The DragonBeam Framework:
Hardware-Protected Security Modules for In-Place Intrusion Detection

Man-Ki Yoon, Mihai Christodorescu, Lui Sha, Sibin Mohan

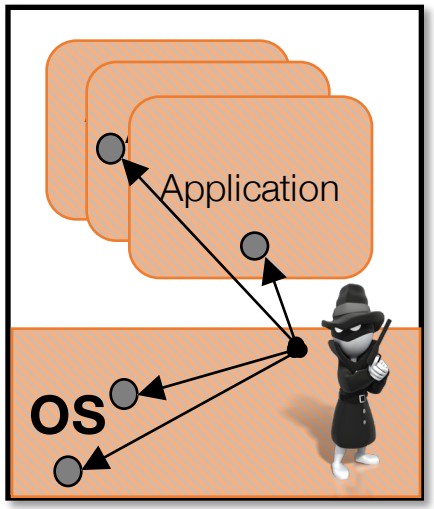
University of Illinois at Urbana-Champaign

Qualcomm Research Silicon Valley

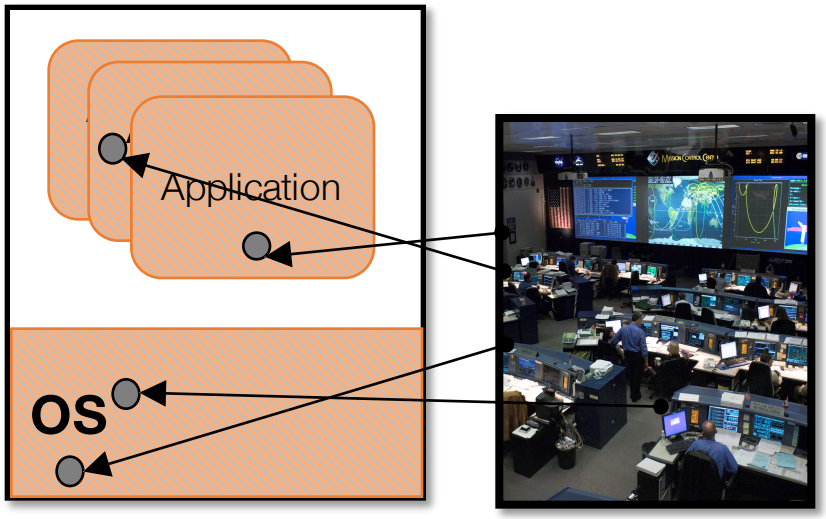
June 6, 2016

Security Monitoring

In-place Monitoring

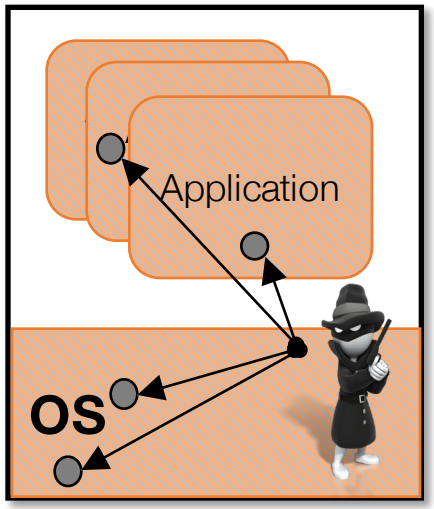


External Monitoring



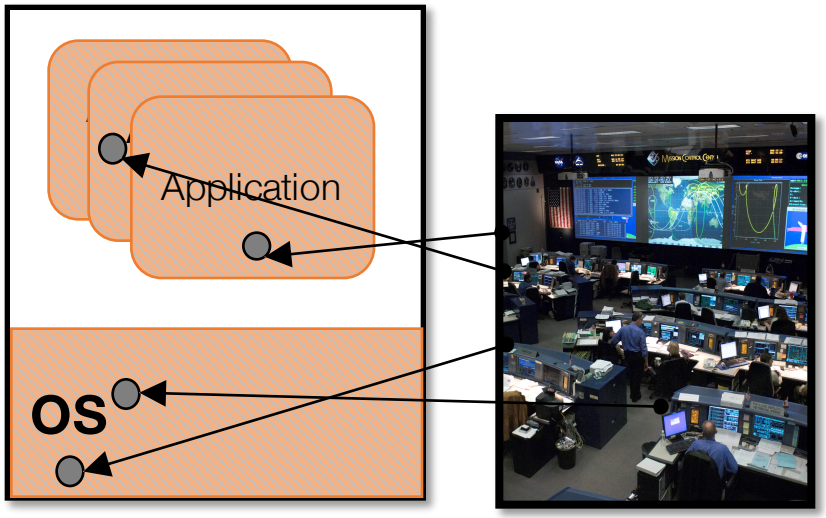
Security Monitoring

In-place Monitoring



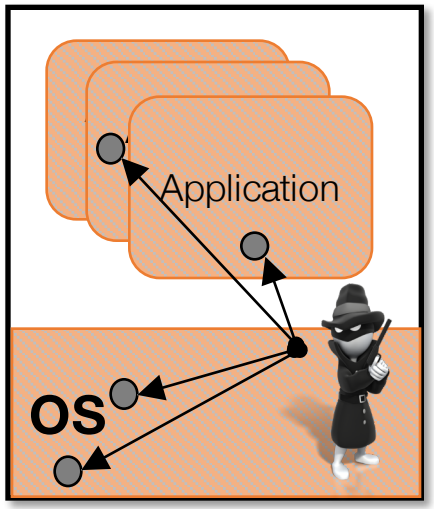
Unsafety of the monitor

External Monitoring



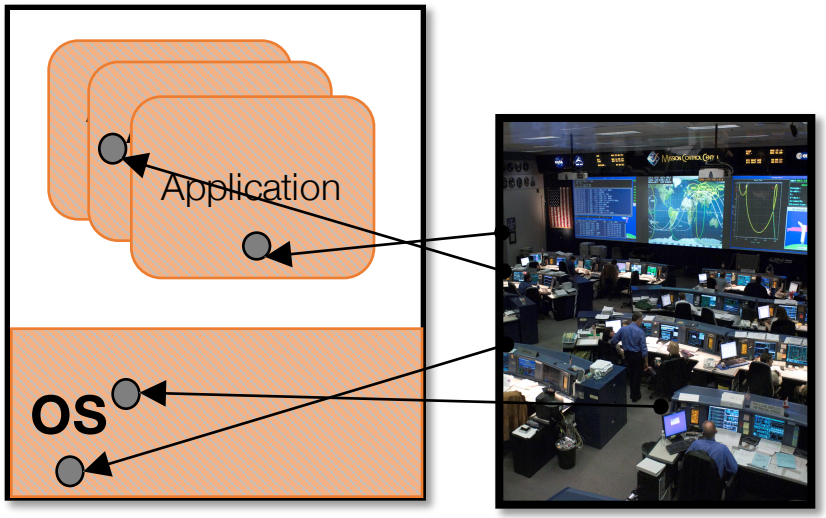
Security Monitoring

In-place Monitoring



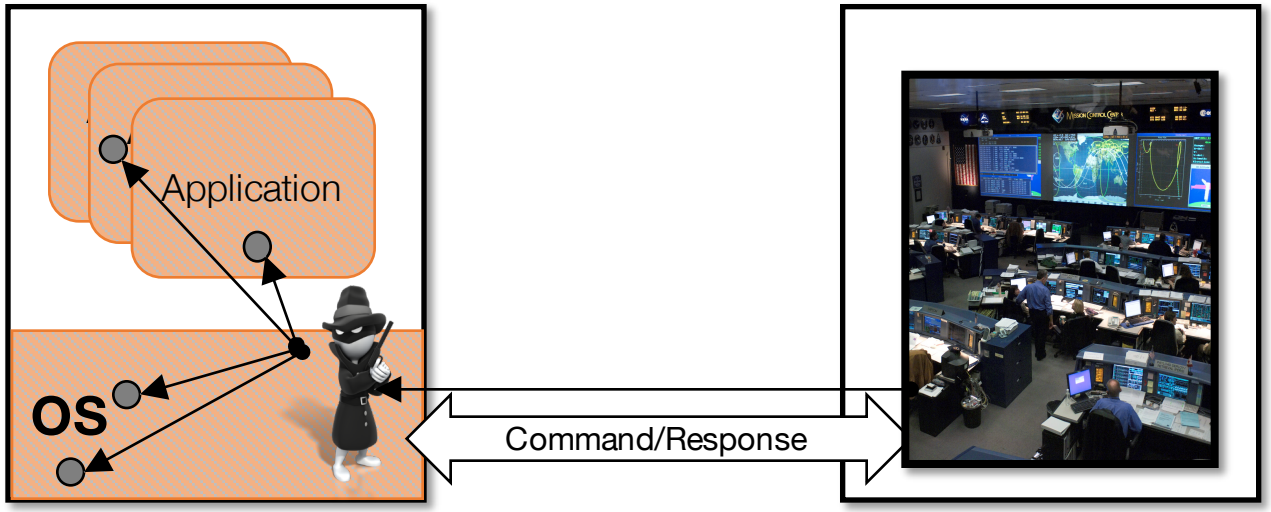
Unsafety of the monitor

External Monitoring



Semantic gap

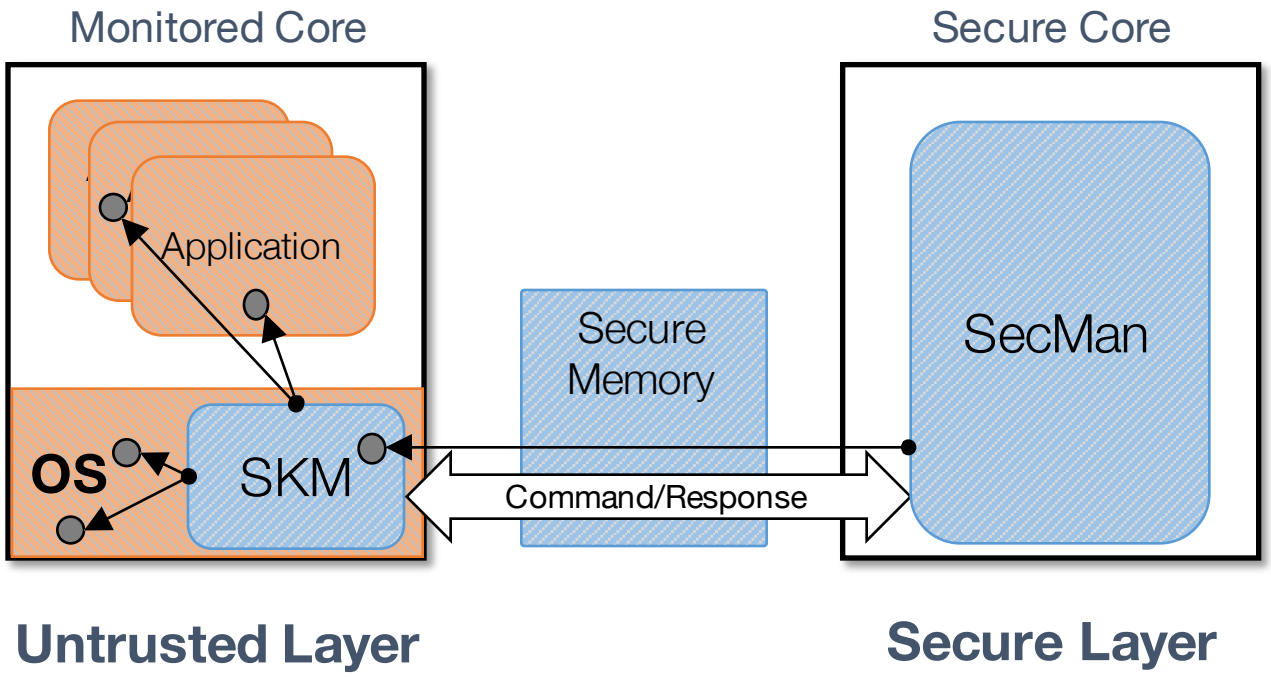
DragonBeam Framework



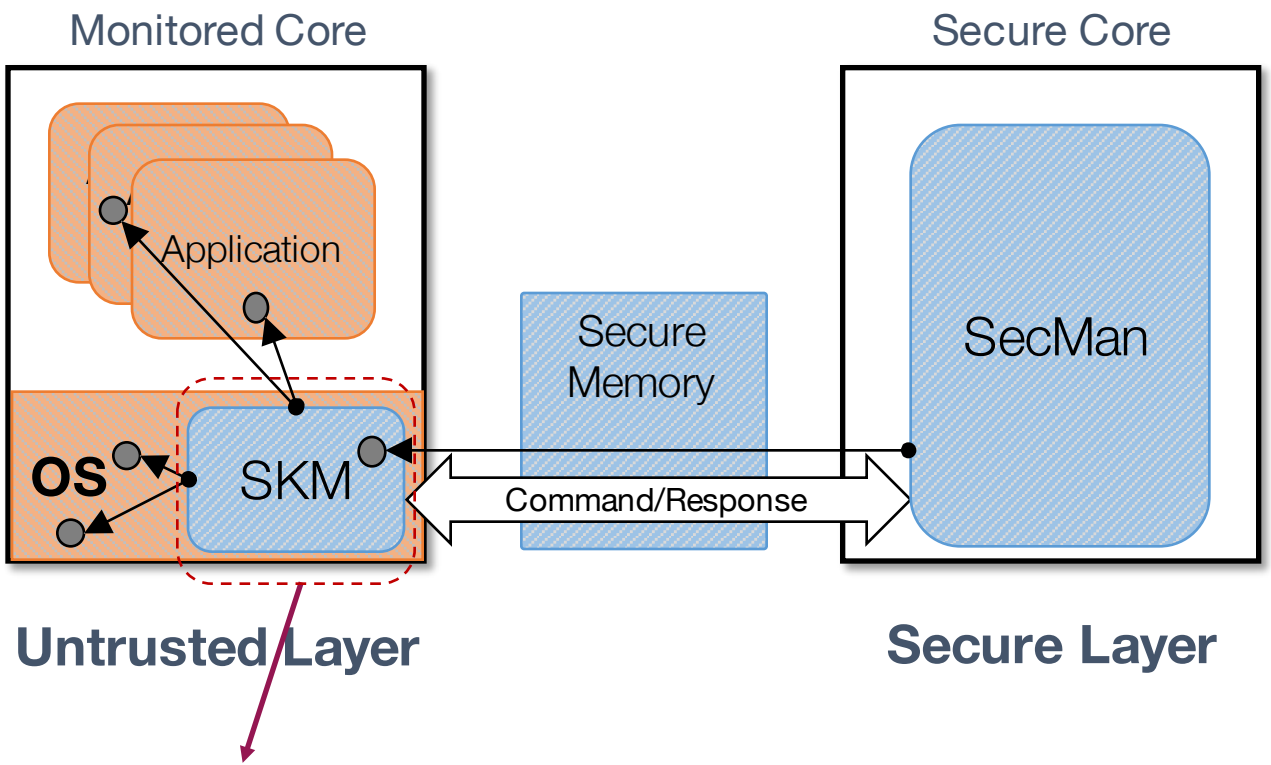
Untrusted Layer

Secure Layer

DragonBeam Framework



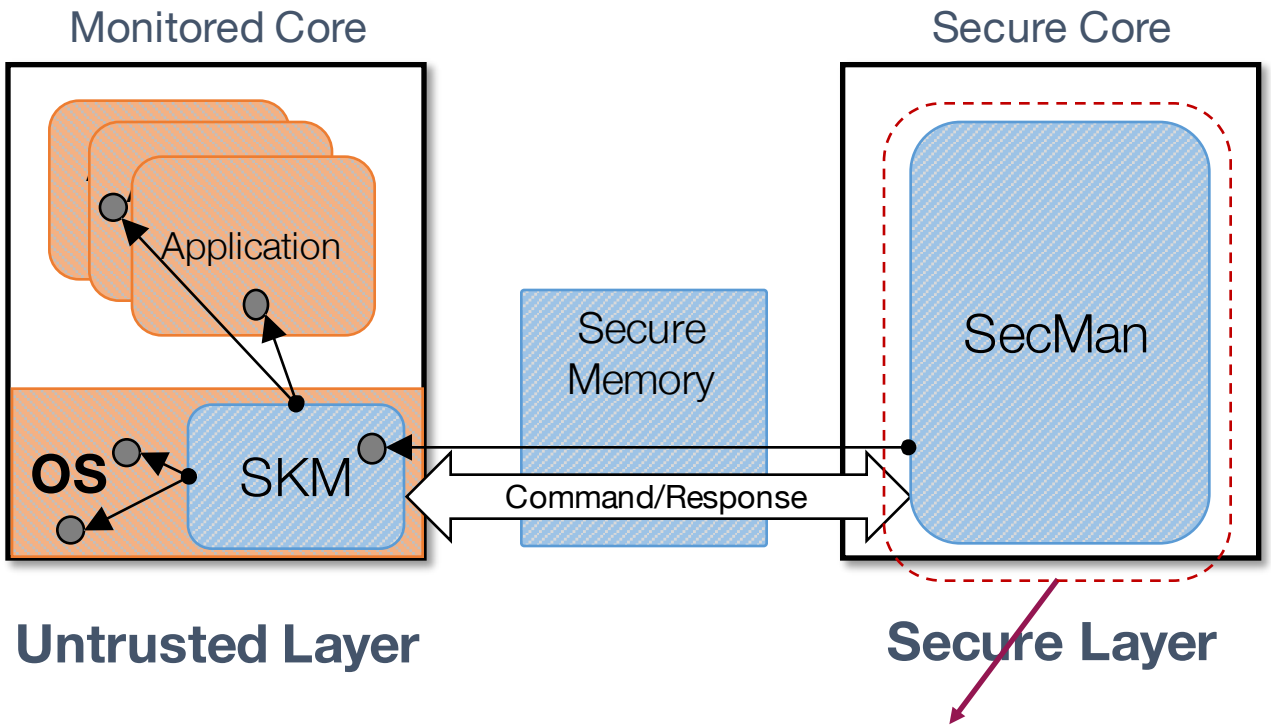
DragonBeam Framework



Secure Kernel Module

- Performs security monitoring operations
- Expands the observability
- Protected by Secure Core

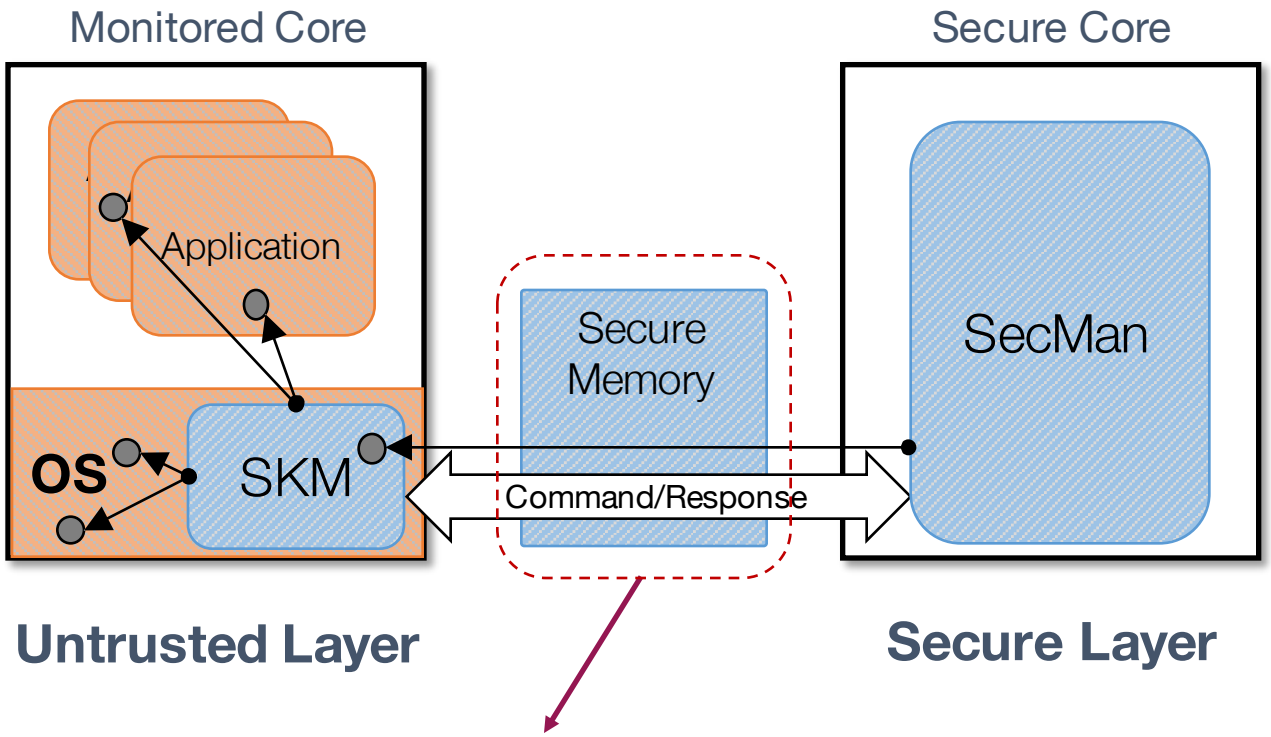
DragonBeam Framework



Secure Kernel Module Manager

- Commands SKM to perform security operations
- Analyzes monitored information
- Guarantees the integrity and the liveness of SKM

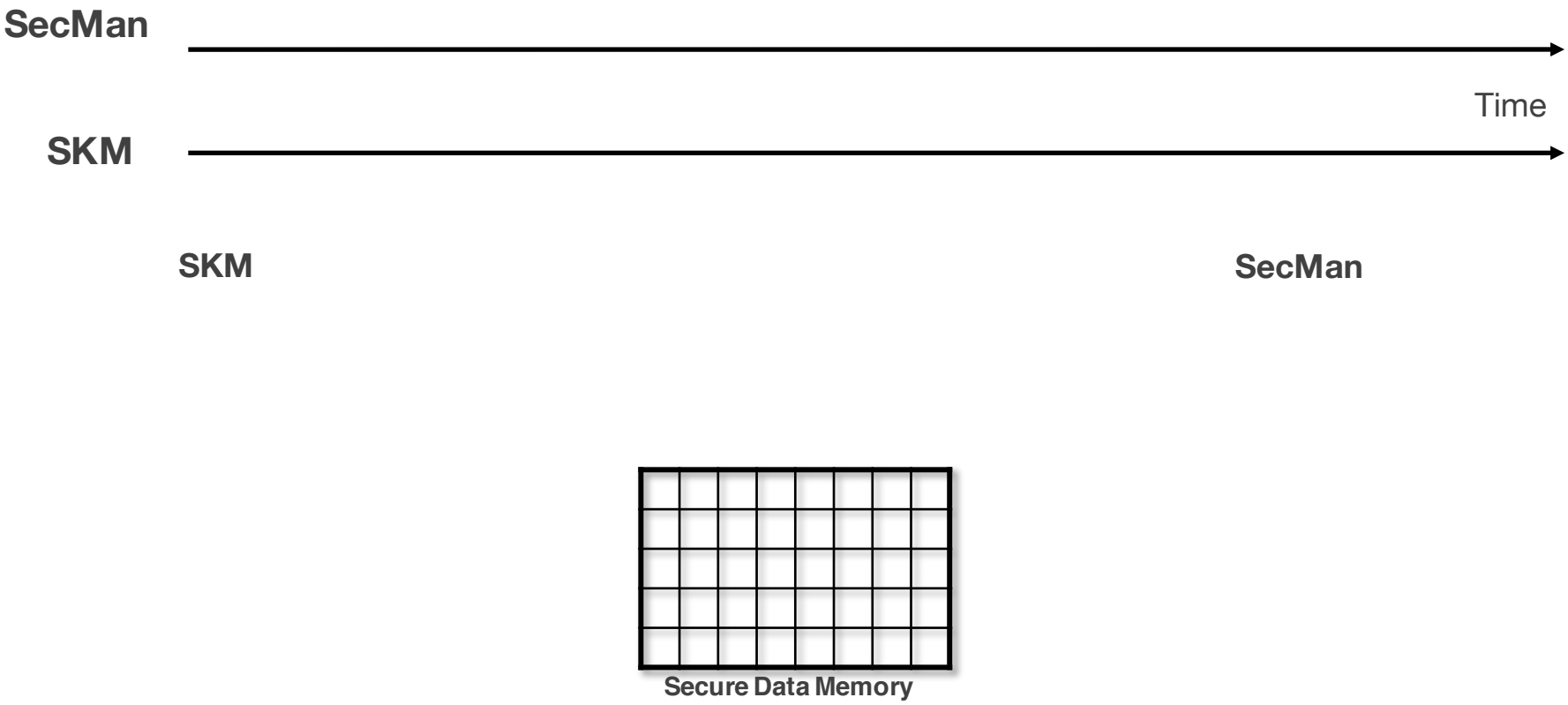
DragonBeam Framework



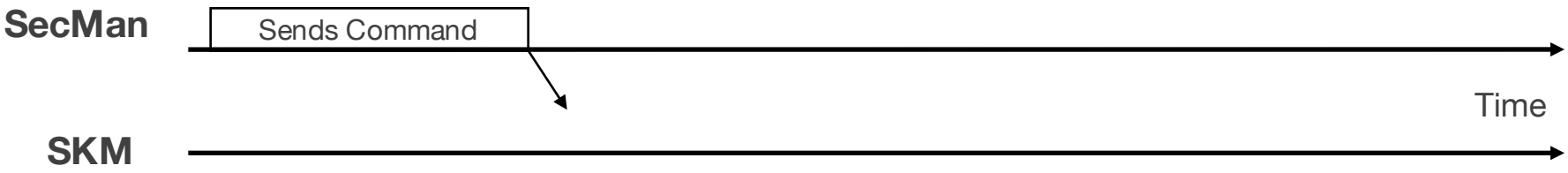
Secure Memory

- Secure communication channel between SKM and SecMan
  - Only accessible by SKM or Secure Core
- Also hosts SecMan code/data

Example Use Case



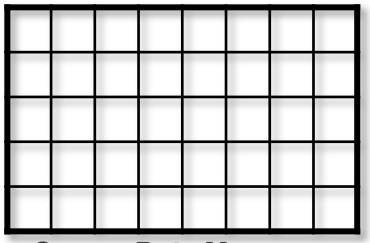
Example Use Case



SKM

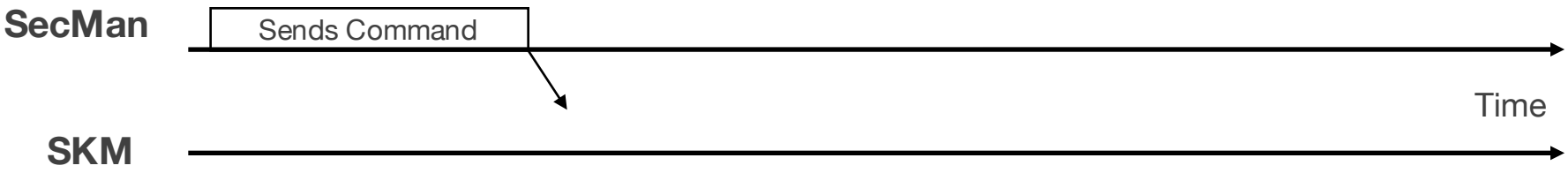
SecMan

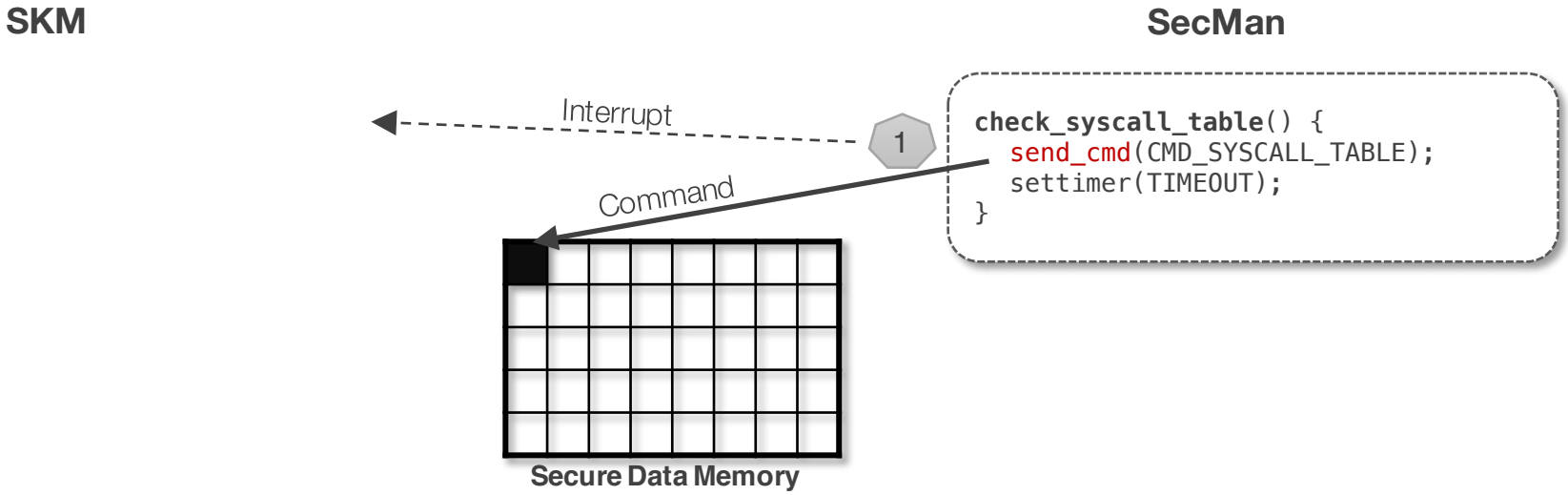
```
check_syscall_table() {
   send_cmd(CMD_SYSCALL_TABLE);
   settimer(TIMEOUT);
}
```



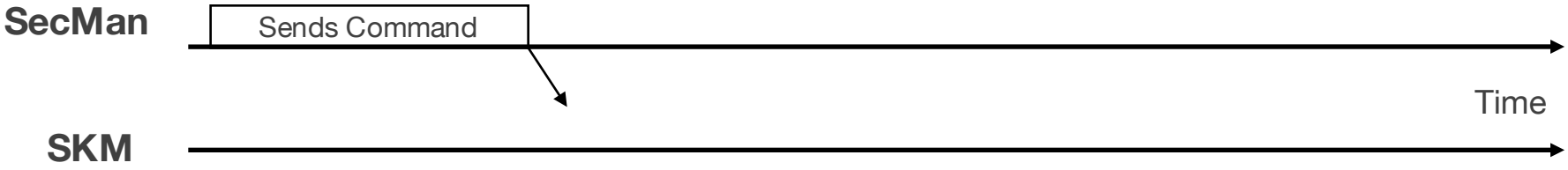
Secure Data Memory

Example Use Case





Example Use Case

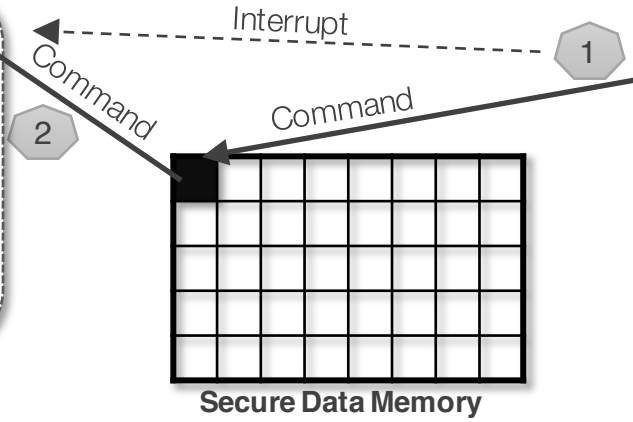


SKM

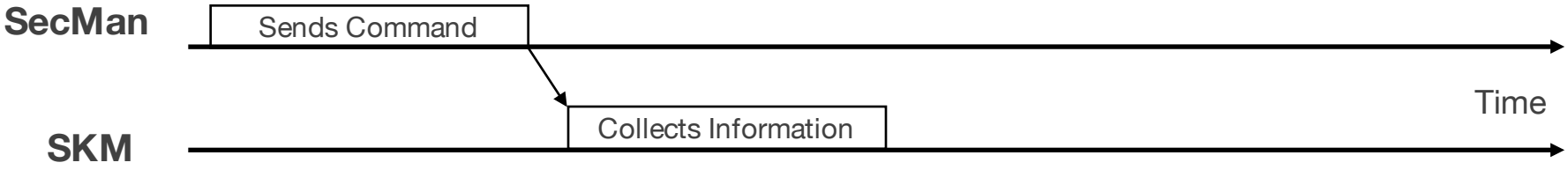
```
skm_ISR() {
   save sp;
   move sp to secure stack;
   switch (*CMD) {
      …
      case CMD_SYSCALL_TABLE:
         send_syscall_table();
         break;
      …
   }
   restore sp;
}
```

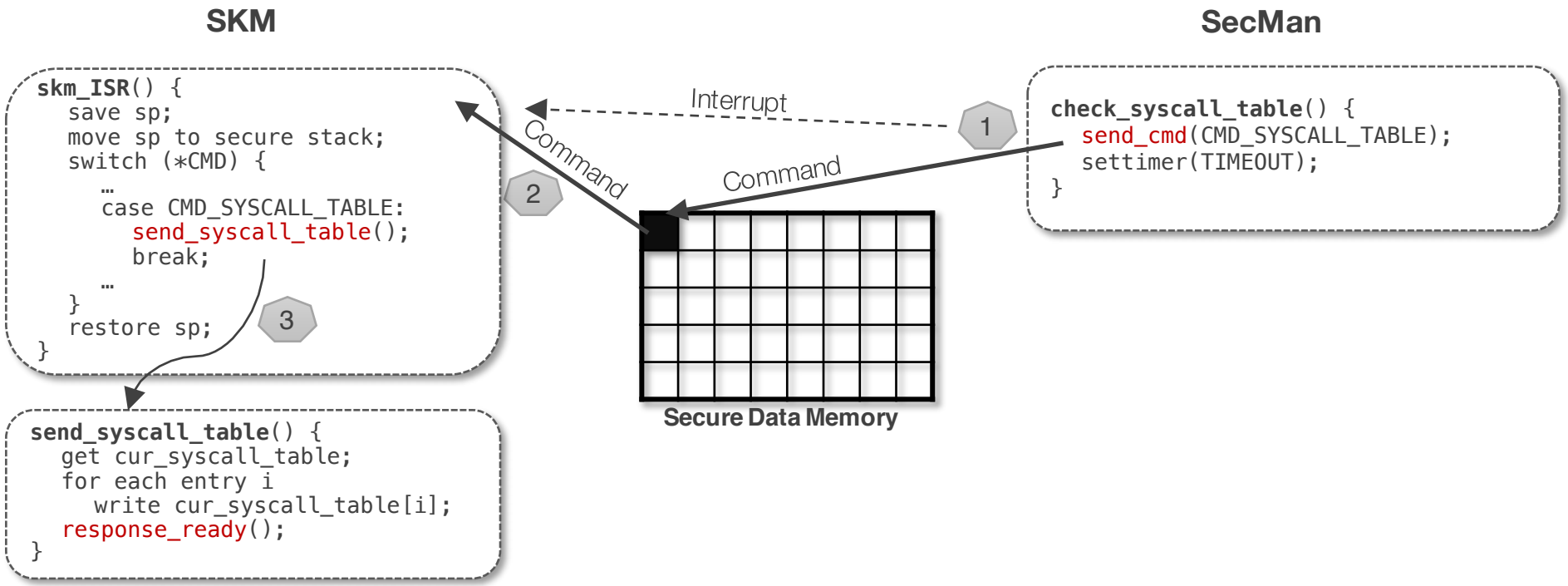
SecMan

```
check_syscall_table() {
   send_cmd(CMD_SYSCALL_TABLE);
   settimer(TIMEOUT);
}
```

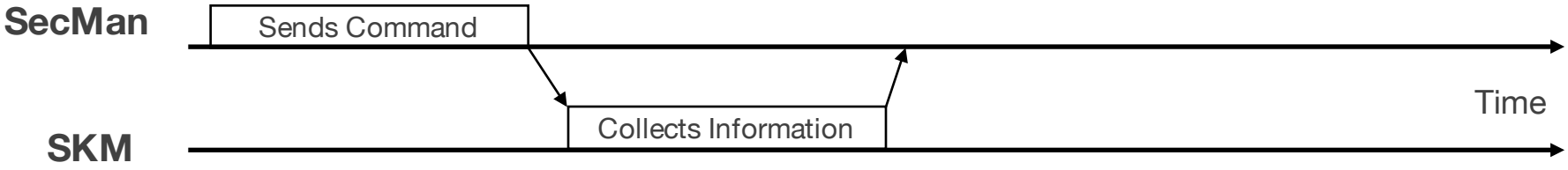


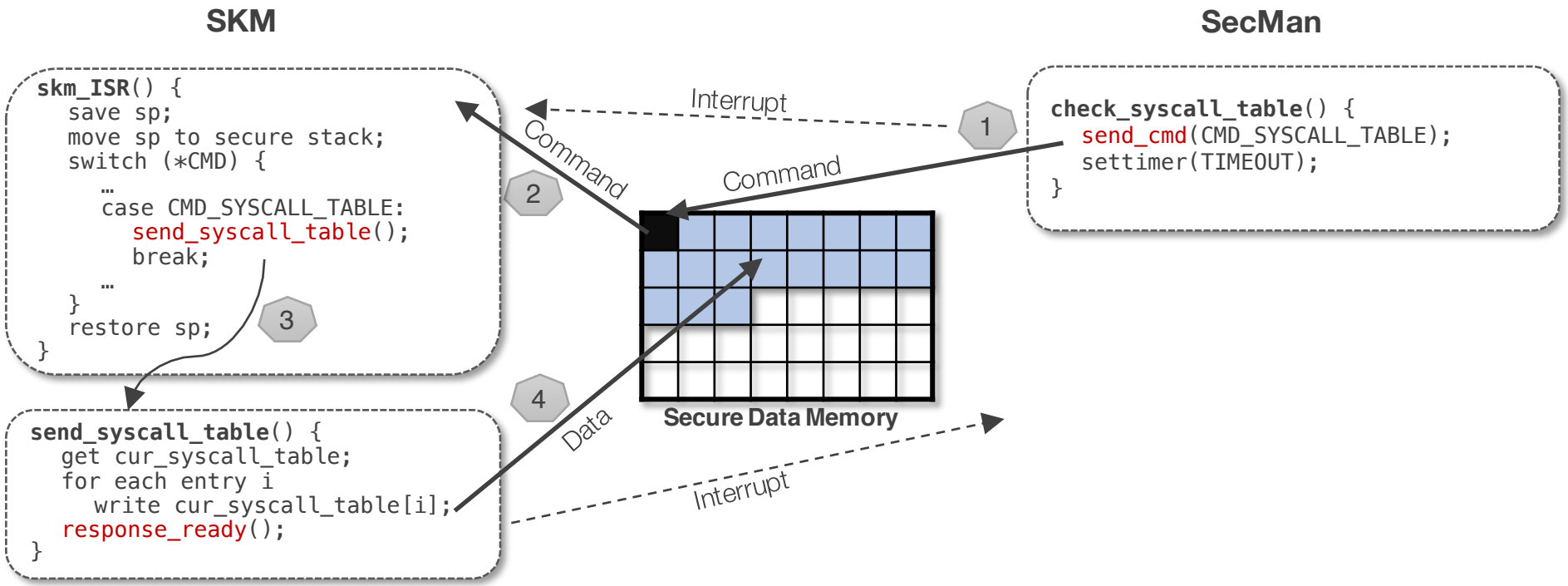
Example Use Case



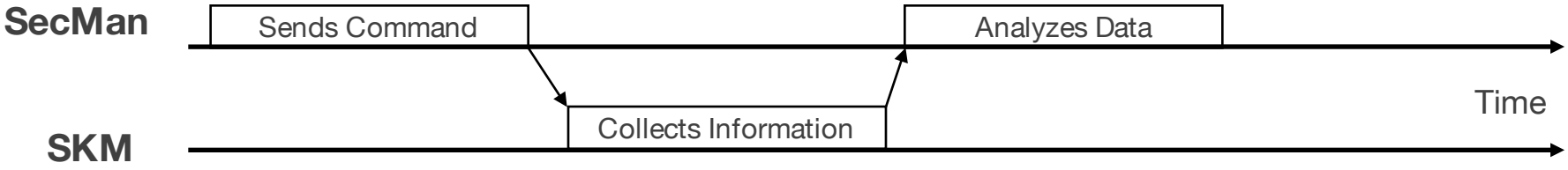


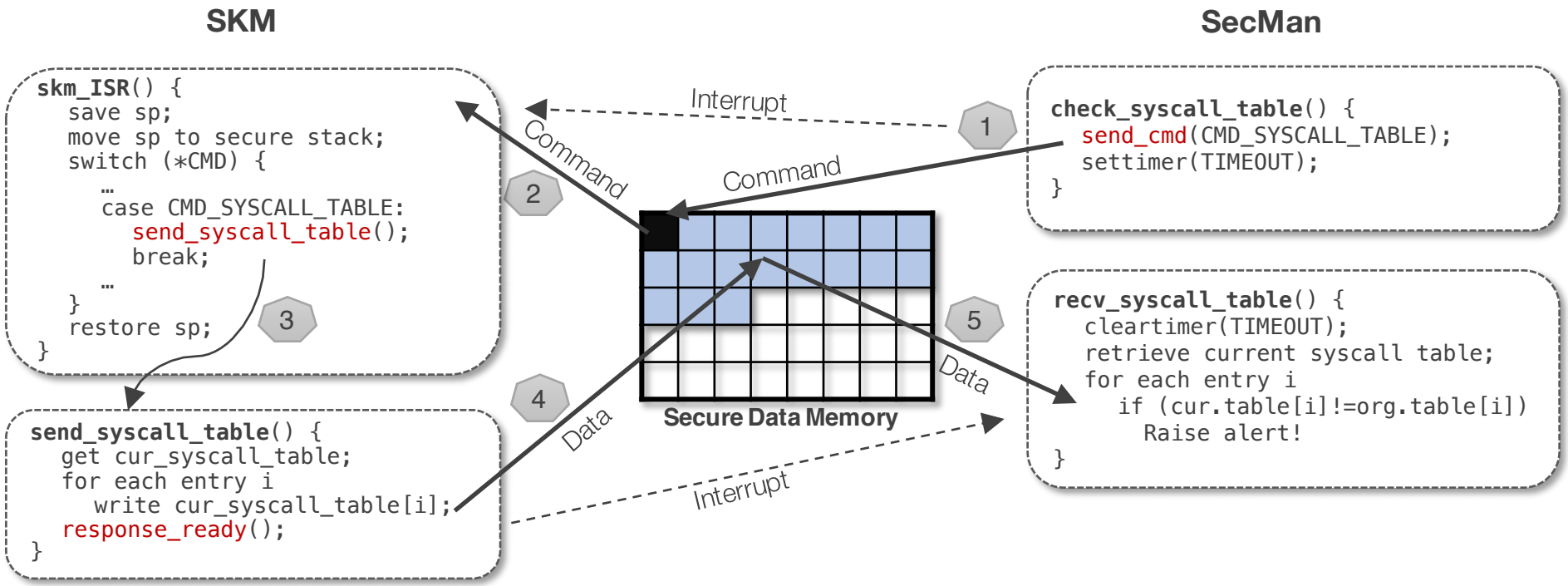
Example Use Case



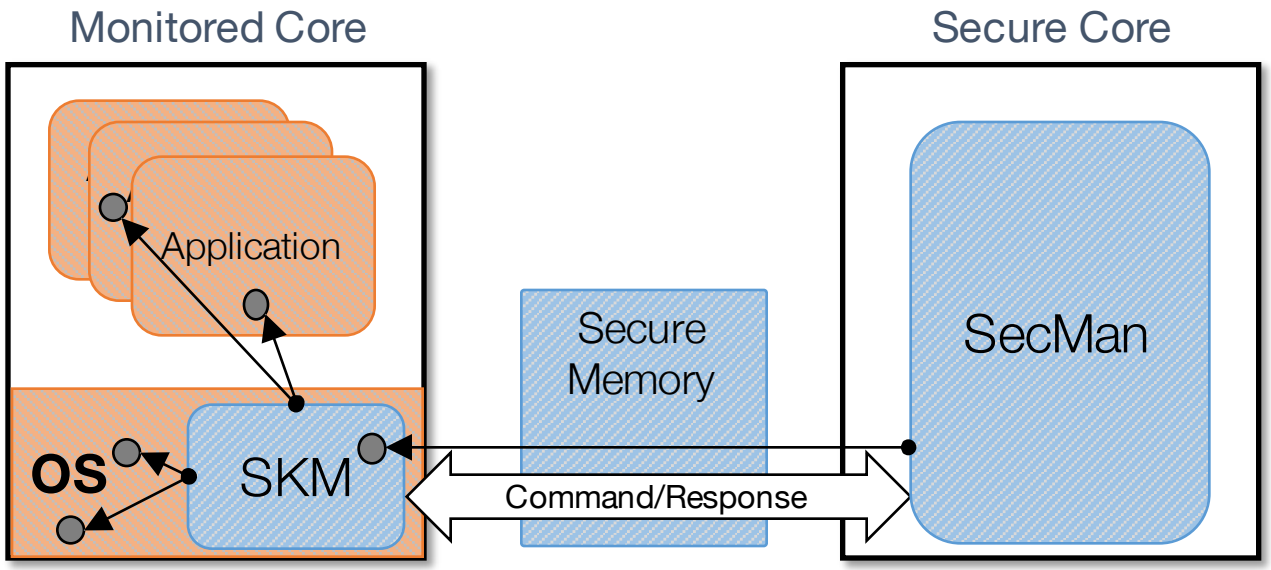


Example Use Case





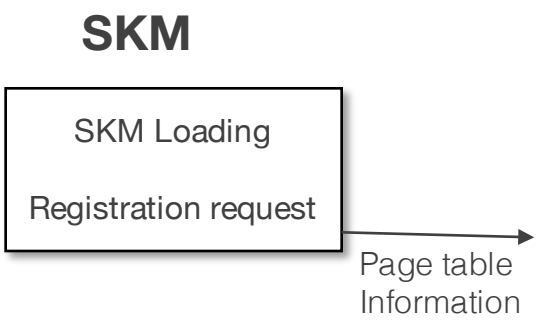
Challenges

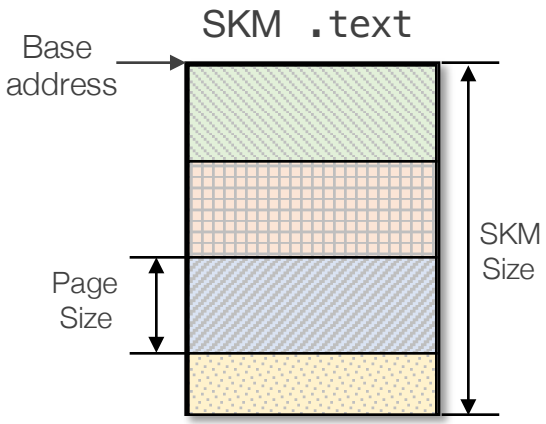


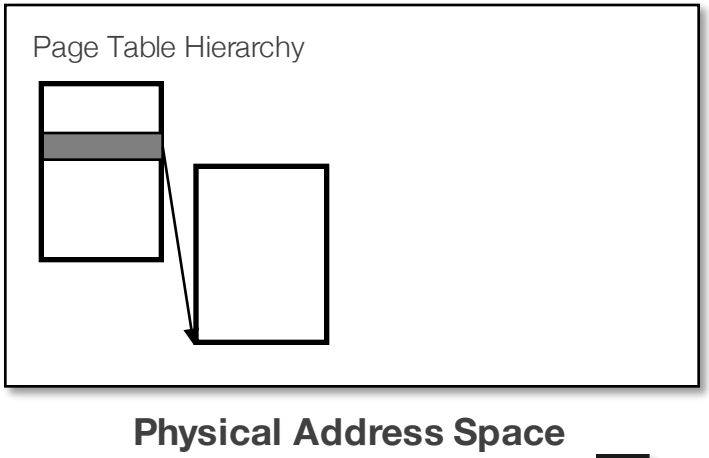
- SKM identification
- Secure memory access control
- SKM integrity and liveness guarantee

SKM Registration

- **Requested by SKM, verified by SecMan**
  - Calculates a hash of SKM's code
  - Directly from physical frames

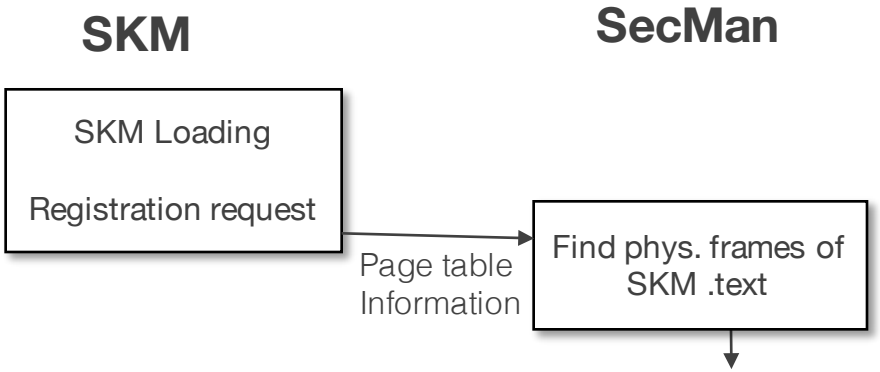


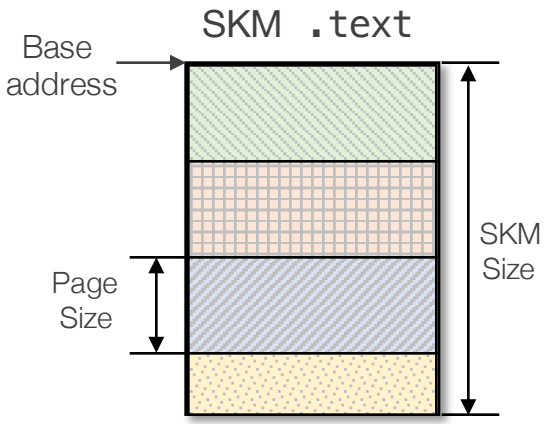


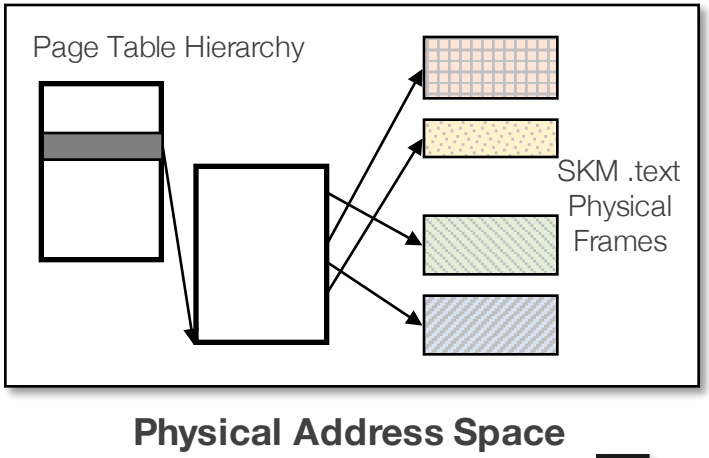


SKM Registration

- **Requested by SKM, verified by SecMan**
  - Calculates a hash of SKM's code
  - Directly from physical frames

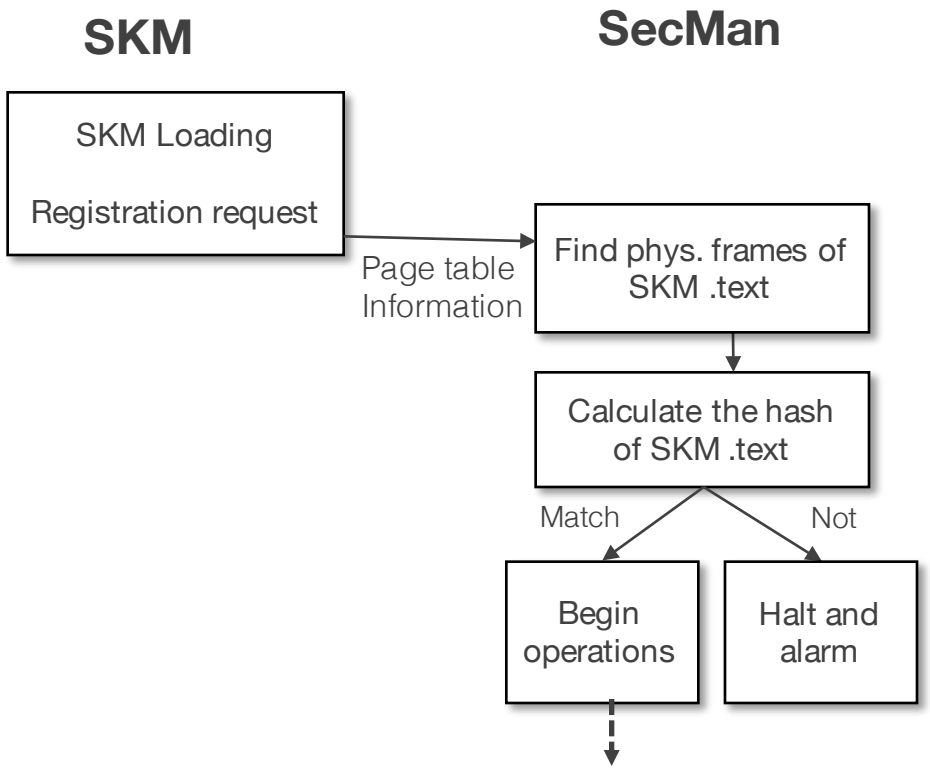


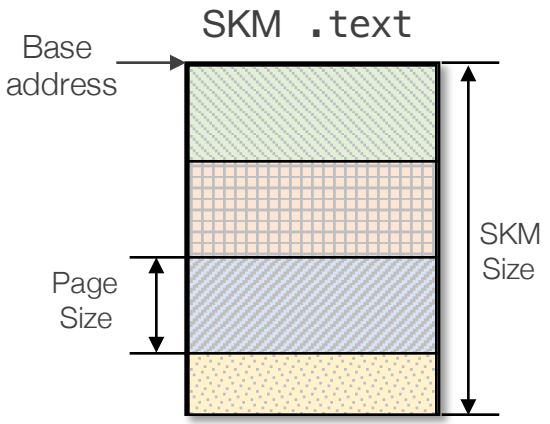


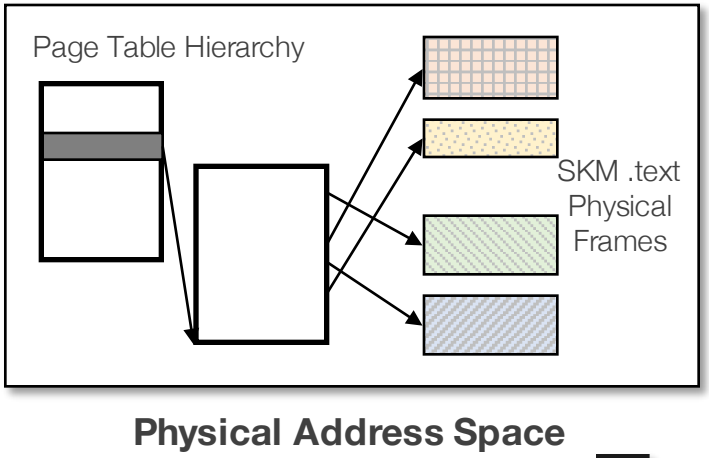


SKM Registration

- **Requested by SKM, verified by SecMan**
  - Calculates a hash of SKM's code
  - Directly from physical frames

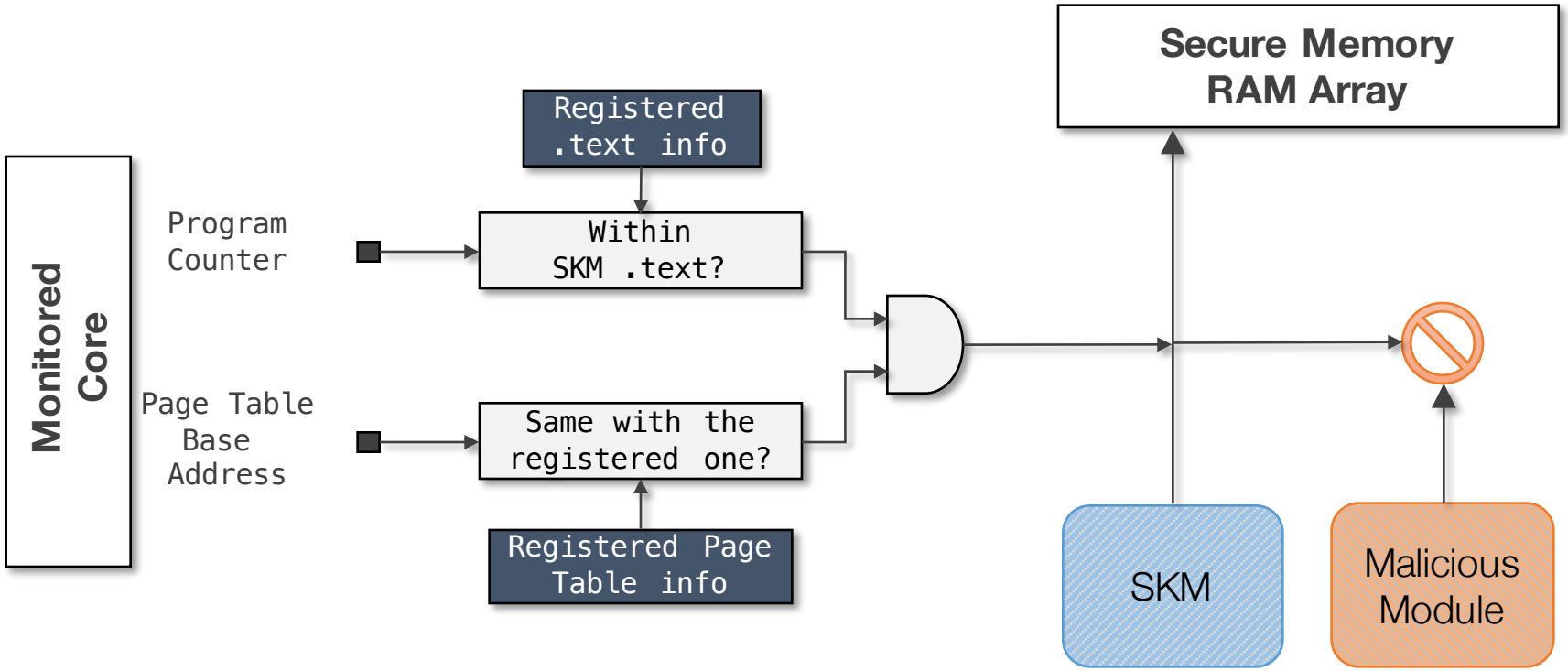






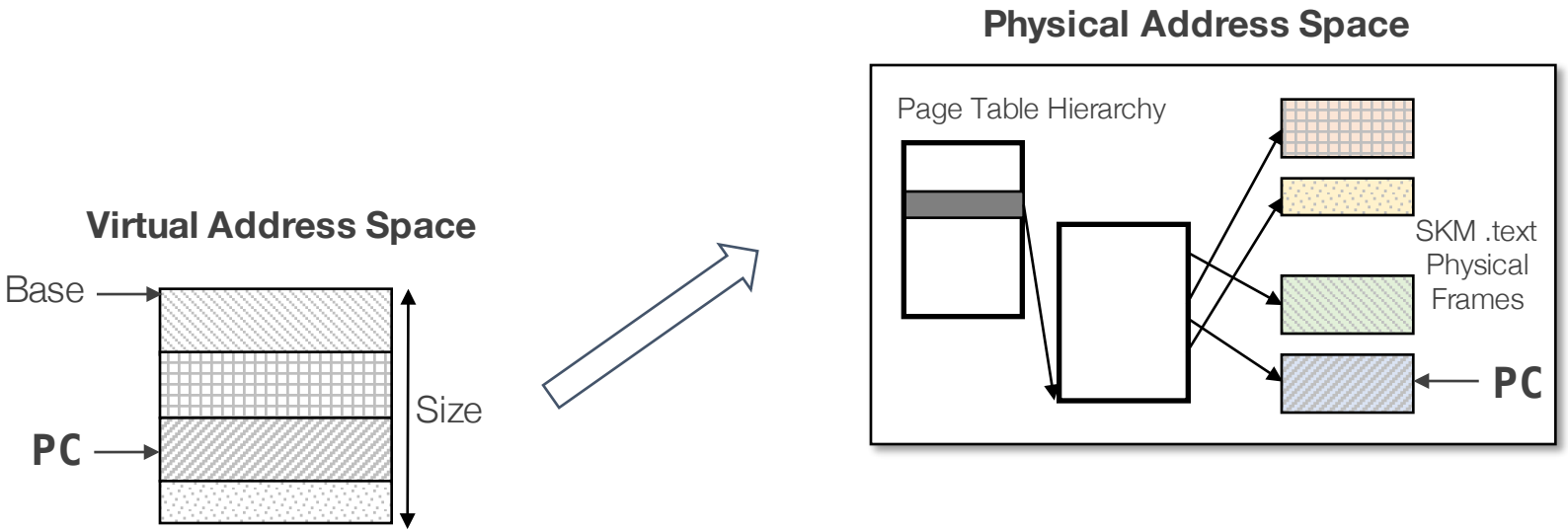
Secure Memory Access Control

- **Who initiated memory transaction?**
  - Use the current program counter and page mapping information



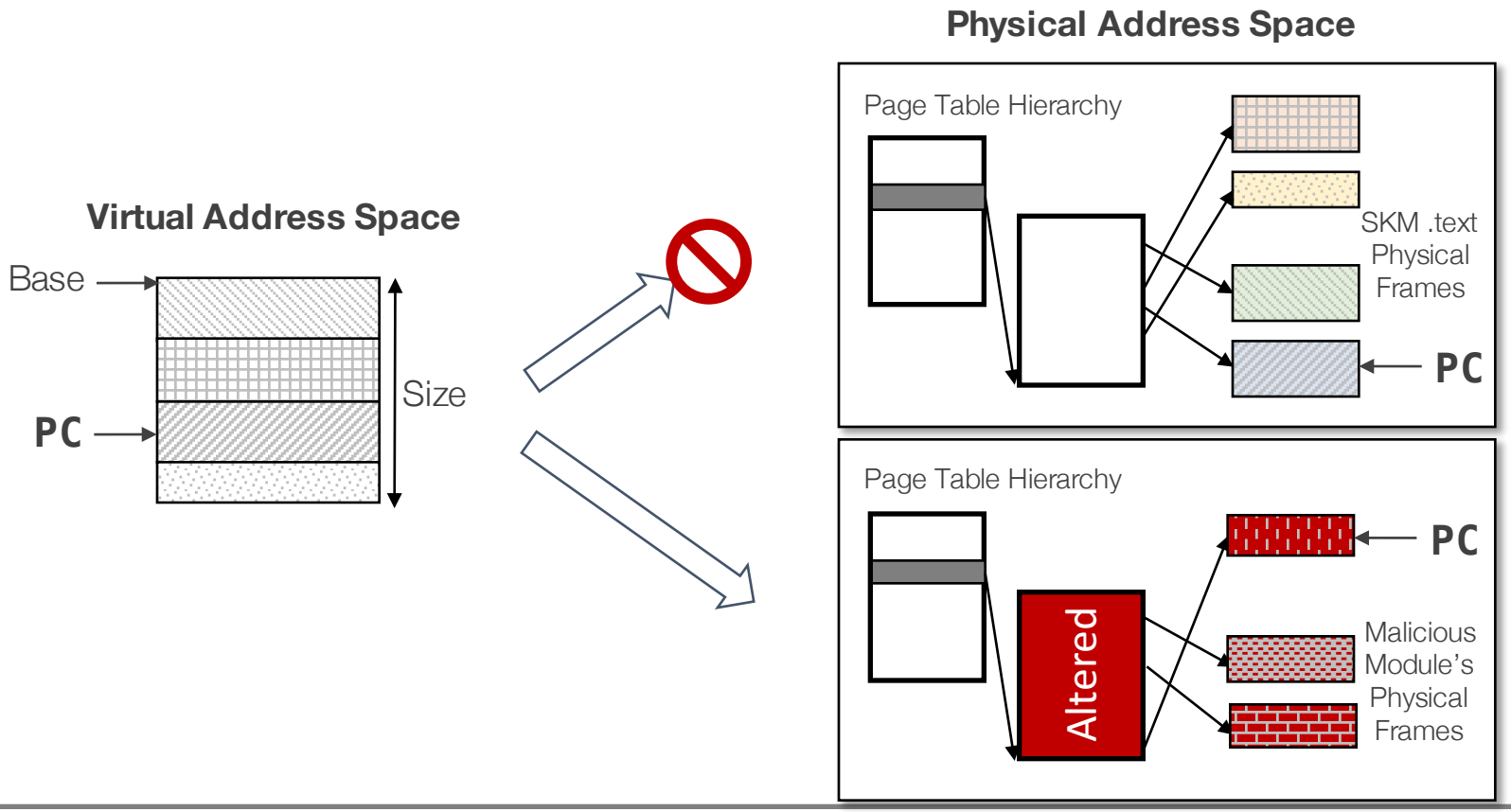
Secure Memory Access Control

- **What if attacker modifies SKM's page mapping?**



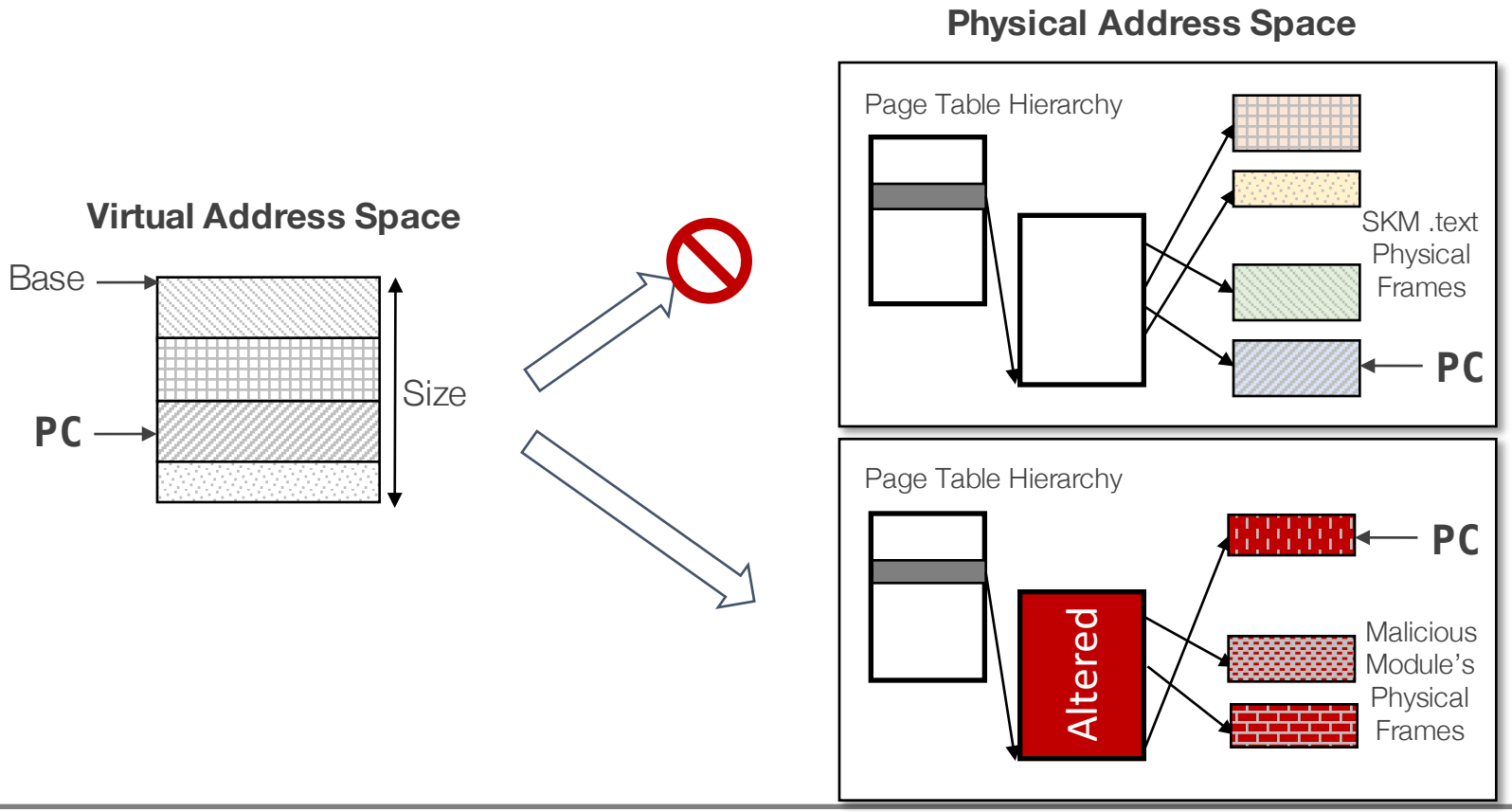
Secure Memory Access Control

- What if attacker modifies SKM's page mapping?



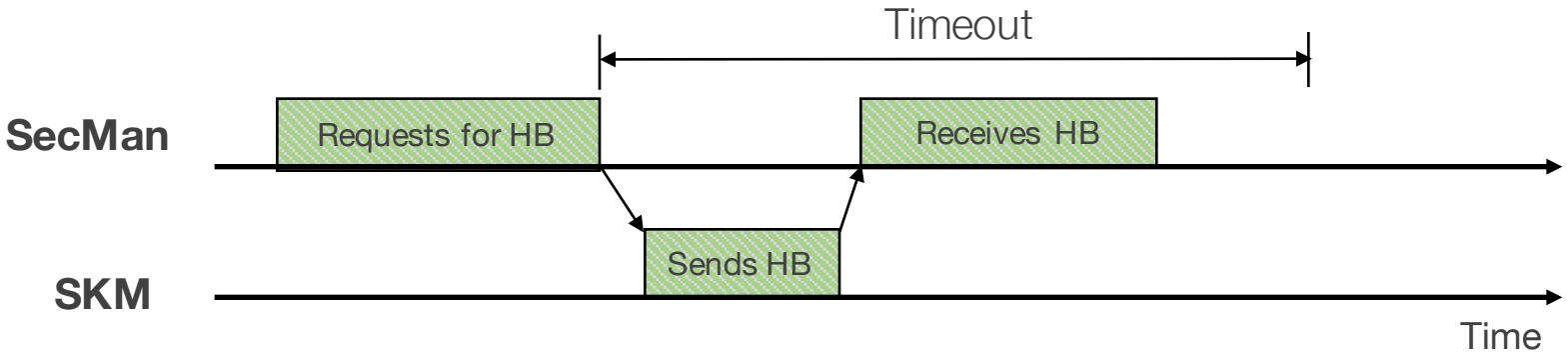
Secure Memory Access Control

- **What if attacker modifies SKM's page mapping?**
  - **Solution**: Regularly translate virt-to-phys address and verifies SKM .text hash



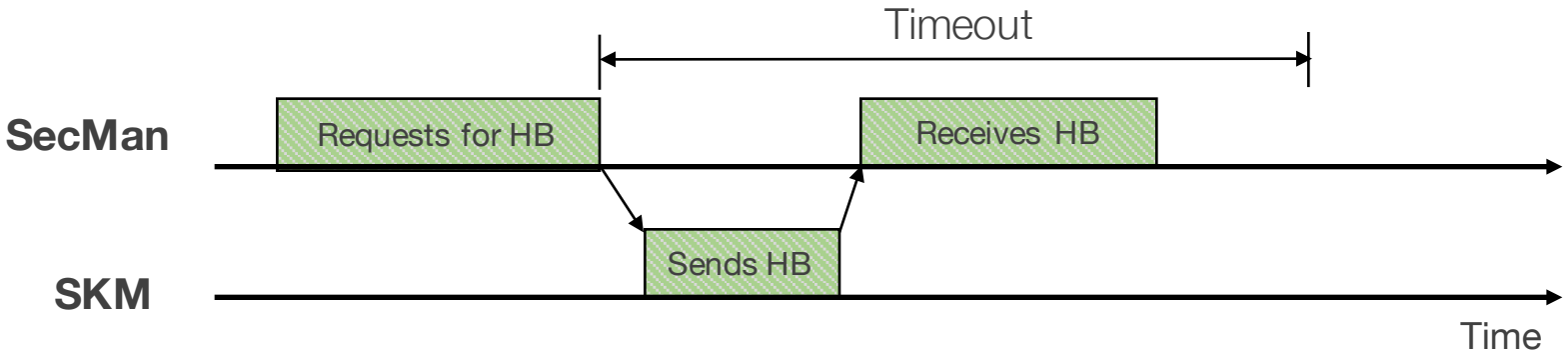
Heartbeat and Hashing

- **Heartbeat**
  - Checks if SKM is alive
  - Only SKM can respond



Heartbeat and Hashing

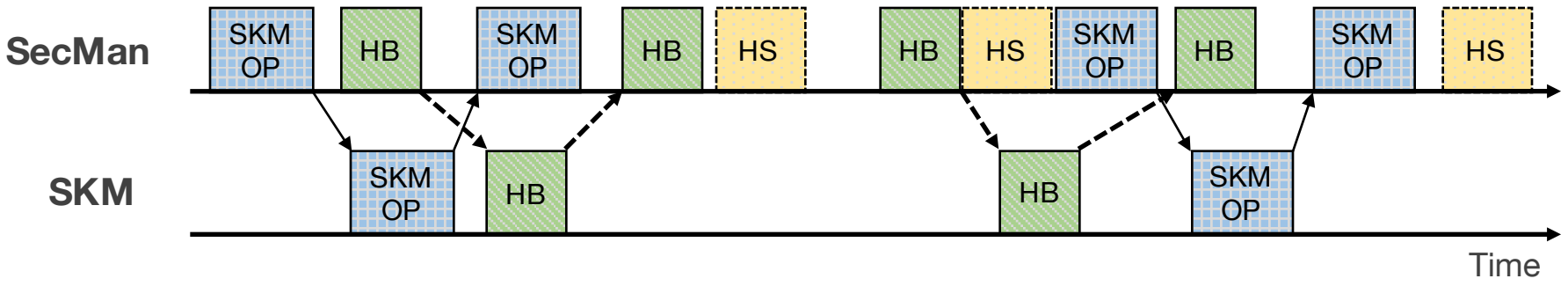
- **Heartbeat**
  - Checks if SKM is alive
  - Only SKM can respond



- **SKM .text hashing**
  - Checks if SKM's code and page mapping have not been altered

Random Check Intervals

- To prevent TOCTTOU (Time Of Check To Time Of Use) attacks
- Attacker *cannot guess* the pattern of checks

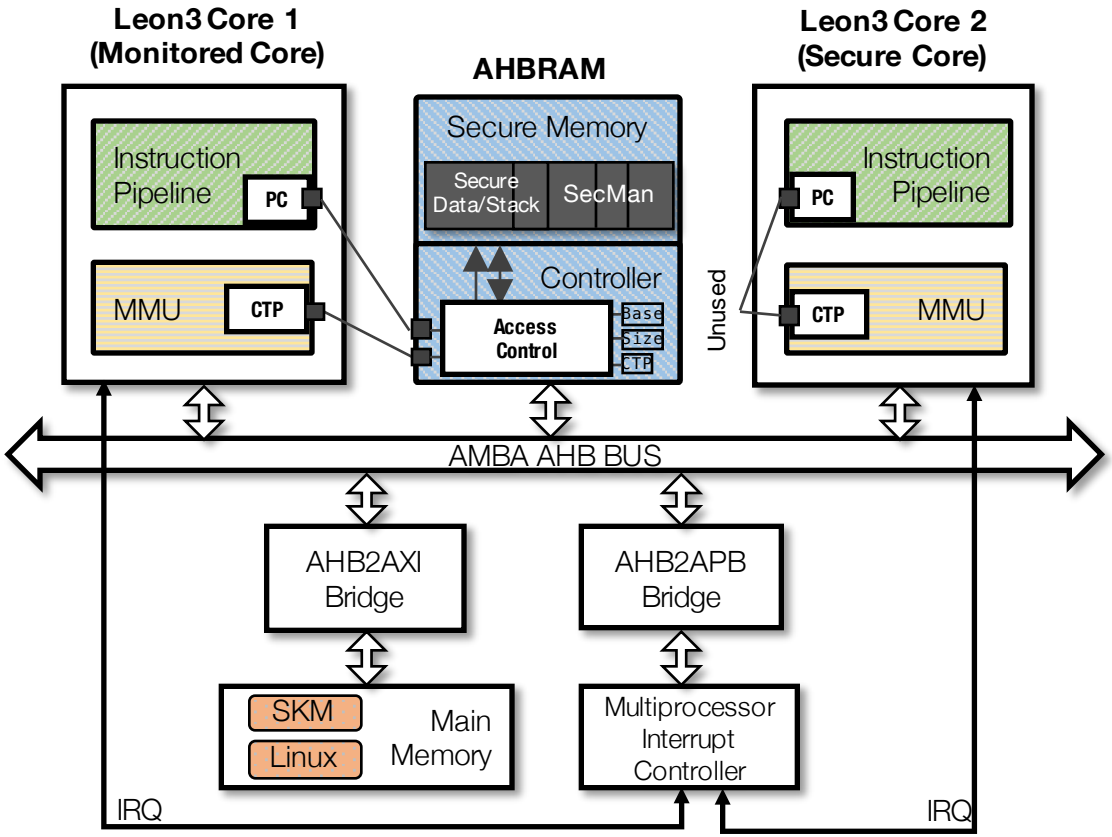


SKM OP **SKM Operation** (Send/Response/Analysis)

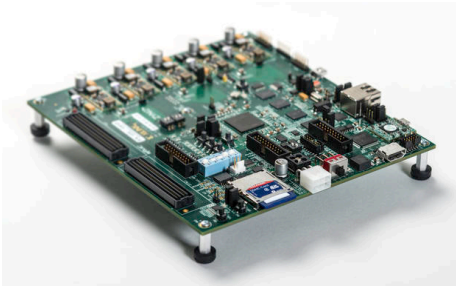
HB **Heartbeat** (Request/Send/Receive)

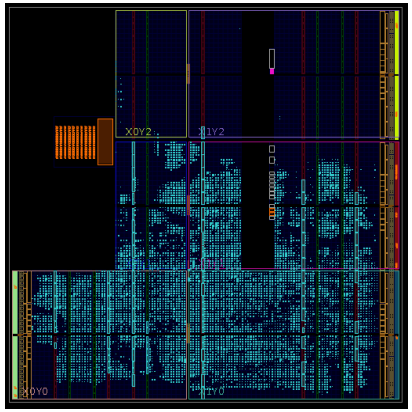
HS **Hashing** SKM .text

Implementation

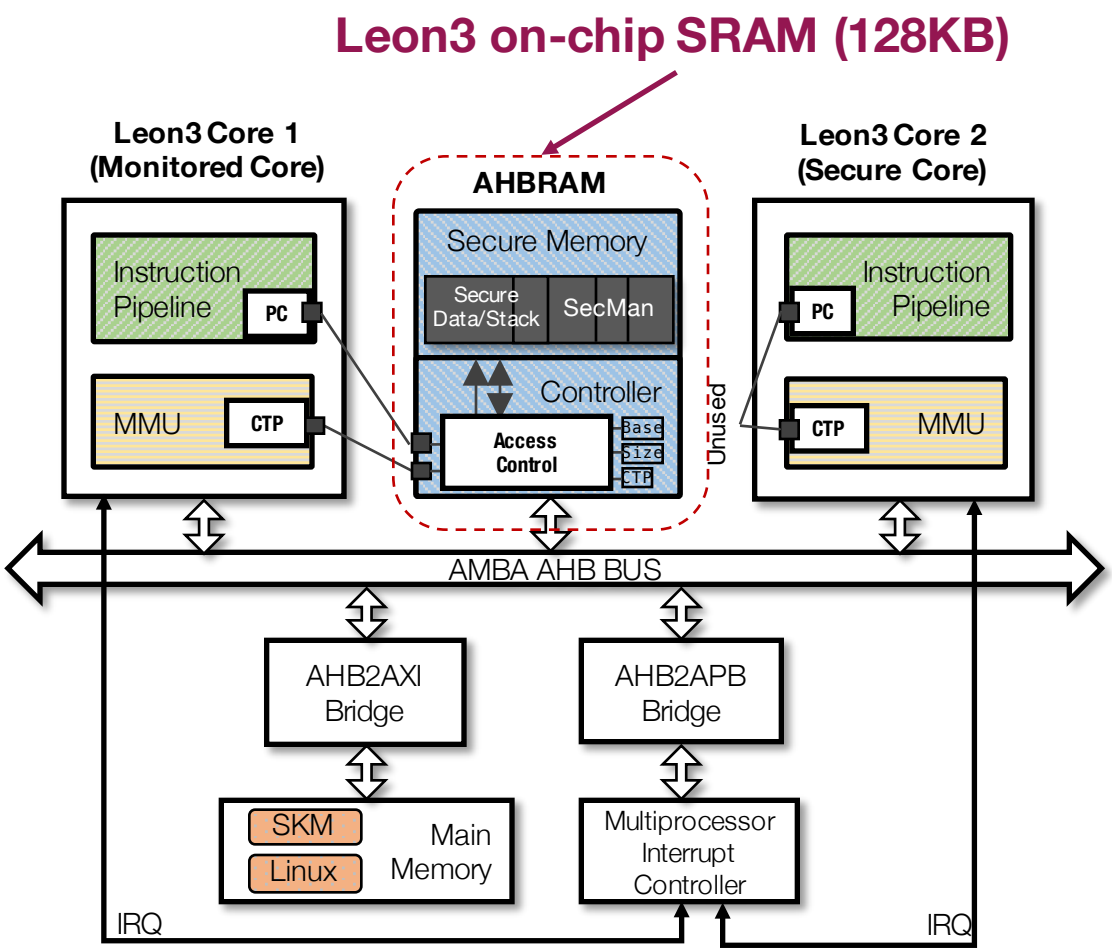


- Leon3 processor on Xilinx ZC702 FPGA
  - SPARC V8, soft-core
  - 83.3 MHz
  - 256 MB

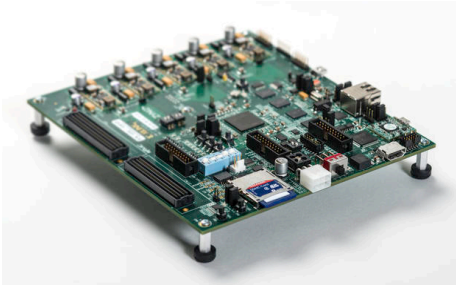


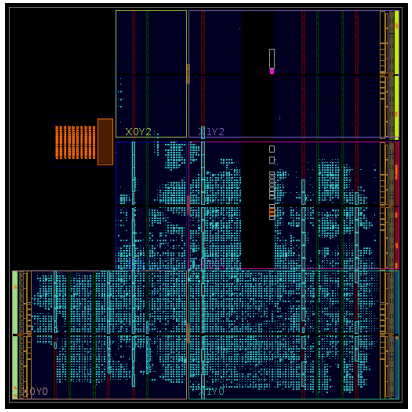


Implementation

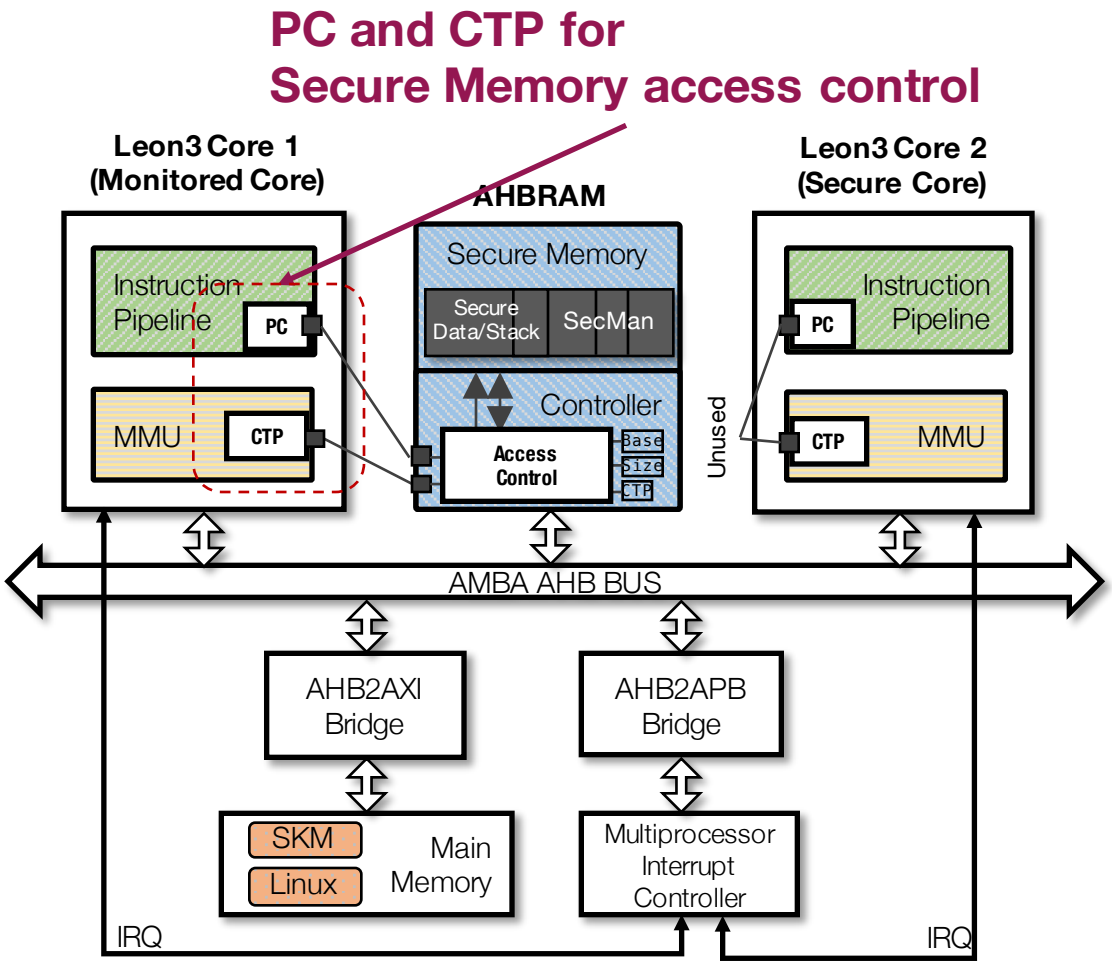


- Leon3 processor on Xilinx ZC702 FPGA
  - SPARC V8, soft-core
  - 83.3 MHz
  - 256 MB

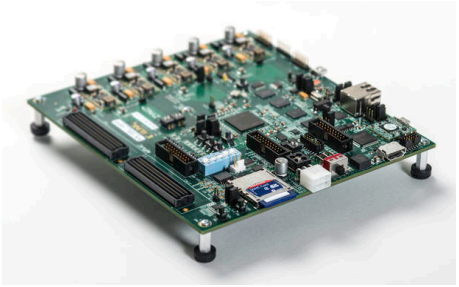


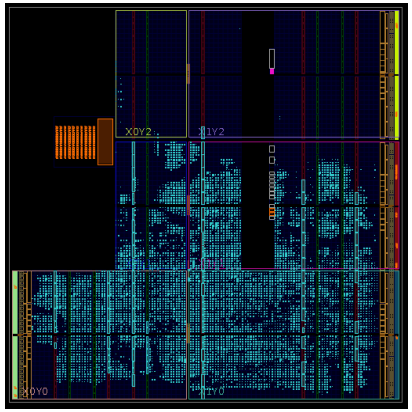


Implementation

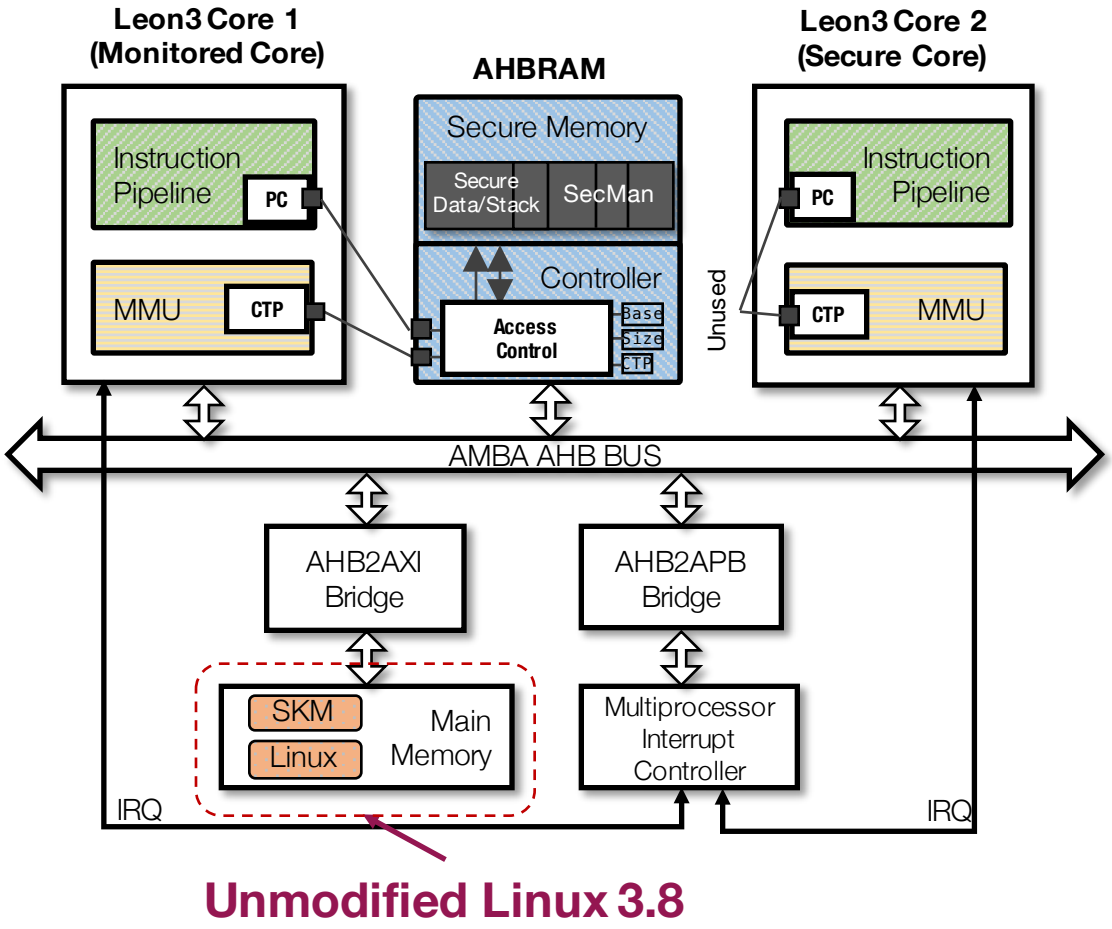


- Leon3 processor on Xilinx ZC702 FPGA
  - SPARC V8, soft-core
  - 83.3 MHz
  - 256 MB

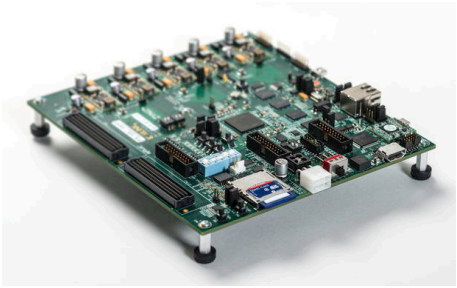


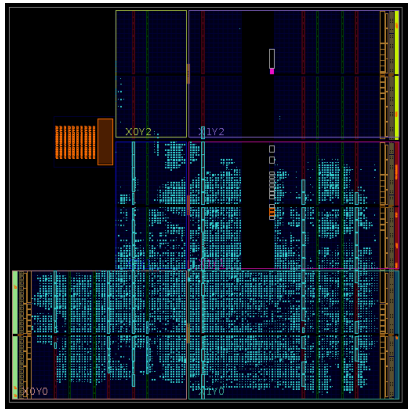


Implementation



- Leon3 processor on Xilinx ZC702 FPGA
  - SPARC V8, soft-core
  - 83.3 MHz
  - 256 MB



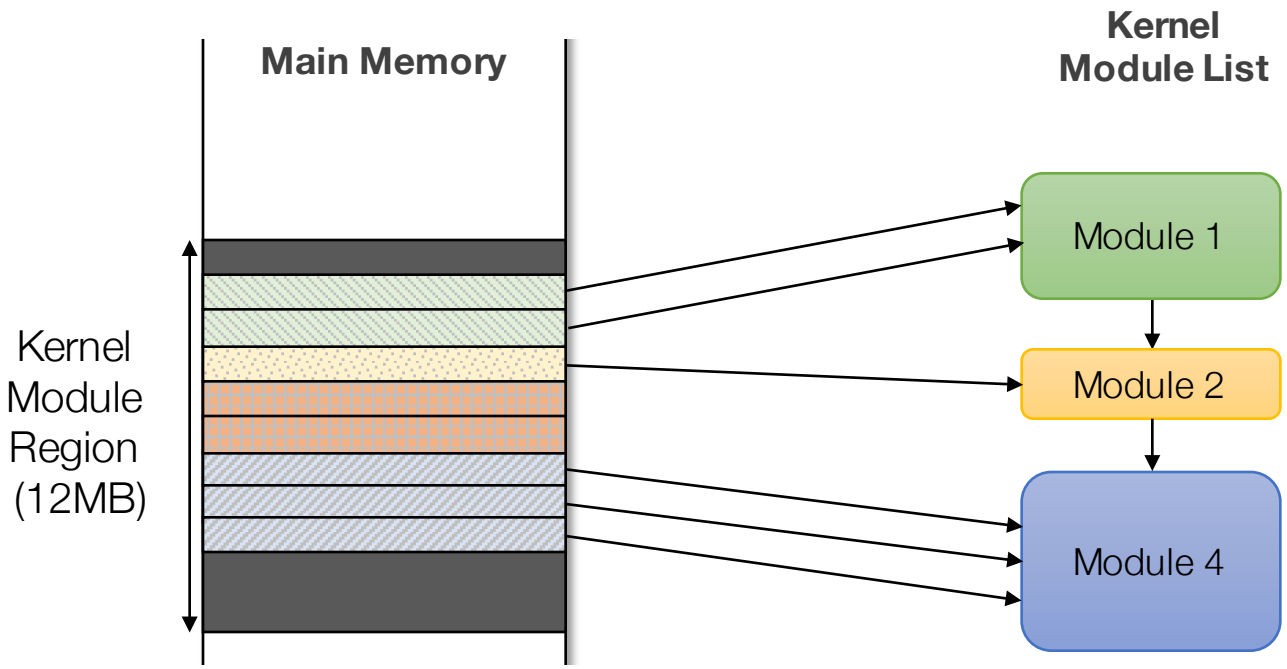


Evaluation – Use Cases

1) System call table integrity check
2) Hidden module detection

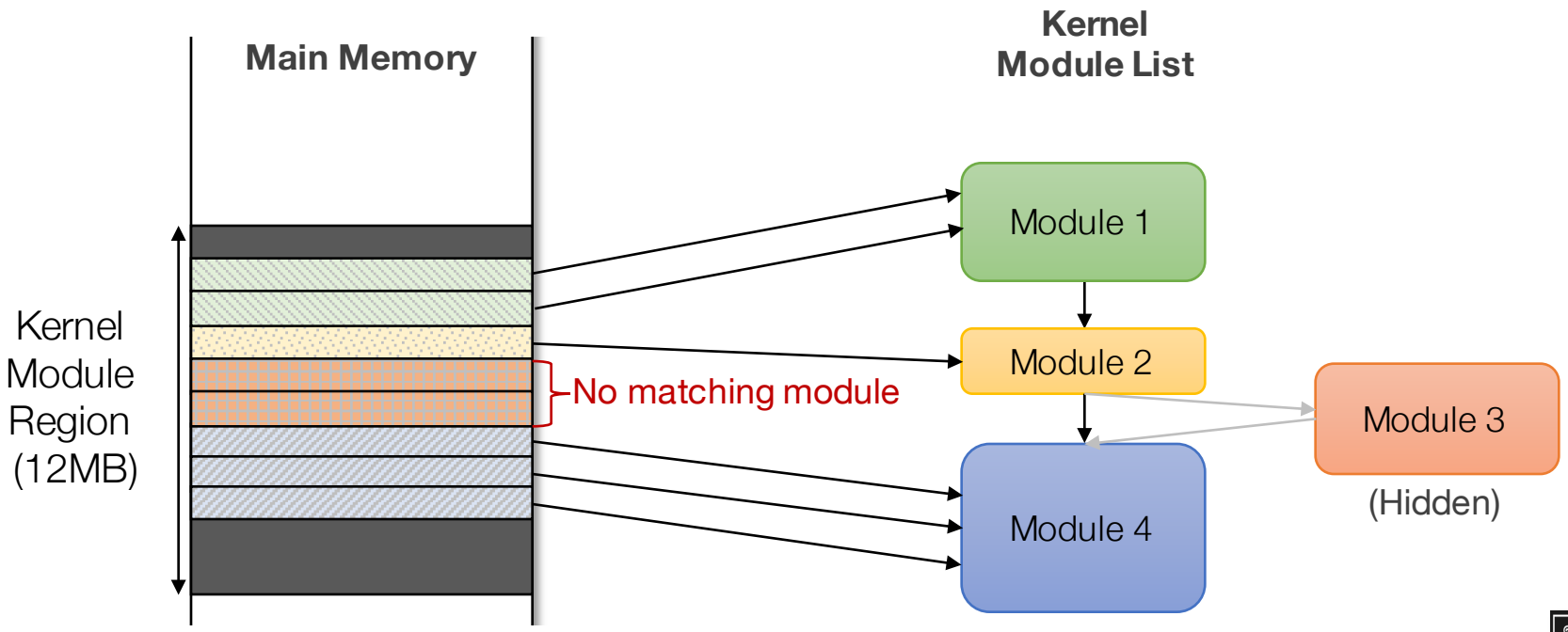
Evaluation – Use Cases

1) System call table integrity check
2) Hidden module detection



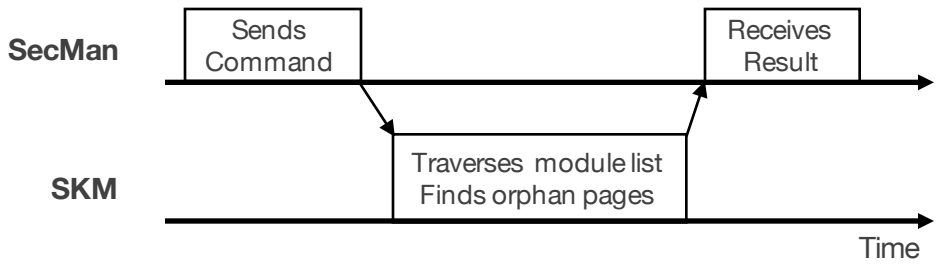
Evaluation – Use Cases

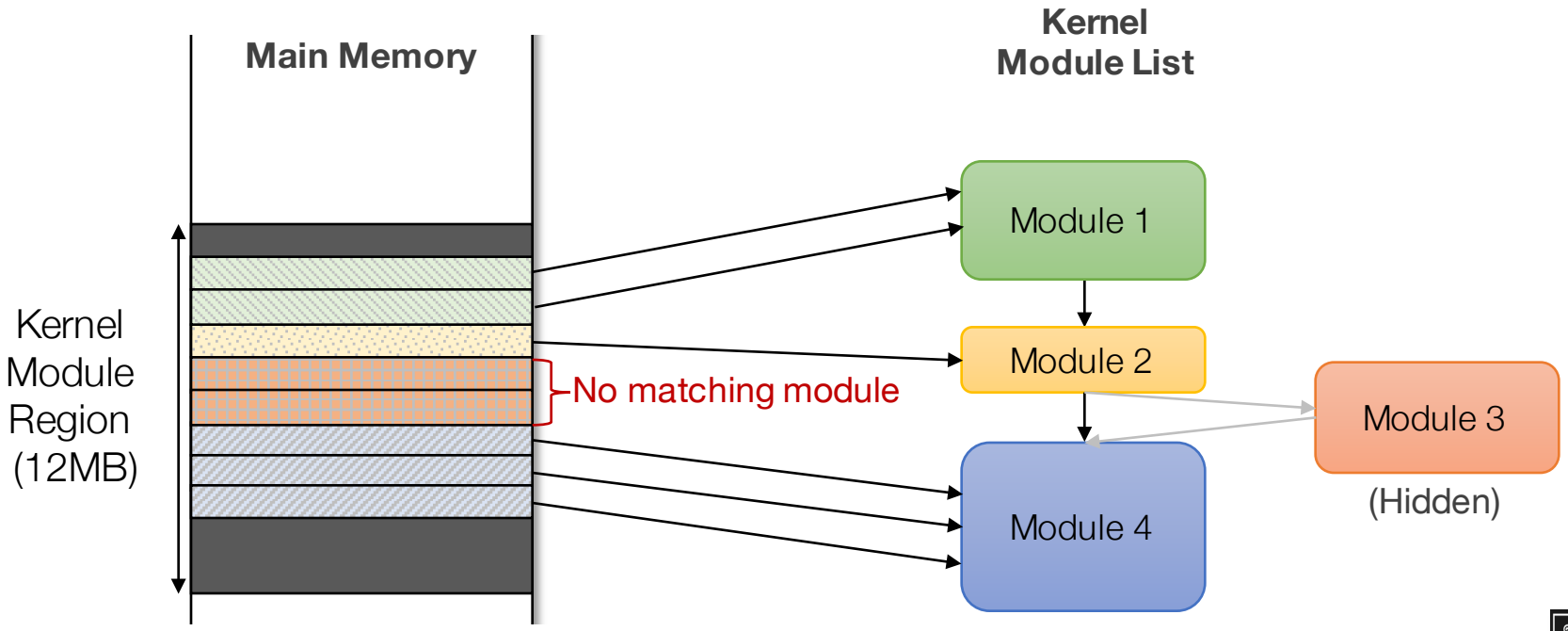
1) System call table integrity check
2) Hidden module detection



Evaluation – Use Cases

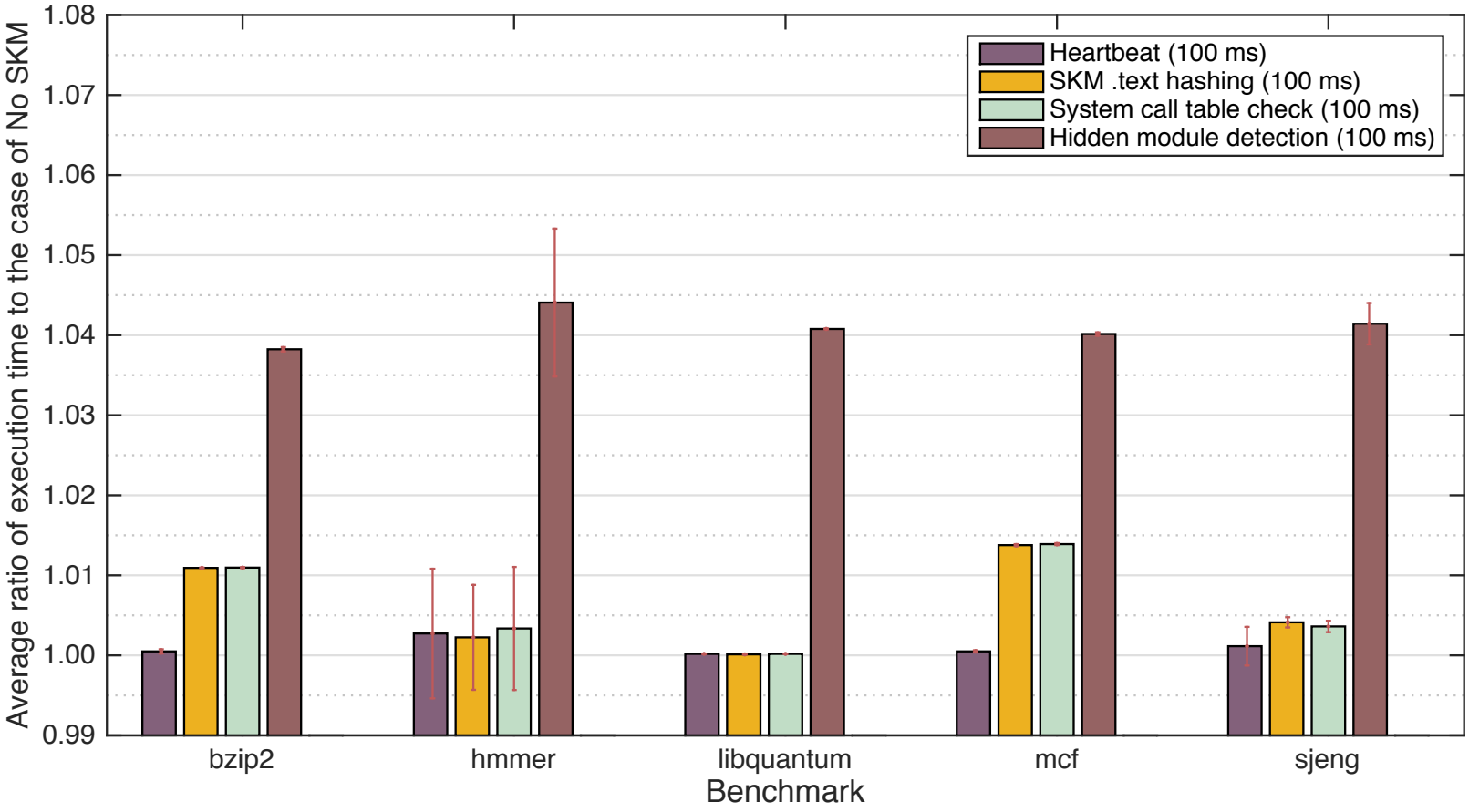
1) System call table integrity check
2) Hidden module detection





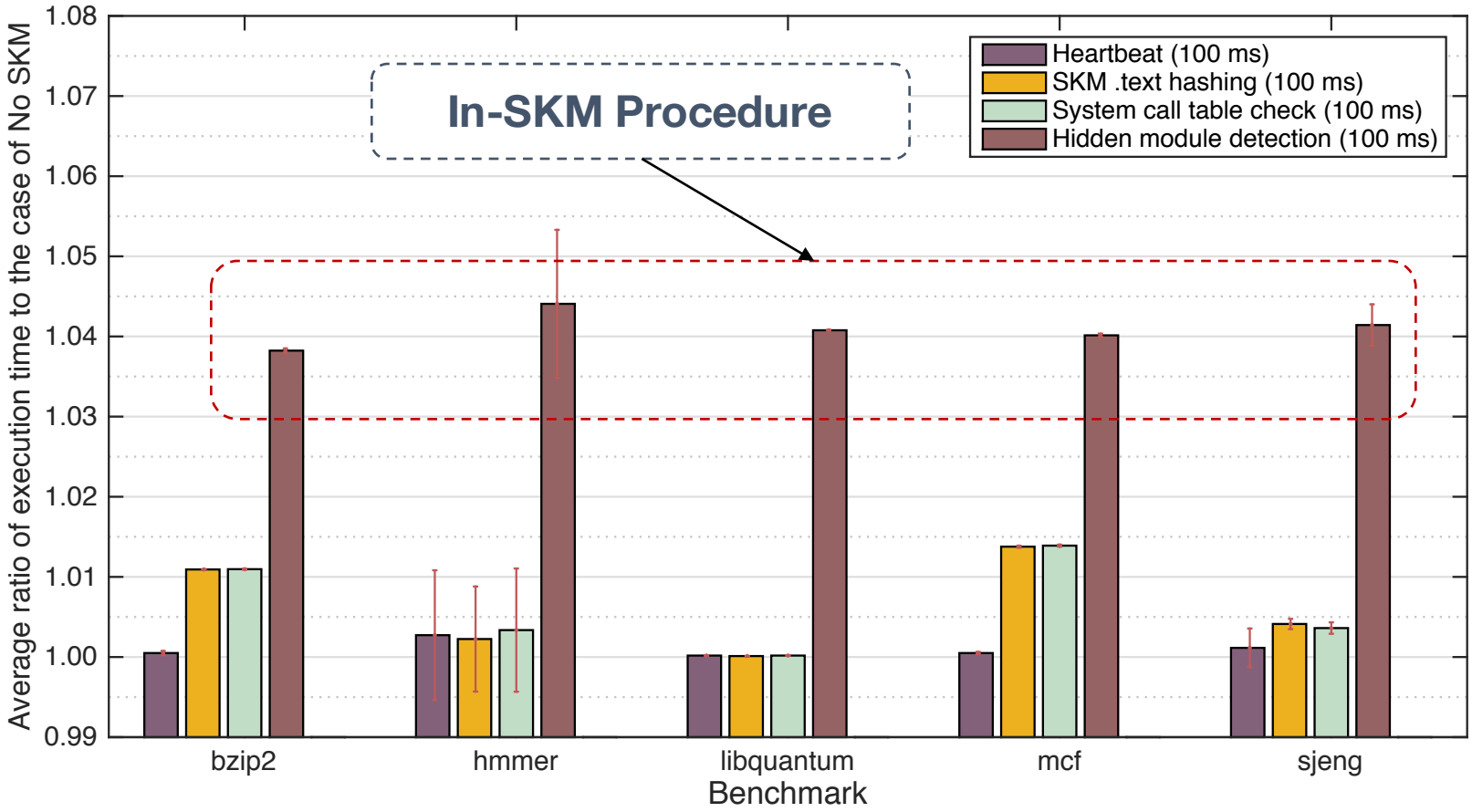
Evaluation – Performance Overhead

- SPEC Benchmarks on the monitored core



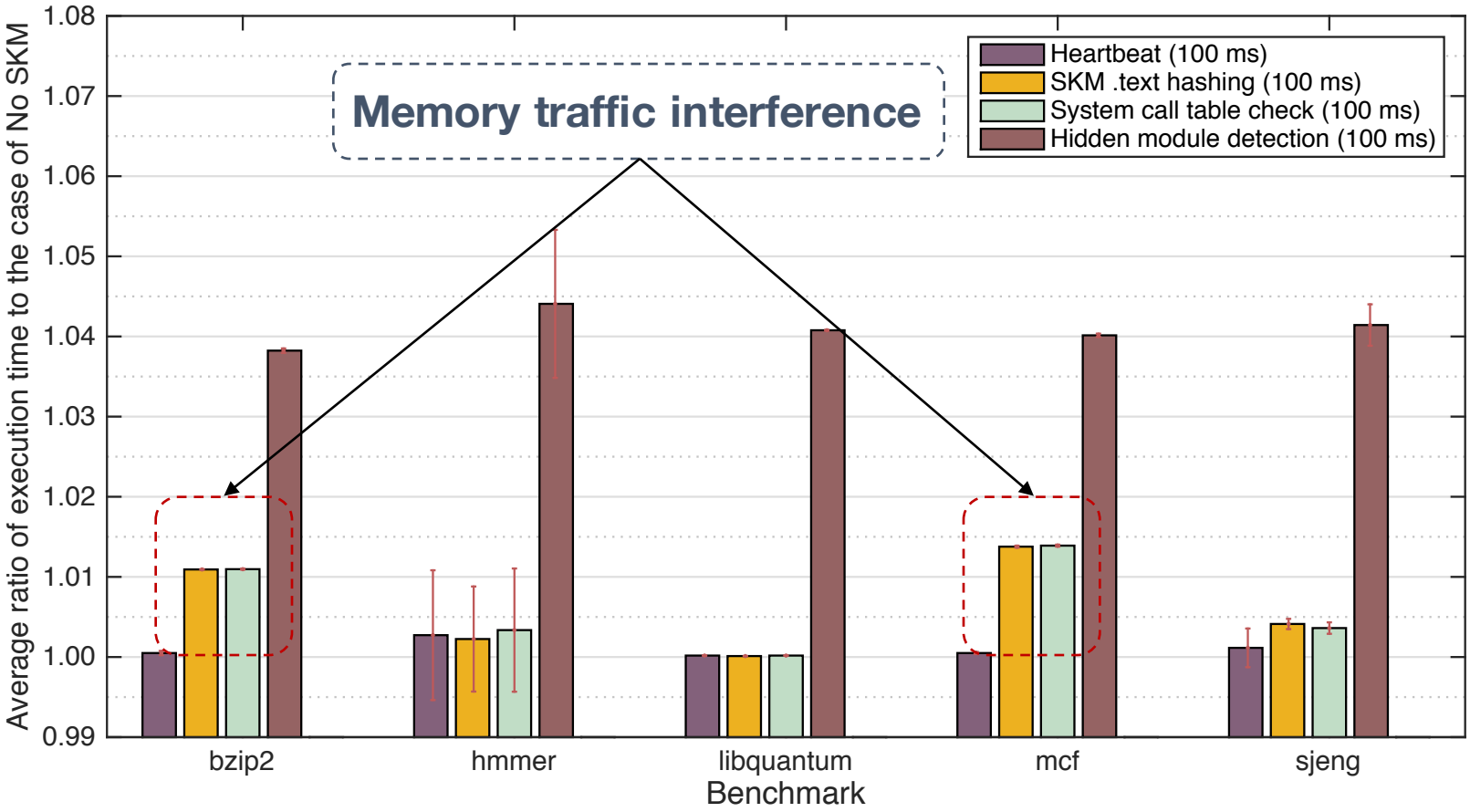
Evaluation – Performance Overhead

- SPEC Benchmarks on the monitored core



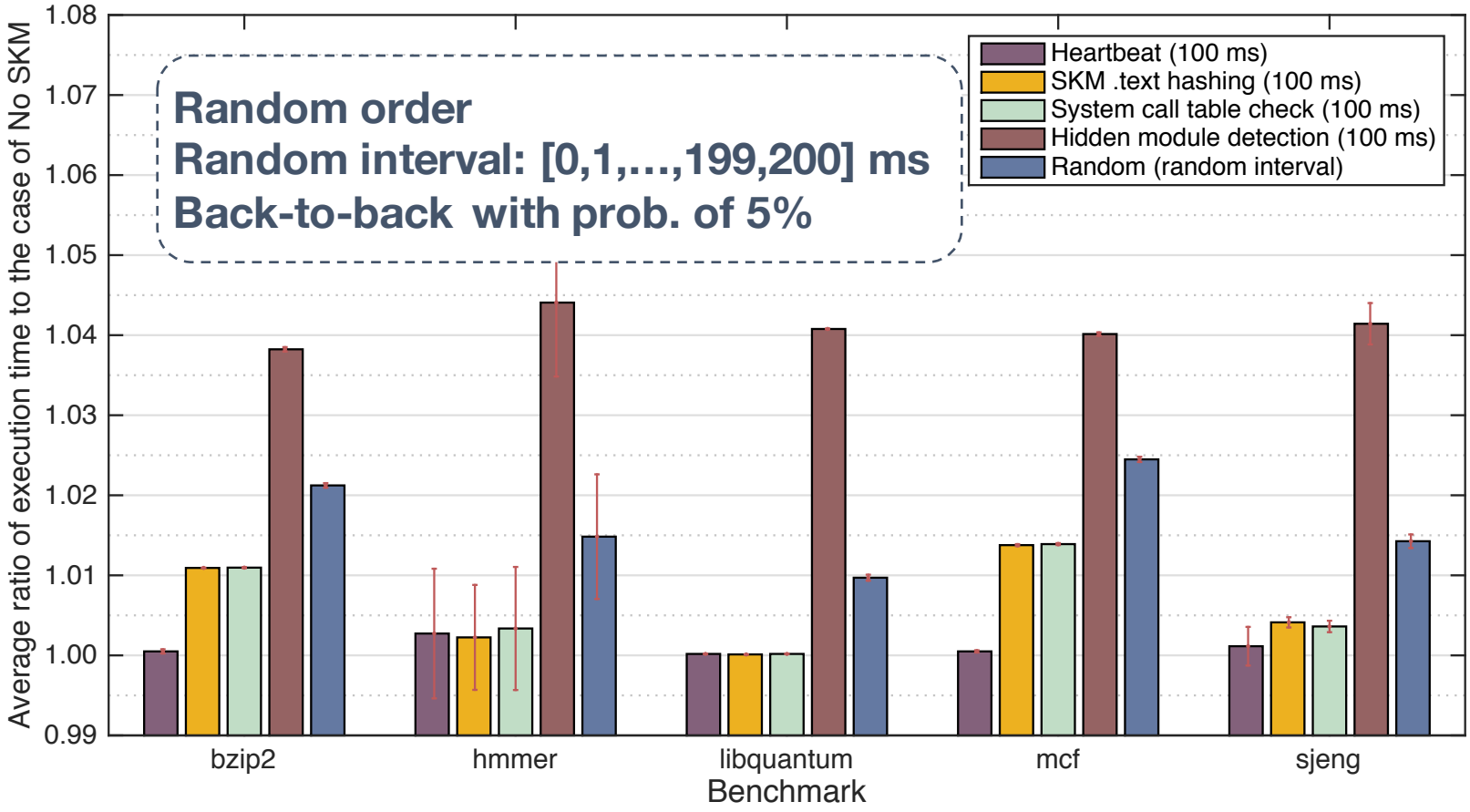
Evaluation – Performance Overhead

- SPEC Benchmarks on the monitored core

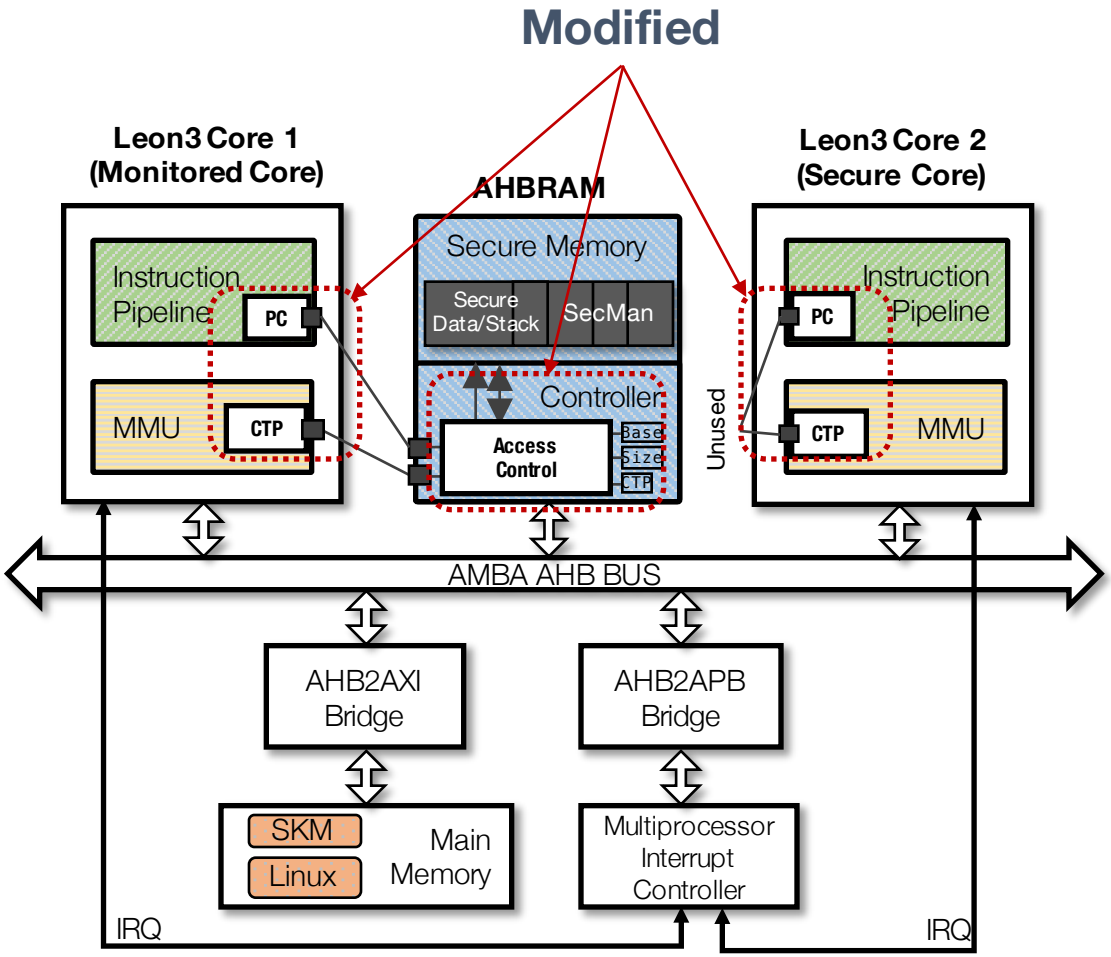


Evaluation – Performance Overhead

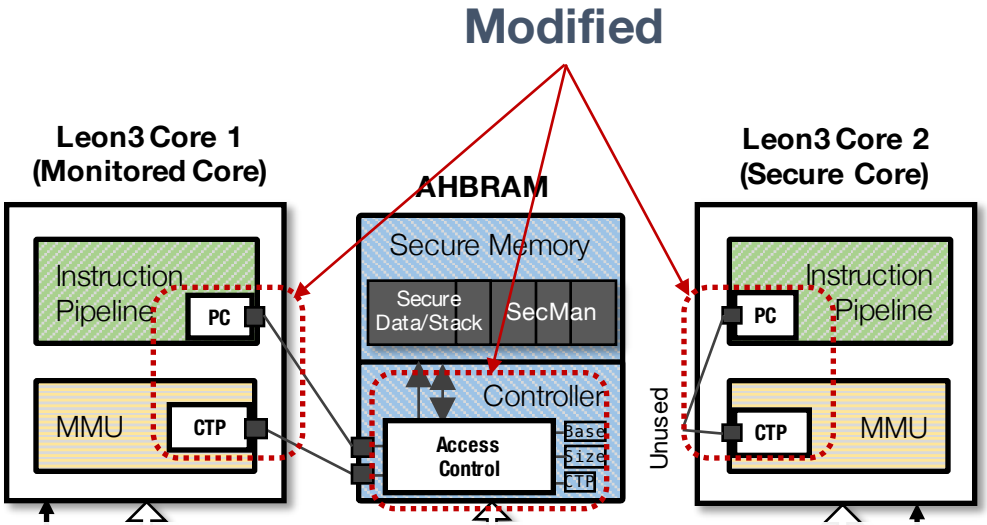
- SPEC Benchmarks on the monitored core



Evaluation – Hardware Cost



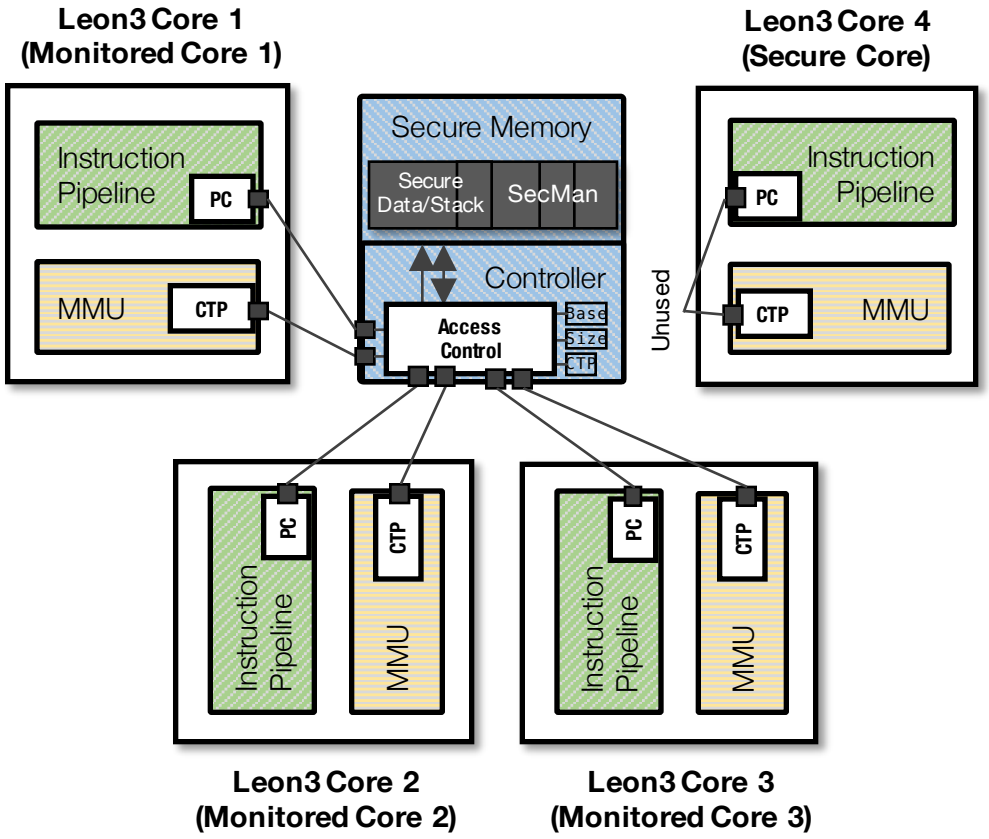
Evaluation – Hardware Cost



Resource	Original	W/ DragonBeam	Δ
Registers	10258	10356	0.96%
Look-up Tables	19482	19511	0.15%

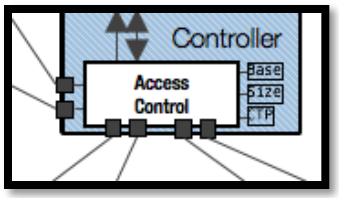
Extension to Multiple Monitored Cores

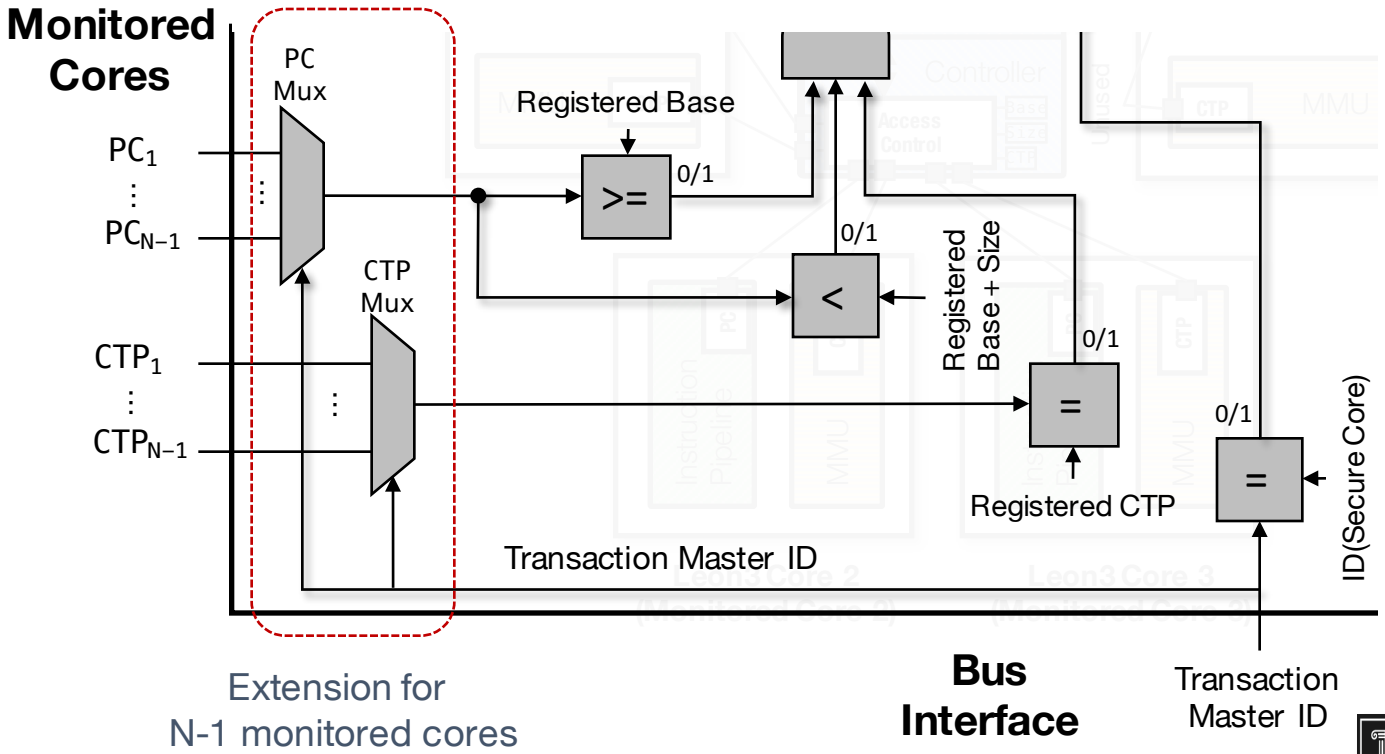
- Extended to quad-core
- Works only for SMP
  - Single SKM



Extension to Multiple Monitored Cores

- Extended to quad-core
- Works only for SMP
  - Single SKM





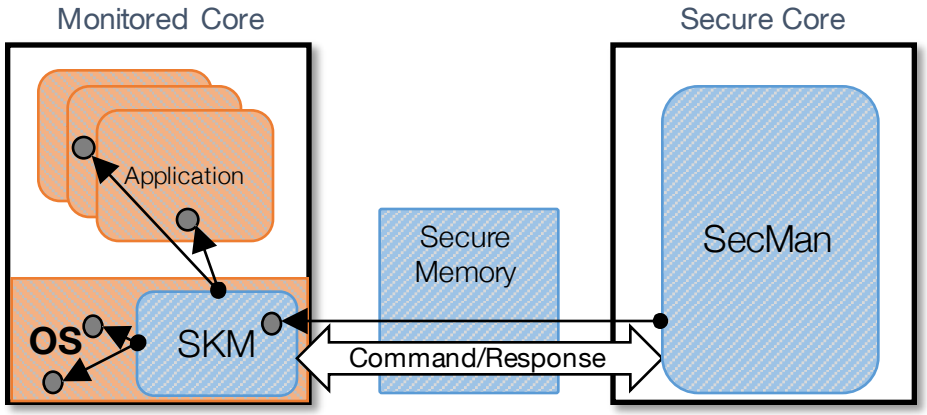
Extension to Multiple Monitored Cores

- Extended to quad-core
- Works only for SMP
  - Single SKM

	Resource	Original	W/ DragonBeam	
Dual Core	Registers	10258	10356	0.96%
	Look-up Tables	19482	19511	0.15%
Quad Core	**Registers**	**18932**	**19029**	**0.51%**
	Look-up Tables	**37777**	**37835**	**0.15%**

Conclusion

- **DragonBeam Framework**
  - HW/SW framework for trusted security monitoring
    - Bootstrap trust into SW layer from trusted HW
    - Multicore-based
  - Expanded observability due to in-place monitoring
    - Secure "Kernel Module"
  - Allows for customized security modules to system developers



Thank you