1. Introduction

The CloudService API provides a simple HTTP-based interface for sending SMS messages. It is designed to be compatible with existing CloudService ZM integrations, allowing you to send single or bulk messages with minimal code changes.

Two endpoints are available:

Endpoint	Method	Format	Use Case
/smsservice/httpapi	GET / POST	Query parameters	Simple single-message sends
/smsservice/jsonapi	POST	JSON body	Bulk messaging, advanced integration

2. Authentication

Both endpoints support two authentication methods. You may use either one:

Option A: Username & Password

Pass your account email and password directly in the request parameters.

Parameter	Type	Description
username	string	Your account email address
password	string	Your account password

Option B: API Key

Pass an API key generated from the platform's API Keys page. Either the `client_access_id` or `client_secret_id` can be used.

Parameter	Type	Description
api_key	string	Your API key (client_access_id or client_secret_id)

> **Recommendation:** Use API keys for production integrations. They can be rotated independently without changing your account password.

3. HTTP API

GET POST `/smsservice/httpapi`

Send a single SMS message using query string parameters.

Parameters

Parameter	Type	Required	Description
username	string	No*	Account email (required if no api_key)

password	string	No*	Account password (required if no api_key)
api_key	string	No*	API key (required if no username/password)
phone	string	Yes	Recipient phone number (Zambian)
msg	string	Yes	Message text
sender_id	string	No	Custom sender ID — max 11 characters, must be approved
shortcode	string	No	Shortcode (alternative to sender_id)

* You must provide either api_key OR username+password.

PHP Example

```
<?php
// Using API key authentication
$params = http_build_query([
    'api_key'   => 'your_api_key_here',
    'phone'     => '260971234567',
    'msg'       => 'Hello from PHP!',
    'sender_id' => 'MyApp'
]);

$url = "https://bulksms.ontech.co.zm/smsservice/httpapi?" . $params;
$response = file_get_contents($url);
$result = json_decode($response, true);

if ($result['status'] == 100) {
    echo "Message sent successfully!";
} else {
    echo "Error: " . $result['message'];
}
?>
```

Python Example

```
import requests

# Using API key
resp = requests.get("https://bulksms.ontech.co.zm/smsservice/httpapi", params={
    "api_key": "your_api_key_here",
    "phone": "260971234567",
    "msg": "Hello from Python!",
    "sender_id": "MyApp"
})
result = resp.json()
print(f"Status: {result['status']}, Message: {result['message']}")
```


Ontech Solutions Limited — Confidential

4. JSON API

POST `/smsservice/jsonapi`

Send one or more SMS messages using a JSON request body. Ideal for bulk sends.

Request Body

```
{
    "auth": {
        "api_key": "your_api_key_here"
    },
    "messages": [
        {"phone": "260971234567", "message": "Hello User 1!"},
        {"phone": "260961234567", "message": "Hello User 2!"},
        {"phone": "260771234567", "message": "Hello User 3!"}
    ]
}
```

Alternatively, using username/password authentication:

```
{
    "auth": {
        "username": "user@example.com",
        "password": "your_password",
        "sender_id": "MyApp"
    },
    "messages": [
        {"phone": "260971234567", "message": "Hello!"}
    ]
}
```

Auth Object Fields

Field	Type	Required	Description
api_key	string	No*	API key (if not using username/password)
username	string	No*	Account email
password	string	No*	Account password
sender_id	string	No	Custom sender ID — max 11 characters, must be approved
short_code	string	No	Shortcode

Message Object Fields

Field	Type	Required	Description
phone	string	Yes	Recipient phone number (Zambian)
message	string	Yes	SMS message text

Ontech Solutions Limited — Confidential

PHP Example

```php
<?php
$data = json_encode([
    'auth' => ['api_key' => 'your_api_key_here'],
    'messages' => [
        ['phone' => '260971234567', 'message' => 'Bulk message 1'],
        ['phone' => '260961234567', 'message' => 'Bulk message 2'],
    ]
]);

$ch = curl_init('https://bulksms.ontech.co.zm/smsservice/jsonapi');
curl_setopt($ch, CURLOPT_POST, true);
curl_setopt($ch, CURLOPT_POSTFIELDS, $data);
curl_setopt($ch, CURLOPT_HTTPHEADER, ['Content-Type: application/json']);
curl_setopt($ch, CURLOPT_RETURNTRANSFER, true);

$response = curl_exec($ch);
curl_close($ch);

$result = json_decode($response, true);
echo "Status: " . $result['status'] . ", Sent: " . $result['sent'];
?>
```

Python Example

```python
import requests

resp = requests.post("https://bulksms.ontech.co.zm/smsservice/jsonapi", json={
    "auth": {"api_key": "your_api_key_here"},
    "messages": [
        {"phone": "260971234567", "message": "Bulk message 1"},
        {"phone": "260961234567", "message": "Bulk message 2"},
    ]
})
result = resp.json()
print(f"Status: {result['status']}, Sent: {result.get('sent', 0)}")
```


Ontech Solutions Limited — Confidential

5. Status Codes

Code	Meaning	Description
100	Success	Message(s) queued for delivery successfully
101	Insufficient Credit	Your account does not have enough SMS credits. Top up your balance.
102	Invalid User	Authentication failed. Check your credentials or API key.
103	Other Error	General error (missing fields, invalid phone number, etc.). Check the `message` field for details.

Response Format

HTTP API returns:

```
{"status": 100, "message": "Success"}
```

JSON API returns (on success):

```
{
    "status": 100,
    "message": "Success",
    "sent": 3,
    "errors": [
        {"index": 2, "phone": "+1555...", "error": "Not a Zambian number"}
    ]
}
```

6. Phone Number Format

All phone numbers must be Zambian numbers. The system accepts various formats and normalizes them:

Input	Normalized	Valid
0971234567	260971234567	Yes
+260971234567	260971234567	Yes
260971234567	260971234567	Yes
+1555123456	1555123456	No (not Zambian)

7. Error Handling

> **Important:** Always check the `status` field in the response. A status of `100` indicates success. Any other code means the message was not sent.

- **102 errors:** Verify your API key is active, or check username/password.
- **101 errors:** Top up your credit balance before retrying.
- **103 errors:** Read the `message` field for specific details on what went wrong.

Ontech Solutions Limited — Confidential

Need help? Contact Ontech Solutions Limited for technical support and integration assistance.

Ontech Solutions Limited — Confidential