

修改频道角色权限设置

接口描述

- 1、修改频道角色权限设置

2、(timestamp, appId) 参与sign签名, 并和sign一起通过url传递, 请求体参数不参与签名, 通过post请求体传递【请设置请求头contentType:application/json】

3、接口支持https协议

接口URL

https://api.polyv.net/live/v4/channel/role-config/update-by-role

[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	Integer	频道号
role	true	String	角色类型 Teacher：讲师、Guest：嘉宾

请求体参数描述

参数名	必选	类型	说明
body	true	Object	字幕信息列表【详见 body参数字段说明 】

body参数字段说明

参数名	必选	类型	说明
webStartCheckInDisplayEnabled	false	String	开播端签到功能显示开关 Y：开启、N：关闭
webStartLotteryDisplayEnabled	false	String	开播端抽奖功能显示开关 Y：开启、N：关闭
webStartQuestionnaireDisplayEnabled	false	String	开播端问卷功能显示开关 Y：开启、N：关闭
webStartAnswerDisplayEnabled	false	String	开播端答题卡功能显示开关 Y：开启、N：关闭
webStartTimerDisplayEnabled	false	String	开播端定时器功能显示开关 (角色类型为讲师时有此字段) Y：开启、N：关闭

参数名	必选	类型	说明
webStartRedPackDisplayEnabled	false	String	开播端红包功能显示开关 Y：开启、N：关闭
webStartCardPushDisplayEnabled	false	String	开播端卡片推送功能显示开关 Y：开启、N：关闭
webStartNotifyDisplayEnabled	false	String	开播端公告功能显示开关 Y：开启、N：关闭
webStartLinkMicDisplayEnabled	false	String	开播端连麦功能显示开关 Y：开启、N：关闭
webStartMultiMediaDisplayEnabled	false	String	开播端多媒体功能显示开关 Y：开启、N：关闭
webStartMembersDisplayEnabled	false	String	开播端成员列表功能显示开关 Y：开启、N：关闭
screenShareRetainCameraEnabled	false	String	共享保持摄像头状态开关 Y：开启、N：关闭
multiplexingDefaultLayout	false	String	混流布局默认应用 (角色类型为讲师时有此字段) 1：单人模式、2：平铺模式、3：主讲模式
cameraSingleModeEnabled	false	String	摄像头单人模式开关 Y：开启、N：关闭
aiVirtualBgUrl	false	String	AI抠像背景图
remixDefaultEnabled	false	String	混音默认开关 Y：开启、N：关闭
showSceneTemplateEnabled	false	String	是否显示场景模板开关 Y：开启、N：关闭
hideFramesDirectionEnabled	false	String	隐藏画面方向切换入口开关 Y：开启、N：关闭
defaultOpenMicLinkEnabled	false	String	是连麦默认方式 (角色类型为讲师时可设置此字段) video：音视频、audio：音频

示例

```
http://api.polyv.net/live/v4/channel/role-config/update-by-role?
channelId=4872391&role=Teacher&timestamp=1721291107000&appId=fso8gzbxlR&sign=D15AFF4476E348FC7F9ECE750B4C96F7
```

请求体json参数：

```
{
  "webStartCheckInDisplayEnabled": "N"
}
```

响应参数描述

参数名	类型	说明
code	Integer	状态码，与 http 状态码相同，用于确定基本的响应状态
status	String	响应结果，由业务决定，成功返回 success，失败返回error
success	Boolean	是否成功响应
requestId	String	请求ID，每次请求生成的唯一的UUID，仅可用于排查、调试，不应该和业务挂上钩
error	Object	状态码非200时的错误信息【详见 Error字段描述 】
data	Object	返回内容

Error参数描述

参数名	类型	说明
code	Integer	错误代码，用于确定具体的错误原因
desc	String	错误描述，与 error.code 对应

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(ChannelOperateTest.class);
/**
 * 修改频道角色权限
 * @throws IOException
 * @throws NoSuchAlgorithmException
 */
@Test
public void updateChannelRoleConfigTest() 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 timestamp = String.valueOf(System.currentTimeMillis());
    int channelId = 100033;

    //业务参数
    String url = "https://api.polyv.net/live/v4/channel/role-config/update-by-role";

    //http 调用逻辑
    Map<String, String> requestMap = new HashMap<>();
    requestMap.put("appId", appId);
    requestMap.put("timestamp", timestamp);
    requestMap.put("channelId", "4872391");
    requestMap.put("role", "Teacher");

    requestMap.put("sign", LiveSignUtil.getSign(requestMap, appSecret));
    url = HttpUtil.appendUrl(url, requestMap);

    Map<String, Object> jsonMap = new HashMap<>();
    jsonMap.put("webStartCheckInDisplayEnabled", "N");

    String json = JSON.toJSONString(jsonMap);

    String response = HttpUtil.postJsonBody(url, json, null);

    log.info("测试修改频道角色权限设置成功: {}", 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（测试示例频道号已隐藏）

```
{
  "code": 200,
  "status": "success",
  "requestId": "14138a5004b1412fbe109f854fec0b52.59.16538790171742883",
  "data": "",
  "success": true
}
```

异常示例

```
{
  "code": 400,
  "status": "error",
  "requestId": "4081dbac03e6441e8bdd301d8feee5a2.124.16360831818611581",
  "error": {
    "code": 20001,
    "desc": "application not found."
  },
  "success": false
}
```