

Trust for Our Time: Confidential Computing in Untrusted Environments

Jo Van Bulck

🏠 DistriNet, KU Leuven, Belgium ✉️ jo.vanbulck@cs.kuleuven.be 🐦 [@jovanbulck](https://twitter.com/jovanbulck) 🌐 vanbulck.net

Inaugural Lecture, February 14, 2025



The Big Picture: Protecting Private Data



Data in transit



Data in use



Data at rest

The Big Picture: Protecting Private Data



Data in transit

- ✓ HTTPS etc.



Data in use



Data at rest

- ✓ Full disk encryption

The Big Picture: Protecting Private Data



Data in transit

- ✓ HTTPS etc.



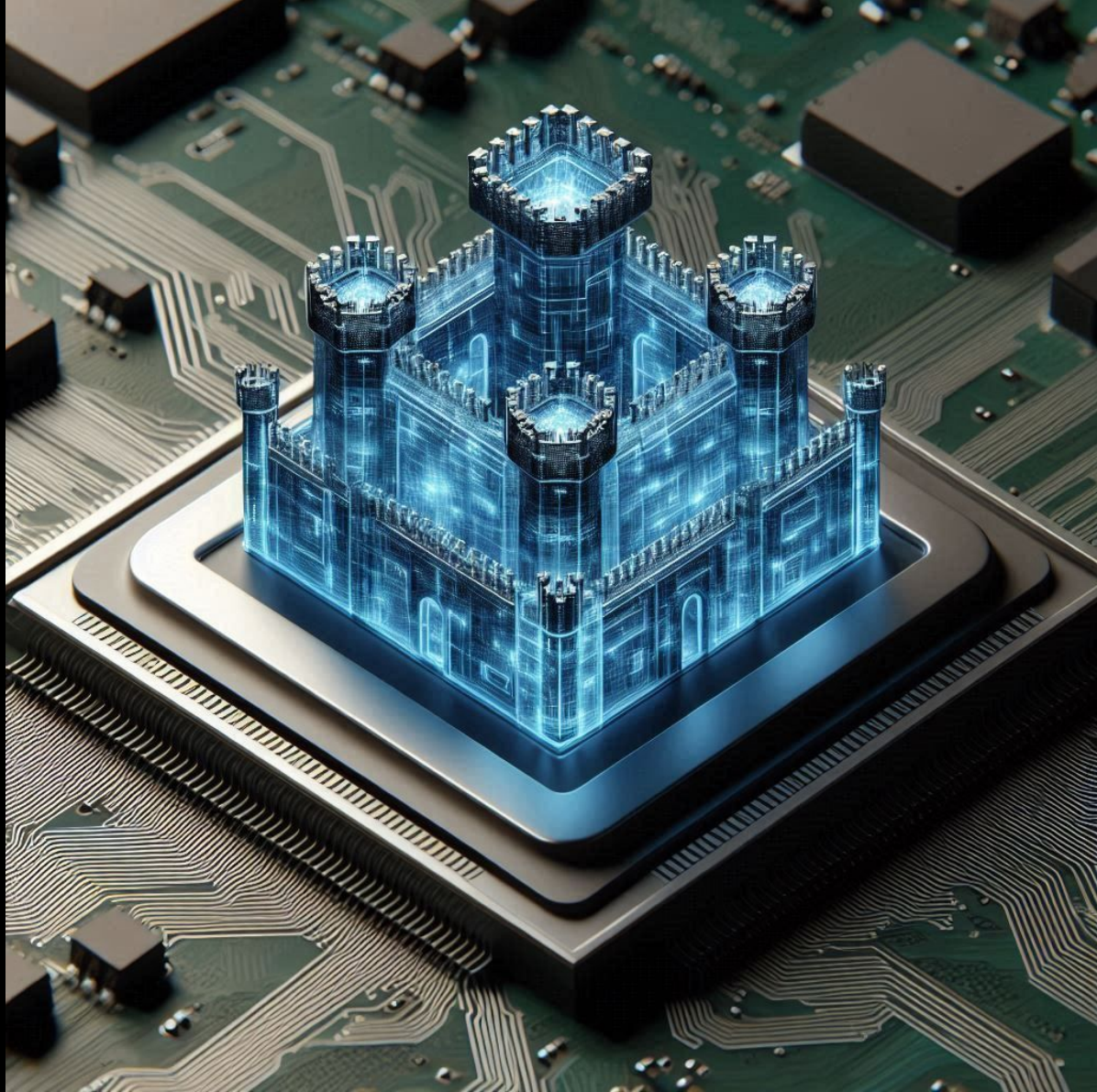
Data in use

- ? Homomorphic encryption?
- ? Trusted Execution?
= Confidential Computing
= Hardware Enclaves

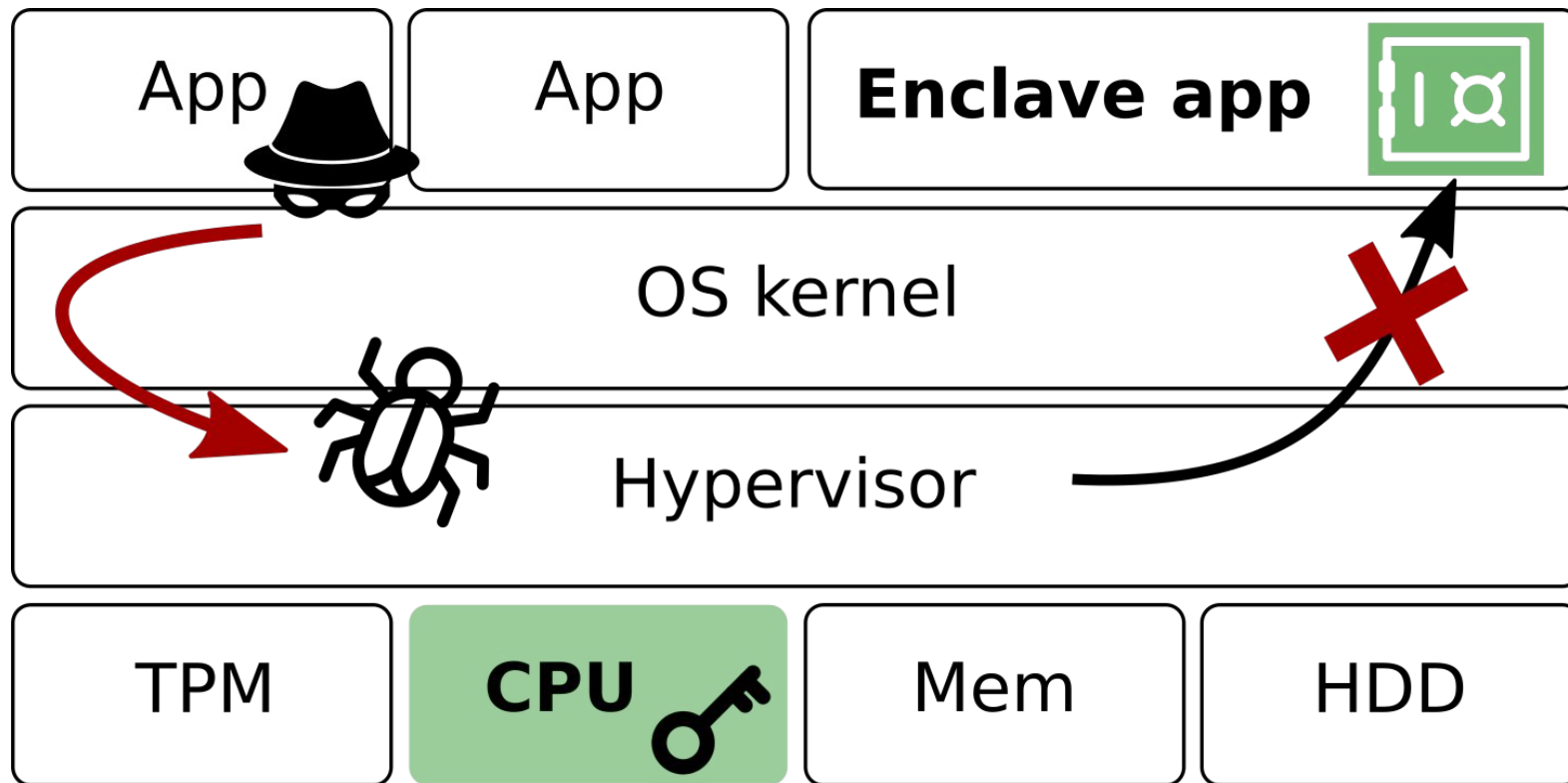


Data at rest

- ✓ Full disk encryption

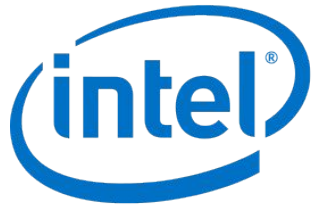


Confidential Computing: Reducing Attack Surface



Trusted execution: Hardware-level **isolation and attestation**

The Rise of Trusted Execution Environments (TEEs)

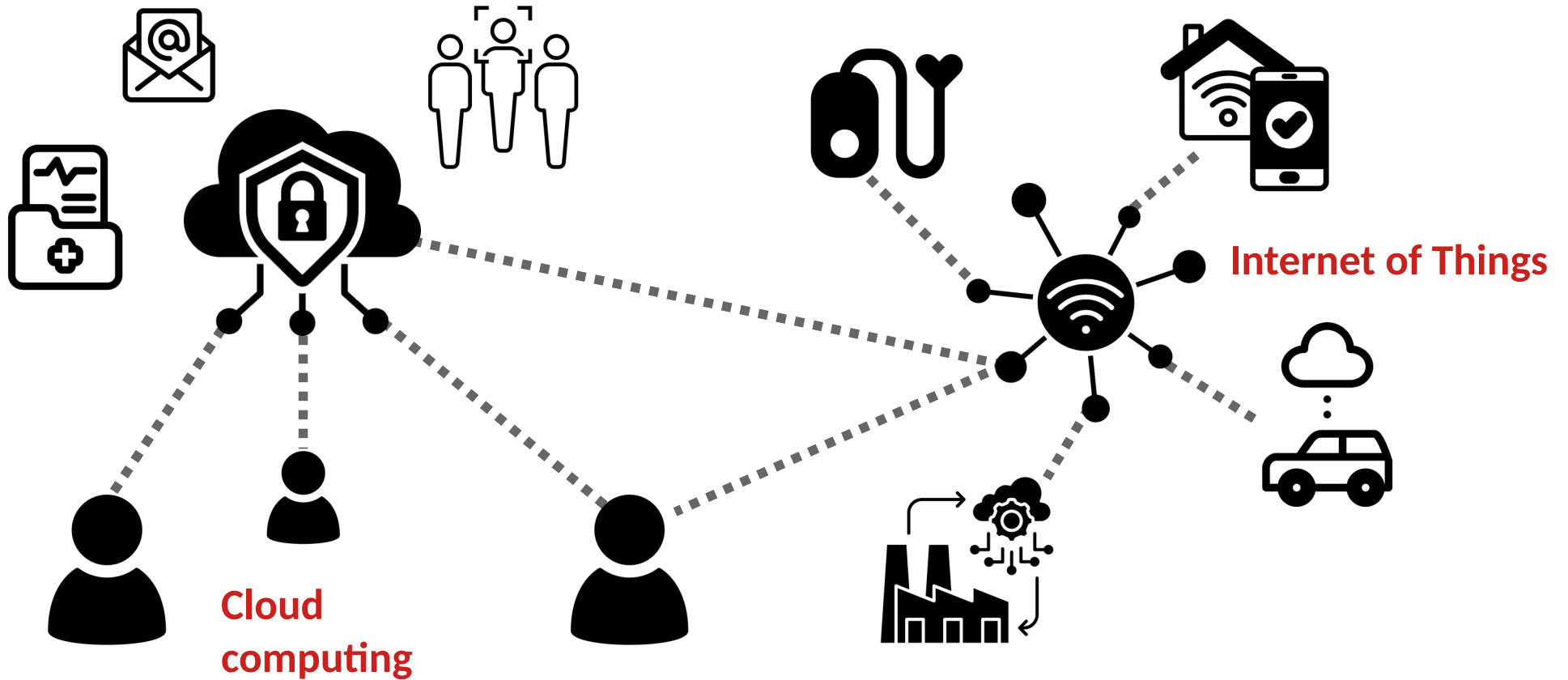


- 2004: ARM TrustZone
- 2015: **Intel Software Guard Extensions (SGX)**
- 2016: AMD Secure Encrypted Virtualization (SEV)
- 2018: IBM Protected Execution Facility (PEF)
- 2020: AMD SEV with Secure Nested Paging (SEV-SNP)
- 2022: Intel Trust Domain Extensions (TDX)
- 2023: ARM Confidential Compute Architecture (CCA)
- 2024: NVIDIA Confidential Computing



TEEs are here to stay...

“Confidential Computing Today, Just Computing Tomorrow” *



Trust for Our Time?



Tim Cook attacks Google over privacy of Photos service

Tim Cook continues to throw barely veiled barbs at Google, in an effort to position Apple as the privacy champion of Silicon Valley.

By Bogdan Petrovan · June 3, 2015 · 0 ·

The Comprehensive Guide to Quitting Google

5 Google Photos privacy problems you should know about

May 24, 2023 · Matt Mills · Tips and Tricks · 0

Google photos privacy issues? Everything you Need to Know

rozepz · May 28, 2018 · Privacy services tutorial

number 8, 2018

Google Photos and Privacy: How Safe Are Your Photos?

PC Software · May 6, 2024

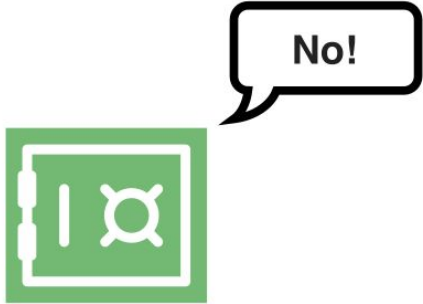




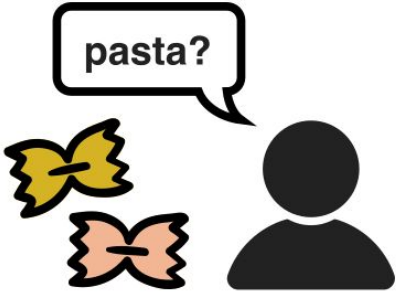
Case Study #1: Secure Login?

Case Study: Comparing a Secret Password

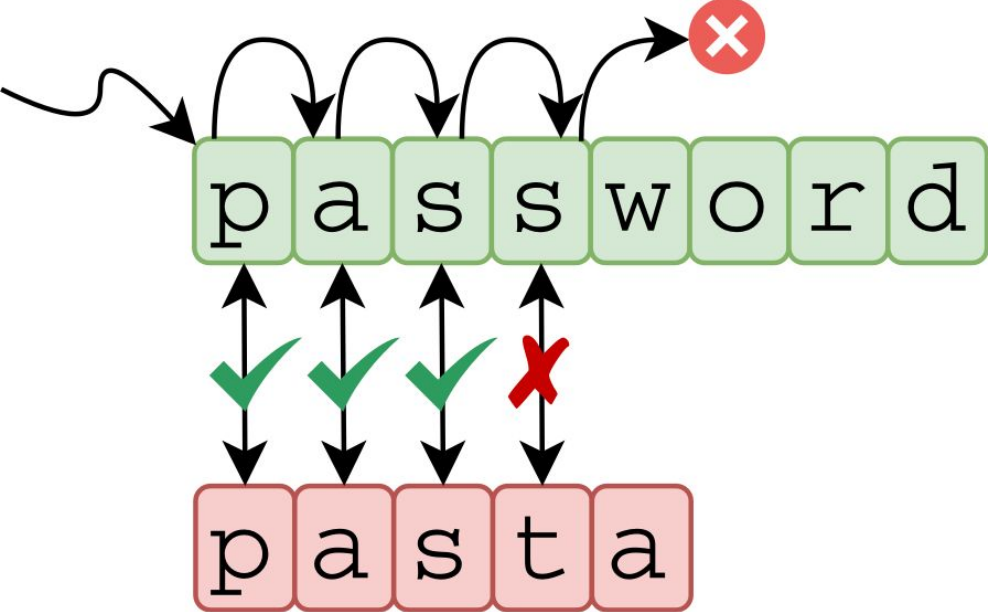
p a s s w o r d



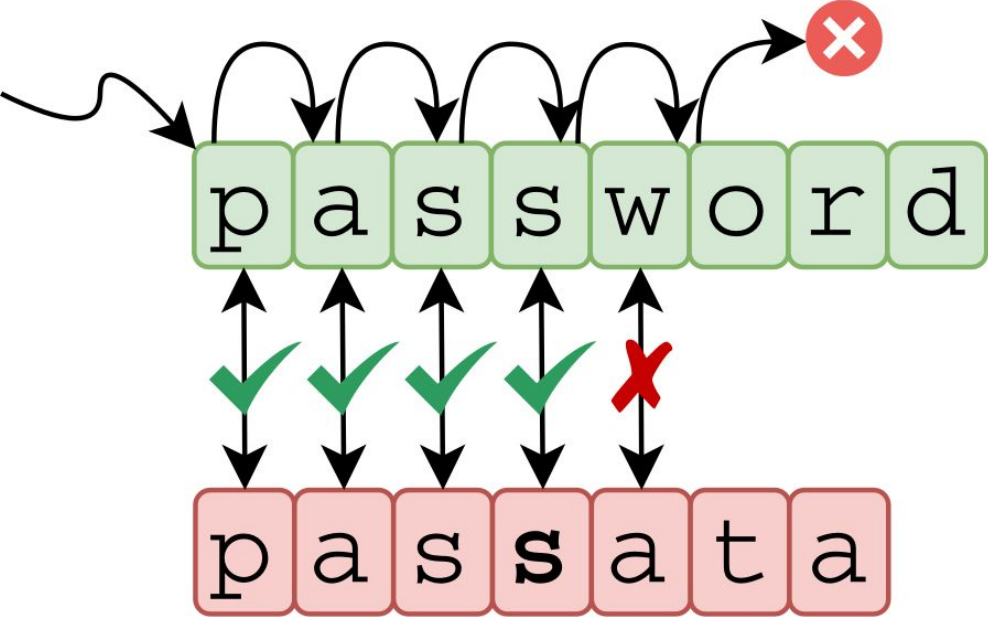
p a s t a



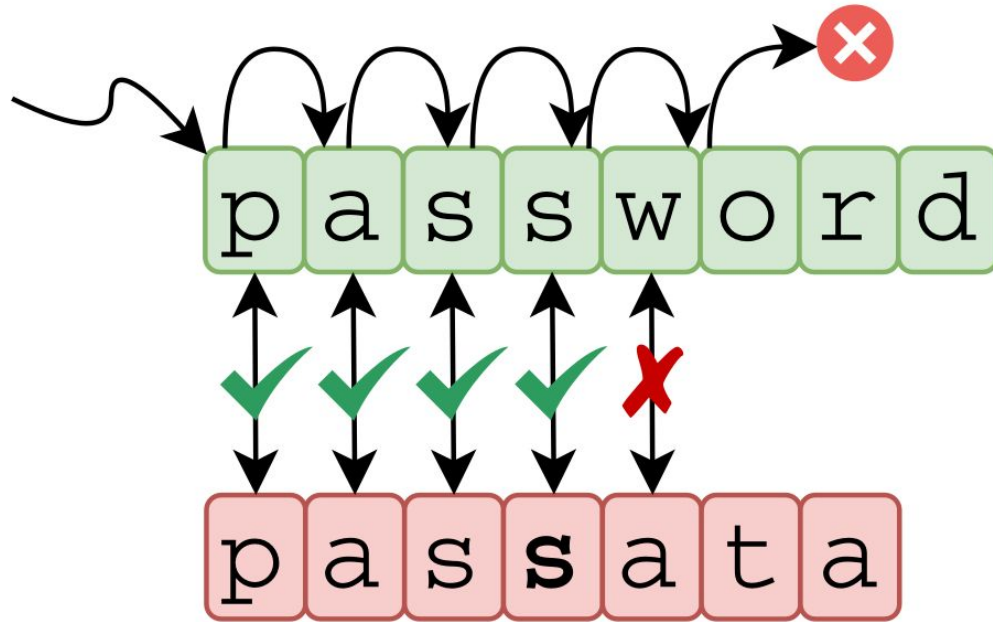
Case Study: Comparing a Secret Password



Case Study: Comparing a Secret Password



Case Study: Comparing a Secret Password

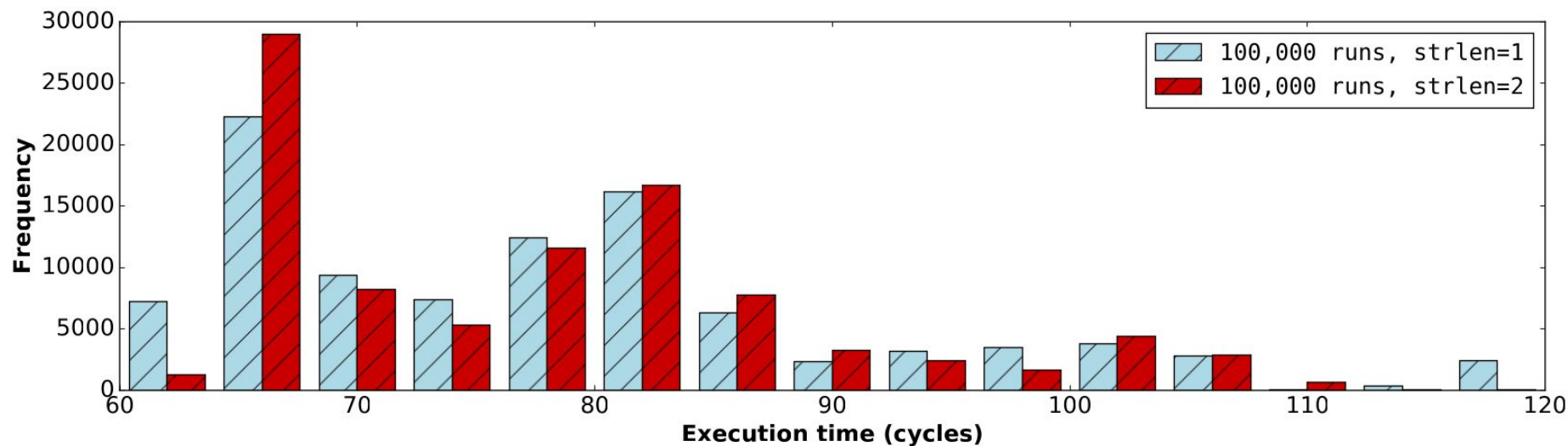


Overall **execution time** reveals correctness of individual password bytes!

Building the Side-Channel Oracle with Execution Timing?



Too noisy: modern x86 processors are lightning fast. . .



Challenge: Side-Channel Sampling Rate



Slow
shutter speed

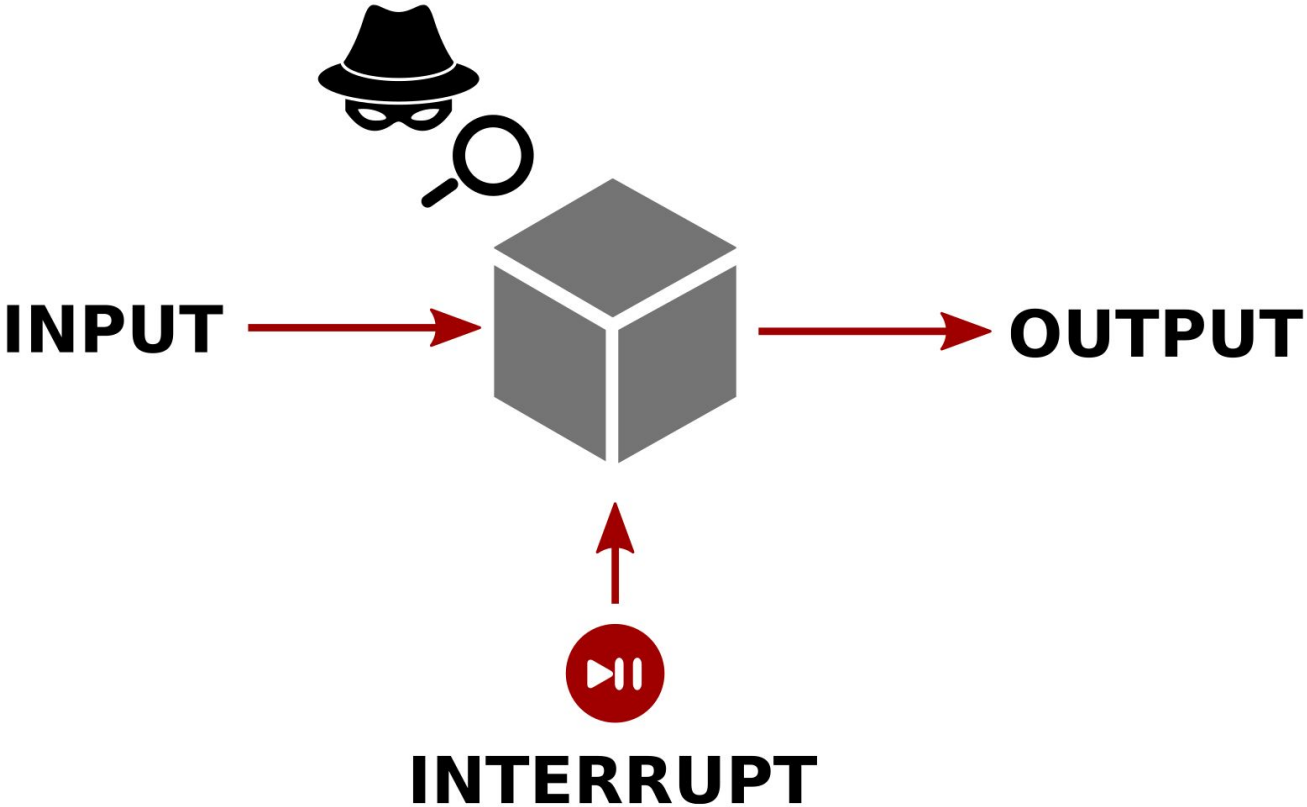


Medium
shutter speed



Fast
shutter speed

SGX-Step: Executing Enclaves one Instruction at a Time



SGX-Step: Executing Enclaves one Instruction at a Time



SGX-Step



**ACSAC 2023
Cybersecurity Artifacts
Impact Award**

 <https://github.com/jovanbulck/sgx-step>

 Watch

22

 Star

245

 Fork

52

CHAPTER 8

ASYNCHRONOUS ENCLAVE EXIT NOTIFY AND THE EDECCSSA USER LEAF FUNCTION

8.1 INTRODUCTION

Asynchronous Enclave Exit Notify (AEX-Notify) is an extension to Intel[®] SGX that allows Intel SGX enclaves to be notified after an asynchronous enclave exit (AEX) has occurred. EDECCSSA is a new Intel SGX user leaf function (ENCLU[EDECCSSA]) that can facilitate AEX notification handling, as well as software exception handling. This chapter provides information about changes to the Intel SGX architecture that support AEX-Notify and ENCLU[EDECCSSA].

The following list summarizes the a details are provided in Section 8.3)

- SECS.ATTRIBUTES.AEXNOTIFY:
- TCS.FLAGS.AEXNOTIFY: This e
- SSA.GPRSGX.AEXNOTIFY: Enclave-writable byte that allows enclave software to dynamically enable/disable AEX notifications.

An AEX notification is delivered by ENCLU[ERESUME] when the following conditions are met:



*SGX-Step led to **new x86 processor instructions!***

→ shipped in millions of devices ≥ 4th Gen Xeon CPU

Intel AEX Notify Support Prepped For Linux To Help Enhance SGX Enclave Security

Written by [Michael Larabel](#) in [Intel](#) on 6 November 2022 at 06:01 AM EST. [5 Comments](#)



Future Intel CPUs and some existing processors via a microcode update will support a new feature called the Asynchronous EXit (AEX) notification mechanism to help with Software Guard Extensions (SGX) enclave security. Patches for the Linux kernel are pending for implementing this Intel AEX Notify support with capable processors.

Intel's Asynchronous EXit (AEX) notification mechanism lets SGX enclaves run a handler after an AEX event. Those handlers can be used for things like mitigating SGX-Step as an attack framework for precise enclave execution control.



Code 1

in intel/linux-sgx

Filter

intel sdk/trts/linux/trts_mitigation.S

```
48 * Description:
49 *   The file provides mitigations for SGX-Step
50 */
71 * Function:
   constant_time_apply_sgxstep_mitigation_and_continue_execution
72 *   Mitigate SGX-Step and return to the point at which the
   most recent
73 *   interrupt/exception occurred.
```

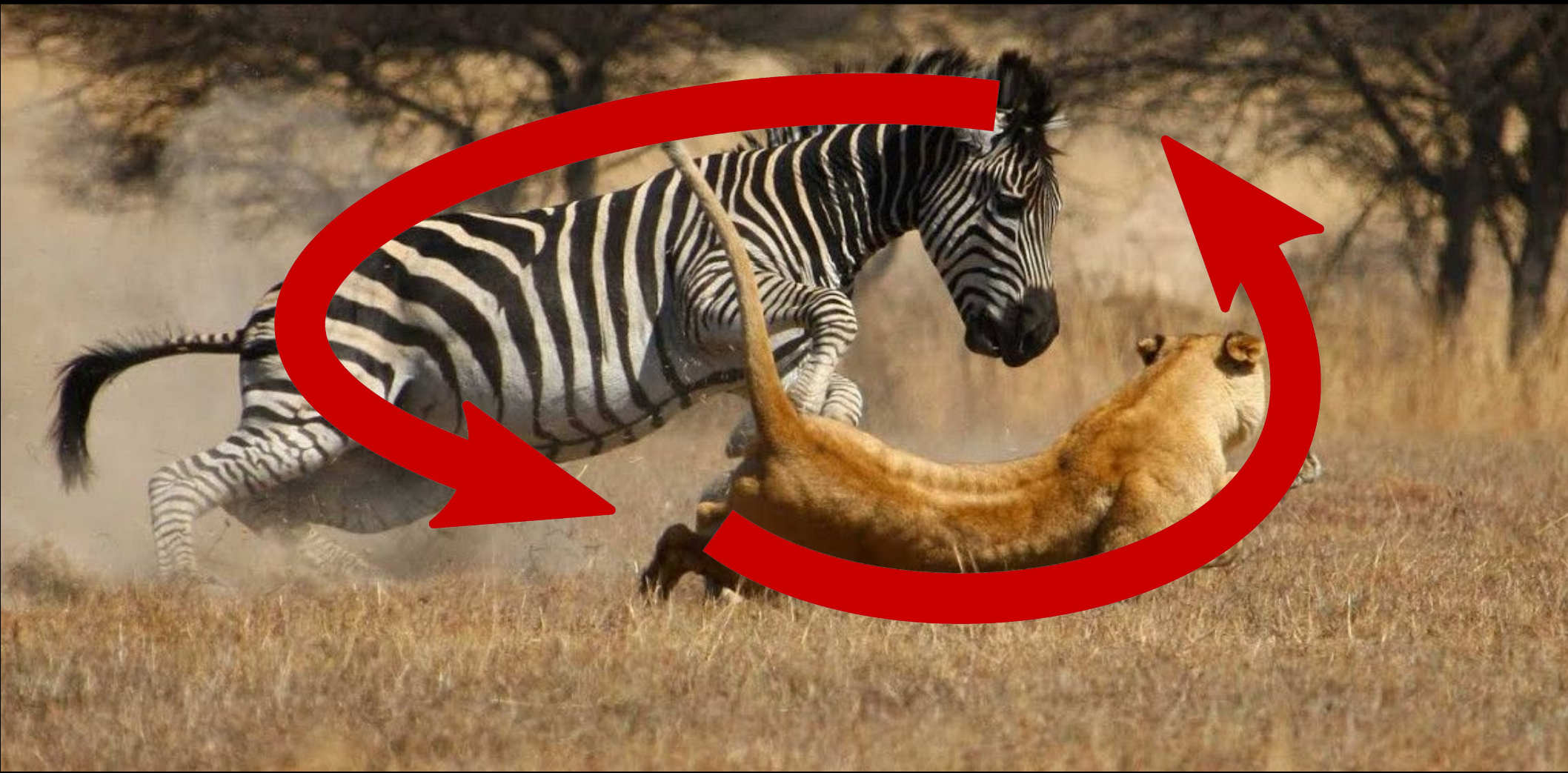


SGX-Step led to **changes in major OSs and enclave SDKs**

Scientific Understanding Driven by Attacker-Defender Race...



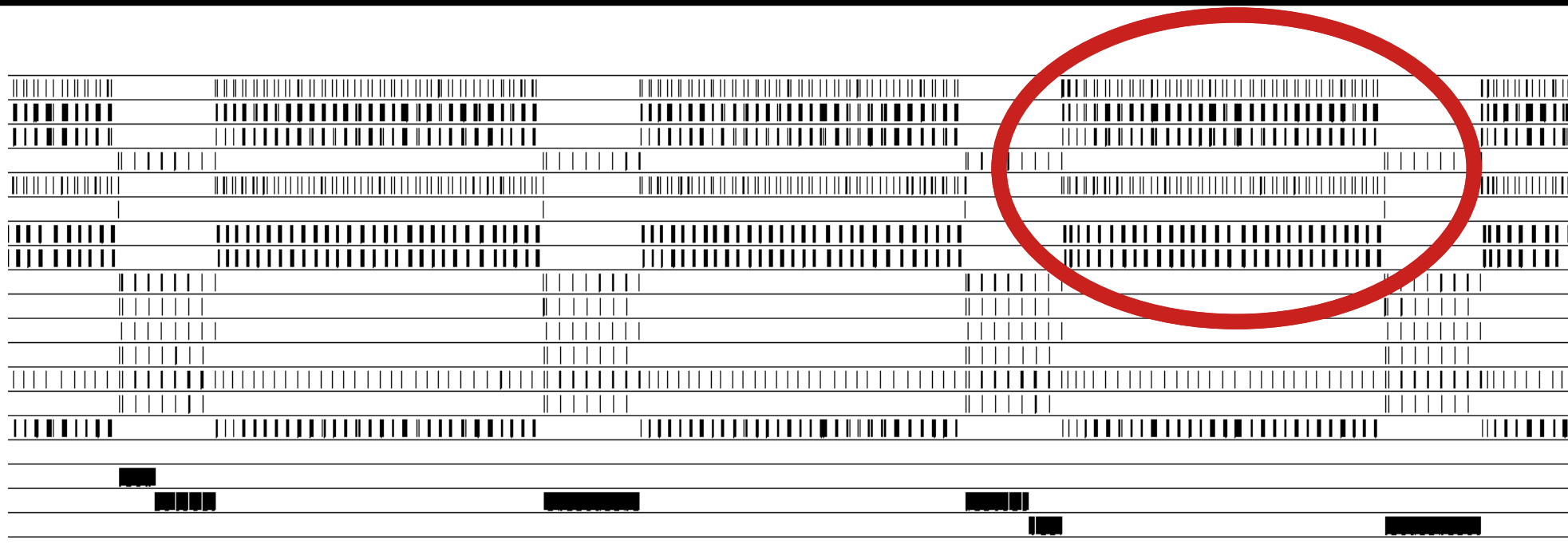
Scientific Understanding Driven by Attacker-Defender Race...





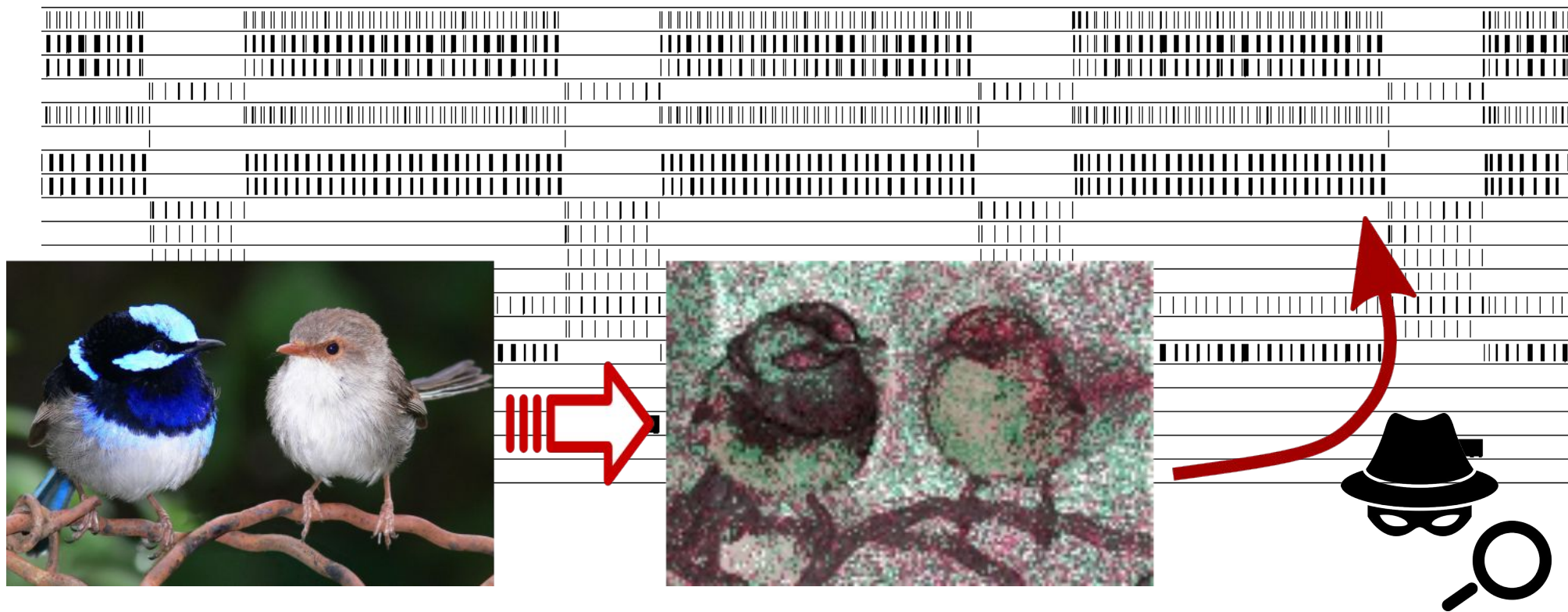
Case Study #2: Image Decoding?

Spatial Resolution: Page-Granular Memory Access Traces



Detailed trace of (coarse-grained) **code and data accesses over time...**

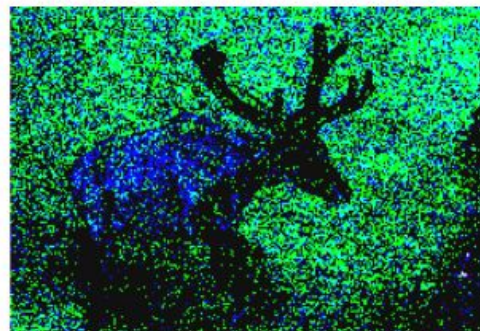
Spatial Resolution: Page-Granular Memory Access Traces



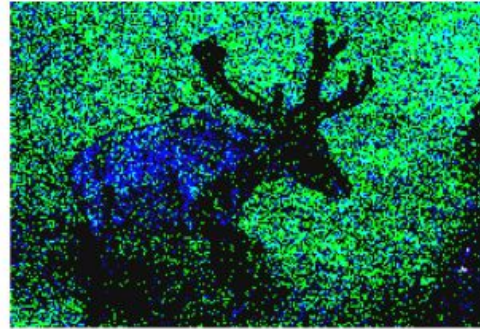
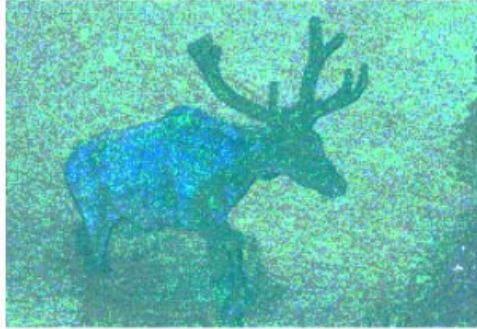
TLBlur: Practical, Compiler-Assisted Leakage Reduction



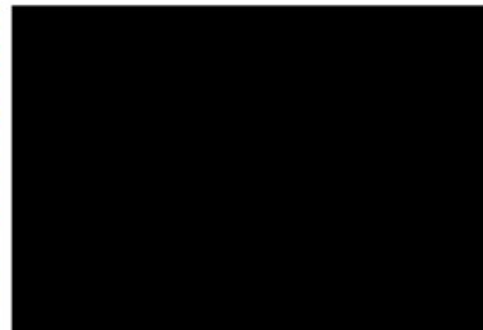
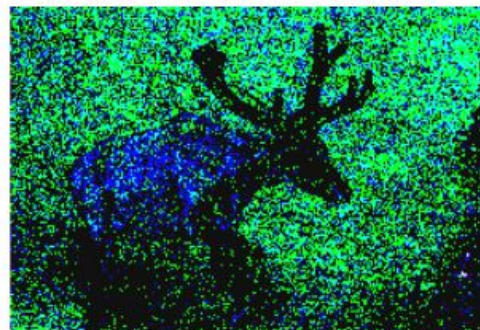
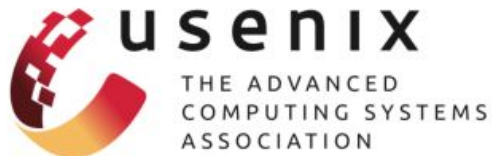
TLBlur: Practical, Compiler-Assisted Leakage Reduction



TLBlur: Practical, Compiler-Assisted Leakage Reduction



TLBlur: Practical, Compiler-Assisted Leakage Reduction



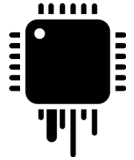
Automated “blurring” of page-access traces in space and time



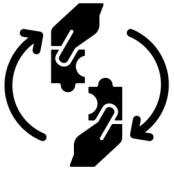
Conclusions and Take-Away



New era of **confidential computing** for the cloud and IoT



... but current architectures are **not perfect!**



Scientific understanding driven by **attacker-defender race**



Thank you!