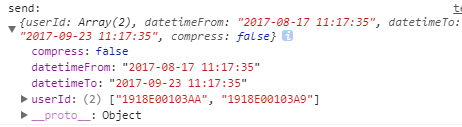
## 查询历史轨迹（有修改）

Send: POST

url: /lbs/get\_history\_location

"userId": ["1918E00103AA", "1918E00103A9"], // "userId": ["all"] for all users  
"datetimeFrom" : "2017-08-17 11:17:35", // 按照时间段查询  
"datetimeTo" : "2017-09-23 11:17:35",  
"compress" : **false,** // 是否压缩，压缩的含义是，服务器会根据特定策略返回部分路径坐标。保留

"page": 1, // 查询的页码。 当记录很多时，需要分页查询。可选参数，默认为第一页  
"rows": 50 // 当前页记录条数。可选参数。默认100条



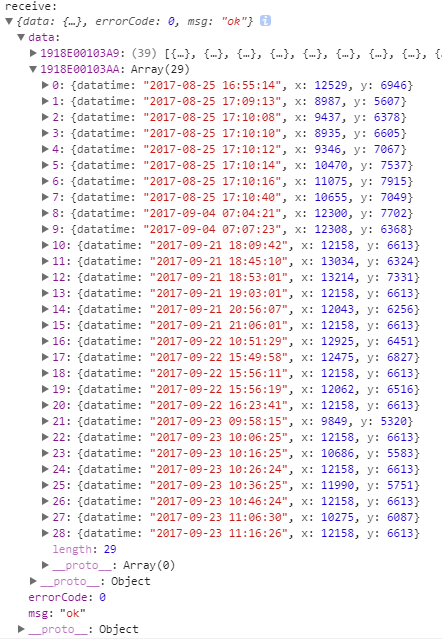
Ajax调用示例：

*/\*\*  
 \* 向服务器发送 Ajax 请求，查询相关数据, POST方式  
 \** ***@param*** *url 请求 url  
 \** ***@param*** *txData 请求 参数，具体参数含义，见对应接口说明  
 \*/***function** ajaxRequest(url, txData) {  
 $.ajax({  
 url: url,  
 async: **true**, // 异步请求  
 type: 'POST',  
 data: {data: JSON.stringify(txData)}

}).done(**function**(data) {

console.log('接口调用成功');  
 }).fail(**function**(jqXHR, textStatus, errorThrown) {  
 console.log('接口调用失败');  
 });  
}

Receive:



返回参数增加 total

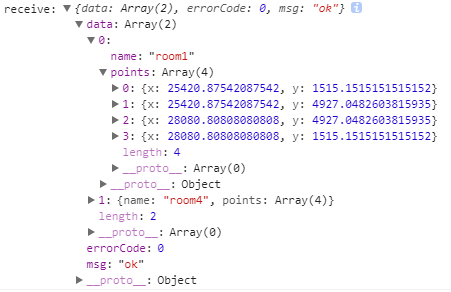
{  
 errorCode 错误码  
 0 成功  
 msg 错误信息  
 'ok' 成功  
 data[{ 历史轨迹记录。排序方式 userId， datatime。如果查询多个用户，无法控制每个用户  
 的记录条数，他们的总条数等于 输入参数 rows  
 x 横坐标，实际物理坐标，单位 mm  
 y 纵坐标  
 datatime 时间格式（北京时间）： yyyy-mm-dd HH-MM-SS  
 }]  
 total 符合条件的记录条数  
}

## 查询电子围栏配置

查询电子围栏配置信息  
输入参数(JSON格式)：  
{  
 data:{ 输入参数  
 floorNo: 查询的楼层  
 }  
}  
:return:  
{  
 errorCode 错误码  
 ==0 成功  
 msg 错误信息  
 =='ok' 成功  
 data:[{ 电子围栏配置  
 name: 围栏名称（所在房间）  
 points:[{ 围栏顶点坐标。（坐标应符合连续逆时针或者顺时针的顺序。当前要求：按逆时针顺序）  
 x: 横坐标，物理坐标，单位 毫米 mm  
 y: 纵坐标，物理坐标，单位 mm  
 }]  
 railNo: 电子围栏编号。后台生成的唯一编号。 added date 2017-10-01  
 id: 电子围栏 Id  
 }]  
}



url = '/lbs/get\_electronic\_rail';  
txData = {  
 "floorNo": "Floor3" // 查询的楼层号  
};



## 电子围栏告警通知

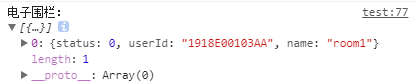
用户进入或者退出电子围栏，服务器推送信息给Web页面

推送给页面的信息是一个数组，数组中每个元素的含义：

userId 用户id，产生电子围栏告警的用户id

name 电子围栏对应的 房间 编号

status 状态, 1 进入围栏，0 退出围栏



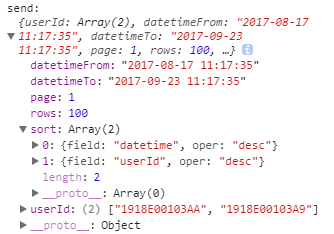
WebSocket接收消息的接口如下：

**var** namespace = '/HeZhong';  
**var** socket = io.connect(location.protocol + '//' + document.domain + ':' + location.port + namespace);  
  
socket.on('hz\_electronic\_tail', **function**(msg) {  
 console.log('电子围栏：', msg);  
});

在多个页面调用WebSocket，注意考虑使用全局WebSocket变量。

## 查询电子围栏告警信息

用户进入或者退出电子围栏，服务器推送信息给Web页面



url = '/lbs/get\_electronic\_rail\_info';  
txData = {  
 "userId": ["1918E00103AA", "1918E00103A9"], // "userId": ["all"] for all users  
 "datetimeFrom": "2017-08-17 11:17:35", // 按照时间段查询, 可选，若不填写，时间段不作为过滤条件  
 "datetimeTo": "2017-09-23 11:17:35", // 结束日期，可选，不填写时，结束日期为 Now  
 "page": 1, // 查询的页码。 当记录很多时，需要分页查询。可选参数，默认为第一页  
 "rows": 100, // 当前页记录条数。可选参数。默认100条  
 "sort": [{"field": "datetime", "oper": "desc"}, // 记录排序规则， 可选参数。  
 {"field": "userId", "oper": "desc"}] // 默认按照时间降序排序  
 // 当前条件只支持 and 。  
 // oper 取值： desc 降序（默认）， asc 升序。 可以不填oper，默认降序  
 // 字段顺序 代表 查询时的排序顺序。  
 // 目前支持 datetime, userId, room排序  
};

可选字段用法：1. 不写该字段，2. 将该字段内容赋值为空字符串。

返回信息：

errorCode 为 0 接口调用成功， 非0失败。具体错误码含义，需要统一定义。

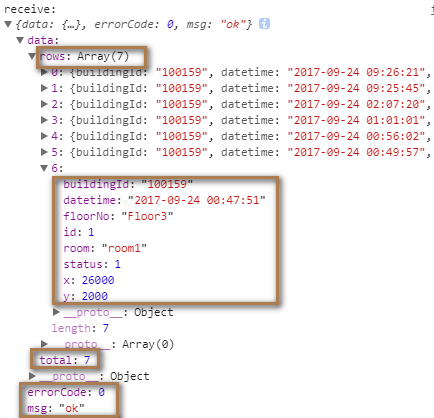
msg 成功为 ‘ok’， 失败时，返回错误原因

data 内为具体的返回记录

其中 total 为 满足查询条件的记录总数。

rows 为记录列表。

记录中的每个字段 见下图



可以参照接口调用示例：

send to server: >>>>>>>>>>  
url: /lbs/get\_electronic\_rail\_info  
{"userId":["1918E00103AA","1918E00103A9"],"datetimeFrom":"2017-08-17 11:17:35","page":1,"rows":100,"sort":[{"field":"datetime","oper":"desc"},{"field":"userId","oper":"desc"}]}

receive from server: <<<<<<<<<<  
{"data":{"rows":[{"buildingId":"100159","datetime":"2017-09-24 09:26:21","floorNo":"Floor3","id":7,"room":"room1","status":1,"x":26000,"y":2000},{"buildingId":"100159","datetime":"2017-09-24 09:25:45","floorNo":"Floor3","id":6,"room":"room1","status":0,"x":26000,"y":5000},{"buildingId":"100159","datetime":"2017-09-24 02:07:20","floorNo":"Floor3","id":5,"room":"room1","status":1,"x":26000,"y":2000},{"buildingId":"100159","datetime":"2017-09-24 01:01:01","floorNo":"Floor3","id":4,"room":"room1","status":0,"x":26000,"y":5000},{"buildingId":"100159","datetime":"2017-09-24 00:56:02","floorNo":"Floor3","id":3,"room":"room1","status":1,"x":26000,"y":2010},{"buildingId":"100159","datetime":"2017-09-24 00:49:57","floorNo":"Floor3","id":2,"room":"room1","status":0,"x":24000,"y":2000},{"buildingId":"100159","datetime":"2017-09-24 00:47:51","floorNo":"Floor3","id":1,"room":"room1","status":1,"x":26000,"y":2000}],"total":7},"errorCode":0,"msg":"ok"}

## 修改/新增电子围栏配置

url: /lbs/electronic\_rail\_cfg\_modify

修改/新增电子围栏信息  
输入参数：  
{  
 data: [{ 信息内容  
 id: 记录ID，新增时 ID可不填或者填 <=0 的数值。 修改时，必须返回记录ID。  
 若查询返回的围栏配置没有记录ID，则不允许修改。  
 name: 电子围栏名称，不允许重复  
 buildingId: 建筑id，保留  
 floorNo: 楼层号  
 points:[{ 围栏顶点坐标。（坐标应符合连续逆时针或者顺时针的顺序。

当前要求：按逆时针顺序）  
 可选参数。 当修改记录时，不涉及顶点修改，可以不传递。

其他情况（新增，修改顶点）都需要  
 传递最终状态的完整顶点信息。  
 x: 横坐标，物理坐标，单位 毫米 mm  
 y: 纵坐标，物理坐标，单位 mm  
 }]  
 }]  
}  
**:return**:  
{  
 errorCode 错误码  
 == 0 成功  
 == 100 [data] field required.  
 == 101 输入参数错误  
 == 102 围栏[%s]已经存在！ (重复添加)  
 == 103 ID为[%d]的电子围栏不存在！ (修改id不存在的电子围栏)  
 == 104 修改后的围栏名称[%s]和现有的重名！  
 == 105 围栏顶点数须大于等于3！  
 msg 错误信息  
 == 'ok' 成功提示  
}

## 删除数据接口

url: /lbs/hz\_data\_del

通用删除入口  
输入参数：  
{  
 who: 要删除的模块。字符串  
 == 'elect\_rail\_cfg' 电子围栏配置 模块  
 ids: [id1, id2, ...] 电子围栏 id list  
}  
**:return**:  
{  
 errorCode 错误码  
 == 0 成功  
 == 101 输入参数错误！  
 == 202 未知模块[ %s ]  
 msg 错误信息  
 == [%d]个围栏，[%d]个顶点信息被删除。 (errorCode = 0 时)  
}