Step-by-Step Script with Explanations

```
#!/bin/bash

# --------------------------
# STEP 1: SETUP
# --------------------------

# Go to GitDorker folder (tool must be cloned from GitHub before this)
cd GitDorker

# --------------------------
# STEP 2: GITHUB TOKEN
# --------------------------

# Save your GitHub token into a file called token.txt
# This token allows authenticated searches and avoids GitHub rate limits
# Replace YOUR_GITHUB_TOKEN with your actual token from GitHub settings
echo "YOUR_GITHUB_TOKEN" > token.txt

# --------------------------
# STEP 3: DORKS FILE (SEARCH QUERIES)
# --------------------------

# Create a list of dorks (GitHub search queries) to find secrets
# org:<org-name> limits the search to a specific GitHub organization
# Replace 'TARGET_ORG' with your real org name in next step
cat <<EOF > dorks.txt
org:TARGET_ORG filename:.env
org:TARGET_ORG AWS_SECRET_ACCESS_KEY
org:TARGET_ORG filename:settings.py
org:TARGET_ORG filename:.git-credentials
org:TARGET_ORG "Authorization: Bearer"
org:TARGET_ORG password
org:TARGET_ORG filename:id_rsa
org:TARGET_ORG filename:config
EOF

# --------------------------
# STEP 4: REPLACE TARGET_ORG
# --------------------------

# Replace 'TARGET_ORG' in the dorks file with your real GitHub org
# Example: microsoft, netflix, my-college-name, etc.
sed -i 's/TARGET_ORG/your_org_name_here/g' dorks.txt

# --------------------------
# STEP 5: RUN GITDORKER
# --------------------------

# This command runs GitDorker with:
```

```
# -tf = token file
# -q  = query file (dorks.txt)
# -o  = output file where results will be saved
python3 GitDorker.py -tf token.txt -q dorks.txt -o github_dork_results.txt

# --------------------------
# STEP 6: CLONE REPOS (OPTIONAL)
# --------------------------

# If you find interesting repo links in github_dork_results.txt, clone them:
# Here's how to clone manually:
# git clone https://github.com/your_org/suspicious-repo.git

# --------------------------
# STEP 7: RUN GITLEAKS ON A CLONED REPO
# --------------------------

# Install GitLeaks if not already installed (Linux example)
# curl -sSL https://github.com/gitleaks/gitleaks/releases/latest/download/gitleaks-linux-amd64 -o gitleaks
# chmod +x gitleaks && sudo mv gitleaks /usr/local/bin/

# Now scan a repo:
# gitleaks detect --source=./suspicious-repo --report=gitleaks-report.json

# This scans the codebase and produces a JSON report with secret details.

# --------------------------
# STEP 8: FINAL OUTPUT
# --------------------------

echo "[OK] GitHub dorking completed!"
echo "[FILE] Check 'github_dork_results.txt' for leaked file links."
echo "[FILE] Run GitLeaks on any cloned repos for deeper analysis."
```

Example Output

```
Example Output (github_dork_results.txt):
https://github.com/netflix/dev-config/blob/main/.env
https://github.com/netflix/web-api/blob/main/settings.py
```