OpenSSF

OPEN SOURCE SECURITY FOUNDATION

OpenSSF Overview

January 19, 2023

Arnaud Le Hors – lehors@us.ibm.com

What is OpenSSF?

MISSION

The purpose of OpenSSF is to inspire and enable the community to secure the open source software we all depend on.

Provides tools, services, training, infrastructure, and resources to achieve this vision.

HISTORY

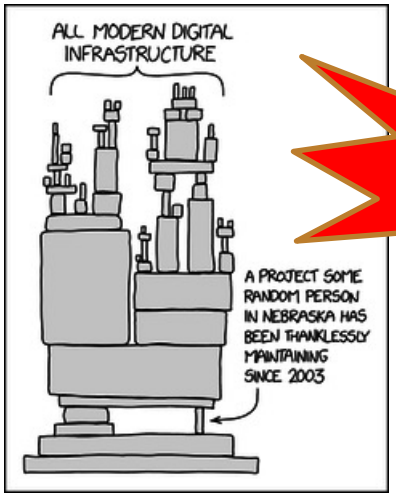
Created in August 2020, as a Linux Foundation project, by a few members with limited budget.

Relaunched in October 2021 with proper funding:

$10.5M in new commitments:
$5.5M from Membership dues + $5M from Microsoft and Google towards “Alpha-Omega”

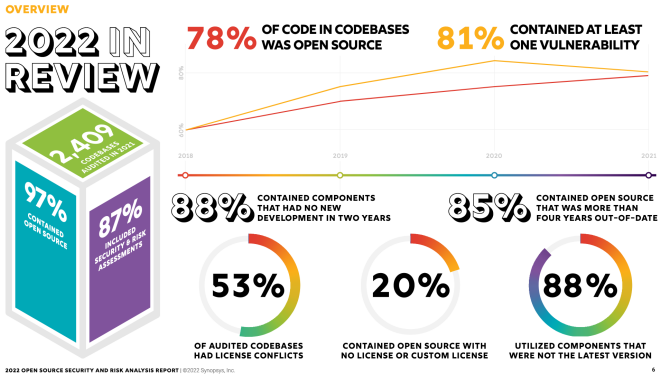
OpenSSF now has 100+ Members coming from IT, Financial, Telecom, and Universities :
AWS, Citi, Google, Huawei, IBM, Intel, Microsoft, Morgan Stanley, GitLab, Goldman Sachs, Harvard, Samsung, ...

Why OpenSSF?



Source: https://xkcd.com/2347/

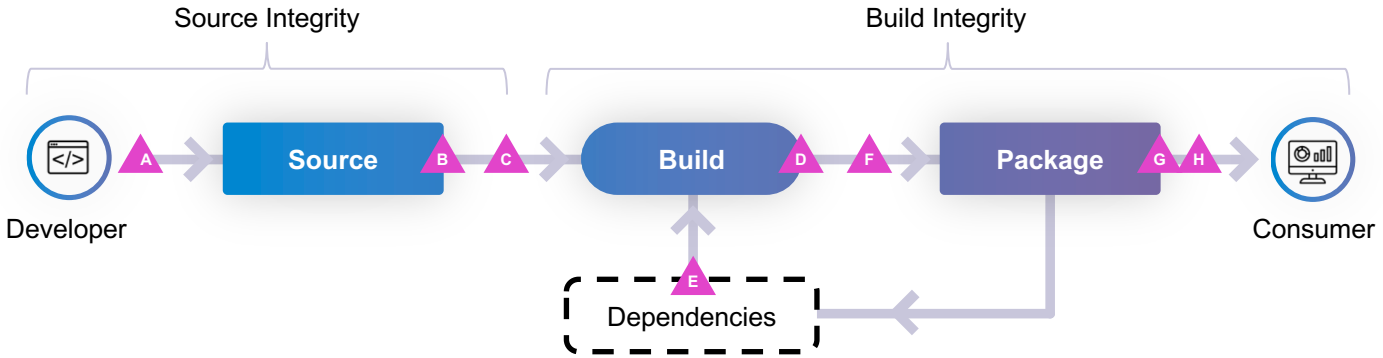
9/12/2021
log4j
NVD Critical 10/10



- Open source makes up at least 70 percent of all software - "2020 Open Source Security and Risk Analysis Report" by Synopsys
- Software supply chain attacks have increased 650% and have a severe impact on business operations - "2021 State of the Software Supply Chain," by Sonatype
- Government leaders worldwide are calling for private and public collaboration – US administration issued Executive Order on May 12th, 2021 directing NIST and others to work with private sector on supply chain security and similar initiatives are in the works in EU: Cyber Resilience Act (CRA)

Our software supply chains are under attack

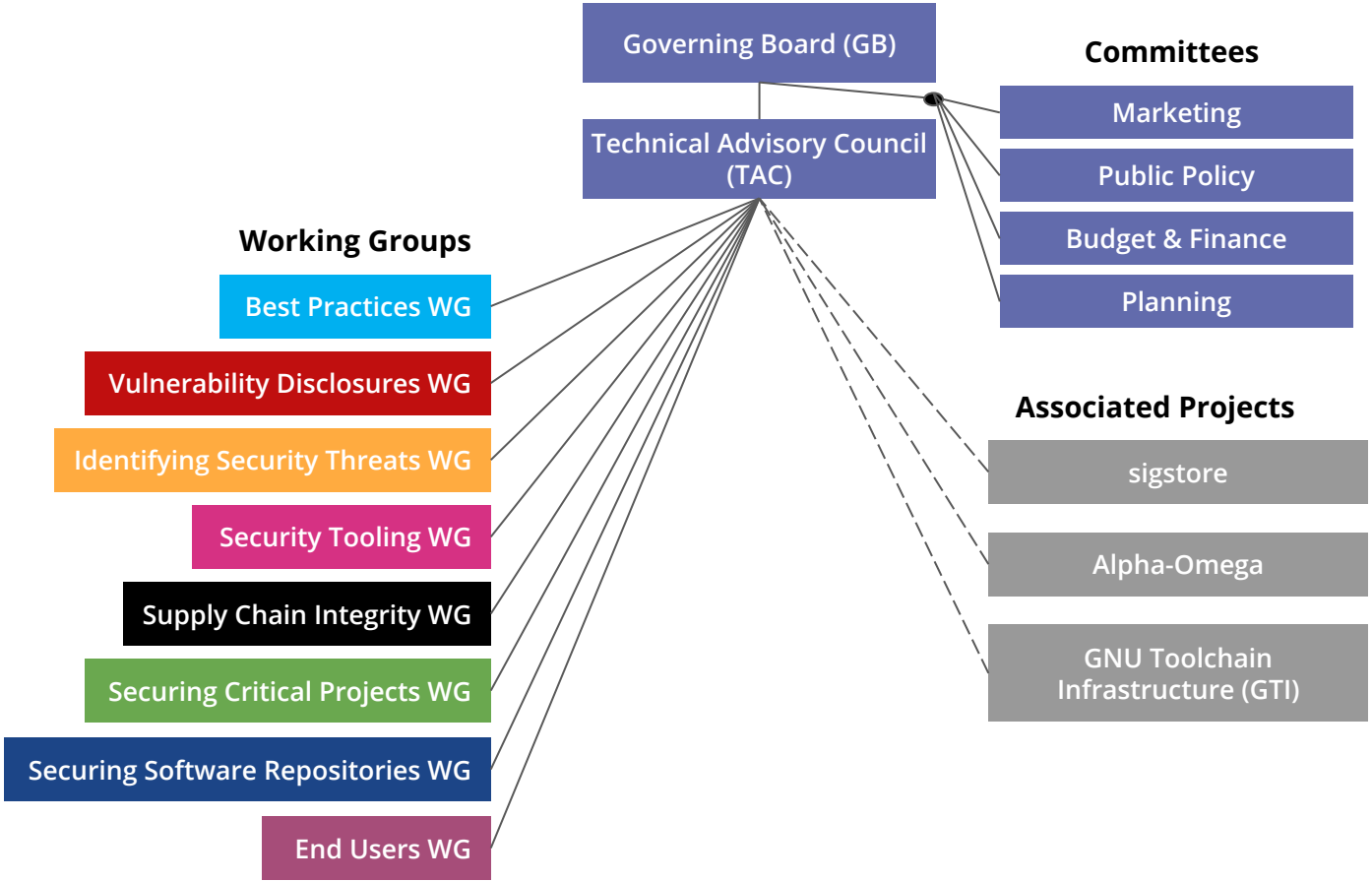
Source integrity and build integrity are critical



A Bypassed code review
B Compromised source control system
C Modified code after source control
D Compromised build platform
E Using a bad dependency
F Bypassed CI/CD
G Compromised package repo
H Using a bad package

OpenSSF
OPEN SOURCE SECURITY FOUNDATION

Open Source Security Foundation (OpenSSF)



OpenSSF
OPEN SOURCE SECURITY FOUNDATION

Working Groups (and their projects & Associated Projects)

Best Practices

Identification, awareness, and education of security best practices

- OpenSSF Best Practices badge
- Scorecard
- Great MFA distribution SIG
- Common Requirements Enumeration (CRE)*
- Secure Software Development Fundamentals courses SIG
- Security Knowledge Framework (SKF)*

Vulnerability Disclosures

Efficient vulnerability reporting and remediation

- Guide to coordinated vulnerability disclosure for OSS projects
- Vulnerability Disclosures Whitepaper
- osv-schema

Identifying Security Threats

Security metrics/reviews for open source projects

- security-reviews,
- Project-Security-Metrics (dashboard)
- SECURITY-IMPACTS.yml spec

Security Tooling

State of the art, globally accessible security tools

- ossf-cve-benchmark
- Web Application Definition spec
- fuzz-introspector

Securing Software Repositories

collaboration of repositories & tools to improve security

- Coming soon!

Supply Chain Integrity

Ensuring the provenance of open source code

- Supply-chain Levels for Software Artifacts (SLSA) [repo]

Securing Critical Projects

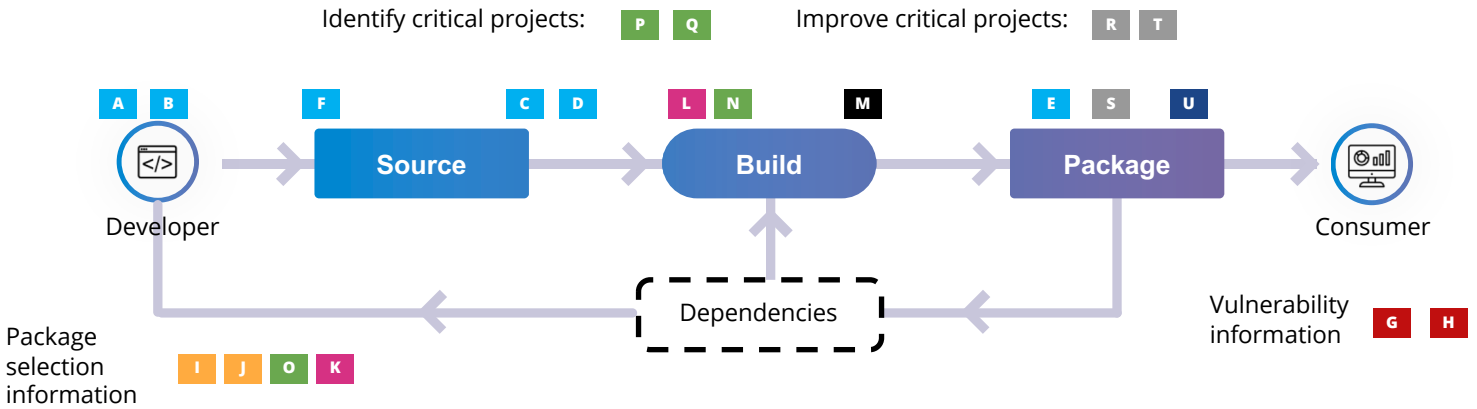
Identification of critical open source projects

- criticality_score
- Harvard research
- package-feeds / package-analysis
- allstar

Associated Projects

- Project Alpha-Omega
- Project Sigstore
- GNU Toolchain Infrastructure (GTI) support

How OpenSSF Projects Work Together



Best Practices WG

A. Secure Software Development Fundamentals courses (education)
B. Security Knowledge Framework (SKF): Hands-on course (education), with OWASP
C. OpenSSF Best Practices Badge
D. Scorecard
E. Great MFA distribution SIG
F. Common Requirements Enumeration (CRE)

Vulnerability Disclosures WG

G. Guide to coordinated vulnerability disclosure for OSS projects; Vulnerability Disclosures Whitepaper
H. osv-schema

Identifying Security Threats WG

I. security-reviews
J. Project-Security-Metrics: Dashboard

Security Tooling WG

K. ossf-cve-benchmark: measure tools
L. Web Application Definition

Supply Chain Integrity WG

M. Supply-chain Levels for Software Artifacts (SLSA)

Securing Critical Projects WG

N. allstar
O. package-feeds / package-analysis
P. criticality_score
Q. Harvard study

End Users WG

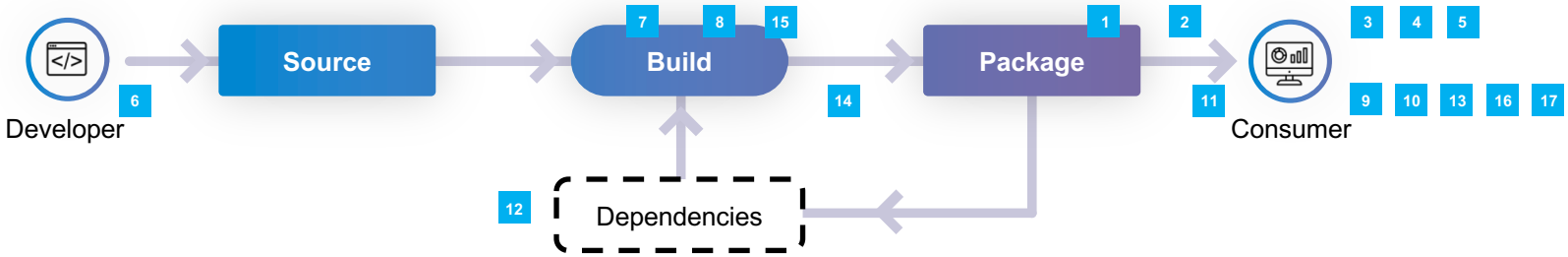
Special Initiative Funds

R. Project Alpha-Omega
S. sigstore
T. GNU Toolchain Infrastructure (GTI) support

Securing Software Repositories WG

U. Securing Software Repositories

Linux Foundation's Security Community other than OpenSSF



1. **SPDX (ISO 5962):** international standard for Software Bill of Materials
2. **CNCF:** Guide for supporting software supply chain best practices
3. **SSDF:** Secure Software Development Fundamentals set courses
4. **Let's Encrypt:** the world's largest certificate authority for the https:// protocol
5. **CCC:** Confidential Computing Consortium protects data in use in memory
6. **CHAOSS:** Community Health Analytics Open Source Software creates analytics and metrics for OSS that define health and identify risk
7. **in-toto:** a framework designed to secure the integrity of software supply chains.
8. **TUF:** The Update Framework maintains the security of software update systems
9. **Uptane:** protects software updates delivered over-the-air to automobiles.
10. **patatt tool:** end-to-end cryptographic attestation to patches sent via email
11. **OpenChain (ISO 5230):** international standard for open source component tracking through supply chain
12. **LFX:** identify OSS vulnerabilities and code secrets, powered by Snyk and BluBracket
13. Tern: software composition analysis tool and library to generates a layer-by-layer view of what's included within a container image
14. SBOM Generator: automatically generate a SBOM from your CI/CD system
15. CatchIT: CI/CD plug-in identifying confidential or sensitive information in code, and catch security violations
16. osquery: performant endpoint visibility
17. CASE Ontology: An international standard supporting automated combination, validation, and analysis of cyber-investigation information

Every OSS project is responsible for its own security (building on capabilities of others)

Highlights for Hyperledger (YMMV)

- Guide to coordinated vulnerability disclosure for OSS projects
- Concise Guide for Developing More Secure Software
- Concise Guide for Evaluating Open Source Software
- npm Best Practices Guide
- Core Infrastructure Initiative (CII) Best Practices Badge (now at OpenSSF)
- Scorecard + Allstar
- Sigstore
- SLSA

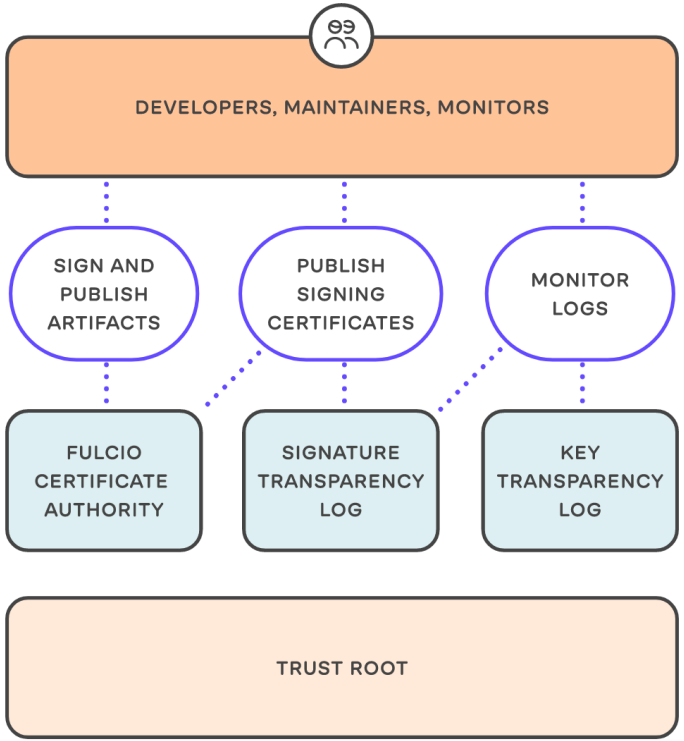
Scorecard / Allstar

- Scorecard: fully automated tool that assesses a number of important heuristics ("checks") associated with software security
  - Helps projects identify areas to improve
  - Enables users (& potential users)) to assess risks & make informed decisions, including evaluating alternatives & working with maintainers to make improvements
  - Growing to provide results on 1M OSS projects
  - Criteria text no longer requires GitHub (need help implementing beyond GitHub)
  - Improved detection of some tools
    - Challenge: Tool detection in large variety of CI/CD pipelines & tools
- Allstar: enables enforcing GitHub security policies
  - Monitor and detect various GitHub settings or repository file contents that may be risky or do not follow security best practices

Sigstore

- Cosign – A tool to easily generate keys and sign artifacts without risk of losing or leaking the keys
- Rekor – A transparency and timestamping service, Rekor records signed metadata to a ledger that can be searched, but can't be tampered with.
- Fulcio – A free root CA, issuing temporary certificates to an authorized identity and publishing them in the Rekor transparency log.

- Uses OpenID Connect as the identity layer



Supply-chain Levels for Software Artifacts (SLSA)

- Each level provides requirements, processes, and best practices to increase trust
- SLSA 1.0 expected in 1Q2023 (unpublished draft: https://slsa.dev/spec/v1.0/

Track/Level	Requirements	Benefits
Build L0	(none)	(n/a)
Build L1	Attestation showing that the package was built as expected	Documentation, mistake prevention, inventorying
Build L2	Signed attestation, generated by a hosted build service	Reduced attack surface, weak tamper protection
Build L3	Hardened build service	Strong tamper protection
Build L4	(not yet defined)	
Source L…	(not yet defined)	

- Tool: GitHub generator (SLSA level 3)

References

- Technical Working Groups - https://github.com/ossf
- Public Meetings Calendar - http://bit.ly/ossf-calendar
- Slack Channel - https://slack.openssf.org
- YouTube Channel - http://bit.ly/ossf-youtube
- Guides: https://openssf.org/resources/guides/

Main website: https://openssf.org