Tamper-Proof Inverse Transparency Logs With Intel SGX

Bachelor's Thesis

Supervisor: Prof. Dr. Alexander Pretschner
Advisor: Valentin Zieglmeier
Email: {pretschn, zieglmev}@in.tum.de
Phone: +49 (89) 289 - 17834
Starting date: immediately

Informatik 4 – Lehrstuhl für Software und Systems Engineering
Prof. Dr. Alexander Pretschner
Fakultät für Informatik
Technische Universität München

Boltzmannstraße 3
85748 Garching bei München

https://www.in.tum.de/i04/

Context

Recent privacy legislation such as the GDPR and CCPA have lead to data owners becoming more self-sovereign in the handling of their data. Yet, these initiatives still do not lead to a true data sovereignty of individuals. Allowing access to data is still in many cases an “all-or-nothing” affair, with those trying to get access to the data having an information advantage. This is especially problematic in case of employee data, where an inherent power asymmetry and the necessity to use data for business processes reduces the choice of data owners to a minimum.

The concept of “Inverse Transparency” tries to improve upon the existing privacy protections of employees by giving data owners more sovereignty in how their data are used. Its core idea is to enable access to data on a more case-by-case basis, but to monitor all accesses and make those visible to data owners [1]. On the one hand, this can help to raise awareness of data usages and better protect the employee's personal data, on the other hand it may enable usages of data useful to teams and individuals alike.

An important prerequisite for Inverse Transparency to function is the ability to record usage logs in a tamper-proof way. Ideally, this can be done without a trusted third party, as in power-asymmetric environments (see above) the logs may be the target of manipulation. The recently developed technology Intel SGX [2] promises to enable this goal. With CPU-based attestation and sealing technology, the correctness of code executed on a remote server can be guaranteed. Furthermore, confidentiality guarantees can be given [3].

There are various ideas of how to utilize this technology within the context of private data analytics and secure logging. Karande et al. describe a secure log based on Intel SGX [4]. Priebe et al. go a similar route to describe EnclaveDB, a SGX-based database [5]. For both, the idea is to give guarantees about the correctness of entries stored in the log, as well as their confidentiality where necessary (although this is limited due to size constraints).

On the other hand, Birrell et al. [6], Schuster et al. [7], and Kelbert et al. [8] describe the idea of securely working with the data with SGX-protected applications, thereby ensuring the correct execution of analytics code. This starts a step earlier to give guarantees even before log entries have been created.

Goal

The goal of this thesis is to research sensible realizations for an SGX-based component for tamper-proof logging in the context of Inverse Transparency. The focus will lie on real-world implementation, considering potential issues of integration into the existing toolchain [1]. That includes the necessity of a discussion where the guarantees of SGX might need to be weakened to enable real world constraints or retain other properties, such as availability and logging performance.

Work is split into three phases: The theoretical, implementation, and evaluation phase.

Theoretical phase: Due to the high level of difficulty of understanding and working with Intel's SGX technology, the theoretical phase will be the most important for the work. A comprehensive understanding of the technology, its core advantages and limitations will be built through thorough literature review based on the given starting points (see below). This will serve as the foundation for an in-depth discussion of the applicability of SGX technology for Inverse Transparency logs for real-world scenarios. Finally, a theoretical model for an SGX-based log component for the toolchain will be designed. The goal of this is to guarantee the integrity and confidentiality of the stored data as much as possible, necessitating the least trust from stakeholders.

Implementation phase: Based on the theoretical work, the logging component will be implemented as a standalone tool. Core functionality to implement will be the storage of new log entries, making available stored entries for access to legitimate stakeholders (data owner and data consumer), and possibly handling access request policies, if sensible. A sensible

Tamper-Proof Inverse Transparency Logs With Intel SGX – Bachelor's Thesis approach to integrate this component into the toolchain could be to make available a REST API. Importantly, the limitations of SGX (storage space, speed) will have to be considered.

Evaluation phase: To evaluate the quality of the implementation, a theoretical analysis of the achieved security guarantees, as well as the limitations in the face of real-world requirements will be conducted. Additionally, the core performance metrics of logging speed and log size for an increasing number of log requests will be measured.

Work Plan

1. Research related literature.
2. Conceptualize a theoretical SGX-based logging component.
3. Implement the conceptualized component, integrating it into the toolchain.
4. Evaluate the solution by performing a security analysis, as well as by measuring core performance metrics.
5. Document the work in the thesis.

Deliverables

- Source code of the implementation.
- Created models and design documents.
- Thesis written in conformance with TUM guidelines.

References

[1] Zieglmeier, Valentin, and Alexander Pretschner. "Trustworthy Transparency by Design." arXiv preprint arXiv:2103.10769 (2021). Available: `https://arxiv.org/pdf/2103.10769`

[2] Anati, Ittai, et al. "Innovative technology for CPU based attestation and sealing." Proceedings of the 2nd international workshop on hardware and architectural support for security and privacy. Vol. 13. ACM New York, NY, USA, 2013. Available: `https://software.intel.com/sites/default/files/article/413939/hasp-2013-innovative-technology-for-attestation-and-sealing.pdf`

[3] Costan, Victor, and Srinivas Devadas. "Intel SGX Explained." IACR Cryptol. ePrint Arch. 2016.86 (2016): 1-118. Available: `http://css.csail.mit.edu/6.858/2020/readings/costan-sgx.pdf`

[4] Karande, Vishal, et al. "Sgx-log: Securing system logs with sgx." Proceedings of the 2017 ACM on Asia Conference on Computer and Communications Security. 2017. Available: `https://dl.acm.org/doi/pdf/10.1145/3052973.3053034`

[5] Priebe, Christian, Kapil Vaswani, and Manuel Costa. "EnclaveDB: A secure database using SGX." 2018 IEEE Symposium on Security and Privacy (SP). IEEE, 2018. Available: `https://ieeexplore.ieee.org/iel7/8418581/8418583/08418608.pdf`

[6] Birrell, Eleanor, et al. "SGX enforcement of use-based privacy." Proceedings of the 2018 Workshop on Privacy in the Electronic Society. 2018. Available: `https://dl.acm.org/doi/pdf/10.1145/3267323.3268954`

[7] Schuster, Felix, et al. "VC3: Trustworthy data analytics in the cloud using SGX." 2015 IEEE Symposium on Security and Privacy. IEEE, 2015. Available: `https://ieeexplore.ieee.org/iel7/7160813/7163005/07163017.pdf`

[8] Kelbert, Florian, et al. "Securecloud: Secure big data processing in untrusted clouds." Design, Automation & Test in Europe Conference & Exhibition (DATE), 2017. IEEE, 2017. Available: `https://ieeexplore.ieee.org/iel7/7919927/7926947/07926999.pdf`