

腾讯云视频

实施指南

产品文档



腾讯云

## 【版权声明】

©2013-2017 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。

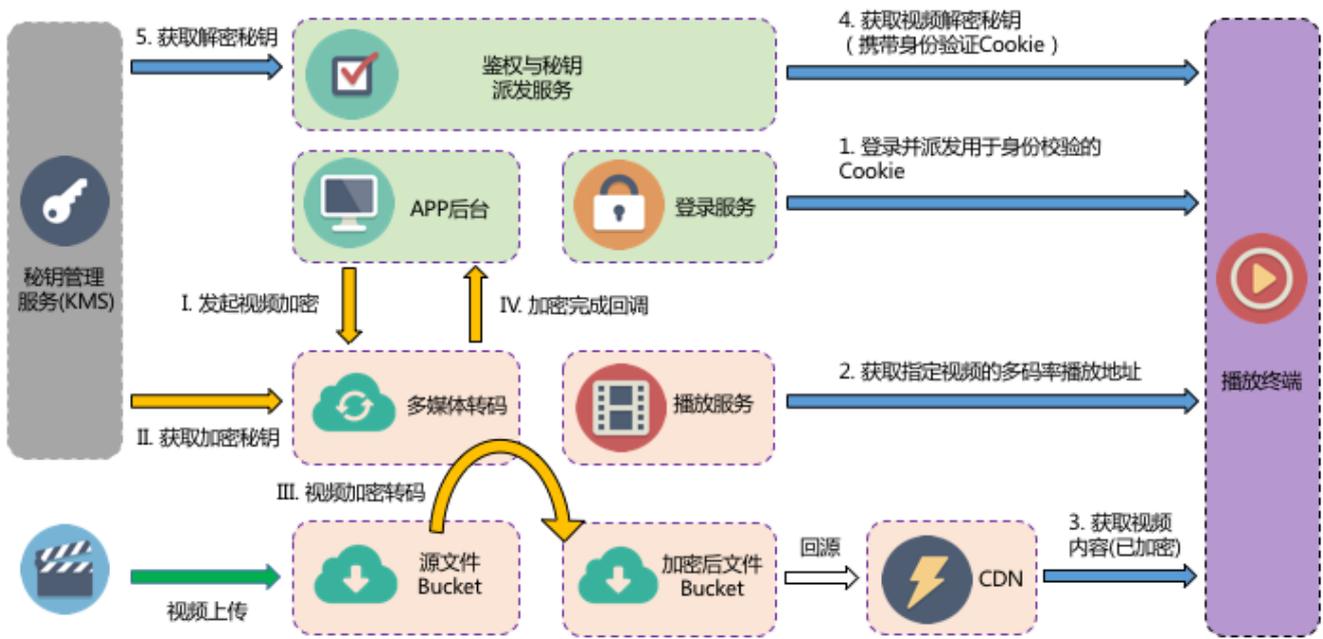
## 文档目录

|            |   |
|------------|---|
| 文档声明.....  | 2 |
| 实施指南.....  | 4 |
| 视频加密 ..... | 4 |
| 云通信集成..... | 7 |

实施指南

视频加密

整体架构



初步实施

建立密钥管理服务 ( KMS )

密钥管理服务主要用于管理视频密钥。视频加密过程需要与 KMS 系统进行交互，步骤如下：

1. 生成用于视频加密的数据密钥，即架构图中第 II 步。这一步将返回数据密钥 ( DK ) 和加密后的数据密钥 ( EDK )。在后续环节中，能够接触到 DK 的角色包括：点播转码服务、App 后台、经过合法身份校验的最终用户。EDK 可以分发给任意用户，但通过 EDK 获取 DK 这一步，必须由 App 后台进行身份校验。
2. 根据 EDK 获取 DK 来进行数据播放，即架构图中的第 4 步。APP 后台在通过用户的身份验证之后，需要调用 KMS 相关接口，使用 EDK 去获取 DK，即架构图中的第 5 步。

## 搭建鉴权与密钥派发服务

对于已经加密的视频，只有经过 App 后台认证过的客户端才能得到 DK。因此，最终客户获取密钥的行为必须要有 App 后台参与鉴权。该服务的主要业务逻辑是：

1. 对于客户端携带 EDK 换取 DK 的请求（即架构图中第 4 步），对请求方进行身份认证。
2. 如果身份认证通过，则去 KMS 系统获取对应的 DK（即架构图中第 5 步），并返回给客户端。

## 配置视频加密模板

为确保点播后台能够进行正确的加密操作，您需要配置视频加密模板。详情参见 [视频加密参数模板](#)。

## 完整实施

### 视频上传

可以通过服务端上传、客户端上传、控制台上传、录制上传、URL 转拉上传等方式来将已有视频文件上传到点播平台。

### 视频加密

视频加密主要分为以下四个步骤：

1. APP 后台发起视频加密

目前您可以通过 [ProcessFile](#) 接口发起视频加密，目前只支持对 HLS 文件进行加密。

2. 点播平台获取加密密钥

点播平台根据调用方指定的加密参数模板，读取密钥获取方式、最终用户获取解密密钥的 URL（假定为

`https://getkey.example.com`

），然后从指定 KMS 系统中获取视频加密密钥 DK、EDK。

3. 点播平台发起视频加密转码

点播转码平台在进行视频加密时，不仅会依照指定的加密算法和密钥对目标输出文件进行加密，而且会将获取解密密钥的 URL 写入视频文件中。

#### 4. 点播平台发起加密完成回调

包含加密操作的任务流状态发生变化（或者执行完毕）之后，点播平台将发起 [任务流状态变更通知](#)。

## 媒资管理

视频加密操作完成之后，可以通过 [GetVideoInfo](#) 接口获取视频的加密信息。

[GetVideoInfo](#) 接口会返回该视频所有转码规格的视频播放地址，包括源文件的播放地址，由于源文件是没有加密处理的，App

服务端可以过滤掉源文件的播放地址，只提供加密视频的播放地址给客户端。[GetVideoInfo](#) 获取到的源文件 `definition` 参数是 0，可以根据这个值来过滤源文件的视频播放地址。

## 视频播放

只有经过合法身份认证的用户才应当得到视频解密密钥。因此在播放过程中，如何对用户的身份信息校验就成为关键因素。

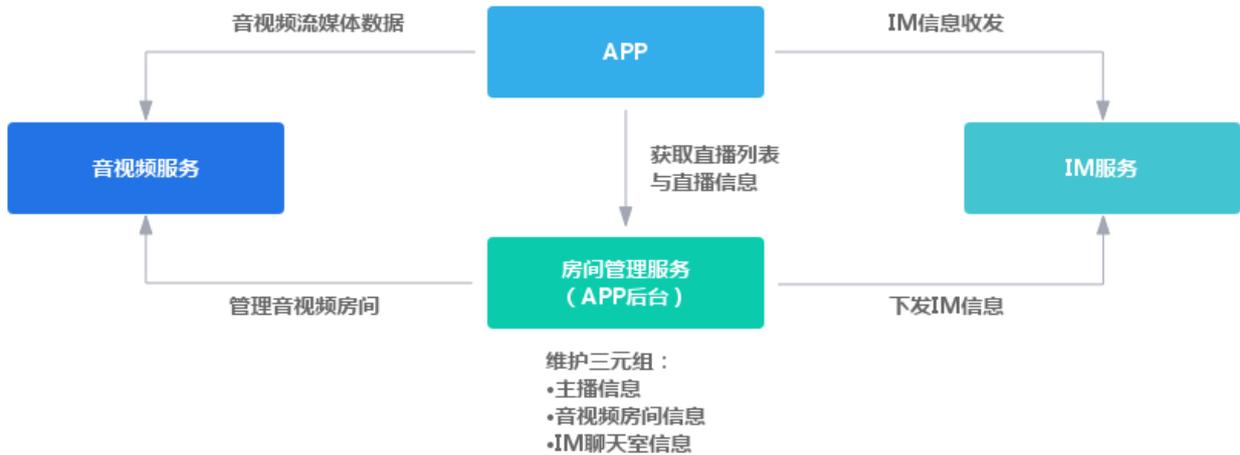
播放过程中，播放器会访问 m3u8 文件中 EXT-X-KEY 标签所标识的 URL

以获取密钥，播放器需要在这一步中携带观看者的身份认证信息。此时有两种方式可以将这一信息传递给 App 鉴权服务：

1. 将用户身份信息通过参数的方式追加到 URL 中，带给 App 的鉴权服务；该方案适用于所有的 HLS 播放器。具体方案参见 [视频播放方案1：通过 QueryString 传递身份认证信息](#)。
2. 将用户身份信息通过 Cookie 带给 App 的鉴权服务；该方案安全性更高，但其仅适用于在访问 EXT-X-KEY 标签所标识的 URL 时会携带 Cookie 的播放器。具体方案参见 [视频播放方案2：通过 Cookie 传递身份认证信息](#)。

## 云通信集成

### 整体架构



### 初步实施

#### 在真正开始集成 IMSDK

之前，您必须完成一些接入准备工作，包括：应用接入和用户体系集成。在完成这两步之后，您需要将 SDK 集成到您的 App 中，然后才能开始调用直播聊天室相关的 SDK 接口。

1. 应用接入 参见：[应用接入指引](#)。
2. 用户体系集成

云通信提供独立模式和托管模式两种用户体系集成。

- 如果您的 App 自主维护用户的注册、用户身份的验证，则应当使用 [独立模式](#)；
- 如果您只是想快速开发一个 App 原型，云通信可以为您提供一套符合业界通用安全标准的用户体系，用户的注册、用户身份的验证将全部由云通信提供，此时应当选用 [托管模式](#)。

3. 客户端集成

客户端集成是将 IMSDK 集成到您的 App 中，参见：[Android 客户端集成](#)、[iOS 客户端集成](#)、[Windows C++ 客户端集成](#)。

### 完整实施

## 创建直播聊天室

通过 IMSDK 创建直播聊天室(AVChatRoom), 参见：[Android 创建群组](#)、[iOS 创建群组](#)、[Windows C++ 创建群组](#)。

通过 REST API 创建直播聊天室，参见：[创建群组](#)。

## 申请加入聊天室

通过 IMSDK 申请加入聊天室，参见：[Android 申请加群](#)、[iOS 申请加群](#)、[Windows C++ 申请加群](#)。

注意：

用户加入 AVChatRoom 后，如果发生异常登出 (logout) 或者 App 进程崩溃 (crash) 的情况，在重新上线 (login) 或者重启 App 进程后，需要再次调用申请加群才能继续在原来的 AVChatRoom 中收发消息。

## 收发消息

通过 IMSDK 收发消息，参见：[Android 消息收发](#)、[iOS 消息收发](#)、[Windows C++ 消息收发](#)。

通过 REST API 发送消息，参见：[在群组中发送普通消息](#)。

## 特殊需求实施

### 聊天室成员进出通知

当聊天室中有用户进入或者离开时，云通信后台会下发成员进出通知。

新用户入群通知参见：[Android 用户入群通知](#)、[iOS 用户入群通知](#)、[Windows C++ 用户入群通知](#)。

用户退群通知参见：[Android 用户退群通知](#)、[iOS 用户退群通知](#)、[Windows C++ 用户退群通知](#)。

云通信后台会对用户进出聊天室的消息进行频率控制，具体参见[消息优先级与频率控制](#)。

### 获取观看直播的用户列表

直播 App 有时需要展示当前用户列表。单个 AVChatRoom

本身支持的用户数量无上限，但是只能够获取到部分群成员列表，最多为 300 个成员。

通过 IMSDK 获取群成员列表，参见：[Android 获取群成员列表](#)、[iOS 获取群成员列表](#)、[Windows C++ 获取群成员列表](#)。

## 获取观看直播的人数

AVChatRoom 的人数统计并不是实时的，有 1 分钟左右的时延。

IMSDK 获取当前群人数参见：[Android 获取群组资料](#)、[iOS 获取群组资料](#)、[Windows C++ 获取群组资料](#)。

通过 REST API 获取当前群人数参见：[获取群组资料](#)（回包中的 MemberNum 字段）。

注意：

- AVChatRoom 针对大规模直播场景，在人数统计上是以“有收发消息的动作”来衡量，统计结果存在时延且不完全准确。在人数较少的场景（比如只有几十人）下，统计的偏差会比较明显。
- 通过 H5 匿名接收 IM 消息的用户也会被计入聊天室人数中。但由于这些用户没有登录，不可能在用户列表中得到展示。

## 在聊天室中下发自定义消息

云通信的消息原生支持文本、图片、语音、表情等。如果 App

存在一些特殊的消息需求（例如点赞消息、红包消息等），可以通过云通信的自定义消息实现。

通过 IMSDK 发送自定义消息，参见：[Android 发送自定义消息](#)、[iOS 发送自定义消息](#)、[Windows C++ 发送自定义消息](#)。

通过 REST API 发送自定义消息，参见：[在群组中发送普通消息](#)。

注意：

- 如果需要使用云通信的脏字过滤功能，请注意自定义消息的结构设计，具体参考[敏感词（脏字）过滤](#)。

- 为方便调试，建议 App 使用 JSON 来组织自定义字段。不建议使用二进制数据。

## 消息优先级与频率控制

当 AVChatRoom 中的人数较多时，发消息的用户可能会非常多，导致一秒之内产生的消息量非常大。消息频率过高，会对云通信服务后台和 App 的渲染造成压力，也会加大客户端流量。此外，如果 App 界面的消息滚动速率过快，可能导致用户无法看清消息。

基于上述考虑，云通信会对单个 AVChatRoom 中的消息进行频率控制。具体可以参见：[消息优先级与频率控制](#)。

## 聊天室禁言与全局禁言

某些用户可能会在聊天室中发送广告或者其它不良信息，此时可以考虑将该用户禁言。禁言分为两个维度：

- 群组维度的禁言：用户在指定群内无法发言，即使退群再加群也不例外。群主和管理员具备此权限；
- 全局维度的禁言：用户在所有群组中都不能发言，同时也不能发送单聊消息。仅 App 管理员具备该权限。

通过 IMSDK 实现群内禁言，参见：[Android 对群成员进行禁言](#)、[iOS 对群成员进行禁言](#)、[Windows C++ 对群成员进行禁言](#)。

通过 REST API 实现群内禁言，参见：[批量禁言和取消禁言](#)。

通过 REST API 获取群内被禁言用户的列表，参见：[获取群组被禁言用户列表](#)。

通过 REST API 设置全局禁言，参见：[设置全局禁言](#)。

通过 REST API 查询全局禁言，参见：[查询全局禁言](#)。

## 敏感词（脏字）过滤

所谓脏字过滤，是指 App 可以在云通信后台配置一批敏感词，如果云通信后台在用户发送的消息中检测到这些敏感词，则拒绝下发该消息，并向发送者返回错误码 80001。

脏字的增删查改参见：[REST API：查询APP自定义脏字](#)、[添加APP自定义脏字](#)、[删除APP自定义脏字](#)。

注意：

- 云通信已经默认配置了政治类、色情类的脏字，能够满足这两方面的大多数过滤需求，App 只需要配置业务场景相关的脏字即可。例如，电商类 App 只需要配置“假货”、“刷单”等业务类脏字即可。
- 默认情况下，云通信后台只会对消息格式中的文本消息进行脏字过滤（TIMTextElem，参见[消息格式描述](#)），如果 App 使用自定义消息格式（TIMCustomElem，参见[消息格式描述](#)），云通信后台无法进行过滤。因此，App 最好在设计之初就确保将需要进行脏字过滤的文本放到 TIMTextElem 中。对于 App 已经使用了自定义消息格式，且已经上线，可以联系云通信客服，将消息格式提供给云通信的技术人员。云通信后台可以据此解析 App 的自定义消息中的文本信息，并进行脏字过滤。

## 其他消息控制策略

除了禁言与脏字过滤之外，云通信还提供了其他消息控制策略，例如群内发言之前回调。另外，如果您的控制策略较为特殊，可以联系云通信客服提出需求，云通信可以为您定制开发消息控制策略。具体参见：[第三方回调简介](#)、[第三方回调接入指引](#)、[群内发言之前回调](#)。

## 在消息中携带用户身份信息

群消息在下发给客户端时，除了会携带消息内容本身，还会携带消息发送者的信息，默认会携带如下字段：发送者的昵称、发送者的头像、发送者在群内的群名片。如果您的 App 在云通信中配置了[用户维度的自定义字段](#)、[群维度的自定义字段](#)（AVChatRoom 不支持群成员维度的自定义字段），亦可在消息中携带这些信息。默认下不会携带自定义字段，如有需要请根据[工单模板](#)提交工单进行处理。

注意：

如果要在消息中携带发送者昵称、头像，必须将这两个信息导入云通信的用户资料。

通过 App 设置设置用户资料，参见：[Android 用户资料](#)、[iOS 用户资料](#)、[Windows C++ 用户资料](#)、[Web 用户资料](#)。

通过 REST API 查询/设置用户资料，参见：[拉取用户资料](#)、[设置用户资料](#)。

除此之外，使用独立模式的 App 还可以在通过 REST API 导入账号的同时设置用户资料，参见：[独立模式账号同步](#)。

## 客户端压力测试

如果要对客户端在 AVChatRoom 下的消息收发性能进行测试，可参考 [压测工具](#)。

## 其他

### 全员消息推送

云通信针对不同的应用场景，提供了 App 消息推送服务和 在线成员广播大群 两种解决方案：

1. 直播场景下，高价值礼物送出时，App 后台有可能会需要向 全体在线用户 下发消息，俗称“大喇叭消息”。云通信为此场景专门提供了 [在线成员广播大群](#)，该方案也支持非登录在线用户接收广播消息。
2. 需要向 全体成员（包含不在线用户）推送 App 通知时，则需要选用 [消息推送服务](#)，除此之外，该方案还支持向全部用户中满足特定条件的用户推送消息，即按用户的标签、属性进行推送。

### 面向单个/一批用户的系统通知

如果 App 后台需要向单个/一批用户发送系统通知，可以参考如下两个接口：[发送消息](#)、[批量发单聊消息](#)。

### App 接收消息

IMSDK 接收消息都是通过统一回调接口 onNewMessage 返回，通过 onNewMessage 回调获得消息后，能根据消息获取相应的会话。具体参见：[Android 新消息通知](#)、[iOS 新消息通知](#)、[Windows 新消息通知](#)、[WebSDK 初始化新消息回调](#)。