HTTP API

Contents

Overview

This guide provides specifications of the HTTP/HTTPS based API provided by ELION Technologies for the automated sending of SMS via the Internet. This guide is intended for developers planning on integrating their systems with ELION's SMS service. It also provides the information about sending the Unicode, Binary and WAP messages.

Assumptions

This guide assumes that you are familiar with and have experience applying the following technologies and concepts:

HTTP/HTTPS communications with the GET and POST methods for parameter through API

A programming language such as Java, C/C++, Visual Basic or PHP to integrate the HTTP URL with the Application

Application designed in any of the above Language should be capable of capturing the Message ID (MID)

Required Environment

In order to use the HTTP/HTTPS based API, you will need to have a system that has internet connectivity on outbound port 80 (HTTP) or 443 (HTTPS). This connectivity may be direct to the internet, or may be via a firewall or proxy server.

Additionally, should you wish to receive delivery reports (DLRs) from the Elion system, you must have a web server that is able to receive incoming requests from the Internet on either port 80 (HTTP) or 443 (HTTPS). This web server does not necessarily have to be running on the same machine that is sending messages to Elion.

This is one of the simpler server-based forms of communication. It can be used either in the form of a HTTP POST or HTTP GET. We recommend POST for larger data transfer and data security. All calls to the API must be URL-encoded. The parameter names are case sensitive.

ELION API Parameter specification

Whenever the Client Application hits the HTTP URL, it should include the following parameters as per the requirement. The following parameters should be used in the same order as given below to call HTTP API using POST or GET method.

Parameters supported by the API call include:

NAME	PARAMETER	DESCRIPTION	REQUIRED VALUE	DEFAULT TYPE	TYPE
Username	uname	Username assigned to account	Username		Required
Password	pass	Password assigned to the account	Password		Required
SenderID	send	Source address for the msg	Sender ID as required		Required
Destination Number	dest	Destination Address/mobile Number of the message	Destination Mobile No.		Required
Message	msg	Text content of the msg (length should not cross 459 characters if concatenation is on, or 160 characters for 8bit msg; 280 characters for binary and 70 characters for Unicode	Body of the msg		Required
Priority	prty	The SMSC will process it based on the priority value. If this value is not set in the URL then our application will take the default value set at the account level	1,2,3,4,5	Account specific	Optional
Validity Period	vp	Validity period to expire the msg in the set time which can be set from 1-180 min	1-180 min	180	Optional
Concatenation	concat	Enables sending multipart msgs to the mobile	0=for limiting 0 length to 160 characters 1=for concatenation		Optional
International	intl	Enable sending the International Messages	0=OFF 1=ON	0	Optional

NAME	PARAMETER	DESCRIPTION	REQUIRED VALUE	DEFAULT TYPE	TYPE
User Data Header	udhi	Indicates that the msg payload is binary	0=normal text msg 1=binary	0	Required for Unicode
Data Coding Scheme	dcs	Indicates the DCS value for the msg: 0 indicates that msg payload is 8bit	0=text, 4=binary, 8=Unicode, 16=Flash, 245=binary including ringtone or picture	0	Required for Unicode
Port	port	Port number to send the port specific messages	Port Number based on Handset	No Value	Optional
WAP URL	wapurl	Set the value for Wap URL	URL	No Value	Required
WAP TEXT	waptxt	Set the value for WAP Text	Text message	No Value	Required

Sample HTTP URL Format

The below HTTP API can be used by Client to send the messages to ELION's Server. A sample of the URL could be in the below format:

SAMPLE FORMAT FOR HTTP:

http://103.250.30.4/SendSMS/sendmsg.php?uname=XXX&pass=XXX&send=91XXXXXXXXX&dest=91XXXXXXXXX&msg=XXXX

SAMPLE API Call made by Client Application:

http://103.250.30.4/SendSMS/sendmsg.php?uname=Username&pass=password&send=919980524000&dest=919243201476&msg=Testing

Successful Transmission of SMS

For each successful submission, the API would return a unique message ID (MID) for that transaction.
The Client's Application should capture the MID and if a Delivery Report (DLR) can be fetched against that MID.

This message ID then can be used to track the status of the message and receive the DLR against this MID through DLR URL.

Error Codes

The following error codes may be displayed while hitting the HTTP API if there is any wrong parameter entered or issue with the Account:

Error Code	Description
0x200	Invalid Username or Password
0x201	Account suspended due to one of several defined reasons
0x202	Invalid Source Address/Sender ID. As per GSM standard, the sender ID should be within 11 characters
0x203	Message length exceeded (more than 160 characters) if concat is set to 0 Message
0x204	length exceeded (more than 459 characters) in concat is set to 1
0x205	DRL URL is not set
0x206	Only the subscribed service type can be accessed - make sure of the service type you are trying to connect with
0x207	Invalid Source IP - kindly check if the IP is responding
0x208	Account deactivated/expired
0x209	Invalid message length (less than 160 characters) if concat is set to 1
0x210	Invalid Parameter values
0x211	Invalid Message Length (more than 280 characters)
0x212	Invalid Message Length
0x213	Invalid Destination Number

Sending normal Text Message (160 characters):

While sending normal 160 character messages Client needs to ensure that any optional parameters are not used (except for dlr-url to receive delivery acknowledgment on to predefined URL). Following is the sample API call that will set the message priority for level 3 and Validity Period to 30 Minutes.

SAMPLE FORMAT:

*http://*103.250.30.4*/SendSMS/sendmsg.php?uname=XYZ&pass=ABC&send=Elion&d*

Sa*mple* Response on XML: *s=*

Sending normal Text Message (more than 160 characters):

The HTTP API support SAR (Segmentation and Reassembly - often called as concatenated SMS) of the message. The API will support maximum of 9 segmentations with max character limit of 1224. Should you need to send concatenated SMS, you need to set 'CONCAT' parameter value as 1. Following is sample API call:

SAMPLE FORMAT:

http://103.250.30.4/SendSMS/sendmsg.php?uname=XXX&pass=XXX&send=XXXXXXX X&dest=91XXXXXXXXX&msg=(Message of more than 160 characters)&concat=1

Sending normal Text Message (more than 160 characters):

Elion HTTP API has end-to-end support for Unicode characters. Should you need to send a Unicode message, you need to set DCS value to 8 while the message text length should not exceed 70 characters. Following is a sample call:

SAMPLE FORMAT:

http://103.250.30.4/SendSMS/sendmsg.php?uname=XXX&pass=XXX&send=XXXXXXX X&dest=91XXXXXXXXX&msg=UnicodeString&udhi=1&dcs=8

When the UDHI value is set to 1 in API, it is required to tack the header before the message contents. If a single part Binary or Unicode message is sent, there is no need to set the udhi value and attach the header in the message. The API considers the default value of the ***'udhi'*** to zero.

SAMPLE FORMAT FOR SINGLE PART MESSAGE:

http://103.250.30.4/SendSMS/sendmsg.php?uname=XXXXX&pass=XXXX&send=XXXX XXXX&dest=91XXXXXXXXX&msg=093090109170 93E092E093E002000200939094 B00200917092F093E&dcs=8

In the above Sample message, it is a single part message and there is no need to tack header and set the '***udhi'*** value.

The header includes the values of length of header in octet, Mobile source and destination ports and message parts information.

One way of sending concatenated SMS (CSMS) is to split the message into parts, and sending each part with a User Data Header (UDH) tacked onto the beginning. A UDH can be used for various purposes and its contents and size varies accordingly, but a UDH for concatenating SMSes look like this:

Field 1 (1 octet): Length of User Data Header, in this case 05.

Field 2 (1 octet): Information Element Identifier, equal to 00 (Concatenated short messages, 8-bit reference number)

Field 3 (1 octet): Length of the header, excluding the first two fields; equal to 03

Field 4 (1 octet): 00-FF, CSMS reference number, must be same for all the SMS parts in the CSMS

Field 5 (1 octet): 00-FF, total number of parts. The value shall remain constant for every short message which makes up the concatenated short message. If the value is zero then the receiving entity shall ignore the whole information element

Field 6 (1 octet): 00-FF, this part's number in the sequence. The value shall start at 1 and increment for every short message which makes up the concatenated short message. If the value is zero or greater than the value in Field 5, then the receiving entity shall ignore the whole information element.

SAMPLE FORMAT FOR MULTIPART MESSAGE:

PART 1

http://103.250.30.4/SendSMS/sendmsg.php?uname=XXXXX&pass=XXXX&send=XXXX
XXXX&dest=91XXXXXXXXXX&msg=050003FA0501093809 4D0926094D092B094D093
8094D0926094D092B094D0938094D0926092B094D0938094D0926094D0917094D
0926094D092B094D0917094D0926094D092B094D0917094D0926094D092B094D0
938094D092B094D0938094D0926094D092B094D0938094D0926094D092B094D09
38094D0926092B094D0938094D0926092B0938094D0926&udhi=1&dcs=8

PART 2

http://103.250.30.4/SendSMS/sendmsg.php?uname=XXXXX&pass=XXXX&send=XXXX
XXXX&dest=91XXXXXXXXXX&msg=050003FA0502094D092B094D0938094D0926094
D092B094D0938094D09260940092B094D0939094D0938094D0926094D0915094D
092B094D0939094D0915094D0938094D09260939094D092B094D0915094D09320
94D0938094D0927094D092B094D0915094D0932094D0938094D0927094D092B09
4D0915094D0938094D0927094D092B094D0915094D0938&udhi=1&dcs=8

PART 3

http://103.250.30.4/SendSMS/sendmsg.php?uname=XXXXX&pass=XXXX&send=XXXX
XXXX&dest=91XXXXXXXXXX&msg=050003FA0503094D0927094D092B094D0915094
D0932094D0938094D0926094D091D094D092B094D0915094D0938094D0927094D
092B094D0915094D0938094D0927094D092B094D0932094D0915094D0938094D0
927094D092B094D0938094D0926094D0915094D0934094D092B094D0926094D09
16094D092B094D0915094D0938094D0926092B094D0915&udhi=1&dcs=8

PART4

http://103.250.30.4/SendSMS/sendmsg.php?uname=XXXXX&pass=XXXX&send=XXXX
XXXX&dest=91XXXXXXXXX&msg=050003FA0505094D0938094D0926094D0915094
D092B09410913092F094D0907092F094D0935093F091A094D0909092F094D09170
94D092E0930094D092E094D0938094D0926003B0027091A094D092A094D&udhi=1
&dcs=8

PART5

http://103.250.30.4/SendSMS/sendmsg.php?uname=XXXXX&pass=XXXX&send=XXXX
XXXX&dest=91XXXXXXXXX&msg=050003FA0504094D091C094D0938094D0926094
D092B094D0915094D0934094D092B094D0938094D0926094D0915094D0934094D
092B094D0939094D0938094D0926094D092B094D0916094D0938094D092609320
94D0915094D092B094D0939094D0938094D0926094D0915094D09320939094D09
2B094D0938094D0926094D0915094D09320939094D092B&udhi=1&dcs=8

In the above multi-part message, the ***'udhi'*** values should be set to 1 and the header should be tacked to every part message.

Sending Binary Messages (160 characters):

The HTTP API supports for 8 bit binary message such as Ring tones, logos, Picture Messages. However, it needs to set certain parameter for the Binary message in HTTP API else the message would be treated as normal text message and may not receive as expected. The following parameters need to be set for sending binary messages:

UDH should be set to 1

DCS value should be 4 or 240 or 245 (based on the spec)

Message should not exceed 280 characters

Following is the sample call:

SAMPLE FORMAT:

http://103.250.30.4/SendSMS/sendmsg.php?uname=XXX&pass=AXX&send=XXXXXXX&dest=91XXXXXXXXX&msg=BinaryMessageString&udhi=1&dcs=4

If the port number is defined in the header, the mobile device automatically identifies the message type whether it is a ringtone, picture, vCard, vCalendar, etc.

There are few standard ports for mobile device that can be included in the messages to send specific message: