Lazart: a symbolic approach for evaluating the robustness of secured codes against control flow fault injections

Marie-Laure Potet, Laurent Mounier, Maxime Puys and Louis Dureuil

Verimag, University of Grenoble Alpes
firstname.name@imag.fr

Article presented at ICST 2014[1]. Smart-cards have nowadays become a ubiquitous information vector and often contain critical data for instance related to banking, identity or medical. Thus they are submitted to drastic secure requirements and certification process in order to resist to high level attack potential (such as multiple attackers with a high level of expertise, using sophisticated equipments, etc.) [CCD09].

We focus on perturbation attacks where an external event (such as a laser beam, a voltage glitch, etc.) modifies the execution. We propose a white-box approach named Lazart, allowing to mesure the robustness of a code against faults attacks targeting control flow. Based on symbolic execution, we are able to either produce attacks or prove their absence. Moreover, thanks to the static analysis, we are able to consider higher order attacks (multiple faults injected in a single run) while mastering the combinatorial explosion inherent.

The Lazart approach relies on a tool-chain based on the LLVM framework [LA04]. Starting for an attack objective (namely a basic block to reach or to avoid), the first step computes the potential injections points. These are embedded in a single higher order mutant which contains all possible fault injections, each being triggerable by a boolean variable. Finally, using the concolic test case generator Klee [CDE08], we are able to find attacks of order less than a given n. Such verdict is obtained by covering all the combinations of faults by playing with the value of their triger variables.

We have tested our approach on two implementations of PIN verification, one of them including robust counter-measures. We also provide a classification criterion in order to compare attacks.

References

[CCD09] Application of attack potential to smartcards. Technical Report CCDB-2009-03-001, Commun Criteria, 2009.

[CDE08] C. Cadar, D. Dunbar, and D. Engler. Klee: Unassisted and automatic generation of high-coverage tests for complex systems programs. In *OSDI*, 2008.

[LA04] C. Lattner and V. Adve. LLVM: A Compilation Framework for Lifelong Program Analysis & Transformation. In *CGO'04*, Palo Alto, California, 2004.

[1] IEEE International Conference on Software Testing, Verification, and Validation, Cleveland, Ohio, USA, March 31- April 4, 2014