



Wallet Binary Transparency

Advisor: **Edona Fasllija**




Motivation

Binary transparency is a security practice that ensures software binaries are verifiable and have not been tampered with. It aims to increase trust in software by providing mechanisms for third parties—like users, security researchers, or auditors—to verify the integrity and authenticity of a software binary. This approach makes it harder for attackers to distribute malicious code under the guise of a legitimate application.

Binary transparency is especially useful in maintaining trust in critical software systems, such as Digital Identity Wallets, where ensuring the integrity of code is paramount for security. It helps mitigate risks in the software supply chain, making it harder for attackers to slip malicious updates or alter software unnoticed.

When applying binary transparency to digital identity wallets, a public, append-only log of all software versions is maintained. These logs store cryptographic hashes of the binaries and metadata like version numbers, release dates, and publisher information. Logs are distributed, publicly auditable, and ensure historical accountability—any modifications to a binary can be detected by comparing it to the hash recorded in the log. Users can verify that updates come from the legitimate developer and match a specific, public version recorded in the transparency log.

Goals and Tasks

-  Get familiar with related literature and open-source implementations of binary transparency systems
-  Design and implement a Wallet Binary Transparency system
-  Write down your findings

Literature

- > [M. Al-Bassam and S. Meiklejohn](#)
Contour: A practical system for binary transparency
[ESORICS 2018](#)
- > Binary Transparency
<https://binary.transparency.dev/>

Courses & Deliverables

- Introduction to Scientific Working**
Short report on background
Short presentation
- Bachelor Project**
Project code and documentation
- Bachelor's Thesis**
Project code
Thesis
Final presentation

Recommended if you're studying

- CS
- ICE
- SEM

Prerequisites

- > Interest and some experience in tree-based data structures
- > Programming skills (Go)

Advisor Contact

edona.fasllija@tugraz.at