

分页查询白名单

接口描述

- 1、分页查询白名单

2、(channelId, timestamp, appId) 参与sign签名, 并和sign一起通过url传递

3、接口支持https协议

接口URL

https://api.polyv.net/meet/v1/channel/auth-whitelist/get

请求方式

GET

接口约束

- 1、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
sign	true	String	签名, 为32位大写的MD5值【详见 签名生成规则 】
channelId	true	Integer	频道号

参数名	必选	类型	说明
pageNumber	false	Integer	当前页码，1~1000之间取值；不传默认1
pageSize	false	Integer	单页数量，1~1000之间取值；不传默认10

示例

```
https://api.polyv.net/meet/v1/channel/auth-whitelist/get?
appId=fyirmfdak4&timestamp=1657786379352&sign=CEA301164F9BA5076F156C1EB60F22F8&channelId=2191532&pageNumber=3&pa
geSize=10
```

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	失败原因，失败时返回。成功时为空【详见 data字段说明 】

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

参数名	类型	说明
desc	String	错误描述, 与 error.code 对应

data字段说明

参数名	类型	说明
pageSize	Integer	每页数据大小, 默认10, 最大值1000
pageNumber	Integer	当前的页数
totalItems	Integer	总的条数
contents	Array	查询的结果列表【详见 Contents参数描述 】
totalPages	Integer	总页数

Contents参数描述

参数名	类型	说明
channelId	Integer	频道号
userId	String	账号id
code	String	会员码
name	String	参会昵称
groupNo	Integer	分组序号
groupName	String	分组名称
groupRole	String	组内身份 leader: 组长 member: 组员

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(WhitelistGet.class);
/**
 * 分页查询白名单
 * @throws IOException
 */
@Test
public void testWhitelistGet() 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/meet/v1/channel/auth-whitelist/get";
    String channelId = "2191532";

    //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("测试分页查询白名单接口返回值: {}",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",
  "requestId": "9d87a0743cd741e2ad7100ee83459f64.62.16655593873260005",
  "data": {
    "pageNumber": 1,
    "pageSize": 10,
    "totalPages": 1,
    "totalItems": 5,
    "contents": [
      {
        "channelId": 2191532,
        "userId": "1b448be323",
        "code": "code1",
        "name": "name1",
        "groupNo": 1,
        "groupName": "1组",
        "groupRole": "leader"
      },
      {
        "channelId": 2191532,
        "userId": "1b448be323",
        "code": "code2",
        "name": "name2",
        "groupNo": 1,
        "groupName": "1组",
        "groupRole": "member"
      },
      {
        "channelId": 2191532,
        "userId": "1b448be323",
        "code": "11",
        "name": "11",
        "groupNo": 1,
        "groupName": "1组",
        "groupRole": "member"
      },
      {
        "channelId": 2191532,
        "userId": "1b448be323",
        "code": "111",
        "name": "name111",
        "groupNo": 1,
        "groupName": "1组",
        "groupRole": "member"
      },
      {
        "channelId": 2191532,
        "userId": "1b448be323",
        "code": "code3",
        "name": "name3",
        "groupNo": 2,
        "groupName": "2组",
        "groupRole": "member"
      }
    ]
  }
}
```

```
    ]
  },
  "success": true
}
```

异常示例

```
{
  "code": 400,
  "status": "error",
  "error": {
    "code": 10003,
    "desc": "时间戳过期"
  },
  "success": false
}
```

```
{
  "code": 403,
  "status": "error",
  "error": {
    "code": 10002,
    "desc": "签名错误"
  },
  "success": false
}
```

```
{
  "code": 400,
  "status": "error",
  "requestId": "9d87a0743cd741e2ad7100ee83459f64.59.16655595201740017",
  "error": {
    "code": 10001,
    "desc": "单次查询不能大于1000条"
  },
  "data": null,
  "success": false
}
```