U.S. HOUSE OF REPRESENTATIVES
COMMITTEE ON SCIENCE, SPACE, AND TECHNOLOGY
SUBCOMMITTEE ON INVESTIGATIONS AND OVERSIGHT
SUBCOMMITTEE ON RESEARCH AND TECHNOLOGY

HEARING CHARTER

Securing the Digital Commons: Open-Source Software Cybersecurity

Wednesday, May 11, 2022
10:00 a.m. EDT – 12:00 p.m. EDT
Zoom

PURPOSE

The purpose of this hearing is to discuss the unique benefits and risks inherent in open-source software, and to explore the ways in which industry and government can collaborate to enhance open-source cybersecurity. The hearing will examine recent open-source software hacks and subsequent efforts to improve security for the development and deployment of open-sourced software. Members and witnesses will discuss the Federal role in improving open-source cybersecurity, particularly at the National Institute of Standards and Technology (NIST). Finally, the hearing will explore the use and potential misuse of open-source software in the development of critical technologies, including artificial intelligence (AI).

WITNESSES

- **Ms. Lauren Knausenberger**, Chief Information Officer, Department of the Air Force
- **Mr. Brian Behlendorf**, General Manager, Open Source Security Foundation
- **Ms. Amélie Erin Koran**, Non-Resident Senior Fellow, The Atlantic Council
- **Dr. Andrew Lohn**, Senior Fellow, Center for Security and Emerging Technology, Georgetown University

OVERARCHING QUESTIONS

- What are the consequences of insecure open-source software and what are organizations in both the public and private sectors doing to help prevent those consequences?
- What further research and standards activities are needed to secure the open-source software ecosystem of the future?
- What policy changes can help secure the open-source software ecosystem? How can the Federal government, including NIST, most effectively collaborate with industry and other stakeholders to help secure open-source software?

What is Open-Source Software?

The 2019 National Defense Authorization Act defines open-source software as software for which the human-readable source code is available for use, study, re-use, modification,

enhancement, and re-distribution by the users of such software.[1] Simply put, open-source software is code that can be used, modified, and distributed by anyone. The software can be a standalone program, such as the web browser *Firefox* or the operating system *Linux*. It also can be software that serves a specific function as a component of larger programs. As a component open-source software is present in 97% of codebases,[2] meaning that essentially everything done on a computer uses open-source software at some level.

Open-source software's ubiquity is due to the advantages it provides. The open nature of the code allows a broader spectrum of people to both improve and to use the software, and its flexibility ensures that it can be altered to serve the specific needs of the user. It can also provide an alternative to proprietary software—commercial software such as Windows Office. Finally, it is distributed for free, allowing access to technical capabilities users may not otherwise be able to afford.[3]

Users gain access to open-source software and all its attendant benefits through online software repositories. Some organizations, such as the Apache Software Foundation, maintain a select set of internally managed projects each with their own repository.[4] Others, such as GitHub (which is owned by Microsoft), provide a place for anyone to host or maintain a repository and from which anyone can download open-source software.[5] Federal agencies are also producers and distributors of open-source software. A 2016 memo from the Office of Management and Budget (OMB) directed agencies to make 20% of their custom software open source through code.gov, and individual agencies such as NASA and NIST have their own websites to distribute the open-source software they create.[6][7]

Open Source vs. Proprietary Software Security

The risks of open-source software are fundamentally the same as proprietary software: the presence of vulnerabilities can permit the intrusion of bad actors. The difference is in the way vulnerabilities are managed for these two types of software. A long running unsettled debate in the cybersecurity community is whether open-source software is more secure because it has more people evaluating it for bugs, or if proprietary software is more secure because hackers have less access to search for vulnerabilities.[8] Analyses of real-world attack data suggest that open-source software is not less secure on average than proprietary software, and when properly managed, may be more secure.[9]

The Security Challenges of Open-Source Software

One challenge for open-source security is the relative lack of resources dedicated to preemptive security, such as the creation of software that is more secure by design, or the operation of

---

[1] Public Law 115-232. https://www.congress.gov/115/plaws/publ232/PLAW-115publ232.pdf
[2] https://www.synopsys.com/software-integrity/resources/analyst-reports/open-source-security-risk-analysis.html
[3] https://www.redhat.com/en/blog/value-open-source
[4] https://www.apache.org/
[5] https://github.com/about
[6] https://obamawhitehouse.archives.gov/sites/default/files/omb/memoranda/2016/m_16_21.pdf
[7] https://code.nasa.gov/
[8] https://dwheeler.com/secure-programs/Secure-Programs-HOWTO/open-source-security.html
[9] https://www.atlanticcouncil.org/resources/breaking-trust-the-dataset/

internal vulnerability checking. Volunteers contribute to open-source projects based on interest, and because of that fewer than 3% of time spent on open-source projects is dedicated to the security of those projects.[10] Identifying which open-source software meets the designation of "critical," and thus is deserving of greater attention to detect vulnerabilities, is an ongoing effort. NIST has issued a definition of critical software, but primarily aimed at government applications. For open-source software the definition of "critical" is largely in the eye of the beholder, though researchers are working now to develop a consensus.

While open-source software has historically been patched quickly following the identification of a vulnerability,[11] the widespread use of the same open-source software by multiple platforms means that a single vulnerability can have a massive impact. Since open-source software is often a component of a larger program, it can also be challenging to determine if a given program is affected by a vulnerability, which in turn makes it hard to know when patching is required. This problem is so prevalent that that 88% of proprietary software contains outdated or otherwise unsafe open-source software.[12]

Noteworthy Open-Source Software Vulnerabilities

On November 24, 2021, a security researcher with Alibaba privately reported to the Apache Software Foundation that their popular Log4j software had a vulnerability which could allow for remote code execution, essentially giving an attacker the ability to run anything—from software that mines cryptocurrency to ransomware—on the underlying server. This vulnerability became public on December 9 and was dubbed Log4Shell.[13] The Log4j software is used to log events by millions of systems, including Amazon Web Services, and one estimate suggested that 10% of all digital assets were vulnerable to it.[14][15] Exploiting Log4Shell is also relatively easy, and proof of concept code was posted on GitHub shortly after the vulnerability was revealed.[16]

The combination of widespread use in software and easy exploitation are why Log4Shell was viewed as a "catastrophic" vulnerability. While the final patch fixing the exploit was issued on December 27, the process of deploying those patches to all affected systems is still ongoing.[17] A full accounting of the damage from hacks exploiting this vulnerability is still unknown.

Another significant open-source vulnerability, called Heartbleed, was discovered in 2014 in the OpenSSL cryptographic software library. OpenSSL provides open-source encryption protocols that are used to protect internet communications, and at the time of the attack at least 66% of all internet sites used servers that relied on these protocols.[18] The bug allowed hackers to bypass encryption to steal information and was relatively simple to exploit.[19] A patch was issued within

[10] *Ibid, page 5*
[11] https://googleprojectzero.blogspot.com/2022/02/a-walk-through-project-zero-metrics.html
[12] https://www.synopsys.com/software-integrity/resources/analyst-reports/open-source-security-risk-analysis.html#
[13] https://www.techtarget.com/whatis/feature/Log4j-explained-Everything-you-need-to-know
[14] https://theconversation.com/what-is-log4j-a-cybersecurity-expert-explains-the-latest-internet-vulnerability-how-bad-it-is-and-whats-at-stake-173896
[15] https://www.tenable.com/blog/one-in-10-assets-assessed-are-vulnerable-to-log4shell?utm_source=charge&utm_medium=social&utm_campaign=internal-comms
[16] https://www.tenable.com/blog/cve-2021-44228-cve-2021-45046-cve-2021-4104-frequently-asked-questions-about-log4shell
[17] https://www.zdnet.com/article/log4j-flaw-thousands-of-applications-are-still-vulnerable-warn-security-researchers/
[18] https://heartbleed.com/
[19] https://www.synopsys.com/blogs/software-security/heartbleed-vulnerability-appsec-deep-dive/

a week of discovery and on the same day the vulnerability was made public. However, even years later, hundreds of thousands of devices were still vulnerable to exploitation.[20]

ONGOING ACTIONS TO ADDRESS OPEN-SOURCE SECURITY

Industry Joint Action

In response to the Heartbleed vulnerability, the Linux Foundation organized the Core Infrastructure Initiative (CII) with support from Google, Microsoft, Facebook, and other major technology companies.[21] Over its lifetime CII funded ten grants to pay for security work on critical open-source projects.[22] In 2020, CII transitioned into the Open Source Security Foundation (OpenSSF) with the goal of tackling open-source security more holistically.

The OpenSSF has developed free training courses on secure software development.[23] They also host working groups with the aim of bringing all relevant stakeholders together to work on topics such as best practices for open-source developers.[24] In response to Log4Shell, OpenSSF established an internal project called "Project Alpha-Omega" to directly improve both the most critical open-source software and to create tools that will raise the baseline of security for all open-source software.[25] For example, the first open-source project targeted for security assistance underlies significant parts of most websites.[26]

Repository managers have also taken careful steps to improve security. On February 1, the npm registry that hosts code for JavaScript, a programming language that underpins many internet applications, began requiring two-factor authentication for the top 100 packages. The npm registry intends to roll out two-factor authentication to the top 500 high-impact packages in early 2022.[27] This helps secure these open-source projects from the submission of malicious code through hacked accounts. To secure open-source code itself, GitHub is trialing an AI programming assistant called Copilot that provides live code suggestions as a programmer is working.[28] This program is intended to help programmers produce more secure software.

White House Actions

On May 12, 2021, the Biden Administration released Executive Order 14028, "Improving the Nation's Cybersecurity."[29] Its goal is to address government supply chain security deficiencies in the wake of the SolarWinds hack. While it did not focus specifically on open-source software, much of the guidance produced as a result would affect how software is analyzed, adopted, and secured in Federal systems. For example, the E.O. took several steps to secure Federal software supply chains, from developing security requirements for newly defined "critical software" to

[20] https://blog.malwarebytes.com/exploits-and-vulnerabilities/2019/09/everything-you-need-to-know-about-the-heartbleed-vulnerability/
[21] https://www.coreinfrastructure.org/faq/
[22] https://web.archive.org/web/20190619184614/https://www.coreinfrastructure.org/grants/
[23] https://openssf.org/training/courses/
[24] https://openssf.org/community/openssf-working-groups/
[25] https://openssf.org/press-release/2022/02/01/openssf-announces-the-alpha-omega-project-to-improve-software-supply-chain-security-for-10000-oss-projects/
[26] https://openssf.org/blog/2022/04/18/openssf-selects-node-js-as-initial-project-to-improve-supply-chain-security/
[27] https://github.blog/2021-12-07-enrolling-npm-publishers-enhanced-login-verification-two-factor-authentication-enforcement/
[28] https://copilot.github.com/
[29] https://www.whitehouse.gov/briefing-room/presidential-actions/2021/05/12/executive-order-on-improving-the-nations-cybersecurity/

creating guidance for minimum standards for vendors' testing of their software source code. Pursuant to the E.O., NIST is scheduled to publish additional guidelines for periodic review of software supply chain security later this month.

The White House also held a summit on the security of open-source software with industry and government experts on January 13, 2022.[30] The summit explored how to improve the process of identifying and mitigating vulnerabilities in open-source software and shorten response times. This dialogue is ongoing and more updates are expected in the spring of 2022.

Cybersecurity and Infrastructure Security Agency

The Department of Homeland Security's CISA helps Federal civilian agencies, critical infrastructure entities, and the private sector share cybersecurity information and respond to emerging incidents. CISA provides interagency guidance, coordination, and education activities. Launched in 2019, CISA's Information and Communications Technology (ICT) Supply Chain Risk Management Task Force is a public-private partnership created to improve the nation's ability to assess and mitigate threats to the ICT supply chain, including those from open-source software.[31] The Task Force is made up of industry representatives from the information technology and communications sectors as well as Federal partners like NIST. In addition, in February 2022, CISA released a catalogue of free cybersecurity tools developed in partnership with the open-source community.[32]

CISA has also taken over a multi-stakeholder initiative from the National Telecommunications and Information Administration to develop a Software Bill of Materials (SBOM).[33] Modern software products depend on a vast number of components from different developers, code repositories, and other sources. Suppliers of software components also use different naming schemes for the same components. As a result, identifying which vulnerabilities compromise which products can be a challenging technical feat. SBOMs may be able to address this challenge by creating a machine-readable inventory that will enable software developers and users to track software components and dependencies and make responding to vulnerabilities in the event of an incident more straightforward. However, as the Investigations and Oversight Subcommittee heard during its hearing on Supply Chain Security in May 2021, questions remain about the effectiveness of SBOMs as well as the ability of organizations to adopt them.[34]

National Institute of Standards and Technology

NIST is the agency primarily in charge of the nation's cybersecurity standards and best practices. In 2014, pursuant to E.O. 13636, NIST published a voluntary framework for reducing cybersecurity risks to critical infrastructure.[35] NIST has since updated and expanded its guidance to apply to new scenarios, many of which are applicable to open-source software. By statute, Federal agencies must secure their systems according to directives from the Office of

[30] https://www.whitehouse.gov/briefing-room/statements-releases/2022/01/13/readout-of-white-house-meeting-on-software-security/
[31] https://www.cisa.gov/sites/default/files/publications/ict-scrm-task-force_year-two-report_508.pdf
[32] https://www.cisa.gov/news/2022/02/18/cisa-launches-new-catalog-free-public-and-private-sector-cybersecurity-services
[33] https://www.ntia.gov/SBOM; https://www.cisa.gov/sbom
[34] https://science.house.gov/hearings/solarwinds-and-beyond-improving-the-cybersecurity-of-software-supply-chains
[35] https://www.nist.gov/cyberframework

Management and Budget. Agencies often choose to use NIST's cybersecurity standards and guidelines to protect their non-national security information and communications infrastructure.

NIST has developed several standards and best practices that apply directly to open-source software. NIST offers guidance for organizations to manage the increasing risk of cyber supply chain compromise.[36] Similarly, NIST has produced guidance for vulnerability remediation.[37] The agency has also developed The Secure Software Development Framework (SSDF) to help software developers mitigate vulnerabilities released in software.[38]

E.O. 14028 required NIST to create several new security standards and guidelines for software in Federal systems, including open-source software:

- In June 2021, NIST published a definition of the term "critical software." The Executive Order also directs CISA to develop a list of software categories and products in use or the acquisition process that meet this definition.[39]
- In July 2021, NIST published security guidelines for critical software and minimum standards for vendors' testing of their software source code.[40,41]
- In February 2022, NIST published standards that enhance the security of the software supply chain based on an updated SSDF, including guidance for software developers to provide SBOMs.[42] NIST published revisions to this document on May 5.[43]
- In February 2022, NIST unveiled guidance on practices software producers can undertake to help strengthen the software supply chain.[44]

National Science Foundation

NSF has traditionally played a role in funding open-source software and data repositories across numerous solicitations.[45] NSF is planning to award grants to help secure elements of the open-source ecosystem as part of its new Pathways to Enable Open-Source Ecosystems (POSE) program.[46] The solicitation for proposals closes on May 12, 2022.

ONGOING CHALLENGES

Understanding The Breadth and Depth of the Open-Source Ecosystem

Because of the protean nature of open-source software projects and the myriad developers who contribute to them, a single comprehensive resource tracking all such projects is not feasible. The percentage of code in surveyed codebases that was open source increased from 36% in 2015 to 78% in 2022.[47] In 2018, the Sloan Foundation's Critical Digital Infrastructure Research Fund

[36] https://nvlpubs.nist.gov/nistpubs/SpecialPublications/NIST.SP.800-161.pdf
[37] https://csrc.nist.gov/publications/detail/sp/800-40/rev-3/final
[38] https://csrc.nist.gov/publications/detail/sp/800-218/final
[39] https://www.nist.gov/itl/executive-order-improving-nations-cybersecurity/critical-software-definition
[40] https://www.nist.gov/itl/executive-order-improving-nations-cybersecurity/recommended-minimum-standard-vendor-or-developer
[41] https://www.nist.gov/itl/executive-order-improving-nations-cybersecurity/security-measures-eo-critical-software-use
[42] https://www.nist.gov/itl/executive-order-improving-nations-cybersecurity/software-supply-chain-security-guidance
[43] https://www.nist.gov/news-events/news/2022/05/nist-updates-cybersecurity-guidance-supply-chain-risk-management
[44] https://www.nist.gov/itl/executive-order-improving-nations-cybersecurity/software-supply-chain-security-guidance
[45] https://www.nsf.gov/awardsearch/showAward?AWD_ID=1348450&HistoricalAwards=false
[46] https://beta.nsf.gov/funding/opportunities/pathways-enable-open-source-ecosystems-pose
[47] *Ibid*

provided funding for thirteen grants to study various aspects of the ecosystem.[48] However, those projects were narrow in scope and did not provide the bird's-eye view of the ecosystem needed to identify crucial open-source software so that resources could be allocated appropriately.

The Linux Foundation has performed a pair of censuses of well-characterized and manageable subsets of the overall ecosystem. The first census, conducted in partnership with the Core Infrastructure Initiative in 2015, sought to identify packages within a particular Linux distribution that were essential to that operating system's security.[49] The second census, completed in March 2022 in partnership with the Harvard Laboratory for Innovation Science, identified the 1,000 most widely deployed open-source libraries in commercial and enterprise applications.[50] This list is meant to help target security efforts towards software with the greatest potential for impactful hacks.

The Federal government also has a role to play in identifying and cataloguing critical software. NIST's publication of a definition of the term "critical software" can assist in identifying the open-source project that most need resources, though NIST's focus is on government software and may not be identical with the software critical to industry.[51] Section 10224 of the *America COMPETES Act of 2022* seeks to address this issue by directing NIST to assess and identify security risks in open-source software.[52]

Resources of the Open-Source Ecosystem Vary Widely

Open-source software is often under-resourced because it lacks the commercial support of proprietary software and because a significant percentage of its developers are volunteers. [53] For instance, prior to the Heartbleed vulnerability, the organization maintaining OpenSSL received an average of just $2,000 per year in donations for their work.[54] With regard to individual developers, a 2020 survey found that 44% received no compensation for their open source work, 49% were compensated by their employer, though many of those also said they worked on additional open-source projects for free. Only 3% were paid by a third party.[55] Focused spending may be needed to encourage work to detect vulnerabilities before they become public. However, it may be challenging to acquire or distribute funds in the volunteer-heavy open-source community.

Industry has attempted to expand the funding dedicated to open-source security in several ways. The Chan Zuckerberg Initiative's Essential Open Source Software for Science program has granted $22.9 million to open-source projects that support biological research since May 2019.[56] In October of 2021 Google announced it was providing $1 million for a pilot program called SOS Rewards to reward improvements in the security of critical open-source software.[57] Google

[48] https://www.fordfoundation.org/campaigns/critical-digital-infrastructure-research/
[49] https://www.coreinfrastructure.org/programs/census-program-i/
[50] https://www.linuxfoundation.org/press-release/the-linux-foundation-and-harvards-lab-for-innovation-science-release-census-of-most-widely-used-open-source-application-libraries/
[51] https://www.nist.gov/system/files/documents/2021/10/13/EO%20Critical%20FINAL.pdf
[52] https://www.congress.gov/bill/117th-congress/house-bill/4521/text/eh
[53] https://xkcd.com/2347
[54] https://www.coreinfrastructure.org/faq/
[55] https://www.linuxfoundation.org/wp-content/uploads/2020FOSSContributorSurveyReport_121020.pdf
[56] https://chanzuckerberg.com/eoss/
[57] https://www.techrepublic.com/article/google-stakes-new-secure-open-source-rewards-program-for-developers-with-1m-seed-money/

also partnered with Microsoft to contribute $5 million to the OpenSSF's Project Alpha-Omega.[58] However, these contributions pale in comparison to the scope of the open-source ecosystem. Moreover, simple infusions of money may not be sufficient to increase security. One of the volunteers working on Log4j said that more funding would have been unlikely to catch the vulnerability, and that instead more involvement by knowledgeable volunteers would go a long way toward increasing the security of the project.[59]

Unique Risks of Open-Source Software for Critical Technologies

Open-source powers technologies ranging from drones to quantum computing to AI. While these applications face similar vulnerabilities to other open-source applications, there are sometimes unique challenges to critical technologies built with open-source software or data. For example, machine learning often requires massive datasets to improve the accuracy of the model. Many organizations or researchers have produced open-source datasets to allow other researchers or developers to train their AI systems.[60] However, malicious actors could theoretically make alterations to these freely available datasets to manipulate an AI system to produce an inaccurate or harmful result. For instance, malicious data might cause an AI to disregard anomalies that would have revealed an intrusion into the system it monitors. One researcher found that poisoning just 0.7% of a dataset was sufficient to bypass defenses.[61]

Another common use of open source within AI development is the sharing of pre-trained models. These models have already been tuned by their developers to produce an intended result, like language processing or image generation. Less technically savvy individuals can then apply the pre-trained model to new tasks or situations.[62] Similar to the issue of poisoned datasets, researchers have found that it is possible to place a backdoor in a model which can provide malicious outcomes only when instructed and is otherwise indistinguishable from a clean model.[63] When the model's code is open source, malicious actors can find weaknesses in the system or create results that were unintended by the developers. For example, when Facebook released a new AI language model called OPT-175B, security researchers were easily able to cause the model to "generate toxic language and reinforce harmful stereotypes."[64] While exploits like these have not yet been detected in the wild, they become more likely as AI systems become more heavily integrated into society.

[58] https://openssf.org/press-release/2022/02/01/openssf-announces-the-alpha-omega-project-to-improve-software-supply-chain-security-for-10000-oss-projects/
[59] https://therecord.media/the-apache-log4j-team-talks-about-the-log4shell-patching-process/
[60] https://cset.georgetown.edu/wp-content/uploads/CSET-Poison-in-the-Well.pdf
[61] https://www.bloomberg.com/opinion/articles/2022-04-24/ai-poisoning-is-the-next-big-risk-in-cybersecurity
[62] https://cset.georgetown.edu/wp-content/uploads/CSET-Poison-in-the-Well.pdf
[63] https://arxiv.org/abs/2204.06974
[64] https://arxiv.org/pdf/2205.01068.pdf