Cloud Alert
Next Generation Notifications

The Cloud Alert Messaging Gateway

Integrator REST API

Developer's Guide & Reference

Document Version 5.2

http://cloudalert.cloud

Table of Contents

Introduction

The Cloud Alert Messaging Gateway is a software gateway that is deployed in a corporate's datacenter to allow applications to send messages using standard technologies. This service enables businesses and corporations to easily transmit alert, OTP and other messages to recipients conveniently and quickly.

If you do not have a copy of the Messaging Gateway, Starter editions of the software are available for free at:

http://cloudalert.cloud

Integrator

The Integrator is a set of APIs that allow software developers to integrate the functionality of the gateway with their applications, for example, send messages on internal application events.

This document details the REST API and its various functions.

REST-API

The REST API enables applications to communicate with the Messaging Gateway over HTTP. Applications can send requests to the server over regular HTTP requests, and receive confirmations and other information over the HTTP Response.

GET & POST

The server allows the functionality to be called using either GET or POST HTTP requests. While GET and POST each have their own limitations, the server does not differentiate based on the type, thereby allowing developers to test the service using the simpler GET, and later switching to POST to allow for longer URLs.

Encoding

URLs can only be sent over HTTP using the ASCII Character-set. Since GET presents the parameters to the server within the URL, the URL may end up containing characters outside the ASCII set such as non-English characters, or characters such as '&' which have a different meaning to the URL. The URL thus has to be converted into a valid ASCII format. URL encoding replaces unsafe ASCII characters with a "%" followed by two hexadecimal digits. URLs also cannot contain spaces; thus URL encoding normally replaces a space with a %20.

For more information about URL encoding, see http://en.wikipedia.org/wiki/Percent-encoding

The REST API requires that these special characters when used within parameter values should be encoded. There is no need to encode parameter names, since they have been designed to not have any disallowed characters.

Notes

- Note that if a message content contains “CR” or “LF” characters, they will be replaced by “\n”, ie, a character containing a Slash, followed by the character 'n'.
- The ip address (192.168.1.1) used in the document is only used as an exmaple. This ip address should be replaced by the hostname or ip address of the gateway deployed at your organization.

Syntax

The REST-API can be invoked using the following syntax:

```
http://<domain_name>/<function_name>?
<param1>=<value1>&<param2>=<value2>&...
```

For example, the following URL is used to send a message by calling the 'SendSMS' function:

http://192.168.1.1/RESTIntegrator_SendSMS_1?u=admin&p=pe56&s=Leonard&
r=true&f=false&n=Penny&d=441234534500&t=Hi%20There%21&o=NV&c=notif

This URL generates a notification on the customer's handset via your mobile app that is installed and registered with the mobile no. 441234534500, with the text 'Hi There!'. The username it uses is 'admin' and the password is 'pe56'. On posting the above request, the application will receive a response in the form of name-value pairs as shown below:

```
POSTED:TRUE

ERROR:

MESSAGE-ID:12383664

NUMBER-OF-PARTS:1
```

Features of the Integrator REST-API

In later sections are detailed the methods contained in the REST API. These methods can be invoked by your application to perform the required functionality.

Authentication

Security in the usage of the APIs is maintained by password-based authentication. Almost all of the functions require a username and password, which identifies a valid user on the gateway.

Multi-format Response

The REST API requests can return the response either in the form of name-value pairs or in XML format as specified by the user in the request. Listed below are the valid response format values for each of the formats that can be used in the REST API request. It is specified with the paramter 'o'.

Valid response format values

1. For responses as name-value pairs: **NV**
2. For responses in the XML format: **XML**

Example

Below is a sample REST API request for sending a notification message to a customer and getting back an XML response:

http://192.168.1.1/RESTIntegrator_SendSMS_1?u=admin&p=pe56&s=Leonard&
r=true&f=false&n=Penny&d=441234534500&t=Hi%20There%21&o=XML&c=notif

As specified in the request, the response would be returned in the XML format as shown below:

```
<?xml version="1.0" encoding="UTF-8"?>

<POSTRESULT>

  <POSTED>TRUE</POSTED>

  <ERROR></ERROR>

  <MESSAGE-ID>12506778</MESSAGE-ID>

  <NUMBER-OF-PARTS></NUMBER-OF-PARTS>

</POSTRESULT>
```

Message Category

Users can specify the category of the message that is to be delivered to the customer. The message category is specified in the REST API request. Messages can be categorized as either Alerts, Promotions or Notifications. Listed below are the valid message category values that can be used in the REST API request. It is specified with the parameter ‘c’.

Valid message category values

1. Alerts: alert, alerts, alerting, a, 1
2. Promotions: promo, promotion, promotions, promotional, p, 2
3. Notifications: notif, notification, notifications, notificational, n, 4

Example

Below is a sample REST API request for sending a promotional message to a customer:

http://192.168.1.1/RESTIntegrator_SendSMS_1?u=admin&p=pe56&s=Leonard&
r=true&f=false&n=Penny&d=441234534500&t=Get%20your%20platinum%20credit%20card%20now
%21&o=NV&c=promo

RESTIntegrator_SendSMS_1

This function posts a single message to the server.

Syntax:

```
http://<domain_name>/RESTIntegrator_SendSMS_1?
u=<username>&p=<password>&s=<sendername>&d=<destinationnumber>&t=<
text>&o=NV|XML&c=alert|notif|promo[&n=<destname>][&r=t|f][&f=t|f]
```

The request parameters to be passed to this function are:

Parameter Name	**Meaning**	**Description**
u	User name	The username (login name) used to access the gateway.
p	Password	The password used to access the gateway.
s	Sender name or short code	The Sender-ID or shortcode that should appear as the sender of the message. This name or shortcode is usually approved and provided to you by the Telco.
r	Report of Delivery	Whether a delivery report is required for the sent message. If set to true, a Delivery Report message will be returned at a later time, through a separate call to one of the getInboundReceipt functions. The recommended values are 't'(True)/'f'(False). For more possible values please refer to the Notes below.
f	Flash message	If the message is intended to be sent as a flash message. The recommended values are 't'(True)/'f'(False). For more possible values please refer to the Notes below.
n	Name of the recipient	The name of the recipient. Optional. This field is useful if names of recipients are to be stored in your Sent Messages for your future reference.
d	Destination phone number	The destination phone number of the recipient (including the country code). The phone number should be numeric, and not include any leading '0's or '+' symbols.

t	Text of the message	The text content of the message.
o	Response format of the request	If NV is specified then a response in the form of name-value pairs is returned. If XML is specified then the response is returned in an XML format.
c	Message category	The category of the message: alert, notification or promotion. Refer to the Message Category section of this document for a detailed list of possible valid values.

Also note that the parameters p(password),s(sender name), n(name of the recipient) and t(message text) would require a possible encoding of the URL. Refer the Encoding section for further details.

The response is multiline, either as name-value pairs or in xml format as specified in the request.

The following information is provided in the response:

Name	**Description**
POSTED	True: if the message was accepted successfully. False: if the message was rejected by the gateway.
ERROR	Error message if any occurred while posting the message to the queue. The value will be blank if the message was accepted.
MESSAGE-ID	Server generated unique ID representing a message successfully posted to the server. It is recommended that you store this value for future reference. The value will be blank if the message post was unsuccessful.
NUMBER-OF-PARTS	Number of parts into which the message was split before posting. Each part is billed separately. The value will be blank if the message post was unsuccessful.

Notes:

All the response titles are provided, irrespective of whether the post was a success or not, for example, the NUMBER-OF-PARTS will be displayed even for messages that could not be posted. The value for these types will be blank, and so should be ignored.

For the boolean types above, besides the recommended values of 't'/'f', more possible values can also be used such as: 'T', 'Y' and 'y' to represent True or 'F', 'N' and 'n' to represent False.

Examples:

Example 1

http://192.168.1.1/RESTIntegrator_SendSMS_1?u=l.hofstadter&p=pe56&s=Leonord&r=true&f=false&n=Penny&d=441234534500&t=Hi%20There%21&o=NV&c=notif

Sample Output (of a successfully posted message):

```
POSTED:TRUE

ERROR:

MESSAGE-ID:12383664

NUMBER-OF-PARTS:2
```

Sample Output (of a failed posted message):

```
POSTED:FALSE

ERROR:The sendername is not allowed for this user

MESSAGE-ID:

NUMBER-OF-PARTS:
```

Example 2

http://192.168.1.1/RESTIntegrator_SendSMS_1?u=l.hofstadter&p=pe56&s=Leonord&r=true&f=false&n=Penny&d=441234534500&t=Hi%20There%21&o=XML&c=alert

Sample Output (of a successfully posted message):

```
<?xml version="1.0" encoding="UTF-8"?>

<POSTRESULT>

  <POSTED>TRUE</POSTED>

  <ERROR></ERROR>

  <MESSAGE-ID>12383664</MESSAGE-ID>

  <NUMBER-OF-PARTS>2</NUMBER-OF-PARTS>

</POSTRESULT>
```

Sample Output (of a failed posted message):

```
<?xml version="1.0" encoding="UTF-8"?>

<POSTRESULT>

  <POSTED>FALSE</POSTED>

  <ERROR>The sendername is not allowed for this user</ERROR>

  <MESSAGE-ID></MESSAGE-ID>

  <NUMBER-OF-PARTS></NUMBER-OF-PARTS>

</POSTRESULT>
```

Additional information and Support

Should you have any additional questions, comments or feedback on the Messaging Gateway, contact us over email at:

support@cloudalert.cloud

or visit our website at

https://cloudalert.cloud