QuecPython Alibaba Cloud IoT Platform Access User Guide

LTE Standard Module Series

Version: 1.0.0

Date: 2020-11-10

Status: Preliminary

Our aim is to provide customers with timely and comprehensive service. For any assistance, please contact our company headquarters:

Quectel Wireless Solutions Co., Ltd.
Building 5, Shanghai Business Park Phase III (Area B), No.1016 Tianlin Road, Minhang District, Shanghai 200233, China
Tel: +86 21 5108 6236
Email: info@quectel.com

Or our local office. For more information, please visit:
http://www.quectel.com/support/sales.htm.

For technical support, or to report documentation errors, please visit:
http://www.quectel.com/support/technical.htm
Or email to support@quectel.com.

General Notes

Quectel offers the information as a service to its customers. The information provided is based upon customers' requirements. Quectel makes every effort to ensure the quality of the information it makes available. Quectel does not make any warranty as to the information contained herein, and does not accept any liability for any injury, loss or damage of any kind incurred by use of or reliance upon the information. All information supplied herein is subject to change without prior notice.

Disclaimer

While Quectel has made efforts to ensure that the functions and features under development are free from errors, it is possible that these functions and features could contain errors, inaccuracies and omissions. Unless otherwise provided by valid agreement, Quectel makes no warranties of any kind, implied or express, with respect to the use of features and functions under development. To the maximum extent permitted by law, Quectel excludes all liability for any loss or damage suffered in connection with the use of the functions and features under development, regardless of whether such loss or damage may have been foreseeable.

Duty of Confidentiality

The Receiving Party shall keep confidential all documentation and information provided by Quectel, except when the specific permission has been granted by Quectel. The Receiving Party shall not access or use Quectel's documentation and information for any purpose except as expressly provided herein. Furthermore, the Receiving Party shall not disclose any of the Quectel's documentation and information to any third party without the prior written consent by Quectel. For any noncompliance to the above requirements, unauthorized use, or other illegal or malicious use of the documentation and information, Quectel will reserve the right to take legal action.

Copyright

About the Document

Revision History

Version	Date	Author	Description
-	2020-11-10	Kinney/Kingka	Initial
1.0.0	2020-11-10	Kinney/Kingka	Initial

Contents

Table Index

Figure Index

1 Introduction

This document takes EC100Y-CN as an example to show how to access Alibaba Cloud with QuecPython class library APIs.

This document is applicable to the following Quectel modules:

- EC100Y-CN
- EC600S-CN

2 Alibaba Cloud IoT Platform

2.1. Platform Introduction

Alibaba Cloud IoT Platform allows stable communication between devices and the IoT platform, connects massive devices downwards, supports to collect device data and uploads to the cloud; provides cloud APIs upwards, and the server sends commands to the device by calling the cloud APIs to achieve remote control. The IoT platform also provides other communication management capabilities, such as device management, rule engine, security capabilities, etc. for rapid data collection, storage, and application development.

Alibaba Cloud IoT platform document center URL: https://www.alibabacloud.com/help?spm=a2c63.l28256.3156523820.dnavd. You can log on to this website to learn more about the IoT platform and device access related knowledge.

2.2. Explanation of Key Terms in the IoT Platform

The following table briefly introduces the key terms in the Alibaba Cloud IoT platform. For detailed information, see Alibaba Cloud official documents in URL: https://www.alibabacloud.com/help/doc-detail/30524.htm?spm=a2c63.l28256.b99.4.67e82b73H77imJ.

Table 1: Key Terms

Key Term	Explanation
product	A product is a set of devices that have the same features. IoT Platform issues a unique ProductKey for each product.
device	A physical device that belongs to a product. IoT Platform issues a DeviceName that is unique under the same product for each device. Devices can directly connect to IoT Platform, or be attached as sub-devices to a gateway that is connected to IoT Platform.
sub-device	Sub-devices cannot directly connect to IoT Platform and must be attached to a gateway.

gateway	A gateway can directly connect to IoT Platform and allows you to manage sub-devices. Sub-devices can communicate with IoT Platform only by using a gateway.
device certificate	A device certificate consists of ProductKey, DeviceName, and DeviceSecret. ● ProductKey is the unique identifier of a product in IoT Platform. This parameter is required in device authentication and communication. You must safely keep this parameter. ● DeviceName is the device name that is generated by IoT Platform during device registration. You can also upload custom device names. Each device has a unique DeviceName under the same product. This parameter is required in device authentication and communication. You must safely keep this parameter. ● DeviceSecret is the private key that is issued by IoT Platform for each device. DeviceSecret is used in pair with DeviceName. This parameter is required in device authentication and communication. You must safely keep this parameter.
ProductSecret	ProductSecret is the private key that is issued by IoT Platform for each product. ProductSecret is used in pair with ProductKey for unique-certificate-per-product authentication. This parameter is required in device authentication and communication. You must safely keep this parameter.
unique-certificate-per-device authentication	A device certificate is burned to each device. The device certificate includes a ProductKey, DeviceName, and DeviceSecret. When you connect a device to IoT Platform, IoT Platform authenticates the device based on the certificate.
unique-certificate-per-product authentication	A product certificate is burned to all devices under the same product. A product certificate includes a ProductKey and ProductSecret. When a device sends an activation request, IoT Platform authenticates the device based on the certificate. If the authentication succeeds, IoT Platform issues a DeviceSecret to the device. Then, the device uses the DeviceSecret to connect with IoT Platform.
Topic	A topic is a UTF-8 character string that is used as a transmission medium during publish/subscribe communication. A device can publish messages to a topic or subscribe to messages from a topic.
rules engine	You can create and configure rules in IoT Platform to achieve the following features: server-side subscription, data forwarding, and scene orchestration.
TSL model	IoT Platform uses the Thing Specification Language (TSL) to describe device features. A TSL model defines the device properties, services, and events. TSL models use the JSON format. You can organize data based on a TSL model and submit the data to IoT Platform.

3 Alibaba Cloud related APIs

3.1. aLiYun

This function configures the product and device information of the Alibaba Cloud IoT suite

- **Prototype**

```
aLiYun(productKey, productSecret, DeviceName, DeviceSecret)
```

- **Parameter**

productKey:
The unique identifier of a product

productSecret:
The product secret which is optional. Default: None.
In unique-certificate-per-device authentication, input None (Cannot be an empty string).
In unique-certificate-per-product authentication, input the real product secret.

deviceName:
Device name

deviceSecret:
Device secret which is optional. Default: None.
In unique-certificate-per-product authentication, input None

- **Return Value**

Return the Alibaba Cloud connection object.

3.2. aLiYun.setMqtt

This function sets the parameters of the MQTT data channel.

- **Prototype**

```
aLiYun.setMqtt(clientID, clean_session, keepAlive)
```

- **Parameter**

clientID:
Customized Alibaba Cloud connection ID

clean_session:
A Boolean value that determines the client type which is optional. Default: False.
If it is True, the agent deletes all information about this client when it disconnects from the client.
If it is False, the client is a persistent client. When the client disconnects, subscription information and queued messages are reserved.

keepAlive:
The maximum time allowed between communications. Range: 60–1200; Unit: second; Default: 300.

- **Return Value**

None

3.3. aLiYun.setCallback

This function registers the callback function.

- **Prototype**

```
aLiYun.setCallback(sub_cb)
```

- **Parameter**

sub_cb:
The callback function

- **Return Value**

None

3.4. aLiYun.subscribe

This function subscribes to MQTT topics.

- **Prototype**

```
aLiYun.subscribe(topic,qos)
```

- **Parameter**

topic:
The subscribed topic

qos:
MQTT message service quality. Default: 0. It can be 0 or 1.

- **Return Value**

None

3.5. aLiYun.publish

This function published a message.

- **Prototype**

```
aLiYun.publish(topic,msg)
```

- **Parameter**

topic:
The published topic

msg:
The data to be sent

- **Return Value**

None

3.6. aLiYun.start

This function starts to run the connection.

- **Prototype**

```
aLiYun.start()
```

- **Parameter**

None

- **Return Value**

None

4 Connecting Alibaba Cloud with QuecPython

4.1. Creating Products and Equipment

When using the Alibaba Cloud IoT platform, you first need to create products and corresponding devices in the cloud, and obtain device certificates (ProductKey, DeviceName, and DeviceSecret). A product is equivalent to a collection of devices of one type, and devices under the same product have the same functions. Download the device certificate issued by the IoT platform to the device and use it for identity verification when the device is connected to the IoT platform.

Step 1: Log in to the Alibaba Cloud IoT platform console by https://www.alibabacloud.com/. After login, Click "**Devices**" and then "**Products**" in the navigation bar to start creating product and configure the parameters.

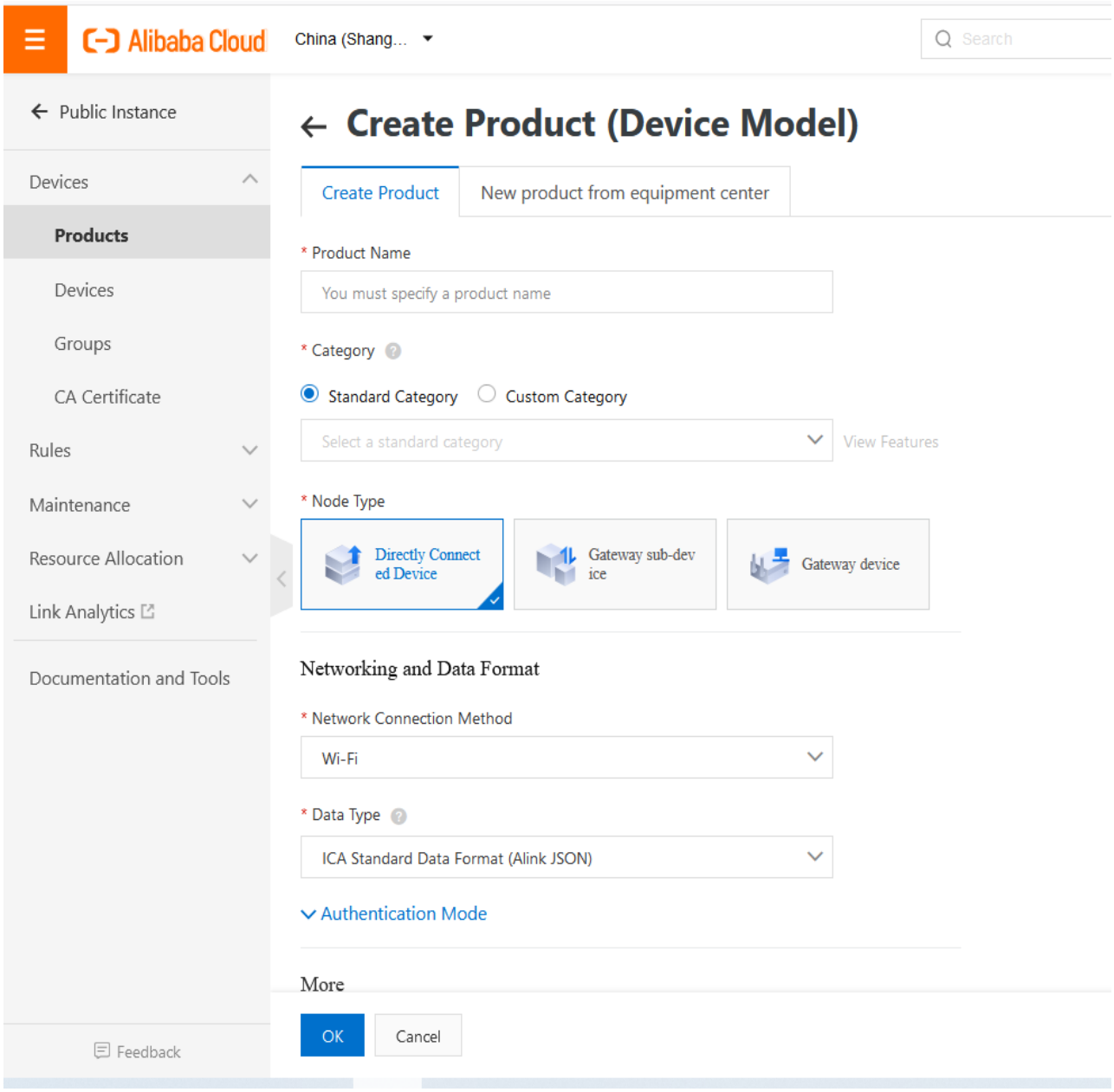

Figure 1: Create Product

Step 2: It is recommended to configure according to the parameters shown in the figure below. To learn more about the parameter configuration, please visit the official address of Alibaba Cloud IoT platform: https://www.alibabacloud.com/help/product/30520.htm.

After creation, click "**OK**" to save the configuration.

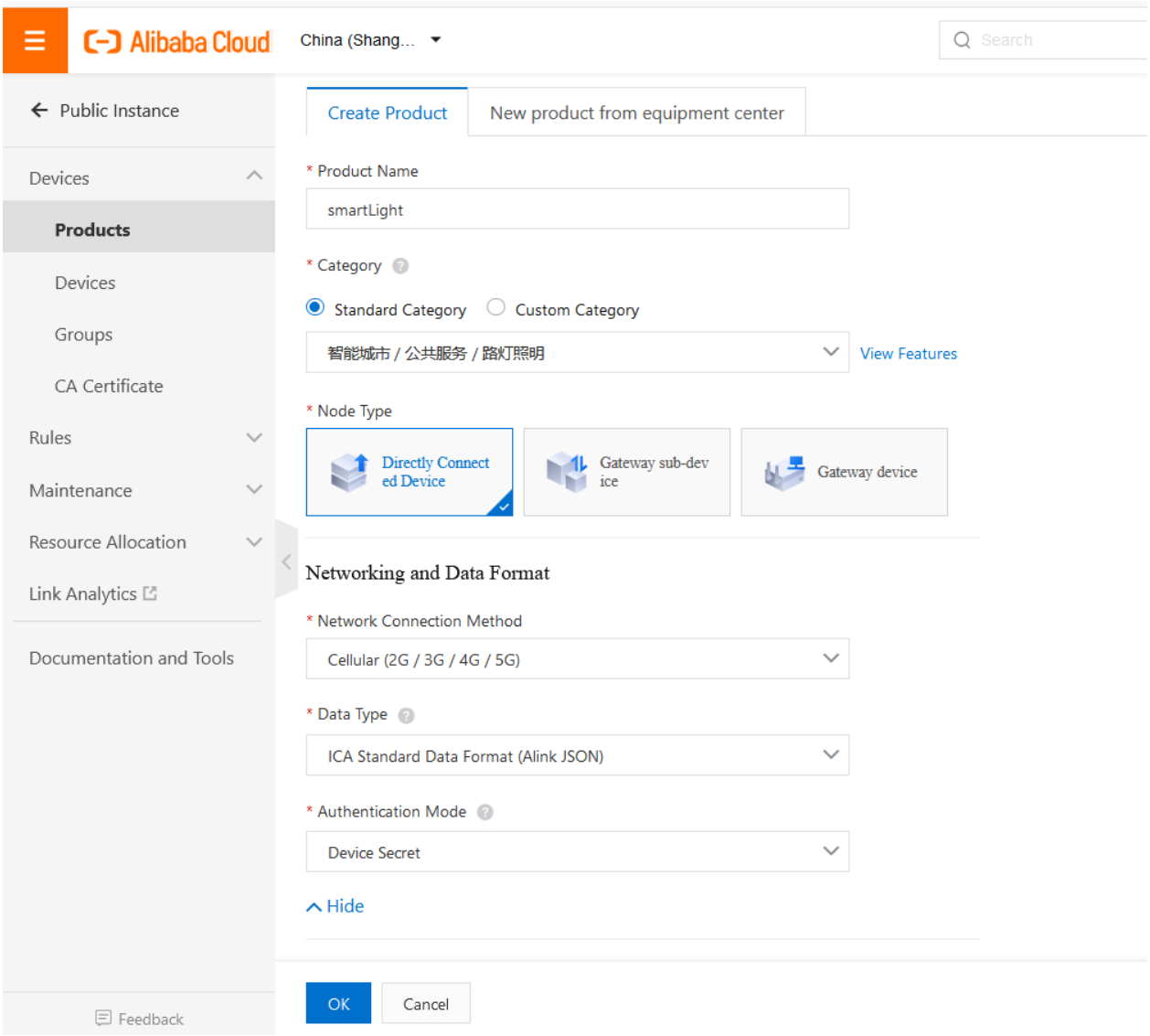

Figure 2: Reference of Product Configurations

Step 3: A product refers to a certain type of device. After creating the product, you need to add an identity for the device. You can add one single device or a batch of devices at a time. Here takes the creation of one single device as an example for detailed introduction. Click "**Devices**" and "**Add Device**" to start adding a device.

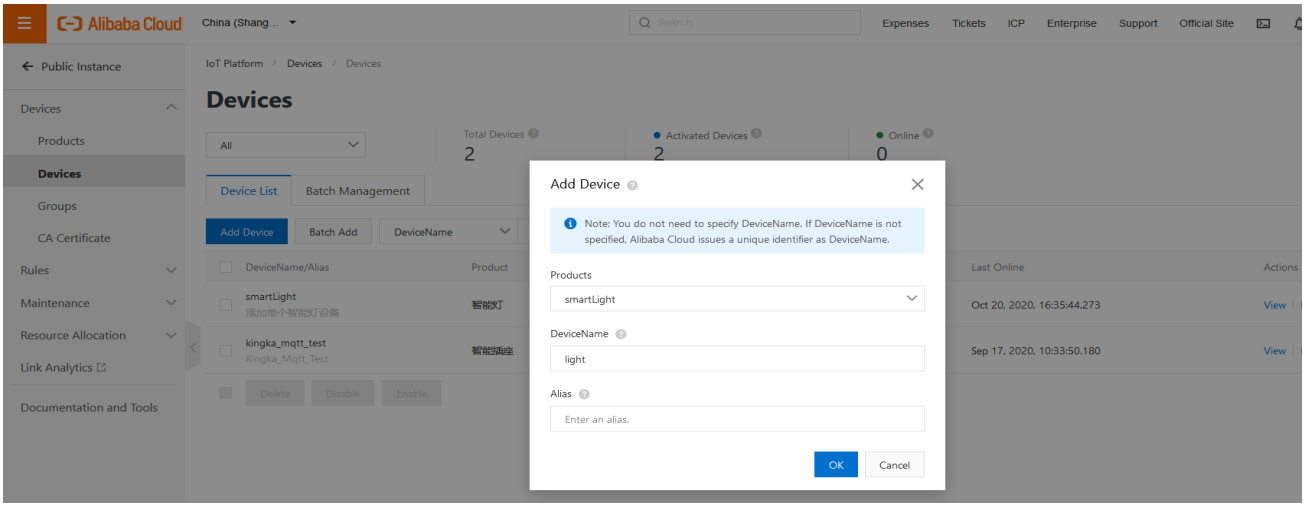

Figure 3: Add Device

Step 4: After the device is successfully added, the "Device Certificate" window automatically pops up, and you can view and copy the device certificate information. Consisting of ProductKey, DeviceName, and DeviceSecret of the device, the device certificate is an important identity authentication for the communication between the device and the IoT platform. Please keep the device certificate properly.

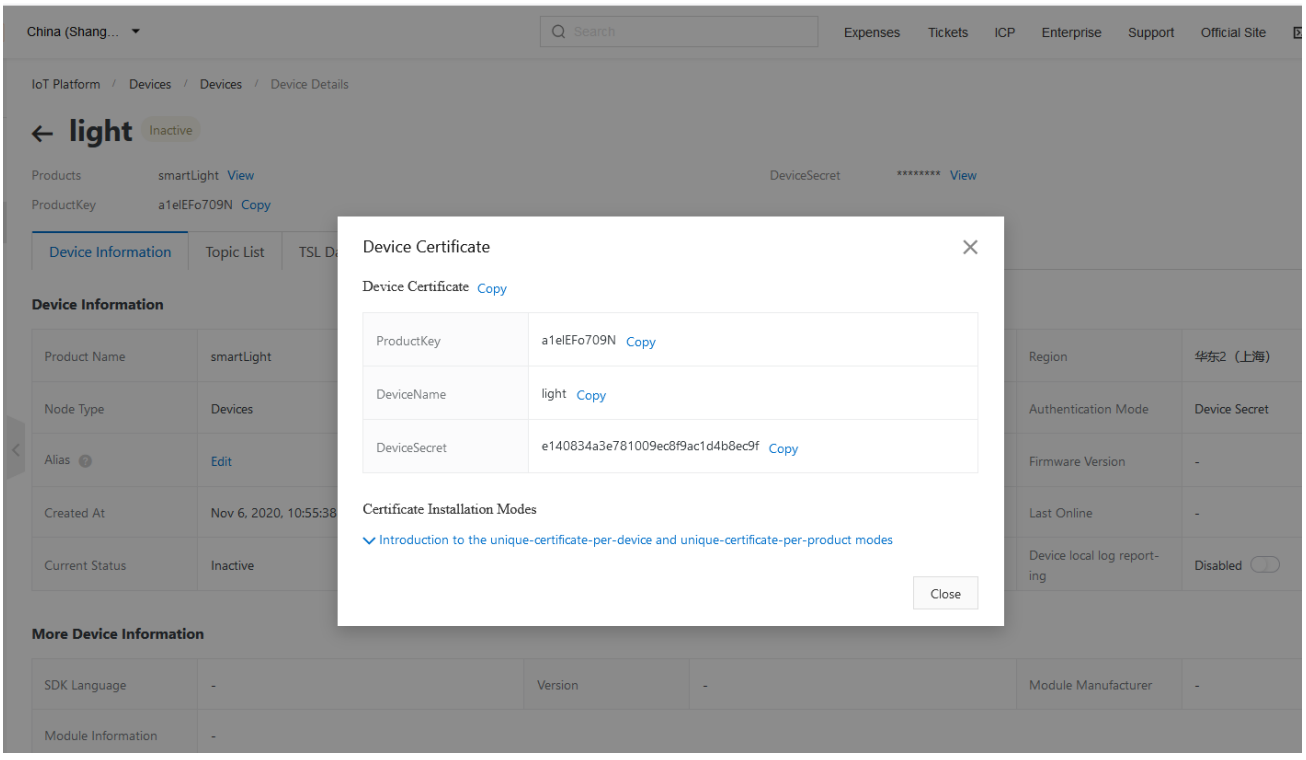

Figure 4: Device Certificate Information

4.2. Connet to Alibaba Cloud IoT Platform

Before accessing the Alibaba Cloud IoT platform, please confirm that you have added products and devices on the IoT platform console and had the device certificate information (ProductKey, DeviceName, and DeviceSecret).

4.2.1. Connect to the IoT Platform through MQTT.fx and Test

Download and install the software MQTT.fx.

Step 1: Open the MQTT.fx software and click the setting icon "".

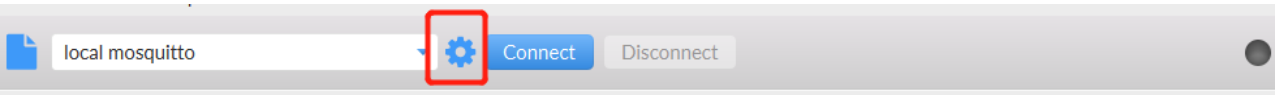

Figure 5: Device Certificate Information

Step 2: Set the connection parameters. The IoT platform currently supports two connection modes, with different settings. See the parameter settings in the table below.

Table 2: Connection Parameters

Name	Input Information
Profile Name	Customized name
Profile Type	MQTT Broker
Broker Address	Indicates the access domain name. Enter the access domain name of the purchased instance. Please enter the instance management page of the Alibaba Cloud IoT platform console to view the connected domain name information in the instance details. The access domain name of the public instance is *${YourProductKey}.iot-as-mqtt.${YourRegionId}.aliyuncs.com*. ● *${YourProductKey}* should be replaced with the ProductKey of the product to which the device belongs (ProductKey can be obtained from the device details page of the IoT platform console). ● See Regions and Availability Zones and replace ${YourRegionId} with a customized region ID.
Broker Port	1883

Client ID	Input the mqttClientId for the underlying protocol message of MQTT Fixed format: ${clientId}\|securemode=3,signmethod=hmacsha1\| A complete example is: *12345*\|*securemode=3,signmethod=hmacsha1*\| ● *${clientId}* is the device ID. It can take any value and the length is within 64 characters. It is recommended to use the MAC address or SN of the device. ● *securemode* is a secure mode, TCP direct connection mode is set *securemode=3*, and TLS direct connection is set *securemode=2*. ● *signmethod* is an algorithm type, supporting hmacmd5 and hmacsha1.
General	The setting items under General can be the default value of the system, or can be set according to specific needs.
User Name	It is composed of DeviceName, symbol (&) and ProductKey. Fixed format: ${YourDeviceName}&${YourProductKey}. A complete example is: device&alxxxxxxxxx.
Password	Download the Password generation tool. Enter https://files.alicdn.com/tpsservice/88413c66e471bec826257781969d1bc7.zip to decompress the download package and double-click the *sign* file. The parameters for generating the password are as follows: ● productKey: Product key to which the device belongs. It can be viewed on the console device details page. ● deviceName: device name. It can be viewed on the console device details page. ● deviceSecret: device secret. It can be viewed on the console device details page. ● timestamp: (optional) timestamp. ● clientId: device ID, consistent with *${clientId}* in the Client ID. ● method: select the type of signature algorithm, which is consistent with the encryption method determined by *signmethod* in Client ID

NOTES

1. After entering the client ID information, do not click "**Generate**".
2. When TCP is connected directly, securemode=3 in Client ID, there is no need to set SSL/TLS information.
3. For TLS direct connection, securemode=2 in the Client ID, you need to set SSL/TLS information.
4. When setting parameters, make sure that there are no spaces in or before or after the parameter value.

An example of setting connection parameter information is shown in the figure below:

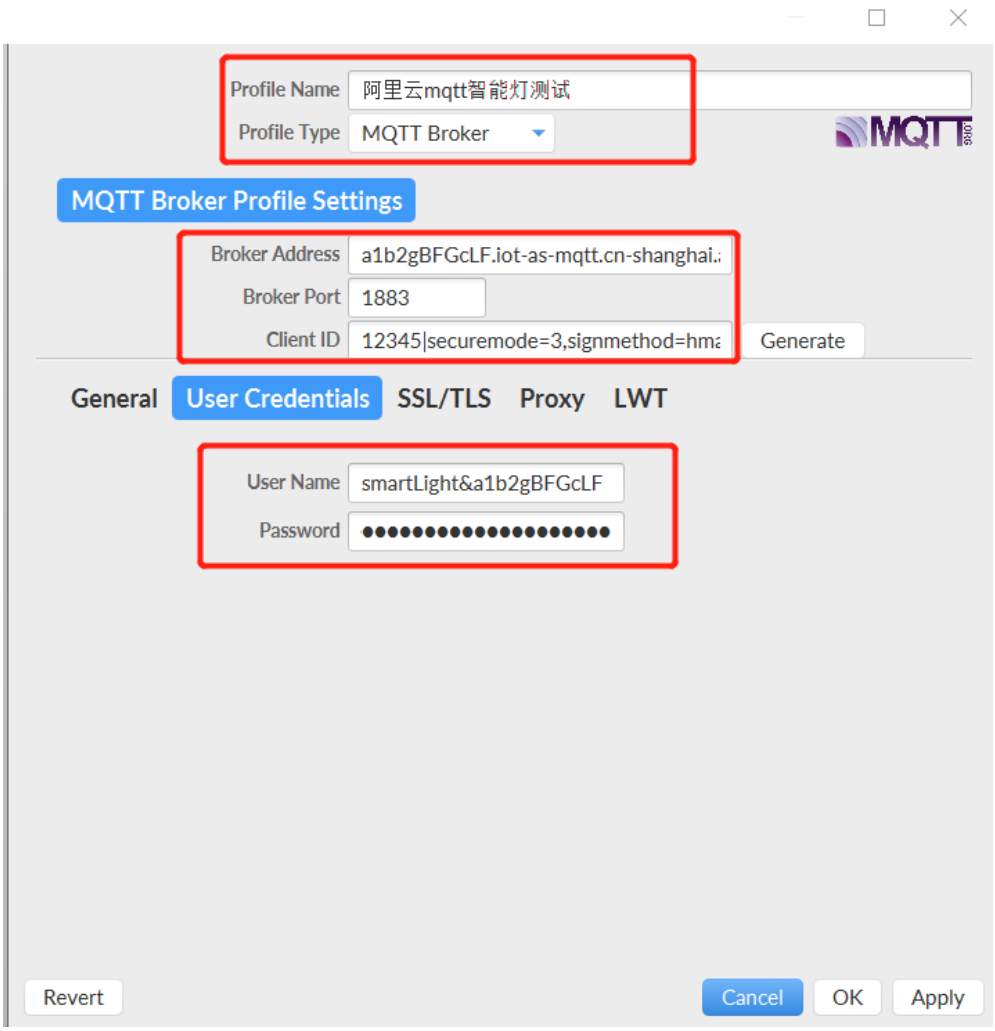

Figure 6: Parameter Configurations

Step 3: After setting the connection parameters, click "**OK**" to confirm the application. Then click the "**Connect**" button to connect to the Alibaba Cloud IoT platform.

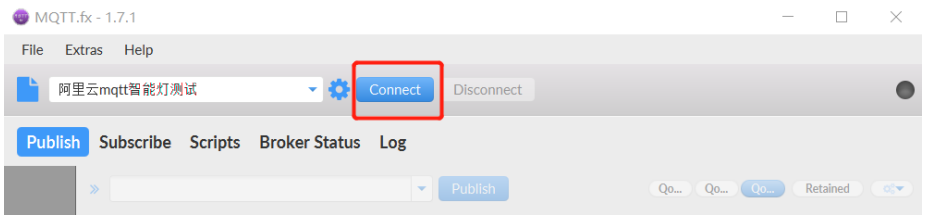

Figure 7: Device Details

4.2.1.1. Data Test

4.2.1.1.1. Downlink Data test

Downlink data test refers to sending messages from the Alibaba Cloud IoT platform and revceiving messages on MQTT.fx. Here is an example to test whether MQTT.fx is successfully connected to the IoT platform.

Step 1: Click "**Subscribe**" of the navigation bar on MQTT.fx.

Step 2: Enter a customized topic for which the device has subscription rights, and click "**Subscribe**" to subscribe to the topic.

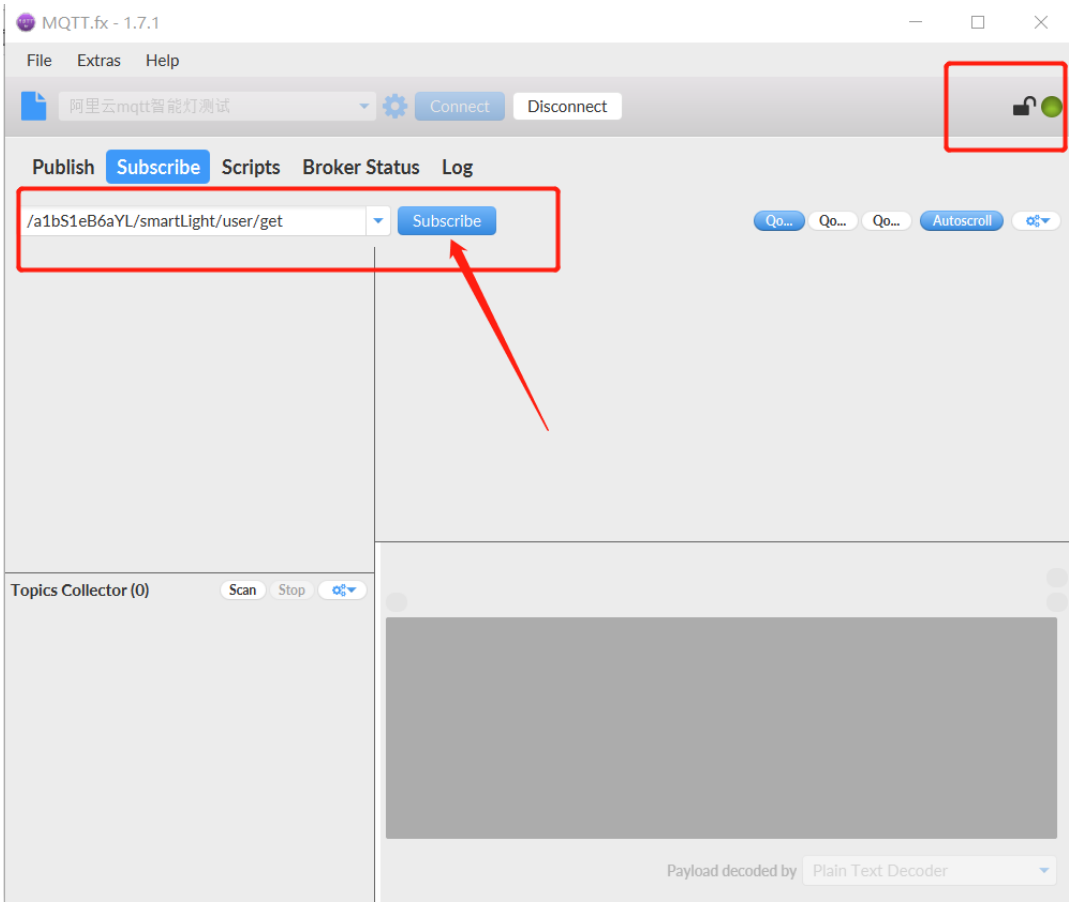

Figure 8: Subscribe to Customized Topic

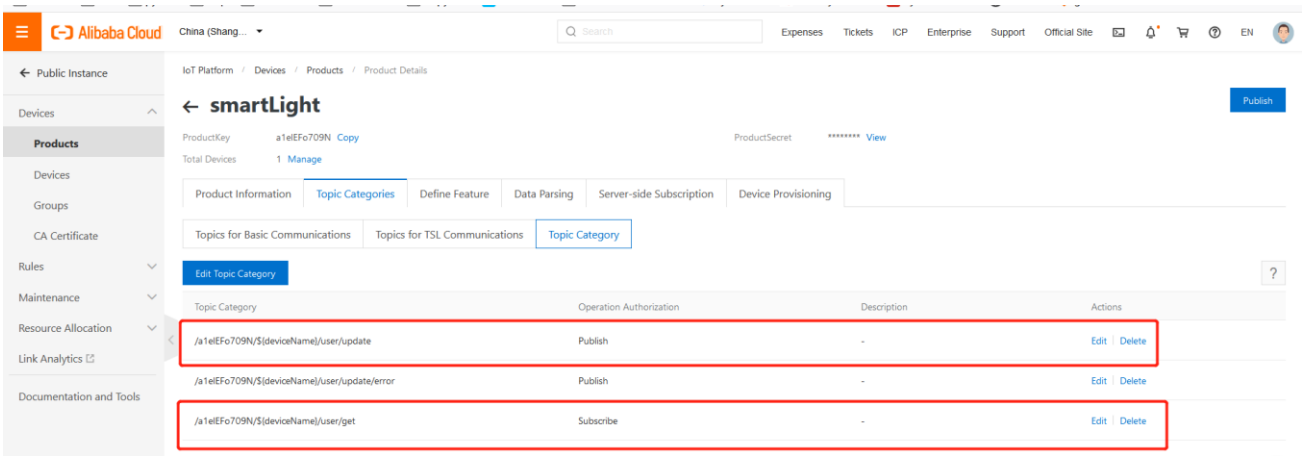

Figure 9: Customized Topic

Step 3: After the subscription is successful, the topic is displayed in the list.

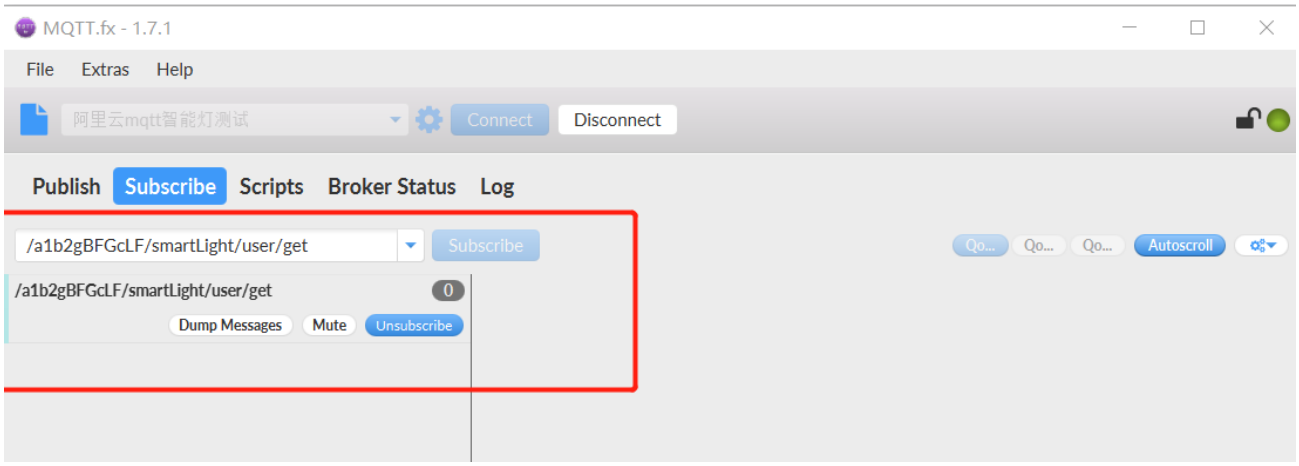

Figure 10: Subscribed Successfully

Step 4: Under the topic list page of the device details of the device in the IoT platform console, click the publish message corresponding to the subscribed topic. Enter the content of the message and click "**OK**".

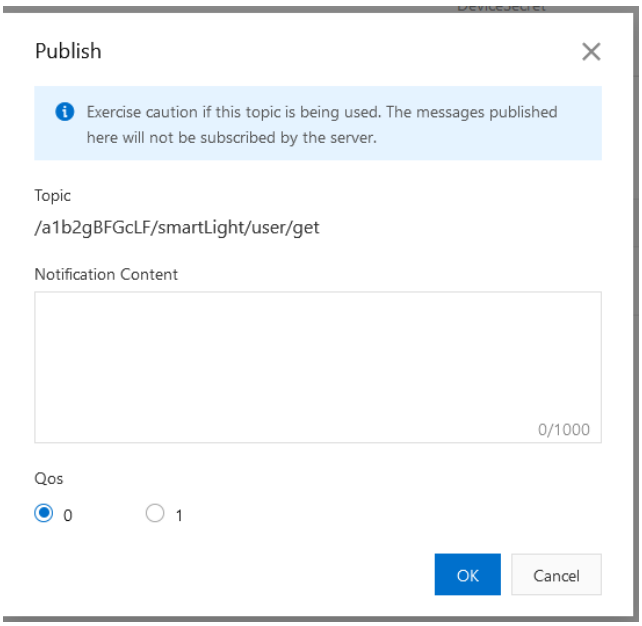

Figure 11: Send a Message from the IoT Platform

Step 5: Log in to the MQTT.fx and check whether the message has been received.

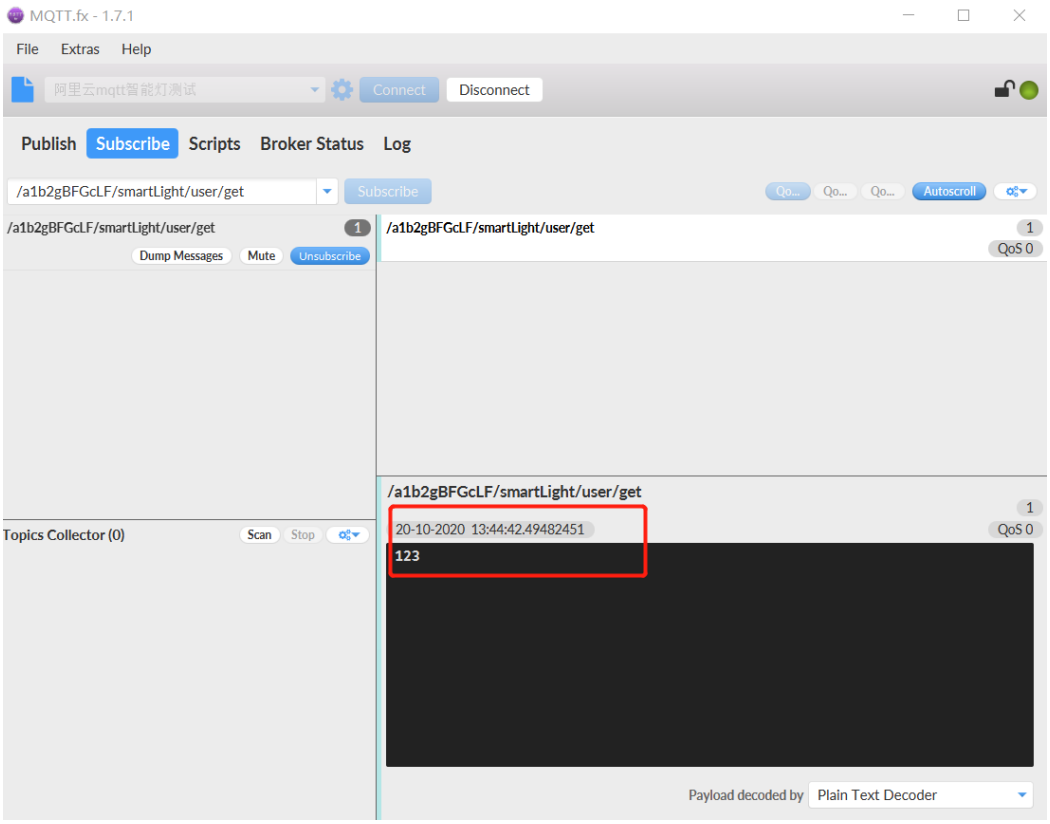

Figure 12: Receive the message sent from the IoT Platform

4.2.1.1.2. Uplink Data Test

Uplink data test refers to sending messages on MQTT.fx and viewing device logs through the IoT platform. Here is an example to test whether MQTT.fx is successfully connected to the IoT platform.

Step 1: Click "**Publish**" of the navigation bar on MQTT.fx.

Step 2: Input a topic for which the device has publishing permission and input the content of the message to be sent, and click "**Publish**" to push a message to this topic.

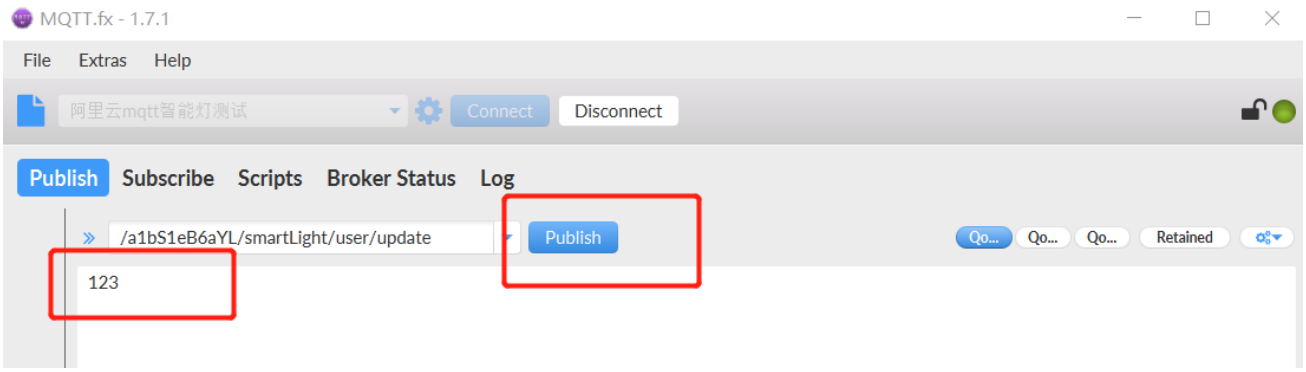

Figure 13: Publish a message to the IoT Platform

Step 3: In the IoT platform console, click "Maintenance", "Device Log", and "Cloud run log" in turn to view the cloud messages of the device.

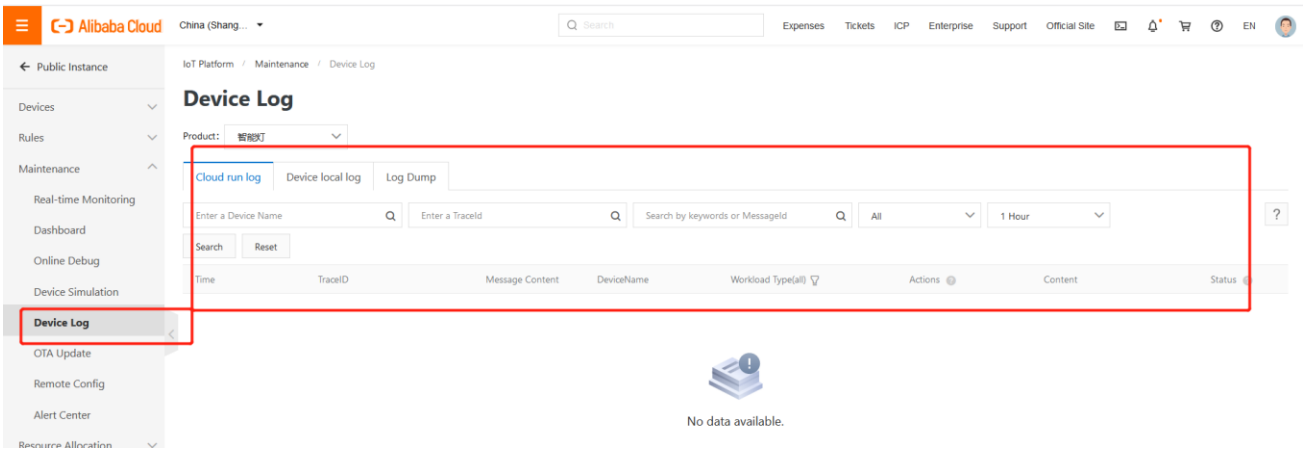

Figure 14: View the Message Content

4.2.2. Connect to the IoT Platform through QuecPython

Run the following code on EC100Y-CN, see *Quectel_QuecPython_Basic_Operation_Instructions* for

details.

```
from aLiYun import aLiYun
import utime

productKey = "a1b2gBFGcLF"         # product key
productSecret = None    # product secret (Input None in unique-certificate-per-device authentication)
DeviceName = "smartLight"         # device name
DeviceSecret = "78a3407e7d43b445cd2dd895cec50ffa"      # device secret (Input None in
unique-certificate-per-product authentication )



# Add Alibaba Cloud connection object
ali = aLiYun(productKey, productSecret, DeviceName, DeviceSecret)

# Configure MQTT connection property
clientID = "12345"    # customized string (within 64)
ali.setMqtt(clientID, clean_session=False, keepAlive=300)



# Callback function
def sub_cb(topic, msg):
       print("subscribe recv:")
       print(topic, msg)

# Configure the callback function
ali.setCallback(sub_cb)

topic = "/a1b2gBFGcLF/smartLight/user/get"    # Topic
# Subscribed topic
ali.subscribe(topic)

topicP="/a1b2gBFGcLF/smartLight/user/update"
# publish a message
ali.publish(topic, "hell world")

# Start to run
ali.start()
utime.sleep(2)
```

The result of running the code are as follows:

1. View the log from the Alibaba Cloud IoT platform.

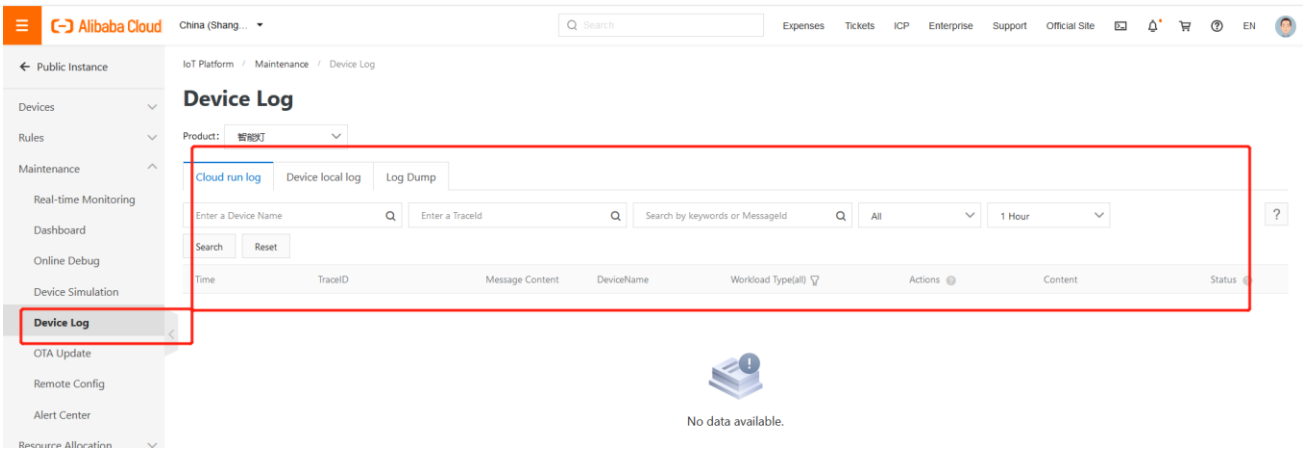

Figure 15: View the Message Content

2. The cloud publishes a message, and the device receives it in the *sub_cb* callback function.

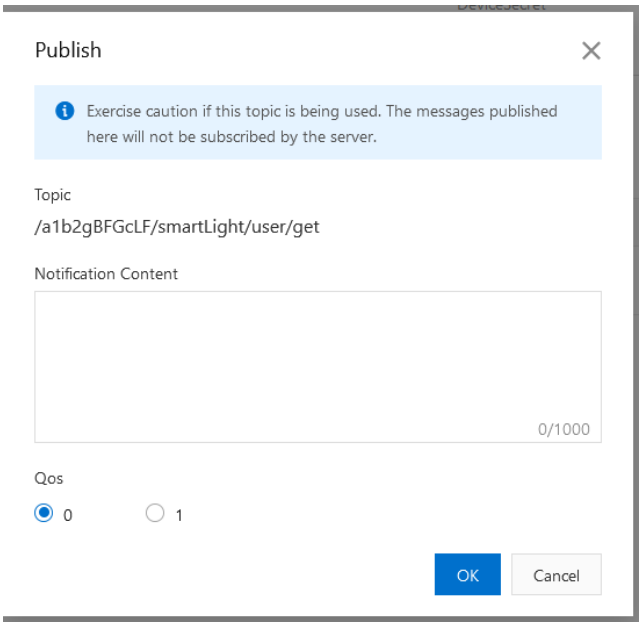

Figure 16: View the Message Content

```
b'/a1b2gBFGcLF/smartLight/user/get' b'wyc mqtt test'
subscribe recv:
b'/a1b2gBFGcLF/smartLight/user/get' b'wyc mqtt test'
```

5 Appendix A References

Table 3: Related Documents

SN	Document Name	Remark
[1]	Quectel_QuecPython_Basic_Operation_Instructions	QuecPython uploading and downloading file introduction

Table 4: Terms and Abbreviations

Abbreviation	Description
API	Application Programming Interface
ID	Mostly refers to Identifier in terms of software
ID²	Internet Device ID
IoT	Internet of Things
MAC	Medium Access Control
MQTT	Message Queuing Telemetry Transport
SN	Serial Number
TCP	Transmission Control Protocol
TLS	Transport Layer Security
TSL	Thing Specification Language