SIM7000 Series_ HTTP(S)_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_HTTP(S)_Application Note
Version:	1.03
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	2018.9.28	Xiaobao.qu	First Release
V1.01	2019.1.23	Xiaobao.qu	Modified Charpter 3.3 and max length of some parameters
V1.02	2019.12.25	Jiangting.ding	Added HTTP and HTTPS Samples
V.103	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 HTTP(S) 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
[2] RFC 2616

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 HTTP(S) Function

HTTP (HyperText Transfer Protocol) is an application layer protocol. When you browse a web page, the browser and the web server will send and receive data on the Internet through the HTTP protocol. HTTP is a stateless protocol based on request and response patterns. That is what we usually call Request/Response.

2.1 Characteristic

- Support client/server mode;
  - Simple and fast

When a client requests a service from a server, it only needs to pass the request method and path. Because the HTTP protocol is simple, the program size of the HTTP server is small, and the communication speed is fast.

- Flexible

HTTP allows the transfer of any type of data object. The type being transferred is marked by Content-Type;

- No connection

No connection means limiting the processing of only one request per link. After the server processes the client's request and receives the customer's response, the server disconnects the link. This way, the transmission time can be saved.

- Stateless

The HTTP protocol is a stateless protocol. Stateless means that the protocol has no memory for transaction processing. A lack of state means that if subsequent processing requires the previous information, it must be retransmitted, which may result in an increase in the amount of data transferred per connection. On the other hand, it responds faster when the server does not need previous information.

2.2 Request Method

According to the HTTP standard, HTTP requests can use a variety of request methods.
HTTP 1.0 defines three request methods: the GET, POST, and HEAD methods.
HTTP1.1 adds six new request methods: OPTIONS, PUT, PATCH, DELETE, TRACE, and CONNECT methods.

No	Method	Description
1	GET	Make a request to a specific resource.
2	HEAD	Ask the server for a response that is consistent with the GET request, except that the response body will not be returned. This method can obtain the meta information contained in the response message header without having to transmit the entire response content.
3	POST	Submit data to a specified resource for processing requests (such as submitting a form or uploading a file). The data is included in the request body. POST requests may result in the creation of new resources and/or modifications to existing resources.
4	PUT	Uploads its latest content to a specified resource location.
5	DELETE	Requests the server to delete the resource identified by the Request-URI.
6	CONNECT	H The HTTP/1.1 protocol is reserved for proxy servers that can connect connections to pipes.
7	OPTIONS	Returns the HTTP request method supported by the server for a particular resource. You can also test the functionality of the server by sending a '*' request to the web server.
8	TRACE	Echoes requests received by the server, primarily for testing or diagnostics.
9	PATCH	It is a supplement to the PUT method for local updating of known resources.

The SIM7000 series supports several methods: GET, POST, PUT, PATCH and HEAD.

3 AT Commands for HTTP(S)

AT Command	Description
AT+SHSSL	Select SSL Configure
AT+SHCONF	Set HTTP(S) Parameter
AT+SHCONN	HTTP(S) Connection
AT+SHBOD	Set Body
AT+SHBODEXT	Set Extension Body
AT+SHAHEAD	Add Head
AT+SHPARA	Set HTTP(S) Para
AT+SHCPARA	Clear HTTP(S) Para
AT+SHCHEAD	Clear Head
AT+SHSTATE	Query HTTP(S) Connection Status
AT+SHREQ	Set Request Type
AT+SHREAD	Read Response Value
AT+SHDISC	Disconnect HTTP(S)
AT+HTTPTOFS	Download file to ap file system
AT+HTTPTOFSRL	State of download file to ap file system

For detail information, please refer to “SIM7000 Series_AT Command Manual”.

4 Bearer Configuration

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
+COPS: 0,0,"CHN-CT",9             mode 9, NB-IOT network

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
                                  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 HTTP(S) Examples

5.1 HTTP Function

5.1.1 HTTP GET

```
//Example of HTTP GET.

AT+SHCONF="URL","http://httpbin.org"          //Set up server URL
OK
AT+SHCONF="BODYLEN",1024                      //Set HTTP body length, for range of max body
OK                                            length
AT+SHCONF="HEADERLEN",350                     //Set HTTP head length, for range of max head
OK                                            length
AT+SHCONN                                     //HTTP build
OK
AT+SHSTATE?                                   //Get HTTP status
+SHSTATE: 1                                   //"+SHSTATE: 1": connected
                                              //"+SHSTATE: 0": disconnected
OK
AT+SHCHEAD                                    //Clear HTTP header, because of http header is
OK                                            appended
AT+SHAHEAD="User-Agent","curl/7.47.0"         //Add header content
OK                                            //For detail, please refer to  document "rfc2616"
AT+SHAHEAD="Cache-control","no-cache"         //Add header content
OK                                            //For detail, please refer to  document "rfc2616"
AT+SHAHEAD="Connection","keep-alive"          //Add header content
OK                                            //For detail, please refer to  document "rfc2616"
AT+SHAHEAD="Accept","*/*"                     //Add header content
OK                                            //For detail, please refer to  document "rfc2616"
AT+SHREQ="/get?user=jack&password=123",1      //Set request type is GET.
OK                                            //Get data size is 387.

+SHREQ: "GET",200,387
AT+SHREAD=0,387                               //Read data length is 387
OK                                            //The data content is follow "+SHREAD: 387"
```

```
+SHREAD: 387
{
  "args": {
    "password": "123",
    "user": "jack"
  },
  "headers": {
    "Accept": "*/*",
    "Cache-Control": "no-cache",
    "Content-Length": "0",
    "Host": "httpbin.org",
    "User-Agent": "curl/7.47.0",
    "X-Amzn-Trace-Id":
"Root=1-5ed706c8-99b97372ae6f043f805cf243"
  },
  "origin": "117.132.195.245",
  "url":
"http://httpbin.org/get?user=jack&password=12
3"
}

AT+SHDISC                                           //Disconnect HTTP connect
OK
```

5.1.2 HTTP POST

```
//Example 1 of HTTP POST.

AT+SHCONF="URL","http://httpbin.org"              //Set up server URL
OK
AT+SHCONF="BODYLEN",1024                          //Set HTTP body length
OK
AT+SHCONF="HEADERLEN",350                         //Set HTTP head length
OK
AT+SHCONN                                         //HTTP build
OK
AT+SHSTATE?                                       //Get HTTP status
+SHSTATE: 1

OK
AT+SHCHEAD                                        //Clear HTTP header
OK
AT+SHAHEAD="Content-Type","application/x-        //Add header content
```

```
www-form-urlencoded"
OK
AT+SHAHEAD="Cache-control","no-cache"         //Add header content
OK
AT+SHAHEAD="Connection","keep-alive"          //Add header content
OK
AT+SHAHEAD="Accept","*/*"                     //Add header content
OK
AT+SHCPARA                                    //Clear body content parameter
OK
AT+SHPARA="product","apple"                   //Add body content parameter
OK
AT+SHPARA="price","1"                         //Add body content parameter
OK
AT+SHREQ="/post",3                            //Set request type is POST
OK                                            //Get data size is 452.

+SHREQ: "POST",200,452
AT+SHREAD=0,452                               //Read data length is 452
OK                                            //The data content is follow “+SHREAD: 452”

+SHREAD: 452
{
  "args": {},
  "data": "",
  "files": {},
  "form": {
    "price": "1",
    "product": "apple"
  },
  "headers": {
    "Accept": "*/*",
    "Cache-Control": "no-cache",
    "Content-Length": "21",
    "Content-Type":
"application/x-www-form-urlencoded",
    "Host": "httpbin.org",
    "X-Amzn-Trace-Id":
"Root=1-5ed633df-058feb6412204392e95333b2
"
  },
  "json": null,
  "origin": "218.204.252.187",
  "url": "http://httpbin.org/post"
}
```

```
AT+SHDISC                                   //Disconnect HTTP connect
OK
```

```
//Example 2 of HTTP POST.

AT+SHCONF="URL","http://httpbin.org"         //Set up server URL
OK
AT+SHCONF="BODYLEN",1024                     //Set HTTP body length
OK
AT+SHCONF="HEADERLEN",350                    //Set HTTP head length
OK
AT+SHCONN                                    //HTTP build
OK
AT+SHSTATE?                                  //Get HTTP status
+SHSTATE: 1

OK
AT+SHCHEAD                                   //Clear HTTP header
OK
AT+SHAHEAD="Content-Type","application/x-    //Add header content
www-form-urlencoded"
OK
AT+SHAHEAD="Cache-control","no-cache"        //Add header content
OK
AT+SHAHEAD="Connection","keep-alive"         //Add header content
OK
AT+SHAHEAD="Accept","*/*"                    //Add header content
OK
AT+SHBOD=" {\"title\":\"Hello http server\"}",29   //Set body content

OK
AT+SHREQ="/post",3                           //Set request type is POST
OK                                           //Get data size is 457.

+SHREQ: "POST",200,457
AT+SHREAD=0,457                              //Read data length is 457
OK                                           //The data content is follow "+SHREAD: 457"

+SHREAD: 457
{
  "args": {},
  "data": "{\"title\":\"Hello http server\"}",
  "files": {},
  "form": {},
  "headers": {
    "Accept": "*/*",
    "Cache-Control": "no-cache",
    "Content-Length": "29",
    "Content-Type": "application/json",
    "Host": "httpbin.org",
    "X-Amzn-Trace-Id":
"Root=1-5ed63fa7-3dda07707b3f2ea63e092a3a"
  },
  "json": {
    "title": "Hello http server"
  },
  "origin": "218.204.252.187",
  "url": "http://httpbin.org/post"
}

AT+SHDISC                                        //Disconnect HTTP connect
OK
```

5.2 HTTPS Function

5.1.1 HTTPS download and convert SSL certificate

```
//Example of HTTPS download and convert SSL Certificate.

AT+CFSINIT
                                              //Init FS AT command
OK
AT+CFSWFILE=3,"httpbin_root_ca.cer",0,1492    //After download, sent certificate file through the
,1000                                         serial port.1492 is certificate size.
DOWNLOAD
                                              //Send CA file success
OK
AT+CFSTERM
                                              //Free data buffer
OK
AT+CSSLCFG="convert",2,"httpbin_root_ca.c     //Conversion CA certificate format.
er"                                           //2 means CA type.
OK                                            //httpbin_root_ca.cer is CA certificate name.
```

5.2.1 HTTPS GET

```
//Example of HTTPS GET.

AT+CSSLCFG="sslversion",1,3                     //Configure SSL/TLS version
OK
                                                //Set HTTP SSL Configure
AT+SHSSL=1,"httpbin_root_ca.cer"                //if you would skip certificate verify, use
OK                                              AT+SHSSL=1,"" instead
AT+SHCONF="URL","https://httpbin.org"           //Set connect server parameter
OK
AT+SHCONF="BODYLEN",1024                        //Set max body length
OK
AT+SHCONF="HEADERLEN",350                       //Set max header length
OK
AT+SHCONN                                       //Connect HTTPS server
OK
AT+SHSTATE?                                     //Get HTTP status
+SHSTATE: 1

OK
AT+SHCHEAD                                      //Clear HTTP header content
OK
AT+SHAHEAD="User-Agent","curl/7.47.0"           //Add header content
OK
AT+SHAHEAD="Cache-control","no-cache"           //Add header content
OK
AT+SHAHEAD="Connection","keep-alive"            //Add header content
OK
AT+SHAHEAD="Accept","*/*"                       //Add header content
OK
AT+SHREQ="/get?user=jack&password=123",1        //Set request type is GET.
OK                                              //Get data size is 388

+SHREQ: "GET",200,388
AT+SHREAD=0,388                                 //Read data length is 388
OK                                              //The data content is follow "+SHREAD: 388"

+SHREAD: 388
{
  "args": {
    "password": "123",
    "user": "jack"
  },
  "headers": {
    "Accept": "*/*",
    "Cache-Control": "no-cache",
    "Content-Length": "0",
    "Host": "httpbin.org",
    "User-Agent": "curl/7.47.0",
    "X-Amzn-Trace-Id":
"Root=1-5ed706c8-99b97372ae6f043f805cf243"
  },
  "origin": "117.132.195.245",
  "url":
"https://httpbin.org/get?user=jack&password=1
23"
}

AT+SHDISC                                         //Disconnect HTTP connect
OK
```

5.2.2 HTTPS POST

```
//Example 1 of HTTPS POST.

AT+CSSLCFG="sslversion",1,3                      //Configure SSL/TLS version
OK
                                                 //Set HTTP SSL Configure
AT+SHSSL=1,"baidu_root_ca.cer"                   //if you would skip certificate verify, use
OK                                               AT+SHSSL=1,"" instead
AT+SHCONF="URL","https://httpbin.org"            //Set connect server parameter
OK
AT+SHCONF="BODYLEN",1024                         //Set max body length
OK
AT+SHCONF="HEADERLEN",350                        //Set max header length
OK
AT+SHCONN                                        //Connect HTTPS server
OK
AT+SHSTATE?                                      //Get HTTP status
+SHSTATE: 1

OK
AT+SHCHEAD                                       //Clear HTTP header
OK
AT+SHAHEAD="Content-Type","application/x-        //Add header content
www-form-urlencoded"
OK
```

```
AT+SHAHEAD="Cache-control","no-cache"       //Add header content
OK
AT+SHAHEAD="Connection","keep-alive"        //Add header content
OK
AT+SHAHEAD="Accept","*/*"                   //Add header content
OK
AT+SHCPARA                                  //Clear body content para
OK
AT+SHPARA="product","apple"                 //Add body content para
OK
AT+SHPARA="price","1"                       //Add body content para
OK
AT+SHREQ="/post",3                          //Set request type is POST
OK                                          //Get data size is 453.

+SHREQ: "POST",200,453
AT+SHREAD=0,453                             //Read data length is 453
OK                                          //The data content is follow "+SHREAD: 453"

+SHREAD: 453
{
  "args": {},
  "data": "",
  "files": {},
  "form": {
    "price": "1",
    "product": "apple"
  },
  "headers": {
    "Accept": "*/*",
    "Cache-Control": "no-cache",
    "Content-Length": "21",
    "Content-Type":
"application/x-www-form-urlencoded",
    "Host": "httpbin.org",
    "X-Amzn-Trace-Id":
"Root=1-5ed633df-058feb6412204392e95333b2
"
  },
  "json": null,
  "origin": "218.204.252.187",
  "url": "https://httpbin.org/post"
}

AT+SHDISC                                   //Disconnect HTTP connect
OK
```

```
//Example 2 of HTTPS POST.

AT+CSSLCFG="sslversion",1,3                 //Configure SSL/TLS version
OK
                                            //Set HTTP SSL Configure
AT+SHSSL=1,"baidu_root_ca.cer"              //if you would skip certificate verify, use
OK                                          AT+SHSSL=1,"" instead
AT+SHCONF="URL","https://httpbin.org"       //Set connect server parameter
OK
AT+SHCONF="BODYLEN",1024                    //Set max body length
OK
AT+SHCONF="HEADERLEN",350                   //Set max header length
OK
AT+SHCONN                                   //Connect HTTPS server
OK
AT+SHSTATE?                                 //Get HTTP status
+SHSTATE: 1

OK
AT+SHCHEAD                                  //Clear HTTP header
OK
AT+SHAHEAD="Content-Type","application/js   //Add header content
on"
OK
AT+SHAHEAD="Cache-control","no-cache"       //Add header content
OK
AT+SHAHEAD="Connection","keep-alive"        //Add header content
OK
AT+SHAHEAD="Accept","*/*"                   //Add header content
OK
AT+SHBOD=" {\"title\":\"Hello http server\"}",29   //Set body content
OK
AT+SHREQ="/post",3                          //Set request type is POST
OK                                          //Get data size is 458.

+SHREQ: "POST",200,458
AT+SHREAD=0,458                             //Read data length is 458
OK                                          //The data content is follow “+SHREAD: 458”

+SHREAD: 458
{
  "args": {},
  "data": "{\"title\":\"Hello http server\"}",
  "files": {},
  "form": {},
  "headers": {
    "Accept": "*/*",
    "Cache-Control": "no-cache",
    "Content-Length": "29",
    "Content-Type": "application/json",
    "Host": "httpbin.org",
    "X-Amzn-Trace-Id":
"Root=1-5ed63fa7-3dda07707b3f2ea63e092a3a
"
  },
  "json": {
    "title": "Hello http server"
  },
  "origin": "218.204.252.187",
  "url": "https://httpbin.org/post"
}

AT+SHDISC                                  //Disconnect HTTP connect
OK
```