Hardware/Software Runtime for GPSA Protection in RISC-V Embedded Cores

Louis SAVARY Simon ROKICKI Steven DERRIEN
Inria, IRISA, Université de Rennes, ENS Rennes
louis.savary@inria.fr

How to protect embedded cores from fault injection whithout modifying the binaries?

Summary

Context: embedded systems are energy constrained and subject to fault attacks.

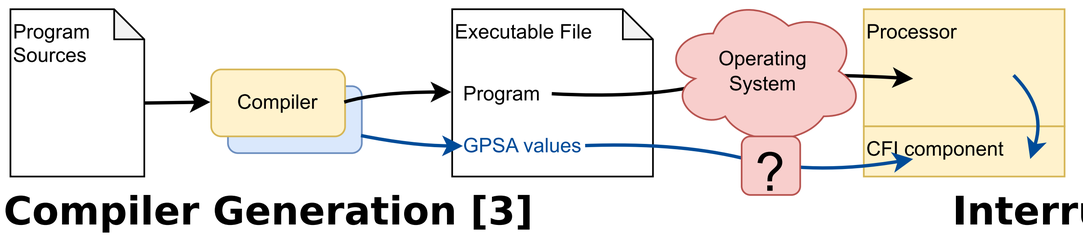
Problem: how to protect the processor against fault attacks without having to modify the binaries ?

Our approach: use known techniques to ensure micro-architectural level integrity properties and adapt them with HW/SW runtime for GPSA mechanisms.

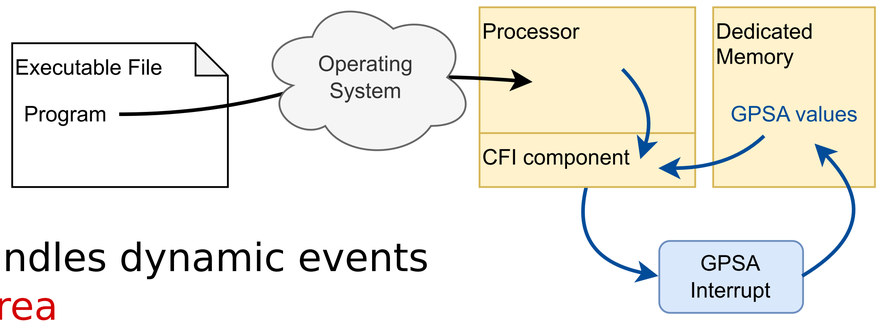
Background

Fault Attacks [1]
- cause an error in a software from hardware
- laser, EM pulse, clock or power glitch
- can impact control flow

GPSA and CSM[1] [2]
- detect control flow errors
- rely on a signature system, encoding each executed instruction



Compiler Generation [3]
- common implementation
- cannot handle dymanic events
- requires a dedicated compiler toolchain

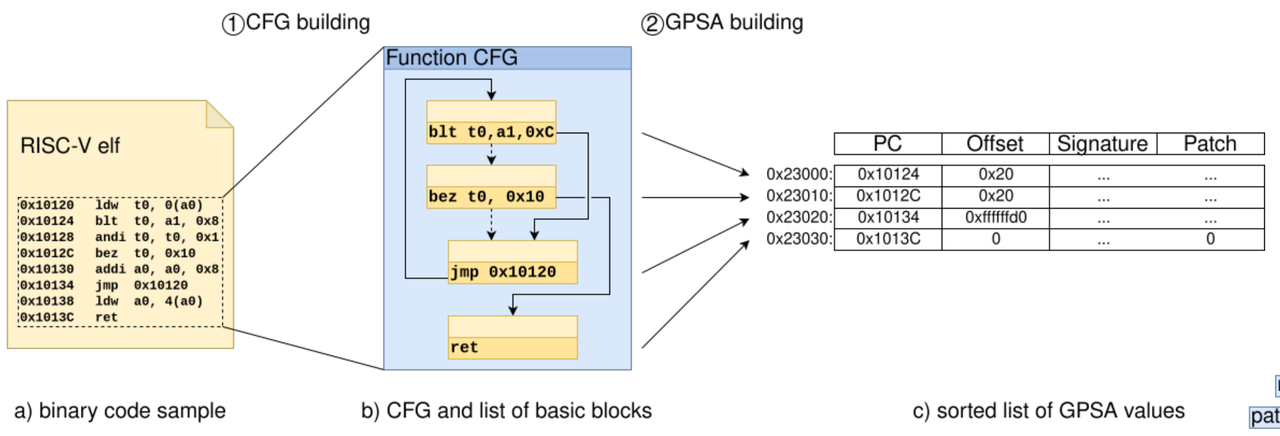


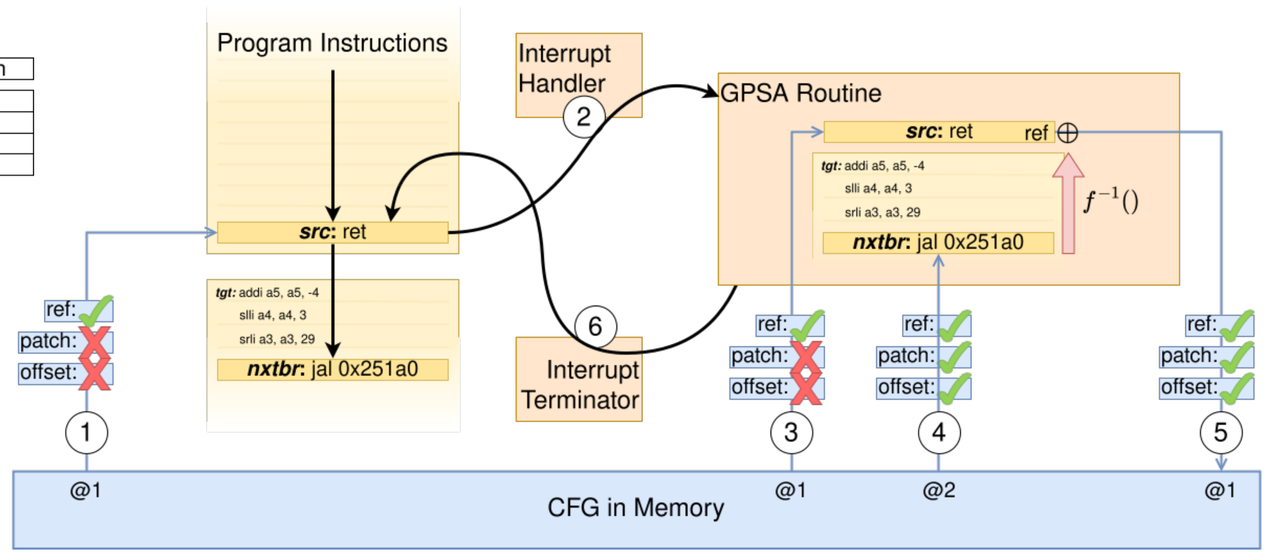
Interrupt Generation [4]
- protects any program and handles dynamic events
- high cost, in both time and area
- does not provide code protection

1 Generalized Path Signature Analysis and Continuous Singature Monitoring

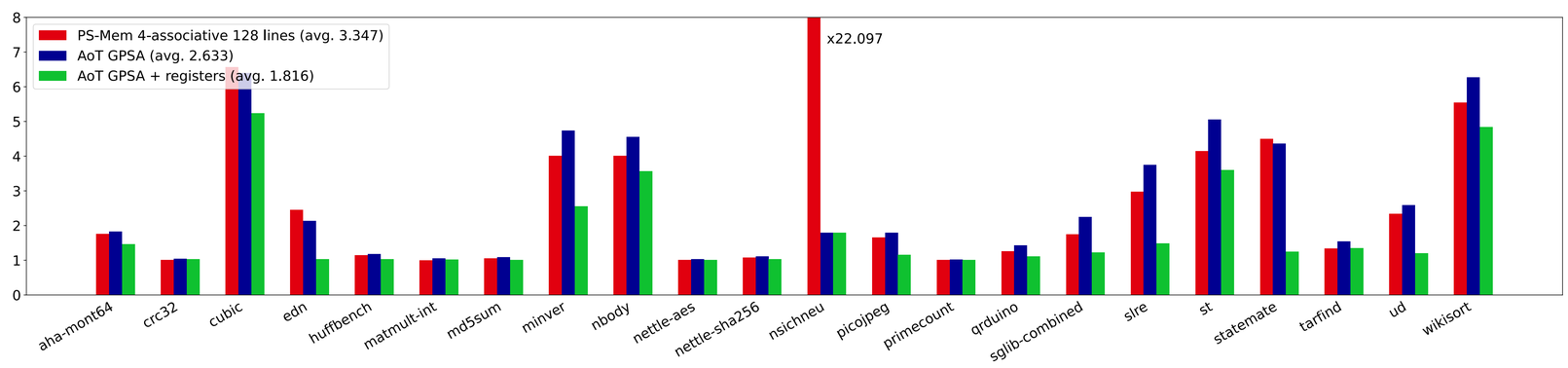
Our approach

Use a static analysis for generation at deployment time, completed with a runtime generator
- can run any RISC-V executable off-the-shelf
- handle indirect jumps and context switches through Dynamic GPSA





Results



- implemented on Comet RISC-V core [5]
- evaluated on embench-iot [6]
- average slowdown of x1.82
- results show ~30% area overhead

References & Acknowledgements

The ARSENE project was funded by the “France 2030” government investment plan managed by the French National Research Agency, under the reference " ANR-22-PECY-0004

[1] J. Laurent, et al. "Fault Injection on Hidden Registers in a RISC-V Rocket Processor and Software Countermeasures." DATE 2019
[2] M. Werner, et al."Protecting the Control Flow of Embedded Processors against Fault Attacks." CARDIS 2015
[3] T. Chamelot et al. "SCI-FI: Control Signal, Code, and Control Flow Integrity against Fault Injection Attacks," DATE 2022
[4] L. Savary, et al. "Hardware/Software Runtime for GPSA Protection in RISC-V Embedded Cores." DATE 2025
[5] S. Rokicki, etal. "What You Simulate Is What You Synthesize: Designing a Processor Core from C++ Specifications." ICCAD 2019
[6] David Patterson and Jeremy Bennett and Palmer Dabbelt, Cesare Garlati and G. S. Madhusudan and Trevor Mudge. Embench: Open Benchmarks for Embedded Platforms. https://github.com/embench/embench-iot