



How to Add Stripe to Your Managed Hosting eCommerce Builder

This guide explains how to connect Stripe to our **managed hosting** eCommerce Builder store so you can accept payments for **digital downloads**.

With managed hosting:

- Your store runs on Djangify's infrastructure
 - You do **not** manage servers, Docker, or `.env` files
 - All Stripe configuration is done through the **admin dashboard**
 - You are still fully responsible for your own Stripe account and email configuration keys
-

What you need before you start

You must have:

- A Stripe account
- A configured email ([see documentation on setting up email](#))
- Admin access to your store (`/admin`)

No server access is required.

How payments work in managed hosting

Quick outline of how it all works:

1. You create products and prices in the eCommerce Builder admin
2. At checkout, the store calculates the total internally
3. Stripe processes the card payment and customer details securely
4. Stripe sends a webhook event to your store
5. The store:
 - Creates the order
 - Unlocks downloads
 - Sends the confirmation email

Stripe does **not** manage products or prices for your shop.

Stripe settings used by managed hosting stores

Your store requires the following Stripe values:

- Stripe Publishable Key
- Stripe Secret Key
- Stripe Webhook Signing Secret

These are entered **via the admin dashboard**. Plus the email configuration added in the same area.

Stripe Test Mode vs Live Mode (must be configured separately)

Stripe has two completely separate environments:

- **Test mode** (sandbox – no real money)
- **Live mode** (real payments)

Each mode has:

- its own API keys
- its own webhook endpoint
- its own signing secret

You must fully test your store in **Test Mode** with **email configuration** in place before switching to Live.

Part A — Set up Stripe in Test Mode

Step 1: Enable Test Mode in Stripe

1. Log in to Stripe: <https://dashboard.stripe.com>
2. Go to the left hand menu with your business name (drop down) and select **Switch To Sandbox** (top-right of the dashboard)
3. You will now see **test data only**

Step 2: Get your Stripe Test API keys

While still in Test Mode:

1. Go to **Developers** → **API keys**
 2. Copy:
 - Publishable key (`pk_test_...`)
 - Secret key (`sk_test_...`)
-

Step 3: Create the Test webhook endpoint

Your managed hosting shop webhook endpoint is:

<https://YOURSTORE.djangify.com/shop/webhook/>

(Use your actual store domain.)

In Stripe Dashboard (still in **Test mode**):

1. While in **Test Mode**, go to **Developers** (*bottom left-hand side of your screen*)
→ **Webhooks**
2. Click **Add destination**
3. Select events:
 - `payment_intent.succeeded`
 - `payment_intent.payment_failed`
4. Enter: <https://yourdomain.com/shop/webhook>
5. Create the endpoint
6. Copy the **Signing secret** (`whsec_...`)

Step 4: Enter Test keys in your store dashboard

1. Log in to [/admin](#)
 2. Go to: Pages → Dashboard Settings
 3. Enter:
 - Stripe Publishable Key ([pk_test_...](#))
 - Stripe Secret Key ([sk_test_...](#))
 - Stripe Webhook Signing Secret ([whsec_...](#))
 4. Save changes. Changes take effect immediately.
-

Step 5: Run a full Test Mode checkout

Use Stripe's standard test card:

Card number: 4242 4242 4242 4242

Expiry date: any future date - e.g. 05/35

CVC: any 3 digits and Postcode: any value

Confirm all of the following:

1. Payment appears in Stripe **Test mode**
2. Order appears in Admin → Orders
3. Download access is unlocked
4. Stripe → Webhooks shows the event delivered successfully

If Stripe shows a successful payment but no order appears, the webhook configuration is incorrect.

Part B — Switch to Stripe Live Mode

Only proceed once Test Mode works end-to-end.

Step 6: Get your Stripe Live API keys

1. In Stripe Dashboard, toggle to **Live mode**
 2. Go to **Developers** → **API keys**
 3. Copy:
 - Publishable key (`pk_live_...`)
 - Secret key (`sk_live_...`)
-

Step 7: Create the Live webhook endpoint

Live mode requires a **separate webhook**.

In Stripe Dashboard (Live mode):

1. Go to **Developers** → **Webhooks**
2. Click **Add endpoint**
3. Select events:
 - `payment_intent.succeeded`
 - `payment_intent.payment_failed`
4. Create the endpoint. Endpoint URL:
`https://YOURSTORE.djangify.com/shop/webhook/`
5. Copy the **Live** signing secret (`whsec_...`)

Step 8: Replace Test keys with Live keys in the dashboard

1. Log in to [/admin](#)
2. Go to: Pages → Dashboard Settings
3. Click the **Stripe Live Mode** box - blue tick appears.
4. Replace:
 - Test Publishable Key → Live Publishable Key
 - Test Secret Key → Live Secret Key
 - Test Webhook Secret → Live Webhook Secret
5. Save changes

Your store is now live.

Step 9: Verify a real payment

Make a small real purchase and confirm:

- Payment appears in Stripe **Live mode**
- Order is created in admin
- Downloads unlock correctly
- Stripe shows a successful webhook delivery
- Issue a refund to yourself (optional)

Common issues and fixes

Payment succeeds in Stripe but no order is created

- Webhook URL incorrect (must be [/shop/webhook/](#))
- Wrong webhook signing secret
- Webhook created in the wrong Stripe mode (Test vs Live)

Checkout fails immediately

- Publishable key and secret key don't match the same mode
- Old keys still saved in Dashboard Settings

Nothing changes after saving keys

- Page not refreshed
- Keys saved in the wrong settings section

Managed Hosting Summary

- Prices managed inside the store
- Stripe configured entirely through admin
- Separate setup required for Test and Live mode
- Webhook endpoint is always [/shop/webhook/](#)

Stripe Pre-Launch Checklist

Use this checklist **before opening your store to customers**.

If every item is confirmed, Stripe payments and downloads should work correctly.

A. Stripe Account & Mode

- Stripe account is active and verified
 - You have completed **Test Mode setup first**
 - You have deliberately switched to **Live Mode** for launch
 - You understand Test and Live mode are completely separate
-

B. Stripe API Keys

Confirm you are using **keys from the same mode** (this matters). **Never mix them.**

Test Mode (before launch)

- Publishable key starts with `pk_test_`
- Secret key starts with `sk_test_`

Live Mode (for launch)

- Publishable key starts with `pk_live_`
- Secret key starts with `sk_live_`

C. Webhook Endpoint (Critical)

Your shop webhook **must** be:

/shop/webhook/

Test Mode

- Webhook created while Stripe is in **Test Mode**
- Endpoint URL is correct and complete
`https://YOURDOMAIN/shop/webhook/`
- Events selected:
 - `payment_intent.succeeded`
 - `payment_intent.payment_failed`
- Webhook signing secret copied (`whsec_...`)
- Test webhook secret saved in the store

Live Mode

- Webhook created again in **Live Mode**
- Endpoint URL is identical
`https://YOURDOMAIN/shop/webhook/`
- Events selected:
 - `payment_intent.succeeded`
 - `payment_intent.payment_failed`
- Live webhook signing secret copied
- Live webhook secret saved in the store

Test and Live webhooks are separate. Both must exist.

D. Store Configuration

Self-Hosted

- Stripe keys are set in `.env`
- `.env` changes have been applied correctly:
 - Docker: container recreated (`--force-recreate`)
 - Non-Docker: correct service restarted
- Application restarted after **every** Stripe change

Managed Hosting

- Stripe keys saved in **Pages** → **Dashboard Settings**
 - Keys were saved in the correct fields
 - Page refreshed after saving
-

E. Frontend Checkout Verification

- Checkout page loads without errors
 - Stripe card form appears correctly
 - No console errors related to Stripe
 - Payment button works and is clickable
-

F. End-to-End Payment Test

Test Mode (required)

- Test card 4242 4242 4242 4242 used
- Payment succeeds in Stripe Test Mode
- Order is created in admin
- Download access unlocks
- Stripe → Webhooks shows **Delivered (200)**

Live Mode (final confirmation)

- Small real payment completed
 - Payment appears in Stripe Live Mode
 - Order created in admin
 - Downloads unlock correctly
 - Webhook delivery successful
-

G. Final Sanity Checks

- HTTPS works on your domain
- Domain resolves publicly (no local-only URLs)
- Server firewall allows outbound HTTPS
- No maintenance mode enabled
- Emails (order confirmation) are sending

If **all boxes are ticked**, your Stripe setup is ready.

If the Webhook Does Not Fire (Troubleshooting)

If Stripe shows a successful payment but **no order is created**, the issue is almost always the webhook. Use this diagnostic path **in order**.

1. Check Stripe → Webhooks → Event Log

In Stripe Dashboard:

- Open the webhook endpoint
- Click the most recent event
- Look for:
 - **Status:** Delivered
 - **Response code:** 200

If you see:

- **No attempts** → Stripe never reached your site
- **4xx / 5xx** → your site rejected the request

2. Confirm the Webhook URL Exactly

It must be **exactly**:

`https://YOURDOMAIN/shop/webhook/`

Common mistakes:

- Missing trailing slash
- `/api/` instead of `/shop/`

- Typo in domain
- Using `http://` instead of `https://`

Stripe will not retry indefinitely if the URL is wrong.

3. Confirm Mode Alignment

This catches more errors than anything else.

- Test payment → Test webhook → Test secret
- Live payment → Live webhook → Live secret

If any one of these is mismatched, signature verification will fail silently.

4. Confirm Webhook Signing Secret

- Make sure the `whsec_...` value:
 - Matches the correct mode
 - Was copied fully (no missing characters)
 - Was saved correctly

If the secret is wrong, Stripe **will deliver the event**, but your store will reject it.

5. Check HTTPS & Reachability

Stripe requires:

- Public HTTPS endpoint (*which your store definitely has available*)
- Valid SSL certificate (*again, definitely available*)

Check:

- Visiting <https://YOURDOMAIN/shop/webhook/> returns a response (not a timeout)
 - Firewall is not blocking inbound HTTPS
 - No IP allow-listing blocking Stripe
-

6. Retry Delivery from Stripe

In Stripe:

- Open the failed webhook event
- Click **Retry**

If it succeeds on retry:

- The issue was temporary

If it fails again:

- The configuration is still wrong. Get help from us – **use the contact form in your dashboard area.**

Confirmation

As long as the above instructions are followed correctly the webhook will always fire.

This really is for information only just in case.

correctly:

- The webhook **will fire**
- Stripe **will deliver**
- Orders **will be created**

Webhook failures almost always come from:

- Wrong URL
- Wrong secret
- Wrong mode
- App not restarted / container not recreated

Not from Stripe itself.

Djangify eCommerce Build

<https://djangify.com>