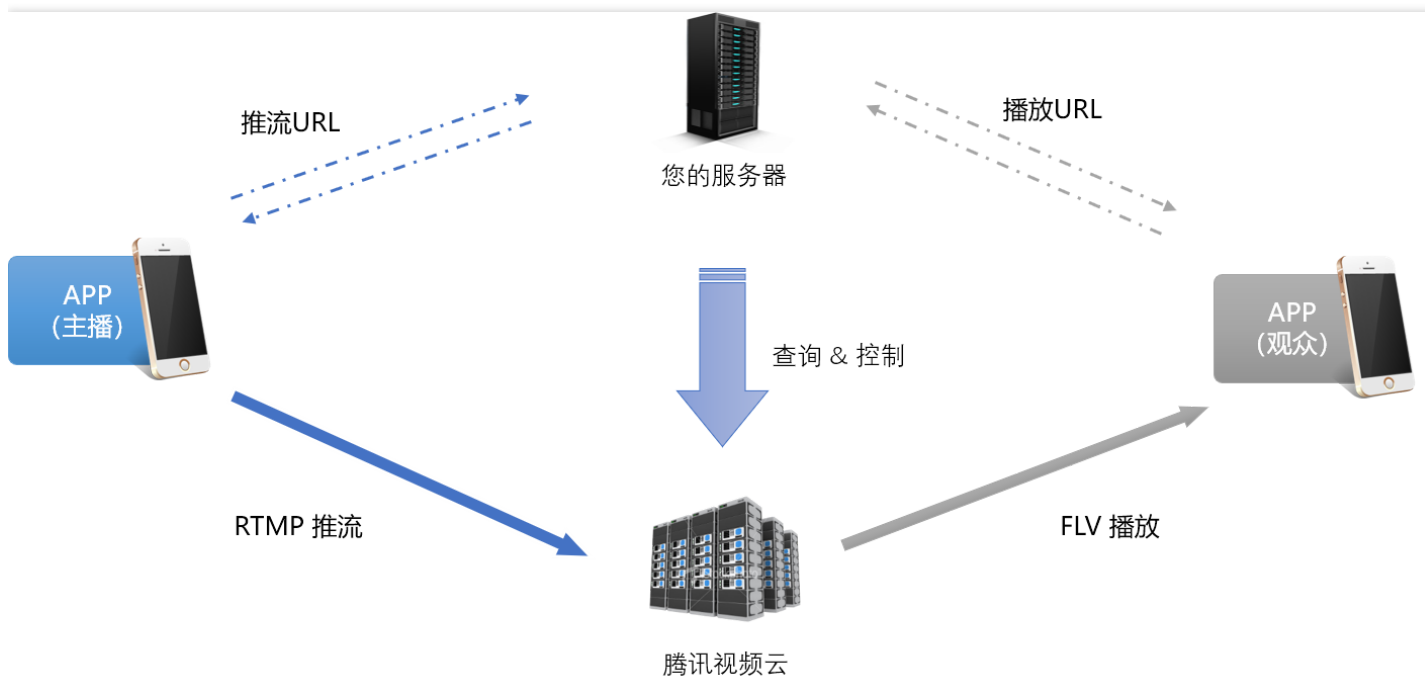
派发UserSig ( IM云通讯 )

操控直播流 ( REST API )

# Server端集成 入门导读

最近更新时间：2018-07-23 16:11:47

由于目前腾讯视频云的定位是 PAAS 而不是 SAAS（通俗的说就是只做平台不做具体的业务），所以您的业务后台工程师就需要参与到对接过程中来，后台工程师的工作分成三个层面：



## 派发 URL

不管您的产品是想做单场次的活动直播，还是主播可以随时开播的直播平台，由后台派发 URL 都要比把 URL 写死在 APP 里要灵活的多。

这里说的“派发”指的是：在 APP（主播端）准备推流（iOS | Android）时返回推流 URL 给 APP，在 APP（观众端）准备播放（iOS | Android）时返回播放 URL 给 APP。

这一部分的详细介绍，可以参考文档 - [派发URL](#)。

## 派发 UserSig

UserSig是使用腾讯云通讯服务（IM）所需的一种安全票据，所以如果您准备使用腾讯云 IM 服务实现聊天室功能，就需要您的后台工程师生成 UserSig 给返回给终端 APP；如果您已经有自己的 IM 解决方案（也就是已经有自己的聊天室了），可以免去这部分工作。

这一部分的详细介绍，可以参考文档 - [派发UserSig](#)。

## 操控直播流

如果您希望查询当前有多少正在直播中的直播流，查询它们的状态，或者对其进行一定的管理，可以使用下面的 REST API 按需进行二次开发：

API	功能介绍
<a href="#">Get_LiveStat</a>	统计信息查询 - 查询推流和播放相关信息
<a href="#">Get_LivePushStat</a>	统计信息查询 - 查询推流相关信息
<a href="#">Get_LivePlayStat</a>	统计信息查询 - 查询播放相关信息
<a href="#">Get_LivePushStatHistory</a>	获取推流历史信息
<a href="#">Get_LivePlayStatHistory</a>	获取播放统计历史信息
<a href="#">Live_Channel_GetStatus</a>	仅查询某条流的状态信息（旧版本接口）
<a href="#">Live_Channel_SetStatus</a>	对某条流实行 <b>禁播</b> 操作，主要用于鉴黄场景
<a href="#">Live_Tape_GetFilelist</a>	查询某条流在直播过程中的 <b>录制</b> 文件列表
<a href="#">Live_Queue_Get</a>	查询某条流在直播过程中的 <b>截图</b> 文件列表
<a href="#">Live_Channel_GetChannelList</a>	查询频道列表
<a href="#">Live_Channel_GetLiveChannelList</a>	查询直播中频道列表
<a href="#">mix_streamv2.start_mix_stream_advanced</a>	云端混流（用于将多条直播流的画面进行合并）操作接口
<a href="#">channel_manager</a>	暂停并延迟恢复——可针对某路流禁止推流
<a href="#">Live_Tape_Start</a>	创建录制任务——可实现定时录制任务或者实时视频录制
<a href="#">Live_Tape_Stop</a>	结束录制任务

这一部分的详细介绍，可以参考文档 - [操控直播流](#)。

# 派发URL（防盗链签名）

最近更新时间：2018-07-25 09:30:29

## 为何要派发URL？

直播推流需要推流 URL，直播播放需要播放 URL，不管您的产品是想做单场次的活动直播，还是主播可以随时开播的直播平台，由后台派发 URL 都要比把 URL 写死在 APP 里要灵活的多。

这里说的“派发”指的是：在 APP（主播端）准备推流（iOS | Android）时返回推流 URL 给 APP，在 APP（观众端）需要播放（iOS | Android）时返回播放 URL 给 APP。

## URL的组成？

### 推流URL的组成

如下是一条标准的推流URL，它由三个部分组成：



- **直播码**

也叫房间号，推荐用随机数字或者用户ID，注意一个合法的直播码需要拼接 BIZID 前缀。

- **txTime**

何时该URL会过期，格式是十六进制的UNIX时间戳，比如 5867D600 代表 2017年1月1日0时0分0分过期。我们的客户一般会将 txTime 设置为当前时间 24 小时以后过期，**过期时间不要太短**，当主播在直播过程中遭遇网络闪断时会重新恢复推流，如果过期时间太短，主播会因为推流 URL 过期而无法恢复推流。

- **txSecret**

防盗链签名，防止攻击者伪造您的后台生成推流URL，计算方法参考[防盗链签名的计算](#)。

- 示例代码

[直播控制台](#)>>[直播码接入](#)>>[推流生成器](#)页面下半部分有示例代码（PHP和Java两个版本）演示如何生成防盗链地址。

## 播放URL的组成

播放URL的拼接跟推流URL一样简单，只是需要把子域名从 **livepush** 改成 **liveplay**：

rtmp	<code>rtmp://8888.liveplay.myqcloud.com/live/8888_test_123</code>
flv	<code>http://8888.liveplay.myqcloud.com/live/8888_test_123.flv</code>
hls	<code>http://8888.liveplay.myqcloud.com/live/8888_test_123.m3u8</code>

## 防盗链签名

### 什么是防盗链签名？

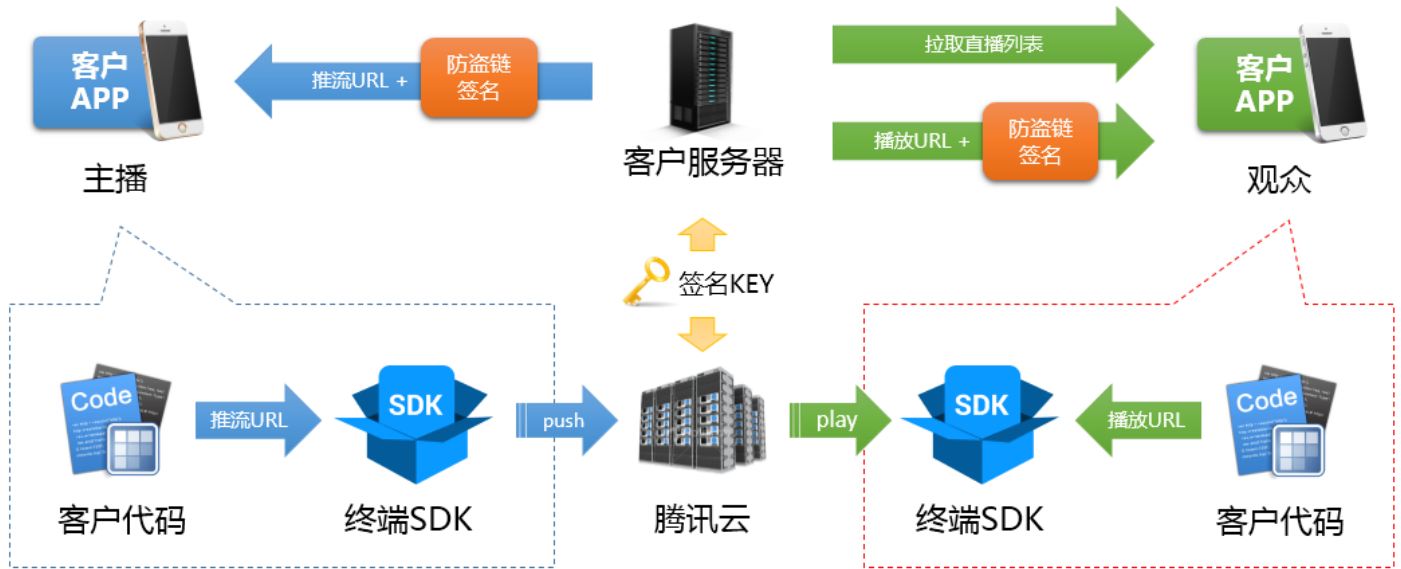
防盗链签名指的是推流 URL 和播放 URL 中的 **txTime** 和 **txSecret** 字段，它的作用是防止攻击者通过伪造您的直播 URL 而盗播您的流量。

防盗链签名

```
rtmp://8888.livepush.myqcloud.com/live/8888_test001?txSecret=xxx&txTime=5C2A3CFF
```

### 防盗链签名的原理？

为了不让攻击者可以伪造您的服务器生成推流URL，我们需要您现在直播管理控制台配置 **防盗链加密KEY**，由于攻击者无法轻易获得加密KEY，也就无法伪造出有效的推流URL，原理如下图所示：



## 防盗链签名的计算？

### step1：交换KEY

首先，您需要在 [直播控制台](#) 设置一个**加密密钥**，该密钥用于在您的服务器上计算防盗链签名，由于腾讯云跟您持有同样的密钥，所以您生成的防盗链签名，腾讯云可以进行合法性确认。

密钥分为**推流防盗链KEY**和**播放防盗链KEY**，前者用于生成推流防盗链URL，后者用于生成播放防盗链URL。目前在 [直播控制台](#) 上可以自助配置推流防盗链KEY，如下图：





### 播放防盗链不支持自助配置

由于播放防盗链KEY的配置需要同步到几千台CDN集群，同步周期一般都很长，不适合调试期频繁修改。如果您需要配置播放防盗链，可以通过客服电话联系我们，正常流程一般需要 1 - 3 天完成全 CDN 集群的配置。

### step2 : txTime

签名中明文部分为txTime，含义是该链接的有效期，比如现在我当前的时间是2016-07-29 11:13:45，而且期望新生成的URL是在 24 小时后即作废，那么txTime就可以设置为 2016-07-30 11:13:45。

为了减少 URL 中的字符串长度，要 2016-07-30 11:13:45 转换成Unix时间戳，也就是1469848425，然后再转换成16进制以进一步压缩字符串长度，也就是 txTime = 1469848425 (十进制) = 579C1B69 (十六进制)。

一般将 txTime 设置为当前时间 12 小时以后过期，**过期时间不要太短**，否则当主播在直播过程中遭遇断网重连时，会因为过期时间太短而无法恢复推流。

### step3 : txSecret

$$\text{txSecret} = \text{MD5}(\text{KEY} + \text{stream\_id} + \text{txTime})$$

这里的 **KEY** 就是您在 step1 中配置的加密KEY，**stream\_id** 为直播码（或称作流ID），**txTime**为刚才计算的579C1B69，**MD5** 即标准的MD5哈希算法。

### step4 : 地址拼接

现在有了推流（或者播放）URL，有了用来告知腾讯云该URL过期时间的txTime，有了只有腾讯云才能解密并且验证的txSecret，就可以拼合成一个防盗链的安全URL了。

```
rtmp://8888.livepush.myqcloud.com/live/8888_test001?txSecret=xxx&txTime=5C2A3CFF
```

[直播控制台](#) >> [直播码接入](#) >> [推流生成器](#) 页面下半部分有示例代码（PHP和Java两个版本）演示如何生成防盗链地址。

# 派发UserSig ( IM云通讯 )

最近更新时间：2018-07-23 16:39:58

## 为何要派发?

UserSig是使用腾讯云通讯服务 ( IM ) 所需的一种安全票据，所以如果您准备使用腾讯云 IM 服务实现聊天室功能，就需要您的后台工程师生成 UserSig 给返回给终端 APP；如果您已经有自己的 IM 解决方案（也就是已经有自己的聊天室了），可以免去这部分工作。

## 什么是UserSig?

腾讯云 IM 的前身是 QQ 的即时通讯消息系统，我们将 QQ 的消息模块进行了抽离，变成了适合移动端接入的 IM SDK。将消息后台进行改造，使其脱离对 QQ 号码的绑定，构成了现在的 IM 云通讯后台。

您可以把 IM SDK 理解为一个没有用户交互界面的 QQ，把 IM SDK 集成到您的 APP 里，就相当于把一个 QQ 的消息内核集成在您的 APP 内部。

我们都知道，QQ 可以用来收发单聊和群聊的消息，但前提是您必须先登录才能使用。我们也都知道，登录 QQ 用的是 QQ 号和密码，登录 IM SDK 也是一样，只是肯定不能再用 QQ 号和密码了，而是使用您指定的用户名 ( userid ) 和密码 ( usersig )。

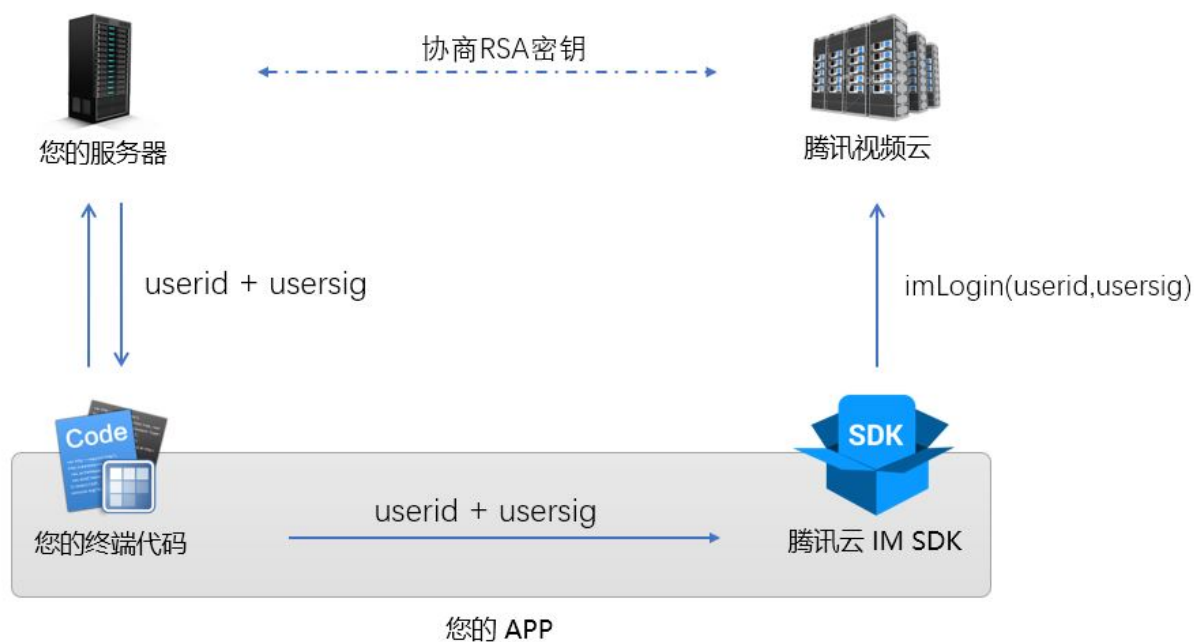
- **用户名 ( userid )**

可以是您目前 APP 里的用户 ID，比如您有一个用户，他/她的账号 ID 是 27149，那么您就可以用 27149 作为登录 IM SDK 的 userid。

- **密码 ( usersig )**

既然您指定了 27149 是您的用户，腾讯云如何才能确认该用户是您认可的合法用户呢？usersig 就是用于解决这个问题的，usersig 本质是对 userid、appid 等信息的非对称加密。

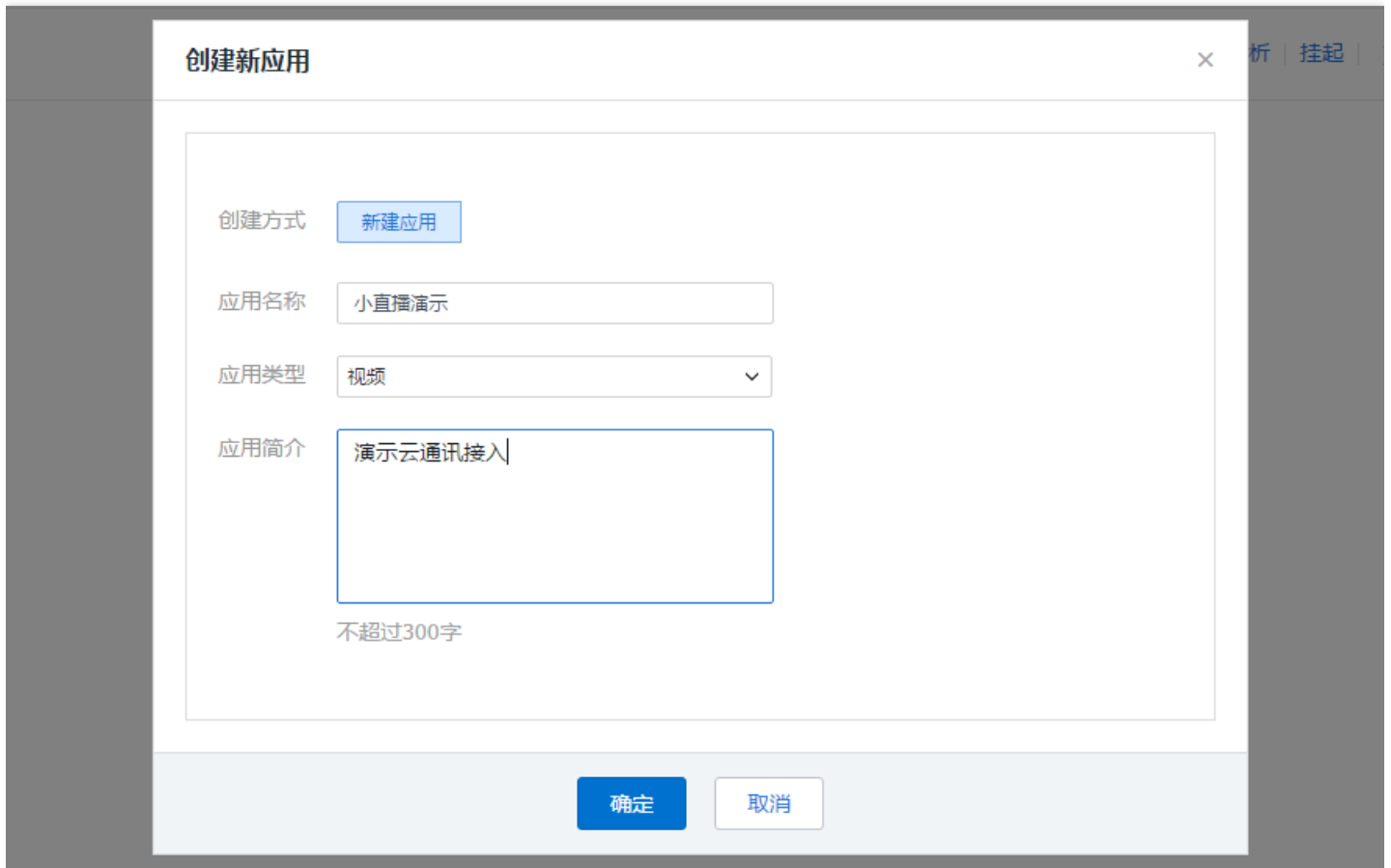
非对称加密用的加密密钥和解密密钥是不同的，您的服务器可以持有私钥并对 userid 和 appid 进行非对称加密，加密之后的结果就是 usersig；而腾讯云同步持有您的公钥，这样一来，腾讯云就可以确认 usersig 是否合法，从而可以确认它是否是由您的服务器签发的。



## 如何生成UserSig?

### step1 : 获取签发UserSig用的私钥

进入云通讯 [管理控制台](#)，如果还没有开通服务，直接点击 **开通云通讯** 按钮即可。新开通的账号下，应用列表是空的，点击 **创建应用接入** 按钮创建一个新的应用：



点击 **确定** 按钮，之后就可以在应用列表中看到刚刚添加的项目了，如下图所示：

### 应用列表

创建应用接入

查看录像

下载云通信SDK

SDKAPPID	应用名称	应用状态	创建时间	操作
1400069256	小直播演示	启用	2018-02-22 20:45:35	<a href="#">应用配置</a>   <a href="#">统计分析</a>
...	...	启用	2018-01-13 22:36:39	<a href="#">应用配置</a>   <a href="#">统计分析</a>
...	...	启用	2017-11-21 17:02:08	<a href="#">应用配置</a>   <a href="#">统计分析</a>

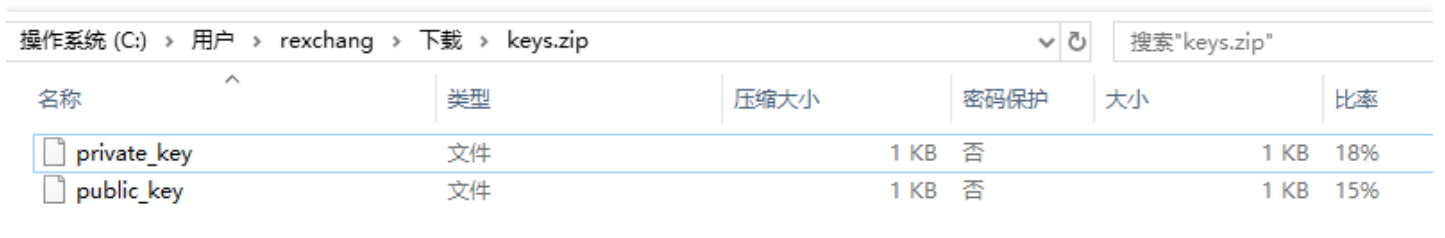
点击 **应用配置** 链接，会进入应用配置界面，再点击 **账号体系集成** 右侧的 **编辑** 按钮，按照下图所示进行配置即可（账号名称和管理员名称推荐用英文，账号名称随便填写，管理员名称在调用 IM 的 REST API 时可以用到）。

The screenshot displays the Tencent Cloud console interface. At the top, there is a navigation bar with '腾讯云' (Tencent Cloud) logo and '常用服务' (Common Services) menu. Below this, a secondary navigation bar lists various services, with '云通信' (Cloud Communication) selected. The main content area is divided into a left sidebar and a main panel. The sidebar contains '云通信' (Cloud Communication) and '应用列表' (Application List). The main panel shows details for an application named '小直播演示' (Small Live Broadcast Demo), including its ID '50KAppID-1400009230', application name, type '视频' (Video), creation time '2018-02-22 20:45:35', and last modification time '2018-02-22 20:45:35'. Below the details, there are three sections: '应用平台' (Application Platform) with an '编辑' (Edit) link, '所属平台' (Associated Platform), and '帐号体系集成' (Account System Integration) with an '编辑' (Edit) link. The '帐号体系集成' section is currently '未配置' (Not Configured). A modal window titled '帐号体系集成' (Account System Integration) is open, showing configuration options. It includes a text box for '帐号名称' (Account Name) with the value 'test' and a note '此项为必填项' (This is a required item). The '集成模式' (Integration Mode) section has two radio buttons: '独立模式' (Independent Mode), which is selected and circled in red, and '托管模式' (Managed Mode). There is a link '了解集成模式' (Learn about integration mode) next to it. The '账号管理员' (Account Administrator) field contains 'admin' and a link '什么是账号管理员' (What is an account administrator). Below this is a '+ 添加管理员111' (Add administrator 111) button. At the bottom of the modal are '保存' (Save) and '取消' (Cancel) buttons. A red arrow points from the '帐号体系集成' link in the sidebar to the modal window.

点击 **保存** 按钮，页面会自动刷新，之后就可以看到 **下载公私钥** 的按钮了。



点击 **下载公私钥** 按钮，会得到一个叫做 **keys.zip** 的压缩包，里面有一个 **private\_key** 和一个 **public\_key**，**private\_key** 就是用来签发 UserSig 的私钥了。



## step2 : 测试private\_key

可以在 **开发辅助工具** 中测试一下 private\_key 是否能正常进行签名。



### step3 : 集成生成代码

我们在官网的 SDK 下载区提供了一份计算UserSig 的简单版源码，有 java 和 php 两个版本的，如果您需要其他版本的，可以到 [DOC](#) 里寻找。

[-UserSig计算源码下载 \( java | PHP | nodejs \)](#)

---

## 如何使用 UserSig ?

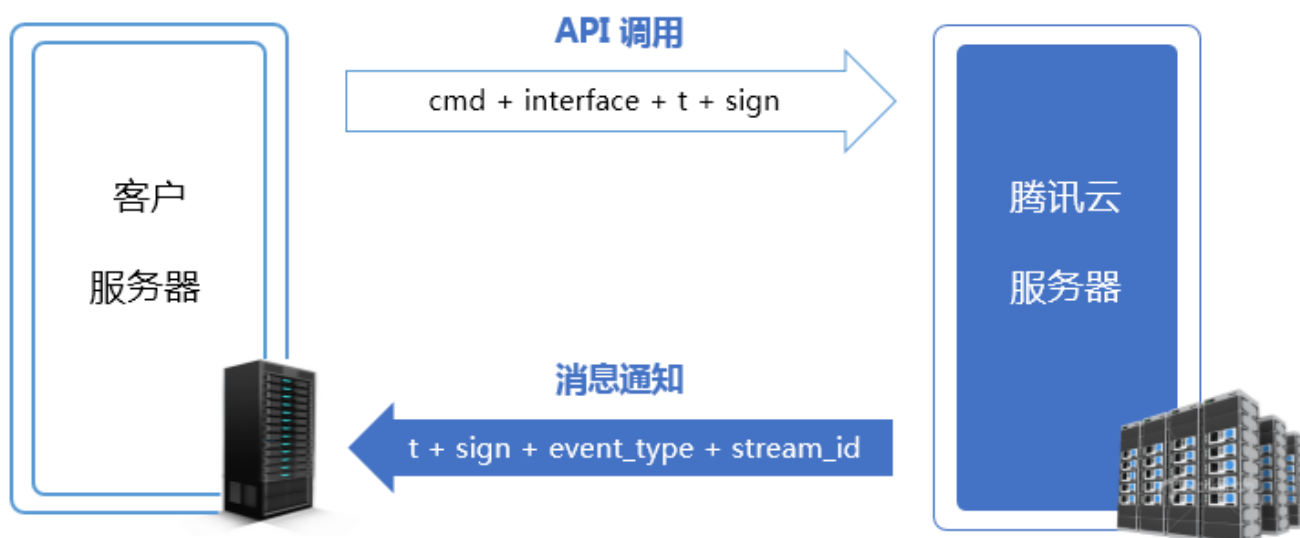
如果您是后台研发工程师，剩下的工作就不需要您操心了，您可以通知您的同事（终端开发工程师）阅读 IM SDK 的接入文档（[iOS](#) | [Android](#)），完成后续接入工作。



# 操控直播流 ( REST API )

最近更新时间：2018-07-23 16:40:46

## 与腾讯云后台通讯



您的服务器与腾讯云服务器的信息同步可以通过两种方式组合实现：

- **API 调用**：腾讯云提供了一组直播码管理API，包括状态查询和状态管理等功能，供您的后台服务器调用。
- **消息通知**：腾讯云在直播流状态变更、录制文件生成等一系列事件发生时，能够以事件消息（JSON）的形式主动通知您的后台服务器，只需要您在腾讯云注册接收事件通知的回调 URL 即可实现。

## API 调用

腾讯云提供了一组直播码管理API，包括状态查询和状态管理等功能，供您的后台服务器调用。

### 1. API 列表

API	功能介绍
<a href="#">Get_LiveStat</a>	统计信息查询 - 查询推流和播放相关信息
<a href="#">Get_LivePushStat</a>	统计信息查询 - 查询推流相关信息
<a href="#">Get_LivePlayStat</a>	统计信息查询 - 查询播放相关信息

API	功能介绍
<a href="#">Get_LivePushStatHistory</a>	获取推流历史信息
<a href="#">Get_LivePlayStatHistory</a>	获取播放统计历史信息
<a href="#">Live_Channel_GetStatus</a>	仅查询某条流的状态信息（旧版本接口）
<a href="#">Live_Channel_SetStatus</a>	对某条流实行 <b>禁播</b> 操作，主要用于鉴黄场景
<a href="#">Live_Tape_GetFilelist</a>	查询某条流在直播过程中的 <b>录制</b> 文件列表
<a href="#">Live_Queue_Get</a>	查询某条流在直播过程中的 <b>截图</b> 文件列表
<a href="#">Live_Channel_GetChannelList</a>	查询频道列表
<a href="#">Live_Channel_GetLiveChannelList</a>	查询直播中频道列表
<a href="#">mix_streamv2.start_mix_stream_advanced</a>	云端混流操作接口
<a href="#">channel_manager</a>	暂停并延迟恢复——可针对某路流禁止推流
<a href="#">Live_Tape_Start</a>	创建录制任务——可实现定时录制任务或者实时视频录制
<a href="#">Live_Tape_Stop</a>	结束录制任务

## 2. 调用方法

在您的 **服务端** 采用 HTTP 协议的 GET 请求方式（即调用参数直接拼接在 URL 中）进行调用即可，详细的调用方法在每个 API 的说明文档中都有示例参考，没有任何对接难度。

## 3. 安全机制

由于对 API 的调用采用的是普通的 HTTP 协议（出于性能考虑），这就需要一套行之有效的办法来确保您的服务器与腾讯云后台之间的通讯安全。

所有直播码相关的云端 API 都采用了同一种安全检查机制，**t + sign 校验**：

- **t (过期时间)**：如果一个API请求或者通知中的 t 值所规定的时间已经过期，则可以判定这个请求或者通知为无效的，这样做可以防止网络重放攻击。t 的格式为UNIX时间戳，即从1970年1月1日（UTC/GMT的午夜）开始所经过的秒数。
- **sign (安全签名)**： $sign = MD5(key + t)$ ，即把加密key 和 t 进行字符串拼接后，计算一下md5值。这里的key 即CGI调用key，您在腾讯云直播管理**控制台** 中可以进行设置：



• 安全原理

由于MD5是不可逆的HASH算法，所以只要确保KEY不泄露，即使攻击者拿到很多对t和sign也无法反算出KEY值，进而无法进行伪装攻击。

• 计算示例

比如我们现在的的时间是 2016-08-22 15:16:27，我们希望有效期是1分钟，也就是 2016-08-22 15:16:27 以后再收到携带这个 t 的请求或者通知即判定为非法的：

$$t = "2016-08-22 15:17:27" = 1471850187$$

假设我们的key是 5d41402abc4b2a76b9719d911017c592，那么我们计算的签名结果就是：

$$sign = MD5(5d41402abc4b2a76b9719d911017c5921471850187) = b17971b51ba0fe5916ddcd96692e9$$

## 错误码

### 1.http 错误

错误码	含义	备注
-----	----	----

错误码	含义	备注
403	Forbidden	接口为了安全考虑开启了校验，若使用浏览器验证发现该错误，可检查下cookie里是否含有skey
404	Not Found	查看请求时是否带上host

## 2.接口通用返回错误

错误码	含义	备注
appid is invalid	appid不合法，表示未开通该功能	

注：以上错误码针对本文1.API列表中的API。不包括[消息事件通知](#)

## 消息通知

详情参考腾讯云事件[消息通知](#)服务。