OpenSSF
OPEN SOURCE SECURITY FOUNDATION

Developing a Secure, Open Future

OSS-NA 2024

OpenSSF
OPEN SOURCE SECURITY FOUNDATION

CRob, n, adj, and v

Pronunciation: U.S. (K-robe)

42nd level Dungeon Master

25th level Securityologist

Security Lorax, Cat-herder

Pirate-enthusiast & hat-owner

OpenSSF
OPEN SOURCE SECURITY FOUNDATION

Agenda

01 - Problem Statement

02 - What is the OpenSSF?

03 - How can the OpenSSF help developers

04 - How can YOU help?

OpenSSF

OPEN SOURCE SECURITY FOUNDATION

The Problem

Developing software is HARD
Developing software SECURELY is **HARDER**
Developers are experts at WRITING SOFTWARE not SECURITY
SECURITY is not always a PRIORITY for projects, developers, or *developer's managers*

Open Source Developer Spectrum

- There are MANY motivations for why people choose to create and collaborate with open source software(1).
- Many large and mature open source projects have specialists or teams that help manage security for the effort.
- The majority of open source projects are single-maintainer projects that do not have access to those resources, nor an understanding of how(2).

(1) - https://www.linuxfoundation.org/hubfs/LF%20Research/MaintainerSecurityBPs_011724.pdf?hsLang=en
(2) - https://anchore.com/blog/open-source-is-bigger-than-you-imagine/

Case Study Persona: Diana the Weekend Warrior

Role:

- Working on open source software in their "spare time"
- Maintains the project(s) by themselves
- Started out as a way to learn new things for fun, has grown beyond that

Background:

- “I maintain a couple of small packages and contribute new medium-size but impactful features to my underlying ecosystem.” (Think a compiler optimisation for floats that takes a few months of work and extremely niche knowledge to get right) This is a really common and critical profile.
- Diana is in a loose network of other niche people doing the same in their ecosystem.
- Diana has challenges keeping their toolchain and CI systems up-to-date and running. C was not made for this kind of work, nor are most of the packaging ecosystems, and they have to fight with them all the time.

Goals:

- Keep the project going and as maintained as possible, given constraints of budget, time, and resources.

https://github.com/ossf/toolbelt/tree/main/personas#name-diana-the-weekend-warrior

Image Source

OpenSSF
OPEN SOURCE SECURITY FOUNDATION

Diana continued

Challenges:

- Diana usually gets something like 2 hours per month to spend on FOSS. Sometimes up to 4h, sometimes less. 1 to 2h per month are dedicated to simply updating base dependencies. This is when it is just a few patches or minor versions. Sometimes up to 4 hours are taken for this. Sometimes a big major version in an important dependency happens and it takes us 10h to fix, over a quarter. This means that nearly all our *time* is spent handling dependency stuff. Basic stuff. Not security emergencies or anything like that. Releasing a new version that just is kept up to date basically.
- Diana does not have more *time* to give. Life is what it is; they have family, a job, friends, etc.
- Diana's tests for the project are in poor shape. This is very well known. They want to make them better, but per the above, there is no additional *time* to devote to this task. Even something like fuzzing would be incredibly challenging to deal with the additional bugs that could be found, prioritized, and eventually (maybe) fixed.
- Diana has no additional *time* to read security-oriented material or the use.
- Most of Diana's *time* is spent fighting build and dependency management tools. This is a constant problem. These tools often break in new and obscure ways. Oftentimes, features of the tool need to be disabled to even make things work and not break their builds. There are probably ways around this, but again, see above, there is no additional time to troubleshoot, research, and adjust. Even the basic tools break too much and eat Diana's *time*.
- Reviewing and reacting to user reports is challenging.

https://github.com/ossf/toolbelt/tree/main/personas#name-diana-the-weekend-warrior

Image Source

Does this sound familiar to anyone?

OpenSSF
OPEN SOURCE SECURITY FOUNDATION

Current State TL/DR

- There are more single-maintainer projects than ones with multiple maintainers.
- There are more consumer requests and priorities than most maintainers ever have planned for.
- More of the developer's time is consumed with operational things (fixing dependency conflicts, infrastructure, creating tests) rather than actual new development.
- Learning new things and "security" rarely make it off the backlog for many.

Image Source

Is *anyone* trying to help?

OpenSSF
OPEN SOURCE SECURITY FOUNDATION

The Rising Tide lifting all boats

- The OpenSSF is a cross-industry collaboration that brings together leaders to improve the security of open source software (OSS) by building a broader community, targeted initiatives, and best practices
- The OpenSSF brings together open source security initiatives under one foundation to accelerate work through cross-industry support. This is beginning with the Core Infrastructure Initiative and the Open Source Security Coalition, and will include new working groups that address vulnerability disclosures, security tooling and more.
- OpenSSF is committed to collaboration and working both upstream and with existing communities to advance open source security for all.

https://openssf.org/

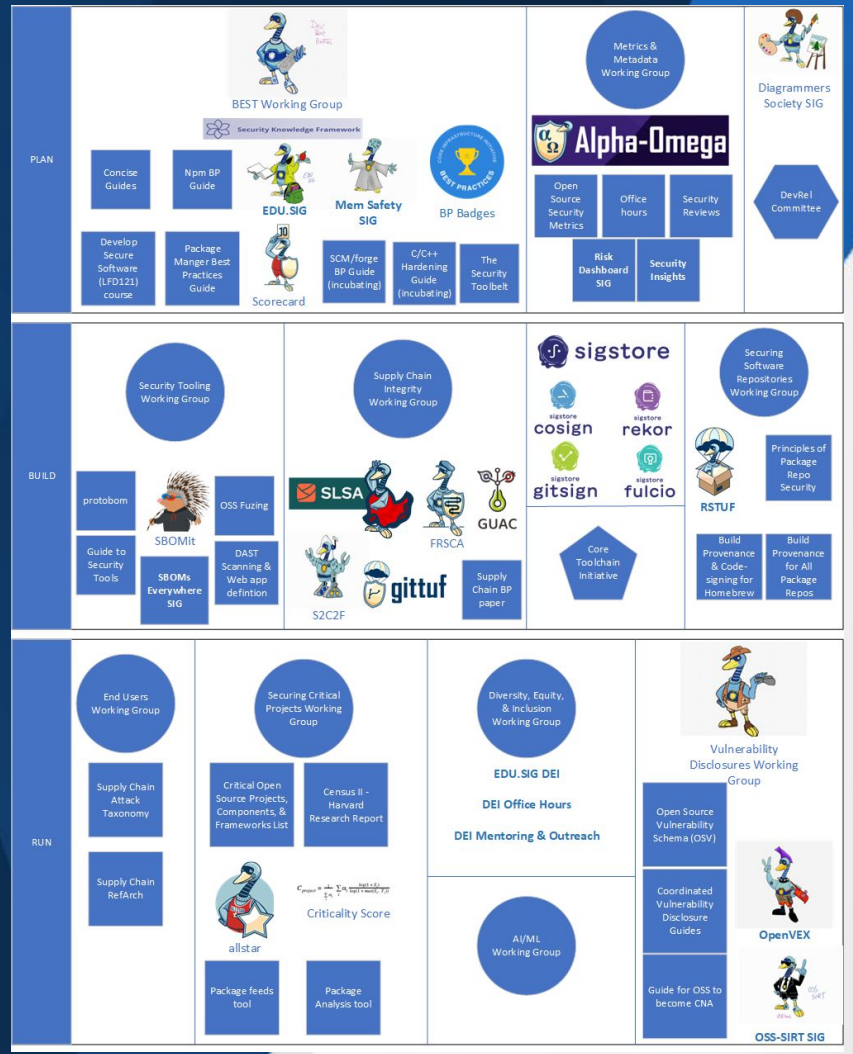
OpenSSF
OPEN SOURCE SECURITY FOUNDATION

A Gaggle of Geese!

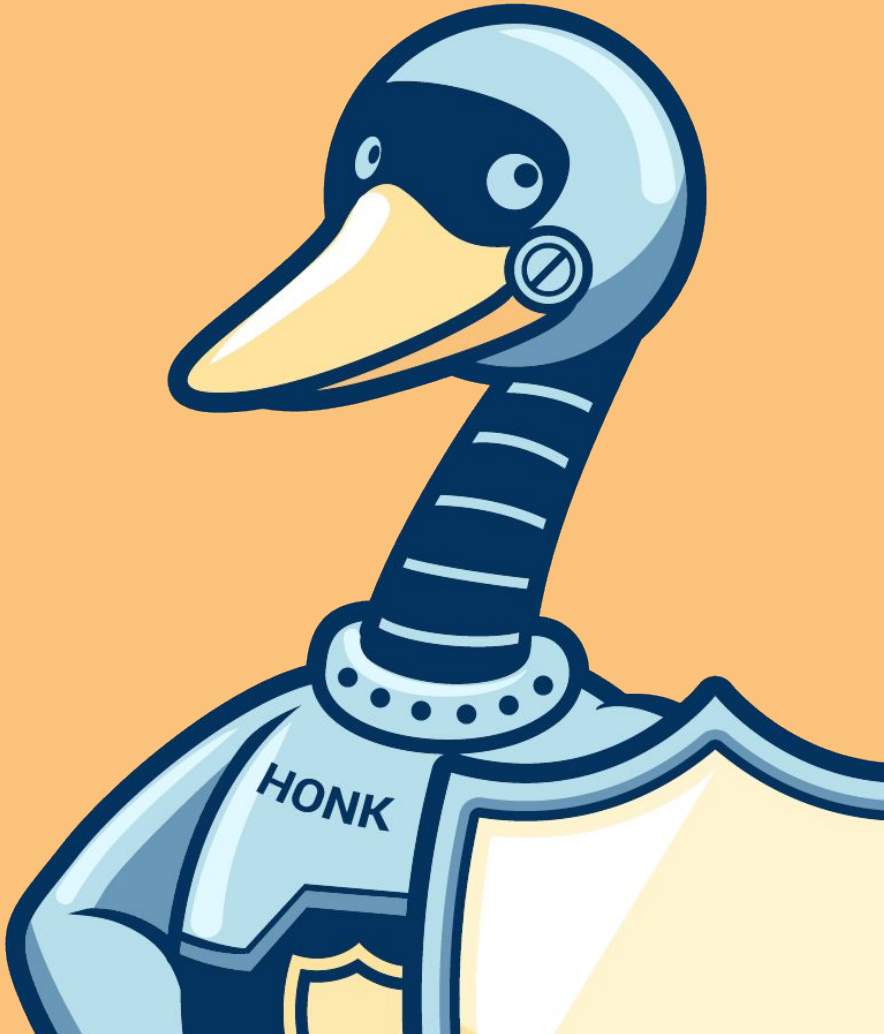
We have numerous software projects, guidelines, specs, and experts to help both upstream OSS developers AND downstream OSS consumers

https://openssf.org/community/openssf-working-groups/

What's up with all the Geese?



Here are some things you can do to set up your project for appsec-success!



The POWER of Education

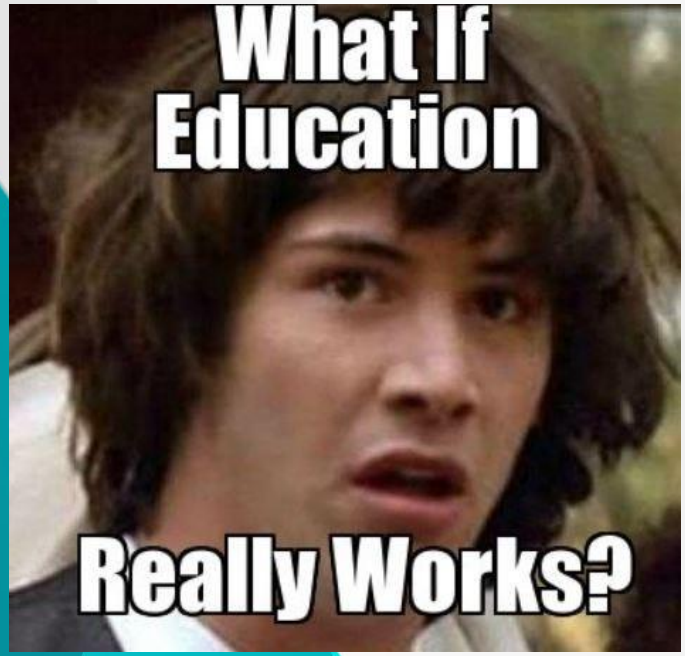


Image Source

Training and education are tools that are **ALWAYS** with you once attained.

- OpenSSF's LFD121 - Developing Secure Software
- OpenSSF's Concise Guide to Developing More Secure Software

Other AMAZING resources

- OWASP Top 10
- SANS Top 25 Most Dangerous Software Errors

OpenSSF
OPEN SOURCE SECURITY FOUNDATION

Protect your accounts

- Don’t reuse passwords
- Use a password vault to store credentials
- Use MFA
- Scrub code and CI for secrets

Image Source

Ensure your SCM & CI Systems are protected

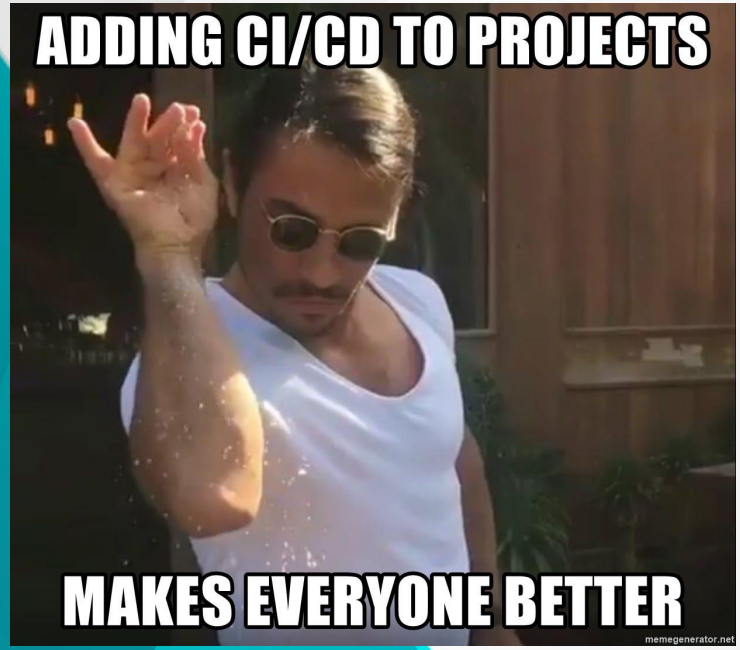


Image Source

- Source Code Management BP guide
- npm BP Guide
- Principles for Package Repository Security
- SLSA
- Enable things like branch protection

OpenSSF
OPEN SOURCE SECURITY FOUNDATION

Validate and Secure the code you're consuming

- Concise Guide to Evaluating OSS Software
- Scorecard & Allstar
- OpenSSF Best Practices Badge
- Implementing Scorecard and Best Practices Badge into your project (video)
- Tools like Dependabot or Renovatebot
- GUAC dependency tree maps
- OSV

Image Source

OpenSSF
OPEN SOURCE SECURITY FOUNDATION

Publish your vuln mgmt process/security policy

- Each project operates differently and has different needs.
- Tell people how you want to handle security reports and how they will be managed.

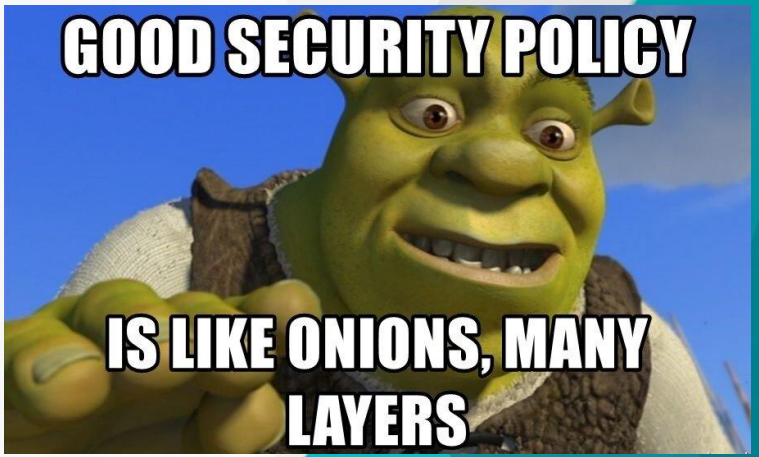


Image Source

https://docs.github.com/en/code-security/getting-started/adding-a-security-policy-to-your-repository
https://gitlab.com/gitlab-org/gitlab-foss/-/blob/read-template-from-repository/doc/release/security.md

OpenSSF
OPEN SOURCE SECURITY FOUNDATION

Establish your Security “team”

- Not every developer is a securityologist.
- Identify people in your project that might have these skills or find some security friends that can help in times of need!

Image Source

OpenSSF
OPEN SOURCE SECURITY FOUNDATION

Establish a CNA contact (or other means of vuln id disclosure)

- As vulns are found and fixed, TELL your downstream about it so they can take action. This is important!
- A CVE Numbering Authority (CNA) is a party authorized to issue CVE IDs for a particular scope of hardware/software, and is the most common way organizations communicate about vulnerabilities.
- There are other ID methods such as GHSA, OSV or GSD that are also CVE-compatible and OSS-workflow-friendly.

Image Source

OpenSSF
OPEN SOURCE SECURITY FOUNDATION

Setup a means for private intake

- A reported vulnerability is a threat to any users of the software if left unfixed.
- Establishing a private way that a Finder can share details or reproducers with the project helps ensure bad actors don't learn about the problem before the project or the users.

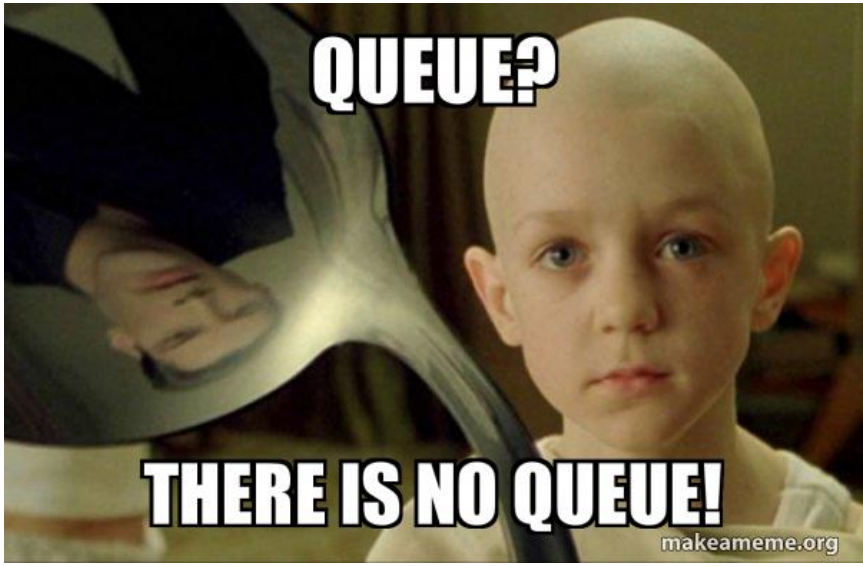


Image Source

OpenSSF
OPEN SOURCE SECURITY FOUNDATION

Establish a means of private patch development & testing

- Like private reporting, it is important that patches that address the vulnerability be kept out of mainline code branches until **after** they have been tested and are ready for public disclosure.
- Bad actors monitor source code repositories for “interesting” (i.e. security-related) PRs and commits.

Image Source

OpenSSF
OPEN SOURCE SECURITY FOUNDATION

Conclusions

- Attackers are increasing their attacks
- Security is important
- OpenSSF provides many resources to help people counter those attacks. See:
  - https://openssf.org
  - https://best.openssf.org/developers

What questions do you have?

What additional resources from the WG would you find helpful?

Image Source

OpenSSF
OPEN SOURCE SECURITY FOUNDATION

Ways to Participate

- Join the OpenSSF Mailing List
- Follow us on Twitter
- Follow us on LinkedIn
- Follow us on Mastodon
- Follow us on Facebook
- Subscribe to our YouTube Channel
- Join a Working Group/Project
- Access the Public Meetings Calendar
- Participate on Slack
- Follow OpenSSF on GitHub
- Become an Organizational Member

Thank You

CRob_at_Intel_dot_com

@SecurityCRob

@SecurityCRob@infosec.exchange

https://github.com/SecurityCRob

The Security Unhappy Hour,
Chips & Salsa
What's in the SOSS? (New)

https://www.linkedin.com/in/darthcrob/