Decentralized distribution of .buildinfo files

JOHAN MORITZ
jmoritz@kth.se

December 15, 2021

1 Thesis title

Decentralized distribution of .buildinfo files

2 Background

The discussion on mechanisms for verifying the lack of malicious code in binaries goes at least back to Ken Thompson's Turing award lecture [1]. Lately there have been attacks on popular packages within the Free Open Source Software ecosystem (FOSS) where built packages have been manipulated to include such malicious code. This raises the question of how much we can actually trust the usage of such dependencies. In an attempt to raise the level of trust and security in FOSS, the reproducible builds [2] projects was started within the Debian community. Its goal is to make software packages build bit-by-bit identically through deterministic builds, mitigating the risk that a certain build is tampered with. Any user of a package that can be reproduced can verify that it has indeed been built from its source code and have not been manipulated after the fact. The meta data of reproducible builds are stored in files with the .buildinfo format, and includes hashes of the build artifacts to allow for verification on the users side. .buildinfo files are currently distributed centrally which makes it a possible single point of failure in the system [3]. This project is about looking into the needs and requirements of moving the distribution of .buildinfo files from a centralized to a decentralized solution and thereby reducing the risks of such systems.

2.1 Subset

The project is carried out at Subset, an IT-security consulting company focusing on critical systems. They see reproducible builds as an important step for raising their trust of FOSS, which is critical for them to be able to use such software and technologies themselves without raising the concern of their customers.

3 Research question

- What are the requirements of a protocol/service for distributing .buildinfo files in a decentralized manner?
- What practical solutions are there to implement a decentralized distribution service for .buildinfo files?

4 Hypothesis

The current hypothesis is that the requirements of such a services should allow use of a consensus algorithm such as Practical Byzantine Fault Tolerance [4].

5 Research method

- Formalize the requirements of .buildinfo files.
- Compare the literature on decentralized consensus and knowledge with the formalized requirements to find possible solutions.
- Implement a prototype system for decentralized distribution of .buildinfo files and compare it to currently used centralized systems.

6 Background of the student

During my masters I have mainly specialized in the construction of programming languages, and taken courses related to formal languages and semantics, compiler construction and type systems. These have given me good practice in formalizing system requirements and properties which is the primary goal of this project, as well as given me tools for building complex systems. Relevant for this project is also a course on ethical hacking, rounding up my experience on computer security from the mandatory security course of the computer science masters program.

7 Supervisor at the company/external organization

Gabriel Bartolini is the primary supervisor of the project and will act as a spring board as well as technical expertise within cryptography and computer security.

8 Suggested examiner at KTH

No suggestion.

9 Suggested supervisor at KTH

Not sure but perhaps Thomas Durieux would be appropriate.

10 Resources

Subset have done a lot of work in IT-security in general and cryptography specifically, so for technical matters they have a lot of resources relevant to this project. They are also open to support the project financially where needed.

11 Eligibility

I will be eligible to start the project by the time this semester concludes. My bachelor was finished earlier this year, I have passed the mandatory research methodology course and, at the time of writing, I have 60 credits of advanced courses in my master. Mid January I finish the last course in my master subtrack on programming languages. As part of my masters program, I have taken courses on computer security as well as ethical hacking which I believe should be enough of a background to start this particular project.

12 Study Planning

During the spring I will finish the last semester of the prosamm (DD2300) course. Besides that I have 10 credits of elected (”valfria”) courses left, which will be concluded once I get them credited (”tillgodoräknade”) from courses outside of KTH.

References

[1] K. THOMPSON, “Reflections on trusting trust,” *Communications of the ACM*, vol. 27, no. 8, 1984.

[2] Reproducible builds - a set of software development practices that create an independently-verifiable path from source to binary code. [Online]. Available: https://reproducible-builds.org/

[3] C. Lamb and S. Zacchiroli, “Reproducible builds: Increasing the integrity of software supply chains,” *IEEE Software*, 2021.

[4] M. Castro, B. Liskov *et al.*, “Practical byzantine fault tolerance.”

A Optional appendix

The approximate word count is
698 (errors:5) words.

Acronyms

FOSS Free Open Source Software