



[www.routomessaging.com](http://www.routomessaging.com)  
[support@routotelecom.com](mailto:support@routotelecom.com)

## Connecting to SMS Gateway

The information contained in this document is proprietary and copyright and for the sole purpose of informing customers about the above service. The service is owned by Routo Telecommunications Ltd, 48 Charlotte Street, London, W1T 2NS, United Kingdom.

**Preface:**

Please register for [SMS/MMS and HLR Lookup account](#) for testing our SMS, MMS and HLR services and integration to our SMS, MMS APIs. The following features and services are available:

- Straightforward, SMS API, MMS API and HLR API integration
- Send and receive SMS and MMS using HTTP and SMPP and make HLR requests using HTTP
- Free sample code on how to send/receive SMS and MMS and make HLR lookups
- Free 24 hours support; our support to answer any questions
- Minimal SMS, MMS and HLR development time
- Deploy with confidence; we have over 6 years of experience in Text Messages Integration
- Sending and Receiving (with delivery reports) of SMS and MMS

**Table of contents:**

- 1 CHANGE HISTORY ..... 4
- 2 INTRODUCTION ..... 5
- 3 HOW TO REGISTER..... 6
- 4 SENDING SMS THROUGH THE SMPP PROTOCOL..... 7
  - 4.1 Connection parameters ..... 7
  - 4.2 Message formatting..... 7
  - 4.3 Server response..... 7
- 5 SENDING SMS THROUGH THE HTTP PROTOCOL ..... 9
  - 5.1 Minimum requirements ..... 9
  - 5.2 Connection Parameters..... 9
  - 5.3 Server response.....10
    - 5.3.1 SMS received response .....11
    - 5.3.2 SMS sent response.....11
  - 5.4 Supported delivery statuses .....12
    - 5.4.1 PHP example .....13
    - 5.4.2 ASP example .....13
  - 5.5 Sending of the bulk messages.....14
    - 5.5.1 Sample bulk SMS message.....14
  - 5.6 Unicode standard implementation.....14
    - 5.6.1 Unicode message sample .....14
    - 5.6.2 Unicode Conversion Tools .....14
- 6 CHECKING THE ACCOUNT BALANCE .....15
- 7 RECEIVING MESSAGES (INCOMING SMS NUMBERS) .....16
  - 7.1 Receiving SMS over Long numbers through HTTP.....16
    - 7.1.1 PHP Example.....17
    - 7.1.2 ASP Example.....18
  - 7.2 Receiving SMS over Long numbers through SMPP .....18
  - 7.3 RotoMessaging Desktop SMS Sender .....20
- 8 CONTACT US .....21

**List of code snippets:**

- Snippet 1: PHP Example .....13
- Snippet 2: ASP Example .....13
- Snippet 3: PHP example .....18
- Snippet 4: ASP example .....18

**List of tables:**

- Table 1: Change history .....4
- Table 2: Message formatting/originator.....7
- Table 3: Message formatting/recipient.....7
- Table 4: Optional HTTP parameters .....10
- Table 5: SMS received responses .....11
- Table 6: Supported delivery statuses.....13
- Table 7: Long numbers parameters.....16

# 1 Change History

Date	Changes	Section
2008-02-09	Initial release	All
2008-02-25	Update	All
2009-03-23	Update	All
2009-07-13	Update	Receiving messages (Incoming SMS Numbers)
2009-08-11	Update	Server response (Delivery reports examples)
2009-10-06	Update	Supported delivery statuses

Table 1: Change history

## 2 Introduction

The RotoMessaging SMS Gateway is a platform capable of delivering SMS messages to any CDMA/GSM mobile handset through partnership with more than 600 mobile operators.

We have a clear focus on SMS mobile messaging with advanced features including sending picture messages, ring-tone, logo, business card, WAP Push, V-Calendar, V-Card, Flash messages and Unicode messages. This document is an introductory guide for sending and receiving SMS through HTTP and SMPP protocols.

By using the RotoMessaging SMS gateway, registered users can send different types of messages:

- **Plain SMS** - standard SMS message
- **Unicode Messages** – SMS with standard character encodings
- **Long and Concatenation**- the method for sending more than 160 characters of information where several separate short messages to our gateway will be displayed as one message
- **Flash SMS** – an SMS message which appears directly on the mobile phone display instead of the usual inbox; this type of message can be useful if you want to catch recipient's attention immediately
- **MMS notifications** – an SMS message with a link for an image, video, audio or rich text which are stored on the Web server
- **Binary Messages/MMS** – rich content SMS messages with more operator logos, ringtones, telephone settings and WAP push messages
- **Ring Tone** – an SMS containing melodies and tunes which can give a unique sound to the GSM phone ringtone you can send to your contacts
- **vCard** – an SMS containing electronic business cards
- **vCalendar** – an SMS containing a calendar entry and schedule information
- **Operator Logos** – an SMS containing a picture that can be sent to a handset as either a single message or a series of concatenated SMS messages
- **Picture Messages** – an SMS containing monochrome pictures which can be sent as a series of SMS messages
- **WAP bookmarks** – an SMS containing a web link which makes it easy to open the page in the browser
- **EMS** – this is an Enhanced Messaging Service which provides rich-media content e.g. picture, ring tones

### 3 How to Register

We do not charge to sign up and use our SMS gateway. In order to register for RotoMessaging SMS test account, please visit: <http://www.routomessaging.com/free-sms-test-account.pmx>

After creating an account you will receive SMS test credit which will allow you to start sending SMS (text messages) immediately.

You can test the features and performance of our routes to over 600 networks with more than 200 countries covered, by any of the following methods:

- **Connecting to our SMS Gateway** - API integration through SMPP or HTTP protocol
- **Sending an SMS from your PC** – “RotoMessaging Desktop SMS Sender“, our Bulk SMS desktop application with integrated feature of 2 way messaging for sending and receiving SMS messages is available at <http://www.routomessaging.com/sms-downloads.pmx> (versions for Windows, Linux and Mac OS)
- **Sending an SMS from our Web Site** – our “Web SMS sender” (embedded in the RotoMessaging Command Centre) available at <https://www.routomessaging.com/cust/index.php>

For more information please contact:

- our **24/7 support team** through real time live chat at: <http://www.routomessaging.com>
- or send us an **Email** [support@routotelecom.com](mailto:support@routotelecom.com)

## 4 Sending SMS through the SMPP protocol

This section of the document will describe the procedure for sending an SMS message through the SMPP (Short Message Peer to Peer) protocol.

RotoMessaging supports the SMPP protocol specifications 3.3 and 3.4. You can start using SMPP connection as soon as you register for an account, as it is enabled by default.

### 4.1 Connection parameters

In order to be able to send SMS through SMPP protocol, you need to setup the following connection parameters:

- **Host:** smsc5.routotelecom.com, smsc6.routotelecom.com
- **Port:** 7777
- **Default encoding:** ISO encoding (DCS=0)
- **For GSM encoding** (DCS=1)
- **Number of concurrent connections:** 5(default)
  - **Note:** more connections are available at user's request

### 4.2 Message formatting

The SMS messages are sent according to the ISO-8859-1 and GSM 3.38 standards. The US-ASCII encoding is supported only in case no special characters are included.

Every short message has a source and destination number **ton/np**i according to the originator and the recipient numbers.

originator	TON	NPI
alphanumeric	5	0
local	2	1
international	1	1

Table 2: Message formatting/originator

recipient	TON	NPI
international	1	1

Table 3: Message formatting/recipient

### 4.3 Server response

The customers who are sending the messages through the SMPP protocol will receive a response from the SMS gateway for every message that has been sent through.

It is up to the client to establish the transceiver or receiver connection to our gateway in order to receive this response.

The response is sent through the delivery\_sm PDU. The following is the list of all supported error codes in the delivery\_sm PDU:

- DELIVRD (0)

- EXPIRED (3,253)
- DELETED (4,252)
- UNDELIV (5,9,251)
- ACCEPTD (6)
- REJECTD (1,8,10,11,49,69,72,73,80,249)
- UNKNOWN (any other value)

The more detailed specification can be found in the [SMPP 3.4 pdf document](#). In addition, please visit <http://www.routomessaging.com/sms-api.pmx> to review all available documentation.

## 5 Sending SMS through the HTTP protocol

This section of the document will describe the procedure for sending an SMS message through the HTTP protocol.

### 5.1 Minimum requirements

The following are the minimum requirements for sending SMS through the HTTP protocol:

- registered account on our web site
- username, password, source and destination address
- message text

A sample SMS message presents mandatory parameters:

<http://sm5c5.routotelecom.com/SMSsend?number=12345678&user=username&pass=password&message=Test+message&ownnum=Routo>

In order to send messages using HTTP connection, customers can use different approaches using a number of development languages such as PHP, ASP, PERL C# and JAVA.

For a higher level of security, our customers can use an SSL connection for sending SMS and MMS content and receiving delivery reports. In this case, sample SMS message should look like this: <https://sm5c5.routotelecom.com/SMSsend?number=12345678&user=username&pass=password&message=Test+message&ownnum=Routo>.

Further information about sending SMS messages through HTTP protocol can be found in RotoMessaging HTTP Specification PDF document which is available at: <http://www.routomessaging.com/sms-api.pmx>

### 5.2 Connection Parameters

There are two types of parameters: mandatory and optional. The following are the mandatory parameters:

- username
- password
- message
- source address
- destination address
- type of message

There are a number of optional parameters. The complete list of the parameters is listed below:

Parameter	Description
<b>user</b>	Your RotoMessaging client <b>username</b>
<b>pass</b>	Your RotoMessaging account <b>password</b> . If the password contains special characters (space,#,\$,+,%,=) it has to be <b>URL encoded</b>
<b>number</b>	The <b>number</b> the message is sent to. It has to be in the international format without '+' or spaces. <b>e.g.</b> 441234567890. Multiple numbers are allowed (up to 10) and should be <b>separated by comma</b>

<b>ownnum</b>	The <b>number from</b> the message appears to have been sent from – it can be <b>numeric</b> or <b>alphanumeric</b> (up to 11 characters)
<b>message</b>	The <b>message body</b> : <ul style="list-style-type: none"> <li>• for operator logos or graphics the message has to be <b>MIME</b> and <b>URL encoded</b></li> <li>• for MMS messages it has to be <b>HEX encoded</b></li> <li>• for all other messages it has to be <b>URL encoded</b>.</li> </ul>
<b>type</b>	<b>Message types</b> : <ul style="list-style-type: none"> <li>• <b>standard SMS</b> (default)</li> <li>• <b>LongSMS</b> (concatenated messages up to 39015 characters)</li> <li>• <b>FlashSMS</b></li> <li>• <b>RingTone</b></li> <li>• <b>OperatorLogo</b></li> <li>• <b>PictureMessage</b></li> <li>• <b>Binary</b></li> <li>• <b>Unicode</b></li> <li>• <b>Longunicode</b></li> <li>• <b>uni_flash</b> (unicode flash)</li> <li>• <b>bookmark</b> (WAP bookmark)</li> <li>• <b>mms-gif</b></li> <li>• <b>mms-jpg</b></li> <li>• <b>mms-jar</b></li> <li>• <b>mms-wav</b></li> <li>• <b>mmsNotification</b></li> </ul>
<b>op</b>	<b>Mobile Operator Code</b> - this parameter must be used when sending Operator Logos
<b>model</b>	The <b>manufacturer</b> of the recipient's phone. It can be: Nokia (default) or Erickson. Set type to Ericsson for all EMS supporting phones. It must be set for RingTone and PictureMessage
<b>delivery</b>	<b>delivery report request</b> – this parameter should be set to 1 if you want to receive the delivery report
<b>mess_id</b>	<b>message ID</b> – this is the message id used for the delivery report; it must be set if the delivery has been set to 1; it can be any string up to 32 characters

Table 4: Optional HTTP parameters

### 5.3 Server response

For every message which has been sent through the HTTP protocol, the customers will receive the response from our SMS gateway. There are two types of gateway responses:

- **SMS received response** - response from our SMS gateway when the message is submitted from client's side (see section 5.3.1)

- **SMS sent response** - response from our SMS gateway with the delivery report for the message that has been sent (see section 5.3.2)

### 5.3.1 SMS received response

The table below lists all possible responses from our SMS gateway when the message is submitted from the client's side.

Parameter	Description
<b>success</b>	sending successful
<b>error</b>	not all required parameters are present
<b>auth_failed</b>	incorrect username and/or password
<b>wrong_number</b>	the number contains non-numeric characters
<b>not_allowed</b>	you are not allowed to send to this number
<b>too_many_numbers</b>	sending to more than 100 numbers per request
<b>no_message</b>	no message given
<b>too_long</b>	message is too long
<b>wrong_type</b>	an incorrect message type was selected
<b>wrong_message</b>	vCalendar contains wrong message
<b>wrong_format</b>	the wrong message format was selected
<b>bad_operator</b>	wrong operator code
<b>failed</b>	internal error
<b>sys_error</b>	the system error

Table 5: SMS received responses

### 5.3.2 SMS sent response

We optionally provide delivery reports for the messages that are successfully sent through our system. The customers need to have a web based script which is capable of receiving handset delivery reports.

In order to be able to receive handset delivery reports, you should first prepare a **CGI script**. The link to this script must be registered on our system and the registration is done through our command centre which is available at: <https://www.routomessaging.com/cust/index.php>.

When requesting handset delivery report you have two choices:

1. you can send message to one subscriber only with unique message ID
2. or you can send messages to more than one subscriber (numbers are separated with commas) with one message ID for whole bulk. Delivery parameter has to be set to 1.

Your CGI script should expect 3 parameters: **mess\_id**, **status** and **number**:

- **mess\_id** - is message ID you sent with message
- **status** - is final status of the message
- **number** - is destination number to which message is sent (useful when you send bulk of messages with one message ID)

## 5.4 Supported delivery statuses

The table below lists all supported statuses which can be delivered to the mobile subscribers.

Parameter	Description
0	Delivered
1	Rejected: Message length is invalid
2	Subscriber absent
3	Device memory capacity exceeded
4	Equipment protocol error
5	Equipment not supported
6	Equipment not SM equipped
7	Unknown service centre
8	Service centre congestion
9	Undeliverable
10	Rejected: Invalid source address
11	Invalid destination address
12	Illegal subscriber
13	Teleservice not provisioned
14	Illegal equipment
15	Call barred
16	Facility not supported
17	Subscriber busy for SM
18	System failure
19	Message waiting, list full
20	Data missing
21	Unexpected data value
22	Resource limitation
23	Initiating release
24	Unknown alphabet
25	USSD busy
26	Duplicated invoke ID
27	No supported service
28	Mistyped parameter
29	Unexpected response from peer
30	Service completion failure
31	No response from peer
32	Invalid response received
34	Invalid destination
49	Message type not supported
50	Destination blocked for sending
51	Not enough money
52	No price
67	Invalid esm_class field data
69	Rejected by SMSC
72	Rejected: Invalid source address TON
73	Rejected: Invalid source address NPI
80	Rejected: Invalid destination address TON
81	Rejected: Invalid destination address NPI
88	Throttling error

97	Rejected: Invalid scheduled delivery time
100	Error sending message
247	Sent
248	Sent
249	Rejected
250	Accepted
251	Undeliverable
252	Deleted
253	Expired
254	Roaming level not supported
255	Unknown error

Table 6: Supported delivery statuses

For further information, please review our technical documentation about HTTP integration and customization options (with lots of sample code in PERL, PHP, ASP, JAVA and C#) at:

<http://www.routomessaging.com/sms-api.pmx>

#### 5.4.1 PHP example

Below is the PHP example.

```
<?php
$mess_id=$_REQUEST["mess_id"];
$status=$_REQUEST["status"];
$number=$_REQUEST["number"];
$file_name = fopen("/tmp/dreports.txt","a+");
fwrite($file_name,"[" . date("d:m:Y:H:i:s") . "]:\n");
fwrite($file_name,"$mess_id\t$number\t$status\n");
fwrite($file_name,"-----\n");
fclose($file_name);
?>
```

Snippet 1: PHP Example

#### 5.4.2 ASP example

Below is the ASP example.

```
<%
dim mess_id,status,number
dim fs,tfile
dim ctime,cdate
mess_id = Request.Form("mess_id")
status = Request.Form("status")
number = Request.Form("number")
cdate = Date
ctime = time
set fs=Server.CreateObject("Scripting.FileSystemObject")
set
tfile=fs.OpenTextFile(server.MapPath("files")+"\dreports.txt",8,true)
tfile.WriteLine(cdate+ctime)
tfile.WriteLine(mess_id+vbt+number+vbt+status)
tfile.WriteLine(1)
tfile.Close
%>
```

Snippet 2: ASP Example

## 5.5 Sending of the bulk messages

The customers who want to send bulk messages can do so by either:

1. sending them one by one
2. or sending the desired number of messages separated by a comma

**Note:** currently our customers can send up to 10 messages at once.

If you would like to receive the delivery report for all of the messages, the delivery parameter has to be set to 1 and the unique message ID has to be specified.

This unique message ID will be used for sending the delivery reports for each of the messages along with the information about the mobile phone number the message has been sent to.

### 5.5.1 Sample bulk SMS message

This is the sample of the bulk SMS message:

[http://sm5c5.routotelecom.com/SMSsend?user=username&pass=password&number=12345678,87654321&ownnum=test&message=bulk&delivery=1&mess\\_id=someID](http://sm5c5.routotelecom.com/SMSsend?user=username&pass=password&number=12345678,87654321&ownnum=test&message=bulk&delivery=1&mess_id=someID)

**Note:** according to our specification, message ID's can be up to 32 characters long.

## 5.6 Unicode standard implementation

RoutoMessaging is following the latest version 5.1.0 Unicode standard. Because we are following this standard, we are able to support the international exchange and presentation of the text messages written in a variety of languages and technical disciplines.

In case a Unicode message is 70 characters long and 2 bytes are allocated for each character, one message is sent. In case of a longer message (Long Unicode) it is divided into a number of messages that are merged when delivered (**note:** Long Unicode message can consist of maximum 4 parts/messages).

### 5.6.1 Unicode message sample

Below is the sample of the Unicode message:

<http://sm5c5.routotelecom.com/SMSsend?user=username&pass=password&number=recipient&ownnum=sender&message=062D062706330628064A06460020062706440630064A0020064A06330645062D002006440644064606270633&type=unicode>

### 5.6.2 Unicode Conversion Tools

There are a number of tools for Unicode conversion. For example:

<http://people.w3.org/rishida/scripts/uniview/conversion.php>

## 6 Checking the Account balance

Our customers can check their current account balance by using the balance script. In order to use the script, the users have to provide their username and password at the following URL:

<http://sm5c5.routotelecom.com/balance.php?username=username&password=password>

If the user credentials have been authenticated, the script will return the balance. If some of the details are incorrect, the script will return the error message accordingly.

The following are the error messages which can be returned by the script:

- **error** – either username or password has not been provided
- **auth\_failed** – invalid username or password
- **sys\_error** – server side error

## 7 Receiving messages (Incoming SMS Numbers)

We can offer our clients the service of renting long numbers, allowing you to receive text messages from a mobile handset regardless of the geographical location.

There are 3 connection methods for receiving these messages and those are:

- **through HTTP** – see section 7.1
- **through SMPP** – see section 7.2
- **through RoutoMessaging Desktop SMS Sender application** - see section 7.3

For further information about renting long numbers please contact our Sales Department at [sales@routotelecom.com](mailto:sales@routotelecom.com) .

### 7.1 Receiving SMS over Long numbers through HTTP

This method is easy to implement and is useful for the customers who want to avoid carrier grade protocols.

The SMS Gateway delivers all of the received messages to the URL specified by the customer. To receive the messages through the HTTP protocol, the customer should provide the SMS Gateway with an URL linked to the CGI script, which will accept the incoming messages.

In case the initial attempt to deliver these messages fails, the SMS Gateway will make another 3 consecutive attempts to resend the data.

In case all 4 attempts are unsuccessful, the process ends. All the requests will be sent as an HTTP POST request and the SMS Gateway will expect to receive a valid HTTP response in order to confirm the delivery.

The format of this request will be as follows:

[http://www.domain.com/cgi-bin/script.php?user=username&pass=password&dcs=0&org=originator\\_number&dest=destination\\_number&message=message\\_text](http://www.domain.com/cgi-bin/script.php?user=username&pass=password&dcs=0&org=originator_number&dest=destination_number&message=message_text)

The table below explains each of the available parameters:

Parameter	Description
<b>user</b>	the customer's RoutoMessaging account username
<b>pass</b>	the customer's RoutoMessaging account password
<b>dcs</b>	the DCS settings of the message (usually '0')
<b>org</b>	the number the message was sent from (originating number)
<b>dest</b>	the long number allocated to the client account; this value will remain constant
<b>message</b>	the message text

Table 7: Long numbers parameters

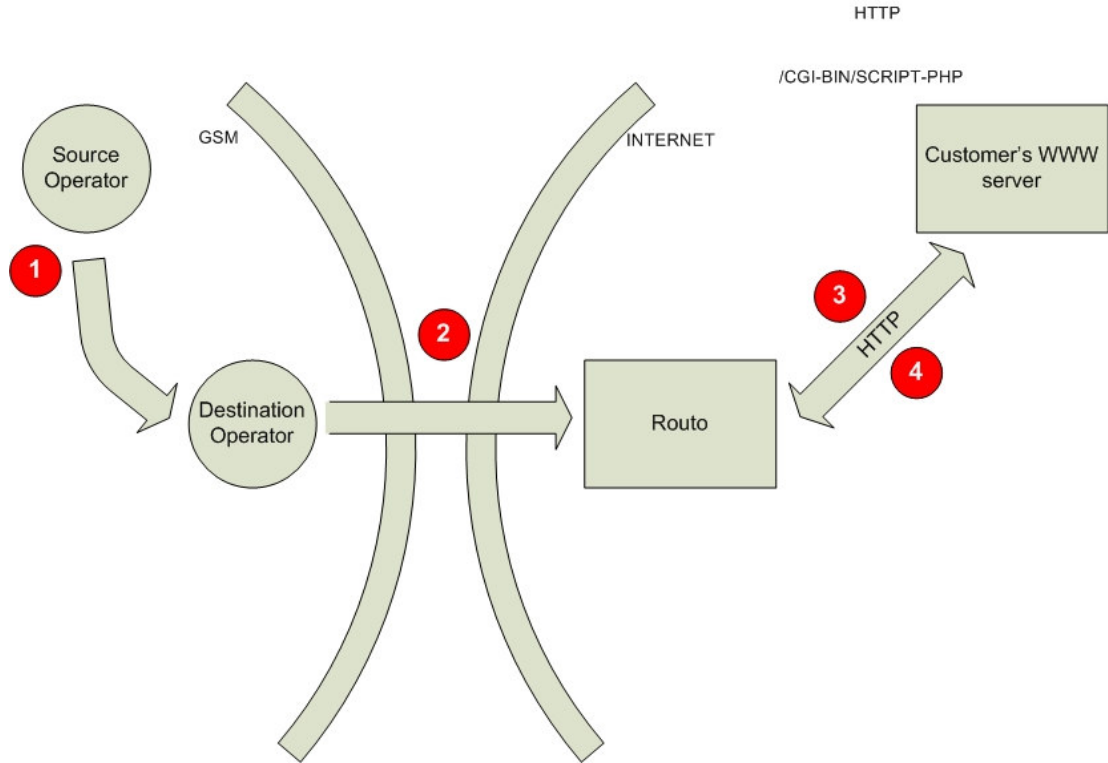


Figure 1: The procedure for receiving messages sent to long number through HTTP protocol

- **Step 1** the end-user's mobile handset sends a message to the long incoming number
- **Step 2** the destination operator receives the message and forwards it to the SMS Gateway
- **Steps 3 and 4:** once the message is received, RotoMessaging will issue an HTTP POST request to the customer's URL
- **(Step 3)** the CGI script should process this request and return a valid HTTP header (Step 4).

In case all of these steps were successfully completed, the SMS Gateway will consider the message as being delivered.

The user's username and password have been used to:

1. authenticate the HTTP request
2. authenticate the customer as a valid SMPP receiver

If necessary, RotoMessaging can also issue an HTTPS request to the customer.

### 7.1.1 PHP Example

The following snippet demonstrates the PHP code for receiving long numbers.

```
<?php
$user=$_REQUEST["user"];
$pass=$_REQUEST["pass"];
$org=$_REQUEST["org"];
$dest=$_REQUEST["dest"];
$dcs=$_REQUEST["dcs"];
$msg=$_REQUEST["message"];
$file_name = fopen("/tmp/incoming.txt","a");
```

```
fwrite($file_name,"[" . date("d:m:Y:H:i:s") . "]: $username\n");  
fwrite($file_name,"$user\t$pass\t$org\t$dest\t$dcs\t$msg\n");  
fwrite($file_name,"-----\n");  
fclose($file_name);  
?>
```

Snippet 3: PHP example

### 7.1.2 ASP Example

The following snippet demonstrates the ASP code for receiving long numbers.

```
<%  
dim username,pass,dcs,org,dest,message  
dim fs,tfile  
dim ctime,cdate  
user = Request.Form("user")  
pass = Request.Form("pass")  
dcs = Request.Form("dcs")  
org = Request.Form("org")  
dest = Request.Form("dest")  
message = Request.Form("message")  
cdate = Date  
ctime = time  
set fs=Server.CreateObject("Scripting.FileSystemObject")  
set tfile=fs.OpenTextFile(server.MapPath("files")+"\incoming.txt",8,true)  
tfile.WriteLine(cdate+ctime)  
tfile.WriteLine(org+vtab+dest+vtab+message+vtab+dcs)  
tfile.WriteLine(1)  
tfile.Close  
%>
```

Snippet 4: ASP example

## 7.2 Receiving SMS over Long numbers through SMPP

The SMPP is a robust industry protocol, which allows higher traffic volumes and better connectivity maintenance. It is possible to receive messages through this protocol by connecting as a valid SMPP v3.4 receiver.

Please note that this service will accept only receiver connections (Rx) and not Transmitter (Tx) or Transceiver (TRx) connections.

Once you have registered for an account please contact your account manager to allocate a Long number.

**Important note:** this service requires dedicated account credentials, and once gained, the customer will be provided with the required connectivity details.

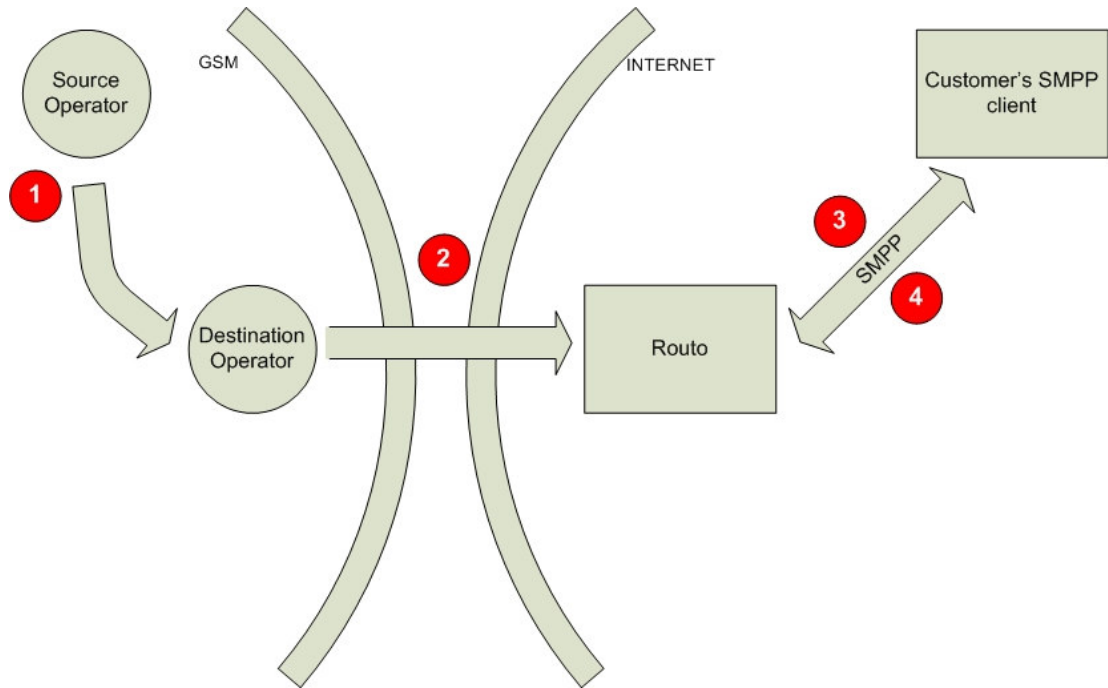


Figure 2: The procedure for receiving messages sent to long number through SMPP protocol

The following are the steps for receiving the message through the SMPP protocol:

- **Step 1:** a text message will be originated and sent to your incoming number
- **Step 2:** this message will be accepted by the destination operator and sent to the SMS Gateway. Once the message is received, the SMS Gateway will store it in a local cache, and wait for the customer to connect as a valid SMPP receiver
- **Step 3:** If the customer is already connected, the SMS Gateway will issue the SMPP command 'delivered\_sm' to the customer's SMPP client
- **Step 4:** after processing this request the customer should reply with a valid SMPP response – 'delivery\_sm\_resp' Once the SMS Gateway receives this response, it will consider the message as being delivered.

### 7.3 RotoMessaging Desktop SMS Sender

The RotoMessaging Desktop SMS Sender is a desktop application for sending and receiving SMS directly from PC, which has an integrated feature for two way messaging allowing full interaction between our clients and their customers.

This functionality is possible because the Desktop SMS Sender already has a UK Long number which can be used immediately. The screenshot below demonstrates the Desktop SMS Sender interface.

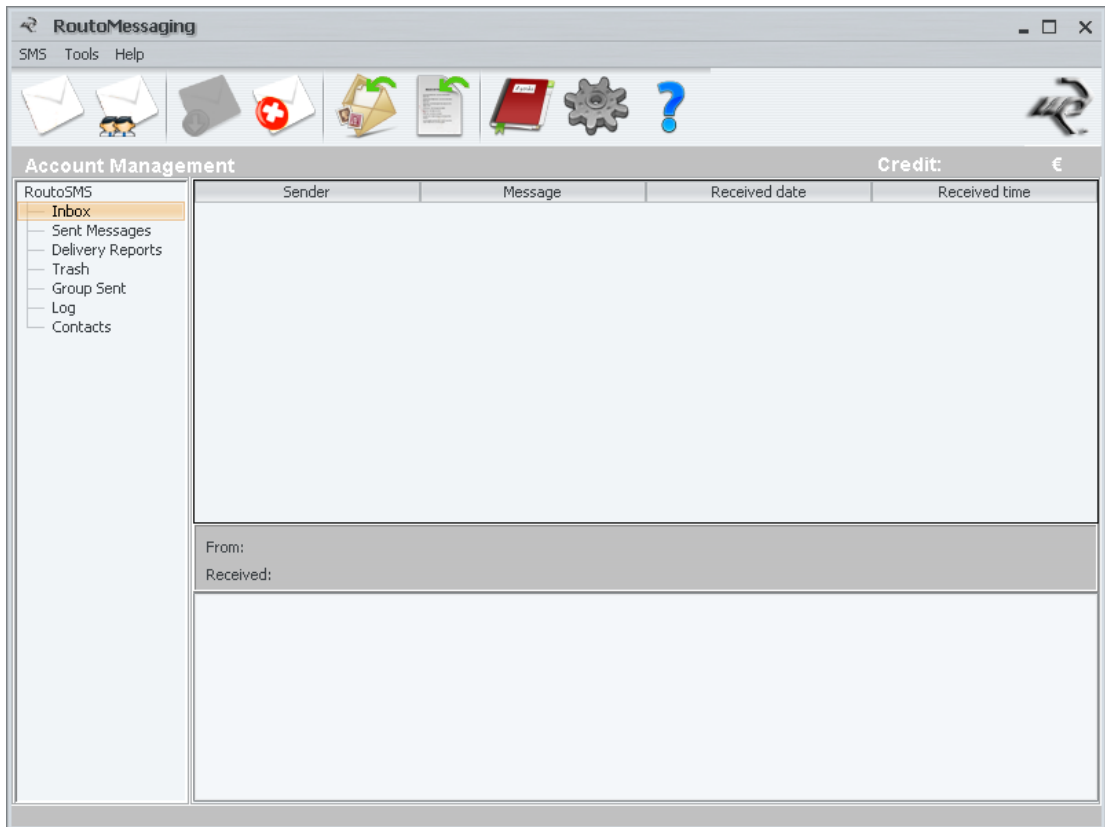


Figure 3: RotoMessaging Desktop SMS Sender interface

You can download the Desktop SMS Sender from the following URL:  
<http://www.rotomessaging.com/sms-downloads.pmx>

## 8 Contact us

For further information please contact:

- our **24/7 support team** through real time live chat at <http://www.routomessaging.com>
- send an email to our **Technical Support**: [support@routotelecom.com](mailto:support@routotelecom.com)
- or contact our **Sales Department** at [sales@routotelecom.com](mailto:sales@routotelecom.com) for further information about renting Long number