



Application Programming Interface

Universal Messaging Gateway

Version 2.2

1 Table of Contents

2	Change History	3
1	Overview	4
1.1	Communication between client and UMG	4
1.1.1	Mobile Originating Message	4
1.1.2	Mobile Terminating Message.....	5
2	Supported Function	6
2.1	Function overview	6
2.2	Message format.....	6
2.2.1	Request Token.....	6
2.2.2	Receive MO	7
2.2.3	submitMT via HTTP	8
2.2.3.1	submitted via HTTP (non-token).....	9
2.2.4	submitMT via raw SMPP	9
2.2.5	submitMTAck	10
2.2.6	Delivery Notification	11
2.3	Status Code.....	12
2.4	Delivery Notification (DN) Status Code	12
2.5	Testing the API (with token)	13
2.6	Testing the API (without token)	14

2 Change History

Date	Version	Change
18-Apr-24	1.0	Initial version
01-Oct-24	2.0	Add: submitted via HTTP (non-token)
13-Dec-24	2.1	Add: swagger API test
01-Apr-24	2.2	Change of API Call domain name

1 Overview

1.1 Communication between client and UMG

Communication between client and Universal Messaging Gateway (UMG) is in synchronous mode. UMG supports Mobile Originating (MO) and Mobile Terminating (MT) messages. For MO messages, UMG will invoke and pass the message to client, while for MT messages, client will submit the messages to UMG.

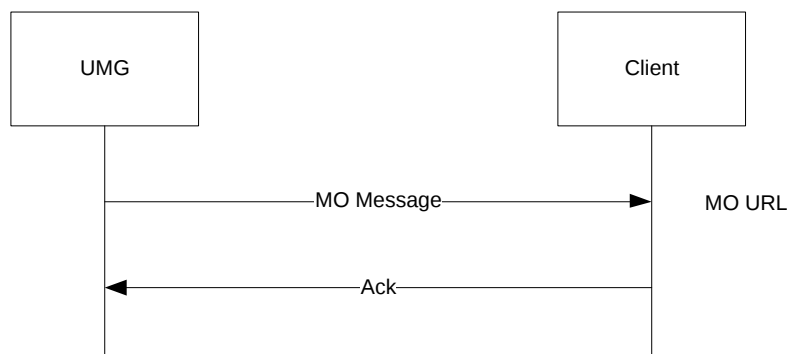
All communication between UMG and client and vice versa can be based on 2 method :

- 1) HTTP POST – it is using Json payload with bearer token authentication
- 2) SMPP version 3.4 – it is using standard smpp protocol

1.1.1 Mobile Originating Message

Mobile Originating (MO) SMS is a SMS initiated by mobile phone. When UMG received user SMS, it will forward to client applications

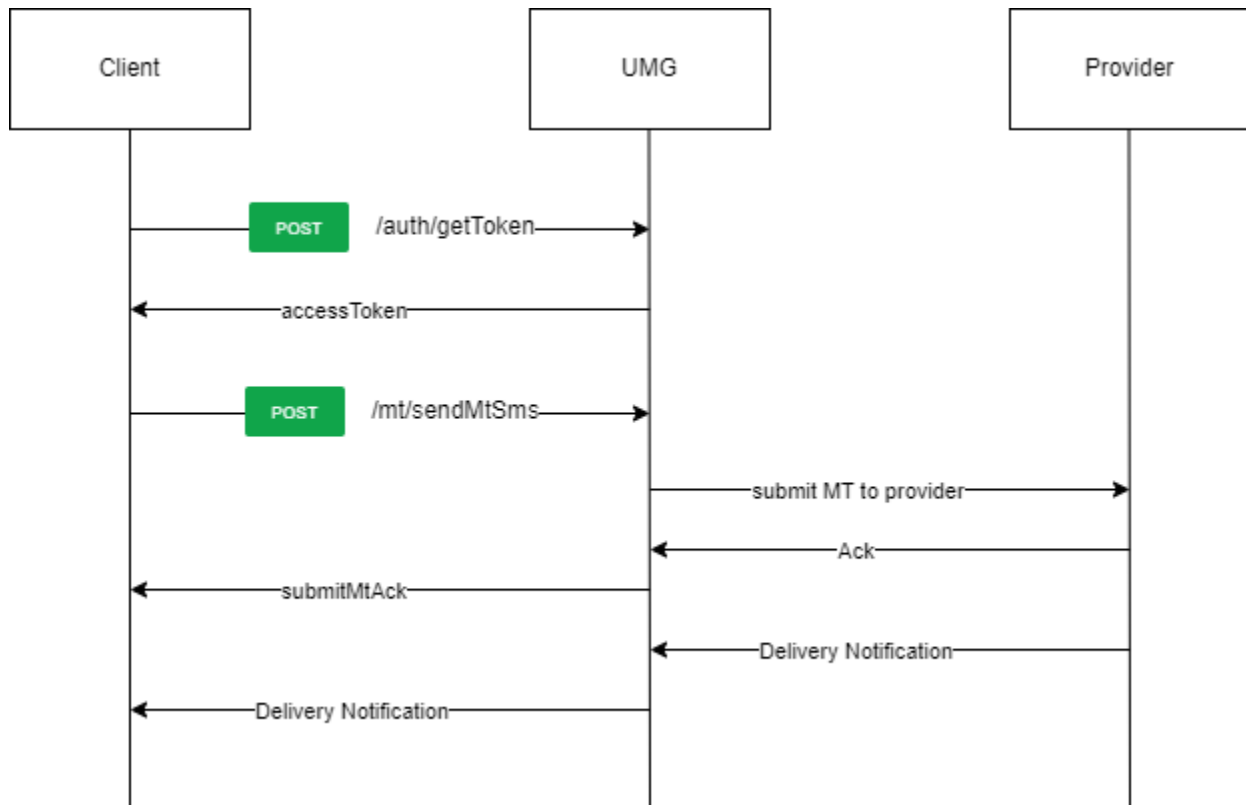
The following diagram shows how client application will receive MO SMS from subscribers:



1.1.2 Mobile Terminating Message

Mobile Terminating (MT) SMS is a SMS terminated to mobile phone. When UMG received SMS from client application, it will send to mobile phone subscriber

The following diagram shows the process of Client submitting MT SMS for delivery by UMG.



When a message has been successfully delivered to the user, UMG will invoke the callback URL of the client to update status as DELIVERED. Client must register with valid callback URL with UMG. This process is done manually when system administrator creates an account for the client application.

2 Supported Function

2.1 Function overview

The following are transaction supported by UMG on MO.

Function description	Function name
Receive Short Message from subscribers	N/A

The following are transaction supported by UMG on MT.

Function description	Function name
Send Short Message	submitMT, submitMTAck
Message delivery notification (DN)	N/A

2.2 Message format

2.2.1 Request Token

Parameter	Description	Data type/max. length
username	API username. UMG admin will provide to customer	Alphanumeric (20)
password	API password. UMG admin will provide to customer	Alphanumeric (20)

e.g: Request

```
{
  "password": "apipassword",
  "username": "apipass@123"
}
```

Response (SUCCESS):

```
{
  "accessToken":
  "eyJhbGciOiJIUzUxMiJ9.eyJzdWIiOiJuZXdhZG1pbilzImhhdCI6MTcxMzQ0NDQ5NCwiZXhwIjoxNzEzNDQ4Mk0fQ._JBFwnfLGV3kMIrIR__fNxjK0areP0x9LYORbq-fyR-42u6yIJdbOF1nqtJ1fdBz356y5H2HPYUs0hUroB_8w",
  "refreshToken": "84b9917c-5693-4aaa-a344-044c2dfc1bdb",
}
```

```
"expiry": 3600000,  
"tokenType": "Bearer"  
}  
Response (Unauthorised):
```

```
{  
  "timestamp": "2024-04-18T12:55:38.339+00:00",  
  "status": 401,  
  "error": "Unauthorized",  
  "message": "",  
  "path": "/auth/getToken"  
}
```

2.2.2 Receive MO

Parameter	Description	Data type/max. length
GwMsgId	UMG Message ID. Mandatory.	Alphanumeric (20)
SvrMsgId	Server Message ID. Optional depending on server side setting.	Alphanumeric (20)
Message	SMS Content	Alphanumeric (480)
SrcAddr	Sender mobile number in international format with country code	Alphanumeric (15)
DestAddr	Destination mobile number in international format with country code or Short-code	Alphanumeric (15)
DataCoding	SMS content encoding. Supported datacoding : 0 - 'ASCII' (example:English word, Malay word) 8 – Unicode (example: Chinese word, korea word)	Num (1)

e.g:

http://www.abc.com/mo_sms.php?GwMsgId=201108031EF00FFF&SvrMsgId=999999999&Message=TR%200088017123456789%2050&SrcAddr=60133331234&DestAddr=22422&DataCoding=0

Based on the above example, the MO URL to be registered in UMG is
http://www.abc.com/mo_sms.php

2.2.3 submitMT via HTTP

Parameter	Description	Data type/length
clientMsgId	Client Message ID. Mandatory.	Alphanum (20)
msgEncoded	SMS Content in base64 encoded	Alphanum (480)
senderId	Sender number in international format with country code or Short-code	Alphanum (15)
destAddr	Destination mobile number in international format with country code	Alphanum (15)
dataCoding	SMS content encoding. Supported datacoding: 0 - 'ASCII' (example:English word, Malay word) 8 - Unicode (example: Chinese word, korea word)	Num(1)
srcTon	Sender number Type of Number(TON). Optional. value: 0	Num(1)
srcNpi	Source number Number Plan Indicator(NPI). Optional. Value: 0	Num(1)
destTon	Destination number TON. Optional. Value:0	Num(1)
destNpi	Destination number Number Plan Indicator(NPI). Optional. Value:0	Num(1)

e.g:

```
{  
  "clientMsgId": "MT001",  
  "dataCoding": 0,  
  "destAddr": "60133731869",  
  "destNpi": "0",  
  "destTon": "0",  
  "msgEncoded": "aGVsbG8=",  
  "senderId": "66001",  
  "srcNpi": "0",  
  "srcTon": "0"  
}
```

2.2.3.1 submitted via HTTP (non-token)

Sending API SMS

This is the version we used most, due to all the PHP or CURL code already generated with HTTP Webhook and the host was simplified to let the user just take a few sentences of important things to embed to our server.

What only needed:

https://api.anchor-sms.com:8443/mt/sendMtSmsNoToken?

**to=60121234567&user=xxxxxxx&pass=xxxxxxxxxxxxxxxx&text=Message+from+Anchor+SMS&from=xxx
xx**

1. headers=application/json
2. to=60121234567 (*This the sms targeted phone number*)
3. text=Message+from+Anchor+SMS (*This the sms content to sent*)
4. user=xxxxxxx (*This is individual account generated APP_KEY/USERNAME*)5. pass=xxxxxxxxxxxxxxxx
(*APP_SECRET /PASSWORD*)3. from=xxxx (*Optional*)4. dataCoding=0 or 8 (*Optional*) ; 0 for ASCII and 8 for Unicode

2.2.4 submitMT via raw SMPP

Client need to do SubmitSM request to UMG using SMPP standard version 3.4

2.2.5 submitMTAck

Parameter	Description	Data type/length
timestamp	Time of submitMT ack	Datetime
status	HTTP response code	Integer(3)
message	HTTP response description	Alphanum (20)
data	Client Message ID	
	Parameter	Description
	clientMsgId	Client Message ID
	umgTxId	UMG Message ID. Mandatory.
	statusCode	Status of the message, refer Status code section 2.3
	statusDescription	Status Description
path	url endpoint path. It is for reference	Alphanum (20)

The submitMTAck message will be in form of JSON.

e.g:

i) Positive response

```
{
  "timestamp": "2024-04-08T06:13:48.925+00:00",
  "status": 201,
  "message": "Created",
  "data": {
    "clientMsgId": "MT001",
    "umgTxId": "240408141348960d4b",
    "statusCode": "001",
    "statusDescription": "Submitted"
  },
  "path": "/mt/sendMtSms"
}
```

ii) Negative response

```
{
  "timestamp": "2024-04-18T12:36:32.464+00:00",
  "status": 401,
```

```
"error": "Unauthorized",  
"message": "",  
"path": "/mt/sendMtSms"  
}
```

2.2.6 Delivery Notification

Parameter	Description	Data type/length
GwMsgId	UMG Message ID. Mandatory.	Alphanum (20)
SvrMsgId	Server Message ID. Optional depending on server side setting.	Alphanum (20)
StatusCode	DN Status, refer Status code section 2.4	Alphanum (3)

e.g:

http://www.abc.com/dn_handler.php?GwMsgId=201108031EF00FFF&SvrMsgId=999999999&StatusCode=100

Based on the above example, the DN URL to be registered in UMG is

http://www.abc.com/dn_handler.php

2.3 Status Code

Code	Description	Possible cause
001	Message received by UMG and in process of delivery	N/A
100	Message delivered to subscriber	N/A
101	Message failed to be delivered	- Subscriber not reachable - Invalid MSISDN - MSISDN being blocked at telco end
104	Invalid message format	- Unrecognized/unsupported message format - Message exceeds limit
201	Client authentication failed	Either System Id or Password is incorrect
202	Client account has insufficient balance	Account has insufficient balance
203	No routing defined in gateway	No route or/and pricing has been define for the MSISDN prefix the client attempting to send
901	Unknown error	Unknown error
902	External connection error	A failure in connectivity with backend provider either telco or other 3 rd party SMS provider
903	System error	Internal system error (please contact Anchor Support)

2.4 Delivery Notification (DN) Status Code

Code	Description	Possible cause
DELIVRD	SMS is delivered to subscriber	N/A
FAIL	Message Fail delivered to subscriber	- Subscriber number is blocked or blacklisted by mobile operator - the destination phone number is invalid or does not exist
EXPIRED	Message validity period has expired. The message has failed to be delivered within its validity period	- Subscriber had not turn on the phone within SMS validity period - subscriber phone is out of coverage within SMS validity period
REJECTD		

2.5 Testing the API (with token)

User can perform tests on all the above function call using the Swagger interface provided.

URL: [**https://api.anchor-sms.com:8443/swagger-ui.html#/**](https://api.anchor-sms.com:8443/swagger-ui.html#/)

POST Method:

Steps:

1. First, get a valid token by triggering getToken function. On the Swagger interface, select auth-controller to expand the API and then click /auth/getToken .

2. In the request box, put the below values

```
{  
  "password": "Kalsym@123",  
  "username": "DFVR001"  
}
```

3. Click the button "Try it out!"

4. From response body, copy the accessToken value. example accessToken value =
eyJhbGciOiJIUzUxMiJ9.eyJzdWUiOiJERlZSMdAxliwiaWF0IjoxNzEzNTE0OTU4LCJleHAiOiJlE3MTM1MTg1NTh
9.Q2cmzaQRxT-b8N-mRUhDWK2cjuRhUMgHyGEfi_xs053mS8f2pA2s-i5e6__6Qxx8svMjyQ-
Rcrjj1hZEUUSZyw

5. go to the right top corner and click button "Authorize".

6. put "Bearer <token u copy at #4>" in the api_key input box and click button "Authorize".

7. to send sms, click "mt-controller" and click "/mt/sendMtSms".

8. in the mtRequest box, put below. ensure you change the receiver phone number "destAddr"

```
{  
  "clientMsgId": "MT001",  
  "dataCoding": 0,  
  "destAddr": "601156301823",  
  "destNpi": "0",  
  "destTon": "0",  
  "msgEncoded": "aGVsbG8=",  
  "senderId": "66001",  
  "srcNpi": "0",  
  "srcTon": "0"  
}
```

9. then click button "Try it out!!".

2.6 Testing the API (without token)

Sending API SMS

This is the version we used most, due to all the PHP or CURL code already generate with HTTP Webhook and the host was simplify to let user just take few sentence of important thing to embedded to our server.

GET Method:

What only needed: <https://api.anchorsms.com:8443/mt/sendMtSmsNoToken?>

`to=60121234567&user=xxxxxxx&pass=xxxxxxxxxxxxxx&text=Message+from+Anchor+SMS&from=xxxxx`

1. **headers**=application/json
2. **to**=60121234567 (*This the sms targeted phone number*)
3. **text**=Message+from+Anchor+SMS (*This the sms content to sent*)
4. **user**=xxxxxxx (*This is individual account generated APP_KEY/USERNAME*)
5. **pass**=xxxxxxxxxxxxxx (*APP_SECRET /PASSWORD*)
6. **from**=xxxx (*Optional*)
7. **dataCoding**=0 or 8 (*Optional*) ; 0 for ASCII and 8 for Unicode