

修改角色信息

旧版接口地址[修改角色信息\(旧版\)](#)

接口描述

- 1、修改助教或嘉宾的信息
- 2、(channelId, 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/channel/account/update

[在线API调用](#)

请求方式

POST

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、不能同时设置监播和翻页以及在线列表权限

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 】
channelId	true	String	频道ID

请求体参数描述

参数名	必选	类型	说明
account	true	String	助教/嘉宾账号
actor	false	String	头衔，助教默认为助教，长度1~10位
nickName	false	String	昵称，默认随机生成昵称，长度1~15位
avatar	false	String	头像，默认初始头像
passwd	false	String	角色密码，长度6~16位
purviewList	false	Array	权限【详见 purviewList字段说明 】

purviewList参数描述

参数名	必选	类型	说明
code	true	String	权限 chatListEnabled：在线列表（仅支持助教） pageTurnEnabled：翻页（仅支持助教，且仅能设置一个助教有翻页权限） monitorEnabled：监播（仅支持助教，且仅能设置一个助教有监播权限） chatAuditEnabled：聊天审核（仅支持助教） chatInteractionEnabled：互动发言（仅支持助教） chatWhenBannedEnabled：禁言时可发言（仅支持嘉宾）
enabled	true	String	开关 Y：开启 N：关闭

示例

```
http://api.polyv.net/live/v4/channel/account/update?
appId=frlr1zazn3&sign=417802BACAD3375E72CA1A53848E0A53&channelId=2824409&timestamp=1648779734640
```

请求体json参数：

```
{
  "account": "0062824409",
  "nickName": "测试修改昵称",
  "purviewList": [
    {
      "code": "chatListEnabled",
      "enabled": "Y"
    },
    {
      "code": "pageTurnEnabled",
      "enabled": "Y"
    },
    {
      "code": "monitorEnabled",
      "enabled": "N"
    }
  ]
}
```

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	【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参数描述

参数名	类型	说明
account	String	助教/嘉宾账号
userId	String	用户ID
channelId	Integer	频道号
passwd	String	角色密码
nickname	String	角色名称
stream	String	角色流名，单独使用无效
status	String	角色状态 Y：开启 N：关闭
createdTime	Long	创建角色时间，13位毫秒级时间戳
lastModified	Long	角色最后修改时间，13位毫秒级时间戳
sort	Integer	频道角色序号
avatar	String	角色头像
actor	String	角色头衔
role	String	角色 Assistant：助教 Guest：嘉宾
monitorEnabled	String	监播权限 Y：开启 N：关闭
pageTurnEnabled	String	翻页权限 Y：开启 N：关闭

参数名	类型	说明
chatListEnabled	String	在线列表权限 Y：开启 N：关闭
chatAuditEnabled	String	聊天审核权限 Y：开启 N：关闭

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);
/**
 * 更新角色信息
 * @throws IOException
 * @throws NoSuchAlgorithmException
 */
@Test
public void testUpdateChannelAccount() 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/channel/account/update";
    String channelId = "2824409";
    String account = "0062824409";
    String nickName = "测试修改昵称";

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

    List<Map<String, String>> purviewList = new ArrayList<>();
    Map<String, String> purviewMap1 = new HashMap<>();
    purviewMap1.put("code", "chatListEnabled");
    purviewMap1.put("enabled", "Y");
    purviewList.add(purviewMap1);
    Map<String, String> purviewMap2 = new HashMap<>();
    purviewMap2.put("code", "pageTurnEnabled");
    purviewMap2.put("enabled", "Y");
    purviewList.add(purviewMap2);
    Map<String, String> purviewMap3 = new HashMap<>();
    purviewMap3.put("code", "monitorEnabled");
    purviewMap3.put("enabled", "N");
    purviewList.add(purviewMap3);

    Map<String, Object> jsonMap = new HashMap<>();
    jsonMap.put("account", account);
    jsonMap.put("nickName", nickName);
    jsonMap.put("purviewList", purviewList);

    requestMap.put("sign", LiveSignUtil.getSign(requestMap, appSecret));
    url = HttpUtil.appendUrl(url, requestMap);
    String response = HttpUtil.postJsonBody(url, JSON.toJSONString(jsonMap), null);
    log.info("测试更新角色信息成功: {}", 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": "86d11d4aa00b49f5993ac0891e670470.59.16487797376158117",
  "data": {
    "account": "0062824409",
    "userId": "1b448be323",
    "channelId": 2824409,
    "passwd": "test123456",
    "nickname": "测试修改昵称",
    "stream": "l1fs56ih",
    "status": "Y",
    "createTime": 1648778492000,
    "lastModified": 1648779737622,
    "sort": 6,
    "avatar": "//liveimages.videocc.net/uploaded/images/2022/04/g8fydrvq51.png",
    "actor": "助教sadboy",
    "pageTurnEnabled": "Y",
    "notifyEnabled": "Y",
    "checkinEnabled": "Y",
    "voteEnabled": "Y",
    "lotteryEnabled": "Y",
    "chatListEnabled": "Y",
    "chatAuditEnabled": "N",
    "monitorEnabled": "N",
    "roundTourEnabled": "N",
    "watchLockEnabled": "N",
    "role": "Assistant"
  },
  "success": true
}
```

异常示例

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