Open Source Software Supply Chain Security
Why does it matters?
Mikaël Barbero
October 26, 2022
EclipseCon 2022
Cybercrime: 3rd Economy in the World

$10.5 Trillions in damages by 2025

https://cybersecurityventures.com/hackerpocalypse-cybercrime-report-2016/
Software Supply Chain Attacks increase 742% in 3 years
Threats Actors are Shifting Targets

App in Production

Software Supply Chain
Blast Radius
Return on Investment
Fortune 500

Don't use Solarwind
5.0%

Use Solarwind Orion
95.0%
Fortune 500

Don't use Solarwind
5.0%

Use Solarwind Orion
95.0%
Fortune 500

Don't use Solarwind
5.0%

Use Solarwind Orion
95.0%
Was Sunburst a Software Supply Chain Attack?

https://slsa.dev/
How was Sunburst a Software Supply Chain Attack?

https://slsa.dev/
Why was is a Software Supply Chain Attack?

https://www.techtarget.com/whatis/feature/SolarWinds-hack-explained-Everything-you-need-to-know
Open Source Software Supply Chain Security

Why does it matter?
Open Source has won
80-90% Open source makes up 80-90% of applications

Source: Forrester
Global Supply Chain
What is a Software Supply Chain?

https://slsa.dev/
Where are the Threats?

https://slsa.dev/
Where are the Threats?

https://slsa.dev/
Where are the Threats?

https://slsa.dev/
Where are the Threats?

https://slsa.dev/
Where are the Threats?

https://slsa.dev/
Software Supply Chain Threats

https://slsa.dev/
Malicious contributions — 2021

> *Hypocrite commits* by some researchers from the University of Minnesota

> Researchers tried to insert deliberately buggy (use-after-free) patches into Linux

From: Kees Cook <keescook@chromium.org>
To: linux-kernel@vger.kernel.org
Cc: Kangjie Lu <kjl@umn.edu>, tech-board@lists.linux-foundation.org
Subject: Report on University of Minnesota Breach-of-Trust Incident
Date: Wed, 5 May 2021 10:07:57 -0700

Report on University of Minnesota Breach-of-Trust Incident

An emergency re-review of kernel commits authored by members of the University of Minnesota, due to the Hypocrite Commits research paper.

May 5, 2021

Prepared by the Linux Foundation's Technical Advisory Board
<tech-board@lists.linux-foundation.org>
Chris Mason (chair)
Steven Rostedt (vice-chair)
Christian Brauner
Dan Williams
Greg Kroah-Hartman
Jonathan Corbet
Kees Cook
Laura Abbott
Sasha Levin
Ted Ts'o

Introduction

On April 20, 2021, in response to the perception that a group of University of Minnesota (UMN) researchers had resumed sending compromised code submissions to the Linux kernel, Greg Kroah-Hartman asked the community to stop accepting patches from UMN and began a re-review of all submissions previously accepted from the University.

https://lore.kernel.org/lkml/202105051005.49BFABCE@keescook/
Possible Mitigations

> Two-persons (or more) reviews of external contributions
> Run Static Code Analysis (SCA) tools
Impersonification — 2021

> **ua-parser-js** by **Faisal Salman**

> 7M downloads per week

> Maintained for 10+ years
Impersonification — 2021

Meanwhile, on the dark web... (Oct 5th, 2021)

Acc development, 7kk installations per week

Posted by: 24 minutes ago (changed)

byte

I sell a development account on npmjs.com, more than 7 million installations every week, more than 1000 others are dependent on this. There is no 2FA on the account. Suitable for distributing installations, miners, creating a botnet.

Start $10k
Stop $1k
Blitz $20k
24 hours after the last bet

Guarantor, we will pay the commission 50/50

Before the conclusion of the transaction, mandatory verification of contacts in PM

Quote
Hi all, very sorry about this.

I noticed something unusual when my email was suddenly flooded by spams from hundreds of websites (maybe so I don't realize something was up, luckily the effect is quite the contrary).

I believe someone was hijacking my npm account and published some compromised packages (0.7.29, 0.8.0, 1.0.0) which will probably install malware as can be seen from the diff here: https://app.renovatebot.com/package-diff?name=ua-parser-js&from=0.7.28&to=1.0.0

I have sent a message to NPM support since I can't seem to unpublish the compromised versions (maybe due to npm policy https://docs.npmjs.com/policies/unpublish) so I can only deprecate them with a warning message.
Impersonification — 2021

On October 22, 2021, 3 new versions of `ua-parser-js` are published with malicious code that steal the OS credentials and the cookies on the machines it’s installed on (+ a cryptocurrency miner; because why not?)

 threaten actor was expecting consumers of `ua-parser-js` to use version ranges on dependencies declaration
Possible Mitigations

> 2FA for committers
Protestware

- node-ipc by Brandon Nozaki Miller
- 1M weekly downloads
- Maintained for 8+ years
- 40 other high profile packages on npmjs
- March 7th, 2022: added code that wipe disk if called from an IP geolocalized in Russia or Belarus
Possible Mitigations

> Two-persons (or more) reviews of all contributions
Maintainer change — 2018

event-stream npm package by Dominic Tarr

> 2 million downloads a week
> Was not actively maintained for years
> Threat actor reportedly offered to help maintain the library
> New owner proceeded to add a malicious library called flatmap-stream to the event-stream package as a dependency
> The malicious code inside the library was obfuscated to evade detection
> The code focused on stealing bitcoins from application, redirecting any mined bitcoins to the attacker’s wallet (instead of the intended target)

https://github.com/dominictarr/event-stream/issues/116#issuecomment-440927400
Possible Mitigations

> Governance and processes for ownership transition
Software Supply Chain Threats?

> IDE plugins
> Shell Prompts
> ...

https://slsa.dev/
Software Supply Chain Threats

https://slsa.dev/
PHP git server compromised — 2021

- PHP's self-hosted git server was compromised
- Threat actor injected two malicious commits attributed to well known developers
- Executes PHP code from within the user-agent HTTP header, if the string starts with zerodium

https://news-web.php.net/php.internals/113838
Possible Mitigations

> Protect source code repository server
> Enforce commit signing would help detect rogue commits
Software Supply Chain Threats

https://slsa.dev/
Build server compromised to modify source between check-in and build

Threat actor introduced backdoor to execute commands as root

> Build server compromised to modify source between check-in and build

> Threat actor introduced backdoor to execute commands as root

Webmin build server compromised — 2018

https://www.webmin.com/exploit.html
Possible Mitigations

- Build from fresh commit only, no cache on build servers
- Provenance attestation of the deployed scripts
Software Supply Chain Threats

https://slsa.dev/
solarwinds
(GitHub) Repo-jacking — 2022

GitHub Actions
on: push
jobs:
test:
  strategy:
  matrix:
    platform: [ubuntu-latest, macos-latest, windows-latest]
  runs-on: ${{ matrix.platform }}
  steps:
    - uses: actions/checkout@v3
    - uses: actions/setup-node@v3
      with:
        node-version: 16
    - run: npm install-ci-test
    - uses:
(GitHub) Repo-jacking — 2022

**Actions**
An entirely new way to automate your development workflow.

15098 results filtered by Actions

- **github-docs-to-wiki**
  By cmbrose
  Converts markdown content in a repository into a wiki

- **Dingtalk Robot Notify**
  By leafney
  Send notifications to Dingtalk

- **Cancel Previous Runs Actions**
  By LarchCiu
  Cancel previous workflow-runs. Skip duplicate workflow-runs. Skip or ignore specific paths. Cancel outdated workflow-runs

- **Run Haskell Tests**
  By sol
  Run all tests of a Haskell package
  2 stars

- **bump2version-action**
  By FragileTech
  Increment the version in one or several

- **commit-environment**
  By iwhiteley
  parse ci tags in a commit and add them as
Changing your GitHub username

You can change the username for your account on GitHub.com.

About username changes

You can change your username to another username that is not currently in use. If the username you want is not available, consider other names or unique variations. Using a number, hyphen, or an alternative spelling might help you find a similar username that's still available.

If you hold a trademark for the username, you can find more information about making a trademark complaint on our Trademark Policy page.

If you do not hold a trademark for the name, you can choose another username or keep your current username. GitHub Support cannot release the unavailable username for you. For more information, see "Changing your username."

After changing your username, your old username becomes available for anyone else to claim. Most references to your repositories under the old username automatically change to the new username. However, some links to your profile won't automatically redirect.
Changing your GitHub username

You can change the username for your account on GitHub.com.

About username changes

You can change your username to another username that is not currently in use. If the username you want is not available, consider other names or unique variations. Using a number, hyphen, or an alternative spelling might help you find a similar username that’s still available.

If you hold a trademark for the username, you can find more information about making a trademark complaint on our Trademark Policy page.

If you do not hold a trademark for the name, you can choose another username or keep your current username. GitHub Support cannot release the unavailable username for you. For more information, see "Changing your username."

After changing your username, your old username becomes available for anyone else to claim. Most references to your repositories under the old username automatically change to the new username. However, some links to your profile won’t automatically redirect.
(GitHub) Repo-jacking — 2022

(GitHub) Repo-jacking — 2022

https://github.com/cyberbob

https://github.com/sbullock
(GitHub) Repo-jacking — 2022

https://github.com/cyberbob

https://github.com/sbullock
(GitHub) Repo-jacking — 2022

https://github.com/cyberbob

https://github.com/sbullock
(GitHub) Repo-jacking — 2022

```yaml
on: push
jobs:
  test:
    strategy:
      matrix:
        platform: [ubuntu-latest, macos-latest, windows-latest]
    steps:
      - uses: actions/checkout@v3
      - uses: actions/setup-node@v3
      with:
        node-version: 16
      - run: npm install-ci-test
      - uses:
```

---

ECLIPSE FOUNDATION

COPYRIGHT (C) 2022, ECLIPSE FOUNDATION | THIS WORK IS LICENSED UNDER A CREATIVE COMMONS ATTRIBUTION 4.0 INTERNATIONAL LICENSE (CC BY 4.0)
Possible Mitigations

> Depends on immutable versions
  
  • git tags are not immutable, git commit digest are
  
  • (same could be said for container images by the way)
Jenkins Miner — 2018

> Exploit outdated / vulnerable Jenkins server
> Plain old Remote Code Execution to mine cryptocurrency
> Victims were paying the cloud bills for the threat actors
> Could have been used to do Sunburst-still attack
Possible Mitigations

> Secure and monitor your servers
> Do not run outdated versions
> Follow vulnerabilities reports
Software Supply Chain Threats

https://slsa.dev/
Given enough eyeballs, all bugs are shallow
Given enough eyeballs, all bugs are shallow

2012

2017

2021
Possible Mitigations

> Continuous analysis of dependencies used by systems (and not only at build time)
> Use Software Bill of Materials for more accuracy and faster time to recover
Typosquatting / Brandjacking — 2020

> Was initially used on domain names to make user go to malicious site rather than the expected one (e.g., https://goolge.com)

> In supply chain attacks, the modus operandi is similar: the goal is to make developers use rogue packages rather than official ones

> NPM
  • Electorn vs electron
  • Loadyaml vs loadyml
  • Twilio-npm (brand jacking)

> Ruby
  • Pretty_color vs colorize

> Python
  • requesys, requesrs, and requesr vs request

<table>
<thead>
<tr>
<th>Group ID</th>
<th>Artifact ID</th>
<th>Version(s)</th>
<th>Vulnerability Tracking Identifier</th>
</tr>
</thead>
<tbody>
<tr>
<td>com.github.codingandcoding</td>
<td>maven-compiler-plugin</td>
<td>3.9.0</td>
<td>sonatype-2021-0012</td>
</tr>
<tr>
<td>com.github.codingandcoding</td>
<td>mail-watcher-plugin</td>
<td>1.16, 1.17</td>
<td>sonatype-2021-0013</td>
</tr>
<tr>
<td>com.github.codingandcoding</td>
<td>servlet-api</td>
<td>3.2.0</td>
<td>sonatype-2021-0014</td>
</tr>
</tbody>
</table>

https://blog.sonatype.com/malware-removed-from-maven-central
Dependency Confusion: How I Hacked Into Apple, Microsoft and Dozens of Other Companies

The Story of a Novel Supply Chain Attack

https://medium.com/@alex.birsan/dependency-confusion-4a5d60fec610
Dependency confusion — 2021

awesomelib 69.69.69
by not_a_hacker
(external)

awesomelib 1.0
(internal)
Possible Mitigations

> Digital signature verification
> Verification of provenance attestations before using a dependency
Software Supply Chain Threats

https://slsa.dev/
Threat actor got credentials to Google Cloud Storage account via leak in Docker image (intermediary layer)

They modified the code of a script hosted at GCS

Script executed by GitHub apps, allowed to read environments variables, potentially secrets

---

https://about.codecov.io/apr-2021-post-mortem/
Possible Mitigations

> Use scanner for secret leaks
> Image scanning + squashed image
> Be mindful about the usage of such apps
> We should require those to run with least privileges principle
Software Supply Chain Threats

https://slsa.dev/
Some developers use custom email domains
When they expire, they can be reused
If emails were used on that domain, they can be used to trigger password recovery on many sites, including package registries
Possible Mitigations

> Use 2FA on all (development) accounts
> Use code signing (and verify) on all published artifacts
> Governance removing inactive accounts
Software Supply Chain Threats

https://slsa.dev/
Star Jacking — 2022

django-filer 2.2.3

A file management application for django that makes handling of files and images a breeze.

Project description

django Filer is a file management application for django that makes handling of files and images a breeze.

Note

This project is endorsed by the Django CMS Association. That means that it is officially accepted by the dCA as being in line with our roadmap vision and development/plugin policy. Join us on Stack.
django-filer 2.2.3

A file management application for django that makes handling of files and images a breeze.

Project description

django Filer is a file management application for django that makes handling of files and images a breeze.

Note

This project is endorsed by the Django CMS Association. That means that it is officially accepted by the dCA as being in line with our roadmap vision and development/plugin policy. Join us on Stack.
Star Jacking — 2022

Bowser

A small, fast and rich-API browser/platform/engine detector for both browser and node.

- **Small.** Use plain EJS-version which is ~4.8kB gzipped.
- **Optimized.** Use only those parsers you need — it doesn’t do useless work.
- **Multi-platform.** It’s browser- and node-ready, so you can use it in any environment.

Don’t hesitate to support the project on Github or OpenCollective if you like it ❤️ Also, contributors are always welcome!

**Contents**

- **Overview**
- **Use cases**
- **Advanced usage**
- **How can I help?**

**Overview**

The library is made to help to detect what browser your user has and gives you a convenient API to filter the users somehow depending on their browsers. Check it out on this page: https://bowser-js.github.io/bowser-online/.
Star Jacking – 2022

**Bowser**

A small, fast and rich-API browser/platform/engine detector for both browser and node.

- **Small.** Use plain ES5-version which is ~4.8kB gziped.
- **Optimized.** Use only those parsers you need — it doesn’t do useless work.
- **Multi-platform.** It’s browser- and node-ready, so you can use it in any environment.

Don’t hesitate to support the project on Github or OpenCollective if you like it. Also, contributors are always welcome!

**Contents**

- **Overview**
- **Use cases**
- **Advanced usage**
- **How can I help?**

**Overview**

The library is made to help to detect what browser your user has and gives you a convenient API to filter the users somehow depending on their browsers. Check it out on this page: https://bowser.js.github.io/bowser-online/.
Star Jacking — 2022

- No validation of the connection between the package and the repository
- Link to any popular repository, and benefit from its good track records
Star Jacking — 2022

What Should I do?
Vision

To be the leading open source foundation globally in implementing supply chain security best practices
Simply putting the burden of added security work on the shoulders of open source projects maintainer is not desirable.
What is SLSA?

Supply chain Levels for Software Artifacts, or SLSA (salsa).

It’s a security framework, a check-list of standards and controls to prevent tampering, improve integrity, and secure packages and infrastructure in your projects, businesses or enterprises. It’s how you get from safe enough to being as resilient as possible, at any link in the chain.

Level 1
Easy to adopt, giving you supply chain visibility and being able to generate provenance

Level 2
Starts to protect against software tampering and adds minimal build integrity guarantees

Level 3
Hardens the infrastructure against attacks, more trust integrated into complex systems

Level 4
The highest assurances of build integrity and measures for dependency management in place
## SLSA, Summary of Levels

<table>
<thead>
<tr>
<th>Level</th>
<th>Description</th>
<th>Example</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Documentation of the build process</td>
<td>Unsigned provenance</td>
</tr>
<tr>
<td>2</td>
<td>Tamper resistance of the build service</td>
<td>Hosted source/build, signed provenance</td>
</tr>
<tr>
<td>3</td>
<td>Extra resistance to specific threats</td>
<td>Security controls on host, non-falsifiable provenance</td>
</tr>
<tr>
<td>4</td>
<td>Highest levels of confidence and trust</td>
<td>Two-party review + hermetic builds</td>
</tr>
</tbody>
</table>
## SLSA, Requirements

### Summary table

<table>
<thead>
<tr>
<th>Requirement</th>
<th>SLSA 1</th>
<th>SLSA 2</th>
<th>SLSA 3</th>
<th>SLSA 4</th>
</tr>
</thead>
<tbody>
<tr>
<td>Source - Version controlled</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
</tr>
<tr>
<td>Source - Verified history</td>
<td>✔</td>
<td>✔</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Source - Retained indefinitely</td>
<td></td>
<td></td>
<td>✔</td>
<td>✔</td>
</tr>
<tr>
<td>Source - Two-person reviewed</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
</tr>
<tr>
<td>Build - Scripted build</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
</tr>
<tr>
<td>Build - Build service</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
</tr>
<tr>
<td>Build - Build as code</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
</tr>
<tr>
<td>Build - Ephemeral environment</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
</tr>
<tr>
<td>Build - Isolated</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
</tr>
<tr>
<td>Build - Parameterless</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
</tr>
<tr>
<td>Build - Hermetic</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
</tr>
<tr>
<td>Build - Reproducible</td>
<td></td>
<td></td>
<td></td>
<td>✔</td>
</tr>
<tr>
<td>Provenance - Available</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
</tr>
<tr>
<td>Provenance - Authenticated</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
</tr>
<tr>
<td>Provenance - Service generated</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
</tr>
<tr>
<td>Provenance - Non-falsifiable</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
</tr>
<tr>
<td>Provenance - Dependencies complete</td>
<td>✔</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Common - Security</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
</tr>
<tr>
<td>Common - Access</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
</tr>
<tr>
<td>Common - Superusers</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
</tr>
</tbody>
</table>

O = required unless there is a justification
SLSA, compliance

SLSA level

Level 2

Declare your project to be compliant with the given SLSA level. See SLSA Security Levels documentation for levels descriptions and the associated normative requirements at each level.

Latest Releases:
From October 6th, 2021 to October 6th, 2021

<table>
<thead>
<tr>
<th>Name</th>
<th>Date</th>
<th>Review</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.0 Handbook</td>
<td>2021-10-06</td>
<td></td>
</tr>
</tbody>
</table>

Active Member Companies:
Member companies supporting this project over the last three months.

Contributors:
Shawn Kilpatrick

GitLab Repositories:
https://gitlab.eclipse.org/eclipse/dash/org.eclipse.dash.handbook

GitLab Excluded Sub-Groups:
eclipse/technology/dash/sync-script-testing/exclude-test

GitLab Project Group:
eclipse/technology/dash

SLSA level:
Level 2

Email Notifications

Documentation
Legal Documentation Generator

Releases
Create a Progress Review
Create a new release

PROJECT LINKS
Website

Eclipse Dash™ declares compliance with SLSA level 2
SLSA: provenance

- SLSA L3 for “Github-native” projects
- Sofrito: Jenkins Shared Library to generate provenance attestation (L1)

https://github.com/slsa-framework/slsa-github-generator
SLSA, sofrito

```json
@Library('sofrito')

pipeline {
  agent any
  stages {
    stage ('Build') {
      steps {
        sh 'mvn clean verify'
        script {
          provenance. generate('target/*.jar')
        }
      }
    }
  }
}
```

---

**ECLIPSE FOUNDATION**

COPYRIGHT (C) 2022, ECLIPSE FOUNDATION | THIS WORK IS LICENSED UNDER A CREATIVE COMMONS ATTRIBUTION 4.0 INTERNATIONAL LICENSE (CC BY 4.0)
SBOMs

PROJECT EE4J.YASSON

2022-10-05 00:34

PROJECT SUMMARY

- Project ID: ee4j.yasson
- Project PMI: https://projects.eclipse.org/projects/ee4j.yasson
- Last analysis: yasson on 202210060034

YASSON

Summary

- Date of last run: 2022-10-06T00:34:00 — Status: Success: Published
- Violations: 45 — download CSV file.
- Reports:
  - WebApp report
  - Static html report
  - Notice file
  - SPDX SBOM
  - CycloneDX SBOM

History
SBOMs

```xml
<?xml version="1.0" encoding="UTF-8"?>
< bom xmlns="urn:uuid:4fa6c0d3e80-2edc-4e3d-80c-7baf64c4d3f" version="1" xmlns:sbom="http://cyclonedx.org/schema/bom/1.3">
  <components>
    <component type="library">
      <group>jakarta.annotation</group>
      <name>jakarta.annotation-api</name>
      <version>2.1.0</version>
      <description>JavaAnnotationsAPI</description>
      <scope>required</scope>
      <hashes>
        <hash alg="SHA-1">4f1cf66b0cede3a75a8ac3d12ee8af2d798e323d</hash>
      </hashes>
    </component>
    <license>
      <license>
        <name>GPL-2.0-only WITH Classpath-exception-2.0</name>
        <ort:origin xmlns:ort="http://www.w3.org/1999/xhtml">concluded license</ort:origin>
      </license>
    </license>
    <license>
      <license>
        <name>GPL-2.0-only WITH Classpath-exception-2.0</name>
        <ort:origin xmlns:ort="http://www.w3.org/1999/xhtml">declared license</ort:origin>
      </license>
    </license>
    <licenses>
      <license>
        <url>pkg:maven/jakarta.annotation/jakarta.annotation-api@2.1.0</url>
        <modified>false</modified>
        <externalReferences>
          <reference type="website">https://projects.eclipse.org/projects/eel4j.ca/</reference>
        </externalReferences>
      </license>
    </licenses>
  </components>
  <component type="library">
    <group>jakarta.el</group>
    <name>jakarta.el-api</name>
    <version>5.0.0</version>
    <description>JavaExpressionLanguage defines an expression language for Java applications</description>
    <scope>required</scope>
    <hashes>
      <hash alg="SHA-1">2a22b304f2082d4d6427cdefb5ce5f6726e2a63a3</hash>
    </hashes>
  </component>
</bom>
```
Other areas

- Measure
- Control
- Improve
- Analyze
Measure

Scorecards
https://github.com/ossf/scorecard

Allstars
https://github.com/ossf/allstar
https://mikael.barbero.tech/blog/post/eclipsefdn-scorecard-aug2022/
Analyze

Token-Permissions

- Non read-only token: 33.1%
- No read-only token: 66.9%

Histogram of Branch-Protection

https://mikael.barbero.tech/blog/post/eclipsefdn-scorecard-aug2022/
Improve

[StepSecurity] ci: Harden GitHub Actions #1766

This is an automated pull request generated by Secure Workflows at the request of @mbarbero. Please merge the Pull Request to incorporate the requested changes. Please tag @mbarbero on your PR. You can also engage with the StepSecurity team by tagging @step-security.

Summary

This is an automated pull request generated by Secure Workflows at the request of @mbarbero. Please merge the Pull Request to incorporate the requested changes. Please tag @mbarbero on your PR. You can also engage with the StepSecurity team by tagging @step-security.

Security Fixes

Least Privileged GitHub Actions Token Permissions

The least privileged token permissions were calculated using Secure Workflows at the request of @mbarbero. Please merge the Pull Request to incorporate the requested changes. Please tag @mbarbero on your PR. You can also engage with the StepSecurity team by tagging @step-security.

- GitHub Security Guide
- The Open Source Security Foundation (OpenSSF) Security Guide

Changes from all commits

<table>
<thead>
<tr>
<th>.github/workflows/</th>
<th>license.yml</th>
<th>quickstarts.yml</th>
</tr>
</thead>
</table>
Improve - otterdog

- GitHub organizations management at scale
- Eclipse Foundation: 50+ organizations, 1000+ repositories
  - 150+ organizations tomorrow
- Side benefit: projects will be able to ask for some tweak by send PR.
  - “As Code” FTW!
2 Factors Authentication

- Physical security keys, FIDO compatible
- Time-based One-Time Password (TOTP) authenticator apps
The Open Source Technology Improvement Fund is a corporate non-profit dedicated to securing open source apps that we all depend on. Securing software isn’t easy, and we know what it takes to succeed. By facilitating security audits and reviews, OSTIF makes it easy for projects to significantly improve security.
Security Audit - some OSTIF audits

Backstage (April - August 2022) — Security Review, Threat Model
sigstore (May 2022) — Security Review, Threat Model
CRI-O (June 2022) — Security Review, Threat Model, Supply Chain Security Assessment
Flux (September 2021) — Security Review
Linux Kernel (April 2021) — Policy Review
Linux Kernel (January 2021) — Policy Review
Unbound (December 2019) — Security Review
OpenSSL (January 2019) — Security Review
OpenSSL PRNG (September 2018) — Security Review
OpenVPN (May 2017) — Security Review
Veracrypt (October 2016) — Security Review
Security Audit - Eclipse IDE p2 PGP signing

SOON
Key Takeaways

- It’s a jungle out there!
- Security is hard, Supply Chain Security is harder
- Eclipse Foundation will provide services, best practices and tools to its projects to be leaders
Thank You!

Mikaël Barbero
Head of Security

mikael.barbero@eclipse-foundation.org
🐦 @mikbarbero
https://mikael.barbero.tech