CISPA
HELMHOLTZ CENTER FOR INFORMATION SECURITY

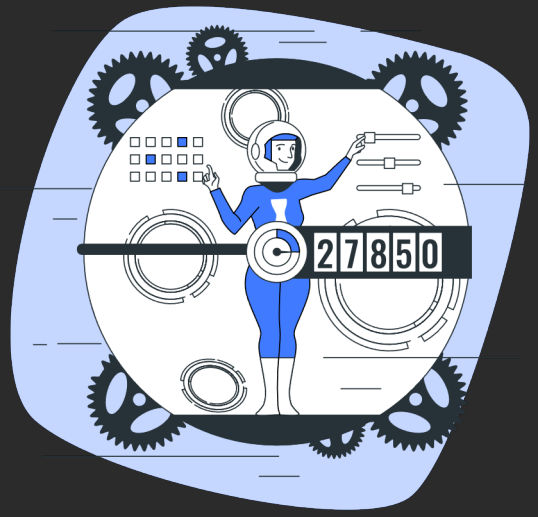
(M)Wait For It:

Bringing the Gap Between Architectural and Microarchitectural Side Channels

USENIX Security ‘23

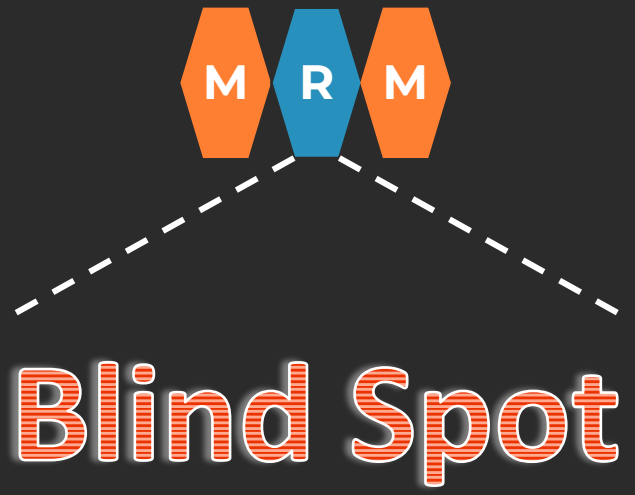
Ruiyi Zhang, Taehyun Kim, Daniel Weber, Michael Schwarz

Motivation



- Timing

- Power



Research Question

- Can we replace measurements with an architecturally-defined interface to leak side-channel information?

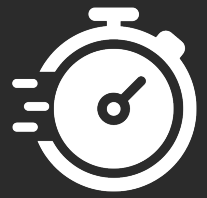
- Can such an interface also reduce the blind spot of existing side channels?

UMONITOR & UMWAIT

- Introduced with Intel Tremont and Alder Lake microarchitecture

- *umonitor* - Arm the hardware with a specified memory range
- *umwait* - Put the CPU into a sleep state

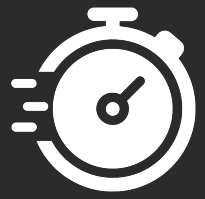
Different Wakeup Triggers



- The default timeout is **100,000** cycles on Linux

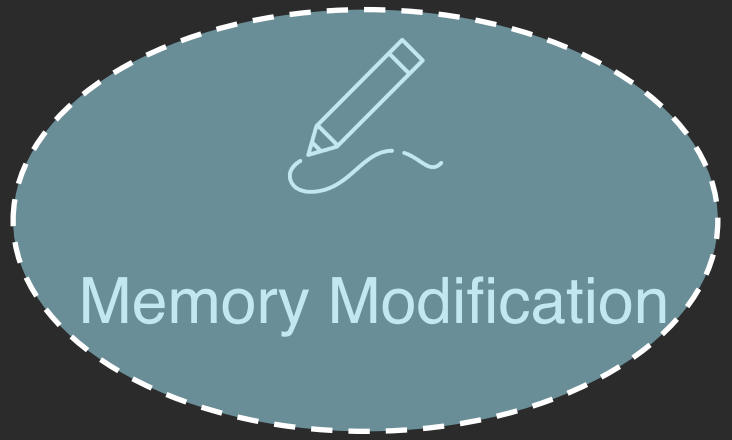
OS-defined Timeout

Carry Flag: 1



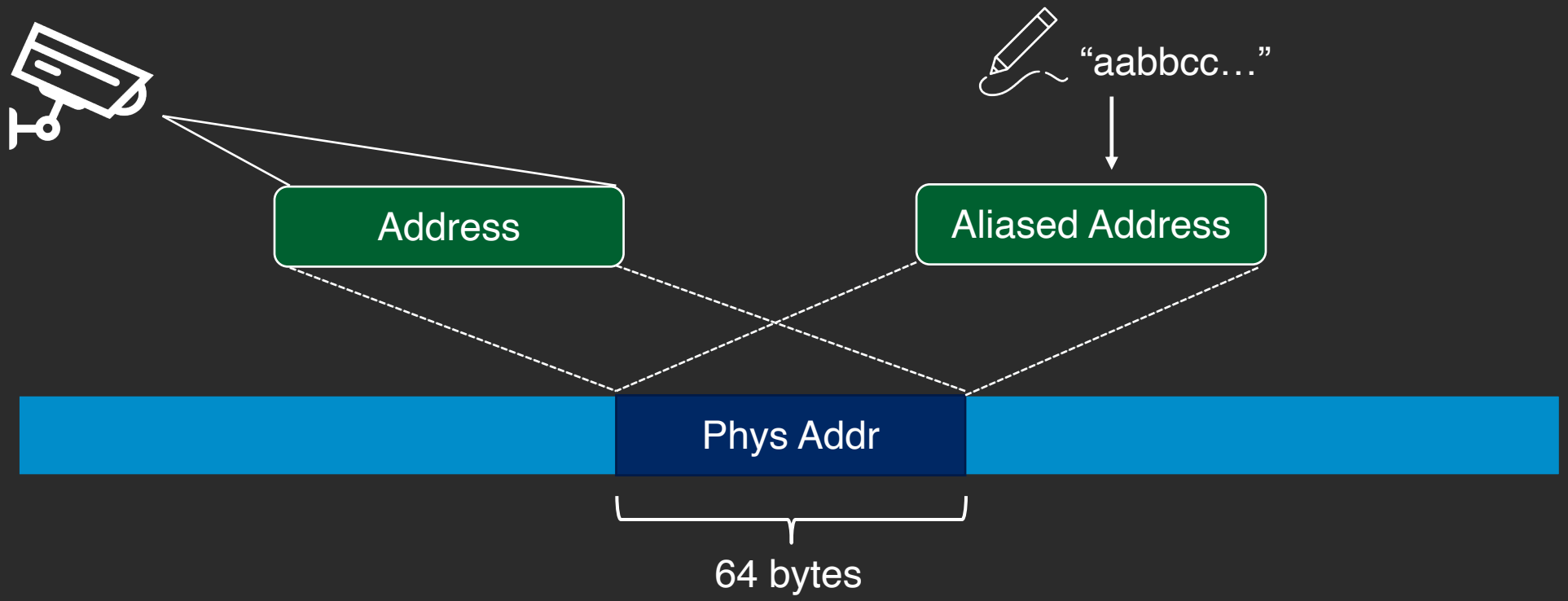
User-defined Timeout

Interrupts



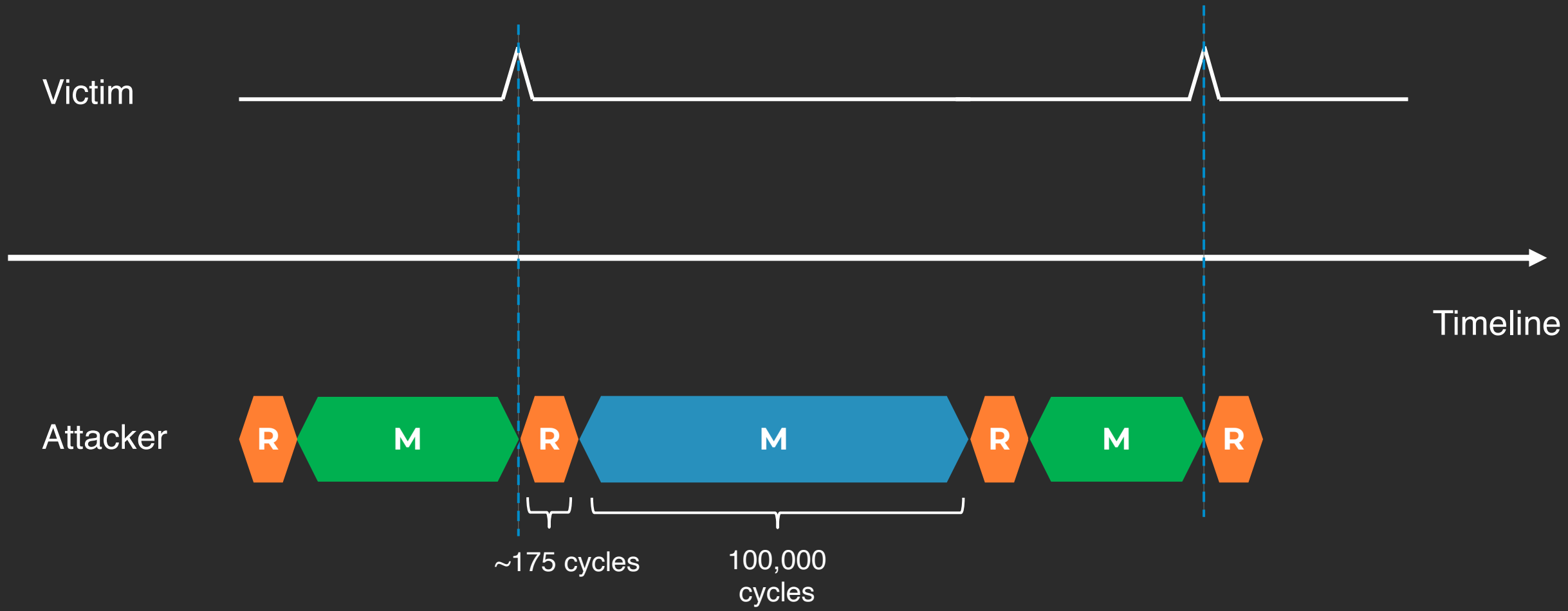
Carry Flag: 0

Analysis of the Monitoring Function

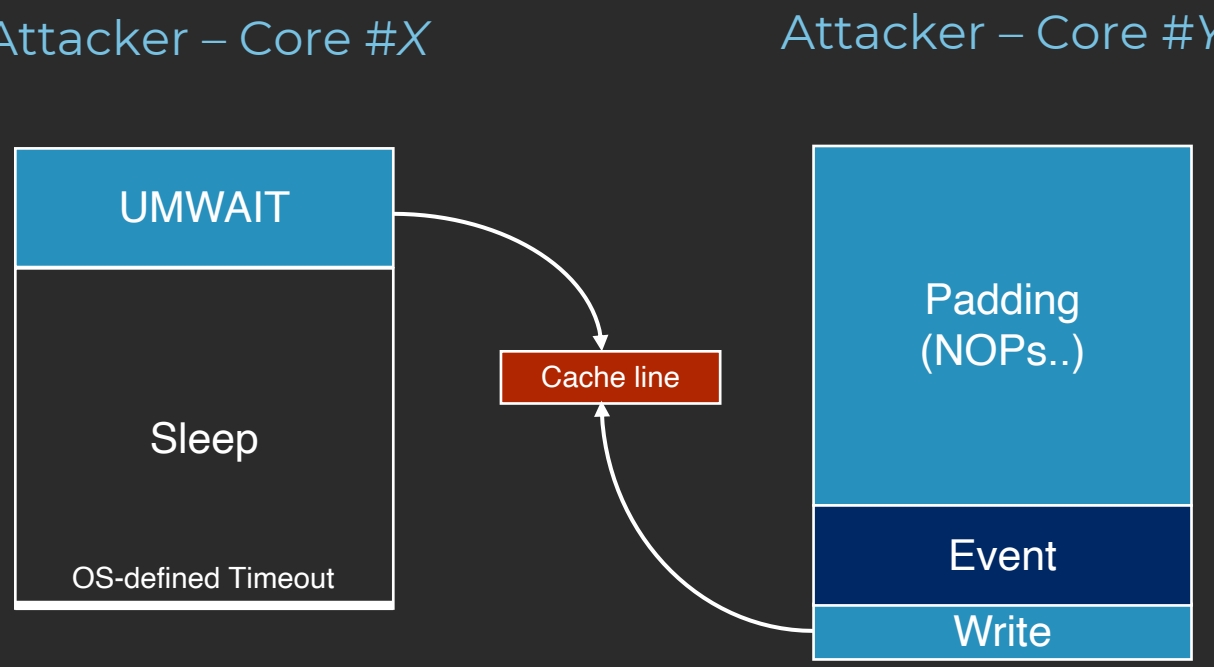


- Although undocumented, **transient** writes also wake up the CPU

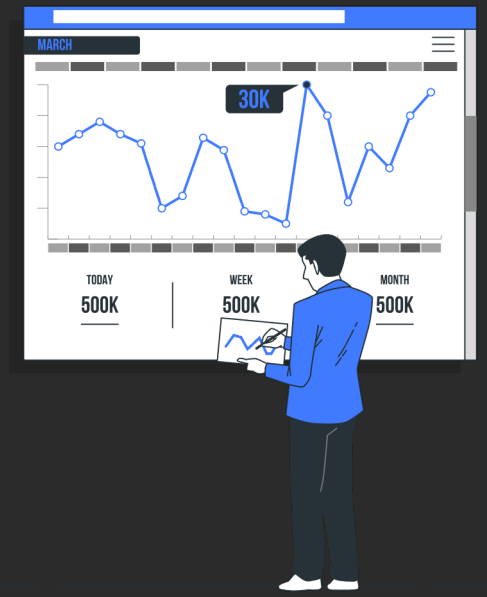
Transient Write Monitor



Timerless Timing Measurement

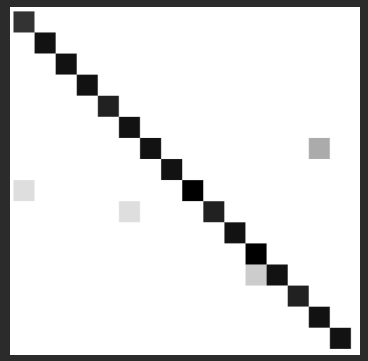


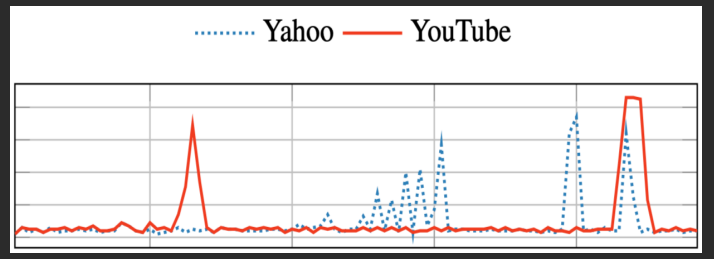
Interrupt Monitor



- Count the number of retired *umwait* within a coarse-grained time bucket (~10ms)
- Also available for AMD and ARM
  - *monitorx* / *mwaitx*
  - *wfi*

Case Studies





- Spectral
  - Up to 200 kbit/s
- AES T-table Attack
- Website Fingerprinting
  - 78 % on AMD
  - 71 % on Intel
  - 67 % on Arm

Takeaway

- First architectural side channel on Intel microarchitecture
- Minimal blind spot, High-precision, Low-noise
- Interrupt-timing attacks on Intel, AMD, and ARM with unpriviledged instructions

Contact:

https://twitter.com/Rayiizzz

ruiyi.zhang@cispa.de

Artifact:

https://github.com/cispa/mwait