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.

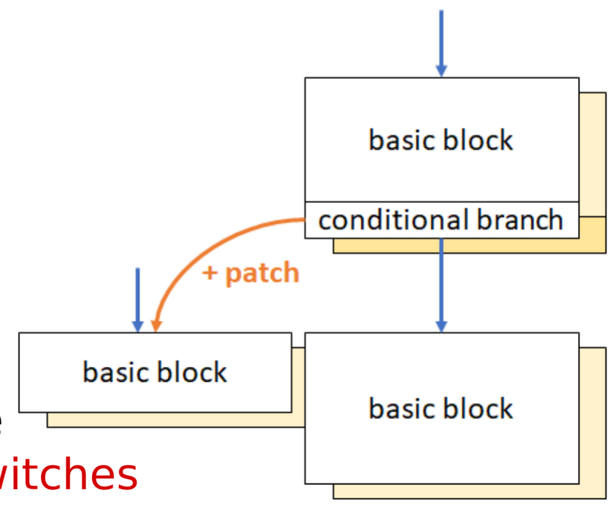
Problem: how to protect the processor against fault attacks without having to modify the binaries ?

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

Background

Fault Attacks [1]

- used to cause a wrong behaviour in a software from faults in the hardware.
- can involve several techniques as laser, EM pulse, clock or power glitch
- impact control flow to skip or re-execute instructions or change branches
- attacker model: at any cycle, the fetched instruction can be randomized



GPSA and CSM[1] [2]

- detect control flow errors
- rely on a signature system, encoding each executed instruction

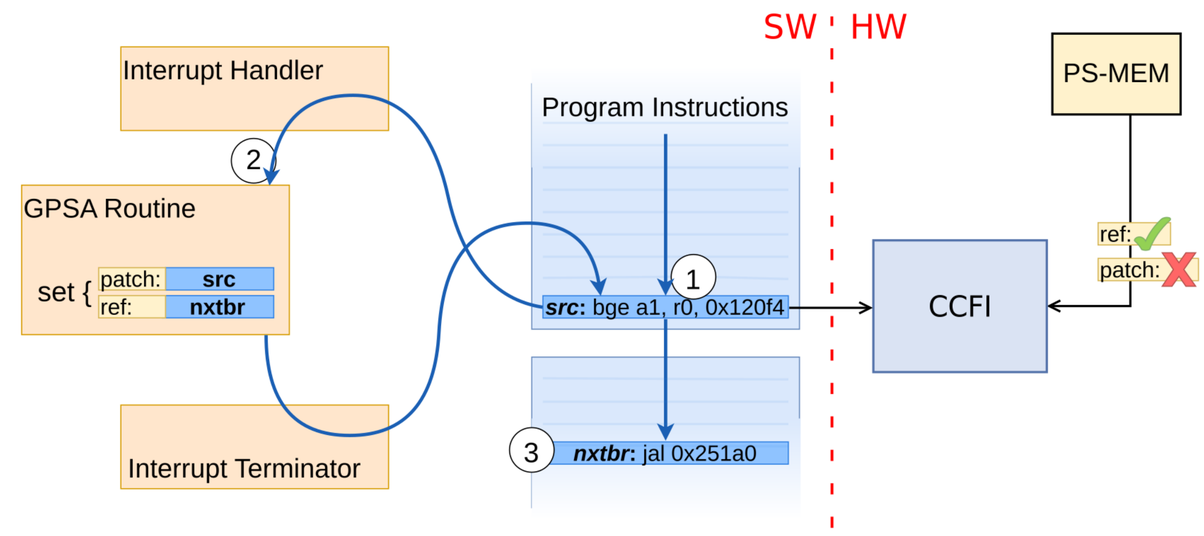
SCI-FI [3]

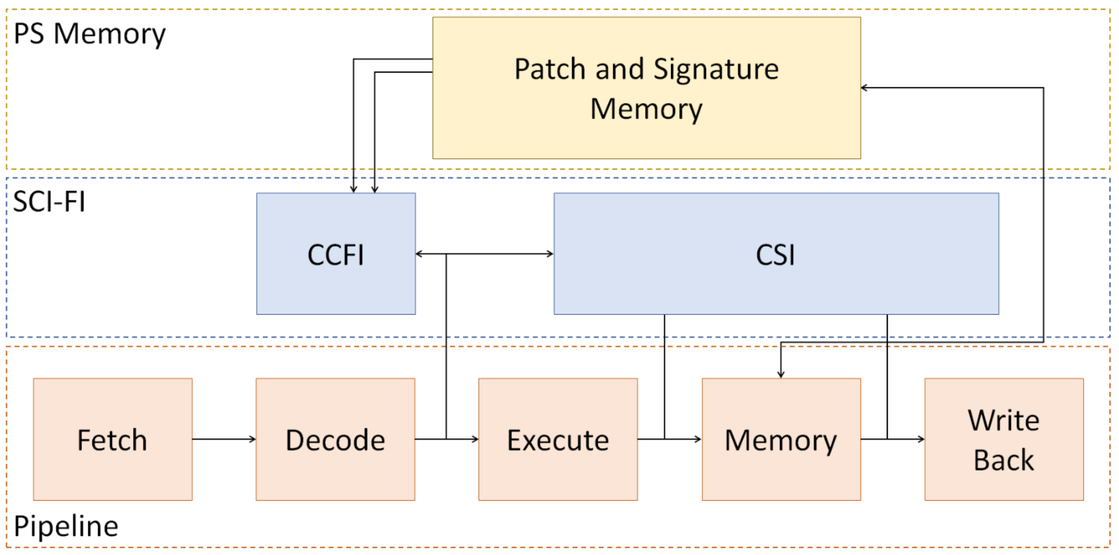
- implements GPSA and CSM within a pipeline
- cannot handle indirect jumps nor context switches
- requires a dedicated compiler toolchain

1 Generalized Path Signature Analysis and Continuous Singature Monitoring

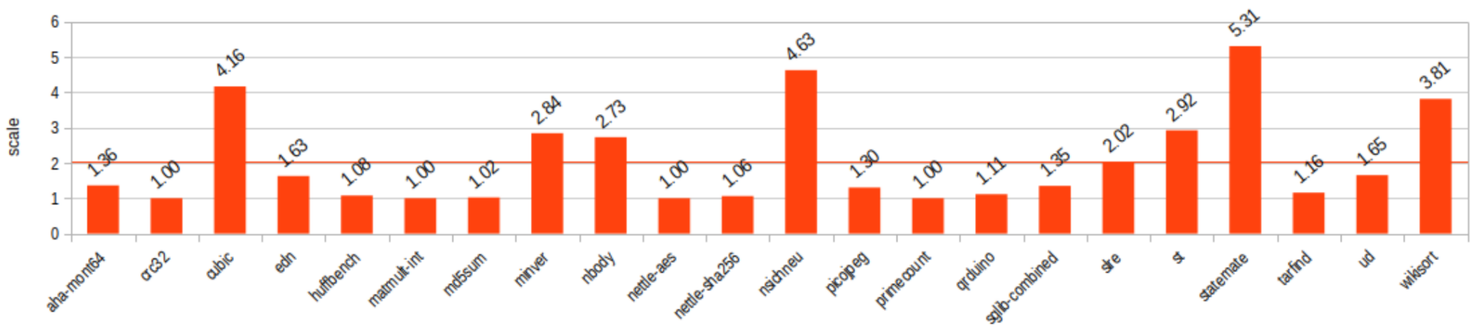
Our approach

Replacing the specialized compiler toolchain with a runtime
- can run any RISC-V executable off-the-shelf
- handle indirect jumps and context switches through Dynamic GPSA





Results



- implemented on the Comet RISC-V core [4]
- evaluated on embench-iot [5]
- worst performance slowdown at x5.31
- average slowdown of x2.05
- early results show important 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] Johan Laurent, Vincent Beroulle, Christophe Deleuze, Florian Pebay-Peyroula. Fault Injection on Hidden Registers in a RISC-V Rocket Processor and Software Countermeasures. 2019 Design, Automation & Test in Europe Conference & Exhibition (DATE), Mar 2019, Florence, Italy. pp.252-255,

[2] Werner, M., Wenger, E., Mangard, S. (2016). Protecting the Control Flow of Embedded Processors against Fault Attacks. In: Homma, N., Medwed, M. (eds) Smart Card Research and Advanced Applications. CARDIS 2015. Lecture Notes in Computer Science(), vol 9514. Springer, Cham.

[3] T. Chamelot, D. Couroussé and K. Heydemann, "SCI-FI: Control Signal, Code, and Control Flow Integrity against Fault Injection Attacks," 2022 Design, Automation & Test in Europe Conference & Exhibition (DATE), Antwerp, Belgium, 2022, pp. 556-559

[4] Simon Rokicki, Davide Pala, Joseph Paturel, Olivier Sentieys. What You Simulate Is What You Synthesize: Designing a Processor Core from C++ Specifications. ICCAD 2019 - 38th IEEE/ACM International Conference on Computer-Aided Design, Nov 2019, Westminster, CO, United States. pp.1-8.

[5] 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