

作业1-2021/7/6

1. UDP测试数据 (学号2018080405)

```
正在打开串口...
[12:20:29.604]发->◇AT
[12:20:29.609]收<-◆AT
OK
[12:20:35.193]发->◇AT+CIMI
[12:20:35.199]收<-◆AT+CIMI
460043331905082
OK
[12:20:38.087]发->◇AT+CSQ
[12:20:38.093]收<-◆AT+CSQ
+CSQ: 15,0
OK
[12:20:40.580]发->◇AT+CGATT?
[12:20:40.585]收<-◆AT+CGATT?
+CGATT: 1
OK
[12:20:45.234]发->◇AT+QIOPEN=1,0,"UDP","47.92.146.210",9999,0,1
[12:20:45.245]收<-◆AT+QIOPEN=1,0,"UDP","47.92.146.210",9999,0,1
[12:20:45.265]收<-◆
OK
+QIOPEN: 0,0
[12:20:49.417]发->◇AT+QISEND=0,10,2018080405
[12:20:49.426]收<-◆AT+QISEND=0,10,2018080405
OK
SEND OK
[12:20:50.697]收<-◆
+QIURC: "recv",0,10
2018080405
```

AT : 测试串口

AT+CIMI: 测试是否有卡

AT+CSQ: 返回信号质量

CGATT=1 已附着网络

连接厂家服务器测试UDP

发送测试数据

接收的数据与发送相同

2. TCP测试数据 (学号2018080405)

[12:29:13.031]发→◇AT+CSQ

□

[12:29:13.037]收←◆AT+CSQ

+CSQ: 15,0

OK

[12:29:13.674]发→◇AT+CGATT?

□

[12:29:13.680]收←◆AT+CGATT?

+CGATT: 1

OK

[12:29:19.957]发→◇AT+QICLOSE=0

□

[12:29:19.963]收←◆AT+QICLOSE=0

OK

关闭套接字

CLOSE OK

[12:29:22.518]发→◇AT+QIOPEN=1,0,"TCP","47.92.146.210",8888,0,1

□

[12:29:22.529]收←◆AT+QIOPEN=1,0,"TCP","47.92.146.210",8888,0,1

OK

TCP测试

[12:29:23.824]收←◆

+QIOPEN: 0,0

[12:29:26.412]发→◇AT+QISTATE?

□

[12:29:26.419]收←◆AT+QISTATE?

+QISTATE: 0,"TCP","47.92.146.210",8888,0,2,1,1

查询服务状态

OK

[12:29:45.751]发→◇AT+QISEND=0,10,2018080405

□

[12:29:45.760]收←◆AT+QISEND=0,10,2018080405

OK

发送数据

SEND OK

[12:29:47.893]收←◆

+QIURC: "recv",0,10
2018080405

→ 接收数据=发送数据

3. MQTT 主题 (fudingyi) 测试数据2018080405

OK

[12:39:14.668]发→◇AT+QMTOPEN=0,"47.92.146.210",1883

□

[12:39:14.678]收←◆AT+QMTOPEN=0,"47.92.146.210",1883

OK

测试MQTT

[12:39:16.156]收←◆

+QMTOPEN: 0,0

[12:39:18.141]发→◇AT+CGSN=1

□

[12:39:18.151]收←◆AT+CGSN=1

+CGSN: 863409053773569

查询IMEI号

OK

[12:39:19.952]发→◇AT+QMTCONN=0,"863409053773569"

□

[12:39:19.962]收←◆AT+QMTCONN=0,"863409053773569"

OK

连接客户端至MQTT服务器

[12:39:20.872]收←◆

+QMTCONN: 0,0,0

[12:39:28.769]发→◇AT+QMTPUB=0,0,0,0,"fudingyi","2018080405"

□

[12:39:28.779]收←◆AT+QMTPUB=0,0,0,0,"fudingyi","2018080405"

OK

发布"fudingyi"主题，2018080405消息

+QMTPUB: 0,0,0

[12:39:31.456]发→◇AT+QMTSUB=0,1,"fudingyi",0

□

[12:39:31.465]收←◆AT+QMTSUB=0,1,"fudingyi",0

[12:39:31.483]收←◆

OK

订阅"fudingyi"主题

[12:39:32.497]收←◆

+QMTSUB: 0,1,0,0

[12:39:35.478]发→◇AT+QMTPUB=0,0,0,0,"fudingyi","2018080405"

□

[12:39:35.488]收←◆AT+QMTPUB=0,0,0,0,"fudingyi","2018080405"

OK

再次发布消息

+QMTPUB: 0,0,0

[12:39:36.657]收←◆

+QMTRECV: 0,0,"fudingyi","2018080405"

接收到消息

```
[12:41:31.095]发->◇AT+QMTUNS=0,2,"fudingyi"  
[12:41:31.104]收<-◆AT+QMTUNS=0,2,"fudingyi"  
[12:41:31.127]收<-◆  
OK  
  
[12:41:33.160]收<-◆  
+QMTUNS: 0,2,0  
  
[12:41:38.458]发->◇AT+QMTPUB=0,0,0,0,"fudingyi","2018080405"  
[12:41:38.468]收<-◆AT+QMTPUB=0,0,0,0,"fudingyi","2018080405"  
OK  
  
+QMTPUB: 0,0,0,0
```

退订fudingyi主题

publish消息之后不会再收到

4. 串口调试软件中保存的命令

-拖动加宽 <input type="checkbox"/> 循环发送 多条帮助 导入ini		顺序 延时	
HEX	字符串(双击注释)	点击发送	ms
<input checked="" type="checkbox"/>	13 00 FF 88	十六进制数据串1	1 1000
<input type="checkbox"/>	AT\r\n	测试	3 1000
<input type="checkbox"/>	AT+CIMI\r\n	测试卡	2 1000
<input type="checkbox"/>	AT+CSQ\r\n	信号质量	0 1000
<input type="checkbox"/>	AT+CGATT?\r\n	注册?	0 1000
<input type="checkbox"/>	AT+QIOPEN=1,0,"UDP","47.92.146.210",9999,0,1\r\n	UDP TEST	0 1000
<input type="checkbox"/>	AT+QISEND=0,10,2018080405\r\n	SEND STUDENT No.	0 1000
<input type="checkbox"/>	AT+QICLOSE=0\r\n	关闭套接字	0 1000
<input type="checkbox"/>	AT+QIOPEN=1,0,"TCP","47.92.146.210",8888,0,1\r\n	TCP TEST	0 1000
<input type="checkbox"/>	AT+QISTATE?\r\n	查询服务状态	0 1000
<input type="checkbox"/>	AT+QMTOPEN=0,"47.92.146.210",1883\r\n	MQTT TEST	0 1000
<input type="checkbox"/>	AT+CGSN=1\r\n	FIND IMEI	0 1000
<input type="checkbox"/>	AT+QMTCONN=0,"863409053773569"\r\n	接客户端至 MQTT 服务	0 1000
<input type="checkbox"/>	AT+QMTPUB=0,0,0,0,"fudingyi","2018080405"\r\n	MQTT Publish	0 1000
<input type="checkbox"/>	AT+QMTSUB=0,1,"fudingyi",0\r\n	MQTT Subscribe	0 1000
<input type="checkbox"/>	AT+QMTUNS=0,2,"fudingyi"\r\n	退订fudingyi	0 1000
<input type="checkbox"/>	AT+QMTSUB=0,1,"linR",0\r\n	Subscribe linR	0 1000
<input type="checkbox"/>	AT+QMTPUB=0,0,0,0,"linR","Welcome to linR's channel"\r\n	linR's publish	0 1000
<input type="checkbox"/>	AT+QMTUNS=0,2,"linR"\r\n	退订linR	0 1000
<input type="checkbox"/>		=	0 1000
<input type="checkbox"/>		21无注释	0 1000
<input type="checkbox"/>		22无注释	0 1000
<input type="checkbox"/>		23无注释	0 1000
<input type="checkbox"/>		24无注释	0 1000
<input type="checkbox"/>		25无注释	0 1000
<input type="checkbox"/>		26无注释	0 1000
<input type="checkbox"/>		27无注释	0 1000
<input type="checkbox"/>		28无注释	0 1000
<input type="checkbox"/>		29无注释	0 1000
<input type="checkbox"/>		30无注释	0 1000

5. 利用云服务完成BC26之间的通信

OK

CLOSE OK

[15:50:37.412]发→◇AT+QMTOPEN=0,"123.60.209.34",1883

□

[15:50:37.421]收←◆AT+QMTOPEN=0,"123.60.209.34",1883

OK

[15:50:37.904]收←◆

+QMTOPEN: 0,0

[15:50:43.608]发→◇AT+QMTCONN=0,"863409053773569"

□

[15:50:43.618]收←◆AT+QMTCONN=0,"863409053773569"

OK

[15:50:44.418]收←◆

+QMTCONN: 0,0,0

[15:50:52.811]发→◇AT+QMTPUB=0,0,0,0,"qwerty","4"

□

[15:50:52.821]收←◆AT+QMTPUB=0,0,0,0,"qwerty","4"

OK

+QMTPUB: 0,0,0

[15:51:46.924]发→◇AT+QMTSUB=0,1,"qwerty",0

□

[15:51:46.933]收←◆AT+QMTSUB=0,1,"qwerty",0

[15:51:46.956]收←◆

OK

[15:51:48.084]收←◆

+QMTSUB: 0,1,0,0

[15:51:50.403]发→◇AT+QMTPUB=0,0,0,0,"qwerty","4"

□

[15:51:50.413]收←◆AT+QMTPUB=0,0,0,0,"qwerty","4"

OK

+QMTPUB: 0,0,0

[15:51:50.944]收←◆

+QMTRECV: 0,0,"qwerty","4"

26与
设备
后将今

SSCOM V5.13.1 串口/网络数据调试器,作者:大虾丁丁,2618058@qq.com. QQ群: 525024

通讯端口 串口设置 显示 发送 多字符串 小工具 帮助 联系作者

```

[16:44:16.308]收←◆
+QMTRECV: 0,0,"qwerty","2018080405"

[16:44:21.864]发→◇13 00 FF 88□
[16:44:21.870]收←◆13 00 FF 88
[16:44:22.865]发→◇AT+CIMI
□
[16:44:22.872]收←◆AT+CIMI
ERROR

[16:44:23.864]发→◇AT
□
[16:44:23.869]收←◆AT
OK

[16:44:27.440]发→◇AT+QMT PUB=0,0,0,0,"qwerty","2018080405"
□
[16:44:27.449]收←◆AT+QMT PUB=0,0,0,0,"qwerty","2018080405"
OK

+QMT PUB: 0,0,0

[16:44:28.587]收←◆
+QMTRECV: 0,0,"qwerty","2018080405"

[16:44:33.151]发→◇AT+QMT PUB=0,0,0,0,"qwerty","2018080405"
□
[16:44:33.161]收←◆AT+QMT PUB=0,0,0,0,"qwerty","2018080405"
OK

+QMT PUB: 0,0,0

[16:44:34.731]收←◆
+QMTRECV: 0,0,"qwerty","2018080405"

[16:45:01.163]发→◇AT+QMT PUB=0,0,0,0,"qwerty","2018080405"
□
[16:45:01.173]收←◆AT+QMT PUB=0,0,0,0,"qwerty","2018080405"
OK

+QMT PUB: 0,0,0

[16:45:04.229]收←◆
+QMTRECV: 0,0,"qwerty","2018080405"

```

清除窗口 打开文件 发送文件 停止 清发

端口号 COM3 USB-SERIAL CH340 ☐ HEX显示 保存数据 ☐ 接收数据到文件 ☐ HEX为

☒ 加时间戳和分包显示 超时时间: 20 ms 第1 字

☐ RTS ☒ DTR 波特率: 115200 AT+QMT SUB=0,1,"qwerty",0\r\n

qwerty

06-07-2021 16:44:33.60273383

2018080405

6. 华为IoTDA

1. 向导式极速体验



产品定义

产品描述了一款物联网设备的属性和功能，本次体验以一款典型的烟感器作演示。产品模型已经定义好，您可以在后续正式使用时修改。

属性列表

一些产品从传感器来的参数，和一个可写参数alarm

属性名称	数据类型	是否必选	访问方式	描述
alarm	int(整型)	是	可读，可写	是否有烟雾报警，0没有报警，1有报警
smokeConcentration	decimal(小数)	是	可读	烟雾浓度值
temperature	decimal(小数)	是	可读	温度值
humidity	int(整型)	是	可读	湿度值

命令列表

命令名称	下发参数	响应参数	描述
setReportingFrequency	value	无	设置设备数据上报频率

可以在最后由用户下发的命令

所属资源空间 DefaultApp_60e4gge0 ?

* 产品名称

mySmokeDetector

用户自行填写

* 设备类型 ?

smokeDetector

本设备名称，自己填写

注册设备

设备是某类产品的具体实现，任何物理设备都需要在平台注册后才能实现设备接入。这里我们基于上一步定义的产品，注册一个烟感设备。

* 所属产品 mySmokeDetector 和前面的产品名相同，已经填写好，用户不能再修改
产品已创建 2021/07/06 17:07:47 GMT+08:00

* 设备标识码 2018080405 填写设备标识码，需要唯一，例如IMEI码

* 设备名称 Device1 自己填写名称

选择设备演示包

本次体验将为您提供定制的设备演示包，模拟设备上报数据。请您根据需要选择：

设备平台



Windows



Linux

设备连接协议

MQTT

开发语言

C#

Java

嵌入式C

使用条件： 请确保您使用的是Windows 7及以上版本的操作系统

- 执行huaweicloud-iot-device-quickstart.exe，查看设备状态的变化（请务必执行解压操作，直接执行bat文件执行将会出错）

设备名称 Device1

设备状态 离线 刷新

自己动手

1. 修改MQTT服务器的地址，为教程的服务器，完成：PC26与MQTT连接，自己的PC26与伙伴PC26的连接，保留数据。
2. 注册华为IoTDA账号，完成向设备快速接入，使用默认设备。
3. 注册属于您的设备，创建设备，享受30天的使用，然后得令无限制使用，实现您的设备接入。

```
connection.PublishMessage:0 - publish message topic = $oc/devices/60e41d63f4b59002867b210c_2018080405/sys/properties/report, msg = [{"service_id":"smokeDetector", "properties":{"temperature":85.62801, "humidity":39, "smokeConcentration":12.45597}, "eventTime":"20210706T091008Z"}]
2021-07-06 17:10:09.4378 | Info | IoT.SDK.Device.Transport.Mqtt.MqttConnection.PublishMessage:0 - publish message topic = $oc/devices/60e41d63f4b59002867b210c_2018080405/sys/properties/report, msg = [{"service_id":"smokeDetector", "properties":{"temperature":34.55146, "humidity":98, "smokeConcentration":61.386246}, "eventTime":"20210706T091009Z"}]
2021-07-06 17:10:10.4257 | Info | IoT.SDK.Device.Transport.Mqtt.MqttConnection.PublishMessage:0 - publish message topic = $oc/devices/60e41d63f4b59002867b210c_2018080405/sys/properties/report, msg = [{"service_id":"smokeDetector", "properties":{"temperature":43.413017, "humidity":29, "smokeConcentration":63.055794}, "eventTime":"20210706T091010Z"}]
2021-07-06 17:10:11.4375 | Info | IoT.SDK.Device.Transport.Mqtt.MqttConnection.PublishMessage:0 - publish message topic = $oc/devices/60e41d63f4b59002867b210c_2018080405/sys/properties/report, msg = [{"service_id":"smokeDetector", "properties":{"temperature":37.066186, "humidity":1, "smokeConcentration":97.5864}, "eventTime":"20210706T091011Z"}]
2021-07-06 17:10:12.4351 | Info | IoT.SDK.Device.Transport.Mqtt.MqttConnection.PublishMessage:0 - publish message topic = $oc/devices/60e41d63f4b59002867b210c_2018080405/sys/properties/report, msg = [{"service_id":"smokeDetector", "properties":{"temperature":95.612015, "humidity":12, "smokeConcentration":92.069855}, "eventTime":"20210706T091012Z"}]
```

请下载设备演示包，然后在电脑上或者嵌入式设备中解压

下载设备演示包

用于模拟设备上报数据，包含您前面配置的产品信息...

执行huaweicloud-iot-device-quickstart.exe，查看设备状态的变化（请务必执行解压操作，直接执行bat文件执行将会出错）

设备名称	Device1	设备状态	在线	刷新
设备在线后，可通过下面的设备影子查看上报的属性数据				
属性	上报时间	上报值	描述	
<可读, 可写>alarm	2021/07/06 17:10:12 GMT+08:00		是否有烟雾报警, 0没有报警, 1有..	
<可读>smokeConcentration	2021/07/06 17:10:12 GMT+08:00	92.069855	烟雾浓度值	
<可读>temperature	2021/07/06 17:10:12 GMT+08:00	95.612015	温度值	
<可读>humidity	2021/07/06 17:10:12 GMT+08:00	12	湿度值	

您可以通过产品定义的命令，调整以上属性数据上报频率。上报频率默认为5秒/次，建议您调整为1秒/次查看上报频率变化

setReportingFrequency

1

下发命令

huaweicloud-iot_...zip

全部显示

在这里输入你要搜索的内容

17:10 2021/7/6

2.使用虚拟设备体验

创建产品



创建产品

★ 所属资源空间: DefaultApp_60e4gge0

如需创建新的资源空间, 您可[前往当前实例详情创建](#)

★ 产品名称: SmartLight

协议类型: MQTT

★ 数据格式: JSON 选择JSON可以不用编解码

★ 厂商名称: HUAWEI

所属行业: 无

★ 设备类型: SmartStreetLight

高级配置: 定制ProductID | 备注信息

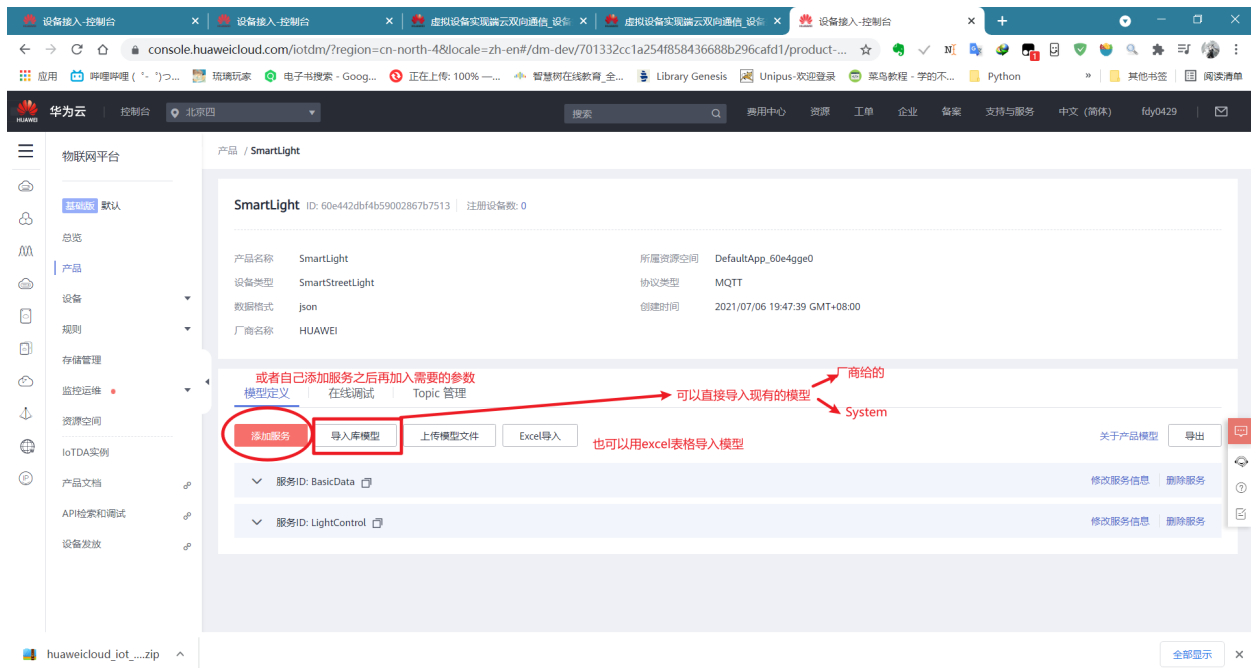
确定 取消

长度不超过32, 只允许中文、字母、数字、及_?#(),.&%@!-等字符的组合。



创建产品成功的窗口。

开发产品模型



服务ID: BasicData

修改服务信息 | 删除服务

服务类型: BasicData

服务描述: 路灯上报的数据

添加属性

属性名称	数据类型	访问方式	描述	操作
luminance	int(整型)	可读/可写		复制 修改 删除

10

总条数: 1 < 1 > 添加luminance参数，存储亮度

添加命令

命令名称	下发参数	响应参数	操作
------	------	------	----

服务ID: LightControl

修改服务信息 | 删除服务

服务类型: LightControl

服务描述:

添加属性

属性名称	数据类型	访问方式	描述	操作
------	------	------	----	----

无数据

添加命令

命令名称	下发参数	响应参数	操作
switch	value		复制 修改 删除

10

总条数: 1 < 1 > switch命令，可以控制灯的开关

修改命令

★ 命令名称

switch

下发参数

新增输入参数

参数名称	数据类型	描述	操作
value	string(字符串)		修改 删除

5

总条数: 1 < 1 >

响应参数

新增响应参数

参数名称	数据类型	描述	操作
------	------	----	----

暂无表格数据

确定

取消



修改参数

* 参数名称

value

参数描述

0/128

* 数据类型

string(字符串)

* 长度

15

枚举值

ON,OFF

路灯的开关由两种状态，
分别是开和关

6/1024

确定

取消

注册虚拟设备

使用在线调试功能，注册一个虚拟设备用于调试。



新增测试设备

设备类型

真实设备

虚拟设备

您正在注册一个虚拟的设备

确定

取消

模型定义 | 在线调试 | Topic 管理

新增测试设备

设备名称	设备标识码	设备ID	类型	操作
20210706T114739ZNo	1625572949767	60e442dbf4b59002867b7513_1625572949767	虚拟设备	调试 删除

虚拟设备名称包含Simulator字样

点此可进入调试界面

数据上报

应用模拟器

全部 数据接收 命令发送

数据接收: 2021/07/06 20:10:16 GMT+08:00
{serviceId: BasicData, data: {"luminance": "50"}}

查看上报结果

服务: BasicData

命令: lightControl

switch:

发送

命令下发

数据上报

IoT Platform

数据上报

命令下发

设备模拟器

全部 数据发送 命令接收

操作时间: 2021/07/06 20:10:15 GMT+08:00
发送消息body信息: { "luminance": "50" }

选择服务: BasicData

* luminance: 50

1 输入亮度数值

周期(秒): 5 ☐ 自动发送 发送

消息跟踪

消息跟踪

[平台收到设备的鉴权消息]clientId is : 60e442dbf4b59002867b7513_1625572949767_0_0_2021070620

2021/07/06 20:07:47 GMT+08:00

[设备在平台的鉴权成功, 设备身份合法]devicelid is : 60e442dbf4b59002867b7513_1625572949767

2021/07/06 20:09:16 GMT+08:00

[平台收到设备的鉴权消息]clientId is : 60e442dbf4b59002867b7513_1625572949767_0_0_2021070620

2021/07/06 20:09:16 GMT+08:00

[设备在平台的鉴权成功, 设备身份合法]devicelid is : 60e442dbf4b59002867b7513_1625572949767

2021/07/06 20:10:16 GMT+08:00

[平台收到设备的属性上报]devicelid is : 60e442dbf4b59002867b7513_1625572949767, topic is : \$oc/devices/60e442dbf4b59002867b7513_1625572949767/sys/properties/report, data toString is : {"services": [{"properties": {"luminance": "50"}, "service_id": "BasicData", "event_time": null}]}

2021/07/06 20:10:16 GMT+08:00

[属性上报设备影子刷新成功]shadow deal property report success ,current effective report: [{"serviceId": "BasicData", "properties": {"luminance": "50"}, "eventTime": "20210706T121016Z"}]

全部显示

命令下发



7. MQTT.fx



查看设备接入信息，记录域名和端口。

接入信息				
接入类型	接入协议（端口号）	接入地址		操作
应用接入	HTTPS (443)	a160e42dd0.iotda.cn-north-4.myhuaweicloud.com		
	AMQPS (5671)	a160e42dd0.iot-amqps.cn-north-4.myhuaweicloud.com		预置接入凭证
设备接入	CoAP (5683) CoAPS (5684)	a160e42dd0.iot-coaps.cn-north-4.myhuaweicloud.com		
	HTTPS (8943)	a160e42dd0.iot-https.cn-north-4.myhuaweicloud.com		
	MQTT (1883) MQTTS (8883)	a160e42dd0.iot-mqtt.cn-north-4.myhuaweicloud.com		

可以看到，我们已经有有了一个MQTT协议的产品，可以直接使用它而不必再次创建产品

连接鉴权

使用MQTT.fx工具接入物联网平台。

1. 安装；

2. 填写设备这侧后生成的设备ID (Device) 和密钥 (DeviceSecret) , 生成连接信息 (ClientID、Username、Password)

设备管理 / 设备详情

概述 | 命令 | 设备影子 | 消息跟踪 | 维护 | 子设备 | 标签

20210706T114739ZNoNBSimulator [🔗](#) ● 离线 ⓘ | 所属产品 SmartLight

所属资源空间	DefaultApp_60e4gge0	设备ID	60e442dbf4b59002867b7513_1625572949767 🔗
设备标识码	1625572949767 🔗	认证类型	密钥 重置密钥
注册时间	2021/07/06 20:02:30 GMT+08:00	节点类型	直连设备
固件版本	--	软件版本	--

最新上报数据 [查看历史数据 ⓘ](#)

luminance

50

<BasicData>

2021/07/06 20:10:16 GMT+08:...

Huaweicloud IoTDA Mqtt ClientId Generator!

这是由华为云设备接入提供的MQTT ClientId生成工具, 设备连接鉴权具体生成算法可以点击下方按钮了解更多

[了解更多](#)

DeviceId	设备ID
60e442dbf4b59002867b7513_1625572949767	
DeviceSecret	密钥
bb804910ef6c65c8c040	

[Generate](#)

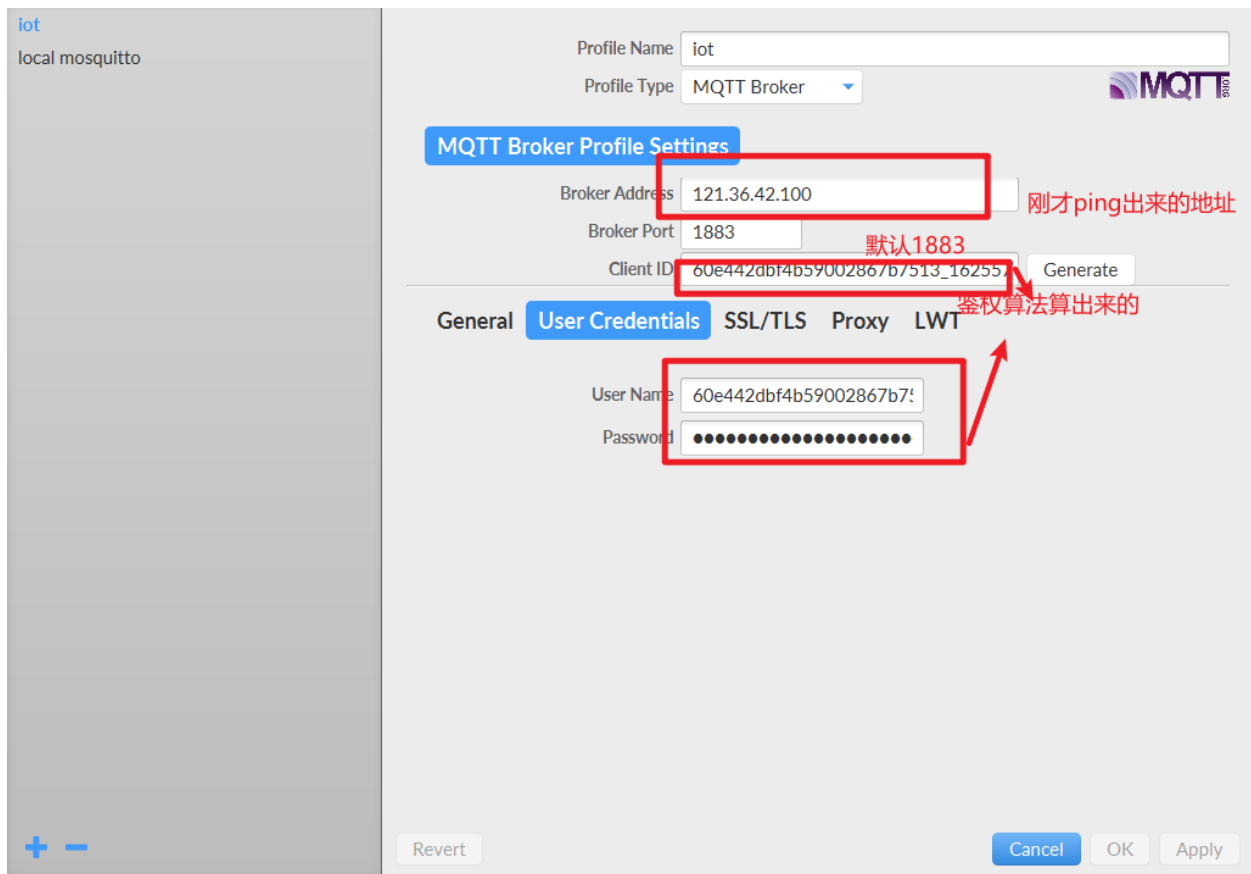
ClientId	60e442dbf4b59002867b7513_1625572949767_0_0_2021070612
Username	60e442dbf4b59002867b7513_1625572949767
Password	25c090f6e44c68baf7c774cb5e4694e6994544fe29d6992b8bb3e0cb5d550568

```
C:\> 选择C:\WINDOWS\system32\cmd.exe - ping a160e42dd0.iot-mqtts.cn-north-4.myhuaweicloud.com
Microsoft Windows [版本 10.0.18363.1440]
(c) 2019 Microsoft Corporation。保留所有权利。

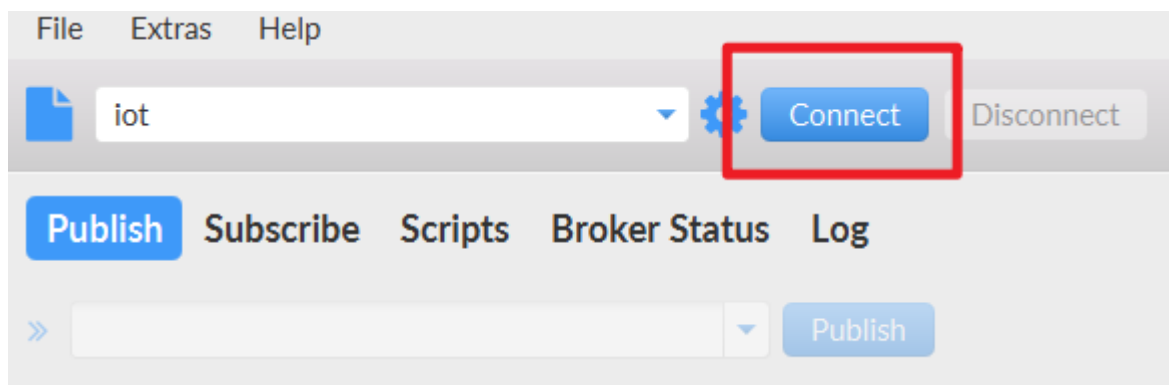
C:\Users\LIGHTBLUE>ping a160e42dd0.iot-mqtts.cn-north-4.myhuaweicloud.com
正在 Ping a160e42dd0.iot-mqtts.cn-north-4.myhuaweicloud.com [121.36.42.100] 具有 32 字节的数据:
来自 121.36.42.100 的回复: 字节=32 时间=32ms TTL=45
来自 121.36.42.100 的回复: 字节=32 时间=27ms TTL=45
来自 121.36.42.100 的回复: 字节=32 时间=28ms TTL=45
```

Broker address 121.36.42.100

Broker Port 1883



然后点击Apply。



点击connect连接。

设备接入服务，从这里开启物联网世界之路！

物联网平台为您提供多种平台服务实例，满足您不同的使用需要，您当前访问的是 **基础版实例**

不同协议设备/应用的接入地址
平台接入地址

了解设备接入典型场景操作流程
新手引导

带您上手玩转平台
快速体验

资源概览

产品数	2	资源空间数	1	数据转发规则	0
设备总数	2	在线设备总数	1	激活设备总数	2

可以再控制台中看到，在线设备总数从0->1。

上报数据

60e442dbf4b59002867b7513_1625572949767

\$oc/devices/60e442dbf4b59002867b7513_1625572949767/sys/properties/report

20210706T114739ZNoNBSim

所属资源空间 DefaultApp_60e4gged

设备标识码 1625572949767

注册时间 2021/07/06 20:02:30

固件版本 --

最新上报数据

luminance

100

<BasicData>

2021/07/06 20:55:52 GMT+08:00

原来是100

File Extras Help

iot Connect Disconnect

Publish Subscribe Scripts Broker Status Log

\$oc/devices/60e442dbf4b59002867b7513_1 Publish

Qo... Qo... Qo... Retained

```
[{"services":[{"properties":{"luminance":"200"}, "service_id":"BasicData","event_time":null}]}
```

上报数据200

20210706T114739ZNoNBSim

所属资源空间

DefaultApp_60e4gge

设备标识码

1625572949767

注册时间

2021/07/06 20:02:30

固件版本

--

最新上报数据

luminance

200

<BasicData>

2021/07/06 20:57:07 GMT+08:...

FileExtrasHelp

iot

ConnectDisconnect

PublishSubscribeScriptsBroker StatusLog

\$oc/devices/60e442dbf4b59002867b7513_1

Publish

{"services":[{"properties":{"luminance":"200"},"service_id":"BasicData","event_time":null}]}

点击Publish之后，亮度变为200，上报成功

如何查错

设备管理 / 设备详情

概述

命令

设备影子

消息跟踪

维护

子设备

标签

消息跟踪可记录设备在运行过程中的各类操作信息，状态和结果。当数据上报、响应命令等业务场景中出现故障时，消息跟踪功能可帮助您快速进行故障定位和原因分析。[了解更多](#)

修改跟踪配置

为避免占用读写计算、存储资源，同时保证数据有效，平台限制最长可跟踪3天的数据，且单个用户下，仅支持同时启动10个设备的消息跟踪。

跟踪类型

设备至平台

平台至设备

应用至平台

平台数据转发

执行情况

运行中

预计结束时间

2021/07/06 21:05:42 GMT+08:00

停止

清除数据

消息跟踪数据

所有消息状态

高级搜索

C

下

业务类型	业务步骤	业务详情	记录时间	消息状态	操作
设备至平台	属性上报设备影子刷新成功	shadow deal property report success ,current effective repo...	2021-07-06 20:57:07.821 GMT+08:00	成功	查看
设备至平台	平台收到设备的属性上报	deviceld is : 60e442dbf4b59002867b7513_1625572949767, ...	2021-07-06 20:57:07.760 GMT+08:00	成功	查看
设备至平台	属性上报设备影子刷新成功	shadow deal property report success ,current effective repo...	2021-07-06 20:55:52.932 GMT+08:00	成功	查看
设备至平台	平台收到设备的属性上报	deviceld is : 60e442dbf4b59002867b7513_1625572949767, ...	2021-07-06 20:55:52.867 GMT+08:00	成功	查看
设备至平台	上行数据异常	Please check whether the json format of the data you repo...	2021-07-06 20:54:08.942 GMT+08:00	失败	定位建议 查看
设备至平台	平台收到设备的属性上报	deviceld is : 60e442dbf4b59002867b7513_1625572949767, ...	2021-07-06 20:54:08.942 GMT+08:00	成功	查看
设备至平台	上行数据异常	Please check whether the json format of the data you repo...	2021-07-06 20:52:52.566 GMT+08:00	失败	定位建议 查看

比较重要的debug的error提示：看消息跟踪的“定位建议”，可以定位到哪里写错了

全部显示

8. 创建仓库，使用git向仓库push今天的作业