OpenSSF
OPEN SOURCE SECURITY FOUNDATION

Addressing the OSS Security challenge

Arnaud Le Hors lehors@us.ibm.com
23 Sept 2024

All software under attack, via vulnerabilities & supply chain

96%

of the total codebases contained open source

84%

of codebases contained at least one open source vulnerability

Log4Shell

https://www.synopsys.com/blogs/software-security/open-source-trends-ossra-report.html
https://www.sonatype.com/state-of-the-software-supply-chain/introduction

Security of OSS

- Some defects are vulnerabilities (just like closed source)
- Also has supply chain (SC) attacks — but the most common may be surprising!
- Most common SC problem: Users download the *wrong software*
  - Typosquatting & dependency confusion
- Less common: Seized OSS developer accounts (typically stolen password)
  - We're working to deploy MFA everywhere, e.g., token giveaways
- Uncommon: Intentional submission/insertion of malicious source code
  - Multi-person OSS projects resilient
  - Linux kernel has fended off 2 attacks

Governmental Agencies World-wide Becoming Proactive

Executive Order (EO) on Improving the Nation's Cybersecurity (12/05/2021)
Executive Order (EO) on Ensuring Responsible Development of Digital Assets (3/9/2022)

- *including Quantum Computing*

www.whitehouse.gov

CYBERSECURITY & INFRASTRUCTURE SECURITY AGENCY

www.cisa.gov

CISA Cybersecurity Advisory Committee (CSAC) (6/2021)

- Cybersecurity Best Practices, Cyber Threats and Advisories

European Commission

The European Union Agency for Cybersecurity (ENISA)

- Cyber Resilience Act (9/15/2022)

外務省 Japan Ministry of Foreign Affairs

National Security Council (NSS)

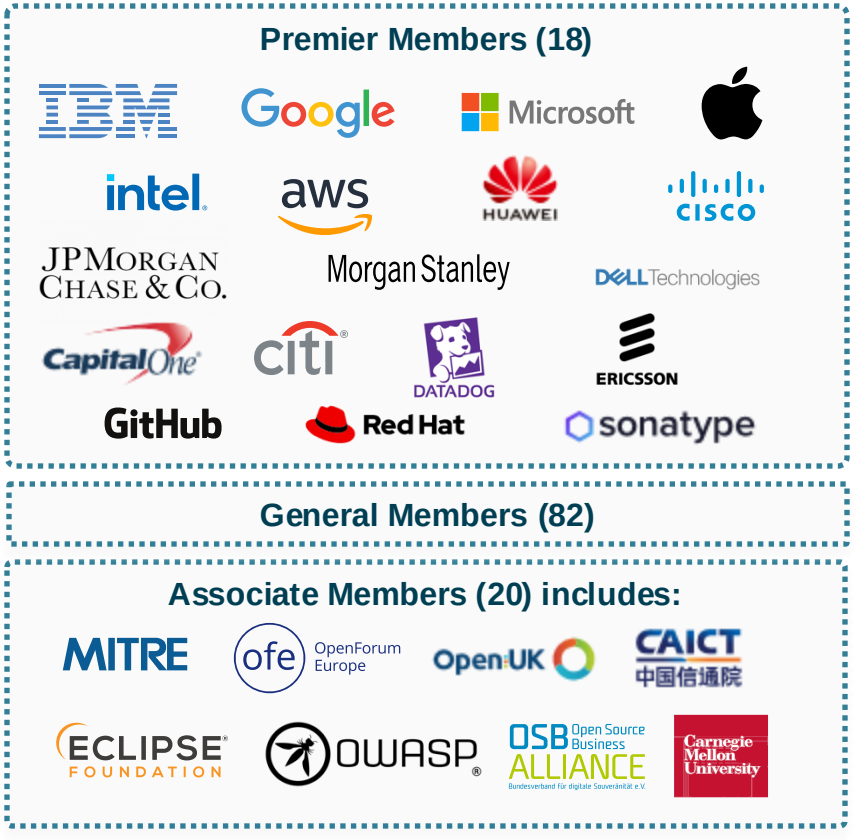
- National Security Strategy (12/16/2022)

OSS, Security, & OpenSSF

- Millions of OSS projects
- Many foundations run OSS projects relevant to security and/or DevSecOps
  - LF Foundations: Continuous Delivery Foundation, Cloud Native Computing Foundation, etc.
  - Other foundations: Apache Software Foundation, Eclipse Foundation, Python Software Foundation, OWASP, etc.

- Open Source Security Foundation (OpenSSF)
  - “Collaboration and working both upstream and with existing communities to advance open source security for all”

Open Source Security Foundation

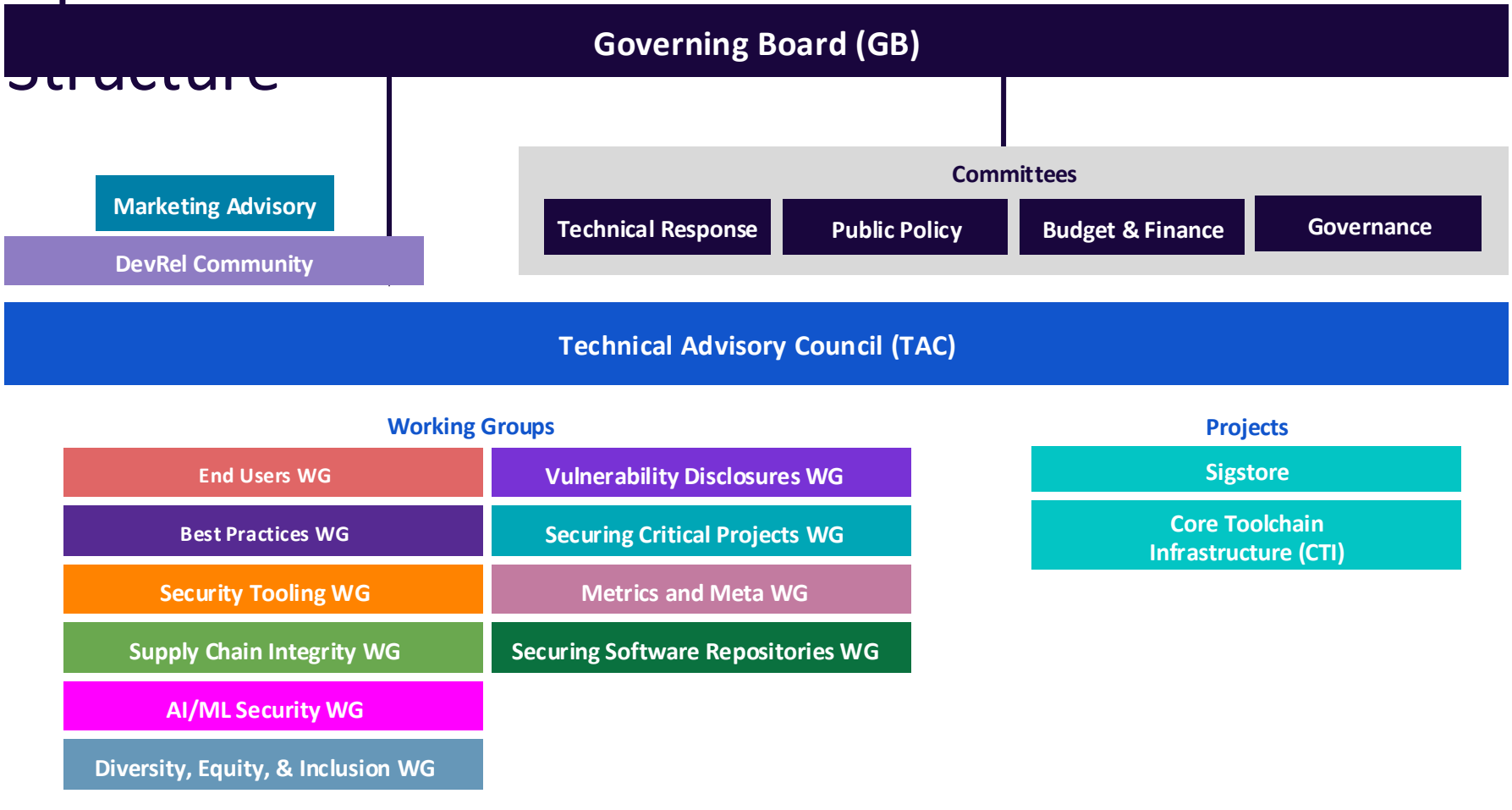


https://openssf.org/about/members/

OpenSSF
OPEN SOURCE SECURITY FOUNDATION

- **August 2020** – Formed under the The Linux Foundation
  - *"The Open Source Security Foundation (OpenSSF) is a cross-industry forum for a collaborative effort to improve open source software security."*
  - Originally, an OSS "Coalition" of companies comprised of **"organic" work groups**
- **Dec 2021** – **OpenSSF "2.0" -** Election of Governing Board (GB) with expanded membership and funding
  - Jamie Thomas, GM Systems Strategy and Development, IBM elected GB Chair
- **Sep. 2023** – OpenSSF Membership Exceeds 110 with Many New Members Dedicated to Securing Open Source Software

OpenSSF Structure



Working Groups, Projects, & SIGs

1. FOUNDATIONAL INFORMATION

2. ENGAGE

3.

Vulnerability Disclosures

Efficient vulnerability reporting and remediation

I. **CVD Guides** SIGs
J. **OSS-SIRT** SIG
K. **Open Source Vuln Schema (OSV)** project
L. **OpenVEX** project
OpenVEX SIG
M. **Vuln Autofix** SIG

Best Practices

Identification, awareness, and education of security practices

A. **Secure Software Development Fundamentals courses** SIG
B. **Security Knowledge Framework (SKF)** project
C. **OpenSSF Best Practices Badge** project
D. **OpenSSF Scorecard** project
E. **Common Requirements Enumeration (CRE)** project
F. **Concise & Best Practices Guides** SIGs
G. **Education** SIG
H. **Memory Safety** SIG
AG. **The Security Toolbelt** SIG
AL. Python Hardening SIG

End Users

Voice of public & private sector orgs that primarily consume open source

Z. **Threat Modeling** SIG

Metrics & Metadata

Security metrics/reviews for open source projects

N. **Security Insights** project
O. **Metrics API** SIG
P. **Security Reviews** project

Security Tooling

State of the art security tools

Q. **SBOM Everywhere** SIG
R. **OSS Fuzzing** project
AI. **SBOMit** project
AJ. **Protobom** project

Supply Chain Integrity

Ensuring the provenance of open source code

S. **Supply-chain Levels for Software Artifacts (SLSA)** project
T. **Secure Supply Chain Consumpt Framework (S2C2F)** project
AJ. **gittuf** project

AK. **GUAC** project
AM. **Zarf** project

Securing Software Repositories

collaboration between repository operators

AB. **RSTUF** Project

Securing Critical Projects

Identification of critical open source projects

U. **List of Critical OS Prj, components, & Frameworks** SIG
V. **criticality_score** project
W. **Census** SIG
X. **Package Analysis** project
Y. **allstar** project

AI/ML Security

AI/ML Security at the Intersection of Artificial Intelligence and Cybersecurity

AD. **Model Signing** SIG

DevRel

Develop Use Cases and help others learn about security

Diversity, Equity, & Inclusion

Increase representation and strengthen the overall effectiveness of the cybersecurity workforce

Projects

Category-leading software initiatives

AE. **Sigstore**
AF. Core Toolchain Infrastructure (CTI)

Sample OpenSSF Project/SIG Results

- Education: *Secure Software Development Fundamentals* (**free** course)
- Guides:
  - *Concise Guide for Developing More Secure Software*
  - *Concise Guide for Evaluating Open Source Software*
  - *Security Baseline (WIP)*
- OSS Security Evaluation:
  - *OpenSSF Scorecard*; auto-measures OSS github.com/ossf/scorecard
  - *OpenSSF Best Practices Badge* (for OSS projects); >6,100 participating, 3 levels
  - *Supply-chain Levels for Software Artifacts (SLSA)*
- Improved tooling: *Sigstore (signing)*
- Vulnerability finding/reporting:
  - *Alpha-Omega:* proactively find/fix vulnerabilities openssf.org/community/alpha-omega
  - *Vulnerability Disclosure Guide* github.com/ossf/oss-vulnerability-guide

Is OSS or proprietary software always more secure?

- Neither. The reality is that neither OSS nor proprietary always more secure
  - If you care, evaluate
- A design principle gives OSS a *potential* security advantage
  - Saltzer & Schroeder [1974/1975] defined secure design principles still valid today
  - Open design principle: “the protection mechanism must not depend on attacker ignorance”
  - OSS better fulfills this principle
  - “Many eyes” theory can work
    - Academics, science & engineering already based on peer review
    - Security experts widely perceive OSS advantage
    - Requires reviewers who know what to look for *and* fixing the problems found
- No software is perfect, so vulnerabilities may be found in well-run projects
  - Continuous careful review is *more* likely to detect vulnerabilities over time

Many technologies at play

- SBOMs (SPDX, CycloneDX)
- Artifact Dependency Graphs (GUAC, dependency track)
- Integrity protections (sigstore, PGP, etc)
- Process-based measurements (SLSA, scorecard, HipCheck, and per-foundation guides)
- Safe consumption frameworks (S2C2F, ESF guide)

Increasing your security posture

Adopt internal policies and processes regarding Open Source use:

- Open source training: open source, open governance, licenses, participation guidelines, security best practices
- Open source clearance: scanning for licenses and vulnerabilities, scorecard
- DevSecOps: scanning for any changes to licenses, vulnerability, scorecard + generation of SBOM+SLSA Provenance

Open source is free – as in puppies

Open source is free, free as in puppies. You might get a free puppy from the shelter, but it is now your responsibility.

Similarly, with open source, you need to:

- Ensure currency with upstream
- Remediate defects and vulnerabilities as they are discovered
- Beware of creating your own fork
- Ideally, contribute fixes and improvements upstream

Get involved!

- To get involved in OpenSSF see openssf.org
  - Biweekly meetings, mailing lists, Slack
  - See our blog for what's going on: openssf.org/blog
- Many other OSS projects & foundations, e.g., Continuous Delivery
- Industry, academia, & government should work together
- The best way to influence an OSS project direction is to get involved!

This presentation released under the CC-BY-4.0 license

This overall presentation is released under the Creative Commons Attribution 4.0 International (CC-BY-4.0). You are free to:

- Share — copy and redistribute the material in any medium or format
- Adapt — remix, transform, and build upon the material

for any purpose, even commercially. This license is acceptable for Free Cultural Works. The licensor cannot revoke these freedoms as long as you follow the license terms. Under the following terms:

Attribution — You must give appropriate credit, provide a link to the license, and indicate if changes were made. You may do so in any reasonable manner, but not in any way that suggests the licensor endorses you or your use.

No additional restrictions — You may not apply legal terms or technological measures that legally restrict others from doing anything the license permits

For full details, see: creativecommons.org/licenses/by/4.0/