



Technical Specification HTTP Gateway API

Sending a MT message

To send a text message, your server has to invoke the URI of Cellcom by means of HTTP(s) GET. It is highly recommended that your server use HTTP keep-alive semantics, when using HTTPS, by maintaining and re-using the same TCP connection to transport multiple sequential HTTP requests and releasing it only when no longer needed. This minimizes the amount of SSL key exchanges. It should be noted that your server is allowed to issue multiple HTTP requests in parallel, but when using HTTPS they must be on distinct SSL connections.

The value of each individual parameter must be URL encoded.

When the URI is opened the gateway will authenticate the request and then send the message using the specified recipient phone number and message text. The server will return a response code as the body of the HTTP response.

The URI to send MT messages:

`http://www.cellcom.be/data/bulkHTTP.php`

The following variables apply:

PARAM.	NAME	INFO
CLID	<i>Client ID</i>	Your customer id (find the value in your Cellcom account)
GWID	<i>Gateway ID</i>	The gateway number of your service (find the value in your Cellcom account)
PW	<i>Password</i>	Your password. This is the md5 hash of your account password. Be aware that this parameter will change when you change your account password. (find the value in your Cellcom account)
TO	<i>Recipient</i>	Number where we will send the message to. Use the international format for ex '32496233133', no spaces!
SID	<i>Sender ID</i>	The sender ID of the message. This will be shown as the sender on the screen of the mobile Phone. You can use an international phone number, or 11 alphanumeric characters. So you can use for example 'Cellcom' or '3228886991'. If the network of the recipient would not support the sender id, Cellcom will replace this value. Since there are more and more operators that do not allow alphanumeric sender id's, there is a high chance we will replace the value to be able to deliver the message.
M	<i>Message</i>	The actual message. If you use more than 160 characters, we will charge extra for every message you start. Some characters count as 2. Please make sure you use URL encoding for this parameter. Use urlencode in PHP, or Server.URLEncode in ASP.

Optional parameters:

PARAM. NAME	INFO
COD <i>Coding</i>	Has to be 2 to send a binary message. M has to be hexadecimal in this case.
UDH <i>User Data Header</i>	Has to be hexadecimal. Please do not put a length byte as first byte. This will be added automatically.
TTL <i>Time To Live</i>	Time to live in minutes. Min value: 60 Max value: 10080 (default) Some operators don't support a TTL longer then 2880, we will automatically adjust the value if necessary.
DR <i>Delivery Receipts</i>	Has to be 1 if you want a delivery receipt trough a HTTP link for this message. (see section "Status Reports") Default value: 0
MID <i>Message ID</i>	You can provide your own message ID using this value. Min value: 1 Max value: 4294967295 If the parameter is defined, this value will be used as message ID in the result code of the request and in the status report. If a message was already send with the same ID, you will get a normal response, but we won't resend the message. This parameter will avoid double sending when a timeout happens during the first request.
ST <i>Send Time</i>	If the message must not be send immediately, but at a time in the future. The messages will be stored in the "outbox" of your account until this time. Min value: current timestamp Max value: current timestamp + 31104000 (1 year) The value must be the Unix timestamp of the time the message must be send. This is the time measured in the number of seconds since the Unix Epoch (January 1 1970 00:00:00 GMT).

Example sms without delivery receipts:

```
http://www.cellcom.be/data/bulkHTTP.php?CLID=12345&GWID=123&PW=256ea978c8f8752eaf87e7a&TO=32496233133&SID=Cellcom&M=This+is+the+message
```

Example sms with delivery receipts:

```
http://www.cellcom.be/data/bulkHTTP.php?CLID=12345&GWID=123&PW=256ea978c8f8752eaf87e7a&TO=32496233133&SID=Cellcom&M=This+is+the+message&DR=1
```

Example wap push sms with delivery receipts:

```
http://www.cellcom.be/data/bulkHTTP.php?CLID=12345&GWID=123&PW=256ea978c8f8752eaf87e7a&TO=32496233133&SID=Cellcom&M=DC0601AE02056A0045C60C03777772e63656c6c636f6d2e62652f636c69656e745f6c6f67696e2e70687000010343656c6c636f6d204c6f67696e000101&COD=2&UDH=0605040b8423f0&DR=1
```

Result codes

When the message has been sent, the response will be:

OK MT_message_ID (for example: OK 785340486)

The MT_message_ID is the unique code of the message. You will need this to trace the message.

If the MID parameter was used during the request, the MT_message_ID will contain the same value.

If not OK, one of the following error-codes will be returned:

CODE	INFO
ERROR 10	Wrong client id or password
ERROR 20	Wrong gateway number (GWID)
ERROR 30	Not enough credit left
ERROR 40	Recipient number is wrong, or destination not covered
ERROR 41	Recipient number is on the accounts' blacklist (eg after stop / unsubscribe MO)
ERROR 50	Sender ID not supported
ERROR 60	Message is too long or invalid
ERROR 61	No binary coding found in M and UDH and COD=2
ERROR 62	In case of binary coding (COD=2), UDH too long (>14 HEX)
ERROR 63	In case of binary coding (COD=2), message (M) too long (>768 HEX)
ERROR 64	TTL value not valid
ERROR 65	DR value not valid
ERROR 66	Send Time (ST) not valid
ERROR 70	Timeout, please retry later (minimum 1 minute later)
ERROR 80	Internal error, please contact Cellcom
ERROR 9xx	Undefined error, please contact Cellcom mentioning the xx errorcode

Status report

If you wish to receive status reports of your messages, you will need to provide us a URL to invoke a HTTP GET request to. Add the DR=1 parameter to your URL when sending the messages.

Your script has to return the message *ok* or *OK* on receiving the delivery report. The response code has to be in the body of the response (Content-Type = "text/plain"). If our server didn't get this response, the script will try maximum 10 times with a 15 minutes pause between every attempt.

When we receive a status message, we will send three parameters to your URL:

PARAM.	NAME	INFO
messageid	<i>Unique Message ID</i>	The unique id of the message where the status report stands for. This is the value we indicate after the OK when sending a message. Optional: you can also define this value when using the optional MID parameter on sending the message.
reportstatus	<i>Status report</i>	The status of the report: 1 - Delivered 2 - Buffered 3 - Failed 5 - Expired 6 - Rejected 7 - Error 10 - Unknown (delivered to the network, but no more info) 11 - Unknown (no DLR received after timeout) 12 - Unknown (not at standard reaction) Other codes indicate an unknown status. If you don't receive a status for a message, it doesn't necessarily mean the message wasn't received. It can also indicate that the SMSC doesn't support status reports, or that the report was lost in the transaction to your server. (eg when your server was offline)
datetime	<i>Timestamp report</i>	The Unix timestamp of the status report.

Receiving a MO text message

When you have a dedicated virtual number or a keyword on a virtual number configured on your account, you will need to provide us a URL to invoke a HTTP GET request to.

Your script has to return the message *ok* or *OK* on receiving the text message. The response code has to be in the body of the response (Content-Type = "text/plain"). If our server didn't get this response, the script will try maximum 10 times with a 15 minutes pause between every attempt.

When we receive a MO text message, we will send these parameters to your URL:

PARAM.	NAME	INFO
TO	<i>Recipient</i>	Your dedicated or shared virtual number.
SID	<i>Sender ID</i>	The cellphone number of the subscriber sending you the inbound text message. Number is in international format, without + or 00 before the country prefix.
MID	<i>Message ID</i>	The unique message id of the text message.
KEY	<i>Keyword</i>	This is the keyword on with your account is subscribed in case you shared a virtual number. This is in fact always the first word of the message text. This value will always be in lower case letters, even if the message text has the keyword in a mix of capital and lower case letters.
TIME	<i>Timestamp</i>	The timestamp of the message like we receive it from the originated network. This is in UNIX timestamp format.
M	<i>Message</i>	The actual message. It's URL encoded.
COD	<i>Coding</i>	The data coding schema of the message. Default is 1 (Plain 7-bit). Flash Plain = 0 / Binary = 2 / Unicode = 4 / Flash Unicode = 5 / Plain GSM encoded = 6 / Flash Plain GSM encoded = 7
UDH	<i>User Data Header</i>	If the message is a binary message, the hex User Date Header will be displayed here.

Example

PHP-example to send a message with CURL:

```
$url =  
"http://www.cellcom.be/data/bulkHTTP.php?CLID=12345&GWID=123&PW=256e  
a978c8f8752eaf87e7a&TO=32496233133&SID=Cellcom&M=This+is+the+message  
";  
  
$ch = curl_init();  
curl_setopt($ch, CURLOPT_URL, $url);  
curl_setopt($ch, CURLOPT_SSL_VERIFYPEER, false);  
curl_setopt($ch, CURLOPT_RETURNTRANSFER, 1);  
$result = curl_exec($ch);  
curl_close($ch);
```

\$result will contain the result or error-code.

Note: use the php function urlencode and urldecode to encode and decode the message.