**HTTP API**

**1. Overview**

This technical document is intended for developers who wish to use the HTTP API for sending messages, and describes the various programming methods and commands used by developers when using this API.

The HTTP API is the most popular API, because there are many ways to utilize it for message sending and it can be used for low or high-volume messaging. As HTTP is a means for relaying information, the HTTP API can be used with practically any web-service application. This is particularly useful for high-volume message sending.

**2. Introduction**

This is one of the simpler server-based forms of communication to our gateway. It can be used either in the form of a HTTP POST, or as a URL (GET). We recommend POST for larger data transfer, due to the size limitations of GET. Communication to our API can be done via HTTP on port 80. All calls to the API must be URL-encoded. The parameter names are case-sensitive. Batch messaging is catered for in a variety of ways.

**3. Getting Started**

The following basic information will help you get started with using the interface. The message text should be URL Encoded. The message should be URL Encoded (also known as percent encoding) string of UTF-8 characters.

For more information on URL encoding, please see this:

[http://www.w3schools.com/tags/ref\_urlencode.asp.](http://www.w3schools.com/tags/ref_urlencode.asp)

Original text:

Hi Amar!

Happy Diwali to you

Regards,

Encoded text:

Hi%20Amar%21%0AHappy%20Diwali%20to%20you%0ARegards%2C

**4. Basic commands**

In order to send a message, the system will firstly need to authenticate you as a valid user. The preferred method of authentication is using username and password.

All other commands are then made up of three segments: authentication, the basic message components (message content and recipients) and the additional message parameters. In the examples below, we will include the authentication and basic message components. The additional message parameters will be included only where they are relevant.

Basic Command Structure: URL Call Authentication Message Parameters

**http://domain/fe/api/v1/send?username={username}&password={password}&unicode={true/false}&from={from}&to={to}&text={text}&templateId={templateId}**

**5. Sending Message**

User can send a text message using the API. To send a message to a mobile number using API, the following will be the request using HTTP GET or POST. The parameters can be sent in any order.

**URL Syntax**

URL Parameters



 **http://domain/fe/api/v1/send?username={username}&password={password}&unicode={true/false}&from ={from}&to={to}&text={text}&templateId={templateId}**

Request Parameters:

|  |  |  |  |
| --- | --- | --- | --- |
| **Parameter** | **Value** | **Required?** | **Description** |
| **Username** | API username | YES | Your API username |
| **Password** | API password | YES | Your API password. |
| **Unicode** | true/ false | YES | False for English message & true for Unicode messages |
| **From** | 6 alpha characters | YES | A Sender ID to appear on the recipients mobile (Cannot contain any special characters). |
| **To** | Recipient number with country code(91 for India) | YES | Recipient mobile number with country code eg:919949345690 |
| **Text** | Text message | YES | Message to recipients mobile (URL Encoded) |
| **templateId** | approved template id | NO | For Transactional messages this parameter is required (this parameter not required for promotional messages) |

**URL Syntax with DLT Content ID**

**http://domain/fe/api/v1/send?username={username}&password={password}&unicode={true/false}&from ={from}&to={to}&text={text}&templateId={templateId}&dltContentId=DLTID**

|  |  |  |  |
| --- | --- | --- | --- |
| **Parameter** | **Value** | **Required?** | **Description** |
| **Username** | API username | YES | Your API username |
| **Password** | API password | YES | Your API password. |
| **Unicode** | true/ false | YES | False for English message & true for Unicode messages |
| **From** | 6 alpha characters | YES | A Sender ID to appear on the recipients mobile (Cannot contain any special characters). |
| **To** | Recipient number with country code(91 for India) | YES | Recipient mobile number with country code eg:919949345690 |
| **Text** | Text message | YES | Message to recipients mobile (URL Encoded) |
| **templateId** | approved template id | NO | For Transactional messages this parameter is required (this parameter not required for promotional messages) |
| **DLTContent ID** | Approved Content id | YES |   |

**6. HTTP API Request & Response Details:**

When the request is successful, the following message is displayed.

**{"transactionId":27547042,"state":"SUBMIT\_ACCEPTED","description":"Message submitted successfully","pdu":1}**

We suggest you store the response text as it may be required to retrieve status / delivery report in future using message id.

Other responses in case of an error:

|  |  |
| --- | --- |
| **Response** | **Description** |
| **INVALID\_SOURCE\_ADDRESS** | Sender ID length != 6 or contains any special characters. |
| **Authentication failure** | Invalid Username/Password/Account expired |
| **status":500,"error":"Internal Server Error"** | Some parameters for request are missing. |
| **"6001:Insufficient Balance !"** | Zero Credit |
| **"5001:Template ID is missing"** | Template ID missing |

**7: Call Back API Format & Parameter Details for Real Time Delivery Report:**

Client has to provide call back api (GET) in below format to configure at the backend, real time delivery reports shall be pushed through the API.

**http://{clienturl path}?txid={transactionId}&to={recpient}&from={sender}&description={description}&pdu={totalPdu}&text={message}&deliverystatus={status}&deliverydt={doneDate}**

**Call Back API Parameters Description:**



**XML API Document**

XML One 2 One

curl -X POST --header 'Content-Type: application/xml' --header 'Accept: application/xml' -d '<?xml version="1.0"?> \

 <OneToOnePush> \

 <credentials> \

 <password>password</password> \

 <user>username</user> \

 </credentials> \

 <from>SENDER</from> \

 <options> \

     <templateId>Template ID</templateId> \

 </options> \

 <shortMessages> \

 <message>This is test message</message> \

 <recipient>9999999999</recipient> \

 </shortMessages> \

 <unicode>true</unicode> \

 </OneToOnePush> \

 ' 'https://domain/fe/api/v1/one2One'

Please find the below description of the parameters.

1. **Password**

 <password>password</password>

 Password will be available over the panel in the HTTP page.

1. **Username**

<user>username</user>

Username will be available over the panel in the HTTP page.

1. **Sender ID**

 <from>SENDER</from>

Is the Sender ID from which message needs to be originated. It should Alphabet and of 6 characters only.

1. **Message Text**

<message>This is test message</message>

This is the message text which will be delivered to the end user.

1. **Mobile Number**

 <recipient>9999999999</recipient>

Enter the mobile number on which the content to be delivered.

1. **Unicode**

 <unicode>true</unicode>

Unicode value to be true in case of vernacular message or in any regional language, in case of English content it should be false.

**Sample XML Request**

<?xml version="1.0"?>

<OneToOnePush>

 <credentials>

 <password>password</password>

 <user>username</user>

 </credentials>

 <from>test</from>

 <options>

 <templateId>all</templateId>

 </options>

 <shortMessages>

 <message><![CDATA[message text]]></message>

 <recipient>9999999999</recipient>

 <corelationId>sssssssssssssssssssss</corelationId>

 </shortMessages>

 <shortMessages>

 <message>test1</message>

 <recipient>8888888888</recipient>

 <corelationId>aaaaaaaaaaa</corelationId>

 </shortMessages>

 <unicode>false</unicode>

 </OneToOnePush>

**Response**

<PushResponse>

 <submitResponses>

 <submitResponses>

 <transactionId>999924278</transactionId>

 <state>SUBMIT\_ACCEPTED</state>

 <description>Message submitted successfully</description>

 <pdu>1</pdu>

 <corelationId>sssssssssssssssssssss</corelationId>

 </submitResponses>

 <submitResponses>

 <transactionId>999924279</transactionId>

 <state>SUBMIT\_ACCEPTED</state>

 <description>Message submitted successfully</description>

 <pdu>1</pdu>

 <corelationId>aaaaaaaaaaa</corelationId>

 </submitResponses>

 </submitResponses>

</PushResponse>

XML One 2 Many

curl -X POST --header 'Content-Type: application/xml' --header 'Accept: application/xml' -d '<?xml version="1.0"?>

 <OneToManyPush>

   <credentials>

        <password>Password</password>

        <user>username</user>

    </credentials>

    <from>Sender ID</from>

    <messageText>text message</messageText>

    <options>

        <a>b</a>

        <templateId>Template ID</templateId>

    </options>

    <recpients>

        <corelationId>ID</corelationId>

        <mobile>Mobile Number 1</mobile>

    </recpients>

    <recpients>

        <corelationId>ID</corelationId>

        <mobile>Mobile Number 2</mobile>

    </recpients>

    <recpients>

        <corelationId>ID</corelationId>

        <mobile>Mobile Number 3</mobile>

    </recpients>

    <unicode>true</unicode>

</OneToManyPush>

 ' 'https://domain/fe/api/v1/one2One'

Please find the below description of the parameters.

1. **Password**

 <password>password</password>

 Password will be available over the panel in the HTTP page.

1. **Username**

<user>username</user>

Username will be available over the panel in the HTTP page. For Promotional user name use the demo.pro and for transactional messages use demo.trans

1. **Sender ID**

 <from>SENDER</from>

Is the Sender ID from which message needs to be originated. It should Alphabet and of 6 characters only.

1. **Message Text**

<message>This is test message</message>

This is the message text which will be delivered to the end user.

1. **Mobile Number**

 <recipient>9999999990</recipient>

 <recipient>9999999991</recipient>

 <recipient>9999999992</recipient>

Enter the mobile number on which the content to be delivered. In case of multiple mobile numbers enter mobile numbers in new row.

1. **Unicode**

 <unicode>true</unicode>

Unicode value to be true in case of vernacular message or in any regional language, in case of English content it should be false.

**Sample XML Request**

<?xml version="1.0"?>

<OneToManyPush>

 <credentials>

 <password><password></password>

 <user><username></user>

 </credentials>

 <from><Sender ID></from>

 <message><Registered></message>

 <options>

<templateId>all</templateId>

 </options>

 <recpients>

<corelationId>hgfg</corelationId>

<mobile>1111111111</mobile>

 </recpients>

 <recpients>

<corelationId>hgfg</corelationId>

<mobile>2222222222</mobile>

 </recpients>

<unicode>false</unicode>

</OneToManyPush>

**Response**

<PushResponse>

<submitResponses>

<submitResponses>

<transactionId>863147912</transactionId>

<state>SUBMIT\_FAILED</state>

<description>4011:Number found in DnD bases</description>

<pdu>0</pdu>

<corelationId>test3</corelationId>

</submitResponses>

<submitResponses>

<transactionId>863147911</transactionId>

<state>SUBMIT\_FAILED</state>

<description>4031:Number found in user blacklist</description>

<pdu>0</pdu>

<corelationId>test2</corelationId>

</submitResponses>

<submitResponses>

<transactionId>863147910</transactionId>

<state>SUBMIT\_ACCEPTED</state>

<description>Message submitted successfully</description>

<pdu>1</pdu>

<corelationId>test1</corelationId>

</submitResponses>

<submitResponses>

<transactionId>863147913</transactionId>

<state>INVALID\_MSISDN</state>

<description>4005:Invalid Msisdn [4747872727] for country [IN]</description>

<pdu>1</pdu>

<corelationId>test4</corelationId>

</submitResponses>

<submitResponses>

<transactionId>863147914</transactionId>

<state>INVALID\_MSISDN</state>

<description>4005:Invalid Msisdn [124774774] for country [IN]</description>

<pdu>1</pdu>

<corelationId>test5</corelationId>

</submitResponses>

</submitResponses>

</PushResponse>

XML One 2 One API With DLT Content ID

curl -X POST --header 'Content-Type: application/xml' --header 'Accept: application/xml' -d '<?xml version="1.0"?> \

 <OneToOnePush> \

 <credentials> \

 <password>password</password> \

 <user>username</user> \

 </credentials> \

 <from>SENDER</from> \

 <options> \

     <templateId>Template ID</templateId> \

 </options> \

 <shortMessages> \

 <message>This is test message</message> \

 <options> \

<dltPrincipalEntityId>2222222222222</dltPrincipalEntityId> \

<dltContentId>1234</dltContentId> \

</options> \

 <recipient>9999999999</recipient> \

 </shortMessages> \

 <unicode>true</unicode> \

 </OneToOnePush> \

 ' 'https://domain/fe/api/v1/one2One'

XML One 2 Many with DLT Content ID

curl -X POST --header 'Content-Type: application/xml' --header 'Accept: application/xml' -d '<?xml version="1.0"?>

 <OneToManyPush>

   <credentials>

        <password>Password</password>

        <user>username</user>

    </credentials>

    <from>Sender ID</from>

    <messageText>text message</messageText>

    <options>

        <a>b</a>

        <templateId>Template ID</templateId>

    </options>

 <options>

 <dltPrincipalEntityId>2222222222222</dltPrincipalEntityId>

 <dltContentId>1234</dltContentId>

 </options>

    <recpients>

        <corelationId>ID</corelationId>

        <mobile>Mobile Number 1</mobile>

    </recpients>

    <recpients>

        <corelationId>ID</corelationId>

        <mobile>Mobile Number 2</mobile>

    </recpients>

    <recpients>

        <corelationId>ID</corelationId>

        <mobile>Mobile Number 3</mobile>

    </recpients>

    <unicode>true</unicode>

</OneToManyPush>

 ' 'https://domain/fe/api/v1/one2One'