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 increasingly 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* [1, 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?

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

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

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

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