SMS REST API

Clickatell

Unlock Possibilities

Contents

1. Overview

This real-time SMS API is built to the very best practice architecture guidelines for RESTful API design. Ideal for developers anticipating a deeper and more extensive integration solution, you can send batches of up to 500 destination numbers per submission in real-time through our international SMS messaging gateway. Interact with Clickatell using JSON or XML.

Why use this API?

- Extremely reliable and convenient to use, with an intuitive, hierarchical, and consistent structure, supported by embedded resource links.
- Version control mechanisms to allow for future upgrades.
- Allows for the sending of multi-language Unicode message.
- Simplified token-based authentication on all API commands.
- Embedded resource links to our online technical documents.
- Schedule a message to be sent at a specific time, of up to 7 days ahead.
- Full two-way messaging capabilities.
- Delivery acknowledgment and Sender ID can be set up.
- Supports extended length messages.
- Provides the ability to receive real-time message status notifications to your system.

When to use this API

The REST API is ideal for improving customer service and delighting your customers by using information from your existing CRM system.

2. Using the SMS REST API

SMS messages are sent by making **HTTP** or **HTTPS** calls to the API. You can loop through your list of mobile numbers and make **HTTP/S** requests to send SMS messages to them. Generally, this API should allow you to submit up to about 100 messages per second to our platform on average (or faster if required).

It's quick to get started and there is no need to understand the additional or advanced features unless you want to use them.

- Supports **JSON** and **XML** data formats.
- Supports **HTTP** (port 80) and **HTTPS** (port 443).
- You can send to multiple handsets in one single **HTTP/S** request (up to 200) – this is useful if you are sending the same message text to all handsets.
- Supports persistent connections (keep-alive) and the use of multiple concurrent connections.
- Various HTTP status codes are used. Click here to view the list.

2.1 Send your first message

- **Step 1:** Replace the example authentication token below with your token.
- **Step 2:** Replace the example mobile numbers below with your number, using the correct international format. This means:
  - o The mobile number needs to start with the correct country code.
  - o There must be no spaces or other non-numeric characters.
  - o If the mobile number starts with zero, remove the zero before typing the rest of the number.
- **Step 3:** Type your message text. Please note that until you make a purchase, your messages will contain pre-defined content.

```
curl -X POST \
-H "X-Version: 1" \
-H "Content-Type: application/json" \
-H "Accept: application/json" \
-H "Authorization: Bearer YourAuthorizationTokenHere" \
-d '{"text":"My message","to":["2799900001","2799900002"]}' \
https://api.clickatell.com/rest/message
```

2.2 HTTP headers explained

Header	Description	Examples
Authorization	Required to authenticate your API requests. The authentication token for this header is displayed in your Developers' Central account.	Authorization: Bearer 4ZnXsl.LcQrDl4ZHWBt6_J1KLqNZVV7Tfg 9KK25nd1EYaQ7SPP2mmLkODKhzJ1S
Version	Specifies which version of the API to use.	X-Version:1
Content-Type	Used to indicate whether you are submitting data in JSON or XML format. Supports application/json and application/xml. There is currently no support for x-www-form-urlencoded.	Content-Type: application/json
Accept	Used to indicate if you want a JSON or XML response format. Defaults to JSON if not specified. If specified, the API expects application/json,application/xml or */*.	Accept: application/json

2.3 Security

For added security, you can opt to:

- Use **HTTPS** for all API requests.
- Enable IP lockdown on your API connection.

2.4 Authenticating

All authentication is done using an authentication token (generated within Developers' Central account). This token needs to be passed onto every REST API call. The header used for this is explained in the section called 'HTTP Headers Explained'.

3. Send message

This command allows you to send one or more SMS messages. To send messages to your database of mobile numbers, you can call this command in a loop.

The server will respond with a unique identifier for each message (referred to as an API message ID). This API message ID can be used to track and monitor the status of your message (see *Message status command* section). You can also send to multiple handsets in one single HTTP/S request (up to 600 messages). This is useful if you are sending the same message text to all handsets.

For high volume messaging, we encourage the use of persistent (keep-alive) **HTTP/S** connections. Multiple concurrent **HTTP/S** connections may also be used for additional performance.

```
POST https://api.clickatell.com/rest/message
```

Note: *the REST API may respond with various HTTP status codes. See the full list in the* Response status codes *section.*

3.1 JSON

Request

```
POST /rest/message HTTP/1.1
HOST: api.clickatell.com
X-Version: 1
Content-Type: application/json
Authorization: Bearer [Your Authorization Token]
Accept: application/json
{"text":"Test Message","to":["2799900001", "2799900002"]}
```

Response

```
HTTP/1.1 202 Accepted
Content-Type: application/json
{
  "data":{
    "message":[
      {
```

```
      "accepted":true,
      "to":"2799900001",
       "apiMessageId":"a55b8f8d56f33440e993aa614c68bf8b"
     },
      {
        "accepted":true,
        "to":"2799900002",
        "apiMessageId":"7f1d32762f6db11f3b7d2aaca2aaf362"
      }
    ]
  }
}
```

3.2 XML

Request

```
1  POST /rest/message HTTP/1.1
2  HOST: api.clickatell.com
3  X-Version: 1
4  Content-Type: application/xml
5  Authorization: Bearer [Your Authorization Token]
6  Accept: application/xml
7  <?xml version "1.0"?>
8  <request>
9      <data>
10         <text>Test Message</text>
11         <to>2799900001</to>
12     </data>
13 </request>
```

Response

```
1  HTTP/1.1 202 Accepted
2  Content-Type: application/xml
3  <?xml version="1.0"?>
4  <response>
5      <data>
6          <message>
7              <accepted>1</accepted>
8              <to>2799900001</to>
9              <apiMessageId>93f924cad9664cfbb0c6ab03a3032ef0</apiMessageId>
10         </message>
11     </data>
12 </response>
```

3.3 Sample code

cURL

```
to="[\"<mobile number>\"]"
message="Test Message"
authToken="<place auth token here>"

curl -X POST \
-H "X-Version: 1" \
-H "Content-Type: application/json" \
-H "Accept: application/json" \
-H "Authorization: Bearer $authToken" \
-d "{\"text\":\"$message\",\"to\":$to}" \
https://api.clickatell.com/rest/message
```

Python

```python
1  import httplib2, json
2
3  to=["mobile number"]
4  message="Test Message"
5  authToken = ""
6
7  resp, content = httplib2.Http().request(
8      "https://api.clickatell.com/rest/message",
9      "POST",
10     body=json.dumps({
11         "text":message,
12         "to":to
13     }),
14     headers={
15         "X-Version" : "1",
16         'Content-Type':'application/json',
17         "Accept" : "application/json",
18         "Authorization" : "Bearer " + authToken
19     }
20 )
```

PHP

```php
1  <?php
2
3  $to="[\"<mobile number>\"]";
4  $message="Test Message";
5  $authToken="<place auth token here>";
6
7  $ch = curl_init();
8
```

```
 9 curl_setopt($ch,CURLOPT_URL,"https://api.clickatell.com/rest/message");
10 curl_setopt($ch, CURLOPT_RETURNTRANSFER, 1);
11 curl_setopt($ch, CURLOPT_POST, 1);
12 curl_setopt($ch, CURLOPT_POSTFIELDS, "{\"text\":\"$message\",\"to\":$to}");
13 curl_setopt($ch, CURLOPT_HTTPHEADER, array(
14     "X-Version: 1",
15     "Content-Type: application/json",
16     "Accept: application/json",
17     "Authorization: Bearer $authToken"
18 ));
19
20 $result = curl_exec ($ch);
21 ?>
```

4. Stop message

This command will attempt to stop the delivery of a particular message. This command can only stop messages that have been scheduled for delivery in the future or may be queued within our router. It cannot stop messages that have already been delivered to an SMSC.

DELETE https://api.clickatell.com/rest/message/[message ID]

Note: *the REST API may respond with various HTTP status codes. See the full list in the* Response status codes *section.*

4.1 JSON

Request

```
DELETE /rest/message/360586640fe9846954897dffe6c13dfc HTTP/1.1
HOST: api.clickatell.com
X-Version: 1
Authorization: Bearer [Your Authorization Token]
Accept: application/json
```

Response

```
HTTP/1.1 200 OK
Content-Type: application/json

{
   "data" : {
       "messageStatus" : "006",
       "description" : "User cancelled message delivery",
        "apiMessageId" : "360586640fe9846954897dffe6c13dfc"
   }
}
```

4.2 XML

Request

```
1 DELETE /rest/message/360586640fe9846954897dffe6c13dfc HTTP/1.1
2 HOST: api.clickatell.com
3 X-Version: 1
4 Authorization: Bearer [Your Authorization Token]
5 Accept: application/xml
```

Response

```
1  HTTP/1.1 200 OK
2  Content-Type: application/xml
3
4  <?xml version="1.0"?>
5  <response>
6      <data>
7      <messageStatus>006</messageStatus>
8      <description>User cancelled message delivery</description>
9      <apiMessageId>b09e1b1bdb5a070c978f03b6836b955c</apiMessageId>
10     </data>
11 </response>
```

4.3 Sample code

cURL

```
msgId=""

authToken=""

curl -X DELETE \
-H "X-Version: 1" \
-H "Accept: application/json" \
-H "Authorization: Bearer $authToken" \

https://api.clickatell.com/rest/message/$msgId
```

Python

```
1  import httplib2, json
2
3  msgId = ""
4
5  authToken = ""
6
```

Clickatell
Unlock Possibilities

```
7  resp, content = httplib2.Http().request(
8      "https://api.clickatell.com/rest/message/" + msgId,
9      "DELETE",
10     headers={
11         "X-Version" : "1",
12         "Accept" : "application/json",
13         "Authorization" : "Bearer " + authToken
14     }
15 )
```

PHP

```
1  <?php
2
3  $msgId = "<api message ID>";
4
5  $authToken = "<place auth token here>";
6
7  $ch = curl_init();
8
9  curl_setopt($ch, CURLOPT_URL, "https://api.clickatell.com/rest/message/$msgId");
10 curl_setopt($ch, CURLOPT_RETURNTRANSFER, 1);
11 curl_setopt($ch, CURLOPT_CUSTOMREQUEST, "DELETE");
12 curl_setopt($ch, CURLOPT_HTTPHEADER, array(
13     "X-Version: 1",
14     "Accept: application/json",
15     "Authorization: Bearer $authToken"
16 ));
17
18 $result = curl_exec ($ch);
19
20 ?>
```

5. Retrieve balance

This will show the number of credits available on your account.

```
GET https://api.clickatell.com/rest/account/balance
```

Note: *the REST API may respond with various HTTP status codes. See the full list in the* Response status codes *section.*

Clickatell
Unlock Possibilities

5.1 JSON

Request

```
GET /rest/account/balance HTTP/1.1
HOST: api.clickatell.com
X-Version: 1
Authorization: Bearer [Your Authorization Token]
Accept: application/json
```

Response

```
HTTP/1.1 200 OK
Content-Type: application/json


{
 "data":{
   "balance":"300.50"
  }
}
```

5.2 XML

Request

```
GET /rest/account/balance HTTP/1.1
HOST: api.clickatell.com
X-Version: 1
Authorization: Bearer [Your Authorization Token]
Accept: application/xml
```

Response

```
1 HTTP/1.1 200 OK
2 Content-Type: application/xml
3
4 <?xml version="1.0"?>
5 <response>
6     <data>
7         <balance>300.50</balance>
8     </data>
9 </response>
```

5.3 Sample code

cURL

```
authToken=""

curl -X GET \
-H "X-Version: 1" \
-H "Accept: application/json" \
```

```
-H "Authorization: Bearer $authToken" \
https://api.clickatell.com/rest/account/balance
```

Python

```python
1  import httplib2, json
2
3  authToken = "place auth token here"
4
5  resp, content = httplib2.Http().request(
6      "https://api.clickatell.com/rest/account/balance",
7      "GET",
8      headers={
9          "X-Version" : "1",
10         "Accept" : "application/json",
11         "Authorization" : "Bearer " + authToken
12     }
13 )
```

PHP

```php
1  <?php
2
3  $authToken = "<place auth token here>";
4
5  $ch = curl_init();
6
7  curl_setopt($ch,CURLOPT_URL, "https://api.clickatell.com/rest/account/balance");
8  curl_setopt($ch, CURLOPT_RETURNTRANSFER, 1);
9  curl_setopt($ch, CURLOPT_HTTPHEADER,      array(
10     "X-Version: 1",
11     "Accept: application/json",
12     "Authorization: Bearer $authToken"
13 ));
14
15 $result = curl_exec ($ch);
16
17 ?>
```

6. Get coverage command

This command allows you to check our coverage of a network or mobile number without sending a message to that handset. This command should **NOT** be used before sending each message.

The returned credit charge value indicates the lowest possible credit charge at that point in time for a single plain-text message. Credit charges may fluctuate based on the availability of delivery routes and features, and there is no guarantee that all your messages will be charged this value.

```
GET https://api.clickatell.com/rest/coverage/[msisdn]
```

Note: *the REST API may respond with various HTTP status codes. See the full list here.*

6.1 JSON

Request

```
GET /rest/coverage/27999123456 HTTP/1.1
HOST: api.clickatell.com
X-Version: 1
Authorization: Bearer [Your Authorization Token]
Accept: application/json
```

Response

```
HTTP/1.1 200 OK
Content-Type: application/json

{
    "data": {
        "routable" : true,
        "destination" : "27999123456",
         "minimumCharge" : 0.8
    }
}
```

6.2 XML

Request

```
1 GET /rest/coverage/27999123456 HTTP/1.1
2 HOST: api.clickatell.com
3 X-Version: 1
4 Authorization: Bearer [Your Authorization Token]
5 Accept: application/xml
```

Response

```
1  HTTP/1.1 200 OK
2  Content-Type: application/xml
3
4  <?xml version="1.0"?>
5  <response>
6      <data>
7          <routable>1</routable>
8          <destination>27999123456</destination>
9          <minimumCharge>0.8</minimumCharge>
10     </data>
11 </response>
```

6.3 Sample code

cURL

```
number=""

authToken=""

curl -X GET \
-H "X-Version: 1" \
-H "Accept: application/json" \
-H "Authorization: Bearer $authToken" \

https://api.clickatell.com/rest/coverage/$number
```

Python

```
1  import httplib2, json
2
3  to = ""
4
5  authToken = ""
6
7  resp, content = httplib2.Http().request(
8      "https://api.clickatell.com/rest/coverage/" + to,
9      "GET",
10     headers={
11         "X-Version" : "1",
12         "Accept" : "application/json",
13         "Authorization" : "Bearer " + authToken
14     }
15 )
```

PHP

```
1  <?php
2
3  $mobileNumber = "<mobile number>";
4
5  $authToken = "<place auth token here>";
6
7  $ch = curl_init();
8
9  curl_setopt($ch,CURLOPT_URL,  "https://api.clickatell.com/rest/coverage/$mobileNumber");
10 curl_setopt($ch, CURLOPT_RETURNTRANSFER, 1);
11 curl_setopt($ch, CURLOPT_HTTPHEADER, array(
12     "X-Version: 1",
13     "Accept: application/json",
14     "Authorization: Bearer $authToken"
15 ));
16
17 $result = curl_exec ($ch);
18
19 ?>
```

7. Message status command

There are multiple ways to track your message delivery status and charges:

- View reports in your Developers' Central account (these can also be exported to a CSV file);
- Enable the Callback system, which will automatically push HTTP requests to your server whenever the status of a message changes. The HTTP request will contain information about the message status and charge. This is triggered in near real-time.
- Use the Query Message API command to query the status of a single message.

The Callback system is the most efficient (and preferred) option, as it does not require you to poll our system to track the progress of a message.

8. Overview of the REST API features

A variety of messaging and SMS features are supported by the gateway and can be activated by including additional parameters. These parameters include the ones stated in the table below:

Feature	Description	Values	Default	Parameter
Binary	Force the message content to be sent as binary (8-bit) data.	0, 1	0	binary
Client Message ID	Client message ID defined by user for message tracking.	2993b6b548	none	clientMessageId
Concatenation	Specifies the maximum number of message parts available for the message.	1, 2, 3	Defaults to setting on your API connection	maxMessageParts
Delivery Queue	Delivers the message through one of three queues assigned to each client account. Messages in the highest priority queue will be delivered first.	1, 2, 3	3	userPriorityQueue
Destination Address	The mobile number of the handset to which the message must be delivered (MSISDN). The number must be in international format with no leading zeros or + symbol.	27831234567 (South Africa) 16501234567 (USA) 44123456789 (UK)	none	to
Enable Callback	Enables you to receive message delivery status updates via an HTTP request made to your server.	0,1,2,3,4,5,6,7	0	callback
Gateway Escalation	Prompts an escalation to an alternative (more expensive) route, if messages are queued or delayed on the least-cost route.	0, 1	0	escalate

Maximum Credits	Overrides the maximum charge specified online in "routing profiles".	0.8,1,1.5,2,2.5,3	none	maxCredits
Message Text	The content of your SMS message.	Hello World		text
Message Type	Send other types of messages such as flash messages.	Refer to description		type
Mobile Originated	This is only applicable to clients that have subscribed to a two-way messaging service. We route via pre-defined carriers to enable the ability for a reply to be received back.	0, 1	0	mo
Required Features	Some parameters and features are not set as "required" by default and may be dropped if the least-cost route does not support them. Set this parameter to ensure that the features set when an SMS is sent are supported by the gateway used.	Refer to description	none	requiredFeatures
Scheduled Time	Specify when a message should get delivered in GMT+0.	2015-06-30T14:00:00Z	none	scheduledDeliveryTime
Sender ID	The source/sender address that the message will appear to come from (also known as the "Sender ID"). These must be registered within your online account and approved by us before they may be used. Two-way numbers rented from us do not require approval.		none	from
Unicode Message	Two-digit language code. Convert your text to Unicode [UCS-2 encoding]. See http://www.Unicode.org/.	0, 1	0	unicode
User Data Header	Set the UDH content of a message.			udh
Validity Period	The validity period in minutes relative to the time at which the SMS was received by our gateway. The message will not be delivered if it is still queued on our gateway after this time period.	25	24 hours	validityPeriod
Bypass DNC Check	Feature to bypass the DNC lookup. Default behavior is to do a lookup unless overridden by this	0, 1	none	bypassDNCCheck

	parameter. A value of "0" will still force a lookup; a value of "1" will force a lookup bypass.			

9. Response status codes

The REST API may respond with any of the following HTTP status codes:

Status code	Description	Detail
200	OK	The request was successfully completed.
202	Accepted	The message(s) will be processed.
207	Multi-status	More than one message was submitted to the API, however, not all messages have the same status.
400	Bad request	Validation failure (such as missing/invalid parameters or headers).
401	Unauthorized	Authentication failure. This can also be caused by IP lockdown settings.
402	Payment required	You do not have enough credits to send messages.
404	Not found	The action was performed on a resource that does not exist.
405	Method not allowed	HTTP method is not supported on the resource.
410	Gone	The mobile number is blocked, delisted, or cannot be routed.
429	Too many requests	Generic rate-limiting error.
503	Service unavailable	A temporary error has occurred on our platform – please retry.

10. API Error Codes

Error Code	Description	Detailed Description	Solution
001	Authentication failed	Authentication details are incorrect.	
007	IP lockdown violation	You have locked down the API instance to a specific IP address but attempted to send from an IP address different to the one you have set.	You can edit the setting to include your new server's IP address or remove the IP lockdown completely within Developers' Central's API settings.

Error Code	Description	Detailed Description	Solution
100	Data malformed	The JSON/XML data submitted is invalid.	Check that the syntax of your request has been formulated correctly and resubmit.
101	Invalid or missing parameters	One or more parameters are missing or invalid.	
102	Invalid user data header	The format of the user data header is incorrect.	Ensure valid UDH data is being passed to the API.
105	Invalid destination address	The destination address you are attempting to send to is invalid.	Check that the number of the handset that you are attempting to send a message to is valid. The number should be in an international format, without a '00' prefix or leading '+' symbol OR begin with a '0' if the default country prefix is enabled on your API.
106	Invalid source address	The specified sender address is incorrect.	The address that the message is sent 'from' has been specified incorrectly. If you are using a Sender ID as your source address, ensure that it has been registered within your online Developers' Central account.
108	Invalid or missing API ID	The API ID is either incorrect or has not been included in the API call.	Include the correct API product ID in your query. You can check the ID that is associated with your API by logging into your Developers' Central account.
109	Missing message ID	This may refer to either a client message ID or API message ID – for example, when using the 'stop message' command.	
113	Maximum message parts exceeded	The text component of the message is greater than the permitted 160 characters (70 Unicode characters). View the concatenation page for help in resolving this issue.	Set concat equal to 1,2,3-N to overcome this by splitting the content across multiple messages. View concatenation information.
114	Cannot route message	This implies that the gateway is not currently routing messages to this network prefix. Please email support@clickatell.com with the mobile number in question.	

Clickatell
Unlock Possibilities

Error Code	Description	Detailed Description	Solution
116	Invalid unicode data	The format of the Unicode data entered is incorrect.	Ensure that the Unicode format is correct and resubmit your query.
120	clientMessageId contains space(s)	Your specified client message ID contains a space. Space characters in client message IDs are not currently supported.	The delivery time must be entered in minutes up to a maximum of 7 days.
121	Destination mobile number blocked	This number is not allowed to receive messages from us and has been put on our blocked list.	
122	Destination mobile opted out	The user has opted out and is no longer subscribed to your service.	
123	Invalid Sender ID	The sender ID is not valid or has not been approved.	A sender ID needs to be registered and approved before it can be successfully used in message sending.
128	Number delisted	This number has been delisted and cannot receive our messages.	
130	Maximum MT limit exceeded until <UNIX TIMESTAMP>	This error is returned when an account has exceeded the maximum number of MT messages that can be sent daily or monthly. You can send messages again on the date indicated by the UNIX TIMESTAMP.	
160	HTTP method is not supported on this resource	An unsupported HTTP method has been performed on the resource. Example: HTTP POST on the Coverage resource.	The response MUST include a Content-Type header that contains a valid method for the requested resource.
161	Resource does not exist	You are attempting to access a REST API resource that does not exist.	
165	Invalid or no version header specified	The expected header that specifies version was either not found or is invalid.	Before continuing, make sure that the correct version is included in the submitted header. The header to use is X-Version: 1

Clickatell
Unlock Possibilities

Error Code	Description	Detailed Description	Solution
166	Invalid accept header specified	The optional header that specifies acceptable content does not contain an allowed value.	
167	Invalid or no content-type specified	The expected header that specifies content-type content was either not found or did not contain an allowed value.	Before continuing, make sure that an allowed value is included in the submitted content-type header. The allowable content-type header values are 'application/json' or application/xml'.
250	Destination address is on DNC list – Do not contact receiver	DNC lookup for the destination address result shows that the receiver should not be contacted.	Use the "bypass_dnc_check" (this name varies for APIs) request parameter to bypass the DNC check.
301	No credit left	Insufficient credits.	Log in to your Developers' Central account and purchase additional credits.
901	Internal error – please retry	An error occurred on our platforms.	Please retry submitting the message. This should be exceptionally rare.

Clickatell
Unlock Possibilities