Formal Security Verification of Processors

Scale4Edge

Scalable Infrastructure for Edge Computing

RPTU Kaiserslautern-Landau, Siemens EDA

SPONSORED BY THE

Federal Ministry of Education and Research

Motivation

"It's time for architects to redefine computer architecture and treat security as a first-class citizen […]" – Hennessy & Patterson's Turing Award lecture, 2018

"The security of our products is one of our most important priorities. […]" – Pat Gelsinger, CEO Intel, 2021

Never ending cycle of new attacks and selective patches calls for security guarantees

Formal Hardware Verification

"Formal" Verification

- Exhausts a design's functional space by rigorous mathematical methods
- Well-defined coverage

State of the art

- Verification of functional correctness
- **Not covered:** Microarchitectural Timing Side Channels

The Guardian

This article is more than **2 years old**

Meltdown and Spectre: 'worst ever' CPU bugs affect virtually all computers

Everything from smartphones and PCs to cloud computing affected by major security flaw found in Intel and other processors – and fix could slow devices

Spectre and Meltdown processor security flaws – explained

Meltdown and Spectre security flaws: so big they have their own logos. Photograph: tcareob72/Natascha Eibl/Getty Images/iStockphoto

Samuel Gibbs

Thu 4 Jan 2018 12.06 GMT

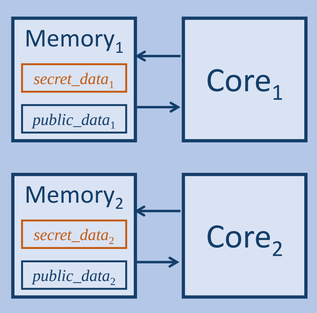
UPEC: Formal Security Verification

Threat model

- User-level program (attacker) steals secret data

UPEC = *Unique Program Execution Checking*

- Formally verifies whether results and timing of a user-level program exhibits dependence on confidential data
- Operates at the Register Transfer Level



$$AG(secret\_data\_protected \wedge \mu\_state_1 = \mu\_state_2 \rightarrow AG\ arch\_state_1 = arch\_state_2)$$

→ UPEC guarantees confidentiality of a processor

UPEC in Action

Case Study: TGC RISC-V cores by Minres

We applied UPEC to the Minres TGC cores developed in S4E and detected 5 security bugs

Core Version	**Security Bug**
TGC_D 0.8.0	*MEPC CSR was not initialized properly*
TGC_D 0.8.0	*Accesses were blocked without PMP configuration*
TGC_D 0.8.1	*PMP NA4 mode granularity too large*
TGC_D 0.8.1	*PMP NAPOT mode: incorrect address masks*
TGC_D 0.8.1	*PMP TOR mode: incorrect address bounds*

All bugs were fixed by Minres, UPEC certified fixed design

UPEC integration into commercial OneSpin 360 tool

- Siemens EDA develops UPEC app
- Integration into SEDA OneSpin with high degree of automation

UPEC Results

UPEC detected several security vulnerabilities in processors.

	Rocket	RI5CY	TGC_D/E	Ariane	BOOM
Pipeline	*5-stage*	*4-stage*	*4/5-stage*	*5-stage*	*12-stage*
Out-of-order execution	*no*	*no*	*no*	*Score-board*	*Deep out-of-order*
Detected Vulnerabilities	*ORC, Bugs*	*Bugs in PMP*	*Bugs in PMP*	*Integrity bugs*	*Spectres and Meltdown*
Vendor/ Organization	*CHIPS Alliance*	*OpenHW Group*	*Minres GmbH*	*lowRISC*	*UC Berkley*

UPEC certified security after fixing all security bugs.

Publications and Awards

- M. R. Fadiheh, J. Müller, R. Brinkmann, S. Mitra, D. Stoffel, and W. Kunz: *A Formal Approach for Detecting Vulnerabilities to Transient Execution Attacks in Out-of-Order Processors*, DAC'20.
- J. Müller, M. R. Fadiheh, A. Duque-Antón, T. Eisenbarth, D. Stoffel, W. Kunz: *A Formal Approach to Confidentiality Verification in SoCs at the Register Transfer Level,* DAC'21. **Intel Hardware Security Academic Award**
- L. Deutschmann, J. Müller, M. R. Fadiheh, D. Stoffel, and W. Kunz: *Towards a Formally Verified Hardware Root-of-Trust for Data-Oblivious Hardware,* DAC'22. **Best Paper Award**
- M. R. Fadiheh, A. Wezel, J. Müller, J. Bormann, S. Ray, J. M. Fung, S. Mitra, D. Stoffel, and W. Kunz: *An Exhaustive Approach to Detecting Transient Execution Side Channels in RTL Designs of Processors,* TC'23.

This work has been developed in the ZuSE project Scale4Edge. Scale4Edge is funded by the German ministry of education and research (BMBF) (reference numbers: 16ME0122K-16ME0140+16ME0465). The authors are responsible for the content of this publication.

RPTU

SIEMENS