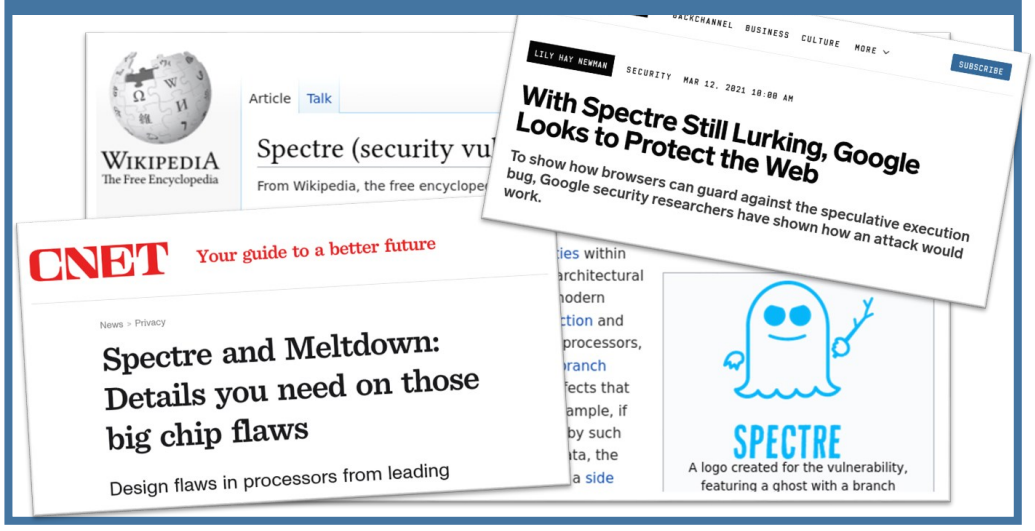
Hardware-Software Co-Design for End-to-End Security

How to get provable security from software down to hardware?

Crypto is everywhere...

... but also side channels



Goal: Hardware-software collaboration for security

PROSPECT: Provably Secure Speculation for the Constant-Time Policy

Lesly-Ann Daniel[1], Marton Bognar[1], Job Noorman[1],

Sébastien Bardin[2], Tamara Rezk[3] and Frank Piessens[1]

Architectural Mimicry: Innovative Instructions to Efficiently Address Control-Flow Leakage in Data-Oblivious Programs

Hans Winderix	Marton Bognar	Job Noorman	Lesly-Ann Daniel	Frank Piessens
imec-DistriNet	*imec-DistriNet*	*imec-DistriNet*	*imec-DistriNet*	*imec-DistriNet*
KU Leuven	*KU Leuven*	*KU Leuven*	*KU Leuven*	*KU Leuven*

Remaining challenges

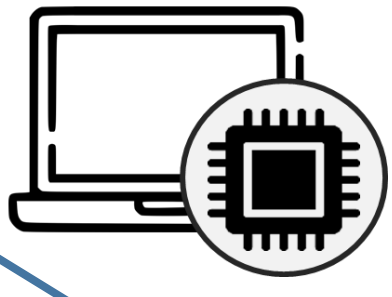
Explore new HW-SW contracts

- Secure balanced branches
- End-to-end provable crypto security
- Protect registers during speculation

Security property

HW/SW contract

Leakage abstraction



Secure compilation

- Compiler-support for HW defense
- ⇒ Larger scale evaluation
- Secure compilation (Jasmin)
- ⇒ Provable end-to-end security

Hardware Validation

- Hardware fuzzing
- ⇒ Correctness vs. contract adherence
- ⇒ From Black-Box to White-Box
- Hardware validation