Committed to Trust

A Qualitative Study on Security & Trust in Open Source Software Projects

Dominik Wermke¹, Noah Wöhler¹, Jan H. Klemmer², Marcel Fourné³, Sascha Fahl¹ und Yasemin Acar⁴

This paper has been published at the 43rd IEEE Symposium on Security and Privacy in 2022.

Open Source Software (OSS) is an unavoidable component in many of today’s software ecosystems. Whether as low-level system drivers in operating systems, as tooling in daily jobs, or simply as dependencies of hobby projects, OSS is an important building block.

In a 2020 report covering 45,000 repositories, GitHub found that most projects on their platform rely on some form of OSS [Gi20]. In recent years, collaborative version control platforms such as GitHub [Mi08] and GitLab [Gi14] introduced a wide field of developers to open source projects. As the complexity of modern software development increased, so did the number of dependencies and involved contributors. Decentralized development and open collaboration of open source projects introduce unique challenges: code submissions from unknown entities, limited personpower for reviewing commits and dependencies, and bringing new contributors up-to-speed in projects’ best practices and processes.

Assessing vulnerabilities in components is a difficult task, as the large number of dependencies required by today’s software result in a complex software supply chain, including software repositories, package managers, and package registries. Recent incidents in the npm ecosystem highlight the large attack surface provided by such registries: in late October 2021, versions of the npm package *ua-parser-JS* with 7 million weekly downloads included malicious code [CC21]. Analogous to a 2020 report from The Linux Foundation [Th20], we consider the software supply chain in this work to include technical features such as how the software is stored, how it can be retrieved, and how it is analyzed during these processes.

The same holds true for commercial software: by building their software as a wrapper or glue around open source components, companies can leverage OSS as building blocks in their processes and products, allowing them to focus their efforts on features and faster delivery. In 2020, 95% of IT departments and companies considered OSS as strategically

¹ CISPA Helmholtz Center for Information Security, Germany, dominik.wermke@cispa.de; noah.woehler@cispa.de; fahl@cispa.de
² Leibniz University Hannover, Germany, klemmer@sec.uni-hannover.de
³ Max Planck Institute for Security and Privacy, Germany, marcel.fourne@mpi-sp.org
⁴ George Washington University, United States, acar@gwu.edu

doi:10.18420/sw2024_46
important to their organization’s overall enterprise infrastructure software strategy [Re20].
By introducing open source components, companies inherit the same challenges and attack
surfaces as open source projects.

These chain effects make the open source ecosystem an important field of research for the
(security and privacy) community. With the introduction of more developer-centered research
approaches arose the need for human-subject research considerations. Recent conflicts
between the research and open source communities such as the „hypocrite commits“ incident
in early 2021 highlight the need for more respectful research approaches for investigating
security and trust in open source projects [Li21]. In this work, we propose a more cooperative
approach for researching open source, working together with committers towards a more
secure and trustworthy ecosystem, instead of against them.

In addition to security, trust also plays an important role in software development and
especially the open source community, as was probably best described in Ken Thompson’s
Turing Award Lecture „Reflections on Trusting Trust“:

[Th84] To what extent should one trust a statement that a program is free of Trojan horses?
Perhaps it is more important to trust the people who wrote the software..

In this work, we aim to shed light on security and trust practices in open source projects —
by exploring projects’ behind-the-scene processes, provided guidance and security policies,
as well as past security challenges and incident handling. We are especially interested in
processes that are often not directly visible from the repository data, e. g., trust relationships,
incident responses, and the handling of suspicious or malicious contributors. For this, we
conducted 27 in-depth, semi-structured interviews with contributors of open source projects.

Literatur

us-cert.cisa.gov/ncas/current-activity/2021/10/22/malware-discovered-popular-npm-package-ua-parser-js,


[Li21] Linux Foundation’s Technical Advisory Board: Report on University of Minnesota Breach-of-Trust
Incident, https://lwn.net/ml/linux-kernel/20210501005.49BFABCE@keescook/, Accessed:


0001-0782, url: https://doi.org/10.1145/358198.358210.