

频道直播录制打点功能

接口描述

- 1、设置直播录制打点

2、接口支持https协议

接口URL

https://api.polyv.net/live/v3/channel/record/add-breakpoint

在线API调用

请求方式

POST

接口约束

1、接口同时支持HTTP 、HTTPS ， 建议使用HTTPS 确保接口安全， 接口调用有频率限制， [详细请查看](#)

请求参数描述

参数名	必选	类型	说明
appId	true	String	账号appId【详见 获取密钥 】
timestamp	true	Long	当前13位毫秒级时间戳， 3分钟内有效

参数名	必选	类型	说明
sign	true	String	签名，为32位大写的MD5值，生成签名的appSecret密钥作为通信数据安全的关键信息，严禁保存在客户端直接使用，所有API都必须通过客户自己服务器中转调用POLYV服务器获取响应数据【详见 签名生成规则 】
channelId	true	String	频道号
type	true	String	打点类型，pause 暂停录制，resume继续录制
fromStartStop	false	String	是否从一开始就暂停录制，Y表示是，其他不是，只有在type为pause时生效。

示例

```
https://api.polyv.net/live/v3/channel/record/add-breakpoint
```

表单参数：

```
appId=frlr1zazn3&origin=playback&sign=01556AABE64ED5C285FF5378392FCD8C&channelId=10003&type=pause
```

响应参数描述

参数名	类型	说明
code	Integer	响应状态码，200为成功返回，非200为失败【详见 全局错误说明 】
status	String	响应状态文本信息
message	String	响应描述信息，当code为400或者500的时候，辅助描述错误原因

参数名	类型	说明
data	Boolean	请求成功返回true，请求失败返回空

Java请求示例

快速接入基础代码请[下载相关依赖源码](#)，[点击下载源代码](#)，下载后加入到自己的源码工程中即可。测试用例中的[HttpUtil.java](#) 和 [LiveSignUtil.java](#) 都包含在下载文件中。

强烈建议您使用[直播Java SDK](#)完成API的功能对接，直播Java SDK 对API调用逻辑、异常处理、数据签名、HTTP请求线程池进行了统一封装和优化。

```
private static final Logger log = LoggerFactory.getLogger(ChannelRecordTest.class);
/**
 * 频道直播录制打点
 * @throws IOException
 * @throws NoSuchAlgorithmException
 */
@Test
public void testAddBreakPoint() throws IOException, NoSuchAlgorithmException {
    //公共参数,填写自己的实际参数
    String appId = super.appId;
    String appSecret = super.appSecret;
    String userId = super.userId;
    String timestamp = String.valueOf(System.currentTimeMillis());
    //业务参数
    String url = "https://api.polyv.net/live/v3/channel/record/add-breakpoint";
    String channelId = "2139283";
    String type = "pause";

    Map<String, String> requestMap = new HashMap<>();
    requestMap.put("appId", appId);
    requestMap.put("timestamp", timestamp);
    requestMap.put("channelId", channelId);
    requestMap.put("type", type);
    requestMap.put("sign", LiveSignUtil.getSign(requestMap, appSecret));
    String response = HttpUtil.postFormBody(url, requestMap);
    log.info("测试修直播录制打点, 返回值: {}", response);
    //do somethings
}
```

响应示例

系统全局错误说明详见[全局错误说明](#)

成功示例

```
{
  "code": 200,
  "status": "success",
  "message": "",
  "data": true
}
```

异常示例

```
{
  "code": 400,
  "status": "error",
  "message": "invalid signature.",
  "data": ""
}
```

```
{
  "code": 400,
  "status": "error",
  "message": "channel in no live", // 频道在直播中才能设置
  "data": ""
}
```