

Flask Webhook + Ngrok + Postman + Logging + Secret Header Auth - Complete Setup

STEP 1: SET UP YOUR PROJECT ENVIRONMENT

```
-----
- Create project folder:
  mkdir flask_ngrok_demo
  cd flask_ngrok_demo
- Create virtual environment:
  python -m venv venv
- Activate virtual environment:
  On Windows PowerShell:
    .\venv\Scripts\Activate.ps1
- Install Flask:
  pip install flask
- Save dependencies (optional):
  pip freeze > requirements.txt
```

STEP 2: CREATE THE FLASK APP FILE

```
-----
- Create a file named: app_with_webhook_logging.py
- Paste the following code:
from flask import Flask, request, jsonify
from datetime import datetime
import json
app = Flask(__name__)
SECRET_KEY = "my_super_secret_key"
LOG_FILE = "webhook_log.jsonl"
REQUIRED_FIELDS = {"event", "user", "status"}
@app.route("/")
def home():
    return "Hello from Flask via Ngrok!"
@app.route("/webhook", methods=["POST"])
def webhook():
    auth_header = request.headers.get("X-Webhook-Secret")
    if auth_header != SECRET_KEY:
        return jsonify({"error": "Unauthorized"}), 401
    data = request.get_json()
    if not data:
        return jsonify({"error": "Invalid JSON"}), 400
    missing_fields = REQUIRED_FIELDS - data.keys()
    if missing_fields:
        return jsonify({"error": f"Missing fields: {' '.join(missing_fields)}"}), 400
    log_entry = {
        "timestamp": datetime.utcnow().isoformat(),
        "data": data
    }
    with open(LOG_FILE, "a") as f:
        f.write(json.dumps(log_entry) + "\n")
    print("Webhook received and logged:", data)
    return jsonify({"status": "received", "data": data}), 200
if __name__ == "__main__":
    app.run(port=5000)
```

STEP 3: RUN THE FLASK APP

```
-----
python app_with_webhook_logging.py
```

STEP 4: START NGROK TUNNEL

```
-----
ngrok http 5000
- Copy the HTTPS URL (e.g. https://abcd1234.ngrok-free.app)
```

STEP 5: TEST USING POSTMAN

```
-----
- Method: POST
- URL: https://your-ngrok-id.ngrok-free.app/webhook
- Headers:
  Content-Type: application/json
  X-Webhook-Secret: my_super_secret_key
- Body → raw → JSON:
{
  "event": "test_event",
```

```
    "user": "john_doe",
    "status": "success"
}
```

STEP 6: VERIFY OUTPUT

- You should see:

```
{
  "status": "received",
  "data": {
    "event": "test_event",
    "user": "john_doe",
    "status": "success"
  }
}
```

- A new entry will be added to webhook_log.jsonl in JSON Lines format.

SUMMARY

You now have a secured, validated, and logged webhook receiver:

- Runs locally via Flask
- Publicly accessible using Ngrok
- Testable with Postman
- Authenticated using a secret key in headers
- Logs all valid requests for later use

This setup is ideal for local development, testing webhooks, and learning API integrations.