Thesis subject

Cross-Layer Fault Analysis for Microprocessor Architectures (CLAM)

LCIS, TIMA, Verimag

Securing components for the IoT market, as well as critical cyberphysical infrastructures, requires analyzing their vulnerabilities and defining hardware and software countermeasures at the fair cost. The increasing complexity of the processors and the applications they run means that the software fault models (such as instruction skips, or instruction replacement) usually used to analyze the vulnerability of their code are no longer sufficient to express the diversity of faulty behaviors in modern architectures [1]. Indeed, architecture designers have progressively added many complex hardware blocks (such as pipeline, cache memory, branch prediction, or speculative execution) to processors in order to optimize program execution. At the same time, fault injection techniques are constantly progressing (e.g., localized ElectroMagnetic or laser attacks, glitch injection attacks) allowing today higher-order injections (both multi-temporal and multi-spatial).
Faced with these attacks, designers must implement countermeasures to detect or mask the effects of injected faults. These countermeasures can be implemented at the hardware level (e.g., duplication of elements, error correcting codes, isolation mechanisms) or at the software level (e.g., duplication of instructions or algorithms, insertion of security tests, signature verification). The design and validation of these countermeasures requires a realistic representation of the effects of faulty attacks (fault model) at the hardware and/or software level. The difficulties encountered are then: (1) the relevance of the fault models: do these fault models correctly represent the effects that an attacker can generate on a target processor? (2) the adequacy of countermeasures: do the countermeasures effectively protect the system, are they oversized and therefore too costly? (3) the link between software and hardware countermeasures: how to combine software and hardware countermeasures efficiently and at the fair cost, how to link the different levels for fault modeling?

The CLAM project and the thesis that we propose aim to answer these questions by associating 3 French laboratories with complementary specialties and experiences: the LCIS (in Valence), specialized in hardware fault simulation at RTL level and the development of fault injection tools (clock and voltage glitch generators, EM attack); the TIMA Laboratory (in Grenoble), with its expertise in fault evaluation, modeling and emulation; the Verimag Laboratory (in Grenoble) specialized in software vulnerability analysis based on static analysis tools. The work in the field of fault injection of each of these laboratories is recognized nationally and internationally.

Figure 1 shows the different levels of analysis (target hardware processor, RTL description and executed code) and the tools we propose to combine to create new realistic software fault models.

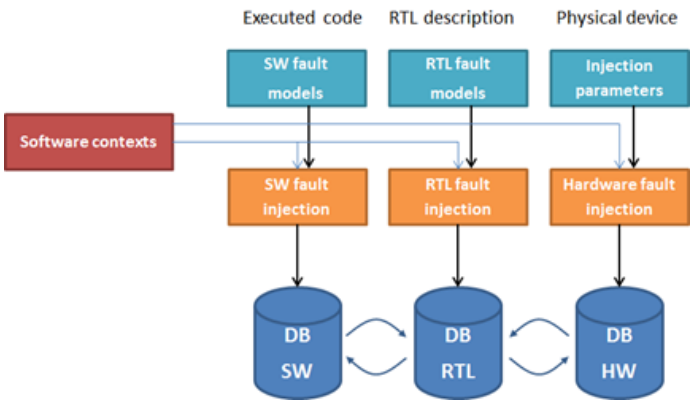

Figure 1: Multi-level fault injection platform: on the physical component, on the RTL description and in the executed code

Attacks on hardware targets (ARM processors) will allow analyzing the realism of RTL faults and these RTL faults to analyze the realism of software faults. For example, software faults whose effects are never observed will be eliminated and, conversely, new software faults will be created. Thanks to the new software fault model inferred for each processor and type of attack, the goal is then to obtain, thanks to static analysis tools (e.g.,with Lazart from Verimag), a code vulnerability analysis that is both more realistic and faster than what is currently done. This analysis will then allow us to verify and improve the effectiveness of the proposed countermeasures at both software and hardware levels.

PhD student profile: Master in Embedded Systems, Master in Computer Science, Master in Microelectronics

Skills: Computer Architecture, Prototyping and Simulation of Digital Systems, Compilation

Location: Grenoble INP LCIS, Valence

Contacts: vincent.beroulle@lcis.grenoble-inp.fr; paolo.maistri@univ-grenoble-alpes.fr

To apply for this offer, please send to the persons indicated above: your CV, a specific letter of motivation (in French or English), a master's transcript (M1 and M2), and letters of recommendation.

References

[1] J. Laurent, V. Beroulle, C. Deleuze, F. Pebay-Peyroula, A. Papadimitriou, "Improving Software Fault Models and Countermeasures in a RISC-V Process".