

查询频道内容保护（防录屏信息）

接口描述

- 1、通过频道号，查询频道内容保护（防录屏信息）

2、接口支持https协议

接口URL

http://api.polyv.net/live/v3/channel/anti/record/get

[在线API调用](#)

请求方式

GET

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	false	String	频道号 传入频道号，查询频道内容保护信息 不传频道号，查询账号内容保护默认模板

示例

```
http://api.polyv.net/live/v3/channel/anti/record/get?
appId=frlr1zazn3&sign=879C66B1CF3AB6C8536294E33D492C53&channelId=2453970&timestamp=1629968257454
```

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
data	Object	响应成功时返回频道内容保护信息【详见 data字段描述 】

data字段描述

参数名	类型	说明
antiRecordType	String	内容保护方式 marquee：跑马灯 watermark：水印
modelType	String	内容保护类型 fixed：固定值 nickname：登录用户名 diyurl：URL自定义设置
content	String	固定值时为设置内容 URL自定义设置时为网址
opacity	Integer	透明度
fontSize	String	字体大小 内容保护方式为跑马灯时，设置数值，范围1-256 内容保护方式为水印时 small：小 middle：中 large：大
fontColor	String	跑马灯字体颜色，色值，水印时不返回此参数
autoZoomEnabled	String	跑马灯自定义缩放，水印时不返回此参数 Y：开启 N：关闭
doubleEnabled	String	双跑马灯，水印时不返回此参数 Y：开启 N：关闭
showMode	String	跑马灯显示方式，水印时不返回此参数 roll：滚动 flicker：闪烁

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(MarqueeTest.class);
/**
 * 查询频道内容保护
 * @throws IOException
 * @throws NoSuchAlgorithmException
 */
@Test
public void marqueeGetTest() 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 = "http://api.polyv.net/live/v3/channel/anti/record/get";
    String channelId = "2453970";

    //http 调用逻辑
    Map<String, String> requestMap = new HashMap<>();
    requestMap.put("appId", appId);
    requestMap.put("timestamp", timestamp);
    requestMap.put("channelId", channelId);
    requestMap.put("sign", LiveSignUtil.getSign(requestMap, appSecret));
    String response = HttpUtil.get(url, requestMap);
    log.info("测试查询频道内容保护: {}", 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": {
    "antiRecordType": "marquee",
    "modelType": "fixed",
    "content": "test",
    "opacity": 80,
    "fontSize": "20",
    "fontColor": "#FFF",
    "autoZoomEnabled": "Y",
    "doubleEnabled": "N",
    "showMode": "flicker"
  }
}
```

异常示例

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