SIM7000 Series_SSL _Application Note

LPWA Module

SIMCom Wireless Solutions Limited
Building B, SIM Technology Building, No.633, Jinzhong Road
Changning District, Shanghai P.R. China
Tel: 86-21-31575100
support@simcom.com
www.simcom.com

Document Title:	SIM7000 Series_SSL_Application Note
Version:	1.01
Date:	2020.07.28
Status:	Released

GENERAL NOTES

SIMCOM OFFERS THIS INFORMATION AS A SERVICE TO ITS CUSTOMERS, TO SUPPORT APPLICATION AND ENGINEERING EFFORTS THAT USE THE PRODUCTS DESIGNED BY SIMCOM. THE INFORMATION PROVIDED IS BASED UPON REQUIREMENTS SPECIFICALLY PROVIDED TO SIMCOM BY THE CUSTOMERS. SIMCOM HAS NOT UNDERTAKEN ANY INDEPENDENT SEARCH FOR ADDITIONAL RELEVANT INFORMATION, INCLUDING ANY INFORMATION THAT MAY BE IN THE CUSTOMER'S POSSESSION. FURTHERMORE, SYSTEM VALIDATION OF THIS PRODUCT DESIGNED BY SIMCOM WITHIN A LARGER ELECTRONIC SYSTEM REMAINS THE RESPONSIBILITY OF THE CUSTOMER OR THE CUSTOMER'S SYSTEM INTEGRATOR. ALL SPECIFICATIONS SUPPLIED HEREIN ARE SUBJECT TO CHANGE.

COPYRIGHT

SIMCom Wireless Solutions Limited
Building B, SIM Technology Building, No.633 Jinzhong Road, Changning District, Shanghai P.R. China
Tel: +86 21 31575100
Email: simcom@simcom.com

For more information, please visit:
https://www.simcom.com/download/list-863-en.html

For technical support, or to report documentation errors, please visit:
https://www.simcom.com/ask/ or email to: support@simcom.com

About Document

Version History

Version	Date	Owner	What is new
V1.00	2019.01.24	Wenjie.lai	First Release.
V1.01	2020.07.28	Wenjie.Lai	All

Scope

This document applies to the following products

Name	Type	Size(mm)	Comments
SIM7000E/C/A/G	Cat-M1(/NB1/EGPRS)	24*24	
SIM7000E-N SIM7000C-N	NB1	24*24	

Contents

1 Introduction

1.1 Purpose of the document

Based on module AT command manual, this document will introduce SSL application process.

Developers could understand and develop application quickly and efficiently based on this document.

1.2 Related documents

[1] SIM7000 Series_AT Command Manual_V1.06

1.3 Conventions and abbreviations

In this document, the GSM engines are referred to as following term:
ME (Mobile Equipment);
MS (Mobile Station);
TA (Terminal Adapter);
DCE (Data Communication Equipment) or facsimile DCE (FAX modem, FAX board);

In application, controlling device controls the GSM engine by sending AT Command via its serial interface. The controlling device at the other end of the serial line is referred to as following term:
TE (Terminal Equipment);
DTE (Data Terminal Equipment) or plainly "the application" which is running on an embedded system;

2 SSL Introduction

SSL (Secure Sockets Layer), a security protocol. It was put forward by Netscape in the first version of Web browser. The aim is to provide security and data integrity for network communications. SSL encrypts the network connections at the transport layer.

SSL uses public key technology to ensure the confidentiality and reliability of communication between two applications and to ensure that communication between client and server applications is not eaves dropped by attackers. It can be supported at both ends of the server and client, and has become an industrial standard for secure communication over the Internet. Current Web browsers generally combine HTTP and SSL to achieve secure communication. This Agreement and its successor are TLS (Transport Layer Security, TLS).

TLS uses key algorithm to provide endpoint authentication and communication security on the Internet, It is based on the public key infrastructure. In typical implementations, however, only the network server is authenticated reliably, while the client is not necessarily. This is because the public key infrastructure is generally commercial, and electronic signature certificates usually need to be paid for. The protocol is designed to enable master-slave architecture application communication itself to prevent tapping, tampering, and message forgery.

SIM7000 series modules currently support TLS1.0, TLS1.1, TLS1.2, DTLS1.0, DTLS1.2.

3 AT Commands that support SSL's TCP/UDP

The module provides AT commands that can be used by device terminals as follows:

AT Command	Description
AT+CACID	Set TCP/UDP Identifier
AT+CASSLCFG	Set SSL certificate and timeout parameters
AT+CAOPEN	Open a TCP/UDP connection
AT+CASEND	Send data via an established connection
AT+CARECV	Receive data via an established connection
AT+CACLOSE	Close a TCP/UDP connection
AT+CSSLCFG	Configure SSL parameters of a context identifier

For more detail introduction, please refer to SIM7000 Series_AT Command Manual.

4 Bearer Configuration

Usually module will register PS service automatically.

Usually module will register PS service automatically.

4.1 PDN Auto-activation

//Example of PDN Auto-activation.	
AT+CPIN?	//Check SIM card status
+CPIN: READY	
OK	
AT+CGDCONT=1,"IP",""	//Configure APN for registration when needed
OK	
AT+CSQ	//Check RF signal
+CSQ: 27,99	
OK	
AT+CGATT?	//Check PS service.
+CGATT: 1	//1 indicates PS has attached.
OK	
AT+COPS?	//Query Network information, operator and network mode 9, NB-IOT network
+COPS: 0,0,"CHN-CT",9	
OK	
AT+CGNAPN	//Query the APN delivered by the network after the CAT-M or NB-IOT network is successfully registered.
+CGNAPN: 1,"ctnb"	//"cmnbiot" is APN delivered by the CAT-M or NB-IOT network. APN is empty under the GSM network.
OK	
AT+CNCFG=1,"ctnb","cdma","1234"	//Before activation please use AT+CNCFG to set APN\user name\password if needed.
OK	
AT+CNACT=1	//Activate network

```
OK

+APP PDP: ACTIVE
AT+CNACT?                                   //Get local IP
+CNACT: 0,1,"10.94.36.44"

OK
```

4.2 APN Manual configuration

If not attached automatically, could configure correct APN setting.

```
//Example of APN Manual configuration.

AT+CFUN=0                                   //Disable RF
+CPIN: NOT READY

OK
AT+CGDCONT=1,"IP","ctnb"                    //Set the APN manually
OK
AT+CFUN=1                                   //Enable RF
OK

+CPIN: READY
AT+CGATT?                                   //Check PS service.
+CGATT: 1                                   //1 indicates PS has attached.

OK
AT+CGNAPN                                   //Query the APN delivered by the network after the
                                            CAT-M or NB-IOT network is successfully
                                            registered.
+CGNAPN: 1,"ctnb"                           //"ctnb" is APN delivered by the CAT-M or NB-IOT

OK                                          network. APN is empty under the GSM network.
AT+CNCFG=1,"ctnb","cdma","1234"             //Before activation please use AT+CNCFG to set
                                            APN\user name\password if needed.
OK
AT+CNACT=1                                  //Activate network
OK

+APP PDP: ACTIVE
```

AT+CNACT?	//Get local IP
+CNACT: 0,1,"10.94.36.44"	
OK	

5 SSL Examples

5.1 Build an ordinary TCP/UDP connection

```
//Example of Build an ordinary TCP/UDP connection

AT+CNACT=1,"cmnet"                          //Open data connection, the parameter "cmnet" is APN. This parameter needs to set different APN values according to different cards.

OK

+APP PDP: ACTIVE
AT+CNACT?                                   //Get local IP
+CNACT: 1,"10.181.182.177"

OK
AT+CACID=0                                  //Device identifier
OK
AT+CASSLCFG=0,ssl,0                         //Whether to use the SSL, If TCP/UDP connection, the parameter is 0.

OK
AT+CASSLCFG=0,protocol,0                    //Set the protocol type. Set to 0 is TCP. If it is UDP, it should be set to 1.

OK
AT+CAOPEN=0,"116.247.119.165",5171          //Create a TCP connection
+CAOPEN: 0,0                                //Return to URC the first parameter is the identifier, the second parameter is the result of the connection, and the 0 indicates success.

OK
AT+CASEND=0,5                               //Request to send 5 bytes of data
>                                           //Input data

OK                                          //Data sent successfully
+CASEND: 0,0,5
+CADATAIND: 0                               //Connection with an identifier of 0 has data.
AT+CARECV=0,100                             //Request to get 100 byte data sent by the server
                                            //Output received data
+CARECV: 0,GFDSGFDGFDSGHFDSHFDS
```

```
OK
AT+CACLOSE=0                    //Close the connection with an identifier of 0.
OK
AT+CNACT=0                      //Disconnect data connection
OK

+APP PDP: DEACTIVE
```

5.2 Build a SSL connection

When SSL establishes communication, it is necessary to verify the identity of both sides of the communication, which is divided into one-way authentication and two-way authentication.
One way authentication is the client to verify the certificate of the server. The server sends the server certificate to the client. The client verifies that the root certificate that issued the server certificate is trustworthy, and if so continues the communication process.
After the two-way authentication client verifies the server certificate, the client needs to send its own certificate to the server and let the server verify its client certificate. The validation process is the same, all need to confirm whether the root certificate of the certificate can be trusted.

5.2.1 Build a one-way authentication SSL connection

Because of modules can only serve as clients. When you need to establish a one-way authentication connection, you need to import the root certificate of the server. If no certificate is imported, the module will default that all the servers can be trusted.

```
//Example of Build a one-way authentication SSL connection

AT+CNACT=1,"cmnet"              //Open data connection, the parameter "cmnet" is
                                APN. This parameter needs to set different APN
                                values according to different cards.
OK

+APP PDP: ACTIVE
AT+CNACT?                       //Get local IP
+CNACT: 1,"10.181.182.177"

OK
AT+CACID=0                      //Device identifier
OK
AT+CSSLCFG="sslversion",0,1     //Set the protocol type of SSL with an identifier of
```

	0.1 indicate TLS1.0
OK	
AT+CASSLCFG=0,ssl,1	//Whether to use SSL, 1 means to turn on the SSL function.
OK	
AT+CASSLCFG=0,crindex,0	//Set protocol type //Identifier for AT+CSSLCFG corresponding SSL configuration
OK	
AT+CASSLCFG=0,"cacert","root.pem"	//Set root certificate. The root certificate must be a certificate that has been converted through AT+CSSLCFG. This item can be omitted. If omitted, all server certificates are trusted by default.
OK	
AT+CAOPEN=0,"116.247.119.165",5171	//Create a SSL connection.
+CAOPEN: 0,0	//Connection success
OK **+ CADATAIND: 0**	//Connection with an identifier of 0 has data. When a connection is successfully established or data is successfully sent, the module actively reads the data once, and if the server data is received, the URC is reported. //If no data is received, the URC will not be reported.
AT+CARECV=0,100	//Read 100 byte data
+CARECV: 0, **220 Serv-U FTP Server v15.0 ready...**	//Output data
OK	
AT+CACLOSE=0	//Close the connection with an identifier of 0.
OK	
AT+CNACT=0	//Disconnect data connection
OK	
+APP PDP: DEACTIVE	

5.2.2 Build a two-way authentication SSL connection

To establish a two-way authentication SSL connection, you need to set up a client certificate. The client certificate needs to be transformed through AT+CSSLCFG first.

The certificate format that the module can support is.PEM, .DER and .P7B.

//Example of Build a two-way authentication SSL connection

```
AT+CNACT=1,"cmnet"                          //Open data connection, the parameter "cmnet" is APN. This parameter needs to set different APN values according to different cards.

OK

+APP PDP: ACTIVE
AT+CNACT?                                   //Get local IP
+CNACT: 1,"10.181.182.177"

OK
AT+CACID=0                                  //Device identifier
OK
AT+CSSLCFG="sslversion",0,1                 //Set the protocol type of SSL with an identifier of 0.
                                            //1 indicate TLS1.0
OK
AT+CASSLCFG=0,ssl,1                         //Whether to use SSL, 1 means to turn on the SSL function.

OK
AT+CASSLCFG=0,crindex,0                     //Set protocol type
                                            //Identifier for AT+CSSLCFG corresponding SSL configuration

OK
AT+CASSLCFG=0,"clientcert","client.pem"     //Set up client certificates.
                                            //The root certificate must be converted to a certificate that can be directly used by AT+CSSLCFG.
OK
AT+CAOPEN=0,"116.247.119.165",5171          //Create a SSL connection
+CAOPEN: 0,0                                //Connection success

OK
AT+CASEND=0,5                               //Request to send 5 bytes of data
>                                           //Input data

OK                                          //Data sent successfully
+CASEND: 0,0,5
AT+CACLOSE=0                                //Close the connection with an identifier of 0.
OK
AT+CNACT=0                                  //Disconnect data connection
OK
```

```
+APP PDP: DEACTIVE
```

5.2.3 Transform SSL certificates

```
//Example of Transform SSL certificates

AT+CSSLCFG="convert",2,"root.pem"        //Configuring the type of certificate to be converted, and 2 is a root certificate.
                                         //Configure the name of the certificate to be converted, and the name after the conversion is consistent with the existing certificate name.
OK
AT+CSSLCFG="convert",1,"client.pem","client.key"   //Configure the type of certificate to be converted, and 1 is client certificate.
                                         //Configure the certificate name that needs to be converted, and the client certificate needs to enter the certificate file and the private key file.
                                         //The name after conversion is identical to the name of the certificate, that is "client.pem".
+CNACT: 1,"10.181.182.177"

OK
```