

# GOIP SMS Interface

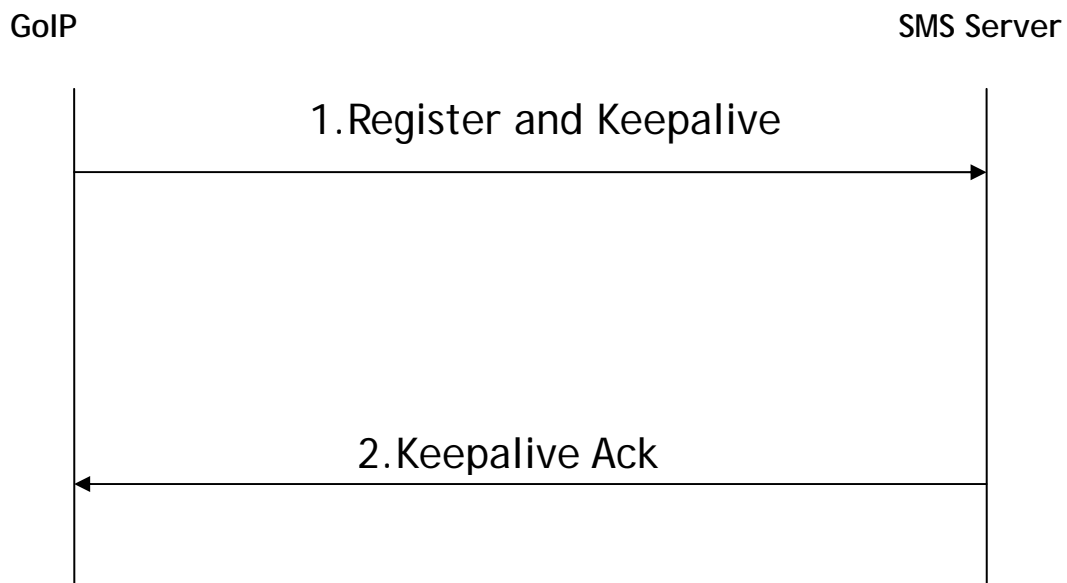
## ● Initialization

Communication of this system use clear text for transmission over UDP transport layer. Please setup the retransmission mechanism for reduces the packet loss between the SMS server and GoIP.

SMS Server Initialization: Set the authentication ID and password for GOIP, and save the list of IDs and passwords. Open the UDP port then start listening.

GOIP Parameters: Enable "SMS Sender" on the configuration page, fills your SMS server address and port, authentication ID and password (refer to above "SMS Server Initialization" for setting). Then restart the GOIP terminal.

## ● Registration and Keepalive



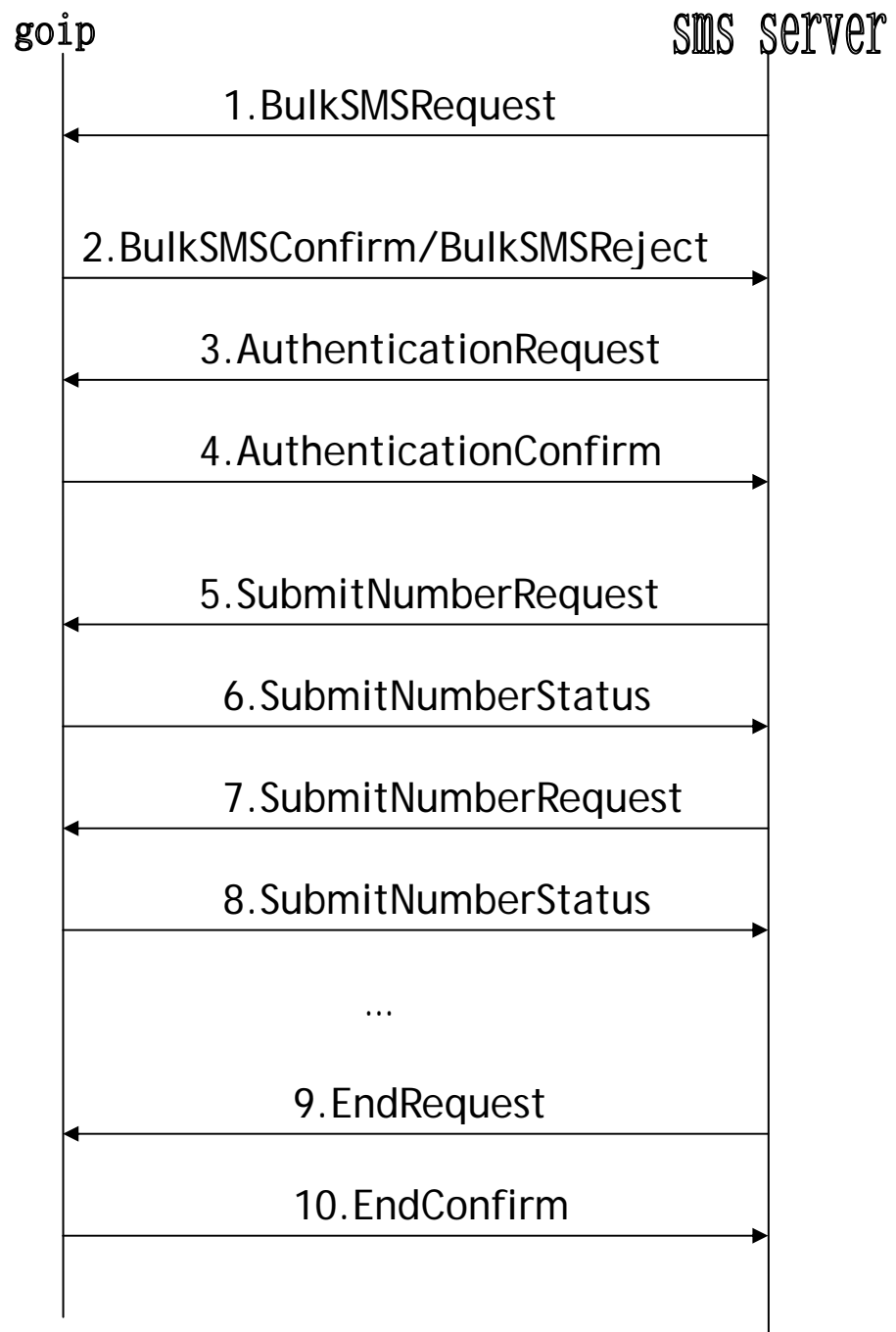
Step	Description
1. Register and Keep alive	GOIP will send a keepalive packet to server every 30s. Format: req:\$count;id:\$id;pass:\$password;num:\$gsm_num;signal:\$gsm_signal; Variable: \$count: counter for sending packets. Initialize to 1 when GOIP power up and increase by 1 after a packet is sent out. \$id: authentication id set in configuration page.

	<p>\$password: authentication password set in configuration page.          \$gsm_num: sim card number          \$gsm_signal: sim signal</p>
2. keepalive ack	<p>SMS server will verify the authentication id and password of the keepalive packet received. And send keepalive ack if the id and password are matched with the authentication list.          Format:          reg:\$count;status:\$status;</p>

e.g.

GOIP send the keepalive with "req:10;id:goipid1;pass:password1;" to SMS server. SMS server send the keepalive ack with "reg:10;status:0;" if goipid1 and password1 is valid.)

- Process for Bulk SMS



Step	Description
1. BulkSMSRequest	<p>At the beginning, SMS server will send a bulk SMS Request to GOIP, <b>include the SMS content and length in utf8 format. The SMS content should be limited to 3000 bytes or less.</b></p> <p>Format: MSG \$sendid \$length \$msg\n</p> <p>Variable:</p> <p>\$sendid: Integer, as a bulk SMS session identifier. Note, all messages described below should use the same "sendid" defined here if they are belong to a same bulk SMS session.</p> <p>\$length: Integer in utf8 format, as the SMS content length</p> <p>\$msg: SMS content in utf8 format</p>
2. BulkSMSConfirm BulkSMSReject	<p>GOIP will do the initialization for the bulk SMS when Bulk SMS Request received. Note, the other messages of this bulk SMS should be received in 90s after Bulk SMS Request received. Or, GOIP will stop the session and release.</p> <p>(1) When the initialization successful, GOIP will send a BulkRequest Confirm to server and request for authentication</p> <p>Format: PASSWORD \$sendid\n</p> <p>Variable:</p> <p>\$sendid: Integer, bulk SMS session identifier in BulkSMSRequest.</p> <p>(2) When the initialization failed, GOIP will send a BulkSMSReject to server, include the error message.</p> <p>Format: ERROR \$sendid \$errmsg\n</p> <p>Variable:</p> <p>\$sendid: Integer, bulk SMS session identifier in BulkSMSRequest.</p> <p>\$errmsg: String, Error message. Now too many request for bulk SMS received at the same time is the main reason of the failure. Note, GOIP only support 5 bulk SMS session at the same time. And, server should stop the session when receive bulkSMSRequest Reject. Then, try again later.</p>
3. AuthenticationRequest	<p>SMS server should send a AuthenticationRequest with the password to GOIP.</p> <p>Format: PASSWORD \$sendid \$password\n</p>

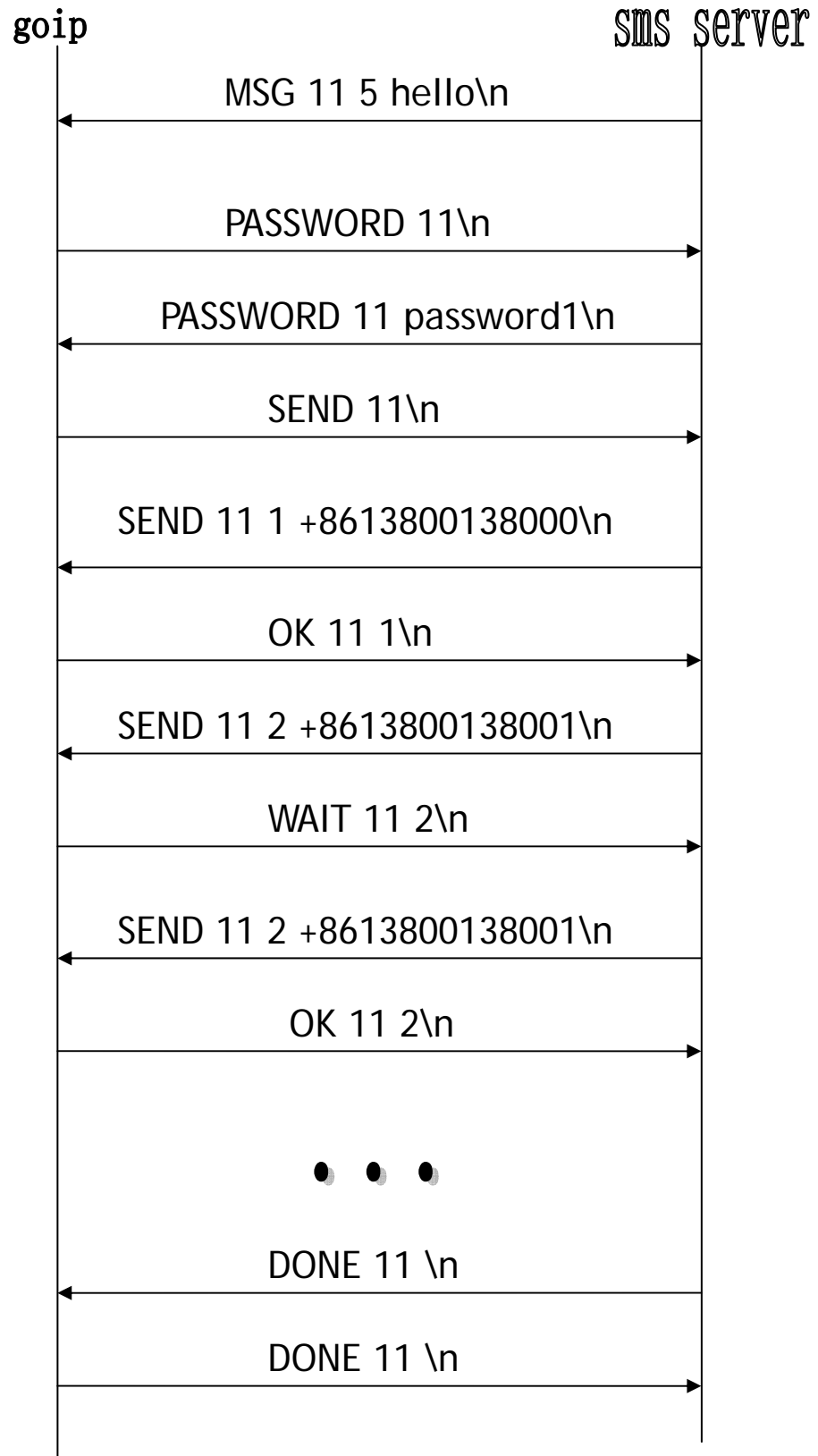
	<p>Variable:</p> <p>\$sendid: Integer, bulk SMS session identifier in BulkSMSRequest.</p> <p>\$password: The registration password of GOIP</p>
4.AuthenticationConfirm AuthenticationReject	<p>GOIP will verify the password in AuthenticationRequest and return the result to server.</p> <p>(1) When authentication successful, GOIP will send a AuthenticationConfirm to server.</p> <p>Format: SEND \$sendid\n</p> <p>Variable:</p> <p>\$sendid: Integer, bulk SMS session identifier in BulkSMSRequest.</p> <p>(2) When authentication failed, GOIP will send a AuthenticationReject to server and wait for the next AuthenticationRequest or the EndRequest</p> <p>Format: ERROR \$sendid PASSWORD\n</p> <p>Variable:</p> <p>\$sendid: Integer, bulk SMS session identifier in BulkSMSRequest.</p>
5.SubmitNumberRequest	<p>GOIP need about 2-5 seconds to send a SMS. So, to avoid packet loos, SMS server could send a SubmitNumberRequest to GOIP to get the sending status of an appointed number every serial seconds until the SMS is sent successfully. Note, the number should be in the full format of international telephone code. E.g. China: +86XXXXXX, H.K.: +852XXXXXXX. And, GOIP only save the sending status of the last 10 numbers in this group</p> <p>Format: SEND \$sendid \$telid \$telnum\n</p> <p>Variable:</p> <p>\$sendid: Integer, bulk SMS session identifier in BulkSMSRequest.</p> <p>\$telid: Integer, the unique SubmitNumberRequest sequence number defined by server.</p> <p>\$teinum: String, telephone number</p>
6. SubmitNumberStatus	<p>GOIP will send a SubmitNumberStatus to server when the SubmitNumberRequest received.</p> <p>(1) when sending SMS to appointed number successful, GOIP will send a SubmitNumberStatus with OK to server.</p> <p>Format: OK \$sendid \$telid\n</p>

	<p>Variable:  \$sendid: Integer, bulk SMS session identifier in BulkSMSRequest.  \$telid: Integer, unique sequence number in SubmitNumberRequest.</p> <p>(2) When sending failed, GOIP will send a SubmitNumberStatus with ERROR to server.  Format: ERROR \$sendid \$telid errorstatus:\$errorid\n</p> <p>Variable:  \$sendid: Integer, bulk SMS session identifier in BulkSMSRequest.  \$telid: Integer, unique sequence number in SubmitNumberRequest.  \$errorid: Integer, error code. Usually it is equal to 1.</p> <p>(3) When the telid in SubmitNumberRequest is not in the list of which save the recent 10 sending number, GOIP will save the telid in the waiting list and send a SubmitNumberStatus with WAIT.  Format: WAIT \$sendid \$telid\n</p> <p>Variable:  \$sendid: Integer, bulk SMS session identifier in BulkSMSRequest.  \$telid: Integer, unique sequence number in SubmitNumberRequest.</p>
7.SubmitNumberRequest	Note, SMS server should send the next SubmitNumberRequest to GOIP after the SubmitNumberStatus with OK or ERROR received Same as 5
8. SubmitNumberStatus	Same as 6
9. EndReqeust	SMS server could send a EndRequest to GOIP to finish the bulk SMS session. Format: DONE \$sendid\n Variable: \$sendid: Integer, bulk SMS session identifier in BulkSMSRequest.
10.End	GOIP will release the bulk SMS session resource when EndRequest received. And return a EndConfirm to server Format: DONE \$sendid\n Variable:

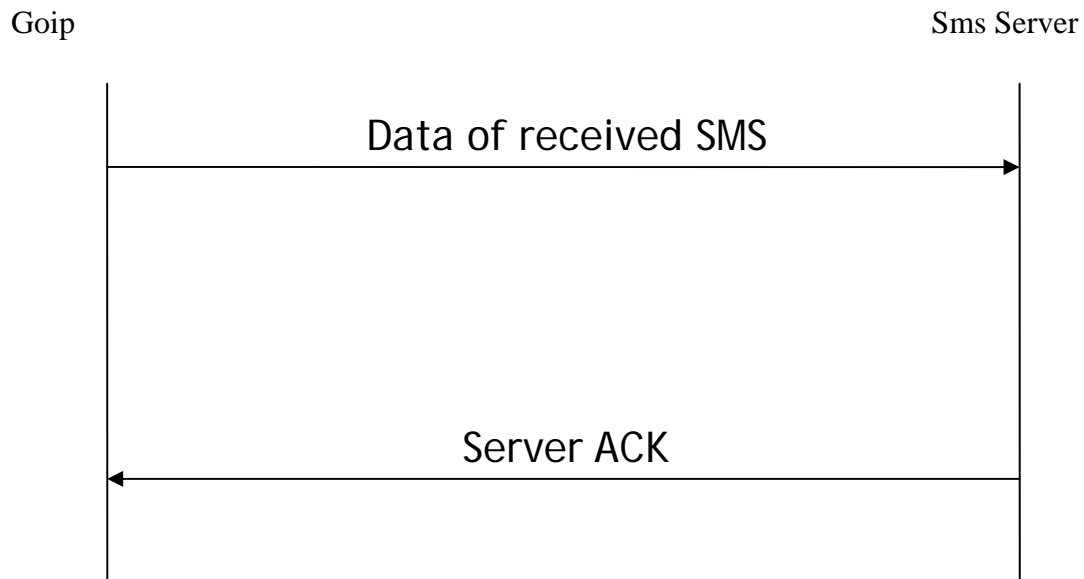
	\$sendid: Integer, bulk SMS session identifier in BulkSMSRequest.

Example:

Assume: sendid: 11, SMS content: hello, Goip password:password1



## RECEIVE SMS



Step	Description
1.Send data of receiverSMS to Server.	<p>If received SMS, Goip will relay the SMS to Server. Goip will resend the data if Server no response in 3 seconds, most resend 3 times.</p> <p>Format: RECEIVE:\$recvid;id:\$id;password:\$password;srcnum:\$srcnum;msg:\$msg</p> <p>Variable: \$recvid:int, count with the current time stamp decreasing; \$id: authentication id set in configuration page. \$password: authentication password set in configuration page. \$srcnum: Source mobile number \$msg: Content of SMS, utf8 format</p>
2.Server ACK	<p>Sms server will receive SMS and check the goip id and password, then send a ACK.</p> <p>Successful Format: RECEIVE \$recvid OK\n</p> <p>Error Format: RECEIVE \$recvid ERROR \$errmsg</p> <p>Variable:\$recvid: int, the goip count; \$errmsg:error message</p>

Example:

Assume: A Goip(id:goipid1, password:password1 )received a SMS "just a test" from mobile "+8613513415667". And it got a count 1270197307, then will send to SMS Server like this:

"RECEIVE:1270197307;id:goipid1;pass:password1;srcnum:+8613513415667;msg:just a test"

Sms server check goip id and password, Saved the SMS data, and reply:  
"RECEIVE 1270197307 OK"