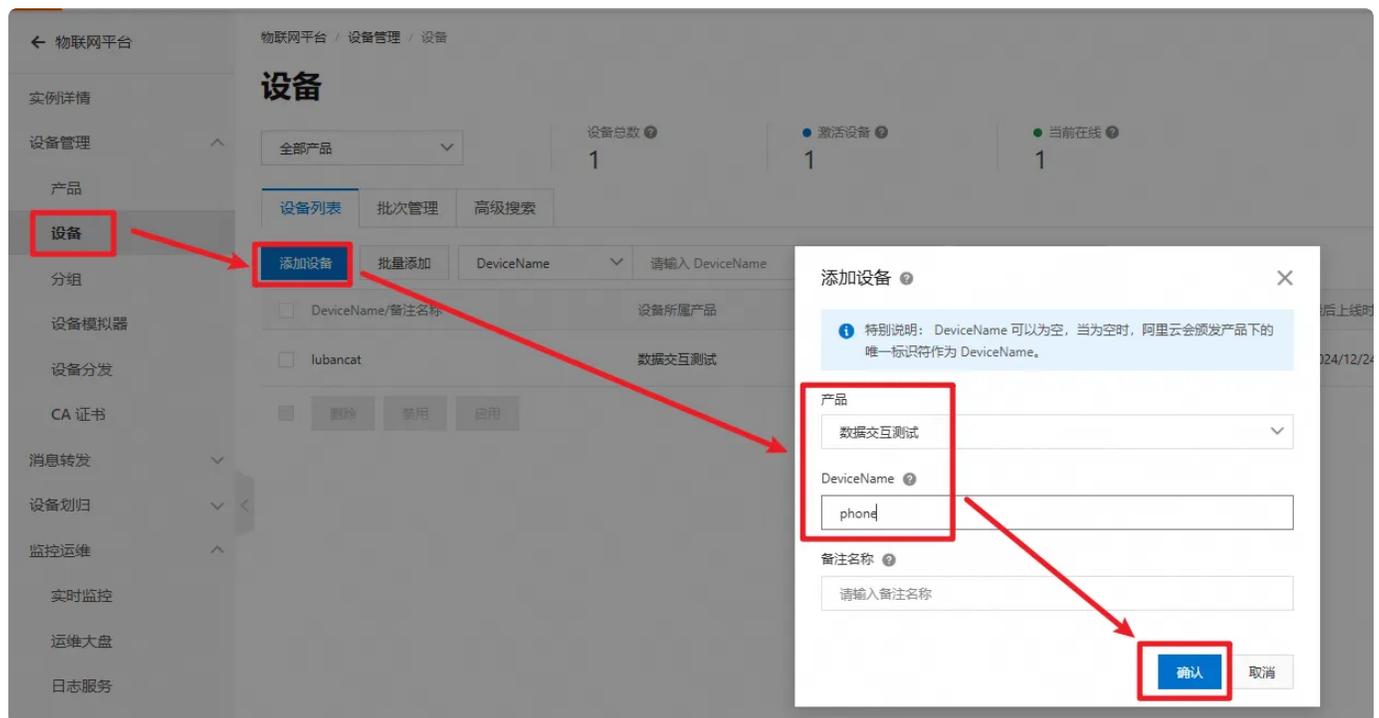


手机MQTT客户端连接阿里云物联网平台

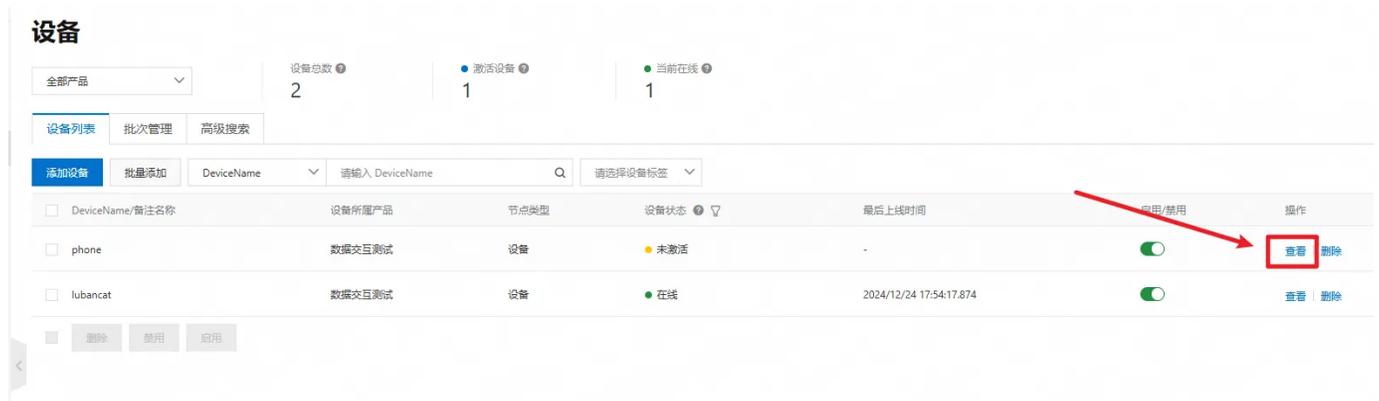
日期：2024-12-24

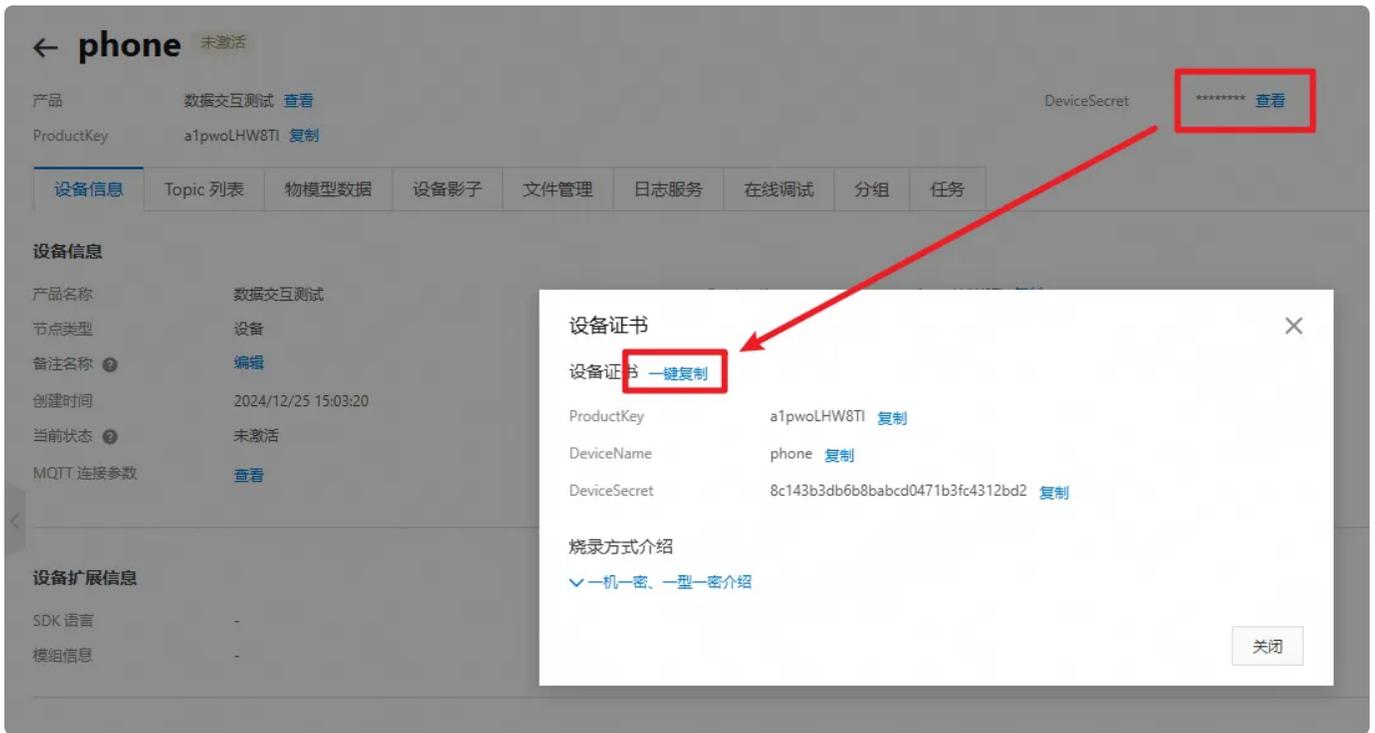
1. 阿里云物联网平台创建手机设备

1. 之前我们在阿里云平台创建了lubancat设备，现在继续创建一个手机设备：



2. 将phone设备的三元素信息保存下来：





3. 下载“MQTT_Password计算工具”，解压后，打开sign.html:

名称	修改日期	类型	大小
hex_hmac_sha1.js	2019/6/20 17:15	JavaScript 文件	4 KB
hex_md5.js	2019/6/20 17:15	JavaScript 文件	9 KB
README.md	2019/6/20 17:15	Markdown File	1 KB
sign.html	2021/9/2 17:36	Microsoft Edge ...	4 KB

[MQTT协议的password计算器，也就是设备端签名](#)

填入设备信息：

productKey:
deviceName:
deviceSecret:
timestamp:
clientId: (注：此处为设备clientId，并非mqttClientId)
method:

点击这里:

计算结果：

mqttClientId: (注：可直接用于mqff.fx的Client ID字段填写)
username:
password:

填入三元素信息，点击“Generate”通过软件获得 MQTT 连接的 Client ID、用户名和密码：

```
{
  "ProductKey": "alpwoLHW8T1",
  "DeviceName": "phone",
  "DeviceSecret": "8c143b3db6b8babcd0471b3fc4312bd2"
}
```

填入设备信息：

productKey: alpwoLHW8TI
deviceName: phone
deviceSecret: 8c143b3db6b8babcd0471b3fc4312bd2
timestamp: 1735110399415
clientId: phone (注：此处为设备clientId，并非mqttClientId) ← 设备名称
method: hmacsha1

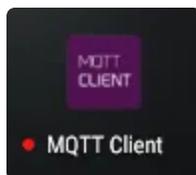
点击这里:

计算结果：

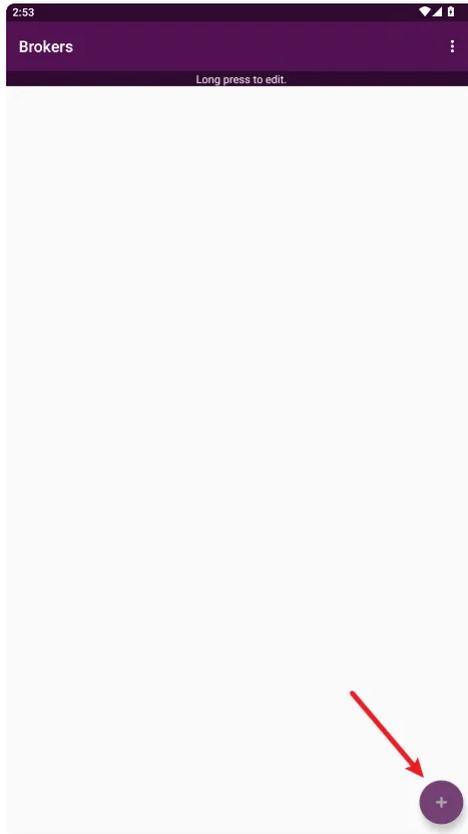
mqttClientId: phone|securemode=2,signmethod=hmacsha1,timestamp=1735110399415 (注：可直接用于mqff.fx的)
username: phone&a1pwoLHW8TI
password: D35B6C8FA98117AC9F4EA5339F1DB2D0DB1048CC

2. 手机安装MQTT客户端软件

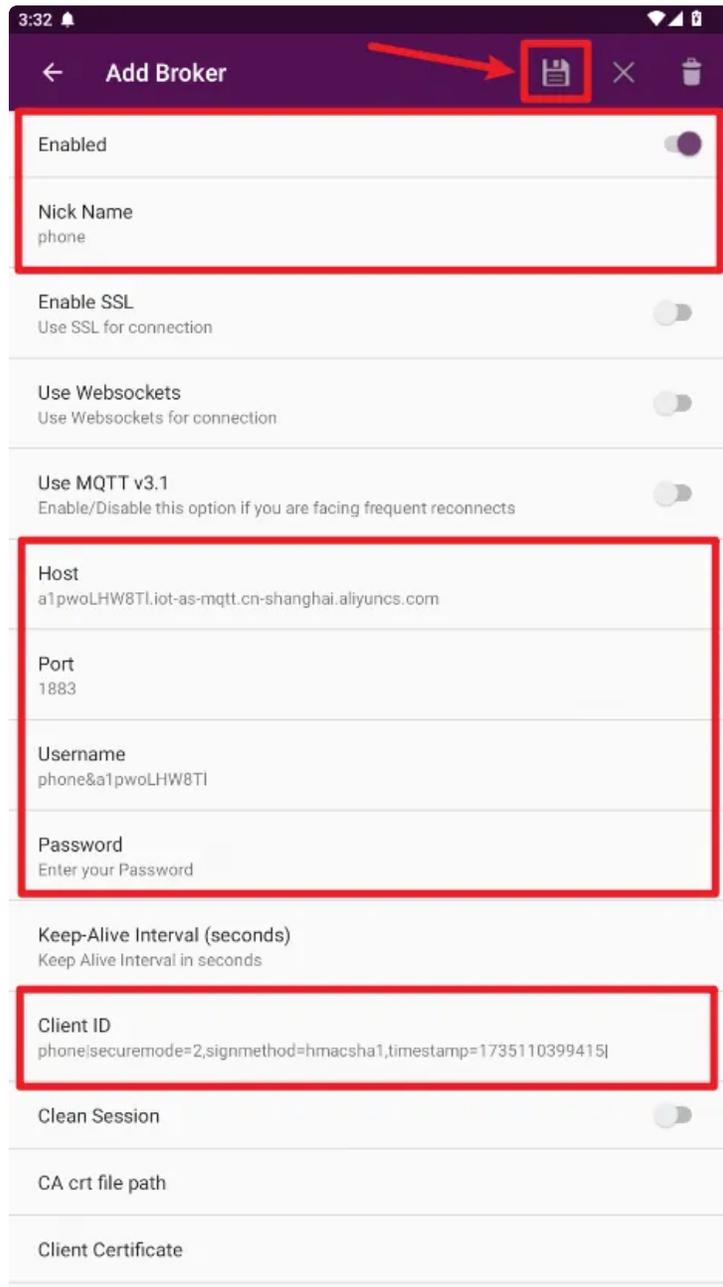
1. 下载“MQTT Client_4.5.1_APKPure.apk”安装包并安装到手机，也可以在电脑端使用安卓模拟器代替没有安卓手机的情况。



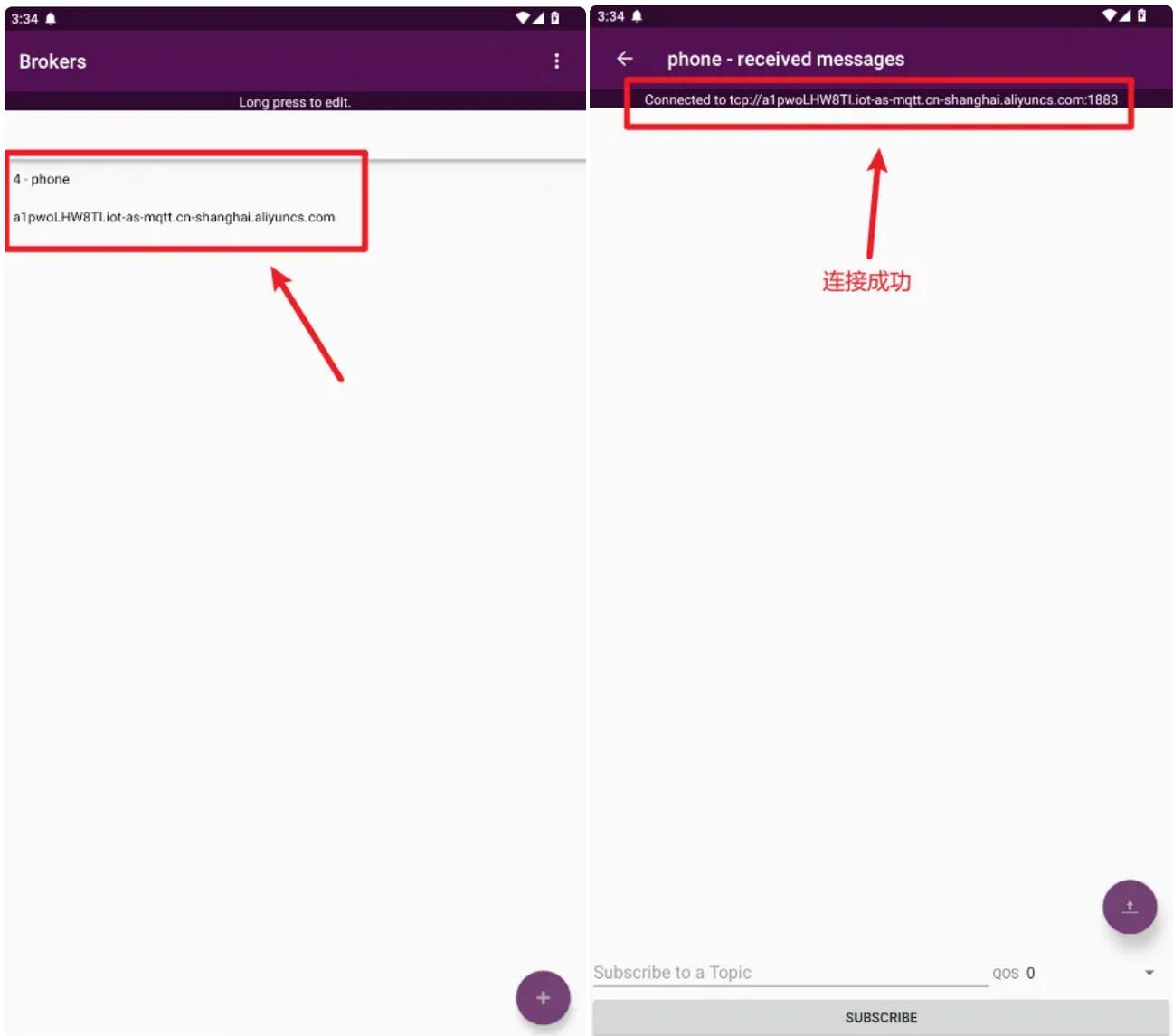
2. 打开MQTT Client，点击右下角的加号图标：



3. 填入昵称、域名、端口、用户名、密码、Client ID, 最后点击右上角的保存按钮:



4. 点击刚刚创建的MQTT客户端：



5. 返回物联网平台，可以看到phone设备已显示在线：

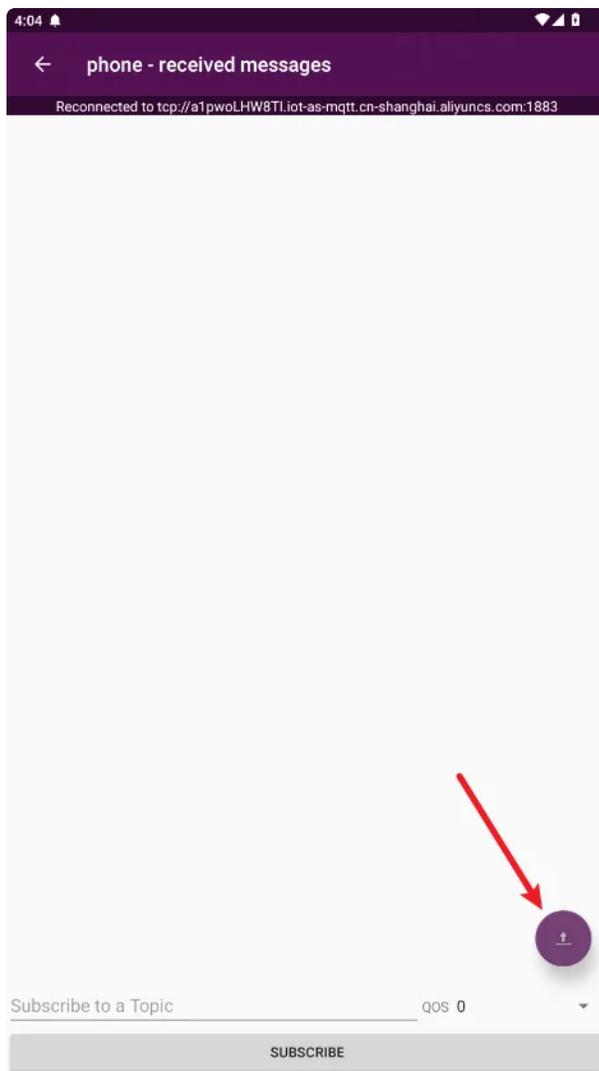
设备

全部产品	设备总数	激活设备	当前在线
全部产品	2	2	2

DeviceName/备注名称	设备所属产品	节点类型	设备状态	最后上线时间
<input type="checkbox"/> phone	数据交互测试	设备	● 在线	2024/12/25 15:34:11.636
<input type="checkbox"/> lubancat	数据交互测试	设备	● 在线	2024/12/24 17:54:17.874

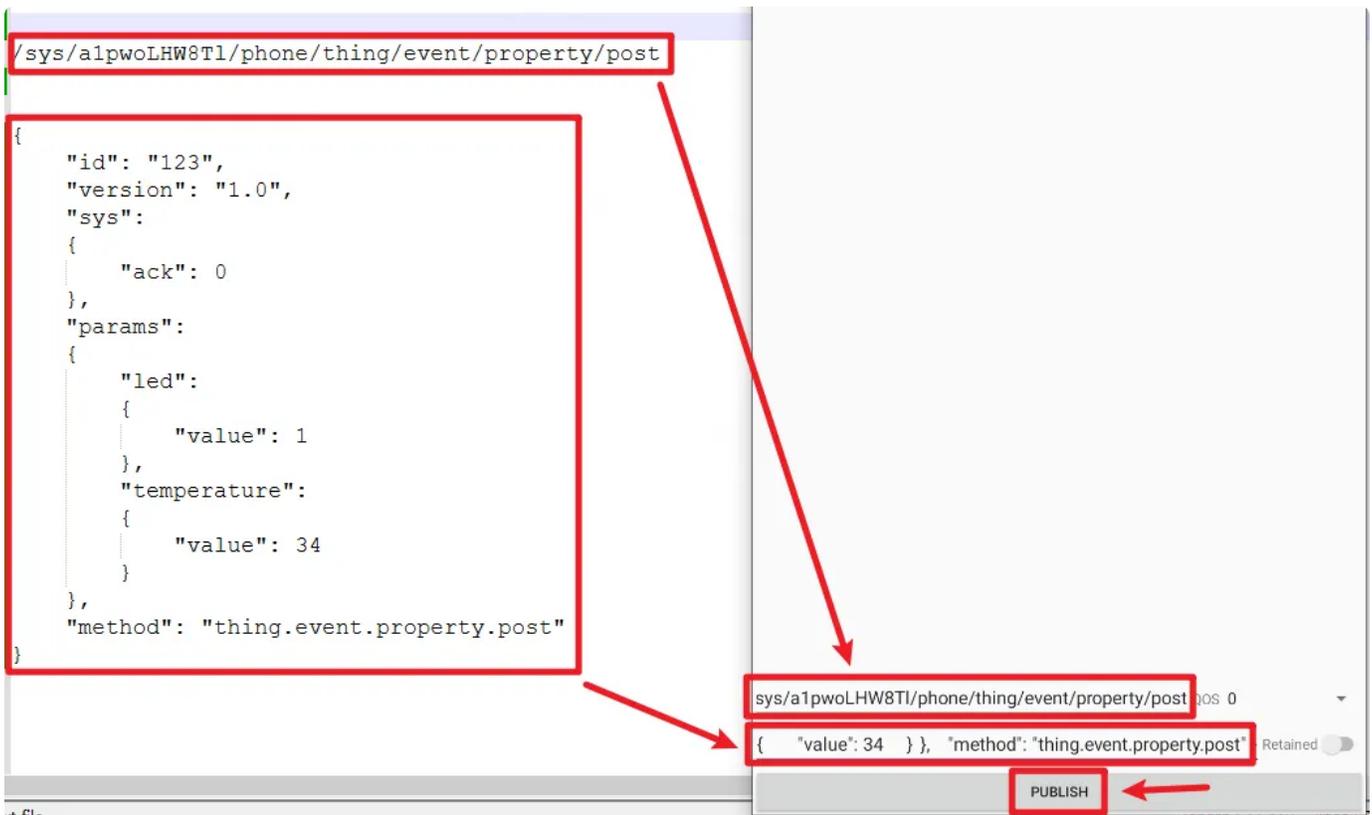
3. 手机发送数据到物联网平台

1. 点击右下角箭头图标：

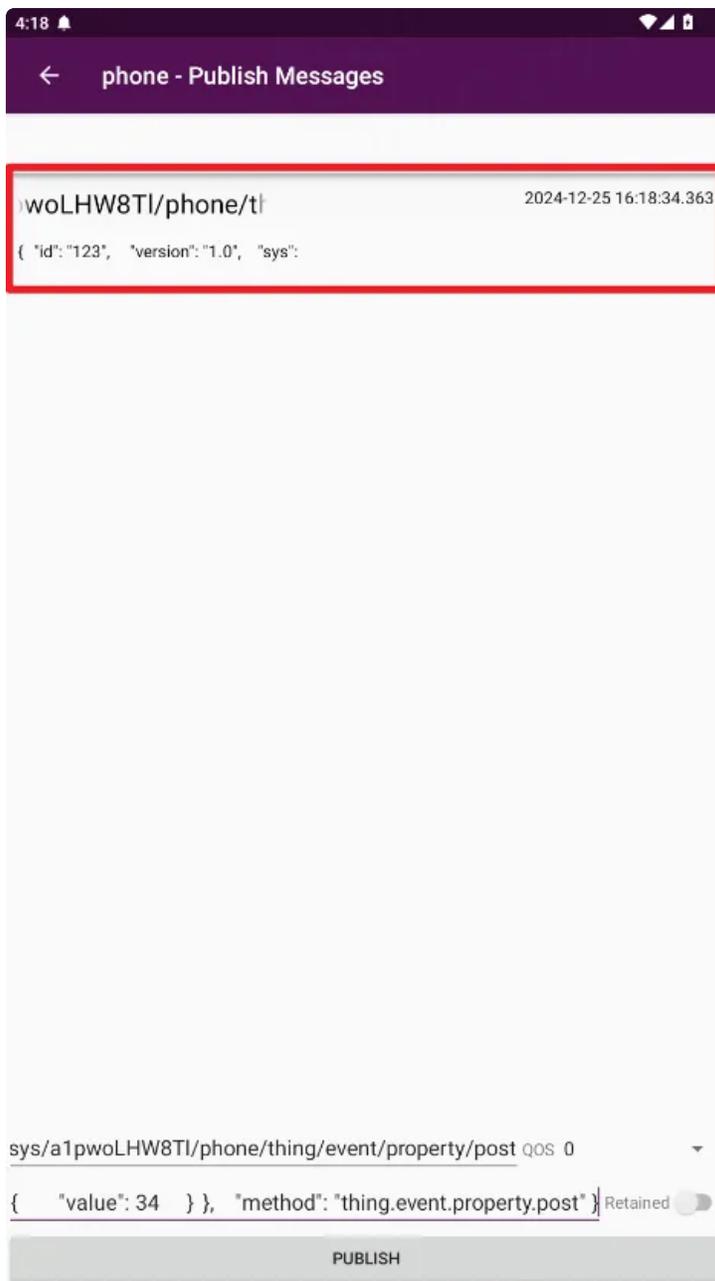


2. 填写Topic和要发送的数据，最后点击最下方的“PUBLISH”按钮：

```
1 # Topic
2 /sys/{your_product_id}/{your_device_name}/thing/event/property/post
3
4 # 要发送的数据, 这里将LED的开关值设置为1, 将温度值设为34摄氏度
5 {
6   "id": "123",
7   "version": "1.0",
8   "sys":
9   {
10    "ack": 0
11  },
12  "params":
13  {
14    "led":
15    {
16      "value": 1
17    },
18    "temperature":
19    {
20      "value": 34
21    }
22  },
23  "method": "thing.event.property.post"
24 }
```



3. 发送成功:



4. 返回物联网平台，查看物模型数据:

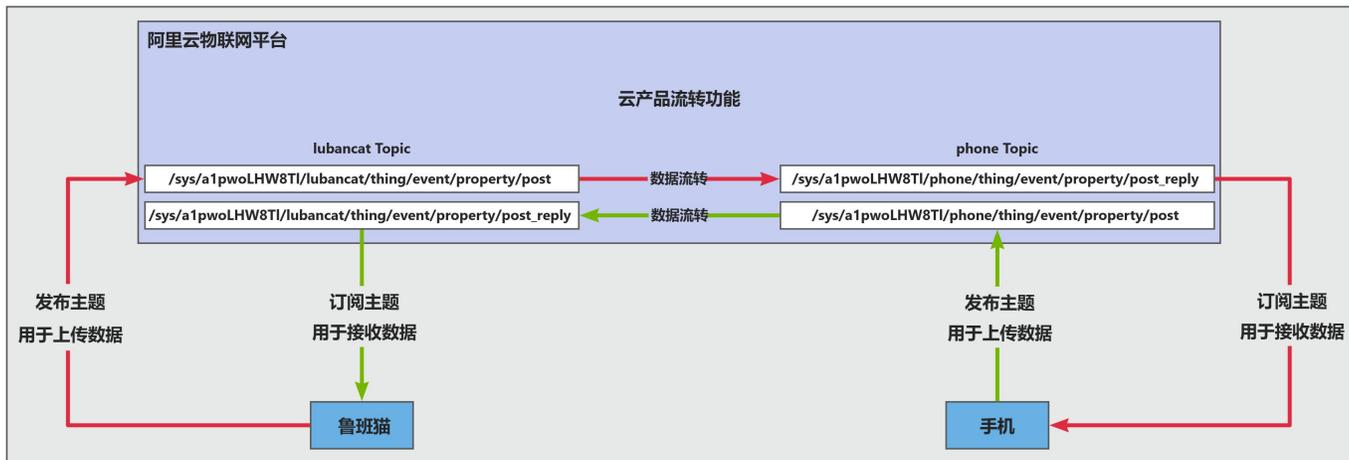
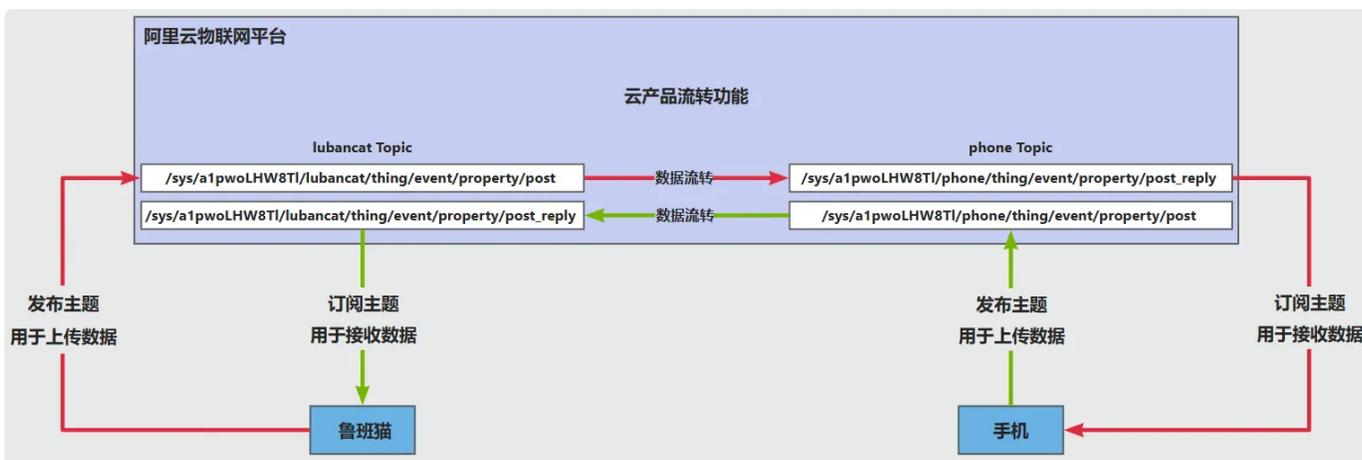


至此，手机发送数据至物联网平台结束。

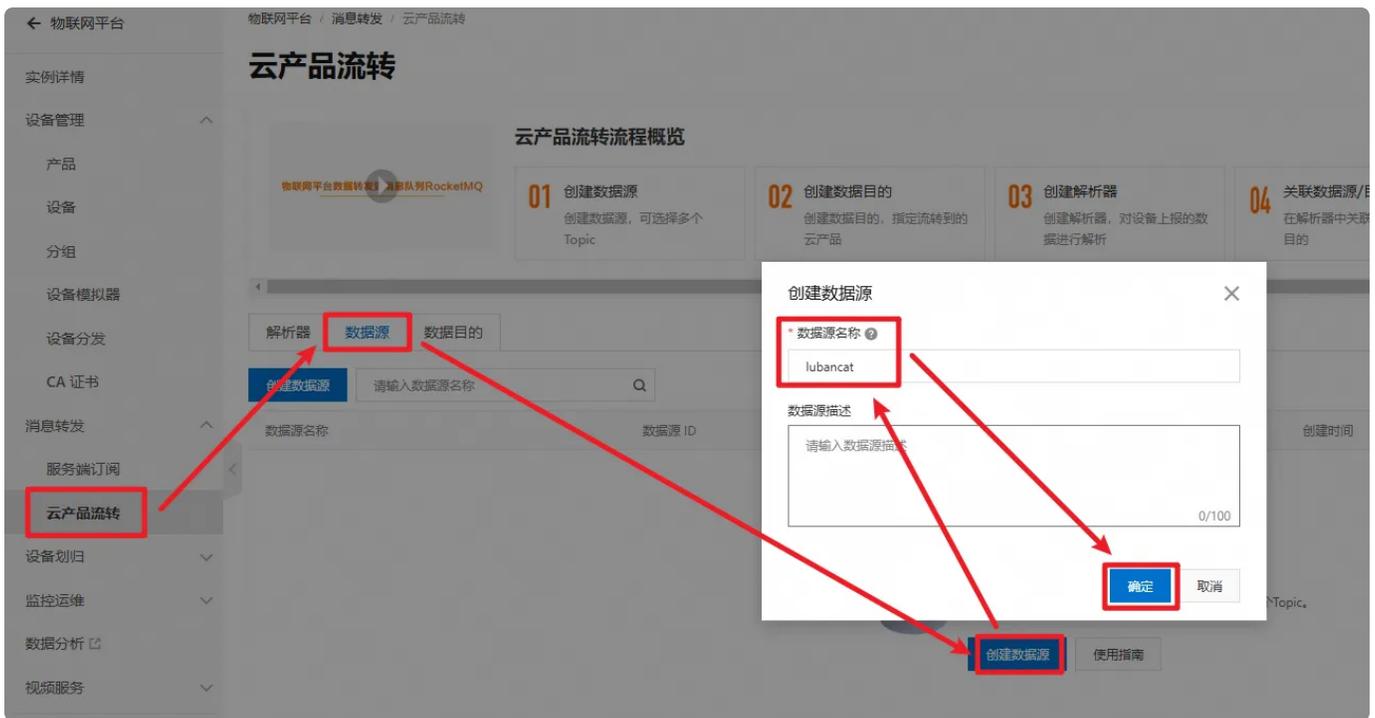
4. 手机和lubancat互收发数据

我们已经在物联网平台创建了两个设备，分别是lubancat和phone设备。目前鲁班猫或手机向云平台发送数据，都是通过订阅系统主题 `/sys/{your_product_id}/{your_device_name}/thing/event/property/post`，然后发送数据。那数据发送完成后如何得知发送结果呢？可以通过订阅 `/sys/{your_product_id}/{your_device_name}/thing/event/property/post_reply` 主题接收返回结果。

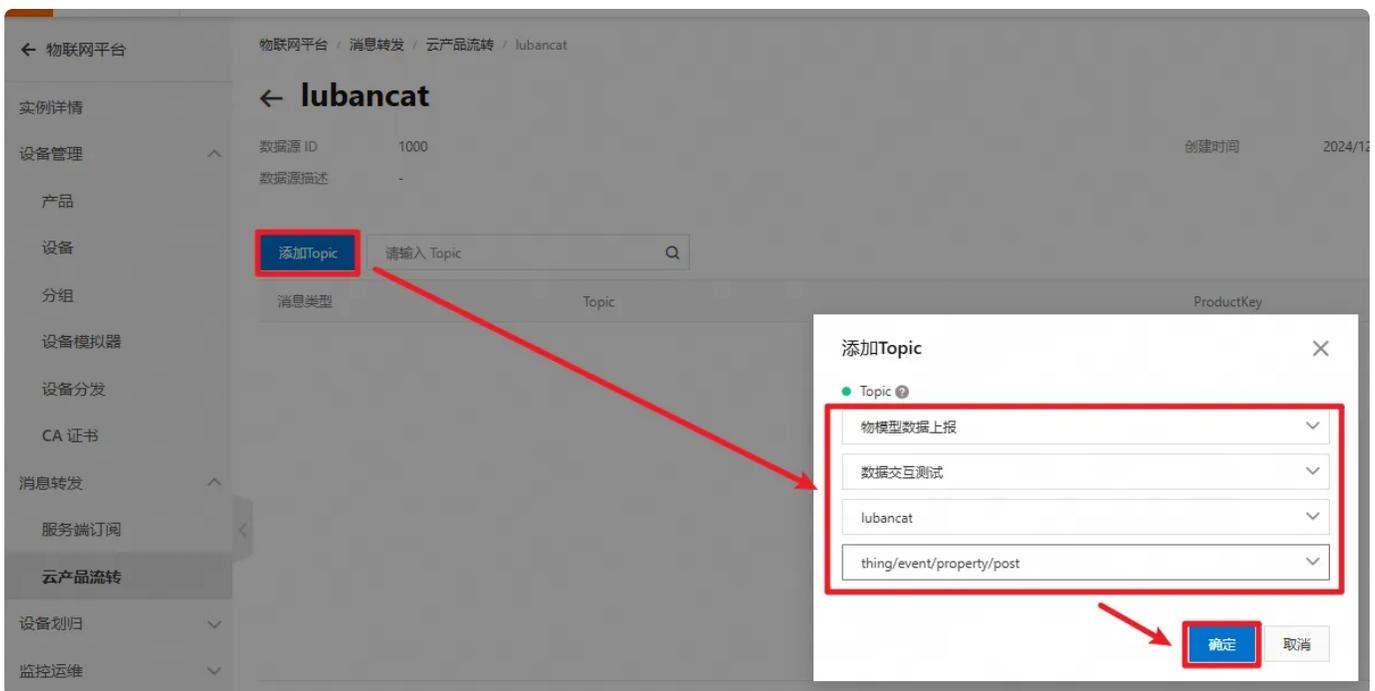
那如何实现手机和鲁班猫之间的数据交互？这里借用云平台的“云产品流转”功能：



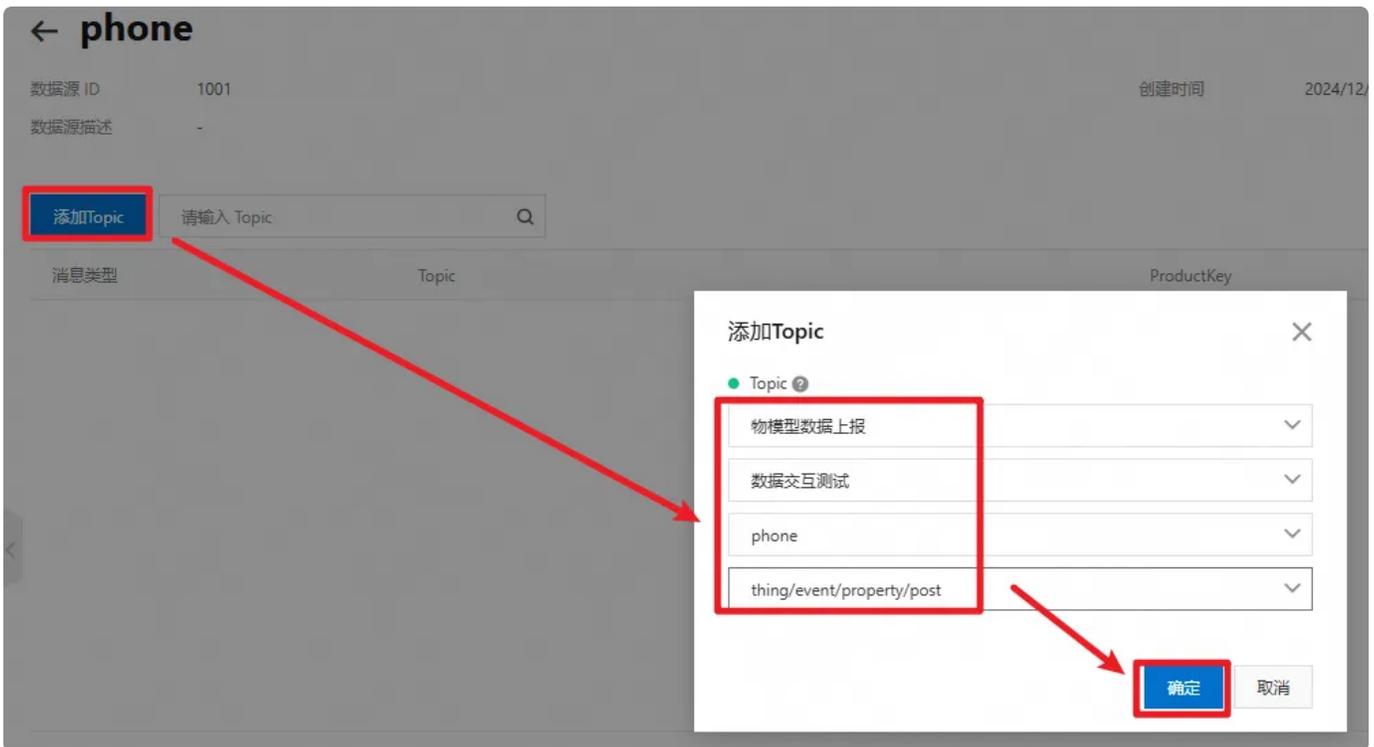
1. 创建数据源：



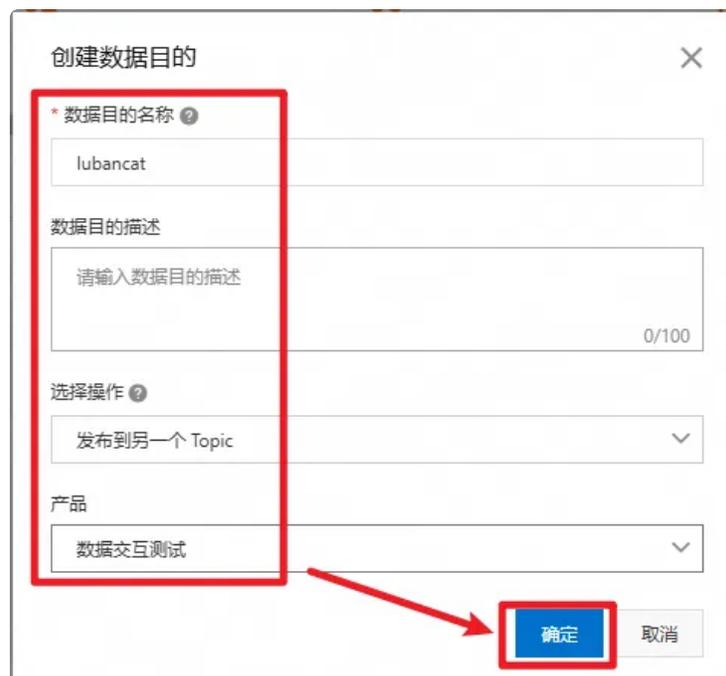
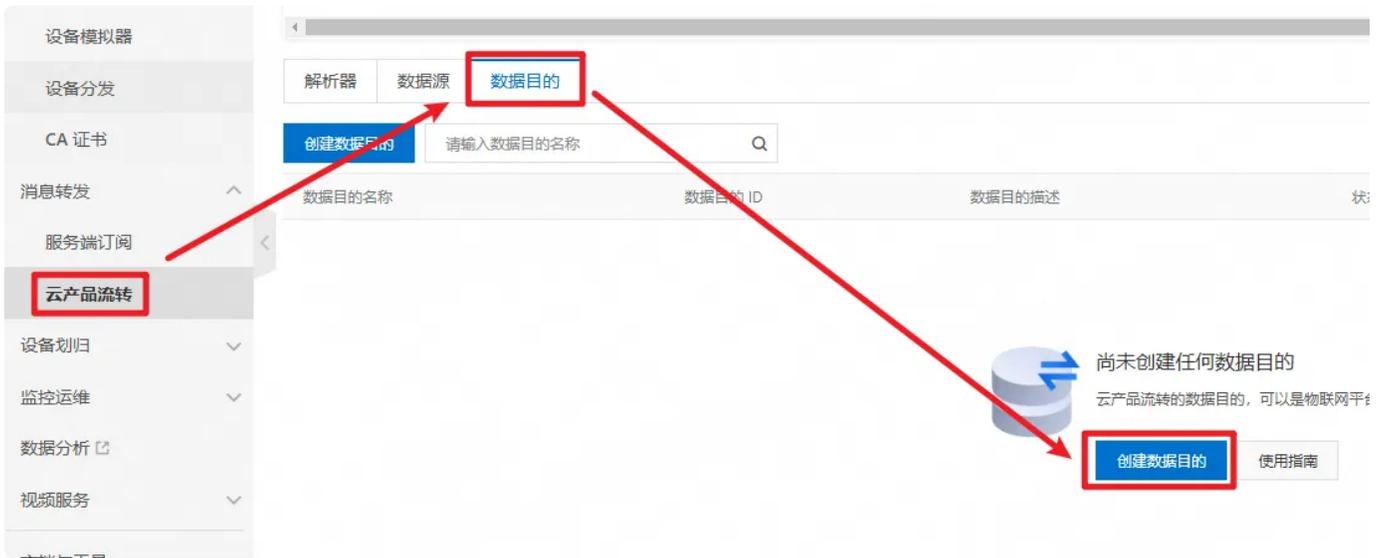
2. 对刚刚创建的数据源添加Topic:



3. 相同的步骤，继续添加手机数据源:



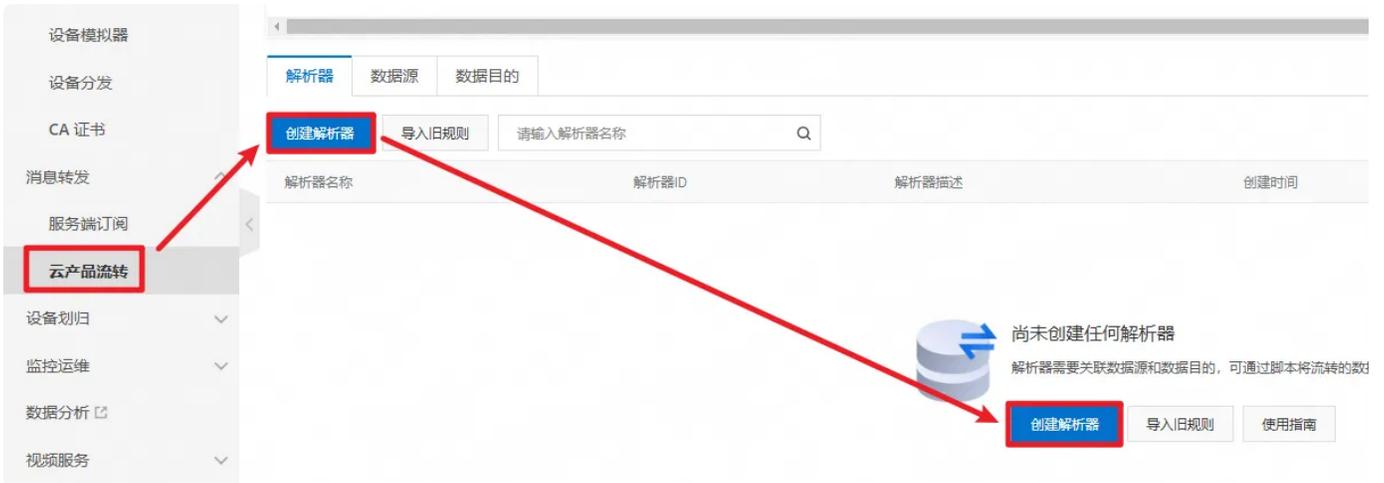
4. 创建lubancat数据目的:



5. 相同的步骤，创建phone数据目的：

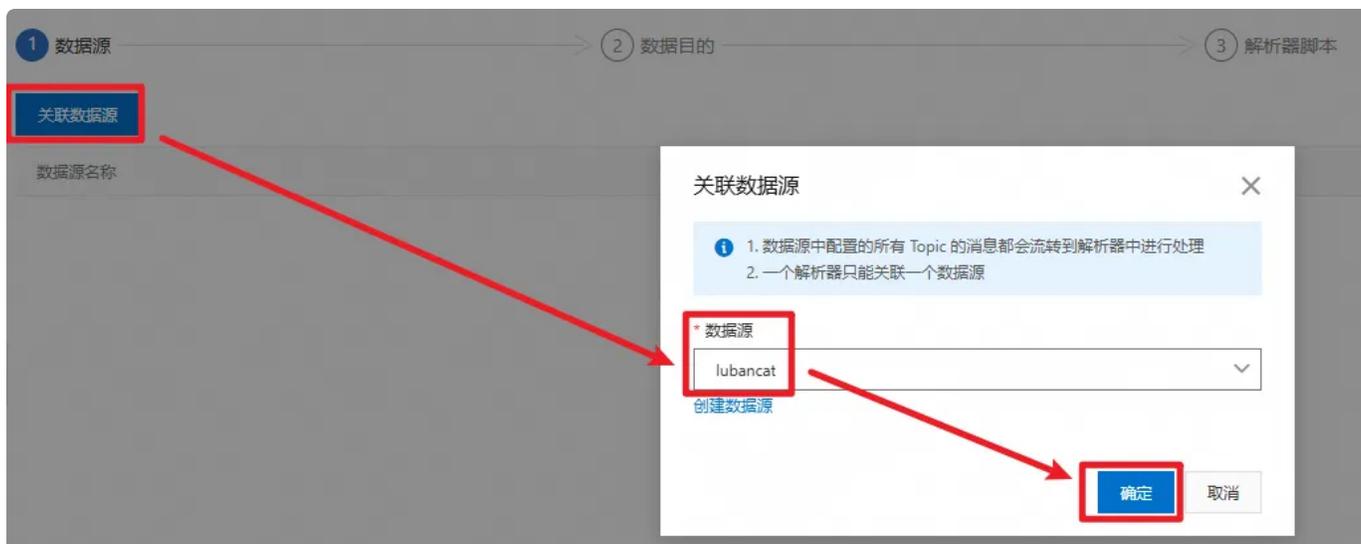


6. 创建lubancat_to_phone解析器：





7. 关联数据源:



8. 关联数据目的:



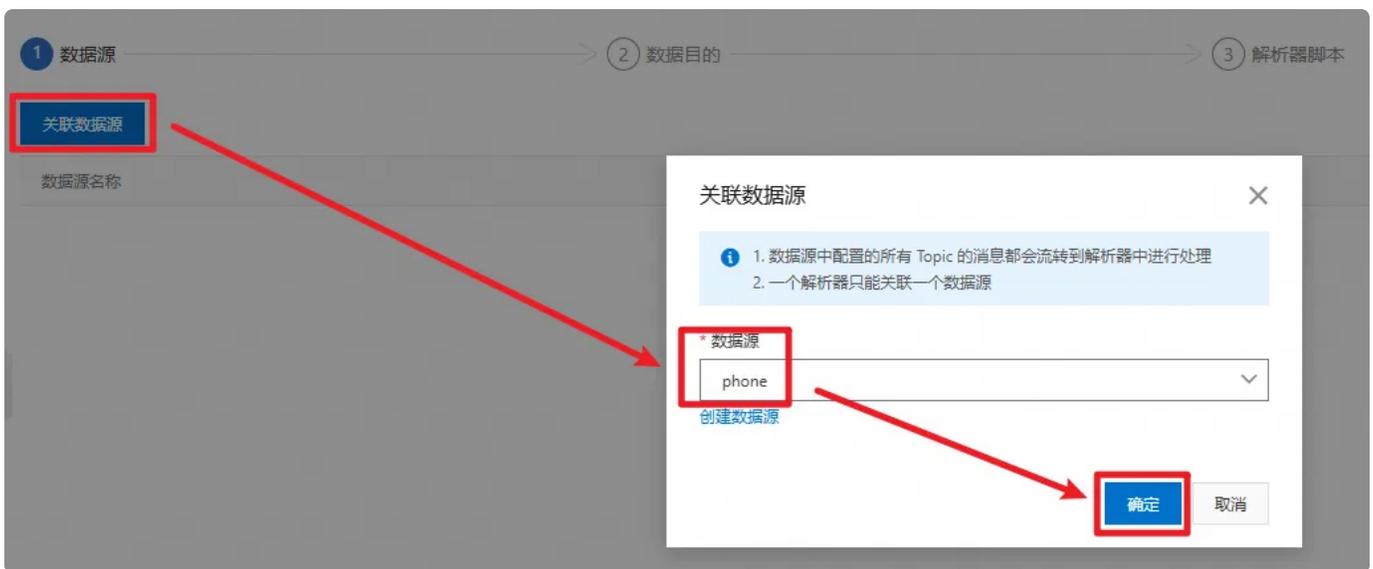
9. 修改解析器脚本，最后保存、发布：



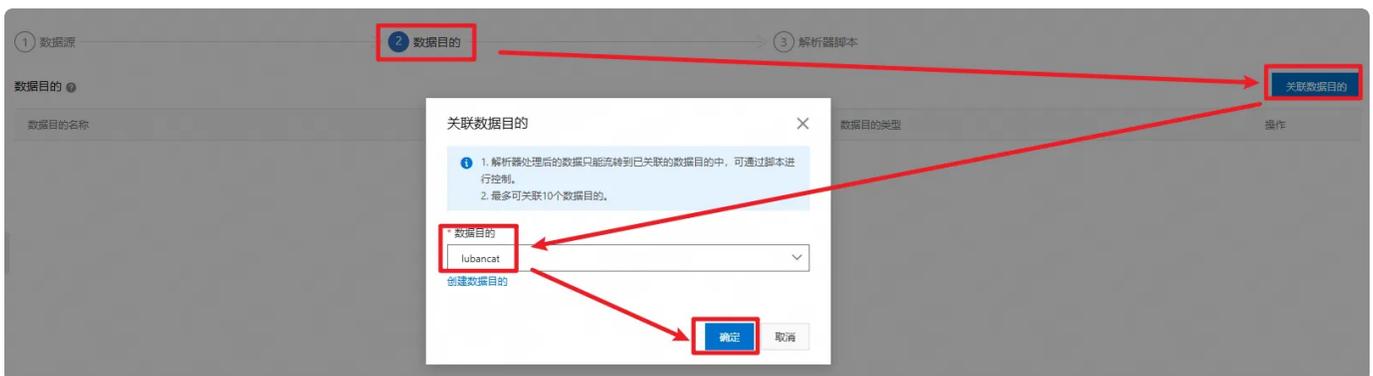
10. 相同的步骤，创建phone_to_lubancat解析器：



11. 关联数据源:



12. 关联数据目的:



13. 修改解析器脚本, 最后保存、发布:

编辑脚本 (当前展示为: 草稿)

```

1 // 草稿页为空时, 进入草稿页会生成默认脚本
2 // 如果默认脚本自动保存过, 继续绑定数据目的, 默认脚本不会自动更新
3 // 此时清空脚本并保存之后, 重新进入草稿页即可重新生成包含最新数据目的的默认脚本
4
5 // 设备上报数据内容, json格式
6 var data = payload('json');
7
8 // 流转到另一个Topic
9 writeIotTopic(1004, "/sys/a1pwoLHW8Tl/lubancat/thing/event/property/post_reply" data);
10

```

↑ 数据目的id

↑ 数据流转到哪个Topic

发布 ← 保存

14. 启动解析器:

云产品流转

解析器名称	解析器ID	解析器描述	创建时间	状态	操作
phone_to_lubancat	1006	-	2024/12/26 17:15:26	未启动	查看 启动 删除
lubancat_to_phone	1005	-	2024/12/26 17:11:50	未启动	查看 启动 删除