Chair of Network Architectures and Services
School of Computation, Information, and Technology
Technical University of Munich

Evaluation of zk-SNARKs inside TEEs

Motivation

Zero-knowledge Succinct Non-interactive Arguments of Knowledge (zk-SNARK): Zero-knowledge proofs with verification in polynomial time [6, 1]

- zk-SNARK proof generation: very resource-demanding [5]
- Burden on the prover → prover is a bottleneck
- Proof generation often not possible on constrained hardware [10]

Possible solution: Outsource zk-SNARK proof generation to the Cloud
Issue: secret witness has to be processed in the Cloud and may be visible to the Cloud provider [2]
Solution: Outsource zk-SNARK proof generation to Trusted Execution Environment (TEE) running in the Cloud

Use Cases

For the thorough evaluation of the combination of zk-SNARKs and TEEs, we have chosen three different use cases:

- Use case with (almost) no time constraints: Electronic voting using zk-SNARKs to prove ballot validity [4]
- Use case with time constraints: Set inclusion proof for authentication [9]
- Use case with various implementations available for in-depth comparisons between frameworks: SHA256 [3]

Experiment Design

Benchmarks have been implemented in **EnGINE**, a framework for the detailed assessment of distributed systems [7, 8]

- Captured metrics: CPU, time, and memory consumption
- VM-based TEE executes Kata container
- Comparison with benchmarks captured in plain Kata container
- Setup: AMD SEV-SNP TEE (32 vCPUs, 128 GB RAM)

Table 1: Framework implementation per use case.

Use case	Circom	Halo2	Gnark	Bellman
Voting	✓	✗	✗	✗
Set inclusion	✓	✗	✗	✗
SHA256	✓	✓	✓	✓

Table 2: Supported backends per framework.

Framework	Groth16	Plonk
Circom	✓	✓
Halo2	✗	✓
Gnark	✓	✓
Bellman	✓	✗

Results & Future Work

Results:

- Time consumption higher in TEE for all use cases and frameworks
- Difference in time consumption dependent on framework/prover
- Comparable memory consumption in TEE for most frameworks
- Mixed results for CPU consumption for many frameworks
- Snarkjs outperformed by all other frameworks

TEEs enable zk-SNARK proof generation in a Cloud environment but come with a time penalty.

Future Work:

- Benchmarks of AMD SEV-SNP and Intel TDX for comparison
- Replace Kata container with QEMU

Evaluation

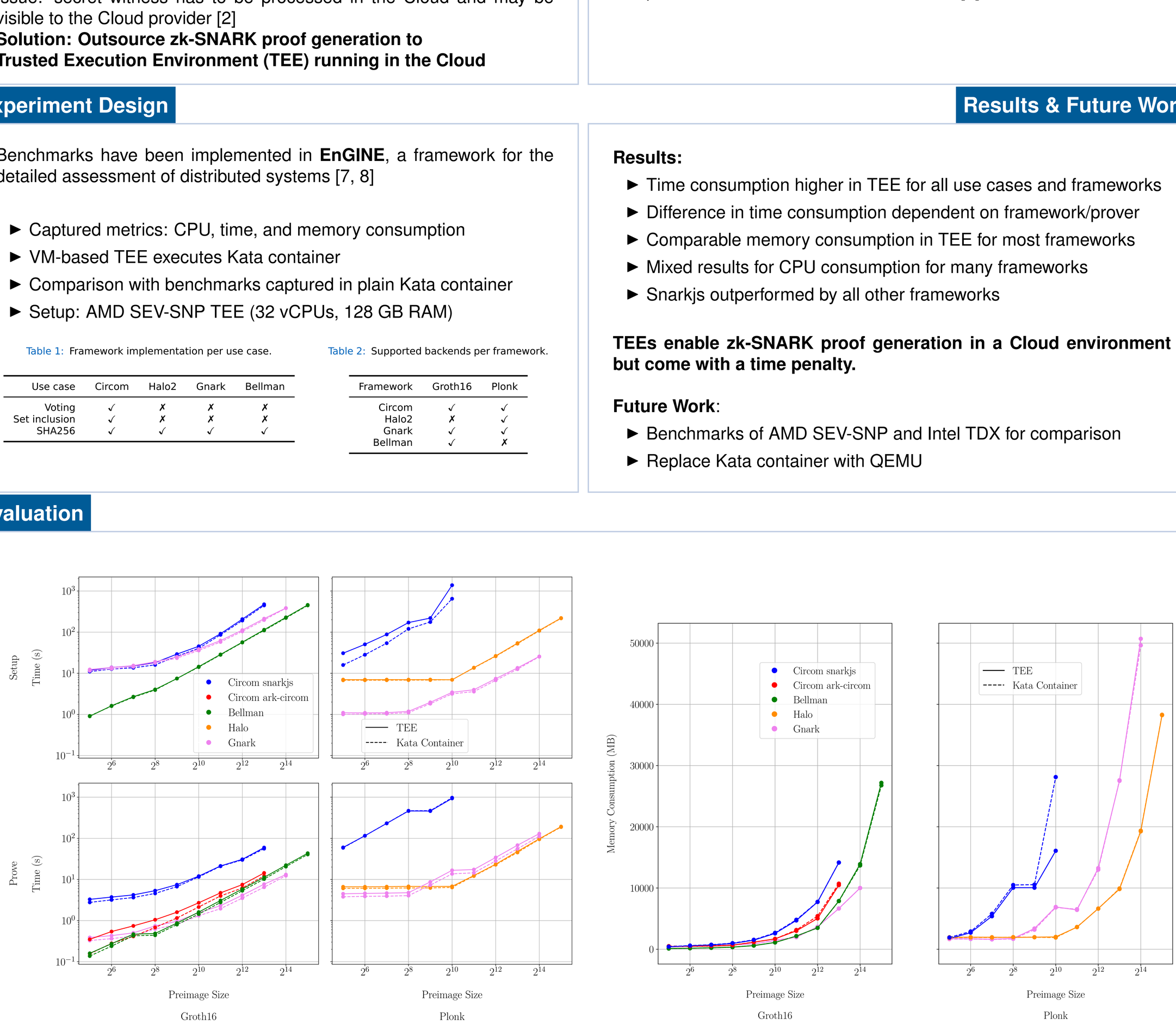

Figure 1: Mean time consumption (left) and mean memory consumption (right) of the SHA256 use case inside a TEE and a Kata container.

[1] O. Amine, K. Baghery, Z. Pindado, and C. Ràfols. Simulation extractable versions of Groth's zk-SNARK revisited. *International Journal of Information Security*, 23(1):431–445, 2024. Publisher: Springer.
[2] R. Buhren, C. Werling, and J.-P. Seifert. Insecure until proven updated: analyzing AMD SEV's remote attestation. In *Proceedings of the 2019 ACM SIGSAC Conference on Computer and Communications Security*, pages 1087–1099, 2019.
[3] J. Ernstberger, S. Chaliasos, G. Kadianakis, S. Steinhorst, P. Jovanovic, A. Gervais, B. Livshits, and M. Orrù. zk-Bench: A Toolset for Comparative Evaluation and Performance Benchmarking of SNARKs, 2024.
[4] N. Huber, R. Küsters, J. Liedtke, and D. Rausch. ZK-SNARKs for Ballot Validity: A Feasibility Study. In D. Duenas-Cid, P. Roenne, M. Volkamer, J. Budurushi, M. Blom, A. Rodríguez-Pérez, I. Spycher-Krivonosova, J. Castellà Roca, and J. Barrat Esteve, editors, *Electronic Voting*, volume 15014, pages 107–123. Springer Nature Switzerland, Cham, 2025. Series Title: Lecture Notes in Computer Science.
[5] X. Liu, Z. Zhou, Y. Wang, B. Zhang, and X. Yang. Scalable Collaborative zk-SNARK: Fully Distributed Proof Generation and Malicious Security, 2024. Published: Cryptology ePrint Archive, Paper 2024/143.
[6] A. Nitulescu. zk-SNARKs: A gentle introduction. *Ecole Normale Superieure*, 2020.
[7] F. Rezabek*, M. Bosk*, T. Paul, K. Holzinger, S. Gallenmüller, A. Gonzalez, A. Kane, F. Fons, Z. Haigang, G. Carle, and J. Ott. EnGINE: Flexible Research Infrastructure for Reliable and Scalable Time Sensitive Networks. *Journal of Network and Systems Management*, 30(4):74, Sept. 2022.
[8] F. Rezabek, K. Glas, R. Von Seck, A. Aroua, T. Leonhardt, and G. Carle. Multilayer environment and toolchain for holistic network design and analysis. Nov. 2023.
[9] D. Soler, C. Dafonte, M. Fernández-Veiga, A. F. Vilas, and F. J. Nóvoa. A privacy-preserving key transmission protocol to distribute qrng keys using zk-snarks. *Computer Networks*, 242:110259, 2024.
[10] H. Wu, W. Zheng, A. Chiesa, R. A. Popa, and I. Stoica. DIZK: A Distributed Zero Knowledge Proof System. In *27th USENIX Security Symposium (USENIX Security 18)*, pages 675–692, Baltimore, MD, Aug. 2018. USENIX Association.

Veronika Bauer, Kilian Glas, Filip Rezabek, Xinxin Fan, and Georg Carle { bav | glas | rezabek | carle }@net.in.tum.de