

获取分配记录

接口描述

- 1、集团账号给分账号分配资源之后，能够通过接口获取分账号的资源
- 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

http://api.polyv.net/live/v4/group/account/allocation-log/list

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](#)
- 2、集团账号接口计算签名请使用集团主账号的appId、appSecret

请求参数描述

参数名	必选	类型	说明
appId	true	String	集团主账号appId【详见 集团账号2.0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 】

请求参数描述

参数名	必选	类型	说明
emails	true	String	分账号邮箱, 多个分账号使用,隔开, 单次请求最大不超过200个
depositStartTime	false	String	开始时间, 格式为yyyy-MM-dd HH:mm:ss
depositEndTime	false	String	结束时间, 格式为yyyy-MM-dd HH:mm:ss
resourceCode	false	String	资源编码,当前支持查询的资源: balance:金额 minutes:直播分钟数 parallelConcurrent:直播并行并发 totalConcurrent:直播累计并发 channels:直播频道数 linkMicDuration:连麦分钟数 guideDuration:导播台分钟数 vodFlow:点播流量 vodSpace:点播空间 groupUserCount:分帐号数量

参数名	必选	类型	说明
alterationType	false	String	变动类型,0：分配 1：回收 2：到期清零
pageNumber	true	Integer	页码,默认为1
pageSize	true	Integer	分页大小,默认为10

示例

```
http://api.polyv.net/live/v4/group/account/allocation-log/list?
appId=g8dtv537aq&timestamp=1653485867000&sign=32389FC010815EDE76028D587FFA6FB3&emails=*****1@polyv.net,*****2@polyv.net
```

响应参数描述

参数名	类型	说明
email	分账号邮箱	
note	分账号备注	
depositTime	分配时间	
resourceCode	资源编码	balance:金额 minutes:直播分钟数 parallelConcurrent:直播并行并发 totalConcurrent:直播累计并发 channels:直播频道数 linkMicDuration:连麦分钟数 guideDuration:导播台分钟数 vodFlow:点播流量 vodSpace:点播空间 groupUserCount:分帐号数量
alterationType	变动类型	0：分配 1：回收 2：到期清零
alteration	变动量，均为正数	点播流量、点播空间传GB

参数名	类型	说明
pageNumber	页码	
pageSize	分页大小	

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);
/**
 * 查询集团账号分配记录
 * @throws IOException
 * @throws NoSuchAlgorithmException
 */
@Test
public void groupAccountBillingTest() throws IOException, NoSuchAlgorithmException {
    //公共参数,填写自己的实际参数
    String appId = super.groupAppId;
    String appSecret = super.groupAppSecret;
    String timestamp = String.valueOf(System.currentTimeMillis());

    //业务参数
    String url = "http://api.polyv.net/live/v4/group/account/allocation-log/list";
    String emails = "*****1@polyv.net,*****1@polyv.net,*****2@polyv.net";

    //http 调用逻辑
    Map<String, String> requestMap = new HashMap<>();
    requestMap.put("appId", appId);
    requestMap.put("timestamp", timestamp);
    requestMap.put("emails", emails);
    requestMap.put("sign", LiveSignUtil.getSign(requestMap, appSecret));

    String response = HttpUtil.get(url, requestMap);
    log.info("查询集团账号分配记录成功: {}", 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

```
{
  "code": 200,
  "status": "success",
  "requestId": "8c0fba9599094d30925ddbd6d6996f6b.63.16902751078127303",
  "data": {
    "pageNumber": 1,
    "pageSize": 20,
    "totalPages": 1,
    "totalItems": 10,
    "contents": [
      {
        "email": "*****@polyv.net",
        "resourceCode": "channels",
        "alteration": "96",
        "depositTime": "2023-01-12 23:47:44",
        "note": "sss",
        "alterationType": 1
      },
      {
        "email": "*****@polyv.net",
        "resourceCode": "channels",
        "alteration": "98",
        "depositTime": "2023-01-12 23:47:36",
        "note": "ss",
        "alterationType": 0
      }
    ]
  },
  "success": true
}
```

异常示例

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