Security Analysis of Transparency Logs and Independent Witnessing in Sigstore

Stavros Kousidis

Master thesis · Published: 2026-10-15

Research question

Under which attacker and network assumptions can independent monitors and witnesses detect inconsistent or manipulated views of Sigstore transparency logs?

Description

Rekor makes software-signing events auditable through cryptographic transparency logs. An inclusion proof provides evidence that an entry is committed to a particular log state, but it does not alone guarantee that all clients observe the same history. Detecting equivocation requires additional monitoring, comparison or witnessing assumptions.

This thesis examines those assumptions and the limits of the guarantees they support. Research should distinguish log compromise, independent witness compromise, communication disruption and inconsistent checkpoints. Rekor already has a monitoring implementation, so implementing a basic monitor is not itself a research contribution.

Scope and evaluation

Develop a clear threat model and a small reproducible log-and-monitor experiment, using a controlled Rekor deployment or simplified Merkle-log prototype. Analyze and inject selected failures or manipulated views. Compare architectures with and without independent checkpoint verification or witnessing. Evaluate which violations are detectable, under what timing and availability assumptions, and where detection cannot be guaranteed.

Any conclusions must distinguish experimental observations from security claims requiring formal assumptions. Compare the relevant Rekor generations only where differences affect the chosen model.

Expected outcome

A structured trust analysis, a reproducible experiment and a precise discussion of detection guarantees, failure modes and remaining assumptions.

Starting points

- Sigstore: Rekor architecture and security considerations.
- Sigstore: Rekor Monitor.
- RFC 9162: Certificate Transparency Version 2.0.

Contact: stavros.kousidis@th-koeln.de
Code publication: open source by default; exceptions require prior agreement (policy).