

查询用户等级列表

接口描述

- 1、查询账号的用户等级列表，按等级顺序升序返回
- 2、用户等级为账号级配置，账号下所有频道共用同一套等级；直播模板和频道只控制是否展示等级
- 3、管理后台【用户】→【用户等级】和开放API读写同一份配置，任一入口保存成功后，另一入口查询到相同结果
- 4、账号从未配置过用户等级时，首次查询会初始化5个默认等级（青铜、白银、黄金、钻石、皇冠）并保存
- 5、接口支持https协议

接口URL

```
http://api.polyv.net/live/v4/user/viewer-level/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](#)
- 2、接口根据签名中的 `appId` 确定目标账号，不接收用户ID或频道号
- 3、用户等级的功能说明详见[用户等级](#)

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 】

参数名	必选	类型	说明
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， 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】

示例

```
http://api.polyv.net/live/v4/user/viewer-level/get?
appId=frlrlzazn3&sign=49428741C1D7BEE8BE1EE545F066ECC7&timestamp=1677167340447
```

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参数描述 】
data	Array	用户等级列表，按等级顺序升序排列【详见 data参数描述 】

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

data参数描述

参数名	类型	说明
levelId	String	等级标识，账号内唯一
levelName	String	等级名称
levelIcon	String	等级图标URL，未设置时为null
levelBgColor	String	等级底色，格式为#RGB或#RRGGBB，未设置时为null
description	String	等级说明，仅用于后台识别和管理，不展示给观众，未设置时为null
requiredPoints	Long	积分门槛，即观众获得该等级所需的最低累计获得积分
sortOrder	Integer	等级顺序，从1开始递增

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(getClass());
/**
 * 查询用户等级列表
 * @throws IOException
 * @throws NoSuchAlgorithmException
 */
@Test
public void testViewerLevelGet() 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());

    //业务参数
    String url = "http://api.polyv.net/live/v4/user/viewer-level/get";

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

    String response = HttpUtil.get(url, requestMap);

    log.info("测试查询用户等级列表,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（首次查询时返回的默认等级）

```
{
  "code": 200,
  "status": "success",
  "requestId": "414afeb28f1c4f0ca17e1c3cle39d247.71.16771673423081591",
  "data": [
    {
      "levelId": "bronze",
      "levelName": "青铜",
      "levelIcon": "https://s1.videocc.net/default-img/level-icon/v1/bronze.png",
      "levelBgColor": "#FF6456",
      "description": null,
      "requiredPoints": 1,
      "sortOrder": 1
    },
    {
      "levelId": "silver",
      "levelName": "白银",
      "levelIcon": "https://s1.videocc.net/default-img/level-icon/v1/silver.png",
      "levelBgColor": "#5E81FF",
      "description": null,
      "requiredPoints": 1000,
      "sortOrder": 2
    },
    {
      "levelId": "gold",
      "levelName": "黄金",
      "levelIcon": "https://s1.videocc.net/default-img/level-icon/v1/gold.png",
      "levelBgColor": "#FF894E",
      "description": null,
      "requiredPoints": 3000,
      "sortOrder": 3
    },
    {
      "levelId": "diamond",
      "levelName": "钻石",
      "levelIcon": "https://s1.videocc.net/default-img/level-icon/v1/diamond.png",
      "levelBgColor": "#A361FF",
      "description": null,
      "requiredPoints": 5000,
      "sortOrder": 4
    },
    {
      "levelId": "crown",
      "levelName": "皇冠",
      "levelIcon": "https://s1.videocc.net/default-img/level-icon/v1/crown.png",
      "levelBgColor": "#FB5C91",
      "description": null,
      "requiredPoints": 10000,
      "sortOrder": 5
    }
  ],
  "success": true
}
```

异常示例

```
{
  "code": 400,
  "status": "error",
  "requestId": "d310b70bc329403f87f77f9203d50f89.128.16360831552223589",
  "error": {
    "code": 20001,
    "desc": "application not found."
  },
  "success": false
}
```