

大家好，由于 ONENET 最新改版上线了，所以之前所使用的多协议接入的 MQTT 方案接入方式已经不能再重新创建新设备，尤其对新用户，直接没有入口，这样就给新用户造成了比较大的麻烦，基于此，墨子号科技针对 ONENET 的改版重新提供新的指导教程以及后续的慢慢改版代码都会更新给到大家的，大家可以关注我们的技术群或者是公众号以及淘宝店铺索取最新资料。

目录

1.ONENET 的注册	1
2.注册设备	3
3.物联网模块连接平台	5
4.数据上报与下发	9

1. ONENET 的注册

在百度当中输入 ONENET 关键词，找到 ONENET 注册平台，登录注册即可。



找到 ONENET 官方平台点击进入注册即可。

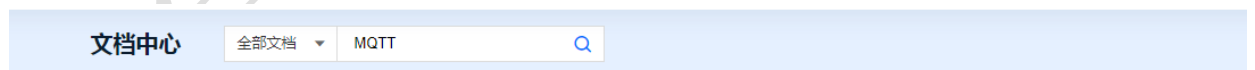


上面就是 ONENET 的注册平台界面了，新用户根据提示注册下帐号即可。注册好之后，进入开发者中心进行产品和设备注册。



进入产品服务里面，左边有产品开发，设备接入管理等。这里我们主要给用户介绍下使用 MQTT 协议来接入 ONENET 该如何注册以及模块端怎么使用。

此外新改版的有很多新的功能，建议用户可以看看他的文档中心，学习下文档使用说明。



MQTT设备连接

MQTT设备连接 服务地址 设备接入支持标准MQTTV3.1.1版本，支持TLS加密，接入服务地址如下：连接协议 证书 地址 端口 说明 MQTT 证书下载 studio-mqtt.heclouds.com studio-mqtts.heclouds.com 加密端口接入 产品与设备创建 按照创建产品与创建设备介绍，创建MQTT，设备登录时需要使用通过key计算出的访问token 来进行访问安全认证，见接入安全认证 设备可通过MQTT

比如我们在文档中心里面输入 MQTT，这个直接提示我们 MQTT 的连接服务器地址以及 MQTT 的版本标准，**注意版本标准是 MQTT3.1.1**。这个很重要，对于移

远的模块需要配置版本的，但是对于移动自家的 4G 模块是不需要配置版本的，这点用户在使用的时候一定要注意。

设备安全认证

设备创建时，平台为每个设备分配了唯一的 key，设备登录时需要使用 通过 key 计算出的访问token 来进行访问安全认证，见[token算法](#)

设备可通过MQTT connect报文进行登录，connect报文中三要素填写方法如下：

参数	是否必须	参数说明
clientId	是	设备名称
username	是	平台分配的产品ID
password	是	填写经过 key 计算的 token

设备接入认证需要有设备 ID，username，password 和之前的多协议接入没有本质的区别，不过 password 需要用工具进行计算，后面的介绍当中会说明如何使用。掌握了其基本的注册说明和接入原理，下一步开始注册设备。

2. 注册设备

创建产品 ×

1 选择产品品类

产品品类 请选择产品品类

免 指平台提供了和物APP标准控制面板

搜索产品品类

2 选择智能化方案

行业

场景

品类

智慧城市

种植养殖

灌溉系统

智慧生活

智能农业

农业监控设备

智慧工业

喷灌智能终端

智慧园区

天窗

智慧商业

侧风机

智慧农业

环流风机

智慧楼宇

外遮阳帘幕

智慧交通

内保温帘幕

智慧物流

湿帘

医疗健康

地源热泵

3 填写信息

电视机

保存 取消

发中

选择合适自己的产品类型进行设备的创建即可。

3.物联网模块连接平台



这里采用移远的 EC800M 模块作为此次连接的说明模块，当然了，只要会一种模块后，后面开发其他的模块基本上都是一样的了。

```
[14:41:58.906]发→◇AT+CSQ
□
[14:41:58.918]收←◆AT+CSQ
+CSQ: 25,99
OK
[14:42:03.258]发→◇AT+CGATT?
□
[14:42:03.270]收←◆AT+CGATT?
+CGATT: 1
OK
[14:42:06.832]发→◇AT+CIMI
□
[14:42:06.836]收←◆AT+CIMI
460081413514499
OK
```

在接入平台之前，一定要先确认卡号，注册网络，信号值等都正常后再进行下一步的连接操作。

准备接入 ONENET 平台，上面有提及 ONENET 接入需要的 MQTT 版本是 V3.1.1。而 EC800M 默认的 MQTT 版本是 3.1.这个对于阿里云就是这样的版本。

所以需要先配置下模块的 MQTT 版本才能进行下一步操作。

```
<vsn>          整型。MQTT 协议版本。
                 3   MQTT 协议 v3.1
                 4   MQTT 协议 v3.1.1
```

在 EC800 MQTT 手册里面有介绍到，设置版本的说明，默认是 3.1，可以设置为 3.1.1。

```
[11:05:38.494]发→◇AT+QMTCFG="version",0,4
□
[11:05:38.499]收←◆AT+QMTCFG="version",0,4
OK
```

按照上面的说明，设置下版本，这个版本一定要设置，否则是登录不了

ONENET 的。

```
[14:49:52.585]发→◇AT+QMTOPEN=0,"183.230.40.96",1883
□
[14:49:52.592]收←◆AT+QMTOPEN=0,"183.230.40.96",1883
OK
[14:49:52.799]收←◆
+QMTOPEN: 0,0
```

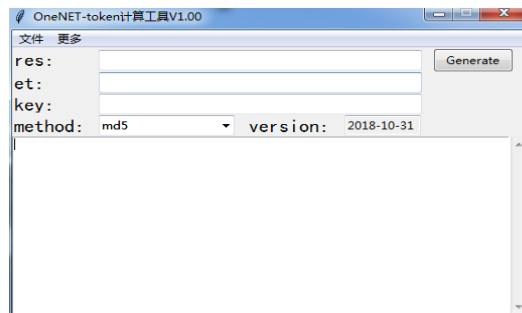
下一步是登录 ONENET 平台，输入 ONENET 平台的 IP 和端口，设置成功后，会返回 0,0，表明地址和端口没问题，下一步进行连接。

因为上面有提供 password 是 TOKEN，所以需要进行下 TOKEN 的计算，需要打开 TOKEN 计算工具。当然了 ONENET 也提供了直接的算法公式进行 TOKEN 的计算，有兴趣的用户可以直接用他的算法进行计算，我们这里采用官方的小工具来获取 TOKEN 值。

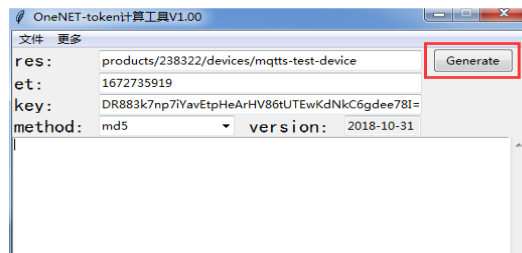
3. Token生成工具

为便于开发者开发，平台提供Token生成工具（[点击下载](#)）

• 打开生成工具 token.exe



• 填写对应参数，点击generate



官方也给出了对应的计算工具说明，用户根据他的介绍填入对应的参数即可。

2. Token算法

Token由多个参数构成，如下表：

名称	类型	是否必须	参数说明	参数示例
version	string	是	参数组版本号，日期格式，目前仅支持"2018-10-31"	2018-10-31
res	string	是	访问资源 resource格式为：products/{产品id}/devices/{设备名字}	products/dafdfadfafdaf/devices/che1
et	int	是	访问过期时间 expirationTime，单位秒，unix时间当一次访问参数中的et时间小于当前时间时，平台会认为访问参数过期从而拒绝该访问	1537255523表示：北京时间 2018-09-18 15:25:23
method	string	是	签名方法 signatureMethod支持hmacmd5、hmacsha1、hmacsha256	sha1（代表使用hmacsha1算法）
sign	string	是	签名结果字符串signature	rBYeJXTp2q4V3C2aj4DBzjaydcw%3D

其中：

1) res:

2



中移物联网有限公司

products/产品 id 号, 或者

products/产品 id 号/devices/设备名称

2) et:

unix 时间, 访问过期时间 expirationTime, 例如 2020-12-31

10:21:08 对应的 et 格式为 1609381268

3) key:

如果 res 填写格式为 products/产品 id 号, 那么 key 填写产品 access_key, 可在产品概述查看;

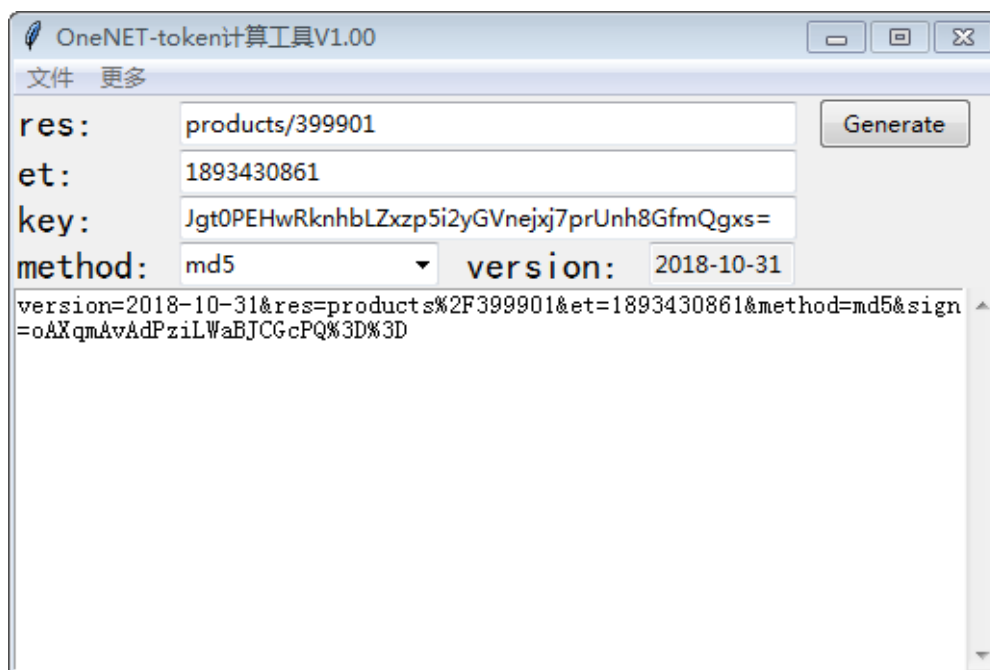
如果 res 填写格式为 products/产品 id 号/devices/设备名称, 那么 key 填写设备 key, 可在设备详情里边查看。

其他参数保持默认值。

他介绍到 res 参数可以有 2 种选择, 目前我们测试来看 res 参数使用的产品 ID 以及 key 使用的 access_key 可以获取到的 TOKEN 值可以正常登录。我们采用这样的方式获取 TOKEN 值进行使用。



在产品里面找到 access_key 进行查看, 这个应该需要输入验证码获取后进行查看。



如上获取到最终的 token 值，然后在使用代码里面进行连接即可。

```
[15:26:19.305]发->◇AT+QMTCONN=0,"mmm","399901","version=2018-10-31&res=products%2F399901&et=1893430861&method=md5&sign=oAXqmAvAdPziLWabJCGcPQ%3D%3D"
[15:26:19.307]收<-◆AT+QMTCONN=0,"mmm","399901","version=2018-10-31&res=products%2F399901&et=1893430861&method=md5&sign=oAXqmAvAdPziLWabJCGcPQ%3D%3D"
JK
[15:26:19.406]收<-◆
+QMTCONN: 0,0,0
```

如上看到返回 0, 0, 0 代表连接成功，此时参数没有问题，就可以设备此时应该在线了。



下一步就可以进行数据的上报与下发测试了。

4. 数据上报与下发

设备在线之后，下一步进行数据的上报与下发测试。需要确认下上报数据的主题以及数据格式这样才好进行测试。

关于数据格式和主题下面这个链接有比较详细的介绍，用户可以学习了解。

<https://open.iot.10086.cn/doc/mqtt/book/device-develop/topics/dp-topics.html>

数据点 topic 簇 topic 列表如下：

系统topic	用途	QoS	可订阅	可发布
\$sys/{pid}/{device-name}/dp/post/json	设备上传数据点	0/1		√
\$sys/{pid}/{device-name}/dp/post/json/accepted	系统通知"设备上传数据点成功"	0	√	
\$sys/{pid}/{device-name}/dp/post/json/rejected	系统通知"设备上传数据点失败"	0	√	

数据点上报的主题，这里我们采用第一个系统 topic 进行数据点上报。

对于我们关注的数据上报问题，主要就是属性上报请求与上报响应。就是对应的发布与订阅，这一点与阿里云极其类似。

设备数据点上传

设备可以通过向系统固定 topic：\$sys/{pid}/{device-name}/dp/post/json 发送数据点存储消息，消息中payload字段数据内容仅支持json格式，示例如下：

```
{
  "id": 123,
  "dp": {
    "temperatrue": [{
      "v": 30,
      "t": 1552289676
    }],
    "power": [{
      "v": 4.5,
      "t": 1552289676
    }],
    "status": [{
      "v": {
        "color": "blue"
      },
      "t": 1552289677
    }, {
      "v": {
        "color": "red"
      },
      "t": 1552289678
    }
  ]
}
```

对于数据格式也有要求，用户需要按照如上的数据格式来实现数据的上报，这样才会在 ONENET 平台进行显示。

其中，参数说明如下：

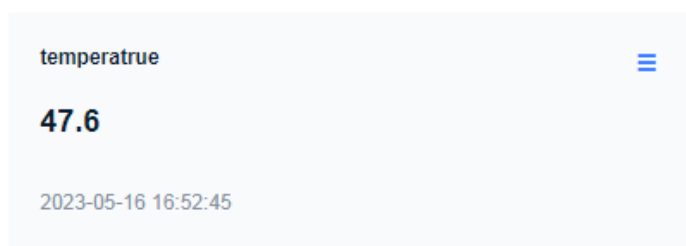
参数	类型	是否必填	说明
id	int	是	消息ID，大于0的整数，数值范围为，4字节有符号数取值范围
dp	object	是	数据点内容，key-value格式，key为数据流名称，value为list格式的一个或者多个数据点值 上例中向temperatrue、power、status三个数据流中上传带时间戳的数据
v	-	是	数据点值，可以为int/float/string/object多种格式
t	int	否	数据点时间戳，unix时间，单位为秒，若不携带，则系统将以数据到达平台时间作为其数据时间

设备数据点上传，支持 QoS1，若 publish QoS=1，则平台回复 publish ack

根据如上描述，下一步进行模块端对接测试实现。

```
[16:52:38.659]发->◇AT+QMQPUBEX=0,0,0,0,"$sys/399901/mmm/dp/post/json",44
[16:52:38.676]收<-◆AT+QMQPUBEX=0,0,0,0,"$sys/399901/mmm/dp/post/json",44
[16:52:43.027]发->◇{"id":111,"dp":{"temperatrue":[{"v":47.6}]}}
[16:52:43.038]收<-◆{"id":111,"dp":{"temperatrue":[{"v":47.6}]}}
[16:52:43.130]收<-◆
OK
+QMQPUBEX: 0,0,0
```

这个是利用 AT 指令进行数据上报，如果主题没有错误的话，那么即便数据格式有问题，那么都会发布成功。在平台端有数据显示。



这样在 ONENET 平台就显示出对应的数据了，这样数据上报就成功了。

```
[16:29:54.418]发->◇AT+QMQPUBEX=0,0,0,0,"$sys/399901/mmm/thing/property/post",87
[16:29:54.427]收<-◆AT+QMQPUBEX=0,0,0,0,"$sys/399901/mmm/thing/property/post",87
[16:29:59.972]发->◇{"id":"123","version":"1.0","params":{"Power":{"value":"12345"},"temp":{"value":23.6}}}}
[16:29:59.983]收<-◆{"id":"123","version":"1.0","params":{"Power":{"value":"12345"},"temp":{"value":23.6}}}}
[16:30:00.081]收<-◆
OK
+QMQPUBEX: 0,0,0
+QMQSTAT: 0,1
```

使用 onenet studio 的数据格式来上报的话，就会提示与服务器断开连接，表明主题或者是数据是有问题的，用户需要查询。

这样完成了 ONENET 的数据的上报，下面测试下 ONENET 的数据下发功能，对于模块而言，需要订阅下主题才可以进行收到 ONENET 平台的数据下发，这点用户需要注意。

```
[17:04:01.308]发->◇AT+QMQSUB=0,1,"$sys/399901/mmm/#",0
[17:04:01.317]收<-◆AT+QMQSUB=0,1,"$sys/399901/mmm/#",0
OK
[17:04:01.540]收<-◆
+QMQSUB: 0,1,0,0
```

首先发个指令订阅下主题，这个是用来接收平台下发数据的，不然不订阅收不到下发数据。这个#代表的是所有的平台主题都进行订阅。

← mmm

设备详情 数据流 文件管理 命令下发 设备镜像

命令内容 字符串 *响应时间 10 ▶ 下发

hello,mozihao

响应时间

返回结果

在平台端找到命令下发的地方，然后输入字符串或者十六进制数据进行数据下发。然后在模块端进行订阅查看。

```
[17:06:52.096]收←◆  
+QMTRECV: 0,0,"$sys/399901/mmm/cmd/request/4c45eea5-8dcb-489c-858d-33fd24801979","hello,mozihao"
```

可以看到接收到了平台下发的数据了。这样对于模块的上传与下发功能都完成了。

后续我们会针对的更新老版本的 MQTT 代码，陆续更新到最新的 ONENET 物联网平台代码来，更加方便用户来使用 ONENET 平台，谢谢！