Has your Microprocessor been attacked? Does it include a malicious component? Machine Learning Can Provide the Answers

Alessandro Palumbo
alessandro.palumbo@inria.fr

Context

Software-exploitable Hardware Trojan Horses (HTHs) can be inserted into microprocessors, allowing attackers to run their own software or gain unauthorized privileges. On the other hand, it has been demonstrated that by observing some features of the Microprocessor (apparently not related to its program run), it is possible to gain information about Microprocessor running operations.

HTHs consist of malicious, undesired circuit modifications. They have always been considered more an academic issue because of the difficulty of insertion in real-world systems, leading to reduced advantages for the attacker. Recently, it has been demonstrated that complex *software-exploitable HTHs* can be inserted in real-world commercial microprocessors. Such HTHs allow the attacker to execute his/her malicious software, modify the running software, or acquire root privileges [1]–[3]. In 2018, the *Rosenbridge* backdoor has been found in a commercial Via Technologies C3 processor [4, 5]. A specific sequence of instructions allowed the attacker to activate the Rosenbridge backdoor and enter supervisor mode[1].

In [7], [8], the authors demonstrate that detecting HTHs implemented in RISC-V ISA-based Microprocessors has been possible by looking at some features. It turned out that the detection was so accurate when looking at features that are close to the circuit (i.e., power consumption and temperature traces, critical path); on the other hand, looking at software features (i.e., performance counters), the detection was not so accurate. The question is: Could the accuracy of the detection be higher if we look for many other software features (in [7], [8], the high-level features were not so many)? The answer seems to be positive: some preliminary evaluation of this scenario has been done previously, but it is needed to investigate more in detail. The goal of this project would be to go ahead and deeper into the yet-done estimations

[1] Via Technologies officially commented that this behavior was due to an undocumented feature meant for debugging

The project

The challenge of this project is to emulate the insertion of different types of HTHs on a RISC-V-based CPU on the gem5 tool simulator [9]. It is a highly configurable and modular simulator for computer processor architectures. Such a simulator provides a huge list of performance counters and Microprocessor features: running programs on the attacked and safe CPUs via Machine Learning computations, looking only at the features, would we be able to detect the attacks? Many HTH models are reported in the TrustHub repository [6]. For this project, we can refer to one of those.

In this topic, we run some evaluations yet, and the response seems to be positive. We have some preliminary scripts written and the simulator up.

Here are the main steps of the project:

1. Get familiar with the gem5 tool for CPU design and simulations.
    a. Implement the attack-free RISC-V ISA and the attacked one.
    b. Run benchmarks with the goal of observing microprocessor feature values (given by the tool).
2. Get familiar with the Machine Learning algorithms and computations.
    a. Implement Python scripts based on Machine Learning models looking for the dumped features with the goal of detecting attacks and safe runs.
        i. Looking for the best model for accuracy detection (i.e., Random Forest, Support Vector Machine, Isolation Forest, and so on).

Required skills or interests

- Software & Application Design Languages (C, C++, Python).
- CPU and microprocessor architectures.

Institute

The internship will take place at CentraleSupélec in Rennes, France, in the SUSHI Inria team[2]. This team is part of the IRISA laboratory[3].

Practical aspects

The intern will receive a ”gratification” of about 600€ per month. Housing options may be available on campus or close to it. This 5/6 months internship is research-oriented and lays

[2] https://team.inria.fr/sushi/

[3] https://www.irisa.fr/en

the groundwork for potential doctoral studies. This opportunity is ideally suited for students interested in pursuing a PhD, given the advanced nature of the work and its potential applications in academic and research settings.

References

[1] Y. Jin, M. Maniatakos, and Y. Makris, “Exposing vulnerabilities of untrusted computing platforms,” in *Proc. Int. Conf. Computer Design*, pp. 131–134, 2012.

[2] N. G. Tsoutsos and M. Maniatakos, “Fabrication attacks: Zero-overhead malicious modifications enabling modern microprocessor privilege escalation,” *IEEE Trans. Emerging Topics in Computing*, vol. 2, no. 1, pp. 81–93, 2014.

[3] X. Wang, T. Mal-Sarkar, A. Krishna, S. Narasimhan, and S. Bhunia, “Software exploitable hardware trojans in embedded processor,” in *2012 IEEE International Symposium on Defect and Fault Tolerance in VLSI and Nanotechnology Systems (DFT)*, pp. 55–58, IEEE, 2012.

[4] C.Domas,“Hardwarebackdoorsinx86cpus.”https://i.blackhat.com/us- 18/Thu-August-9/us-18-Domas-God-Mode-Unlocked-Hardware- Backdoors-In-x86-CPUs-wp.pdf, 2018.

[5] project:rosenbridge, last access Feb. 2022. URL: https://github. com/xoreaxeaxeax/rosenbridge.

[6] B. Shakya, T. He, H. Salmani, D. Forte, S. Bhunia, M. Tehranipoor, Benchmarking of hardware trojans and maliciously affected circuits, Journal of Hardware and Systems Security 1 (2017) 85–102.

[7] A. Palumbo, L. Cassano, B. Luzzi, J. A. Hernández, P. Reviriego, G. Bianchi, and M. Ottavi. “Is your FPGA bitstream HardwareTrojan-free? Machine learning can provide an answer”. In: Journal of Systems Architecture (2022), p. 102543.

[8] Ribes, S., Malatesta, F., Garzo, G., & Palumbo, A. (2024). *Machine learning-based classification of hardware Trojans in FPGAs implementing RISC-V cores*. In Proceedings of the 10th International Conference on Information Systems Security and Privacy (ICISSP) (pp. 717-724).

[9] Binkert, N., Beckmann, B., Black, G., Reinhardt, S. K., Saidi, A., Basu, A., Hestness, J., Hower, D., Krishna, T., Sardashti, S., Schuette, B., Sen, R., Sewell, K., Shoaib, M., Sinkar, S., Turner, A., Wenisch, T. F., & Hill, M. D. (2011). *The gem5 simulator*. ACM SIGARCH Computer Architecture News, 39(2), 1-7.