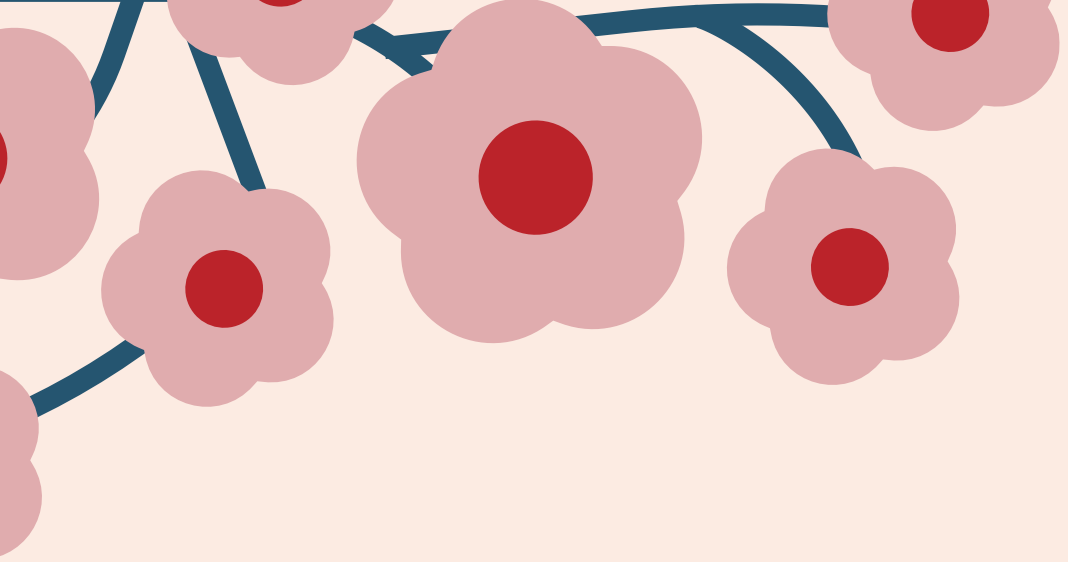
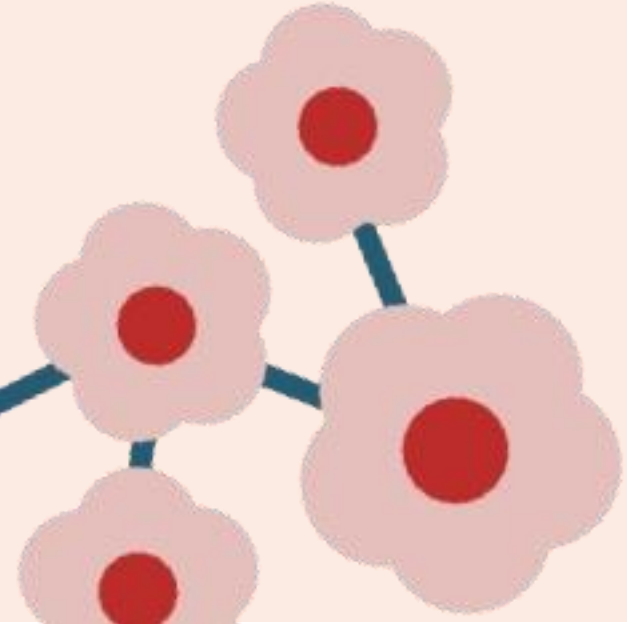
06 April 2026



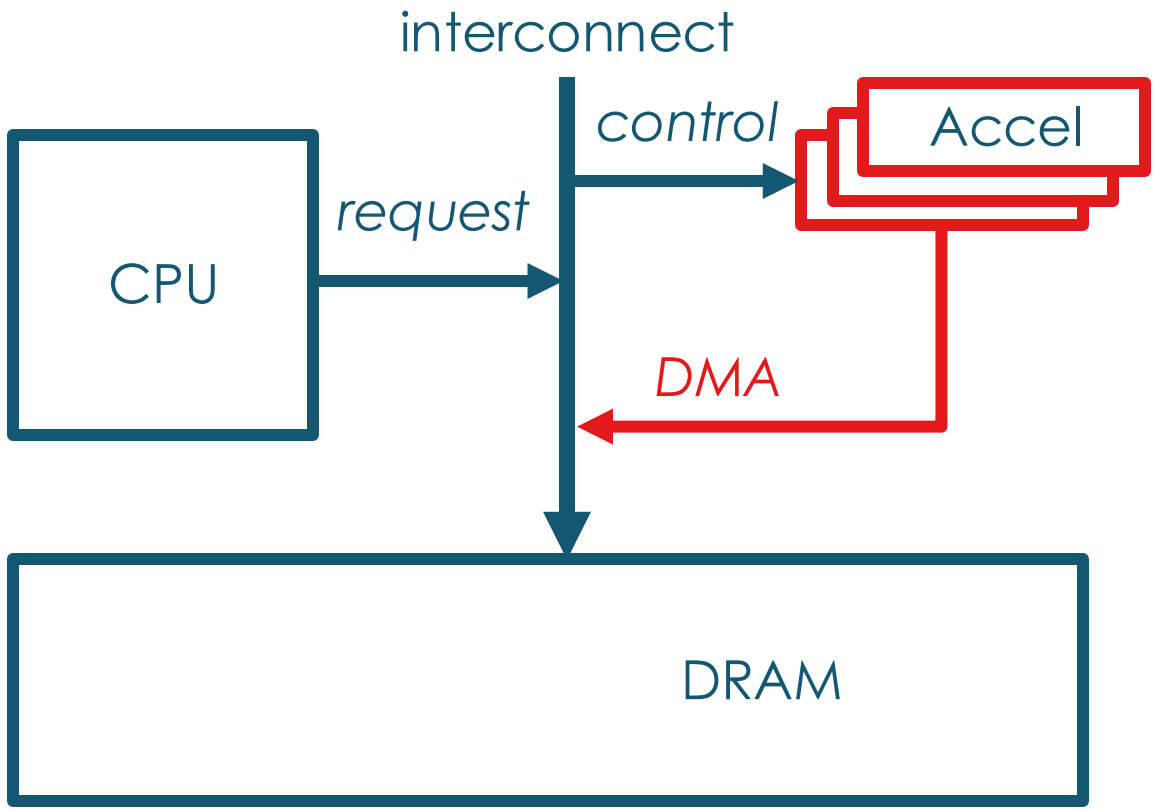


Towards Heterogeneous CHERI Compartmentalization

Jianyi Cheng
Assistant Professor – University of Edinburgh

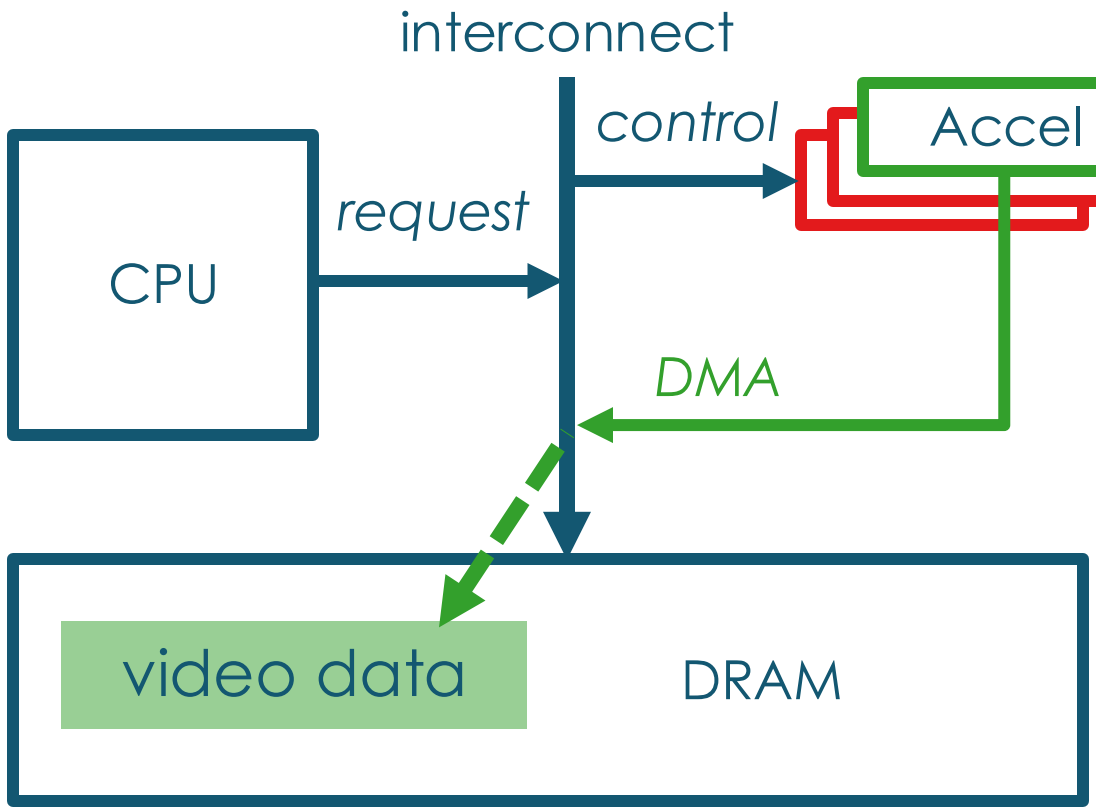
Accelerators are good but…

- Attack vectors: malicious use of accelerators



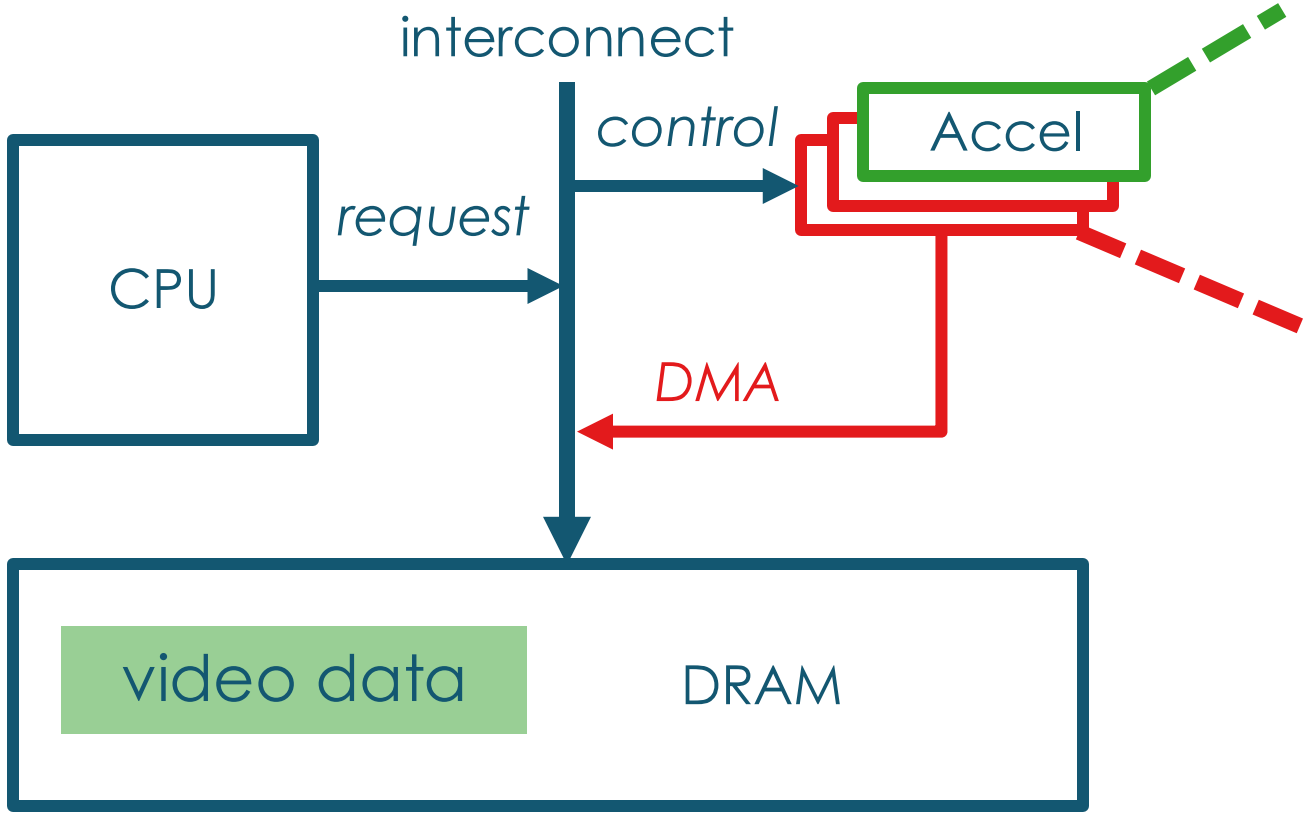
Accelerators are good but…

- Attack vectors: malicious use of accelerators



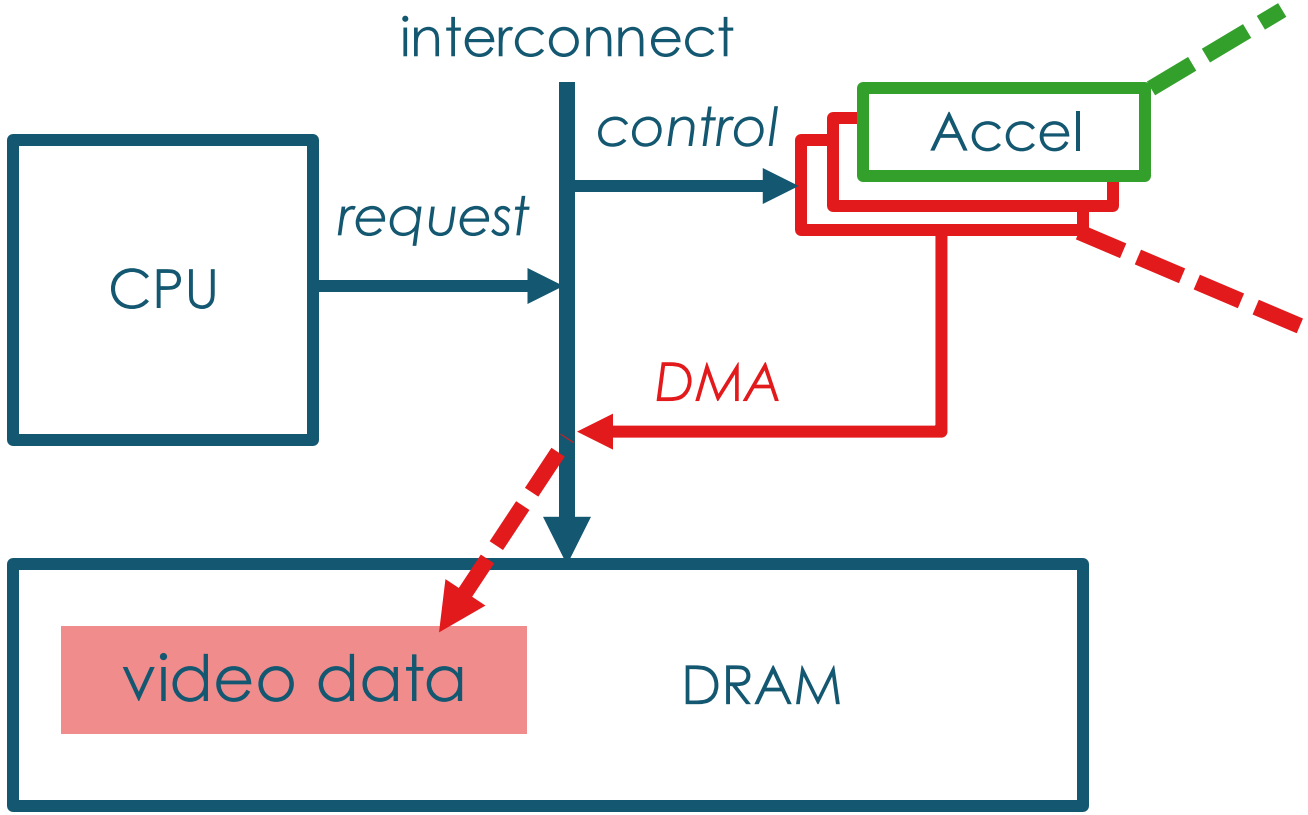
Accelerators are good but...

- Attack vectors: malicious use of accelerators



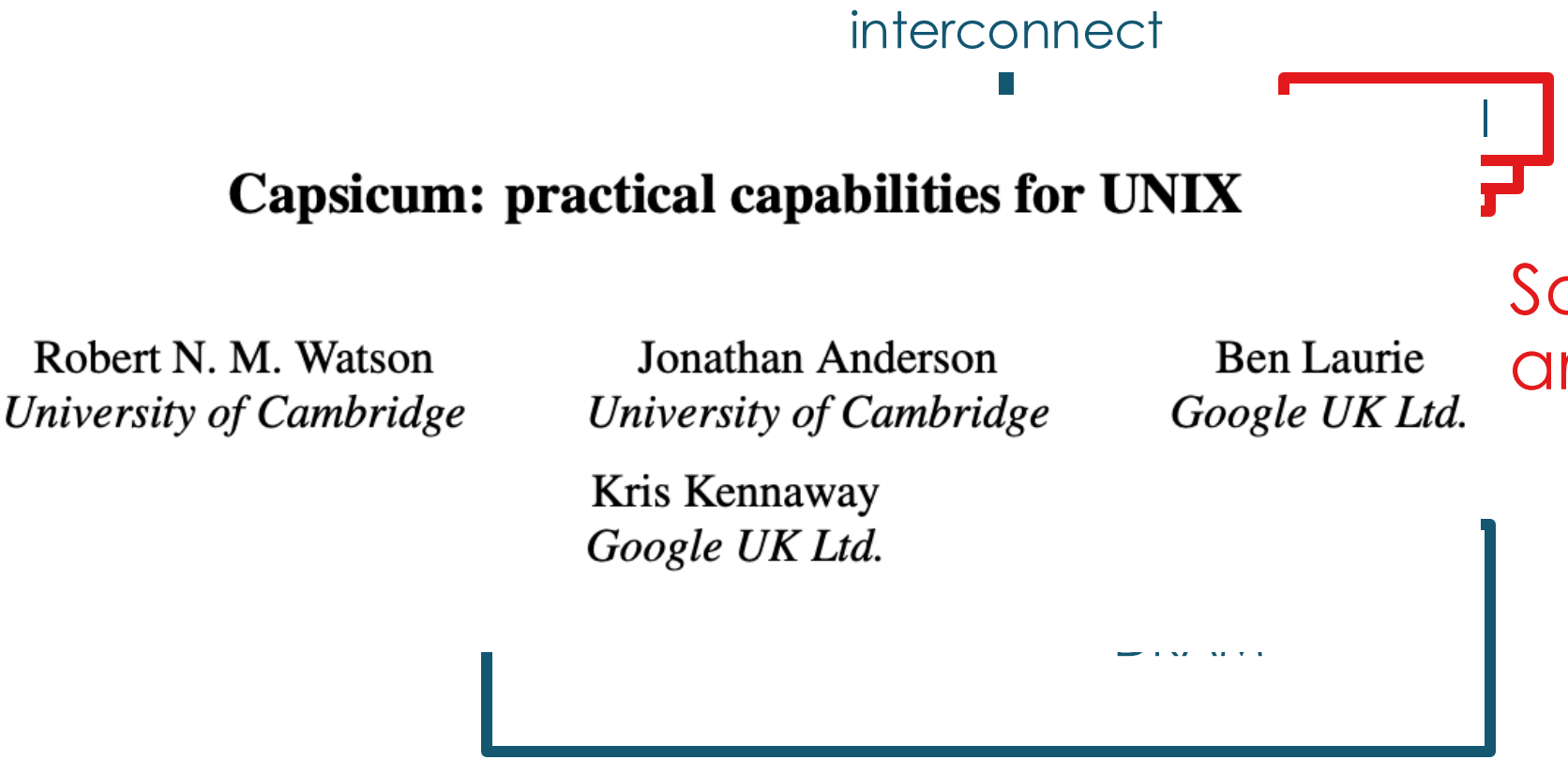
Accelerators are good but…

- Attack vectors: malicious use of accelerators



Accelerators are good but…

- Attack vectors: malicious use of accelerators



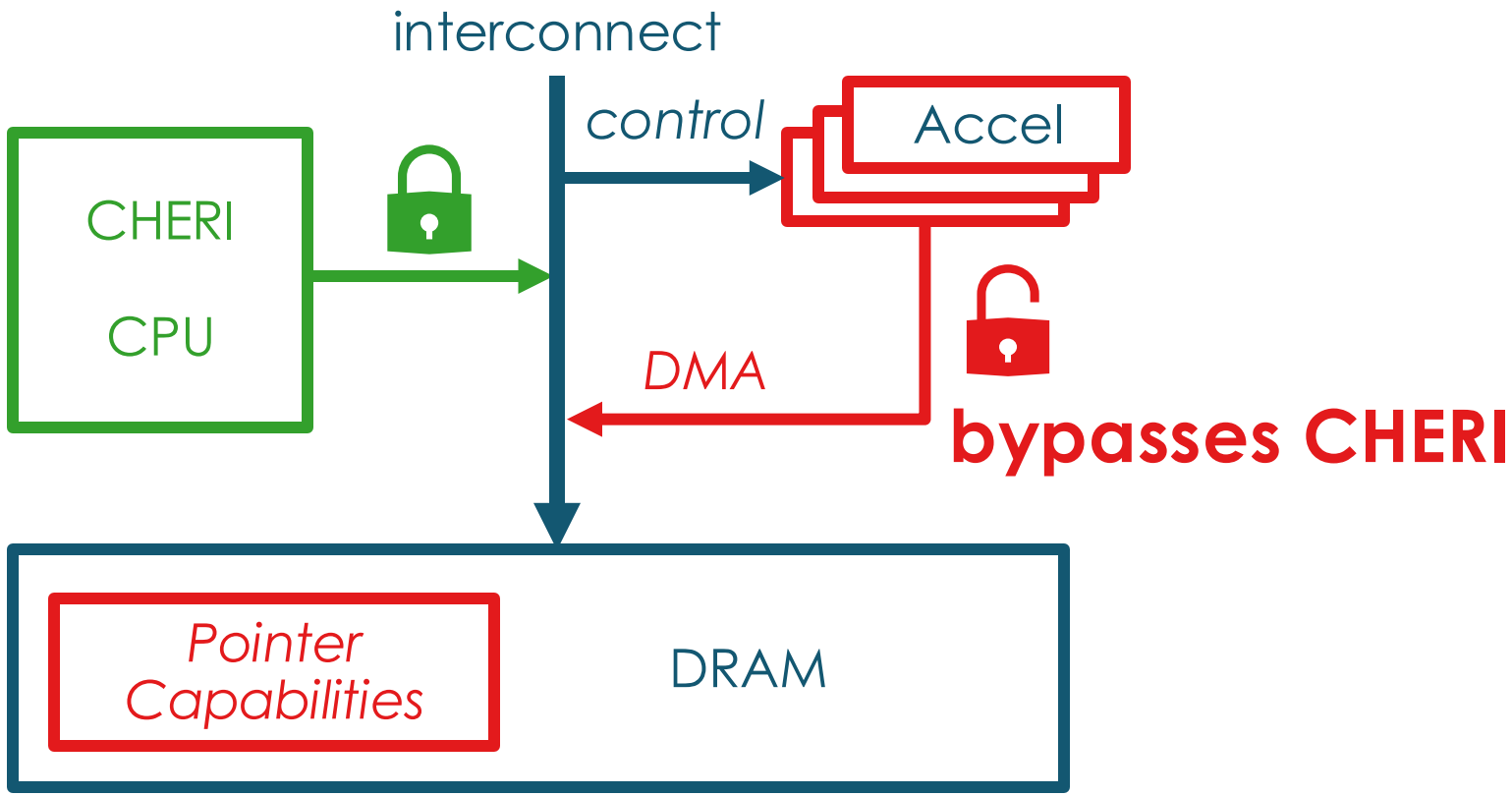
Software solutions are complex…

How about hardware solutions?

CHERI

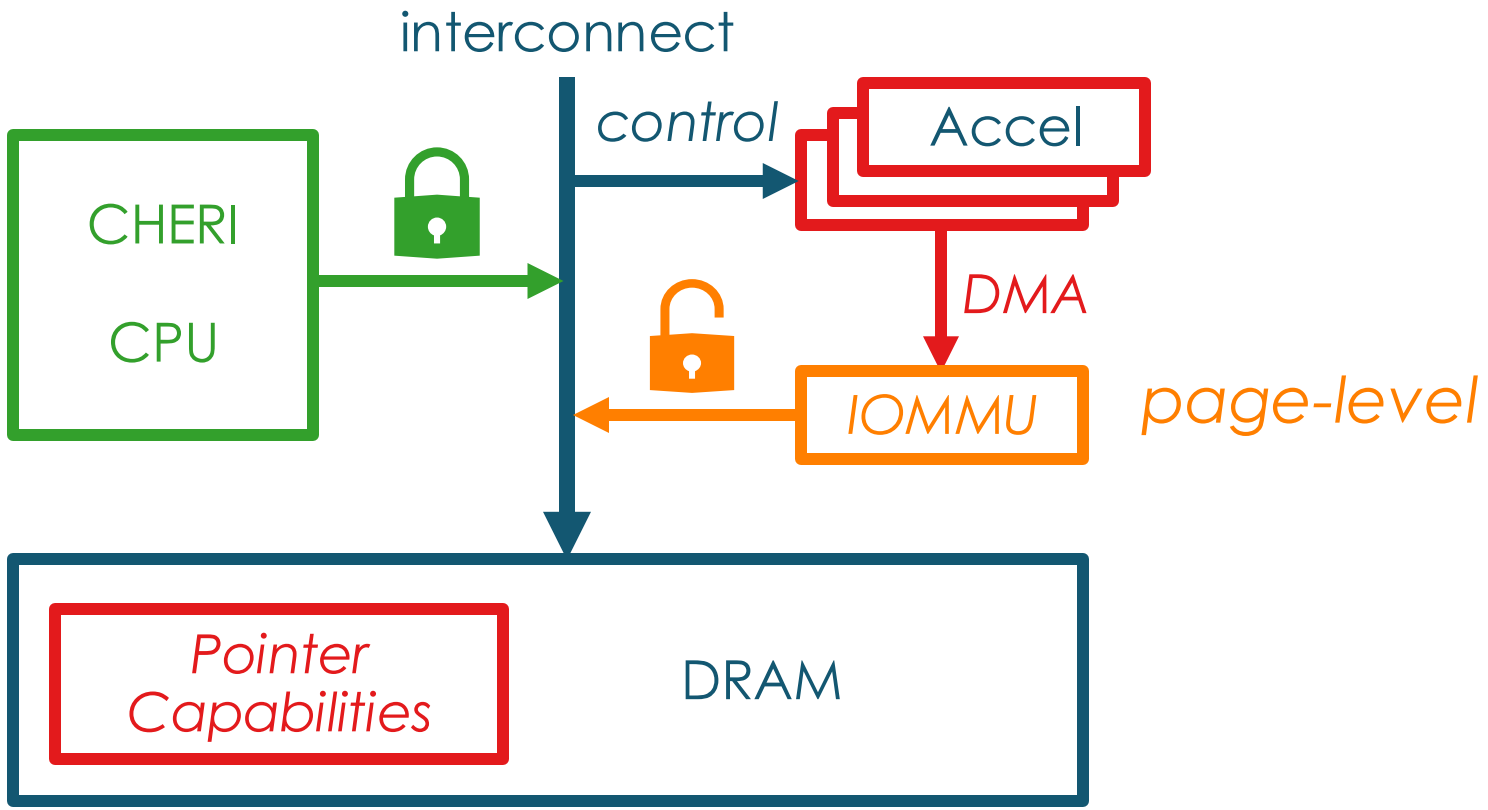
Memory safety issues are concerning

- Attack vectors: malicious use of accelerators



Memory safety issues are concerning

- Attack vectors: malicious use of accelerators



Memory safety issues are concerning

- Attack vectors: malicious use of accelerators

2024 ACM/IEEE 51st Annual International Symposium on Computer Architecture (ISCA)

sNPU: Trusted Execution Environments on Integrated NPUs

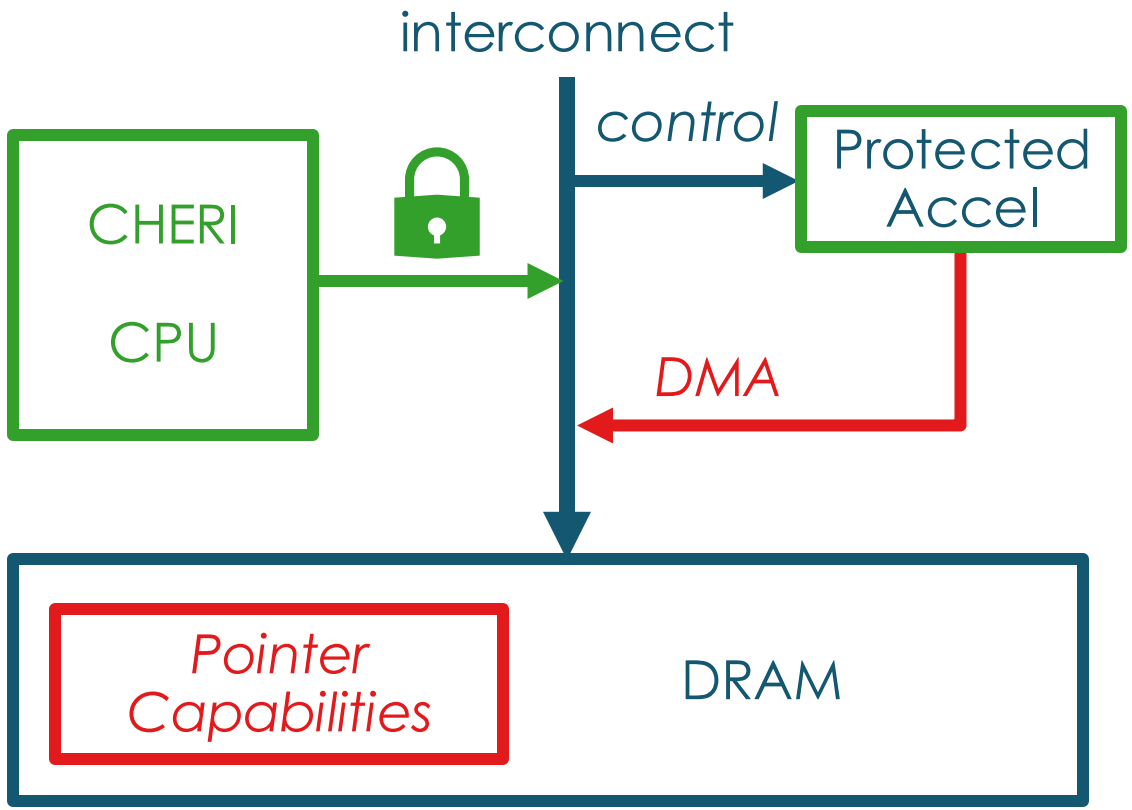
Erhu Feng[1†◇], Dahu Feng[1‡], Dong Du[†◇], Yubin Xia[†◇], Haibo Chen[†◇§]
[†]*Institute of Parallel and Distributed Systems, SEIEE, Shanghai Jiao Tong University*
[‡]*Department of Precision Instrument, Tsinghua University*
[◇]*Engineering Research Center for Domain-specific Operating Systems, Ministry of Education, China*
[§]*Key Laboratory of System Software (Chinese Academy of Sciences)*

Capabilities

CHERI

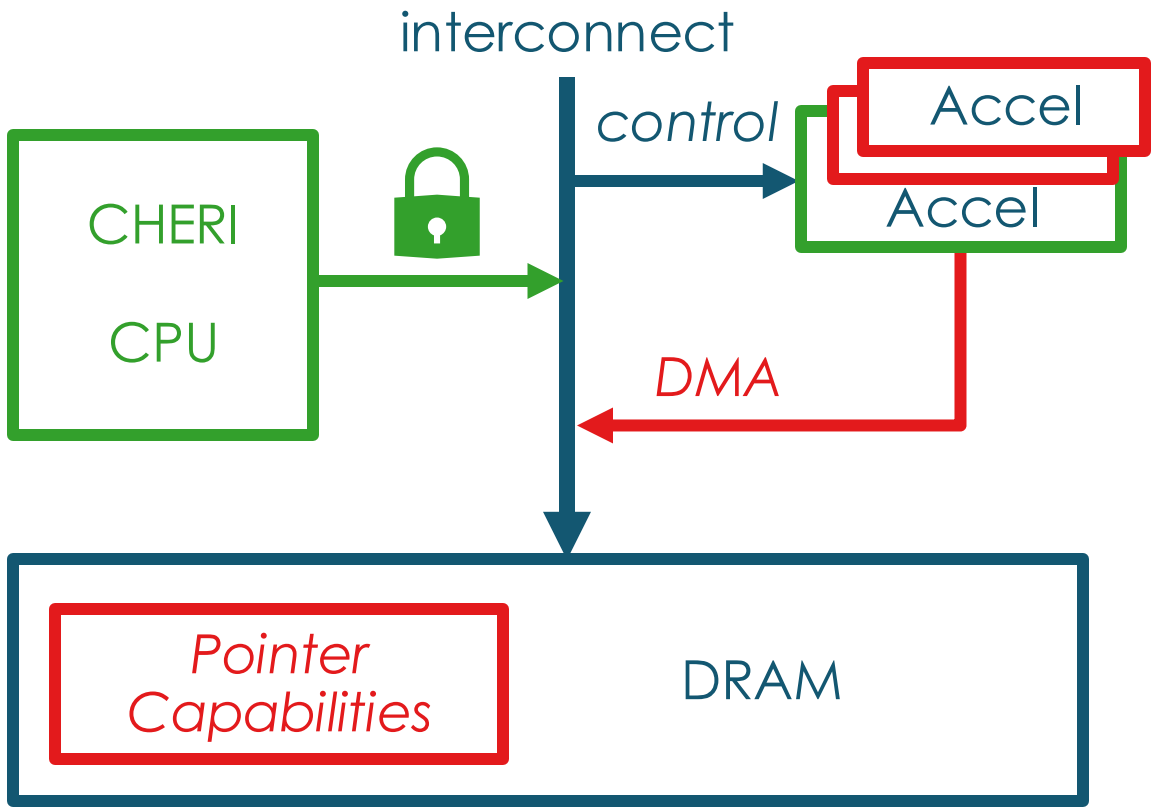
Memory safety issues are concerning

- Attack vectors: malicious use of accelerators



Memory safety issues are concerning

- Attack vectors: malicious use of accelerators



Research Challenges

Coarse Granularity

IOMMUs provide memory protection at *the page level.*

Protection Heterogeneity

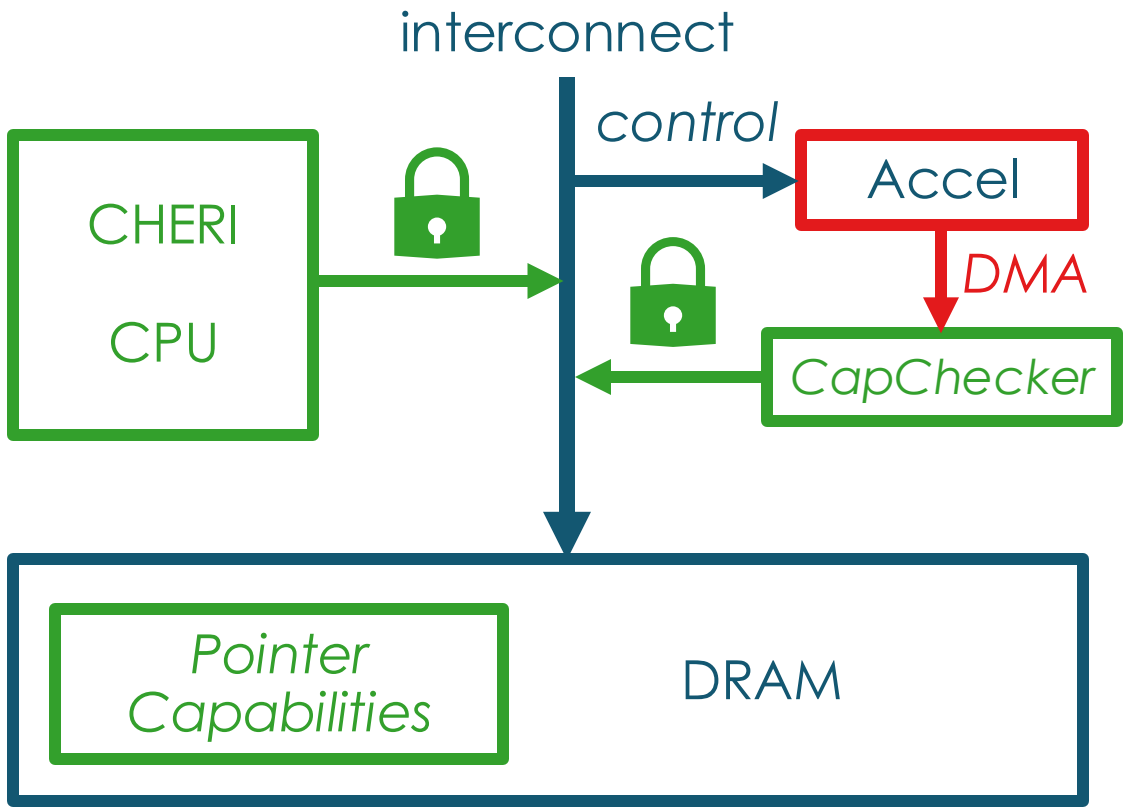
Different targets speak *different policies* for protection.

Our Goals

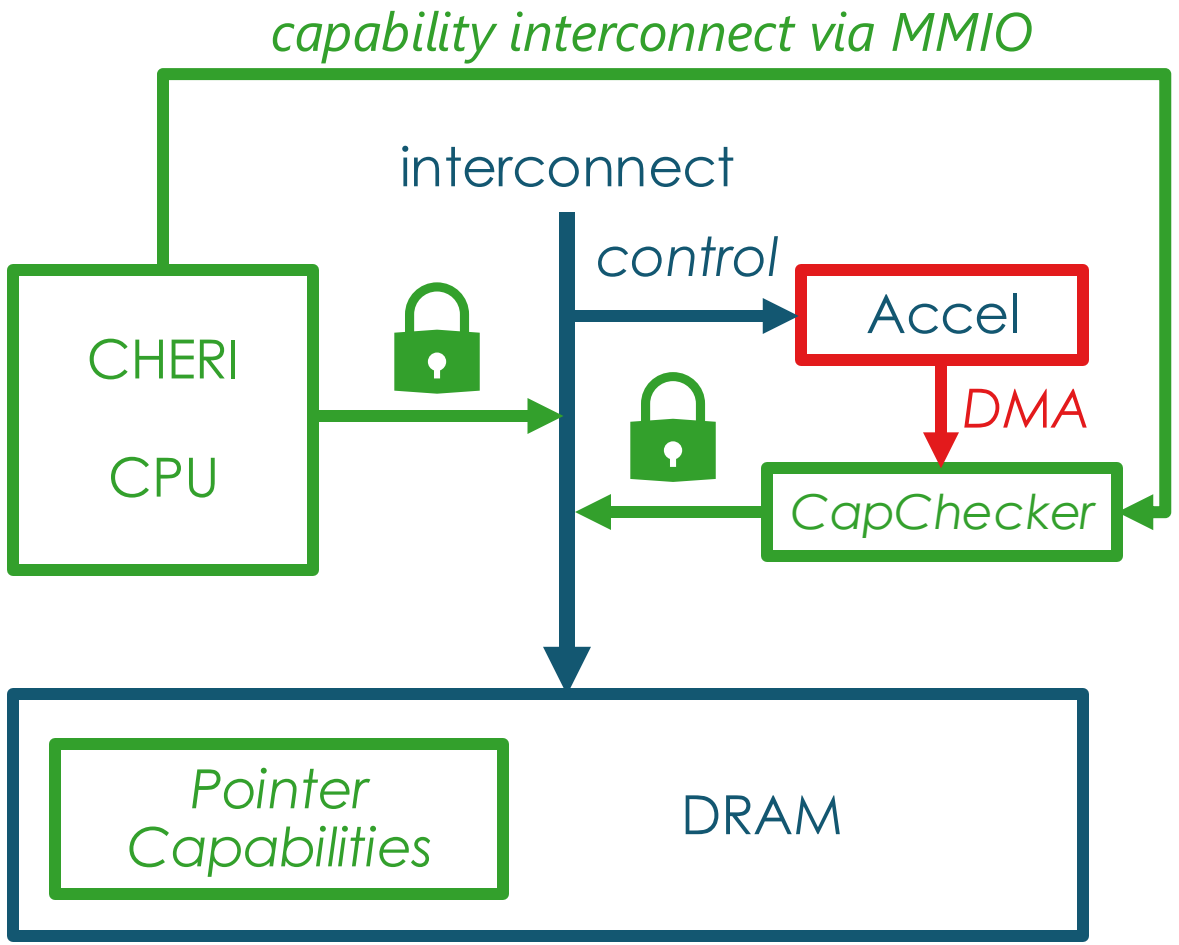
Fine Granularity

Accelerator Independent

CapChecker: Hardware Capability Checker



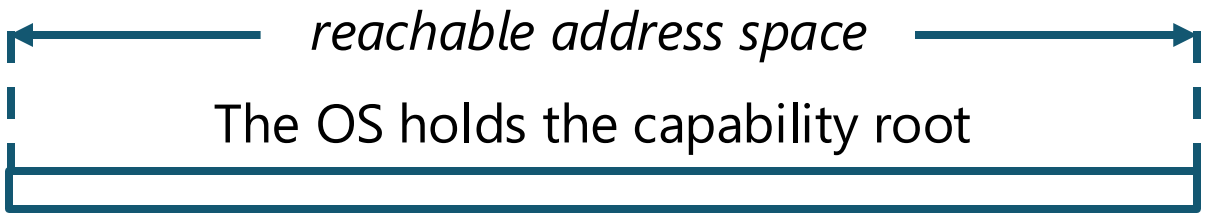
CapChecker: Hardware Capability Checker



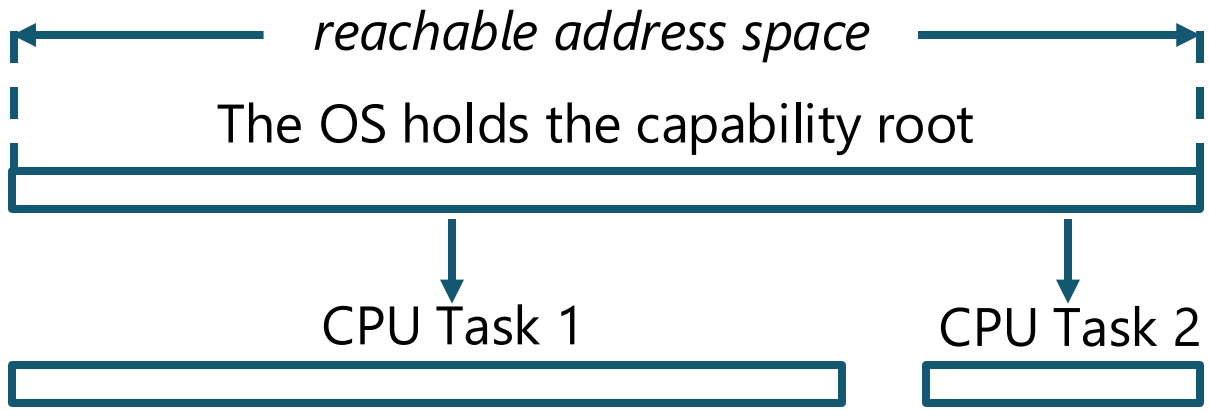
Threat Model

- CPU is already protected by CHERI
- No dynamic memory management on accelerators
- Software components are trustworthy.

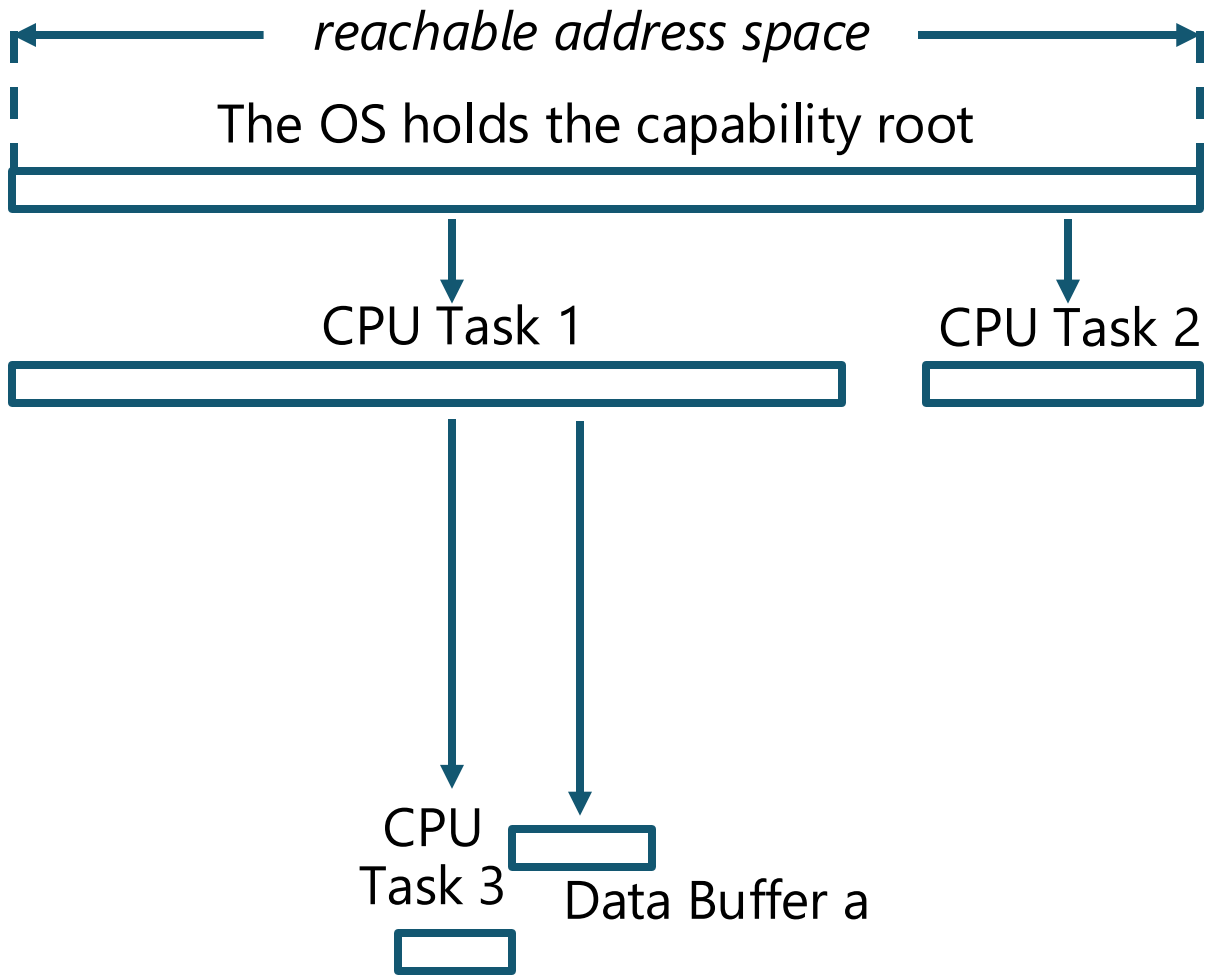
Capability Model



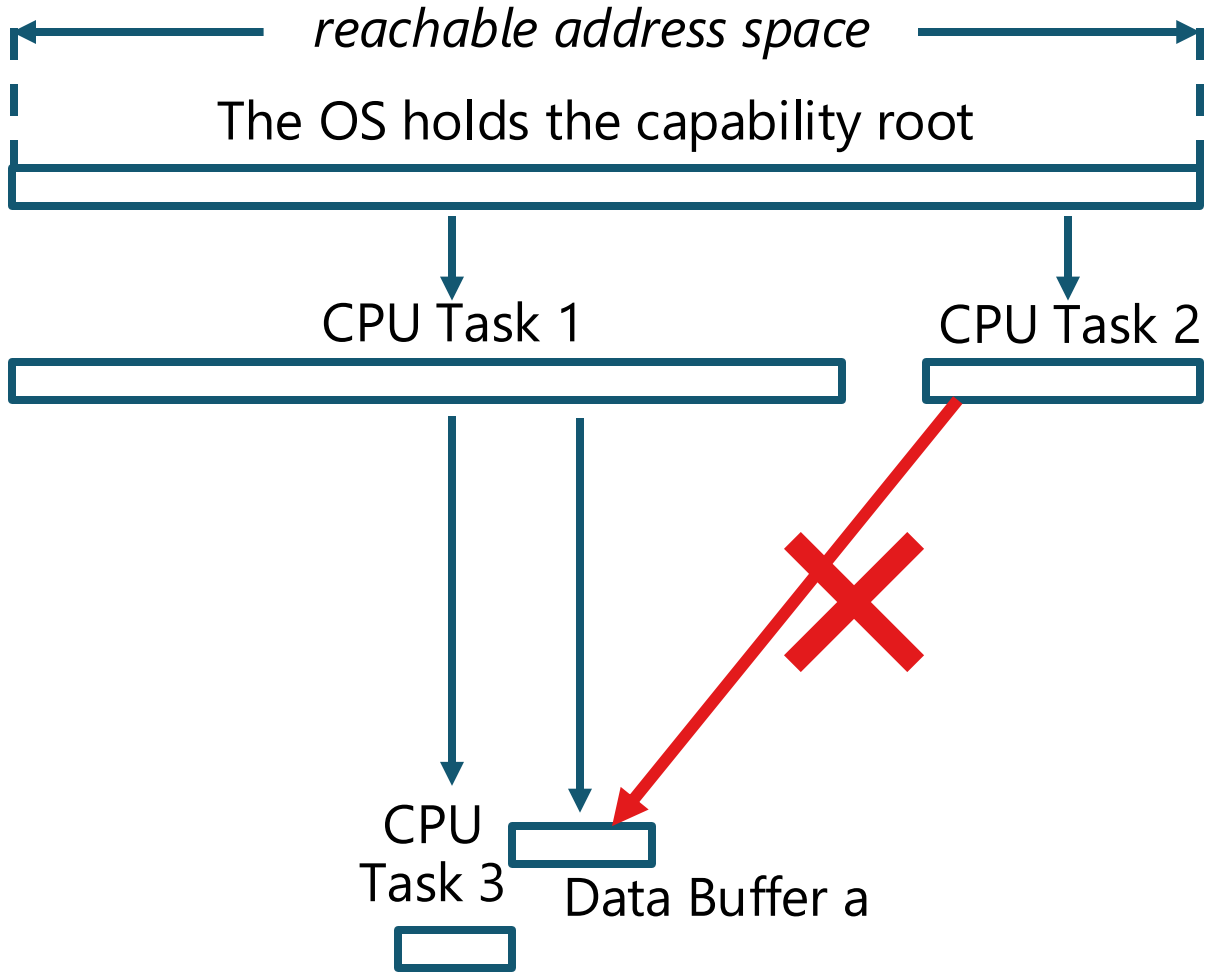
Capability Model



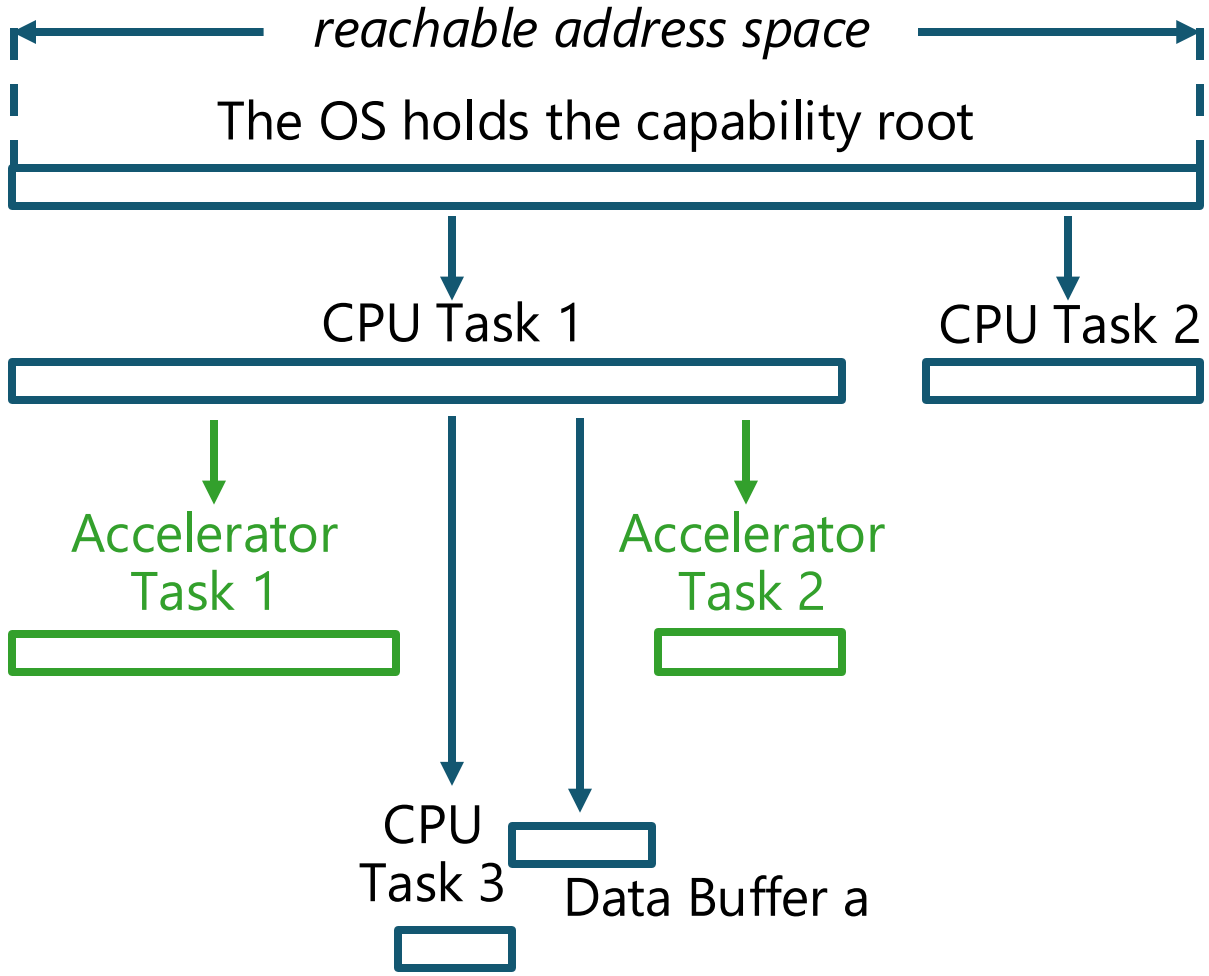
Capability Model



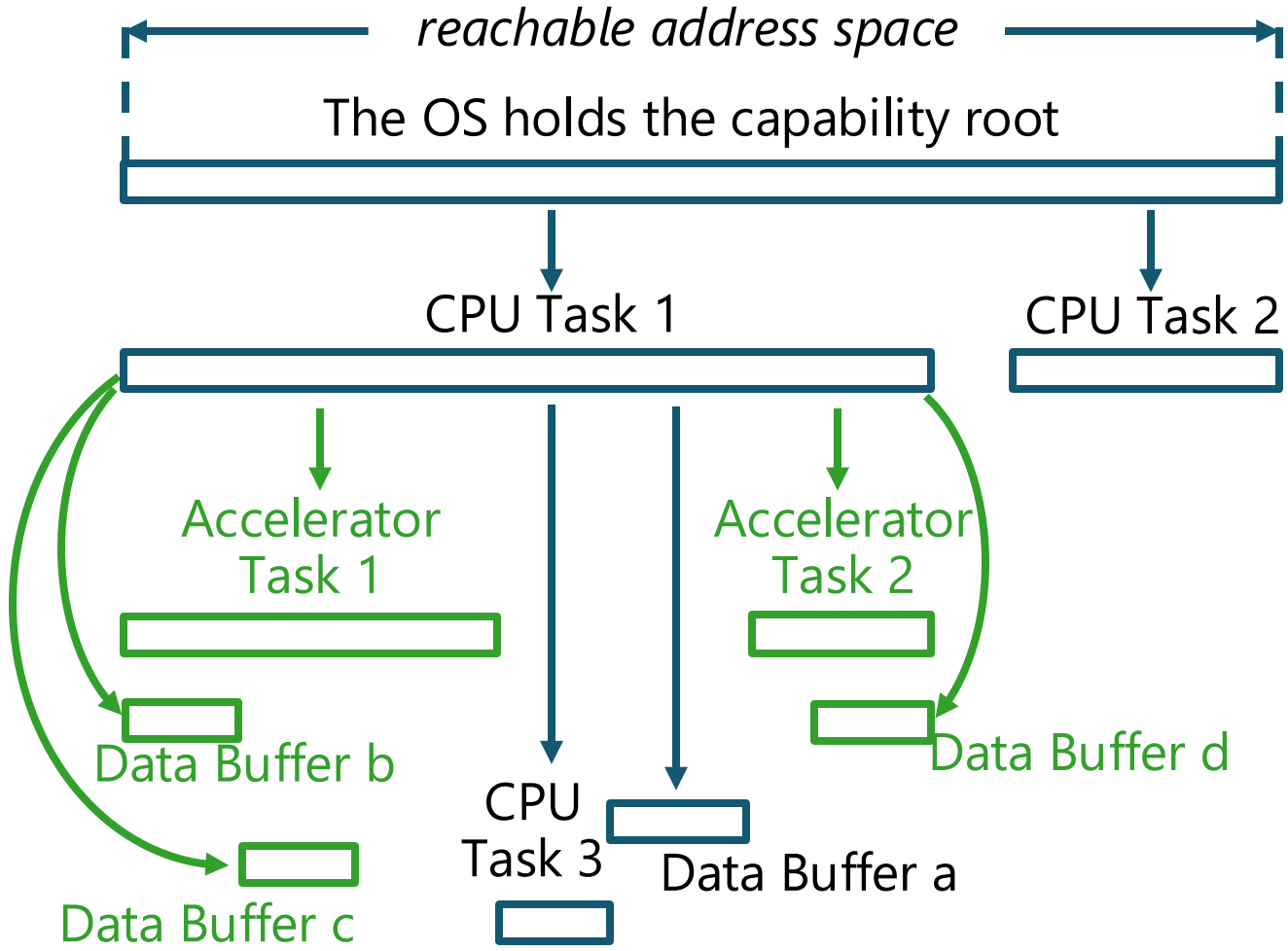
Capability Model



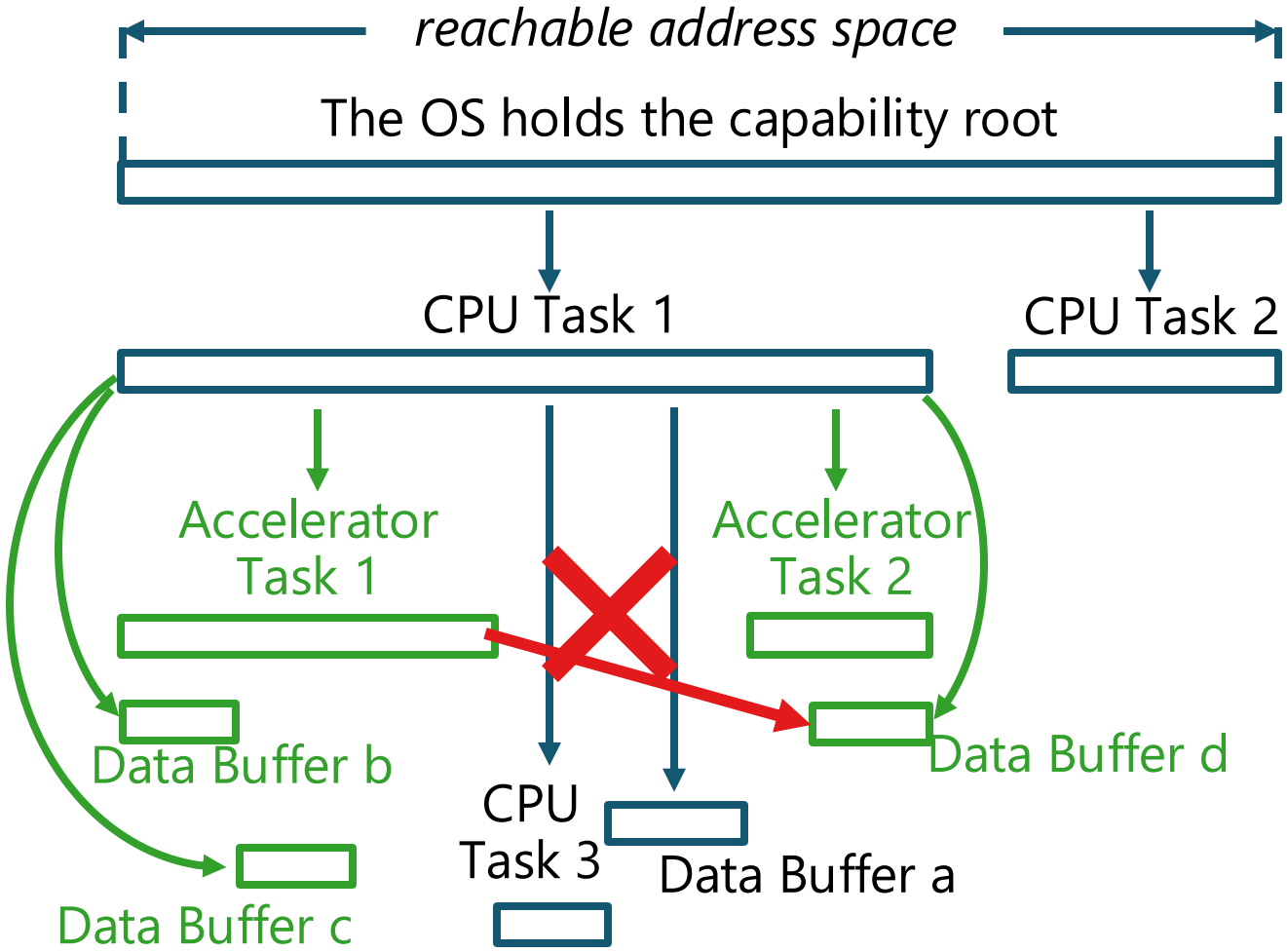
Capability Model



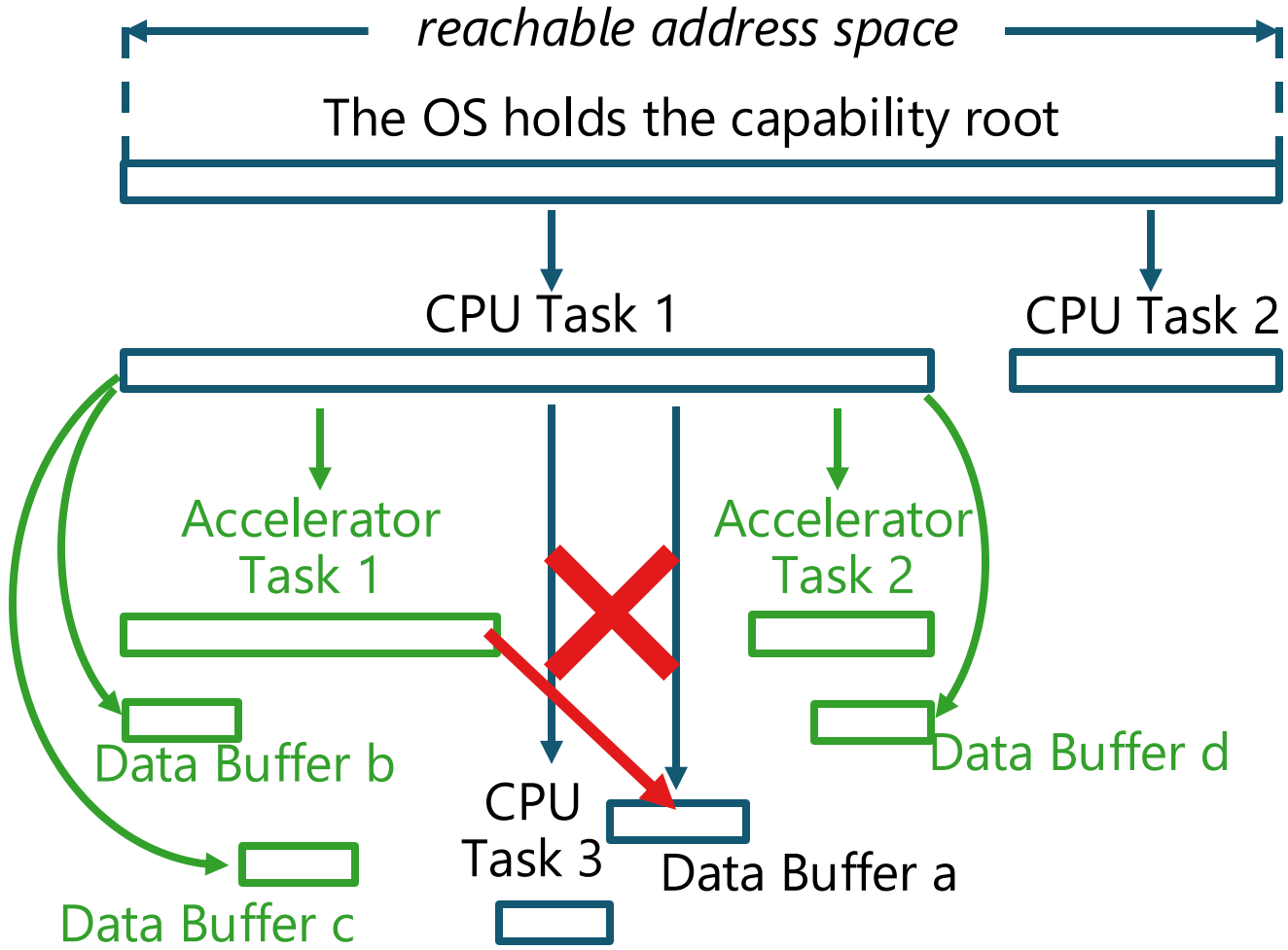
Capability Model



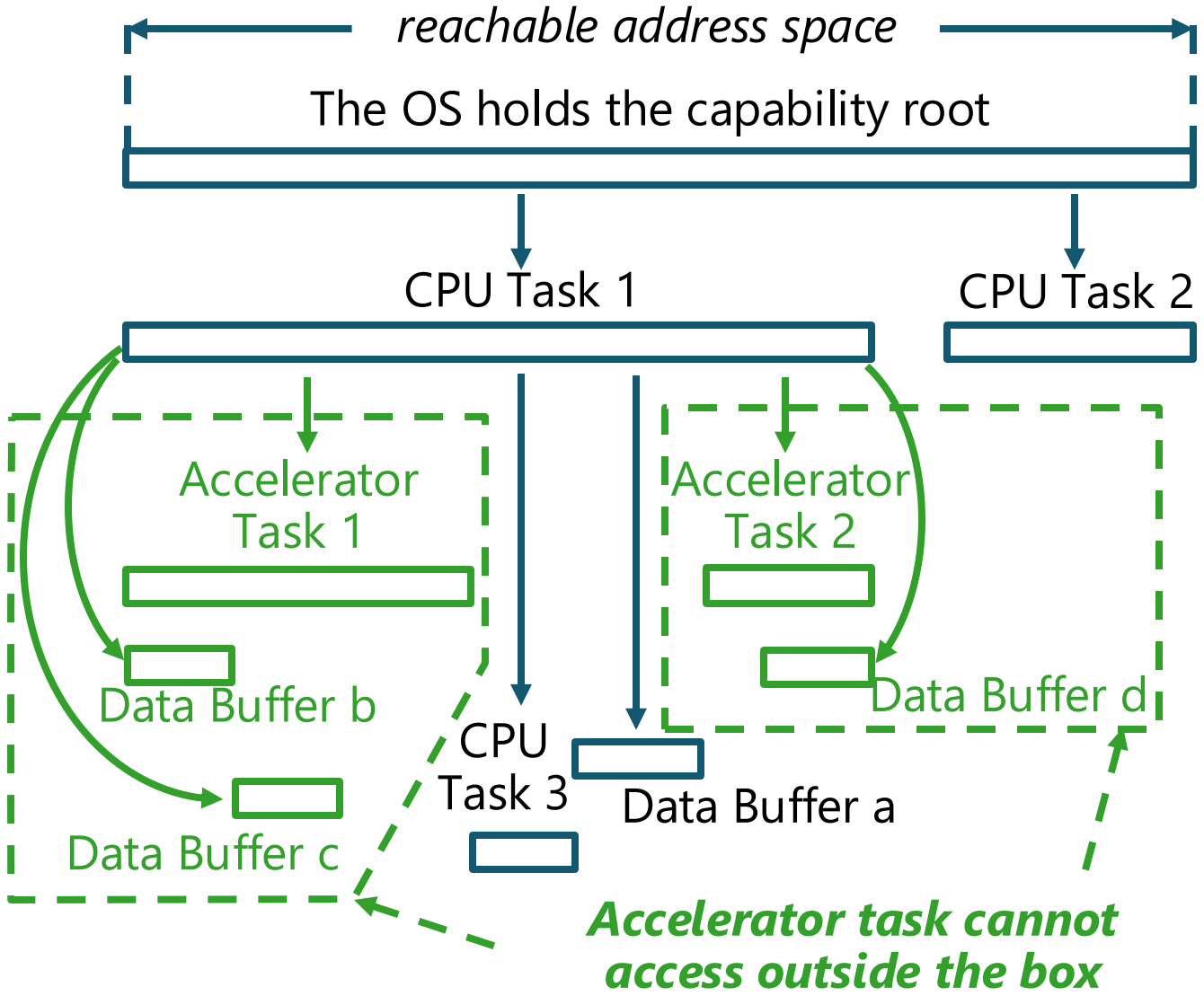
Capability Model



Capability Model

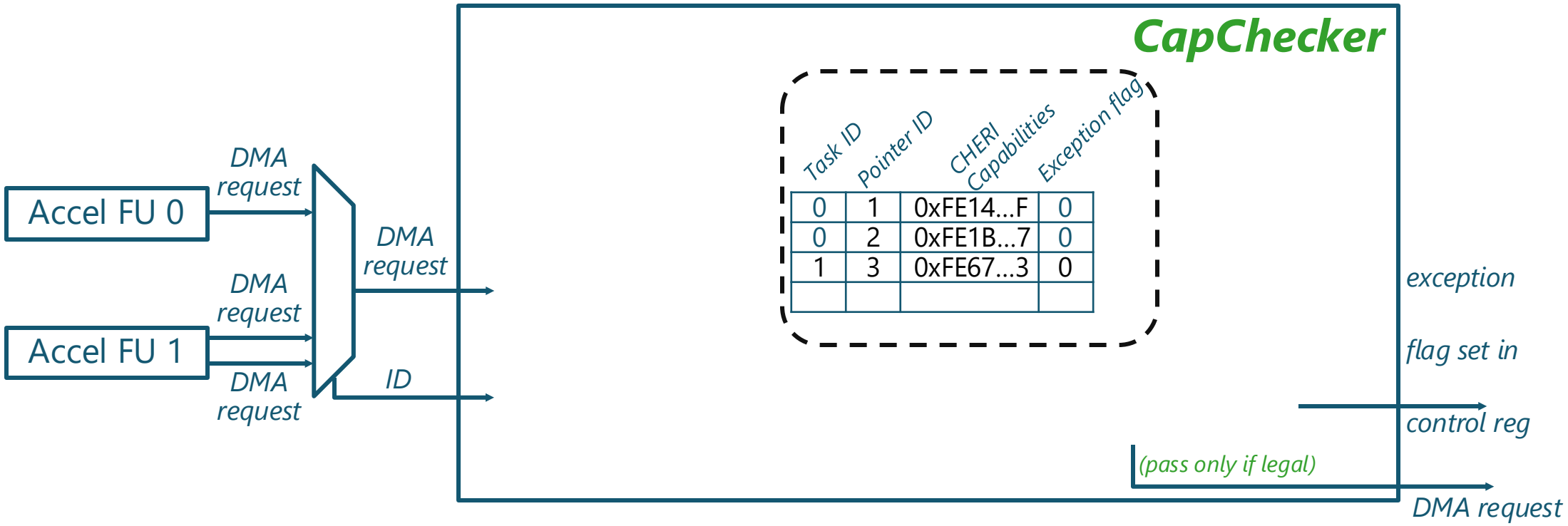


Capability Model

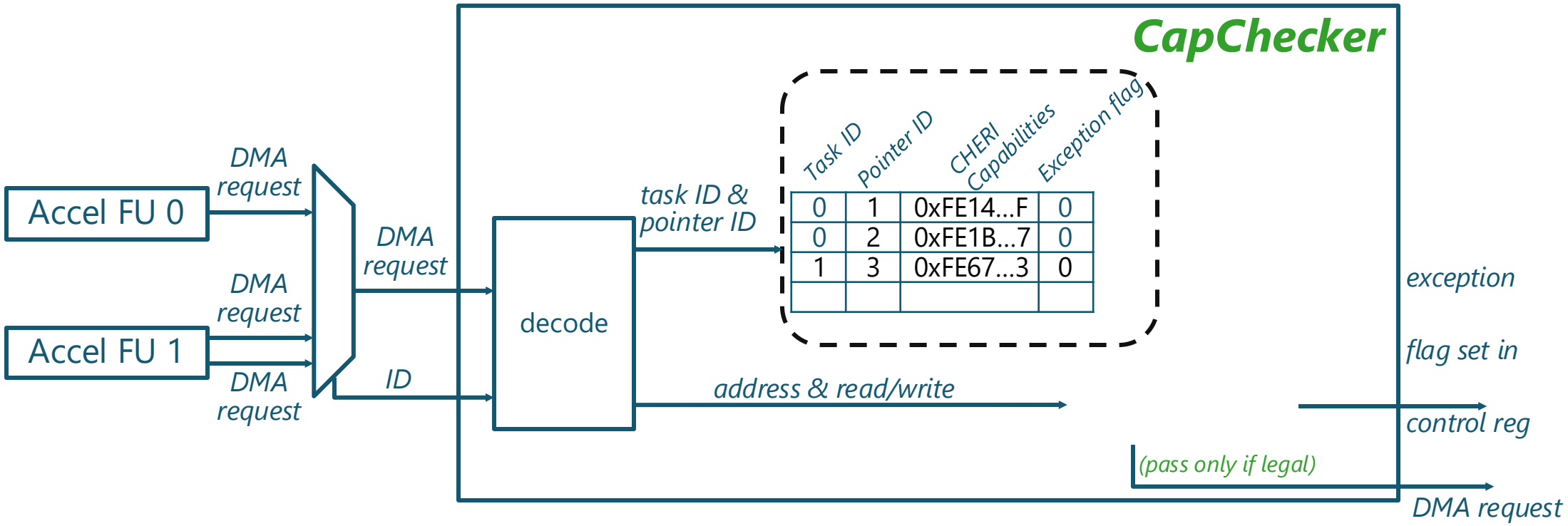


CapChecker Hardware Design

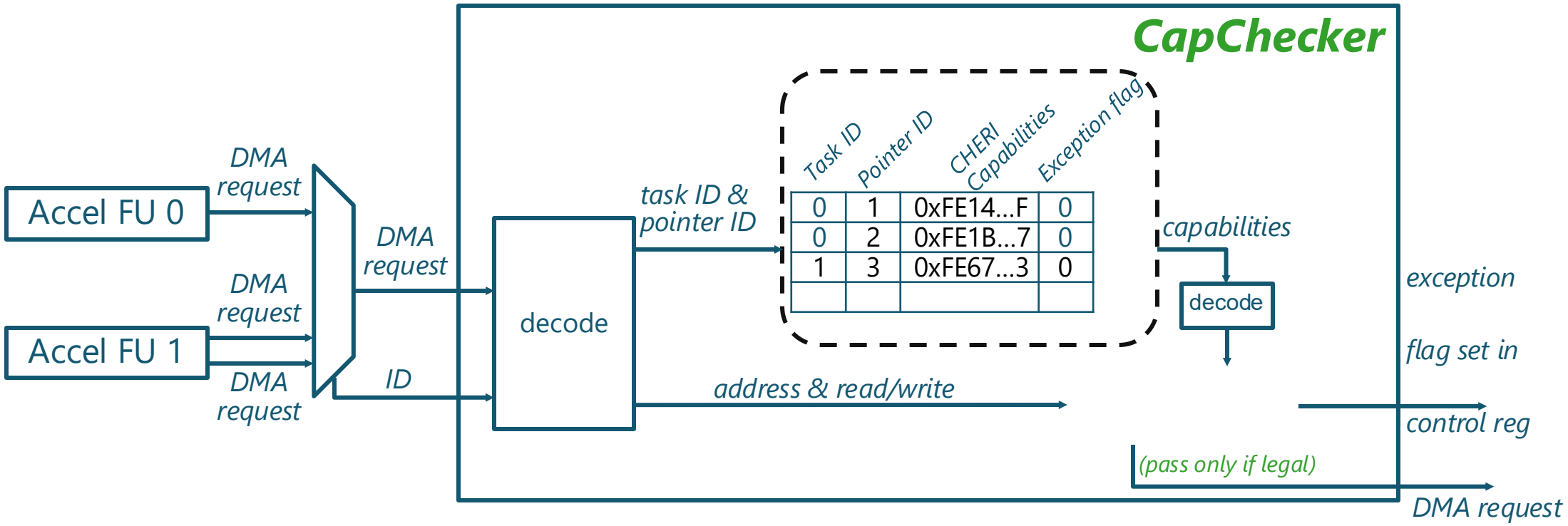
CapChecker Hardware Design



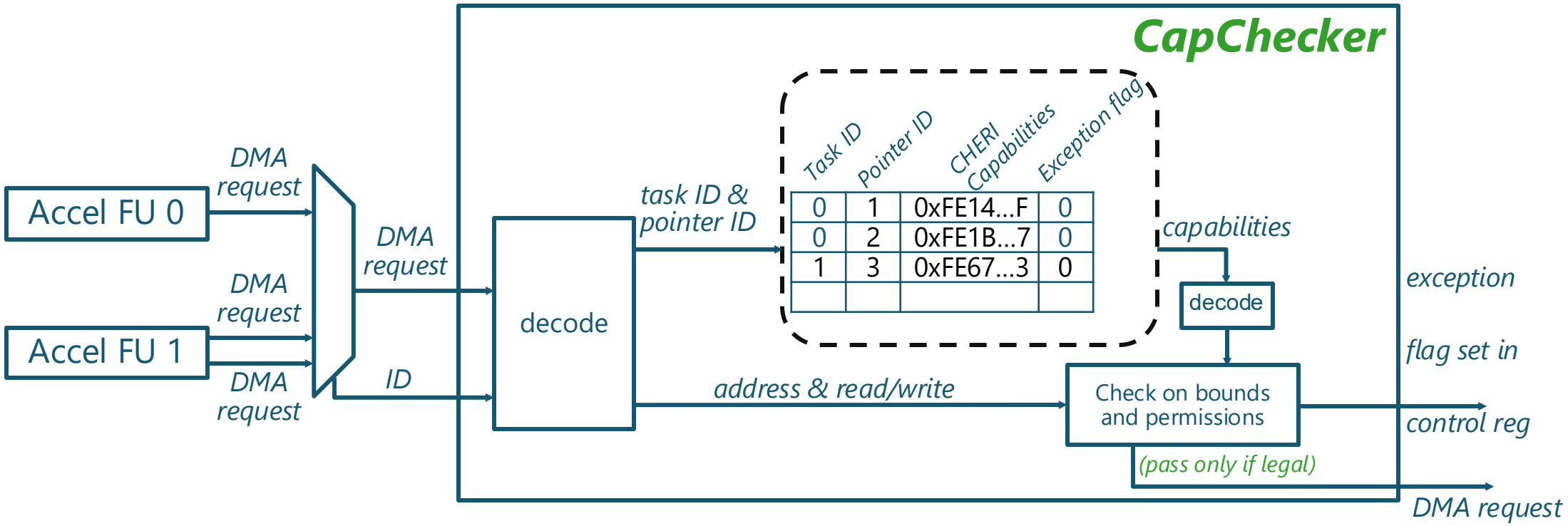
CapChecker Hardware Design



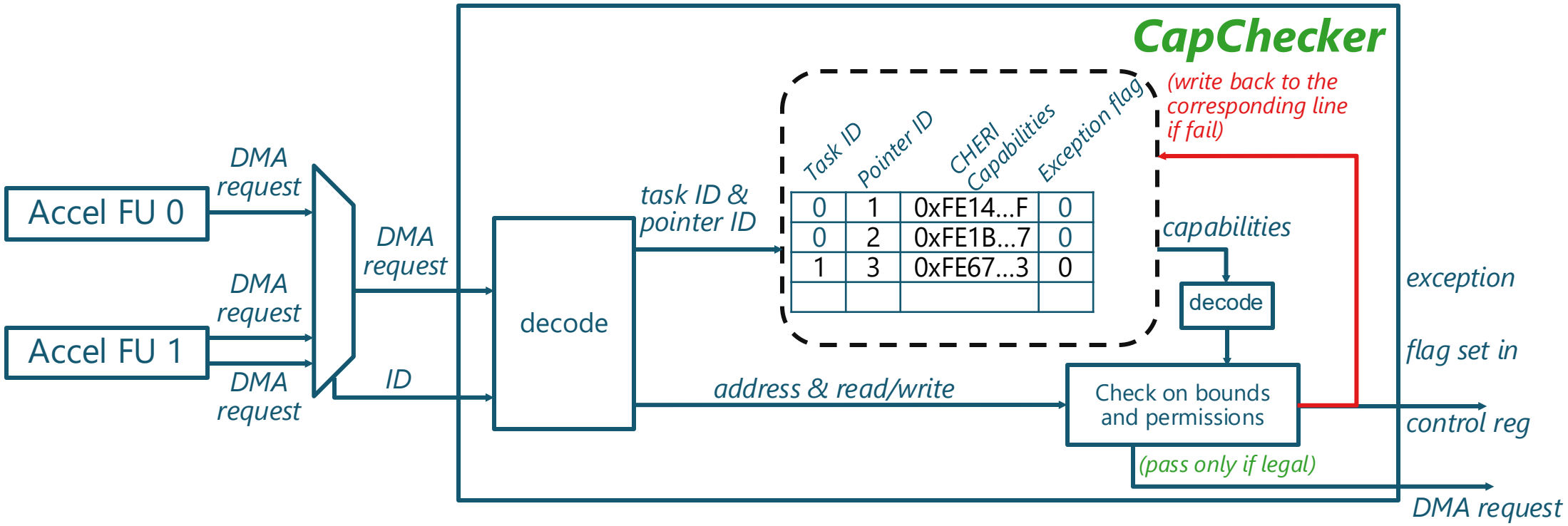
CapChecker Hardware Design



CapChecker Hardware Design

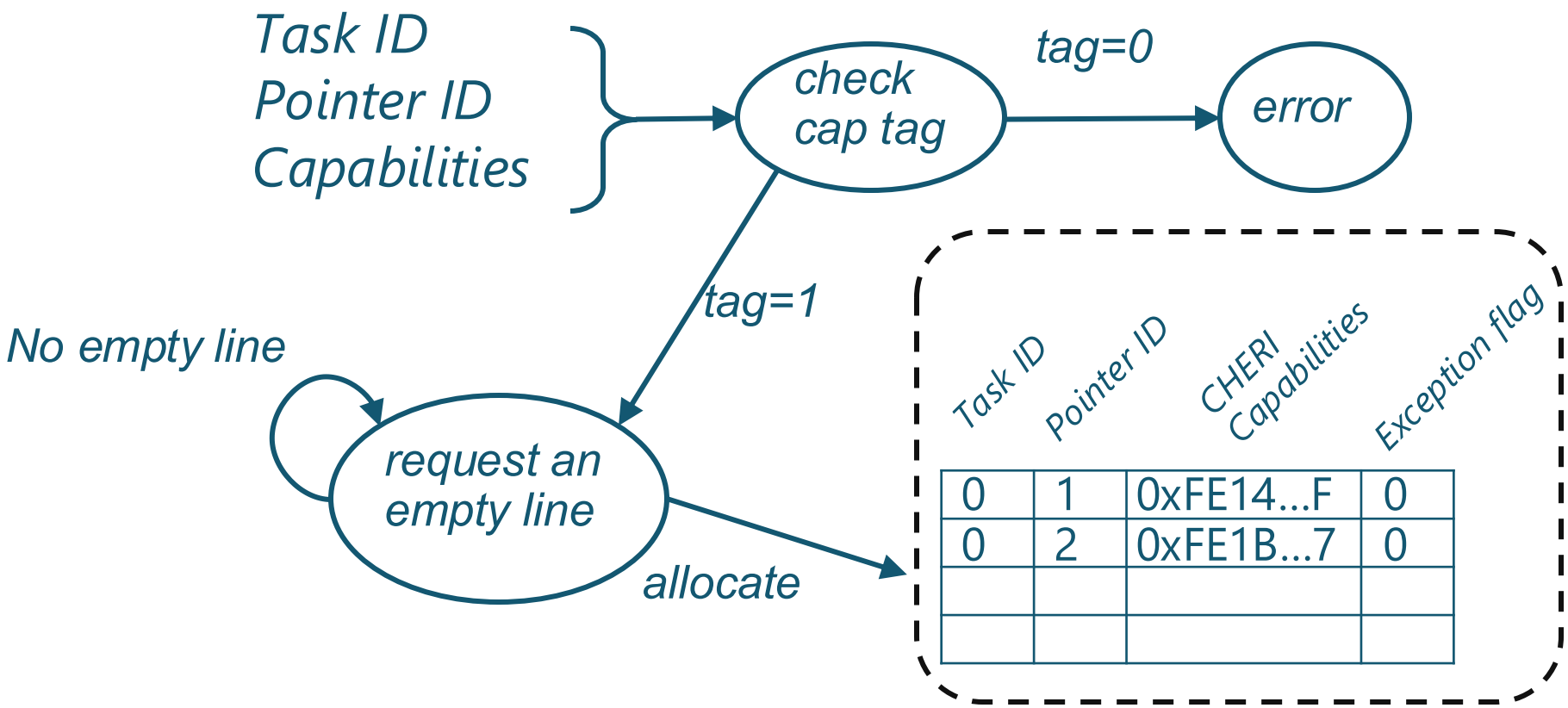


CapChecker Hardware Design



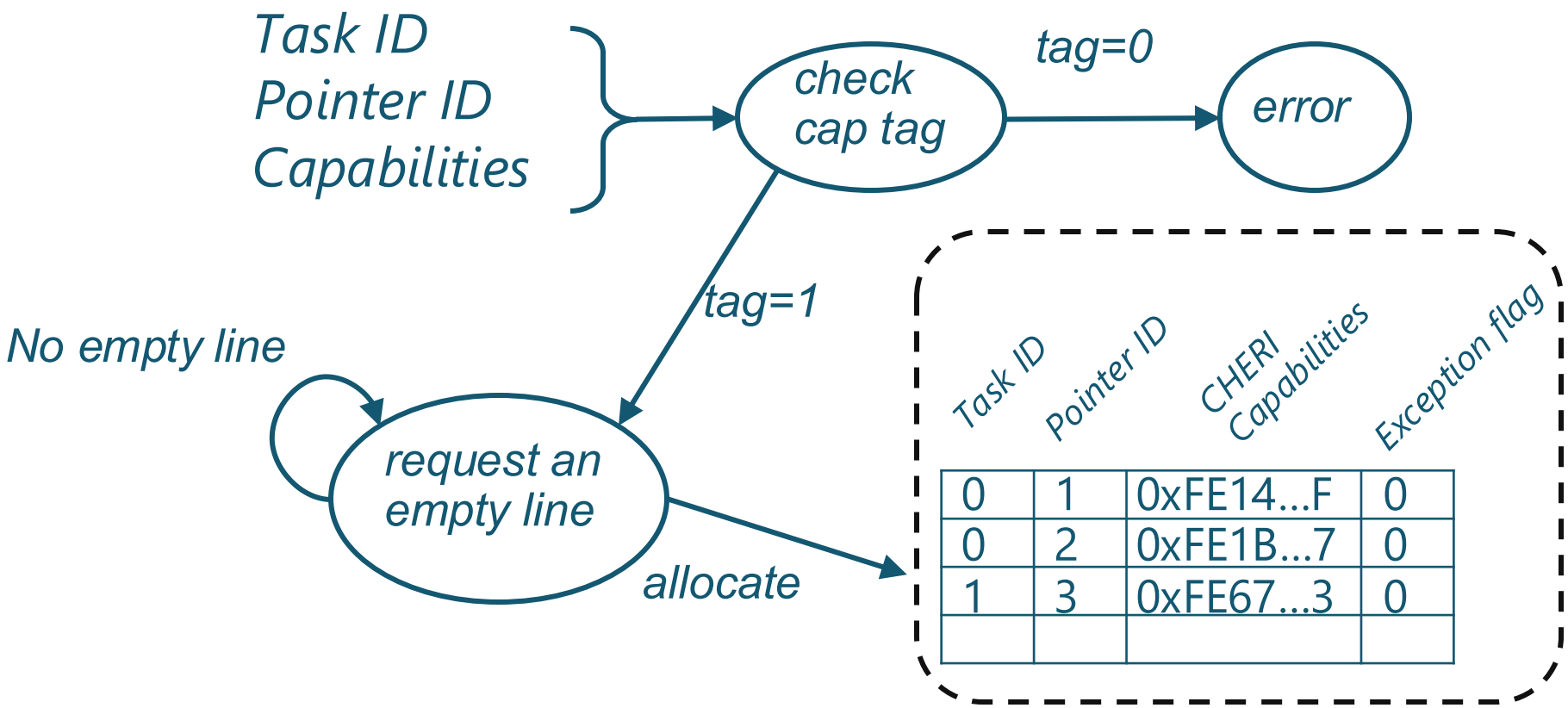
CapChecker Driver

Allocate capabilities



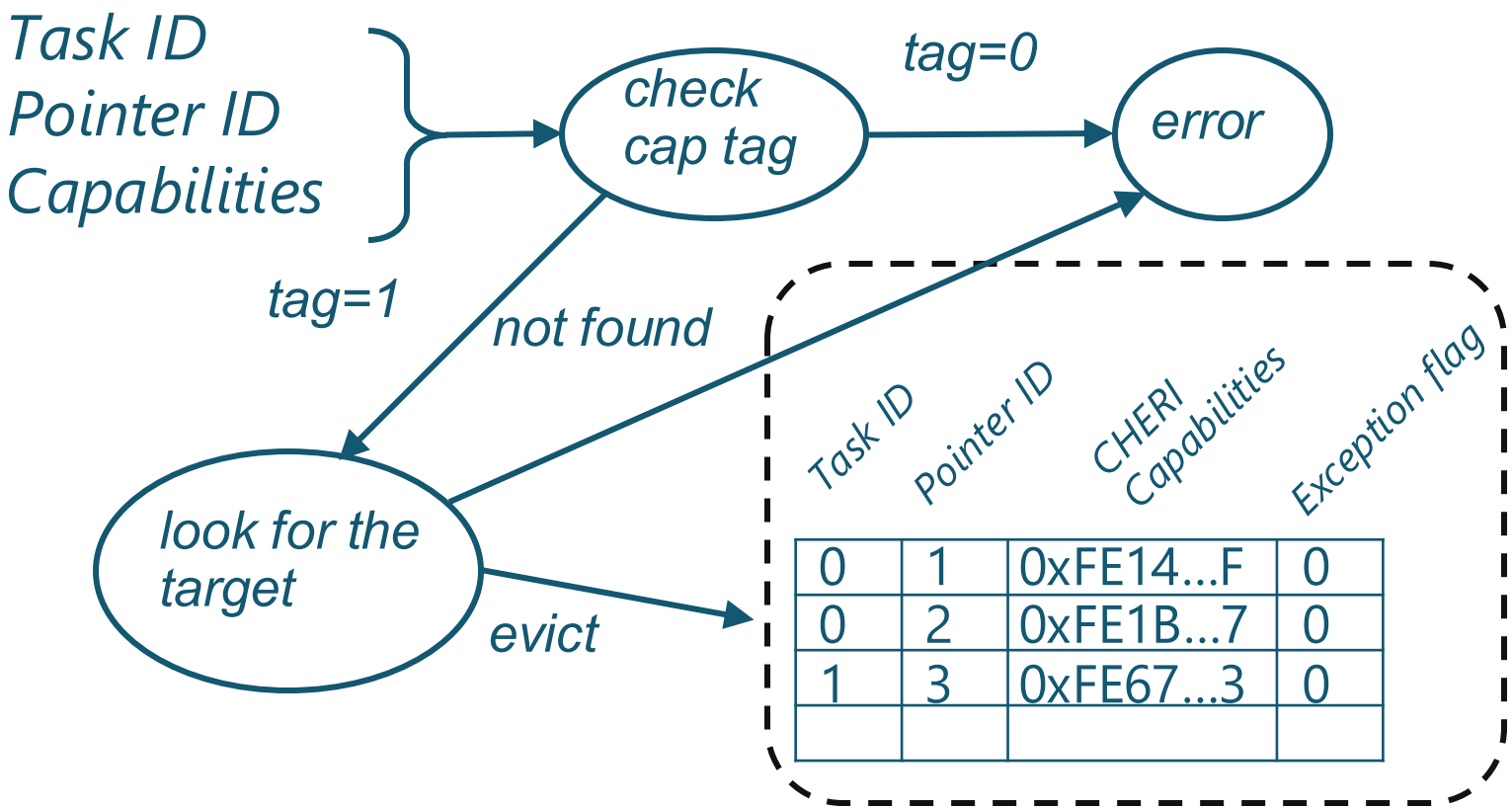
CapChecker Driver

Allocate capabilities



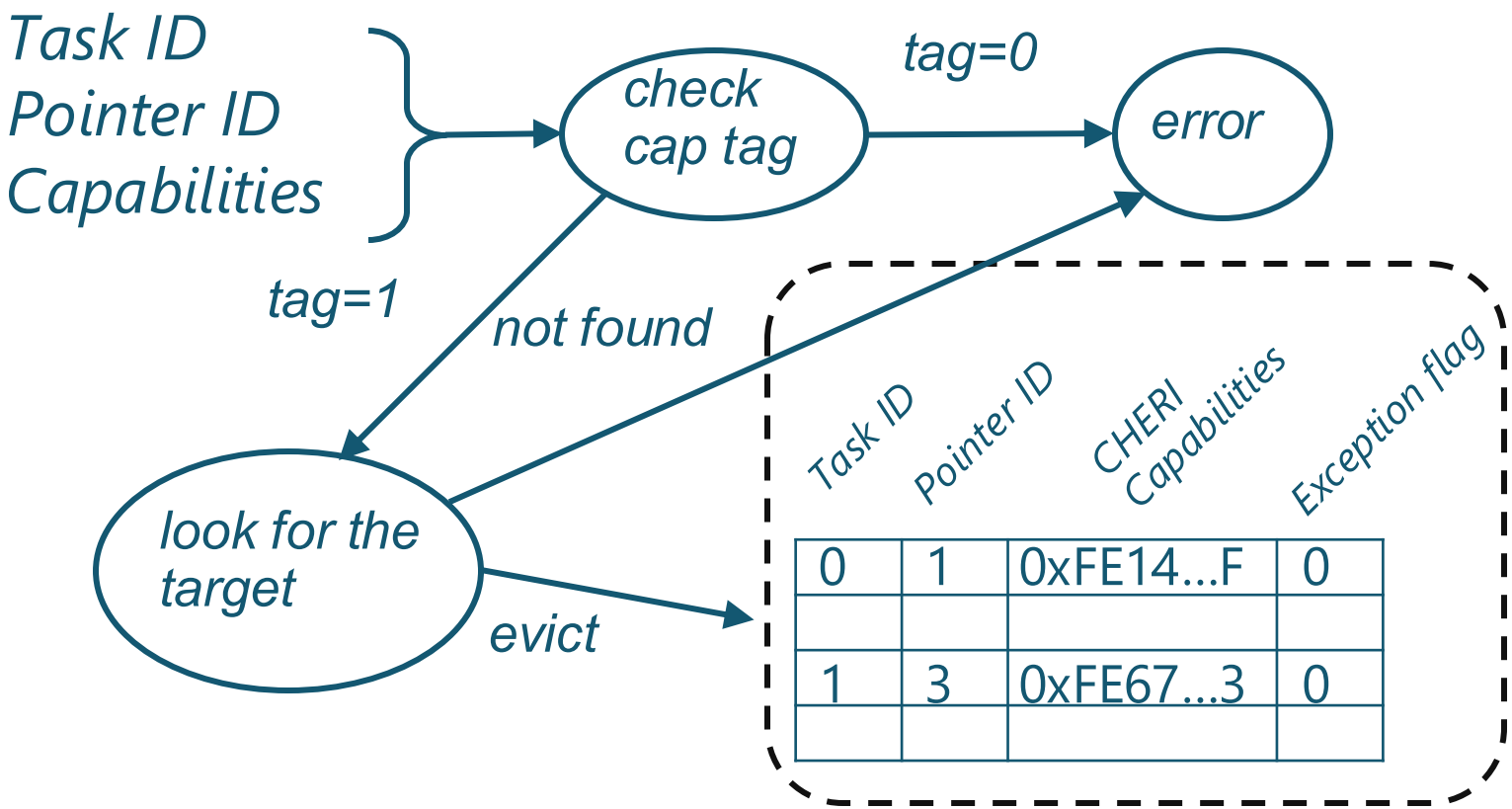
CapChecker Driver

Evict capabilities



CapChecker Driver

Evict capabilities



Experimental Evaluation

- Security
- Overhead

Security: CapChecker has the finest grain

- CWE Classification

- Metric: granularity

ID	Memory Weaknesses	No Method	IOMMU	sNPU	CapChecker
…	Buffer overreads or overwrites				
761	Free of Pointer not at Start of Buffer				
822	Untrusted Pointer Dereference				
823	Untrusted Pointer Offset				

Security: CapChecker has the finest grain

- CWE Classification

- Metric: granularity

ID	Memory Weaknesses	No Method	IOMMU	sNPU	CapChecker
…	Buffer overreads or overwrites	❌	Page	Task	**Object**
761	Free of Pointer not at Start of Buffer	❌	Page	Task	**Object**
822	Untrusted Pointer Dereference				
823	Untrusted Pointer Offset				

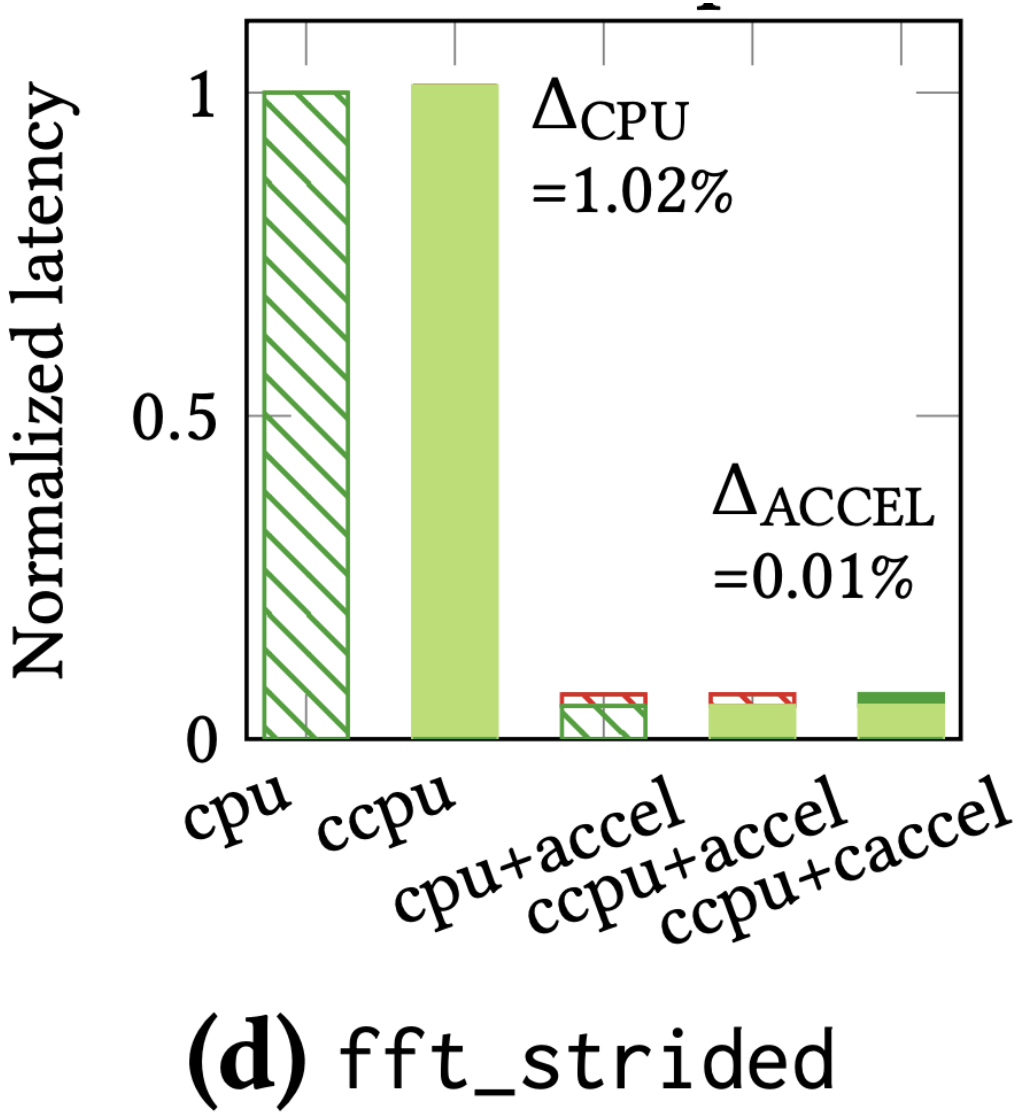
Security: CapChecker has the finest grain

- CWE Classification

- Metric: granularity

ID	Memory Weaknesses	No Method	IOMMU	sNPU	CapChecker
…	Buffer overreads or overwrites	❌	Page	Task	**Object**
761	Free of Pointer not at Start of Buffer	❌	Page	Task	**Object**
822	Untrusted Pointer Dereference	❌	❌	Task	**Object**
823	Untrusted Pointer Offset	❌	❌	Task	**Object**

Performance: CapChecker has <5% overhead



(d) `fft_strided`

CHERI-unaware CPU
CHERI-unaware accelerator
CHERI-aware CPU
CHERI-aware accelerator

CHERI overhead:

Accelerators < CPU

Conclusion and Future Work

What we did

- Compartmentalization for heterogenous systems
- CapChecker presents fine-grained hardware protection
- CapChecker is affordable and scalable

What we plan to do

- Dynamic memory managements
- Micro-architecture exploration

CHERI

THANK YOU

Contact contact@cheri-alliance.org

Web www.cheri-alliance.org

Link to the paper:

06 April 2026