

# What a Webhook Is and How to Receive One

The difference between asking and being told

---

From Nolan Law Firm — [nemolegal.com/tools](https://nemolegal.com/tools)

## The Plain English Version

Most software works by asking: your browser asks a server for a page, the server responds. A webhook flips that. Instead of you asking, the other system tells you when something happens.

Example: Stripe processes a payment. Instead of you checking every few seconds to see if it went through, Stripe sends a webhook — an HTTP POST request — to a URL you provide the moment it happens. Your server receives it and does whatever you've programmed: update a database, send an email, trigger a workflow.

**One-line definition:** A webhook is a URL that receives data when something happens elsewhere.

## How It Works in Practice

Step	What Happens
1	You create a URL that can receive POST requests (your webhook endpoint)
2	You give that URL to the sending service (Stripe, GitHub, Zapier, n8n, etc.)
3	When the event fires, they POST JSON data to your URL
4	Your server receives the request and processes the data
5	You return a 200 OK response so they know it was received

## How to Receive One (n8n — No Code)

In n8n, add a Webhook node. Set Method to POST. Copy the URL it gives you. That URL is now live — anything POSTed to it triggers your workflow. The data arrives in the node as JSON you can map to other steps.

## How to Receive One (Python — Minimal)

```
from flask import Flask, request

app = Flask(__name__)
```

```
@app.route('/webhook', methods=['POST'])

def webhook():

    data = request.json

    print('Received:', data)

    return '', 200 # Always return 200 or the sender will retry

app.run(port=5000)
```

## Testing Without a Live Server

Use [webhook.site](#) or [RequestBin](#). These give you a temporary public URL that captures and displays everything sent to it. Paste the URL into whatever service you're testing, trigger the event, and watch the data arrive. Zero setup.

**Security note:** Always verify webhook signatures if the sending service provides them. Anyone who knows your URL can POST to it. Stripe, GitHub, and others send a signature header you can validate to confirm the request is genuine.

**Using AI to Help With This** You don't have to fully understand this to use it. Here are prompts that work:

```
"I need to receive a Stripe webhook in Python/n8n when a payment succeeds. Walk me through the exact setup."
```

```
"Here is the JSON payload my webhook is receiving: [paste]. Help me write code to extract [specific field] and use it."
```

```
"My webhook is receiving data but my script returns a 500 error. Here is the error: [paste]. What is wrong?"
```