Perfect Wireless Experience
完美无线体验

FIBOCOM NL668 AT Commands User Manual_SSL

Version: V1.1.4

Date: 2019-10-30

Applicability Type

No.	Type	Note
1	NL668-CN-00/01/02/03/04/10	NA
2	NL668-EAU-00	NA
3	NL668-EU-00/01	NA
4	NL668-AM-00/01	NA
5	NL668-JP-00/01	NA
6	NL668-LA-00	NA
7	NL661-EU-00	NA

Copyright

Attention

The document is subject to update from time to time owing to the product version upgrade or other reasons. Unless otherwise specified, the document only serves as the user guide. All the statements, information and suggestions contained in the document do not constitute any explicit or implicit guarantee.

Trademark

The trademark is registered and owned by Fibocom Wireless Inc.

Versions

Version	Author	Assessor	Approver	Update Date	Description
V1.0.0				2015-01-26	Initial version
V1.0.1				2015-01-31	Added +GTSSLERR AT command
V1.0.2				2015-03-09	Increase applicable modes
V1.0.3				2017-04-10	Added +GTSSLVER command using to set theversion of handshake protocol; Updated memo notes for certificate related files;
V1.0.4				2017-11-16	Change to new template
V1.0.5				2018-04-21	Add NL668 serial
V1.0.6				2018-11-14	Add NL668-AM serial and NL668-EU
V1.0.7				2018-12-26	Add AT example and modify SSL error code
V1.0.8				2019-04-17	List all the specific application types
V1.1.1				2019-05-30	Add Note for GTSSLVER command
V1.1.2				2019-06-06	Add +GTSSLCIPHER command and Note
V1.1.3	Shi Jin		Pu Long	2019-10-14	Add NL668-CN-10
V1.1.4	Zhang Miao	Wang Bin	Pu Long	2019-10-30	Modify +GTSSLCIPHER command description

Contents

1 SSL

1.1 +GTSSLFILE,Load certificates or keys

1.1.1 Description

This command is used to load the CA certificate of SSL, KEY, or the local certificate of trust..

1.1.2 Syntax

Command	Response/Action	Note
+GTSSLFILE=<file_type>,<file_len>	OK or: ERROR	Set command sets the type and length of the loading certificate, CERTFILE and KEYFILE indicate which type of CA certificate or key are loaded (generally for the situation of two-way authentication, when the client sends the certificate or key to the server, i.e., the server needs to validate the client's legitimacy). TRUSTFILE indicates that the trust certificate is loaded into the machine, where the purpose of the certificate is to verify the validity of the server (both one-way authentication and bidirectional authentication may need to be validated (specifically by +GTSSLMODE chose)), Loading TRUSTFILE file support up to 40.
+GTSSLFILE?	+GTSSLFILE:<file_type>,<file_num> OK e.g. +GTSSLFILE: CERTFILE,0 +GTSSLFILE:KEYFILE,0	Read command queries whether the type of certificate has been loaded, where CERTFILE and KEYFILE can be at most 1,TRUSTFILE can be more than one. For example, there is no certificate file type of CAFILE or KEYFILE; but have a TRUSTFILE certificate type ;

Command	Response/Action	Note
	+GTSSLFILE:TRUSTFILE,1	
+GTSSLFILE=?	+GTSSLFILE:(“CERTFILE,KEYFILE ,TRUSTFILE”),(list of supproted < file_len>s) OK	Test command can query the type and length of the loading certificate.

1.1.3Attributes

Pin Restricted	Persistent	Sync Mode	Effect Immediately	Time of duration
No	No	Yes	Yes	< 1s

1.1.4 Defined Values

< file_type>：Only use one type of file among " CERTFILE", "KEYFILE" and "TRUSTFILE"

< file_len>：length of certificate (which is the length of the file after encoding by Base64 format),range is 4-8192bytes

< file_num>：Represents the number of loaded certificates

Note:

If module power down, all certificates are lost. The CERTFILE and KEYFILE at module can only load one of them; TRUSTFILE, trust certificate, can load up to 40. At present, Command only can support to add (loading) and query certificates, but not support to delete and modify certificates. And most importantly, any type of file loaded into module must be encoded through base64 format. When module goes to ODM mode and appear ">", if module do not receive any data for more than 12 seconds, it will automatically exit ODM mode and return ERROR.

1.2 +GTSSLMODE,Set whether to verify the certificate of the server

1.2.1 Description

This command can set whether the client (module)verifies certificate downloaded from server or not, 1 indicates verify and 0 indicates not. If verify set, then there must be at least one trust certificate in the local trust client list, (i.e., we can get at least one file in the TRUSTFILE field of AT+GTSSLFILE?)

1.2.2 Syntax

Command	Response/Action
+GTSSLMODE=<checkmode>	OK or: ERROR
+GTSSLMODE?	+GTSSLMODE:<checkmode> OK
+GTSSLMODE=?	+GTSSLMODE: (list of supported < checkmode>s) OK

1.2.3Attributes

Pin Restricted	Persistent	Sync Mode	Effect Immediately	Time of duration
No	No	Yes	Yes	< 1s

1.2.4 Defined Values

< checkmode>：0 indicates no verify (default setting), and 1 indicates need to verify

1.3 +GTSSLERR,Get the SSL error code

1.3.1 Description

The function of this command is to query the error code generated by the last SSL error connection.

1.3.2 Syntax

Command	Response/Action	Note
+GTSSLERR	OK or: +GTSSLERR: <err_code> OK	If SSL is no error in the connection, return OK, otherwise the error code returned from the last connection is returned .
+GTSSLERR?	OK or: +GTSSLERR: <err_code> OK	If there are no error occurred in the SSL connection, then modulewill returnOK, otherwise will return the error code occurred at the last connection.

1.3.3 Attributes

Pin Restricted	Persistent	Sync Mode	Effect Immediately	Time of duration
No	No	Yes	Yes	< 1s

1.3.4 Defined Values

< err_code>：
0: normal
10: unexpected message
20: bad record mac
21: decryption failed reserved
22: record overflow
30: decompression failure
40: handshake failure
41: no certificate reserved(only used to SSL3.0)
42: bad certificate
43: unsupported certificate
44: certificate revoked
45: certificate expired
46: certificate unknown
47: illegal parameter
48: unknown ca
49: access denied
50: decode error
51: decrypt error
60: export restriction reserved

70: protocol version
71: insufficient security
80: internal error
90: user canceled
100: no renegotiation
150: bad certificate sign
151: bad certificate issuer
152: invalid
153: host mismatch (Common Name)
154: alert for viewing the certificate
155: an unrecognized server name list

1.4 +GTSSLVER,Set and query the version of the SSL handshake protocol

1.4.1 Description

The command function is to set up or query the version of the SSL handshake protocol.

1.4.2 Syntax

Command	Response/Action
+GTSSLVER=<sslver>	OK or: ERROR
+GTSSLVER?	+GTSSLVER:< sslver > OK
+GTSSLVER=?	+GTSSLVER: (list of supported < sslver>s) OK

1.4.3Attributes

Pin Restricted	Persistent	Sync Mode	Effect Immediately	Time of duration
No	No	Yes	Yes	< 1s

1.4.4 Defined Values

< sslver>:
1 : indicates the protocol version is SSL3.0;

2 : indicates the protocol version is TLS1.0 （Default）;
3 : indicates the protocol version is TLS1.1;
4 : indicates the protocol version is TLS1.2

Note:
SSL3.0 does not support on NL668_AM_00/01, NL668_JP_00/01, NL668_CN_03/04, NL668_LA_00.

1.5 +GTSSLCIPHER, Configure the encryption algorithm when establishing the connection

1.5.1 Description

The command function is to configure the encryption algorithm supported by the current product.

1.5.2 Syntax

Command	Response/Action
+GTSSLCIPHER=<cipalgID>,<cmd>	OK or ERROR
+GTSSLCIPHER?	+GTSSLCIPHER: (list of supported <cipalgID>s) OK list cipher algorithm ID that can be used or OK no cipher algorithm ID enabled, modem will load default cipher algorithms
+GTSSLCIPHER=?	+GTSSLCIPHER: (range of valid <cipalgID>s), (range of supported <cmd>s) OK

1.5.3Attributes

Pin Restricted	Persistent	Sync Mode	Effect Immediately	Time of duration
No	No	Yes	Yes	< 1s

1.5.4 Defined Values

<cipalgID>: integer type and range 0-31 and 255, each cipher algorithm ID is related with the corresponding algorithm. the following are the correspondence between cipalgID and the related algorithm:

0: indicates all default algorithms set;
1: indicates the algorithm is TLS_DHE_RSA_WITH_3DES_EDE_CBC_SHA;
2: indicates the algorithm is TLS_DHE_RSA_WITH_AES_128_CBC_SHA;
3: indicates the algorithm is TLS_RSA_WITH_AES_128_CBC_SHA;
4: indicates the algorithm is TLS_DHE_RSA_WITH_AES_256_CBC_SHA;
5: indicates the algorithm is TLS_RSA_WITH_AES_256_CBC_SHA;
6: indicates the algorithm is TLS_RSA_WITH_3DES_EDE_CBC_SHA;
7: indicates the algorithm is TLS_RSA_WITH_AES_128_CBC_SHA256;
8: indicates the algorithm is TLS_RSA_WITH_AES_256_CBC_SHA256;
9: indicates the algorithm is TLS_DHE_RSA_WITH_AES_128_CBC_SHA256;
10: indicates the algorithm is TLS_DHE_RSA_WITH_AES_256_CBC_SHA256;
11-31: reserved;
255: indicates all surpported algorithms set;

Note:

different product have different default algorithms and different supported algorithms, all cipher algorithm ID are normalized. if all of the algorithms are not configured, the device will load default algorithms as follows,

for these products M910/NL668_CN_00/ NL668_CN_01/ NL668_CN_02/NL668_EAU/ NL668_EU_00/ NL668_EU_01,
default algorithms are:
TLS_DHE_RSA_WITH_3DES_EDE_CBC_SHA, cipher ID 1;
TLS_DHE_RSA_WITH_AES_128_CBC_SHA, cipher ID 2;
TLS_RSA_WITH_AES_128_CBC_SHA, cipher ID 3;
other non-default algorithms are:
TLS_DHE_RSA_WITH_AES_256_CBC_SHA, cipher ID 4;
TLS_RSA_WITH_AES_256_CBC_SHA, cipher ID 5;
TLS_RSA_WITH_3DES_EDE_CBC_SHA, cipher ID 6;

for these products NL668_AM/NL668_JP/ NL668_LA/NL668_CN_04,
default algorithms are:
TLS_DHE_RSA_WITH_AES_128_CBC_SHA, cipher ID 2;
TLS_RSA_WITH_AES_128_CBC_SHA, cipher ID 3;
TLS_DHE_RSA_WITH_AES_256_CBC_SHA, cipher ID 4;
other non-default algorithms are:
TLS_RSA_WITH_AES_128_CBC_SHA256, cipher ID 7;
TLS_RSA_WITH_AES_256_CBC_SHA256, cipher ID 8;
TLS_DHE_RSA_WITH_AES_128_CBC_SHA256, cipher ID 9;
TLS_DHE_RSA_WITH_AES_256_CBC_SHA256, cipher ID 10;

<cmd>: integer type, range 0, 1. cmd indicates whether to load the algorithm binded with the current cipher algorithm ID.

0 the algorithm shouldn't be loaded when establishing the connection;

1 the algorithm should be loaded when establishing the connection;

2 Example

2.1 +GTSSLFILE

//write ca file,key file,cert file

at+gtsslfile="certfile",1176

>

// input the certificate content

OK

Note: The certfile must start with "-----BEGIN CERTIFICATE-----" and end with "-----END CERTIFICATE-----". The certificate content must be in base64 encoding format

at+gtsslfile="trustfile",1241

>

// input the certificate content

OK

Note: The trust must start with "-----BEGIN CERTIFICATE-----" and end with "-----END CERTIFICATE-----". The certificate content must be in base64 encoding format

at+gtsslfile="keyfile",1675

>

// input the certificate content

OK

Note: The certfile must start with " -----BEGIN RSA PRIVATE KEY-----" and end with " -----END RSA PRIVATE KEY-----". The certificate content must be in base64 encoding format2.1 Non-authentication mode

2.2 +GTSSLMODE

//set the verification server certificate

at+gtsslmode=1

OK

//set not to verify the server certificate

at+gtsslmode=0

OK

2.3 +GTSSLVER

//set the authentication protocol : SSL3.0

AT+GTSSLVER=1

OK

//set the authentication protocol : TLS1.0

AT+GTSSLVER=2

OK

//set the authentication protocol : TLS1.1

AT+GTSSLVER=3

OK

//set the authentication protocol : TLS1.2

AT+GTSSLVER=4

OK

2.4 +GTSSLERR

//Query the error code returned when the authentication fails.

AT+GTSSLERR?

+GTSSLERR: 42

OK

2.5 +GTSSLCIPHER

//enable all supported cipalgID

AT+GTSSLCIPHER=255,1

OK

//Query the cipalgID can be used

AT+GTSSLCIPHER?

+GTSSLCIPHER: 2,3,4,5,7,8,9

OK

//disable all default cipalgID

AT+GTSSLCIPHER=0,0

OK

//enable specific cipalgID

AT+GTSSLCIPHER=3,1

OK

AT+GTSSLVER=4

AT+GTSSLCIPHER=9,1

2.6 SSL application example

Example 1: Do not verify the server certificate

AT+GTSSLVER=2

OK

AT+GTSSLFILE?

+GTSSLFILE: CERTFILE,0

+GTSSLFILE: KEYFILE,0

+GTSSLFILE: TRUSTFILE,0

OK

AT+GTSSLMODE?

+GTSSLMODE: 0 //normal connection mode, do not need to verify the certificate

OK

AT+MIPCALL=1,"cmnet"

OK

+MIPCALL: 10.79.220.142

AT+MIPOPEN=1,,"www.baidu.com",443,2

OK

+MIPOPEN: 1,1

AT+MIPCLOSE=1

OK

+MIPCLOSE: 1,1

Example 2: No no trustfile and verify server certificate

at+gtsslmode=1

OK

at+gtsslmode?

+GTSSLMODE: 1

OK

at+mipopen=1,,"www.baidu.com",443,2

OK

+MIPOPEN 1,0 // connect fail

AT+GTSSLERR? // Query reason

+GTSSLERR: 42 //The certificate validation failed.

OK

//the reason is no TRUSTFILE certificate

at+gtsslfile? // query loading certificate through the

+GTSSLFILE: CERTFILE,0

+GTSSLFILE: KEYFILE,0

+GTSSLFILE: TRUSTFILE,0

OK

Example 3: load expired certificate and verify server certificate

at+gtsslmode=1

OK

at+gtsslmode?

+GTSSLMODE: 1

OK

AT+GTSSLFILE="TRUSTFILE",850

>

// input the certificate content

OK

at+gtsslfile?

+GTSSLFILE: CERTFILE,0

+GTSSLFILE: KEYFILE,0

+GTSSLFILE: TRUSTFILE,1

OK

at+mipopen=1,,"114.255.225.39",20444,2

OK

+MIPOPEN 1,0 / / the connection failed

AT+GTSSLERR?

+GTSSLERR: 45

OK

Note: Whether the certificate expires depends on the time of the module. If the current time of the module is outside the validity period of the certificate, the module will consider it to be an expired certificate.

Example 4: Load a valid certificate and verify the server certificate

at+gtsslmode=1

OK

at+gtsslmode?

+GTSSLMODE: 1

OK

AT+GTSSLFILE="TRUSTFILE",850

>

// input the certificate content

OK

at+gtsslfile?

+GTSSLFILE: CERTFILE,0

+GTSSLFILE: KEYFILE,0

+GTSSLFILE: TRUSTFILE,1

OK

at+mipopen=1,,"114.255.225.39",20444,2

OK

+MIPOPEN: 1,1 //Connection success

AT+MIPCLOSE=1

OK

+MIPCLOSE: 1,1

Example 5: Single authentication Write KEYFILE and CERTFILE,

do not write TRUSTFILE, do not verify server certificate

AT+GTSSLMODE?

+GTSSLMODE: 0

OK

AT+GTSSLFILE?

+GTSSLFILE: CERTFILE,0

+GTSSLFILE: KEYFILE,0

+GTSSLFILE: TRUSTFILE,0

OK

AT+GTSSLFILE=" CERTFILE ",1334

>

….

OK

AT+GTSSLFILE="KEYFILE",1675

>

….

OK

AT+GTSSLFILE?

+GTSSLFILE: CERTFILE,1

+GTSSLFILE: KEYFILE,1

+GTSSLFILE: TRUSTFILE,0

OK

AT+MIPOPEN=1,,"188.93.19.231",5555,2

OK

+MIPOPEN: 1,1

+MIPRTCP:
1,0,350A576564204A616E2032312031353A35383A3034204D534B20323031350A576564204A
616E2032312031353A35383A3035204D534B20323031350A576564204A616E2032312031353A3
5383A3036204D534B20323031350A576564204A616E2032312031353A35383A3037204D534B20
323031350A576564204A616E2032312031353A35383A3038204D534B20323031350A576564204
A616E2032312031353A35383A3039204D534B20323031350A576564204A616E2032312031353A
35383A3130204D534B20323031350A576564204A616E2032312031353A35383A3131204D534B2
0323031350A576564204A616E2032312031353A35383A3132204D534B20323031350A576564204
A616E2032312031353A35383A3133204D534B20323031350A576564204A616E2032312031353A
35383A3134204D534B20323031350A576564204A616E2032312031353A35383A3135204D534B2
0323031350A576564204A616E2032312031353A35383A3136204D534B20323031350A576564204
A616E2032312031353A35383A3137204D534B20323031350A576564204A616E2032312031353A
35383A3138204D534B20323031350A576564204A616E2032312031353A35383A3139204D534B2
0323031350A576564204A616E2032312031353A35383A3230204D534B20323031350A576564204
A616E2032312031353A35383A3231204D534B20323031350A576564204A616E2032312031353A
35383A3232204D534B20323031350A576564204A616E2032312031353A35383A3233204D534B2
0323031350A576564204A616E2032312031353A35383A3234204D534B20323031350A576564204
A616E2032312031353A35383A3235204D534B20323031350A576564204A616E2032312031353A
35383A3236204D534B20323031350A576564204A616E2032312031353A35383A3237204D534B2
0323031350A576564204A616E2032312031353A35383A3238204D534B20323031350A576564204
A616E2032312031353A35383A3239204D534B20323031350A576564204A616E2032312031353A
35383A3330204D534B20323031350A576564204A616E2032312031353A35383A3331204D534B2
0323031350A576564204A616E2032312031353A35383A3332204D534B20323031350A576564204
A616E2032312031353A35383A3333204D534B20323031350A576564204A616E2032312031353A
35383A3334204D534B20323031350A576564204A616E2032312031353A35383A3335204D534B2
0323031350A576564204A616E2032312031353A35383A3336204D534B20323031350A576564204
A616E2032312031353A35383A3337204D534B20323031350A576564204A616E2032312031353A

35383A3338204D534B20323031350A576564204A616E2032312031353A35383A3339204D534B2
0323031350A576564204A616E2032312031353A35383A3430204D534B20323031350A576564204
A616E2032312031353A35383A3431204D534B20323031350A576564204A616E2032312031353A
35383A3432204D534B20323031350A576564204A616E2032312031353A35383A3433204D534B2
0323031350A576564204A616E2032312031353A35383A3434204D534B20323031350A576564204
A616E2032312031353A35383A3435204D534B20323031350A576564204A616E2032312031353A
35383A3436204D534B20323031350A576564204A616E2032312031353A35383A3437204D534B2
0323031350A576564204A616E2032312031353A35383A3438204D534B20323031350A576564204
A616E2032312031353A35383A3439204D534B20323031350A576564204A616E2032312031353A
35383A3530204D534B20323031350A576564204A616E

+MIPRTCP: 1,0,2032312031353A35383A3531204D534B20323031350A576564204A61

3 Appendix

CA certificate instance, note that this program supports only Base64 encoding format, as shown below:

-----BEGIN CERTIFICATE-----

MIIEUjCCA7ugAwIBAgIMaNRI/dS0fjCIWMzpMA0GCSqGSIb3DQEBBQUAMEExDTAL
BgNVBAYeBABDAE4xETAPBgNVBAseCABJAEMAQgBDMR0wGwYDVQQDHhQAcgBvAG8A
dABDAEEANwA4ADEAMDAeFw0wOTEyMDMxMTU3NTFaFw0zOTExMjYxMTU3NTFaMH4x
DTALBgNVBAYeBABDAE4xGzAZBgNVBAgeEgBHAHUAYQBuAGcAZABvAG4AZzENMAsG
A1UEBx4EAEcAWjERMA8GA1UECh4IAEIAYQBuAGsxETAPBgNVBAseCABJAEMAQgBD
MRswGQYDVQQDHhIAcwB1AGIAQwBBADcAOAAyADgwgZ8wDQYJKoZIhvcNAQEBBQAD
gY0AMIGJAoGBAJPVCgBKBJxImKAvTdJmnOJCN8xaO9to+mqKd0dIuyorMkBCBSsC
LNbveFTy4YzUQrwZKKbSYxOHFBpwSXMLWMzvQasU1QO6nM1pt6agDKFhyS0g07Md
eXurWZBPHjU5Kh6kNAtUgGCwCdwwy7kPqJU+hO6EhMEClzTxiTE0WIULAgMBAAGj
ggIQMIICDDAOBgNVHQ8BAQAEBAMCAIYwDwYDVR0TAQEABAUwAwEB/zCByQYDVR0f
AQEABIG+MIG7MIG4oIG1oIGyhoGvbGRhcDovLzEyMi4xMzYuNzguMTg6Mzg5L0NO
PXJvb3RDQTc4MTAsIENOPXJvb3RDQTc4MTAsIE9VPUNSTERpc3RyaWJ1dGVQb2lu
dHMsIERDPWI1Y2FkLCBEQz1pY2JjLCBEQz1jb20sIERDPWNuP2NlcnRpZmljYXRl
UmV2b2NhdGlvbkxpc3Q/YmFzZT9vYmplY3RjbGFzcz1jUkxEaXN0cmlidXRpb25Q
b2ludDAUBglghkgBhvhCAQEBAQAEBAMCAAQwIgYDVR0jAQEABBgwFoAU03ocnyxJ
tUIQ6aT51gu0K7uVgRgwgcAGCCsGAQUFBwEBAQEABIGwMIGtMIGqBggrBgEFBQcw
AoaBnWxkYXA6Ly8xMjIuMTM2Ljc4LjE4OjM4OS9DTj1yb290Q0E3ODEwLENOPXJv
b3RDQTc4MTAsT1U9Y0FDZXJ0aWZpY2F0ZXMsREM9aXVjYWQsIERDPWIjYmMsIERD
PWNvbSwgREM9Y24/Y0FDZXJ0aWZpY2F0ZT9iYXNlP29iamVjdENsYXNzPWNlcnRp

ZmljYXRpb25BdXRob3JpdHkwIAYDVR0OAQEABBYEFNzkkw+GM/atvbKrgpXzayzB
FrtiMA0GCSqGSIb3DQEBBQUAA4GBAGjfYnEvvYcoEpeHq+Uv/ZBA9lmcbCdUZ/9h
2QBw8SfR4Lv8LAB9Kp+23oOQTQeEsi5MNlQDxOGKxOUUsmt4DBCLNRevxBmWEpur
rUlM/Ar4xte+LXoItI1ZCVDSPjnvLXGopQfaUtS3IlIWYvU5XG9fpGgUX02cqAN2
d5TgyvLY

-----END CERTIFICATE-----