Digipay Merchant Integration Instructions

Version 10.0 Last Revision: January 2026

Table of Contents

Introduction

The Digipay payment gateway allows merchants to accept credit card payments by redirecting customers to a secure Digipay-hosted payment page. After the customer completes payment, Digipay notifies your server via a postback (callback) and redirects the customer back to your site.

How It Works

1. Customer clicks "Pay" on your site
2. Your site redirects customer to Digipay with encrypted order details
3. Customer enters payment information on Digipay's secure page
4. Digipay processes the payment
5. Digipay sends a postback to your server with the result
6. Customer is redirected back to your site

Prerequisites

- **SSL Certificate**: Your site must use HTTPS
- **PHP 7.2+** with OpenSSL extension
- **Site ID**: Provided by Digipay after registration
- **Encryption Key**: Provided by Digipay (store securely, never commit to version control)

Information to Provide to Digipay

Contact tech@digipay.co with:

1. Your website URL
2. Your postback URL (e.g., `https://yoursite.com/digipay-postback.php`)
3. Your return URL (where customers go after payment)
4. Products/subscriptions you'll be selling

Digipay will provide:

- Your **Site ID**

Encryption Key

The shared encryption key for all Digipay integrations is:

```
fluidcastplgpaygowoo22
```

Store this securely in your configuration file. Never expose it in client-side code.

Quick Start Guide

Step 1: Create Your Postback Handler

Create a file called `digipay-postback.php` on your server:

```php
<?php
// Your postback handler - see "Complete Code Examples" section
```

Step 2: Store Your Encryption Key Securely

Add to your configuration file (e.g., `wp-config.php` for WordPress):

```php
define('DIGIPAY_ENCRYPTION_KEY', 'your-key-from-digipay');
```

IMPORTANT: Never hardcode the encryption key in your source code or commit it to version control.

Step 3: Redirect Customers to Digipay

Build the payment URL with your order details and redirect the customer.

Step 4: Test

Use Digipay's testing bed to verify your integration works.

Payment Flow Overview

```
┌─────────────┐    ┌─────────────┐    ┌─────────────┐
│  Your Site  │    │   Digipay   │    │  Postback   │
│ (Checkout)  │    │  (Payment)  │    │  Handler    │
└──────┬──────┘    └──────┬──────┘    └──────┬──────┘
       │                  │                  │
       │ 1. Redirect ───► │                  │
       │                  │                  │
       │                  │ 2. Customer pays │
       │                  │                  │
       │                  │ 3. Postback ───► │
       │                  │                  │
       │                  │ ◄─── 4. XML OK   │
```

```
     │                    │                    │
     │ ◄── 5. Redirect    │                    │
     │    (to tcomplete)  │                    │
     │                    │                    │
```

Sending Customers to Digipay

Endpoint

```
https://secure.digipay.co/order/creditcard/cc_form_enc.php
```

Required Parameters

Parameter	Type	Description
`site_id`	integer	Your Digipay Site ID
`charge_amount`	decimal	Amount to charge (e.g., `19.99`)
`type`	string	Transaction type. Use `purchase` for one-time payments
`order_description`	string	Description of the purchase (max 255 chars)
`session`	string	Your unique order/session ID. This is returned in the postback
`encrypt`	integer	Set to `1` to indicate encrypted request
`pburl`	URL	Your postback handler URL
`tcomplete`	URL	Where to redirect customer after successful payment

Optional Billing Parameters

These pre-fill the payment form for the customer:

Parameter	Type	Description
`first_name`	string	Customer's first name
`last_name`	string	Customer's last name
`email`	string	Customer's email address
`address`	string	Street address
`city`	string	City
`state`	string	State (2-letter code or full name)
`zip`	string	Postal code (no spaces)
`country`	string	2-letter country code (ISO 3166-1 alpha-2)

Optional Parameters

Parameter	Type	Description
`woocomerce`	integer	Set to `1` for WooCommerce integrations. **Note: This parameter is intentionally spelled without the second 'm'**
`shipped`	integer	Set to `0` for digital goods, `1` for physical goods
`trans`	integer	Customer's historical transaction count from your site (optional, for fraud prevention)

Example: Building the Payment URL

```php
$params = array(
    'site_id'           => 'YOUR_SITE_ID',
    'charge_amount'     => '29.99',
    'type'              => 'purchase',
    'order_description' => 'Premium Subscription',
    'session'           => 'ORDER-12345',
    'woocomerce'        => '1',
    'encrypt'           => '1',
    'first_name'        => 'John',
    'last_name'         => 'Smith',
    'email'             => 'john@example.com',
    'address'           => '123 Main St',
    'city'              => 'New York',
    'state'             => 'NY',
    'zip'               => '10001',
    'country'           => 'US',
    'shipped'           => '0',
    'pburl'             => 'https://yoursite.com/digipay-postback.php',
    'tcomplete'         => 'https://yoursite.com/thank-you/'
);

// Build and encrypt the URL (see Encryption section)
$payment_url = build_digipay_url($params);

// Redirect customer
header('Location: ' . $payment_url);
exit;
```

Postback Handler Setup

When a payment is completed, Digipay sends a notification to your `pburl`. Your handler must:

1. Receive the postback data
2. Validate and process the payment
3. Return an XML response

Postback Data Format

IMPORTANT: Digipay sends postback data as JSON embedded in a POST key. This requires special parsing:

```php
// Extract JSON from POST key
foreach ($_POST as $key => $value) {
    if (strpos($key, '{') === 0) {
        $data = json_decode($key, true);
        break;
    }
}
```

Postback Parameters Received

Parameter	Type	Description
`session`	string	Your order ID (passed in original request)
`amount`	string	Amount charged, formatted as `XX_XX` (e.g., `29_99` = $29.99). **Note: Uses underscore, not decimal point**
`description`	string	Order description with underscores instead of spaces
`email`	string	Customer email (may be null)
`last`	string	Customer last name (may be empty)

Amount Format Conversion

The amount is sent with an underscore instead of a decimal point. Convert it:

```php
$raw_amount = $data['amount'];  // e.g., "29_99"
$amount = str_replace('_', '.', $raw_amount);  // "29.99"
```

Required XML Responses

Success Response:

```xml
<?xml version="1.0" encoding="UTF-8"?>
<rsp stat="ok" version="1.0">
    <message id="100">Purchase successfully processed</message>
    <receipt>ORDER-12345</receipt>
</rsp>
```

Failure Response:

```xml
<?xml version="1.0" encoding="UTF-8"?>
<rsp stat="fail" version="1.0">
    <error id="102">Invalid session variable: 'XYZ'</error>
</rsp>
```

Postback Error Codes

Code	Description

100	Success
101	Request from unauthorized IP
102	Invalid session variable
103	Invalid product code
104	Unable to process purchase

Security: IP Whitelisting

Only accept postbacks from Digipay's servers. Whitelist this IP address:

```
$allowed_ips = array(
    '185.240.29.227'
);
```

Note: Contact tech@digipay.co if you need updated IP addresses.

Return URL Handling

After successful payment, the customer is redirected to your `tcomplete` URL.

The following variables can be included in your return URL:

- `$session` - Your order ID
- `$receipt` - Receipt value from your postback response

Example:

```
https://yoursite.com/thank-you/?order=$session&receipt=$receipt
```

Encryption

All requests to Digipay must be encrypted using AES-256-CBC.

Encryption Process

1. Build the full URL with all parameters
2. Encrypt the URL string
3. Send as `?param=ENCRYPTED_STRING`

PHP Encryption Function

```
function digipay_encrypt($string, $key) {
    $encrypt_method = 'AES-256-CBC';
    $iv_length = openssl_cipher_iv_length($encrypt_method);
    $iv = openssl_random_pseudo_bytes($iv_length);
    $salt = openssl_random_pseudo_bytes(256);
    $iterations = 999;
    $hash_key = hash_pbkdf2(
        'sha512',
        $key,
        $salt,
        $iterations,
        64  // 256 / 4
    );

    $encrypted = openssl_encrypt(
        $string,
        $encrypt_method,
        hex2bin($hash_key),
        OPENSSL_RAW_DATA,
        $iv
    );

    $output = array(
        'ciphertext' => base64_encode($encrypted),
        'iv'         => bin2hex($iv),
        'salt'       => bin2hex($salt),
        'iterations' => $iterations
    );

    return base64_encode(json_encode($output));
}
```

Building the Encrypted URL

```
function build_digipay_url($params) {
    $base_url = 'https://secure.digipay.co/order/creditcard/cc_form_enc.php';
    $query_string = http_build_query($params);
    $full_url = $base_url . '?' . $query_string;

    $encrypted = digipay_encrypt($full_url, DIGIPAY_ENCRYPTION_KEY);

    return $base_url . '?param=' . urlencode($encrypted);
}
```

Testing

Test Environment

Use Digipay's testing bed to test your integration:

1. Log in to the Digipay merchant portal: https://secure.digipay.co/cp/login.php
2. Navigate to the testing section
3. Enter your postback URL
4. Run test transactions

Test Session Value

Implement a test session value in your postback handler for automated testing:

```
$test_session = 'YOUR_SITE_ID_test';  // e.g., "1234_test"

if ($session === $test_session) {
    // Return success without processing
    output_xml_response('ok', 'Test successful', 100, 'TEST-' . time());
    exit;
}
```

Contact tech@digipay.co with your test session value.

Testing Checklist

- [ ] Customer redirects to Digipay payment page
- [ ] Order details display correctly on payment page
- [ ] Postback is received after payment
- [ ] Postback returns correct XML response
- [ ] Order is marked as paid in your system
- [ ] Customer is redirected to thank you page

Error Codes

API Error Codes (from Digipay)

Code	Name	Description
101	INVALID_CP	Invalid Content Partner ID
102	INVALID_USER	Invalid username/password
103	INVALID_SITE	Invalid Site ID
104	IP_NOT_ALLOWED	Your server IP is not authorized
105	API_NOT_ALLOWED	API not enabled for your site
106	INVALID_ACTION	Invalid action parameter
201	INVALID_TRANS_ID	Invalid transaction ID
202	NO_CREDIT_CARD	Customer has no card on file
203	TRANSACTION_MISMATCH	Transaction doesn't match site
204	AMOUNT_TOO_LARGE	Amount exceeds site limit
205	CHARGE_NOT_AUTHORIZED	Payment declined
206	INVALID_AMOUNT	Invalid amount format
301	SYSTEM_ERROR	Internal server error

Complete Code Examples

Complete Postback Handler (PHP)

```php
<?php
/**
 * Digipay Postback Handler
 * Place this file at your pburl location
 */

// Load your framework (WordPress example)
require_once dirname(__FILE__) . '/wp-load.php';

// Set XML content type
header('Content-Type: application/xml; charset=utf-8');

$version = '1.0';

/**
 * Output XML response and exit
 */
function xml_response($status, $message, $code, $receipt = null) {
    global $version;
    echo '<?xml version="1.0" encoding="UTF-8"?>' . "\n";

    if ($status === 'ok') {
        echo '<rsp stat="ok" version="' . $version . '">' . "\n";
        echo '    <message id="' . $code . '">' . htmlspecialchars($message) .
'</message>' . "\n";
        if ($receipt) {
            echo '    <receipt>' . htmlspecialchars($receipt) . '</receipt>' . "\n";
        }
        echo '</rsp>';
    } else {
        echo '<rsp stat="fail" version="' . $version . '">' . "\n";
        echo '    <error id="' . $code . '">' . htmlspecialchars($message) .
'</error>' . "\n";
        echo '</rsp>';
    }
    exit;
}

// Security: Validate IP
$allowed_ips = array('185.240.29.227');
$remote_ip = $_SERVER['REMOTE_ADDR'];

if (isset($_SERVER['HTTP_X_FORWARDED_FOR'])) {
    $ips = explode(',', $_SERVER['HTTP_X_FORWARDED_FOR']);
    $remote_ip = trim($ips[0]);
}
```

```php
if (!empty($allowed_ips) && !in_array($remote_ip, $allowed_ips)) {
    xml_response('fail', 'Request from unauthorized IP ' . $remote_ip, 101);
}

// Parse JSON from POST key (Digipay's actual format)
$data = array();
foreach ($_POST as $key => $value) {
    if (strpos($key, '{') === 0) {
        $data = json_decode($key, true);
        break;
    }
}

// Get session from JSON or GET fallback
$session = !empty($data['session']) ? $data['session'] :
           (isset($_GET['session']) ? $_GET['session'] : '');

// Handle test session
if ($session === 'YOUR_SITE_ID_test') {
    xml_response('ok', 'Test successful', 100, 'TEST-' . time());
}

// Validate session
if (empty($session)) {
    xml_response('fail', 'Invalid session variable: empty', 102);
}

// Convert amount format (29_99 -> 29.99)
$amount = !empty($data['amount']) ? str_replace('_', '.', $data['amount']) : '0.00';

// Convert description (underscores to spaces)
$description = !empty($data['description']) ? str_replace('_', ' ',
$data['description']) : '';

// Get your order
$order_id = intval($session);
$order = wc_get_order($order_id);  // WooCommerce example

if (!$order) {
    xml_response('fail', "Invalid session variable: '{$session}'", 102);
}

// Process the payment
try {
    if ($order->is_paid()) {
        xml_response('ok', 'Order already processed', 100, $order_id);
    }

    $order->payment_complete();
    $order->add_order_note('Digipay payment completed. Amount: $' . $amount);
    $order->save();
    xml_response('ok', 'Purchase successfully processed', 100, $order_id);

} catch (Exception $e) {
    xml_response('fail', 'Unable to process purchase: ' . $e->getMessage(), 104);
}
```

Complete Payment Redirect (PHP)

```
<?php
/**
 * Redirect customer to Digipay for payment
 */

// Your encryption key (store securely!)
define('DIGIPAY_ENCRYPTION_KEY', 'your-key-here');

function digipay_encrypt($string, $key) {
    $encrypt_method = 'AES-256-CBC';
    $iv_length = openssl_cipher_iv_length($encrypt_method);
    $iv = openssl_random_pseudo_bytes($iv_length);
    $salt = openssl_random_pseudo_bytes(256);
    $iterations = 999;

    $hash_key = hash_pbkdf2('sha512', $key, $salt, $iterations, 64);

    $encrypted = openssl_encrypt(
        $string,
        $encrypt_method,
        hex2bin($hash_key),
        OPENSSL_RAW_DATA,
        $iv
    );

    $output = array(
        'ciphertext' => base64_encode($encrypted),
        'iv'         => bin2hex($iv),
        'salt'       => bin2hex($salt),
        'iterations' => $iterations
    );

    return base64_encode(json_encode($output));
}

function redirect_to_digipay($order) {
    $base_url = 'https://secure.digipay.co/order/creditcard/cc_form_enc.php';

    $params = array(
        'site_id'           => 'YOUR_SITE_ID',
        'charge_amount'     => number_format($order['total'], 2, '.', ''),
        'type'              => 'purchase',
        'order_description' => $order['description'],
        'session'           => $order['id'],
        'woocomerce'        => '1',
        'encrypt'           => '1',
        'first_name'        => $order['billing']['first_name'],
        'last_name'         => $order['billing']['last_name'],
        'email'             => $order['billing']['email'],
        'address'           => $order['billing']['address'],
        'city'              => $order['billing']['city'],
        'state'             => $order['billing']['state'],
        'zip'               => preg_replace('/\s+/', '', $order['billing']['zip']),
        'country'           => $order['billing']['country'],
        'shipped'           => '0',
        'pburl'             => 'https://yoursite.com/digipay-postback.php',
        'tcomplete'         => 'https://yoursite.com/thank-you/?order=' .
$order['id']
    );

    $query_string = http_build_query($params, '', '&');
    $full_url = $base_url . '?' . $query_string;

    $encrypted = digipay_encrypt($full_url, DIGIPAY_ENCRYPTION_KEY);
    $payment_url = $base_url . '?param=' . urlencode($encrypted);

    header('Location: ' . $payment_url);
    exit;
}

// Example usage:
$order = array(
    'id' => '12345',
    'total' => 29.99,
    'description' => 'Premium Subscription',
    'billing' => array(
        'first_name' => 'John',
        'last_name'  => 'Smith',
        'email'      => 'john@example.com',
        'address'    => '123 Main St',
        'city'       => 'New York',
        'state'      => 'NY',
        'zip'        => '10001',
        'country'    => 'US'
    )
);

redirect_to_digipay($order);
```

Refunds

Refunds are not processed through the API. To issue a refund:

1. Log in to the Digipay merchant portal: https://secure.digipay.co/cp/login.php

2. Contact Digipay support at tech@digipay.co with:
    - Your Site ID
    - The order/session ID
    - The amount to refund
    - Reason for refund

Digipay support will process the refund and confirm when completed.

Support

For technical questions or issues:

- **Email**: tech@digipay.co
- **Merchant Portal**: https://secure.digipay.co/cp/login.php

---

Changelog

Version 10.0 (January 2026)

- Documented actual postback data format (JSON in POST key)
- Added correct amount format (`XX_XX` with underscore)
- Added Digipay server IP for whitelisting
- Removed references to transaction ID in postback (not sent)
- Added complete working code examples
- Fixed `woocomerce` parameter spelling note
- Reorganized document structure for clarity