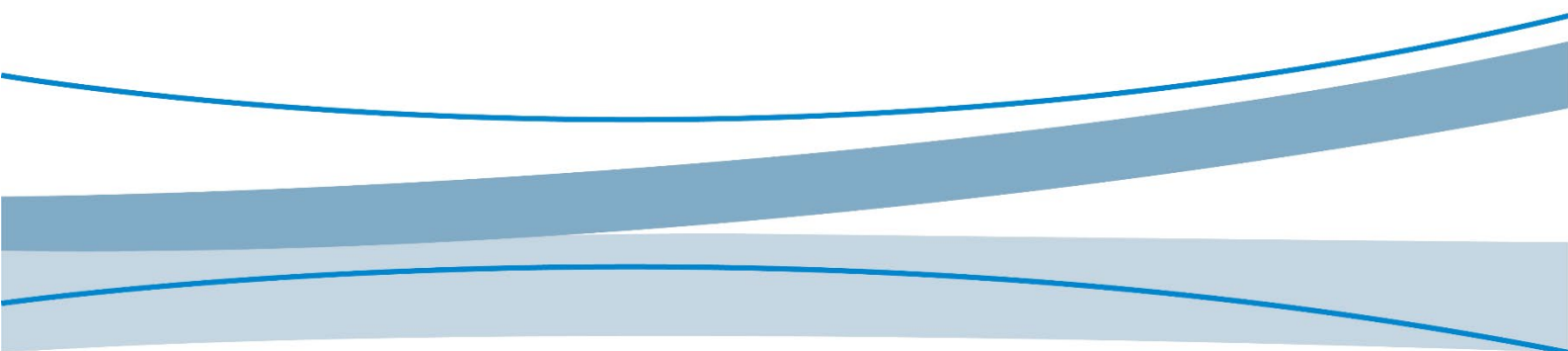




# MTC

## Application Guide\_Integrated Cloud

V1.0



## Disclaimer

Any action you take in the course of using this document is at your own risk, and Fibocom shall not be liable for any damages or losses under any circumstances. Due to product version upgrade or other reasons, Fibocom reserves the right to modify any information in this document at any time without prior notice and any responsibility. Unless otherwise agreed, all statements, information and suggestions in this document do not constitute any express or implied guarantee.

This document may include the third-party information covering products, services, software, data, and so on. Fibocom does not control and assumes no responsibility for the third-party content, including but not limited to the accuracy, compatibility, reliability, availability, legitimacy, appropriateness, performance, non-infringement, and status update, unless otherwise specified in this document. Fibocom does not provide any guarantee or authorization for the third-party content mentioned or referenced in this document. If you need a third-party license, obtain it in an authorized or legal way, unless otherwise specified in this document.

## Copyright Notice

Copyright © 2025 Fibocom Wireless Inc. All rights reserved.

Unless specially authorized by Fibocom, the recipient of the documents shall keep the documents and information received confidential, and shall not use them for any purpose other than the implementation and development of this project. Without the written permission of Fibocom, no unit or individual shall extract or copy part or all of the contents of this document without authorization, or transmit them in any form. Fibocom has the right to investigate legal liabilities for any offense and tort in connection with violation of confidentiality obligations, or unauthorized use or malicious use of the said documents and information in other illegal forms.

## Trademark Statement

 The trademark is registered and owned by Fibocom Wireless Inc.

Other trademarks, product names, service names and company names appearing in this document are owned by their respective owners.

## Contact Information

Website: <https://www.fibocom.com>

Address: 10/F-14/F, Block A, Building 6, Shenzhen International Innovation Valley, Dashi First Road, Xili Community, Xili Subdistrict, Nanshan District, Shenzhen

Tel: 0755-26733555

# Contents

Applicable Model .....	2
Change History .....	3
1 Overview .....	4
2 Amazon AWS .....	5
2.1 Create an AWS Account .....	5
2.2 AWS IoT Account Configuration .....	5
2.2.1 Create Items .....	7
2.2.2 Create Policy .....	11
2.2.3 Certificate Attachment Policy .....	13
2.3 Connection Example .....	14
3 Alibaba Cloud .....	20
3.1 One Secret for One Device Authentication .....	20
3.2 One Secret for One Type Authentication .....	21
3.3 Connection Example .....	23
3.3.1 Unencrypted Connection of One Secret for One Device Authentication .....	23
3.3.2 Pre-registration of One Secret for One Type .....	24
4 Huawei Cloud .....	26
4.1 Create Product .....	26
4.2 Create Device .....	28
4.3 Connection Example .....	28
5 AT Commands Used by MQTT Service .....	30

# Applicable Model

No.	Applicable Model	Description
1	All MTC products	NA

# Change History

---

V1.0 (2024-03-18)	Initial version.
-------------------	------------------

# 1 Overview

This document is to guide the basic operations for connecting the current module to the cloud platforms.

## 2 Amazon AWS

AWS IoT provides cloud services to connect IoT devices to other devices or AWS. AWS IoT communication supports MQTT (Message Queuing Telemetry Transport) and HTTPS (Hyper Text Transfer Protocol).

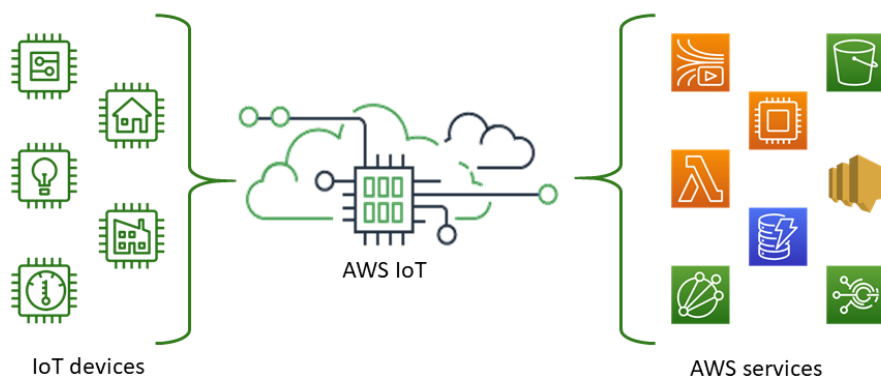


Figure 1. Amazon IoT cloud structure

### 2.1 Create an AWS Account

Before using AWS IoT Core for the first time, complete the following tasks:

- Register an AWS account.  
[https://docs.aws.amazon.com/zh\\_cn/iot/latest/developerguide/setting-up.html#aws-registration](https://docs.aws.amazon.com/zh_cn/iot/latest/developerguide/setting-up.html#aws-registration)
- Create a user and grant permissions  
[https://docs.aws.amazon.com/zh\\_cn/iot/latest/developerguide/setting-up.html#create-iam-user](https://docs.aws.amazon.com/zh_cn/iot/latest/developerguide/setting-up.html#create-iam-user)
- Open the AWS IoT control panel  
[https://docs.aws.amazon.com/zh\\_cn/iot/latest/developerguide/setting-up.html#iot-console-signin](https://docs.aws.amazon.com/zh_cn/iot/latest/developerguide/setting-up.html#iot-console-signin)

If you already have an AWS account and an IAM user, you can use them and jump forward to open the AWS IoT control panel.

### 2.2 AWS IoT Account Configuration

After entering the AWS control panel, you will see the following interface. This interface contains all the portals of Amazon Cloud. Select **IoT Core** to enter.

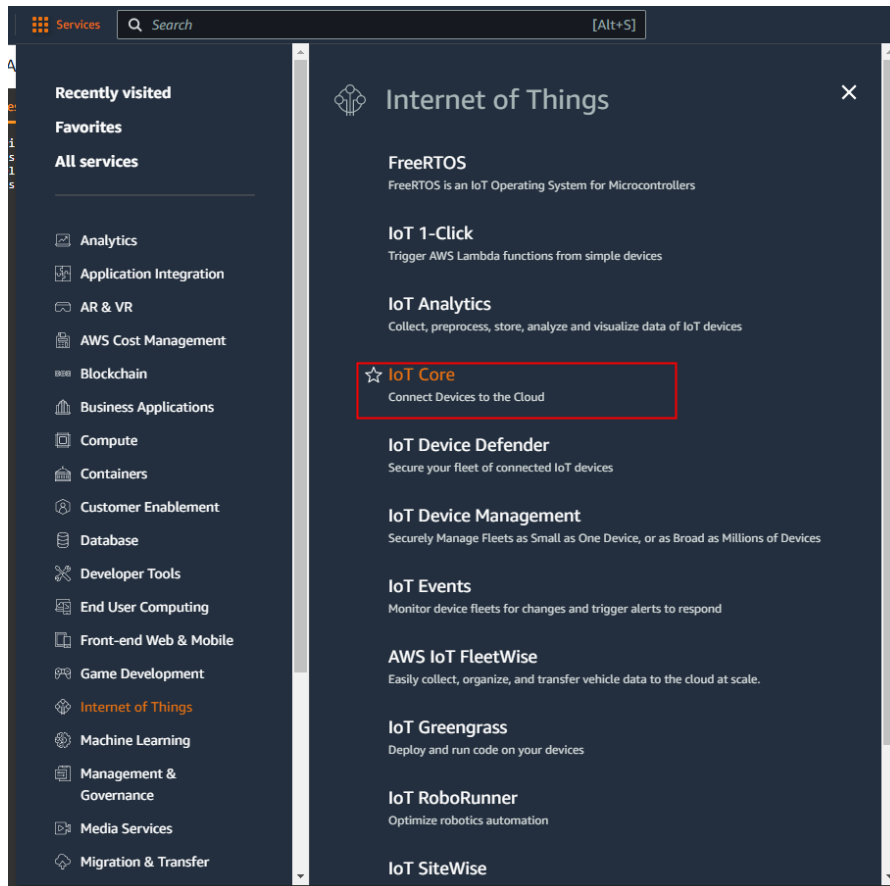


Figure 2. Amazon cloud control panel

The interface of AWS IoT is as follows:

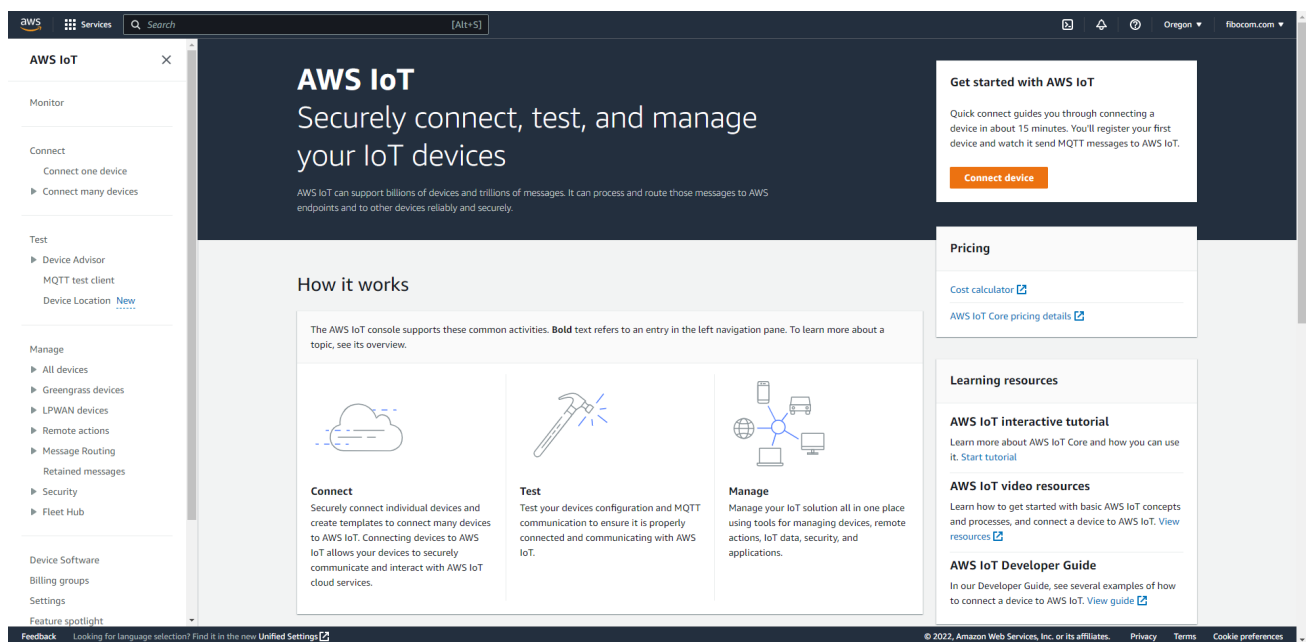


Figure 3. Amazon cloud IoT interface



Select **Manage > Things** to enter the Items interface, and click **Create things**. This interface is a list of devices to be created, as shown in the following figure.

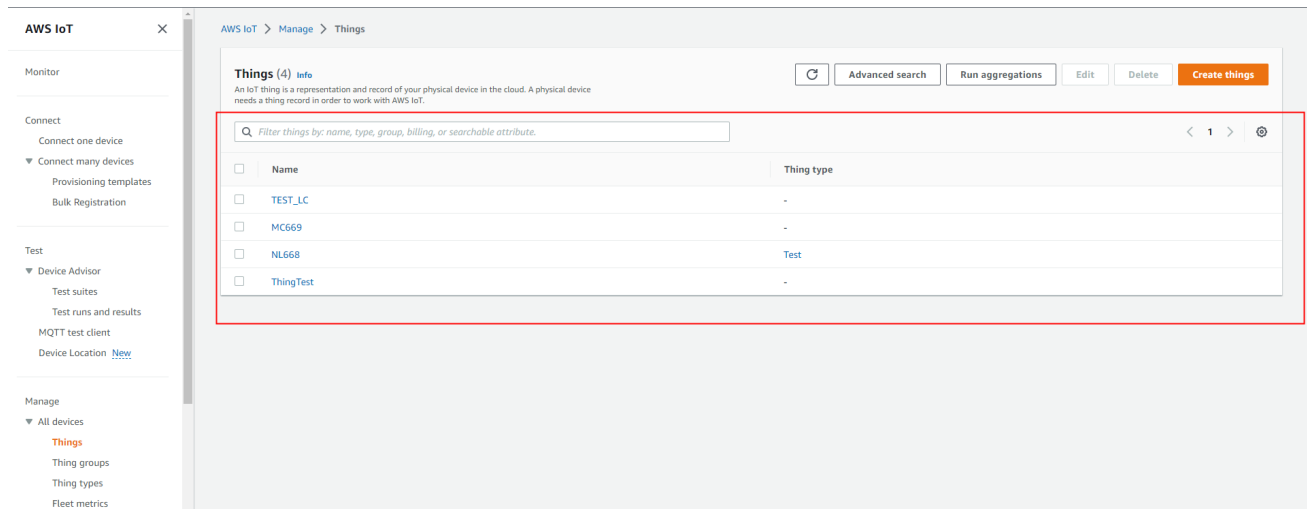


Figure 4. Amazon cloud device creating

## 2.2.1 Create Items

Enter the **Create things** interface, and select **Create single thing**, and click **Next**.

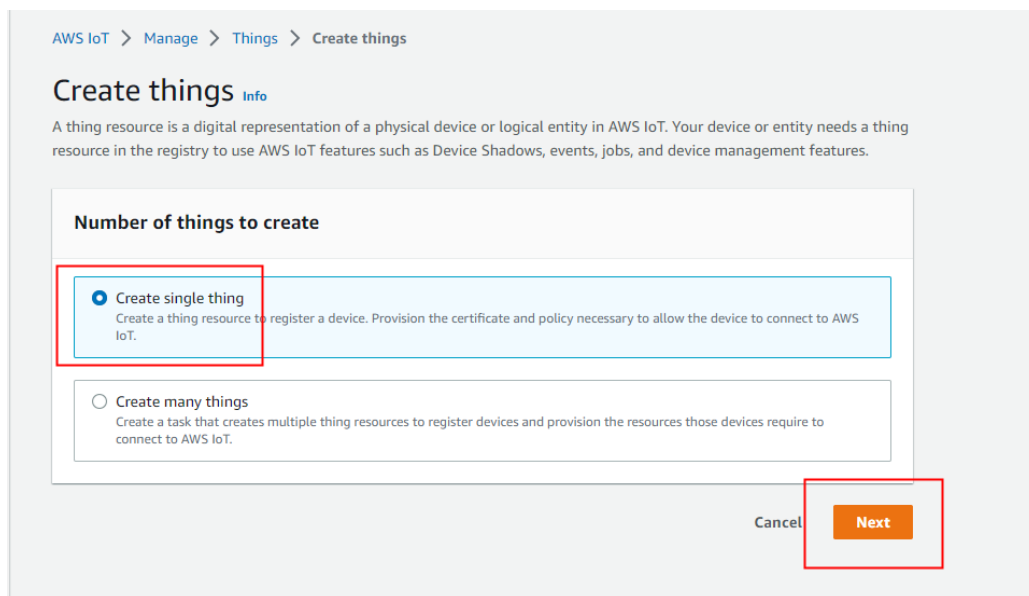


Figure 5. Creating single thing

In the interface of **Specify thing properties**, enter the thing name, for example, **fibocomTest**, and select the **Thing type**. If there is no creation type, you need to create types. Click the next step at the bottom page after creation.

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

Step 1  
**Specify thing properties**

Step 2 - optional  
Configure device certificate

Step 3 - optional  
Attach policies to certificate

### Specify thing properties [Info](#)

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 [Info](#)

Thing name

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

Figure 6. Creating thing

Create type interface: select **Create thing type** when you are finished.

### Create thing type

Thing types store description and configuration information that is common to similar devices.

Thing type name

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

Description - optional

#### Additional configuration

You can add additional information to the thing type that can help you to organize, manage, and search your things.

- ▶ Searchable attributes - optional
- ▶ Tags - optional

Cancel Create

Figure 7. Creating thing type

Create item certificates.

The screenshot shows the AWS IoT console interface for configuring a device certificate. The breadcrumb trail at the top reads: AWS IoT > Manage > Things > Create things > Create single thing. On the left sidebar, the steps are: Step 1: Specify thing properties; Step 2 - optional: **Configure device certificate**; Step 3 - optional: Attach policies to certificate. The main heading is 'Configure device certificate - optional' with an 'Info' link. Below the heading is a paragraph: '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.' The 'Device certificate' section contains four radio button options: 1. 'Auto-generate a new certificate (recommended)' (selected), with subtext 'Generate a certificate, public key, and private key using AWS IoT's certificate authority.' 2. 'Use my certificate', with subtext 'Use a certificate signed by your own certificate authority.' 3. 'Upload CSR', with subtext 'Register your CA and use your own certificates on one or many devices.' 4. 'Skip creating a certificate at this time', with subtext 'You can create a certificate for this thing and attach a policy to the certificate at a later time.' At the bottom right are three buttons: 'Cancel', 'Previous', and 'Next'.

Figure 8. Configure device certificate

After the creation, it will display the created certificates.

1. Save the public and private key files of the certificates
2. Download the root certificates.
3. After the saving the certificates, click activate, indicating that the certificates are available.
4. After the activation, select additional policy to complete the creation.

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

82cef407459...te.pem.crt

Deactivate certificate

Download

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

82cef4074592abb8ceff311...e8d960c-public.pem.key

Download

Private key file

82cef4074592abb8ceff311...8d960c-private.pem.key

Download

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

Download

Amazon trust services endpoint

ECC 256 bit key: Amazon Root CA 3

Download

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](#)

Done

Figure 9. Saving and activating certificates

Register things.

Copyright © Fibocom Wireless Inc.

10

The screenshot shows the AWS IoT console interface for creating a single thing. The breadcrumb trail is 'AWS IoT > Manage > Things > Create things > Create single thing'. The left sidebar shows three steps: 'Step 1: Specify thing properties', 'Step 2 - optional: Configure device certificate', and 'Step 3 - optional: Attach policies to certificate', which is currently selected. The main heading is 'Attach policies to certificate - optional' with an 'Info' link. Below the heading is a description: 'AWS IoT policies grant or deny access to AWS IoT resources. Attaching policies to the device certificate applies this access to the device.' The main content area is titled 'Policies (1/2)' and includes a 'Create policy' button with an external link icon. A search bar labeled 'Filter policies' is present. Below it is a table with two rows: 'fibocomTEST' (checked) and 'TEST' (unchecked). At the bottom right are 'Cancel', 'Previous', and 'Create thing' buttons.

Figure 10. Thing creation completed



You need to select a policy here, and if there is no policy, you can omit it.

## 2.2.2 Create Policy

You need to configure the policy because Amazon cloud policy will affect the access logic. You can select policies on Amazon Cloud IoT homepage.

The screenshot shows the 'Create policy' interface in the AWS IoT console. The breadcrumb trail is 'AWS IoT > Secure > Policies > Create policy'. The heading is 'Create policy' with an 'Info' link. Below it is a description: 'AWS IoT Core policies allow you to manage access to the AWS IoT Core data plane operations.' The 'Policy properties' section includes a 'Policy name' input field with a placeholder 'PolicyName' and a note: 'A policy name is an alphanumeric string that can also contain period (.), comma (,), hyphen(-), underscore (\_), plus sign (+), equal sign (=), and at sign (@) characters, but no spaces.' There is a 'Tags - optional' section. Below this are tabs for 'Policy statements' and 'Policy examples'. The 'Policy document' section has a 'Builder' and 'JSON' tab. It includes a 'Policy effect' dropdown set to 'Allow', a 'Policy action' dropdown set to 'Choose an action', and a 'Policy resource' input field with a placeholder 'arn:aws:iot:region:account:resource/resourceName' and a 'Remove' button. An 'Add new statement' button is at the bottom left. At the bottom right are 'Cancel' and 'Create' buttons.

Figure 11. Policy creation

On the **Create Policy** interface, enter a policy name, and select advanced mode to write the policy configuration. Click **Create** after the configuration.

AWS IoT > Secure > Policies > Create policy

## Create policy [Info](#)

AWS IoT Core policies allow you to manage access to the AWS IoT Core data plane operations.

**Policy properties**  
AWS IoT Core supports named policies so that many identities can reference the same policy document.

**Policy name**  
  
A policy name is an alphanumeric string that can also contain period (.), comma (,), hyphen(-), underscore (\_), plus sign (+), equal sign (=), and at sign (@) characters, but no spaces.

**Tags - optional**

**Policy statements** | Policy examples

**Policy document** [Info](#)  
An AWS IoT policy contains one or more policy statements. Each policy statement contains actions, resources, and an effect that grants or denies the actions by the resources.

**Policy effect**

**Policy action**

**Policy resource**

Figure 12. Policy statement

The policy used in the example is as follows:

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": "iot:*",
      "Resource": "*"
    },
    {
      "Effect": "Allow",
      "Action": "iot:Publish",
      "Resource": "arn:aws:iot:us-west-2:014957689912:topic/my/TEST"
    }
  ]
}
```

## 2.2.3 Certificate Attachment Policy

Select the certificate you have created on the **Certificates** interface, and enter the configuration page.

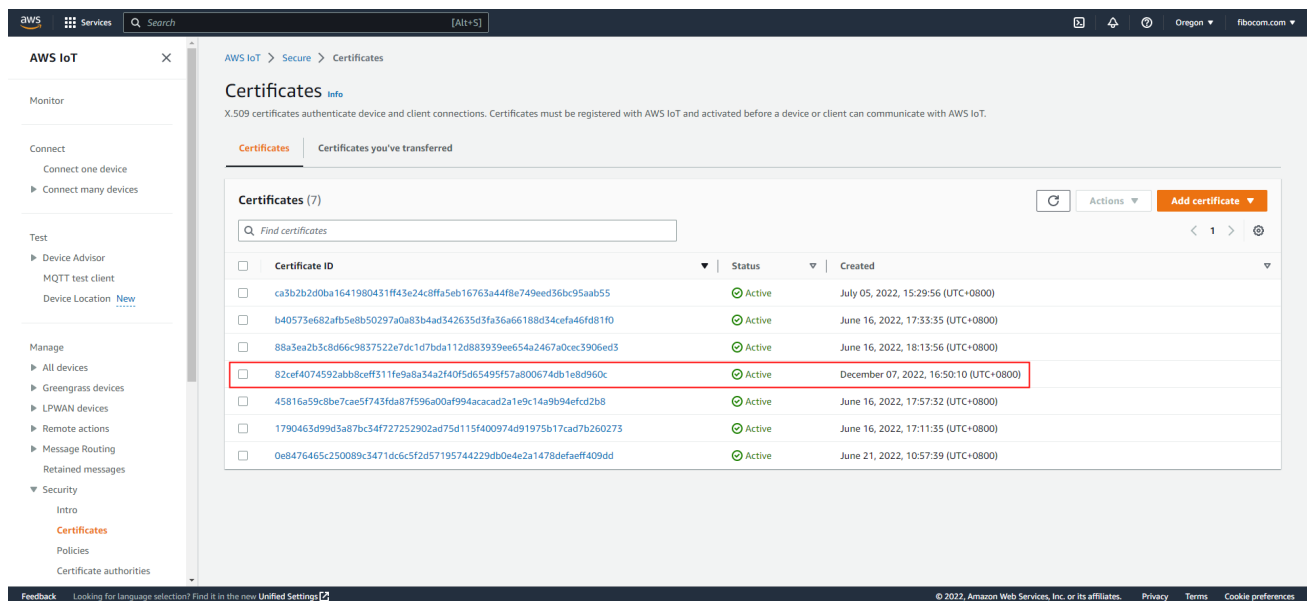


Figure 13. Selecting certificates

Choose **Actions** > **Attach to things**. Choose the created fibocomPolicy.

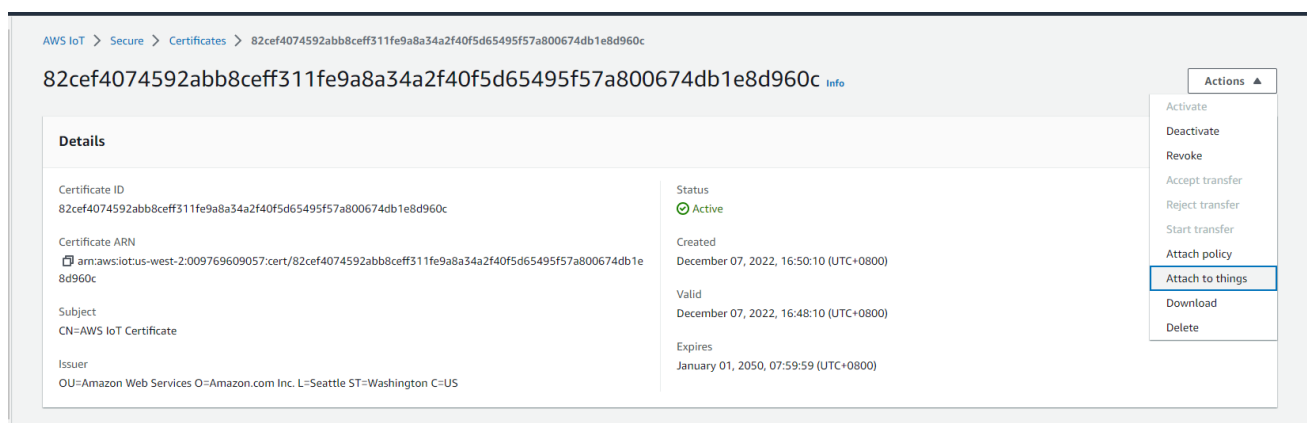


Figure 14. Attachment policy

Select the fibocomPolicy:

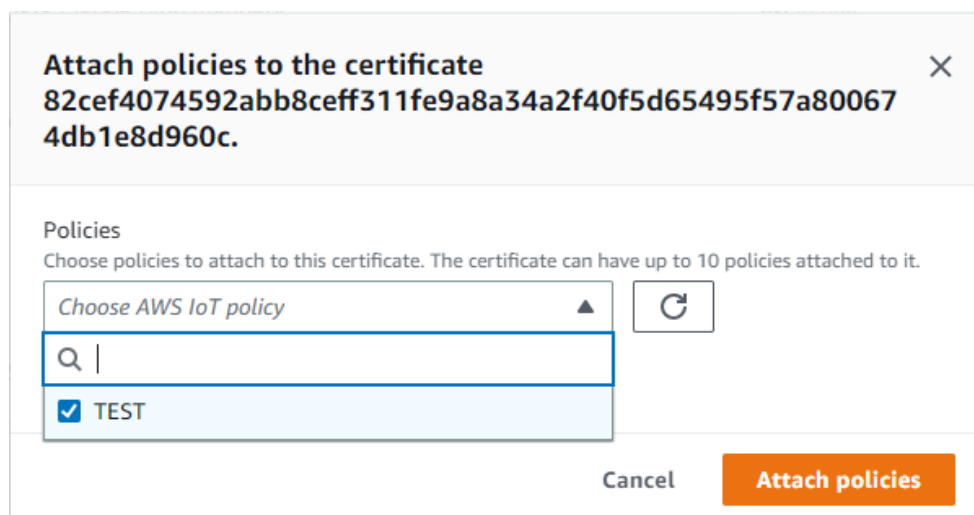


Figure 15. Attaching policies to certificate

After the above operations, the process of creating items, certificates and policies in Amazon Cloud has been completed. Items can be accessed normally.

## 2.3 Connection Example

```

AT                                // Send AT commands to the module continuously until the module
returns OK.

AT                                // Echo. If the first AT command is a setting command, it's
recommended to resend it after OK is returned.

AT command ready                  // Active report upon startup; it indicates the module is ready,
its time is not fixed, and it can be disabled using AT+MSTART.

+SIM READY                       // Active report that indicates the SIM card is identified upon
startup; its time is not fixed, and it can be disabled using AT+MSTART.

AT

AT                                // Echo. It can be disabled using ATE0. Subsequent echoes are
omitted for brevity.

OK

// The baud rate adaptation may cause the first setting command to be invalid. ATE0
is not saved upon power-down but takes effect immediately.

ATE0                             // Disable echo.

ATE0

OK

AT+GTRAT=6,3,2                   // Prioritize LTE in the network search sequence.

OK

AT+CFUN=1                         // Set the operating mode to normal operating mode.

```



```
OK
AT+CPIN?           // Check whether the SIM card is identified.
+CPIN: READY       // READY indicates that the SIM card is identified. If it's ERROR,
the SIM card may not be identified or may require a PIN to enter.

OK
AT+CIMI?           // Query IMSI
+CIMI: 460027295794151

OK
AT+CGDCONT=1,"IP","cmnet" // Set the APN first.
OK
AT+CSQ?            // Confirm the current signal strength.
+CSQ: 25,99        // 25 indicates the reference value of signal strength.

OK
AT+COPS?           // Query operator information.
+COPS: 0,0,"CHINA MOBILE",7 // 7 indicates 4G.

OK
AT+CGREG?          // Query whether the PS domain is registered. For 4G SIM card, it's
recommended to run the AT+CEREG? command to check again.
+CGREG: 0,1        // The value 1 or 5 indicates the SIM card is available.

OK
AT+CGREG?          // Query whether the LTE is registered.
+CEREG: 0,1        // The value 1 or 5 indicates the SIM card is available.

OK
AT+CREG?           // Query whether the CS domain is registered; this command can be
ignored if the CS domain related services are not required.
+CREG: 0,1         // The value 1 or 5 is returned. // The value 1 or 5 is returned.

OK
AT+MIPCALL?
+MIPCALL: 0

OK
AT+MIPCALL=1       // The module works with the network to activate PDP and requests
```



```
-----BEGIN CERTIFICATE-----
MIIDWTCCAKGgAwIBAgIUUge6GqtXTDaOV6+ZoJVnaIjFuUQwDQYJKoZIhvcNAQEL
BQAwTTFLEkGA1UECwxQW1hem9uIFdlYiBTZXJ2aWNlcyBPPUFTYXpvi5jb20g
SW5jLiBMPVNlYXR0bGUgU1Q9V2FzaGluZ3RvbiBDPVVTMB4XDTIwMTIzMDAzMDEy
NloXDTQ5MTIzMTIzNTk1OVowHjEcmBoGA1UEAwTQVdTIElvcyBBDXJ0aWZpY2F0
ZTCCASIwDQYJKoZIhvcNAQEBBQADggEPADCCAQoCggEBAA0zoIs0WJTazUAjtVRU5
gKAFwiwFEN9mi9fQjCrFDInRe+EzTvpwrR7/KF2yPWdNh2H5pwpH3Lvw2jBq+o8J
LrtXkFNaUw10LflMco0Vl6JvxmPhrpkh8aIW0+rolgpjUtcM/bhDTSaV0BsNQNbK
qjSpP/OxUdUwsnVlXNZLuoM+r51AC2ywPsPC7MURG5I0ibYoPx5P+AidaafTBqk/
37+1utwBBWoRPUwESCIRzVRcZ0cBWu7d87/FBUfcYzBdNb0IxpXasdAhZM+ayWh7
Kd0aRnXgUfZzqtXuWRmbf8WsiGdD3UDsmlzxq3AmRiMkToqBfboM2Yw7ggHp2e/z
ssUCAwEAANgMF4wHwYDVR0jBBgwFoAUdG/NaSspjIj4aD8t8FbGktuMLQwHQYD
VR00BBYEfBt0zxEPXj1k21bI9PSVKqpx3+B/MAWGA1UdEwEB/wQCMAAwDgYDVR0P
AQH/BAQDAgeAMA0GCSqGSIb3DQEBCwUAA4IBAQC1BB4ExyncVzFzAwgfbbJidZCS
baiaKvkr8lpsSYQVlD6Eeqi3Ng3IgZ066N8moxE8WBAQ2MEnbM1LRvCgthJXzns/
4js8fWchQh5VPxCTkPYXgws81njwZzFFBRn/xS0oeAl0PKUWzQTFqx9Fd9S+4X31
kiuMHPS+Jk9LLlmpy/HsDlmpSmDrSpcBsdR27NBE083PLPdZo8j0RCIISIXEQtbV
huzyatQXBxAcPzCf6r800CnpbI/dg9fS/NzXDaybrqxZxzM/RtcD9+NYtZ4DRGNq
qXdFQu5gtPEX3q905vitrjN8jpZEzKWeuSoGaekNf6z4bM681UMIhDvlp68Z
-----END CERTIFICATE-----
OK
```

```
AT+GTSSSLFILE="KEYFILE",1679 // Configure the private key file.
```

```
>
```

```
-----BEGIN RSA PRIVATE KEY-----
MIIEpAIBAAKCAQEA70iWzRYlMDNQC01VFTmAoAVaLB8Q32aL19CMkt8MidF74TNO
+nCtHv8oXbI9Z02HYfmnCkfCu/DaMGr6jwkuu1eQU1pTCXQt+UxyjRWXom/GY+Gu
mSHxohbT6uiWCmNS1wz9uENNjPQXGw1A1sqqOyk/87FR1RaydWVc1ku6gz6vnUAL
bLA+w8LsxREbkjSJtig/Hk/4CJ1pp9MGqT/fv7W63AEFAhE9TARIiHhNVFkx5wFa
7t3zv8UFR9xjMF01s4jGnFqx0CFkz5rJaHsp3RpGdeBR9nOq3G5ZExt/xayIZ0Pd
Q0yaXPGrcCZGIyR0ioF9ugzZjDuCaenZ7/OyxQIDAQABAOIBAALHo+VaKiuLnpfM
OSBboV+/UFSONMN6T4/DubFTFrGTTLHSyU10/wMLGW0oB0JsDXef107uku99nwHM
b444HF1EZMQlni3ROCzhscJoC3RzfU90uXjAhjZRsEDkY8ZRbgpp85wy5ffa8Csx
F56tmBvDmyV/4ibu8j8gZlt5XgHmN59dSg7SA2VRV1Lvt004SF8yiTGjBue/at8c
ex2VuSzkh3tvRvEtujhIgDrHsDEgLzmMWLCwEMl4cUB4NIuLOQzJmVR+3oAp2ToG
GPEqMgTOes8BMF53QASIrLSAXByuOmnHaH3u6321wjeVersHdQfZyXFfGEBN8Yz7
xY+p0E0CgYEA/B0/4qbeECMSLL+13uJWQNHLmYUI3e2xxk5DL9b/QZv9kigkyfi
DEdtfLc489gjCpHd+RNipnbECnv9PffYyzDj1fNEklAVSy8s50Y4br4DEJ/QTcbI
```

```
MdvGKyI1NKKDux3m/gUUJbZ3HBsuUpq6wqHyLro3D3n6V0uC29YxgwsCgYEA8I9X
53zka3AtCn/qc06Xg1KVgxys1fj9u1k4XG4a5NlUSmnYKjNQWUuW90+zkvjvq1FJ
Q00NdA5WS2MwXjrsv7qwODPW35KSy3ANBVV7Q0bfaUZAKZX/M9TRC+a3URc4QCJ5
FA0a0oQbX/IyxRmp8VckrtJhu5+sE28N7U+QQ28CgYA4mXRzNNUbuFPn0etKG3QT
v0WoAXxJ7iFZl18Ibp1/N4fB0dLa1pRX/10kJ5ognEBZqh+0QFbgWj9uvyE+XTsN
et9bc+7MDjCJnYCuN0SKEBxuCgLgwyTElLtriwsLVHpi8raeBpPcV5dr9uoyXryq
X2MYfHPKbK9gE7LAGdkpuwKBgQCz0aLXAm/Is/CnQZkimmP9oVTrzhqQU/BvBDFy
PkfS03abrWUNno/4P93Wt/tW6MhWwy7T14VcdH8jGUEFnyxr4YRqbq23L9yVP+wr
yYs7bhYAnqfFrj83ImUlbVAoyJ+eqWYyi9zIUwoXfXdgCGAVbJuaJt8xnhjF6iTR
+MCQ9wKBgQDHQqNz6uZlnQwA58+krdAc7QrpqZHOkWYj8WaQC5GY/lkitkmMIYBZ
JVfYE5TYQ+Q0hoywhV49I4EXEvZ8BPQ51eyREbQhzVLAPG4PFB9yKKftn5SAtkTi
5Ua6x4rSxj6x/8hfsk6+AkP3YyiwcW9U5jjxu0F8MfXtvUMOMD1pQQ==
-----END RSA PRIVATE KEY-----
```

OK

```
AT+GTSSLMODE=1 // Set up authentication server certificates.
```

OK

```
AT+MQTTUSER=1, "", "", "test" // Configure authentication information. Amazon Cloud
uses certificate connection, so name and password are empty.
```

OK

```
AT+MQTTOPEN=1,"aqb3vda67tst3-ats.iot.us-west-2.amazonaws.com",8883,1,300,2 //
Establish a connection.
```

OK

```
+MQTTOPEN: 1,1: // Connection succeeded.
```

```
AT+MQTTSUB=1, "my/TEST", 0 // Subscribe to messages.
```

OK

```
+MQTTSUB: 1,1 // Subscribed successfully.
```

```
AT+MQTTPUB=1,"my/TEST",0,0,"11111111" // Subscribe to messages.
```

OK

```
+MQTTPUB: 1,1 // Published successfully.
```

```
+MQTTMSG: 1,0,"my/TEST","11111111" // Received the published messages.
```

```
AT+MQTTCLOSE=1 // Disconnect from the server.
```

```
OK
```

```
+MQTTCLOSE: 1,1 // Disconnection succeeded.
```

## 3 Alibaba Cloud

Alibaba Cloud IoT platform, used as a device management platform, manages a large number of IoT devices to realize uplink and downlink data sending and receiving.

### 3.1 One Secret for One Device Authentication

One secret for one device authentication is to burn the unique certificate (ProductKey, DeviceName and DeviceSecret) for each device in advance. The IoT platform authenticates the device certificate information of the device when it is connected with the platform. After the authentication is passed, the device is activated and communicates with the IoT platform.

One secret for one device authentication for its high security to be recommended.

Before one secret for one device authentication, you should create products and devices. Select **Products > Create Product** on Alibaba Cloud menu page. Select **Directly Connected Device** for **Node Type** and **Device Secret** for **Authentication Mode**. The results are as follows:

IoT Platform / Devices / Products / Create Product


### ← Create Product (Device TSL)


[Create Product](#) [Create Product from Device Center](#)


\* Product Name

\* Category ⓘ  
☒ Standard Category ☐ Custom Category  
 [View Features](#)

\* Node Type

 Directly Connected Device

 Gateway sub-device

 Gateway device

Networking and Data Format

\* Network Connection Method

\* Data Type ⓘ

✓ Checksum Type

✓ Authentication Mode

More

✓ Product Description

Figure 16. Creating device

The screenshot shows the 'AUTODEVICE' product page in the IoT Platform console. The breadcrumb is 'IoT Platform / Devices / Products / AUTODEVICE'. The page has a 'Publish' button in the top right. Below the breadcrumb, there's a 'ProductKey' field with the value 'a1h0Epe3Ebv' and a 'Copy' link, and a 'ProductSecret' field with a masked value '\*\*\*\*\*' and a 'View' link. A 'Total Devices' count of '1' with a 'Manage' link is also present. A horizontal tab bar includes 'Product Information' (selected), 'Topic Categories', 'Define Feature', 'Data Parsing', 'Server-side Subscription', 'Device Provisioning', and 'File Uploading Configurations'. The 'Product Information' section has an 'Edit' link and displays the following details: Product Name (AUTODEVICE), Category (Roadway Lighting), Authentication Mode (Device Secret), Connection Protocol (Wi-Fi), Node Type (Directly Connected Device), Data Type (ICA Standard Data Format (Alink JSON)), Enabled status (toggle on), Created At (Nov 22, 2021, 10:11:59), Data Verification Level (Weak Verification), and Status (Developing). Below this is the 'Tag Information' section with an 'Edit' link, showing 'Product Tag' with 'No results found.'

Figure 17. Product details

You can get the ProductKey and ProductSecret information on the product page, and then select **Devices** > add devices from the menu to create devices for the related products. The result is as follows:

The screenshot shows the 'TestDevice' device page in the IoT Platform console. The breadcrumb is 'IoT Platform / Devices / Devices / TestDevice'. The page has an 'Inactive' status tag. Below the breadcrumb, there's a 'Products' field with the value 'AUTODEVICE' and a 'View' link, and a 'DeviceSecret' field with a masked value '\*\*\*\*\*' and a 'View' link. A 'ProductKey' field with the value 'a1h0Epe3Ebv' and a 'Copy' link is also present. A horizontal tab bar includes 'Device Information' (selected), 'Topic List', 'TSL Data', 'Device Shadow', 'Manage Files', 'Device Log', 'Online Debug', 'Groups', and 'Task'. The 'Device Information' section has an 'Edit' link and displays the following details: Product Name (AUTODEVICE), Node Type (Devices), Alias (TestDevice), Created At (Nov 22, 2022, 14:52:59), Current Status (Inactive), MQTT Connection Parameters (link to 'Here'), ProductKey (a1h0Epe3Ebv), DeviceName (TestDevice), IP Address (-), Activated At (-), Real-time Delay (Test), Region (China (Shanghai)), Authentication Mode (Device Secret), Firmware Version (-), Last Online (-), Device local log reporting (Disabled), ClientID (-), SDK Language (-), Version (-), and Module Manufacturer (-). Below this is the 'Tag Information' section with an 'Edit' link, showing 'Device Tag' with 'No results found.'

Figure 18. Device details

You can obtain the connection parameters of ProductKey, DeviceName, and DeviceSecret from the product page.

## 3.2 One Secret for One Type Authentication

In the one secret for one type authentication, all devices under the same product can burn the same device information, that is, the same product certificates (ProductKey and ProductSecret) that all devices contain. When a device sends an activation request, the IoT platform will confirm its identity and deliver the information required for the device access after it passes the authentication.

One secret for one type authentication supports two ways of use: free pre-registration (not supported at present) and pre-registration. The following figure describes the comparison.

Before one secret for one type authentication, you should create products and devices. Select **Products** > **Create Product** on Alibaba Cloud menu page. Choose **Directly Connected Device** for **Node Type** and **Device Secret** for **Authentication Mode**. The results are as follows:

IoT Platform / Devices / Products / Create Product

## ← Create Product (Device TSL)

[Create Product](#) [Create Product from Device Center](#)

\* Product Name

You must specify a product name

\* Category ?

☒ Standard Category ☐ Custom Category

Select a standard category View Features

\* Node Type

Directly Connected Device

Gateway sub-device

Gateway device

### Networking and Data Format

\* Network Connection Method

Wi-Fi

\* Data Type ?

ICA Standard Data Format (Alink JSON)

✓ Checksum Type

\* Authentication Mode ?

Device Secret

^ Hide

Figure 19. Creating product

IoT Platform / Devices / Products / AUTODEVICE

## ← AUTODEVICE

ProductKey a1h0Epe3Ebv Copy ProductSecret \*\*\*\*\* View

Total Devices 1 Manage

[Product Information](#) [Topic Categories](#) [Define Feature](#) [Data Parsing](#) [Server-side Subscription](#) [Device Provisioning](#) [File Uploading Configurations](#)

### Product Information

[Edit](#)

Product Name	AUTODEVICE	Node Type	Directly Connected Device	Created At	Nov 22, 2021, 10:11:59
Category	Roadway Lighting	Data Type	ICA Standard Data Format (Alink JSON)	Data Verification Level	Weak Verification
Authentication Mode	Device Secret	Device Secret	Enabled <input checked="" type="checkbox"/>	Status	Developing
Connection Protocol	Wi-Fi	Product Description	-		

### Tag Information

[Edit](#)

Product Tag No results found.

Figure 20. Product details

You can get the ProductKey and ProductSecret information on the product page, and then select



**Devices** > add devices from the menu to create devices for the related products. The result is as follows:

IoT Platform / Devices / Devices / TestDevice

← **TestDevice** Inactive

Products AUTODEVICE [View](#) DeviceSecret \*\*\*\*\* [View](#)  
 ProductKey a1h0Epe3Ebv [Copy](#)

Device Information Topic List TSL Data Device Shadow Manage Files Device Log Online Debug Groups Task

**Device Information**

Product Name	AUTODEVICE	ProductKey	a1h0Epe3Ebv <a href="#">Copy</a>	Region	China (Shanghai)
Node Type	Devices	DeviceName	TestDevice <a href="#">Copy</a>	Authentication Mode	Device Secret
Alias	TestDevice <a href="#">Edit</a>	IP Address	-	Firmware Version	-
Created At	Nov 22, 2022, 14:52:59	Activated At	-	Last Online	-
Current Status	Inactive	Real-time Delay	Test	Device local log reporting	Disabled <input type="checkbox"/>
MQTT Connection Parameters	<a href="#">Here</a>				
ClientID					

**More Device Information**

SDK Language	-	Version	-	Module Manufacturer	-
Module Information					

**Tag Information** [Edit](#)

Device Tag No results found.

Figure 21. Device details

You can obtain the connection parameters of ProductKey, DeviceName, and DeviceSecret from the product page.

## 3.3 Connection Example

### 3.3.1 Unencrypted Connection of One Secret for One Device

#### Authentication

AT+MIPCALL=1,"CMNET" // Request the operator to allocate an IP.

OK

+MIPCALL: 10.32. 232.1 // The return value is the IP address assigned by the operator.

AT+MQTTAUTHCFG=1,"a1Q0oULxrcp","TestDevice1","08ca857ade24c35ee06e37d225337599",0 // Obtain the cloud parameter configuration. The parameters are: cloudtype, ProductKey, DeviceName, DeviceSecret, and tls.

+MQTTAUTHCFG: 1,"TestDevice1&a1Q0oULxrcp","872ACD7B4758A99929B4CB58E5121B280C3F1C0D42B78492F62DE5CB1E9970D8","TestDevice1|securemode=3,signmethod=hmacsha256,timestamp=946745780|" // The return value parameters are cloudtype, Username, Password, and ClientID Str.

OK

```
AT+MQTTUSER=1,"TestDevice1&a1Q0oULxrcp","872ACD7B4758A99929B4CB58E5121B280C3F1C0D42B78492F62DE5CB1E9970D8","TestDevice1|securemode=3,signmethod=hmacsha256,timestamp=946745780|"
```

// Set MQTT authentication information with the following parameters: Client ID, Username, Password, and ClientIDStr.

OK

```
AT+MQTTOPEN=1,"iot-as-mqtt.cn-shanghai.aliyuncs.com",1883,0,300
```

// Establish an MQTT connection with the following parameters: Clientid, Remote IP/URL, RemotePort, Cleansession flag, and Keepalive time.

OK

```
+MQTTOPEN: 1,1
```

// MQTT connection established successfully.

```
AT+MQTTCLOSE=1
```

// Close MQTT connection.

OK

```
+MQTTCLOSE: 1,1
```

// MQTT connection closed successfully.

### 3.3.2 Pre-registration of One Secret for One Type

```
AT+MIPCALL=1
```

// Request the operator to allocate an IP.

OK

```
+MIPCALL: 100.104.156.37
```

// The return value is the IP address assigned by the operator.

```
AT+MQTTAUTHDYN=1,"iot-as-mqtt.cn-shanghai.aliyuncs.com",1883,"a1klgAXldch","HPoiKreay5vmZEHJ","867567040119311"
```

// Dynamic authentication with the following parameters: cloudtype, Host, Port, ProductKey, ProductSecret, and DeviceName.

OK

```
+MQTTAUTHDYN: 1,"a1klgAXldch","867567040119311","35057b24c65af058985d07cae8dcc2f8"
```

// The return value parameters are cloudtype, ProductKey, DeviceName, and DeviceSecret.

```
AT+MQTTAUTHCFG=1,"a1klgAXldch","867567040119311","35057b24c65af058985d07cae8dcc2f8",0
```

// Obtain the cloud parameter configuration. The parameters are: cloudtype, ProductKey, DeviceName, DeviceSecret, and tls.

```
+MQTTAUTHCFG:
```

```
1,"867567040119311&a1klgAXldch","26FD8D31AE53274507D43D4F5F8436004325AD8B78C5AFA1BFE9742FD32BB490","867567040119311|securemode=3,signmethod=hmacsha256,timestamp=1631931961|"
```

```
// Set MQTT authentication information with the following parameters: Client ID, Username, Password, and ClientIDStr.
```

```
OK
```

```
AT+MQTTUSER=1,"867567040119311&a1klgAXldch","26FD8D31AE53274507D43D4F5F8436004325AD8B78C5AFA1BFE9742FD32BB490","867567040119311|securemode=3,signmethod=hmacsha256,timestamp=1631931961|"
```

```
// Set MQTT authentication information with the following parameters: Client ID, Username, Password, and ClientIDStr.
```

```
OK
```

```
AT+MQTTOPEN=1,"iot-as-mqtt.cn-shanghai.aliyuncs.com",1883,0,300
```

```
// Establish an MQTT connection with the following parameters: Clientid, Remote IP/URL, RemotePort, Cleansession flag, and Keepalive time.
```

```
OK
```

```
+MQTTOPEN: 1,1 // MQTT connection established successfully.
```

```
AT+MQTTCLOSE=1 // Close MQTT connection.
```

```
OK
```

```
+MQTTCLOSE: 1,1 // MQTT connection closed successfully.
```

## 4 Huawei Cloud

### 4.1 Create Product

Enter the Huawei Cloud platform and click **Products** in the left-side navigation bar. On the **Products** page, click **Create Product** to enter the **Create Product** dialog box and complete product creation as instructed.

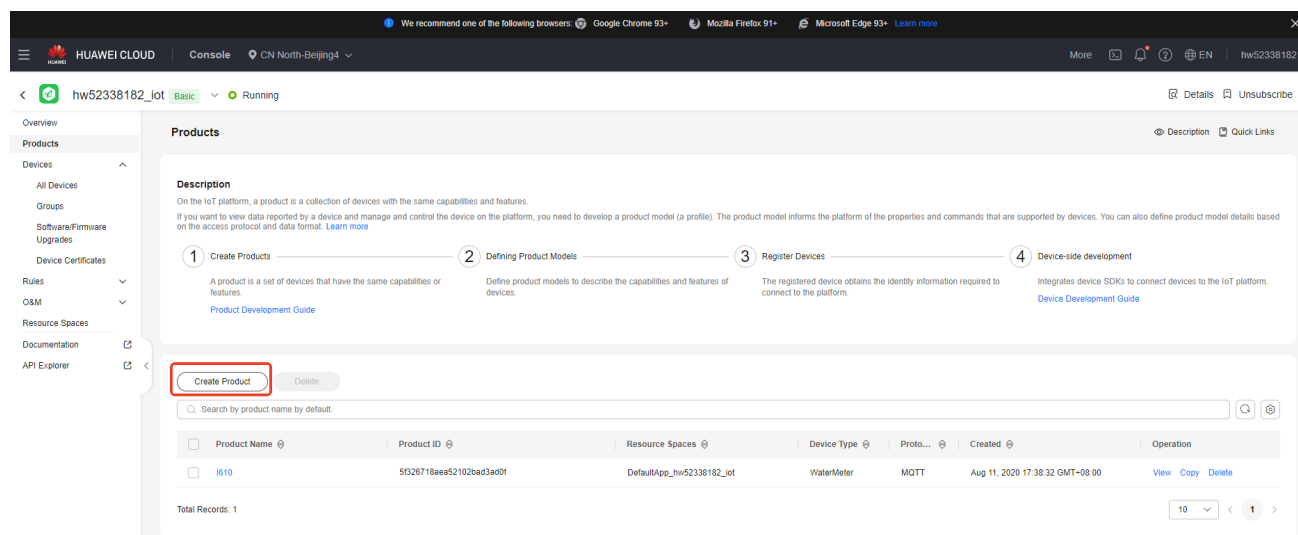


Figure 22. Products page

### Create Product

★ Resource Space ?

DefaultApp\_hw52338182\_iot

To create a new resource space, you can [go to the instance details page](#).

★ Product Name

TEST\_MQTT

Protocol ?

MQTT

★ Data Type ?

JSON

Device Type Selection

Standard profile Custom

★ Device Type ?

TEST\_MQTT

Advanced Settings ▾

Custom Product ID | Description

Cancel

OK

Figure 23. Create Product

On the **Basic Information** page of the product you just created, click **Customize Model** to add a service.

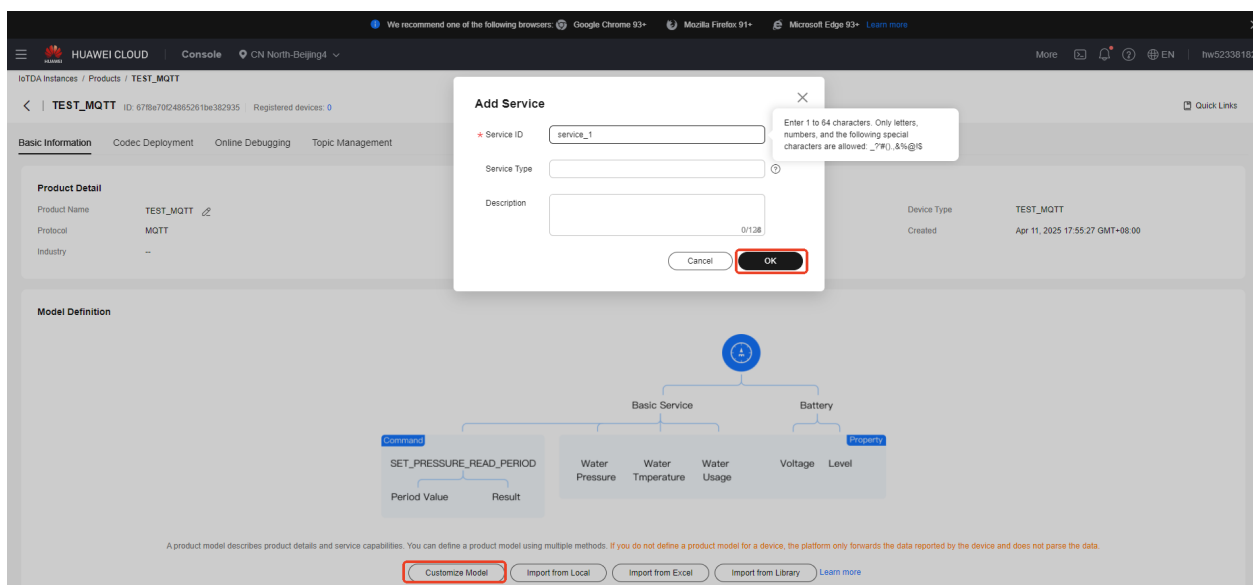


Figure 24. Add custom model

In the service list, select the service you added and add properties and commands for the service. You need to enter the correct parameter type and range based on the actual needs.

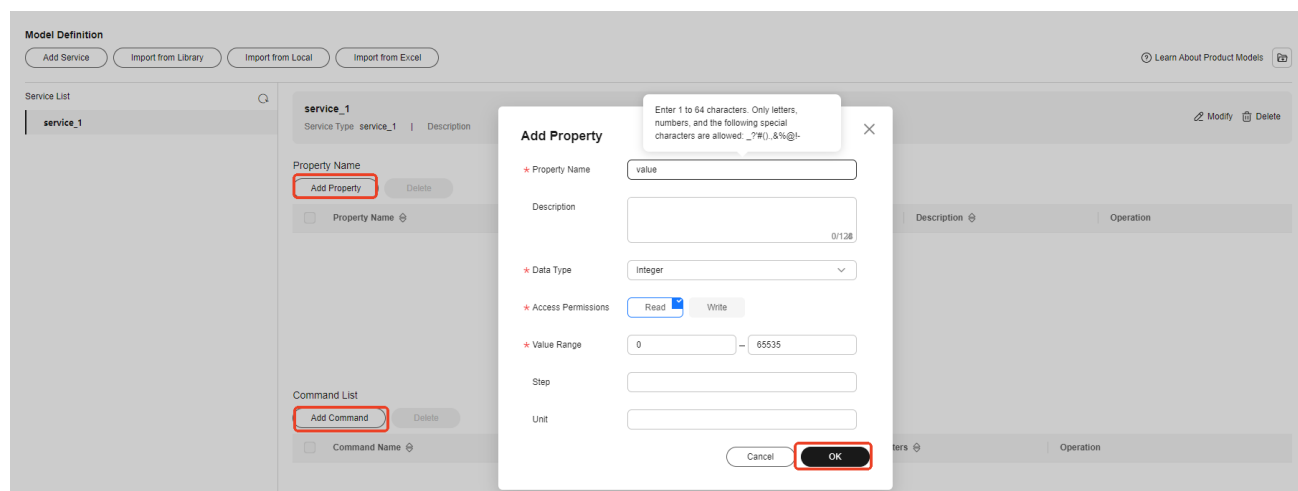


Figure 25. Add properties for custom service

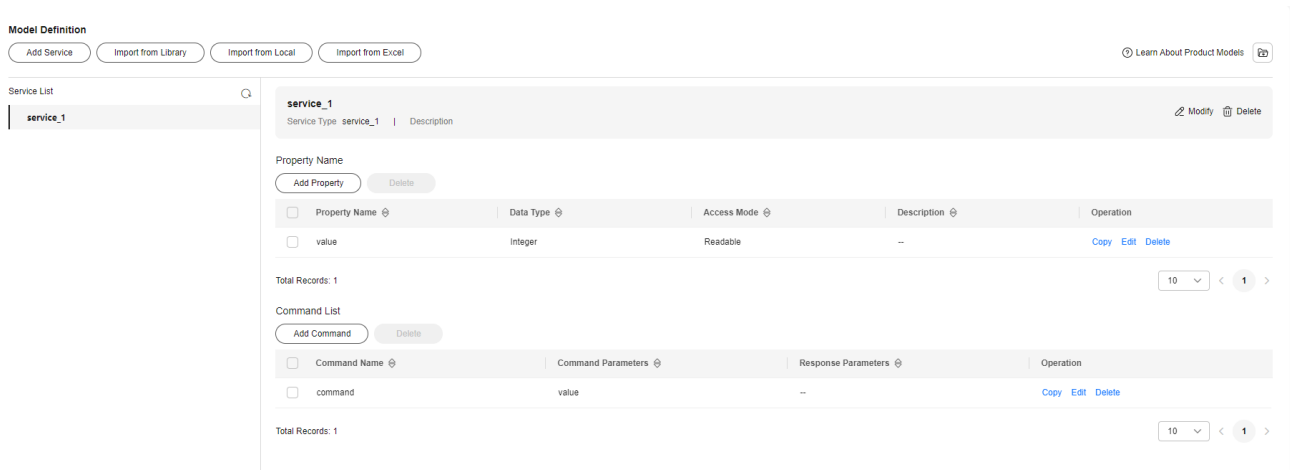


Figure 26. Result of adding custom model

After the custom model is added, the product creation is completed.

## 4.2 Create Device

Choose **Devices** > **All Devices** in the left-side navigation bar. Click the **Register Device** button and the **Register Device** dialog box will pop up. Fill in the product name you created and select the device ID to complete device creation.

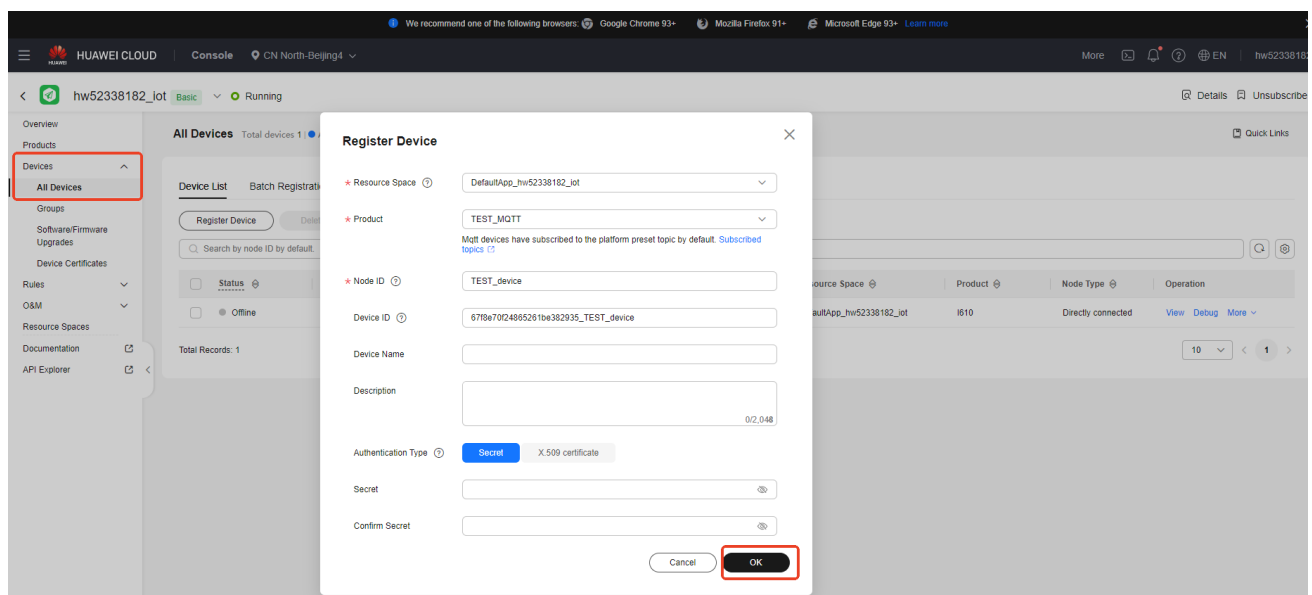


Figure 27. Create Device

## 4.3 Connection Example

```
AT+MIPCALL=1 //Request the operator to assign an IP address.  
OK
```

```
+MIPCALL: 100.100.87.143          //The return value is the IP address assigned by the operator.

AT + MQTTAUTHCFG = 3, "63351d7206cae4010b4b7750 _ TEST _ device",
"059801b5b3c5fed33e0c51af0398fc38" //Obtain the cloud parameter configuration. The parameters
are cloudtype, DeviceName, DeviceSecret.

+MQTTAUTHCFG: 3,"63351d7206cae4010b4b7750_TEST_device","57c2a50ab8234600e26f85fcd841
29058639d1d7a8fa94e6e3dfff1299ca347b","63351d7206cae4010b4b7750_TEST_device_0_0_2000
000105" //The return value parameters are cloudtype, Username, Password, ClientIDStr.
OK

AT + MQTTUSER = 1, "63351 d7206cae4010b4b7750 _ TEST _ device",
"57c2a50ab8234600e26f85fcd84129058639d1d7a8fa94e6e3dfff1299ca347b", "63351
d7206cae4010b4b7750 _ TEST _ device _ 0 _ 0 _ 2000000105" //Set the user configuration.
The parameters are clientid, Username, Password, ClientIDStr.
OK

AT+MQTTOPEN=1,"iot-mqtts.cn-north-4.myhuaweicloud.com",1883,1,300 //Establish an MQTT
connection. The parameters are clientid, Remote IP/URL, RemotePort, Cleansession flag, Keepalive
time.
OK

+MQTTOPEN: 1,1          //MQTT connection established successfully.

AT+MQTTCLOSE=1          //Close MQTT connection.
OK

+MQTTCLOSE: 1,1          //MQTT connection closed successfully.
```

## 5 AT Commands Used by MQTT Service

AT Command	Function Description
AT+MIPCALL	Ask the operator for an IP address. Fill the APN of the SIM card in the position of <b>cmnet</b> , which can be obtained from the SIM card supplier. Generally, China Mobile uses CMNET or CMIOT as its APN, and China Unicom uses 3gnet as its APN. If you need to configure a username and password, refer to +MIPCALL in AT Manual .
AT+MQTTAUTHDYN	Dynamic registration instruction configuration, supporting Alibaba Cloud and Tencent Cloud
AT+MQTTAUTHCFG	Cloud platform MQTT connection parameter calculation
AT+MQTTUSER	Set MQTT authentication information.
AT+MQTTOPEN	Establish MQTT connection.
AT+MQTTSUB	Subscribe to a topic.
AT+MQTTPUB	Publish a message to a topic.
AT+MQTTPUB	Publish a message to a topic (ODM mode. JSON supported).
AT+MQTTUNSUB	Unsubscribe from a topic.
AT+MQTTCLOSE	Close MQTT connection.
AT+MQTTWILL	Set the will message; the setting command must be sent before MQTT connection is established.
AT+GTSSLFILE	Load TLS certificate and key.
AT+GTSSLVER	Query or set TLS version.
AT+GTSSLMODE	Set whether to verify the server certificate.
AT+GTSSLERR	Get TLS error code.
AT+CCLK	Query command, query module current time