Build a Smarter World

M95 FTPS

Application Note

GSM/GPRS Module Series

Rev. M95_FTPS_Application_Note_V1.0

Date: 2018-11-20

Status: Released

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.
7th Floor, Hongye Building, No.1801 Hongmei Road, Xuhui District, Shanghai 200233, China
Tel: +86 21 5108 6236
Email: info@quectel.com

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

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

GENERAL NOTES

QUECTEL OFFERS THE INFORMATION AS A SERVICE TO ITS CUSTOMERS. THE INFORMATION PROVIDED IS BASED UPON CUSTOMERS' REQUIREMENTS. QUECTEL MAKES EVERY EFFORT TO ENSURE THE QUALITY OF THE INFORMATION IT MAKES AVAILABLE. QUECTEL DOES NOT MAKE ANY WARRANTY AS TO THE INFORMATION CONTAINED HEREIN, AND DOES NOT ACCEPT ANY LIABILITY FOR ANY INJURY, LOSS OR DAMAGE OF ANY KIND INCURRED BY USE OF OR RELIANCE UPON THE INFORMATION. ALL INFORMATION SUPPLIED HEREIN IS SUBJECT TO CHANGE WITHOUT PRIOR NOTICE.

COPYRIGHT

About the Document

History

Revision	Date	Author	Description
1.0	2018-11-20	Jaryoung LI	Initial

Contents

Table Index

1 Introduction

File Transfer Protocol with SSL Secure (FTPS) is a combination of the File Transfer Protocol (FTP) with SSL/TLS protocols to provide encrypted communication and secure identification of an FTPS server.

In order to ensure communication privacy, the communication between a server and a client should be in an encrypted way so as to prevent data from eavesdropping, tampering or forging during the communication process.

FTPS function can add security capabilities of SSL/TLS protocols to standard FTP communication through simply layering the FTP on the top of the SSL/TLS protocols.

This document mainly introduces how to use the FTPS function of Quectel M95 module.

1.1. SSL/TLS Versions and Cipher Suites

The following are SSL/TLS versions supported by M95 module currently.

- SSL3.0
- TLS1.0
- TLS1.1
- TLS1.2

The following table shows TLS cipher suites supported by M95 module. For detailed description of cipher suites, please refer to *RFC 2246-The TLS Protocol Version 1.0*.

Table 1: Supported TLS Cipher Suites

Codes of Cipher Suites	Names of Cipher Suites
0X0035	TLS_RSA_WITH_AES_256_CBC_SHA
0X0005	TLS_RSA_WITH_RC4_128_SHA
0X0004	TLS_RSA_WITH_RC4_128_MD5
0X000A	TLS_RSA_WITH_3DES_EDE_CBC_SHA
0X002F	TLS_RSA_WITH_AES_128_CBC_SHA
0X003D	TLS_RSA_WITH_AES_256_CBC_SHA256

1.2. The Procedure of Using SSL Function

Step 1: Install certificate and key to RAM or NVRAM by **AT+QSECWRITE** command. **AT+QSECDEL** command is used to delete the certificate and key, and **AT+QSECREAD** command is used to check the checksum of certificate and key. If authentication for server and client is not needed, please skip this step.

Step 2: Configure the APN, username, password of context by **AT+QICSGP** command. **AT+QIREGAPP** command is used to register on TCP/IP stack. For details, please refer to ***document [1].***

Step 3: Activate GPRS PDP context by **AT+QIACT** command, then the local IP address can be queried by **AT+QILOCIP** command. For details, please refer to ***document [1].***

Step 4: Configure SSL version, cipher suit, server authentication, client authentication, CA certificate, client certificate and client key by **AT+QSSLCFG** command.

Step 5: Configure FTPS mode, enable FTPS function and configure SSL context index by **AT+QSSLCFG** command.

Step 6: Upload a file to or download a file from the FTPS server. For more details, please refer to the ***document [2]***.

2 Description of FTPS AT Commands

2.1. AT Command Syntax

Table 2: Types of AT Commands and Response

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.
Execute Command	**AT+<*x*>**	This command reads non-variable parameters affected by internal processes in the GSM engine.

2.2. AT+QFTPCFG Configure Parameters for FTP(S) Server

This command is used to configure the parameters for FTP(S) server.

AT+QFTPCFG Configure Parameters for FTP(S) Server	
Write Command **AT+QFTPCFG=<type>[,<value>]**	Response If **<value>** is present and **<type>** is legal **OK** **+QFTPCFG: 0** If **<value>** is omitted and **<type>** is legal, query the current setting of FTP(S) server by **<type>** **+QFTPCFG: <value>** If there is any error **ERROR** or

	+QFTPCFG: <err>
Reference	

Parameter

<type>	The types of configurable parameters to be set. 1 The mode of data connection 2 The transfer type 3 The resuming point to resume file transfer 4 The local position of the file to transfer 5 FTP(S) non-transparent mode 6 Ignore FTP(S) data link address returned by server in passive mode 7 The connection mode of FTPS
<value>	The value of parameters to be set. If **<type>**=7, 0 Implicit connection 1 Explicit connection For **<type>**=1-6, please refer to ***document [2]*** for detailed information.

2.3. AT+QSSLCFG SSL Configuration

This AT command is used to configure the following parameters: SSL version, cipher suite, secure level, server root certificate, client certificate, client key, RTC time ignorance and SSL context index of FTPS. These parameters will be used in the handshake procedure.

The parameter <ctxindex> is the index of the SSL context. The module supports six SSL contexts at most. And on the basis of an SSL context, several SSL connections can be established. The settings such as SSL version and cipher suite are stored in the SSL context, and they will be applied to a new SSL connection which is associated with the SSL context.

AT+QSSLCFG SSL Configuration	
Test Command **AT+QSSLCFG=?**	Response **+QSSLCFG: "type",(0-5),"value"** **OK**
Write Command Query the context setting **AT+QSSLCFG="ctxindex",<ctxindex>**	Response **+QSSLCFG: <ctxindex>,<sslversion>,<seclevel>,<ciphersuite>,<cacertname>,<clientcertname>,<clientkeyname>** **OK**

	If there is any error **ERROR**
Write Command Configure SSL/TLS version **AT+QSSLCFG="sslversion",<ctxindex>[,<sslversion>]**	Response If **<sslversion>** is present **OK** If **<sslversion>** is omitted, query the SSL/TLS version **+QSSLCFG: "sslversion",<sslversion>** **OK** If there is any error **ERROR**
Write Command Configure cipher suite **AT+QSSLCFG="ciphersuite"<ctxindex>[,**list of supported **<ciphersuite>**s]	Response If **<ciphersuite>** is present **OK** If **<ciphersuite>** is omitted, query the cipher suite **+QSSLCFG: "ciphersuite",<ciphersuite>** **OK** If there is any error ERROR
Write Command Configure authentication mode **AT+QSSLCFG="seclevel",<ctxindex>[,<seclevel>]**	Response If **<seclevel>** is present **OK** If **<seclevel>** is omitted, query the SSL/TLS verification level **+QSSLCFG: "seclevel",<seclevel>** **OK** If there is any error **ERROR**
Write Command Configure the path of root certificate **AT+QSSLCFG="cacert",<ctxindex>[,<cacertname>]**	Response If **<cacertname>** is present **OK** If **<cacertname>** is omitted, query the server CA certificate **+QSSLCFG: "cacert",<cacertname>** **OK**

	If there is any error ERROR
Write Command Configure the path of client certificate **AT+QSSLCFG="clientcert",<ctxindex>[,<clientcertname>]**	Response If **<clientcertname>** is present **OK** If **<clientcertname>** is omitted, query the client certificate **+QSSLCFG: "clientcert",<clientcertname>** **OK** If there is any error **ERROR**
Write Command Configure the path of client key **AT+QSSLCFG="clientkey",<ctxindex>[,<clientkeyname]**	Response If **<clientkeyname>** is present **OK** If **<clientkeyname>** is omitted, query the client private key **+QSSLCFG: "clientkey",<clientkeyname>** **OK** If there is any error ERROR
Write Command Configure whether to ignore the RTC time **AT+QSSLCFG="ignorertctime"[,<ignorertctime>]**	Response If **<ignorertctime>** is present **OK** If **<ignorertctime>** is omitted, query the setting of RTC time ignorance **+QSSLCFG: "ignorertctime",<ignorertctime>** **OK** If there is any error **ERROR**
Write Command Enable/Disable the FTPS function **AT+QSSLCFG="ftps"[,<ftpsenable>]**	Response If **<ftpsenable>** is present **OK** If **<ftpsenable>** is omitted, query the enabling or disabling setting of FTPS function **+QSSLCFG: "ftps",<ftpsenable>**

	OK If there is any error ERROR
Write Command Configure SSL/TLS context index for FTPS **AT+QSSLCFG="ftpsctxi"[,<ftpsctxindex>]**	Response If **<ftpsctxindex>** is present **OK** If **<ftpsctxindex>** is omitted, query the SSL/TLS context for FTPS **+QSSLCFG: "ftpsctxi",<ftpsctxindex>** **OK** If there is any error ERROR
Reference	

Parameter

<ctxindex>	SSL context index. Range: 0-5	
<sslversion>	The supported SSL/TLS versions.	
	0	SSL3.0
	1	TLS1.0
	2	TLS1.1
	3	TLS1.2
	4	All Supported
<ciphersuite>	Codes and names of cipher suites.	
	0X0035	TLS_RSA_WITH_AES_256_CBC_SHA
	0X002F	TLS_RSA_WITH_AES_128_CBC_SHA
	0X0005	TLS_RSA_WITH_RC4_128_SHA
	0X0004	TLS_RSA_WITH_RC4_128_MD5
	0X000A	TLS_RSA_WITH_3DES_EDE_CBC_SHA
	0X003D	TLS_RSA_WITH_AES_256_CBC_SHA256
<seclevel>	SSL/TLS verification level decided by the requirement of a remote server.	
	0	No authentication
	1	Server authentication
	2	Server authentication and client authentication
<cacertname>	String format. The server CA certificate. For details of the parameter format, please refer to ***Table 3***.	
<clientcertname>	String format. The client certificate. For details of the parameter format, please refer to ***Table 3.***	

<clientkeyname> String format. The client private key. For details of the parameter format, please refer to ***Table 3.***

<ignorertctime> RTC time ignorance. The parameter is used to configure whether to ignore the RTC time or not.

0 Do not ignore the RTC time
1 Ignore the RTC time

<ftpsenable> Enable/disable the FTPS function

0 Disable FTPS
1 Enable FTPS

<ftpsctxindex> Configure the SSL/TLS context for FTPS. The range of SSL context is 0-5. If the host does not configure the <ftpsctxindex>, the value of <ftpsctxindex> is -1.

Table 3: Formats of <cacertname>, <clientcertname> and <clientkeyname>

When Upload the Files to RAM			
Parameter	**Format**	**Description**	**Comment**
<cacertname>	RAM:ca_cert.pem	Identify a server root certificate	
<clientcertname>	RAM:client_cert.pem	Identify a client certificate	
<clientkeyname>	RAM:client_key.pem	Identify a client private key	
When Upload the Files to NVRAM			
Parameter	**Format**	**Description**	**Comment**
<cacertname>	NVRAM:CA[0,1]	Identify a CA certificate	Two CA certificates can be uploaded: client certificate and client private key. The filename of CA certificate must be CA0 or CA1.
<clientcertname>	NVRAM:CC0	Identify a client certificate	The filename of client certificate must be CC0.
<clientkeyname>	NVRAM:CK0	Identify a client private key	The filename of client private key must be CK0.

NOTE

If no authentication is set, security data will not be needed. If server authentication has been set, customers need to configure server CA certificate. If both server and client authentications have been set, customers need to configure client certificate, server CA certificate and client private key.

2.4. AT+QSECWRITE Upload a Certificate or Key

This command is used to upload server CA certificate, client certificate and client private key to RAM or NVRAM. And the certificate and key will be stored in these storage devices in an encrypted way. After the certificate and key are stored in these storages, the host cannot read the data from these storages; instead, the host can only query the checksum of them. Please note that before adding a certificate or key to RAM or NVRAM, it should not be existed in the corresponding storage. If it already exists, the host should delete it first, and then add it to the corresponding storage device.

AT+QSECWRITE Upload a Certificate or Key	
Test Command **AT+QSECWRITE=?**	Response **+QSECWRITE: <filename>,<filesize>**[,list of supported **<timeout>**s] **OK**
Read Command **AT+QSECWRITE?**	Response **OK**
Write Command **AT+QSECWRITE=<filename>,<filezsize>[,<timeout>]**	Response If the format is correct **Connect** After the module switches to data mode, the certificate or key data can be input. When the size of the input data reaches <filesize> (unit: byte) or the module receives “+++” sequence from UART, the module will return to command mode and reply the following codes: **+QSECWRITE: <uploadsize>,<checksum>** **OK** If there is any error **+CME ERROR: <err>**
Reference	

Parameter

<filename>	The name of the certificate or key to be uploaded. For detailed format of the files, please refer to ***Table 3***.
<filesize>	The size of the certificate or key to be uploaded. Unit: byte If the file is uploaded to the RAM, the maximum size is 32768 bytes. If the file is uploaded to NVRAM, the maximum size is 2017 bytes. The minimum size is 1 byte.
<timeout>	The time in seconds to wait for data input via UART port. Unit: second. Range: 3-200. The default value is 100.
<uploadsize>	The size of the actually uploaded data. Unit: byte
<checksum>	The checksum of the uploaded data.

2.5. AT+QSECREAD Query the Checksum of a Certificate or Key

This command is used to query the checksum of a certificate or key. If the checksum is not the same as the original one owned by the server or client, certain mistakes will occur.

AT+QSECREAD Query the Checksum of a Certificate or Key	
Test Command **AT+QSECREAD=?**	Response **+QSECREAD: <filename>** **OK**
Read Command **AT+QSECREAD?**	Response **OK**
Write Command **AT+QSECREAD=<filename>**	Response **+QSECREAD: <good>,<checksum>** OK If there is any error **+CME ERROR: <err>**
Reference	

Parameter

<filename>	The name of the certificate or key which has been uploaded. For detailed format of the files, please refer to ***Table 3***.
<good>	Indicate whether the certificate or key is correct or not. When uploading the certificate or key by **AT+QSECWRITE**, the checksum of certificate or key will be stored at the same time. After executing **AT+QSECREAD**, it will calculate the checksum of the certificate or key again, and then compare the checksum with the one stored by

	AT+QSECWRITE. If they are the same, the certificate or key is correct, otherwise the certificate or key is wrong. 0 The certificate or key is wrong. 1 The certificate or key is correct.
<checksum>	The checksum of the file.

2.6. AT+QSECDEL Delete a Certificate or Key

This command is used to delete a certificate or key.

AT+QSECDEL Delete a Certificate or Key	
Test Command **AT+QSECDEL=?**	Response **+QSECDEL: <filename>** **OK**
Read Command **AT+QSECDEL?**	Response **OK**
Write Command **AT+QSECDEL=<filename>**	Response **OK** If there is any error **+CME ERROR: <err>**
Reference	

Parameter

<filename>	The name of the certificate or key to be deleted. For detailed format of the files, please refer to ***Table 3***.

3 Error Codes

When no command is executed and any error happens, final result codes "**+QFTPERROR:<err>**" and "**+CME ERROR: <err>**" will indicate an error related to mobile equipment or network. The details about **<err>** are described in the following table.

Table 4: Summary of Error Codes

<err>	Description of Error Codes
-1	Unknown error
-3	The FTP(S) server is busy. For example, opening FTP(S) service, controlled by another virtual UART, etc.
-4	Failed to get an IP address according to domain name.
-5	Network error. For example, it is failed to activate GPRS/CSD context, or establish the TCP connection with the FTP(S) server or send FTP(S) command to the FTP(S) server.
-6	The FTP(S) session is closed by the FTP(S) server.
-7	The data connection of the FTP(S) service is closed by the FTP(S) server.
-8	GPRS/CSD context is deactivated.
-9	Timeout
-10	The input parameter is illegal.
-11	The file is not found in a local position (UFS or SD or RAM).
-12	Failed to get the file in a local position (UFS or SD or RAM).
-13	There is no enough memory for attachment.
-421	Service not available, closing control connection. This may be a reply to any command if the service knows it must shut down.
-425	Cannot open data connection.
-426	Connection closed; transfer aborted.

-450	Request file action not taken. File unavailable (e.g., file busy).
-452	Requested action not taken. Insufficient storage space in system.
-500	Syntax error, command unrecognized. This may include errors such as command line too long.
-501	Syntax error in parameters or arguments.
-502	Command not implemented.
-530	Not logged in.
-532	Need account for storing files.
-550	Requested action not taken. File unavailable ((e.g., file not found, no access).
-551	Requested action aborted: page type unknown.
-552	Requested file action aborted. Exceeded storage allocation (for current directory or dataset).
-553	Requested action not taken. File name not allowed.
4008	No space for uploading files.
4009	Timeout
4010	The file is not found.
4011	The file is too large.
4012	The file already exists.
4013	Invalid parameter

4 Examples

4.1. Examples of Uploading Certificate and Key

4.1.1. Uploading Certificate and Key to RAM

This is an example about how to upload CA certificate, client certificate and client private key to RAM. If customers do not need server and client authentication, please skip this step.

```
//Upload CA certificate, client certificate and client private key to RAM.
AT+QSECWRITE="RAM:ca_cert.pem",1614,100         //Upload the CA certificate to RAM.
CONNECT

+QSECWRITE:  1614,4039                          //The ca_cert.pem data has been uploaded
                                                  and the size is 1614 bytes.
OK

AT+QSECWRITE="RAM:client_cert.pem",1419,100     //Upload the client certificate to RAM.
CONNECT

+QSECWRITE: 1419,618                            //The client_cert.pem data has been uploaded
                                                  and the size is 1419 bytes.
OK

AT+QSECWRITE="RAM:client_key.pem",1679,100      //Upload the client private key to RAM.
CONNECT

+QSECWRITE: 1679,83a7                           //The client_key.pem data has been uploaded
                                                  and the size is 1679 bytes.
OK
```

4.1.2. Uploading Certificate and Key to NVRAM

This is an example about how to upload CA certificate, client certificate and client private key to NVRAM. If customers do not need server and client authentication, please skip this step.

```
//Upload CA certificate, client certificate and client private key to NVRAM.
AT+QSECWRITE="NVRAM:CA0",1614,100       //Upload the CA certificate to NVRAM.
CONNECT

+QSECWRITE: 1614,4039                   //The CA0 data has been uploaded and the size is 1614
                                          bytes.
OK

AT+QSECWRITE="NVRAM:CC0",1419,100       //Upload the client certificate to NVRAM.
CONNECT

+QSECWRITE: 1419,618                    //The CC0 data has been uploaded and the size is 1419
                                          bytes.
OK

AT+QSECWRITE="NVRAM:CK0",1679,100       //Upload the client private key to NVRAM.
CONNECT

+QSECWRITE: 1679,83a7                   //The CK0 data has been uploaded and the size is 1679
                                          bytes.
OK
```

4.2. Examples of FTPS Service

4.2.1. Open and Close FTPS Service

```
//Step 1: Configure and activate the PDP context.
AT+QIFGCNT=0                              //Set the foreground context as context 0.
OK

AT+QICSGP=1,"CMNET"                       //Set the bearer type as GPRS and the APN as
                                            “CMNET”, and there is no username and password
                                            for the APN.
OK

AT+QIREGAPP                               //Register on TCP/IP stack.
OK
```

```
AT+QIACT                                      //Activate GPRS PDP context.
OK

AT+QILOCIP                                    //Query the local IP address.
10.1.83.188

AT+QSECWRITE="NVRAM:CA0",1273,100             //Upload the CA certificate to NVRAM.
CONNECT

+QSECWRITE: 1273,1a38
OK

AT+QSSLCFG="cacert",0,"NVRAM:CA0"             //Configure the path of root certificate.
OK

AT+QSSLCFG="ignorertctime",1                  //Ignore the RTC time.
OK

AT+QSSLCFG="seclevel",0,1                     //Set the SSL verify level as 1 which means only
                                                server authentication.
OK

AT+QSSLCFG="sslversion",0,4                   //Configure SSL version.
OK

AT+QSSLCFG="ctxindex",0                       //Query the context setting.
+QSSLCFG
0,4,1,"0X0005,0X0004,0X0035,0X002F,0X003D,0X000A","NVRAM:CA0","",""

OK

AT+IFC=2,2                                    //Set TE-TA Local Data Flow Control to RTS flow
                                                control and CTS flow control.
OK

AT+QSSLCFG="ftps",1                          //Enable FTPS function.
OK

AT+QSSLCFG="ftpsctxi",0                      //Configure SSL context index as 0.
OK

AT+QFTPUSER="admin"                          //Set the user name as "admin".
OK
```

```
AT+QFTPPASS="123456"                        //Set the password as "123456".
OK

AT+QFTPOPEN="220.180.239.212",990
OK

+QFTPOPEN: 0                              //Successfully opened FTP service.

AT+QFTPSTAT                               //Query the status of FTPS Service.
+QFTPSTAT: OPENED

OK

AT+QFTPCLOSE                              //Close the connection with FTPS.
OK

+QFTPCLOSE: 0                             //Successfully closed the connection.

AT+QIDEACT                                //Deactivate the context.
DEACT OK
```

4.2.2. Upload a File to FTPS Server

```
AT+QFTPPATH="/"                          //Set the path to upload file as "/".
OK

+QFTPPATH: 0                             //Successfully set the path.

AT+QFTPPUT="test.txt",10220,200          //Upload the file "test.txt" via UART and then the file will
                                           be stored in the FTPS server. The expected file size is
                                           10220 bytes. If the read size of the file "test.txt" is less
                                           than 10220 bytes, it will upload file with actual size. The
                                           maximum time to read file data is 200 seconds.
OK

CONNECT                                 //Opened data mode and uploaded data.
<Input data>                            //Input data via UART.
+QFTPPUT: 10220                         //Successfully uploaded the file "test.txt" to the FTPS
                                          server. The size of uploaded data is 10220 bytes.
```

4.2.3. Download a File from FTPS Server

```
AT+QFTPPATH="/"                     //Set the path to download file as "/".
OK

+QFTPPATH: 0                        //Successfully set the path.

AT+QFTPGET="test.txt"               //Download the file "test.txt" from the FTPS server.
OK

CONNECT                             //Successfully opened data connection to download file.
<Output data>                       //Output data via UART.
+QFTPGET: 10220                     //Successfully downloaded the file "test.txt" from the FTPS
                                      server. And the size of the data successfully downloaded is
                                      10220 bytes.

OK                                  //Successfully finished the download operation.
```

4.2.4. Switch to FTP Service

```
AT+QSSLCFG="ftps",0                 //Set <ctxindex> to 0 and switch to FTP service.
OK

AT+QFTPUSER="test"                  //Set the user name as "test".
OK

AT+QFTPPASS="test"                  //Set the password as "test".
OK

AT+QFTPOPEN="ftp2.quectel.com",21
OK

+QFTPOPEN: 0                        //Successfully opened the FTP service.
```

5 Error Handling

5.1. PDP Activation Error Handling

If it is failed to activate PDP context by **AT+QIACT** command, please check the following configurations:

1. Query whether the PS domain is attached or not by **AT+CGATT?** command. If not, execute **AT+CGATT=1** command to attach PS domain.
2. Query the CGREG status by **AT+CGREG?** command and make sure the PS domain has been registered.
3. Query the PDP context parameters by **AT+QIREGAPP** command and make sure the APN of specified PDP context has been set.
4. Make sure the specified PDP context ID is neither used by PPP nor activated by **AT+CGACT** command.
5. The module only supports two PDP contexts activated simultaneously, so customers must make sure the amount of activated PDP context is less than or equal to two.

If all configurations above are correct, but activating the PDP by **AT+QIACT** command still fails, please reboot the module to resolve this issue. After rebooting the module, please check the configurations above for at least three times and each time at an interval of 10 minutes to avoid frequent rebooting the module.

5.2. FTPS Error Handling

If it is failed to connect with FTPS server, please check the connection mode and the SSL encryption authentication mode of the server.

1. Query the connection mode by **AT+QFTPCFG=<type>[,<value>]** command. For explicit connection mode, please make sure **<type>** and **<value>** are set as 7 and 1 respectively. For implicit connection mode, the **<value>** is set as 0 by default.
2. Query the SSL context setting by **AT+QSSLCFG=”ctxindex”,<ctxindex>** command.

6 Appendix A References

Table 5: Related Documents

SN	Document Name	Remark
[1]	Quectel_M95_AT_Commands_Manual	M95 AT Commands Manual
[2]	Quectel_GSM_FTP_AT_Commands_Manual	GSM FTP AT Commands Manual
[3]	GSM 07.07	Digital Cellular Telecommunications (Phase 2+); AT Command Set for GSM Mobile Equipment (ME)
[4]	GSM 07.10	Terminal Equipment to Mobile Station (TE-MS) Multiplexer Protocol

Table 6: Terms and Abbreviations

Abbreviation	Description
APN	Access Point Name
CA	Certificate Authority
FTPS	File Transfer Protocol Secure
NVRAM	Non Volatile Random Access Memory
PDP	Packet Data Protocol
PPP	Point-to-Point Protocol
PS	Packet Switching
RAM	Random Access Memory
SSL	Security Socket Layer
TCP/IP	Transmission Control Protocol/Internet Protocol

TLS	Transport Layer Security