Selcom Payment Gateway — API Documentation

Version: 2.0.0 **Base URL:** `https://your-domain.com/api` **Content-Type:** `application/json`

Table of Contents

---

1. Authentication

Every request must include the following **4 headers**. There is no login step — each request is self-authenticated using a cryptographic signature.

Header	Value
`Authorization`	`SELCOM <base64(client_key)>`
`Digest-Method`	`RS256`
`Timestamp`	Current UTC time in ISO 8601 format
`Digest`	Cryptographic signature of the timestamp (see below)

RS256 — RSA Asymmetric Signing

Uses **RSA asymmetric cryptography**. You sign requests with your **private key**. We verify using your **public key** stored on our server. Your private key is never shared or transmitted — even if our server is compromised, your key remains safe.

How it works:

```
Your Private Key  →  RSA-SHA256( "timestamp=<value>" )  →  Digest header
Our Server        →  verify( Digest, Your Public Key )  →  PASS / FAIL
```

Step 1 — You will receive your credentials:

```
client_key:  client-vSDOkTzbufJv
private_key: [2048-bit RSA Private Key PEM — keep this secret, never share]
```

Step 2 — Get the current UTC timestamp in ISO 8601 format:

```
2026-05-06T18:30:00+00:00
```

Step 3 — Build the signature string:

```
timestamp=2026-05-06T18:30:00+00:00
```

Step 4 — Sign with your RSA private key (SHA-256) and Base64-encode:

```
Digest = base64( RSA-SHA256( signatureString, your_private_key ) )
```

Step 5 — Send these 4 headers on every request:

```
Authorization:  SELCOM Y2xpZW50LXZTRk9rVHpidWZKdg==
Digest-Method:  RS256
Timestamp:      2026-05-06T18:30:00+00:00
Digest:         <base64 encoded RSA-SHA256 signature>
```

> ***Security rules:***
>
> - *Timestamp must be within **±5 minutes** of server time (UTC). Requests outside this window are automatically rejected to prevent replay attacks.*
> - *Always use **UTC** for the timestamp.*
> - *Never share your RSA private key with anyone.*

---

2. Create Order

Creates a full checkout order with complete buyer information.

Endpoint

```
POST /api/checkout/create-order
```

Request Body

Field	Type	Required	Description
`transid`	string	Yes	Your unique transaction ID (max 100 chars)
`order_id`	string	Yes	Your unique order reference (max 100 chars)
`amount`	number	Yes	Amount to charge (must be >= 1)
`currency`	string	Yes	3-letter currency code e.g. `TZS`, `USD`
`vendor`	string	Yes	Your Selcom vendor ID
`buyer_name`	string	Yes	Full name of the buyer
`buyer_email`	string	Yes	Email address of the buyer
`buyer_phone`	string	Yes	Phone number e.g. `255712345678`
`no_of_items`	integer	Yes	Number of items in the order (min 1)
`redirect_url`	string	Yes	URL to redirect buyer after payment
`cancel_url`	string	No	URL to redirect buyer if they cancel

`webhook`	string	No	URL to receive payment notification callback

Example Request

```
POST /api/checkout/create-order
Authorization: SELCOM Y2xpZW50LXZTRk9rVHpidWZKdg==
Digest-Method: RS256
Timestamp: 2026-05-06T18:30:00+00:00
Digest: <RSA-SHA256 signature>
Content-Type: application/json

{
    "transid":      "TXN-20260506-001",
    "order_id":     "ORDER-20260506-001",
    "amount":       50000,
    "currency":     "TZS",
    "vendor":       "VENDOR123",
    "buyer_name":   "John Doe",
    "buyer_email":  "john@example.com",
    "buyer_phone":  "255712345678",
    "no_of_items":  1,
    "redirect_url": "https://yoursite.com/payment/success",
    "cancel_url":   "https://yoursite.com/payment/cancel",
    "webhook":      "https://yoursite.com/payment/webhook"
}
```

Success Response `200 OK`

```
{
    "result":  "SUCCESS",
    "message": "Request successful",
    "data": {
        "order_id":           "ORDER-20260506-001",
        "payment_gateway_url": "https://checkout.selcommobile.com/pay/abc123"
    }
}
```

---

3. Create Order Minimal

Creates a checkout order with minimum required fields.

Endpoint

```
POST /api/checkout/create-order-minimal
```

Request Body

Field	Type	Required	Description

`transid`	string	Yes	Your unique transaction ID (max 100 chars)
`order_id`	string	Yes	Your unique order reference (max 100 chars)
`amount`	number	Yes	Amount to charge (must be >= 1)
`currency`	string	Yes	3-letter currency code e.g. `TZS`, `USD`
`vendor`	string	Yes	Your Selcom vendor ID
`buyer_phone`	string	Yes	Phone number e.g. `255712345678`
`no_of_items`	integer	Yes	Number of items in the order (min 1)
`redirect_url`	string	Yes	URL to redirect buyer after payment

Example Request

```
POST /api/checkout/create-order-minimal
Authorization: SELCOM Y2xpZW50LXZTRk9rVHpidWZKdg==
Digest-Method: RS256
Timestamp: 2026-05-06T18:30:00+00:00
Digest: <RSA-SHA256 signature>
Content-Type: application/json

{
    "transid":      "TXN-20260506-002",
    "order_id":     "ORDER-20260506-002",
    "amount":       10000,
    "currency":     "TZS",
    "vendor":       "VENDOR123",
    "buyer_phone":  "255712345678",
    "no_of_items":  1,
    "redirect_url": "https://yoursite.com/payment/success"
}
```

Success Response `200 OK`

```
{
    "result":  "SUCCESS",
    "message": "Request successful",
    "data": {
        "order_id":           "ORDER-20260506-002",
        "payment_gateway_url": "https://checkout.selcommobile.com/pay/xyz789"
    }
}
```

---

4. Cancel Order

Cancels a previously created order.

Endpoint

```
DELETE /api/checkout/cancel-order
```

Request Body

Field	Type	Required	Description
`transid`	string	Yes	The transaction ID used when creating the order
`order_id`	string	Yes	The order ID used when creating the order

Example Request

```
DELETE /api/checkout/cancel-order
Authorization: SELCOM Y2xpZW50LXZTRk9rVHpidWZKdg==
Digest-Method: RS256
Timestamp: 2026-05-06T18:35:00+00:00
Digest: <RSA-SHA256 signature>
Content-Type: application/json

{
    "transid":  "TXN-20260506-001",
    "order_id": "ORDER-20260506-001"
}
```

Success Response `200 OK`

```
{
    "result":  "SUCCESS",
    "message": "Order cancelled successfully"
}
```

---

5. Get Order Status

Retrieves the current payment status of an order.

Endpoint

```
GET /api/checkout/order-status
```

Query Parameters

Parameter	Type	Required	Description
`transid`	string	Yes	The transaction ID used when creating the order
`order_id`	string	No	The order ID used when creating the order

Example Request

```
GET /api/checkout/order-status?transid=TXN-20260506-001&order_id=ORDER-20260506-001
Authorization: SELCOM Y2xpZW50LXZTRk9rVHpidWZKdg==
Digest-Method: RS256
Timestamp: 2026-05-06T18:40:00+00:00
Digest: <RSA-SHA256 signature>
```

Success Response `200 OK`

```
{
    "result":  "SUCCESS",
    "message": "Request successful",
    "data": {
        "order_id":   "ORDER-20260506-001",
        "transid":    "TXN-20260506-001",
        "status":     "COMPLETED",
        "amount":     50000,
        "currency":   "TZS",
        "msisdn":     "255712345678",
        "created_at": "2026-05-06T18:30:00+00:00"
    }
}
```

---

6. Error Responses

Authentication Errors `401 Unauthorized`

```
{ "result": "FAIL", "message": "Missing required headers: Authorization, Digest-
Method, Digest, Timestamp." }
```

```
{ "result": "FAIL", "message": "Timestamp is outside the +-5-minute window (drift:
360s). Check your clock." }
```

```
{ "result": "FAIL", "message": "RS256 signature verification failed. Check your
private key and timestamp." }
```

Validation Errors `422 Unprocessable Entity`

```
{
    "result":  "FAIL",
    "message": "Validation failed.",
    "errors": {
        "amount":   ["The amount field is required."],
        "currency": ["The currency must be 3 characters."]
```

```
    }
}
```

---

7. Code Examples

PHP — RS256

```php
$clientKey     = 'your-client-key';
$privateKeyPem = file_get_contents('/path/to/your/private_key.pem');

$timestamp       = (new DateTime('now', new DateTimeZone('UTC')))->format('c');
$signatureString = "timestamp={$timestamp}";

openssl_sign($signatureString, $rawSignature, $privateKeyPem, OPENSSL_ALGO_SHA256);
$digest = base64_encode($rawSignature);

$headers = [
    'Authorization' => 'SELCOM ' . base64_encode($clientKey),
    'Digest-Method' => 'RS256',
    'Timestamp'     => $timestamp,
    'Digest'        => $digest,
    'Content-Type'  => 'application/json',
];

$response = Http::withHeaders($headers)->post(
    'https://your-domain.com/api/checkout/create-order-minimal',
    [
        'transid'      => 'TXN-001',
        'order_id'     => 'ORDER-001',
        'amount'       => 10000,
        'currency'     => 'TZS',
        'vendor'       => 'VENDOR123',
        'buyer_phone'  => '255712345678',
        'no_of_items'  => 1,
        'redirect_url' => 'https://yoursite.com/success',
    ]
);
```

JavaScript (Node.js) — RS256

```javascript
const crypto = require('crypto');
const fs     = require('fs');

const clientKey     = 'your-client-key';
const privateKeyPem = fs.readFileSync('/path/to/private_key.pem', 'utf8');

const timestamp       = new Date().toISOString();
const signatureString = `timestamp=${timestamp}`;
```

```
const sign   = crypto.createSign('SHA256');
sign.update(signatureString);
const digest = sign.sign(privateKeyPem, 'base64');

const headers = {
    'Authorization': 'SELCOM ' + Buffer.from(clientKey).toString('base64'),
    'Digest-Method': 'RS256',
    'Timestamp':     timestamp,
    'Digest':        digest,
    'Content-Type':  'application/json',
};
```

Python — RS256

```
import base64, datetime, requests
from cryptography.hazmat.primitives import hashes, serialization
from cryptography.hazmat.primitives.asymmetric import padding

client_key = 'your-client-key'
with open('/path/to/private_key.pem', 'rb') as f:
    private_key = serialization.load_pem_private_key(f.read(), password=None)

timestamp        = datetime.datetime.now(datetime.timezone.utc).isoformat()
signature_string = f'timestamp={timestamp}'.encode()

raw_signature = private_key.sign(signature_string, padding.PKCS1v15(),
hashes.SHA256())
digest = base64.b64encode(raw_signature).decode()

headers = {
    'Authorization': 'SELCOM ' + base64.b64encode(client_key.encode()).decode(),
    'Digest-Method': 'RS256',
    'Timestamp':     timestamp,
    'Digest':        digest,
    'Content-Type':  'application/json',
}
```

For credentials and RSA key provisioning, please contact your account manager.