SIM800 Series_SSL _Application Note

GPRS 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:	SIM800 Series_SSL_Application Note
Version:	1.05
Date:	2020.10.16
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	2013.10.18	Hanjun.Liu	First Release
V1.01	2013.06.30	Ping.Zhang/ Hanjun.Liu	Add scope Chapter2.4, Add description of TCP over SSL Chapter2.5, Add description of SSL certificate Chapter2.6, Add SSL option Chapter3.8,3.9,3.10 Add examples
V1.02	2016.11.17	Wenjie.Lai	Scope
V1.03	2019.12.10	Chengliang.Wang Xiaohui.Xu	Chapter 2.7,Add AT+SSLSETROOT Chapter 2.8,Add AT+SSLDEROOT Chapter 2.9,AT+SSLDECLI Chapter 3.11,Add examples Chapter 3.12, Add examples Chapter 3.13, Add examples
V1.04	2020.6.15	Liuyang.Zhang /Wenjie.Lai	All
V1.05	2020.10.16	Jia.tao/Wenjie.Lai	Chapter 3.5,Add client certificate file type Document style

Scope

This document presents the AT command of SSL operation and application examples. This document can apply to SIM800 series modules with SSL function.

Contents

1 Introduction

1.1 Purpose of the document

Based on module AT command manual, this document will introduce SSL operations, including HTTP, EMAIL and FTP function.

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

1.2 Related Documents

[1] SIM800 Series_AT Command Manual

1.3 Conventions and abbreviations

Abbreviation	Description
URC	Unsolicited request code
TE	Terminal Equipment
TA	Terminal Adapter
DTE	Data Terminal Equipment or plainly "the application" which is running on an embedded system
DCE	Data Communication Equipment or facsimile DCE(FAX modem, FAX board)
ME	Mobile Equipment
MS	Mobile Station
SSL	Secure Socket Layer
TLS	Transport Layer Security

2 SSL Function

2.1 SSL Description

Secure socket layer (SSL), a security protocol, is first put forward by Netscape at the same time as they lunch the first version of Web Browser, the purpose is to provide security and data integrity for network communication. SSL encrypts network connection at the transport layer.

SSL uses public key technology to ensure the confidentiality and reliability of communication between applications, so that the communication between client and server application will not be intercepted by the aggressor. It can be supported on both the server and the client ends, has become the industry standard secure communication on the internet. The current Web browsers generally combine the HTTP and SSL, enabling secure communication. This Agreement and its successor is TLS (Transport Layer Security).

TLS using the key algorithm provided endpoint authentication and secure communication on the Internet, which is based on public key infrastructure (PKI). However, in the example of a typical implementation, only the network service provider is reliable authentication, the client is not necessarily. This is because the public key infrastructure common in commercial operation, electronic signature certificate is usually required to pay for. Protocol is designed in a way to make the master-slave architecture application communication itself prevent eavesdropping, tampering, and message forgery.

SIM800 series support SSL2.0, SSL3.0, TLS1.0 and TLS1.2.

2.2 HTTPS Description

HTTPS is the HTTP channel which targets secure, in simple terms is safe version of HTTP. Added layer of SSL below HTTP, security of HTTPS is based on SSL, so the details please see the SSL encryption.

It is a URI scheme (abstract identifier system), syntax similar to http: System. For secure HTTP data transmission. HTTPS:URL shows that it uses HTTP, but HTTPS exists a default port different with HTTP and has an encryption / authentication layer (between HTTP and TCP). This system was originally developed by Netscape for providing authenticated and encrypted communication method, and now it is widely used in security-sensitive communication on the World Wide Web, such as transaction payment.

2.3 FTPS Description

FTPS is a multi-transmission protocol, equivalent to the encrypted version of the FTP. It is an enhanced FTP protocol which uses standard FTP protocol and commands in the Secure Sockets Layer. It add SSL security features for FTP protocol and data channels. FTPS is also known as "FTP-SSL" and "FTP-over-SSL". SSL is a protocol which encrypts and decrypts data in secure connection between client and an SSL-enabled server.

2.4 EMAIL Encrypted Transmission Description

To receive Email, SIM800 series support SSL encrypted POP3 protocol which is called POP3S. It will use special port, default port: 995. To send Email, SIM800 series use HTTPS communication, default port: 465. SIM800 series also supports the use of ordinary port, through the STARTTLS (SMTP) and STLS (POP3) to enable encryption transmission.

3 AT command

SIM800 series modules provide encrypted link AT command is as follows:

Command	Description
AT+EMAILSSL	Set EMAIL to use SSL function
AT+HTTPSSL	Set HTTP to use SSL function
AT+FTPSSL	Set FTP to use SSL function
AT+CIPSSL	Set TCP to use SSL function
AT+SSLSETCERT	Import SSL client certificate file
AT+SSLOPT	SSL option
AT+SSLSETROOT	Import SSL root certificate file
AT+SSLDEROOT	Delete SSL root certificate file
AT+SSLDECLI	Delete SSL Client Certificate File

3.1 AT+EMAILSSL Set Email to Use SSL Function

AT+EMAILSSL Set EMAIL to Use SSL Function	
Test Command **AT+EMAILSSL=?**	Response **+EMAILSSL:** (list of supported <**n**>s) **OK**
Read Command **AT+EMAILSSL?**	Response **+EMAILSSL: <n>** **OK**
Write Command **AT+EMAILSSL=<n>**	Response **OK**
Parameter Saving Mode	NO_SAVE
Max Response Time	-
Reference	An error code will return if the SSL channel setup failure or communication errors happened when sending mail: **+SMTPSEND: <code>** An error code when sign POP3 server : **+POP3IN: <code>**

	<code> 71 SSL failed to establish channels 72 SSL alert message with a level of fatal result in the immediate termination of the connection.

Defined Values

<n>	0 Not use encrypted transmission 1 Begin encrypt transmission with encryption port 2 Begin encrypt transmission with normal port

3.2 AT+HTTPSSL Set HTTP to Use SSL Function

AT+HTTPSSL Set HTTP to Use SSL Function	
Test Command **AT+HTTPSSL=?**	Response **+HTTPSSL:** (range of supported <**n**>s) **OK**
Read Command **AT+HTTPSSL?**	Response **+HTTPSSL: <n>** **OK**
Write Command **AT+HTTPSSL=<n>**	Response **OK**
Parameter Saving Mode	NO_SAVE
Max Response Time	-
Reference	An error code will return if HTTPACTION command fail： **+HTTPACTION: <code>** **<code>** 605 SSL failed to establish channels 606 SSL alert message with a level of fatal result in the immediate termination of the connection

Defined Values

<n>	0 Disable SSL function 1 Enable SSL function

3.3 AT+FTPSSL Set FTP to Use SSL Function

AT+FTPSSL Set FTP to Use SSL Function	
Test Command **AT+FTPSSL=?**	Response **+FTPSSL:** (range of supported <**n**>s) **OK**
Read Command **AT+FTPSSL?**	Response **+FTPSSL:** <**n**> **OK**
Set Command **AT+FTPSSL=<n>**	Response **OK**
Parameter Saving Mode	NO_SAVE
Max Response Time	-
Reference	An error code will return if FTP operation fail, case in FTPGET: **+FTPGET: <code>** **<code>** 80 SSL failed to establish channels 81 SSL alert message with a level of fatal result in the immediate termination of the connection 82 FTP AUTH error 83 FTP PBSZ error 84 FTP PROT error

Defined Values

<n>	0 Disable SSL function 1 Use FTPS with Implicit mode 2 Use FTPS with Explicit mode

3.4 AT+CIPSSL Set TCP to Use SSL Function

AT+CIPSSL Set TCP to Use SSL Function	
Test Command **AT+CIPSSL=?**	Response **+CIPSSL:** (range of supported <**n**>s) **OK**
Read Command **AT+CIPSSL?**	Response **+CIPSSL: <n>**

	OK
Write Command **AT+CIPSSL=<n>**	Response **OK**
Parameter Saving Mode	NO_SAVE
Max Response Time	-
Reference	After set AT+CIPSSL=1, module will automatic begin SSL certificate after TCP connected Currently, we just support SSL Client function.

Defined Values

<n>	0 Disable SSL function 1 Enable SSL function

3.5 AT+SSLSETCERT Import SSL Client Client Certificate File with Private Key

AT+SSLSETCERT	**Import SSL Client Client Certificate File with Private Key**
Test Command **AT+SSLSETCERT=?**	Response **+SSLSETCERT:** max length of field <**file**>,max length of field <**password**> **OK**
Write Command **AT+SSLSETCERT=<file>[,<password>]**	Response **OK** If import succeed **+SSLSETCERT: 0** If import failed **+SSLSETCERT: 1**
Parameter Saving Mode	NO_SAVE
Max Response Time	-
Reference	Just one file can be imported. If import more than once, module will keep last imported file. Support ".crt", ".cer" or "p12" certificate file.

Defined Values

<file>	File to be imported. Alphanumeric ASCII text string up to 100

	characters.
<password>	Password required to parse the certificate file. Alphanumeric ASCII text string up to 32 characters.

3.6 AT+SSLOPT SSL Option

AT+SSLOPT SSL Option	
Test Command **AT+SSLOPT=?**	Response **+SSLOPT:** (range of supported <**opt**>s),(range of supported <**enable**>s) **OK**
Read Command **AT+SSLOPT?**	Response +**SSLOPT: 0,<enable>** **+SSLOPT: 1,<enable>** **OK**
Write Command **AT+SSLOPT=<opt>,<enable>**	Response **OK**
Parameter Saving Mode	NO_SAVE
Max Response Time	-
Reference	If need server authentication, please set AT+SSLOPT=0,0 If do not need server authentication, please set AT+SSLOPT=0,1 If need client authentication, please set AT+SSLOPT=1,1 If do not need client authentication, please set AT+SSLOPT=1,0

Defined Values

<opt>	0 Ignore invalid certificate 1 Client authentication
<enable>	0 Close 1 Open

3.7 AT+SSLSETROOT Import SSL Root Certificate File

AT+SSLSETROOT Import SSL Root Certificate File	
Write Command **AT+SSLSETROOT=<filename>,<filesize>**	Response **OK** or **Certificate already exists!** **OK** or **ERROR**
Parameter Saving Mode	NO_SAVE
Max Response Time	-
Reference	The files to be imported must be binary encoded

Defined Values

<filename>	File to be imported
<filesize>	File to be imported of size

3.8 AT+SSLDEROOT Delete SSL Root Certificate File

AT+SSLDEROOT Delete SSL Root Certificate File	
Test Command **AT+SSLDEROOT?**	Response list of supported <**filename**>s **OK**
Write Command **AT+SSLDEROOT=<filename>**	Response If delete succeed **+SSLDEROOT: 0** **OK** If delete failed **+SSLDEROOT: 13** File does not exist or **+SSLDEROOT: 42** Not enough permissions **OK**
Parameter Saving Mode	NO_SAVE
Max Response Time	-
Reference	

Defined Values

<filename>	The file name you want to delete

3.9 AT+SSLDECLI Delete SSL Client Certificate File

AT+SSLDECLI Delete SSL Client Certificate File	
Test Command **AT+SSLDECLI?**	Response **OK** or list of the imported client certificate file **<filename>** **OK**
Write Command **AT+SSLDECLI=<filename>**	Response If delete succeed **+SSLDECLI: 0** **OK**
Parameter Saving Mode	NO_SAVE
Max Response Time	-
Reference	

Defined Values

<filename>	The file name you want to delete

4 SSL Examples

The following table provides some using method of the SSL function.

4.1 EMAIL Send Encrypted Mail with Normal Port

//Email send with normal port	
AT+SAPBR=3,1,"APN","CMNET"	//Configure bearer profile 1
OK	
AT+SAPBR=1,1	//To open a GPRS context
OK	
AT+EMAILCID=1	//Set EMAIL Use bear profile 1
OK	
AT+EMAILTO=30	//Set EMAIL timeout
OK	
AT+EMAILSSL=2	//Set EMAIL begin encrypt transmission with normal port
OK	
AT+SMTPSRV="SMTP.GMAIL.COM"	//Set SMTP server address, port is omitted, means use the default ports: 25
OK	
AT+SMTPAUTH=1,"account","password"	//Set user name and password
OK	
AT+SMTPFROM="account@GMAIL.COM","account"	//Set sender address and name
OK	
AT+SMTPSUB="Test"	//Set the subject
OK	
AT+SMTPRCPT=0,0,"john@sim.com","john"	//Set the recipient (To:)
OK	
AT+SMTPBODY=19	//Set the body
DOWNLOAD	
This is a new Email	
OK	
AT+SMTPSEND	Send the Email
OK	
+SMTPSEND: 1	

4.2 EMAIL Send Encrypted Mail with Encryption Port

```
//Email send with encryption port
AT+SAPBR=3,1,"APN","CMNET"                    //Configure bearer profile 1
OK
AT+SAPBR=1,1                                  //To open a GPRS context
OK
AT+EMAILCID=1                                 //Set EMAIL Use bear profile 1
OK
AT+EMAILTO=30                                 //Set EMAIL timeout
OK
AT+EMAILSSL=1                                 //Set EMAIL begin encrypt transmission with
OK                                            encryption port
AT+SMTPSRV="SMTP.GMAIL.COM"                   //Set SMTP server address, port is omitted, means
OK                                            use the default ports: 465
AT+SMTPAUTH=1,"account","password"            //Set user name and password
OK
AT+SMTPFROM="account@GMAIL.COM","acc          //Set sender address and name
ount"
OK
AT+SMTPSUB="Test"                             //Set the subject
OK
AT+SMTPRCPT=0,0,"john@sim.com","john"         //Set the recipient (To:)
OK
AT+SMTPBODY=19                                //Set the body
DOWNLOAD
This is a new Email

OK
AT+SMTPSEND                                   //Send the Email
OK

+SMTPSEND: 1
```

4.3 EMAIL Receive Encrypted Mail with Normal Port

```
//Email receive with normal port
```

```
AT+SAPBR=3,1,"APN","CMNET"                    //Configure bearer profile 1
OK
AT+SAPBR=1,1                                  //To open a GPRS context.
OK
AT+EMAILCID=1                                 //Set EMAIL Use bear profile 1
OK
AT+EMAILTO=30                                 //Set EMAIL timeout
OK
AT+EMAILSSL=2                                 //Set EMAIL begin encrypt transmission with
OK                                            normal port
AT+POP3SRV="mail.sim.com","john","123456      //Set POP3 server and account, port is omitted,
"                                             means use the default ports 110
OK
AT+POP3IN                                     //Log in POP3 server
OK

+POP3IN: 1
AT+POP3NUM                                    //Get Email number and total size
OK

+POP3NUM: 1,2,11124
AT+POP3LIST=1                                 //Get the specific Email's size
OK

+POP3LIST: 1,1,5556
AT+POP3CMD=4,1                                //Retrieve the specific Email
OK

+POP3CMD: 1
AT+POP3READ=1460                              //Get the Email content
+POP3READ: 1,1460
…

OK

AT+POP3READ=1460
+POP3READ: 1,1460
…

OK
AT+POP3READ=1460                              //The Email's content is read completely
+POP3READ: 2,1183
…

OK
```

```
AT+POP3OUT                                  //Log out POP3 SERVER
OK

+POP3OUT: 1
```

4.4 EMAIL Receive Encrypted Mail with Encryption Port

```
//Email receive with encryption port
AT+SAPBR=3,1,"APN","CMNET"                  //Configure bearer profile 1
OK
AT+SAPBR=1,1                                //To open a GPRS context.
OK
AT+EMAILCID=1                               //Set EMAIL Use bear profile 1
OK
AT+EMAILTO=30                               //Set EMAIL timeout
OK
AT+EMAILSSL=1                               //Set EMAIL begin encrypt transmission with
OK                                          encryption port
AT+POP3SRV="mail.sim.com","john","123456    //Set POP3 server and account, port is omitted,
"                                           means use the default ports 995
OK
AT+POP3IN                                   //Log in POP3 server
OK

+POP3IN: 1
AT+POP3NUM                                  //Get Email number and total size
OK

+POP3NUM: 1,2,11124
AT+POP3LIST=1                               //Get the specific Email's size
OK

+POP3LIST: 1,1,5556
AT+POP3CMD=4,1                              //Retrieve the specific Email
OK

+POP3CMD: 1
AT+POP3READ=1460                            //Get the Email content
+POP3READ: 1,1460
…
```

```
OK

AT+POP3READ=1460
+POP3READ: 1,1460
…

OK
AT+POP3READ=1460                    //The Email's content is read completely
+POP3READ: 2,1183
…

OK
AT+POP3OUT                          //Log out POP3 SERVER
OK

+POP3OUT: 1
```

4.5 HTTPS Get Method with HTTPS

```
// Use HTTPS download data
AT+HTTPINIT                         //Init HTTP service
OK
AT+HTTPPARA="CID",1                 //Set parameters for HTTP session
OK
AT+HTTPPARA="URL","www.gmail.com"
OK
AT+HTTPPARA="REDIR",1
OK
AT+HTTPSSL=1                        //Enable HTTPS function
OK
AT+HTTPACTION=0                     //GET session start
OK
+HTTPACTION: 0,200,84200            //GET successfully
AT+HTTPREAD                         //Read the data of HTTP server
+HTTPREAD: 84200
….
OK
AT+HTTPTERM                         //Terminate HTTP service
OK
```

4.6 FTP Get Method with Implicit FTPS

//Use Implicit FTPS mode download	
AT+FTPCID=1	//Set parameters for FTP session.
OK	
AT+FTPSERV="116.228.221.52"	
OK	
AT+FTPUN="sim.cs1"	
OK	
AT+FTPPW="****"**	
OK	
AT+FTPGETNAME="1K.txt"	
OK	
AT+FTPGETPATH="/"	
OK	
AT+FTPSSL=1	//Open Implicit FTPS mode
OK	
AT+FTPGET=1	//Open the FTP get session.
OK	
+FTPGET: 1,1	//Data are available.
AT+FTPGET=2,1024	//Request to read 1024 bytes, but
+FTPGET: 2,50	//Only 50 bytes are now available.
01234567890123456789012345678901234567890123456789	
OK	
AT+FTPGET=2,1024	//Request to read 1024 bytes again.
+FTPGET: 2,0	//No byte is now available, but it is not the end of session.
OK	
+FTPGET: 1,1	//If the module receives data but user do not input "AT+FTPGET=2,<reqlength>" to read data, "+FTPGET: 1,1" will be shown again in a certain time.
AT+FTPGET=2,1024	//Request to read 1024 bytes.
+FTPGET: 2,1024	//1024 bytes are now available.
01234567890123456789012345678901234567890123456789…..1234	
OK	
+FTPGET:1,0	//Data transfer finished. The connection to the FTP server is closed.

4.7 Set FTP Get Method with Explicit FTPS

//Use Explicit FTPS mode download	
AT+FTPCID=1	//Set parameters for FTP session.
OK	
AT+FTPSERV="116.228.221.52"	
OK	
AT+FTPUN="sim.cs1"	
OK	
AT+FTPPW="****"**	
OK	
AT+FTPGETNAME="1K.txt"	
OK	
AT+FTPGETPATH="/"	
OK	
AT+FTPSSL=2	//Open Explicit FTPS mode
OK	
AT+FTPGET=1	//Open the FTP get session.
OK	
+FTPGET: 1,1	//Data are available.
AT+FTPGET=2,1024	//Request to read 1024 bytes, but
+FTPGET: 2,50	//Only 50 bytes are now available.
01234567890123456789012345678901234567890123456789	
OK	
AT+FTPGET=2,1024	//Request to read 1024 bytes again.
+FTPGET: 2,0	//No byte is now available, but it is not the end of session.
OK	
+FTPGET: 1,1	//If the module receives data but user do not input "AT+FTPGET=2,<reqlength>" to read data, "+FTPGET: 1,1" will be shown again in a certain time.
AT+FTPGET=2,1024	//Request to read 1024 bytes.
+FTPGET: 2,1024	//1024 bytes are now available.
01234567890123456789012345678901234567890123456789…..1234	
OK	
+FTPGET:1,0	//Data transfer finished. The connection to the FTP server is closed.

4.8 Establish a TCP Client Connection over SSL

//Establish a TCP Client Connection over SSL	
AT+CGATT?	//GPRS Service's status
+CGATT: 1	
OK	
AT+CSTT="CMNET"	//Start task and set APN.
OK	//The default APN is "CMNET", with no username or password. Check with local GSM provider to get the APN.
AT+CIICR	//Bring up wireless connection (GPRS or CSD)
OK	
AT+CIFSR	//Get local IP address
10.78.245.128	
AT+CIPSSL=1	//Enable SSL function
OK	
AT+CIPSTART="TCP","116.228.221.51","8500"	//Start up the connection
OK	//The TCP connection has been established successfully. SSL certificate finished.
CONNECT OK	
AT+CIPSEND	//Send data to remote server, CTRL+Z (0x1a) to send. User should write data only after the promoting mark ">", and then use CTRL+Z to send. User can use command "AT+CIPSPRT" to set whether echo promote ">" after issuing "AT+CIPSEND".
> hello TCP serve	
SEND OK	//Remote server receives data. For TCP, "SEND OK" means data has been sent out and received successfully by the remote server, due to the TCP connection-oriented protocol;
hello SIM800	//Received data from remote server
CLOSED	//Remote server closed the connection

4.9 Establish a TCP Client Connection over SSL in Multi Connection

AT+CIPSSL=1 must be set first if customer want to start a TCP connection over SSL. Any TCP connection established before AT+CIPSSL=1 will not try SSL certificate.

//Establish a TCP Client Connection over SSL in Multi Connection	
AT+CGATT?	//GPRS Service's status
+CGATT: 1	

```
OK
AT+CIPMUX=1                                  //Enable multi connection
OK
AT+CSTT="CMNET"                              //Start task and set APN.
OK
AT+CIICR                                     //Bring up wireless connection
OK                                             (GPRS r CSD)
AT+CIFSR                                     //Get local IP address
10.78.245.128
AT+CIPSTART=0,"TCP","116.228.221.51","850    //Establish a TCP connection, connection number
0"                                           0
OK

0,CONNECT OK
AT+CIPSSL=1                                  //Enable SSL function. Connection 0 will not start
OK                                           SSL certificate
AT+CIPSTART=1,"TCP","116.228.221.51","960    //Establish a TCP connection, connection number
0"                                           1. SSL certificate finished.
OK

1,CONNECT OK
AT+CIPSEND=0                                 //Send data to connection 0
> TCP test

0,SEND OK
AT+CIPSEND=1                                 //Send data to connection 1
> TCP Over SSL test

1,SEND OK
+RECEIVE,0,17:
SIM800 TCP test                              //Received data from connection 0, data length 17
+RECEIVE,1,26:                               //Received data from connection 1, data length 26
SIM800 TCP Over SSL test
0,CLOSED                                     //Connection 0 is closed by remote server
AT+CIPSTATUS                                 //Query the current connection status
OK

STATE: IP PROCESSING

C:
0,0,"TCP","116.228.221.51","8500","CLOSED "
C:
1,0,"TCP","116.228.221.51","9600","CONNECT
ED "
C: 2,,"","","","INITIAL"
```

```
C: 3,,"","","","INITIAL"
C: 4,,"","","","INITIAL"
C: 5,,"","","","INITIAL"
```

4.10 Import a SSL Client Certificate File with Private Key

```
//Import a SSL Client certificate file with private key
AT+FSCREATE=C:\USER\HENRY_SSL.CRT          //Create certificate file on FS.
OK
AT+FSWRITE=C:\USER\HENRY_SSL.CRT,0,1196,10  //Write file to FS.
>
OK
AT+SSLSETCERT="C:\USER\HENRY_SSL.CRT","********"  //Import certificate file
OK                                         //Import succeed

+SSLSETCERT: 0
```

4.11 Import a SSL Root Certificate File

```
//Import a SSL root certificate file
AT+FSCREATE=C:\USER\HENRY_SSL.CRT          //Create certificate file on FS.
OK
AT+FSWRITE=C:\USER\HENRY_SSL.CRT,0,1196,10  //Write file to FS.
>
OK
AT+SSLSETROOT="C:\USER\HENRY_SSL.CRT",1196  //Import certificate file
OK                                         //Import succeed
```

4.12 Delete a SSL Root Certificate File

```
//Delete a SSL root certificate file
AT+SSLDEROOT?                    //Certificate file list
1001
1002
...
1013

OK
AT+SSLDEROOT=1013                //Delete Certificate file 1013
+SSLDEROOT: 0                    //Delete succeed

OK
```

4.13 Delete a SSL Client Certificate File

```
//Delete a SSL client certificate file
AT+SSLDECLI?                     //Client Certificate file name
1014
1014

OK
AT+SSLDECLI=1014                 //Delete Certificate file 1014
+SSLDECLI: 0                     //Delete succeed

OK
```