BG95&BG77&BG600L Series
DFOTA Upgrade Guide

LPWA Module Series

Version: 1.4

Date: 2024-09-26

Status: Released

At Quectel, our aim is to provide timely and comprehensive services to our customers. If you require any assistance, please contact our 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 offices. 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 us at: support@quectel.com.

Legal Notices

We offer information as a service to you. The provided information is based on your requirements and we make every effort to ensure its quality. You agree that you are responsible for using independent analysis and evaluation in designing intended products, and we provide reference designs for illustrative purposes only. Before using any hardware, software or service guided by this document, please read this notice carefully. Even though we employ commercially reasonable efforts to provide the best possible experience, you hereby acknowledge and agree that this document and related services hereunder are provided to you on an “as available” basis. We may revise or restate this document from time to time at our sole discretion without any prior notice to you.

Use and Disclosure Restrictions

License Agreements

Documents and information provided by us shall be kept confidential, unless specific permission is granted. They shall not be accessed or used for any purpose except as expressly provided herein.

Copyright

Our and third-party products hereunder may contain copyrighted material. Such copyrighted material shall not be copied, reproduced, distributed, merged, published, translated, or modified without prior written consent. We and the third party have exclusive rights over copyrighted material. No license shall be granted or conveyed under any patents, copyrights, trademarks, or service mark rights. To avoid ambiguities, purchasing in any form cannot be deemed as granting a license other than the normal non-exclusive, royalty-free license to use the material. We reserve the right to take legal action for noncompliance with abovementioned requirements, unauthorized use, or other illegal or malicious use of the material.

Trademarks

Except as otherwise set forth herein, nothing in this document shall be construed as conferring any rights to use any trademark, trade name or name, abbreviation, or counterfeit product thereof owned by Quectel or any third party in advertising, publicity, or other aspects.

Third-Party Rights

This document may refer to hardware, software and/or documentation owned by one or more third parties ("third-party materials"). Use of such third-party materials shall be governed by all restrictions and obligations applicable thereto.

We make no warranty or representation, either express or implied, regarding the third-party materials, including but not limited to any implied or statutory, warranties of merchantability or fitness for a particular purpose, quiet enjoyment, system integration, information accuracy, and non-infringement of any third-party intellectual property rights with regard to the licensed technology or use thereof. Nothing herein constitutes a representation or warranty by us to either develop, enhance, modify, distribute, market, sell, offer for sale, or otherwise maintain production of any our products or any other hardware, software, device, tool, information, or product. We moreover disclaim any and all warranties arising from the course of dealing or usage of trade.

Privacy Policy

To implement module functionality, certain device data are uploaded to Quectel's or third-party's servers, including carriers, chipset suppliers or customer-designated servers. Quectel, strictly abiding by the relevant laws and regulations, shall retain, use, disclose or otherwise process relevant data for the purpose of performing the service only or as permitted by applicable laws. Before data interaction with third parties, please be informed of their privacy and data security policy.

Disclaimer

a) We acknowledge no liability for any injury or damage arising from the reliance upon the information.
b) We shall bear no liability resulting from any inaccuracies or omissions, or from the use of the information contained herein.
c) While we have made every effort to ensure that the functions and features under development are free from errors, it is possible that they could contain errors, inaccuracies, and omissions. Unless otherwise provided by valid agreement, we make no warranties of any kind, either implied or express, and exclude all liability for any loss or damage suffered in connection with the use of features and functions under development, to the maximum extent permitted by law, regardless of whether such loss or damage may have been foreseeable.
d) We are not responsible for the accessibility, safety, accuracy, availability, legality, or completeness of information, advertising, commercial offers, products, services, and materials on third-party websites and third-party resources.

About the Document

Revision History

Version	Date	Author	Description
1.0	2020-01-15	Matt YE/ Lane HAO	Initial
1.1	2020-07-29	Matt YE	1. Added the applicable module BG600L-M3. 2. Added supported baud rates for reporting of DFOTA URCs and develop the main UART baud rate to follow the configuration by AT+IPR. 3. Updated the range of the <file_size> parameter for delta package.
1.2	2022-01-25	Billie Xing/ Matt Ye/ Adonis Chen	1. Added parameters <breakpoint> and <request_size> in AT+QFOTADL=<HTTP_URL>. 2. Added command AT+QFOTADL=<FTP_URL>. 3. Modified the note in AT+QFOTADL="COM:",<file_size>,<timeout>. 4. Added more parameters for command AT+QFOTADL=<para1>,<para2>,<para3>. 5. Added command AT+QFOTADL=<file_name>,<type>. 6. Added command AT+QCFGEXT="sni". 7. Added FTP(S) DFOTA example. 8. Added HTTP(S) header configuration. 9. Added watch dog configuration for DFOTA. 10. Added error code number.
1.3	2024-06-18	Billie XING	1. Updated the response of AT+QFOTADL=<para1>[,<para2>[,<para3>]] and updated the parameters of <para1>, <para2> and <para3> (Chapter 2.2.5.5). 2. Added a note about the applicable situation of AT+QFOTADL=<file_name>,<type> (Chapter 2.2.5.6). 3. Added an example of resuming downloading for

			HTTP(S) DFOTA (Chapter 2.3.7). 4. Added a note for getting a delta firmware package (Chapter 2.1.1).
1.4	2024-09-26	Billie XING	1. Updated the document name from "Application Note" to "Upgrade Guide". 2. Added the chapter of DFOTA implementation and user responsibility (Chapter 1.1). 3. Optimized the DFOTA firmware upgrade flowchart to distinguish between the user side and the Quectel side (Figure 1). 4. Optimized the description of the DFOTA firmware upgrade steps and added relevant notes to distinguish between the user side and the Quectel side (Chapter 2.1). 5. Updated the declaration of AT command examples (Chapter 2.2.3). 6. Added the description of <HTTP_URL> and optimized the explanation of the <HTTP_server_URL> (Chapter 2.2.5.1). 7. Added the description of <CoAP_URL> and optimized the explanation of the <CoAP_server_URL> (Chapter 2.2.5.2). 8. Added the description of <FTP_URL> and optimized the explanation of the <FTP_server_URL> (Chapter 2.2.5.3). 9. Updated the description of <para3> if <para1>=10 (Chapter 2.2.5.5). 10. Updated the description of <file_name> and the notes about UFS and EUFS (Chapter 2.2.5.6). 11. Updated the URL address and related instructions in the command example (Chapter 2.3).

Contents

Table Index

Figure Index

1 Introduction

Quectel BG95 series, BG77 and BG600L-M3 modules support DFOTA (Delta Firmware Over-The-Air) feature, which allows users to update the module firmware wirelessly.

In DFOTA, a delta firmware package, which contains only differences between the original firmware version and target firmware version, is needed before firmware updating. Therefore, DFOTA is time-saving and can reduce unnecessary data transmission.

This document introduces how to update the firmware of Quectel BG95 series, BG77 and BG600L-M3 modules via DFOTA, which can be triggered either by **AT+QFOTADL** or via LwM2M.

NOTE

1. It is mandatory to ensure a stable power supply during DFOTA upgrade process.
2. If there is an external watchdog monitoring of the module, the DAM could not run during DFOTA process, and customer can't feed the watch dog in the DAM App. So, it needs to remove the watch dog temporarily or fed it by AP kernel (refer to ***Chapter 2.3.6*** for details) to prevent unexpected restart behavior that interrupts the DFOTA upgrade process and causes module to be damaged possibly.

1.1. DFOTA Implementation and User Responsibility

Quectel follows industry best practices with regard to firmware updates for its modules by enabling users to offer DFOTA updates. Please note that Quectel does not have the ability to unilaterally push updates to users' devices. Quectel hands full control over the DFOTA process to users. In the process, Quectel solely provides the updated firmware but cannot initiate DFOTA updates on users' devices.

Users can determine when to push the update to the Quectel modules using the DFOTA mechanism by configuring corresponding parameters for the update that the users host on their own infrastructures.

2 Trigger DFOTA by AT Commands

2.1. Firmware Update Procedure

The following chart illustrates the DFOTA procedure triggered by the AT command under the condition that the firmware package be stored on an HTTP(S)/CoAP(S)/FTP(S) server.

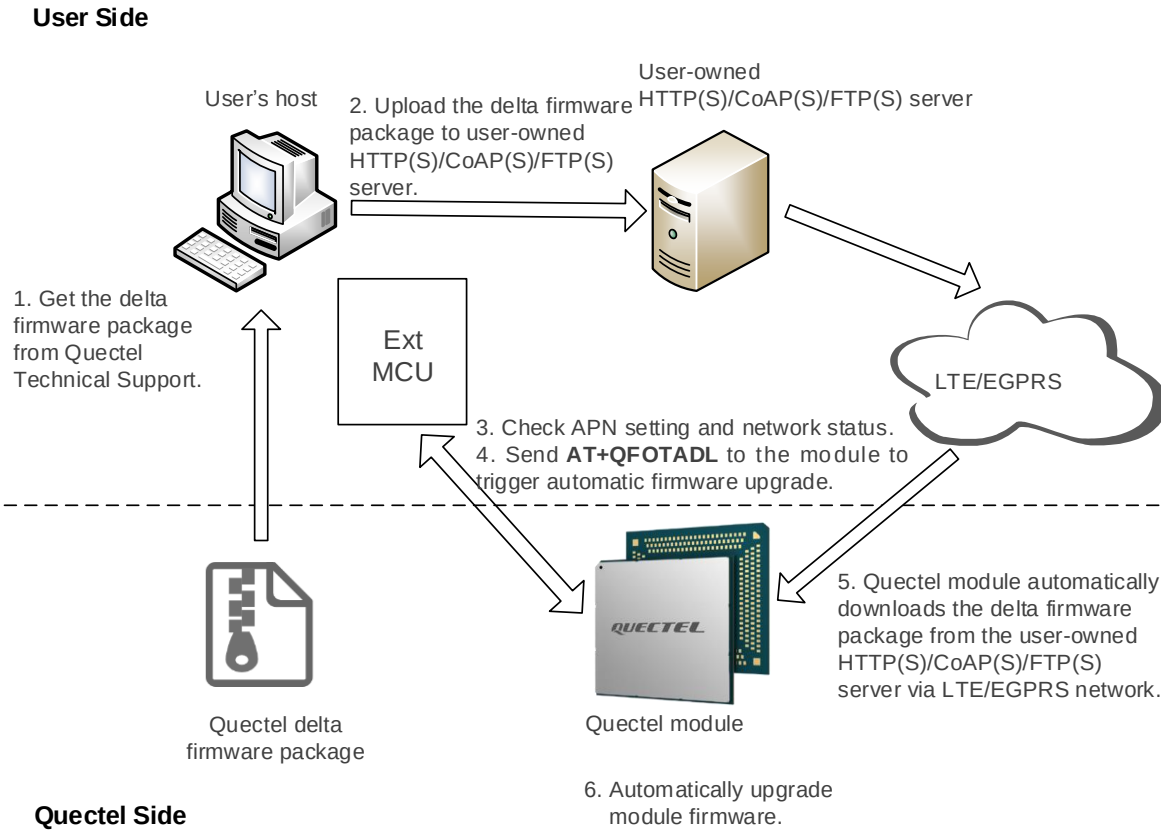

Figure 1: DFOTA Procedure (Triggered by AT Command)

As shown in the above figure, the following steps need to be performed to update the firmware when the firmware package is stored on an HTTP(S)/CoAP(S)/FTP(S) server:

Step 1: Get the delta firmware package from Quectel Technical Support (see ***Chapter 2.1.1*** for details).
Step 2: Upload the delta firmware package to your HTTP(S)/CoAP(S)/FTP(S) server (see ***Chapter 2.1.2*** for details).

Step 3: Check APN setting and network status (see ***Chapter 2.1.3*** for details).
Step 4: Send **AT+QFOTADL** to the module to trigger automatic firmware update (see ***Chapter 2.1.4*** for details).
Step 5: The delta firmware package is downloaded to the module from your HTTP(S)/CoAP(S)/FTP(S) server via LTE/EGPRS network.
Step 6: The module internally updates the firmware automatically.

NOTE

You are responsible for providing and managing the HTTP(S)/CoAP(S)/FTP(S) server for the firmware update. Quectel does not supply the server or assist with its setup.

2.1.1. Get a Delta Firmware Package

Before firmware updating, send the original firmware version and the selected target firmware version to Quectel Technical Support to obtain the corresponding delta firmware package.

NOTE

If there is a large difference between the original firmware version and the target firmware version, the size of the delta firmware package will be large, thereby reducing the success rate of DFOTA downloading. It is recommended to avoid such scenarios. However, if there is a strong demand for this scenario, please contact Quectel Technical Support to provide the intermediate firmware version and get relevant DFOTA package for transitional upgrade.

2.1.2. Upload Delta Package to HTTP(S)/CoAP(S)/FTP(S) Server

Step 1: Set up the HTTP(S)/CoAP(S)/FTP(S) server before using the DFOTA function (Quectel does not provide such servers.)
Step 2: Upload the delta firmware package to your server and save its storage path.
Step 3: The module downloads the delta package through the path after the corresponding AT command is executed.

2.1.3. Check APN Setting and Network Status

Step 1: Check APN setting: check whether the APN is set properly after uploading the delta firmware package to the HTTP(S)/CoAP(S)/FTP(S) server. For instance, the APN for a Telstra (U)SIM card should be "**Telstra.internet**", and the relevant AT commands are listed below:

- Query DFOTA APN: **AT+QCFGEXT="fota_apn"**
- Set DFOTA APN: **AT+QCFGEXT="fota_apn",0,"Telstra.internet"**

Step 2: Check network status: after APN setting is confirmed, make sure that the data network is registered before firmware updating, relevant AT commands are listed below:

- **AT+CSQ**: Query signal quality
- **AT+CEREG?**: Query network registration status (Optional: **AT+CGREG?**)
- **AT+COPS?**: Query the registered operator

For more details about the above commands, see ***document [1]***.

2.1.4. Execute AT Command to Update the Firmware

After the APN setting and network status are confirmed to be ready, execute **AT+QFOTADL** to download the firmware package and update the firmware. The package is downloaded to the module from the HTTP(S)/CoAP(S)/FTP(S) server via LTE/EGPRS network. The firmware is updated automatically. For more details about the DFOTA AT commands, see ***Chapter 2.3***.

NOTE

The module supports firmware update via the HTTP(S)/CoAP(S)/FTP(S) server, the COM port and the local file system. For more information on the firmware update via the COM port or the local file system, see ***Chapter 2.2.5.4*** or ***Chapter 2.2.5.5***.

2.2. DFOTA Related AT Commands

2.2.1. Definitions

- **<CR>** Carriage return character.
- **<LF>** Line feed character.
- **<...>** Parameter name. Angle brackets do not appear on the command line.
- **[...]** Optional parameter of a command or an optional part of TA information response. Square brackets do not appear on the command line. When an optional parameter is not given in a command, the new value equals its previous value or the default settings, unless otherwise specified.
- **Underline** Default setting of a parameter.

2.2.2. AT Command Syntax

All command lines must start with **AT** or **at** and end with **<CR>**. Information responses and result codes always start and end with a carriage return character and a line feed character: **<CR><LF><response><CR><LF>**. In tables presenting commands and responses throughout this document, only the commands and responses are presented, and **<CR>** and **<LF>** are deliberately omitted.

Table 1: Types of AT Commands

Command Type	Syntax	Description
Test Command	**AT+<cmd>=?**	Test the existence of the corresponding command and return information about the type, value, or range of its parameter.
Read Command	**AT+<cmd>?**	Check the current parameter value of the corresponding command.
Write Command	**AT+<cmd>=<p1>[,<p2>[,<p3>[...]]]**	Set user-definable parameter value.
Execution Command	**AT+<cmd>**	Return a specific information parameter or perform a specific action.

2.2.3. Declaration of AT Command Examples

The AT command examples in this document are provided to help you learn about the use of the AT commands introduced herein. The examples, however, should not be taken as Quectel's recommendations or suggestions about how to design a program flow or what status to set the module into. Sometimes multiple examples may be provided for one AT command. However, this does not mean that there is a correlation among these examples, or that they should be executed in a given sequence. The URLs, domain names, IP addresses, usernames/accounts, and passwords (if any) in the AT command examples are provided for illustrative and explanatory purposes only, and they should be modified to reflect your actual usage and specific needs.

2.2.4. Description of AT Command Mode and Data Mode

The COM port of BG95 series, BG77 and BG600L-M3 modules provide two working modes: AT command mode and data mode. In the AT command mode, the inputted data via COM port is treated as AT commands. In data mode, it is treated as data.

Inputting **+++** or pulling up MAIN_DTR (**AT&D1** should be set first) can make the COM port exit data mode. To prevent **+++** from being misinterpreted as data, the following sequence should be followed:

1. Do not input any character for at least 1 s before inputting **+++**.
2. Finish the input of **+++** within 1 s, and no other characters can be inputted during this time.
3. Do not input any character within 1 s after **+++** has been inputted.

When **AT+QFOTADL="COM:",<file_size>,<timeout>** is executed, the COM port enters the data mode. The port exits the data mode forcibly and the command is interrupted immediately when **+++** is inputted or MAIN_DTR is pulled up before the command response is returned.

2.2.5. AT+QFOTADL Update Firmware via DFOTA

AT+QFOTADL Update Firmware via DFOTA	
Test Command **AT+QFOTADL=?**	Response **OK**

AT+QFOTADL starts automatic firmware update for the modules via DFOTA. When the **AT+QFOTADL** is executed, the package is automatically downloaded to the module. After the package download is finished, the module automatically updates the firmware. The module reboots automatically if the firmware is updated successfully, otherwise it returns an error and exits from DFOTA.

2.2.5.1. AT+QFOTADL=<HTTP_URL>[,<breakpoint>[,<request_size>]] Update Firmware When the Delta Package is Stored on an HTTP(S) Server

If the delta firmware package is stored on the HTTP(S) server, **AT+QFOTADL=<HTTP_URL>[,<breakpoint>[,<request_size>]]** is executed to start automatic firmware update via DFOTA. The module downloads the delta package from the HTTP(S) server over the air and updates the firmware automatically. Besides, it also supports the breakpoint to resume downloading, the **<breakpoint>** is 0 by default. And customer also can configure DFOTA package size for each request through **<request_size>**.

AT+QFOTADL=<HTTP_URL>[,<breakpoint>[,<request_size>]] Update Firmware When the Delta Package is Stored on an HTTP(S) Server	
Write Command **AT+QFOTADL=<HTTP_URL>[,<breakpoint>[,<request_size>]]**	Response If the optional parameter is omitted: **OK** **+QIND: "FOTA","HTTPSTART"** **+QIND: "FOTA","DOWNLOADING",<percent>** **+QIND: "FOTA","DOWNLOADING",<percent>** **...** **+QIND: "FOTA","HTTPEND",<HTTP_err>** **+QIND: "FOTA","RESETTING"** **+QIND: "FOTA","START"** **+QIND: "FOTA","UPDATING",<percent>** **+QIND: "FOTA","UPDATING",<percent>** **...** **+QIND: "FOTA","END",<update_err>** If any of the optional parameters is specified, set the breakpoint position and request size to update the firmware:

	OK **+QIND: "FOTA","HTTPSTART"** **+QIND: "FOTA","DOWNLOADING",<percent>** **+QIND: "FOTA","DOWNLOADING",<percent>** **...** **+QIND: "FOTA","HTTPEND",<HTTP_err>** If there is any error: **+CME ERROR: <err>**
Maximum Response Time	300 ms
Characteristics	The command takes effect immediately. The configurations are not saved.

Parameter

<HTTP_URL>	String type. The URL that the delta firmware package stored on the HTTP(S) server. The maximum length is 700 bytes. It should be started with "HTTP://" or "HTTPS://", for example: **"HTTP://<HTTP_server_URL>:<HTTP_port>/<HTTP_file_path>"**.
<breakpoint>	Integer type. The offset location of the delta package file. Default value: 0.
<request_size>	Integer type. The size of the custom segment downloaded. It must be an integer multiple of 2048. Default value: 0.
<HTTP_server_URL>	String type. The IP address or domain name of the HTTP(S) server owned and operated by you.
<HTTP_port>	Integer type. The port number of the HTTP(S) server. Default value: 80. Range: 1–65535.
<HTTP_file_path>	String type. The file path in the HTTP(S) server.
<HTTP_err>	Integer type. The HTTP(S) error code. 0 Downloaded successfully. Any other value Errors. See ***Chapter 4*** for more details.
<percent>	Integer type. The download or update progress in percentage.
<update_err>	Integer type. 0 Updated successfully. Any other value Errors. See ***Chapter 4*** for more details.
<err>	Error codes. See ***Chapter 4*** for more details.

NOTE

1. In the DFOTA mode, only the APP works normally and the modem is not loaded. In this period, the main UART cannot receive AT commands anymore, and can only report DFOTA URCs at the baud rate configured by **AT+IPR**. Supported baud rates for the reporting of URCs are 4800, 9600, 19200,

38400, 115200, 230400, 460800 and 921600 bps. The default baud rate 115200 bps is used if any unsupported baud rate is set by **AT+IPR** for the URC reporting. Additionally, the USB port cannot be used either.

2. After a smooth **"DOWNLOADING"** process, the module is reset to enter the DFOTA mode. When the **"UPDATING"** process is finished, the module restarts and start up in a normal mode.
3. If the module is powered down during **"UPDATING"** process, the module will automatically enter the DFOTA mode and continue the previous firmware update when it is powered up next time.
4. (1) If SSL certificates are needed while downloading the firmware package from an HTTPS server, they should be uploaded to */datatx* in APP EFS, and must be renamed in accordance with the following rules:
   1) The root certificate is renamed as *fota_cacert.pem;*
   2) The client certificate is renamed as *fota_client_cert.pem*;
   3) The private key is renamed as *fota_client_key.pem*.

   (2) Certificate upload methods:
   - Quectel QEFS Explorer tool can be used to upload the certificates. For more details about the tool, refer to ***document [2]***.
   - **AT+QFUPL** can also be used to upload the certificates. For more details about the command, see ***document [3]***.

2.2.5.2. AT+QFOTADL=<CoAP_URL> Update Firmware When the Delta Package is Stored on a CoAP(S) Server

If the delta firmware package is stored on a CoAP(S) server, the **AT+QFOTADL=<CoAP_URL>** is executed to start automatic firmware update via DFOTA. The module downloads the delta package from the CoAP(S) server wirelessly and updates the firmware automatically.

AT+QFOTADL=<CoAP_URL> Update Firmware When the Delta Package is Stored on a CoAP(S) Server	
Write Command **AT+QFOTADL=<CoAP_URL>**	Response **OK** **+QIND: "FOTA","COAPSTART"** **+QIND: "FOTA","COAPEND",<CoAP_err>** **+QIND: "FOTA"," RESETTING"** **+QIND: "FOTA","START"** **+QIND: "FOTA","UPDATING",<percent>** **+QIND: "FOTA","UPDATING",<percent>** **...** **+QIND: "FOTA","END",<update_err>** If there is any error: **+CME ERROR: <err>**

Maximum Response Time	300 ms
Characteristics	The command takes effect immediately.

Parameter

<CoAP_URL>	String type. The URL that the delta firmware package stored on the CoAP(S) server. The maximum length is 700 bytes. It should be started with "CoAP://" or "CoAPS://", for example: **"CoAP://<CoAP_server_URL>:<CoAP_port>/<CoAP_file_path>"**.
<CoAP_server_URL>	String type. The IP address or domain name of the CoAP(S) server owned and operated by you.
<CoAP_port>	Integer type. The port number of the CoAP(S) server. Default value: 80. Range: 1–65535.
<CoAP_file_path>	String type. The file path in CoAP(S) server.
<CoAP_err>	Integer type. The CoAP(S) error code. 0 Downloaded successfully. Any other value Errors. See ***Chapter 4*** for more details.
<percent>	Integer type. The update progress in percentage.
<update_err>	Integer type. 0 Updated successfully. Any other value Errors. See ***Chapter 4*** for more details.
<err>	Error codes. See ***Chapter 4*** for more details.

NOTE

The modules support two security modes for CoAP over DTLS: certificates mode and pre-shared key (PSK) mode.

- In certificates mode, certificates are necessary while the firmware package is downloaded from a CoAP server. They should be uploaded to */datatx/ssl* in APP EFS, and must be renamed in accordance with the following rules:
  1) The root certificate is renamed as *coap-cert.pem;*
  2) The client certificate is renamed as *coap-cacert.pem*;
  3) The private key is renamed as *coap-key.pem*.
- In PSK mode, the PSK is required by server. The PSK key file should be renamed to *coap_fota.psk* and uploaded into the */datatx/ssl* folder.

2.2.5.3. AT+QFOTADL=<FTP_URL> Update Firmware When Delta Package is Stored on FTP(S) Server

If the delta firmware package is stored on an FTP(s) server, **AT+QFOTADL=<FTP_URL>** should be executed to start automatic firmware update via DFOTA. Then the module will download the delta package from the FTP(s) server over the air and update the firmware automatically.

AT+QFOTADL=<FTP_URL> Update Firmware When Delta Package is Stored on FTP(S) Server	
Write Command **AT+QFOTADL=<FTP_URL>**	Response **OK** **+QIND: "FOTA","FTPSTART"** **+QIND: "FOTA","DOWNLOADING",<percent>** **+QIND: "FOTA","DOWNLOADING",<percent>** **...** **+QIND: "FOTA","FTPEND",<FTP_err>** **+QIND: "FOTA","RESETTING"** **+QIND: "FOTA","START"** **+QIND: "FOTA","UPDATING",<percent>** **+QIND: "FOTA","UPDATING",<percent>** **...** **+QIND: "FOTA","END",<update_err>** If there is any error: **+CME ERROR: <err>**
Maximum Response Time	300 ms
Characteristics	This command takes effect immediately. The configuration is not saved.

Parameter

<FTP_URL>	String type. The URL that the delta firmware package stored on the FTP(S) server. The maximum length is 700 bytes. It should be started with "FTP://" or "FTPS://", for example: **"FTP://<FTP_USER>:<FTP_PASS>@<FTP_server_URL>:<FTP_port>/<FTP_file_path>".**
<FTP_USER>	String type. The username for FTP(S) server login.
<FTP_PASS>	String type. The password of FTP(S) server login.
<FTP_server_URL>	String type. The IP address or domain name of FTP(S) server owned and operated by you.
<FTP_port>	Integer type. The port number of FTP(S) server. Default value: 21.

	Range: 1–65535.
<FTP_file_path>	String type. The file path in FTP(S) server.
<FTP_err>	Integer type. The FTP(S) error code. 0 Downloaded successfully. Any other value Errors. Refer to ***Chapter 4*** for more details.
<percent>	Integer type. The download or update progress in percentage.
<update_err>	Integer type. 0 Updated successfully. Any other value Errors. See ***Chapter 4*** for more details.
<err>	Error codes. See ***Chapter 4*** for more details.

NOTE

1. If SSL certificates are needed while the firmware package is downloaded from an FTPS server, it is necessary to upload the certificate to */datatx* in APP EFS. Before the upload, rename the certificates in accordance with the following rules:
   1) The root certificate must be renamed into *ftp_fota_cacert.pem;*
   2) The client certificate must be renamed into *ftp_fota_client_cert.pem*;
   3) The private key must be renamed into *ftp_fota_client_key.pem*.
2. Certificate upload methods:
   - Quectel QEFS Explorer tool can be used to upload the certificates. For more details about the tool, please refer to ***document [2]***.
   - **AT+QFUPL** can also be used to upload the certificates. For more details about the command, please refer to ***document [3]***.

2.2.5.4. AT+QFOTADL="COM:",<file_size>,<timeout> Upload a Delta Package to Flash

This command uploads a delta package to the NAND flash. If a delta package has existed in the NAND flash, it will be overwritten when a new package is uploaded. After this command is executed and **CONNECT** is returned, the module switches to the data mode. When the uploaded data reaches **<file_size>**, or there is no any data inputted when **<timeout>** reaches, it exits from the data mode automatically. During data transmission, inputting **+++** or pulling up MAIN_DTR can make the module exit from the data mode, and more details are provided in ***Chapter 2.2.4***.

AT+QFOTADL="COM:",<file_size>,<timeout> Upload a Delta Package to Flash	
Write Command **AT+QFOTADL="COM:",<file_size>[,<timeout>]**	Response **CONNECT** TA switches to the data mode (transparent access mode), and the delta package in the binary form can be inputted. When the total size of the inputted data reaches **<file_size>**, TA returns to the command mode and replies with the following codes:

	+QFOTADL: <upload_size>,<check_sum> **OK** If there is any error: **+CME ERROR: <err>**
Maximum Response Time	300 ms
Characteristics	The command takes effect immediately.

Parameter

<file_size>	Integer type. The file size expected to be uploaded. Unit: byte.
<timeout>	Integer type. The time waiting for data to be inputted to USB/UART. Default value: 5. Unit: second.
<upload_size>	Integer type. The file size that has been uploaded. Unit: byte.
<check_sum>	The checksum of the uploaded data.
<err>	Error code. See ***Chapter 4*** for more details.

NOTE

1. **<file_size>** is limited by the flash space for saving DFOTA package. Since the flash space for saving DFOTA package is shared by DFOTA algorithm and DFOTA package, **<file_size>** may be changed in the future. Besides, the maximum flash space for saving DFOTA package is calculated automatically by the module, and you can query it by **AT+QFOTADL=12**.
2. **<check_sum>** is a 16-bit checksum based on bitwise Exclusive-OR (XOR).
   If the number of the characters is odd, set the last character as the high 8 bit, and the low 8 bit to 0, and then use an XOR operator to calculate the checksum. **+++** sequence causes TA to end the command and switch to the command mode. However, the data previously uploaded is preserved into the file.
3. When executing the command, the data must be entered after **CONNECT** is returned.

2.2.5.5. AT+QFOTADL=<para1>[,<para2>[,<para3>]] Extended DFOTA Function

This command supports several extended functions through different parameters, such as triggering the DFOTA update with a local DFOTA firmware package, getting DFOTA firmware package information, and deleting a DFOTA firmware package.

AT+QFOTADL=<para1>[,<para2>[,<para3>]] Extended DFOTA Function	
Read Command **AT+QFOTADL?**	Response **+QFOTADL: (**list of supported **<state>**s**)**

	OK If **<state>** is 3, return the current delta package size downloaded by DFOTA: **+QFOTADL: 3,<delta_size>** **OK**
Write Command **AT+QFOTADL=<para1>[,<para2>[,<para3>]]**	Response If **<para1>**=1, start DFOTA update: **OK** If **<para2>** is omitted and **<para1>**=3, return the current DFOTA update trigger mode: **+QFOTADL: <para1>,<para2>** **OK** If **<para2>** is specified and **<para1>**=3, configure DFOTA update trigger mode: **OK** If **<para2>** is omitted and **<para1>**=4, return the size of local DFOTA package: **+QFOTADL: <para1>,<para2>** **OK** If **<para2>** is omitted and **<para1>**=5, delete the DFOTA package: **OK** **If <para2>** and **<para3>** are omitted and **<para1>**=8, query retry times and intervals: **+QFOTADL: <para1>,<para2>,<para3>** **OK** **If <para2>** and **<para3>** are present and **<para1>**=8, configure retry times and intervals: **OK** If **<para3>** is omitted, **<para2>** is present and **<para1>**=10, configure downloading files to EUFS from HTTP server, the

	downloaded file path will be renamed as */datatx/update.zip*. **OK** If **<para3>** and **<para2>** are present and **<para1>**=10, configure downloading files to EUFS from HTTP server, the downloaded file path will be renamed as **<para3>**. **OK** **If <para2>** and **<para3>** are omitted and **<para1>**=12, get the maximum size for saving DFOTA package**:** **+QFOTADL: <para1>,<para2>** **OK** If **<para3>** is omitted, **<para1>** and **<para2>** are present and **<para1>**=13, configure DFOTA to support resuming downloading in the next power-on. If **<para1>**, **<para2>** and **<para3>** are present and **<para1>**=13, configure DFOTA to support resuming downloading in the next power-on, either from the beginning or from the last downloaded point. If there is any error: **+CME ERROR: <err>** If there is any other error: **ERROR**
Maximum Response Time	10 s
Characteristics	The command takes effect immediately. The configurations are not saved.

Parameter

<para1>	Integer type. 1 Start DFOTA update 2 Reserved 3 Configure whether to trigger DFOTA update immediately after DFOTA package is downloaded 4 Get the size of local DFOTA package 5 Delete the local DFOTA package 6 Cancel the current DFOTA package download 7 Reserved

	8 Configure retry times and intervals when the connection breaks down 9 Reserved 10 Download file from HTTP(S) server and save it into alternate file system 11 Reserved 12 Get the maximum size of storage space for storing DFOTA package 13 Configure DFOTA to support resuming downloading in the next power-on
<para2>	If **<para1>**=3: 0 Trigger DFOTA update with **AT+QFOTADL=1**. 1 Trigger DFOTA update automatically when the package is downloaded. If **<para1>**=4: **<para2>** represents the size of local DFOTA package. Unit: byte. If **<para1>**=8: **<para2>** is used to configure retry times. Range:1–2147483647. If **<para1>**=10: **<para2>** represents the URL address of file stored on the HTTP(S) server. If **<para1>**=12: **<para2>** represents the maximum size for saving DFOTA package. Unit: byte. If **<para1>**=13: **<para2>** represents the URL address of file stored on the HTTP(S) server. In other cases, **<para2>** is omitted.
<para3>	If **<para1>**=8: **<para3>** is used to configure the time interval between two retries. Unit: second. If **<para1>**=10: **<para3>** represents the file name for saving the downloaded file, and the local file path is */datatx/* in EUFS. If **<para1>**=13: **<para3>** is used to configure downloading from the beginning or last downloaded point in the next power-on. If the **<para3>** is omitted or is 0: **<para3>** represents downloading from last downloaded point. If the **<para3>** is 1, **<para3>** represents downloading from the beginning. In other cases, **<para3>** is omitted.
<state>	Integer type. DFOTA package download. 0 Not start downloading 1 Downloading 2 Downloaded 3 Download failure
<delta_size>	Integer type. The current delta package size downloaded by DFOTA.
<err>	Error code. See ***Chapter 4*** for more details.

2.2.5.6. AT+QFOTADL=<file_name>,<type> Copy Delta Package to Flash from File System

This command copies delta package to the Nand flash that saves delta package from Modem or AP file system. If the delta package has existed on the Nand flash, it will be overwritten by a new package.

AT+QFOTADL=<file_name>,<type>	Copy Delta Package to Flash from File System
Write Command **AT+QFOTADL=<file_name>,<type>**	Response **OK** If there is any error: **+CME ERROR: <err>**
Maximum Response Time	/
Characteristics	The command takes effect immediately.

Parameter

<file_name>	String type. Name of the file to be copied. The maximum length is 128 bytes. "UFS:**<file_name>**" The file from Modem file storage to FOTA partition "EUFS:**<file_name>**" The file from AP file storage to FOTA partition
<type>	Integer type. The type of file to be copied to FOTA partition. 0 The first file to FOTA partition 1 The intermediate file to FOTA partition 2 The last file to FOTA partition
<err>	Error Codes. See ***Chapter 4*** for more details.

NOTE

1. This command is useful in the following situations:
   a) When the DFOTA download command provided by Quectel is not used to download the delta package and store it in the file system, this command can be used to copy the delta package data into the flash inside the module for subsequent system upgrade.
   b) When the delta package is transmitted into the module file system in a wired way through external devices (such as MCU), this command can be used to copy the delta package data into the flash inside the module for subsequent system upgrade.
2. For details on UFS and EUFS, please refer to ***document [3]***.
3. If there is only one file in UFS/EUFS, configure **<type>** as 2.

2.2.6. AT+QCFGEXT="fota_apn" Configure IP Family and APN for DFOTA

This command configures the IP family and APN for DFOTA downloading service.

AT+QCFGEXT="fota_apn" Configure IP Family and APN for DFOTA	
Write Command **AT+QCFGEXT="fota_apn",[<IP_type>,<APN>,[<username>,<password>]]**	Response If the optional parameters are omitted, query the current setting: **+QCFGEXT: "fota_apn",<IP_type>,<APN>,<username>,<password>** **OK** If any of the optional parameters is specified, configure the IP family and APN for DFOTA: **OK** If there is any error: **ERROR**
Maximum Response Time	300 ms
Characteristics	The command takes effect immediately. The configurations are saved automatically.

Parameter

<IP_type>	Integer type. IP family. 0 IPv4 address family 1 IPv6 address family 2 IPv4 and IPv6 address family
<APN>	String type. Access point name.
<username>	String type. Username of the selected APN.
<password>	String type. Password of the selected APN.

2.2.7. AT+QCFGEXT="fota_http_header" Configure HTTP(S) Header for DFOTA over HTTP(S)

This command configures HTTP(S) header for DFOTA over HTTP(S).

AT+QCFGEXT="fota_http_header" Configure HTTP(S) Header for DFOTA over HTTP(S)	
Write Command **AT+QCFGEXT="fota_http_header"[,<k**	Response If the optional parameters are omitted, query the current

ey>,<value>]	setting: **+QCFGEXT: "fota_http_header"[,<key>,<value>]** **OK** If the optional parameters are specified, configure HTTP header for DFOTA over HTTP(S): **OK** If there is any error: **+CME ERROR: <err>**
Maximum Response Time	300 ms
Characteristics	The command takes effect immediately. The configurations are not saved.

Parameter

<key >	String type. Key of HTTP(S) header.
<value>	String type. Value of HTTP(S) header.
<err>	Error codes. Refer to ***Chapter 4*** for more details.

NOTE

The maximum number of HTTP(S) header is 5.

2.2.8. AT+QCFGEXT="fota_wd_gpio" Configure Watch Dog Pin and Feeding Interval During DFOTA

As the DAM cannot run during DFOTA process, you can't feed the watch dog in the DAM App if external watch dog is used, and it may cause the module to reset by external watch dog because of watch dog timeout. In order to avoid this problem, Quectel implements a feature to feed the watch dog automatically in AP kernel during DFOTA. This command enables/disables the external watch dog feeding and configures the pin and the feeding interval.

AT+QCFGEXT=" fota_wd_gpio" Configure Watch Dog Pin and Feeding Interval During DFOTA	
Write Command **AT+QCFGEXT="fota_wd_gpio"[,<switch>,<pin_num>,<feed_interval>]**	Response If the optional parameters are omitted, query the current setting: **+QCFGEXT: "fota_wd_gpio",<switch>[,<pin_num>,<feed_interval>]**

	OK If the optional parameters are specified, configure the watch dog pin and feeding interval during DFOTA **OK** If there is any error: **ERROR**
Maximum Response Time	300 ms
Characteristics	The command takes effect immediately. The configurations are saved automatically.

Parameter

<switch >	Integer type. Enable/disable DFOTA watch dog. 0 Disable 1 Enable
<pin_num>	Integer type. GPIO pin number for watch dog. BG95 series module supports the pin numbers below: 4–7, 18, 19, 22, 23,25–28, 40, 41, 64–66, 85–88. BG600L-M3 supports the pin numbers below: 9–12, 22,23, 29,30, 53,54, 57–62. BG77 supports the pin numbers below: 1–5, 8, 9, 33–37, 40, 41, 48–51, 57, 60, 61, 63, 67–71, 77, 80–82, 90, 91, 93.
<feed_interval>	Integer type. The interval to feed the watch dog. The period of square wave is 2***<feed_interval>**. Unit: millisecond. Range: 100–3600000.
<err>	Error codes. See ***Chapter 4*** for more details.

2.2.9. AT+QCFGEXT="sni" Configure Whether to Enable Server Name Indication for DFOTA

AT+QCFGEXT="sni" Configure Whether to Enable Server Name Indication for DFOTA	
Write Command **AT+QCFGEXT="sni"[,<SNI>]**	Response If the optional parameter is omitted, query the current setting: **+QCFGEXT: "sni",<SNI>** **OK** If the optional parameter is specified, configure whether to enable server name indication for DFOTA: **OK**

	If there is any error: **ERROR**
Maximum Response Time	300 ms
Characteristics	The command takes effect immediately. The configuration is not saved.

Parameter

<SNI>	Integer type. Whether to enable server name indication. 0 Disable 1 Enable

2.3. Examples

2.3.1. Update Firmware from HTTP(S) Server by DFOTA

//You can perform the firmware upgrade after storing the delta firmware package on your HTTP(S) server. "https://www.example.com:100/update.zip" is used as an example URL below (The URL is provided for illustrative purpose only. Please replace it with a valid URL that corresponds to your HTTP(S) server and firmware package).

```
AT+CSQ;+CEREG?;+CGREG?;+COPS?     //Query network status.
+CSQ: 26,99

+CEREG: 0,1

+CGREG: 0,4

+COPS: 0,0,"Telstra Mobile Telstra",8

OK
```

//Execute **AT+QFOTADL** command to start automatic firmware update via DFOTA, and then the module downloads the delta package and updates firmware automatically.

```
AT+QFOTADL="https://www.example.com:100/update.zip"
OK
+QIND: "FOTA","HTTPSTART"
+QIND: "FOTA","DOWNLOADING",5%
…
+QIND: "FOTA","HTTPEND",0          //Finish downloading the package from the HTTP(S) server.
+QIND: "FOTA","RESETTING"          //The module is reset and then enters DFOTA mode.
+QIND: "FOTA","START"
+QIND: "FOTA","UPDATING",1%
```

```
+QIND: "FOTA","UPDATING",2%
...
+QIND: "FOTA","UPDATING",100%
+QIND: "FOTA","END",0                    //Finish updating the firmware.
```

2.3.2. Update Firmware from FTP(S) Server by DFOTA

```
//You can perform the firmware upgrade after storing the delta firmware package on your FTP(S) server. "ftp://test:test@192.0.2.2:8309/TEST1/upgrade.bin" is used as an example URL below. (The URL is provided for illustrative purposes only. Please replace it with a valid URL that corresponds to your FTP server and firmware package.) Execute AT+QFOTADL command to start automatic firmware update via DFOTA, and then the module downloads the delta package and updates the firmware automatically.
AT+QFOTADL="ftp://test:test@192.0.2.2:8309/TEST1/upgrade.bin"
OK
+QIND: "FOTA","FTPSTART"
+QIND: "FOTA","DOWNLOADING",5%
+QIND: "FOTA","DOWNLOADING",10%
+QIND: "FOTA","DOWNLOADING",15%
+QIND: "FOTA","DOWNLOADING",20%
...
+QIND: "FOTA","DOWNLOADING",95%
+QIND: "FOTA","DOWNLOADING",100%
+QIND: "FOTA","FTPEND",0                  //Finish the package downloading from FTP(S) server.
+QIND: "FOTA","RESETTING"                 //The module resets and then enters DFOTA mode.
+QIND: "FOTA","START"
+QIND: "FOTA","UPDATING",1%
+QIND: "FOTA","UPDATING",4%
...
+QIND: "FOTA","UPDATING",99%
+QIND: "FOTA","UPDATING",100%
+QIND: "FOTA","END",0                     //Finish the firmware update.
```

2.3.3. Copy Delta Package to FOTA Partition from EUFS

```
//Assume that the DFOTA package is divided the into four parts, update1.bin, update2.bin, update3.bin and update4.bin.
//Download or uploaded update1.bin toEUFS.
AT+QFOTADL="EUFS:update1.bin",0     //Copy update1.bin to FOTA partition.
OK
//Download or uploaded update2.bin to EUFS
AT+QFOTADL="EUFS:update2.bin",1    //Copy update2.bin to FOTA partition.
OK
//Download or uploaded update3.bin to EUFS
```

```
AT+QFOTADL="EUFS:update3.bin",1    //Copy update3.bin to FOTA partition.
OK
//Download or uploaded update4.bin to EUFS
AT+QFOTADL="EUFS:update4.bin",2    //Copy update4.bin to FOTA partition.
OK
```

2.3.4. Set Breakpoint Download and Single Request Size for HTTP(S) DFOTA Downloading

```
//Update firmware when delta firmware package is stored on HTTP(S) server.
"https://www.example.com:100/update.zip" is used as an example URL below. (The URL is provided for illustrative purpose only. Please replace it with a valid URL that corresponds to your HTTP(S) server and firmware package.)
AT+CSQ;+CEREG?;+CGREG?;+COPS?          //Query network status
+CSQ: 26,99

+CEREG: 0,1

+CGREG: 0,4

+COPS: 0,0,"Telstra Mobile Telstra",8

OK
```

//Set breakpoint download and single request size. Execute **AT+QFOTADL** command to start automatic firmware update via DFOTA, and then the module downloads the delta package and update firmware automatically.

```
AT+QFOTADL="https://www.example.com:100/update.zip",0,10240
OK
+QIND: "FOTA","HTTPSTART"
+QIND: "FOTA","DOWNLOADING",5%
…
+QIND: "FOTA","HTTPEND",712        //Finish a single download of the package from the HTTPS server.
AT+QFOTADL?
+QFOTADL:3,10240              //Query the size of a single download package from the HTTPS server.

OK
```

```
//Set the breakpoint position to 10240 and the request size to 0 by default, and download the package from the breakpoint until all packages are downloaded completely.
AT+QFOTADL="https://www.example.com:100/update.zip",10240
OK
+QIND: "FOTA","HTTPSTART"
+QIND: "FOTA","DOWNLOADING",10%
```

```
…
+QIND: "FOTA","DOWNLOADING",100%
+QIND: "FOTA","HTTPEND",0
+QIND: "FOTA","RESETTING"                    //The module resets and then enters DFOTA mode.
+QIND: "FOTA","START"
+QIND: "FOTA","UPDATING",1%
+QIND: "FOTA","UPDATING",2%
...
+QIND: "FOTA","UPDATING",100%
+QIND: "FOTA","END",0                        //Finish the firmware update.
```

2.3.5. Configure HTTP(S) Header for DFOTA over HTTP(S)

```
//Set the user-defined HTTP header configuration.
AT+QCFGEXT="fota_http_header","Auth","test"
OK
//Query the user-defined HTTP header configuration.
AT+QCFGEXT="fota_http_header"
+QCFGEXT: key:Auth    value:test
OK
//Delete one of the user-defined HTTP header configuration.
AT+QCFGEXT="fota_http_header","Auth",""
OK
```

2.3.6. Set Watch Dog Configuration for DFOTA

```
//Set the watch dog configuration.
AT+QCFGEXT="fota_wd_gpio",1,6,3000    //Enable the watch dog, and set pin number to 6 and feed
OK                                     intervals to 3000 milliseconds.
AT+QCFGEXT="fota_wd_gpio",0           //Disable DFOTA watch dog.
OK
AT+QCFGEXT="fota_wd_gpio"             //Query DFOTA watch dog configuration.
"fota_wd_gpio",1,6,3000

OK
```

2.3.7. Configure Support Resuming Downloading for HTTP(S) DFOTA

```
//Update firmware when delta firmware package is stored on HTTP(S) server.
"https://www.example.com:100/update.zip" is used as an example URL below.(The URL is provided for
illustrative purpose only. Please replace it with a valid URL that corresponds to your HTTP(S) server and
firmware package.).
AT+CSQ;+CEREG?;+CGREG?;+COPS?          //Query network status
```

```
+CSQ: 26,99

+CEREG: 0,1

+CGREG: 0,4

+COPS: 0,0,"Telstra Mobile Telstra",8

OK
AT+QFOTADL=13,"https://www.example.com:100/update.zip"
OK

+QIND: "FOTA","HTTPSTART"
+QIND: "FOTA","DOWNLOADING",5%
+QIND: "FOTA","DOWNLOADING",10%
+QIND: "FOTA","DOWNLOADING",15%
RDY                                          //power off the module

APP RDY
AT+QFOTADL=13,"https://www.example.com:100/update.zip"
OK

+QIND: "FOTA","HTTPSTART"
+QIND: "FOTA","DOWNLOADING",15%
…
+QIND: "FOTA","DOWNLOADING",100%
+QIND: "FOTA","HTTPEND",0
+QIND: "FOTA","RESETTING"                    //The module resets and then enters DFOTA mode.
+QIND: "FOTA","START"
+QIND: "FOTA","UPDATING",1%
+QIND: "FOTA","UPDATING",2%
...
+QIND: "FOTA","UPDATING",100%
+QIND: "FOTA","END",0                        //Finish the firmware update.
```

3 Trigger DFOTA with LwM2M

3.1. Firmware Update Procedure

OMA defines the application layer communication protocol between a LwM2M server and a LwM2M client, which is located in a LwM2M device. For the modules, the LwM2M protocol is used to manage device provisioning and is a trigger of a DFOTA procedure. For the DFOTA update on the modules, a firmware update object is provided by the LwM2M specification.

The following are the routine DFOTA steps with LwM2M:

1. Write “Package”/”Package URI” to trigger the downloading of a firmware image, so that the DFOTA application downloads the firmware image from LwM2M server automatically.
2. Post “Update” to trigger firmware update on the modules, so that the DFOTA application finishes firmware update automatically with the candidate firmware image.

The LwM2M client is notified about the new firmware by the LwM2M server using CoAP(S). The client sends indication to a registered DFOTA application. The DFOTA application downloads the firmware delta package from the server and stores it on the file system of BG95 series, BG77 and BG600L-M3 modules.

The following figure shows key components involved in the LwM2M DFOTA process on the modules.

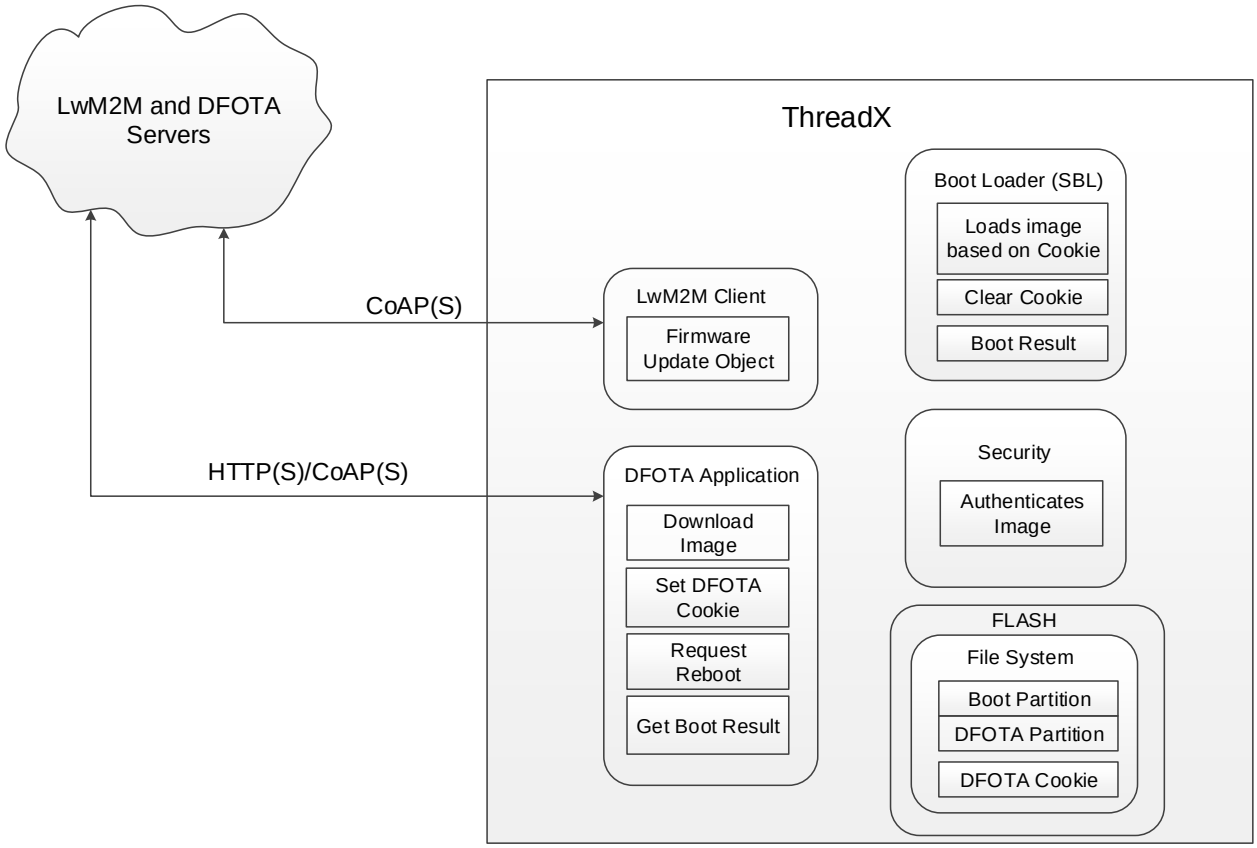

Figure 2: LwM2M Firmware Image Download

3.1.1. Download a Firmware Image

About “Firmware Update Object” in LwM2M, the detailed interaction process between LwM2M client and LwM2M server is described as below.

The LwM2M client should register and observe the “Package URI” resource in the Firmware Update Object, which is shown in the server when the client is registered successfully.

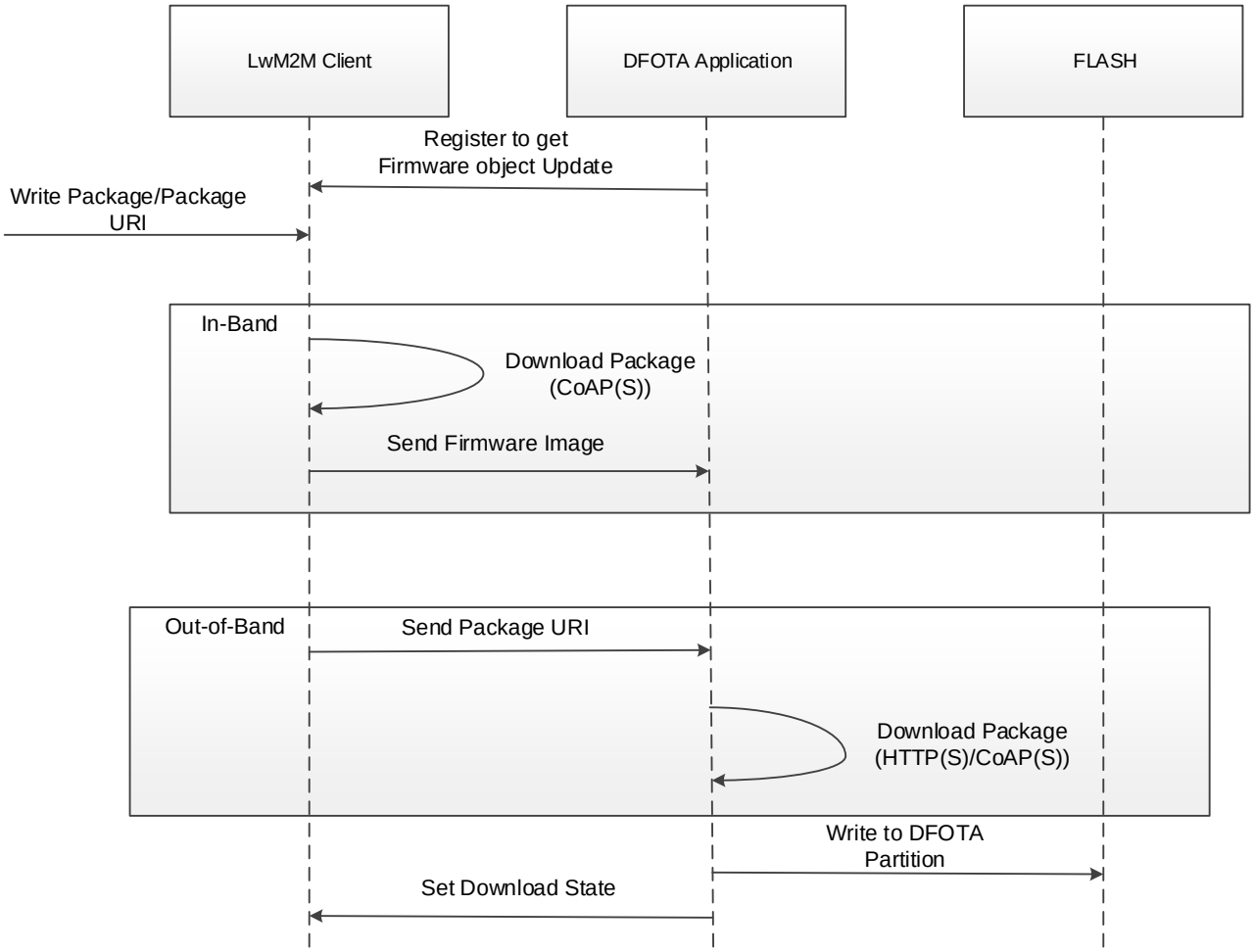

Figure 3: Call Flow of Package URI Observing and Firmware Downloading

NOTE

In the in-band mode, the module downloads a firmware package from the LwM2M server directly via the CoAP(S) protocol and sends the downloaded package to the DFOTA application. All "firmware update" actions are done in the application. But in the out-of-band mode, the module downloads a firmware package from the HTTP(S)/CoAP(S)/FTP(S) server to the module.

3.1.2. Update the Firmware

The LwM2M server should observe the “State” resource in the Firmware Update Object. If the “State” has changed to “Downloaded”, “Update” resource is executed to trigger the firmware update process.

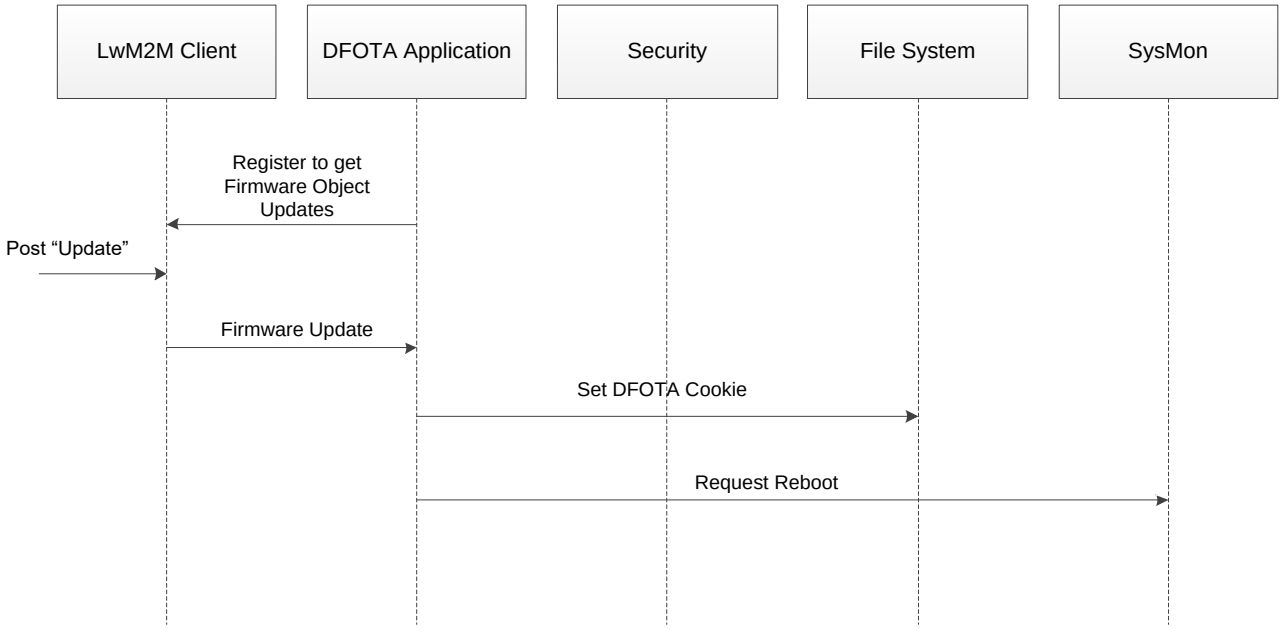

Figure 4: Call Flow of “Update” Resource Observing and Update Event Processing

3.2. URC of Firmware Update with LwM2M

3.2.1. Download Progress URCs

3.2.1.1. In-Band Mode

When the DFOTA firmware update is triggered in the LwM2M in-band mode, the following URCs are outputted to notify the download progress.

+LWM2M: "FOTA","DOWNLOADING" Notify the Download Progress	
URC Format: **+LWM2M: "FOTA","DOWNLOADING","START"**	Download process started.
URC Format: **+LWM2M: "FOTA","DOWNLOADING","COMPLETED"**	Download process completed.

3.2.1.2. Out-of-Band HTTP(S) Mode

When the DFOTA firmware update is triggered in the LwM2M out-of-band HTTP(S) mode, the following URCs are outputted to notify the download progress.

+QIND: "FOTA","HTTP" Notify the Download Progress	
URC Format: **+QIND: "FOTA","HTTPSTART"**	Download process started.
URC Format: **+QIND: "FOTA","DOWNLOADING",<percent>** **...**	Download progress in percentage.
URC Format: **+LWM2M: "FOTA","DOWNLOADED",<HTTP_err>**	Download operation result.

Parameter

<percent>	Integer type. The download progress in percentage.
<HTTP_err>	Integer type. The HTTP(S) error code. 0 Downloaded successfully. Any other value Error. Refer to ***Chapter 4*** for more details.

3.2.1.3. Out-of-Band CoAP(S) Mode

When the DFOTA firmware update is triggered in the LwM2M out-of-band CoAP(S) mode, the following URCs are outputted to notify the download progress.

+QIND: "FOTA","COAP" Notify the Download Progress	
URC Format: **+QIND: "FOTA","COAPSTART"**	Download process started.
URC Format: **+LWM2M: "FOTA","DOWNLOADED",<CoAP_err>**	Download operation result.

Parameter

<CoAP_err>	Integer type. The CoAP(S) error code. 0 Downloaded successfully. Any other value Error. Refer to ***Chapter 4*** for more details.

3.2.2. Update Progress URCs

3.2.2.1. +QIND: "FOTA","START" Updating Started

+QIND: "FOTA","START" Updating Started	
URC Format: **+QIND: "FOTA","START"**	Notify the start of the updating.

3.2.2.2. +QIND: "FOTA","UPDATING",<percent> Updating Progress

+QIND: "FOTA","UPDATING",<percent> Updating Progress	
URC Format: **+QIND: "FOTA","UPDATING",<percent>**	Notify the updating progress in percentage.

Parameter

<percent>	Integer type. The updating progress in percentage.

3.2.2.3. +QIND: "FOTA","END",<err> Updating Finished

+QIND: "FOTA","END",<err> Updating Finished	
URC Format: **+QIND: "FOTA","END",<err>**	Notify the completion of updating.

Parameter

<err>	Error code. See ***Chapter 4*** for more details.

4 Summary of Error Codes

These error codes indicate errors related to mobile equipment or network. The details about **<HTTP_err>**, **<CoAP_err>**,**<FTP_err>**,**<update_err>** and **<err>** are described in the following tables.

Table 2: Summary of <HTTP_err>/<CoAP_err>/<FTP_err> Codes

<HTTP_err>/<CoAP_err>/<FTP_err>	Meaning
0	Download successfully
701	Unknown error
702	Server connection failed
703	Request failed
704	Download timeout
705	URL error
706	File does not exist
707	Failed to write data to file
708	Downloaded file is too large
709	Download cancelled
710	Downloaded package check failed
711	Data call disconnected
712	Indicates that it is a custom segmented download

Table 3: Summary of <update_err> Codes

<update_err>	Meaning
0	Updated successfully

504	Firmware update failed
505	Update package does not exist
506	Update package check failed
507	Decompress failed
508	Compress failed
509	Failed to copy compress data
510	Restore failed
511	Package is mismatched with the current firmware
512	DFOTA unknown error

Table 4: Summary of <err> Codes

<err>	Meaning
590	Invalid input parameter
591	DFOTA is ongoing
592	DFOTA is not in downloading
593	DFOTA package does not exist or failed to open
594	DFOTA memory allocation failed

5 Appendix References

Table 5: Related Documents

Document Name
[1] Quectel_BG95&BG77&BG600L_Series_AT_Commands_Manual
[2] Quectel_BG95&BG77&BG600L_Series_QEFS_Explorer_User_Guide
[3] Quectel_BG95&BG77&BG600L_Series_FILE_Application_Note

Table 6: Terms and Abbreviations

Abbreviation	Description
AP	Application Processor
APN	Access Point Name
bps	Bit(s) Per Second
CoAP(S)	Constrained Application Protocol (Secure)
DFOTA	Delta Firmware Upgrade Over-the-Air
EFS	Embedded File System
EGPRS	Enhanced General Packet Radio Service
FOTA	Firmware Over-the-Air
HTTP(S)	Hyper Text Transport Protocol (Secure)
FTP(s)	File Transfer Protocol (Secure)
IP	Internet Protocol
IPv4	Internet Protocol Version 4
IPv6	Internet Protocol Version 6

KB	Kilobytes
LTE	Long Term Evolution
LPWA	Low Power Wide Area
LwM2M	Lightweight Machine to Machine
NAND	Not And
OMA	Open Mobile Alliance
SSL	Secure Sockets Layer
URC	Unsolicited Result Code
URI	Uniform Resource Identifier