

设置频道虚拟人数

接口URL

https://api.polyv.net/live/v3/channel/basic/update-rebot-setting

接口说明

- 1、接口用于设置频道虚拟人数信息
- 2、接口支持https

支持格式

JSON

请求方式

POST

请求数限制

TRUE

请求参数

参数名	必选	类型	说明
appId	是	string	从API设置中获取，在直播系统登记的appId
timestamp	是	long	13位当前毫秒级时间戳

参数名	必选	类型	说明
sign	是	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	是	int	频道ID
robotNumber	是	int	虚拟人数,最小0,最大2999
robotNumberTimes	否	int	虚拟人数倍数,默认1,最小1,最大20
addRobotModel	是	string	虚拟人数显示模式，timely立即生效、fixed_time分时生效
changeTime	否	int	分时增加虚拟观众的分时时间，单位毫秒，参数addRobotModel为timely有效，默认20000，最小20000，最大1800000

响应成功JSON示例：

```
{
  "code": 200,
  "status": "success",
  "message": "",
  "data": null
}
```

响应字段说明

参数名	说明
code	状态码,成功为200，签名失败为403，参数错误为400，服务端错误为500

参数名	说明
status	成功为success,错误时为error
message	成功为"", 错误时为错误描述信息
data	响应数据

响应失败JSON示例：

参数错误

```
{
  "code": 400,
  "status": "error",
  "message": "param validate error",
  "data": ""
}
```

未输入appId

```
{
  "code": 400,
  "status": "error",
  "message": "appId is required.",
  "data": ""
}
```

appId不正确

```
{
  "code": 400,
  "status": "error",
  "message": "application not found.",
  "data": ""
}
```

时间戳错误

```
{
  "code": 400,
  "status": "error",
  "message": "invalid timestamp.",
  "data": ""
}
```

签名错误

```
{
  "code": 403,
  "status": "error",
  "message": "invalid signature.",
  "data": ""
}
```

非法频道号

```
{
  "code": 403,
  "status": "error",
  "message": "illegal channel id: %s",
  "data": ""
}
```

java请求示例

```
public void testAddRecordTask() {
    String url = "https://api.polyv.net/live/v3/channel/basic/update-rebot-setting";
    // appId和加密串
    String appId = "xxxxxxx";
    String appSecret = "xxxxxxxxxxxxxxxxxxx";

    Map<String, String> params = new HashMap<>();
    params.put("channelId", "2180007");
    params.put("robotNumber", "10");
    params.put("addRobotModel", "fixed_time");
    params.put("changeTime", "30000");

    // 调用Polyv的工具类方法设置sign
    PolyvTool.setLiveSign(params, appId, appSecret);
    String content = HttpClientUtil.getInstance()
        .sendHttpPost(url, params);
    System.out.println(content);
}
```