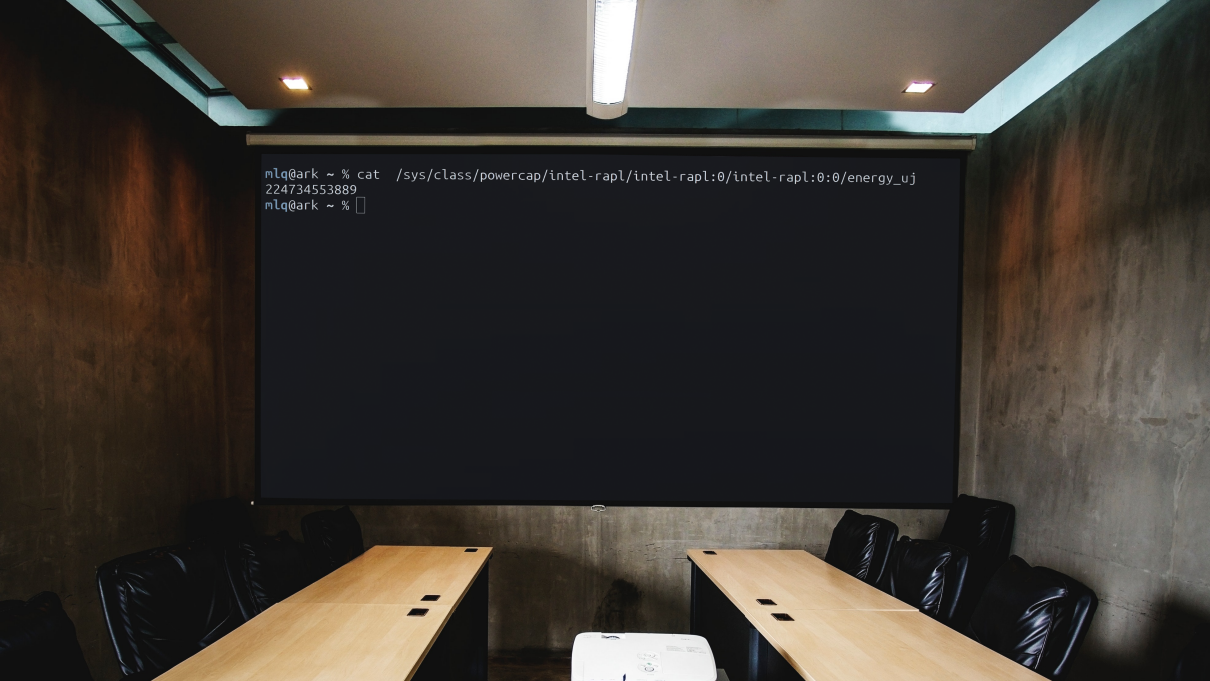
Warum leaked hier Strom?

Attacking CPUs with Power Side Channels from Software

Moritz Lipp, Michael Schwarz, Daniel Gruss, Andreas Kogler

December 27, 2020

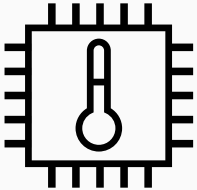

mlq@ark ~ % cat /sys/class/powercap/intel-rapl/intel-rapl:0/intel-rapl:0:0/energy_uj
224734553889
mlq@ark ~ %


0.0065
0.0060
0.0055
0.0050
0.0045
0.0040
0.0035
0.0030
9000
10000
11000
12000
13000
14000


Name 1
Name 2
Name 3
Name 4

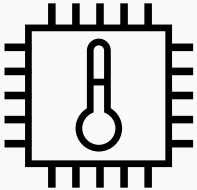
CPU Power Management

- **Thermal Design Power** (TDP) is the power consumption under the maximum theortical load of the CPU

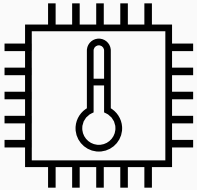


CPU Power Management

- **Thermal Design Power** (TDP) is the power consumption under the maximum theortical load of the CPU
  - Give integrators a target to find proper thermal solutions

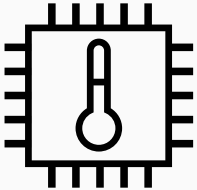


CPU Power Management



- **Thermal Design Power** (TDP) is the power consumption under the maximum theortical load of the CPU
  - Give integrators a target to find proper thermal solutions
  - For short period of times, more power can be consumed.

CPU Power Management



- **Thermal Design Power** (TDP) is the power consumption under the maximum theortical load of the CPU
  - Give integrators a target to find proper thermal solutions
  - For short period of times, more power can be consumed.

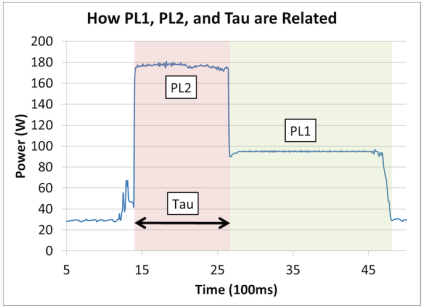


Figure 1: AnandTech

CPU Power Management

In order to save power, you can …

CPU Power Management

In order to save power, you can …

Shut down resources

CPU Power Management

In order to save power, you can …

Shut down resources

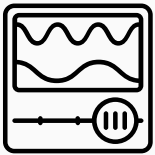
Reduce voltage

CPU Power Management

In order to save power, you can …

Shut down resources

Reduce voltage



Reduce frequency

Intel RAPL

- Need for Platform Thermal Management, Platform Power Limiting, Power/Performance Budgeting

Intel RAPL

- Need for Platform Thermal Management, Platform Power Limiting, Power/Performance Budgeting
- **Intel Running Average Power Limit** (RAPL) provides …

Intel RAPL

- Need for Platform Thermal Management, Platform Power Limiting, Power/Performance Budgeting
- **Intel Running Average Power Limit** (RAPL) provides …

power limiting

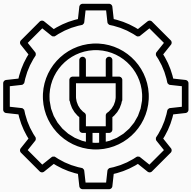
Intel RAPL

- Need for Platform Thermal Management, Platform Power Limiting, Power/Performance Budgeting
- **Intel Running Average Power Limit** (RAPL) provides …

power limiting

accurate energy reading

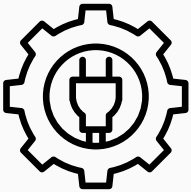
Intel RAPL



- On **Linux**, counters can be accessed using the `powercap` framework

  `/sys/devices/virtual/powercap/intel-rapl`

Intel RAPL

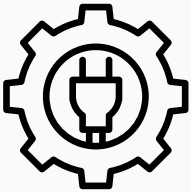


- On **Linux**, counters can be accessed using the `powercap` framework

  `/sys/devices/virtual/powercap/intel-rapl`

- On **macOS** and **Windows**, a driver from Intel needs to be installed

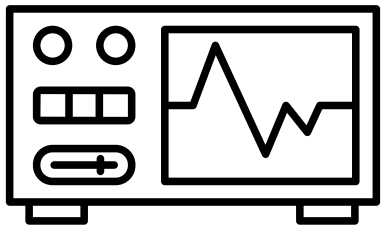
Intel RAPL



- On **Linux**, counters can be accessed using the `powercap` framework

  `/sys/devices/virtual/powercap/intel-rapl`

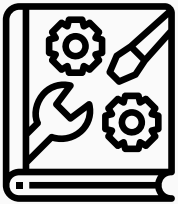
- On **macOS** and **Windows**, a driver from Intel needs to be installed
  - Graham Sutherland (@gsuberland) found other Windows drivers exposing the MSRs



What can we do with this?

Distinguishing Instructions

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



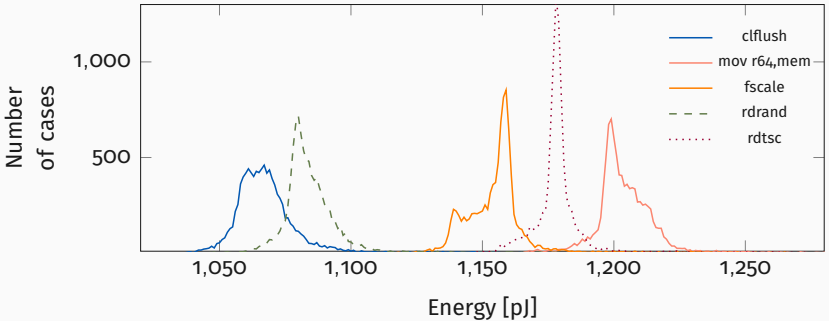
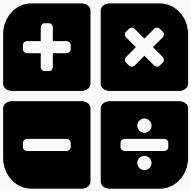


Figure 2: A histogram of the power consumption of various instructions on the i7-6700K (desktop) system.

Distinguishing Operands



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

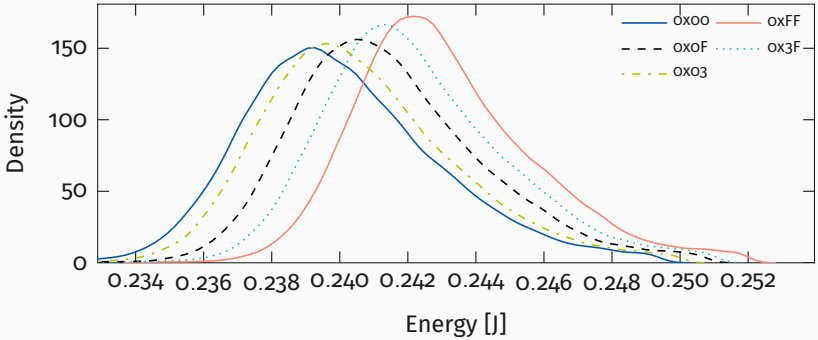


Figure 3: Measured energy consumption of the `imul` instruction with one operand fixed to 8 and the other varying in its Hamming weight.

Distinguishing Data

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

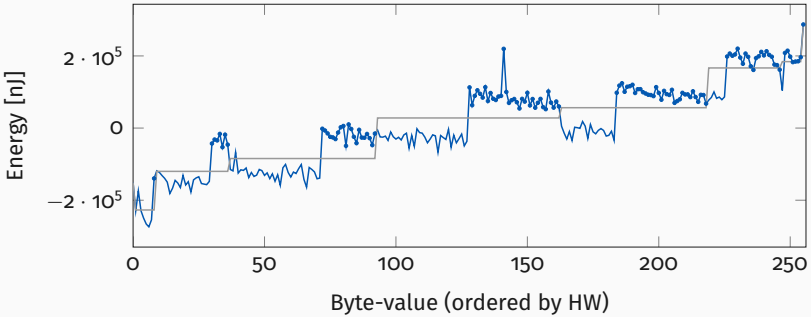
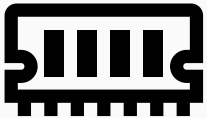


Figure 4: Energy consumption of the `movb` instruction for all byte values, ordered by Hamming Weight (HW) and value. The circle marks values where the most-significant bit is set.

Distinguishing Load Targets

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



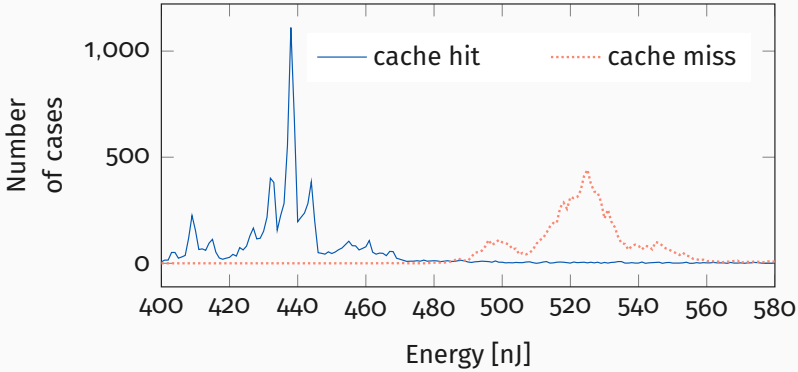


Figure 5: Using RAPL to distinguish whether the target of a memory load is cached (cache hit) or not (DRAM access).

Let’s exploit this!

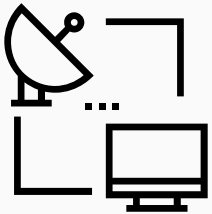
Covert Channel

- Communication Channel between two parties that are not allowed to communicate

Covert Channel

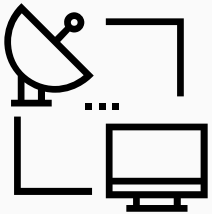
- Communication Channel between two parties that are not allowed to communicate
- Leveraging the power side channel

Covert Channel



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

Covert Channel



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

$\rightarrow$ Deduces transmitted bit

Covert Channel

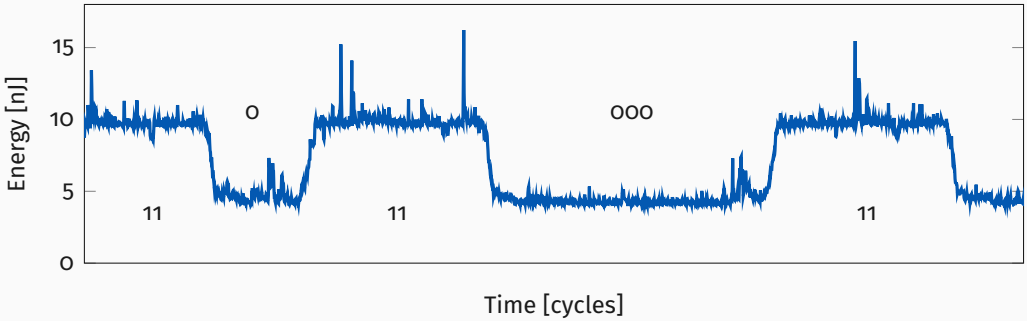
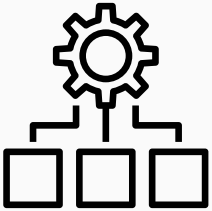


Figure 6: Transmission of bits `1101100011` using the time-less covert channel.

Covert Channel



- Not limited to 2 processes
- **Xen Hypervisor** granted guests access to the RAPL registers
- Establish a covert channel between 2 guests

Covert Channel

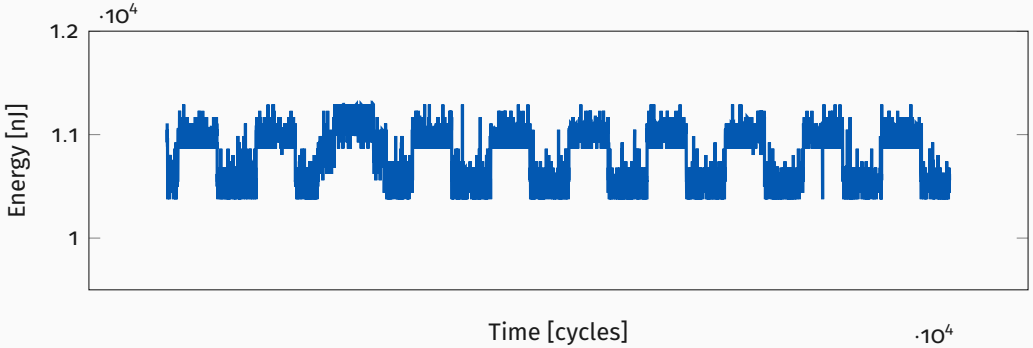
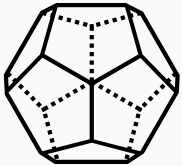


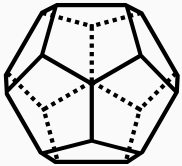
Figure 7: Transmission of bits between 2 Xen guests.

Breaking KASLR



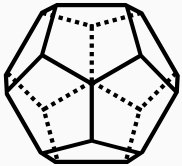
- Kernel Address Space Layout Randomization (KASLR) randomizes kernel location

Breaking KASLR



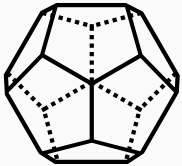
- Kernel Address Space Layout Randomization (KASLR) randomizes kernel location
- Exploit **energy consumption differences** between

Breaking KASLR



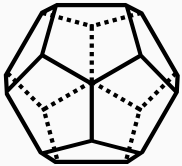
- Kernel Address Space Layout Randomization (KASLR) randomizes kernel location
- Exploit **energy consumption differences** between
  - Mapped addresses

Breaking KASLR



- Kernel Address Space Layout Randomization (KASLR) randomizes kernel location
- Exploit **energy consumption differences** between
  - Mapped addresses
  - Unmapped addresses

Breaking KASLR



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

Breaking KASLR

Figure 8: Page-table walks for unmapped pages require more power

Breaking KASLR

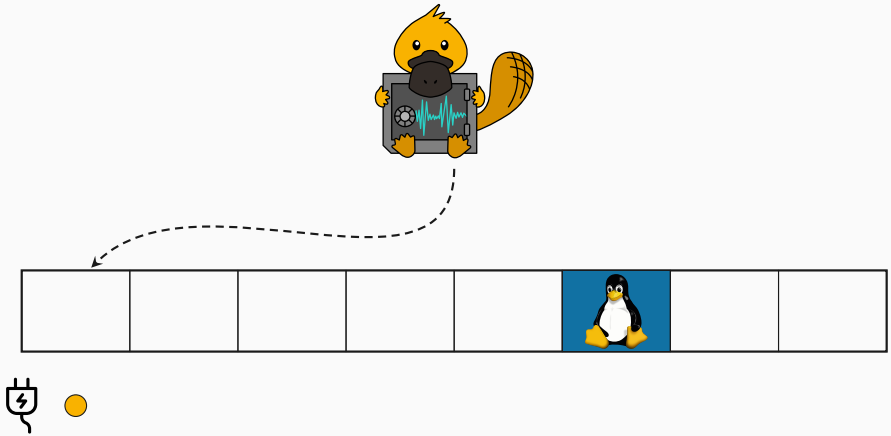


Figure 8: Page-table walks for unmapped pages require more power

Breaking KASLR

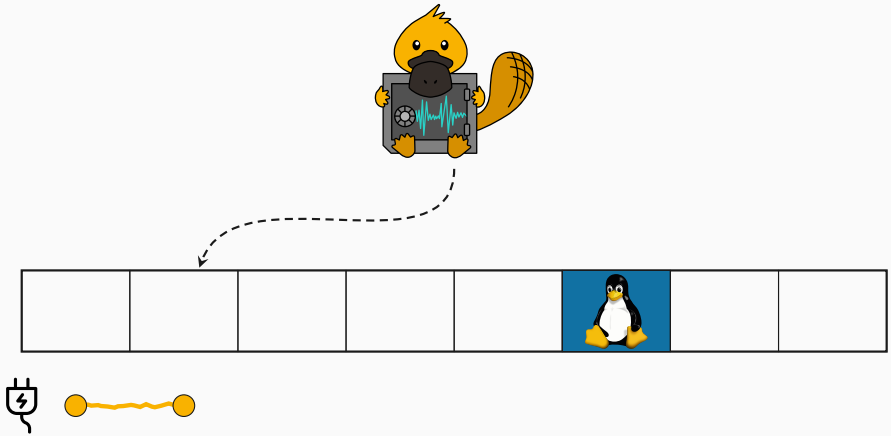


Figure 8: Page-table walks for unmapped pages require more power

Breaking KASLR

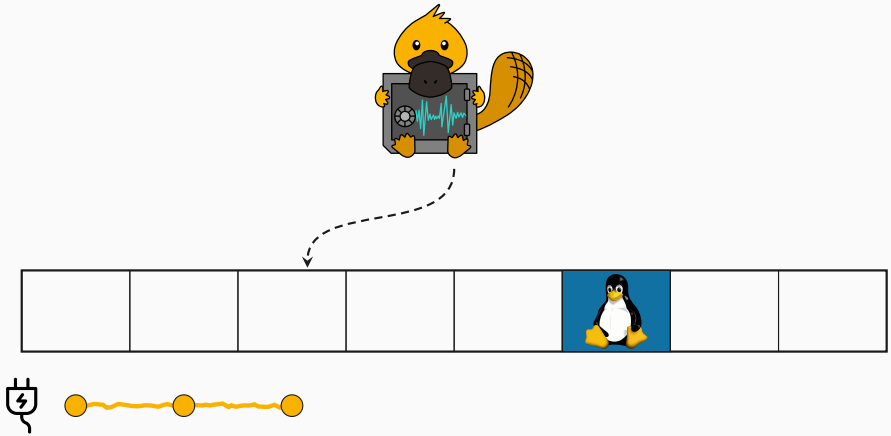


Figure 8: Page-table walks for unmapped pages require more power

Breaking KASLR

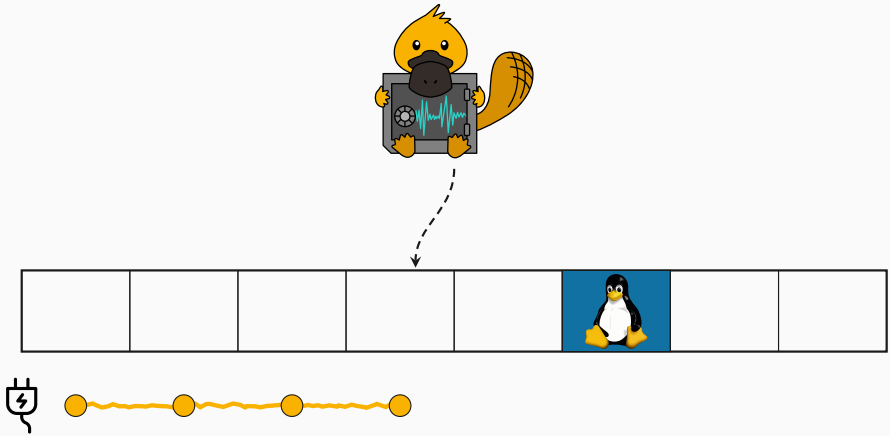


Figure 8: Page-table walks for unmapped pages require more power

Breaking KASLR

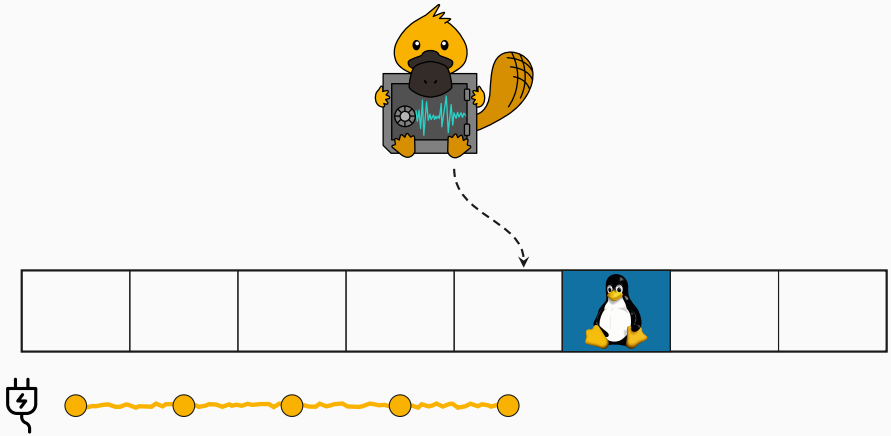


Figure 8: Page-table walks for unmapped pages require more power

Breaking KASLR

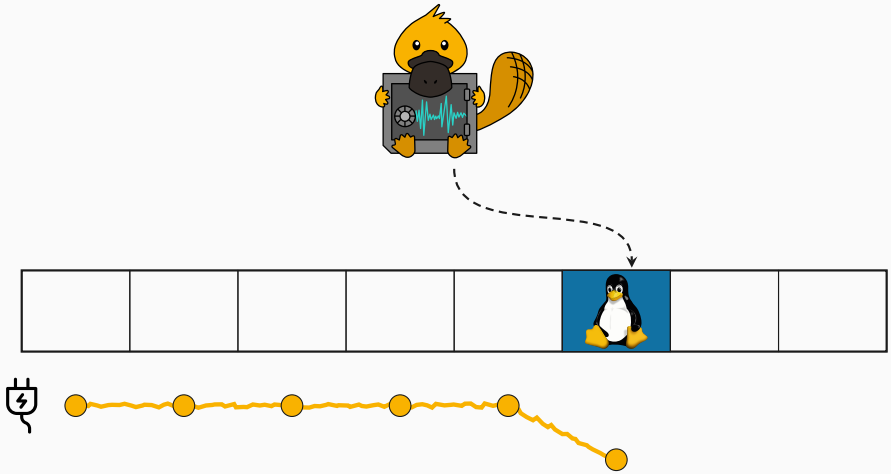


Figure 8: Page-table walks for unmapped pages require more power

Breaking KASLR

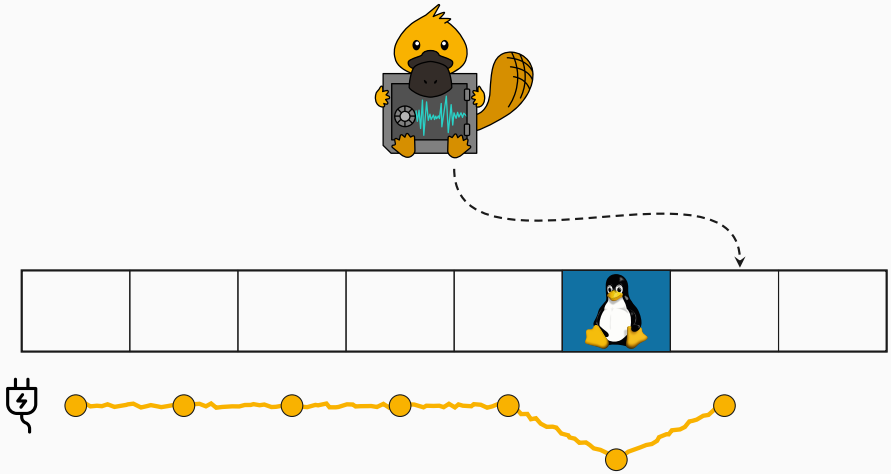


Figure 8: Page-table walks for unmapped pages require more power

Breaking KASLR

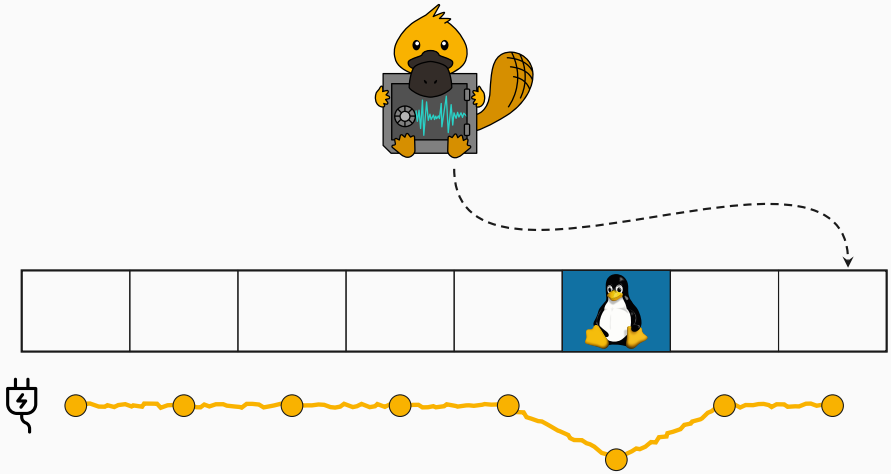
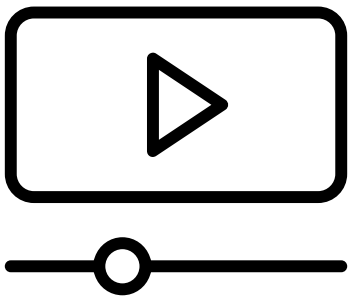
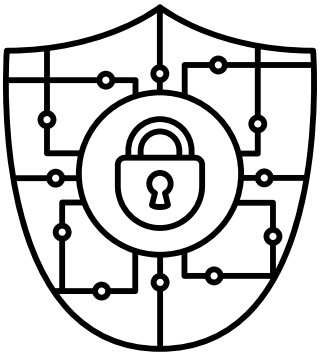


Figure 8: Page-table walks for unmapped pages require more power

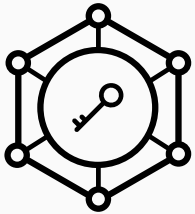


Demo Video



We want real attacks: Let’s attack crypto!

Attacking RSA



- RSA is a **widely-used** public-key cryptosystem

Encrypt with public key

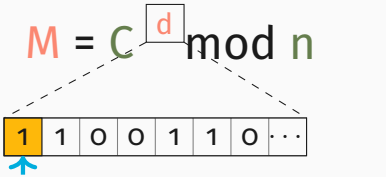
Decrypt with private key

- Get the private key: Profit!

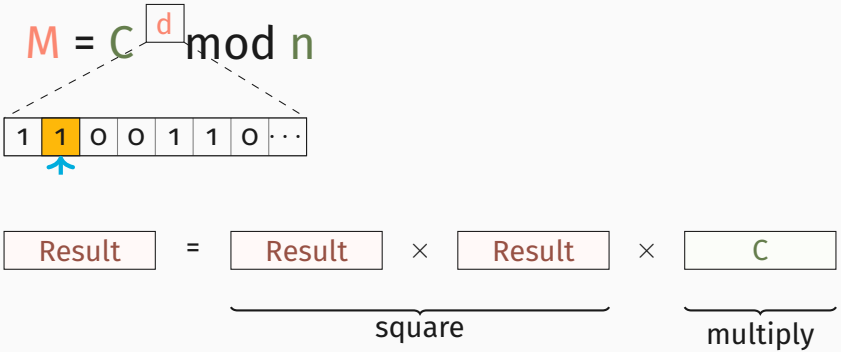
Attacking RSA

$$M = C^{d} \bmod n$$

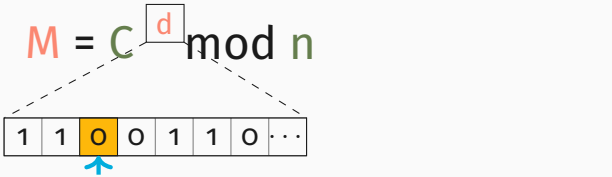
Attacking RSA



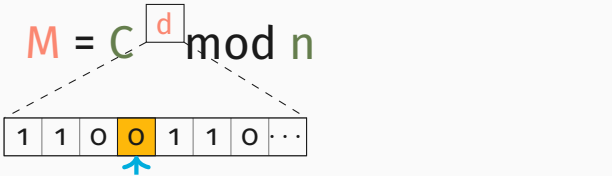
Attacking RSA



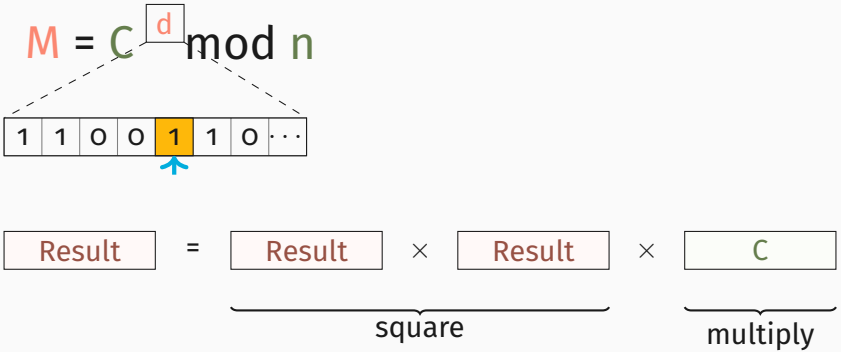
Attacking RSA



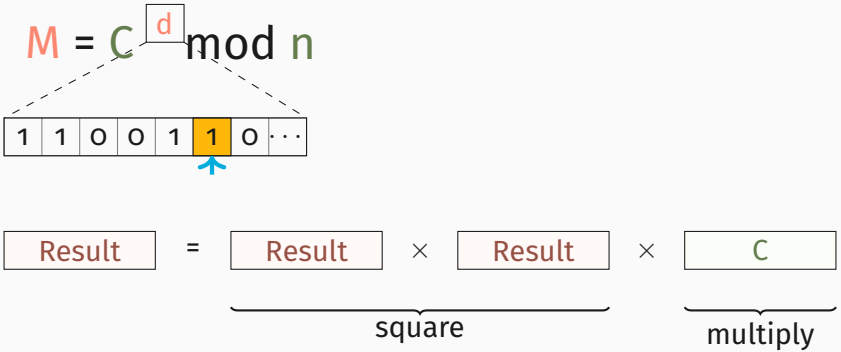
Attacking RSA



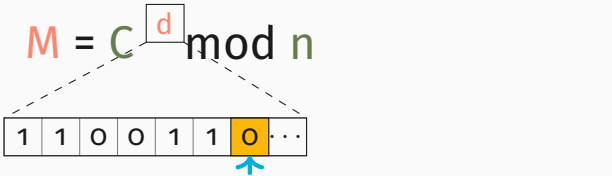
Attacking RSA



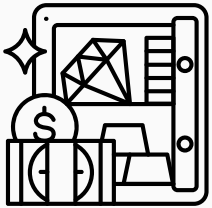
Attacking RSA



Attacking RSA

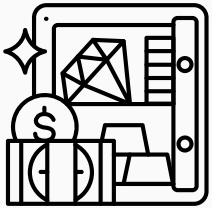


Attacking Intel SGX



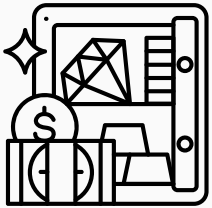
- Instruction-set extension

Attacking Intel SGX



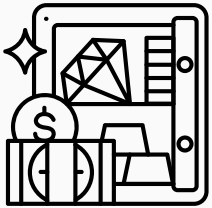
- Instruction-set extension
- Integrity and confidentiality of code and data in **untrusted environments**

Attacking Intel SGX



- Instruction-set extension
- Integrity and confidentiality of code and data in **untrusted environments**
- Run programs in **enclaves** using protected areas of memory

Attacking Intel SGX



- Instruction-set extension
- Integrity and confidentiality of code and data in **untrusted environments**
- Run programs in **enclaves** using protected areas of memory
- Operating system can be **compromised**

Taming the enclaves

SGX-Step: A Practical Attack Framework for Precise Enclave Execution Control

Jo Van Bulck
imec-DistriNet, KU Leuven
jo.vanbulck@cs.kuleuven.be

Frank Piessens
imec-DistriNet, KU Leuven
frank.piessens@cs.kuleuven.be

Raoul Strackx
imec-DistriNet, KU Leuven
raoul.strackx@cs.kuleuven.be

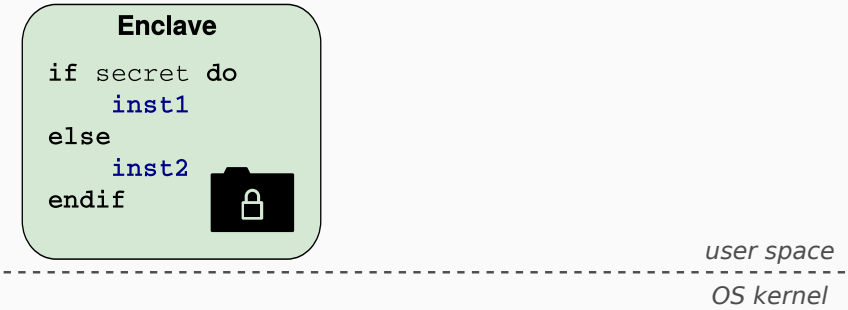
Abstract

Protected module architectures such as Intel SGX hold the promise of protecting sensitive computations from a potentially compromised operating system. Recent research convincingly demonstrated, however, that SGX's strengthened adversary model also gives rise to to a new class of powerful, low-noise side-channel attacks leveraging first-rate control over hardware. These attacks commonly rely on frequent enclave preemptions to obtain fine-grained side-channel observations. A maximal temporal resolution is achieved when the victim state is measured after every instruction. Current state-of-the-art enclave execution control schemes, however, do not generally achieve such instruction-level granularity.

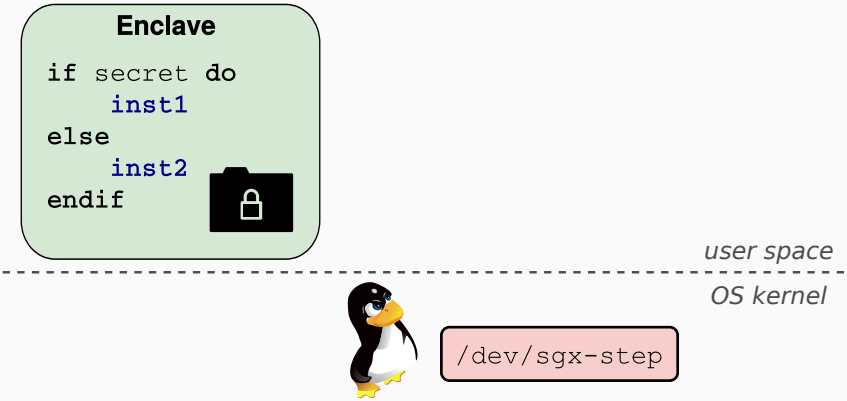
This paper presents SGX-Step, an open-source Linux kernel framework that allows an untrusted host process to

concerns, the past years have seen a significant research effort [3, 6, 9] on Protected Module Architectures (PMAs) that support isolated execution of security-sensitive application components or *enclaves* with a minimal Trusted Computing Base (TCB). These proposals have in common that they enforce security primitives directly in hardware, or in a small hypervisor, so as to prevent the untrusted OS from accessing enclaved code or data directly, while still leaving it in charge of shared platform resources such as system memory or CPU time. With the arrival of Intel's Software Guard eXtensions (SGX) [6, 7], such strong hardware-enforced trusted computing guarantees are now available on mainstream consumer devices.

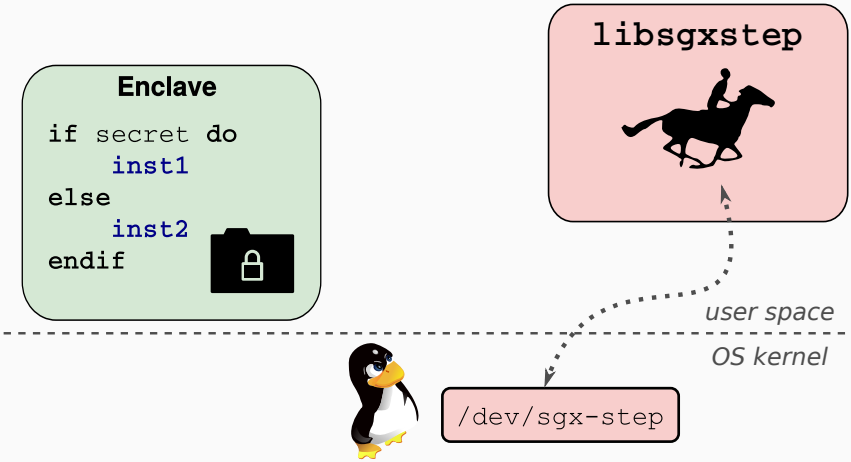
Recent research demonstrated, however, that the increased capabilities of a privileged PMA attacker allow her to con-


Enclave
if secret do
inst1
else
inst2
endif
user space
OS kernel

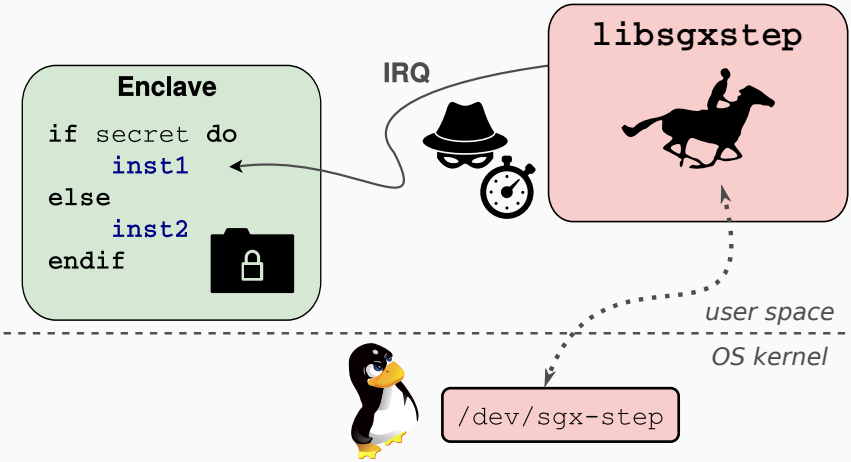
SGX-Step


Enclave
if secret do
inst1
else
inst2
endif
user space
OS kernel
/dev/sgx-step

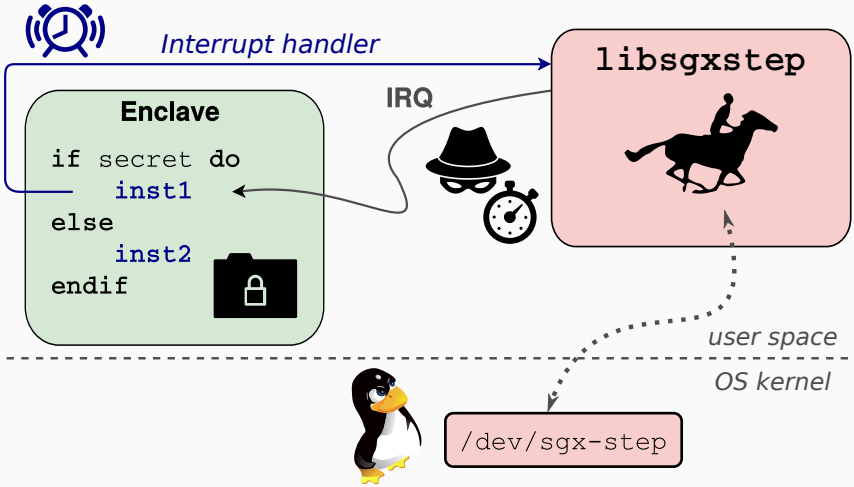
SGX-Step


Enclave
if secret do
inst1
else
inst2
endif
libsgxstep
user space
OS kernel
/dev/sgx-step

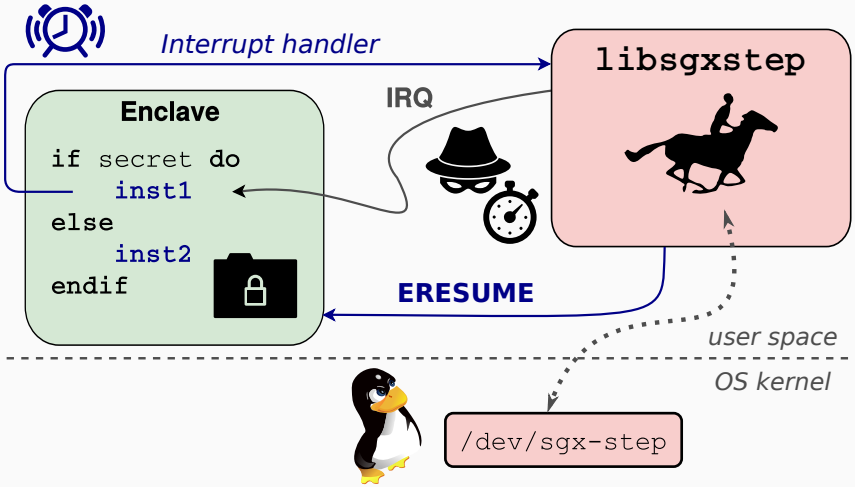
SGX-Step


libsgxstep
IRQ
Enclave
if secret do
inst1
else
inst2
endif
user space
OS kernel
/dev/sgx-step

SGX-Step


Interrupt handler
libsgxstep
IRQ
Enclave
if secret do
inst1
else
inst2
endif
user space
OS kernel
/dev/sgx-step

SGX-Step


Interrupt handler
libsgxstep
IRQ
Enclave
if secret do
inst1
else
inst2
endif
ERESUME
user space
OS kernel
/dev/sgx-step

SGX Step

- **SGX-step** is an open-source Linux kernel framework

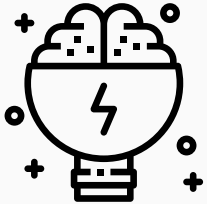
SGX Step

- **SGX-step** is an open-source Linux kernel framework
- Configure APIC timer interrupts

SGX Step

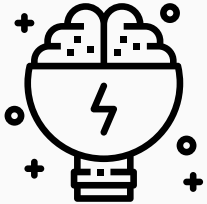
- **SGX-step** is an open-source Linux kernel framework
- Configure APIC timer interrupts
- Single and zero-step enclave execution

SGX Step



- **Combine** Intel RAPL with SGX-step

SGX Step

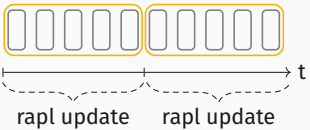


- **Combine** Intel RAPL with SGX-step
- Measure the energy consumption of **single instructions**

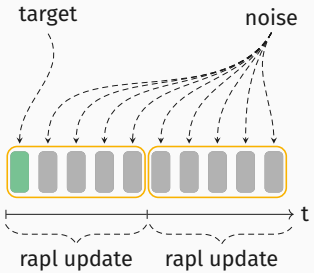
SGX Step + RAPL

target noise

- Measure an **instruction** by

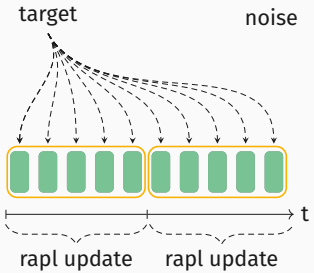


SGX Step + RAPL



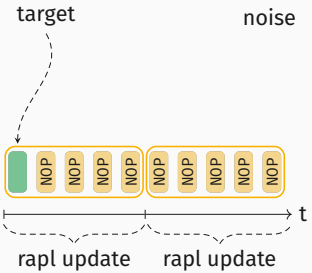
- Measure an **instruction** by
  - executing it once

SGX Step + RAPL



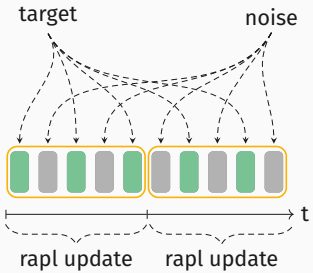
- Measure an **instruction** by
  - executing it once
  - executing it repeatedly

SGX Step + RAPL



- Measure an **instruction** by
  - executing it once
  - executing it repeatedly
  - padding it with known instructions

SGX Step + RAPL



- Measure an **instruction** by
  - executing it once
  - executing it repeatedly
  - padding it with known instructions
  - using SGX-Step to reissue the instruction

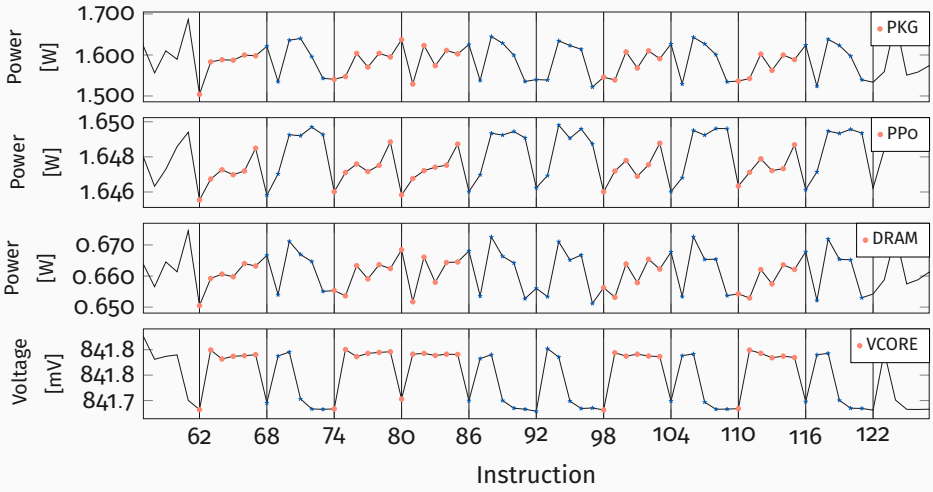
Toy Cipher

- Enclave imitating Square-and-multiply

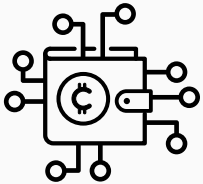
Toy Cipher

- Enclave imitating Square-and-multiply
- Single step every instruction
- Measure the energy consumption of several zero steps

RSA Toy Cipher

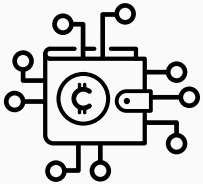


Attacking mbed TLS



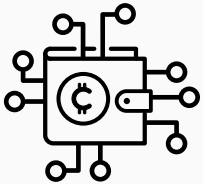
- **Extract RSA** key from mbed TLS 2.13.0

Attacking mbed TLS



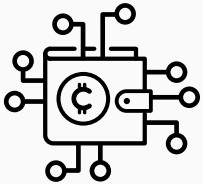
- **Extract RSA** key from mbed TLS 2.13.0
- Square-and-multiply algorithm

Attacking mbed TLS



- **Extract RSA** key from mbed TLS 2.13.0
- Square-and-multiply algorithm
- Multiplication function uses AVX memset

Attacking mbed TLS



- **Extract RSA** key from mbed TLS 2.13.0
- Square-and-multiply algorithm
- Multiplication function uses AVX memset
- Number of instructions executed depends on the key

Attacking mbed TLS

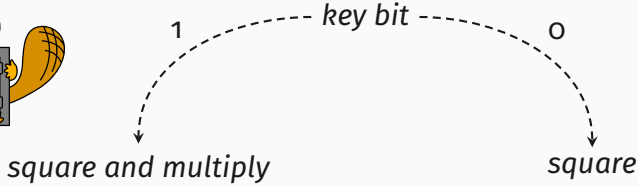
key bit

Attacking mbed TLS

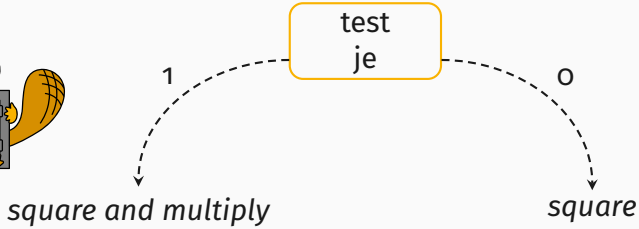
1 ------ *key bit*

square and multiply

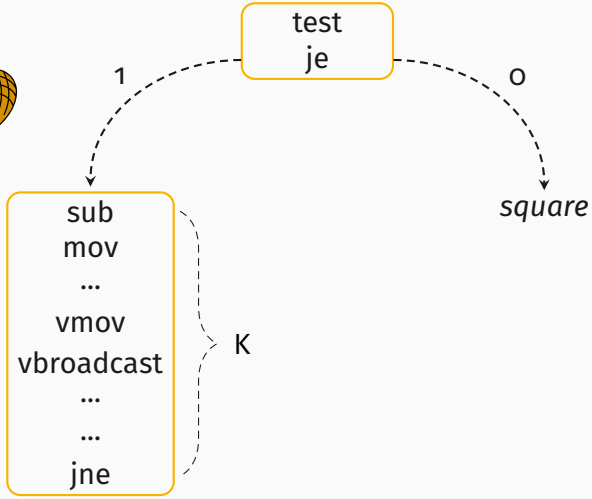
Attacking mbed TLS



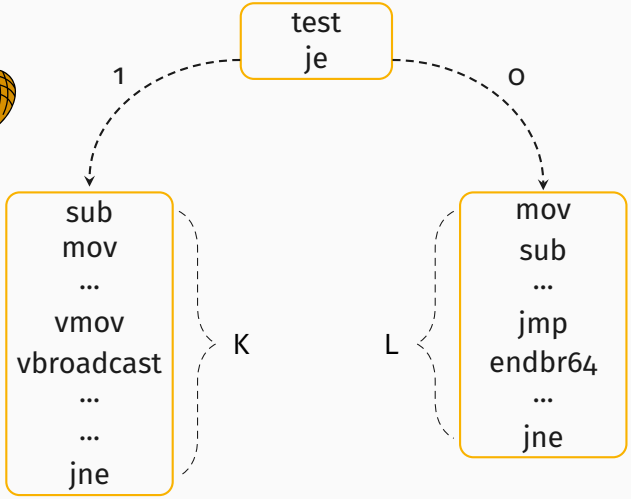
Attacking mbed TLS



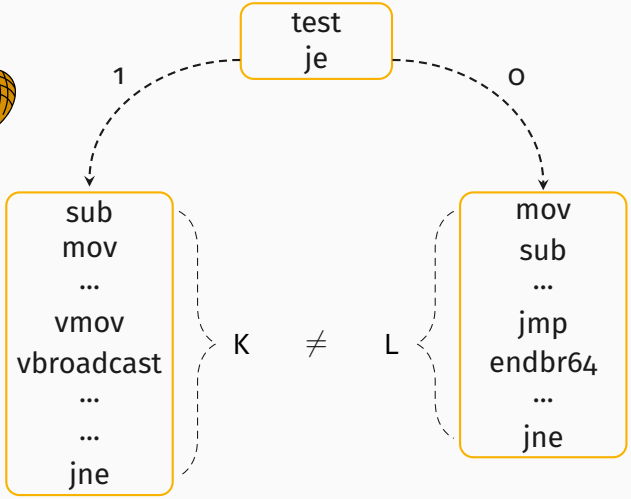
Attacking mbed TLS



Attacking mbed TLS



Attacking mbed TLS

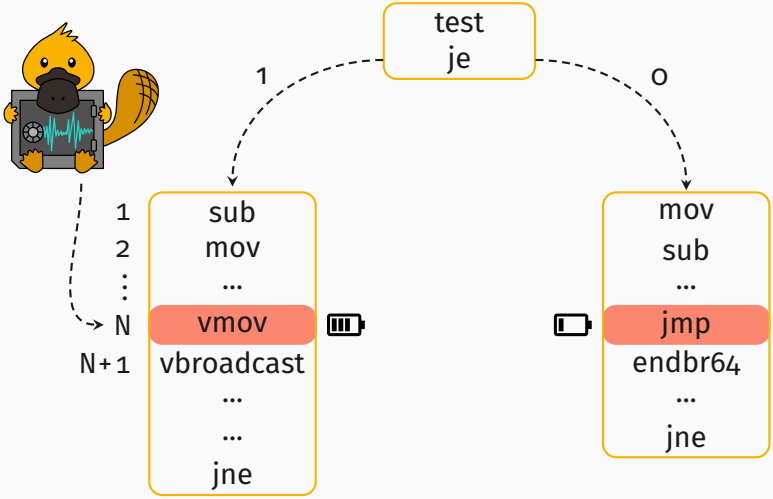


Attacking mbed TLS

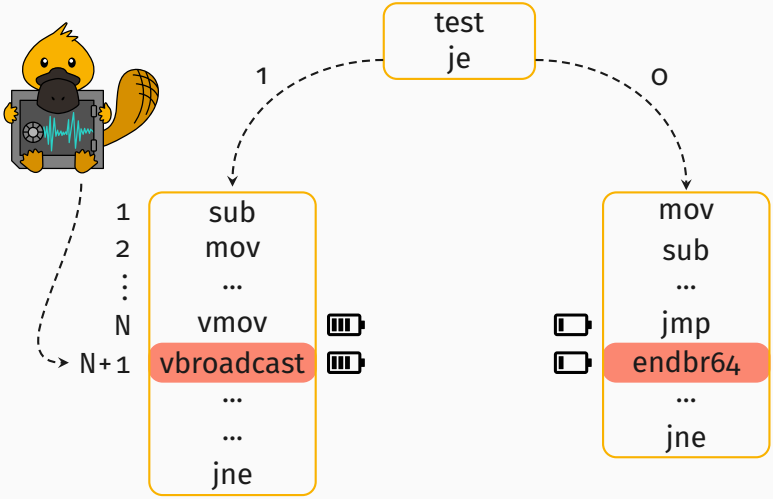
Attacking mbed TLS

Attacking mbed TLS

Attacking mbed TLS

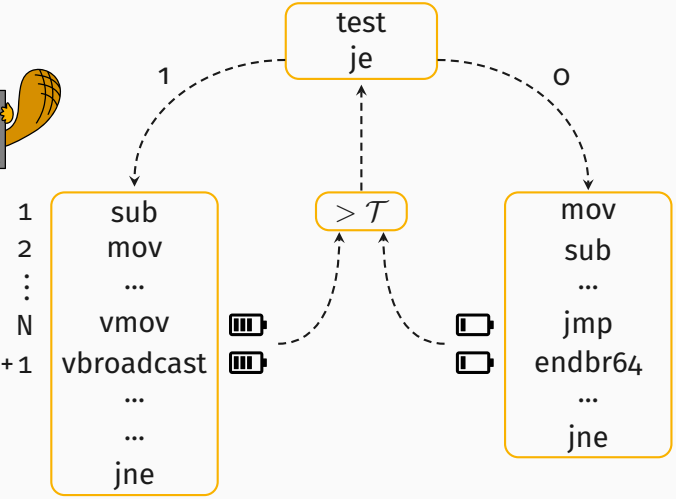


Attacking mbed TLS

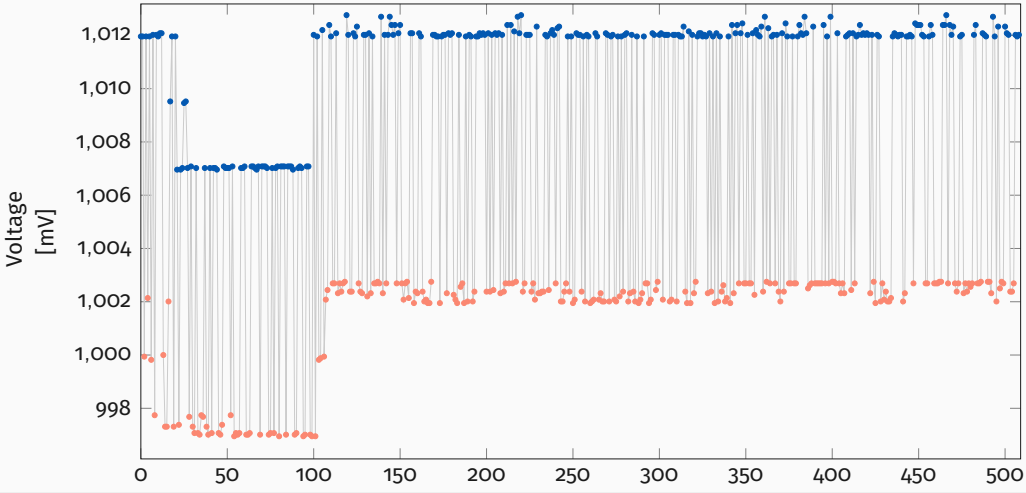


Attacking mbed TLS

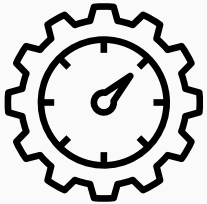
Attacking mbed TLS



RSA Toy Cipher

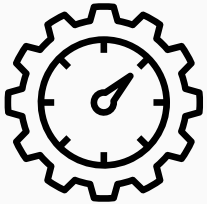


Performance



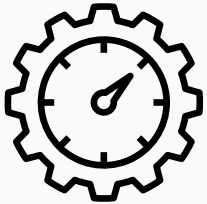
- Time per key bit increases linearly based on the index

Performance



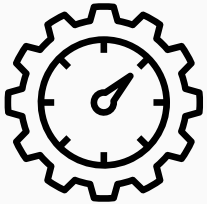
- Time per key bit increases linearly based on the index
- 3h 31m for a **512 bit**

Performance



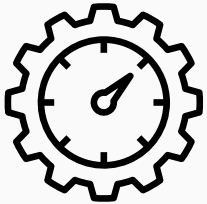
- Time per key bit increases linearly based on the index
- 3h 31m for a **512 bit**
  - 52 minutes spend for finding target instruction

Performance



- Time per key bit increases linearly based on the index
- 3h 31m for a **512 bit**
  - 52 minutes spend for finding target instruction
- Record 3 samples per key bit

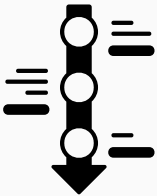
Performance



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

Timeline

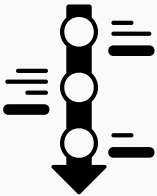
2017



Timeline

2017

February 27 — Discovery of the RAPL interface

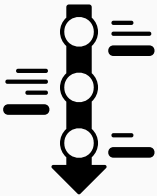


Timeline

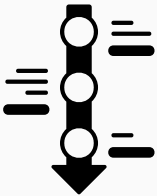
2017

February 27 — Discovery of the RAPL interface

March 31 — First version of the lyrics



Timeline



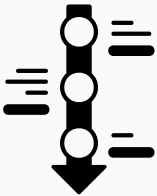
2017

February 27 — Discovery of the RAPL interface

March 31 — First version of the lyrics

May 18 — Coming up with this talk title

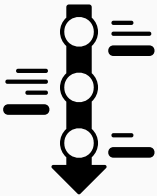
Timeline



2017

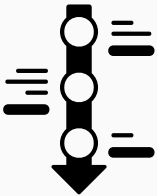
February 27	Discovery of the RAPL interface
March 31	First version of the lyrics
May 18	Coming up with this talk title
Fall	First toy attack on RSA + covert channel

Timeline



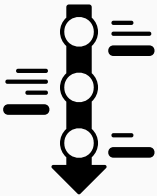
2017	
February 27	Discovery of the RAPL interface
March 31	First version of the lyrics
May 18	Coming up with this talk title
Fall	First toy attack on RSA + covert channel
2018	

Timeline



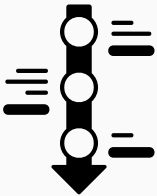
2017	
February 27	Discovery of the RAPL interface
March 31	First version of the lyrics
May 18	Coming up with this talk title
Fall	First toy attack on RSA + covert channel
2018	
Spring	KASLR break + 8-bit blocks for RSA

Timeline



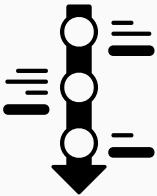
2017	
February 27	Discovery of the RAPL interface
March 31	First version of the lyrics
May 18	Coming up with this talk title
Fall	First toy attack on RSA + covert channel
2018	
Spring	KASLR break + 8-bit blocks for RSA
2019	

Timeline



2017	
February 27	Discovery of the RAPL interface
March 31	First version of the lyrics
May 18	Coming up with this talk title
Fall	First toy attack on RSA + covert channel
2018	
Spring	KASLR break + 8-bit blocks for RSA
2019	
September	Full attack on RSA

Timeline



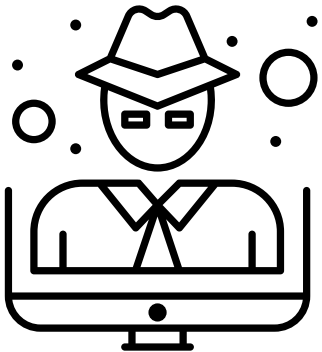
2017	
February 27	Discovery of the RAPL interface
March 31	First version of the lyrics
May 18	Coming up with this talk title
Fall	First toy attack on RSA + covert channel
2018	
Spring	KASLR break + 8-bit blocks for RSA
2019	
September	Full attack on RSA
November	Submission, Responsible Disclosure + Start of Embargo

Embargo

- Issue was embargoed for **almost a year**

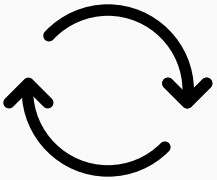
Embargo

- Issue was embargoed for **almost a year**
- Disclosure date: Nov 10th, 2020



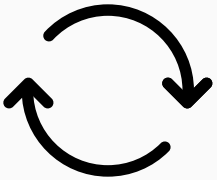
Crypto Attacks from User Space

Repeat Repeat Repeat



- **Difficult** to measure parts without SGX-step

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:

Correlation Power Analysis

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

Hamming Weight

Number of bits set

Correlation Power Analysis

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

Hamming Weight

Number of bits set

Hamming Distance

Bits flipping between operations

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

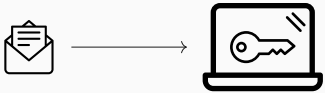
CPA Attack

- **SGX Minimal I/O noise**: 2 million traces in 26 hours
- **SGX Real-world conditions**: 16 million traces in 277 hours
- **Kernel**: 4 million traces in 50 hours

CPA Attack

CPA Attack

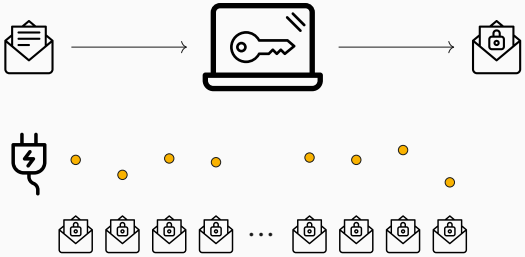
CPA Attack



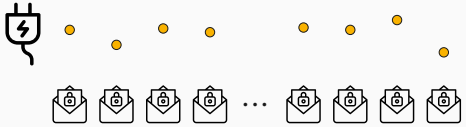
CPA Attack

CPA Attack

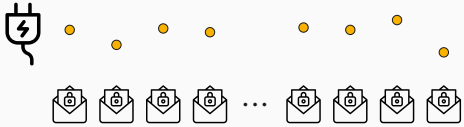
CPA Attack



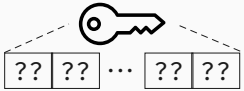
CPA Attack

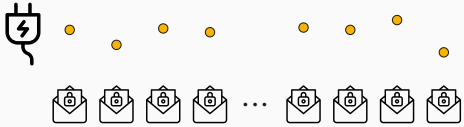


CPA Attack

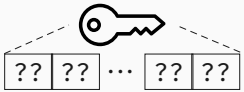


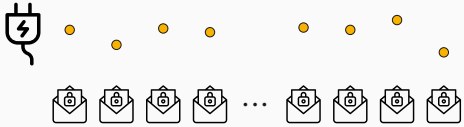
CPA Attack



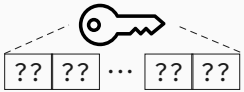


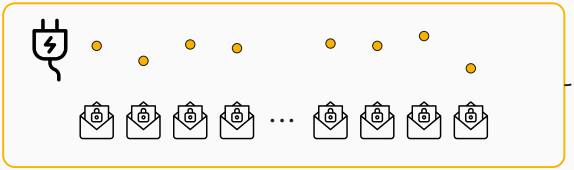
CPA Attack



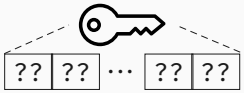


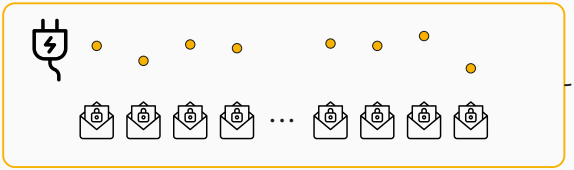
CPA Attack



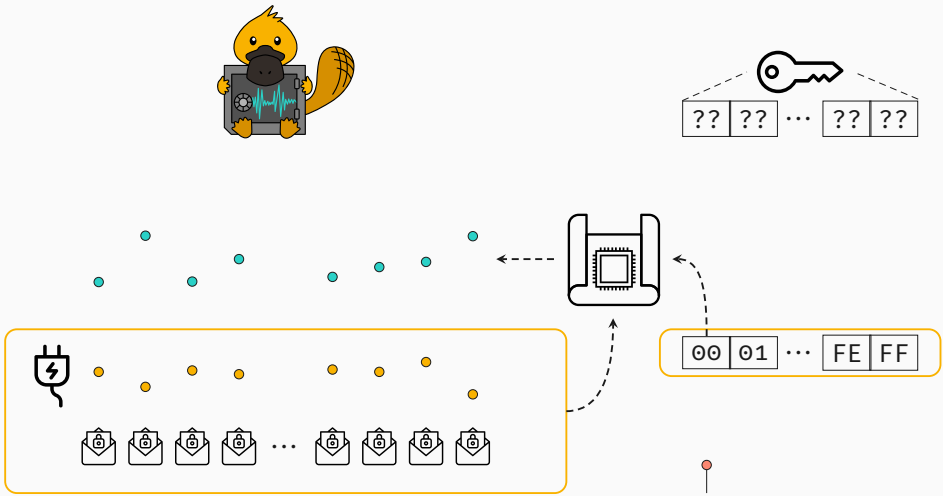


CPA Attack

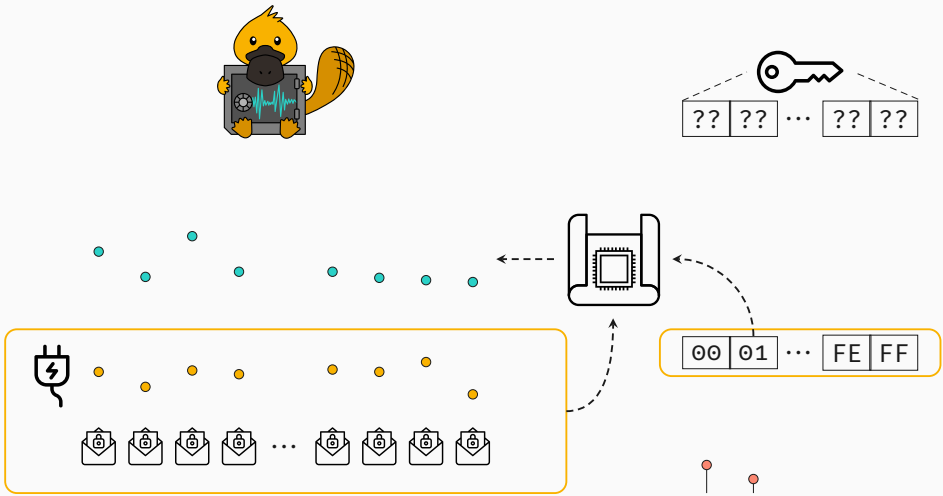




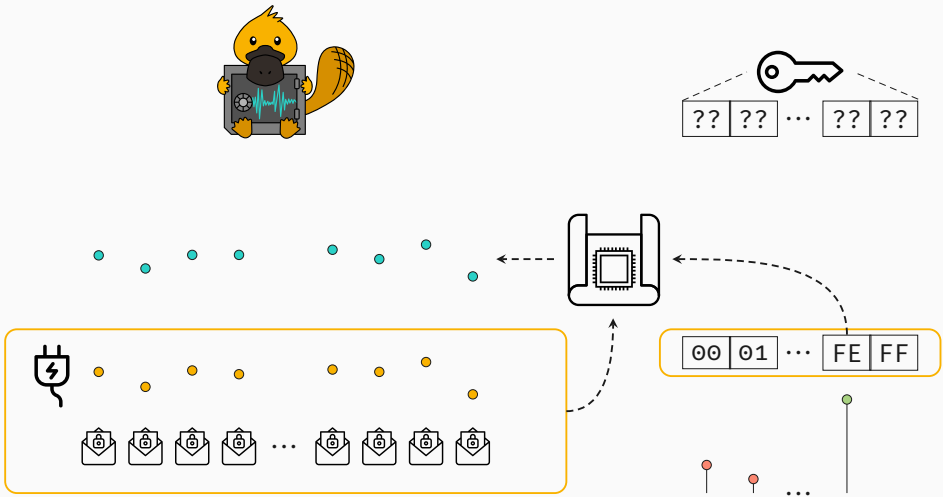
CPA Attack



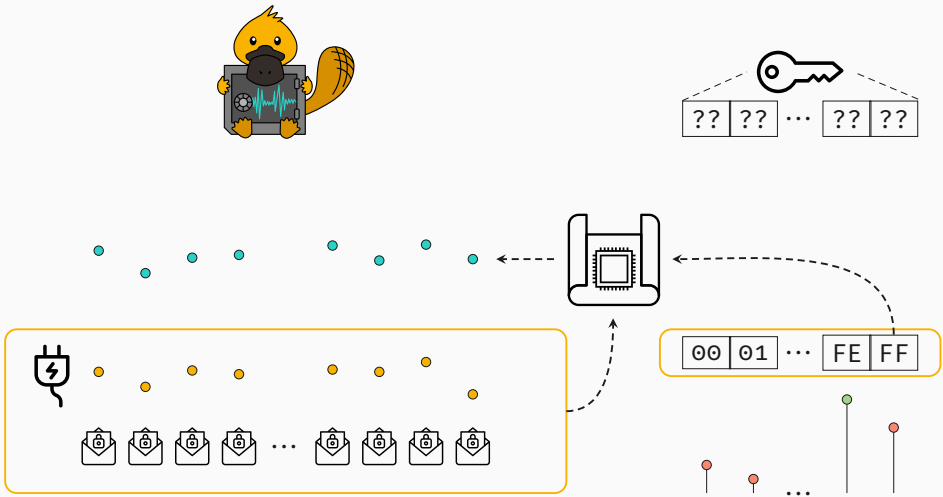
CPA Attack



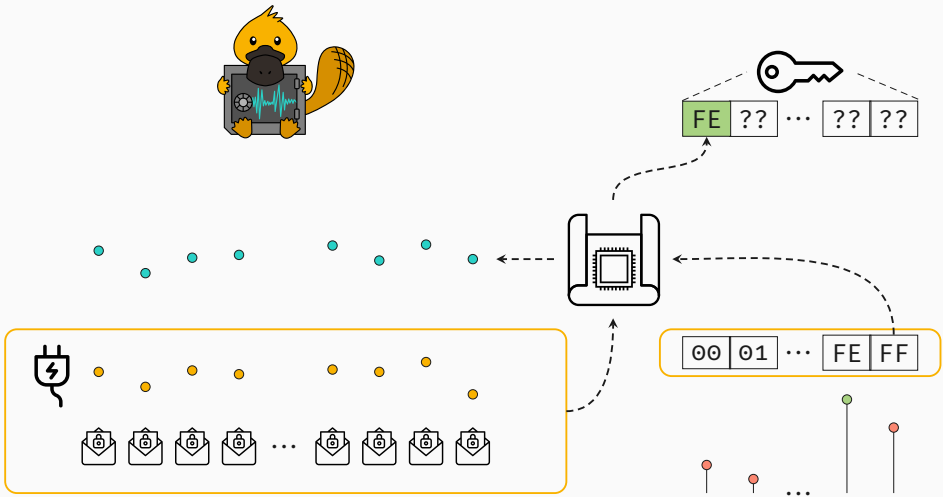
CPA Attack



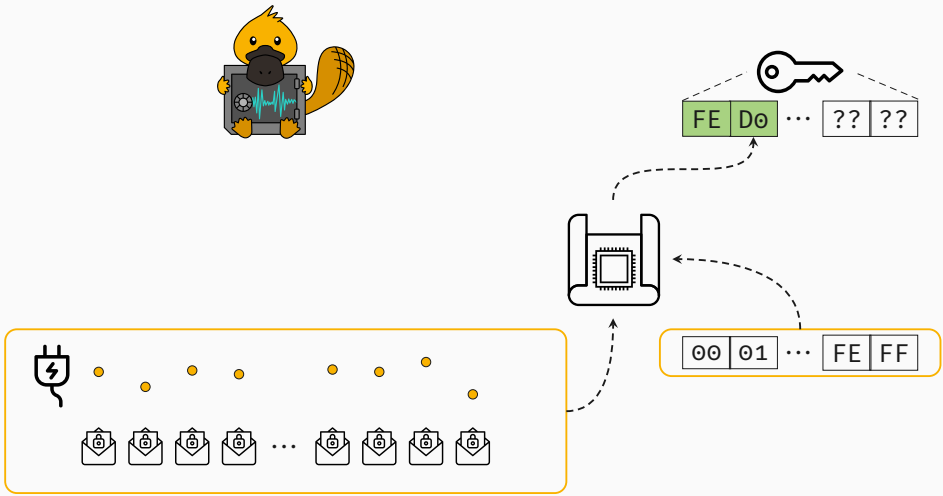
CPA Attack



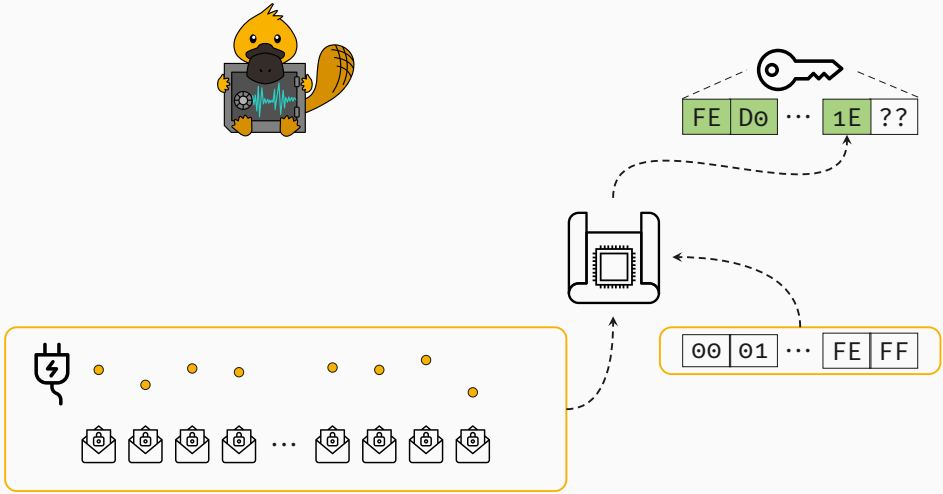
CPA Attack



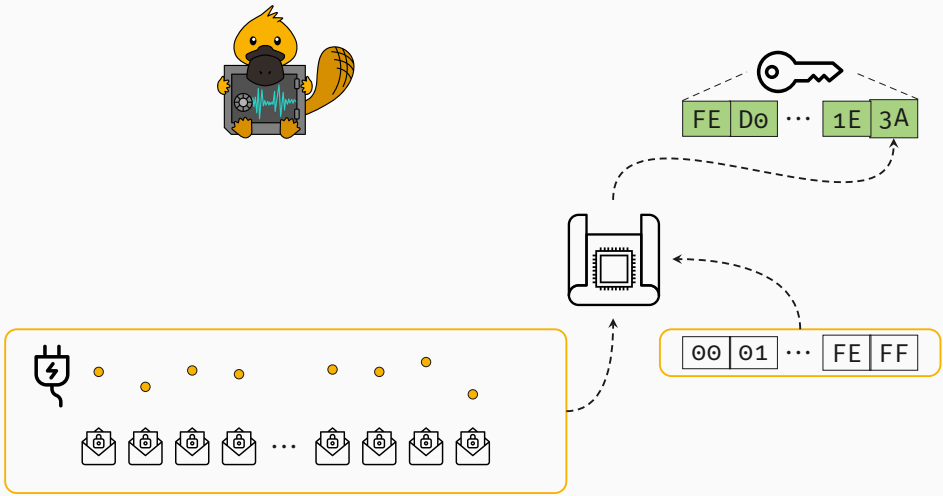
CPA Attack

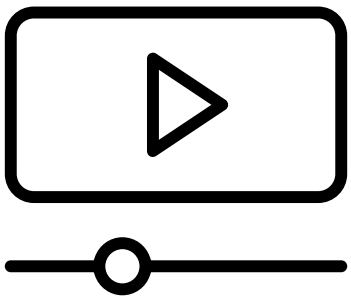


CPA Attack



CPA Attack





Demo Video

Other CPU Vendors

- AMD affected as well

Other CPU Vendors

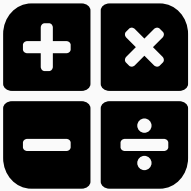
- AMD affected as well
- Never heard back after disclosure

Other CPU Vendors

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

Distinguishing Operands on AMD

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



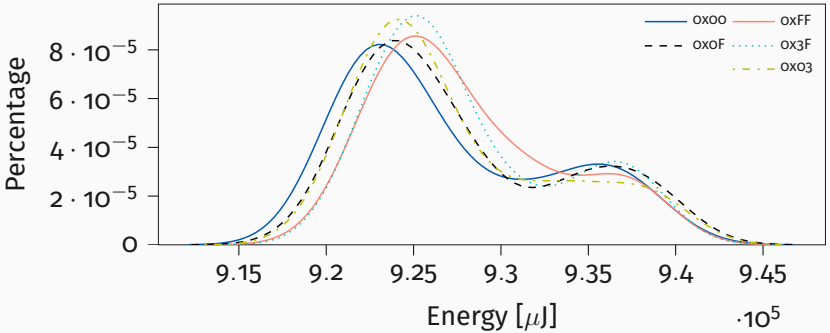
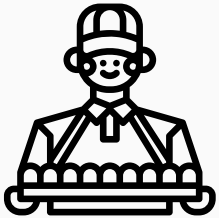


Figure 9: Measured energy consumption of the `shr` instruction with varying Hamming weights.

Other CPU Vendors



Other CPUs also have interfaces for measuring power

- Some ARM development boards (odroid XU+E, SAML11)
- NVIDIA Jetson TX2
- IBM POWER9
- Marvell ThunderX2
- Ampere Altra
- Hygon Dhyana CPU family

Media Coverage

heise online heise+

IT Mobiles Entertainment Wissen Netzpolitik Wirtschaft Journal

TOPTHEMEN: XBOX SERIES X/S PS5 E-AUTO SECURITY WINDOWS 10 CORONAVIRUS

heise online) News) 11/2020) "Platypus": Sicherheitslücke missbraucht Messfunktion von...

"Platypus": Sicherheitslücke missbraucht Messfunktion von Intel-Prozessoren

Sicherheitsforscher nutzen die zur Leistungsmessung gedachte RAPL-Schnittstelle aktueller Prozessoren, um geheime Daten zu ergattern, die die CPU verarbeitet.

Lesezeit: 3 Min. In Pocket speichern 93

(Bild: Natascha Eibl/CC0 1.0)

Media Coverage

heise online

IT Mobiles Entertainment Wissen Netzpolitik Wirtschaft Journal

TOPTHEMEN: XBOX SERIES X/S PS5 E-AUTO SECURITY WINDOWS 10 CORONAVIRUS

heise online | News | 11/2020 | "Platypus": Sicherheitslücke missbraucht Messfunktion von...

"Platypus": Sicherheitslücke missbraucht Messfunktion von Intel-Prozessoren

Sicherheitsforscher nutzen die zur Leistungsmessung gedachte RAPL-Schnittstelle aktueller Prozessoren, um geheime Daten zu ergattern, die die CPU verarbeitet.

Lesezeit: 3 Min. In Pocket speichern 93

(Bild: Natascha Eibl/CC0 1.0)

golem.de IT-NEWS FÜR PROFIS

HOME TICKER VIDEOS VORGELESEN FORUM | ANMELDEN

golem.de jetzt werbefrei lesen

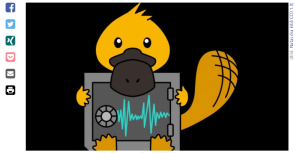
IT-KARRIERE: STELLENMARKT AKADEMIE SHOP | IT-KÖPFE GEHALTSCHECK | SERVICES: GOLEM PC PREISVERGLEICH TOP-ANGEBOTE

SICHERHEITSLÜCKE PLATYPUS

Passwörter per Leistungsmessung aus der CPU ausleiten

Wieder gelingt es einem Forscherteam, eigentlich geschützte Daten aus Intel-CPUs abzuleiten. Diesmal per Software-Leistungsmessung.

11. November 2020, 13:01 Uhr, Sebastian Grüner



Der Platypus-Angriff ist noch Schnäbelchen beigetan.

Eine Forschergruppe der Unis Graz und Birmingham sowie des Cispa-Helmholtz-Zentrums hat eine alte Angriffsmethode auf CPUs neu aufgelegt und den Angriff Platypus (Power Leakage Attacks Targeting Your Protected User Secrets) getauft. Das Team nutzt dazu minimale Unterschiede in der Leistungsaufnahme von CPUs, um geheimes Schlüsselmaterial aus der CPU auszuleiten.

Media Coverage

heise online

IT Mobiles Entertainment Wissen Netzpolitik Wirtschaft Journal

TOPTHEMEN: XBOX SERIES X/S PS5 E-AUTO SECURITY WINDOWS 10 CORONAVIRUS

heise online | News | 11/2020 | "Platypus": Sicherheitslücke missbraucht Messfunktion von...

"Platypus": Sicherheitslücke missbraucht Messfunktion von Intel-Prozessoren

Sicherheitsforscher nutzen die zur Leistungsmessung gedachte RAPL-Schnittstelle aktueller Prozessoren, um geheime Daten zu ergattern, die die CPU verarbeitet.

Lesezeit: 3 Min. In Pocket speichern 93

(Bild: Natascha Eibl/CC0 1.0)

ZDNet

SCANDINAVIA AFRICA UK ITALY SPAIN MORE NEWSLETTERS ALL WRITERS

New Platypus attack can steal data from Intel CPUs

Intel has released microcode updates today to prevent attackers from abusing the Intel RAPL mechanism to steal sensitive data from its CPUs.

By Catalin Cimpanu for Zero Day | November 10, 2020 -- 18:00 GMT (18:00 GMT) | Topic: Security

A team of academics has disclosed today a new attack method that can extract data from Intel CPUs.

Named Platypus, an acronym for "Power Leakage Attacks: Targeting Your Protected User Secrets," the attack targets the RAPL interface of Intel processors.

RAPL, which stands for *Running Average Power Limit*, is a component that allows firmware or software applications to monitor power consumption in the CPU and DRAM.

RAPL, which effectively lets firmware and software apps read how much electrical power a CPU is pulling in to perform its tasks, is a system that has been used for years to track and debug application and hardware performance.

RESEARCHER STEAL ENCRYPTION KEYS VIA INTEL RAPL

golem.de

IT-NEWS FÜR PROFIS

HOME TICKER VIDEOS VORGELESEN FORUM | ANMELDEN

golem.de jetzt werbefrei lesen

TOP-THEMEN: Arbeit Auto Künstliche mehr...

IT-KARRIERE: STELLENMARKT AKADEMIE SHOP IT KÖPFE GEHALTSCHECK | SERVICES: GOLEM PC PREISVERGLEICH TOP-ANGEBOTE

SICHERHEITSLÜCKE PLATYPUS

Passwörter per Leistungsmessung aus der CPU ausleiten

Wieder gelingt es einem Forscherteam, eigentlich geschützte Daten aus Intel-CPUs abzuleiten. Diesmal per Software-Leistungsmessung.

11. November 2020, 13:01 Uhr, Sebastian Grüner

Der Platypus-Angriff ist noch Schwachstellen bekannt.

Eine Forschergruppe der Unis Graz und Birmingham sowie des Cispa-Helmholtz-Zentrums hat eine alte Angriffsmethode auf CPUs neu aufgelegt und den Angriff Platypus (Power Leakage Attacks Targeting Your Protected User Secrets) getauft. Das Team nutzt dazu minimale Unterschiede in der Leistungsaufnahme von CPUs, um geheimes Schlüsselmaterial aus der CPU auszuleiten.

Media Coverage

heise online

IT Mobiles Entertainment Wissen Netzpolitik Wirtschaft Journal

"Platypus": Sicherheitslücke m… Messfunktion von Intel-Prozes…

Sicherheitsforscher nutzen die zur Leistungsmessung … Schnittstelle aktueller Prozessoren, um geheime Date… verarbeitet.

(Bild: Natascha Eibl/CC0 1.0)

ZDNet

New Platypus attack can steal data from Intel CPUs

Intel has released microcode updates today to prevent attackers from abusing the Intel RAPL mechanism to steal sensitive data from its CPUs.

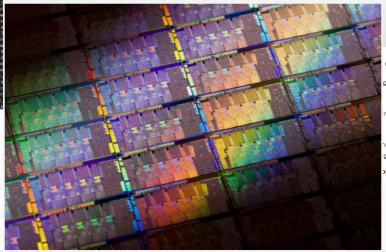
ars TECHNICA

PLATYPUS —

Intel SGX defeated yet again—this time thanks to on-chip power meter

New research sends chipmaker scrambling to fix side channel that exposes secret data.

DAN GOODIN - 11/10/2020, 7:00 PM



…can extract data from Intel CPUs.

…g Your Protected User Secrets," the attack targets …

…rt that allows firmware or software applications to …

…uch electrical power a CPU is pulling in to perform …g application and hardware performance.

golem.de

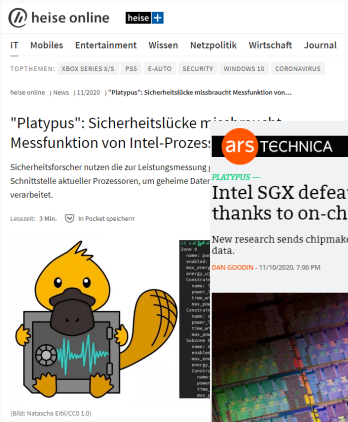
Passwörter per Leistungsmessung aus der CPU ausleiten

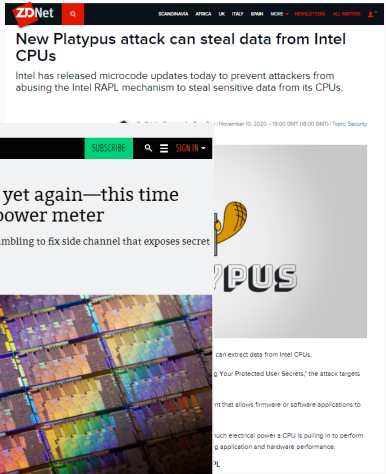
Wieder gelingt es einem Forscherteam, eigentlich geschützte Daten aus Intel-CPUs abzuleiten. Diesmal per Software-Leistungsmessung.

11. November 2020, 13:01 Uhr, Sebastian Grüner

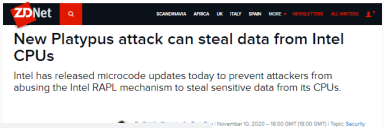
Eine Forschergruppe der Unis Graz und Birmingham sowie des Cispa-Helmholtz-Zentrums hat eine alte Angriffsmethode auf CPUs neu aufgelegt und den Angriff Platypus (Power Leakage Attacks Targeting Your Protected User Secrets) getauft. Das Team nutzt dazu minimale Unterschiede in der Leistungsaufnahme von CPUs, um geheimes Schlüsselmaterial aus der CPU auszuleiten.

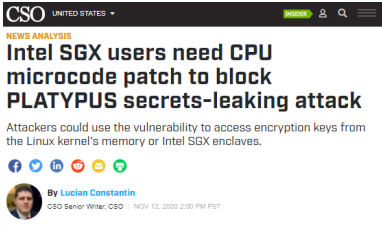
Media Coverage

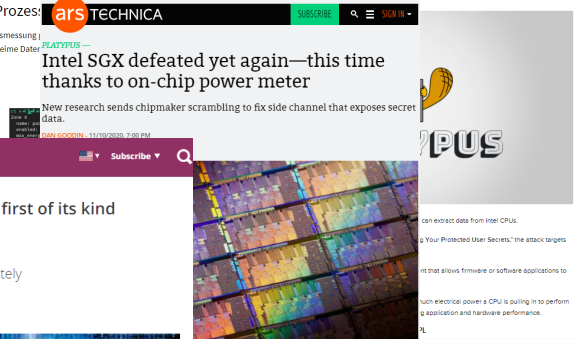




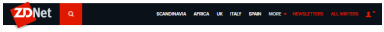
Media Coverage

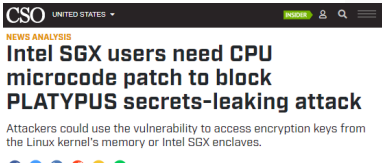


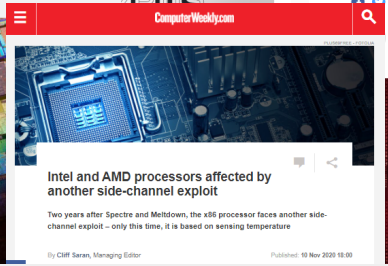




Media Coverage







Moritz Lipp, Michael Schwarz, Daniel Gruss, Andreas Kogler

Countermeasures

Countermeasures

- Remove the unprivileged access to the RAPL MSRs

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**

Countermeasures

- Threat model of SGX allows a **compromised operating system**
  - Operating system patch does not help

Countermeasures

- Threat model of SGX allows a **compromised operating system**
  - Operating system patch does not help
- **Microcode updates** are necessary

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

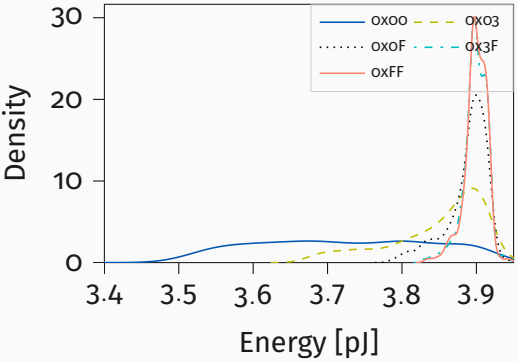
Countermeasures

- 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

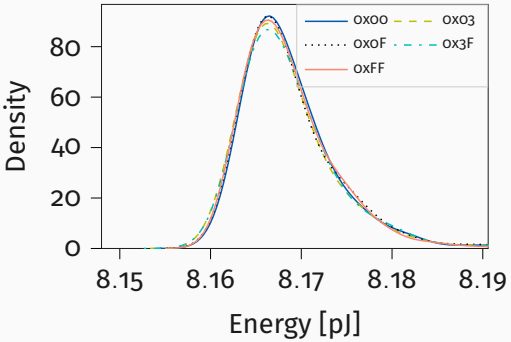
Countermeasures

- 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

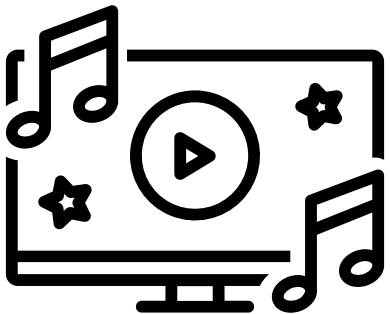
Mitigation



Without Mitigation



With Mitigation



Lady Ada - Powertrace

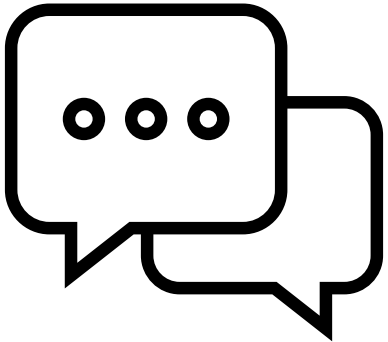
PLATYPUS: Software-based Power Side-Channel Attacks on x86
Moritz Lipp, Andreas Kogler, David Oswald, Michael Schwarz, Catherine Easdon, Claudio Canella, Daniel Gruss

Takeaways

- Power side-channel attacks can be exploited **from software** on modern CPUs

Takeaways

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



Questions & Discussion

References i