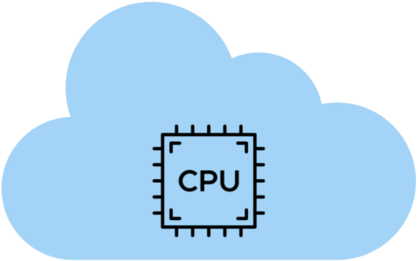
Acai: Protecting Accelerator Execution with Arm Confidential Computing Architecture

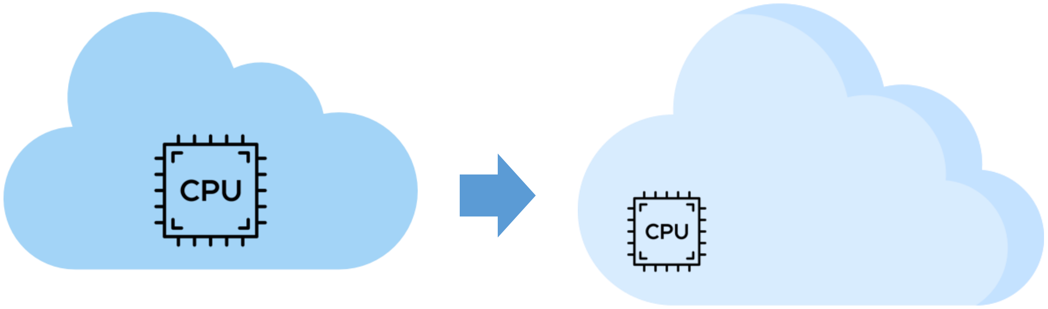
Supraja Sridhara, Andrin Bertschi, Benedict Schlüter, Mark Kuhne, Fabio Aliberti, and Shweta Shinde

ETH Zurich

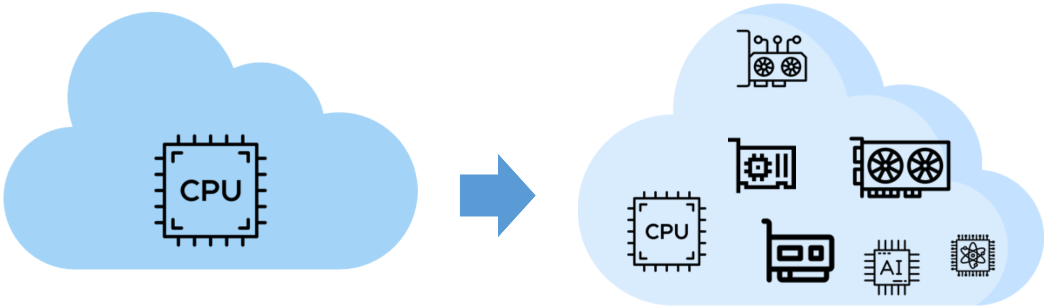
Cloud & Accelerators



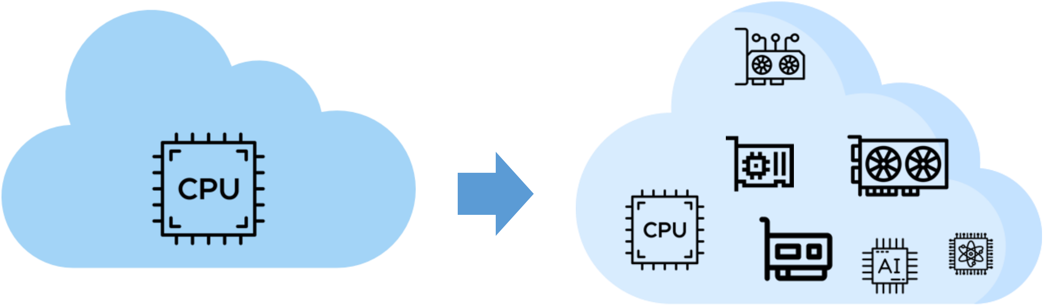
Cloud & Accelerators



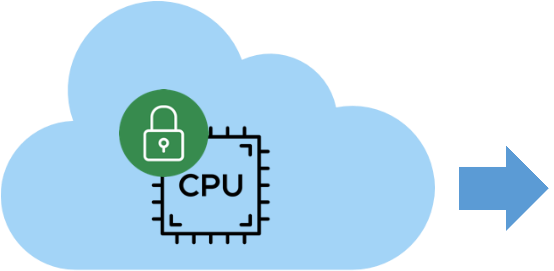
Cloud & Accelerators



Cloud & Accelerators

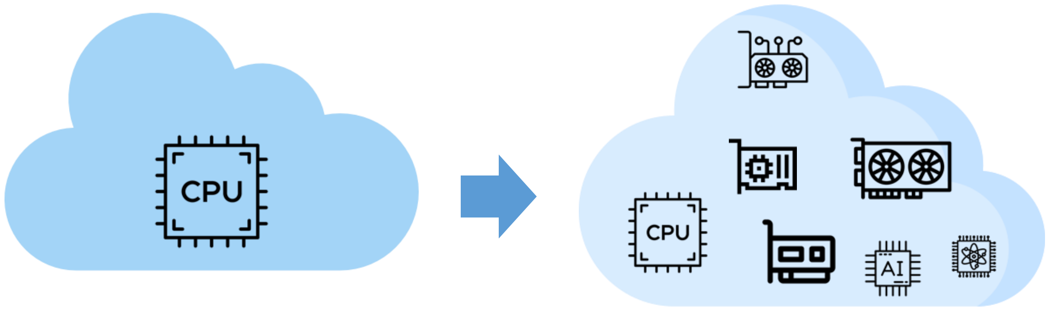


Confidential Computing with TEEs

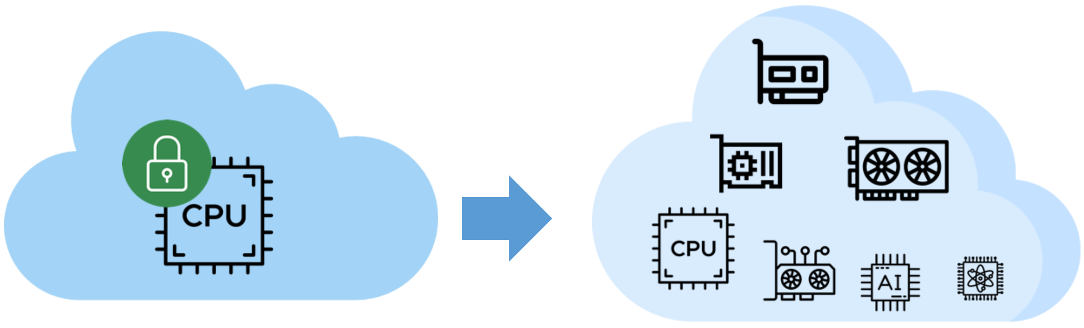


Intel SGX
Intel TDX
AMD SEV-SNP
Arm CCA

Cloud & Accelerators

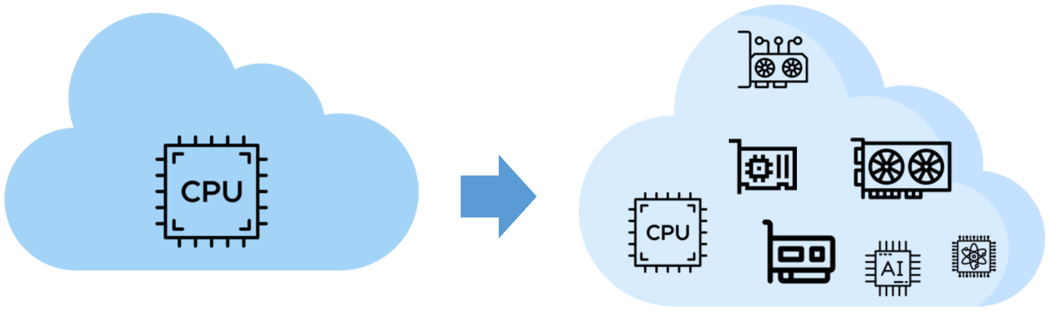


Confidential Computing with TEEs

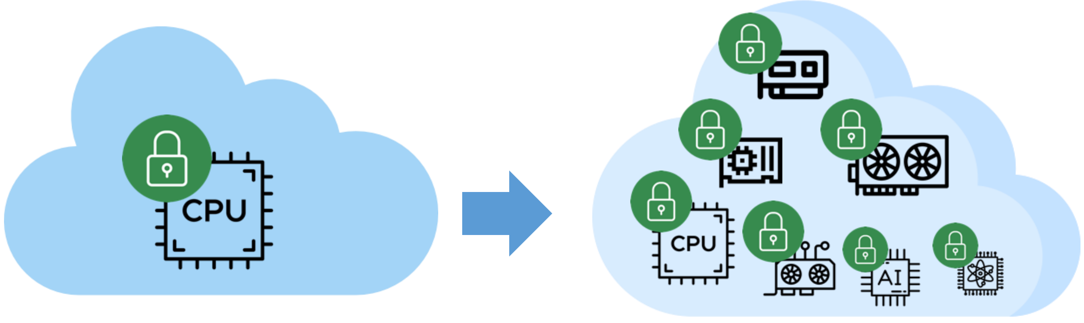


Intel SGX
Intel TDX
AMD SEV-SNP
Arm CCA

Cloud & Accelerators

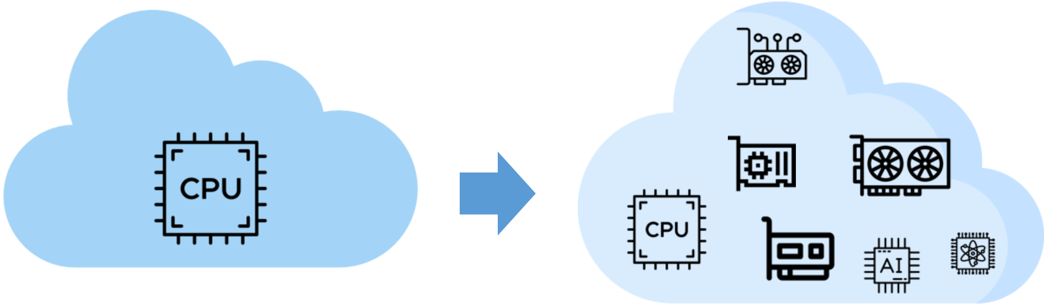


Confidential Computing with TEEs

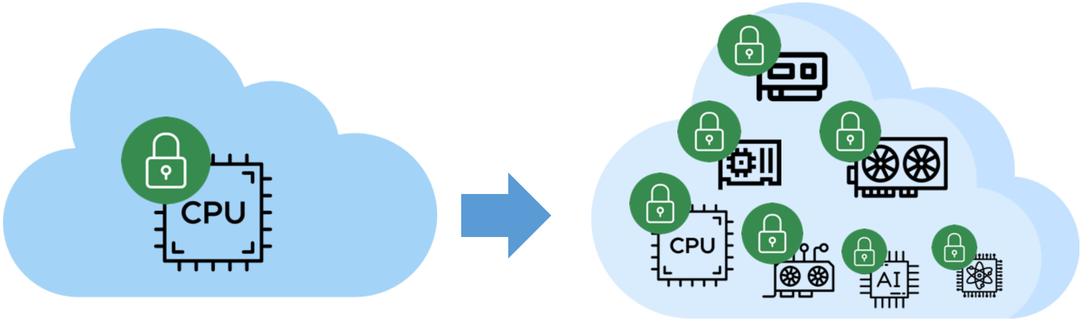


Intel SGX
Intel TDX
AMD SEV-SNP
Arm CCA

Cloud & Accelerators



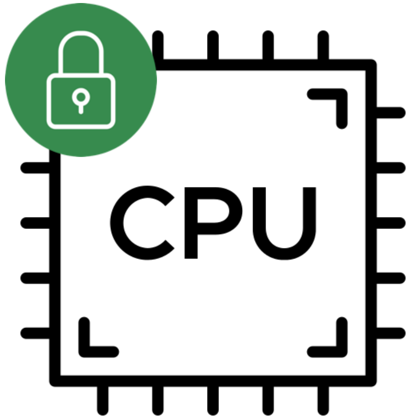
Confidential Computing with TEEs

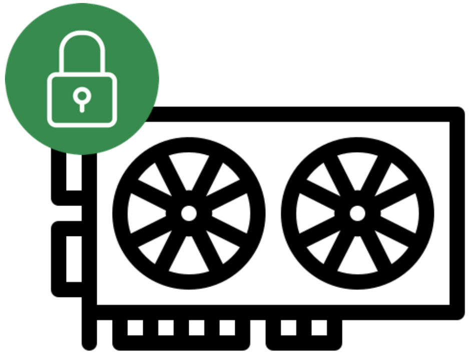


Intel SGX
Intel TDX
AMD SEV-SNP
Arm CCA

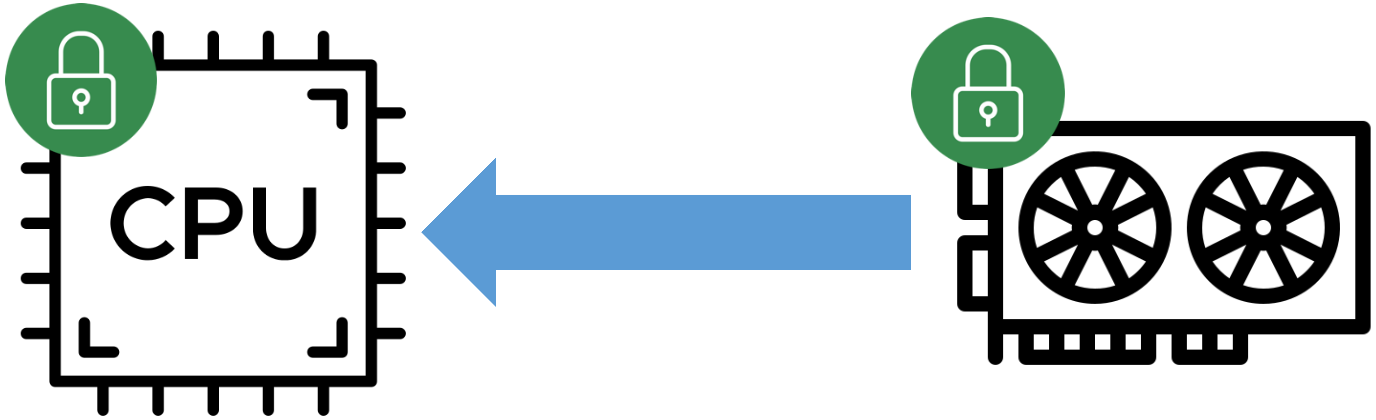
Nvidia H100

Securely Compose CC CPU + Accelerator

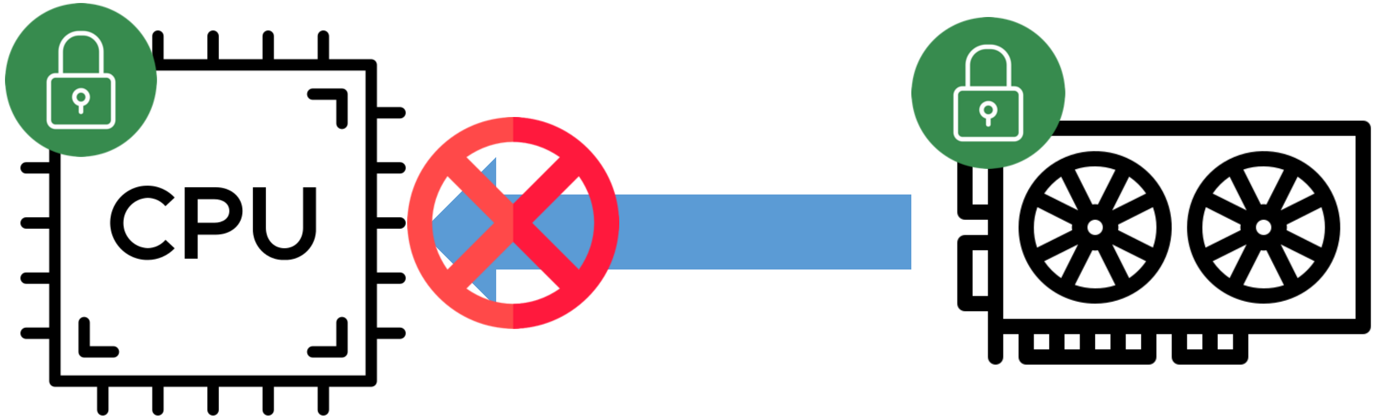




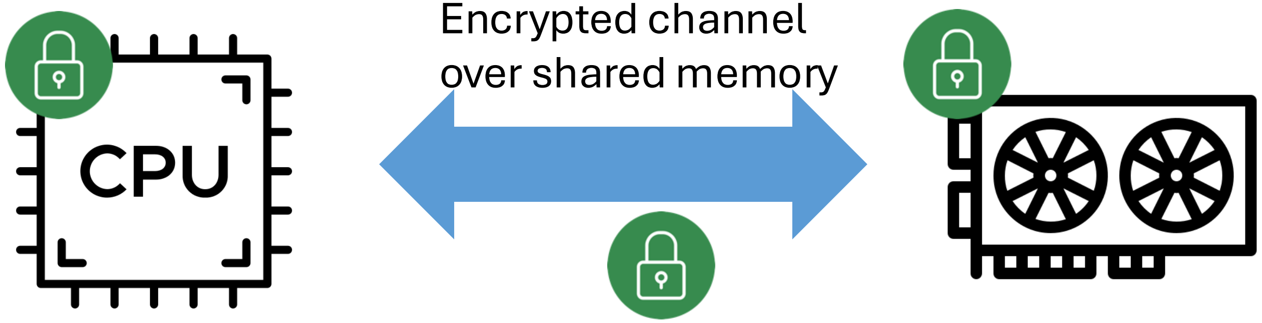
Securely Compose CC CPU + Accelerator



Securely Compose CC CPU + Accelerator



Bounce Buffers



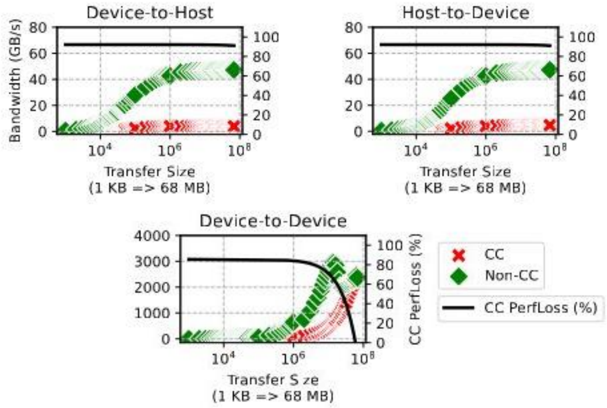
Bounce Buffers

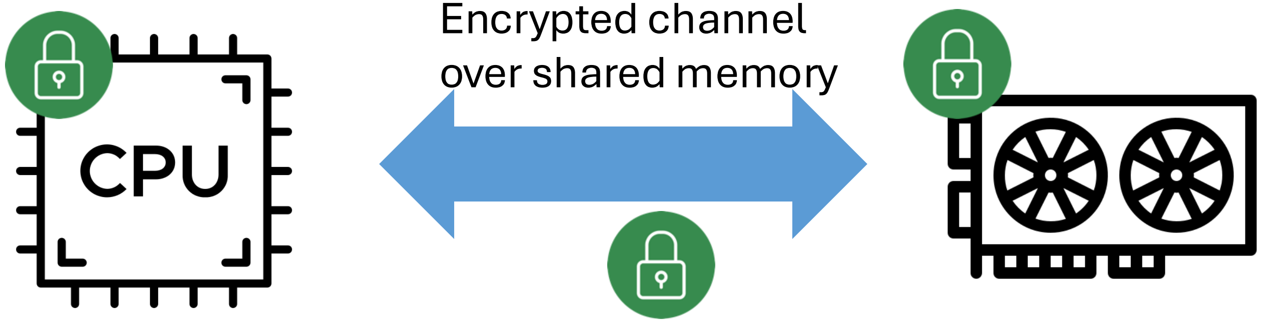
Data from Nvidia [1]

Nvidia H100 : ~4GBps

PCIe 6 : upto 256 GBps

Data from TDX+H100 benchmarking [2]



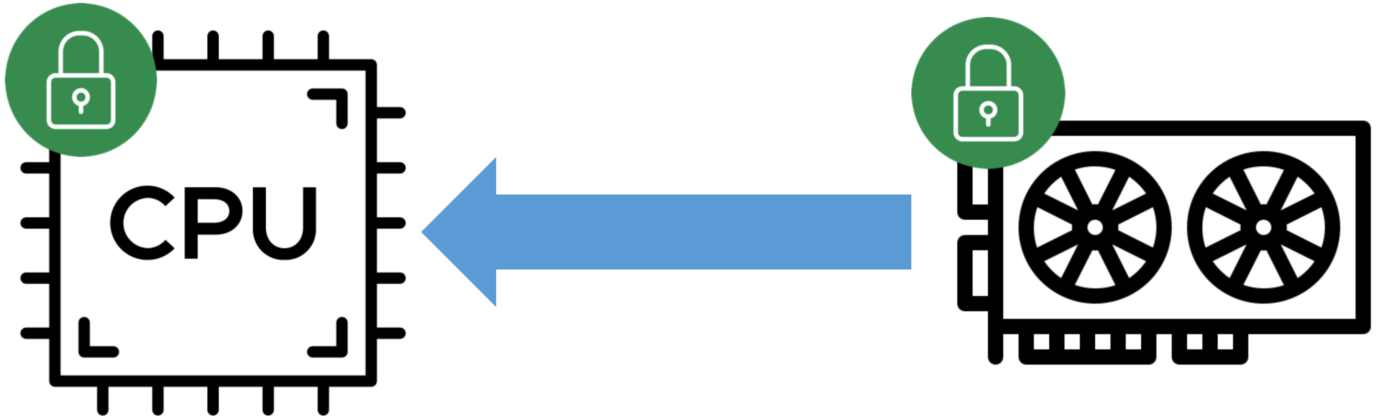


Encrypted channel over shared memory

[1] https://developer.nvidia.com/blog/confidential-computing-on-h100-gpus-for-secure-and-trustworthy-ai/

[2] https://research.ibm.com/publications/securing-ai-inference-in-the-cloud-is-cpu-gpu-confidential-computing-ready

Allow protected memory access



Acai

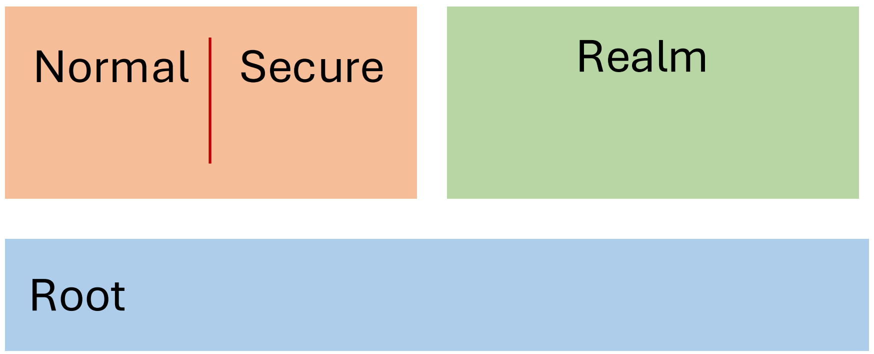
FIRST SYSTEM FOR PCIE DEVICES WITH CCA

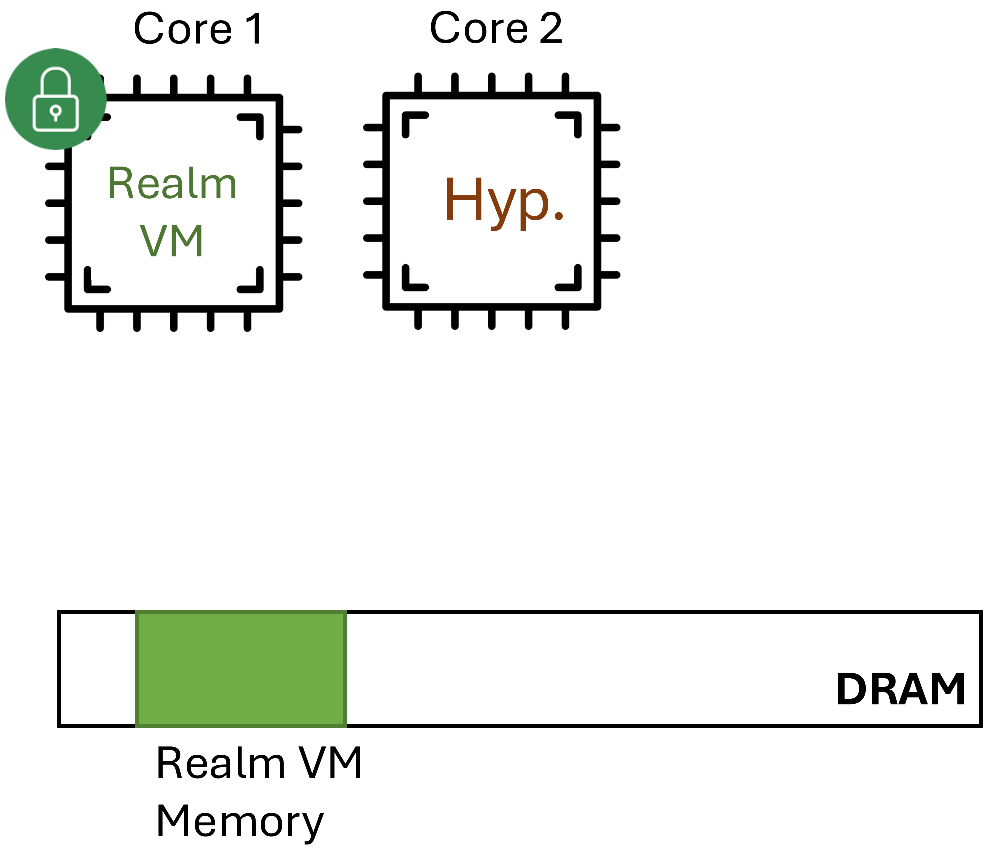
EXTEND CCA’S INVARIANTS FOR SECURITY

BUILD A CONCRETE DESIGN

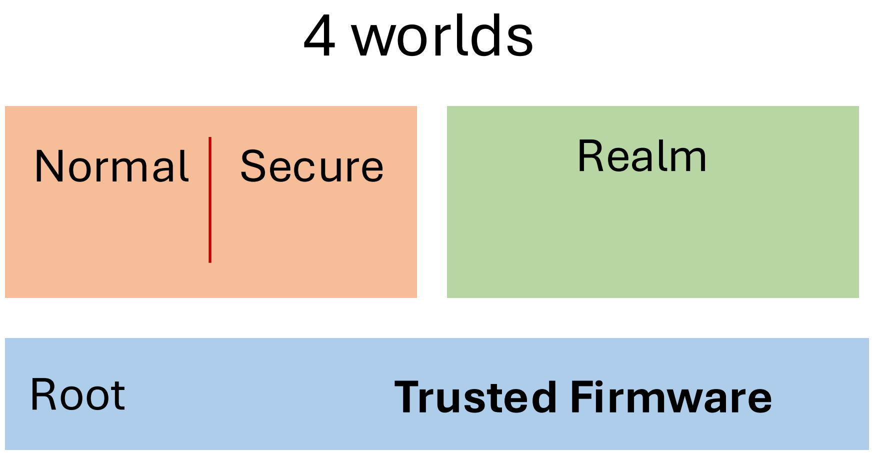
Background: Arm CCA

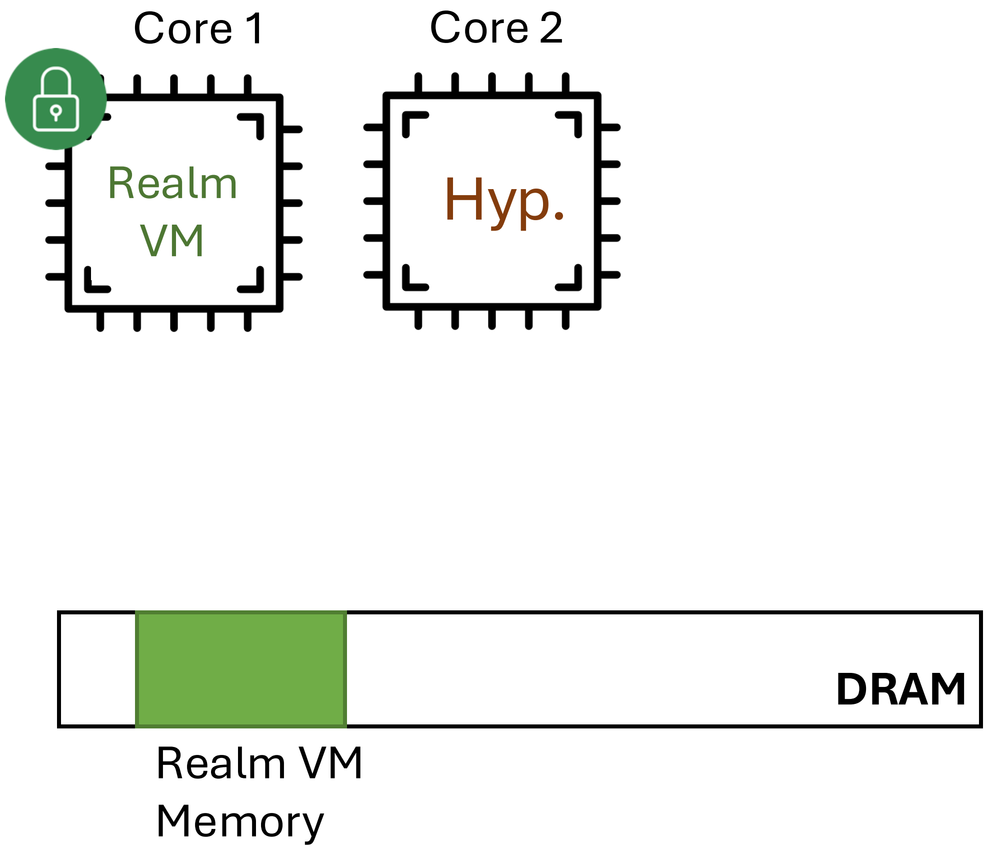
4 worlds



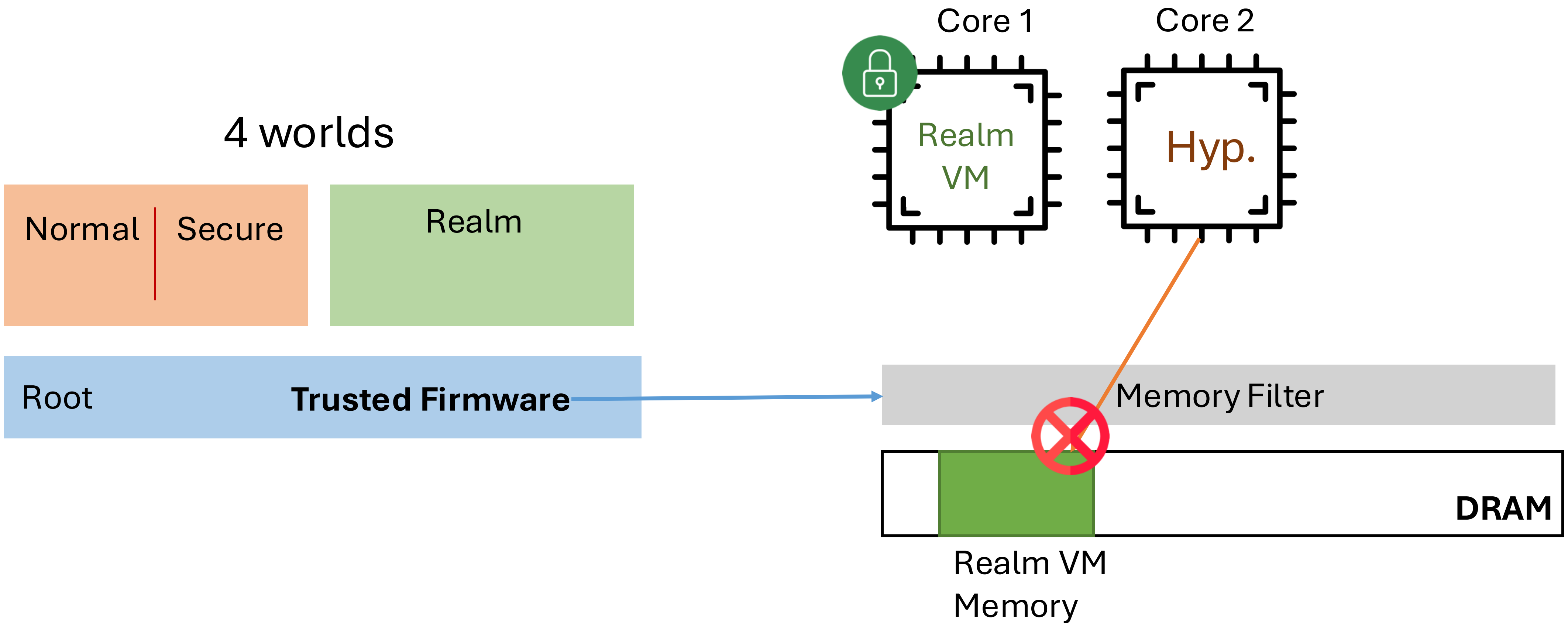


Background: Arm CCA





Background: Arm CCA



4 worlds

Normal

Secure

Realm

Root

Trusted Firmware

Core 1

Core 2

Realm VM

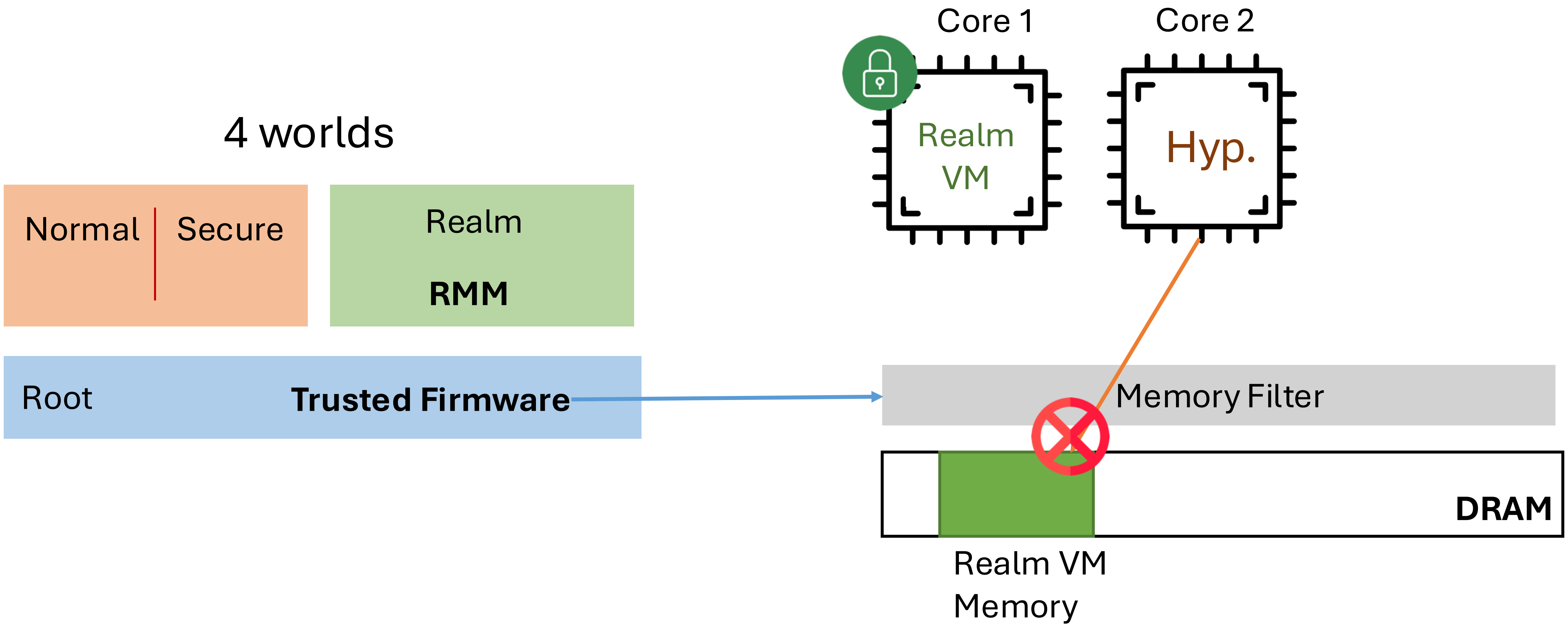
Hyp.

Memory Filter

DRAM

Realm VM Memory

Background: Arm CCA



4 worlds

Normal | Secure

Realm

RMM

Root

Trusted Firmware

Core 1

Core 2

Realm VM

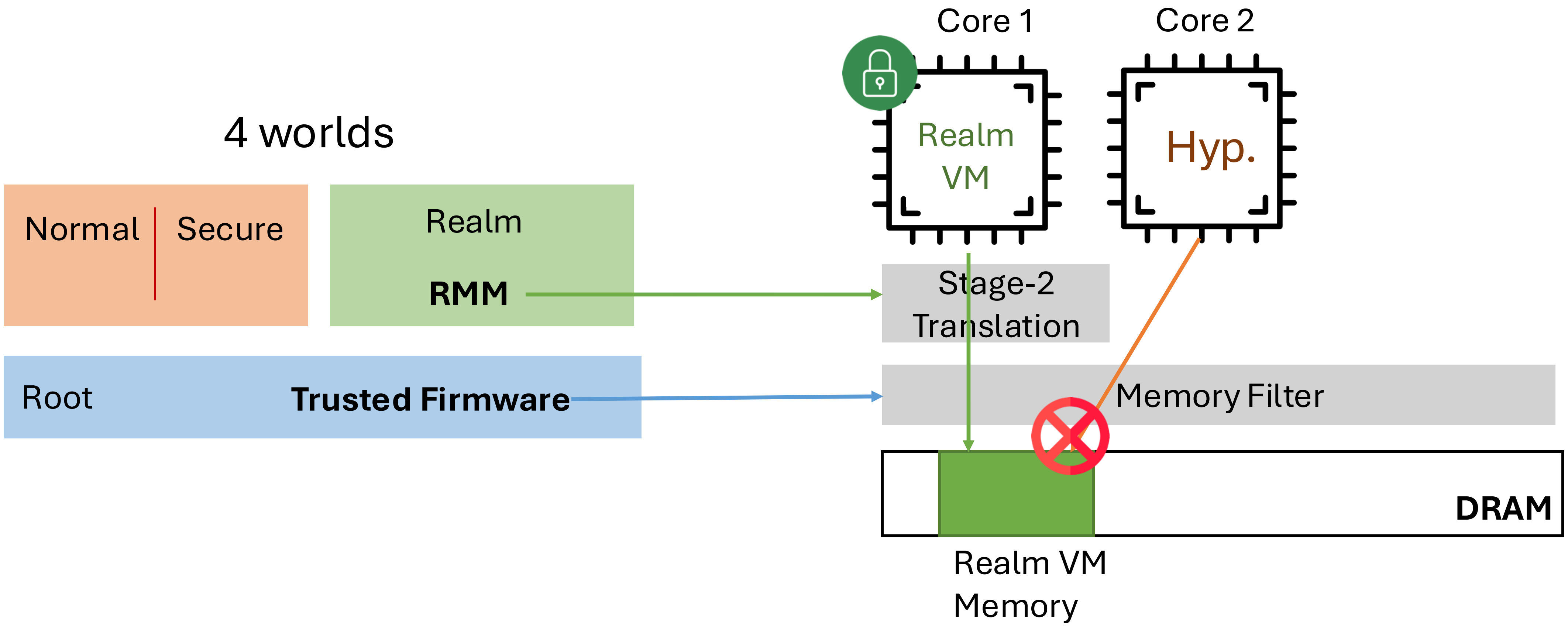
Hyp.

Memory Filter

DRAM

Realm VM Memory

Background: Arm CCA



4 worlds

Normal

Secure

Realm

RMM

Root

Trusted Firmware

Core 1

Core 2

Realm VM

Hyp.

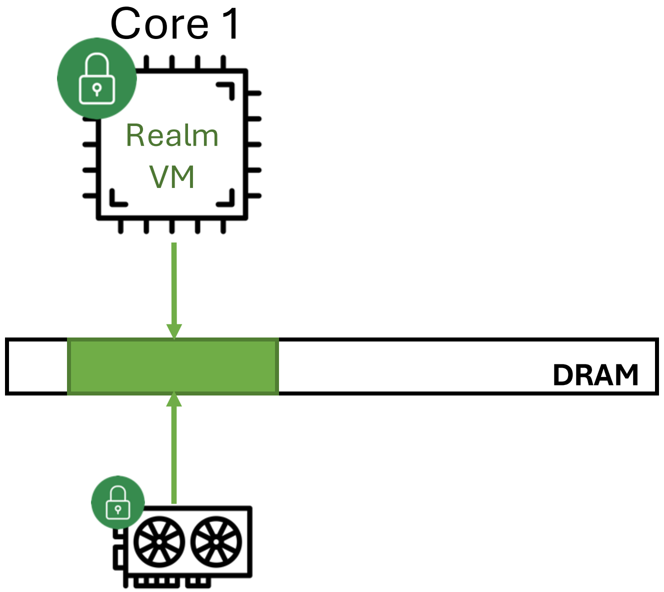
Stage-2 Translation

Memory Filter

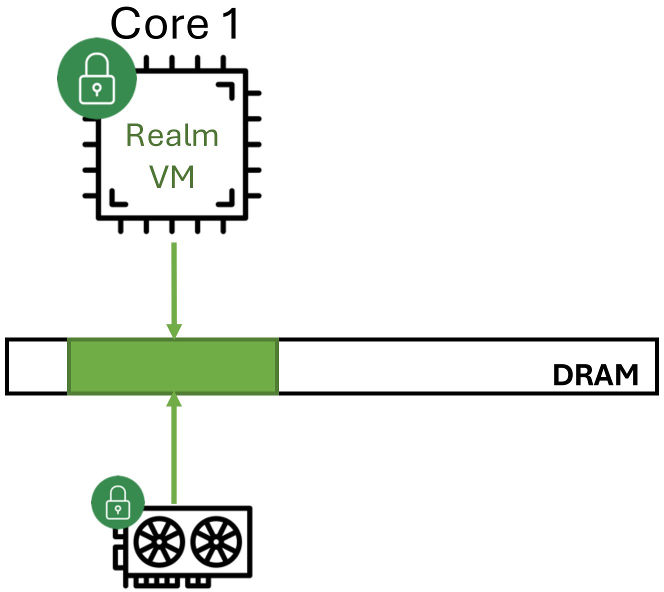
DRAM

Realm VM Memory

Attaching devices to CVMs



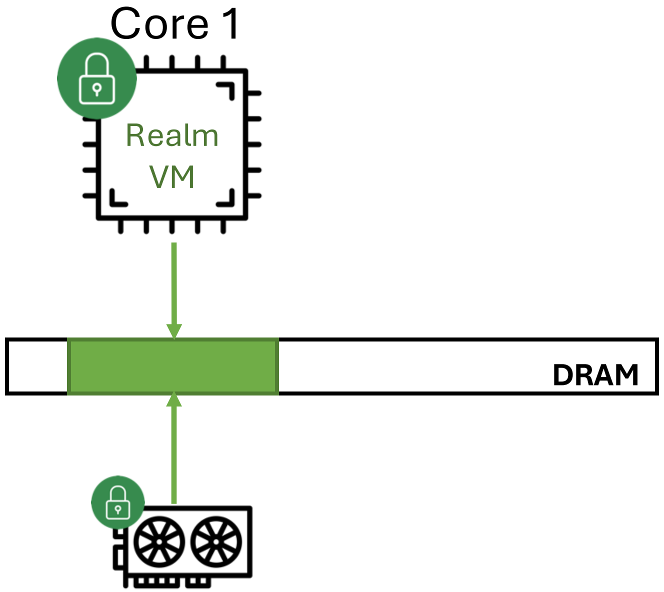
Attaching devices to CVMs



Time-sharing

Time-slice the device between different Realm VMs

Attaching devices to CVMs



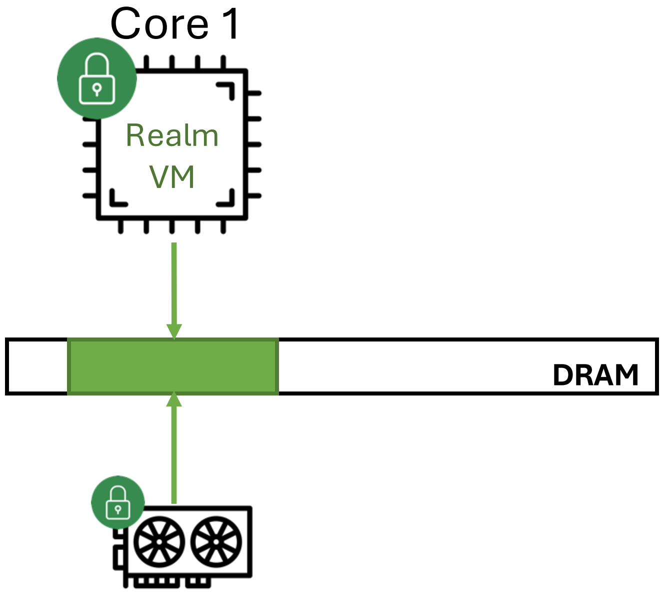
Time-sharing

Time-slice the device between different Realm VMs

Hotplugging

Attach and detach during Realm VM lifecycle

Attaching devices to CVMs



Time-sharing

Time-slice the device between different Realm VMs

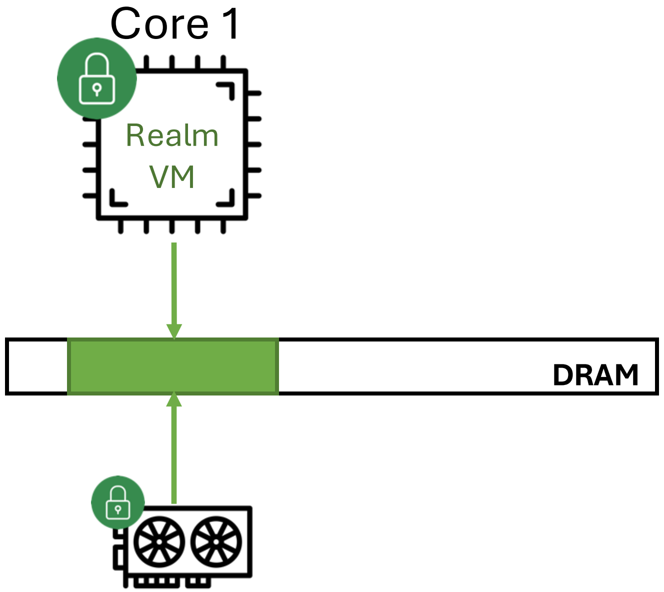
Hotplugging

Attach and detach during Realm VM lifecycle

Multi-tenancy

Share a device spatially between different Realm VMs

Attaching devices to CVMs



Time-sharing

Time-slice the device between different Realm VMs

Hotplugging

Attach and detach during Realm VM lifecycle

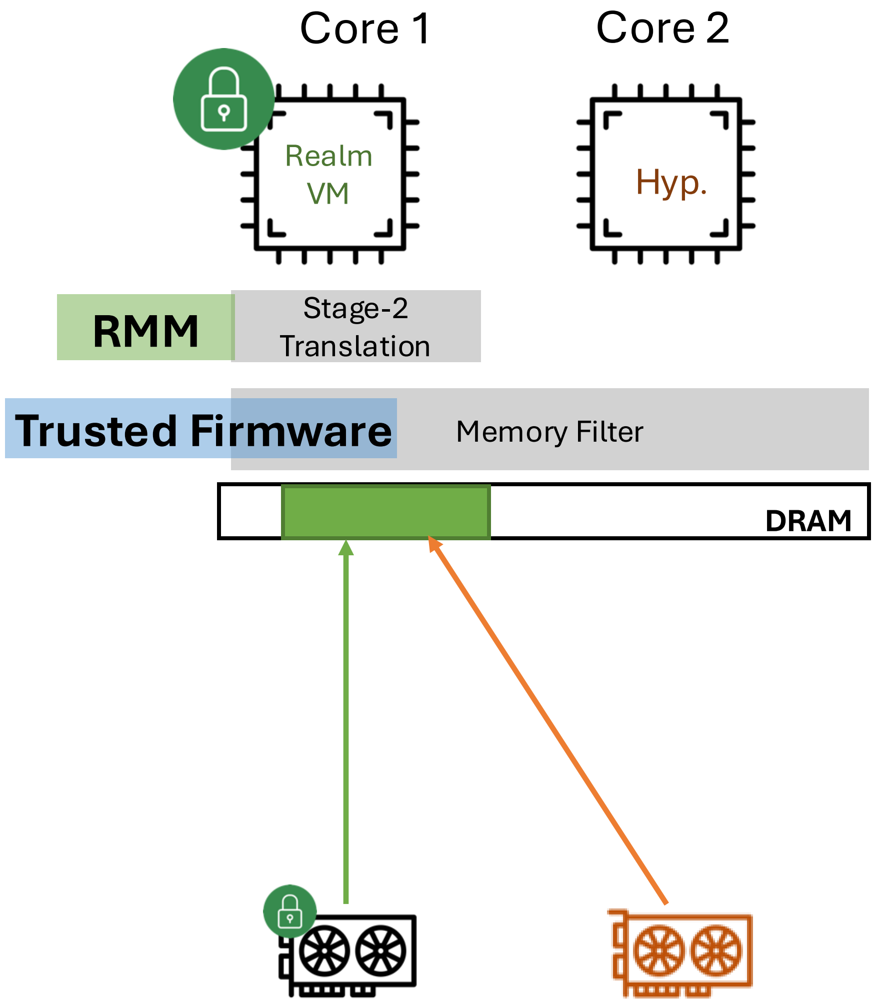
Multi-tenancy

Share a device spatially between different Realm VMs

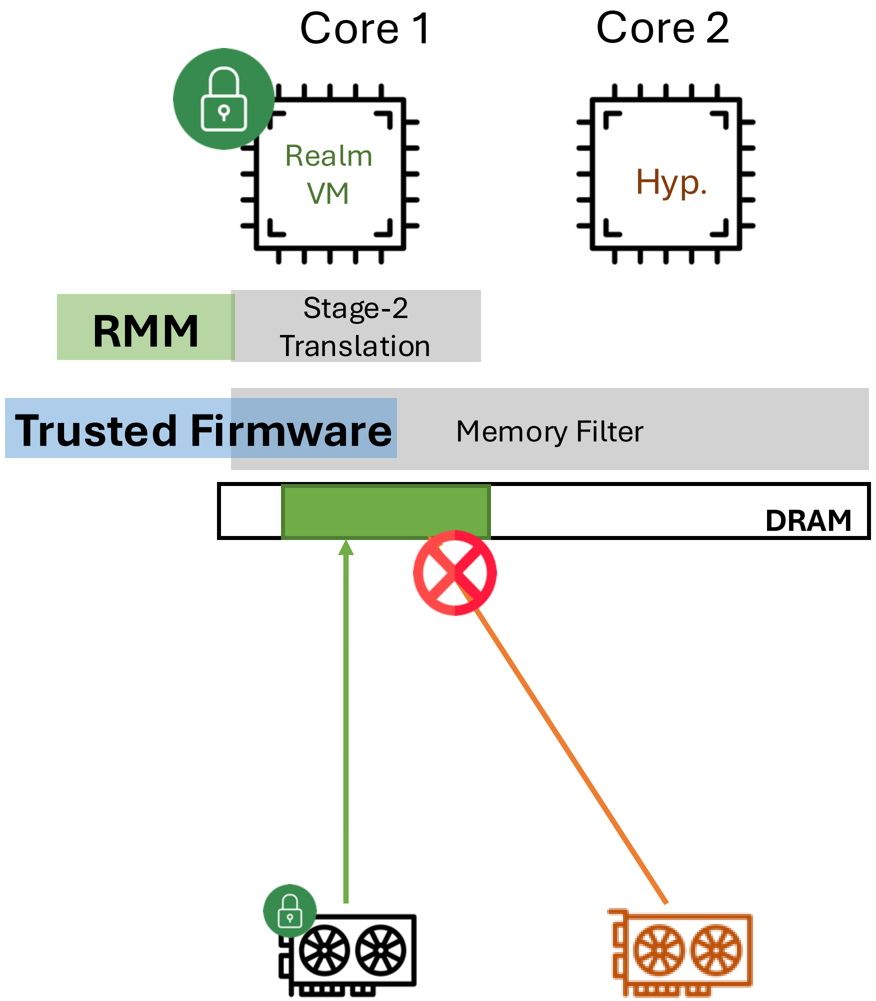
Map to one VM

Attach device to one VM throughout its lifecycle

Isolate device accesses



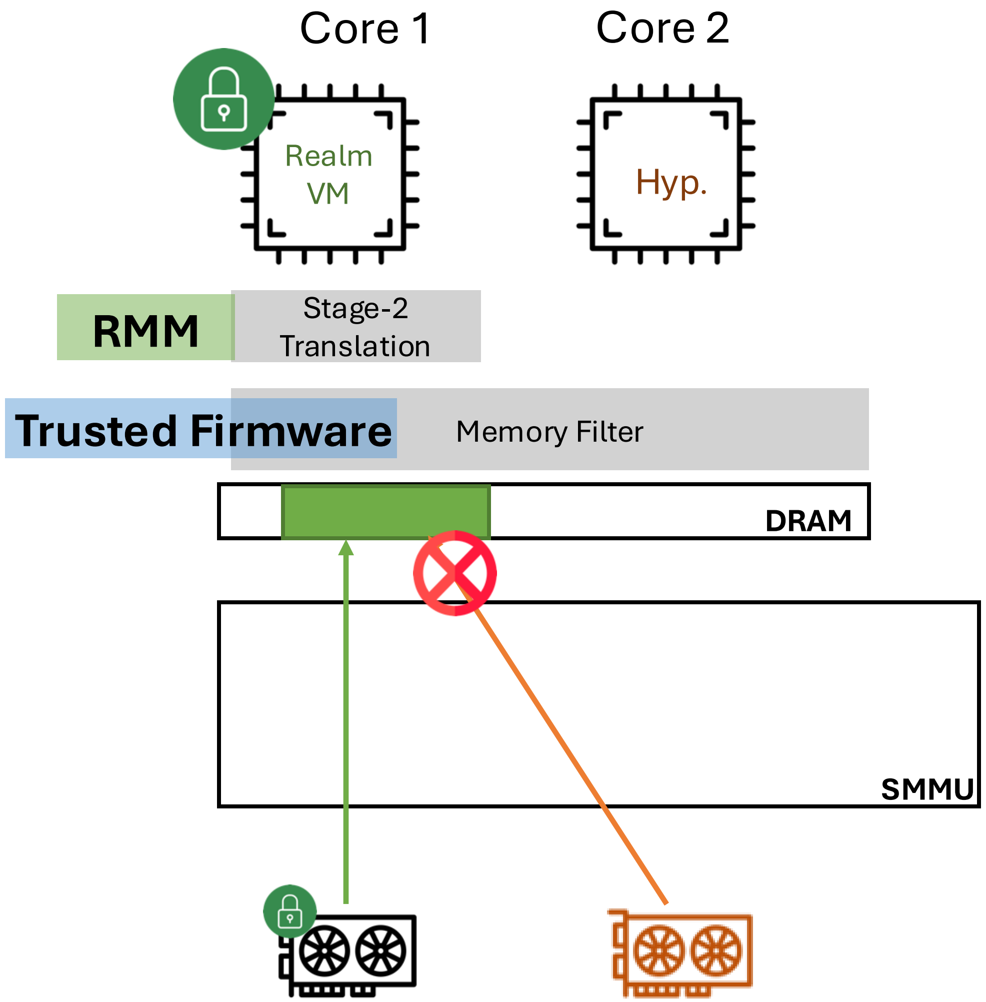
Isolate device accesses



Invariant:

- Isolate devices to their CVM memory

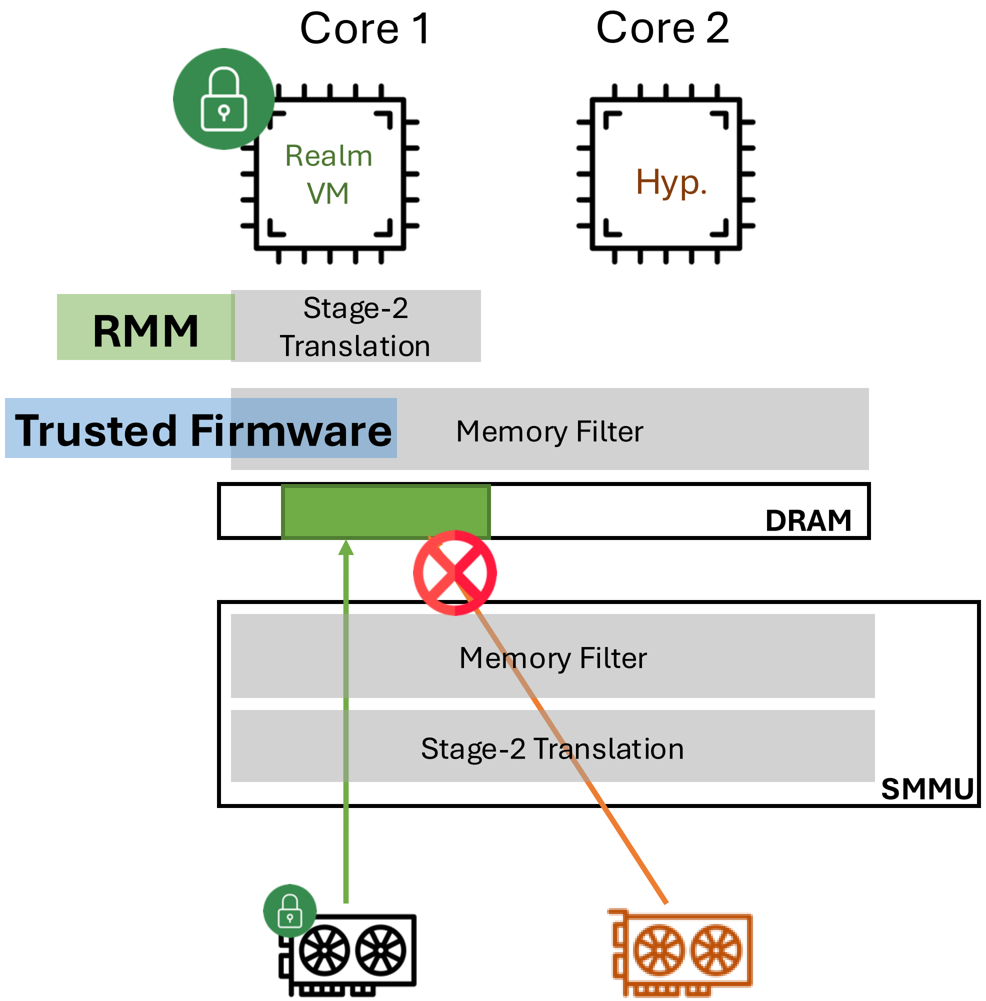
Isolate device accesses



Invariant:

- Isolate devices to their CVM memory

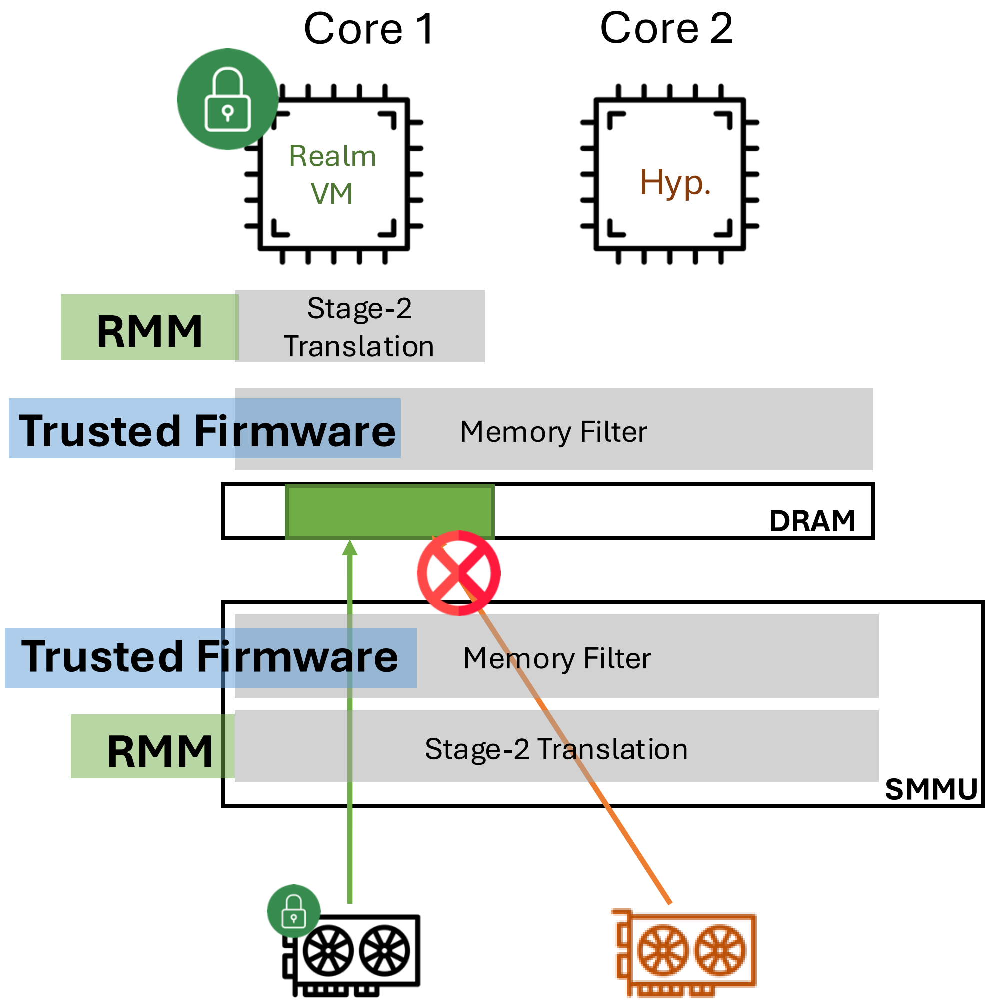
Isolate device accesses



Invariant:

- Isolate devices to their CVM memory

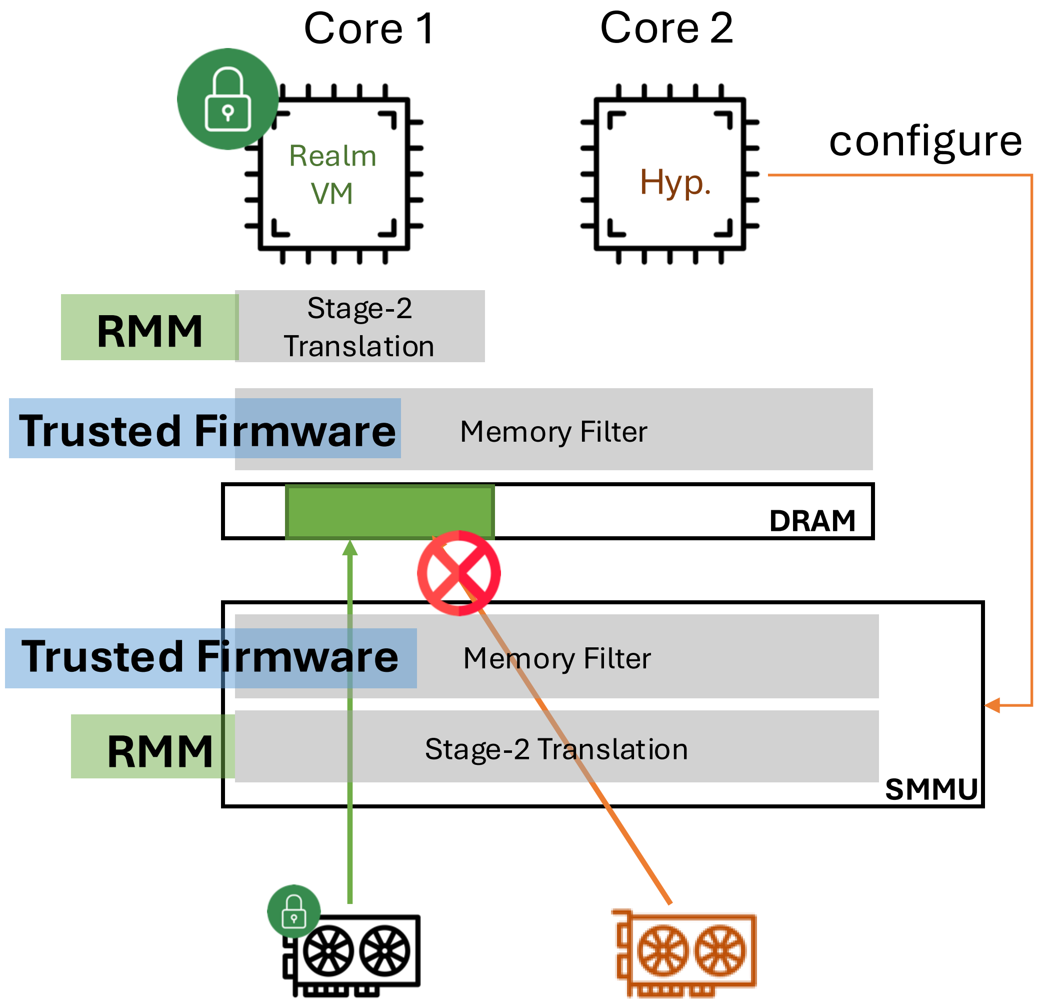
Isolate device accesses



Invariant:

- Isolate devices to their CVM memory

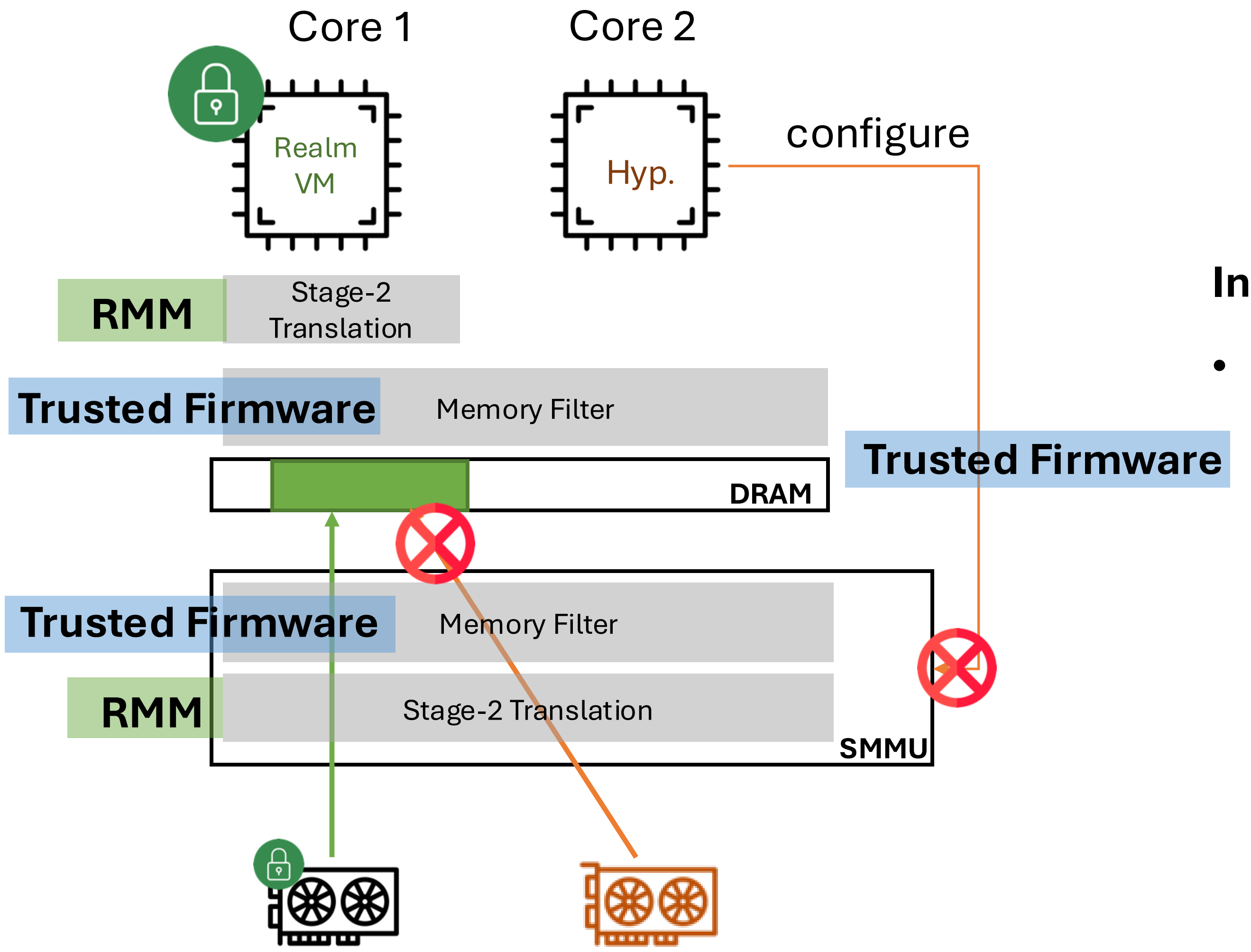
Isolate device accesses



Invariant:

- Isolate devices to their CVM memory

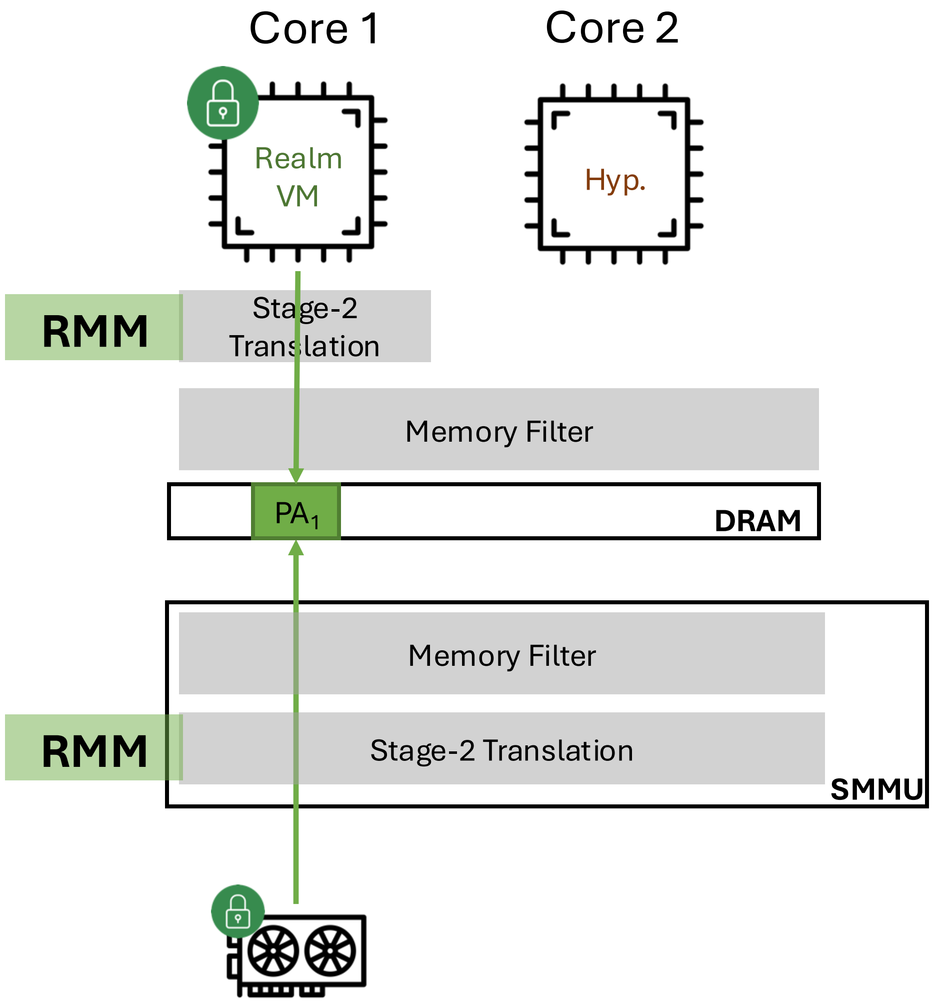
Isolate device accesses



Invariant:

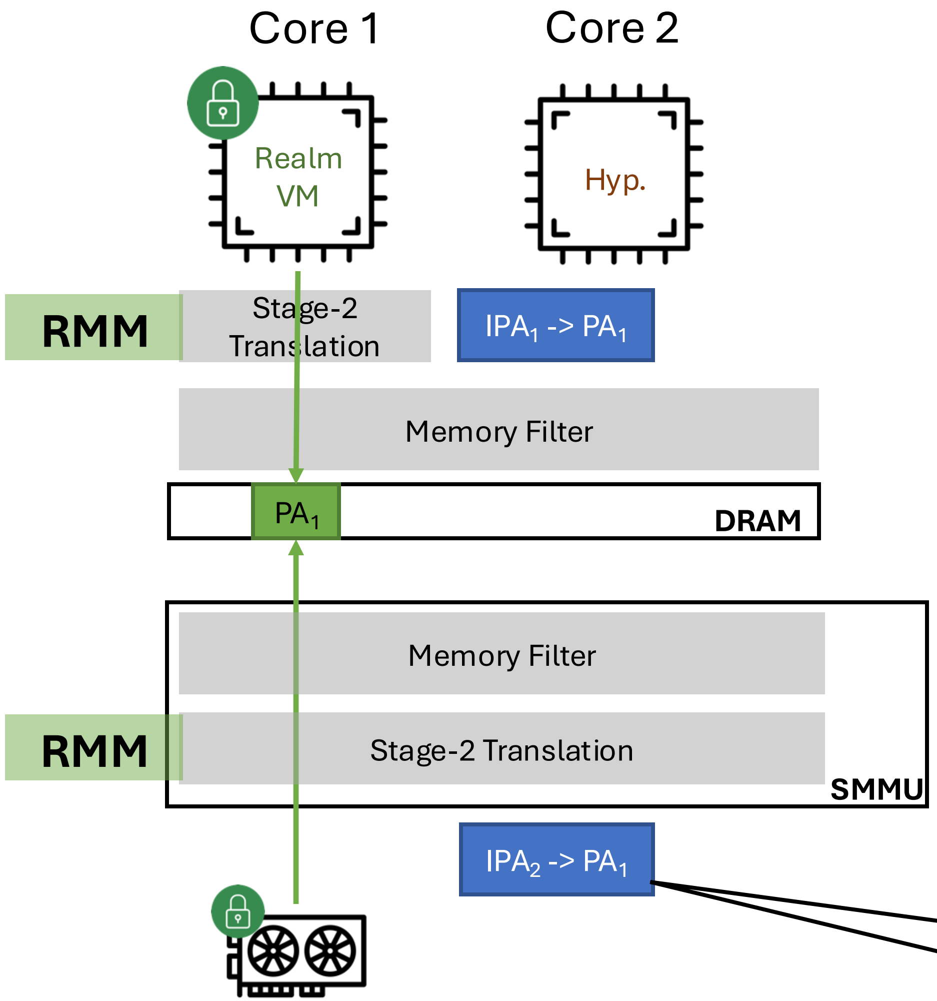
- Isolate devices to their CVM memory

Synchronize views



Invariant:

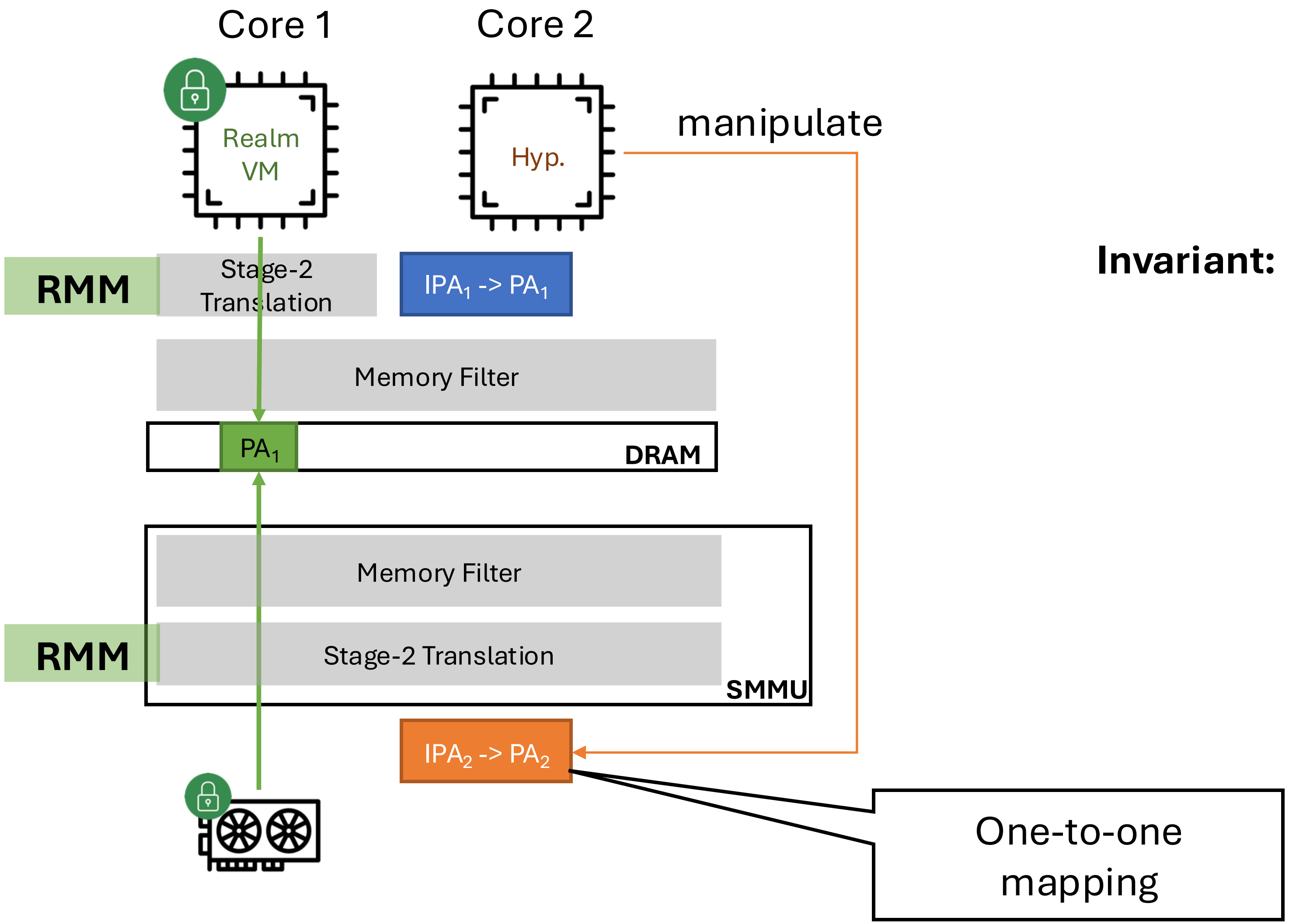
Synchronize views



Invariant:

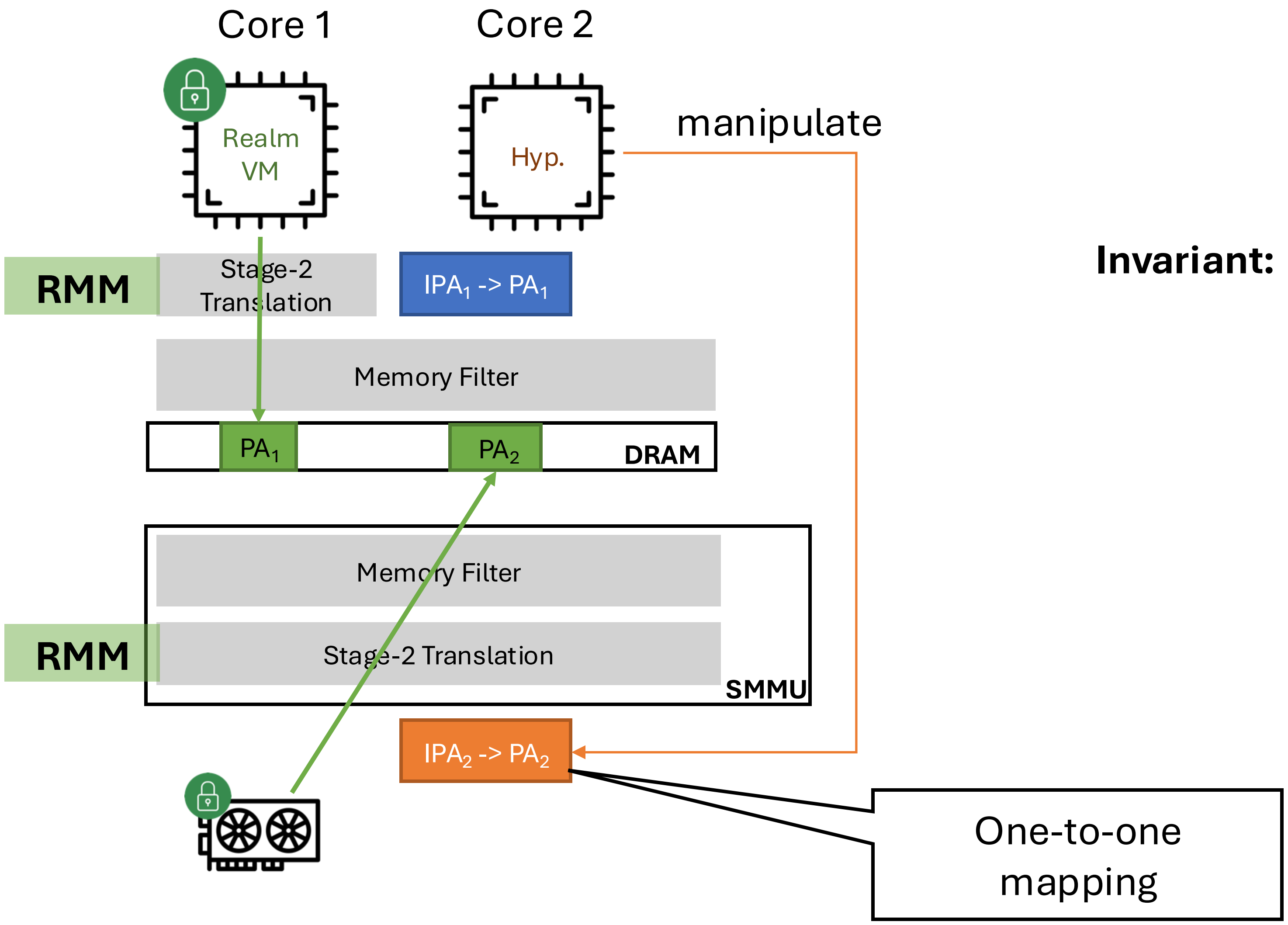
One-to-one mapping

Synchronize views



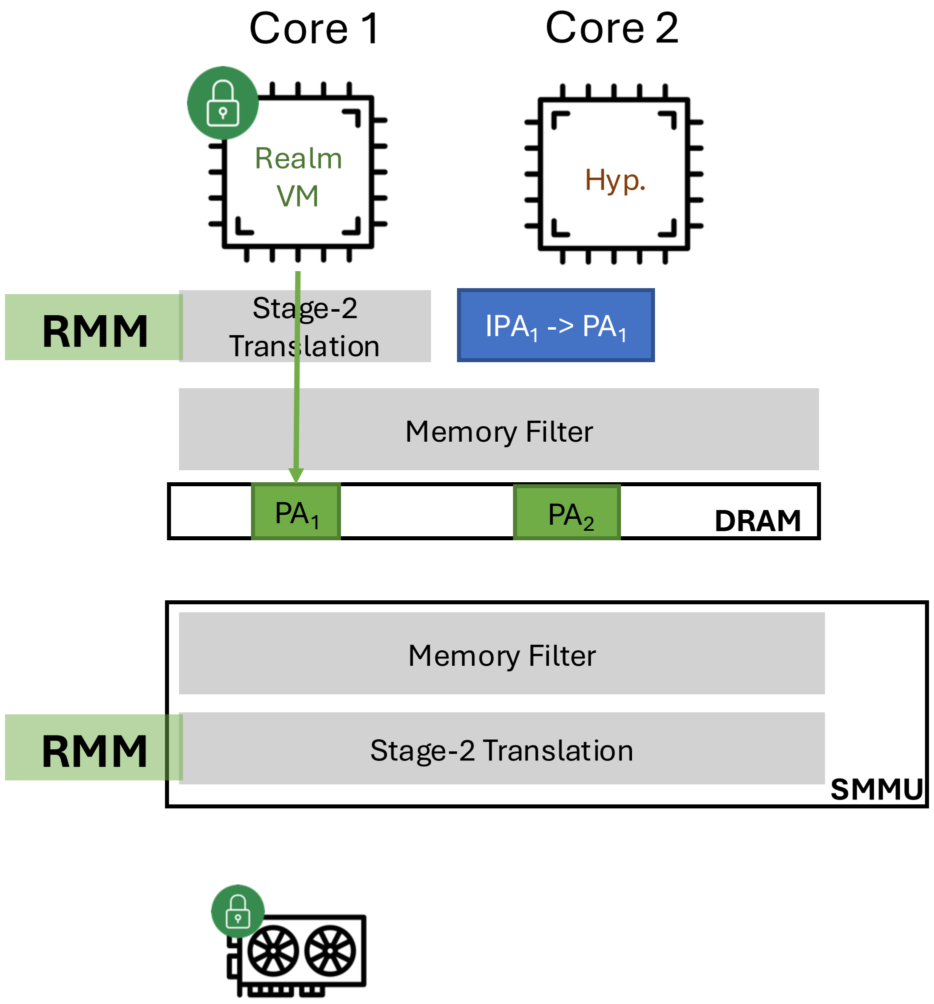
Invariant:

Synchronize views



Invariant:

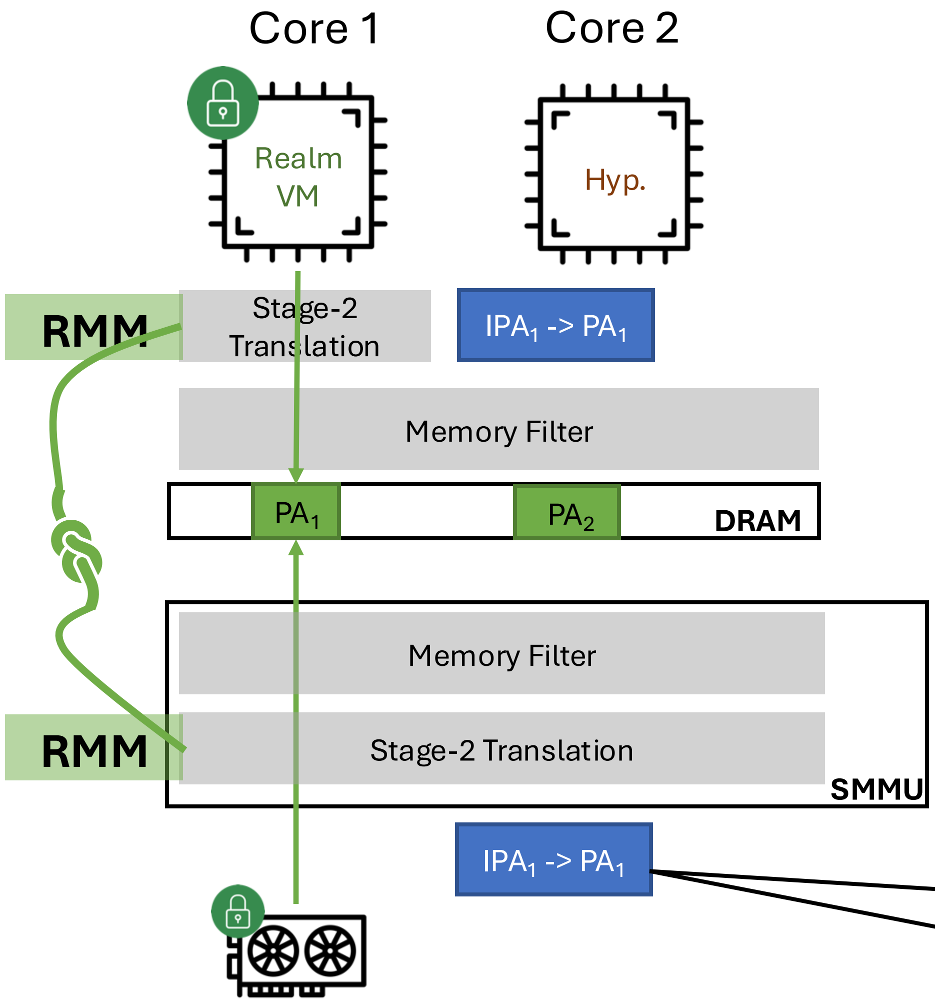
Synchronize views



Invariant:

- Isolate devices to their CVM memory
- One-to-one mapping between IPA->PA
- Ensure device and VM always see the same view

Synchronize views

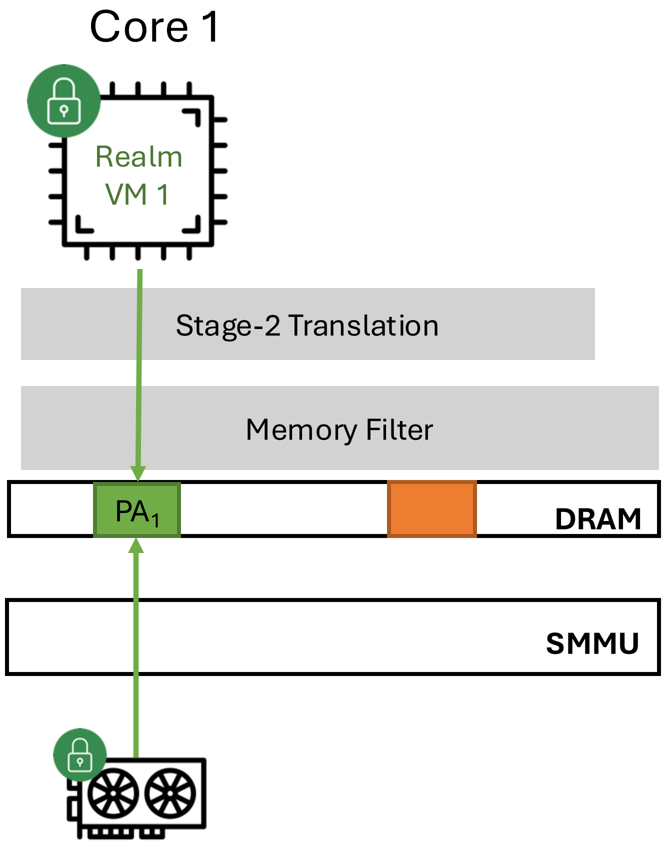


One-to-one mapping with same view

Invariant:

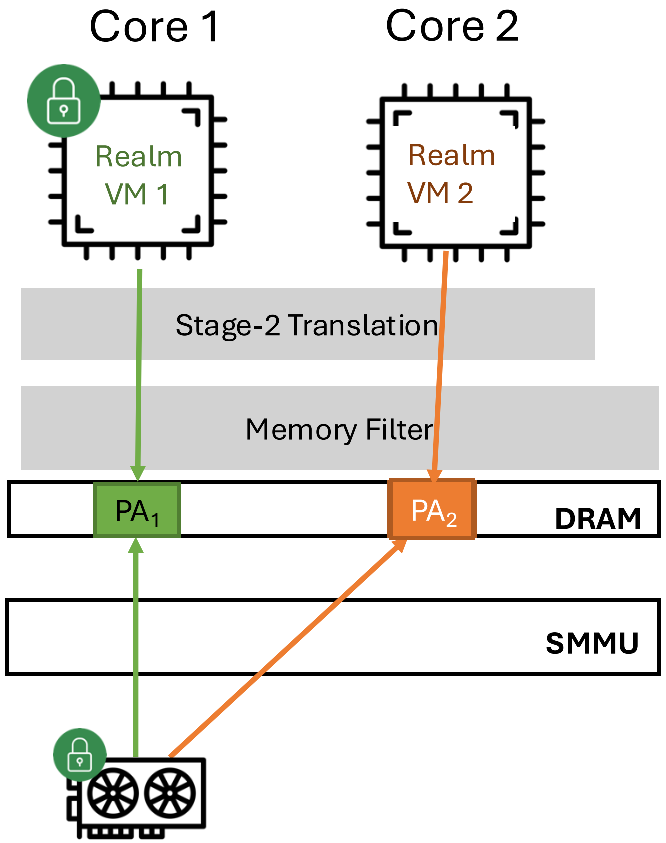
- Isolate devices to their CVM memory
- One-to-one mapping between IPA->PA
- Ensure device and VM always see the same view

Exclusive device ownership



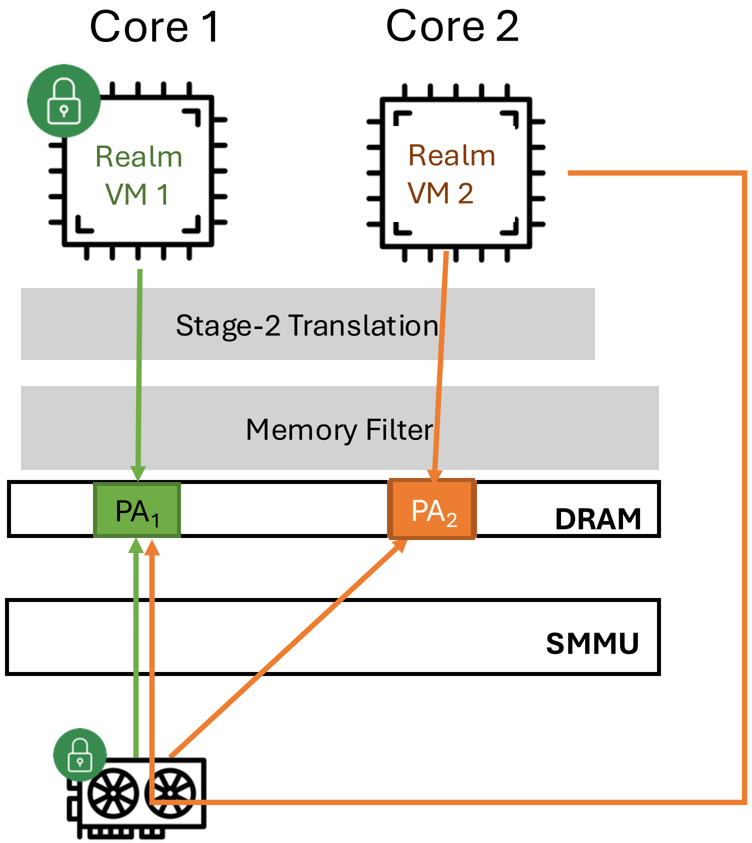
Invariant:

Exclusive device ownership



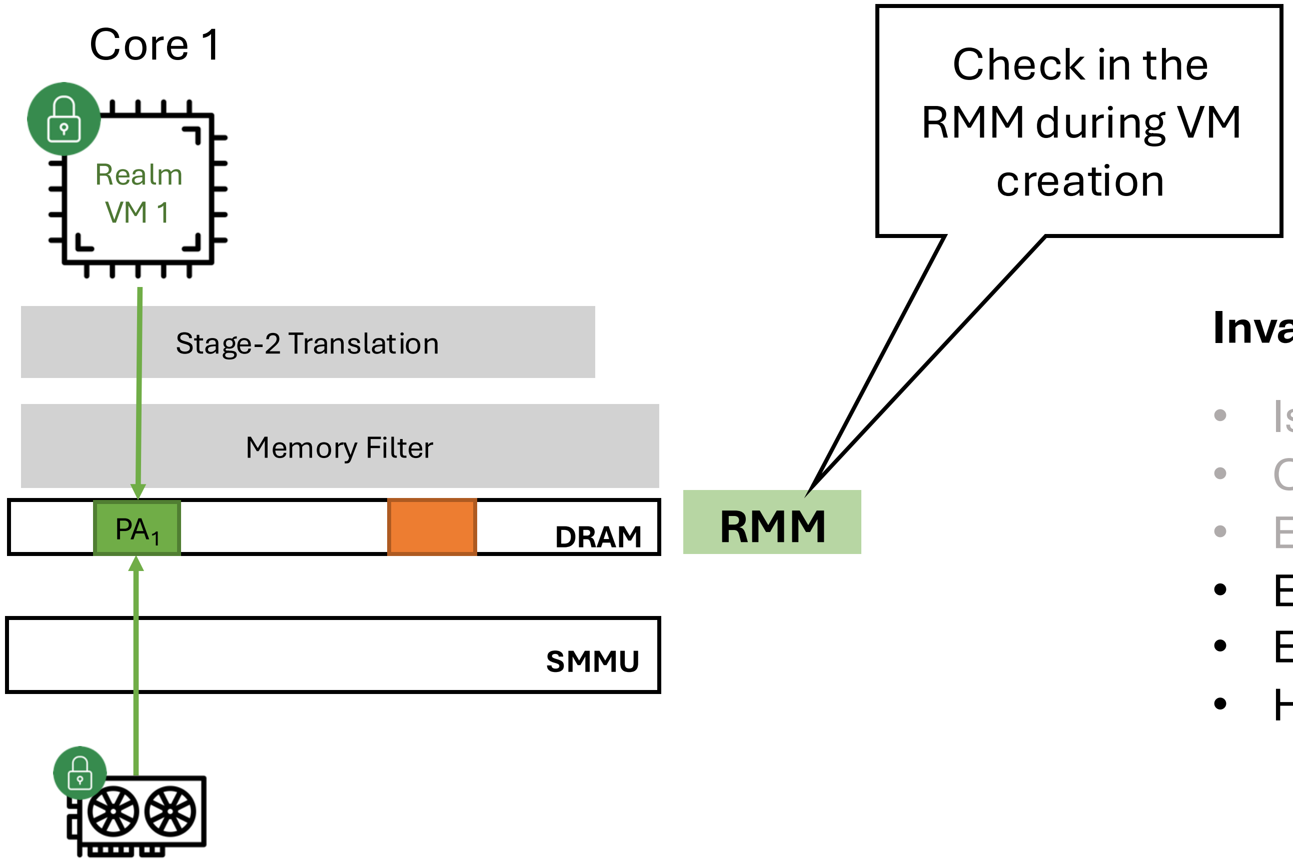
Invariant:

Exclusive device ownership



Invariant:

Exclusive device ownership

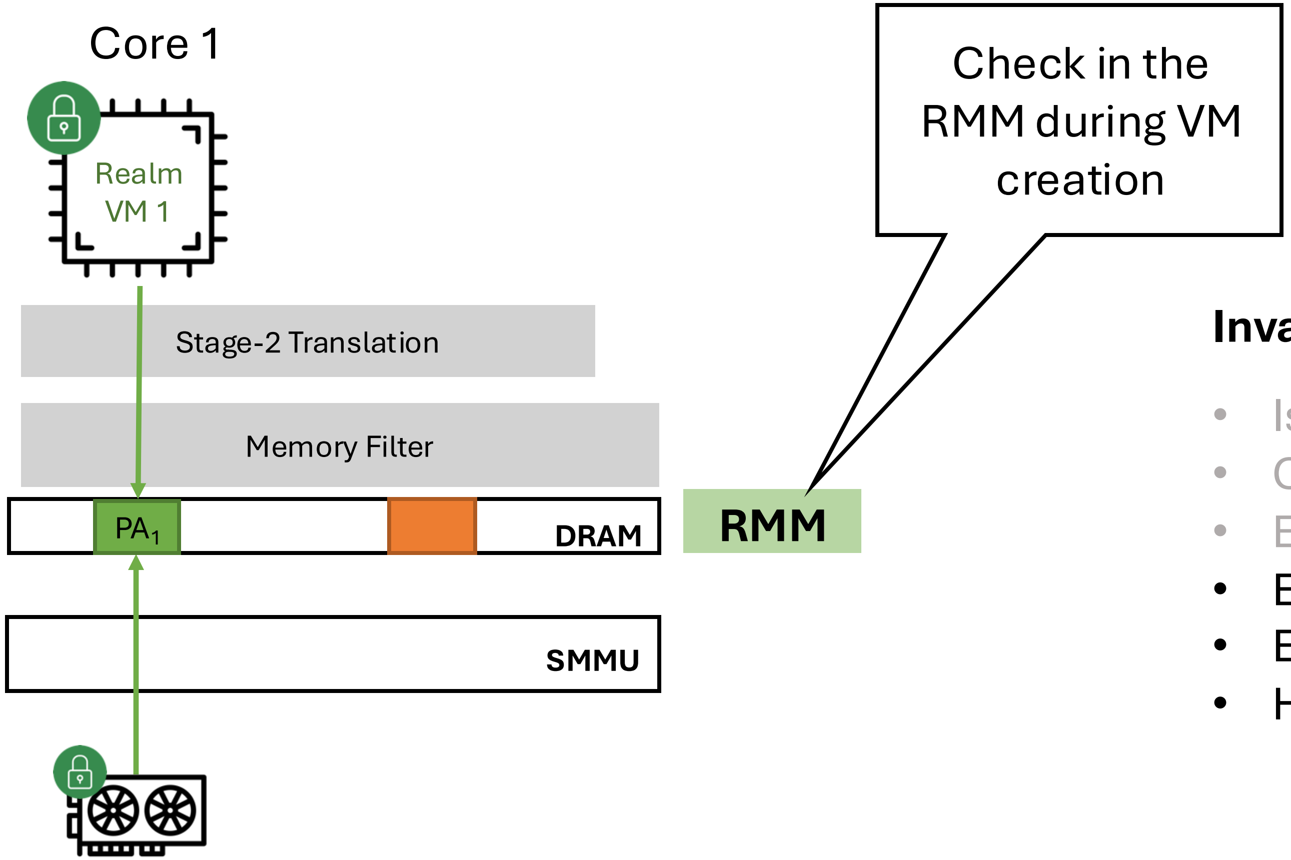


Check in the RMM during VM creation

Invariant:

- Isolate devices to their CVM memory
- One-to-one mapping between IPA->PA
- Ensure device and VM always see the same view
- Ensure exclusive device ownership
- Establish unforgereable identity with attestation
- Hardware based memory encryption on PCIe bus

Exclusive device ownership

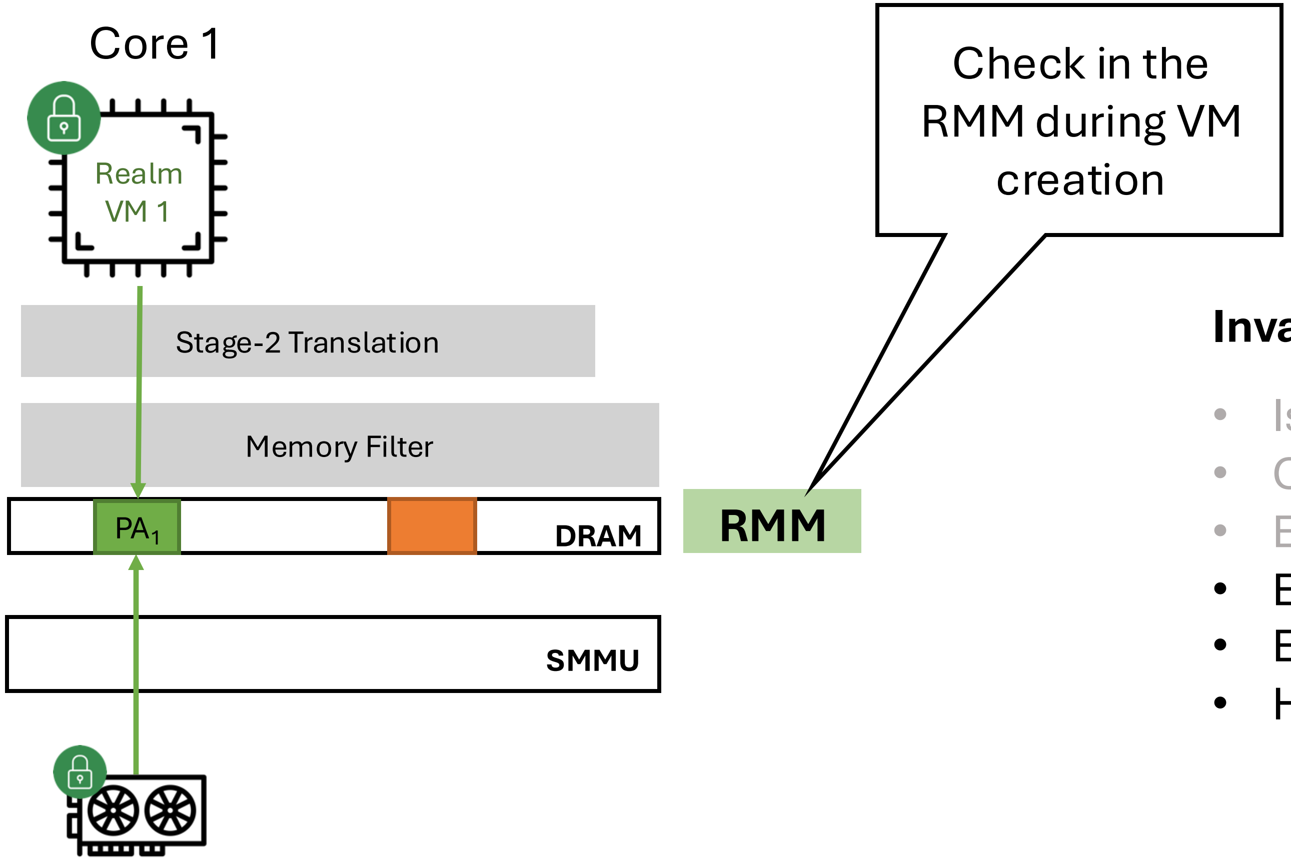


Check in the RMM during VM creation

Invariant:

- Isolate devices to their CVM memory
- One-to-one mapping between IPA->PA
- Ensure device and VM always see the same view
- Ensure exclusive device ownership
- Establish unforgereable identity with attestation
- Hardware based memory encryption on PCIe bus

Exclusive device ownership

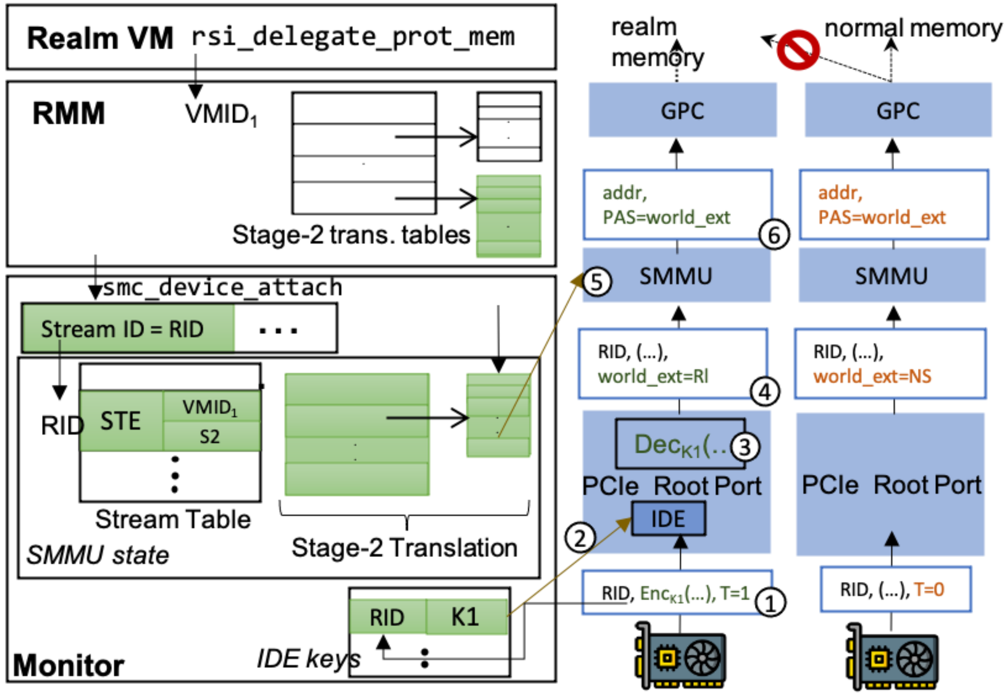


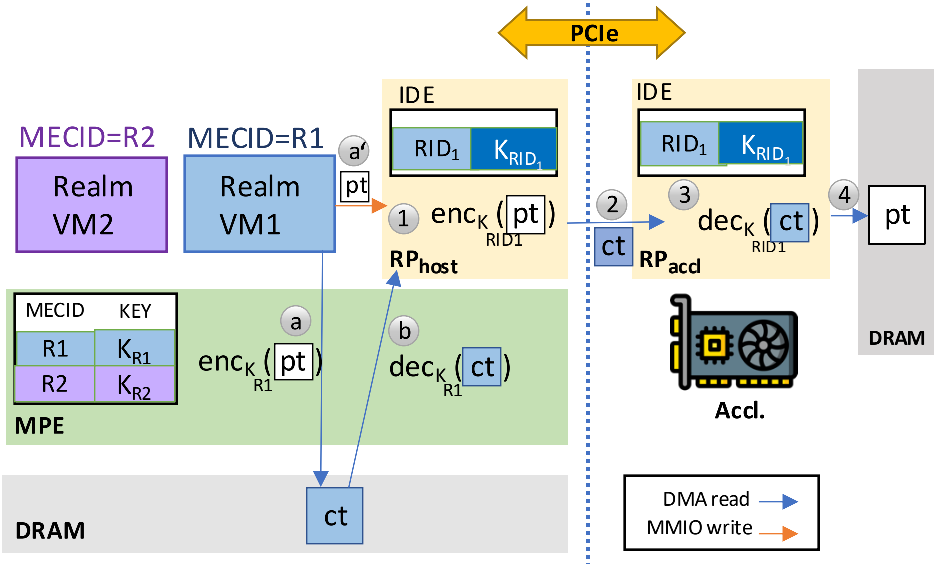
Check in the RMM during VM creation

Invariant:

- Isolate devices to their CVM memory
- One-to-one mapping between IPA->PA
- Ensure device and VM always see the same view
- Ensure exclusive device ownership
- Establish unforgereable identity with attestation
- Hardware based memory encryption on PCIe bus

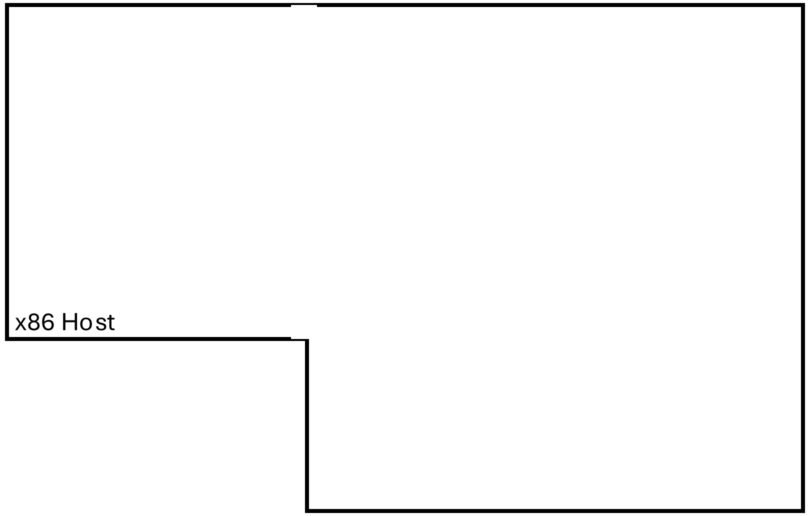
Putting it together





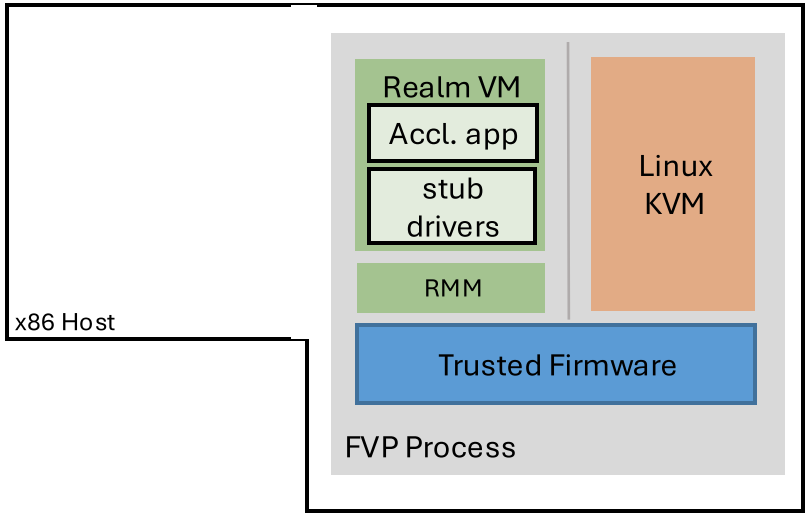
MECID	KEY
R1	K$_{R1}$
R2	K$_{R2}$

Implementation



Implementation

- No hardware with ARM CCA yet, but
  - Arm's simulator (FVP) supports CCA
  - Little/No support for PCIe devices
- Performance evaluation prototype: Arm Cortex-A53



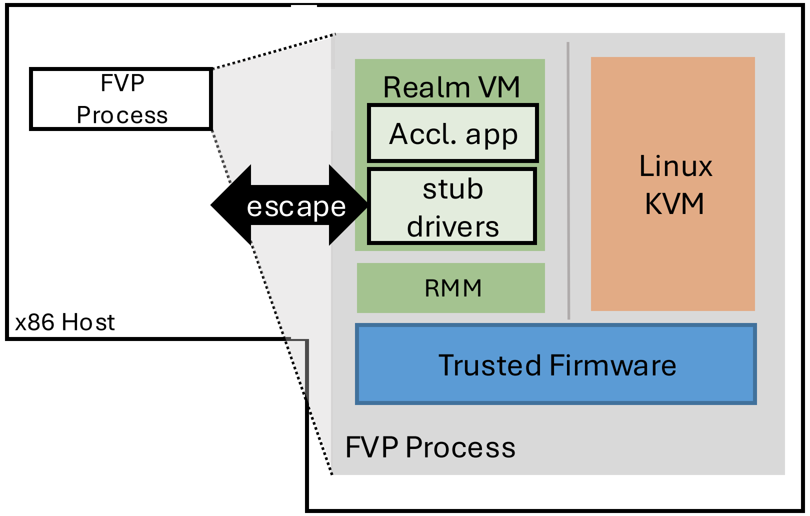
Compatibility

- We only change the RMM, trusted firmware, the guest Linux kernel
- No changes to the device drivers, runtime, or applications

- Monitor: 1588 LoC
- RMM: 382 LoC
- Guest kernel: 1734 LoC

Implementation

- No hardware with ARM CCA yet, but
  - Arm's simulator (FVP) supports CCA
  - Little/No support for PCIe devices
- Performance evaluation prototype: Arm Cortex-A53



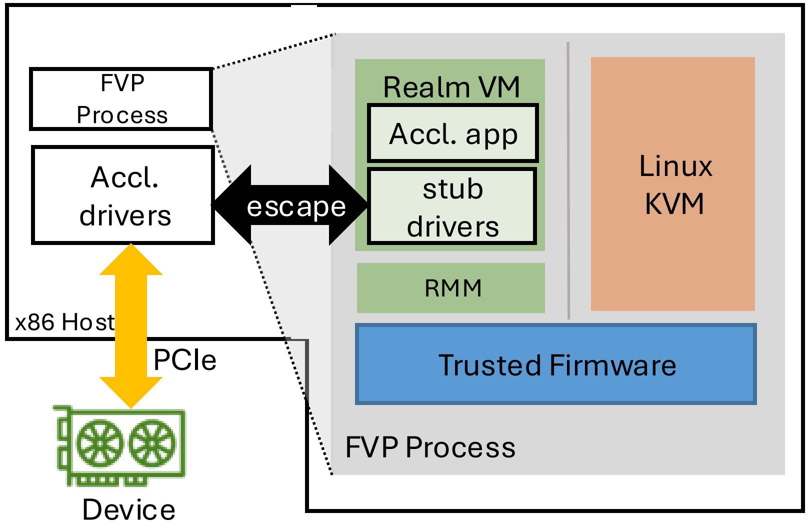
Compatibility

- We only change the RMM, trusted firmware, the guest Linux kernel
- No changes to the device drivers, runtime, or applications

- Monitor: 1588 LoC
- RMM: 382 LoC
- Guest kernel: 1734 LoC

Implementation

- No hardware with ARM CCA yet, but
  - Arm’s simulator (FVP) supports CCA
  - Little/No support for PCIe devices
- Performance evaluation prototype: Arm Cortex-A53



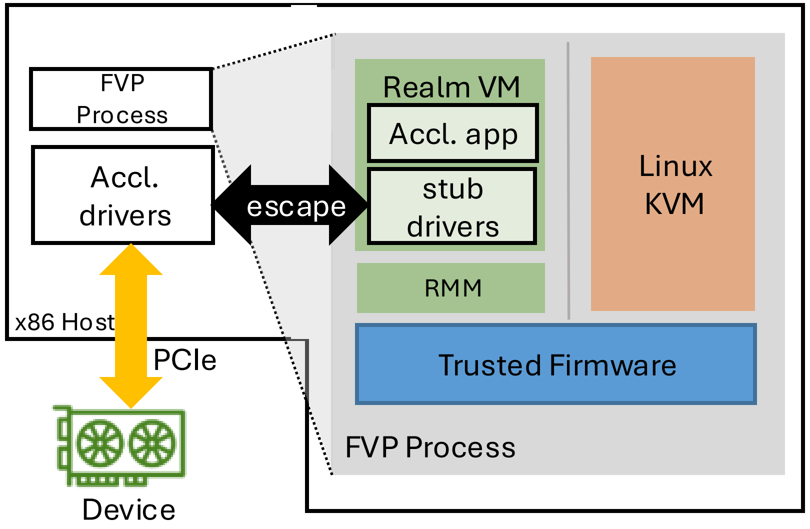
Compatibility

- We only change the RMM, trusted firmware, the guest Linux kernel
- No changes to the device drivers, runtime, or applications

- Monitor: 1588 LoC
- RMM: 382 LoC
- Guest kernel: 1734 LoC

Implementation

- No hardware with ARM CCA yet, but
  - Arm’s simulator (FVP) supports CCA
  - Little/No support for PCIe devices
- Performance evaluation prototype: Arm Cortex-A53



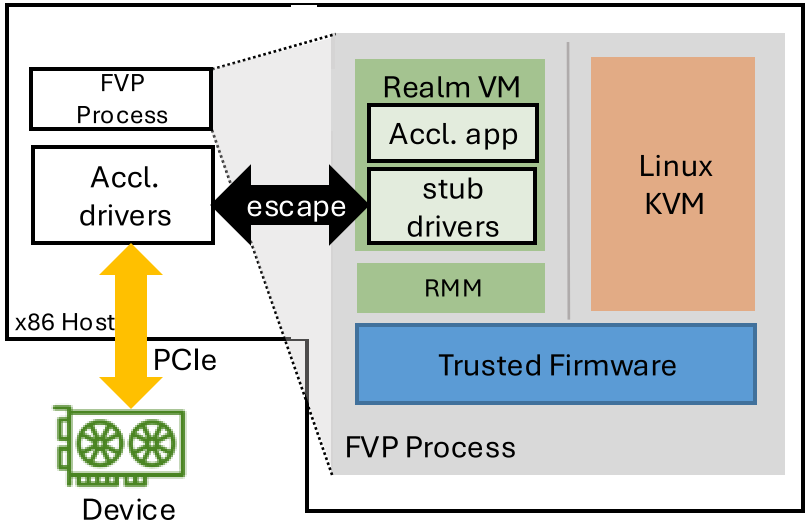
Compatibility

- We only change the RMM, trusted firmware, the guest Linux kernel
- No changes to the device drivers, runtime, or applications

- Monitor: 1588 LoC
- RMM: 382 LoC
- Guest kernel: 1734 LoC

Implementation

- No hardware with ARM CCA yet, but
    - Arm's simulator (FVP) supports CCA
    - Little/No support for PCIe devices
- Performance evaluation prototype: Arm Cortex-A53



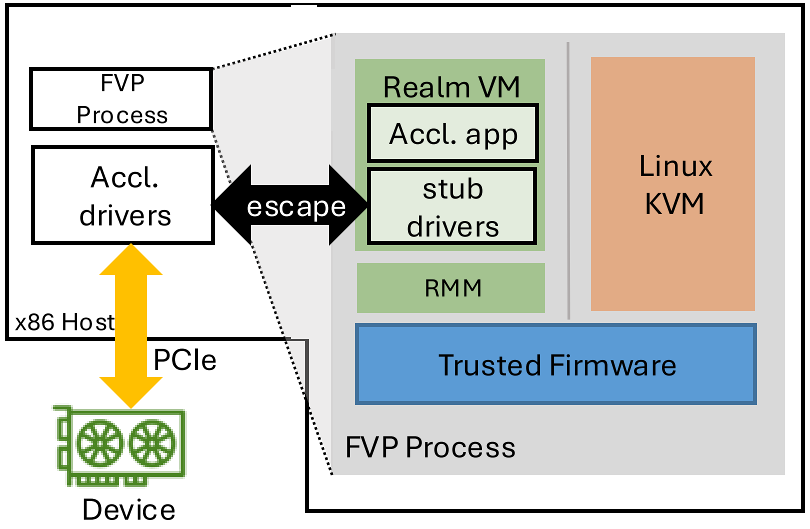
Compatibility

- We only change the RMM, trusted firmware, the guest Linux kernel
- No changes to the device drivers, runtime, or applications

- Monitor: 1588 LoC
- RMM: 382 LoC
- Guest kernel: 1734 LoC

Implementation

- No hardware with ARM CCA yet, but
  - Arm’s simulator (FVP) supports CCA
  - Little/No support for PCIe devices
- Performance evaluation prototype: Arm Cortex-A53



Compatibility

- We only change the RMM, trusted firmware, the guest Linux kernel
- No changes to the device drivers, runtime, or applications

API	Status	Description
`rmi_data_create`	changed	add data from normal world to realm memory. ACAI adds `attach_dev` flag.
`rsi_delegate_prot_mem`	new	delegate realm memory to protected memory. calls `smc_delegate_prot_mem`.
`smc_device_attach`	new	attach and detach a device from realm.
`smc_delegate_prot_mem`	new	delegate realm memory to protected memory. add stage-2 translation for the SMMU.

- Monitor: 1588 LoC
- RMM: 382 LoC
- Guest kernel: 1734 LoC

Evaluation Setup

Evaluation Setup

- We benchmark on a GPU and FPGA
- Measure number of instructions on the simulator as a performance measure

Evaluation Setup

- We benchmark on a GPU and FPGA
- Measure number of instructions on the simulator as a performance measure

GPU Benchmarks

App	Domain	Tasks	T Size	P Size
nn	Dense linear algebra	1	1	42764
gaussian	Dense linear algebra	3148	38	1575 × 1575
needle	Dynamic programming	229	39	1840
pathfinder	Dynamic programming	5	20	50000 × 100
bfs	Graph traversal	2	3	1840
srad_v1	Structured grid	102	2	502 × 458
srad_v2	Structured grid	4	64	2048 × 2048
hotspot	Structured grid	5	3	512 × 512
backprop	Unstructured grid	2	71	262144 × 16 × 1

FPGA Benchmarks

App	Domain	T Size	P Size
matmul5	Matrix Multiplication	300 B	42764
matmul10	Matrix Multiplication	1200 B	1575 × 1575
svd32	Singular Value Decomposition	320KB	1840
svd64	Singular Value Decomposition	20	50000 × 100

Evaluation Setup

- We benchmark on a GPU and FPGA
- Measure number of instructions on the simulator as a performance measure

- **Baseline**: Encryption with Bounce Buffers
  Realm VM encrypts and copies to Normal world
- **Acai**
  Setup realm memory that device directly accesses

GPU Benchmarks

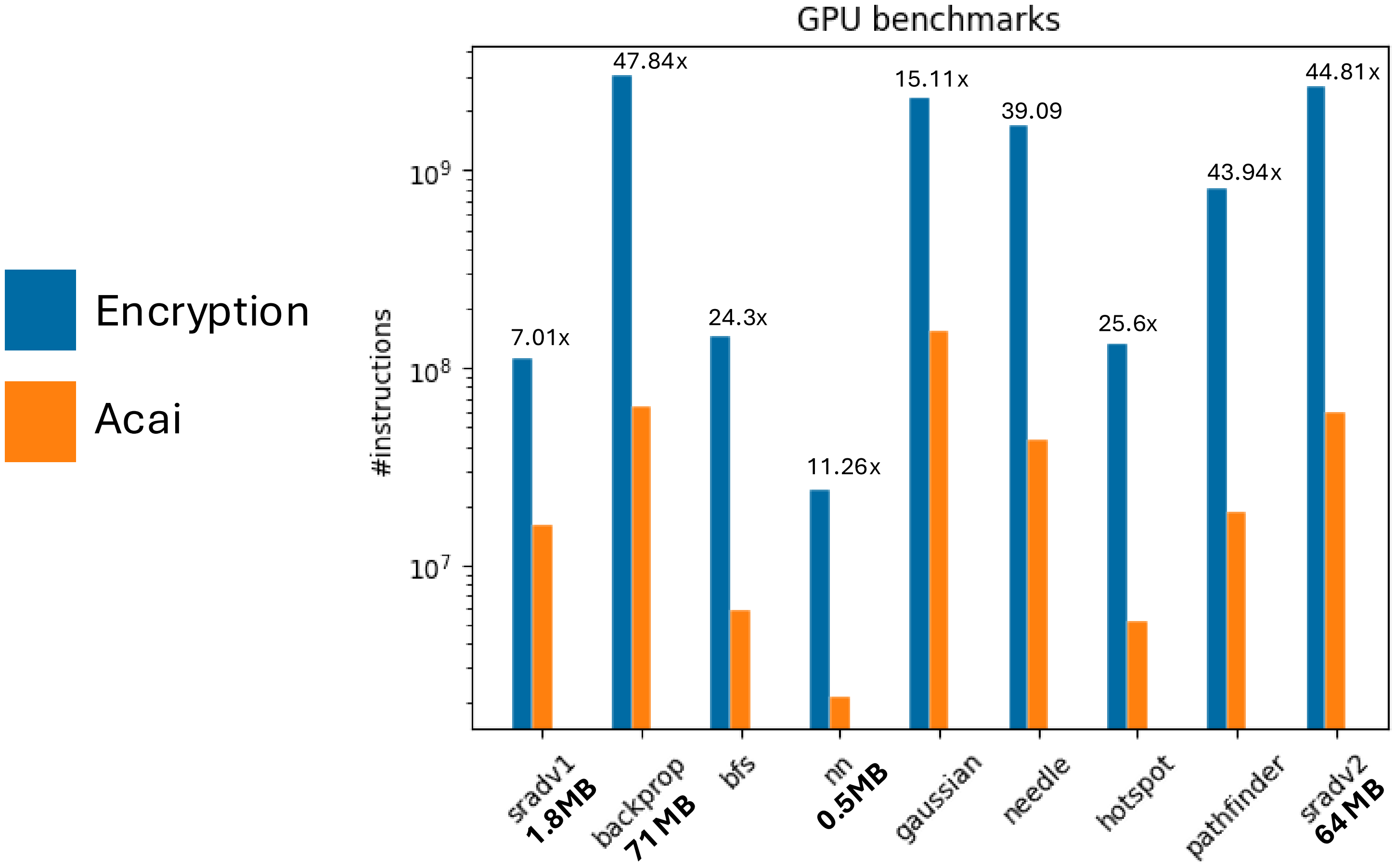
App	Domain	Tasks	T Size	P Size
nn	Dense linear algebra	1	1	42764
gaussian	Dense linear algebra	3148	38	1575 × 1575
needle	Dynamic programming	229	39	1840
pathfinder	Dynamic programming	5	20	50000 × 100
bfs	Graph traversal	2	3	1840
srad_v1	Structured grid	102	2	502 × 458
srad_v2	Structured grid	4	64	2048 × 2048
hotspot	Structured grid	5	3	512 × 512
backprop	Unstructured grid	2	71	262144 × 16 × 1

FPGA Benchmarks

App	Domain	T Size	P Size
matmul5	Matrix Multiplication	300 B	42764
matmul10	Matrix Multiplication	1200 B	1575 × 1575
svd32	Singular Value Decomposition	320KB	1840
svd64	Singular Value Decomposition	20	50000 × 100

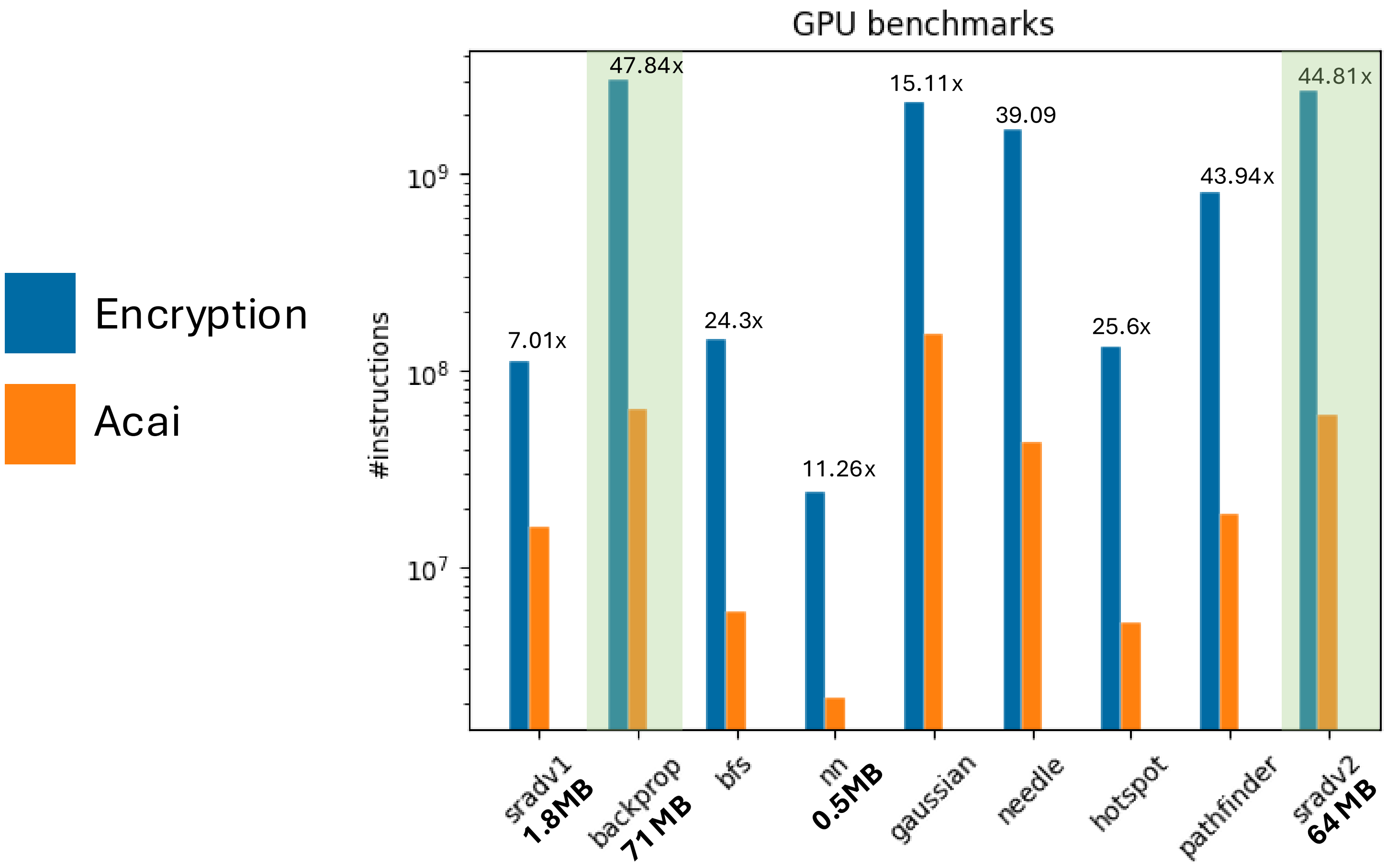
Impact of removing bounce buffers

Almost **26x** faster than encrypted mode for GPU



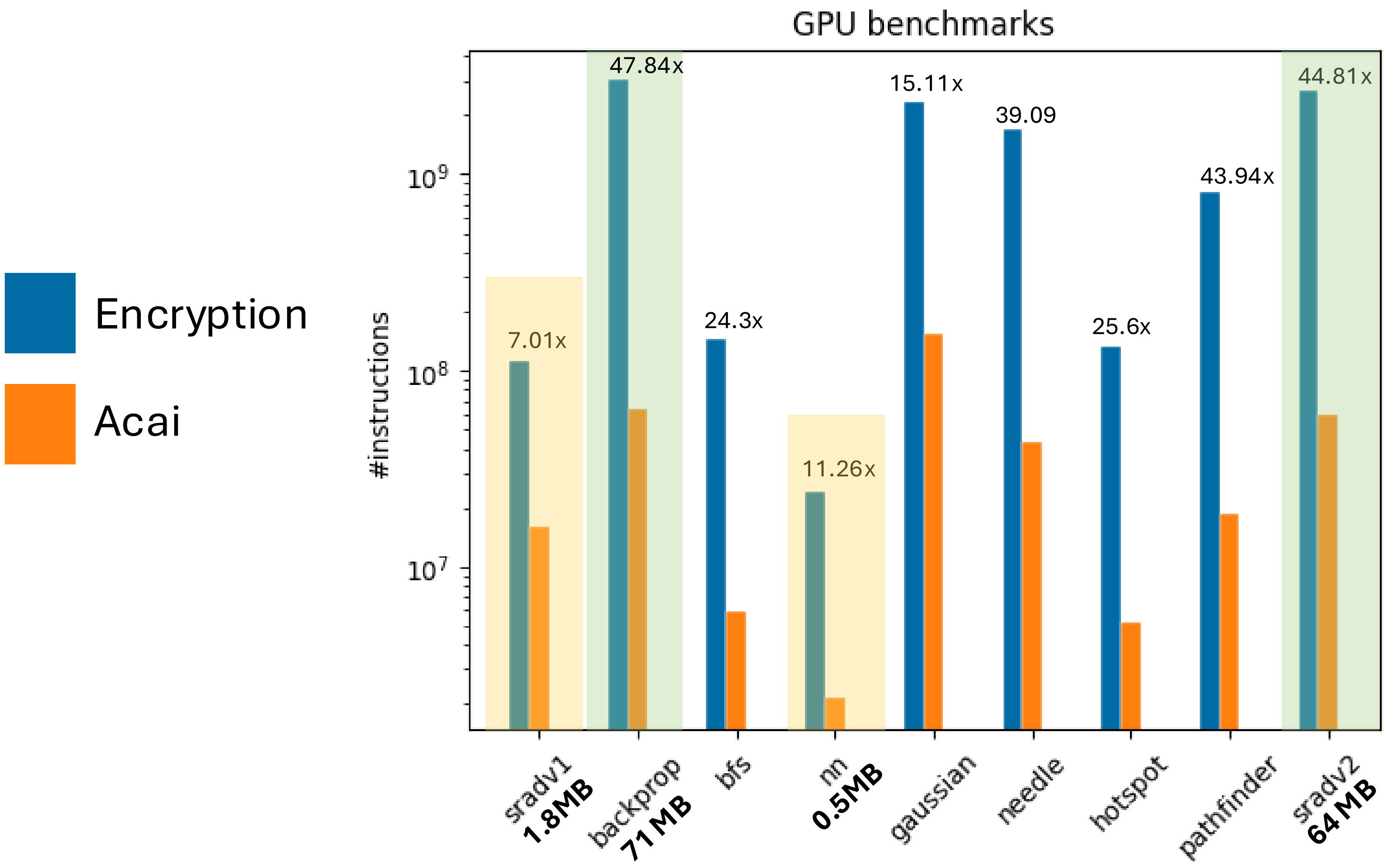
Impact of removing bounce buffers

Almost **26x** faster than encrypted mode for GPU



Impact of removing bounce buffers

Almost **26x** faster than encrypted mode for GPU

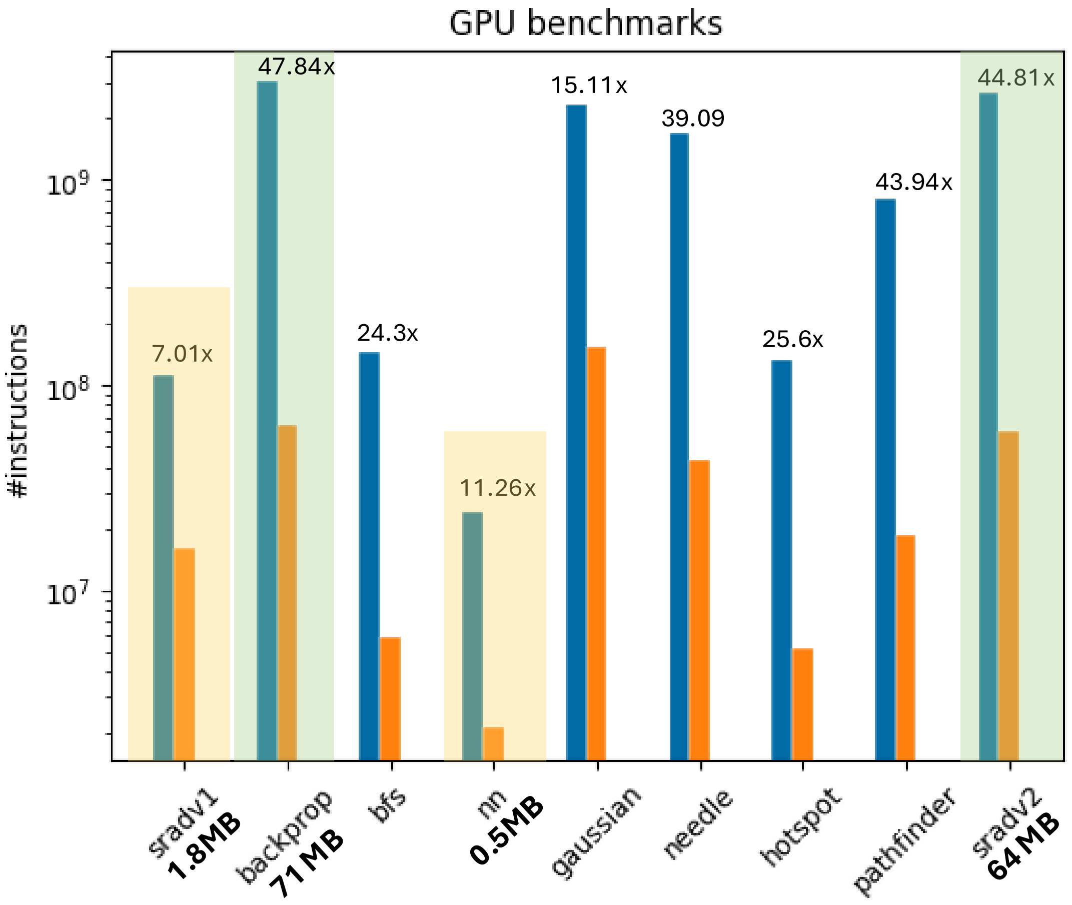


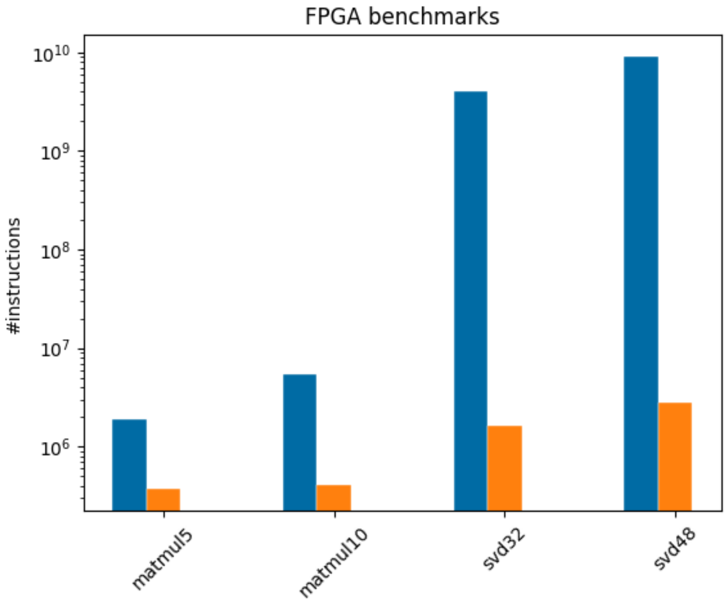
Impact of removing bounce buffers

Almost **26x** faster than encrypted mode for GPU

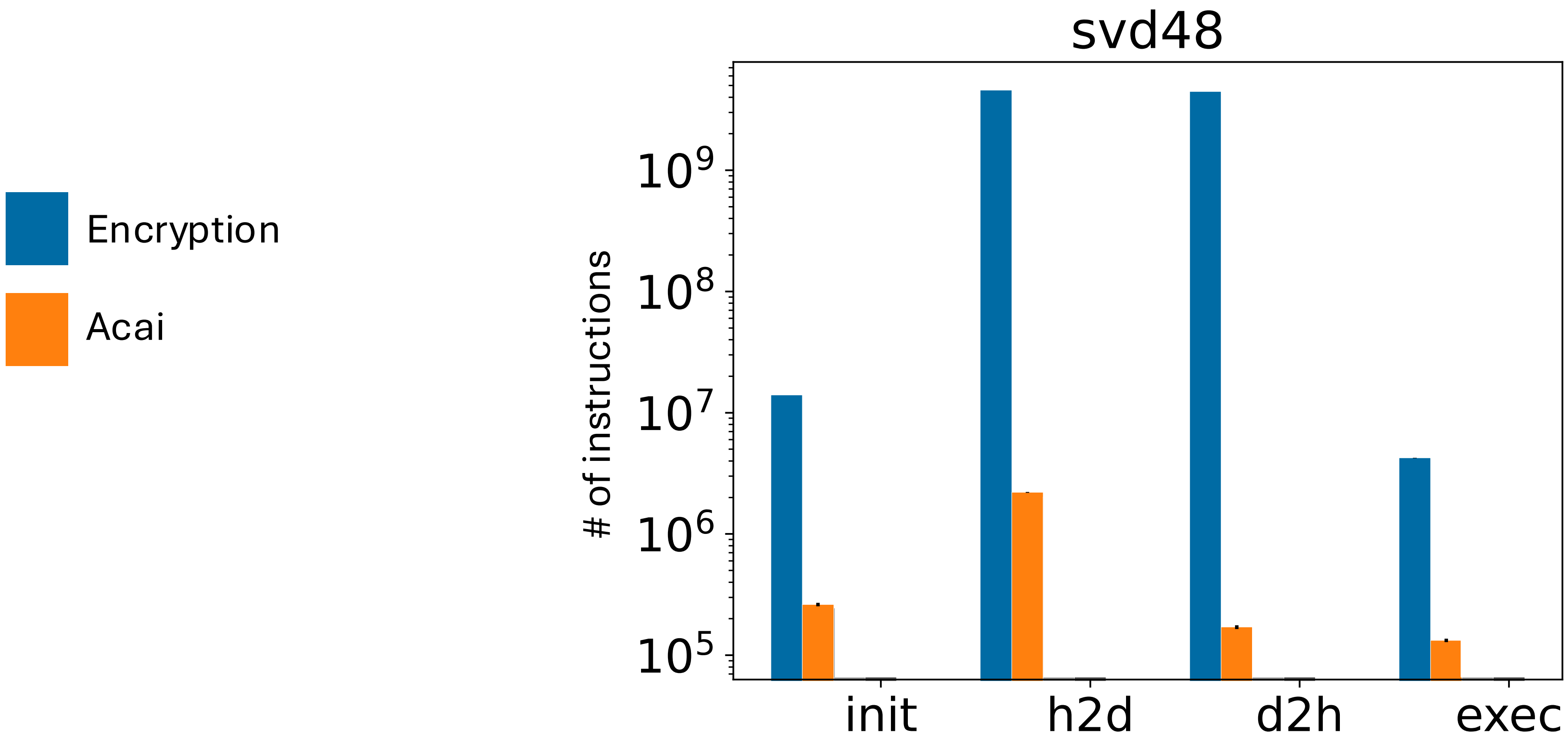
🟦 Encryption

🟧 Acai

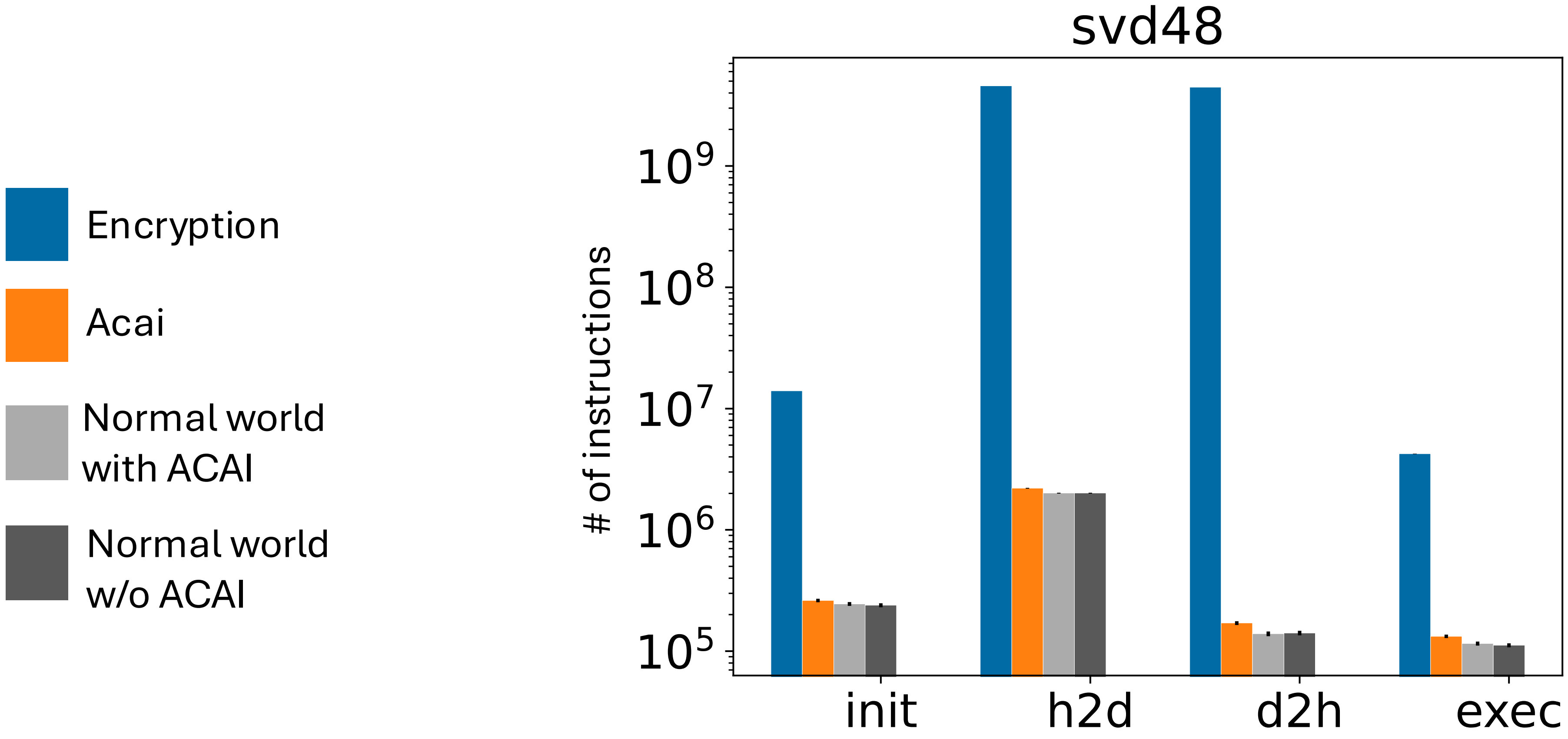




Other measurements



Other measurements



Effect on the normal world:
3.8% for GPU and 1.9% for FPGA benchmarks

Estimates on Arm Board

Estimates on Arm Board

- Measure the performance of context switches, interface calls, and memory operations
- Measure the performance for transferring a 4KB page with AES-GCM 256-bit block size
- Use FVP measurements to estimate the performance of Bounce Buffers and Acai
- Even with fast hardware encryption, **Acai is 2 orders of magnitude faster.**

Estimates on Arm Board

- Measure the performance of context switches, interface calls, and memory operations
- Measure the performance for transferring a 4KB page with AES-GCM 256-bit block size
- Use FVP measurements to estimate the performance of Bounce Buffers and Acai
- Even with fast hardware encryption, **Acai is 2 orders of magnitude faster.**

Estimates on Arm Board

- Measure the performance of context switches, interface calls, and memory operations
- Measure the performance for transferring a 4KB page with AES-GCM 256-bit block size
- Use FVP measurements to estimate the performance of Bounce Buffers and Acai
- Even with fast hardware encryption, **Acai is 2 orders of magnitude faster.**

Summary

- Confidential Computing is becoming ubiquitous, from mobiles to cloud
- Research question:
  How to extend the notion of Confidential Computing to peripherals and accelerators?
- Acai is one concrete instance to showcase the challenges
- We add device support to the simulator
- Acai is open source!

  https://github.com/sectrs-acai

ARTIFACT EVALUATED usenix ASSOCIATION AVAILABLE

ARTIFACT EVALUATED usenix ASSOCIATION FUNCTIONAL