



API v3.2 Reference

LAST UPDATE: February 2015



- General Overview 4**
 - API Version 3.2 4
 - Getting Started 4
 - API URL 4
 - HTTP and HTTPS 4
- Send An SMS..... 5**
 - Overview 5
 - HTTP POST Parameters*..... 5
 - Structure of the compressed pipe-delimited file* 13
 - Response to your HTTP POST*..... 15
- Receive an SMS 16**
 - Overview 16
 - Receiving SMS Messages On Demand..... 17
 - HTTP POST Parameters*..... 17
 - Response to your HTTP POST*..... 20
 - Receiving SMS Messages Real-Time..... 22
 - HTTP GET Parameters*..... 24
- Receiving a Delivery Receipt (DLR)..... 30**
 - Overview 30
 - HTTP GET Parameters*..... 32
- Check Delivery Status of a Message..... 36**
 - Overview 36
 - Delivery Status Check - API URL* 36
 - HTTP POST Parameters*..... 36
 - Response To Your HTTP POST*..... 38
- Create a Sub-Account..... 39**
 - Overview 39
 - Account - API URL* 39
 - HTTP POST Parameters*..... 39
 - Response To Your HTTP POST*..... 42
- Create a User 44**
 - Overview 44
 - User - API URL* 44
 - HTTP POST Parameters*..... 44
 - Response To Your HTTP POST*..... 48



- Check Your Account Balance 49**
 - Overview 49
 - HTTP POST Parameters*..... 49
 - Response To Your HTTP POST*..... 50

- Success/ Error Response Codes 51**
 - When sending an SMS message..... 51
 - When receiving a delivery receipt 51
 - Response Codes..... 52

- API Support 54**
 - Email..... 54
 - CONTINUAL IMPROVEMENT... 54

General Overview

API VERSION 3.2

This is the reference document for version 3.2 of the SMS API provided by SMS Central.

This API, based on the HTTP protocol, is for developers who want to integrate SMS messaging with their own standalone or web applications.

GETTING STARTED

The API is based on the HTTP protocol. This means you can use any web development language to access the API. If you're using PHP, .NET or JAVA check out our user guides and code samples to help you get on your way to integrating SMS messaging into your application.

API URL

All HTTP POSTS sent to SMS Central's API must go to the following URL:

```
https://my.smscentral.com.au/api/v3.2
```

HTTP AND HTTPS

The SMS Central API is served over HTTPS to ensure secure and private messaging. Non-encrypted HTTP is supported, however this is not recommended.



Send an SMS

OVERVIEW

To send an SMS, simply send a HTTP POST to the API URL with the required POST Parameters.

SMS Central will then output a string response with the response code indicating the status of your message.

You can check the delivery status of your messages via the API (more info on pg. 36).

You may also receive Delivery Receipts to your email address or URL (more info in pg. 30), which provide the final delivery status of the SMS.

HTTP POST Parameters

The parameters below are the POST parameters that can be provided in order to send an SMS message.

Please ensure these parameter names are submitted in uppercase.

Parameter	Required	Type	Possible/ Example Value(s)
USERNAME	Yes	String	
PASSWORD	Yes	String	
ACTION	Yes	String	'send'

ORIGINATOR	Yes	String (Alphanumeric, max 11 chars)	Mobile Number, Sender Name, 'shared' , 'dedicated'
RECIPIENT	Yes	String	
CAMPAIGN	No	String	*max length 100 characters
REFERENCE	No	String	*max length 100 characters
SCHEDULE	No	Datetime	2011-12-25 00:00:00
MESSAGE_TEXT	Yes	String	<i>See definition for allowed chars</i>
URL	No	String	
RECIPIENTMESSAGES	No	JSON Data (array)	Allows you to send up to 100 messages with one HTTP POST.
REPLYPATH	No	String	URL (for POST) or Email Address
REPLYTYPE	No	String	WWW, HTTP, EMAIL
FILE_LIST	No	String	filename
FILE_HASH	No	String	MD5 hash value of uploaded file

POST Parameter Definitions

USERNAME

- **Type:** string
- **Description:** This is your Username for your SMS Central account

PASSWORD

- **Type:** string

- **Description:** This is your Password for your SMS Central account

ACTION

- **Type:** string
- **Description:** To send an SMS message, please provide the string value 'send' (without single quotes) as this POST parameter value.

ORIGINATOR

- **Type:** string
- **Description:** This is the name or number your SMS message is to appear to come from, it can be alphanumeric (letters and numbers only) and can be up to a maximum of 11 characters.
 - * To send from one of SMS Central's pool of shared numbers, please use the string value 'shared' (without single quotes).
 - * To send from your own dedicated numbers, cycling through them so the recipient does not receive an SMS from the same number, please use the string value 'dedicated' (without single quotes).

RECIPIENT

- **Type:** string
 - Description:** This is the number you wish to send your SMS message to. SMS Central recommends that you provide the number in International format.
 - * An example Australian recipient value: 61435800957

CAMPAIGN

- **Type:** string
 - Description:** This string value allows you to group messages by providing a new campaign name, or provide an existing campaign name to associate these messages to that campaign. This parameter is useful in reporting.
 - * If you specify a 'shared' or a dedicated number as the ORIGINATOR, responses to your message will also provide this campaign value.

See Receive an SMS (pg. 16).

REFERENCE

- **Type:** string
- **Description:** This string value allows you specify your own unique value to identify the message you are sending.
 - * This value will also be returned in delivery receipts (pg. 30).
 - * If you specify a 'shared' or a dedicated number as the ORIGINATOR, responses to your message will also provide this reference value, allowing you to match replies with the message you sent. See Receive an SMS (pg. 16) for more info.
 - * The 'REFERENCE' value must be a unique value for each message, an attempt to send 2 messages with the same 'REFERENCE' value will result in the message being rejected with a 513 error code.

SCHEDULE

- **Type:** datetime
- **Description:** This datetime value allows you to schedule the message to be delivered on the day and time of your SCHEDULE parameter value. The message will sit in SMS Central' s queues until the scheduled date and time, upon which the message will be sent.
 - * An example datetime value, representing December 25th, 2011 and 3:35pm is:

```
2011-12-25 15:35:00
```

MESSAGE_TEXT

- **Type:** string
- **Description:** This is the text of the SMS message to be sent and this text must be URL encoded.

Please note that if any characters outside of the below list of ASCII characters are submitted, the message will be treated as a Unicode message.

A-Z
0-9
~!@#\$%^&*()-_=[?<>,'.";:/\{\}
(whitespace)

** Please note:* A standard GSM SMS message has a maximum of 160 characters. Longer messages are definitely possible, however please be aware that exceeding 160 characters will constitute a 'second' message. When a message is longer than 160 characters, this is referred to as a multi-part message as it contains multiple messages (or multiple-parts). The total SMS limit then becomes 153 characters per 'part' as the 7 characters are used up by invisible headers and footers which denote which part of the message is being sent (i.e. Part 1 of 2). For example:

1 message = 160 characters available

2 messages = 306 characters available (153+153)

3 messages = 459 characters available (153+153+153)

4 messages = 612 characters available (153+153+153+153)

5 messages = 760 characters available (153+153+153+153+153)

6 messages = 918 characters available (153+153+153+153+153+153)

7 messages = 1071 characters available (153+153+153+153+153+153+153)

** A maximum of 7 parts are possible*

Note: If your message is treated as a Unicode message (due to having characters not within the GSM character set), a single Unicode SMS message has a maximum of 70 characters. Longer messages are possible, however in this case each message part will contain a maximum of 67 characters.

URL

- **Type:** string
- **Description:** By providing a URL to content in this optional parameter, the message will be converted into a WAP Push message. In this case the value within the MESSAGE_TEXT field will be the WAP message label.

RECIPIENTMESSAGES

- **Type:** JSON data (array)
- **Description:** The value for this parameter is JSON data (array) which contains the 'RECIPIENT' , 'MESSAGE_TEXT' and 'REFERENCE' (optional) values for each of the (up to) 100 messages, where 'RECIPIENT' is the mobile number of the recipient, and 'MESSAGE_TEXT' is the specific text of the message to be delivered to that recipient and 'REFERENCE' is the string value of a unique reference that you can apply to the message (also to be used for delivery receipt tracking).

This parameter will accept up to 100 messages in the one HTTP POST. An example value, in JSON format is:

```
[{"RECIPIENT": "61435800957", "MESSAGE_TEXT": "Message 1",  
  "REFERENCE": "unique01"},  
 {"RECIPIENT": "61406040271", "MESSAGE_TEXT": "Message 2",  
  "REFERENCE": "unique02"}]
```

If a value for the 'RECIPIENTMESSAGES' parameter is provided in your HTTP POST, the 'RECIPIENT' and 'MESSAGE_TEXT' and 'REFERENCE' POST Parameters will be ignored and the values in this JSON data will be applied.

Please ensure you send via HTTP POST if you intend to utilise the RECIPIENTMESSAGES parameter

* **Please note:** Sending (up to) 100 messages via the RECIPIENTMESSAGES parameter will still receive the same 'Response to your HTTP POST' . If any of the 100 messages is rejected due to an invalid number, all 100 messages will be rejected and the entire JSON Data batch (HTTP POST) will need to be re-sent, to deliver the messages.

REPLYTYPE

- **Type:** String
- **Description:** This is an optional field, allowing you to indicate that you would like Reply messages to the SMS message you are sending, to be forwarded to an Email Address or web URL (as specified in REPLYPATH). The acceptable values for this parameter are: EMAIL, WWW, HTTP. These values are case-sensitive.

A value of EMAIL will indicate the reply should be forwarded to an email address. A value of WWW or HTTP will indicate the reply should be forwarded to a URL (this also includes an HTTPS URL).

Note: If the 'REPLYTYPE' is provided, then a value for REPLYPATH is mandatory.

REPLYPATH

- **Type:** String
- **Description:** If the REPLYTYPE parameter is provided, the REPLYPATH parameter is also required. This parameter allows you to indicate the location where Reply messages to the SMS message you are sending, should be forwarded.

If the 'REPLYTYPE' parameter value is 'EMAIL' the REPLYPATH acceptable value is a valid format email address.

If your 'REPLYTYPE' parameter value is 'WWW' or 'HTTP' the REPLYPATH acceptable value is a URL or IP address (preceded by HTTP).

Within the WWW or HTTP option for REPLYPATH, you have additional functionality available as provided below

Within the URL provided as your REPLYPATH value, you may also specify query string parameters. There are also specific query string parameters you may specify, with tags, which we will then substitute with actual values. For example:

```
http://www.yourcompany.com?phone=#FROM#&msg=#MESSAGE_TEXT#&date  
_sent=#SENTDTS#&DATE_RECEIVED=#RCVDDTS#&recipient=#RECIPIENT#
```

The above query string parameters of #FROM# and #MESSAGE_TEXT#, #RECIPIENT# and #SENTDTS# and #RCVDDTS# will be replaced with the actual values such. The following list indicates what each of those tags represent.

- #FROM# - This represents the mobile number of the handset
- #ORIGINATOR# - This represents the mobile number of the handset
- #RECIPIENT# - This represents the number being replied to
- #MESSAGE_TEXT# - This represents the text of the message
- #SENTDTS# - Datetime of the time the original outbound message was sent
- #RCVDTS# - Datetime of the time the reply message was received

Note: When sending the REPLYPATH value as a POST or GET Parameter, it must be URL-encoded. If you are providing the REPLYPATH within the JSON-encoded RECIPIENTMESSAGE Parameter, it does not need to be URL-encoded as the

JSON-encoding will be sufficient.

Attaching a compressed pipe-delimited file

SMS Central provides functionality allowing you to send a much larger batch of messages than the 'RECIPIENTMESSAGES' limitation of 100 messages. You may attach a compressed pipe-delimited file with the API request, which may contain up to One Million SMS messages (1,000,000 SMS). In doing so, the 'RECIPIENTMESSAGES' parameter will be ignored and the following will apply:

FILE_LIST

- **Type:** String
- **Description:** This is the file name of the file being attached to the request.
Please note (1): the uncompressed file name must end in ".csv" or "txt" to be considered valid.
Please note (2): the file must contain the header (or empty line) in the first line of the file, as the first line is ignored.

FILE_HASH

- **Type:** String
- **Description:** This is the MD5 hash value of the file you are attaching, essentially allowing SMS Central to validate the file and data being provided.

Structure of the compressed pipe-delimited file

- The file may be compressed using ZIP format.
- Structure of the pipe-delimited file is:
Header: id|recipient|message_text|originator

Sample: 123456|61406040271|" Hello World" | 61435800957

The fields available (as listed in the Header above) are:

id

This is equivalent to the 'REFERENCE' parameter and is a unique value that you may provide for the message.

recipient

This is the mobile number of the recipient of the message to be sent

message_text

This is the text of the message. This is an optional value, if no value is provided, the value in the 'MESSAGE_TEXT' POST/GET parameter will be used.

originator

This is the name or number your SMS message is to appear to come from, it can be alphanumeric (letters and numbers only) and can be up to a maximum of 11 characters. This is an optional value and if none is provided, the 'ORIGINATOR' POST/GET parameter value will be used.

Response to your HTTP POST

Your HTTP POST will give you a response providing you with an immediate result of your SMS send.

SMS Central will always return a "HTTP/1.1 200 OK" HTTP status code in the header of the response to your HTTP POST.

The content of the response will contain the Success or Error response code and the corresponding success or error response message. Please see Success and Error Response Codes (pg. 51) for a list of all possible response codes.

A response code with the number '0' indicates that your HTTP POST was successful and the message is being processed for delivery.

A successful HTTP POST to Send an SMS will result in the following content in the HTTP Response:

```
0
```

A response to your HTTP POST indicating an error will contain the error response code and description in the same line, separated by whitespace. An example of such a response is as follows:

```
511 Username or Password incorrect
```

Please note: These results are not delivery receipt results; they are there to provide you with information on the success of your HTTP POST, not the message delivery itself.

Please see Receive an SMS (pg. 16) for receiving final carrier delivery confirmation status via Delivery Receipts.

Receive an SMS

OVERVIEW

If you have a dedicated mobile number, or if you have sent messages using a shared mobile number, you are able to receive SMS messages as replies, or run an inbound SMS campaign.

There are two ways to receive an SMS using SMS Central's API, these are 'on-demand' and 'real-time'.

The difference between the two methods is that "on-demand" allows you to view messages you have received whenever you like, while "real-time" provides you the ability to receive SMS messages in real-time, forwarded to a URL that you provide.

Of course, you can also always check the messages you have received by messages by logging in to your SMS Central account and viewing your Inbox, or by generating an 'Inbound' report.

RECEIVING SMS MESSAGES ON DEMAND

In order to periodically poll SMS Central' s server to check for received SMS messages, you require an HTTP POST to SMS Central API URL.

SMS Central will always return an "HTTP/1.1 200 OK" HTTP status code in the header of the response to your HTTP POST, together with the data requested in the content of the response to your HTTP POST.

HTTP POST Parameters

To send an on-demand request to retrieve SMS messages that have been received, please send a HTTP POST to the API URL with the following POST Parameters.

Parameter	Required	Type	Possible/ Example Value(s)
USERNAME	Yes	String	
PASSWORD	Yes	String	
ACTION	Yes	String	'read'
DATESTART	No	Datetime	2011-12-25 00:00:00
DATEEND	No	Datetime	2011-01-25 23:59:59
STATUS	No	String (all uppercase)	"ALL" , "UNREAD" , "READ"
CAMPAIGN	No	String	* Name of your campaign (URL encoded value)

POST Parameter Definitions

USERNAME

- **Type:** string
- **Description:** This is your Username for your SMS Central account

PASSWORD

- **Type:** string
- **Description:** This is your Password for your SMS Central account

ACTION

- **Type:** string
- **Description:** To view received SMS messages, please provide the string value 'read' (without single quotes) as this POST parameter's value.

DATESTART

- **Type:** datetime
- **Description:** To specify a day and time for how far back you wish to retrieve the received SMS messages, please specify this day and time in datetime format.
* For example, if you wish to retrieve messages newer than and including 25 December 2011, please provide a value of 2011-12-25 00:00:00

DATEEND

- **Type:** datetime
- **Description:** To specify a day and time for an end period for which you wish to retrieve the received SMS messages, please specify this day and time in datetime format.

For example, if you wish to receive messages up until and including 25 December 2011, please provide the value of 2011-12-25 23:59:59

STATUS

- **Type:** string
- **Description:** This STATUS parameter allows you to specify which type of messages you would like to retrieve, the options, to be provided as values (all characters uppercase) are:
 - 'ALL' – this will retrieve all received SMS messages, for the dates specified in DATESTART to DATEEND (or every single message, if no values are provided for those parameters)
 - 'UNREAD' – this will retrieve all SMS messages, for the dates specified in DATESTART to DATEEND, which are marked as unread. Please note that retrieving unread messages will result in them consequently being marked as read
 - 'READ' – this will retrieve all received SMS messages, which have been marked as read (either by retrieving unread messages via the API, or marking them as read by logging in to your SMS Central account).

CAMPAIGN

- **Type:** string
- **Description:** The CAMPAIGN parameter allows you to retrieve the reply messages to the campaign you may have just sent. By providing the name of the campaign (URL encoded) as per the value provided via the CAMPAIGN parameter when sending an SMS, you can retrieve reply messages to that campaign. For example, you may provide the CAMPAIGN value of the name of your campaign and the STATUS value of 'UNREAD' in order to retrieve new (unread) reply messages to your campaign.

Response to your HTTP POST

SMS Central will always return a "HTTP/1.1 200 OK" HTTP status code in the header of the response to your HTTP POST.

The 'content' of the response to your HTTP POST will be in XML format when messages are received, or a simple string if there was an error with your request.

The XML Response will contain a collection of <message> tags, inside a single <messages> tag.

Each <message> tag will contain the following tags wrapped within:

<datestamp> - this is the date the message was received

<direction> - this will always be 'MO' which stands for Mobile Originated and represents an inbound message

<originator> - this is the mobile number of the handset which sent in the message

<recipient> - this is the mobile number that the handset was sending the message to

<messagetext> - this is the text message content that was received

Example of a Successful Response To Your Request (type: XML)

```
<messages>
  <message>
    <datestamp>2011-10-13 21:28:44</datestamp>
    <direction>MO</direction>
    <originator>61xxxxxxxx</originator>
    <recipient>61xxxxxxxx</recipient>
    <messagetext>Lorem ipsum dolor sit
amet</messagetext>
  </message>
</messages>
```

Possible Error Responses to Your Request

(type: string)

```
Invalid From date, Please input date in this format YYYY-  
mm-dd. Eg; 2011-12-25
```

```
Invalid End date, Please input date in this format YYYY-  
mm-dd. Eg; 2011-12-25
```

```
Unable to fetch unread messages
```

RECEIVING SMS MESSAGES REAL-TIME

To receive SMS messages in real-time, please log in to your SMS Central account to set up the URL where you would like received messages to be forwarded. SMS Central recommends using a HTTPS URL.

When Receiving SMS Messages in Real-Time, SMS Central's server will send a HTTP GET to the URL you have specified, as soon as a message is received.

IMPORTANT NOTE

SMS Central requires a response from your server to the HTTP GET.

Please ensure that you respond to the HTTP GET with a "HTTP/1.1 200 OK" HTTP status code in the header of your response and please ensure that you output a "0" in the content of your response to the HTTP GET.

SMS Central also provides the option to retry the forwarding of the message to your URL, in the event that your server may be having issues or experiencing downtime.

Within the 'Rules & Triggers' section of the SMS Central portal, you're able to enable the retry logic when setting up your forwarding rules.

In doing so, in the event that the nominated URL does not provide an adequate response or is experiencing downtime, SMS Central will continually retry the forwarding to your URL up to 10 times, with an increased delay of 5 minutes between each attempt i.e:

attempt 1: 2014-01-02 10:15:00
attempt 2: 2014-01-02 10:20:00
attempt 3: 2014-01-02 10:30:00
attempt 4: 2014-01-02 10:45:00
attempt 5: 2014-01-02 11:05:00
attempt 6: 2014-01-02 11:30:00
attempt 7: 2014-01-02 12:00:00



api

attempt 8: 2014-01-02 12:35:00

attempt 9: 2014-01-02 13:15:00

attempt 10: 2014-01-02 14:00:00

HTTP GET Parameters

When forwarding received SMS messages to the URL you have specified with a HTTP GET, SMS Central will provide the following parameters:

Parameter	Type	Possible/ Example Value(s)
USER_NAME	String	
PASSWORD	String	
ORIGINATOR	String	Mobile number (in international format): eg 61435748922
RECIPIENT	String	
PROVIDER	String	"telstra" , "optus" , "vodafone" , "virgin" , "hutc3g"
CAMPAIGN	String	
REFERENCE	String	
MESSAGE_TEXT	String	
UDH	OCTET Encoded User Data Header	
BINARY	OCTET Encoded Binary Data	
DCS	Integer	0, 1, 3, 8
VALUE	String	
ID	String	

HTTP GET Parameter Definitions

USER_NAME

- **Type:** string
- **Description:** This is the Username you set for the URL in Rules & Triggers i.e. the username required for Basic Authentication.

PASSWORD

- **Type:** string
- **Description:** This is the Password you set for the URL in Rules & Triggers i.e. the password required for Basic Authentication.

ORIGINATOR

- **Type:** string
- **Description:** This is the mobile number of the handset that has sent in the SMS message.

* In international format, an example of an Australian originator value is:
[61435800957](#)

RECIPIENT

- **Type:** string
- **Description:** This is the number that the message was being sent to. This would either be your dedicated number, or a shared virtual mobile number provided by SMS Central.

* An example Australian recipient value: [61435800957](#)

CAMPAIGN

- **Type:** string
- **Description:** When the message you are receiving is found to be a reply to an SMS message you have sent, if you provided a 'CAMPAIGN' value in that SMS message you sent, this CAMPAIGN parameter will provide that value, allowing you to match received messages to the campaign.

REFERENCE

- **Type:** string
- **Description:** When the message you are receiving is found to be a reply to an SMS message you have sent, if you provided a unique 'REFERENCE' value in that SMS message you sent, this REFERENCE parameter will provide that value, allowing you to match the received message as a reply to that specific SMS message you sent.

MESSAGE_TEXT

- **Type:** string
- **Description:** This will contain up to 160 characters forming the text of the SMS message received. Please note, in the case of a multi-part message, each part of the message may be received as an individual SMS, therefore containing up to 153 characters for each individual part.

UDH

- **Type:** OCTET Encoded User Data Header
- **Description:** If there is no value in the 'MESSAGE_TEXT' parameter and the received message is longer than 160 characters long, the UDH parameter will be provided containing an Octet-encoded user data header. In this case, the message will be split into 153 character parts, provided in the 'BINARY' parameter. This UDH value provides information about the message, including which part of the message this is.

In the case where the message contains non-GSM characters, the standard message length can contain up to 70 characters. In this case, each multi-part message will contain 67 characters.

An example value is: 0500038A0302

- The first 3 octets will always be "050003".

- The next octet ("8A" in the example) identifies the group. However, since we will pass your reference back to you in the delivery receipt, this octet will probably be useful for double checking only.

- The next octet ("03" in the example) specifies the total number of parts in the group.
- The last octet ("02" in the example) specifies the particular part of this message/delivery receipt. In this case, the delivery receipt would be for the second part of a 3-part message.

In order to check that the full message is delivered, you will need to look at the last 2 octets to check how many parts a message has, and check that you received a positive delivery for each part.

BINARY

- **Type:** OCTET Encoded Binary Data
- **Description:** If there is no value in the 'MESSAGE_TEXT' parameter and the received message is longer than 160 characters long, or the message contains characters not within the standard GSM character set, the BINARY parameter will be provided containing an Octet-encoded binary data. Where the message contains only standard GSM characters, In this case, the message will be split into 153 character parts, with each part of the message providing a User Data Header (UDH) PARAMETER value and the content of the message part provided in the BINARY parameter. In the case where the message contains non-GSM characters, it will be split into 70 character parts for a single message or 67 character parts for a multi-part message, with each part of the message providing a User Data Header (UDH) which provides information for determining which part of the multi-part message you have just received.

DCS

- **Type:** integer
- **Description:** The Data Coding Scheme (DCS) value represents the character set with which the 'BINARY' value should be decoded. A value of 8 should be treated as UCS-2 as it will contain characters outside of the ASCII/GSM character set. Any other value should be treated as ASCII/Latin-1.

VALUE

- **Type:** string
- **Description:** If the received message is a MO-billed Premium SMS message (the user sending the message will pay a fee, charged to their mobile phone bill, for

sending the message), the VALUE parameter will contain the amount billed to the handset, in cents.

* For example, if the charge is 55cents, the value for the VALUE parameter would be "55" .

ID

- **Type:** string
- **Description:** This is an internal reference ID for this message, provided by SMS Central. This is used for internal SMS Central tracking and you may quote this ID in support requests. You may also use this unique ID as a possible way to double check whether this particular inbound message has already been forwarded on to your URL.

Receiving a Delivery Receipt (DLR)

OVERVIEW

When you send an SMS (pg. 5), once your message has been sent to the Mobile Network Operator (also known as a carrier), SMS Central shortly thereafter receives a delivery receipt from that carrier.

These delivery receipts provide a final status of the message delivery, indicating whether the message was successfully delivered, or whether the message failed.

You can also elect to receive the Delivery Receipt to a URL you specify (HTTPS URLs are recommended). This is done with a HTTP GET, pushing the delivery receipt values to your URL in real-time, as we receive the delivery receipt.

SMS Central also provides the option to retry the forwarding of the delivery receipt to your URL, in the event that your server may be having issues or experiencing downtime.

Within the 'Rules & Triggers' section of the SMS Central portal, you're able to enable the retry logic when setting up your forwarding rules. In doing so, in the event that the nominated URL does not provide an adequate response or is experiencing downtime, SMS Central will continually retry the forwarding to your URL up to 10 times, with an increased delay of 5 minutes between each attempt. For example:

```
attempt 1: 2014-01-02 10:15:00
attempt 2: 2014-01-02 10:20:00
attempt 3: 2014-01-02 10:30:00
attempt 4: 2014-01-02 10:45:00
attempt 5: 2014-01-02 11:05:00
attempt 6: 2014-01-02 11:30:00
attempt 7: 2014-01-02 12:00:00
attempt 8: 2014-01-02 12:35:00
```



attempt 9: 2014-01-02 13:15:00

attempt 10: 2014-01-02 14:00:00

HTTP GET Parameters

When forwarding delivery receipts to the URL you have specified with a HTTP GET, the following parameters will be provided:

Parameter	Type	Possible/ Example Value(s)
USER_NAME	String	
PASSWORD	String	
ORIGINATOR	String	Mobile number (in international format)
RECIPIENT	String	
PROVIDER	String	"telstra" , "optus" , "vodafone" , "virgin" , "hutc3g"
MESSAGE_TEXT	String	
ID	String	
REFERENCE	String	
RESULT	Integer	See Success & Error Response Codes (pg. 41)
UDH	String	Example: 0500038A0302 (see definition below)
STATUS	String	DELIVRD, BUFFRED, FAILED
STATUSDESCRIPTION	String	A textual description of the reason for failure (if failure).

HTTP GET Parameter Definitions

USER_NAME

- **Type:** string
- **Description:** This is the Username you set for the URL in Rules & Triggers i.e. the username required for Basic Authentication.

PASSWORD

- **Type:** string
- **Description:** This is the Username you set for the URL in Rules & Triggers i.e. the username required for Basic Authentication.

ORIGINATOR

- **Type:** string
- **Description:** This is the phone number, in international format, of the originator or sender of the message. This is the number you sent the message 'from' .

RECIPIENT

- **Type:** string
- **Description:** This is the phone number, in international format, of the recipient of the message. This is the number you send the message 'to' .
*An example Australian recipient value: 61435800957

PROVIDER

- **Type:** string
- **Description:** This is a text representation identifying the Mobile Network Operator (carrier). This value will be 'default' (string, without single quotes) for standard SMS messages.

MESSAGE_TEXT

- **Type:** string
- **Description:** This is the text of the message you send to the recipient.

ID

- **Type:** string
- **Description:** This is an internal reference ID for this message, provided by SMS Central. This is used for internal SMS Central tracking and you may quote this ID in support requests.

REFERENCE

- **Type:** string
- **Description:** This is the (optional) unique REFERENCE value for the message that you provided when sending an SMS. This allows you to match the delivery receipt to the message you sent.

RESULT

- **Type:** string
- **Description:** This is a numerical result of the delivery status, providing a final confirmation value of the delivery or failure of the message. Please see Success and Error Response Codes (pg. 51) for possible values.

UDH

- **Type:** string
- **Description:** This is the standard User Data Header used in GSM to identify different parts of a multipart message. The value consists of 6 octets of hex digits. An example value is: 0500038A0302
 - The first 3 octets will always be "050003".
 - The next octet ("8A" in the example) identifies the group. However, since we will pass your reference back to you in the delivery receipt, this octet will probably be useful for double checking only.
 - The next octet ("03" in the example) specifies the total number of parts in the group.
 - The last octet ("02" in the example) specifies the particular part of this message/delivery receipt. In this case, the delivery receipt would be for the second part of a 3-part message.

In order to check that the full message is delivered, you will need to look at the last 2 octets to check how many parts a message has, and check that you received a positive delivery for each part.

STATUS

- **Type:** string
- **Description:** This value indicates the delivery status of the message and may include values such as:
 - DELIVERD – this indicates the message, or this part of the message has been delivered
 - BUFFRED – this indicates the message has been sent to the carrier and is currently undelivered by the carrier with a temporary failure result (i.e phone may be out of range)
 - FAILED – this indicates that the carrier was unable to deliver the message

STATUSDESCRIPTION

- **Type:** string
- **Description:** This field provides a verbose string representation of the reason for the message failure. You may see values such as:
 - “0001: Phone related” – Indicates the delivery failure is a handset issue
 - “0008: Message expired within the carrier and failure reason unknown”
 - “0002: Deliverer related: message within carrier”
 - “0004: Delivered to mobile service” (when the errorcode is DELIVRD)

Check Delivery Status of a Message

OVERVIEW

SMS Central provides you the ability to check for the delivery status of an individual outbound message or messages.

You can do this by providing the RECIPIENT mobile number OR the unique REFERENCE value provided when sending the message.

SMS Central recommends using the POST method for this API request.

Delivery Status Check - API URL

All HTTP POSTS sent to SMS Central's API to check status must go to the following URL:

<https://my.smscentral.com.au/api/v3.2/checkstatus>

HTTP POST Parameters

The parameters below are the POST parameters that can be provided, in order to retrieve your account balance.

Please ensure these parameters are submitted in uppercase.

Parameter	Required	Type	Possible/ Example Value(s)
USERNAME	Yes	String	
PASSWORD	Yes	String	
RECIPIENT	No	String	61435800957
REFERENCE	No	String	

Note: Either the 'RECIPIENT' OR the 'REFERENCE' parameters must be provided.

HTTP POST Parameter Definitions

USERNAME

- **Type:** string
- **Description:** This is your Username for your SMS Central account.

PASSWORD

- **Type:** string
- **Description:** This is your Password for your SMS Central account

RECIPIENT

- **Type:** string
- **Description:** This is the recipient value of the number you sent the message to. By providing a value in this parameter (as opposed to using the REFERENCE parameter) you will receive the status value for up to the 10 most recent messages (in the current month) that you have sent to that customer. Note, you can supply a comma separated list of values, of up to 10 mobile numbers, for the RECIPIENT parameter. This will return the status of (up to) the last 10 messages in the current month, for each of those recipient values.

REFERENCE

- **Type:** string
- **Description:** This is the unique reference value you provided when sending the original outbound message. This allows you to check the status for an individual message. Note, you can supply a comma separated list of values, of up to 100 values, for the REFERENCE parameter. This will return the status of all 100 messages matching to the unique reference values provided.

Response To Your HTTP POST

After sending your HTTP POST with the required parameters to the API URL for checking the delivery status of your message, SMS Central will return a "HTTP/1.1 200 OK" HTTP status code in the header of the response to your HTTP POST and XML-formatted content providing the details of the status of the message(s).

If you are providing the 'REFERENCE' value, then a single XML <message> tag will be provided for the individual message matching that REFERENCE. If you provide a comma-separated list of (up to 100) REFERENCE values, a single '<message>' block will be returned for each individual message matching each of the REFERENCE values provided.

If you are providing the 'RECIPIENT' value, then you will receive up to 10 <message> blocks, containing the 10 most recent messages sent to that recipient within the current month. If you provide a comma-separated list of (up to 10) RECIPIENT values, you will be returned up to 10 <message> blocks for each of the RECIPIENT values, showing the 10 most recent messages to those recipient values within the current month.

The XML Response will be formatted as per the following tags and XML structure. The following contains example values:

```
<messages>
  <message>
    <reference>ABC123</reference>
    <timestamp>2015-01-01 15:00:00</timestamp>
    <recipient>61400000000</recipient>
    <status>DELIVRD</status>
    <statusdescription>0004: Delivered to mobile service
  </statusdescription>
  </message>
  <message>
    <reference>ABC456</reference>
    <timestamp>2015-01-02 16:00:00</timestamp>
    <recipient>61400000001</recipient>
    <statusdescription>0004: Delivered to mobile service
  </statusdescription>
  </message>
</messages>
```

Create a Sub-Account

OVERVIEW

As of API V3.0, SMS Central provides you the ability to create sub-accounts via the API as well as add users to that sub-account. To do this, you would send a HTTP GET or POST to the API URL.

Of course, you can also log into your SMS Central account to create sub-accounts via the web portal

Note: This functionality requires your account to be configured as a partner account. Please contact SMS Central's support team for this to be enabled.

Account - API URL

All HTTP POSTS to SMS Central's API to create a subaccount must go to the following URL:

```
https://my.smscentral.com.au/api/v3.2/account/create
```

HTTP POST Parameters

The parameters below are the POST parameters that can be provided, in order to create a subaccount.

Please ensure these parameters are submitted in uppercase.

Parameter	Required	Type	Possible/Example Value(s)
USERNAME	Yes	String	
PASSWORD	Yes	String	
ACCOUNT	YES	JSON-format	See POST Parameter definition

POST Parameter Definitions

USERNAME

- **Type:** string
- **Description:** This is your Username for your SMS Central account.

PASSWORD

- **Type:** string
- **Description:** This is your Password for your SMS Central account

ACCOUNT

- **Type:** JSON-formatted string
- **Description:** This is where you will provide the details and settings, in JSON-format, of the account being created.

JSON Field	Required	Value Type & Possible/Example Value(s)
ACCOUNTNAME	Yes	String containing the name of the account to be created
PARENTACCOUNT	No	Integer value containing the ID of the parent account. See further notes
PAYMENTMODEL	Yes	String value 'prepaid' or 'postpaid' denoting a Pre-paid or Post-paid subaccount, respectively. See further notes
MONTHLYVOLUMELIMIT	No	Integer with monthly volume limit, leaving this out means no limit will be applied

INVOICINGACCOUNT	No	ID of the account that should be invoiced for usage. Leave out to invoice to account being created. See further notes.
------------------	----	--

ACCOUNT – JSON Further Notes

PARENTACCOUNT

If you wish to create a multi-level hierarchical structure, you may do so by providing the ID of the account which should be the ‘parent’ (one level higher in hierarchical structure) of the account being created. As this is an optional field, if you do not provide a value or do not provide this field at all, the account (for which you are submitting your username & password) will be determined as the ‘Parent’ account of the subaccount being created.

Note: You will have the ID an account by having created that account via the API first, saving the Account ID provided in the response. See “Response to your HTTP POST” .

MONTHLYVOLUMELIMIT

You are able to set a monthly volume limit on the account you are creating by providing an integer value to this JSON field. If you do not provide this field in the JSON object, no monthly volume limit will be set for the account being created.

PAYMENTMODEL

This allows you to determine whether the account you are creating will require credits to be purchased (and/or allocated via the UI) up-front (making them prepaid) or whether they are able to send SMS messages without restriction and will be invoiced by SMS Central for the usage at the end of the month (making them ‘postpaid’). If your account (the top-level account in the hierarchical structure) is a

pre-paid account, then setting a 'postpaid' subaccount will mean messages sent from that subaccount will automatically deduct from the pre-paid balance of your parent account.

INVOICINGACCOUNT

You are able to determine your billing & invoicing structure by providing the ID of the account that should be receiving the invoices for the usage of the subaccount being created. You may provide the ID of the immediate parent account, or the ID of an account higher up the hierarchy, or you may leave this value out entirely in which case the account being created will be determined as the account being invoiced for its own usage.

Note: This only applies if the sub-account being created will have a 'postpaid' value for its PAYMENTMODEL.

Response to Your HTTP POST

After sending your HTTP POST with the required parameters to the Account API URL for creating a sub-account, SMS Central will return a "HTTP/1.1 200 OK" HTTP status code in the header of the response to your HTTP POST and JSON-formatted content providing your details of the account created.

The JSON Response will be of the following format:

On Success

```
{  "account" :
  {
    "name" : "String value containing the name of the account, as per your
              ACCOUNTNAME value" ,
    "ID" : [integer value containing the unique ID of this account]
  }
}
```

On Failure

```
{  "error" :  
    "errorcode" : [integer value containing an error code unique to error type],  
    "errordescription" : "String description of the error that has occurred"  
}
```

Create a User

OVERVIEW

As of API V3.0, SMS Central provides you the ability to create users via the API as well as assign those users to any number of accounts within your account hierarchy. To do this, you would send a HTTP GET or POST to the API URL.

Of course, you can also log into your SMS Central account to create users via the web portal

Note: This functionality requires your account to be configured as a partner account. Please contact SMS Central's support team for this to be enabled.

User - API URL

All HTTP POSTS to SMS Central's API to create a user must go to the following URL:

```
https://my.smscentral.com.au/api/v3.2/user/create
```

HTTP POST Parameters

The parameters below are the POST parameters that can be provided, in order to create a user.

Please ensure these parameters are submitted in uppercase.

Parameter	Required	Type	Possible/Example Value(s)
USERNAME	Yes	String	
PASSWORD	Yes	String	
USER	YES	JSON-format	See POST Parameter definition

POST Parameter Definitions

USERNAME

- **Type:** string
- **Description:** This is your Username for your SMS Central account.

PASSWORD

- **Type:** string
- **Description:** This is your Password for your SMS Central account

USER

- **Type:** JSON-formatted string
- **Description:** This is where you will provide the details of the user being created and the account(s) they are to be assigned to.

JSON Field	Required	Value Type & Possible/Example Value(s)
USERNAME	Yes	String containing the Username for the user being created
PASSWORD	Yes	String containing the Password for the user being created
FIRSTNAME	Yes	String containing first name of the user being created
LASTNAME	No	String containing last name of the user being created
EMAILADDRESS	No	String containing the email address of the user being created
MOBILENUMBER	No	String containing the mobile number (in international format) of the user being created

ACCOUNTS	Yes	JSON-formatted array of accounts (and user level) the user should be assigned to. See further notes
----------	-----	---

USER – JSON Further Notes

ACCOUNTS

In order to assign the user being created into account(s), you must provide an array of JSON-formatted account objects with the "ID" integer value of the account (refer to the Account API and the integer value of the "Userlevel"). You must provide 1 or more accounts.

Note: Userlevel correlates with the user levels in the web portal with integer values as follows:

- Administrator user, integer value 1
- Advanced user, integer value 2
- Standard user, integer value 3

This would be in the format described below

```
{
  account: {
    "ID" : [Integer value of the account ID],
    "Userlevel" : [integer value 1, 2 or 3 for "admin" "advanced"
and "standard respectively] },
  account: {
    "ID" : [Integer value of the account ID],
    "Userlevel" : [integer value 1, 2 or 3 for "admin" "advanced"
and "standard respectively] },
  account: {
```

```
    "ID" : [Integer value of the account ID],  
    "Userlevel" : [integer value 1, 2 or 3 for "admin" "advanced"  
and "standard respectively] },  
}
```

Response To Your HTTP POST

After sending your HTTP POST with the required parameters to the User API URL for creating a user, SMS Central will return a "HTTP/1.1 200 OK" HTTP status code in the header of the response to your HTTP POST and JSON-formatted content providing your details of the user created.

The JSON Response will be of the following format:

On Success

```
{  "user" :
  "username" : "String value containing the username of the created user, as per
                your USERNAME value in the JSON-formatted 'user' object" ,
  "ID" : [integer value containing the unique ID of this user]
}
```

On Failure

```
{  "error" :
  "errorcode" : [integer value containing an error code unique to error type],
  "errordescription" : "String description of the error that has occurred"
}
```



Check Your Account Balance

OVERVIEW

SMS Central provides you the ability to check your account balance via the API. To do this, you would send a HTTP GET or POST to the API URL.

Of course, you can also log into your SMS Central account to check your account balance.

HTTP POST Parameters

The parameters below are the POST parameters that can be provided, in order to retrieve your account balance.

Please ensure these parameters are submitted in uppercase.

Parameter	Required	Type	Possible/ Example Value(s)
USERNAME	Yes	String	
PASSWORD	Yes	String	
ACTION	Yes	String	'balance'

POST Parameter Definitions

USERNAME

- **Type:** string
- **Description:** This is your Username for your SMS Central account.

PASSWORD

- **Type:** string
- **Description:** This is your Password for your SMS Central account

ACTION

- **Type:** string
- **Description:** To check your account balance, please provide the string value 'balance' (without single quotes) as this POST parameter's value.

Response To Your HTTP POST

After sending your HTTP POST with the required parameters to the API URL for checking your account balance, SMS Central will return a "HTTP/1.1 200 OK" HTTP status code in the header of the response to your HTTP POST and XML-formatted content providing your account balance.

The XML Response will contain a single <balance> tag, containing your account balance.

An example XML Response showing a remaining account balance of \$125.34 is:

```
<balance>125.34</balance>
```

Success/ Error Response Codes

The following Success/Error response codes relate to the response you will receive by SMS Central to your HTTP POST when sending an SMS, as well as the response codes provided when Receiving a Delivery Receipt SMS (pg. 30).

WHEN SENDING AN SMS MESSAGE

The response code and description will be provided as a string (separated by whitespace) in the content output to your HTTP POST. An example response is as follows:

```
1 Success; Message has been successfully delivered
```

The HTTP header response will always be 200 (OK). Please see Send an SMS (pg. 5) for details.

WHEN RECEIVING A DELIVERY RECEIPT

The response code (numerical value) based on the list below will be provided in the 'RESULT' parameter. Please see 'Receiving a Delivery Receipt' for details.

RESPONSE CODES

Parameter	Possible/ Example Value(s)
0	Success; Message has been sent, delivery confirmation is pending
1	Success; Message has been successfully delivered
500	Failed; Internal SMS Central Server Error
503	Failed; Message has expired within the carrier
511	Rejected; Your username or password is incorrect
513	Rejected; Message rejected due to being a duplicate message
514	Rejected; No 'RECIPIENT' value was provided
519	Rejected; The number you tried sending to is blacklisted
531	Rejected; Your message contains no content
532	Rejected; Your message content has invalid characters. Please check your message text
534	Rejected; Credit limit exceeded. You have run out of SMS credit
535	Rejected: Your 'ORIGINATOR' value is not valid



536	Notice; Your message has been temporarily delayed
-----	---

550	Failed: Message could not be delivered by carrier
-----	---

API Support

EMAIL

If you have any questions or want to learn more, get in touch with our Technical Support Team. You can address your email to our team at support@smscentral.com.au.

Make sure you add API to the Subject of your email!

CONTINUAL IMPROVEMENT...

We are working really hard to improve SMS Central API. As part of this project, we will be adding new support tools shortly.

Thanks for your patience!