Verification of Side-Channel Leakage Contracts on Binary Code

Yanis Sellami
CEA, List

Sébastien Bardin
CEA, List

yanis.sellami@cea.fr, sebastien.bardin@cea.fr

Automated program analysis is a research topic that tries to design tools and analyses to detect bugs in programs. Existing techniques typically target memory access vulnerabilities (*e.g.* buffer overflows) and numerical vulnerabilities (*e.g.* zero division) that can corrupt program data or crash the execution. Yet, even when one can prove that a program contain no such bug, an attacker can still exploit some physical properties of program execution (such as execution time, cache values, ...) to understand its behavior and extract secret data. Such attacks are called side-channel attacks.

The binary code analysis platform BINSEC [1] developped at CEA List possesses a relational analysis [2] that can detect some side-channel vulnerabilities, typically when the code is not constant-time. Unfortunately, the constant-time programming paradigm is often insufficient to model correctly a increasing set of possible side-channel attacks. On one hand, it may raise alerts that will not lead to any pair or distinguishable observation. On the other hand, it fails to capture finer leakages that may be present on the hardware.

To tackle this problem, we will consider the concept of hardware-software leakage constracts [3, 4]. Such contracts provide, for the software, an insight on the actual behavior of the hardware. This insight can be exploited to perform a more detailed analysis of the vulnerability of a given software when running on the corresponding hardware. The goal of this project is to extend the capabilities of BINSEC to verify binary code against such contracts.

Logistics. The internship will be homed at Nano-INNOV in Saclay, co-supervised by Yanis Sellami and Sébastien Bardin.

Candidate. To candidate or obtain additional information, please contact the supervisors by email.

References

[1] "Plateforme binsec." `https://binsec.github.io/`.

[2] L.-A. Daniel, S. Bardin, and T. Rezk, "Binsec/rel: Symbolic binary analyzer for security with applications to constant-time and secret-erasure," *ACM Trans. Priv. Secur.*, vol. 26, apr 2023.

[3] G. Mohr, M. Guarnieri, and J. Reineke, “Synthesizing hardware-software leakage contracts for risc-v open-source processors,” in *2024 Design, Automation Test in Europe Conference Exhibition (DATE)*, pp. 1–6, 2024.

[4] H. Winderix, M. Bognár, J. Noorman, L.-A. Daniel, and F. Piessens, “Architectural mimicry: Innovative instructions to efficiently address control-flow leakage in data-oblivious programs,” 2024-01-01.

[5] Y. Sellami, G. Girol, F. Recoules, D. Couroussé, and S. Bardin, “Inference of robust reachability constraints,” *Proc. ACM Program. Lang.*, vol. 8, Jan. 2024.