Developing a Test Suite for Transient-Execution Attacks on RISC-V and CHERI-RISC-V

Franz A. Fuchs, Jonathan Woodruff, Simon W. Moore, Peter G. Neumann, Robert N. M. Watson

University of Cambridge and SRI International

CARRV – Online, 17 June 2021

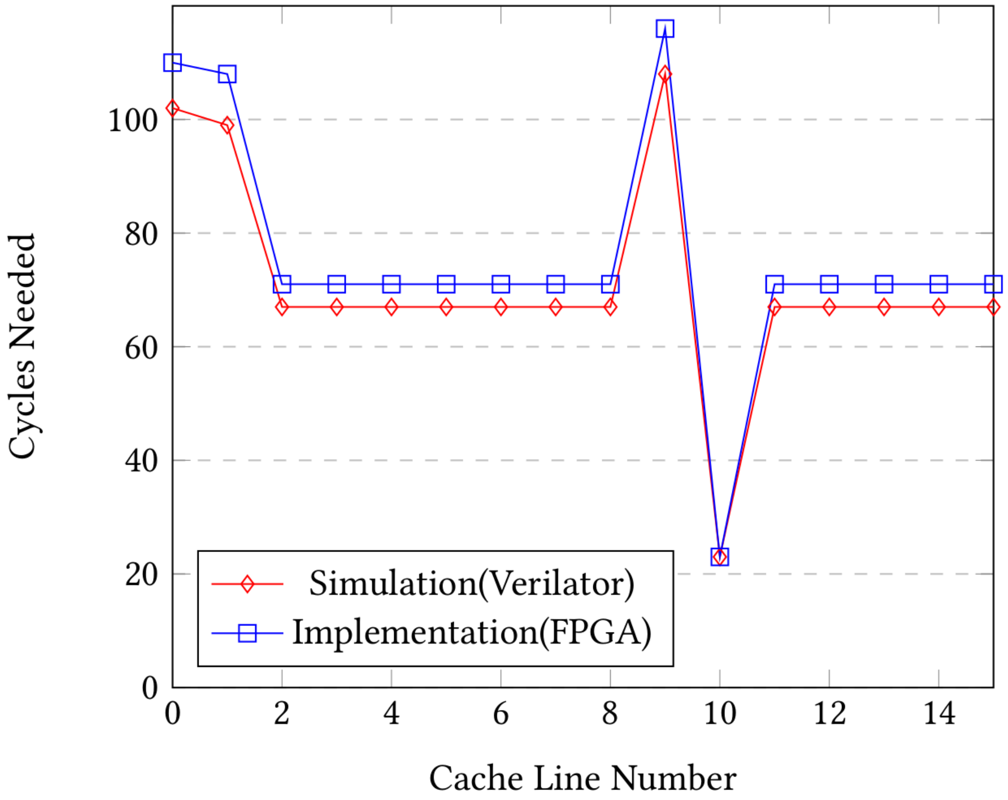
Approved for public release; distribution is unlimited. This research is sponsored by the Defense Advanced Research Projects Agency (DARPA) and the Air Force Research Laboratory (AFRL), under contract FA8750-10-C-0237 ("CTSRD") and HR0011-18-C-0016 ("ECATS"). The views, opinions, and/or findings contained in this presentation are those of the author(s)/presenter(s) and should not be interpreted as representing the official views or policies of the Department of Defense or the U.S. Government.

Overview

- Transient-execution attacks
- Capability Hardware Enhanced RISC Instructions (CHERI)
- Test suite for transient-execution attacks
- Explaining sample attacks
- Discussing the test framework

Transient-Execution Attacks

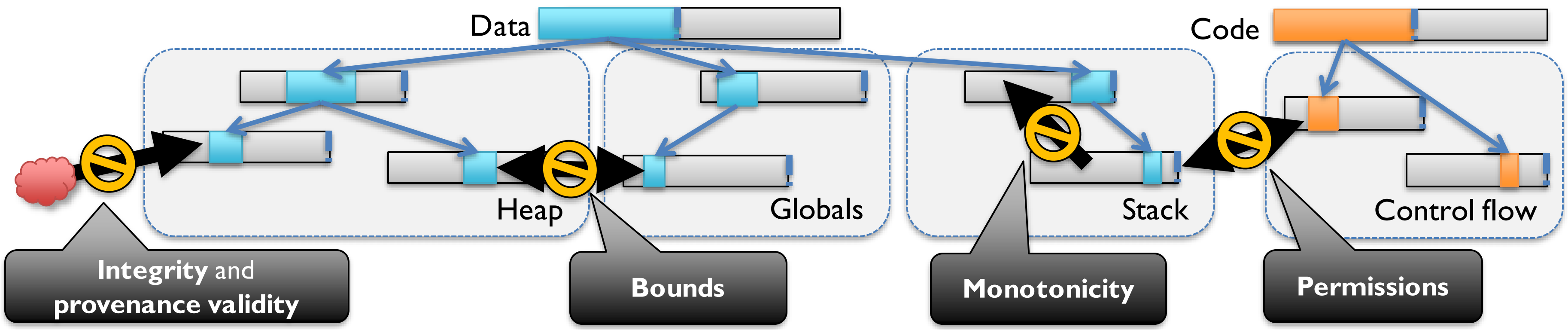
- Facilitated by speculation and out-of-order execution
- Attacks trick microarchitectures into performing actions that are architecturally prohibited
- Leads to microarchitectural state changes
- Obtain secret via side channels



Test Framework

- Problem: most attacks have been detected in shipped processors that cannot be fixed until the next generation
- Proposed approach: detect attacks at design time in simulation and on FPGA
- Demonstration on a superscalar RISC-V core
- Demonstrate that CHERI fine-grained memory protection can mitigate a subclass of transient-execution attacks
- Lays foundation for developing/implementing mitigation mechanisms

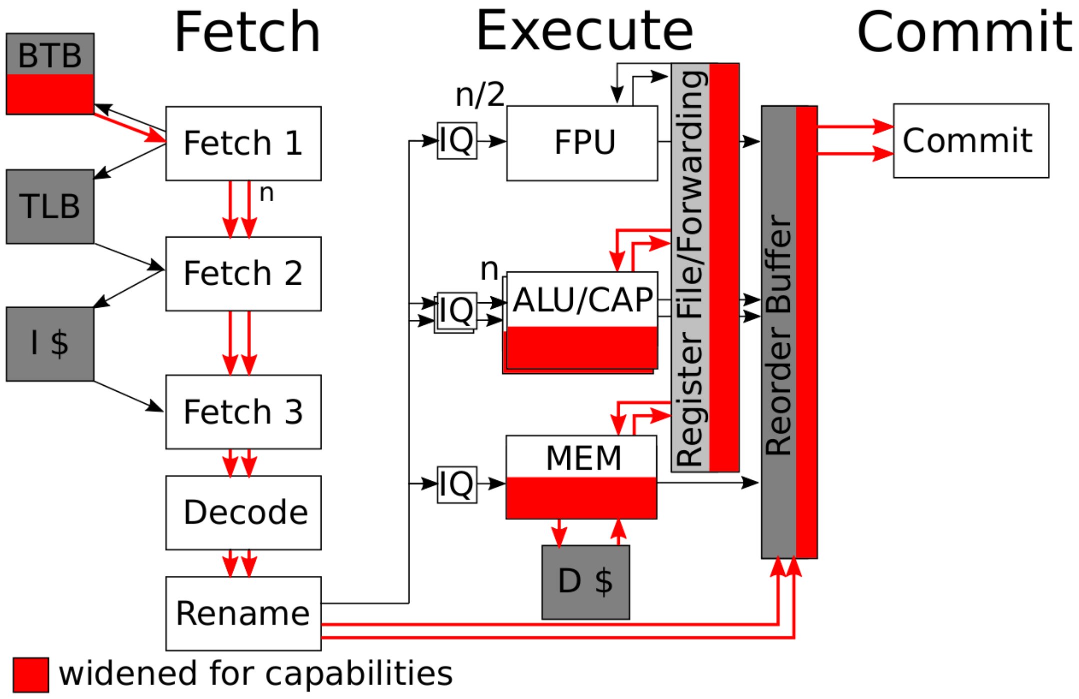
CHERI Architecture: Pointers become Capabilities



- CHERI Architectural Capabilities (Watson et al., CHERI ISAv8, 2020).
  - Architecturally-defined, "fat pointer" with one-bit validity tag.
  - Carry *cursor*, *base*, and *top* addresses (and *permission* and *type* bits).
  - CPU enforces bounds and permissions checks on dereference operations.
  - Overwriting a capability with data clears the validity tag.
  - CPU instructions ensure no de-novo validity or enlarging of bounds.
- CHERI composes with "host" ISAs: here, RISC-V; but also MIPS & Arm's Morello.

CHERI RiscyOO

- 2 superscalar processor
- 64-bit integer width / 129-bit capability address width
- 129-bit general-purpose registers
- Implements both RISC-V and CHERI-RISC-V



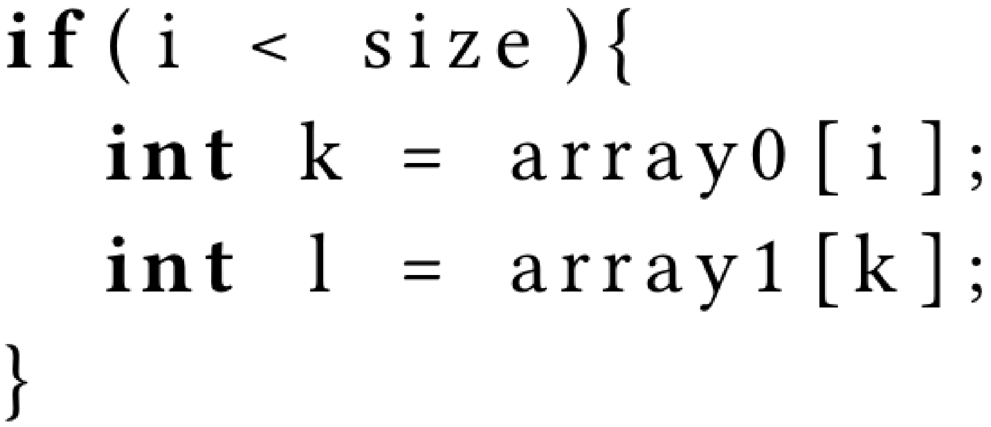
Results

	RISC-V	CHERI-RISC-V
Spectre-PHT	S	U
Spectre-BTB	S	S
Spectre-RSB	S	S
Spectre-STL	S	S
Meltdown-US	U	n/a
Meltdown-US-CHERI	n/a	U
Meltdown-GP	U	n/a
Meltdown-GP-CHERI	n/a	U

(S)uccessful, (U)nsuccessful

Spectre-PHT

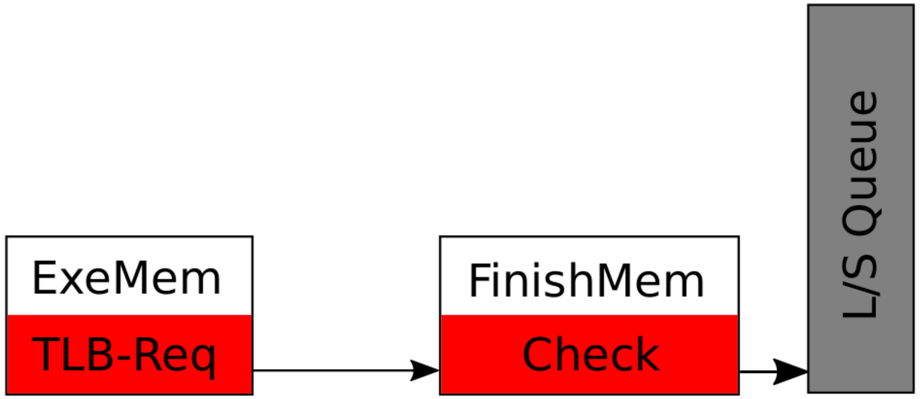
- Speculative Bounds Bypass
- Arrays become capabilities
- CHERI systems can mitigate this attack
- Tight capability configuration needed, otherwise it will be successful



```
if(i < size){
    int k = array0[i];
    int l = array1[k];
}
```

Mitigating Meltdown-style attacks

- Caused by late pipeline checks
- (CHERI) RiscyOO mitigates all attempted Meltdown-style attacks
- Checks are performed before memory addresses are issued



Discussing the Test Framework

- Spectre-PHT and Spectre-STL violate the RISC-V security model since the software's guarantees cannot be held
- In addition, Spectre-BTB and Spectre-RSB break CHERI's security model
- CHERI partially mitigates the transient-execution attack class when operating in pure-capability mode

Discussing the Test Framework

- Extensible for future RISC-V/CHERI-RISC-V implementations (or iterations of current ones) and yet to be detected transient-execution attacks
- Lays a foundation for hardware verification and development / implementation of mitigation mechanisms both in hardware and software

Conclusions

- We demonstrate that:
  - RiscyOO is vulnerable to all Spectre-style attacks and is, therefore, a good target for research into speculative execution
  - CHERI fine-grained memory protection can mitigate one of the four attacks
- Our test suite is a valuable tool to check mitigation mechanisms at design time
  - Find our test suite at:

https://github.com/CTSRD-CHERI/Test-Suite-Transient-Execution

- Thanks to the entire CHERI team and our sponsors: NCSC under the UK RISE Initiative, Defense Advanced Research Projects Agency (DARPA), EPSRC REMS Programme,Arm Limited, and Google, Inc