

# BG95 Series AWS IoT Platform Access User Guide

**LPWA Module Series** 

Rev. 1.0

Date: 2020-08-25

Status: Preliminary

www.quectel.com



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: <u>http://www.quectel.com/support/sales.htm</u>.

#### 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.

# COPYRIGHT

THE INFORMATION CONTAINED HERE IS PROPRIETARY TECHNICAL INFORMATION OF QUECTEL WIRELESS SOLUTIONS CO., LTD. TRANSMITTING, REPRODUCTION, DISSEMINATION AND EDITING OF THIS DOCUMENT AS WELL AS UTILIZATION OF THE CONTENT WITHOUT PERMISSION ARE FORBIDDEN. OFFENDERS WILL BE HELD LIABLE FOR PAYMENT OF DAMAGES. ALL RIGHTS ARE RESERVED IN THE EVENT OF A PATENT GRANT OR REGISTRATION OF A UTILITY MODEL OR DESIGN.

Copyright © Quectel Wireless Solutions Co., Ltd. 2020. All rights reserved.



# **About the Document**

# **Revision History**

Version	Date	Author	Description
1.0	2020-08-25	Jaryoung LI	Initial



Со	nten	tSAbout the Document	2
Сог	ntents		3
Tab	le Inde	ЭХ	1
Fig	ure Inc	lex	5
1	Introd	luction	ò
	1.1.	Brief Introduction on AWS IoT	3
2	AWS	IoT Platform Access	7
	2.1.	Register Device and Get Certificates	7
	2	.1.1. Register Device	7
	2	.1.2. Get Certificates	)
	2.2.	Create Policies and Attach	2
	2.3.	Find MQTT Connection Address and Port 17	7
	2.4.	Import Certificates and Connect to AWS IoT Platform 19	9
	2.5.	Use the Device Shadow Service	1
3	Exam	ple	3
	3.1.	Configure the Network	3
	3.2.	Load Certificates	4
	3.3.	Active PDP Context	3
	3.4.	Configure SSL Option	7
	3.5.	Configure MQTT Option	7
	3.6.	MQTT Connection and Data Interaction	7
4	Supp	ort Band List	•
5	Appe	ndix A References	I.



# **Table Index**

Table 1: Support Band List	. 29
Table 2: Related Documents	. 31
Table 3: Terms and Abbreviations	. 31



# **Figure Index**

Figure 1: Communication between AWS IOT and the Device	6
Figure 2: AWS IoT Console	8
Figure 3: Register a Thing	8
Figure 4: Create a Single Thing	9
Figure 5: Add Device to the Thing Registry	9
Figure 6: Add Certificate for Thing	10
Figure 7: Download Certificates	11
Figure 8: Select Root CA	11
Figure 9: Policies	12
Figure 10: Create a Policy	12
Figure 11: Select Certificate	13
Figure 12: Attach Policy	14
Figure 13: Attach Policy to the Certificate	14
Figure 14: Attach Thing	15
Figure 15: Attach Thing to the Certificate	16
Figure 16: Select the Certificate	16
Figure 17: List Attached Things	16
Figure 18: List Attached Policies	17
Figure 19: Things Interface	17
Figure 20: Find MQTT Connection Address	18
Figure 21: MQTT Connection Port	18
Figure 22: Quectel EVB	19
Figure 23: Quectel LTE Windows USB Driver	19
Figure 24: PC serial port	20
Figure 25: AWS Connection Certificates	20
Figure 26: Upload Certificates to Module	21
Figure 27: Configure TLS Option	21
Figure 28: Configure MQTT Option	21
Figure 29: Connect Server	21
Figure 30: Shadow Topics	22
Figure 31: Publish message	22



# **1** Introduction

This document provides users with AWS IoT Cloud platform access method, including how to connect Quectel module to AWS IoT Cloud platform with MQTTS and the related AT command involved in the AWS IoT platform access process.

# 1.1. Brief Introduction on AWS IoT

AWS IoT provides secure, bi-directional communication between Internet-connected devices such as sensors, actuators, embedded micro-controllers, or smart appliances and the AWS Cloud. This enables to collect telemetry data from multiple devices, and store and analyze the data.

AWS IoT Cloud platform supports TLS dual authentication for client certificates, in which MQTT can act as a message broker which provides a secure mechanism for devices and AWS IoT applications to publish and receive messages from each other. After importing certificates into Quectel module, the module can access to AWS IoT Cloud platform through MQTTS.



Devices publish & subscribe Billions of devices can publish and subscribe to messages





Devices communicate AWS IoT Core enables devices to communicate with AWS services and each other

Figure 1: Communication between AWS IOT and the Device

bandwidth requirements



# **2** AWS IoT Platform Access

AWS IoT platform supports TLS client and server certificates authentication, with the MQTT protocol as a message broker. The certificates imported by the module can be used to connect the AWS IoT platform with MQTTS.

# NOTE

Before using AWS IoT services, an AWS account must be created. Please refer to the AWS official link <u>https://docs.aws.amazon.com/iot/latest/developerguide/setting-up.html#aws-registration</u> for details on how to create an AWS account.

# 2.1. Register Device and Get Certificates

Sign in to the AWS IoT platform and register a device in the registry. Certificates will be created in the process of device registration.

# NOTE

The certificates created in the process of device registration to be imported into the module later should downloaded to the local computer.

# 2.1.1. Register Device

1. Navigate to the AWS IoT Console at <u>https://console.aws.amazon.com/iot/home</u>, in the navigation pane, choose Manage, and then choose Things.







	Â	
AWS IoT		
Monitor		
Onboard		Houtes 1
Manage		
Things		
Types	E	
Thing groups		
Billing Groups		
Jobs		
Tunnels		
Greengrass		You don't have any things yet
		A thing is the representation of a device in the cloud
Secure		
Defend		Learn more Register a thing
Act		
Test		

Figure 3: Register a Thing

2. In the Creating AWS IoT things page, click "Create a single thing".



Creating AWS IoT things	
An IoT thing is a representation and record of your physical device in the cloud. Any physical device needs a thing record in order to work with AWS IoT. Learn more.	
Register a single AWS IoT thing Create a thing in your registry	Create a single thing
Bulk register many AWS IoT things Create things in your registry for a large number of devices already using AWS IoT, or register devices so they are ready to connect to AWS IoT.	Create many things
Cancel	Create a single thing

# Figure 4: Create a Single Thing

3. Then add your device to the thing registry according to the provided steps. Take the device name MyIoTDevice as an example:

Add your device to the	thing registry	STEP 1/3
This step creates an entry in the thing re	egistry and a thing shadow for your device.	
Name		
MyloTDevice		
Apply a type to this thing		
	nagement by providing consistent registry data for things that share a t	ype. Types provide things with a
Using a thing type simplifies device man common set of attributes, which describ	be the identity and capabilities of your device, and a description.	
Using a thing type simplifies device mar common set of attributes, which describ Thing Type	be the identity and capabilities of your device, and a description.	

Figure 5: Add Device to the Thing Registry



# 2.1.2. Get Certificates

X.509 certificates protects the connection between device and AWS IoT platform. The certificates have to be activated before using.

1. On the Add a certificate for your thing page, under One-click certificate creation, choose Create certificate.

CREATE A THING Add a certificate for your thing	STEP 2/3
A certificate is used to authenticate your device's connection to AWS IoT.	
<b>One-click certificate creation (recommended)</b> This will generate a certificate, public key, and private key using AWS IoT's certificate authority.	Create certificate
Create with CSR Upload your own certificate signing request (CSR) based on a private key you own.	▲ Create with CSR
Use my certificate Register your CA certificate and use your own certificates for one or many devices.	Get started
Skip certificate and create thing You will need to add a certificate to your thing later before your device can connect to AWS IoT.	Create thing without certificate

# Figure 6: Add Certificate for Thing

 Download certificates, keys and root CA and save them in your PC. If your device supports it, you should select the RSA 2048 bit key: Amazon Root CA 1 These are cross-signed by Starfield. <u>https://docs.aws.amazon.com/iot/latest/developerguide/server-authentication.html</u>.



# Certificate created!

Download these files and save them in a safe place. Certificates can be retrieved at any time, but the private and public keys cannot be retrieved after you close this page.

#### In order to connect a device, you need to download the following:

A certificate for this thing	f2cec8702d.cert.pem	Download			
A public key	f2cec8702d.public.key	Download			
A private key	f2cec8702d.private.key	Download			
You also need to download a root CA for AWS IoT: A root CA for AWS IoTDownload					

Activate

#### **Figure 7: Download Certificates**

# CA certificates for server authentication

Depending on which type of data endpoint you are using and which cipher suite you have negotiated, AWS IoT Core server authentication certificates are signed by one of the following root CA certificates:

#### VeriSign Endpoints (legacy)

• RSA 2048 bit key: VeriSign Class 3 Public Primary G5 root CA certificate 🗹

Amazon Trust Services Endpoints (preferred)

#### Note

You might need to right click these links and select Save link as... to save these certificates as files.

#### • RSA 2048 bit key: Amazon Root CA 1 🗹.

- RSA 4096 bit key: Amazon Root CA 2. Reserved for future use.
- ECC 256 bit key: Amazon Root CA 3 🗹.
- ECC 384 bit key: Amazon Root CA 4. Reserved for future use.

These certificates are all cross-signed by the Starfield Root CA Certificate 2. All new AWS IoT Core regions, beginning with the May 9, 2018 launch of AWS IoT Core in the Asia Pacific (Mumbai) Region, serve only ATS certificates.

Figure 8: Select Root CA



# 2.2. Create Policies and Attach

1. Create a policy.

In the AWS IoT console page, select "**Secure**" in the left navigation bar, click "**Policies**" and "**Create**" to create a policy. In the "Create a policy" page, input policy name, take policy name MyIoTDevicePolicy as an example:

AWS IoT	Policies	
Monitor  Monboard	Search policies Q	C
<ul> <li>Manage</li> <li>Greengrass</li> </ul>	mydevice1-Policy gsm_iot_policy cyz_test_001-Policy cs_iot_aws_policy_2	
Secure Certificates Policies CAs	c.s_iot_policy_aws_ca c.s_iot_policy My_lot_Policy MylotThing-1	
Role Aliases Authorizers Defend	MylotPolicy-2 MyGreengrass_Core-p Default 01234cd72d1bac290	

# Figure 9: Policies

Create a policy	
Create a policy to define a set of authorized actions. You can authorize actions on one or more resources (things, topics, topic filte more about IoT policies go to the AWS IoT Policies documentation page. Name MyIoTDevicePolicy	ers). To learn
Add statements Policy statements define the types of actions that can be performed by a resource.	Advanced mode
Action	
*	
Resource ARN	
*	
Effect	
Allow Deny	Remove

Figure 10: Create a Policy



# NOTE

This policy grants unrestricted access for all iot operations, and is to be used only in a development environment. For non-dev environments, all devices in your fleet must have credentials with privileges that authorize intended actions only, which include (but not limited to) AWS IoT MQTT actions such as publishing messages or subscribing to topics with specific scope and context. The specific permission policies can vary for your use cases. Identify the permission policies that best meet your business and security requirements.

For sample policies, refer to

https://docs.aws.amazon.com/iot/latest/developerguide/example-iot-policies.html. Also refer to https://docs.aws.amazon.com/iot/latest/developerguide/security-best-practices.html.

2. Attach the policy to a certificate.

In the AWS IoT console page, select "**Secure**" in the left navigation bar, click "**Certificates**", choose a certificate:



Figure 11: Select Certificate

Click "..." to open the drop-down menu and click "Attach policy":



AWS IoT	Certificates				
Monitor	Search certificates	Q			
Onboard					
Manage	f2cec8702c	01110bfbeb83f2c179	 056766295328ac71d	 feb4756fcf6b6ce4e25	
Greengrass	ACTIVE	ACTIVE	ACTIVE	ACTIVE	
▼ Secure		Revoke			
Certificates	575ea7deC	transfer cb0ef7846b9fb4f55c	d553ece06b13a6d9d	19d74a7aaeee27cf66	
Policies	ACTIVE Reject	transfer ACTIVE	ACTIVE	ACTIVE	
CAs	Revoke				
Role Aliases	Start	transfer			
Authorizers	ACTIVE Atta	ACTIVE 43529eadad42baet39	e74d1280c106a5e85 ACTIVE	ab94c9d05397940e3 Active	
Defend	D	ownload			
▶ Act	c02baa970360019	Delete 51cacdcb29942d252f	f286c188c42c04e1e1	63a54ff3351c705229	
Test					

Figure 12: Attach Policy

Choose the policy you have created previously and click "Attach":

# Attach policies to certificate(s)

Policies will be attached to the following certificate(s):

f2cec8702d1c8038257515745e832ffa86cd65c2753b501ebcaacbe386666b6d

#### Choose one or more policies

Q Search policies	
My_lot_Policy	View ^
MylotThing-1	View
MylotPolicy-2	View
MyGreengrass_Core-policy	View
Default	View
01234cd72d1bac2901-Policy	≡ View
VyloTDevicePolicy	View

1 policy selected	Cancel	Attach

#### Figure 13: Attach Policy to the Certificate



3. Attach the certificate to a thing.

Click "..." of the specified certificate to open the drop-down menu and click "Attach thing":

AWS IoT	Certificate	s			
Monitor					
Onboard	Search certifica	ites	Q		
Manage	f2cec8702c	 Activate	01110bfbeb83f2c179	056766295328ac71d	feb4756fcf6b6ce4e25
Greengrass	ACTIVE	Deactivate	ACTIVE	ACTIVE	ACTIVE
▼ Secure	-	Revoke			
Certificates	575ea7de0	Accept transfer	cb0ef7846b9fb4f55c	d553ece06b13a6d9d	19d74a7aaeee27cf66
Policies	ACTIVE		ACTIVE	ACTIVE	ACTIVE
CAs		Revoke transfer			
Role Aliases	761.0.0	Start transfer			
Authorizers	a36de2e2a	Attach thing	43529eadad42baet39 Active	e74d1280c106a5e85 ACTIVE	ab94c9d05397940e3 Active
Defend		Download			
▶ Act	c02baa970	Delete 3600199a	51cacdcb29942d252f	f286c188c42c04e1e1	63a54ff3351c705229
Test					

Figure 14: Attach Thing

Choose the thing you have created previously and click "Attach":

Attach things to certificate(s)
Things will be attached to the following certificate(s): f2cec8702d1c8038257515745e832ffa86cd65c2753b501ebcaacbe386666b6d
Choose one or more things
Q Search things
MyloTDevice
□ 01234cd72d1bac2901
gsm_iot
012368c4532d8d9c01
testDev_1
cs_iot_dev_2-1
cs_iot_aws_ca_1
1 thing selected Cancel Attach



# Figure 15: Attach Thing to the Certificate

#### 4. Verification.

To verify whether the policy and thing are attached successfully, select the certificate to list all attached policies and things.

AWS IoT	Certificates			
Monitor	Search certificates	0		
Onboard		~		
Manage	f2cec8702d1c803825	 01110bfbeb83f2c179	 056766295328ac71d	 feb4756fcf6b6ce4e25
Greengrass	ACTIVE	ACTIVE	ACTIVE	ACTIVE
▼ Secure				
Certificates	575ea7de0d0f09e928	cb0ef7846b9fb4f55c	d553ece06b13a6d9d	19d74a7aaeee27cf66
Policies	ACTIVE	ACTIVE	ACTIVE	ACTIVE
CAs				
Role Aliases	a36de2e2a6c2808b6	43529eadad42baef39	e74d1280c106a5e85	••• ab94c9d05397940e3
Authorizers	ACTIVE	ACTIVE	ACTIVE	ACTIVE
Defend				
▶ Act	c02baa970360d199a	51cacdcb29942d252f	•••• f286c188c42c04e1e1 Астіvе	63a54ff3351c705229
Test				

# Figure 16: Select the Certificate

	d1c80382575	6d
ACTIVE		Actions -
Details	Things	
Policies Things	MyloTDevice	

#### Figure 17: List Attached Things



Certificates > f2cec8702d	Ic80382575	
CERTIFICATE f2cec8702d1	c8038257515745e832ffa86cd65c2753b501ebcaacbe386666b6d	
ACTIVE		Actions -
Details	Policies	
Policies Things Non-compliance	MyIoTDevicePolicy	



# 2.3. Find MQTT Connection Address and Port

Find the MQTT connection server as below:

AWS IoT	Things	Create
Monitor		
Onboard	Search things Q Fleet Indexing Info	List 💌
▼ Manage	E Name T	уре
Types	MyloTDevice	0 TYPE ••••
Thing groups Billing Groups	01234cd72d1bac2901	0 TYPE ***
Jobs	gsm_jot N	0 TYPE •••
<ul> <li>Greengrass</li> </ul>	012368c4532d8d9c01	0 TYPE •••
	terflow 1 a	0 TVDE ++++

# Figure 19: Things Interface

Choose the things you just created and choose Interact.



Things > MyloTDevice

THING MyloTDevice NO TYPE	Actions -
Details Security Thing groups Billing Groups Shadows	This thing already appears to be connected.       Connect a device         HTTPS       Update your Thing Shadow using this Rest API Endpoint. Learn more         affect14colfeen at a jett us wort 1 contractory acres
Interact Activity Jobs Violations Defender metrics	MQTT Use topics to enable applications and things to get, update, or delete the state information for a Thing (Thing Shadow) Learn more

#### Figure 20: Find MQTT Connection Address

The MQTTS connection port supported by AWS IoT platform is shown as below.

# Protocols, Port Mappings, and Authentication

The following table shows each protocol supported by AWS IoT, the authentication method, and port used for each protocol.

Protocol, Authentication, and Port Mappings				
Protocol	Authentication	Port	ALPN ProtocolName	
MQTT	X.509 client certificate	8883, 443 <sup>†</sup>	x-amzn-mqtt-ca	
HTTPS	X.509 client certificate	8443, 443 <sup>†</sup>	x-amzn-http-ca	
HTTPS	SigV4	443	N/A	
MQTT over WebSocket	SigV4	443	N/A	

Figure 21: MQTT Connection Port



# 2.4. Import Certificates and Connect to AWS IoT Platform

Quectel EVB can test the connection between AWS IoT platform and the module. The EVB has antenna, SIM card and USB power as the figure below shows. Such as BG95-M3 can use GSM/CAT-M/NB-IoT network. The USB port can be a virtual port to connect with the serial port tool. Please install the *Quectel\_LTE\_Windows\_USB\_Driver* in your PC first. Then you can use the *QCOM* tool to communicate with the EVB.



# Figure 22: Quectel EVB

Quectel_LTE_Windows_USB_Driver_V2.1.6			
Include in library	folder		
Name	Date modified	Туре	Size
🛎 setup.exe	2019/4/2 17:15	Application	12,994 KB

# Figure 23: Quectel LTE Windows USB Driver

Select Setup.exe to install *Quectel\_LTE\_Windows\_USB\_Driver*, after install finished PC serial port as below:



Ports (COM & LPT)	
	DM1)
	M83)
	COM82)
	main serial port
USB Serial Port (COM32)	

# Figure 24: PC serial port

Import certificates into the module and connect the module to AWS IoT platform with AT command related to MQTTS. The process is shown as below.

Step 1: Upload the certificates into the module

🔄 6d1fa87feb-certificate.pem.crt	2020/1/20 14:48	安全证书	2 KB	
6d1fa87feb-private.pem.key -	2020/1/20 14:49	KEY 文件	2 KB	• client.pem
6d1fa87feb-public.pem.key	2020/1/20 14:48	KEY 文件	1 kb user	kev. pem
AmazonRootCA1.pem	2020/1/20 15:00	PEM 文件	2 KB	
			cacert.pe	em

# Figure 25: AWS Connection Certificates

Upload the certificates into the module:

QCOM_V1.6				_ 0 %
About				
COM Port Setting	Command List			
COM Port: 84 - Baudrate: 115200 - StopBits: 1 - Parity: None -	Choose All Commands	нех 🗆	Enter	Delay(mS)
	□ 1: AT+QFUPL="cacert.pem",1446,100	ন 🕇	1	8096
Byte Size: 8 V Flow Control: No Ctrl Flow V Close Port	2: AT+QFUPL="client.pem",1220,100	<b>v</b>	2	8096
	3: AT+QFUPL="user_key1.pem" 1679,100	<b>I P</b>	3	8096
AT+QFUPL="cacert.pem".1446.100	4: AT+QSSLCFG="cacert",2,"cacert.pem"		4	400
CONNECT	5: AT+QSSLCFG="clientcert",2,"client.pem"		5	400
*GFUFL. 1440,3030	6: AT+QSSLCFG="clientkey",2,"user_key1.pen		6	400
OK AT+QFUPL="client.pem".1220.100	7: AT+QSSLCFG="ciphersuite",2,0XFFFF		7	400
CONNECT	8: AT+QSSLCFG="sslversion",2,4		8	400
+QFUFL. 1220,3023	9: AT+QSSLCFG="seclevel",2,2		9	400
OK AT+QFUPL="user kev1.pem": 1679.100	10: AT+QMTCFG="ssl",2,1,2		10	400
CONNECT	11: AT+QMTCFG="version",2,4		11	400
+QFUPE: 1673,3820	12: AT+QMTOPEN=2,"af5c7l4oo3fcr-ats.iot.us-w		12	400
ок	T13: AT+QMTCONN= 2,"MyloTDevice"		13	3000
	V 14: AT+QFDEL="cacert.pem"		14	3000
	15: AT+QFDEL="client.pem"		15	400
	I6: AT+QFDEL="user_key1.pem"		16	1000
	□ 17: □		17	400
	□ 18: □		18	400
	□ 19: □		19	400
	20:		20	400
[2020-08-05_16:06:48:407] DCD:0 CTS:1 RI:0	21:		21	
[2020-08-05_16:06:52:275] This File Size is 1679 Bytes [2020-08-05_16:06:54:437] DCD:1 CTS:1 RI:0	22:		22	
[2020-08-05_16:06:55:719] DCD:0 CTS:1 RI:0	23:		23	
	24:		24	
	25:		25	
Clear Information	26:		26	
Input String: HEX String Show In HEX V Send With Enter	27:		27	
	28:		28	
Send Command	29:		29	
②Select file Select File D:\360MoveData\Users\Janyoung\Desktop\新建: Send 100.0% Send File	Load Test Script Clear All Commands	R Delay	un Times Time(mS)	: 10000



## Figure 26: Upload Certificates to Module

Step 2: Configure TLS option:

```
AT+QSSLCFG="cacert",2,"cacert.pem"
OK
AT+QSSLCFG="clientcert",2,"client.pem"
OK
AT+QSSLCFG="clientkey",2,"user_key1.pem"
OK
AT+QSSLCFG="ciphersuite",2,0XFFFF
OK
AT+QSSLCFG="sslversion",2,4
OK
AT+QSSLCFG="sslversion",2,4
OK
AT+QSSLCFG="sslversion",2,2
OK
AT+QMTCFG="ssl",2,1,2
OK
```

# Figure 27: Configure TLS Option

**Step 3**: Configure MQTTS and MQTT protocol version

AT+QMTCFG="ssl",2,1,2 OK AT+QMTCFG="version",2,4 OK

# Figure 28: Configure MQTT Option

Step 4: Connect to AWS IoT with MQTTS.

AT+QMTOPEN=2,"af5c7l4oo3fcr-ats.iot.us-west-2.amazonaws.com",8883 OK +QMTOPEN: 2,0 AT+QMTCONN= 2,"MyloTDevice" OK +QMTCONN: 2,0,0

Figure 29: Connect Server

# 2.5. Use the Device Shadow Service

Use topics to enable applications and things to get, update, or delete the state information for a Thing (Thing Shadow).



Refer to https://docs.aws.amazon.com/iot/latest/developerguide/device-shadow-mqtt.html

# **Shadow topics**

The topics in this section are used by named and unnamed shadows. The topics used by each differ only in the topic prefix. This table shows the topic prefix used by each shadow type.

ShadowTopicPrefix value	Shadow type
<pre>\$aws/things/thingName/shadow</pre>	Unnamed (classic) shadow
<pre>\$aws/things/thingName/shadow/name/shadowName</pre>	Named shadow

To create a complete topic, select the *ShadowTopicPrefix* for the type of shadow to which you want to refer, replace *thingName*, and *shadowName* if applicable, with their corresponding values, and then append that with the topic stub as shown in the following table. Remember that topics are case sensitive.

# Figure 30: Shadow Topics

Step 5: Subscribe and publish messages:

AT+QMTPUB=2,1,1,0,"\$aws/things/MyloTDevice/shadow/get/accepted" > Hello AWS IoT! OK +QMTPUB: 2,1,0 +QMTRECV: 2,1,"\$aws/things/MyloTDevice/shadow/get/accepted","Hello AWS IoT!"

Figure 31: Publish message



# **3** Example

This chapter provides examples for AWS IoT platform access authentication. The following shows the whole process of accessing to AWS IoT with MQTTS.

# 3.1. Configure the Network

//AT+QCFG='	"nwscan	mode",[, <scanmo< td=""><td>ude&gt;]</td></scanmo<>	ude>]
// <scanmode></scanmode>	>: 0	Automatic	
//	1	GSM only	
//	3	LTE only	
AT+QCFG="r	nwscan	mode",3	//Configure the scan mode to LTE only.
ок			
//AT+QCFG=	"iotopmo	ode"[, <mode>]</mode>	
// <mode></mode>	Number	format. Network	category to be searched under LTE RAT.
//	0 eM	ТС	
//	1 NB-	-loT	
//	2 eM	TC and NB-loT	
AT+QCFG="i	otopmo	ode",1	//Configure the network to NB-IoT.
ОК			
//AT+QCFG='	"band"[,·	<gsmbandval>,<e< td=""><td>mtcbandval&gt;,<nbiotbandval>]</nbiotbandval></td></e<></gsmbandval>	mtcbandval>, <nbiotbandval>]</nbiotbandval>
// <gsmbandva< td=""><td>al&gt;: A h</td><td>exadecimal value</td><td>that specifies the GSM frequency band. If it is set to 0, it means</td></gsmbandva<>	al>: A h	exadecimal value	that specifies the GSM frequency band. If it is set to 0, it means
	not	to change GSM fi	requency band.
// <emtcbandv< td=""><td colspan="3">//<emtcbandval>: A hexadecimal value that specifies the eMTC frequency band. If it is set to 0 or</emtcbandval></td></emtcbandv<>	// <emtcbandval>: A hexadecimal value that specifies the eMTC frequency band. If it is set to 0 or</emtcbandval>		
	0x4	0000000, it mean	s not to change the frequency band
// <nbiotbandv< td=""><td>al&gt;: A h</td><td>exadecimal value</td><td>that specifies the NB-IoT frequency band. If it is set to 0 or</td></nbiotbandv<>	al>: A h	exadecimal value	that specifies the NB-IoT frequency band. If it is set to 0 or
	0x4	0000000, it mean	s not to change the frequency band
//For the supp	orted ba	and list, please ref	ier to <b>Chapter 4</b> .
AT+QCFG="l	oand",0	,0,10	//Configure NB-IoT network BAND 5, the hexadecimal value
			is 0x10, only need input 10.
ОК			
AT+CEREG?	;+QNW	INFO;+QCSQ	//Query the network status.
+CEREG: 0,1			//Registered NB-IoT network



+QNWINFO: "CAT-NB1","46011","LTE BAND 5",2506

+QCSQ: "CAT-NB1",-80,-94,103,-15

ок

# 3.2. Load Certificates

//If the module already has certificates, please delete the certificates with AT+QFDEL first.

AT+QFDEL="cacert.pem"

ΟΚ

AT+QFDEL="client.pem"

ΟΚ

AT+QFDEL="user\_key1.pem"

ΟΚ

//The server certificate size is 1446 bytes, the timeout is 5000 s, upload the certificates after echoing **CONNECT**.

# AT+QFUPL="cacert.pem",1446,5000

# CONNECT

-----BEGIN CERTIFICATE-----

MIID7zCCAtegAwIBAgIBADANBgkghkiG9w0BAQsFADCBmDELMAkGA1UEBhMCVVMx EDAOBgNVBAgTB0FyaXpvbmExEzARBgNVBAcTCINjb3R0c2RhbGUxJTAjBgNVBAoT HFN0YXJmaWVsZCBUZWNobm9sb2dpZXMsIEluYy4xOzA5BgNVBAMTMIN0YXJmaWVs ZCBTZXJ2aWNIcyBSb290IENIcnRpZmljYXRIIEF1dGhvcml0eSAtIEcyMB4XDTA5 MDkwMTAwMDAwMFoXDTM3MTIzMTIzNTk1OVowgZgxCzAJBgNVBAYTAIVTMRAwDgYD VQQIEwdBcml6b25hMRMwEQYDVQQHEwpTY290dHNkYWxIMSUwIwYDVQQKExxTdGFy ZmllbGQgVGVjaG5vbG9naWVzLCBJbmMuMTswOQYDVQQDEzJTdGFyZmllbGQgU2Vy dmljZXMgUm9vdCBDZXJ0aWZpY2F0ZSBBdXRob3JpdHkgLSBHMjCCASIwDQYJKoZI hvcNAQEBBQADggEPADCCAQoCggEBANUMOsQq+U7i9b4Zl1+OiFOxHz/Lz58gE20p OsgPfTz3a3Y4Y9k2YKibXlwAgLlvWX/2h/klQ4bnaRtSmpDhcePYLQ1Ob/blSdm2 8xpWriu2dBTrz/sm4xq6HZYuajtYIIIHVv8loJNwU4PahHQUw2eeBGq6345AWh1K Ts9DkTvnVtYAcMtS7nt9rjrnvDH5RfbCYM8TWQIrgMw0R9+53pBlbQLPLJGmpufe hRhJfGZOozptqbXuNC66DQO4M99H67FrjSXZm86B0UVGMpZwh94CDkIDhbZsc7tk 6mFBrMnUVN+HL8cisibMn1IUaJ/8viovxFUcdUBgF4UCVTmLfwUCAwEAAaNCMEAw DwYDVR0TAQH/BAUwAwEB/zAOBgNVHQ8BAf8EBAMCAQYwHQYDVR0OBBYEFJxfAN+q AdcwKzilorhtSpzyEZGDMA0GCSqGSIb3DQEBCwUAA4IBAQBLNqaEd2ndOxmfZyMI bw5hyf2E3F/YNoHN2BtBLZ9g3ccaaNnRbobhiCPPE95Dz+I0swSdHynVv/heyNXB ve6SbzJ08pGCL72CQnqtKrcqfU28eIUSwhXqvfdqlS5sdJ/PHLTyxQGjhdByPq1z qwubdQxtRbeOlKyWN7Wg0l8VRw7j6IPdj/3vQQF3zCepYoUz8jcl73HPdwbeyBkd iEDPfUYd/x7H4c7/I9vG+o1VTqkC50cRRj70/b17KSa7qWFiNyi2LSr2EIZkyXCn 0q23KXB56jzaYyWf/Wi3MOxw+3WKt21gZ7leyLnp2KhvAotnDU0mV3HaIPzBSICN



# sSi6

-----END CERTIFICATE-----+QFUPL: 1188,2d13

OK

//The client certificate size is 1220 bytes, the timeout is 5000 s, upload the certificates after echoing **CONNECT**.

# AT+QFUPL="client.pem",1220,5000

# CONNECT

-----BEGIN CERTIFICATE-----

MIIDWTCCAkGgAwIBAgIUEU8Sdtdxv7TSMa+qSJctGwP/ef4wDQYJKoZIhvcNAQEL BQAwTTFLMEkGA1UECwxCQW1hem9uIFdlYiBTZXJ2aWNlcyBPPUFtYXpvbi5jb20g SW5jLiBMPVNIYXR0bGUgU1Q9V2FzaGluZ3RvbiBDPVVTMB4XDTlwMDgwNTA2MTgz NIoXDTQ5MTIzMTIzNTk1OVowHjEcMBoGA1UEAwwTQVdTIElvVCBDZXJ0aWZpY2F0 ZTCCASIwDQYJKoZIhvcNAQEBBQADggEPADCCAQoCggEBANyZlfrksZfD2Wz07SHU BMSGhxABZFmJeW8+/R1j88imNG+EK32pDiS6foD1zoYkpZF5bZyqSgn+GW4WYLPB yU2AAG2lewsF9eOXc87Nk4GdtS+1Qrbjf+JkcmrDMHDFq6ugzEhjSP3CHMIU5SJy TJTdY0HxuR+aQbty8d6zyqNe6nFLe8NuUbQ7AmSMAtJQDODTPsVGv3dzbO6FI3TR IQCCp33sLZ4AYhRE9y+jhJZ0hHk/ehe9D68sbeWOfbAkFySCmIDFdBGLIaQ6z+jW 7uOxZhBaxbxoS1dqT7j3lNqsoR59hrwz8lKD29HQPRrjSFv1x+nSV/tfUAl/c+Tn 0psCAwEAAaNgMF4wHwYDVR0jBBgwFoAUoJJ7YR4/P2r0/vbPNTRM7EXwcoAwHQYD VR0OBBYEFOrBYZ82vY5gWGR2AoWDRLvmUHpyMAwGA1UdEwEB/wQCMAAwDgYDVR0P AQH/BAQDAgeAMA0GCSqGSIb3DQEBCwUAA4IBAQCeSbUzBjF9wVN8x+a0F7Zjtq0Z /CJAK6g+iNOcSpXBCXeER0mjijFfiGQUN0By6kTf67yv5RS3yZZtFS+3g62psf7g LDWm/vnClp4B4ZehNiGyZHfC5X+mN9HCe2ej+npukHBwPFgzf0e0PsY25LrnNTtW sRcWzQAdG1YJuk0u4ai+NcThYrJKNcNj2BHRk1+rZiSRp19Jwzks6xaJrYAT2zcx tcT5gJql6yCq1HSmF603JxksSlYFdSSc/Yx+7O5zR5g/sNehh/BL43SZFAhly2r1 tOg3g6iwlLrKD0s3/VdoK8p8xfZwF8MksKpKi14LKwAb2eGnCAhgfB8JKvjK -----END CERTIFICATE-----

+QFUPL: 1224,7a6c

# OK

//The client key size is 1679 bytes, the timeout is 5000 s, upload the client key after echoing **CONNECT**. **AT+QFUPL="user\_key1.pem",1679,5000** 

# CONNECT

-----BEGIN RSA PRIVATE KEY-----

MIIEpAIBAAKCAQEA3JmV+uSxl8PZbPTtldQExIaHEAFkWYl5bz79HWPzyKY0b4Qr fakOJLp+gPXOhiSlkXltnKpKCf4ZbhZgs8HJTYAAbYh7CwX145dzzs2TgZ21L7VC tuN/4mRyasMwcMWrq6DMSGNI/clcyVTIInJMIN1jQfG5H5pBu3Lx3rPKo17qcUt7 w25RtDsCZIwC0IAM4NM+xUa/d3Ns7oUjdNEhAIKnfewtngBiFET3L6OEInSEeT96 F70Pryxt5Y59sCQXJIKYgMV0EYshpDrP6Nbu47FmEFrFvGhLV2pPuPeU2qyhHn2G vDPyUoPb0dA9GuNIW/XH6dJX+19QCX9z5OfSmwIDAQABAoIBAB7Zh6TfiIptxCE9



m0/ow4XsPkUZvLvcwtOw3IrW2IfN3nVd6WsYUjcGsZw9Q9V0mlbgkVigY9xD4bCI hpt81Tb2WJj4xiRDgCCE1JguHZu5v1XdpmfHJul1I98UTtjme9xzjz3sTI/YLohl S+nmTBlhYHkfYcEupFSffX2kbQktgWotYbUX9f9h2KIXAyvASekUz6EgyZXsHRV2 963uuJgZl2kYinQ9zaku01x14qtMIFfAZRLI/xZol4Qr65d+582FO1+EjZmffV55 TfV2a2wnE/RtGRQyq9Z8I2EPOsP8DuHogsHZwAV7761Z5n7DqF4DoKanR7W8SqCV zH2FbjkCgYEA7g4DlpxsrG8sH3nnUeb5V09Skdy1i1ldRRsGELTpyfXRFKnYsEIA e51GMNd19L5Q+7e87LpjBZLKsV3DwIFNiRIwS6YHp6b2WuOxVkwHnsthqLe+zfac xmi/WFCkEcwGEoCkEW55uut2X1UzZ8zt6ZTGuiREYa1mp3MYo2u/MIUCgYEA7Tq6 j4bEjhOJ1+fxaDCnak7rcOIv3hECnBemXrp5PnA4+a8ffDhg8KHxC3gR4Jd0BHGa U9jPjfC0XMjyJqzIjXH1/JyfNZOjovjGIcy7LUZ4PRESsbXE0INL6pvmeAvAJZeG I59U4aTIy85+gMM9/ZxOy1c5Qta6Yx+Jna7uQS8CgYEAtiRC2poVDFqDiBHdy+zO Gt+2baRRw6ewfY+PtHi36K8MqqLKCRJ3DB3p9rTkq75yeCa9huOVoE4QiGUVwzgx +w7PpRIECWLWW/SbW0wRCI7UyiLM+woWKjk7LneFEZjpjH6hCjVdLoe6qnamPmWu I2qSIxpct9/VC4ok7+UhBYECqYEA5Wr9Tp4jac8FaHJbqMocrZeC45kqd5R1tKS+ dS/a3wJCZ1zvkv8m6K+D3/aPO2bcgQuoFtw+5OLaWjzOyY4hKYEDRffllDeicZlq cjd+8KsMzum67Xd+zbOb8Fgyive6K+Cc/fbNbKYitc6N0IJ+mcvW+5jvvG7Ss9el C4RHz2sCgYBhykXUv0zuAYHhmUQhu/5YspuacwZTm0P27TQm7iX9wPaa7vwUPCEB v4AWjORZqCURbRiaonLCQ15ZZwg/GV1lfzsX1+WIrv3V4LhQ6glkpuN8pGOMv3vA GQEUj6zg6ALKdYwmFd8J3BYE3m3OGeOifk5OzDqCyyuNxOwAwh1peQ== -----END RSA PRIVATE KEY-----

+QFUPL: 1675,150

# OK

#### AT+QFLST

+QFLST: "cacert.pem",1446 +QFLST: "client.pem",1220 +QFLST: "user\_key1.pem",1679 //Query the list file.

#### ΟΚ

# **3.3. Active PDP Context**

//AT+QICSGP= <contextid>[,<context_type>,<apn>[,<username>,<password>[,<authentication>]]]</authentication></password></username></apn></context_type></contextid>		
// <contextid></contextid>	Integer type. The context ID. Range: 1–16.	
//< context_type >:	Integer type. The protocol type.	
	1 IPv4	
// <apn> :</apn>	String type. The access point name.	
// <username>:</username>	String type. The username.	
// <password> :</password>	String type. The password	
// <authentication></authentication>	Integer type. The authentication methods.	
//	0 None	
//	1 PAP	



// 2 CHAP // 3 PAP or CHAP AT+QICSGP=1,1,"m2m64.com.attz","",",0 OK AT+QIACT=1 //Active PDP contextID 1. OK

# 3.4. Configure SSL Option

AT+QSSLCFG="cacert",2,"cacert.pem" //0 OK	Configure the path of server certificate for SSL context 2.
AT+QSSLCFG="clientcert",2,"client.pem" //0 OK	Configure the path of client certificate for SSL context 2.
AT+QSSLCFG="clientkey",2,"user_key1.per	n" //Configure the path of client private key for SSL context 2.
OK	
AT+QSSLCFG="seclevel",2,2	<ul><li>//Configure the authentication mode for SSL context 2,</li><li>SSL authentication mode: server and client</li><li>authentication if requested by the remote server</li></ul>
OK	
AT+QSSLCFG="sslversion",2,4 OK	//SSL authentication version
AT+QSSLCFG="ciphersuite",2,0xFFFF OK	//Cipher suite
AT+QSSLCFG="ignorelocaltime",2,1 OK	//Ignore the time of authentication

# 3.5. Configure MQTT Option

AT+QMTCFG="SSL",2,1,2	//Configure MQTT session to SSL mode for SSL context 2
ОК	
AT+QMTCFG="version",2,4	//Configure MQTT protocol version to MQTT v3.1.1
ОК	

# **3.6. MQTT Connection and Data Interaction**

//Start MQTT SSL connection, please refer to chapter 2.3.
AT+QMTOPEN=2,"a2sgasbshsff52-ats.iot.us-west-2.amazonaws.com",8883



OK

+QMTOPEN: 2,0 AT+QMTCONN=2,"Quectel" OK

//Connect to MQTT server

+QMTCONN: 2,0,0

AT+QMTSUB=2,1,"\$aws/things/MyIoTDevice/shadow/get/accepted",0 //Subscribe to topics,

//**MyloTDevice** is device //name

OK

+QMTSUB: 2,1,0,

//Publish messages. After echoing >, input the payload, and tap "ctrl+Z" send the data.
AT+QMTPUB=2,1,1,0,"\$aws/things/MyloTDevice/shadow/get/accepted "
>Hello AWS IoT!

ΟΚ

+QMTPUB: 2,1,0

+QMTRECV: 2,1,"\$aws/things/MyIoTDevice/shadow/get/accepted","Hello AWS IoT!" AT+QMTCLOSE=2 //Close MQTTS connection

ΟΚ

+QMTCLOSE: 2,0

NOTE

For the details of above commands, please refer to **Document [1]**, **Document [2]**, **Document [3]** and **Document [4]**.



# **4** Support Band List

# Table 1: Support Band List

Parameter	Supported List		Note
<gsmbandval></gsmbandval>	00000000 No change 00000001 GSM 900MHz 00000002 GSM 1800MHz 00000004 GSM 850MHz 00000008 GSM 1900MHz 0000000F Any frequency band		eg.: 0x0a=0x02(G SM1800)+0x08(G SM1900) This parameter is valid on BG95-M3 module only.
<emtcbandval></emtcbandval>	0x1 (BAND_PREF_LTE_BAND1) 0x2 (BAND_PREF_LTE_BAND2) 0x4 (BAND_PREF_LTE_BAND3) 0x8 (BAND_PREF_LTE_BAND4) 0x10 (BAND_PREF_LTE_BAND5) 0x80 (BAND_PREF_LTE_BAND5) 0x800 (BAND_PREF_LTE_BAND12) 0x1000 (BAND_PREF_LTE_BAND13) 0x2000 (BAND_PREF_LTE_BAND14) 0x20000 (BAND_PREF_LTE_BAND18) 0x40000 (BAND_PREF_LTE_BAND19) 0x80000 (BAND_PREF_LTE_BAND20) 0x1000000 (BAND_PREF_LTE_BAND20) 0x2000000 (BAND_PREF_LTE_BAND25) 0x2000000 (BAND_PREF_LTE_BAND26) 0x4000000 (BAND_PREF_LTE_BAND27) 0x8000000 (BAND_PREF_LTE_BAND28) 0x4000000 (BAND_PREF_LTE_BAND28) 0x4000000 (BAND_PREF_LTE_BAND31) 0x200000000000000000 (BAND_PREF_LTE 0x8000000000000000000000 (BAND_PREF_LTE 0x100000000000000000000000000000000000	LTE B1 LTE B2 LTE B3 LTE B4 LTE B5 LTE B8 LTE B12 LTE B13 LTE B14 LTE B18 LTE B19 LTE B20 LTE B25 LTE B26 LTE B27 LTE B27 LTE B28 LTE B31 =BAND66) LTE B66 E_BAND72) LTE B72 TE_BAND73) LTE B73 F_LTE_BAND85) LTE B85	eg.: 0x15=0x01(LT EB1)+0x04(LTE B 3)+0x10(LTE B5)
<nbiotbandval></nbiotbandval>	0x1 (BAND_PREF_LTE_BAND1)	LTE B1	



0x2 (BAND_PREF_LTE_BAND2)	LTE B2
0x4 (BAND_PREF_LTE_BAND3)	LTE B3
0x8 (BAND_PREF_LTE_BAND4)	LTE B4
0x10 (BAND_PREF_LTE_BAND5)	LTE B5
0x80 (BAND_PREF_LTE_BAND8)	LTE B8
0x800 (BAND_PREF_LTE_BAND12)	LTE B12
0x1000 (BAND_PREF_LTE_BAND13)	LTE B13
0x2000 (BAND_PREF_LTE_BAND14)	LTE B14
0x20000 (BAND_PREF_LTE_BAND18)	LTE B18
0x40000 (BAND_PREF_LTE_BAND19)	LTE B19
0x80000 (BAND_PREF_LTE_BAND20)	LTE B20
0x1000000 (BAND_PREF_LTE_BAND25)	LTE B25
0x2000000 (BAND_PREF_LTE_BAND26)	LTE B26
0x4000000 (BAND_PREF_LTE_BAND27)	LTE B27
0x8000000 (BAND_PREF_LTE_BAND28)	LTE B28
0x40000000 (BAND_PREF_LTE_BAND31)	LTE B31
0x200000000000000000000000000000000000	BAND66)
	LTE B66
0x800000000000000000 (BAND_PREF_LTE_	BAND72
	LTE B72
0x10000000000000000000000 (BAND_PREF_LTE	_BAND73)
	LTE B73
0x100000000000000000000000000000000000	LTE_BAND85)
	LTE B85



# **5** Appendix A References

#### **Table 2: Related Documents**

SN	Document Name	Remark
[1]	Quectel_BG95&BG77_AT_Commands_Manual	AT command manual
[2]	Quectel_BG95&BG77&BG600L_Series_MQTT_Application_Note	MQTT application note
[3]	Quectel_BG95&BG77_SSL_Application_Note	SSL application note
[4]	Quectel_BG95&BG77_FILE_Application_Note	FILE application note
NOTE		

The *Quectel\_LTE\_Windows\_USB\_Driver*, *QCOM* tool and documents please obtain from Quectel. <u>https://www.quectel.com/support/download.htm</u>

# **Table 3: Terms and Abbreviations**

Abbreviation	Description
AWS	Amazon Web Services
юТ	Internet of Things
MQTT(S)	Message Queuing Telemetry Transport (Security)
TLS	Transport Layer Security
SSL	Secure Sockets Layer
QCOM	Serial port tool
EVB	Evaluation Board