

获取应用角色权限

接口描述

- 1、获取应用角色权限

2、接口支持https协议

接口URL

http://api.polyv.net/live/v4/user/webapp-role/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](#)

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 】
roleId	true	Integer	角色ID

示例

```
http://api.polyv.net/live/v4/user/webapp-role/get?roleId=1&appId=frrlr1zazn3&sign=4D418459FA42A9F88475022E33945955&timestamp=1659056102639
```

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	角色权限信息【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
role	Object	角色信息【详见 Role参数描述 】
permissions	Array	权限列表【详见 Permissions参数描述 】

Role参数描述

参数名	类型	说明
id	Integer	角色ID
name	String	角色名称
userId	String	用户ID
description	String	角色描述
status	String	角色状态
createdTime	Long	创建时间（13位毫秒级时间戳）
lastModified	Long	最后修改时间（13位毫秒级时间戳）
permissionName	String	权限名称
isOldRole	Boolean	是否为旧角色
type	String	角色类型(root:仅主账号可见， child: 主账号和子账号都可见)

Permissions参数描述

参数名	类型	说明
id	Integer	权限ID
type	String	权限类型
name	String	权限名称
code	String	权限代码
parentId	Integer	父权限ID
lft	Integer	左节点值（用于树形结构）
rgt	Integer	右节点值（用于树形结构）
createTime	Long	创建时间（13位毫秒级时间戳）
lastModified	Long	最后修改时间（13位毫秒级时间戳）
depth	Integer	权限层级深度
owned	Boolean	是否拥有该权限

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());
/**
 * 获取应用角色权限
 * @throws IOException
 * @throws NoSuchAlgorithmException
 */
@Test
public void getRoleTest() 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/webapp-role/get";
    String roleId = "1";

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

    requestMap.put("sign", LiveSignUtil.getSign(requestMap, appSecret));

    String response = HttpUtil.get(url, requestMap);
    log.info("测试获取应用角色权限成功: {}", 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": "8352399b-d440-4e21-affa-2ea10e734c52",
  "data": {
    "role": {
      "id": 3521,
      "name": "测试角色",
      "userId": "2b3ccac944",
      "description": "这是一个测试角色",
      "status": null,
      "createdTime": 1752731090000,
      "lastModified": 1752731090000,
      "permissionName": null,
      "isOldRole": null,
      "type": "root"
    },
    "permissions": [
      {
        "id": 102,
        "type": "LIVE_V2",
        "name": "直播权限",
        "code": "live_v2",
        "parentId": 0,
        "lft": 1,
        "rgt": 310,
        "createdTime": 1619769766000,
        "lastModified": 1619769766000,
        "depth": 0,
        "owned": false
      },
      {
        "id": 103,
        "type": "LIVE_V2",
        "name": "新建直播",
        "code": "live-create-channel-center",
        "parentId": 1,
        "lft": 2,
        "rgt": 7,
        "createdTime": 1619769799000,
        "lastModified": 1678786231000,
        "depth": 1,
        "owned": false
      }
    ]
  },
  "success": true
}

```

异常示例

```
{
  "code": 400,
  "status": "error",
  "requestId": "d310b70bc329403f87f77f9203d50f89.128.16360831552223589",
  "error": {
    "code": 20001,
    "desc": "角色不存在"
  },
  "success": false
}
```