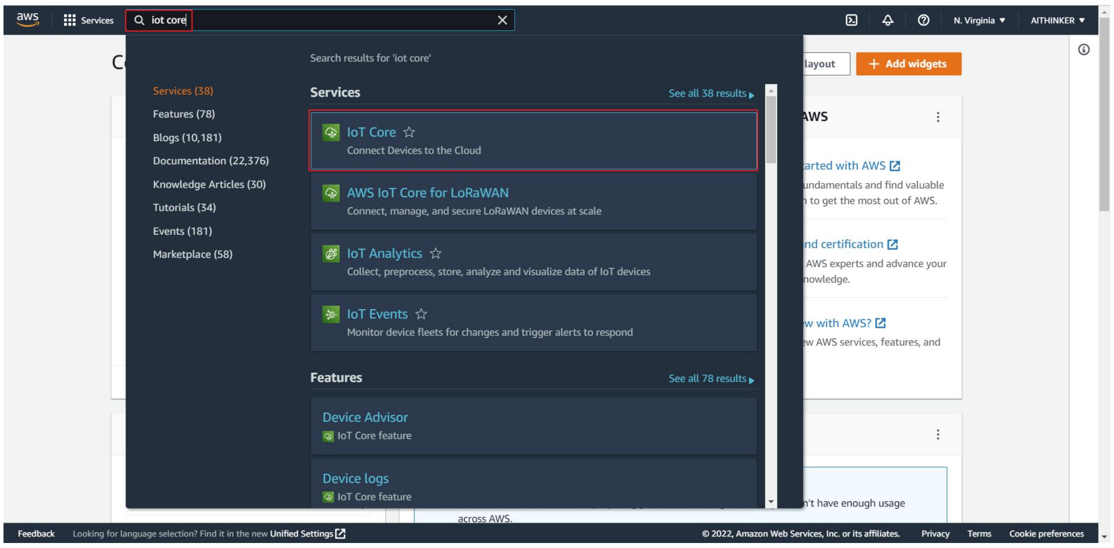
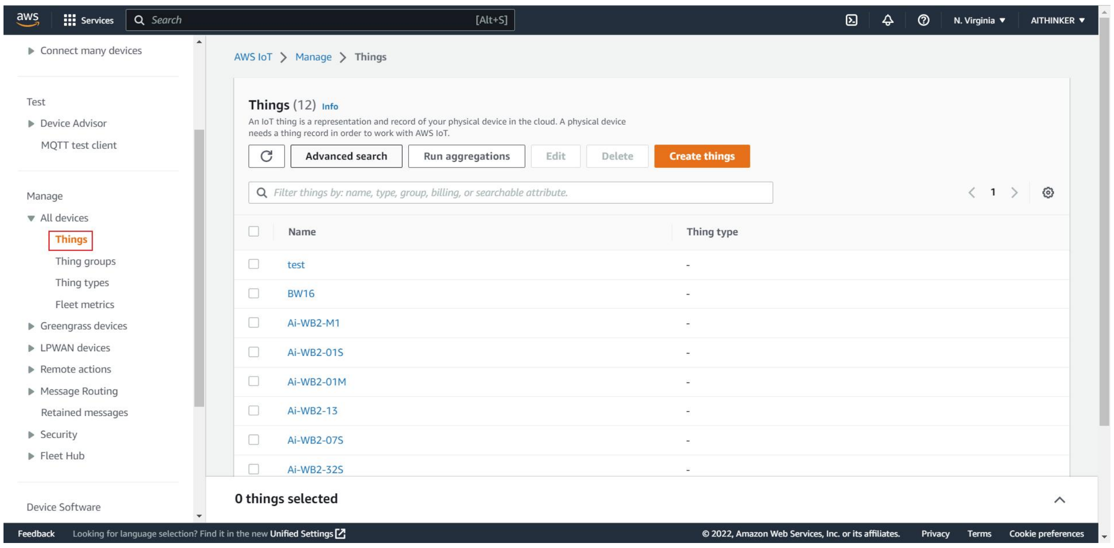
# 1. Create devices on the cloud platform

## 1.1 Enter Iot Core console

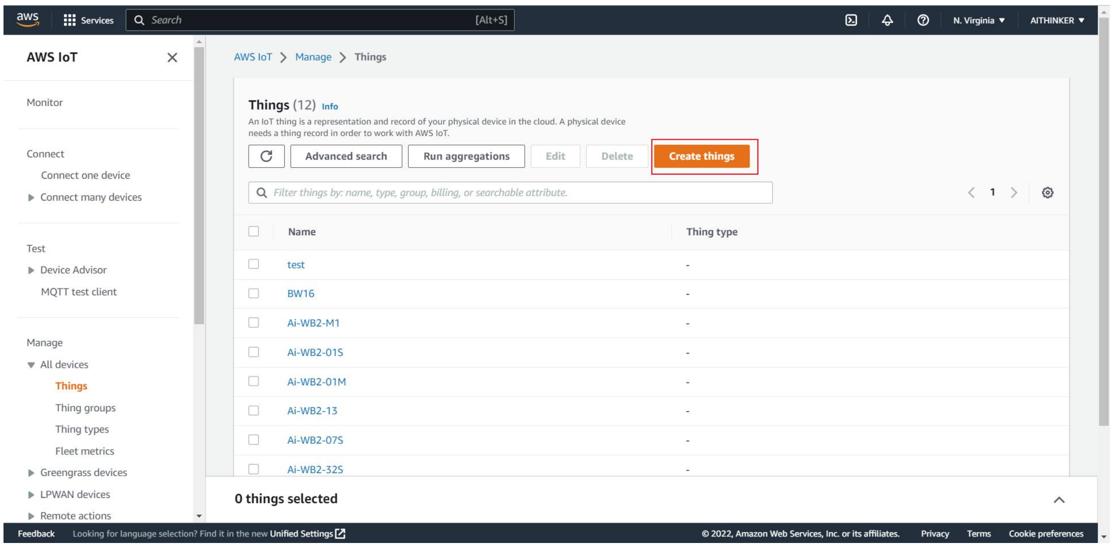


## 1.2 Click item to enter the Create item interface



| Name | Thing type |
|---|---|
| test | - |
| BW16 | - |
| Ai-WB2-M1 | - |
| Ai-WB2-01S | - |
| Ai-WB2-01M | - |
| Ai-WB2-13 | - |
| Ai-WB2-07S | - |
| Ai-WB2-32S | - |

## 1.3 Create things



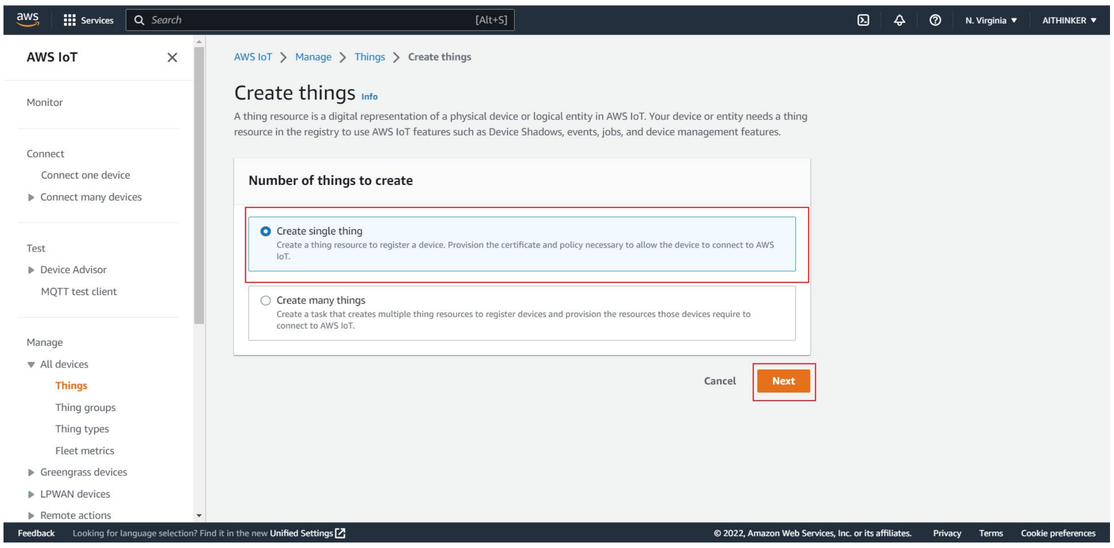
Things (12)

An IoT thing is a representation and record of your physical device in the cloud. A physical device needs a thing record in order to work with AWS IoT.

| Name | Thing type |
|---|---|
| test | - |
| BW16 | - |
| Ai-WB2-M1 | - |
| Ai-WB2-01S | - |
| Ai-WB2-01M | - |
| Ai-WB2-13 | - |
| Ai-WB2-07S | - |
| Ai-WB2-32S | - |

0 things selected

## 1.4 Create single thing



Create things

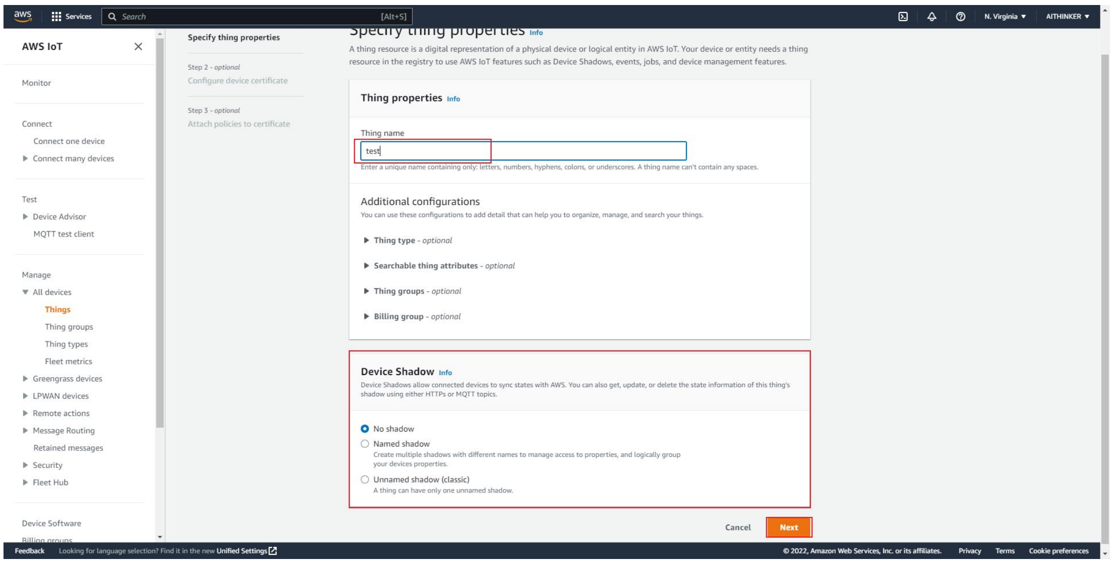
A thing resource is a digital representation of a physical device or logical entity in AWS IoT. Your device or entity needs a thing resource in the registry to use AWS IoT features such as Device Shadows, events, jobs, and device management features.

Number of things to create

[✓ Create single thing]
Create a thing resource to register a device. Provision the certificate and policy necessary to allow the device to connect to AWS IoT.

Create many things
Create a task that creates multiple thing resources to register devices and provision the resources those devices require to connect to AWS IoT.

## 1.5 Specify thing properties


Specify thing properties

A thing resource is a digital representation of a physical device or logical entity in AWS IoT. Your device or entity needs a thing resource in the registry to use AWS IoT features such as Device Shadows, events, jobs, and device management features.

**Thing properties**

Thing name: test

Enter a unique name containing only: letters, numbers, hyphens, colons, or underscores. A thing name can't contain any spaces.

Additional configurations

You can use these configurations to add detail that can help you to organize, manage, and search your things.

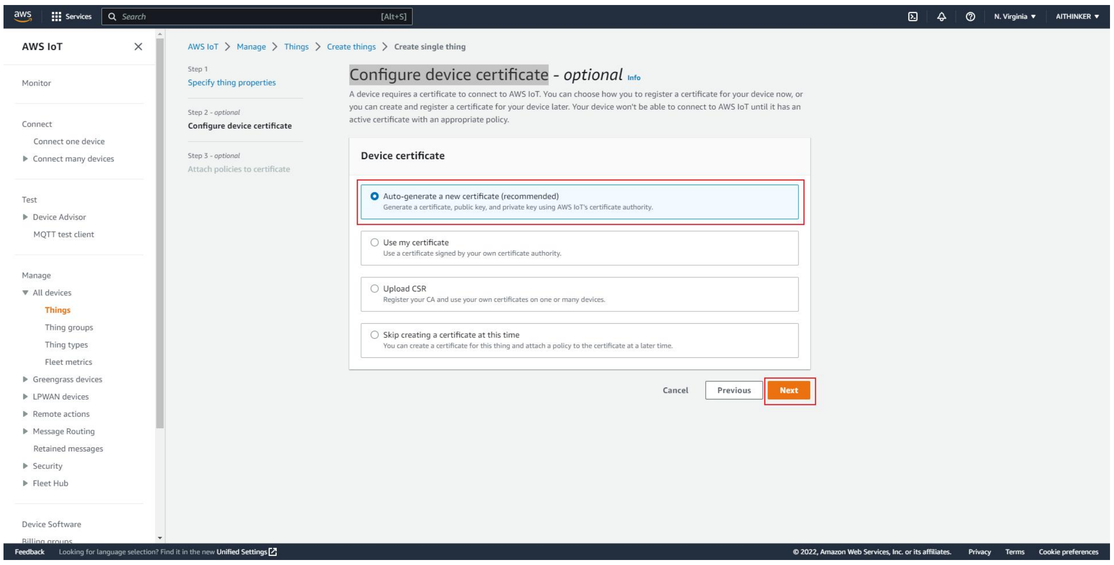
- Thing type - *optional*
- Searchable thing attributes - *optional*
- Thing groups - *optional*
- Billing group - *optional*

**Device Shadow**

Device Shadows allow connected devices to sync states with AWS. You can also get, update, or delete the state information of this thing's shadow using either HTTPs or MQTT topics.

- [✓ No shadow]
- Named shadow: Create multiple shadows with different names to manage access to properties, and logically group your devices properties.
- Unnamed shadow (classic): A thing can have only one unnamed shadow.

## 1.6 Configure device certificate


AWS IoT > Manage > Things > Create things > Create single thing

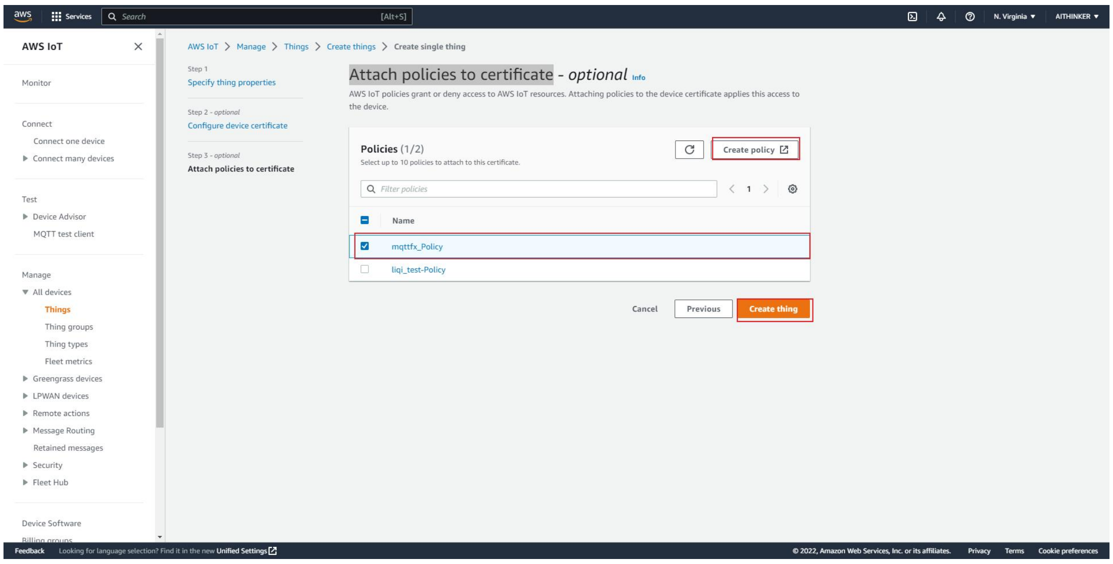
Configure device certificate - *optional*

A device requires a certificate to connect to AWS IoT. You can choose how you to register a certificate for your device now, or you can create and register a certificate for your device later. Your device won't be able to connect to AWS IoT until it has an active certificate with an appropriate policy.

**Device certificate**

- [✓ Auto-generate a new certificate (recommended)]: Generate a certificate, public key, and private key using AWS IoT's certificate authority.
- Use my certificate: Use a certificate signed by your own certificate authority.
- Upload CSR: Register your CA and use your own certificates on one or many devices.
- Skip creating a certificate at this time: You can create a certificate for this thing and attach a policy to the certificate at a later time.

## 1.7 Attach policies to certificate



Attach policies to certificate - *optional* Info

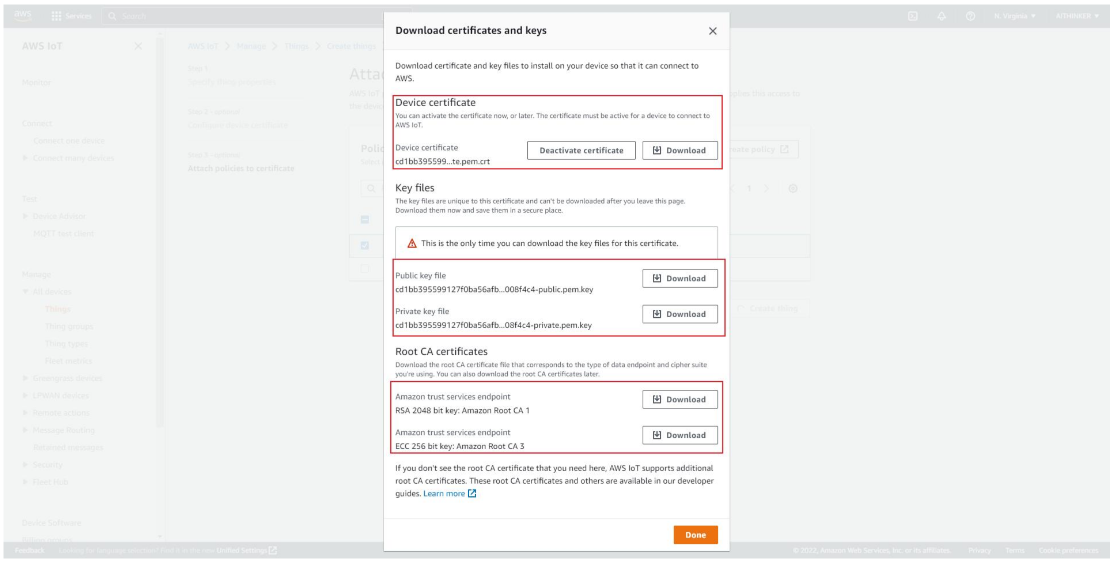
AWS IoT policies grant or deny access to AWS IoT resources. Attaching policies to the device certificate applies this access to the device.

Policies (1/2)

Select up to 10 policies to attach to this certificate.

| | Name |
|---|---|
| ✓ | mqttfx_Policy |
| | liqi_test-Policy |

## 1.8 Download and save the certificate



Download certificates and keys

Download certificate and key files to install on your device so that it can connect to AWS.

Device certificate

You can activate the certificate now, or later. The certificate must be active for a device to connect to AWS IoT.

Device certificate
cd1bb395599...te.pem.crt

Key files

The key files are unique to this certificate and can't be downloaded after you leave this page. Download them now and save them in a secure place.

This is the only time you can download the key files for this certificate.

Public key file
cd1bb395599127f0ba56afb...008f4c4-public.pem.key

Private key file
cd1bb395599127f0ba56afb...08f4c4-private.pem.key

Root CA certificates

Download the root CA certificate file that corresponds to the type of data endpoint and cipher suite you're using. You can also download the root CA certificates later.

Amazon trust services endpoint
RSA 2048 bit key: Amazon Root CA 1

Amazon trust services endpoint
ECC 256 bit key: Amazon Root CA 3

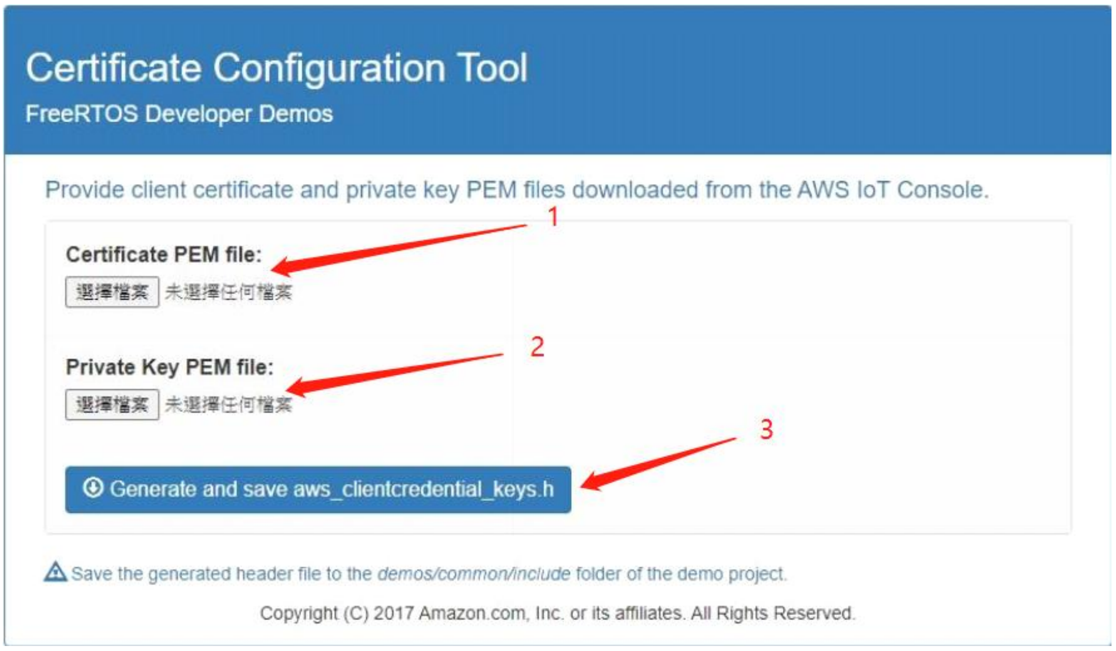
If you don't see the root CA certificate that you need here, AWS IoT supports additional root CA certificates. These root CA certificates and others are available in our developer guides. Learn more

# 2. SDK Amend

## 2.1 SDK certificate change

### 2.1.1 Generating Certificates

Use amazon-freertos/tools/certificate_configuration/CertificateConfigurator.html to generate aws_clientcredential_keys.h.



Certificate Configuration Tool

FreeRTOS Developer Demos

Provide client certificate and private key PEM files downloaded from the AWS IoT Console.

Certificate PEM file:

選擇檔案 未選擇任何檔案

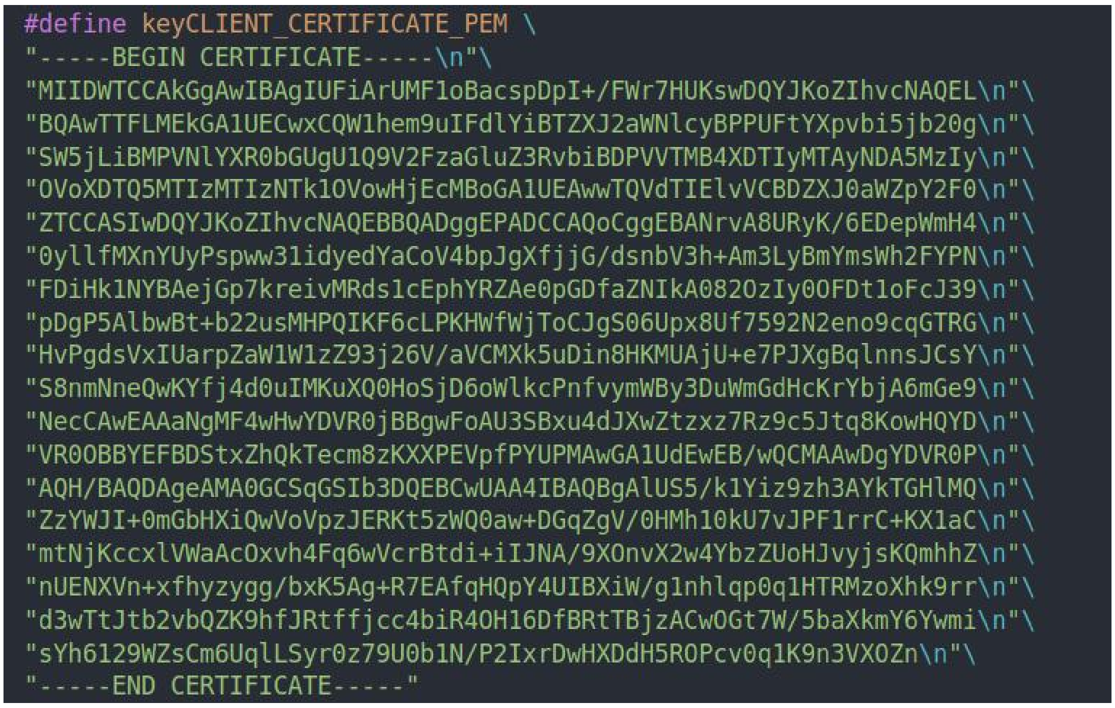
Private Key PEM file:

選擇檔案 未選擇任何檔案

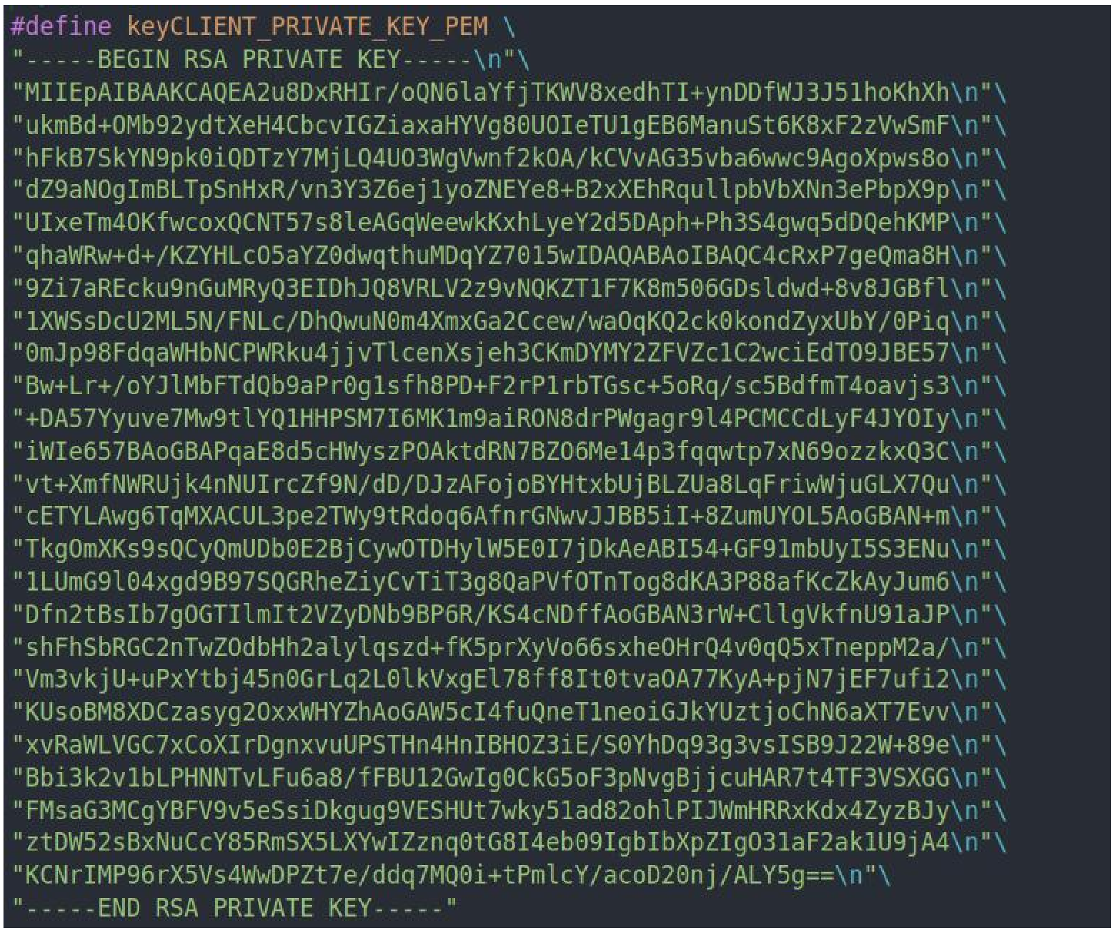
Save the generated header file to the *demos/common/include* folder of the demo project.

Copyright (C) 2017 Amazon.com, Inc. or its affiliates. All Rights Reserved.

## 2.1.2 Replacing a Certificate File



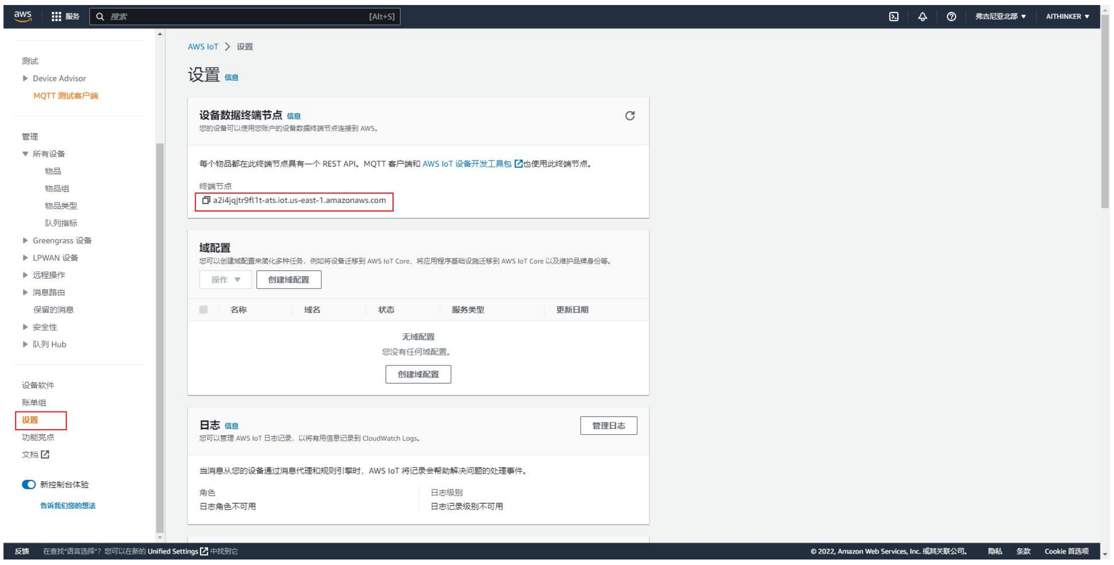
```
#define keyCLIENT_CERTIFICATE_PEM \
"-----BEGIN CERTIFICATE-----\n"\
"MIIDWTCCAkGgAwIBAgIUFiArUMF1oBacspDpI+/FWr7HUKswDQYJKoZIhvcNAQEL\n"\
"BQAwTTFLMEkGA1UECwxCQW1hem9uIFdlYiBTZXJ2aWNlcyBPPUFtYXpvbi5jb20g\n"\
"SW5jLiBMPVNlYXR0bGUgU1Q9V2FzaGluZ3RvbiBDPVVTMB4XDTIyMTAyNDA5MzIy\n"\
"OVoXDTQ5MTIzMTIzNTk1OVowHjEcMBoGA1UEAwwTQVdTIElvVCBDZXJ0aWZpY2F0\n"\
"ZTCCASIwDQYJKoZIhvcNAQEBBQADggEPADCCAQoCggEBANrvA8URyK/6EDepWmH4\n"\
"0yllfMXnYUyPspww31idyedYaCoV4bpJgXfjjG/dsnbV3h+Am3LyBmYmsWh2FYPN\n"\
"FDiHk1NYBAejGp7kreivMRds1cEphYRZAe0pGDfaZNIkA082OzIy0OFDt1oFcJ39\n"\
"pDgP5AlbwBt+b22usMHPQIKF6cLPKHWfWjToCJgS06Upx8Uf7592N2eno9cqGTRG\n"\
"HvPgdsVxIUarpZaW1W1zZ93j26V/aVCMXk5uDin8HKMUAjU+e7PJXgBqlnnsJCsY\n"\
"S8nmNneQwKYfj4d0uIMKuXQ0HoSjD6oWlkcPnfvymWBy3DuWmGdHcKrYbjA6mGe9\n"\
"NecCAwEAAaNgMF4wHwYDVR0jBBgwFoAU3SBxu4dJXwZtzxz7Rz9c5Jtq8KowHQYD\n"\
"VR0OBBYEFBDStxZhQkTecm8zKXXPEVpfPYUPMAwGA1UdEwEB/wQCMAAwDgYDVR0P\n"\
"AQH/BAQDAgeAMA0GCSqGSIb3DQEBCwUAA4IBAQBgAlUS5/k1Yiz9zh3AYkTGHlMQ\n"\
"ZzYWJI+0mGbHXiQwVoVpzJERKt5zWQ0aw+DGqZgV/0HMh10kU7vJPF1rrC+KX1aC\n"\
"mtNjKccxlVWaAcOxvh4Fq6wVcrBtdi+iIJNA/9XOnvX2w4YbzZUoHJvyjsKQmhhZ\n"\
"nUENXVn+xfhyzygg/bxK5Ag+R7EAfqHQpY4UIBXiW/g1nhlqp0q1HTRMzoXhk9rr\n"\
"d3wTtJtb2vbQZK9hfJRtffjcc4biR4OH16DfBRtTBjzACwOGt7W/5baXkmY6Ywmi\n"\
"sYh6129WZsCm6UqlLSyr0z79U0b1N/P2IxrDwHXDdH5ROPcv0q1K9n3VXOZn\n"\
"-----END CERTIFICATE-----"
```



```
#define keyCLIENT_PRIVATE_KEY_PEM \
"-----BEGIN RSA PRIVATE KEY-----\n"\
"MIIEpAIBAAKCAQEA2u8DxRHIr/oQN6laYfjTKWV8xedhTI+ynDDfWJ3J51hoKhXh\n"\
"ukmBd+OMb92ydtXeH4CbcvIGZiaxaHYVg80UOIeTU1gEB6ManuSt6K8xF2zVwSmF\n"\
"hFkB7SkYN9pk0iQDTzY7MjLQ4UO3WgVwnf2kOA/kCVvAG35vba6wwc9AgoXpws8o\n"\
"dZ9aNOgImBLTpSnHxR/vn3Y3Z6ej1yoZNEYe8+B2xXEhRqullpbVbXNn3ePbpX9p\n"\
"UIxeTm4OKfwcoxQCNT57s8leAGqWeewkKxhLyeY2d5DAph+Ph3S4gwq5dDQehKMP\n"\
"qhaWRw+d+/KZYHLcO5aYZ0dwqthuMDqYZ7015wIDAQABAoIBAQC4cRxP7geQma8H\n"\
"9Zi7aREcku9nGuMRyQ3EIDhJQ8VRLV2z9vNQKZT1F7K8m506GDsldwd+8v8JGBfl\n"\
"1XWSsDcU2ML5N/FNLc/DhQwuN0m4XmxGa2Ccew/waOqKQ2ck0kondZyxUbY/0Piq\n"\
"0mJp98FdqaWHbNCPWRku4jjvTlcenXsjeh3CKmDYMY2ZFVZc1C2wciEdTO9JBE57\n"\
"Bw+Lr+/oYJlMbFTdQb9aPr0g1sfh8PD+F2rP1rbTGsc+5oRq/sc5BdfmT4oavjs3\n"\
"+DA57Yyuve7Mw9tlYQ1HHPSM7I6MK1m9aiRON8drPWgagr9l4PCMCCdLyF4JYOIy\n"\
"iWIe657BAoGBAPqaE8d5cHWyszPOAktdRN7BZO6Me14p3fqqwtp7xN69ozzkxQ3C\n"\
"vt+XmfNWRUjk4nNUIrcZf9N/dD/DJzAFojoBYHtxbUjBLZUa8LqFriwWjuGLX7Qu\n"\
"cETYLAwg6TqMXACUL3pe2TWy9tRdoq6AfnrGNwvJJBB5iI+8ZumUYOL5AoGBAN+m\n"\
"TkgOmXKs9sQCyQmUDb0E2BjCywOTDHylW5E0I7jDkAeABI54+GF91mbUyI5S3ENu\n"\
"1LUmG9l04xgd9B97SQGRheZiyCvTiT3g8QaPVfOTnTog8dKA3P88afKcZkAyJum6\n"\
"Dfn2tBsIb7gOGTIlmIt2VZyDNb9BP6R/KS4cNDffAoGBAN3rW+CllgVkfnU91aJP\n"\
"shFhSbRGC2nTwZOdbHh2alylqszd+fK5prXyVo66sxheOHrQ4v0qQ5xTneppM2a/\n"\
"Vm3vkjU+uPxYtbj45n0GrLq2L0lkVxgEl78ff8It0tvaOA77KyA+pjN7jEF7ufi2\n"\
"KUsoBM8XDCzasyg2OxxWHYZhAoGAW5cI4fuQneT1neoiGJkYUztjoChN6aXT7Evv\n"\
"xvRaWLVGC7xCoXIrDgnxvuUPSTHn4HnIBHOZ3iE/S0YhDq93g3vsISB9J22W+89e\n"\
"Bbi3k2v1bLPHNNTvLFu6a8/fFBU12GwIg0CkG5oF3pNvgBjjcuHAR7t4TF3VSXGG\n"\
"FMsaG3MCgYBFV9v5eSsiDkgug9VESHUt7wky51ad82ohlPIJWmHRRxKdx4ZyzBJy\n"\
"ztDW52sBxNuCcY85RmSX5LXYwIZznq0tG8I4eb09IgbIbXpZIgO31aF2ak1U9jA4\n"\
"KCNrIMP96rX5Vs4WwDPZt7e/ddq7MQ0i+tPmlcY/acoD20nj/ALY5g==\n"\
"-----END RSA PRIVATE KEY-----"
```

## 2.2 Replace MQTT addr and device name

### 2.2.1 Setting the Terminal Node



设置 信息

设备数据终端节点 信息

您的设备可以使用您账户的设备数据终端节点连接到 AWS。

每个物品都在此终端节点具有一个 REST API。MQTT 客户端和 AWS IoT 设备开发工具包也使用此终端节点。

终端节点
a2i4jqjtr9fl1t-ats.iot.us-east-1.amazonaws.com

域配置

您可以创建域配置来简化多种任务，例如将设备迁移到 AWS IoT Core、将应用程序基础设施迁移到 AWS IoT Core 以及维护品牌身份等。

| 名称 | 域名 | 状态 | 服务类型 | 更新日期 |
|---|---|---|---|---|

无域配置
您没有任何域配置。

日志 信息

您可以管理 AWS IoT 日志记录，以将有用信息记录到 CloudWatch Logs。

当消息从您的设备通过消息代理和规则引擎时，AWS IoT 将记录会帮助解决问题的处理事件。

角色
日志角色不可用

日志级别
日志记录级别不可用

在查找"语言选择"？您可以在新的 Unified Settings 中找到它

© 2022, Amazon Web Services, Inc. 或其关联公司。

### 2.2.2 Change the path and device name

```
#define clientcredentialMQTT_BROKER_ENDPOINT        "tbnn221tbgfpu.deviceadvisor.iot.us-east-1.amazonaws.com"

/*
 * @brief Host name.
 *
 * @todo Set this to the unique name of your IoT Thing.
 * Please note that for convenience of demonstration only we
 * are using a #define here. In production scenarios the thing
 * name can be something unique to the device that can be read
 * by software, such as a production serial number, rather
 * than a hard coded constant.
 */
#define clientcredentialIOT_THING_NAME              "BW16"
```

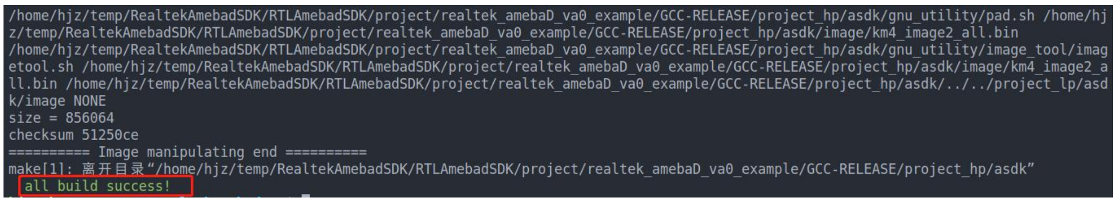
### 2.2.3 Change Wifi SSID and PASSWORD

```
#define clientcredentialWIFI_SSID                   "TEST1"

/*
 * @brief Password needed to join Wi-Fi network.
 * @todo If you are using WPA, set this to your network password.
 */
#define clientcredentialWIFI_PASSWORD               "123456789"
```

# 3. SDK compile and burn

## 3.1 SDK compile

Compile by executing ./make_all.sh.



```
/home/hjz/temp/RealtekAmebadSDK/RTLAmebadSDK/project/realtek_amebaD_va0_example/GCC-RELEASE/project_hp/asdk/gnu_utility/pad.sh /home/hj
z/temp/RealtekAmebadSDK/RTLAmebadSDK/project/realtek_amebaD_va0_example/GCC-RELEASE/project_hp/asdk/image/km4_image2_all.bin
/home/hjz/temp/RealtekAmebadSDK/RTLAmebadSDK/project/realtek_amebaD_va0_example/GCC-RELEASE/project_hp/asdk/gnu_utility/image_tool/imag
etool.sh /home/hjz/temp/RealtekAmebadSDK/RTLAmebadSDK/project/realtek_amebaD_va0_example/GCC-RELEASE/project_hp/asdk/image/km4_image2_a
ll.bin /home/hjz/temp/RealtekAmebadSDK/RTLAmebadSDK/project/realtek_amebaD_va0_example/GCC-RELEASE/project_hp/asdk/../../project_lp/asd
k/image NONE
size = 856064
checksum 51250ce
========== Image manipulating end ==========
make[1]: 离开目录“/home/hjz/temp/RealtekAmebadSDK/RTLAmebadSDK/project/realtek_amebaD_va0_example/GCC-RELEASE/project_hp/asdk”
 all build success!
```

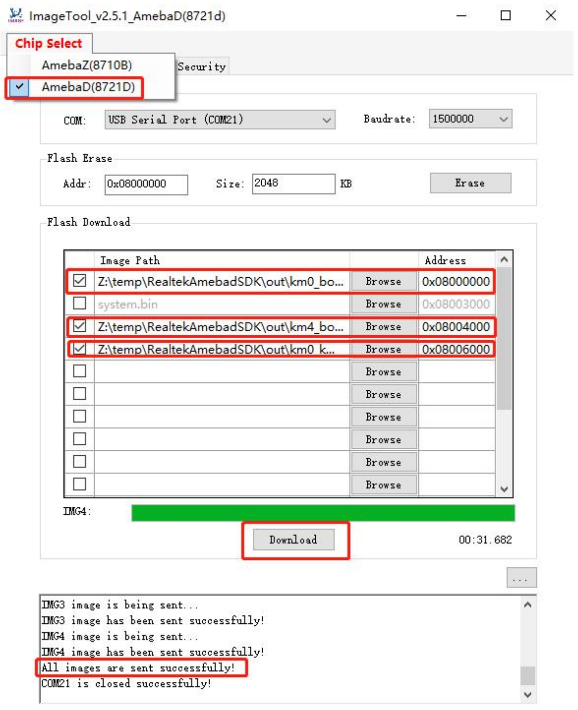
## 3.2 Burning firmware

### 3.2.1 Enter the burning mode

The boot pin is pulled down and the module is reset to enter the burning mode.

### 3.2.2 Burn

Select the chip model, select the relevant firmware, and click Download. If the output All image are send successfully, it indicates that the burning is successful.



ImageTool_v2.5.1_AmebaD(8721d)

Chip Select: AmebaZ(8710B) [✓ AmebaD(8721D)]

COM: USB Serial Port (COM21)    Baudrate: 1500000

Flash Erase — Addr: 0x08000000    Size: 2048 KB    Erase

Flash Download

| | Image Path | | Address |
|---|---|---|---|
| ☑ | Z:\temp\RealtekAmebadSDK\out\km0_bo... | Browse | 0x08000000 |
| ☐ | system.bin | Browse | 0x08003000 |
| ☑ | Z:\temp\RealtekAmebadSDK\out\km4_bo... | Browse | 0x08004000 |
| ☑ | Z:\temp\RealtekAmebadSDK\out\km0_k... | Browse | 0x08006000 |
| ☐ | | Browse | |
| ☐ | | Browse | |
| ☐ | | Browse | |
| ☐ | | Browse | |
| ☐ | | Browse | |
| ☐ | | Browse | |

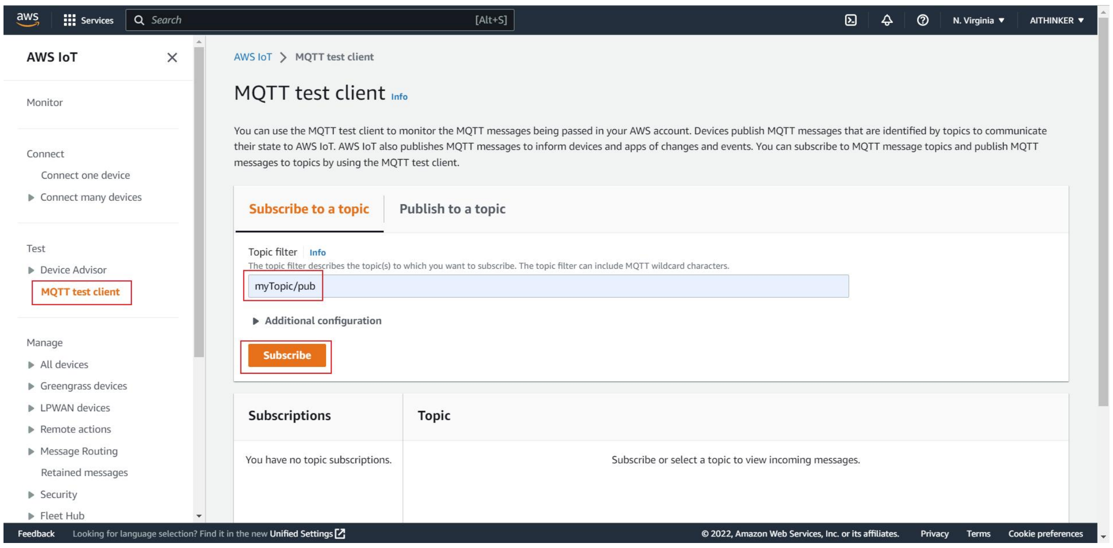
IMG4:    Download    00:31.682

```
IMG3 image is being sent...
IMG3 image has been sent successfully!
IMG4 image is being sent...
IMG4 image has been sent successfully!
All images are sent successfully!
COM21 is closed successfully!
```

# 4. Cloud connection test

## 4.1 AWS IOT Subscribe to the test

### 4.1.1 Subscribe to BW16/example/topic topics on the AWS IOT console



AWS IoT > MQTT test client

MQTT test client Info

You can use the MQTT test client to monitor the MQTT messages being passed in your AWS account. Devices publish MQTT messages that are identified by topics to communicate their state to AWS IoT. AWS IoT also publishes MQTT messages to inform devices and apps of changes and events. You can subscribe to MQTT message topics and publish MQTT messages to topics by using the MQTT test client.

Topic filter Info

The topic filter describes the topic(s) to which you want to subscribe. The topic filter can include MQTT wildcard characters.

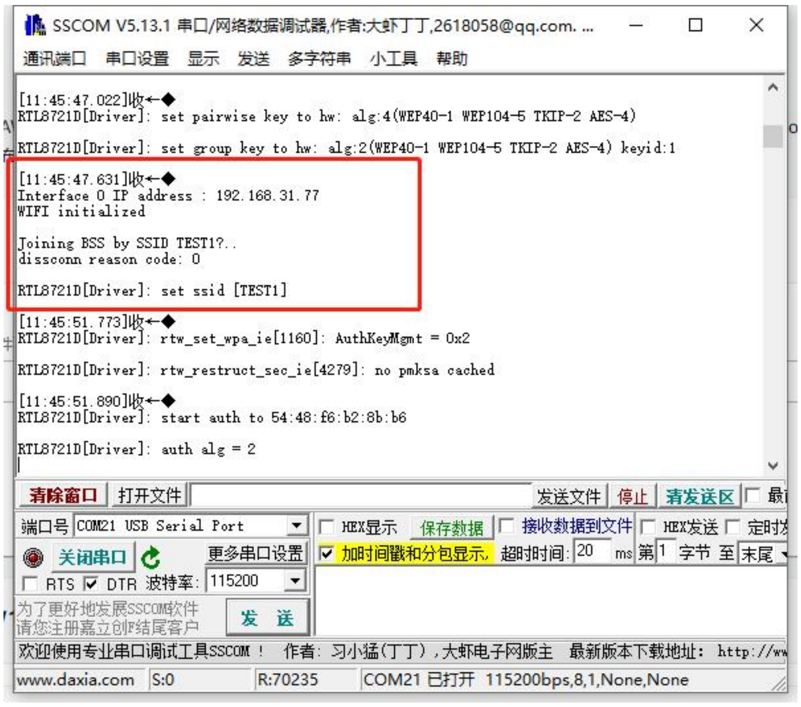
myTopic/pub

You have no topic subscriptions.

Subscribe or select a topic to view incoming messages.

## 4.2 Reset the device and view the log output

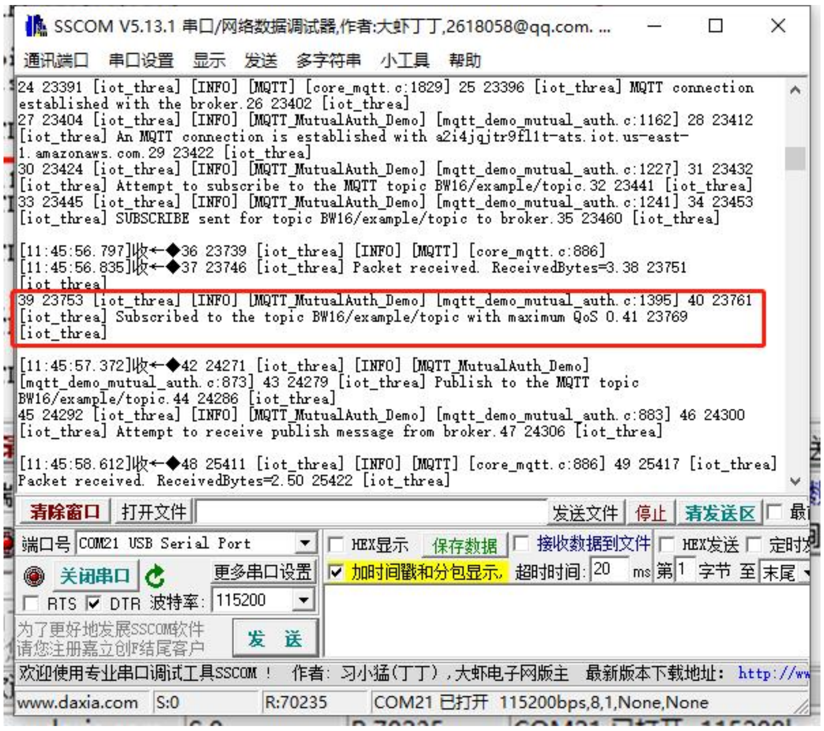
### 4.2.1 Connect with wifi



```
[11:45:47.022]收←◆
RTL8721D[Driver]: set pairwise key to hw: alg:4(WEP40-1 WEP104-5 TKIP-2 AES-4)

RTL8721D[Driver]: set group key to hw: alg:2(WEP40-1 WEP104-5 TKIP-2 AES-4) keyid:1

[11:45:47.631]收←◆
Interface 0 IP address : 192.168.31.77
WIFI initialized

Joining BSS by SSID TEST1?..
dissconn reason code: 0

RTL8721D[Driver]: set ssid [TEST1]

[11:45:51.773]收←◆
RTL8721D[Driver]: rtw_set_wpa_ie[1160]: AuthKeyMgmt = 0x2

RTL8721D[Driver]: rtw_restruct_sec_ie[4279]: no pmksa cached

[11:45:51.890]收←◆
RTL8721D[Driver]: start auth to 54:48:f6:b2:8b:b6

RTL8721D[Driver]: auth alg = 2
```

SSCOM V5.13.1 串口/网络数据调试器,作者:大虾丁丁,2618058@qq.com. ...
通讯端口 串口设置 显示 发送 多字符串 小工具 帮助
清除窗口 打开文件 发送文件 停止 清发送区
端口号 COM21 USB Serial Port
HEX显示 保存数据 接收数据到文件 HEX发送
关闭串口 更多串口设置 加时间戳和分包显示, 超时时间: 20 ms 第 1 字节 至 末尾
RTS DTR 波特率: 115200
为了更好地发展SSCOM软件 请您注册嘉立创F结尾客户
发 送
欢迎使用专业串口调试工具SSCOM ! 作者: 习小猛(丁丁), 大虾电子网版主 最新版本下载地址: http://ww
www.daxia.com S:0 R:70235 COM21 已打开 115200bps,8,1,None,None

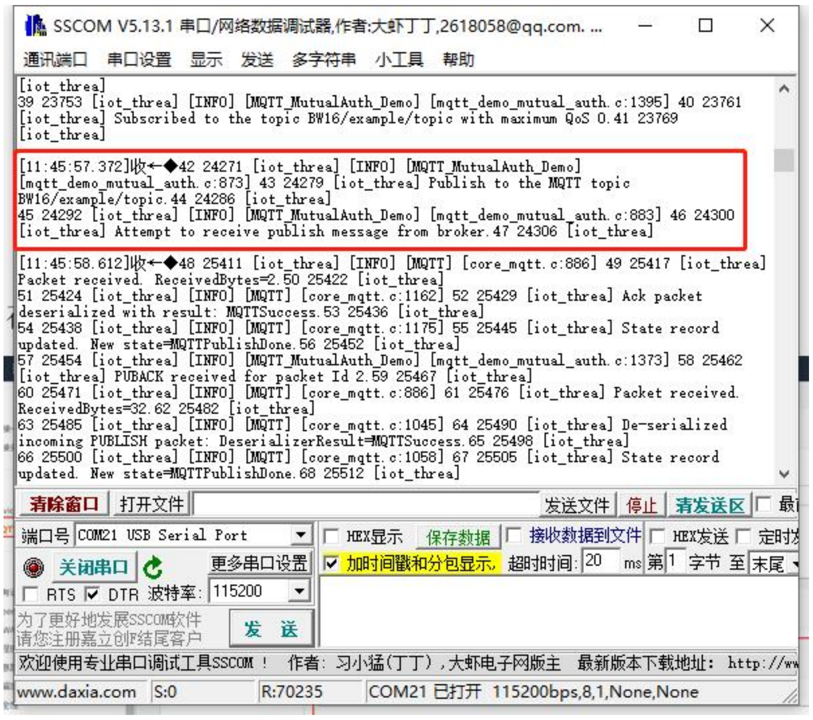
### 4.2.2 Subscribe to Topics



```
24 23391 [iot_threa] [INFO] [MQTT] [core_mqtt.c:1829] 25 23396 [iot_threa] MQTT connection
established with the broker.26 23402 [iot_threa]
27 23404 [iot_threa] [INFO] [MQTT_MutualAuth_Demo] [mqtt_demo_mutual_auth.c:1162] 28 23412
[iot_threa] An MQTT connection is established with a2i4jqjtr9fl1t-ats.iot.us-east-
1.amazonaws.com.29 23422 [iot_threa]
30 23424 [iot_threa] [INFO] [MQTT_MutualAuth_Demo] [mqtt_demo_mutual_auth.c:1227] 31 23432
[iot_threa] Attempt to subscribe to the MQTT topic BW16/example/topic.32 23441 [iot_threa]
33 23445 [iot_threa] [INFO] [MQTT_MutualAuth_Demo] [mqtt_demo_mutual_auth.c:1241] 34 23453
[iot_threa] SUBSCRIBE sent for topic BW16/example/topic to broker.35 23460 [iot_threa]

[11:45:56.797]收←◆36 23739 [iot_threa] [INFO] [MQTT] [core_mqtt.c:886]
[11:45:56.835]收←◆37 23746 [iot_threa] Packet received. ReceivedBytes=3.38 23751
[iot_threa]
39 23753 [iot_threa] [INFO] [MQTT_MutualAuth_Demo] [mqtt_demo_mutual_auth.c:1395] 40 23761
[iot_threa] Subscribed to the topic BW16/example/topic with maximum QoS 0.41 23769
[iot_threa]

[11:45:57.372]收←◆42 24271 [iot_threa] [INFO] [MQTT_MutualAuth_Demo]
[mqtt_demo_mutual_auth.c:873] 43 24279 [iot_threa] Publish to the MQTT topic
BW16/example/topic.44 24286 [iot_threa]
45 24292 [iot_threa] [INFO] [MQTT_MutualAuth_Demo] [mqtt_demo_mutual_auth.c:883] 46 24300
[iot_threa] Attempt to receive publish message from broker.47 24306 [iot_threa]

[11:45:58.612]收←◆48 25411 [iot_threa] [INFO] [MQTT] [core_mqtt.c:886] 49 25417 [iot_threa]
Packet received. ReceivedBytes=2.50 25422 [iot_threa]
```

SSCOM V5.13.1 串口/网络数据调试器,作者:大虾丁丁,2618058@qq.com. ...
通讯端口 串口设置 显示 发送 多字符串 小工具 帮助
清除窗口 打开文件 发送文件 停止 清发送区
端口号 COM21 USB Serial Port
HEX显示 保存数据 接收数据到文件 HEX发送
关闭串口 更多串口设置 加时间戳和分包显示, 超时时间: 20 ms 第 1 字节 至 末尾
RTS DTR 波特率: 115200
为了更好地发展SSCOM软件 请您注册嘉立创F结尾客户
发 送
欢迎使用专业串口调试工具SSCOM ! 作者: 习小猛(丁丁), 大虾电子网版主 最新版本下载地址: http://ww
www.daxia.com S:0 R:70235 COM21 已打开 115200bps,8,1,None,None

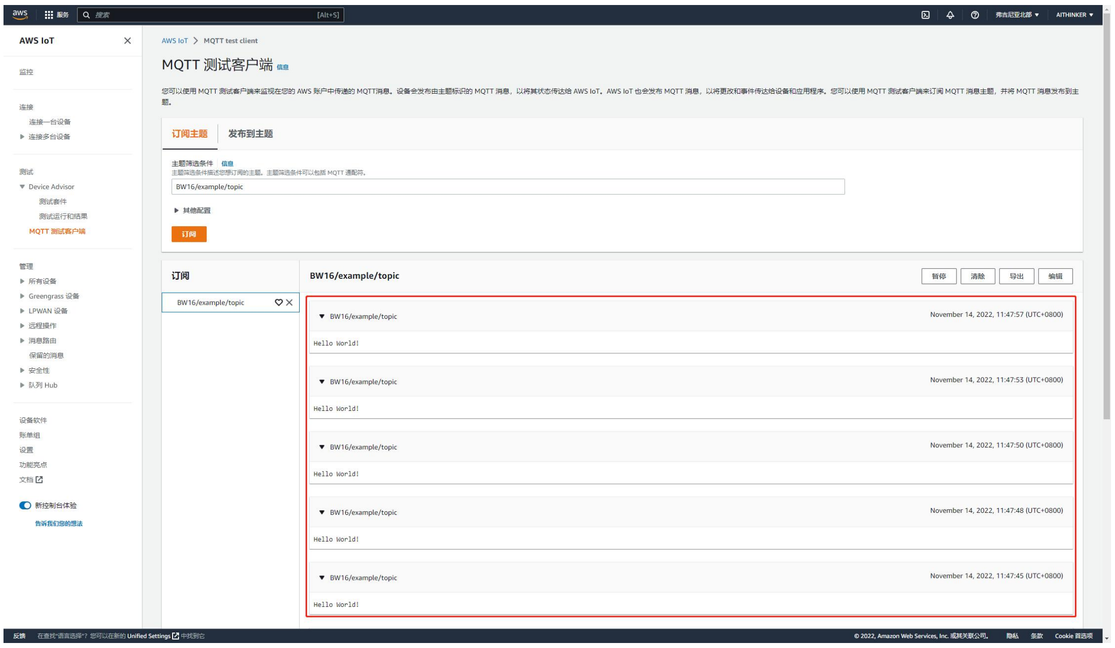
## 4.2.3 Publishing a Topic Message



```
[iot_threa]
39 23753 [iot_threa] [INFO] [MQTT_MutualAuth_Demo] [mqtt_demo_mutual_auth.c:1395] 40 23761
[iot_threa] Subscribed to the topic BW16/example/topic with maximum QoS 0.41 23769
[iot_threa]

[11:45:57.372]收←◆42 24271 [iot_threa] [INFO] [MQTT_MutualAuth_Demo]
[mqtt_demo_mutual_auth.c:873] 43 24279 [iot_threa] Publish to the MQTT topic
BW16/example/topic.44 24286 [iot_threa]
45 24292 [iot_threa] [INFO] [MQTT_MutualAuth_Demo] [mqtt_demo_mutual_auth.c:883] 46 24300
[iot_threa] Attempt to receive publish message from broker.47 24306 [iot_threa]

[11:45:58.612]收←◆48 25411 [iot_threa] [INFO] [MQTT] [core_mqtt.c:886] 49 25417 [iot_threa]
Packet received. ReceivedBytes=2.50 25422 [iot_threa]
51 25424 [iot_threa] [INFO] [MQTT] [core_mqtt.c:1162] 52 25429 [iot_threa] Ack packet
deserialized with result: MQTTSuccess.53 25436 [iot_threa]
54 25438 [iot_threa] [INFO] [MQTT] [core_mqtt.c:1175] 55 25445 [iot_threa] State record
updated. New state=MQTTPublishDone.56 25452 [iot_threa]
57 25454 [iot_threa] [INFO] [MQTT_MutualAuth_Demo] [mqtt_demo_mutual_auth.c:1373] 58 25462
[iot_threa] PUBACK received for packet Id 2.59 25467 [iot_threa]
60 25471 [iot_threa] [INFO] [MQTT] [core_mqtt.c:886] 61 25476 [iot_threa] Packet received.
ReceivedBytes=32.62 25482 [iot_threa]
63 25485 [iot_threa] [INFO] [MQTT] [core_mqtt.c:1045] 64 25490 [iot_threa] De-serialized
incoming PUBLISH packet: DeserializerResult=MQTTSuccess.65 25498 [iot_threa]
66 25500 [iot_threa] [INFO] [MQTT] [core_mqtt.c:1058] 67 25505 [iot_threa] State record
updated. New state=MQTTPublishDone.68 25512 [iot_threa]
```

SSCOM V5.13.1 串口/网络数据调试器,作者:大虾丁丁,2618058@qq.com. ...
通讯端口 串口设置 显示 发送 多字符串 小工具 帮助
清除窗口 打开文件 发送文件 停止 清发送区
端口号 COM21 USB Serial Port
HEX显示 保存数据 接收数据到文件 HEX发送 定时发送
关闭串口 更多串口设置
加时间戳和分包显示, 超时时间: 20 ms 第 1 字节 至 末尾
RTS DTR 波特率: 115200
为了更好地发展SSCOM软件 请您注册嘉立创F结尾客户
发 送
欢迎使用专业串口调试工具SSCOM ! 作者: 习小猛(丁丁), 大虾电子网版主 最新版本下载地址: http://ww
www.daxia.com S:0 R:70235 COM21 已打开 115200bps,8,1,None,None

## 4.3 View the subscription data in the AWS IOT console



AWS IoT > MQTT test client

MQTT 测试客户端 信息

您可以使用 MQTT 测试客户端来监视在您的 AWS 账户中传递的 MQTT消息。设备会发布由主题标识的 MQTT 消息，以将其状态传达给 AWS IoT。AWS IoT 也会发布 MQTT 消息，以将更改和事件传达给设备和应用程序。您可以使用 MQTT 测试客户端来订阅 MQTT 消息主题，并将 MQTT 消息发布到主题。

主题筛选条件 信息
主题筛选条件描述您想订阅的主题。主题筛选条件可以包括 MQTT 通配符。
BW16/example/topic

订阅: BW16/example/topic

```
BW16/example/topic                      November 14, 2022, 11:47:57 (UTC+0800)
Hello World!

BW16/example/topic                      November 14, 2022, 11:47:53 (UTC+0800)
Hello World!

BW16/example/topic                      November 14, 2022, 11:47:50 (UTC+0800)
Hello World!

BW16/example/topic                      November 14, 2022, 11:47:48 (UTC+0800)
Hello World!

BW16/example/topic                      November 14, 2022, 11:47:45 (UTC+0800)
Hello World!
```

aws 服务 搜索 [Alt+S]
弗吉尼亚北部 AITHINKER
AWS IoT
监控
连接 连接一台设备 连接多台设备
测试 Device Advisor 测试套件 测试运行和结果 MQTT 测试客户端
管理 所有设备 Greengrass 设备 LPWAN 设备 远程操作 消息路由 保留的消息 安全性 队列 Hub
设备软件 账单组 设置 功能亮点 文档
新控制台体验
订阅主题 发布到主题
其他配置
订阅
暂停 清除 导出 编辑
© 2022, Amazon Web Services, Inc. 或其关联公司。 隐私 条款 Cookie 首选项