

阿里云

视频直播

最佳实践

文档版本 : 20181213

法律声明

阿里云提醒您在阅读或使用本文档之前仔细阅读、充分理解本法律声明各条款的内容。如果您阅读或使用本文档，您的阅读或使用行为将被视为对本声明全部内容的认可。

1. 您应当通过阿里云网站或阿里云提供的其他授权通道下载、获取本文档，且仅能用于自身的合法合规的业务活动。本文档的内容视为阿里云的保密信息，您应当严格遵守保密义务；未经阿里云事先书面同意，您不得向任何第三方披露本手册内容或提供给任何第三方使用。
2. 未经阿里云事先书面许可，任何单位、公司或个人不得擅自摘抄、翻译、复制本文档内容的部分或全部，不得以任何方式或途径进行传播和宣传。
3. 由于产品版本升级、调整或其他原因，本文档内容有可能变更。阿里云保留在没有任何通知或者提示下对本文档的内容进行修改的权利，并在阿里云授权通道中不时发布更新后的用户文档。您应当实时关注用户文档的版本变更并通过阿里云授权渠道下载、获取最新版的用户文档。
4. 本文档仅作为用户使用阿里云产品及服务的参考性指引，阿里云以产品及服务的“现状”、“有缺陷”和“当前功能”的状态提供本文档。阿里云在现有技术的基础上尽最大努力提供相应的介绍及操作指引，但阿里云在此明确声明对本文档内容的准确性、完整性、适用性、可靠性等不作任何明示或暗示的保证。任何单位、公司或个人因为下载、使用或信赖本文档而发生任何差错或经济损失的，阿里云不承担任何法律责任。在任何情况下，阿里云均不对任何间接性、后果性、惩戒性、偶然性、特殊性或惩罚性的损害，包括用户使用或信赖本文档而遭受的利润损失，承担责任（即使阿里云已被告知该等损失的可能性）。
5. 阿里云网站上所有内容，包括但不限于著作、产品、图片、档案、资讯、资料、网站架构、网站画面的安排、网页设计，均由阿里云和/或其关联公司依法拥有其知识产权，包括但不限于商标权、专利权、著作权、商业秘密等。未经阿里云和/或其关联公司书面同意，任何人不得擅自使用、修改、复制、公开传播、改变、散布、发行或公开发表阿里云网站、产品程序或内容。此外，未经阿里云事先书面同意，任何人不得为了任何营销、广告、促销或其他目的使用、公布或复制阿里云的名称（包括但不限于单独为或以组合形式包含“阿里云”、“Aliyun”、“万网”等阿里云和/或其关联公司品牌，上述品牌的附属标志及图案或任何类似公司名称、商号、商标、产品或服务名称、域名、图案标示、标志、标识或通过特定描述使第三方能够识别阿里云和/或其关联公司）。
6. 如若发现本文档存在任何错误，请与阿里云取得直接联系。

通用约定

格式	说明	样例
	该类警示信息将导致系统重大变更甚至故障，或者导致人身伤害等结果。	禁止： 重置操作将丢失用户配置数据。
	该类警示信息可能导致系统重大变更甚至故障，或者导致人身伤害等结果。	警告： 重启操作将导致业务中断，恢复业务所需时间约10分钟。
	用于补充说明、最佳实践、窍门等，不是用户必须了解的内容。	说明： 您也可以通过按 Ctrl + A 选中全部文件。
>	多级菜单递进。	设置 > 网络 > 设置网络类型
粗体	表示按键、菜单、页面名称等UI元素。	单击 确定。
courier 字体	命令。	执行 <code>cd /d C:/windows</code> 命令，进入Windows系统文件夹。
斜体	表示参数、变量。	<code>bae log list --instanceid Instance_ID</code>
<code>[]</code> 或者 <code>[a b]</code>	表示可选项，至多选择一个。	<code>ipconfig [-all/-t]</code>
<code>{}</code> 或者 <code>{a b}</code>	表示必选项，至多选择一个。	<code>swich {stand / slave}</code>

目录

法律声明.....	1
通用约定.....	1
1 直播答题解决方案.....	1
1.1 方案介绍.....	1
2 纯音频与纯视频直播.....	8
3 Web端直播实践.....	9
3.1 如何推流与播放.....	9
3.2 直播转码.....	11
4 直播安全.....	13
4.1 鉴权代码示例.....	13
4.2 如何保护直播安全？.....	14
5 高频率智能审核方案.....	17

1 直播答题解决方案

1.1 方案介绍

直播问答模式是用户在指定时间内登陆直播间，在主持人引导下进行线上答题，答对12道题目即可冲顶奖金，瓜分每期设定的奖金。从2018年伊始，互联网圈就刮起了一阵“大佬狂撒币，网友喜答题”的热潮，那么如何在自己的业务中增加这样强互动的方式，促成用户活跃的提升呢？下面以阿里视频云提供的方案为例，描述如何搭建直播答题功能。

- 如果业务中还没有直播的服务，请 [注册阿里云账号](#) 并 [开通视频直播服务](#)。
- 目前我们提供的方式是在直播流中通过APP Server调用openAPI来插入SEI信号，使用阿里云答题播放器解析SEI信号。

实现方案

步骤一：流程说明

1. 主持人提出问题，此时准备推送题目。
2. 现场人员发出信息，通过接入方的App Server，调用阿里云的OpenAPI，在直播视频流当前位置中插入若干SEI帧，帧内容可由业务自定义。
3. 播放SDK接收到视频流后，解析出SEI帧，并回调给APP。此时APP立即向App Server请求问题信息，然后显示在APP上，完成整个出题过程。



说明：

在这个方案中，由于推流会产生固定的延时，而现场人员是通过云端服务插入SEI信号，有可能插入SEI的画面比真实插入时间早一点。因此现场人员在出题前需要做好校准，在插入SEI信号时加入一个固定延迟。

举例：现场主持人在12:00:00时提出问题，假设现场到阿里云的推流延时有1秒，那么现场工作人员应在12:00:01时调用添加SEI的OpenAPI（或者在12:00:00时调用，但是OpenAPI里面的delay参数设置为1000），这样正好保证SEI插入的是12:00:00主持人提出问题时的画面。

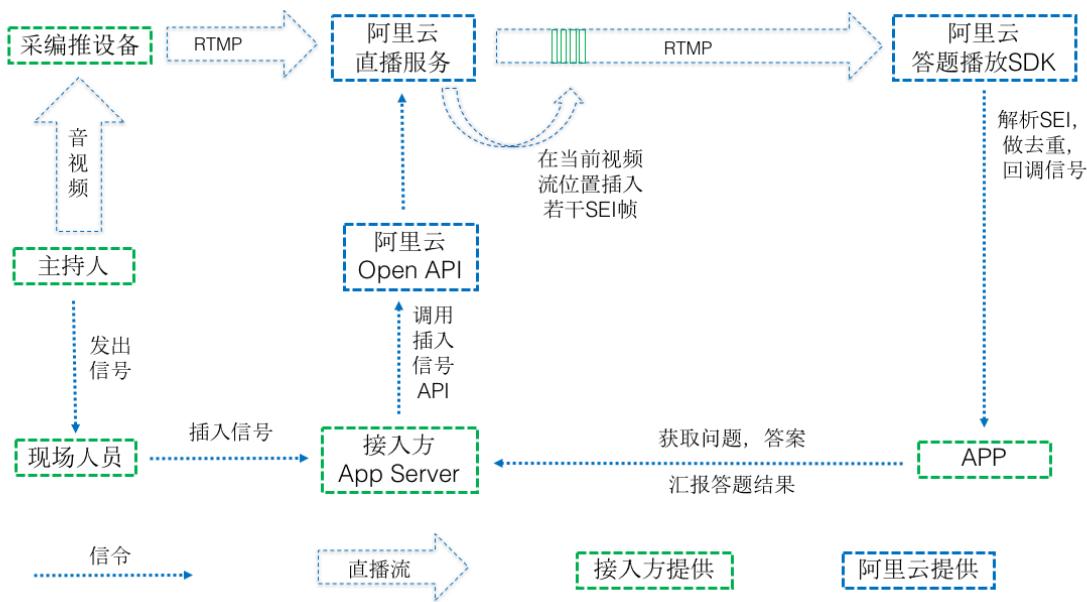


说明：

- 客户端在收到SEI信号后，需要向服务器获取问题，这会产生一定的通信耗时。因此实际答题系统中，可酌情延长答题时间，比如在原有答题时间上+1秒。

- 用户答题后，客户端需要上报答案，这会产生一定的通信耗时。因此实际答题系统中，AppServer在接收到答案时，答题超时时间允许有一个宽限值，比如原有的答题时限+1秒之内的均认为及时答题。

详见如下直播问答服务的架构图：



步骤二：配置与开通

- 首先需要开通阿里云视频直播服务。
- 提出工单，要求开通视频云添加SEI功能（提交域名和appname信息）。
- 在推流时，需要增加一个推流参数wm=1。

步骤三：服务端API调用

如前流程介绍，在接入方的appserver完成视频流中添加SEI信号时需要调用以下接口。



说明：

除了通过服务端api方式插入SEI信号完成题目下发，还可以通过我们提供的 [定制OBS工具](#)完成推题。

1. 添加SEI的OpenAPI

- 请求参数

参数	类型	是否必须	描述
Action	String	是	操作接口名，系统规定参数，取值： <code>AddTrancodeSEI</code>
DomainName	String	是	您的加速域名。
AppName	String	是	直播流所属应用名称。
StreamName	String	是	流名 (SEI 将会被添加到该流以及该流所产生的所有转码流上)。
Text	String	是	SEI 文本，长度限制：4000 字节。
Pattern	String	是	追加到每一个关键帧、每一帧。 取值： <code>keyframe</code> 、 <code>frame</code> 。
Repeat	Integer	是	重复次数，-1 代表无限。
Delay	Integer	是	接收到命令后的 delay 时间 (毫秒)。

- 返回参数

名称	类型	描述
RequestId	String	该条任务请求ID

- 特殊错误码

错误代码	描述	Http 状态码	语义
<code>InvalidDomain.NotFound</code>	The domain provided does not exist in our records.	404	当前账户下未查到域名
<code>IllegalOperation</code>	Illegal domain operate is not permitted.	403	不支持当前操作，如：非直播类域名

错误代码	描述	Http 状态码	语义
InvalidParams	Invalid parameters	400	参数不合法
InternalError	The request processing has failed due to some unknown error.	500	后台发生未知错误

2. 调用示例

```
https://live.aliyuncs.com?Action=AddTrancodeSEI&Domain=test101.cdnpe.com&App=xxx&Stream=xxx&Text=hahaha&Pattern=frame&Repeat=300&Delay=0&<公共请求参数>
```



说明：

在直播答题这种场景下，参数Pattern建议取值frame，Repeat建议为3倍的关键帧间隔（示例中为300）。这样设置可以保证在3个关键帧间隔内，即便发生丢帧，只要播放器收到1帧，即可取得SEI信息。由于SEI在多帧内重复，所以播放器需要做SEI去重的工作。

步骤四：接入答题播放器

首先需要集成阿里云播放器SDK，具体参考如下链接：播放器相关文档参见 [播放器SDK > 产品介绍](#)。

针对直播答题，播放器SDK提供了SEI数据的回调，分别来看Android和iOS对应的接口

1. Android SEI数据回调

通过播放器设置SEI数据回调接口，来获取到SEI的数据，从而进行业务的处理。方法如下：

- **setSEIDataListener**

```
public void setSEIDataListener(MediaPlayerSEIDataListener listener)
```

描述：给播放器设置SEI数据的回调，获取流中的SEI数据。

参数：回调监听。

- **MediaPlayer.MediaPlayerSEIDataListener**

```
public void onSEIdata(String data)
```

描述：SEI数据的回调监听

参数：data：SEI的数据。



说明：

同一次播放，如果SEI数据相同，则只回调一次，后续的相同数据不会回调。

使用示例

```
aliVcMediaPlayer = new AliVcMediaPlayer(DemoLiveAnswerActivity.this, videoSurface);
aliVcMediaPlayer.setSEIDataListener(new MediaPlayer.MediaPlayerSEIDataListener() {
    @Override
    public void onSEIData(String data) {
        praseData(data); //解析SEI data数据
    }
});
aliVcMediaPlayer.setMaxBufferDuration(3 * 1000);
aliVcMediaPlayer.prepareAndPlay(liveUrl);

private void praseData(String data) {
    //具体格式按照自己业务的需求去解析，然后做业务处理
    JSONObject jsonObject = null;
    try {
        jsonObject = new JSONObject(data);
    } catch (JSONException e) {
        e.printStackTrace();
    }

    SeiInfo seiInfo = SeiInfo.getFromJson(jsonObject);
    if (seiInfo == null) {
        Toast.makeText(getApplicationContext(), "SEI 数据错误", Toast.LENGTH_SHORT).show();
        return;
    }

    if (seiInfo.type.equals("startAnswer")) {
        //到服务器请求题目
        requestQuestion(seiInfo.questionId);
    } else if (seiInfo.type.equals("showResult")) {
        //到服务器请求答题汇总
        requestQuestionReport(seiInfo.questionId, seiInfo.showTime);
    }
}
```

以上方法均可在Demo中找到使用方法。

2. iOS SEI数据回调

通过监听播放器SEI数据通知来获取SEI数据，具体的通知如下：

AliVcMediaPlayerSeiDataNotification

使用示例

添加监听通知接口：

```
// 直播答题接收的消息
[[NSNotificationCenter defaultCenter] addObserver:self
                                         selector:@selector(
onSeiData:)
                                         name:AliVcMedia
PlayerSeiDataNotification object:self.mediaPlayer];
```

处理监听通知中SEI数据：

```
// 答题
- (void)onSeiData:(NSNotification *)notification{
    /*
    {
        "questionId": "001",
        "type": "startAnswer"
    } 这个是开始答题

    {
        "questionId": "001",
        "type": "showResult",
        "showTime": "5"
    } 这个是显示答题结果

    */
    NSDictionary* dict = [notification userInfo];
    if (dict) {
        NSString* seiData = [dict objectForKey:@"seiData"];
        if (seiData) {
            NSLog(@"%@", seiData);
        }
    }
}
```

可以在Demo中查询具体的使用方法。

注意事项

- 播放器延迟处理

需要控制每个端上的延迟是一致的，可以设置播放器最大的延迟参数来处理，一般的原则是延迟越大，越能够抗网络抖动，延迟越小，则卡顿越频繁。针对直播答题来说，需要考虑到整条链路上的整体延迟，所以这个延迟参数对题目的有效性时间会有影响。如果播放器延迟控制在3秒，答题时间是10秒，则题目的有效性时间需要加上播放器的延迟时间则是13秒。

ios延迟控制参数，单位毫秒 `@property(nonatomic, readwrite) int dropBufferDuration;`

android延迟控制接口：单位毫秒 `public void setMaxBufferDuration(int duration);`

- 下发题目延迟处理

客户端在收到SEI信号后，需要向服务器获取问题，这会产生一定的通信耗时。因此在下发题目时，可酌情加大`expiredSeconds`，比如在原有答题失效时间上+1秒。

2 纯音频与纯视频直播

阿里云支持纯音频与纯视频直播，您可以推送纯音频或纯视频的直播流到阿里视频云中心，也可以直接播放纯音频或纯视频的直播流。

- **推流**

支持推流RTMP协议的音视频流。

- **播放**

— 您可以在地址后端增加参数，来单独播放纯音频或纯视频流。

■ 单独播放音频：`onlyaudio=1`

■ 单独播放视频：`onlyvideo=1`

— **示例**

■ `rtmp://域名/APPName/Streamname?onlyaudio=1`

■ `FLV://域名/APPName/Streamname.flv?onlyaudio=1`



说明：

仅支持FLV与RTMP格式的播流地址。

- **使用场景**

— **推流音视频**

`rtmp://推流域名/APPName/Streamname?`

— **输入纯音频、视频地址，播出纯音频或纯视频内容**

■ 播放纯音频：`rtmp://播流域名/APPName/Streamname?onlyaudio=1`

■ 播放纯视频：`FLV://播流域名/APPName/Streamname.flv?onlyvideo=1`

3 Web端直播实践

3.1 如何推流与播放

场景

- 我有一台 PC，上面有摄像头，怎么进行一场直播？

准备工作

- 开通阿里云直播服务：阿里云直播服务接收直播流，并提供直播播放地址，完成一场直播需要先开通阿里云直播服务，在[视频直播产品主页](#)中单击立即开通，开通阿里云直播服务。
- 添加直播加速域名：您需要做直播服务需要提供一个经过备案的域名，并且在视频直播控制台中[添加直播加速域名](#)。

系统架构



开始直播

直播需要如下步骤：

- 获取推流地址：假如我们的直播加速域名为 a.com，那么推流地址为：`rtmp://video-center.alivecdn.com/AppName/StreamName?vhost=a.com`。
- 推流：可以使用阿里云提供的直播 demo 工具进行推流测试：https://promotion.aliyun.com/ntms/act/streamplay.html?spm=5176.137251.709409.4.MiBmaJ&wh_ttid=pc，将第一步中的推流地址填入推流工具中的推流地址处，并单击开始直播即可。
- 播放：对于正在进行的推流，可以使用如下几个地址进行播放，将任一地址填入直播 demo 工具的拉流地址，单击开始点播即可。



说明：

需要把下面的 a.com 换成自己的直播域名。

- `http://a.com/AppName/StreamName.flv`
- `rtmp://a.com/AppName/StreamName`
- `http://a.com/AppName/StreamName.m3u8`

常见问题

有没有其他工具可以进行推流？

示例中使用的是 Flash 进行推流，还可以使用 OBS 进行推流，[OBS 下载及使用教程](#)。

如果我需要同时进行多场直播怎么办？

示例中使用的是 Flash 进行推流

阿里云直播设计逻辑为：完成配置后，可以自行管理推流地址，无需创建直播地址，相应的播放地址通过规则来映射。假如用户的直播加速域名为 a.com，那么推流地址为：rtmp://video-center.alivecdn.com/AppName/StreamName?vhost=a.com。其中 AppName 和 StreamName 用户可以根据自己需求任意设置和管理。

例如，如下的直播推流地址都是可以使用的（相应的播放地址列在推流地址后面）。

- 推流地址：rtmp://video-center.alivecdn.com/live/abc?vhost=a.com
 - rtmp 播放地址：rtmp://a.com/live/abc
 - httpflv 播放地址：http://a.com/live/abc.flv
 - hls 播放地址：http://a.com/live/abc.m3u8
- 推流地址：rtmp://video-center.alivecdn.com/app/s1?host=a.com
 - rtmp 播放地址：rtmp://a.com/app/s1
 - httpflv 播放地址：http://a.com/app/s1.flv
 - hls 播放地址：http://a.com/app/s1.m3u8

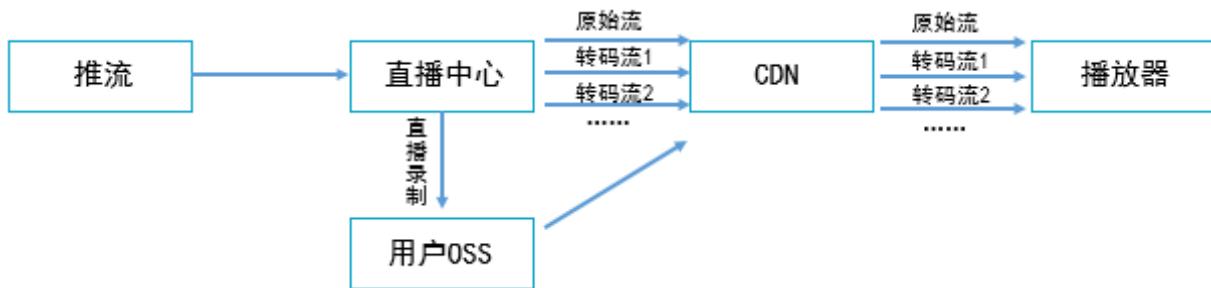
上面提到的三种地址，有什么区别？

- 三种播放地址的延迟不一样：httpflv 与 rtmp 延迟较小，适合交互比较强的直播，hls 延迟较大，适合交互比较弱的直播。
- 三种播放地址的兼容性不一样：httpflv 和 rtmp 需要 flash 播放器或则客户端播放器，hls 可以直接在浏览器中播放。

3.2 直播转码

直播常见一个问题是，推流的码率设置得比较高、很清晰，但是网络差的观众无法观看，我们可以使用直播转码功能，将推上来的直播流转成多种码率，满足不同网络的用户观看需求。

架构图



配置规则

阿里云直播转码的设计逻辑与录制类似，对域名进行配置，推流地址符合配置规则直播流将被转码。

转码模版

每个转码模版会有一个 **ID**，内容为一组转码参数的集合，制定了转码输出流的码率、分辨率等参数。

阿里云提供 2 组转码模版：高质量转码模版和低质量转码模版，每组转码模版分别有四种分辨率、码率的组合，这些模版分别为：

- 高质量模版：**ld**、**sd**、**hd**、**ud**
- 低质量模版：**lld**、**lsd**、**lhd**、**lud**

转码配置规则

用户的推流地址形式如下：

```
rtmp://video-center.alivecdn.com/AppName/StreamName?vhost=a.com
```

这个推流地址中，有三部分是由用户管理的：域名 (a.com)、AppName、StreamName。用户可以配置对某个 **域名 +AppName** 使用某几个转码模版，配置后，当用户推流时，符合该域名与AppName 的直播流将被转码。

示例

配置：域名为 a.com, AppName 为 live 进行转码，如下直播流将被转码。

rtmp://video-center.alivecdn.com/live/StreamName?vhost=a.com

rtmp://video-center.alivecdn.com/live/xyz?vhost=a.com

用户可以在控制台上进行 [通用转码配置](#)、[自定义转码配置](#)。

转码流播放地址：转码流播放地址由原始流播放地址+转码 ID 构成。例如，在域名 a.com，AppName 为 live 的直播域名下配置了 **Id**、**sd** 的转码，则相应的推流、播放地址分别为：

- 推流地址：rtmp://video-center.alivecdn.com/live/abc?vhost=a.com
- 原始流播放地址：
 - rtmp : rtmp://a.com/live/abc
 - httpflv : http://a.com/live/abc.flv
 - hls : http://a.com/live/abc.m3u8
- 转码流 **Id** 播放地址：
 - rtmp : rtmp://a.com/live/abc_ld
 - httpflv : http://a.com/live/abc_ld.flv
 - hls : http://a.com/live/abc_ld.m3u8
- 转码流 **sd** 播放地址：
 - rtmp : rtmp://a.com/live/abc_sd
 - httpflv : http://a.com/live/abc_sd.flv
 - hls : http://a.com/live/abc_sd.m3u8

参见 [推流地址和播流地址#原画#](#)、[推流地址和播流地址#转码#](#)。

4 直播安全

4.1 鉴权代码示例

概述

URL鉴权规则参见 [直播鉴权](#)，通过这个 demo 您可以根据业务需要，方便的对URL进行鉴权处理。

以下Python Demo包含直播所用的A种鉴权方式。

Python版本

```
import re
import time
import hashlib
import datetime
def md5sum(src):
    m = hashlib.md5()
    m.update(src)
    return m.hexdigest()
def a_auth(uri, key, exp):
    p = re.compile("^(rtmp://)?([^.?]+)(/[^.?]*)(\\?.*)?$")
    if not p:
        return None
    m = p.match(uri)
    scheme, host, path, args = m.groups()
    if not scheme: scheme = "rtmp://"
    if not path: path = "/"
    if not args: args = ""
    rand = "0"          # "0" by default, other value is ok
    uid = "0"          # "0" by default, other value is ok
    sstring = "%s-%s-%s-%s" %(path, exp, rand, uid, key)
    hashvalue = md5sum(sstring)
    auth_key = "%s-%s-%s" %(exp, rand, uid, hashvalue)
    if args:
        return "%s%s%s&auth_key=%s" %(scheme, host, path, args,
        auth_key)
    else:
        return "%s%s%s?auth_key=%s" %(scheme, host, path, args,
        auth_key)
def main():
    uri = "rtmp://video-center.alivecdn.com/test/test?vhost=xc.cdnpe.
    com"          # original uri
    key = "<input private key>"          # private key
    of      authorization
    exp = int(time.time()) + 1 * 3600          # expiration
    time: 1 hour after current time
    authuri = a_auth(uri, key, exp)          # auth type:
    print("URL : %s\nAUTH: %s" %(uri, authuri))
if __name__ == "__main__":
```

```
main()
```

4.2 如何保护直播安全？

准备工作

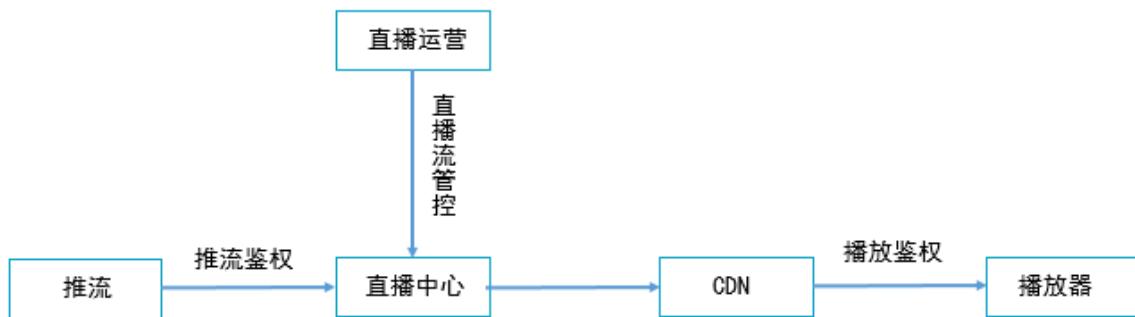
- 开通阿里云直播服务

阿里云直播服务接收直播流，并提供直播播放地址，完成一场直播需要先开通阿里云直播服务，在[视频直播官网详情页](#)中单击立即开通，开通阿里云直播服务。

- 如您已经能够使用阿里云进行直播，但还没有进行过直播，参见[如何使用PC推流](#)。

系统架构

由下图可知，我们可以通过三个方面控制直播的安全。



- 对推流进行鉴权
- 对直播流的管控
- 对播放进行鉴权

直播鉴权

推流鉴权与播放鉴权的机制是一样的，都是属于直播鉴权。

鉴权原理

用户对加速域名设置一个密钥，并根据将访问阿里云的 URI、时间戳、密钥构成的字符串进行 hash，将 hash 值以及时间戳作为请求参数的一部分，当如下两项通过时，允许访问该 URL。

- 时间戳没有超过当前时间。
- 服务端用同样的 hash 算法计算 hash 值，请求参数中的 hash 值一致。

鉴权算法

未鉴权URL：<http://DomainName/AppName/StreamName.flv>。

- 鉴权后的URL：`http://DomainName/Filename?auth_key=timestamp-rand-0-md5hash`。

参数说明

- `timestamp`：失效时间，整形正数，固定长度 10，1970 年 1 月 1 日以来的秒数。用来控制失效时间。
- `rand`：随机数，一般设成 0。
- `md5hash`：通过md5算法计算出的验证串，算法（假设用户的密钥为 123456：`md5hash=md5(timestamp-rand-0-123456)`）。

鉴权示例

用户给 a.com 这个域名配置了一个鉴权密钥：xyz，希望鉴权有效时间为：2017-07-28 05：43：20，准备用`rtmp://video-center.alivecdn.com/live/abc?vhost=a.com`进行推流，相应的鉴权计算逻辑为：

推流

- 获取推流uri：`/live/abc`。
- 获取 `timestamp`：过期时间 2017-07-28 05：43：200 的unixtimestamp为：1501191800。
- 组装加密字符串：`/live/abc-1501191800-0-0-xyz`。
- 计算加密后的字符串：`md5("/live/abc-1501191800-0-0-xyz") = 7856896499b661b1123dca82d7406aa3`。
- 拼接加密后的推流URL：`rtmp://video-center.alivecdn.com/live/abc?vhost=a.com?auth_key=1501191800-0-0-7856896499b661b1123dca82d7406aa3`。

播放

播放的计算方法与推流类似，以 `httpflv` 播放地址为例。

- 未鉴权的 URL 为：`http://a.com/live/abc.flv`。
- 获取播放URL：`/live/abc.flv`。
- 获取 `timestamp`：过期时间 2017-07-28 05：43：200 的 unixtimestamp 为：1501191800。
- 组装加密字符串：`/live/abc.flv-1501191800-0-0-xyz`。
- 计算加密后的字符串：`md5("/live/abc.flv-1501191800-0-0-xyz") = b022238fd0cd1c8efae2ba84fc0c9119`。

- 拼接加密后的推流URL：`http://a.com/live/abc.flv?auth_key=1501191800-0-0-b022238fd0cd1c8efae2ba84fc0c9119`。
- 设置方法：可以在直播控制台进行设置，控制台中鉴权计算器，可以方便的生成鉴权URL。详情参考[直播鉴权](#)。

直播流管控

直播流管控针对直播推流，在直播过程中，发现直播流内容不合法，除了能够通过终止推流停止直播以外，还能够通过直播流管控手段，对直播流进行封禁。

登录[视频直播控制台](#)，在流管理列表中可以看到加入黑名单选项。将视频流设置黑名单后，此条流执行禁推功能，将不会再接收推流的状态。



推流地址	App-Name	画质	推流时间	操作
rtmp://video-center.alivecdn.com/AppName/StreamName?host=live.aliyuntest.club	AppName	· 原画(Original) · 流畅(LLD)	起: 2018-01-23 19:14:03 止: 2018-01-23 19:14:14	历史监控 加入黑名单 直播地址

5 高频率智能审核方案

近些年来，视频直播快速发展，国家对于直播内容监管的要求日益严格，如何在满足业务需求的情况下，选择成本最优的智能审核方案，是每个直播平台面对的难题。本文介绍如何以最优成本配置直播审核模板。

功能介绍

- 审核场景

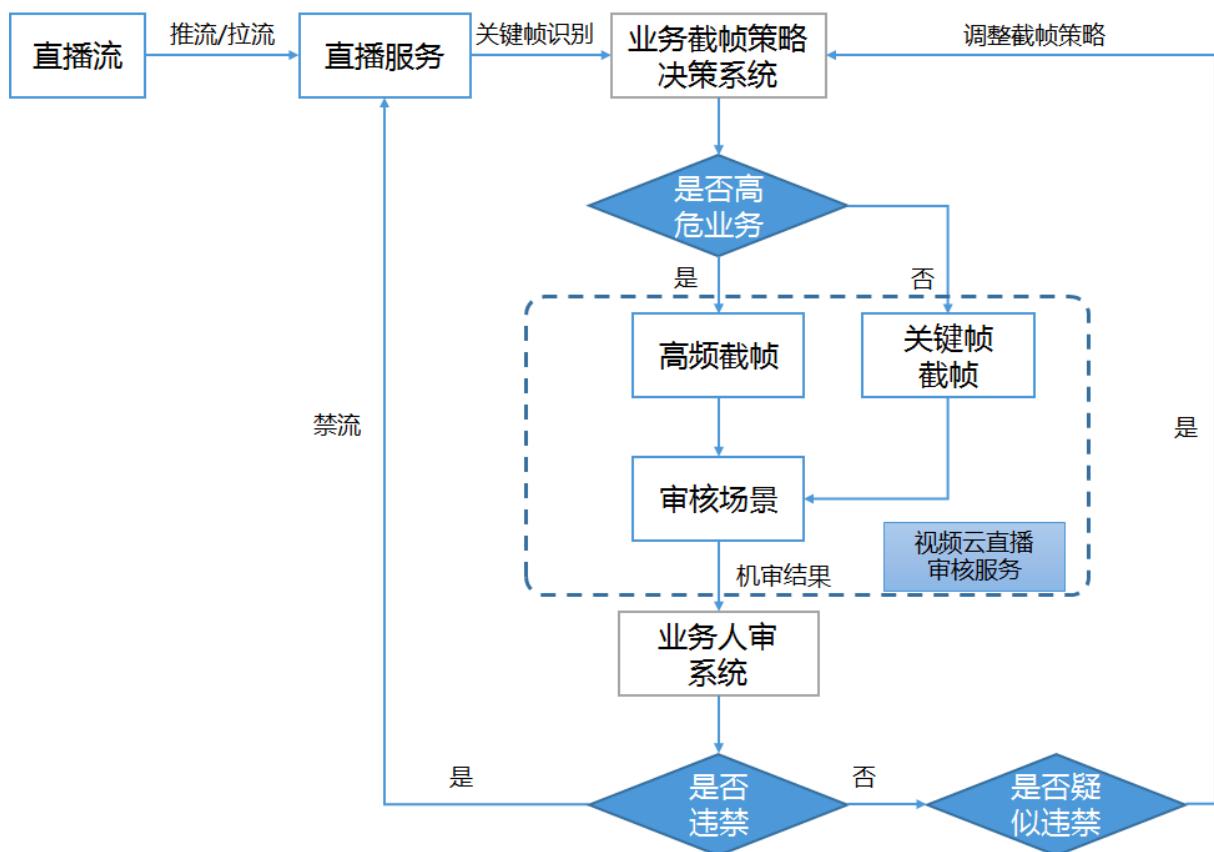
审核场景包含对直播画面的涉黄、暴恐涉政、广告、不良场景四大维度进行审核。

场景	描述	Scene	Label
智能监黄	识别直播中的色情内容。	porn	<ul style="list-style-type: none">normal：正常图片，无色情内容sexy：性感图片porn：色情图片
暴恐涉政识别	识别直播中的暴恐涉政内容。	terrorism	<ul style="list-style-type: none">normal：正常图片bloody：血腥explosion：爆炸烟光outfit：特殊装束logo：特殊标识weapon：武器politics：涉政violence：打斗crowd：聚众parade：游行carcrash：车祸现场others：其他
广告识别	识别直播中的广告信息。	ad	<ul style="list-style-type: none">normal：正常图片，无广告ad：含广告图片
不良场景识别	识别直播中出现的黑屏、黑边、昏暗画面，画中画，抽	live	<ul style="list-style-type: none">normal：正常图片

场景	描述	Scene	Label
	烟，打架等不良场景图片。		<ul style="list-style-type: none"> • meaningless : 无意义图片 • PIP : 画中画 • smoking : 吸烟 • drivelive : 车内直播

- 截帧频率
 - 高频截帧：对于内容风险高的业务如核心主播或交友等类型，采用高频截帧方案，截帧频率为1-2秒，大幅度规避危险内容曝光的可能性。
 - 关键帧截帧：对于业务风险相对不高的业务如游戏直播或赛事直播，采用关键帧截帧方案，截帧频率为3s，可做到优化成本。

方案介绍



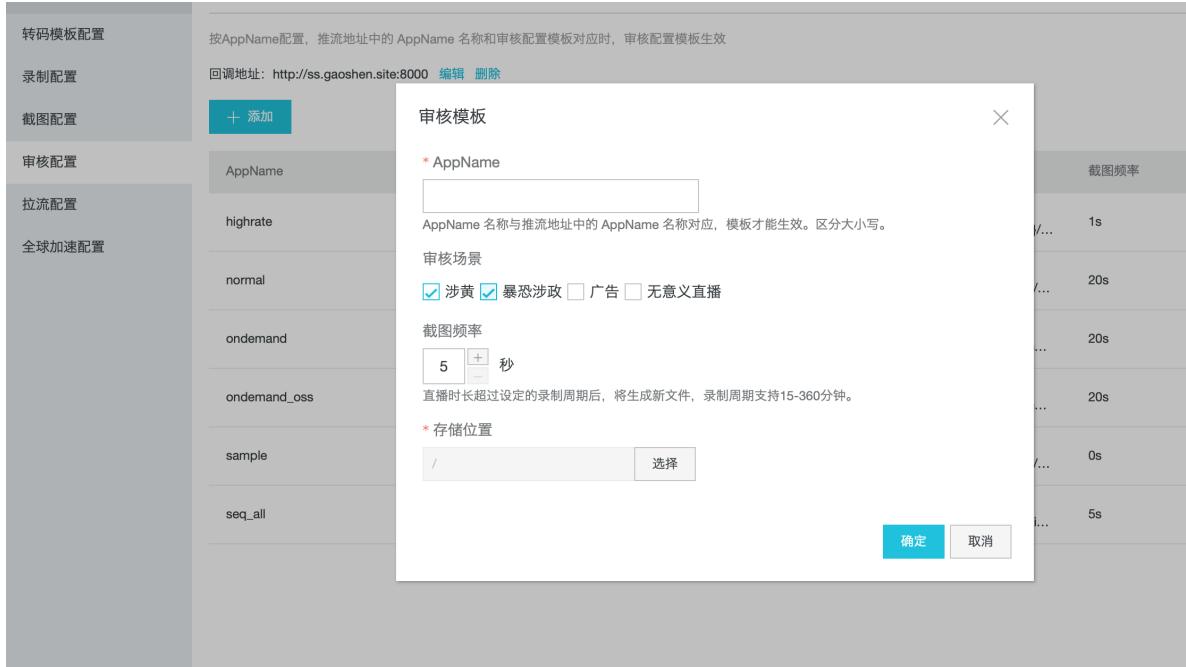
接入步骤



说明：

接入前，您需要开通直播服务，并完成默认的OSS授权，问题图片会依赖此授权保存到您的OSS中。开通直播服务，参见 [购买流程](#)。

1. 您需要准备一个HTTP server作为接收回调的服务。
2. 根据您的需求通过 [视频直播控制台](#) 配置审核策略。



您还可以通过OpenAPI配置审核策略。



3. 配置回调地址，并单击 确定。



4. 内容审核。

- 推流后，立即开始按照规则进行直播内容审核。
- 当发现异常情况，会通过回调通知给使用方，并把有问题的图片保存到用户指定的OSS下。

格式如下：

```
{  
    "DomainName": "qt01.alivecdn.com",  
    "AppName": "ondemand_oss",  
    "StreamName": "gs0911_4",  
    "OssEndpoint": "oss-cn-hangzhou.aliyuncs.com",  
    "OssBucket": "mts-bucket-test",  
    "OssObject": "customer/ondemand_oss/gs0911_4/1505131029218.jpg"  
},  
    "Result": [  
        {  
            "BizType": "default",  
            "Result": [  
                { "Label": "Porn", "Rate":100, "Suggestion": "review", "Scene": "porn", "Extent": {}},  
                { "Label": "Ad", "Rate":100, "Suggestion": "review", "Scene": "ad", "Extent": {}},  
            ]  
        }  
    ]  
}
```



说明：

- **DomainName:** 域名
- **AppName:** app
- **StreamName:** 流名
- **OssEndpoint:** Oss endpoint地址
- **OssBucket:** Oss Bucket名字
- **OssObject:** 存储对象名字
- **Result:** 检测结果
 - **Label:** 对应检测结果标签
 - **Rate:** 置信度
 - **Suggestion:** 建议处理方式
 - **Scense:** 检测场景
 - **Extent:** 扩展字段