Bredband2

SMS-API Advanced

Interface description

HTTP GET v1.0

Contents

Introduction.....3
Sending SMS.....4
Server Hostnames and Ports.....4
Certificate Information 4
Parameters.....4
Response codes.....6
Examples.....6
Sending an SMS.....6
Sending an SMS to multiple recipients.....6
Sending a FLASH SMS.....7
Sending an SMS with Alphanumeric address.....7
Sending an SMS with ÅÄÖ 7
Sending a Binary SMS.....7

Receiving Delivery Reports.....7
Example.....8
Receiving SMS.....9
Push.....9
Parameters.....9
Example.....10
Pull.....10
Obtaining a list of waiting messages.....10
Retrieving a single message.....11
Retrieving next message.....12
Result codes.....13
Response codes.....13

Introduction

This document describes the Bredband2 SMS-API Advanced (from now on referred to as 'Interface').
It describes the available features, and provides instructions on how to use them.

This document is intended for application developers and/or integrators. Basic knowledge of the HTTP protocol is assumed.

Sending SMS

Bredband2 SMS-API Advanced
provides a HTTP GET interface for sending outgoing SMS. The interface supports regular HTTP as well as HTTPS. It is strongly recommended to use HTTPS to secure the communication. If desired, GotaSMS can also us e SSL-encryption when sending delivery reports back to the client.

Server Hostnames and Ports

This table defines which hostname and port to use for a selected service when submitting the GET-request. Please note that this information may be subject to change and that different information may have been provi ded along with the username and password.

Service	Hostname	Port	Path
HTTP GET	smsgateway.gotanet.se	40080	/submit
HTTP GET using SSL	smsgateway.gotanet.se	40443	/submit

Certificate Information

Bredband2 SMS-API Advanced uses a certificate issued by *Geotrust, Inc.* for authentication.
Unless not already done, the Geotrust, Inc.
root certificate must be installed on the client system. For instructions on how this is done, please refer to www.geotrust.com.

Parameters

The table below lists available parameters of the Interface. Some parameters are mandatory, while others can be left out to use their default values. If mandatory parameters are left out, the request will be rejected.

All parameters must be URI encoded1.

Parameter	Mandatory	Description	Example
username	Yes	Username of client.	username=myuser
password	Yes	Password of client.	password=mypassword
to	Yes	Recipient of the message, specified as MSISDN number in International format. Multiple recipients can be specified, separated by semi-colon.	To=46701234567 to=46701234567;46707654321

1 Uniform Resource Identifiers (URI): Generic Syntax, RFC2396

Parameter	Mandatory	Description	Example
from	Yes	Sender of message. Can be an International number (max 15 digits), Alphanumeric (max 11 characters) or a Short number (up to 6 digits). Only characters 0-9 and '+' is allowed for International and Short numbers. Alphanumeric addresses also support A-Z, a-z, ! #%&*().	from=46701234567 from=MyCompanyAB
message	Yes	The actual message. If message type is binary this parameter should be given in hexadecimal form.	message=Hello+world
type	No	Message type can be 'text', 'flash' or 'binary'. Defaults to 'text'.	type=flash
charset	No	Specifies the character set for the message of the request. Defaults to 'ISO-8859-1'. See appendix A for supported character sets.	charset=ISO-8859-1
validity-period	No	Number of minutes that message is valid. Defaults to 10080 minutes (7 days). Maximum value is 20160 minutes.	validity-period=1440
delivery-time	No	The preferred time when message delivery should first be attempted. The parameter is given in absolute time, UTC.	delivery-time=2014-10-07+12%3a00%3a00
user-data-header	No	User Data Header, given in hexadecimal form. For advanced use.	user-data-header=050003FF0201
data-coding-scheme	No	Data Coding Scheme, used when message type is binary. Acceptable values are 0-255.	data-coding-scheme=0

Response codes

This is not an exhaustive list of response codes, but the ones most likely to be returned.

Response	Description
202 Accepted	Message has been accepted for delivery. A unique transaction id is given in the response body. See examples below.
400 Bad Request	The request is malformed.
403 Forbidden	Invalid credentials.
409 Conflict	Rate limit exceeded.

Examples

- **Sending an SMS**

```
http://smsgateway.gotanet.se:40080/submit?
username=myuser&password=mypassword&t
o=46701234567&from=46701234567&message=Hello+world
```

If everything is correct the response will contain a message similar to:

```
Accepted for delivery, eda0ae82-98d7-40b3-bbaa-205b68b63c0b
```

- Sending an SMS to multiple recipients

```
http://smsgateway.gotanet.se:40080/submit?username=myuser&password=mypasswo
rd&t
o=46701234567;46776543210&from=46701234567&message=Hello+world
```

If everything is correct the response will contain a message similar to:

```
Accepted for delivery, eda0ae82-98d7-40b3-bbaa-205b68b63c0b;7e513c4e-24d1-434e-
a580-71c7c418a4cc
```

- Sending a FLASH SMS

  The message is a flash SMS containg the message “Hello World”.

  ```
  http://smsgateway.gotanet.se:40080/submit?username=myuser&password=mypassword&to=46701234567&from=46701234567&message=Hello+world&type=flash
  ```

  Sending an SMS with Alphanumeric address

- The sending address for this message is “Test”.

  ```
  http://smsgateway.gotanet.se:40080/submit?username=myuser&password=mypassword&to=46701234567&from=Test&message=Hello+world
  ```

  Sending an SMS WITH ÅÄÖ

  The message contains ”ääöÅÄÖ”.

- http://smsgateway.gotanet.se:40080/submit?username=myuser&password=mypassword&to=46701234567&from=46701234567&message=%E5%E4%F6%C5%C4%D6

  Sending a Binary SMS

- The binary message contains ”Hello world”.

  ```
  http://smsgateway.gotanet.se:40080/submit?username=myuser&password=mypassword&to=46701234567&from=46701234567&type=binary&message=48656c6c6f20776f726c64&user-data-header=050003C80101&data-coding-scheme=4
  ```

Receiving Delivery Reports

Bredband2 SMS-API Advanced supports sending of delivery reports to the customer.
In order to receive delivery reports, the customer account must be configured with a delivery report URL.

When configured, delivery reports will be forwarded, as soon as they are received by Bredband2 SMS-API Advanced. The delivery report is sent using a GET-request.

Information needed to enable Delivery Reports:
- URL (hostname, port, path) to a customer system that will accept the reports through HTTP GET
- Optional username and password for basic authentication
- Optional certificate for HTTPS/SSL connection

In addition, the customer system must be prepared to accept connections to the given hostname and port.
It is recommended to use HTTPS to secure the transmission.7

The report contains the following headers:

Header	Description
X-result	The delivery outcome.
X-Transaction-Id	Transaction Id corresponding to the id received when the message was sent.

Possible values for X-Result:

X-result	Description
Delivered	Message has been delivered to the requested destination.
Undeliverable	Message has not been delivered. Common causes for this problem are phone might be switched off or out of network.
Expired	Message could not be delivered within the specified validity period.

The GET-request must be responded to by the customer, using a 200-response. Please refer to the example section below for an example of how this can be done.

Example

The message below is an example of a GET-request sent from Bredband2 SMS-API Advanced, containing the report for transaction id bc757d8a-a0db-4e0d-a388-bf6d6eee0d87, which was delivered successfully.

```
GET /delivery_report.php HTTP/1.1
Host: 1.2.3.4:55520
Connection: keep-alive
User-Agent: Leissner-SMS-Gateway/svn
X-Transaction-Id: bc757d8a-a0db-4e0d-a388-bf6d6eee0d87
X-Result: Delivered
```

The message below is an example of a valid response to a delivery report:

```
HTTP/1.1 200 OK
Connection: close
```

Receiving SMS

Using Bredband2 SMS-API Advanced it is possible to receive MO-messages to virtual numbers. SMS can be received in two ways, push or pull. If using push, MO-messages will be sent to the client when available. If using pull, the client will poll Bredband2 SMS-API Advanced for received messages at their own discretion.
When a message has been delivered to the client, either by push or pull, it will be deleted from Bredband2 SMS-API Advanced.

Push

If configured, Bredband2 SMS-API Advanced will initiate HTTP GET requests to the client whenever a new MO-message arrives. Information needed to enable push:

- URL (hostname, port, path) to a customer system that will accept the messages through HTTP GET
- Optional username and password for basic authentication
- Optional certificate for HTTPS/SSL connection

In addition, the customer system must be prepared to accept connections to the given hostname and port.

It is recommended to use HTTPS to secure the transmission.

The GET request has a format similar to the MT request message. The GET-

request must be responded to by the
customer using a 200-response. Please refer to the example section below for an example of how this can be done.

If the HTTP GET request fails for some reason it will be rescheduled to be re-

sent at a later time. A request can fail
if the customer's HTTP server is unavailable or if the request was not replied to with a 200-response.
Re-sends are scheduled as follows:

- Every 10 seconds during the first minute
- Every 60 seconds during the first hour
- Every 15 minutes during the first 24 hours
- Every second hour after 24 hours

No more re-sends will be performed if the request has still not succeeded after a week has passed.

- Parameters

The table below lists available parameters for MO-messages. Parameters marked as 'Mandatory' will always be present in the request. Optional parameters may be left out of the request.

All parameters are URI encoded.

Parameter	Mandatory	Description	Example
to	Yes	Recipient of the message, specified as MSISDN number in International format.	To=46701234567
from	Yes	Sender of message.	from=46701234567

Parameter	Mandatory	Description	Example
message	Yes	The actual message.	message=Hello+world
type	Yes	Message type can be ‘text’, ‘flash’ or ‘binary’.	type=flash
user-data-header	No	User Data Header, given in hexadecimal form. For advanced use.	user-data-header=050003FF0201
data-coding-scheme	No	Data Coding Scheme, used when message type is binary. Acceptable values are 0-255.	data-coding-scheme=0

- Example The message below is an example of a GET-request sent from GotaSMS, containing an incoming MO SMS:

```
GET /incoming_sms.cgi HTTP/1.1URI: /incoming_sms.cgi?

from=Test&to=46701234567&message=Test&type=text&charset=

ISO-8859-1
Host: 1.2.3.4:55520
Connection: keep-alive
User-Agent: Leissner-SMS-Gateway/svn
X-Transaction-Id: bc757d8a-a0db-4e0d-a388-bf6d6eee0d87
```

The message below is an example of a valid response:

```
HTTP/1.1 200 OK
Connection: close
```

Pull

As an alternative to the push method the client can pull waiting MO-messages from GotaSMS at any time.

- Obtaining a list of waiting messages

To obtain a list of waiting messages construct a GET request to this path:

cgi-bin/getmessagelist

The table below specifies the parameters for the request:

Parameter	Mandatory	Description	Example
username	Yes	Username of client.	username=myuser
password	Yes	Password of client.	password=mypassword

The body of the response from GotaSMS will contain a parameter specifying the number of messages waiting a nd zero or more transaction ids.

Examples of possible response bodies:

Response	Description
{"Result": 1, "Messages": [{"TransactionId": "aa313168-c9f9-4b25-aa14-38dadfb8ef8e"}, {"TransactionId": "56ef83b9-a535-46ec-8511-bcb30370f51b"}]}	Two messages (with transaction id aa313168-c9f9-4b25-aa14-38dadfb8ef8e and 56ef83b9-a535-46ec-8511-bcb30370f51b) are waiting.
{"Result": 1, "Messages": []}	No messages are waiting.

- Retrieving a single message

A single message can then be retreived using a specific transaction id together with a GET request to this path:

```
cgi-bin/getmessageforid
```

The table below specifies the parameters for the request:

Parameter	Mandatory	Description	Example
username	Yes	Username of client.	username=myuser
password	Yes	Password of client	password=mypassword
transaction-id	Yes	Transaction id of message to b e retrieved.	transaction-id=56ef83b9-a535-46ec-8511-bcb30370f51b

Example of possible response bodies:

Response	Description
{"Result": 1, "From": "Test", "FromType": "A", "To": "+46701234567", "ToType": "I", "Msg": "Test", "MsgType": 0, "CharSet": "ISO-8859-1", "Timestamp": "2014-10-14 08:46:09"}	A message from "Test" to number "4671234567" containing the message "Test". Timestamp is given in UTC.
{"Result": 2}	The given transaction id is unknown.

- Retrieving next message

The client can also use a single request to directly retreive a message, if there are any available. With this request the oldest message will be returned.

cgi-bin/getnextmessage

The table below specifies the parameters for the request:

Parameter	Mandatory	Description	Example
username	Yes	Username of client.	username=myuser
password	Yes	Password of client	password=mypassword

Example of possible response bodies:

Response	Description
{"Result": 1, "From": "Test", "FromType": "A", "To": "+46701234567", "ToType": "I", "Msg": "Test", "MsgType": 0, "CharSet": "ISO-8859-1", "Timestamp": "2014-10-14 08:46:09"}	A message from "Test" to number "+4671234567" containing the message "Test". Timestamp is given in UTC.
{"Result": 3}	No more messages are waiting.
{"Result": 4}	An internal error occurred.

- Result codes

The following result codes are used in getmessagelist, getmessageforid and getnextmessage.

Response	Description
1	No error.
2	The given transaction id is unknow
3	n. No more messages are waiting.
4	An internal error occurred.

- Response codes

This is not an exhaustive list of response codes, but the ones most likely to be returned.

Response	Description
202 Accepted	Message has been accepted for delivery. A unique transaction id is given i n the response body. See examples below.
400 Bad Request	The request is malformed.
403 Forbidden	Invalid credentials.
409 Conflict	Rate limit exceeded.