

查询研讨会参会条件

接口描述

1、查询研讨会参会条件
2、接口支持https协议

接口URL

https://api.polyv.net/live/v3/channel/seminar/setting/auth/get

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	true	String	频道号

示例

```
https://api.polyv.net/live/v3/channel/seminar/setting/auth/get?
appId=frrlr1zazn3&sign=8661982EB3646269F9824A6ACCACE25C&timestamp=1621843737789
```

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	Array	请求成功为数据内容，错误为空串【详见 data字段说明 】

data字段说明

参数名	类型	说明
channelId	String	频道号
userId	String	用户ID

参数名	类型	说明
rank	Integer	用于实现一个频道设置两个参会条件，为1或2（1为主要条件，2为次要条件）
enabled	String	是否开启参会条件 N：关闭 Y：开启
authType	String	参会条件类型 password：密码登录 direct：独立授权 phone：白名单
roleCode	String	角色 host：主持人 attendee：参会人
password	String	密码登录的password
directKey	String	独立授权key

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(SeminarAuthTest.class);
/**
 * 查询研讨会参会条件
 * @throws IOException
 */
@Test
public void testGetSeminarAuth() 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/seminar/setting/auth/get";

    //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("测试查询研讨会参会条件接口返回值: {}",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": [{
    "roleId": 12796,
    "userId": "1b448be323",
    "channelId": 2191532,
    "rank": 1,
    "enabled": "Y",
    "authType": "password",
    "roleName": "主持人",
    "roleCode": "host",
    "password": "464nk4",
    "directKey": null,
    "createdTime": 1626830659000,
    "lastModified": 1626830659000
  }, {
    "roleId": 12797,
    "userId": "1b448be323",
    "channelId": 2191532,
    "rank": 1,
    "enabled": "Y",
    "authType": "password",
    "roleName": "参会人",
    "roleCode": "attendee",
    "password": "w8u81f",
    "directKey": null,
    "createdTime": 1626830659000,
    "lastModified": 1626830659000
  }, {
    "roleId": 15832,
    "userId": "1b448be323",
    "channelId": 2191532,
    "rank": 2,
    "enabled": "Y",
    "authType": "direct",
    "roleName": "主持人",
    "roleCode": "host",
    "password": "464nk4",
    "directKey": "b3z0fy1y3z",
    "createdTime": 1628053046000,
    "lastModified": 1628053046000
  }, {
    "roleId": 15833,
    "userId": "1b448be323",
    "channelId": 2191532,
    "rank": 2,
    "enabled": "Y",
    "authType": "direct",
    "roleName": "参会人",
    "roleCode": "attendee",
    "password": "w8u81f",
    "directKey": "1v2n30e15i",
    "createdTime": 1628053046000,
```

```
"lastModified": 1628053046000
  ]
}
```

异常示例

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