Chapter 8 - Software-based Power Attacks

Attacking CPUs with Power Side Channels from Software

Mathias Oberhuber

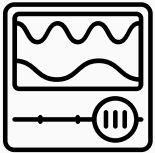
3rd April 2025

CPU Power Management

- CPU power management is complex
- In order to save power, you can …

Shut down resources

Reduce voltage



Reduce frequency

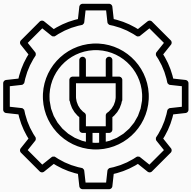
Intel RAPL

- Therefore, the CPU requires:
  - Thermal Management
  - Platform Power Limiting
  - Power/Performance Budgeting
- Domains: PKG, CORE, MC
- **Intel Running Average Power Limit** (RAPL) provides:

power limiting

energy reading

Intel RAPL



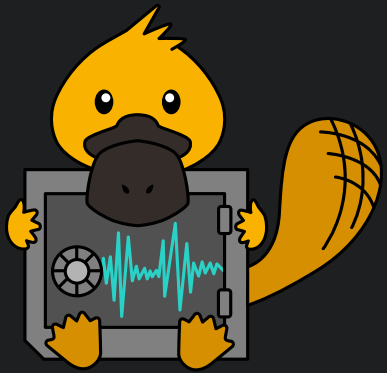
- **Linux**: accessed via `powercap` framework
  `/sys/devices/virtual/powercap/intel-rapl`
- **macOS** and **Windows**: Intel driver needs to be installed

Intel RAPL: Properties

Unprivileged power meter

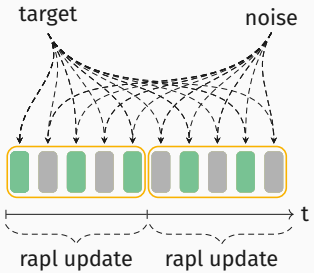
No physical access

Low refresh rate

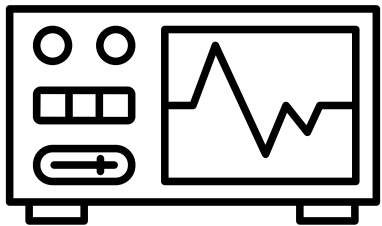


Platypus Attack

RAPL: Measurement Techniques



- Measure an **instruction** by
  - executing it once
  - executing it repeatedly
  - padding it with known instructions
  - reissue the instruction after an interrupt



What can we do with this?

Distinguishing Instructions

- Measure the energy consumption of **different instructions**

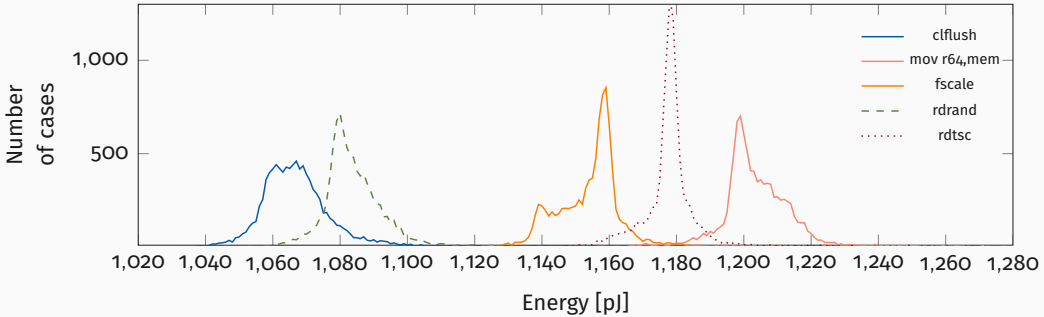
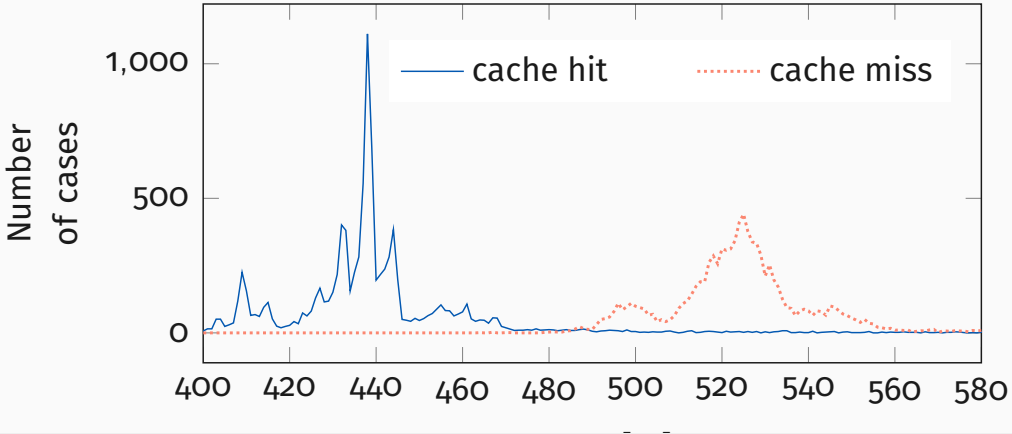


Figure 1: A histogram of the power consumption of various instructions on the i7-6700K (desktop) system.

Distinguishing Load Targets

- Measure the energy consumption of **different load targets**



Distinguishing Operands

- Measure the energy consumption of **different operands**

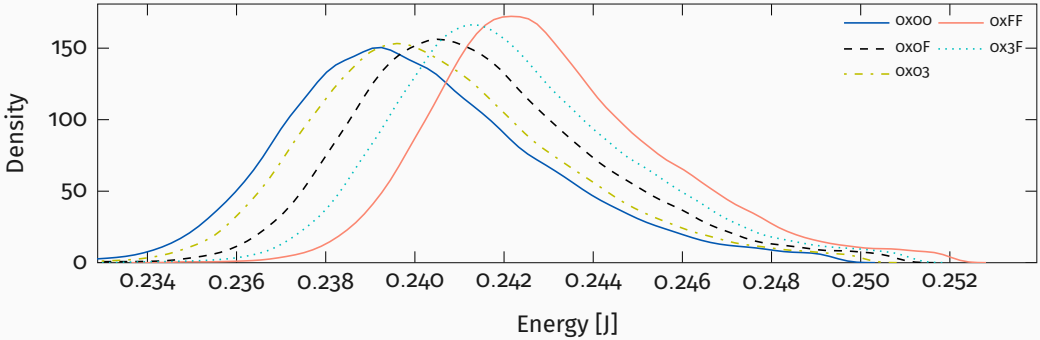


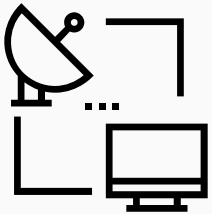
Figure 3: Measured energy consumption of the `imul` instruction with one operand fixed to 8 and the other varying in its Hamming weight.

Let’s exploit this!

Covert Channel

- Hidden communication channel
- Leveraging the power side channel

Covert Channel



- 2 Processes, Sender and Receiver
  - **Send a 1**: Perform energy-consuming instructions
  - **Send a 0**: Idle
- Receiver measures power consumption

→ **Deduces transmitted bit**

Covert Channel

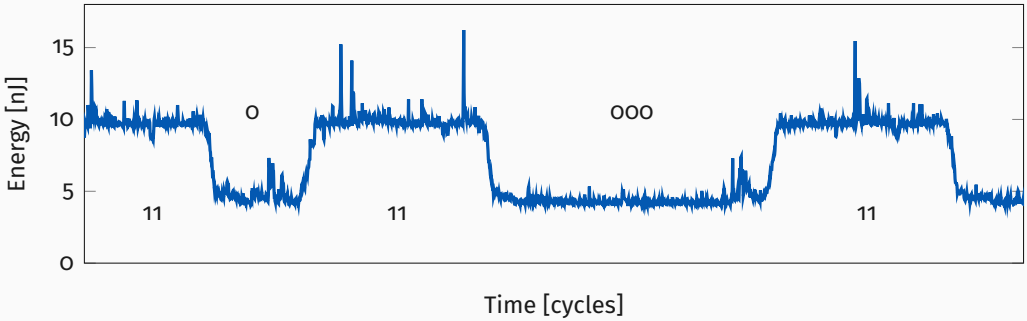


Figure 4: Transmission of bits `1101100011` using the time-less covert channel.

Breaking KASLR

- Kernel Address Space Layout Randomization (KASLR)
- Exploit **energy consumption differences** between
  - Mapped addresses
  - Unmapped addresses
- **Valid address translations** are cached in the **TLB**

Breaking KASLR

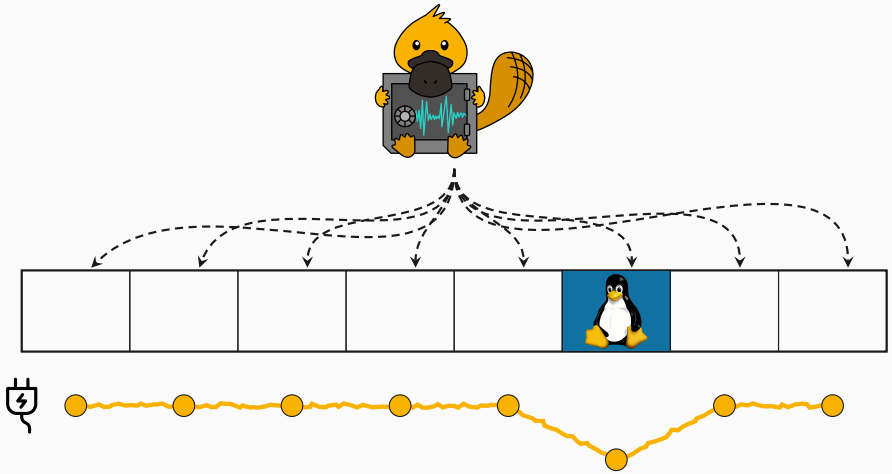


Figure 5: Repeated Page-table walks for unmapped pages require more power

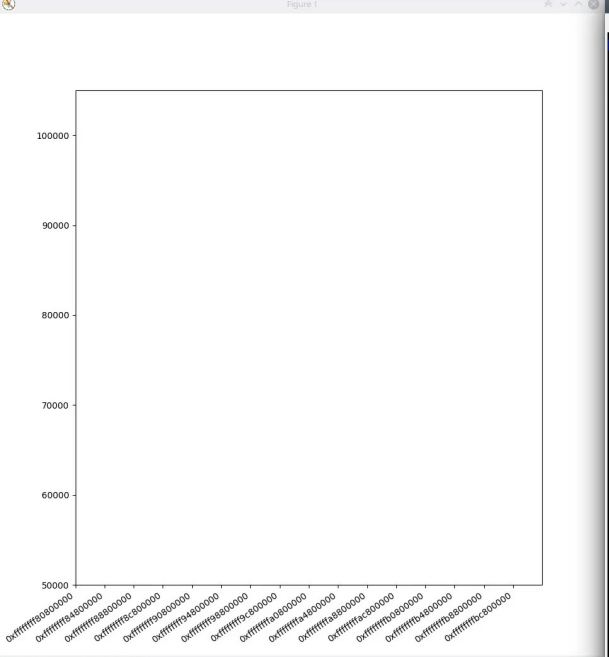
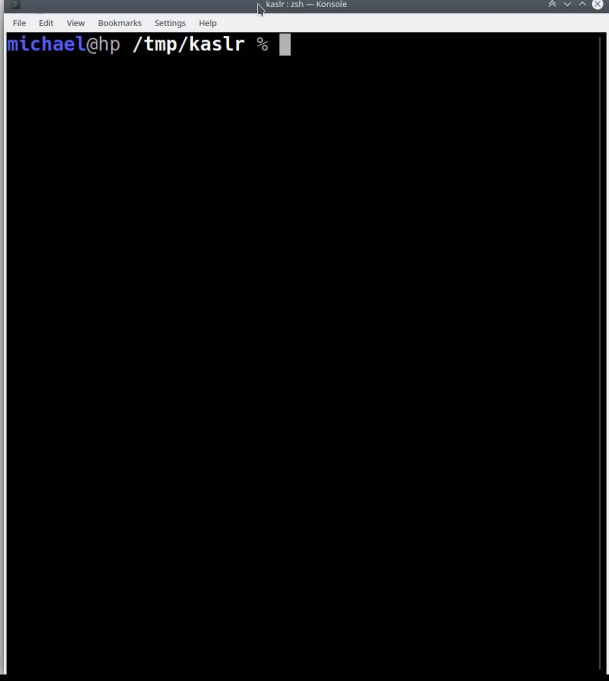
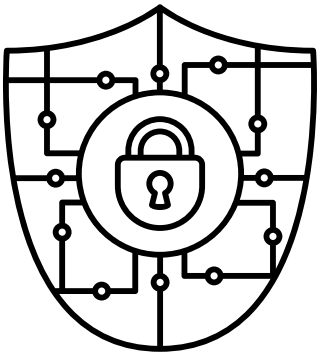


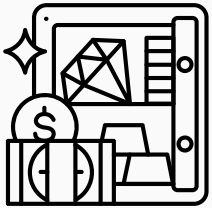
Figure 1
100000
90000
80000
70000
60000
50000
0xffffff8e8800000
0xffffff8f4800000
0xffffff8f8800000
0xffffff8fc800000
0xffffff908800000
0xffffff914800000
0xffffff918800000
0xffffff91c800000
0xffffffa0800000
0xffffffa4800000
0xffffffa8800000
0xffffffac800000
0xffffffb0800000
0xffffffb4800000
0xffffffb8800000
0xffffffbc800000


kaslr : zsh — Konsole
File
Edit
View
Bookmarks
Settings
Help
michael@hp /tmp/kaslr %



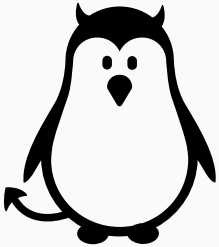
Attacking Intel SGX: RSA Key Recovery

Attacking Intel SGX



- Instruction-set extension
- Integrity and confidentiality in **untrusted environments**
- **Enclaves** offer protected areas of memory
- Operating system can be **compromised**

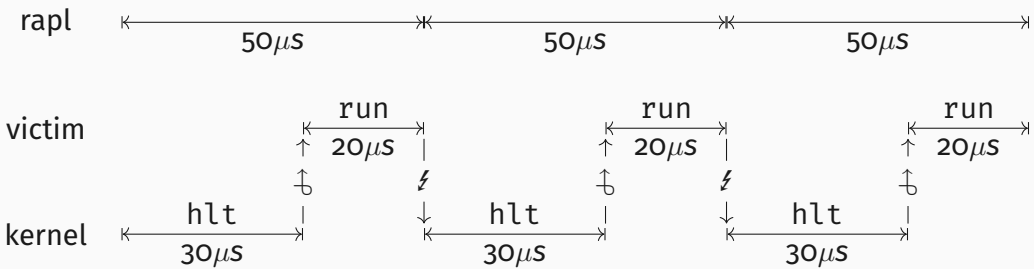
Attacking Intel SGX: Threat Model



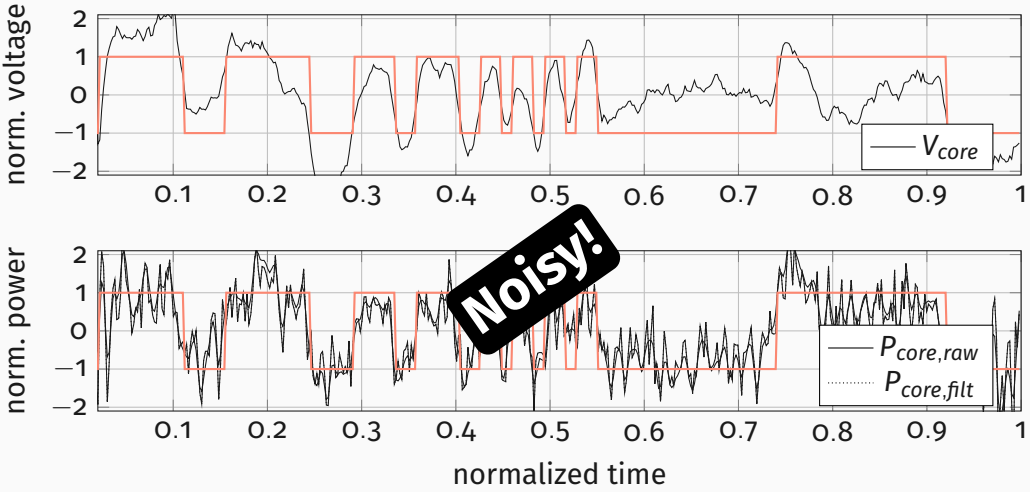
- More power as an evil operating system
- Hook the SGX Enclave exit point
- Directly read out the **RAPL values** from the MSRs
- No operating system overhead!
- Interrupt victim often to increase resolution

Halt Delay

- RAPL domains have a nearly fixed update interval
- Delay the interrupt return with the halt delay in the ISR
- Reduces the execution time of the victim in the current interval



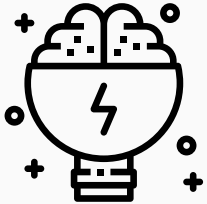
SPA Attack - Results



SGX Step

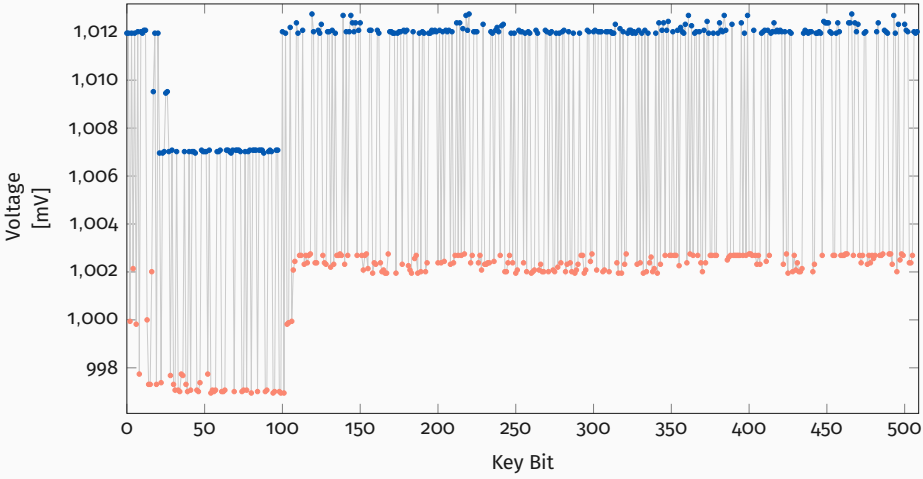
- **SGX-step** is an open-source Linux kernel framework
- Configure APIC timer interrupts
- Single and zero-step enclave execution

SGX Step

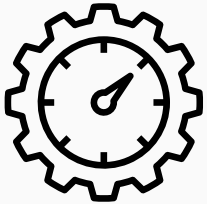


- **Combine** Intel RAPL with SGX-step
- Measure the energy consumption of **single instructions**

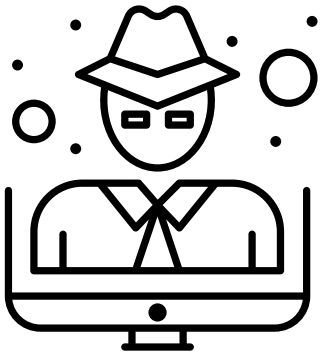
Attacking mbed TLS



Performance

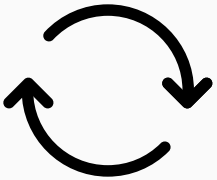


- Time per key bit increases linearly based on the index
- 3h 31m for a **512 bit**
  - 52 minutes for finding target instruction
- Record 3 samples per key bit
  - This could be extend to a single trace attack



Crypto Attacks from User Space

Repeat Repeat Repeat



- **Difficult** to measure parts without SGX-step
- **Can** measure over the **overall execution**

Correlation Power Analysis

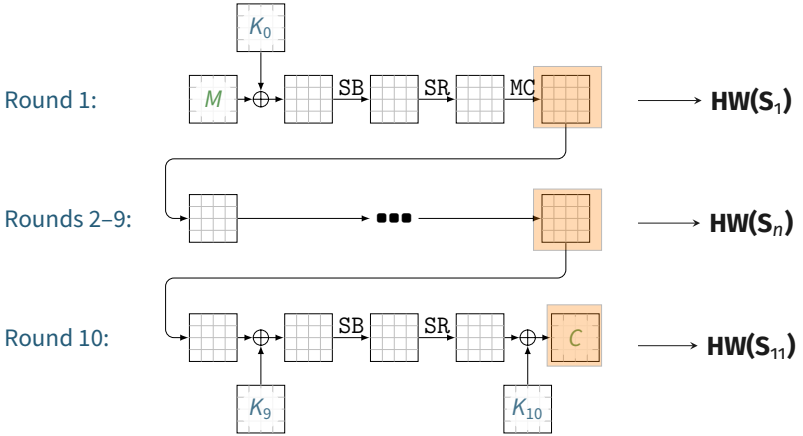
- Building a power consumption **model** of the device:

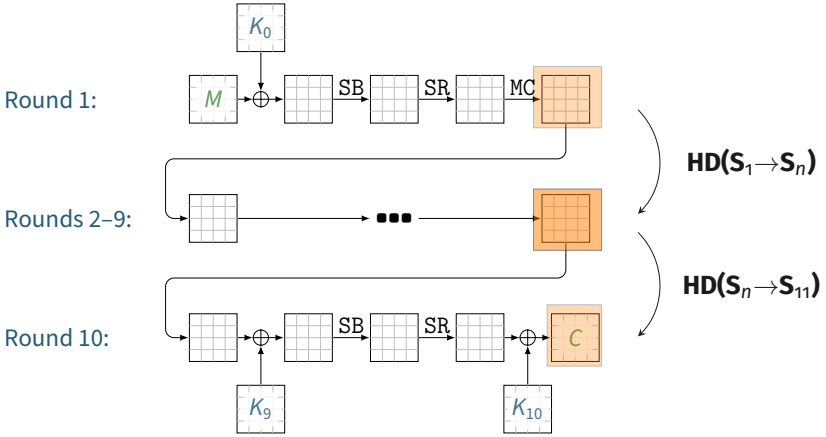
Hamming Weight

Number of bits set

Hamming Distance

Bits flipping between operations


K_0
Round 1:
M
SB
SR
MC
HW(S_1)
Rounds 2–9:
HW(S_n)
Round 10:
SB
SR
C
HW(S_{11})
K_9
K_{10}


K_0
Round 1:
M
SB
SR
MC
Rounds 2–9:
Round 10:
SB
SR
C
K_9
K_{10}
$\mathbf{HD(S_1 \to S_n)}$
$\mathbf{HD(S_n \to S_{11})}$

Correlation Power Analysis

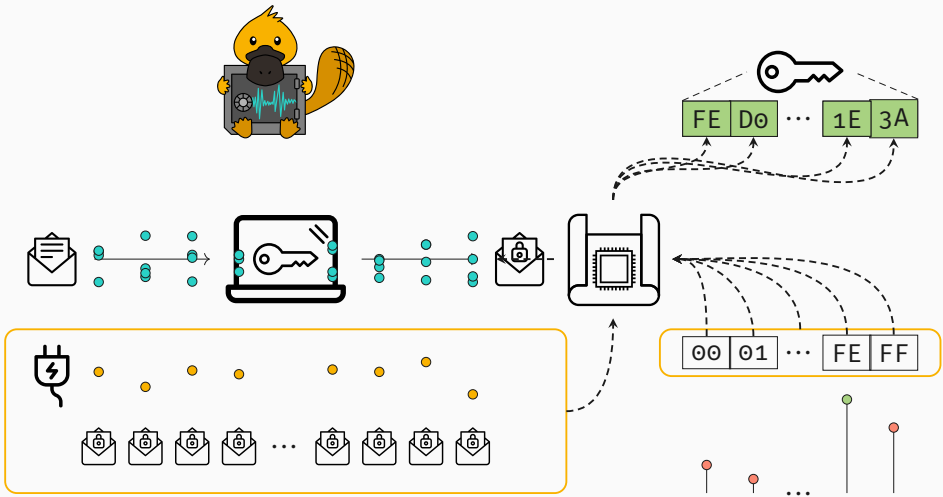
- **AES-NI**: Side-channel resilient instruction-set extension
- Target **AES-NI** in a scenario where we can trigger encryption/decryption of many blocks
  - Disk encryption/decryption
  - TLS
  - (Un)sealing SGX enclave state

Correlation Power Analysis

- We control the plain text
- We observe the cipher text
- We measure the energy consumption over many operations
- We guess the key

- With our model and all possible values, **where** is the **correlation** the highest?

CPA Attack



Other CPU Vendors

- AMD affected as well
- Never heard back after disclosure
- Similar Linux patch as Intel

Countermeasures

Countermeasures

- Remove the unprivileged access to the RAPL MSRs
- **1 Line Patch** for the Linux Kernel

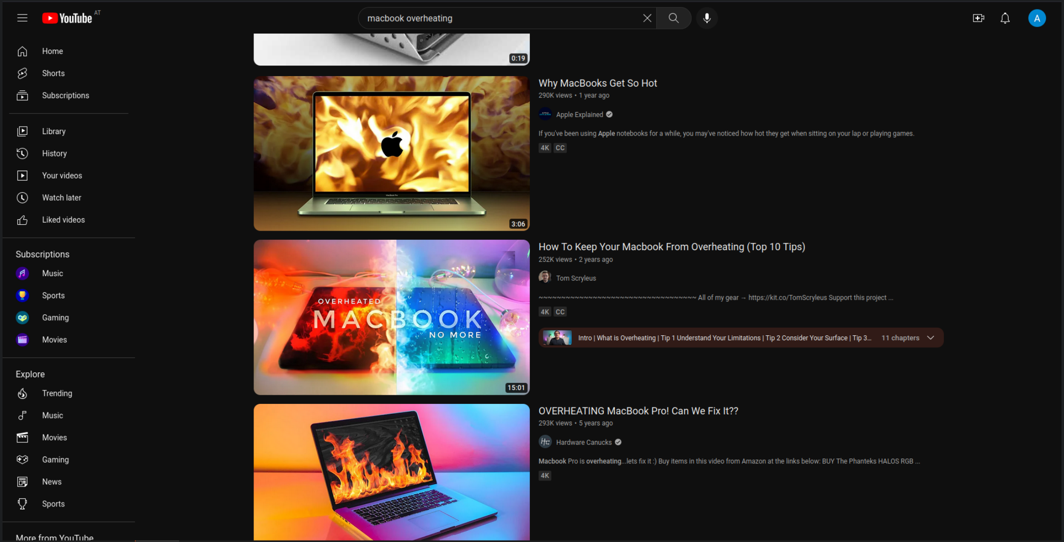
Countermeasures

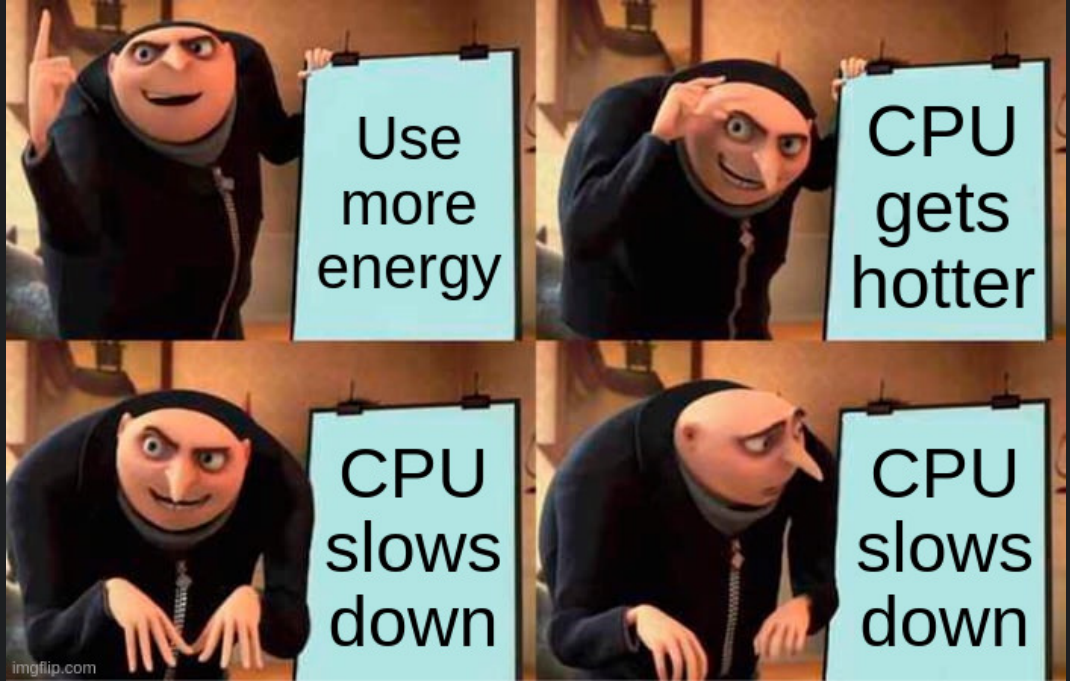
- Threat model of SGX allows a **compromised operating system**
  - Operating system patch does not help
- **Microcode updates** are necessary
  - Fallback to a model of the energy consumption
  - Does **not allow** to distinguish data/operands any more
  - **Constant-time implementations** are necessary

Takeaways

- Power side-channel attacks can be exploited **from software** on modern CPUs
- Threat model of Intel SGX requires more complex mitigations

Remove Interface = The End?


YouTube AT
macbook overheating
Home
Shorts
Subscriptions
Library
History
Your videos
Watch later
Liked videos
Subscriptions
Music
Sports
Gaming
Movies
Explore
Trending
Music
Movies
Gaming
News
Sports
More from YouTube
0:19
3:06
Why MacBooks Get So Hot
290K views • 1 year ago
Apple Explained
If you've been using Apple notebooks for a while, you may've noticed how hot they get when sitting on your lap or playing games.
4K CC
OVERHEATED
MACBOOK
NO MORE
15:01
How To Keep Your Macbook From Overheating (Top 10 Tips)
252K views • 2 years ago
Tom Scryleus
~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~ All of my gear → https://kit.co/TomScryleus Support this project ...
4K CC
Intro | What is Overheating | Tip 1 Understand Your Limitations | Tip 2 Consider Your Surface | Tip 3...
11 chapters
OVERHEATING MacBook Pro! Can We Fix It??
293K views • 5 years ago
Hardware Canucks
Macbook Pro is overheating...lets fix it :) Buy items in this video from Amazon at the links below: BUY The Phanteks HALOS RGB ...
4K
~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

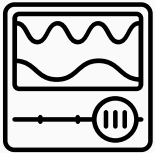

Use more energy
CPU gets hotter
CPU slows down
CPU slows down
imgflip.com

Remember?

- CPU power management is complex
- In order to save power, you can …

Shut down resources

Reduce voltage



Reduce frequency

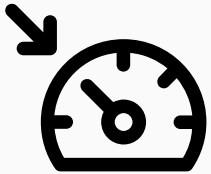
Hertzbleed

- The Hertzbleed attack from Wang et al. shows:
- If more **energy** is used
- The CPU gets **hotter**
- Until the frequency is no longer sustainable

→ The runtime of the executed code slows down

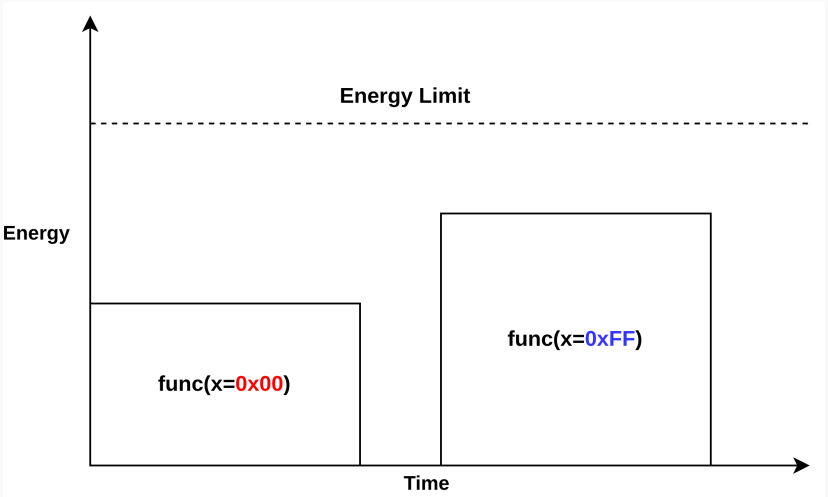
- Measure with fixed clock, e.g., `rdtsc`

How to Throttle the CPU

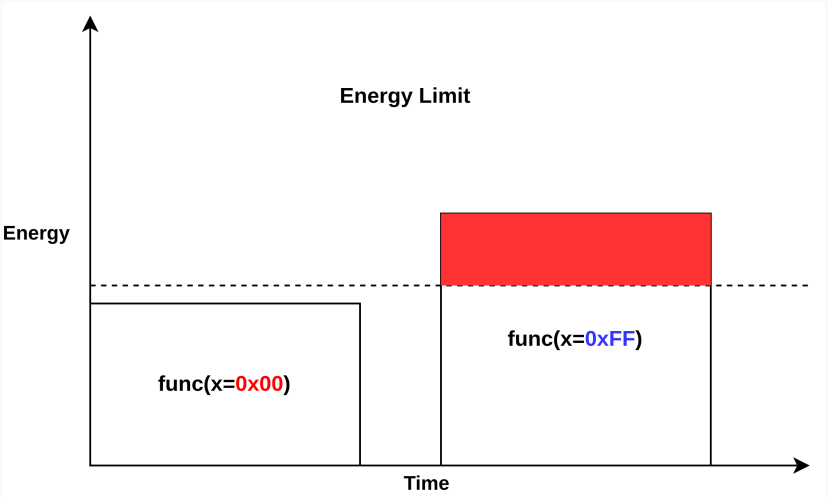


- RAPL provides energy **limits**
  - If exhausted CPU throttles the frequency
- Run **Stress** on the system
  - CPUs start throttling when using many threads

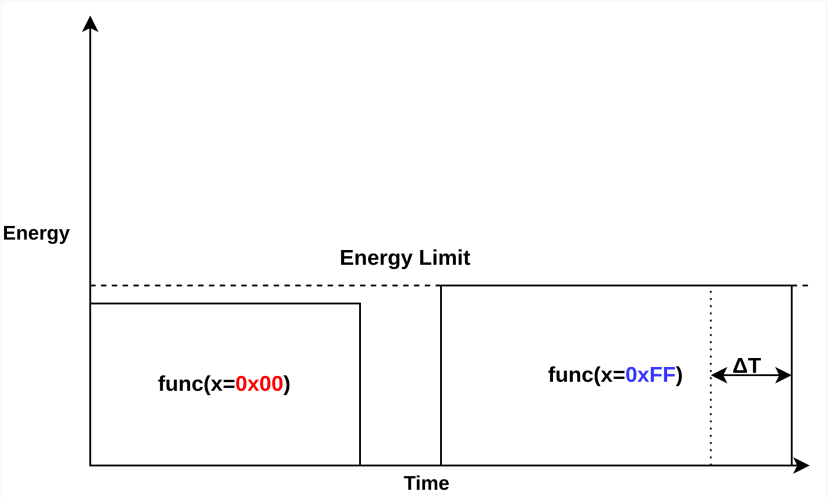
Converting Energy Differences



Convert Energy Differences



Convert Energy Differences



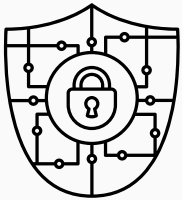

NYPD
POWER
TIME
imgflip.com

What can we do with this?

Covert Channel

- Hidden communication channel
- No power interface required
- Time/Frequency measurements proxy power interface

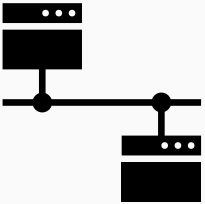
Attacking Crypto



- AES Correlation Power Analysis
  - Measure execution time of AES encryptions

  $\rightarrow$ Apply CPA technique to recover key

Remote Attack



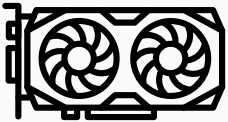
- Remote attacker requests service from server
  - Cryptographic operation, i.e. encryption, signature
- Server computes respose using secret

→ **Hertzbleed-effect** influences response times

- **Calculations using secret** influences server CPU frequency

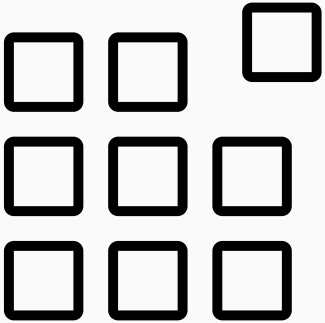
→ Attacker recovers secret using collected timings

GPU Throttling



- **Integrated** GPUs share power limits with the CPU
  - → **CPU throttling** indicates high GPU consumption
- **Dedicated** GPUs have power limits too
  - → **Observable** by timing a GPU workload

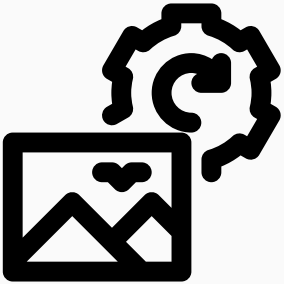
Pixel Stealing



- What secrets are *“inside”* a GPU?
  - GPU renders windows and screen

  → **Privacy** related information
- **Pixel** color represents the information

Pixel Stealing



- **Post-processing** without revealing the pixels
- Pixel value is the **data operand**
- Distinguishable power consumption
  - **Bright** pixel → less power
  - **Dark** pixel → more power

→ Measure timing and infer pixel value

The End?

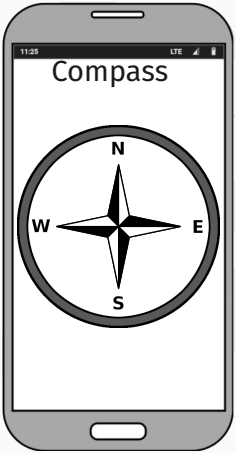
Are there other exploitable power-related signals?

About

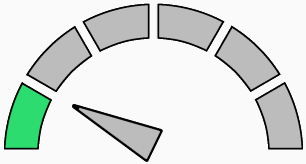
Android **power-related** side channel

- Android sensor interface as a proxy for power measurements purely from software
- Systematic analysis of 9 Android smartphones:
  - Recovering leakage properties: Integration interval, rotation-dependent leakage
- Local attack:
  - Malicious app leaking processed AES key bytes
- Remote web-based JavaScript attack:
  - JavaScript sensor-based pixel-stealing attack leaking cross-origin pixels up to $5\,\mathrm{s/pixel}$

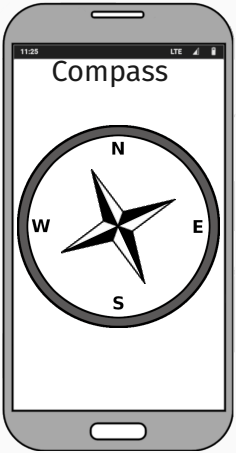
Motivation



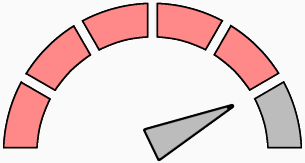
CPU utilization



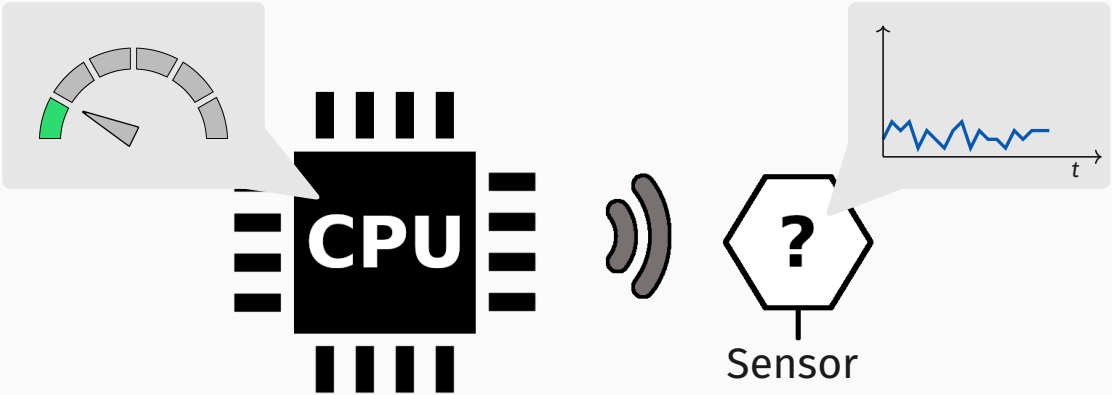
Motivation



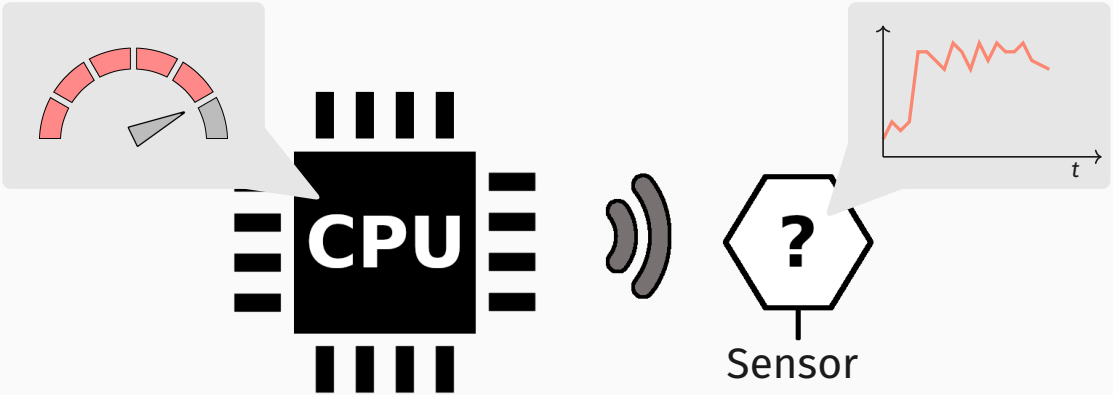
CPU utilization



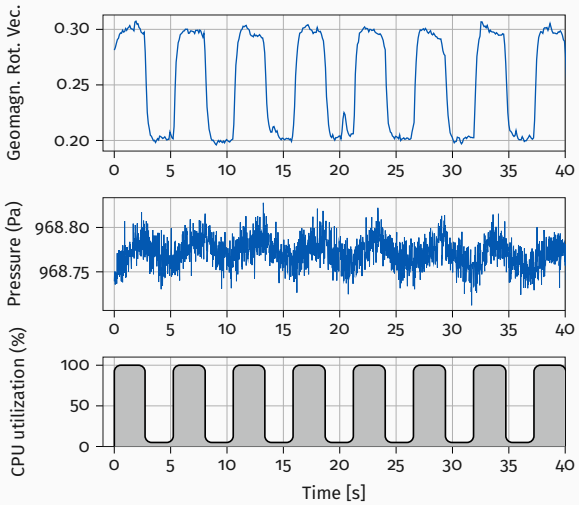
Motivation



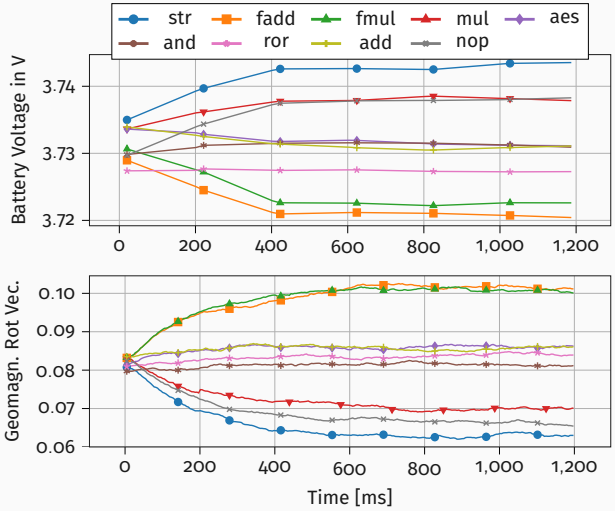
Motivation



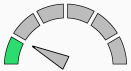
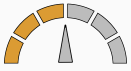
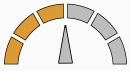
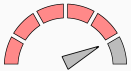
Systematic Evaluation: Varying CPU Utilization

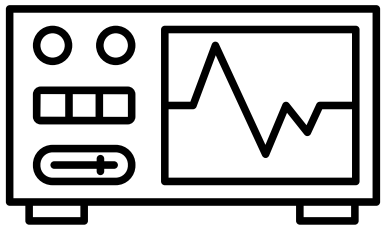


Systematic Sensor Analysis: Executed Instructions



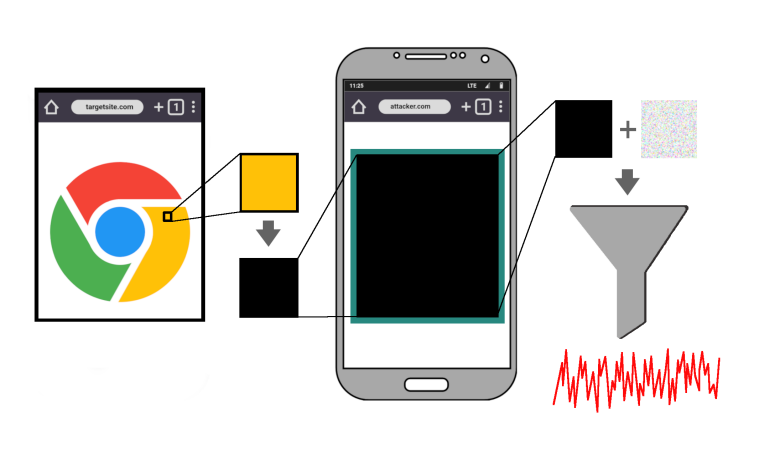
Systematic Sensor Analysis: Varying Data Operands

a	⊕	b	=	c	Power
a_0	⊕	b_0	=	00_2	
a_2	⊕	b_2	=	01_2	
a_2	⊕	b_2	=	10_2	
a_4	⊕	b_4	=	11_2	

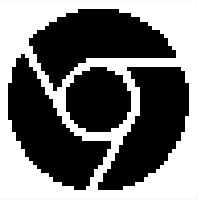
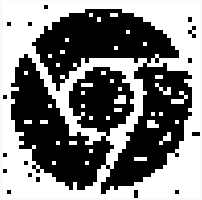


What can we do with this?

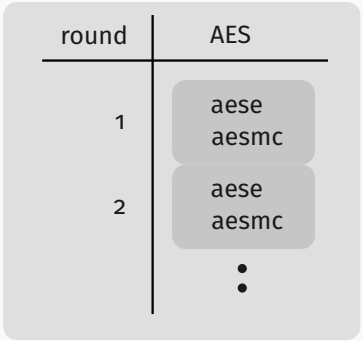
JavaScript Pixel-Stealing Attack



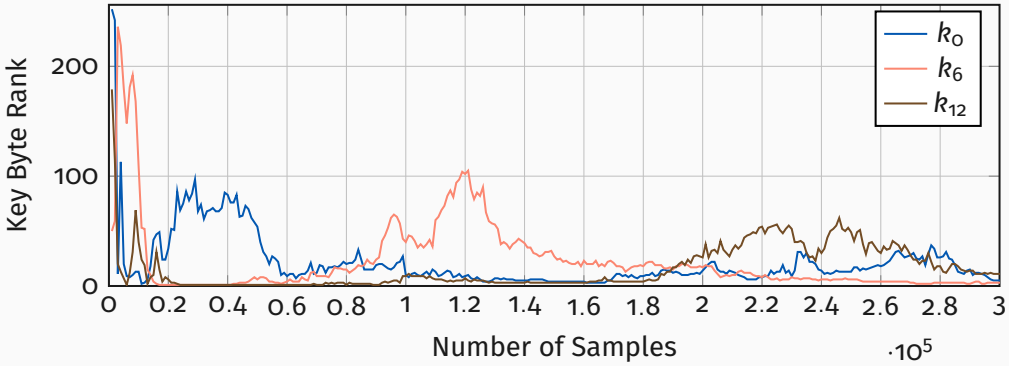
JavaScript Pixel Stealing: Evaluation

			
Image:	Original	Magnetometer	Abs. Orientation
Time/Pixel (s):		5	10
Accuracy (%):		90.2	70

AES Attack Case Study

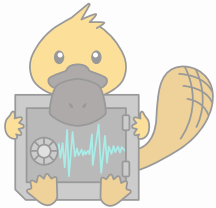


AES Correlation Power Analysis



How can we transform power side channels towards a broader scope?

Motivation



Software-based Power Side C

- **Specific** targets: Algorithms
- Leak edge cases
- **Limited** to a side channel

ecution Attacks

Generic targets: CPU components

- Leak arbitrary data
- **Agnostic** to side channels

Collide+Power

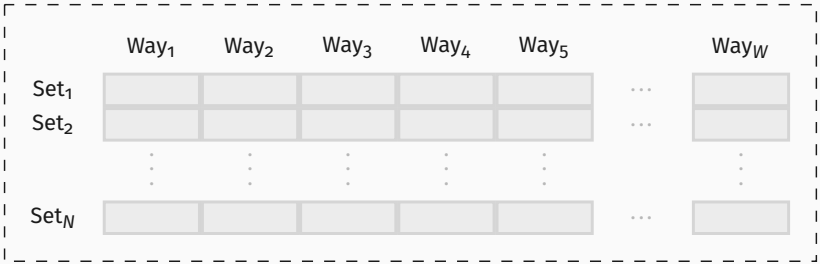
Collide+Power - Overview

- **Collide+Power** exploits leakage between:
  - **Guess** $\mathcal{G}$**:** Attacker-controlled data
  - **Value** $\mathcal{V}$**:** Victim secret data

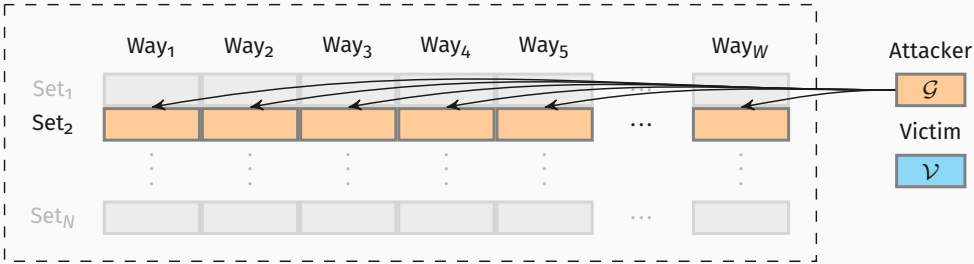
💡 Hamming distance: $\mathsf{hd}(\mathcal{G}, \mathcal{V})$

$\rightarrow$ How to exploit this limited information?

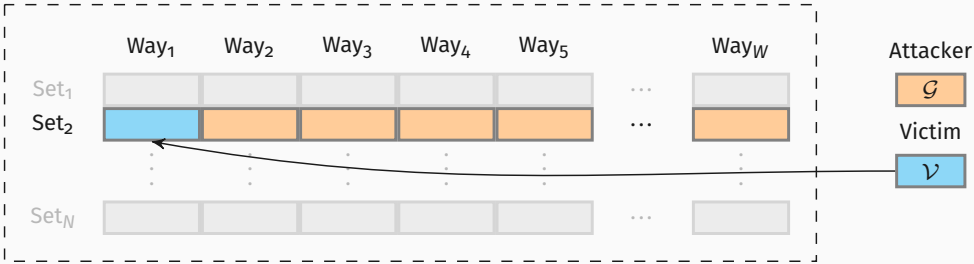
Collide+Power - Memory Subsystem



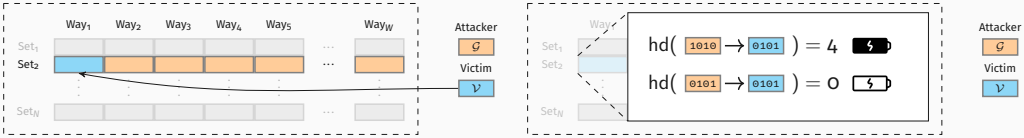
Collide+Power - Memory Subsystem



Collide+Power - Memory Subsystem

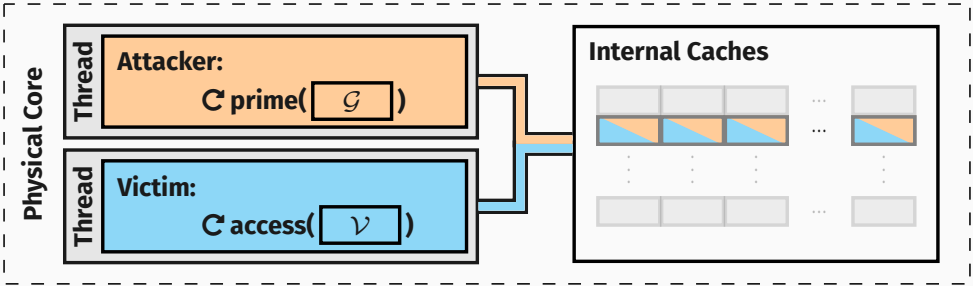


Collide+Power - Memory Subsystem

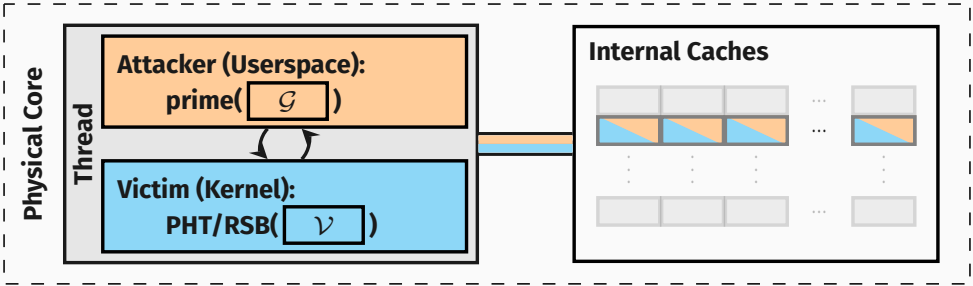


Generic Attacks

MDS-style Attack



Meltdown-style Attack



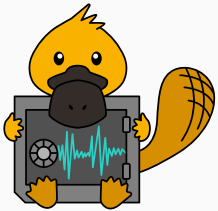
This must be slow?

NO!

It is EXTREMELY slow![1]

[1] With the current state-of-the-art.

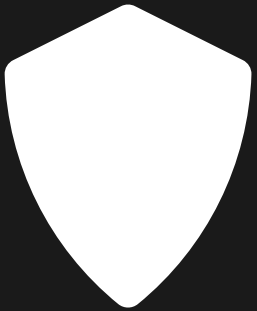
Software-based Power Side Channels



- **MDS-style:**
  4.82 bit/h
- **Meltdown-style (RSB):**
  0.84 bit/h

- **MDS-style:**
  0.065 to 0.68 bit/h
- **Meltdown-style estimate (PHT):**
  99.95 days/bit to 2.86 years/bit

Mitigations



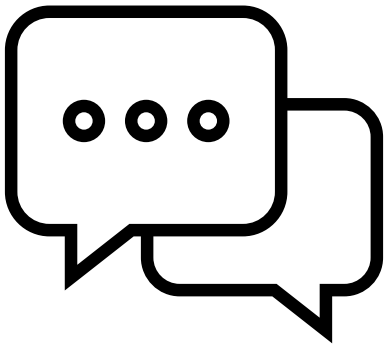
Mitigations

- **Preventing data collisions:**
  - **Redesign** of the complete shared data path
  - **Costly** to deploy
  - **Missed** components re-enable Collide+Power

Mitigations

- **Preventing observable power consumption:**
  - **Restricting** all direct power interfaces
- **Mitigating** Hertzbleed is challenging
  - Thermal and power management is required

→ **Collide+Power** is slow but unmitigated on modern CPUs!



Questions?