

# **BG96 MQTT**

# **Application Note**

**LTE Module Series**

Rev. BG96\_MQTT\_Application\_Note\_V1.0

Date: 2018-01-04

Status: Released



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

**Quectel Wireless Solutions Co., Ltd.**

7<sup>th</sup> Floor, Hongye Building, No.1801 Hongmei Road, Xuhui District, Shanghai 200233, China

Tel: +86 21 5108 6236

Email: [info@quectel.com](mailto:info@quectel.com)

**Or our local office. For more information, please visit:**

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

**For technical support, or to report documentation errors, please visit:**

<http://quectel.com/support/technical.htm>

Or email to: [support@quectel.com](mailto: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.

**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 ARE FORBIDDEN WITHOUT PERMISSION. 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. 2018. All rights reserved.***

# About the Document

## History

Revision	Date	Author	Description
1.0	2018-01-04	Louis GU/ Serena SHEN	Initial

---

## Contents

About the Document .....	2
Contents .....	3
Table Index .....	4
<b>1 Introduction .....</b>	<b>5</b>
<b>2 MQTT Data Interaction.....</b>	<b>6</b>
<b>3 MQTT Related AT Commands .....</b>	<b>7</b>
3.1. AT Command Syntax.....	7
3.2. Description of MQTT Related AT Commands.....	7
3.2.1. AT+QMTCFG Configure Optional Parameters of MQTT .....	7
3.2.2. AT+QMTOPEN Open a Network for MQTT Client.....	11
3.2.3. AT+QMTCLOSE Close a Network for MQTT Client .....	12
3.2.4. AT+QMTCONN Connect a Client to MQTT Server.....	13
3.2.5. AT+QMTDISC Disconnect a Client from MQTT Server .....	14
3.2.6. AT+QMTSUB Subscribe to Topics .....	15
3.2.7. AT+QMTUNS Unsubscribe from Topics.....	16
3.2.8. AT+QMTPUB Publish Messages .....	17
<b>4 Summary of Error Codes .....</b>	<b>19</b>
<b>5 MQTT Related URCs .....</b>	<b>21</b>
5.1. "+QMTSTAT" URC to Indicate State Change in MQTT Link Layer .....	21
5.2. "+QMTRECV" URC to Notify the Host to Read MQTT Packet Data .....	22
<b>6 Examples .....</b>	<b>23</b>
6.1. Example of MQTT Operation without SSL.....	23
6.2. Example of MQTT Operation with SSL.....	25
<b>7 Appendix A References.....</b>	<b>28</b>

## Table Index

TABLE 1: DESCRIPTION OF <ERR> CODES .....	19
TABLE 2: MQTT RELATED URCS .....	21
TABLE 3: ERROR CODES OF THE URC .....	21
TABLE 4: RELATED DOCUMENTS .....	28
TABLE 5: TERMS AND ABBREVIATIONS .....	28

# 1 Introduction

MQTT (Message Queuing Telemetry Transport) is a broker-based publish/subscribe messaging protocol designed to be open, simple, lightweight and easy to implement. It is designed for connections with remote locations where a "small code footprint" is required or the network bandwidth is limited.

This document mainly introduces how to use the MQTT function of Quectel BG96 module through AT commands.

# 2 MQTT Data Interaction

This chapter gives the data interaction mechanism of MQTT function.

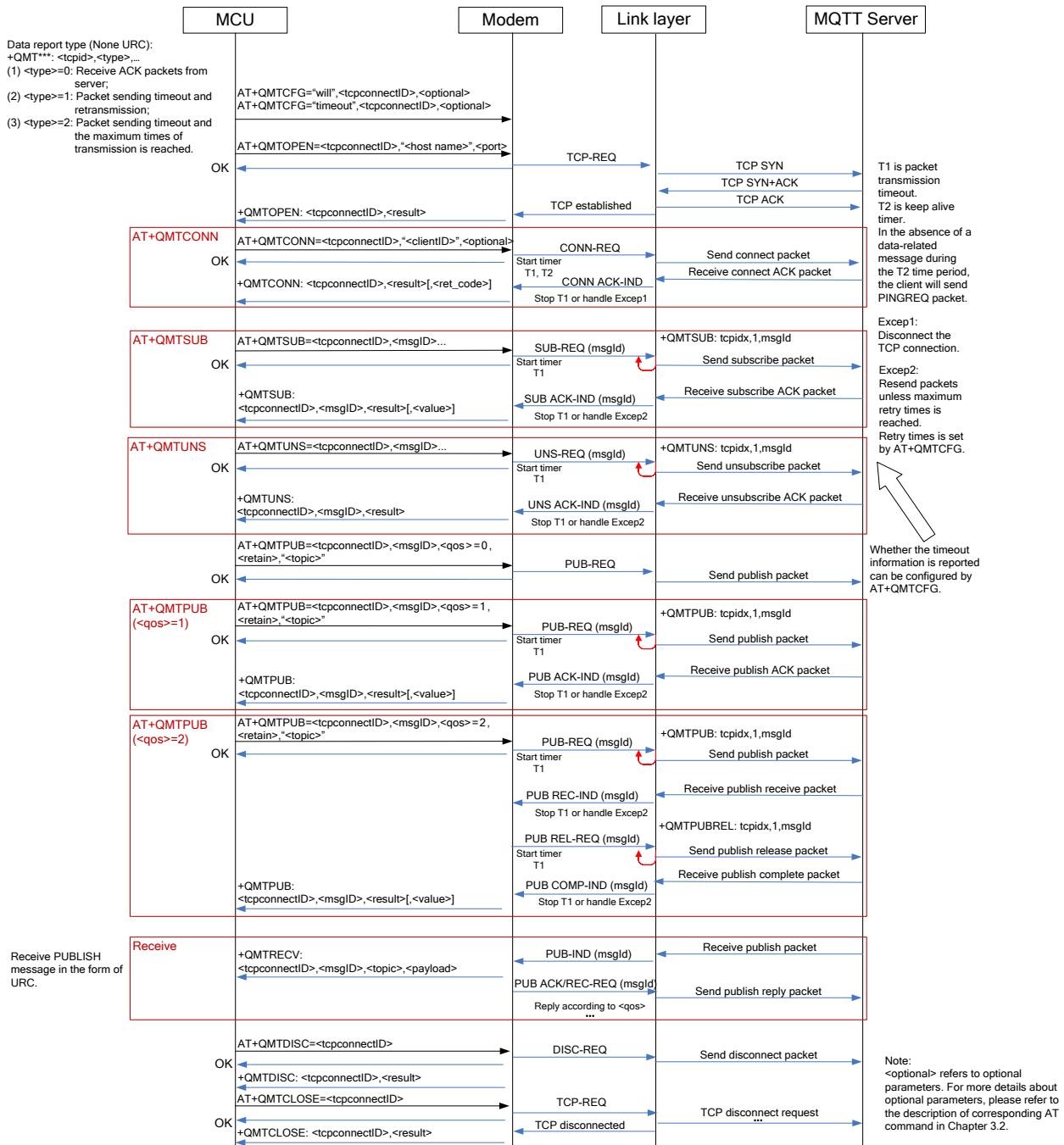


Figure 1: MQTT Data Interaction Diagram

# 3 MQTT Related AT Commands

This chapter presents the AT commands for operating MQTT function.

## 3.1. AT Command Syntax

Table 1: Types of AT Commands and Responses

Test Command	AT+<x>=?	This command returns the list of parameters and value ranges set by the corresponding Write Command or internal processes.
Read Command	AT+<x>?	This command returns the currently set value of the parameter or parameters.
Write Command	AT+<x>=<...>	This command sets the user-definable parameter values.
Execution Command	AT+<x>	This command reads non-variable parameters affected by internal processes in the UE.

## 3.2. Description of MQTT Related AT Commands

### 3.2.1. AT+QMTCFG Configure Optional Parameters of MQTT

The command is used to configure optional parameters of MQTT.

AT+QMTCFG Configure Optional Parameters of MQTT	
Test Command AT+QMTCFG=?	Response +QMTCFG: "version",(0-5),(3,4) +QMTCFG: "pdpcid",(0-5),(1-16) +QMTCFG: "ssl",(0-5),(0,1),(0-5) +QMTCFG: "keepalive",(0-5),(0-3600) +QMTCFG: "session",(0-5),(0,1) +QMTCFG: "timeout",(0-5),(1-60),(1-10),(0,1) +QMTCFG: "will",(0-5),(0,1),(0-2),(0,1),"willtopic","willmessage"



	<p><b>+QMTCFG:</b> "aliauth",(0-5),"productkey","devicename","devicesecret"</p> <p><b>OK</b></p>
<p>Write Command Configure the MQTT protocol version <b>AT+QMTCFG="version",&lt;tcpconnectID&gt;[,&lt;vsn&gt;]</b></p>	<p>Response <b>OK</b></p> <p>If &lt;vsn&gt; is omitted, query the MQTT protocol version. Response <b>+QMTCFG: "version",&lt;vsn&gt;</b></p> <p><b>OK</b></p> <p>If there is an error related to ME functionality, response: <b>+CME ERROR: &lt;err&gt;</b></p>
<p>Write Command Configure the PDP to be used by the MQTT client <b>AT+QMTCFG="pdpcid",&lt;tcpconnectID&gt;[,&lt;cid&gt;]</b></p>	<p>Response <b>OK</b></p> <p>If &lt;cid&gt; is omitted, query the PDP to be used by the MQTT client. Response <b>+QMTCFG: "pdpcid",&lt;cid&gt;</b></p> <p><b>OK</b></p> <p>If there is an error related to ME functionality, response: <b>+CME ERROR: &lt;err&gt;</b></p>
<p>Write Command Configure Will information <b>AT+QMTCFG="will",&lt;tcpconnectID&gt;[,&lt;will_fg&gt;[,&lt;will_qos&gt;,&lt;will_retain&gt;,"&lt;will_topic&gt;","&lt;will_msg&gt;"]]</b></p>	<p>Response <b>OK</b></p> <p>If &lt;will_fg&gt;, &lt;will_qos&gt;, &lt;will_retain&gt;, "&lt;will_topic&gt;" and "&lt;will_msg&gt;" are omitted, query the Will information. Response <b>+QMTCFG: "will",&lt;will_fg&gt;[,&lt;will_qos&gt;,&lt;will_retain&gt;,"&lt;will_topic&gt;","&lt;will_msg&gt;"]]</b></p> <p><b>OK</b></p> <p>If there is an error related to ME functionality, response: <b>+CME ERROR: &lt;err&gt;</b></p>

<p>Write Command Configure timeout of message delivery <b>AT+QMTCFG="timeout",&lt;tcpconnectID&gt;[,&lt;pkt_timeout&gt;[,&lt;retry_times&gt;]][,&lt;timeout_notice&gt;]]</b></p>	<p>Response <b>OK</b></p> <p>If &lt;pkt_timeout&gt;, &lt;retry_times&gt; and &lt;timeout_notice&gt; are omitted, query the timeout of message delivery.</p> <p>Response <b>+QMTCFG: "timeout",&lt;pkt_timeout&gt;,&lt;retry_times&gt;,&lt;timeout_notice&gt;</b></p> <p><b>OK</b></p> <p>If there is an error related to ME functionality, response: <b>+CME ERROR: &lt;err&gt;</b></p>
<p>Write Command Configure the session type <b>AT+QMTCFG="session",&lt;tcpconnectID&gt;[,&lt;clean_session&gt;]</b></p>	<p>Response <b>OK</b></p> <p>If &lt;clean_session&gt; is omitted, query the session type.</p> <p>Response <b>+QMTCFG: "session",&lt;clean_session&gt;</b></p> <p><b>OK</b></p> <p>If there is an error related to ME functionality, response: <b>+CME ERROR: &lt;err&gt;</b></p>
<p>Write Command Configure the keep-alive time <b>AT+QMTCFG="keepalive",&lt;tcpconnectID&gt;[,&lt;keep-alive time&gt;]</b></p>	<p>Response <b>OK</b></p> <p>If &lt;keep-alive time&gt; is omitted, query the keep-alive time.</p> <p>Response <b>+QMTCFG: "keepalive",&lt;keep-alive time&gt;</b></p> <p><b>OK</b></p> <p>If there is an error related to ME functionality, response: <b>+CME ERROR: &lt;err&gt;</b></p>
<p>Write Command Configure the MQTT SSL mode and SSL context index <b>AT+QMTCFG="ssl",&lt;tcpconnectID&gt;[,&lt;sslenable&gt;[,&lt;ctxindex&gt;]]</b></p>	<p>Response <b>OK</b></p> <p>If &lt;sslenable&gt; and &lt;ctxindex&gt; are omitted, query the MQTT SSL mode and SSL context index.</p> <p>Response <b>+QMTCFG: "ssl",&lt;sslenable&gt;[,&lt;ctxindex&gt;]</b></p> <p><b>OK</b></p>

	If there is an error related to ME functionality, response: <b>+CME ERROR: &lt;err&gt;</b>
Write Command Configure Alibaba device information for Alibaba Cloud <b>AT+QMTCFG="aliauth",&lt;tcpconnectID&gt;[,&lt;product_key&gt;,&lt;device_name&gt;,&lt;device_secret&gt;"]</b>	Response <b>OK</b>  If "<product_key>",<device_name>",<device_secret>" are omitted, query the device information. Response <b>[+QMTCFG: "aliauth",&lt;product_key&gt;,&lt;device_name&gt;,&lt;device_secret&gt;"]</b>  <b>OK</b>  If there is an error related to ME functionality, response: <b>+CME ERROR: &lt;err&gt;</b>
Maximum Response Time	300ms

## Parameter

<b>&lt;tcpconnectID&gt;</b>	MQTT socket identifier. The range is 0-5.
<b>&lt;vsn&gt;</b>	MQTT protocol version 3 MQTT protocol v3 4 MQTT protocol v4
<b>&lt;cid&gt;</b>	The PDP to be used by the MQTT client. The range is 1-16. The default value is 1.
<b>&lt;will_fg&gt;</b>	Configure the Will flag 0 Ignore the Will flag configuration 1 Require the Will flag configuration
<b>&lt;will_qos&gt;</b>	Quality of service for message delivery 0 At most once 1 At least once 2 Exactly once
<b>&lt;will_retain&gt;</b>	The Will retain flag is only used on PUBLISH messages. 0 When a client sends a PUBLISH message to a server, the server will not hold on to the message after it has been delivered to the current subscribers 1 When a client sends a PUBLISH message to a server, the server should hold on to the message after it has been delivered to the current subscribers
<b>&lt;will_topic&gt;</b>	Will topic string
<b>&lt;will_msg&gt;</b>	The Will message defines the content of the message that is published to the will topic if the client is unexpectedly disconnected. It can be a zero-length message.
<b>&lt;pkt_timeout&gt;</b>	Timeout of the packet delivery. The range is 1-60. The default value is 5. Unit: second.
<b>&lt;retry_times&gt;</b>	Retry times when packet delivery times out. The range is 0-10. The default value

	is 3.
<b>&lt;timeout_notice&gt;</b>	0 Not report timeout message when transmitting packet 1 Report timeout message when transmitting packet
<b>&lt;clean_session&gt;</b>	Configure the session type 0 The server must store the subscriptions of the client after it disconnects. 1 The server must discard any previously maintained information about the client and treat the connection as “clean”.
<b>&lt;keep-alive time&gt;</b>	Keep-alive time. The range is 0-3600. The default value is 120. Unit: second. It defines the maximum time interval between messages received from a client. If the server does not receive a message from the client within 1.5 times of the keep alive time period, it disconnects the client as if the client has sent a DISCONNECT message. 0 The client is not disconnected
<b>&lt;sslenable&gt;</b>	Configure the MQTT SSL mode 0 Use normal TCP connection for MQTT 1 Use SSL TCP secure connection for MQTT
<b>&lt;ctxindex&gt;</b>	SSL context index. The range is 0-5.
<b>&lt;product_key&gt;</b>	Product key issued by Alibaba Cloud
<b>&lt;device_name&gt;</b>	Device name issued by Alibaba Cloud
<b>&lt;device_secret&gt;</b>	Device secret key issued by Alibaba Cloud

#### NOTES

1. If **<will\_flag>**=1, then **<will\_qos>**, **<will\_retain>**, **<will\_topic>** and **<will\_msg>** must be present. Otherwise they will be omitted.
2. **<clean\_session>**=0 is only effective when the server supports the operation.
3. If MQTT connection is configured to SSL mode, **<ctxindex>** must be present. Also, customers need to use **AT+QSSLCFG** command to configure the SSL version, cipher suite, secure level, CA certificate, client certificate, client key and ignorance of RTC time, which will be used in MQTT SSL handshake procedure.
4. Care must be taken to ensure message delivery does not time out while it is still being sent.
5. **AT+QMTCFG=“aliauth”** command is only used for Alibaba Cloud. If it is configured, the parameters **<username>** and **<password>** in command **AT+QMTCONN** can be omitted.

### 3.2.2. AT+QMTOPEN Open a Network for MQTT Client

The command is used to open a network for MQTT client.

#### AT+QMTOPEN Open a Network for MQTT Client

Test Command  
**AT+QMTOPEN=?**

Response  
**+QMTOPEN:**  
(list of supported **<tcpconnectID>**s),“**<host\_name>**”,(list of supported **<port>**s)

	<b>OK</b>
Read Command <b>AT+QMTOPEN?</b>	Response <b>[+QMTOPEN: &lt;tcpconnectID&gt;,"&lt;host_name&gt;",&lt;port&gt;]</b>
	<b>OK</b>
Write Command <b>AT+QMTOPEN=&lt;tcpconnectID&gt;,"&lt;host_name&gt;",&lt;port&gt;</b>	Response <b>OK</b>  <b>+QMTOPEN: &lt;tcpconnectID&gt;,&lt;result&gt;</b>  If there is an error related to ME functionality, response: <b>+CME ERROR: &lt;err&gt;</b>
Maximum Response Time	75s, determined by network

## Parameter

<b>&lt;tcpconnectID&gt;</b>	MQTT socket identifier. The range is 0-5.
<b>&lt;host_name&gt;</b>	The address of the server. It could be an IP address or a domain name. The maximum size is 100 bytes.
<b>&lt;port&gt;</b>	The port of the server. The range is 1-65535.
<b>&lt;result&gt;</b>	Result of the command execution -1 Failed to open network 0 Network opened successfully 1 Wrong parameter 2 MQTT identifier is occupied 3 Failed to activate PDP 4 Failed to parse domain name 5 Network disconnection error

### 3.2.3. AT+QMTCLOSE Close a Network for MQTT Client

The command is used to close a network for MQTT client.

#### **AT+QMTCLOSE Close a Network for MQTT Client**

Test Command <b>AT+QMTCLOSE=?</b>	Response <b>+QMTCLOSE: (list of supported &lt;tcpconnectID&gt;s)</b>  <b>OK</b>
Write Command <b>AT+QMTCLOSE=&lt;tcpconnectID&gt;</b>	Response <b>OK</b>

	<p><b>+QMTCLOSE: &lt;tcpconnectID&gt;,&lt;result&gt;</b></p> <p>If there is an error related to ME functionality, response: <b>+CME ERROR: &lt;err&gt;</b></p>
Maximum Response Time	300ms

### Parameter

<b>&lt;tcpconnectID&gt;</b>	MQTT socket identifier. The range is 0-5.
<b>&lt;result&gt;</b>	Result of the command execution
	-1 Failed to close network
	0 Network closed successfully

### 3.2.4. AT+QMTCONN Connect a Client to MQTT Server

The command is used when a client requests a connection to MQTT server. When a TCP/IP socket connection is established from a client to a server, a protocol level session must be created using a CONNECT flow.

<b>AT+QMTCONN Connect a Client to MQTT Server</b>	
Test Command <b>AT+QMTCONN=?</b>	<p>Response</p> <p><b>+QMTCONN:</b> (list of supported &lt;tcpconnectID&gt;s), “&lt;clientID&gt;”[“&lt;username&gt;”[“&lt;password&gt;”]]</p> <p><b>OK</b></p>
Read Command <b>AT+QMTCONN?</b>	<p>Response</p> <p><b>[+QMTCONN: &lt;tcpconnectID&gt;,&lt;state&gt;]</b></p> <p><b>OK</b></p>
Write Command <b>AT+QMTCONN=&lt;tcpconnectID&gt;,”&lt;clientID&gt;”[“&lt;username&gt;”[“&lt;password&gt;”]]</b>	<p>Response</p> <p><b>OK</b></p> <p><b>+QMTCONN: &lt;tcpconnectID&gt;,&lt;result&gt;[,&lt;ret_code&gt;]</b></p> <p>If there is an error related to ME functionality, response: <b>+CME ERROR: &lt;err&gt;</b></p>
Maximum Response Time	<b>&lt;pkt_timeout&gt;</b> (default 5s), determined by network

## Parameter

<b>&lt;tcpconnectID&gt;</b>	MQTT socket identifier. The range is 0-5.
<b>&lt;clientID&gt;</b>	The client identifier string.
<b>&lt;username&gt;</b>	User name of the client. It can be used for authentication.
<b>&lt;password&gt;</b>	Password corresponding to the user name of the client. It can be used for authentication.
<b>&lt;result&gt;</b>	Result of the command execution 0 Packet sent successfully and ACK received from server 1 Packet retransmission 2 Failed to send packet
<b>&lt;state&gt;</b>	MQTT connection state 1 MQTT is initial 2 MQTT is connecting 3 MQTT is connected 4 MQTT is disconnecting
<b>&lt;ret_code&gt;</b>	Connection status return code 0 Connection Accepted 1 Connection Refused: Unacceptable Protocol Version 2 Connection Refused: Identifier Rejected 3 Connection Refused: Server Unavailable 4 Connection Refused: Bad User Name or Password 5 Connection Refused: Not Authorized
<b>&lt;pkt_timeout&gt;</b>	Timeout of the packet delivery. The range is 1-60. The default value is 5. Unit: second.

### NOTE

If a client with the same Client ID is already connected to the server, the "older" client must be disconnected by the server before completing the CONNECT flow of the new client.

### 3.2.5. AT+QMTDISC Disconnect a Client from MQTT Server

The command is used when a client requests a disconnection from MQTT server. A DISCONNECT message is sent from the client to the server to indicate that it is about to close its TCP/IP connection.

#### AT+QMTDISC Disconnect a Client from MQTT Server

Test Command <b>AT+QMTDISC=?</b>	Response <b>+QMTDISC:</b> (list of supported <b>&lt;tcpconnectID&gt;</b> s)
	<b>OK</b>
Write Command	Response

AT+QMTDISC=<tcpconnectID>	<p>OK</p> <p>+QMTDISC: &lt;tcpconnectID&gt;,&lt;result&gt;</p> <p>If there is an error related to ME functionality, response: +CME ERROR: &lt;err&gt;</p>
Maximum Response Time	300ms

### Parameter

<tcpconnectID>	MQTT socket identifier. The range is 0-5.
<result>	Result of the command execution
	-1 Failed to close connection
	0 Connection closed successfully

### 3.2.6. AT+QMTSUB Subscribe to Topics

The command is used to subscribe to one or more topics. A SUBSCRIBE message is sent by a client to register an interest in one or more topic names with the server. Messages published to these topics are delivered from the server to the client as PUBLISH messages.

AT+QMTSUB Subscribe to Topics	
Test Command AT+QMTSUB=?	<p>Response</p> <p>+QMTSUB: (list of supported &lt;tcpconnectID&gt;s),(list of supported &lt;msgID&gt;s),“&lt;topic&gt;”,(list of supported &lt;qos&gt;s)</p> <p>OK</p>
Write Command AT+QMTSUB=<tcpconnectID>,<msgID>,”<topic1>”,<qos1>[,”<topic2>”,<qos2>...]	<p>Response</p> <p>OK</p> <p>+QMTSUB: &lt;tcpconnectID&gt;,&lt;msgID&gt;,&lt;result&gt;[,&lt;value&gt;]</p> <p>If there is an error related to ME functionality: +CME ERROR: &lt;err&gt;</p>
Maximum Response Time	<pkt_timeout> * <retry_times> (default 15s), determined by network

### Parameter

<tcpconnectID>	MQTT socket identifier. The range is 0-5.
<msgID>	Message identifier of packet. The range is 1-65535.
<topic>	Topic that the client wants to subscribe to or unsubscribe from



<b>&lt;qos&gt;</b>	The QoS level at which the client wants to publish the messages. <u>0</u> At most once 1 At least once 2 Exactly once
<b>&lt;result&gt;</b>	Result of the command execution 0 Sent packet successfully and received ACK from server 1 Packet retransmission 2 Failed to send packet
<b>&lt;value&gt;</b>	If <b>&lt;result&gt;</b> is 0, it is a vector of granted QoS levels. If <b>&lt;result&gt;</b> is 1, it means the times of packet retransmission. If <b>&lt;result&gt;</b> is 2, it will not be presented.
<b>&lt;pkt_timeout&gt;</b>	Timeout of the packet delivery. The range is 1-60. The default value is 5. Unit: second.
<b>&lt;retry_times&gt;</b>	Retry times when packet delivery times out. The range is 0-10. The default value is 3.

**NOTE**

The **<msgid>** is only present in messages where the QoS bits in the fixed header indicate QoS levels 1 or 2. It must be unique amongst the set of "inflight" messages in a particular direction of communication. It typically increases by exactly one from one message to the next, but is not required to do so.

### 3.2.7. AT+QMTUNS Unsubscribe from Topics

The command is used to unsubscribe from one or more topics. An UNSUBSCRIBE message is sent by the client to the server to unsubscribe from named topics.

#### AT+QMTUNS Unsubscribe from Topics

Test Command <b>AT+QMTUNS=?</b>	Response <b>+QMTUNS:</b> (list of supported <b>&lt;tcpconnectID&gt;</b> s),(list of supported <b>&lt;msgid&gt;</b> s), <b>"&lt;topic&gt;"</b>  <b>OK</b>
Write Command <b>AT+QMTUNS=&lt;tcpconnectID&gt;,&lt;msgid&gt;,"&lt;topic1&gt;"["&lt;topic2&gt;"...]</b>	Response <b>OK</b>  <b>+QMTUNS:</b> <b>&lt;tcpconnectID&gt;,&lt;msgid&gt;,&lt;result&gt;</b>  If there is an error related to ME functionality, response: <b>+CME ERROR: &lt;err&gt;</b>
Maximum Response Time	<b>&lt;pkt_timeout&gt;</b> * <b>&lt;retry_times&gt;</b> (default 15s), determined by network

## Parameter

<b>&lt;tcpconnectID&gt;</b>	MQTT socket identifier. The range is 0-5.
<b>&lt;msgID&gt;</b>	Message identifier of packet. The range is 1-65535.
<b>&lt;topic&gt;</b>	Topic that the client wants to subscribe to or unsubscribe from
<b>&lt;result&gt;</b>	Result of the command execution 0 Sent packet successfully and received ACK from server 1 Packet retransmission 2 Failed to send packet
<b>&lt;pkt_timeout&gt;</b>	Timeout of the packet delivery. The range is 1-60. The default value is 5. Unit: second.
<b>&lt;retry_times&gt;</b>	Retry times when packet delivery times out. The range is 0-10. The default value is 3.

### 3.2.8. AT+QMTPUB Publish Messages

The command is used to publish messages by a client to a server for distribution to interested subscribers. Each PUBLISH message is associated with a topic name. If a client subscribes to one or more topics, any message published to those topics are sent by the server to the client as a PUBLISH message.

#### AT+QMTPUB Publish Messages

Test Command <b>AT+QMTPUB=?</b>	Response <b>+QMTPUB:</b> (list of supported <b>&lt;tcpconnectID&gt;</b> s),(list of supported <b>&lt;msgID&gt;</b> s),(list of supported <b>&lt;qos&gt;</b> s),(list of supported <b>&lt;retain&gt;</b> s)," <b>&lt;topic&gt;</b> ", " <b>&lt;msg&gt;</b> "  <b>OK</b>
Write Command <b>AT+QMTPUB=&lt;tcpconnectID&gt;,&lt;msgID&gt;,&lt;qos&gt;,&lt;retain&gt;,"&lt;topic&gt;"</b> After ">" is responded, input the data to be sent. Tap " <b>CTRL+Z</b> " to send, and tap " <b>ESC</b> " to cancel the operation.	Response <b>OK</b>  <b>+QMTPUB:</b> <b>&lt;tcpconnectID&gt;,&lt;msgID&gt;,&lt;result&gt;[,&lt;value&gt;]</b>  If there is an error related to ME functionality: <b>+CME ERROR:</b> <b>&lt;err&gt;</b>
Maximum Response Time	<b>&lt;pkt_timeout&gt;</b> * <b>&lt;retry_times&gt;</b> (default 15s), determined by network

## Parameter

<b>&lt;tcpconnectID&gt;</b>	MQTT socket identifier. The range is 0-5.
<b>&lt;msgID&gt;</b>	Message identifier of packet. The range is 0-65535. It will be 0 only when <b>&lt;qos&gt;=0</b> .

<b>&lt;qos&gt;</b>	The QoS level at which the client wants to publish the messages. <u>0</u> At most once 1 At least once 2 Exactly once
<b>&lt;retain&gt;</b>	Whether or not the server will retain the message after it has been delivered to the current subscribers. <u>0</u> The server will not retain the message after it has been delivered to the current subscribers 1 The server will retain the message after it has been delivered to the current subscribers
<b>&lt;topic&gt;</b>	Topic that needs to be published
<b>&lt;msg&gt;</b>	Message to be published
<b>&lt;result&gt;</b>	Result of the command execution 0 Packet sent successfully and ACK received from server (message that published when <b>&lt;qos&gt;</b> =0 does not require ACK) 1 Packet retransmission 2 Failed to send packet
<b>&lt;value&gt;</b>	If <b>&lt;result&gt;</b> is 1, it means the times of packet retransmission. If <b>&lt;result&gt;</b> is 0 or 2, it will not be presented.
<b>&lt;pkt_timeout&gt;</b>	Timeout of the packet delivery. The range is 1-60. The default value is 5. Unit: second.
<b>&lt;retry_times&gt;</b>	Retry times when packet delivery times out. The range is 0-10. The default value is 3.
<b>&lt;retry_times&gt;</b>	Retry times when packet delivery times out. The range is 0-10. The default value

## NOTES

1. If this command is executed successfully and gets **OK** back, the client can continue to publish new packet. The maximum quantity of transmitting packet should not be greater than that of inflight windows (5);
2. After executing this command, the client will be ready to send data, which will be sent as payload. The maximum length of the input data is 1548 bytes at a time and tap "Ctrl+Z" to send the data.
3. PUBLISH messages can be sent either from a publisher to the server, or from the server to a subscriber. When a server publishes messages to a subscriber, the following URC will be returned to notify the host to read the received data that is reported from MQTT server:  
**+QMTRECV: <tcpconnectID>,<msgID>,<topic>,<payload>**  
 For more details about the URC description, please refer to **Chapter 4.2**.

## 4 Summary of Error Codes

Final result code **+CME ERROR: <err>** indicates an error related to mobile equipment or network. The following table lists some of the general error codes.

**Table 1: Description of <err> Codes**

Code of <err>	Meaning
0	UE failure
1	No connection to UE
2	Adaptor link reserved
3	Operation not allowed
4	Operation not supported
5	PH-SIM PIN required
6	PH-FSIM PIN required
7	PH-FSIM PUK required
10	(U)SIM not inserted
11	(U)SIM PIN required
12	(U)SIM PUK required
13	(U)SIM failure
14	(U)SIM busy
15	(U)SIM wrong
16	Incorrect password
17	(U)SIM PIN2 required
18	(U)SIM PUK2 required

---

20	Memory full
21	Invalid index
22	Not found
23	Memory failure
24	Text string too long
25	Invalid characters in text string
26	Dial string too long
27	Invalid characters in dial string
30	No network service
31	Network timeout
32	Network not allowed - emergency calls only
40	Network personalization PIN required
41	Network personalization PUK required
42	Network subset personalization PIN required
43	Network subset personalization PUK required
44	Service provider personalization PIN required
45	Service provider personalization PUK required
46	Corporate personalization PIN required
47	Corporate personalization PUK required

---

# 5 MQTT Related URCs

This chapter gives MQTT related URCs and their descriptions.

**Table 2: MQTT Related URCs**

SN	URC Format	Description
[1]	<b>+QMTSTAT: &lt;tcpconnectID&gt;,&lt;err_code&gt;</b>	When the state of MQTT link layer is changed, the client will close the MQTT connection and report the URC.
[2]	<b>+QMTRECV: &lt;tcpconnectID&gt;,&lt;msgID&gt;,&lt;topic&gt;,&lt;payload&gt;</b>	Reported when the client has received the packet data from MQTT server.

## 5.1. “+QMTSTAT” URC to Indicate State Change in MQTT Link Layer

The URC begins with “+QMTSTAT:”. It will be reported when there is a change in the state of MQTT link layer.

### “+QMTSTAT” URC to Indicate State Change in MQTT Link Layer

<b>+QMTSTAT: &lt;tcpconnectID&gt;,&lt;err_code&gt;</b>	When the state of MQTT link layer is changed, the client will close the MQTT connection and report the URC.
Reference	

#### Parameter

<b>&lt;tcpconnectID&gt;</b>	MQTT socket identifier. The range is 0-5.
<b>&lt;err_code&gt;</b>	An error code. Please refer to the table below for details.

**Table 3: Error Codes of the URC**

<err_code>	Description	How to do
1	Connection is closed or reset by peer.	Execute <b>AT+QMTOPEN</b> command and reopen MQTT connection.

2	Sending PINGREQ packet timed out or failed.	Deactivate PDP first, and then active PDP and reopen MQTT connection.
3	Sending CONNECT packet timed out or failed.	<ol style="list-style-type: none"> <li>1. Check whether the inputted user name and password are correct.</li> <li>2. Make sure the client ID is not used.</li> <li>3. Reopen MQTT connection and try to send CONNECT packet to server again.</li> </ol>
4	Receiving CONNECK packet timed out or failed.	<ol style="list-style-type: none"> <li>1. Check whether the inputted user name and password are correct.</li> <li>2. Make sure the client ID is not used.</li> <li>3. Reopen MQTT connection and try to send CONNECT packet to server again.</li> </ol>
5	The client sends DISCONNECT packet to sever and the server is initiative to close MQTT connection.	This is a normal process.
6	The client is initiative to close MQTT connection due to packet sending failure all the time.	<ol style="list-style-type: none"> <li>1. Make sure the data is correct.</li> <li>2. Try to reopen MQTT connection since there may be network congestion or an error.</li> </ol>
7	The link is not alive or the server is unavailable.	Make sure the link is alive or the server is available currently.
8-255	Reserved for future use.	

## 5.2. “+QMTRECV” URC to Notify the Host to Read MQTT Packet Data

The URC begins with “+QMTRECV:”. It is mainly used to notify the host to read the received MQTT packet data that is reported from MQTT server.

### “+QMTRECV” URC to Notify the Host to Read MQTT Packet Data

<b>+QMTRECV:</b> <tcpconnectID>,<msgid>,<topic>,<payload>	Notify the host to read the received data that is reported from MQTT server.
Reference	

#### Parameter

<tcpconnectID>	MQTT socket identifier. The range is 0-5.
<msgid>	The message identifier of packet
<topic>	The topic that received from MQTT server
<payload>	The payload that relates to the topic name

# 6 Examples

This chapter gives the examples to explain how to use MQTT related AT commands.

## 6.1. Example of MQTT Operation without SSL

```
//Configure Alibaba device information for Alibaba cloud.
AT+QMTCFG="aliauth",0,"oyjtmPI5a5j","MQTT_TEST","wN9Y6pZSIly7Exa5qVzcmigEGO4kAazZ"
OK

AT+QMTOPEN=?
+QMTOPEN: <tcpconnectID>,"<host_name>",<port>

OK

//Open a network for MQTT client.
AT+QMTOPEN=0,"iot-as-mqtt.cn-shanghai.aliyuncs.com",1883
OK

+QMTOPEN: 0,0 //Opened the MQTT client network successfully.

AT+QMTOPEN?
+QMTOPEN: 0,"iot-as-mqtt.cn-shanghai.aliyuncs.com",1883

OK

AT+QMTCONN=?
+QMTCONN: <tcpconnectID>,"<clientID>" [,"<username>"] [,"<password>"]

OK

//Connect a client to MQTT server.
//If Alibaba Cloud is connected, customers can use AT+QMTCFG="aliauth" command to configure the
device information in advance, and do not need to provide username/password here anymore.
AT+QMTCONN=0,"clientExample"
OK

+QMTCONN: 0,0,0 //Connected the client to MQTT server successfully.
```



**AT+QMTSUB=?**

**+QMTSUB** : <tcpconnectID>,<msgID>,"<topic>",<qos>

OK

//Subscribe to topics.

**AT+QMTSUB=0,1,"topic/example",2**

OK

**+QMTSUB: 0,1,0,2**

**AT+QMTSUB=0,1,"topic/pub",0**

OK

**+QMTSUB: 0,1,0,0**

//If a client subscribes to a topic and other devices publish the same topic to the server, the module will report the following information.

**+QMTRECV: 0,0, "topic/example", "This is the payload related to topic"**

//Unsubscribe from topics.

**AT+QMTUNS=0,2, "topic/example"**

OK

**+QMTUNS: 0,2,0**

**AT+QMTPUB=?**

**+QMTPUB** : <tcpconnectID>,<msgID>,<qos>,<retain>,"<topic>"

OK

//Publish messages.

**AT+QMTPUB=0,0,0,0,"topic/pub"**

**>This is test data, hello MQTT.**      //After receiving '>', input data "This is test data, hello MQTT." and then send it. The maximum length of the data is 1548 bytes and the data that beyond 1548 bytes will be omitted. After inputting data, tap "Ctrl+Z" to send.

OK

**+QMTPUB: 0,0,0**

//If a client subscribes to a topic named "topic/pub" and other devices publish the same topic to the server, the module will report the following information.

**+QMTRECV: 0,0, "topic/pub",This is test data, hello MQTT.**

```
//Disconnect a client from MQTT server.  
AT+QMTDISC=0  
OK  
  
+QMTDISC: 0,0 //Disconnection closed successfully.
```

## 6.2. Example of MQTT Operation with SSL

```
//Configure MQTT session into SSL mode.  
AT+QMTCFG="SSL", 0, 1, 2  
OK  
  
//If SSL authentication mode is "server authentication", store CA certificate to RAM.  
AT+QFUPL="cacert.pem",1758,100  
CONNECT  
<Input the cacert.pem data, the size is 1758 bytes>  
+QFUPL: 1758,384a  
  
OK  
  
//If SSL authentication mode is "server authentication", store CC certificate to RAM.  
AT+QFUPL="client.pem",1220,100  
CONNECT  
<Input the client.pem data, the size is 1220 bytes>  
+QFUPL: 1220,2d53  
  
OK  
  
//If SSL authentication mode is "server authentication", store CK certificate to RAM.  
AT+QFUPL="user_key.pem",1679,100  
CONNECT  
<Input the client.pem data, the size is 1679 bytes>  
+QFUPL: 1679,335f  
  
OK  
  
//Configure CA certificate.  
AT+QSSLCFG="cacert",2,"cacert.pem"  
OK  
  
//Configure CC certificate.  
AT+QSSLCFG="clientcert",2,"client.pem"
```

OK

//Configure CK certificate.

**AT+QSSLCFG="clientkey",2,"user\_key.pem"**

OK

//Configure SSL parameters.

**AT+QSSLCFG="secllevel",2,2** //SSL authentication mode: server authentication

OK

**AT+QSSLCFG="sslversion",2,4** //SSL authentication version

OK

**AT+QSSLCFG="ciphersuite",2,"0xFFFF"** //Cipher suite

OK

**AT+QSSLCFG="ignorelocaltime",1** //Ignore the time of authentication.

OK

//Start MQTT SSL connection

**AT+QMTOPEN=0, "a1zgnxur10j8ux.iot.us-east-1.amazonaws.com", "8883"**

OK

**+QMTOPEN: 0,0**

//Connect to MQTT server

**AT+QMTCONN=0,"M26\_0206"**

OK

**+QMTCONN: 0,0,0**

//Subscribe to topics.

**AT+QMTSUB=0,1,"\$aws/things/M26\_0206/shadow/update/accepted",1**

OK

**+QMTSUB: 0,1,0,1**

//Publish messages.

**AT+QMTPUB=0,1,1,0,"\$aws/things/M26\_0206/shadow/update/accepted"**

>This is publish data from client

OK

**+QMTPUB: 0,1,0**

//If a client subscribes to a topic named "\$aws/things/M26\_0206/shadow/update/accepted" and other devices publish the same topic to the server, the module will report the following information.

**+QMTRECV: 0,1,"\$aws/things/M26\_0206/shadow/update/accepted",This is publish data from**

client

//Disconnect a client from MQTT server.

**AT+QMTDISC=0**

OK

+QMTDISC: 0,0

# 7 Appendix A References

**Table 4: Related Documents**

SN	Document Name	Remarks
[1]	MQTT V3.1 Protocol Specification	MQTT protocol specification version 3.1
[2]	MQTT V3.1.1 Protocol Specification	MQTT protocol specification version 3.1.1

**Table 5: Terms and Abbreviations**

Abbreviation	Description
ACK	Acknowledgement
MQTT	Message Queuing Telemetry Transport
QoS	Quality of Service
RAM	Random Access Memory
SSL	Secure Sockets Layer
TCP	Transmission Control Protocol
URC	Unsolicited Result Code