THE LINUX FOUNDATION | Research

Strengthening License Compliance and Software Security with SBOM Adoption

A Definitive SBOM Guide for Enterprises

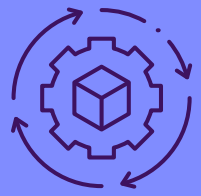
Ibrahim Haddad, Ph.D., *The Linux Foundation*
Foreword by Melissa Evers, *Intel*

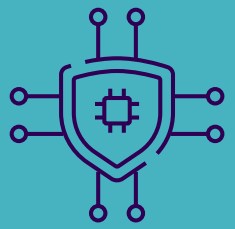
August 2024

Strengthening License Compliance and Software Security with SBOM Adoption

The Linux Foundation launched the Software Package Data Exchange (**SPDX**) project in 2009, a **major milestone toward SBOM standardization**.

The US Executive Order 14028 **mandates federal agency use of SBOMs** for software procurement to **enhance supply chain security** amid rising cyber threats.





One of the key components of the European Union's **Cyber Resilience Act** is the introduction of a **recommended SBOM**, ensuring products are secure by design.

SBOMs safeguard software supply chains and **bolster national cybersecurity posture**, regardless of industry type or technology domain.

A **SBOM** is a **comprehensive, machine-readable inventory** detailing the constituent software components within an application, system, or software stack.

SBOMs typically comprise **5 key elements**: component inventory, origin information, dependency relationships, vulnerability intelligence, and metadata & annotations.

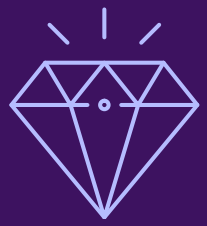
SBOMs are crucial for license compliance & cybersecurity, offering organizations essential insights into software components to **ensure license adherence & enhance cyber defenses**.

SBOMs empower license compliance teams to **mitigate legal, reputational, technical, & financial risks** associated with license violations.

SBOMs serve as early warning systems, enabling preemptive mitigation of security risks before they escalate & facilitating streamlined incident response & patch management efforts.

SBOM functionality is typically embedded as part of **software composition analysis (SCA) tools** to ensure open source license compliance & improve code security.

For effective implementation, organizations must **establish clear policies & roles** that help integrate SBOMs into compliance & security practices.



Organizations must **perform regular & timely updates** of SBOMs & **monitor the effectiveness of their implementation**.

Abstract

Software Bill of Materials (SBOMs) have emerged as a critical tool for enhancing transparency, license compliance, and security within software supply chains. This paper delves into the role of SBOMs, discussing their importance in ensuring license compliance, mitigating cybersecurity risks, and facilitating regulatory adherence. The paper offers a comprehensive overview of SBOMs, including a historical perspective, and highlights their importance in today's complex software landscape. Furthermore, it explores the legislative context surrounding SBOMs, including recent mandates such as the United States' Executive Order 14028, Improving the Nation's Cybersecurity, and the E.U. Cyber Resilience Act, as well as the industry-driven effort from the Linux Foundation—SPDX—an open standard for SBOM representation. The report concludes by offering actionable recommendations for effective SBOM implementation to enable organizations to enhance transparency, mitigate risks, and build more resilient software ecosystems.

Contents

Foreword

In the intricate network of today's software supply chains, Software Bills of Materials (SBOMs) stand as both the blueprint and the toolset that guide us toward a future of greater transparency, compliance, and security within our software infrastructure. Far from being mere instruments, SBOMs are the strategic keystones that deepen our grasp and governance of the myriad software components, licenses, and potential vulnerabilities.

The value of SBOMs extends well beyond simple record-keeping. They are pivotal in enabling license compliance, reducing cybersecurity risks, and meeting the demands of rigorous regulatory frameworks. In the current era, it is uncommon to find software developed entirely by a single party. Instead, most software emerges as a complex mosaic, richly interlaced with contributions from countless open-source initiatives and commercial ventures. Amidst this dense network of interdependencies, a solitary component can become the critical factor in the security of the entire construct. As the next layers of software abstractions and compound AI system use cases become mainstream, the complexity and interdependency of our software as an industry draws even more attention. And when the interconnectivity of software systems grows, the protective role of SBOMs in defending these systems from threats and maintaining their operational integrity becomes increasingly crucial.

Recent governmental mandates have propelled SBOMs to the forefront of cybersecurity strategy, reflecting a global recognition of the need for standardized practices in software transparency. And it's in this rapidly evolving landscape that the Linux Foundation's stewardship, along with the long-established contributions from Intel and other participating entities, that we realize the value of work in the Software Package Data Exchange (SPDX), OpenChain, and other projects that establish the open standard for SBOM representation and the processes of handling them, uniting industry stakeholders in a common goal to streamline and improve compliance.

As we champion the strategic implementation of SBOMs, we are not just advocating for a set of practices; we are fostering a culture of openness, collaboration, and re-use that strengthens the very fabric of our digital infrastructure. The collective efforts to adopt and refine SBOMs are a testament to the industry's dedication to building resilient software ecosystems.

We stand at a crossroads where the decisions we make today will shape the security and compliance landscape of tomorrow. By embracing SBOMs and the collaborative spirit of initiatives like SPDX and OpenChain, we can ensure a future where software not only powers our world but does so with the utmost integrity and trust. Welcome to the era of enhanced transparency—a future we secure together.

Melissa Evers, Intel Corporation Vice President, Software and Advanced Technology

Introduction

In today's fast-paced technology landscape, software in general and open source software in particular play an increasingly pivotal role in driving innovation and competitiveness, requiring robust governance mechanisms to ensure license compliance and security. Enter the Software Bill of Materials (SBOM)—a sophisticated and indispensable tool that offers unprecedented visibility and control over software components within an organization's ecosystem. This paper aims to highlight the strategic significance of SBOMs in strengthening the practice of software security and license compliance, navigating through legislative requirements and industry initiatives, and offering actionable strategies to support organizations in their SBOM adoption plans.

Historical Perspective

While the concept of a bill of materials has been a standard part of the product supply chain for the last 50 years, as functionality shifts to software, the software supply chain takes on greater importance. The concept of SBOMs has steadily evolved over the years in response to the growing complexity of software supply chains and the increasing importance of transparency and accountability in software development practices. Initially, SBOMs gained traction within highly regulated industries, such as aerospace, automotive, and defense, where stringent compliance requirements necessitated meticulous documentation of software components. However, it wasn't until the proliferation of open source software and the advent of cloud computing that SBOMs began to garner broader attention across diverse sectors. As software ecosystems expanded and dependencies proliferated, the need for standardized SBOM frameworks became increasingly apparent, laying the groundwork for industry-driven initiatives aimed at promoting SBOM adoption.

The Linux Foundation launched the SPDX (Software Package Data Exchange) project in 2009, a major milestone toward SBOM standardization. SPDX aimed to develop a common standard for documenting and sharing SBOMs, fostering interoperability and collaboration. Recently, initiatives such as OWASP's CycloneDX emerged to address the need for tracking dependencies as another lightweight, machine-readable format for SBOMs, further advancing the cause of SBOM standardization.

SBOM momentum received a boost with the issuance of the United States' **Executive Order (EO) 14028**, Improving the Nation's Cybersecurity, in May 2021. This EO underscored the critical importance of enhancing software supply chain security in light of escalating cyber threats and directed federal agencies to adopt SBOMs for software procurement. It called for the establishment of minimum elements for SBOMs (discussed in a later section) and laid the groundwork for broader adoption through collaborative efforts. As a result, SBOMs have transitioned from being used in specific, highly regulated industries within a narrow context to a broader component of legislative and regulatory efforts aimed at safeguarding software supply chains and bolstering the national cybersecurity posture—regardless of industry type or technology domain.

SBOM Basics

In understanding the critical role of SBOMs in improving software security and compliance, it's essential to learn the fundamental concepts underlying their utility and composition. At its core, an SBOM serves as a comprehensive, machine-readable inventory detailing the constituent software components within an application, system, or software stack. Analogous to a bill of materials in traditional manufacturing, an SBOM offers transparency and visibility into the software supply chain, facilitating robust software governance and risk management practices.

An SBOM typically comprises the following five key elements: component inventory, origin information, dependency relationships, vulnerability intelligence, and metadata and annotations. We briefly explain each of these elements in the following subsections.

Component Inventory

The component inventory is a detailed list of all software components, including both open source and proprietary components, along with their respective versions and dependencies.

Provenance Information

The provenance information includes metadata covering the origin and ownership of each software component, including licensing terms, copyright attributions, and contributors.

Dependency Relationships

The dependency relationships are hierarchical relationships delineating dependencies between software components, facilitating traceability and impact analysis.

Vulnerability Intelligence

Vulnerability intelligence includes detailed information related to known security vulnerabilities associated with each software component in the component inventory. The goal of this intelligence is to enable proactive risk mitigation and vulnerability management.

Metadata and Annotations

Metadata and annotations are additional contextual information enriching the SBOM, such as build instructions, release notes, and compliance attestations.

Criticality of SBOMs

SBOMs play a pivotal role in both license compliance and cybersecurity. From ensuring adherence to applicable licenses (including both commercial and open source licenses) to improving defenses against cyber threats, SBOMs provide organizations with invaluable insights into their software components. In the following subsections, we will delve deeper into the role SBOMs play from both license compliance and security perspectives.

License Compliance

From the standpoint of license compliance, SBOMs provide a strategic advantage by offering comprehensive insights into the intricate web of software components underpinning organizational operations. In today's environment, where organizations depend on open source software in building their software stack, it is critical to catalog open source and proprietary software assets, understand license obligations, and have a plan to execute and fulfill these obligations. The **Synopsys 2024 report** declared that 77% of all source code in the total codebases that they scanned originated from open source. This is a very significant number, showcasing the degree of dependency on open source software. SBOMs empower license compliance teams to mitigate legal, reputational, technical, and financial risks associated with license violations. The absence of SBOMs exposes organizations to many license compliance pitfalls, necessitating manual and error-prone methods of software asset tracking that undermine operational efficiency and expose organizations to unnecessary risks.

Security

In the realm of cybersecurity, SBOMs emerge as a critical asset in improving organizational resilience against evolving threats within the software supply chain. By proactively identifying and addressing vulnerabilities in third-party components, SBOMs serve as early warning systems, enabling organizations to preemptively mitigate security risks before they escalate into full-blown breaches. Moreover, SBOMs facilitate streamlined incident response and patch management efforts, equipping organizations with the agility and foresight needed to safeguard critical assets and avoid exploits of security vulnerabilities.

Legislative Context

United States

Against the backdrop of escalating cyber threats and regulatory scrutiny, legislative efforts such as the **United States' EO 14028** (previously mentioned) underscore the strategic importance of enhancing software supply chain security. Central to this executive mandate is the directive to delineate minimum elements for SBOMs, positioning SBOMs as linchpins in fostering transparency and accountability across the software development lifecycle. In a follow-up section, we will highlight the minimum elements for SBOMs as defined by EO 14028 to bring awareness and visibility.

KEY MILESTONES LEADING TO THE ANNOUNCEMENT OF EO 14028:

1. **February 2013:** Establishment of the National Institute of Standards and Technology (NIST) Cybersecurity Framework: The NIST released the first version of its Cybersecurity Framework, which provides guidelines and best practices for improving cybersecurity infrastructure in organizations.
2. **February 2013:** Presidential Policy Directive 21: Presidential Policy Directive 21 aimed to advance national efforts to strengthen and secure critical infrastructure by enhancing resilience and security.
3. **June 2015:** U.S. Office of Personnel Management (OPM) Data Breach: The U.S. OPM experienced a massive data breach, exposing the personal information of over 21 million federal employees. This breach highlighted significant cybersecurity vulnerabilities within government agencies.
4. **May 2017: EO 13800:** President Trump signed EO 13800, Strengthening the Cybersecurity of Federal Networks and Critical Infrastructure. This EO focused on enhancing the cybersecurity of federal networks and critical infrastructure.
5. **January 2019:** Cybersecurity Maturity Model Certification (CMMC) Initiative: The Department of Defense (DoD) introduced the CMMC to enhance the cybersecurity posture of its supply chain by requiring third-party certification of cybersecurity practices.
6. **December 2020:** SolarWinds Cybersecurity Breach: The SolarWinds attack, a significant supply chain cyber incident, compromised numerous federal agencies and private sector organizations, emphasizing the need for enhanced cybersecurity measures.
7. **January 2020:** National Defense Authorization Act (NDAA) for Fiscal Year 2021: The NDAA included provisions for improving cybersecurity, particularly within the DoD supply chain, and emphasized the importance of SBOMs.
8. **May 2021:** Colonial Pipeline Ransomware Attack: The Colonial Pipeline ransomware attack disrupted fuel supplies on the East Coast of the United States, further underscoring the vulnerability of critical infrastructure to cyber threats.
9. **May 12, 2021:** Announcement of EO 14028: President Biden signed EO 14028, Improving the Nation's Cybersecurity. This EO mandates federal agencies to adopt SBOMs, enhance software supply chain security, and implement comprehensive cybersecurity measures.

Against the backdrop of escalating cyber threats and regulatory scrutiny, legislative efforts such as the United States' EO 14028 (previously mentioned) underscore the strategic importance of enhancing software supply chain security.

European Union

Furthermore, across the pond, on March 12, 2024, the European Parliament approved the **E.U. Cyber Resilience Act** (CRA), with a large majority in favor of the legislation. The CRA aims to strengthen cybersecurity across the E.U. by setting comprehensive requirements for hardware and software products sold within the region. The Act emphasizes the need for manufacturers to ensure that their products are secure by design, regularly updated, and resilient against cyber threats. One of the key components of the Act is the introduction of implicit SBOM requirements, which requires manufacturers to produce them on the request of market surveillance authorities. This transparency helps identify potential vulnerabilities and ensures that all components, including third-party and open source software, meet security standards. By implementing these measures, the Act seeks to enhance the overall security of digital products and protect consumers and businesses from cyber risks.

Japan

The Japanese Ministry of Economy, Trade and Industry (**METI**) has emphasized the importance of SBOMs to manage the increasing threats to software security as supply chains grow more complex. METI has developed a **guide** for software suppliers that outlines the benefits of SBOM adoption and provides key implementation points. The guide encourages companies to adopt SBOMs to enhance software management, reduce response times to vulnerabilities, lower management costs, and improve development productivity and cybersecurity performance.

METI expects the widespread adoption of this guide to lead to better software management practices and to enhance cybersecurity across the industry.

Canada

Canada hasn't yet implemented specific legislation mandating the use of SBOMs. However, there are indications that the country's administration is aware of the importance of SBOMs for enhancing cybersecurity and managing software supply chains. The **Canadian Centre for Cyber Security** is actively promoting best practices for cybersecurity, including adopting SBOMs as a transparency and risk management tool. The emphasis is on encouraging organizations to integrate SBOMs into their software development and procurement processes to improve vulnerability management and overall cybersecurity posture. These efforts center on guidance and adopting best practices rather than legislative requirements at this stage.

China

China has been proactive in bolstering its legislative framework to support the implementation of SBOMs. As part of its 2024 legislative agenda, the Chinese government is placing a strong emphasis on cybersecurity and the management of software vulnerabilities, reflecting its broader goals of enhancing national security and technological advancement. The National People's Congress Standing Committee is addressing these issues through several legislative projects. These initiatives include efforts to improve cyber governance, manage software supply chain risks, and ensure the security of software components. Although specific legislation mandating SBOMs has not been explicitly mentioned, the legislative efforts encompass broader principles that align with the objectives of SBOMs, such as transparency and vulnerability management in software.

SBOM Minimum Elements

As previously mentioned, the United States' EO 14028 mandates several measures to enhance the cybersecurity of federal networks and the software supply chain. One of the key requirements is the establishment of minimum elements for an SBOM. These minimum elements are the following.

Identification of Software Components

SBOMs should identify each software component within a system or application, including open source software and proprietary/third-party commercial components.

Version Numbering

SBOMs should specify the version numbers of each software component to ensure accurate tracking and management of dependencies.

Provenance Information

SBOMs should contain metadata detailing the origin and ownership of each software component, including licensing information, copyright attributions, and contributors.

Dependency Relationships

SBOMs should delineate the hierarchical relationships between software components, including dependencies and interactions between various modules or libraries.

Security Attributes

SBOMs can be paired with information about known security vulnerabilities associated with each software component, enabling organizations to assess and mitigate security risks. Security information is typically pulled from a CVE (Common Vulnerabilities and Exposures), a database of publicly disclosed security vulnerabilities.

SBOM Format

SBOMs should adhere to a standardized format or schema to facilitate interoperability and exchange of information between different organizations and systems.

These SBOM minimum elements offer a foundational framework for promoting transparency, accountability, and security within software supply chains, helping organizations better understand and manage the risks associated with third-party software components.

System[1] Package Data Exchange Open Standard

The Linux Foundation's open source project **SPDX** has been at the forefront of standardizing SBOMs and promoting their adoption across the software industry since 2009, more than 11 years before EO 14028. The Linux Foundation has demonstrated incredible leadership and recognized the need for a common framework to facilitate the exchange of software component information in its effort to support open source license compliance practices. SPDX brings together organizations from various industries to develop standardized formats and tools for creating, sharing, and analyzing SBOMs. Through the typical open source collaborative effort and community engagement, SPDX has established a robust ecosystem of tools, resources, and best practices to support SBOM adoption and implementation. By providing guidance, developing specifications, and fostering collaboration, the Linux Foundation, through its SPDX efforts, has played a crucial role in advancing SBOM standardization, enabling organizations to enhance transparency, streamline license compliance, and improve security within their software supply chains.

Recommendations for Effective SBOM Implementation

To harness the full potential of SBOMs, organizations must embrace a strategic approach to implementation that is underpinned by automation and strategic utilization. In this section, we share some of the recommended practices we observe in the industry through our exchanges with leading organizations.

Establish Clear Policies and Procedures

Organizations should develop comprehensive policies and procedures outlining the generation, maintenance, and utilization of SBOMs throughout the software development lifecycle. Such policies and procedures (or processes) are typically part of a large effort defining the organization's internal compliance and security infrastructure.

Outline Clear Roles and Responsibilities

Organizations should ensure that all of their employees understand their roles and responsibilities regarding SBOMs and, at a broader level, regarding the practices to ensure compliance with applicable software licenses and software security guidelines. Organizations have used various methods to structure their teams responsible for fulfilling this function. Some companies have opted for a centralized team; others have opted for a cross-functional team consisting of a dedicated Open Source Compliance Officer who has access to various individuals and teams (DevOps, Engineering, Legal) that contribute to the compliance effort without being part of a centralized team. For a detailed discussion on setting up an open source license compliance infrastructure, we recommend "**Open Source License Compliance in the Enterprise**," 2nd edition, published by the Linux Foundation.

Automate SBOM Generation

Organizations should integrate SBOM generation as part of their continuous integration and delivery pipelines to ensure that SBOMs are automatically created with each software build. SBOM functionality is typically embedded as part of Software Composition Analysis (SCA) tools that support software

1 Prior to release 3.0, the System Package Data Exchange was called the Software Package Data Exchange.

development teams to ensure open source license compliance and improve the security of the code. Among their core functionalities, SCA tools perform automated scans on source codebases and help identify open source components and their license, flag any known vulnerabilities, and generate an SBOM of the scanned code.

Enrich with Metadata and Vulnerability Information

Organizations should enhance the production of their SBOMs with additional metadata about software components, such as version numbers, dependencies, licenses, and known security vulnerabilities associated with each software component.

Integrate into Compliance and Security Practices

It is recommended that organizations incorporate SBOMs into their open source license compliance and security practices to demonstrate robust governance and effective risk management.

Educate and Train Staff

Organizations should provide training and education to staff involved in license compliance and security practices. The Linux Foundation offers **free training courses** on license compliance and software security that are readily accessible.

Regularly Update and Review SBOMs

Organizations should maintain their SBOMs regularly as the use of software components changes or new vulnerabilities are discovered. Typically, organizations establish a process for monitoring changes and ensuring that SBOMs accurately reflect the current state of deployed or used software assets. Maintaining compliance and security is an ongoing effort that depends on discipline and a commitment to incorporate such activities into existing practices. One effective practice is to conduct recurring open source risk audits at predefined intervals for all source code, ensuring that the SBOMs are kept updated.

Collaborate and Get Involved in SPDX

Organizations should engage or participate at various levels with industry initiatives working on SBOM standardization efforts. Such engagements will help organizations stay informed about the latest developments in SBOM practices and tools and provide insights into best practices and emerging trends. The Linux Foundation's **SPDX** project is the leading project, and there are many benefits to such participation, including collaborating with peers to share best practices, co-developing supporting tools, evangelizing the open standard, and contributing to the advancement of SBOM adoption.

Monitor and Evaluate Effectiveness

Organizations should implement mechanisms to monitor the effectiveness of their SBOM implementation, such as tracking license compliance with SBOM policies, assessing the impact on security incident response times, and measuring improvements in vulnerability management processes. Organizations often rely on these insights to refine and improve their SBOM practices over time.

By following these recommendations, organizations can effectively implement SBOMs to enhance transparency, license compliance, and security across their software supply chains.

Conclusion

Integrating SBOMs into software development and governance processes is an important advancement in cybersecurity and license compliance. As discussed in this paper, SBOMs offer a structured and comprehensive view of software components, allowing organizations to manage and mitigate risks associated with software supply chains more effectively.

SBOMs contribute significantly to license compliance and cybersecurity, empowering organizations to track and manage software components meticulously, ensuring adherence to licensing terms, and proactively addressing security vulnerabilities. By embracing SBOMs, organizations can achieve greater transparency, enhance security, and ensure robust compliance across their software supply chains, thereby navigating the complexities of today's technological landscape with confidence and resilience.

Acknowledgments

The author wishes to express his gratitude to the Linux Foundation Research staff, Hilary Carter (SVP, Linux Foundation Research), and Kate Stewart (VP, Dependable Embedded Systems, Linux Foundation) for their reviews and valuable suggestions, which have significantly improved this paper.

About the Author

Dr. **Ibrahim Haddad** serves as the Executive Director of the LF AI & Data Foundation, fostering a vendor-neutral ecosystem for advancing the open source AI platform. In this role, he ensures a trusted and sustainable environment for developers to innovate, manage, and scale open source AI projects. Haddad's career includes research, technology, and management positions at prominent organizations such as Ericsson Research, the Open Source Development Labs, Motorola, Palm, Hewlett-Packard, and Samsung Research. He earned his Ph.D. in Computer Science with honors from Concordia University in Montréal, Canada.

THE LINUX FOUNDATION | Research

Founded in 2021, **Linux Foundation Research** explores the growing scale of open source collaboration, providing insight into emerging technology trends, best practices, and the global impact of open source projects. Through leveraging project databases and networks, and a commitment to best practices in quantitative and qualitative methodologies, Linux Foundation Research is creating the go-to library for open source insights for the benefit of organizations the world over.

x.com/linuxfoundation

facebook.com/TheLinuxFoundation

linkedin.com/company/the-linux-foundation

youtube.com/user/TheLinuxFoundation

github.com/LF-Engineering

To reference this work, please cite as follows: Ibrahim Haddad, "Strengthening License Compliance and Software Security with SBOM Adoption: A Definitive SBOM Guide for Enterprises," foreword by Melissa Evers, The Linux Foundation, August 2024.