# Decoding Log4j Vulnerability Lessons Learned

Organizations that adopt technology for their business must carefully tread the path of managing the risk (e.g., financial, reputational, operational, legal) associated with vulnerabilities. However, vulnerabilities in the information age are increasingly tough to manage and their impact goes far beyond quantification. One such global vulnerability is the Log4j weakness. Attackers' exploitation of vulnerabilities in the Log4j component in November and December 2021 impacted countless organizations around the world and will continue to be a weakness for years to come. More than 35,000 Java packages, amounting to more than 8 percent of the Maven Central Repository (the most significant Java package repository), were affected by Log4j vulnerabilities.[1] A report by the Cyber Safety Review Board (CSRB) noted that the Log4j vulnerability impacted virtually every networked organization, and the severity of the threat required fast action.[2] However, the absence of a comprehensive customer list for Log4j, or even a list of where it is integrated as a subsystem, hindered a speedy response. Organizations and vendors alike struggled to discover where they used Log4j, and security enthusiasts and hackers compounded the problem by combining vulnerabilities, which contributed to confusion and response fatigue.[3] Given the significance of this event, there are many lessons to be learned and improvements to be made to organizations' cyber capabilities.

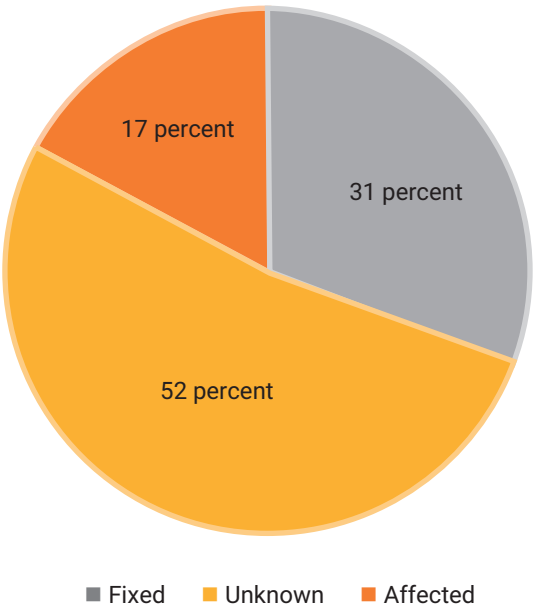
## Log4j Vulnerability

Log4j is a Java-based open-source software product used by many developers to collect and manage information about system activities.[4] Although Log4j has been around since 2001, the vulnerability was identified in Log4j version 2, released in 2014 for general use by the Apache Software Foundation (ASF).

Based on crowdsourced information, the Cybersecurity and Infrastructure Security Agency (CISA) tracked a list of vendors affected by the Log4j vulnerability.[5] Although this list is still evolving, an analysis of trends revealed (**figure 1**):

**FIGURE 1**

Percentage of Vendors Impacted by Log4j




- Out of a total of 2,100 software vendors listed across 26 pages, the Log4j vulnerability has been fixed by approximately one-third of those vendors.
- For more than 50 percent of the tracked vendors, the remediation status of this vulnerability and/or its applicability are unknown.[6]

Given the widespread use of this software, the Log4j vulnerability has already been dubbed an "endemic" vulnerability.[7]

**NINAD DHAVASE**

Works with a big four consulting firm in Australia and has 13 years of experience in cybersecurity, with a focus on the financial services sector, including banking, capital markets, insurance and payments domains. His previous experience includes working with India's largest stock exchange and clearing corporation, leading the cybersecurity governance, risk and compliance management areas. He volunteers as an *ISACA® Journal* article reviewer and has been a presenter at public forums and training sessions. The views expressed here are his own.

## Cyber Safety Review Board

In light of events surrounding the Log4j vulnerability, US President Joe Biden issued an executive order establishing the US Cyber Safety Review Board (CSRB) to review major cyberevents, make concrete recommendations to enhance cybersafety in the public and private sectors, and improve the state of national cybersecurity in the United States. One of the key advantages of working in the cybersecurity field is its close-knit community, which shares and exchanges information to build collective cyberdefense capabilities. The CSRB includes both US government and private-sector members, and it has a direct path to the US secretary of homeland security and the US president to ensure that its recommendations are addressed and implemented.[8]

The CSRB's first assignment was to review the significant cyberevents associated with exploitation of the Log4j vulnerability and make recommendations to help the cybercommunity improve its defense and response capabilities.

## Bringing CSRB Recommendations to Life

The CSRB's report is organized into four themes and 19 recommendations (**figure 2**).

**FIGURE 2**

### CSRB Themes and Recommendations

| Theme 1 | Address the continuing risk of Log4j. |
|---|---|
| 1.1 Organizations should be prepared to address Log4j vulnerabilities for years to come.<br>1.2 Organizations should continue to report (and escalate) observations of Log4j exploitation.<br>1.3 CISA should expand its capability to develop, coordinate and publish authoritative cyberrisk information.<br>1.4 US Federal and state regulators should drive implementation of CISA guidance through their own regulatory authorities. | |
| **Theme 2** | **Drive existing best practices for security hygiene.** |
| 2.1 Organizations should invest in capabilities to identify vulnerable systems.<br>2.2 Organizations should develop the capacity to maintain an accurate IT asset and application inventory.<br>2.3 Organizations should have a documented vulnerability response program.<br>2.4 Organizations should have a documented vulnerability disclosure and handling process.<br>2.5 Software developers and maintainers should implement secure software practices. | |
| **Theme 3** | **Build a better software ecosystem.** |
| 3.1 Open-source software developers should participate in community-based security initiatives.<br>3.2 Organizations should invest in training for software developers in secure software development.<br>3.3 Organizations should improve software bill of materials (SBOM) tooling and adoptability.<br>3.4 Organizations should increase investments in open-source software security.<br>3.5 Organizations should pilot open-source software maintenance support for critical services. | |
| **Theme 4** | **Invest in the future.** |
| 4.1 Organizations should explore a baseline requirement for software transparency for US federal government vendors.<br>4.2 Organizations should examine the efficacy of a cybersafety reporting system.<br>4.3 Organizations should explore the feasibility of establishing a software security risk assessment center of excellence (SSRACE).<br>4.4 Organizations should study the incentive structures required to build secure software.<br>4.5 Organizations should establish a government-coordinated working group to improve identification of software with known vulnerabilities. | |

Source: Adapted from Cyber Safety Review Board (CSRB), *Review of the December 2021 Log4j Event*, USA, 2022, *https://www.cisa.gov/sites/default/files/publications/CSRB-Report-on-Log4-July-11-2022_508.pdf*

**FIGURE 3**

### Key Attributes and Dimensions of Identified Themes

| Theme | Approximate Time Horizon | Investments/Sponsorship | External Collaboration |
|---|---|---|---|
| 1 | Immediate | None–minimal | Minimal |
| 2 | Near term | Minimal | Moderate |
| 3 | Mid term | Moderate | Moderate |
| 4 | Long term | High | High |

**FIGURE 4**

## Breakdown of CSRB Recommendations by Stakeholder

| # | CSRB Recommendation | Organizations | Regulators/ Government Agencies | Software Vendors/ Related Ecosystem |
|---|---|---|---|---|
| 1.1 | Organizations should be prepared to address Log4j vulnerabilities for years to come. | O | O | O |
| 1.2 | Organizations should continue to report (and escalate) observations of Log4j exploitation. | O | O | O |
| 1.3 | CISA should expand its capability to develop, coordinate and publish authoritative cyberrisk information. | C | O | C |
| 1.4 | US federal and state regulators should drive implementation of CISA guidance through their own regulatory authorities. | C | O | C |
| 2.1 | Organizations should invest in capabilities to identify vulnerable systems. | O | O | O |
| 2.2 | Organizations should develop the capacity to maintain an accurate IT asset and application inventory. | O | O | O |
| 2.3 | Organizations should have a documented vulnerability response program. | O | O | O |
| 2.4 | Organizations should have a documented vulnerability disclosure and handling process. | O | O | O |
| 2.5 | Software developers and teams who maintain the software should implement secure software practices. | O | O | O |
| 3.1 | Open-source software developers should participate in community-based security initiatives. | C | C | O |
| 3.2 | Organizations should invest in secure software development training for software developers. | O | O | O |
| 3.3 | Organizations should improve SBOM tooling and adoptability. | O | O | O |
| 3.4 | Organizations should increase investments in open-source software security. | C | O | O |
| 3.5 | Organizations should pilot open-source software maintenance support for critical services. | C | O | C |
| 4.1 | Organizations should explore a baseline requirement for software transparency for US federal government vendors. | O | O | C |
| 4.2 | Organizations should examine the efficacy of a cybersafety reporting system. | C | O | C |
| 4.3 | Organizations should explore the feasibility of establishing an SSRACE. | C | O | C |
| 4.4 | Organizations should study the incentive structures required to build secure software. | C | O | O |
| 4.5 | Organizations should establish a US government-coordinated working group to improve identification of software with known vulnerabilities. | C | O | C |

**FIGURE 5**

## Theme 1 Actions and Considerations

| Action/Initiative | Key Considerations for Organizations |
|---|---|
| 1.A Continue to assess for Log4j vulnerability. | • Organizations should diligently assess areas where technology might be impacted by Log4j vulnerabilities and upgrade those technologies.<br>• Technologies that are vulnerable to previous versions of Log4j should not be introduced, and other emerging vulnerabilities should not be ignored. Discovery and remediation of vulnerabilities are baseline expectations in security, and vigilance helps secure the technology environment and reduce exposure. |
| 1.B Continue to report and share Log4j-related information. | • Organizations should continue to identify, share and report incidents and information about Log4j.<br>• A key challenge identified in the CSRB report is the lack of authoritative and credible information sources, especially about cyberevents that affect a large number of organizations. Therefore, all stakeholders across the cyberecosystem should work together to develop credible information that can be validated and shared within the community to enable a faster response. Many regulators and government agencies have platforms that enable this information exchange, and they should be utilized for optimal coordination. |
| 1.C Collaborate and exchange information about Log4j. | |

**FIGURE 6**

## Theme 2 Actions and Considerations

| Action/Initiative | Key Considerations for Organizations |
|---|---|
| 2.A Maintain an accurate and up-to-date asset and application inventory. | • Automation and tooling for the maintenance of IT asset management information, including applications, should be considered. Key characteristics should be aligned with US National Institute of Standards and Technology (NIST) Special Publication (SP) 1800-5.[a] |
| 2.B Scan and remediate vulnerabilities. | • Vulnerability scanning technologies should be deployed to identify vulnerable software and enhance the speed of remediation. |
| 2.C Maintain an accurate SBOM. | • This is one of the CSRB's most important and impactful recommendations. An SBOM is a nested inventory—a list of ingredients that make up software components. It has emerged as a key building block in software security and software supply chain risk management.[b]<br>• The US National Telecommunications and Information Administration (NTIA) is developing standards that organizations can adopt to develop SBOMs (the equivalent of an ingredients list on prepackaged food).[c]<br>• Leading software providers are adopting these standards,[d, e] and organizations that use software should demand this practice as part of their procurement and software supply chain risk management process. |
| 2.D Create a vulnerability response playbook. | • The objective is to help organizations address Log4j-like vulnerabilities, whether such weaknesses are reported publicly or privately.<br>• A playbook is especially useful when software code is not available and a swift response is needed.<br>• CISA has released a reference vulnerability response playbook that includes four phases: identification, evaluation, remediation and reporting/notification.[f]<br>• To make the CISA playbook more relevant to a particular organization, it can be expanded by outlining roles and responsibilities, definitions and timelines and linking these aspects to existing policies and processes, such as cybercrisis and vulnerability management.<br>• This plan must be tested and amended over time to improve capabilities. |
| 2.E Create a vulnerability disclosure and handling process. | • Vulnerabilities are found not only in code; there may be misconfigurations in applications, infrastructure and network systems.<br>• A misconfiguration somewhere in a system is likely, making it important to respond quickly, before an attacker discovers it. A vulnerability disclosure and handling process enables end users to report vulnerabilities, improving an organization's security posture and avoiding systemic risk.<br>• Aspects such as prioritization, remediation, communication (internal and external) and integration with third-party suppliers must be considered.<br>• There are many popular technology platforms that can be leveraged to run a disclosure and remediation program, leading to benefits when integrated with in-house policies and processes. |
| 2.F Require secure development practices by software developers. | • All stakeholders should encourage their software developers to implement industry best practices, such as those outlined in International Organization for Standardization/International Electrotechnical Commission (ISO/IEC) standard ISO/IEC 27034-1:2011 *Information Technology—Security Techniques—Application Security*[g] and NIST's Secure Software Development Framework.[h]<br>• Development teams should embed practices such as secure code scanning, vulnerability assessment and management to better equip themselves to deal with threats that may be injected inadvertently.<br>• Development teams should implement consistent and comprehensive approaches to document information about software components that could lead to better understanding, discovery, remediation and coordination during cyberevents. |

Sources: a) Stone, M.; C. Irrechukwu; H. Perper; D. Wynne; L. Kauffman; National Institute of Standards and Technology (NIST) Special Publication (SP) 1800-5 *IT Asset Management*, USA, September 2018, *https://nvlpubs.nist.gov/nistpubs/SpecialPublications/NIST.SP.1800-5.pdf*; b) Cybersecurity and Infrastructure Security Agency (CISA), "Software Bill of Materials (SBOM)," USA, *https://www.cisa.gov/sbom*; c) National Telecommunications and Information Administration (NTIA), "Software Bill of Materials," USA, *https://ntia.gov/page/software-bill-materials*; d) Badlani, D. K.; A. Diglio; "Microsoft Open Sources Its Software Bill of Materials (SBOM) Generation Tool," Engineering@Microsoft, 12 July 2022, *https://devblogs.microsoft.com/engineering-at-microsoft/microsoft-open-sources-software-bill-of-materials-sbom-generation-tool/*; e) Lum, B.; M. Maruseac; I. Hepworth; "Announcing GUAC, a Great Pairing With SLSA (and SBOM)!" Google Security Blog, 10 October 2022, *https://security.googleblog.com/2022/10/announcing-guac-great-pairing-with-slsa.html*; f) Cybersecurity and Infrastructure Security Agency, *Cybersecurity Incident and Vulnerability Response Playbooks*, USA, November 2021, *https://www.cisa.gov/sites/default/files/publications/Federal_Government_Cybersecurity_Incident_and_Vulnerability_Response_Playbooks_508C.pdf*; g) International Organization for Standardization/International Electrotechnical Commission (ISO/IEC), ISO/IEC 27034-1:2011 *Information technology—Security techniques—Application security*, Switzerland, 2011, *https://www.iso.org/standard/44378.html*; h) Souppaya, M.; K. Scarfone; D. Dodson; National Institute of Standards and Technology (NIST) SP 800-218 *Secure Software Development Framework, Version 1.1*, USA, February 2022, *https://csrc.nist.gov/publications/detail/sp/800-218/final*

An analysis of these themes indicates that the Log4j vulnerability highlighted several capability gaps that need to be fixed. The widespread attack was a wake-up call for both private- and public-sector organizations to adopt an all-hands-on-deck approach to addressing these gaps and collectively reducing the impact of similar events in the future. The foundation of cybersecurity risk management is to minimize the impact of attacks across various dimensions—business, revenue, compliance, reputation—and to recover the systems, data and business within a reasonable time. Further, because resources (i.e., scope, cost, schedule) are always limited, prioritization is necessary to reap the benefits from investments.

Broadly, if the CSRB's four themes were categorized based on a few key attributes, the distribution would be as illustrated in **figure 3**.

The themes and recommendations from the CSRB report are directed toward a range of stakeholders, including organizations, regulators, government agencies, software vendors, developers and technology professionals. **Figure 4** illustrates how these recommendations would be distributed among stakeholders based on primary ownership (O) and contribution/consultation (C). This is only an approximation; the reality may vary, depending on the context and technologies under consideration.

### Theme 1: Address the Continuing Risk of Log4j

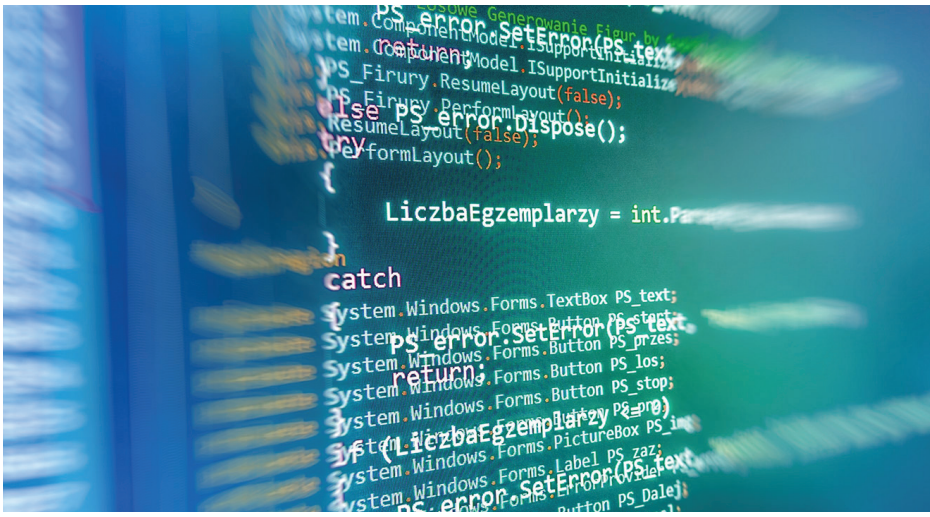
**Figure 5** outlines the key actions and considerations recommended by the CSRB under Theme 1.

### Theme 2: Drive Existing Best Practices for Security Hygiene

**Figure 6** outlines the key actions and considerations recommended by the CSRB under Theme 2.

### Theme 3: Build a Better Software Ecosystem

**Figure 7** outlines the key actions and considerations recommended by the CSRB under Theme 3.




### Theme 4: Invest in the Future

**Figure 8** outlines the key actions and considerations recommended by the CSRB under Theme 4.

## Conclusion

The CSRB report is an important compilation of lessons learned from the Log4j cyberevent.[9] Although such events can shake the belief (i.e., digital trust) in organizations and technology, they also provide opportunities to advance research and strengthen

**FIGURE 7**

## Theme 3 Actions and Considerations

| Action/Initiative | Key Considerations for Organizations |
|---|---|
| 3.A Participate and invest in better security in open-source development. | There are a number of areas in which organizations can diversify their budgets and contribute to building a secure software ecosystem:<br>• Open Source Security Foundation (OpenSSF) offers free training resources.[a] It has also developed a project to automate analysis and trust decisions based on the security posture of open-source projects.[b]<br>• The Open Web Application Security Program (OWASP) Foundation is a leading community-led open-source initiative that focuses on improving software security by tracking weaknesses and providing tools and resources for all.[c]<br>• Led by the Linux Foundation and OpenSSF, a platform of community representatives including open-source developers, commercial representatives and experts from US federal agencies has committed to common minimum and high-impact actions to improve the resilience and security of open-source software. This detailed plan outlines goals and a number of activity streams in the areas of education, risk assessment, digital signatures, memory safety, incident response and code audits.[d]<br>• Those new to software development and computer science should adopt secure software development practices through education and appropriate funding.<br>• Adequate funding and incentives should be provided to help the open-source community upgrade and maintain existing open-source software. This ensures that vulnerabilities are remediated as they are discovered and that code is maintained as technology advances. |
| 3.B Train software developers in secure software development. | |
| 3.C Collaborate to improve open-source software maintenance support. | |

Sources: a) Open Source Security Foundation (OpenSSF), "Secure Software Development Fundamentals Courses," *https://openssf.org/training/courses/*; b) Github, "OSSF Scorecard," *https://github.com/ossf/scorecard*; c) Open Web Application Security Program (OWASP), "About the OWASP Foundation," *https://owasp.org/about/*; d) Open Source Security Foundation (OpenSSF), "The Open Source Security Mobilization Plan," *https://openssf.org/oss-security-mobilization-plan/*

**FIGURE 8**

## Theme 4 Actions and Considerations

| Action/Initiative | Key Considerations for Organizations |
|---|---|
| 4.A Develop a baseline requirement for software transparency information. | • Organizations should define minimum or baseline requirements that outline expectations related to understanding the open-source components used, known vulnerabilities software may be exposed to, service levels to fix vulnerabilities when discovered and the type of support expected when a critical security event occurs.<br>• These requirements should be built into contracts, and organizations and vendors should work together to develop, distribute and maintain secure software. Risk management practices can help operationalize such requirements. |
| 4.B Collaborate externally to identify, coordinate and report software vulnerabilities. | • Organizations and their external stakeholders, such as sectoral regulators, industry associations or government agencies, must jointly develop platforms that encourage and incentivize the discovery, reporting and remediation of software vulnerabilities.<br>• Such platforms may serve as a catalyst for coordinated detection and response in the case of a large security event such as Log4j that impacts industry and government agencies alike. |
| 4.C Collaborate externally to establish an SSRACE. | • Organizations reuse numerous open-source software products and components as part of their software development and adoption life cycles, and each organization conducts its own risk assessment and maintains inventory for these components. What if this were centralized?<br>• A mutually trusted party, such as an industry association, could be funded to provide centralized services consisting of asset or inventory management, code maintenance, response to vulnerabilities, prioritization and risk management and lead the overall effort for secure software adoption at the industry level. |
| 4.D Build and adopt secure software. | • Although a lot needs to be done at the broader software ecosystem level to develop and adopt secure software, organizations should establish their own specific criteria and incentivize in-house or vendor-supported application maintenance and development teams to adopt secure development practices.<br>• By providing the right set of tools and incentives for individuals to develop secure software (e.g., swag material for shipping a zero-issue code module), an industry can be encouraged to adopt best practices. |

**LOOKING FOR MORE?**

- Learn more about, discuss and collaborate on information and cybersecurity in ISACA's Online Forums. *https://engage.isaca.org/onlineforums*

the control ecosystem. This leads to technologies that are resilient and engender confidence in the tech ecosystem. Both private and public organizations should do more than secure their own environments; they should also help secure the technology ecosystem. After all, as Vincent Van Gogh said, "Great things are done by a series of small things brought together."[10]

## Endnotes


1 Wetter, J.; N. Ringland; "Understanding the Impact of Apache Log4j Vulnerability," Google Security Blog, 17 December 2021, *https://security.googleblog.com/2021/12/understanding-impact-of-apache-log4j.html*

2 Cyber Safety Review Board (CSRB), *Review of the December 2021 Log4j Event*, USA, 2022, *https://www.cisa.gov/sites/default/files/publications/CSRB-Report-on-Log4-July-11-2022_508.pdf*

3 *Ibid*.

4 *Ibid*.

5 Github, "Log4j Affected DB/Software Lists," *https://github.com/cisagov/log4j-affected-db/tree/develop/software_lists*

6 *Ibid*.

7 Volz, D.; "Major Cyber Bug in Log4j to Persist as 'Endemic' Risk for Years to Come, U.S. Government Board Finds," *The Wall Street Journal*, 14 July 2022, *https://www.wsj.com/articles/major-cyber-bug-in-log4j-to-persist-as-endemic-risk-for-years-to-come-u-s-government-board-finds-11657796400*

8 Cybersecurity and Infrastructure Security Agency (CISA), "Cyber Safety Review Board (CSRB)," USA, *https://www.cisa.gov/cyber-safety-review-board*

9 *Op cit* CSRB

10 Van Gogh, V.; *https://www.brainyquote.com/quotes/vincent_van_gogh_120866*