



How to Use NeoGate TG API

Version V1.0

Date: November, 2013

Yeastar Information Technology Co. Ltd

1. Overview

This protocol allows external programs to control NeoGate to send/receive SMS and send USSD.

2. TG Configuration

API should be enabled in NeoGate. Click SMS tab on the upper right corner, and click API Settings. Tick the box in front of "Enable API". Enter the IP address that is allowed to run SMS Client. Save and apply changes.

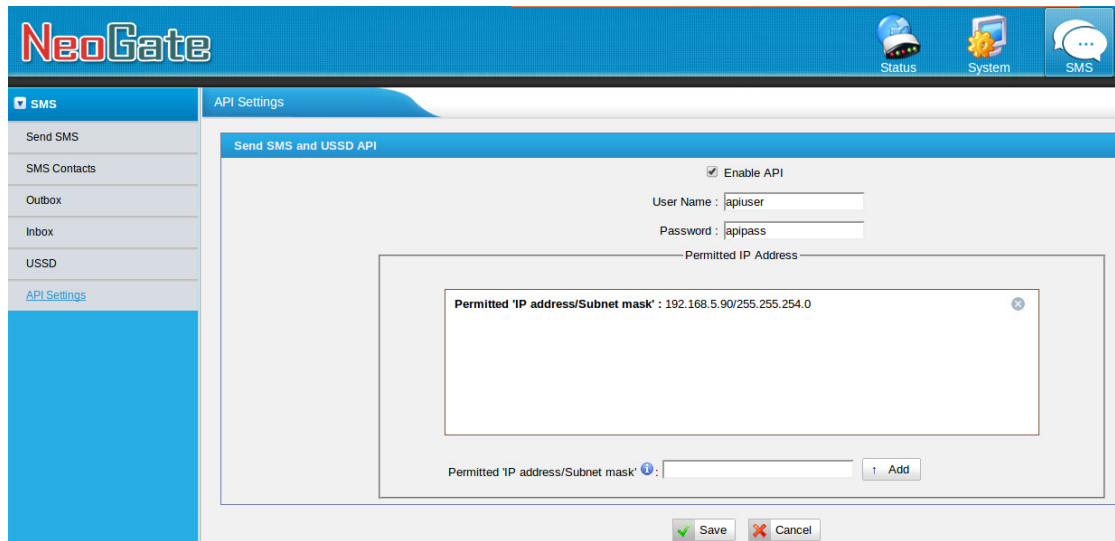
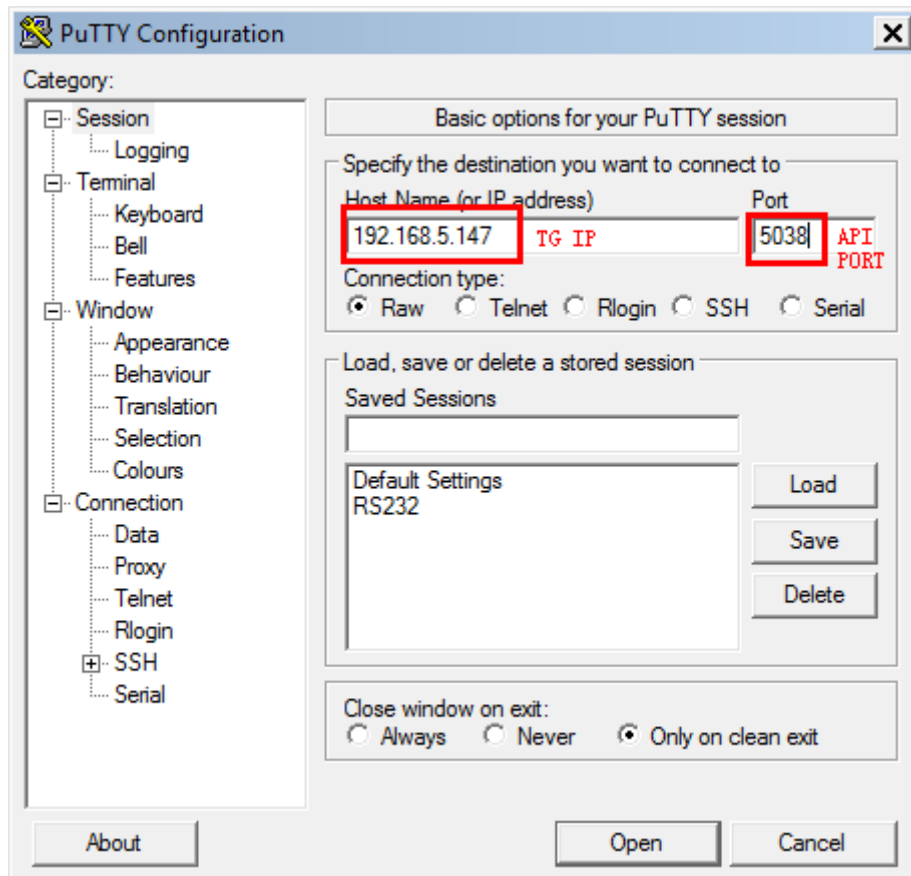
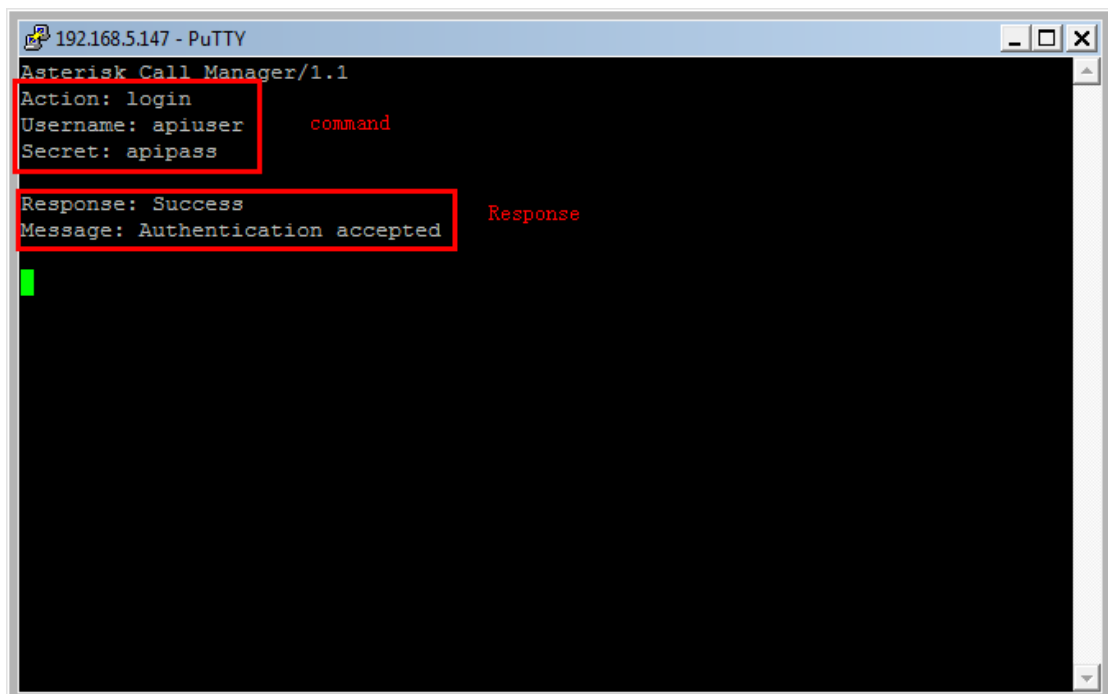


Figure 1 Enable API on TG

3. Connect SMS Client to NeoGate SMS server

SMS Client is connected to NeoGate SMS server through TCP. The IP address is the IP of NeoGate, and the port is the default 5038. User name and password are those configured in the previous step. After TCP connection is established, "Action: Login\r\nUsername: apiuser\r\nSecret: apipass\r\n\r\n" will be sent the server. If the server replies "Response: Success", it indicates the connection is successful. Other responses indicate unsuccessful connection.

Below is an example using putty to connect SMS server:

**Figure 2 Using Putty to Connect TG API****Figure 3 Login Command and Successful**

4. Execute smscommand

Check port status:

Client->Server command: "Action: smscommand\r\ncommand: gsm show spans\r\n\r\n\r\n"

Server->Client

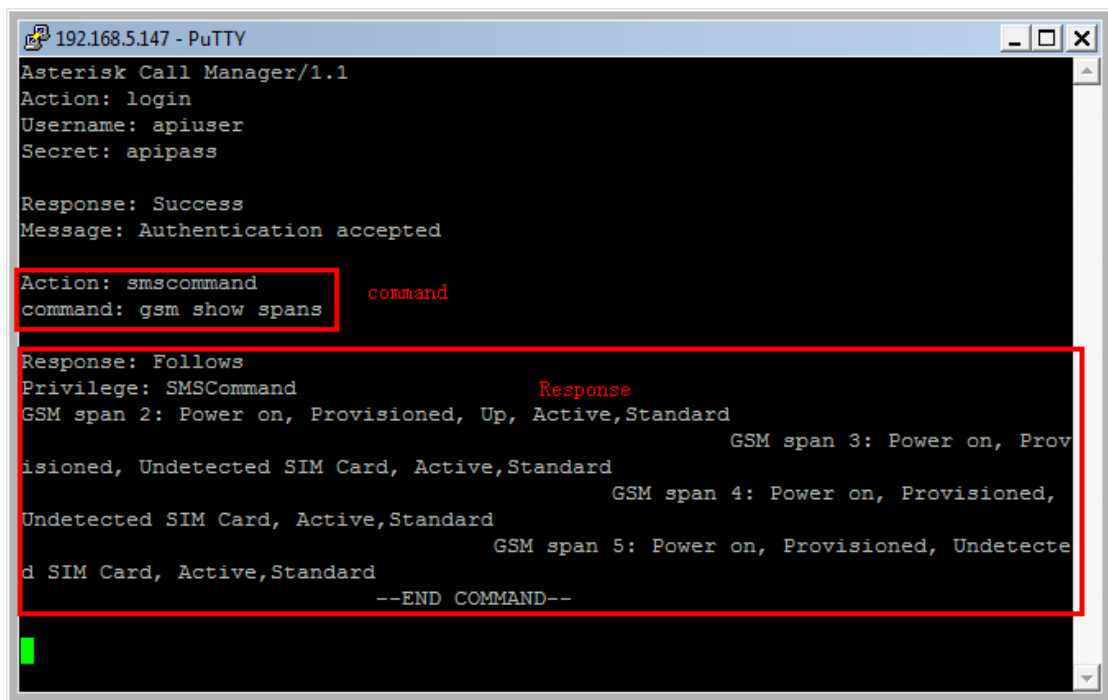
GSM span 2: Power on, Provisioned, Up, Active,Standard

GSM span 3: Power on, Provisioned, Undetected SIM Card, Active,Standard

GSM span 4: Power on, Provisioned, Undetected SIM Card, Active,Standard

GSM span 5: Power on, Provisioned, Undetected SIM Card, Active,Standard

--END COMMAND--



```

192.168.5.147 - PuTTY
Asterisk Call Manager/1.1
Action: login
Username: apiuser
Secret: apipass

Response: Success
Message: Authentication accepted

Action: smscommand
command: gsm show spans

Response: Follows
Privilege: SMSCommand
GSM span 2: Power on, Provisioned, Up, Active,Standard
GSM span 3: Power on, Provisioned, Undetected SIM Card, Active,Standard
GSM span 4: Power on, Provisioned, Undetected SIM Card, Active,Standard
GSM span 5: Power on, Provisioned, Undetected SIM Card, Active,Standard
--END COMMAND--

```

Figure 4 Example Command

Check specific port information:

If detailed information of a specific port to be checked, use the following command.

Client->Server

command: "Action: smscommand\r\ncommand: gsm show \$port+1\r\n\r\n"

\$port: the port you want to check

Server>Client

Response: Follows

Privilege: SMSCommand

D-channel: 2

Status: Power on, Provisioned, Up, Active,Standard

Type: CPE

Manufacturer: SIMCOM_Ltd

Model Name: SIMCOM_SIM900
 Model IMEI: 013227009596135
 Model CBAND: EGSM_MODE,ALL_BAND
 Revision: 1137B13SIM900M64_ST
 Network Name: CHINA MOBILE
 Network Status: Registered (Home network)
 Signal Quality (0,31): 30
 SIM IMSI: 460021059661838
 SIM SMS Center Number: +8613800592500
 Send SMS Center Number: Undefined
 Last event: Hangup
 State: READY
 Last send AT: AT+CREG?\r\n--END COMMAND--

```

192.168.5.147 - PuTTY
Asterisk Cell Manager/1.1
Action: login
Username: apiuser
Secret: apipass

Response: Success
Message: Authentication accepted
Action: smscommand
command: gsm show span 2

Response: Follows
Privilege: SMSCommand
D-channel: 2
Status: Power on, Provisioned, Up, Active,Standard
Type: CPE
Manufacturer: SIMCOM_Ltd
Model Name: SIMCOM_SIM900
Model IMEI: 013227009596135
Model CBAND: EGSM_M0
DE,ALL_BAND
Revision: 1137B13SIM900M64_ST
Network Name: CHINA MOBILE
Network Status: Registered (Home network)
Signal Quality (0,31): 30
SIM IMSI: 460021059661838
SIM SMS Cen
ter Number: +8613800592500
Send SMS Center Number: Undefined
Last event: Hangup
State: READY
Last send AT: AT+CREG?\r\n--END COMMAND--
  
```

Figure 5 Example Command

5. Send & Receive SMS

As the connection is successful, NeoGate SMS server can be used to send SMS and USSD. The server will also send a message to the Client upon receiving a SMS.

Send SMS

Client->Server

Command: "Action: smscommand\r\ncommand: gsm send sms \$port+1 \$dest \"\$message\"
\$id\r\n\r\n"

\$port: the trunk used to send the SMS

\$dest: the destination number

\$message: contents of the message. Note that if the message contains special characters like

\r\n, it is better that the message is encoded using url_encode. The maximum byte before encoding is 1024.

\$id: the unique identifier of the SMS

Response status: after the SMS is sent, the following message will be sent to the client. In the message, "ID" is the designated unique identifier when sending the SMS, "Smcs" is the SMS center number, "Status: 1" indicates successful sending, and "0" indicates failure.

Server->Client

Event: UpdateSMSSend

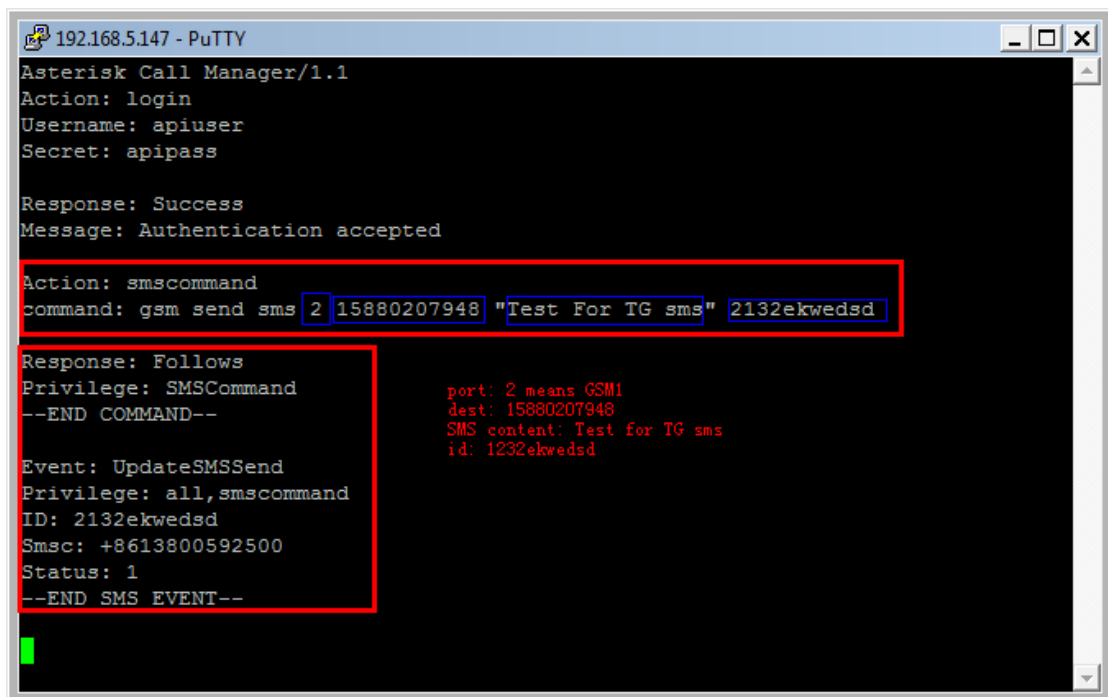
Privilege: all,smscommand

ID: 2

Smcs: +8613800592500

Status: 1

--END SMS EVENT--



```

192.168.5.147 - PuTTY
Asterisk Call Manager/1.1
Action: login
Username: apiuser
Secret: apipass

Response: Success
Message: Authentication accepted

Action: smscommand
command: gsm send sms 2 15880207948 "Test For TG sms" 2132ekwedsd

Response: Follows
Privilege: SMSCommand
--END COMMAND--

Event: UpdateSMSSend
Privilege: all,smscommand
ID: 2132ekwedsd
Smcs: +8613800592500
Status: 1
--END SMS EVENT--

port: 2 means GSM1
dest: 15880207948
SMS content: Test for TG sms
id: 1232ekwedsd

```

Figure 6 Send SMS

Receive SMS

Upon receiving an SMS, NeoGate will send the following contents to the Client, which will integrate the SMS based on the contents. The example is a message from China Mobile service hotline 10086 replying a balance inquiry. It is integrated by 2 pieces of a whole SMS.

"ID": the unique identifier of these SMS;

"GsmPort": the port that received the SMS;

"Sender": the number that sent the SMS;

"Recvtime": received time;

"Index": sequence number of the SMS piece

"Total": how many pieces of SMS is sent as a whole

"Smsc": SMS center number

Content: SMS content

--END SMS EVENT--

Event: ReceivedSMS

Privilege: all,smscommand

ID: E9510086

GsmPort: 5

Sender: 10086

Recvtime: 2013-11-13 14:59:13

Index: 2

Total: 2

Smsc: +8613800591551

Content: %EF%BB%BF%E8%BF%87%E7%9F%AD%E4%BF%A1%E3%80%81%E8%AF%AD%E9%9F%B3%E3%80%81%E7%BD%91%E7%AB%99%E8%87%AA%E5%8A%A9%E4%BA%A4%E8%B4%B9%EF%BC%8C%E6%9B%B4%E5%8F%AF%E4%BA%AB%E5%8F%979.8%E6%8A%98%E4%BC%98%E6%83%A0%EF%BC%8C%E8%AF%A6%E6%83%85%E7%99%BB%E9%99%86www.10086.cn%E6%9F%A5%E8%AF%A2%E3%80%82%E4%B8%AD%E5%9B%BD%E7%A7%BB%E5%8A%A8

```

192.168.5.147 - PuTTY
Response: Follows
Privilege: SMSCommand
--END COMMAND--

Event: UpdateSMSSend
Privilege: all,smscommand
ID: 2132ekwedsd
Smsc: +8613800592500
Status: 1
--END SMS EVENT--

Event: ReceivedSMS
Privilege: all,smscommand
ID:
GsmPort: 1
Sender: +8615880207948
Recvtime: 2013-11-27 09:39:46
Index: 1
Total: 1
Smsc: +8613800592500
Content: Reply+Test+For+TG+sms
--END SMS EVENT--

"ID": the unique identifier of these SMS;
"GsmPort": the port that received the SMS;
"Sender": the number that sent the SMS;
"Recvtime": received time;
"Index": sequence number of the SMS piece
"Total": how many pieces of SMS is sent as a whole
"Smsc": SMS center number

```

Figure 7 Receive SMS

6. Send USSD

Client->Server

Command: "Action: SMSCommand \r\ncommand: gsm send ussd \$port+1 \"\$message\""

[\$timeout]\r\n\r\n"

\$port: the port that sent the USSD

\$message: the contents of the USSD

\$timeout: timeout period, the default is 30 seconds.

Success:

1:Received USSD success on span: 6

\tUSSD Responses: 1

\tUSSD Code: 72

\tUSSD Len: 276

\tUSSD Message:

6B228FCE4F7F7528638C4E0A670D52A15385000A003830014F1860E06D3B52A8000A00313
001670065B04F1860E0000A003230018BDD8D3967E58BE2000A003330014E1A52A167E58
BE2000A003430014E1A52A1529E7406000A003630016570636E4E1A52A1000A003730015E
3875284FE1606F000A00393001004700334ECB7ECD000A002D002D002D

--END COMMAND--

USSD Code: the code value

USSD Message: the encoded content of USSD, the Client needs to decode the USSD with
USSD code.

Failure:

0:Send USSD failed on span 6

Timeout:

0:Send USSD timeout on span 6

<END>