Extending a Binary Code Vulnerability Analysis for Side-Channel Attacks

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 possesses a relational analysis [2] that can detect some side-channel vulnerabilities, typically when the code is not constant-time. This analysis can be taken as a first steop to build more complex procedures able to detect more diverse vulnerabilities.

The goal of this project is to extend this relational analysis to target new vulnerabilities. We are particularly interested in power side-channel attacks [3, 4], port contention attacks and ciphertext attacks [5]. We will expect a state of the art on these attacks, the related vulnerabilities, as well as an understanding of the BINSEC platform. The student shall then suggest an extension of the BINSEC relational analysis for a new type of vulnerability, with an implementation if possible.

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] R. Tsoupidi, R. Lozano, E. Troubitsyna, and P. Papadimitratos, "Securing optimized code against power side channels," in *2023 IEEE 36th Computer Security Foundations Symposium (CSF)*, (Los Alamitos, CA, USA), pp. 340–355, IEEE Computer Society, jul 2023.

[4] Y. Wang, R. Paccagnella, E. T. He, H. Shacham, C. W. Fletcher, and D. Kohlbrenner, "Hertzbleed: Turning power Side-Channel attacks into remote timing attacks on x86," in *31st USENIX Security Symposium (USENIX Security 22)*, (Boston, MA), pp. 679–697, USENIX Association, Aug. 2022.

[5] S. Deng, M. Li, Y. Tang, S. Wang, S. Yan, and Y. Zhang, "CipherH: Automated detection of ciphertext side-channel vulnerabilities in cryptographic implementations," in *32nd USENIX Security Symposium (USENIX Security 23)*, (Anaheim, CA), pp. 6843–6860, USENIX Association, Aug. 2023.