System & Software Security
SBOM

Nuno Pereira
João Pereira
Davide Baggio

October 2024

We confirm that this report was fully produced by the team members **Davide Baggio, João Pereira and Nuno Pereira** and we are jointly responsible for all content presented in this work. All used sources were attributed properly.

Davide Baggio, João Pereira, Nuno Pereira

Contents

1 Introduction

Software systems, now more than ever, outsource parts of application logic to first- or third-party *dependencies* — pieces of code that are used in conjunction or as a part of the application's business logic but that are not part of the application itself. (Needs references)

Keeping track of dependencies can be an arduous task, which can be more easily managed using software tools and components such as *package managers* that keep track of an application's dependencies and their versions (Needs references) , like *npm* or *cargo* [8, 9].

SBOMs [1], proposed by the U.S. National Telecommunications and Information Administration, are a formal way of describing the software dependencies of an application and the relations between these dependencies. Other metadata can be attached to SBOM entries for a more comprehensive outline of the software components being dependended upon.

Currently, there are 3 SBOM standards used in practice: OWASP CycloneDX [6], Software Product Data eXchange (SPDX) [7] and Software Identification Tagging (SWID) [5]. However, little consensus exists between these 3 standards, making for one of the many challenges against general SBOM adoption [2].

2 Overview of reviewed papers

To assess the state-of-the-art with respect to SBOMs and their industry-wide use, we conducted a critical review of some examples from the literature, which are enumerated and described below:

Software Bills of Materials Are Required. Are We There Yet? [3]

An Empirical Study on Software Bill of Materials: Where We Stand and the Road Ahead [2]

On the Way to SBOMs: Investigating Design Issues and Solutions in Practice [4]

etc...

3 Metrics defined

4 Analysis and Comparison of reviewed papers

5 Conclusion

Supply Chain Security is an increasingly important factor of modern Software Development. Ensuring that companies adopt good practices in regards to the code they outsource is a pivotal step in ensuring the security requirements of software products. SBOMs (Needs references) are an example of such a practice that promotes good secure software engineering practices.

In this paper, we analyzed N examples of the state-of-the-art with regards to SBOMs and conducted a critical review of these papers according to our own metrics:

- Metric 1
- Metric 2
- ...

We found out that ...
Future work can expand on the methodology and results of this paper by ...

References

[1] Éamonn Ó Muirí. "Framing software component transparency: Establishing a common software bill of material (SBOM)". In: *NTIA, Nov* 12 (2019). (accessed: 15.10.2024).

[2] Boming Xia et al. "An Empirical Study on Software Bill of Materials: Where We Stand and the Road Ahead". In: *2023 IEEE/ACM 45th International Conference on Software Engineering (ICSE)*. 2023, pp. 2630–2642. DOI: 10.1109/ICSE48619.2023.00219.

[3] Nusrat Zahan et al. "Software Bills of Materials Are Required. Are We There Yet?" In: *IEEE Security & Privacy* 21.2 (2023), pp. 82–88. DOI: 10.1109/MSEC.2023.3237100.

[4] Tingting Bi et al. "On the Way to SBOMs: Investigating Design Issues and Solutions in Practice". In: *ACM Trans. Softw. Eng. Methodol.* 33.6 (June 2024). ISSN: 1049-331X. DOI: 10.1145/3654442. URL: https://doi.org/10.1145/3654442.

[5] National Institute of Standards and Technology (NIST). *Software Identification (SWID) Tagging — CSRC — CSRC — csrc.nist.gov*. https://csrc.nist.gov/projects/Software-Identification-SWID. [Accessed 15-10-2024]. (accessed: 15.10.2024).

[6] Open Worldwide Application Security Project (OWASP). *OWASP CycloneDX Software Bill of Materials (SBOM) Standard — cyclonedx.org*. https://cyclonedx.org/. [Accessed 15-10-2024]. (accessed: 15.10.2024).

[7] Linux Foundation. *SPDX Linux Foundation Projects Site — spdx.dev*. https://spdx.dev/. [Accessed 15-10-2024].

[8] NodeJS Team. *NPM: Node Package Manager*. URL: https://www.npmjs.com/. (accessed: 15.10.2024).

[9] Rust Team. *Cargo*. URL: https://doc.rust-lang.org/cargo/. (accessed: 15.10.2024).