Varys

Protecting SGX Enclaves From Practical Side-Channel Attacks

Oleksii Oleksenko, Bohdan Trach
Robert Krahn, Andre Martin, Christof Fetzer

TECHNISCHE UNIVERSITÄT DRESDEN

Mark Silberstein

TECHNION Israel Institute of Technology

Key issue of the cloud:

We cannot trust it

We cannot trust the cloud

- Thousands of employees
- Legal obligations
- Infrastructure vulnerabilities

GERALD SAUER OPINION 02.28.17 10:00 AM

A MURDER CASE TESTS ALEXA'S DEVOTION TO YOUR PRIVACY

Patch alert! Easy-to-exploit flaw in Linux kernel rated 'high risk'

Urgent security triage needed

Cloud Data Leak Exposes Information on 123 Million Americans

By: Sean Michael Kerner | December 20, 2017

We cannot trust the cloud

- Thousands of employees
- Legal obligations
- Infrastructure vulnerabilities

GERALD SAUER OPINION 02.28.17 10:00 AM

A MURDER CASE TESTS ALEXA'S DEVOTION TO YOUR PRIVACY

Patch alert! Easy-to-exploit flaw in Linux kernel rated 'high risk'

Urgent security triage needed

Cloud Data Leak Exposes Information on 123 Million Americans

By: Sean Michael Kerner | December 20, 2017

We cannot trust the cloud

- Thousands of employees
- Legal obligations
- Infrastructure vulnerabilities

GERALD SAUER OPINION 02.28.17 10:00 AM

A MURDER CASE TESTS ALEXA'S DEVOTION TO YOUR PRIVACY

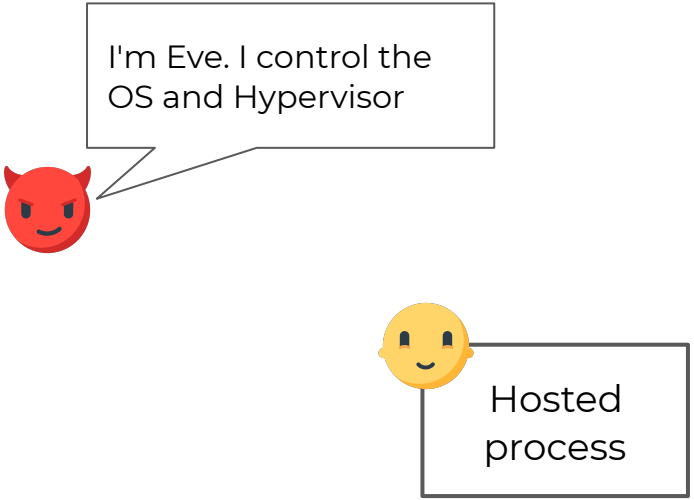
Patch alert! Easy-to-exploit flaw in Linux kernel rated 'high risk'

Urgent security triage needed

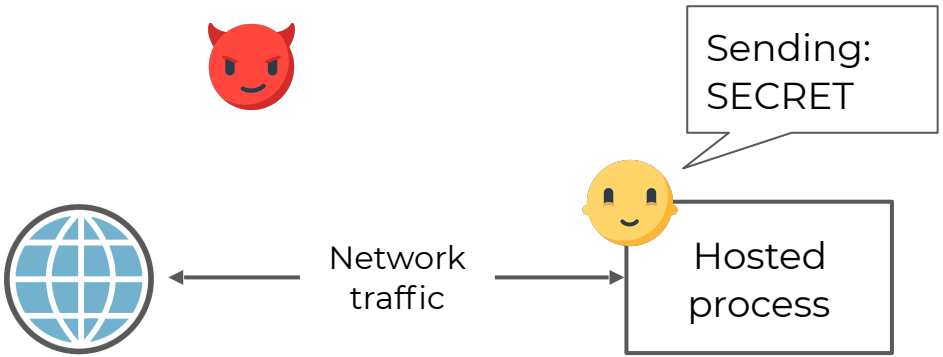
Cloud Data Leak Exposes Information on 123 Million Americans

By: Sean Michael Kerner | December 20, 2017

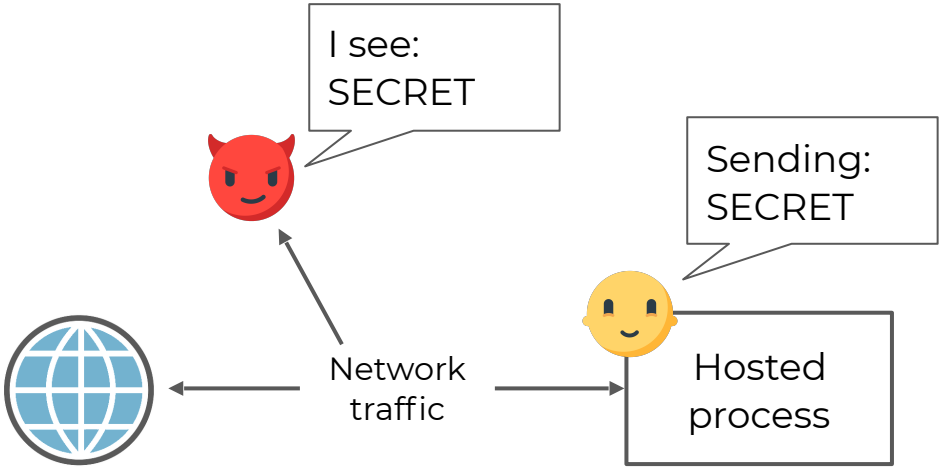
Privileged attack vectors



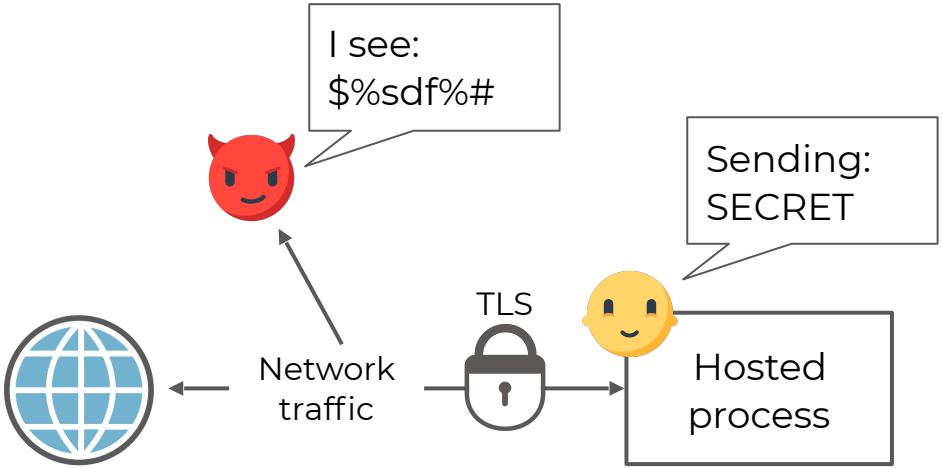
Privileged attack vectors: Network



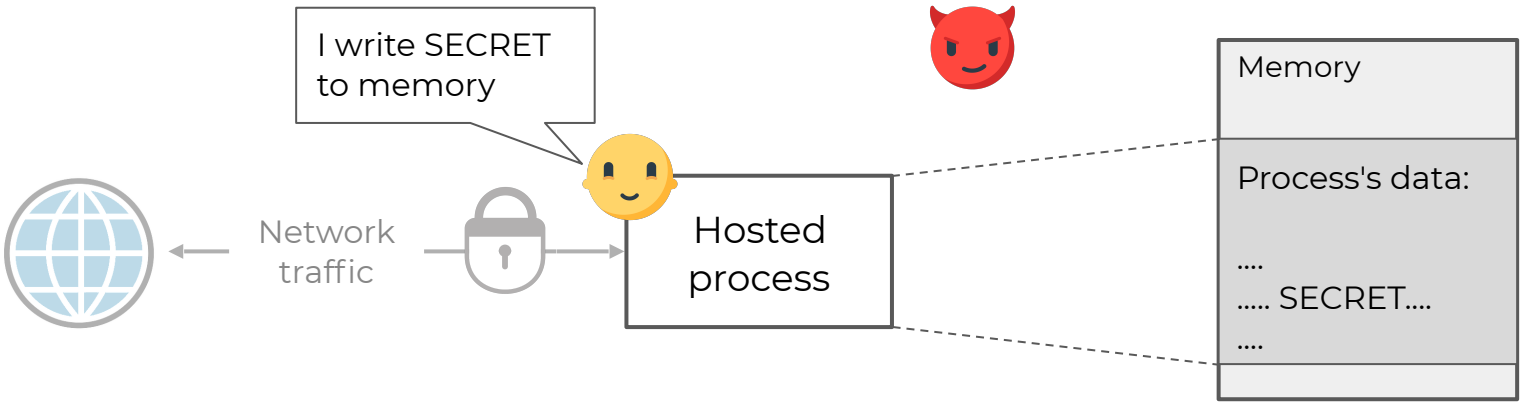
Privileged attack vectors: Network



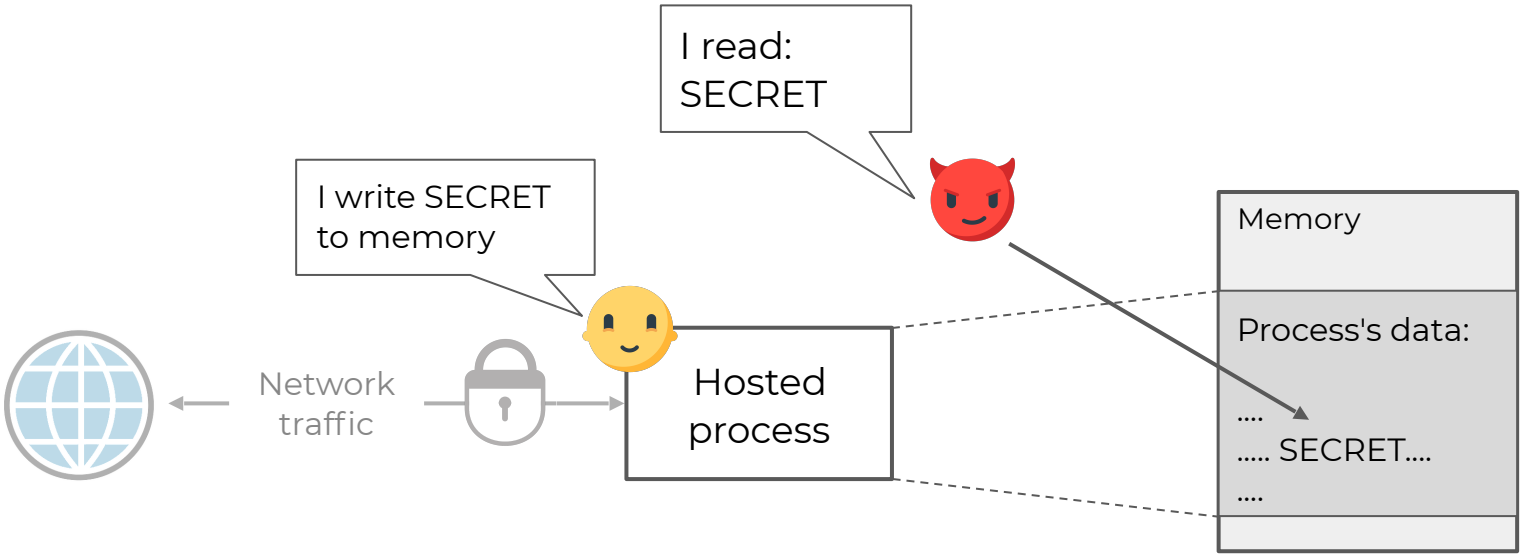
Privileged attack vectors: Network



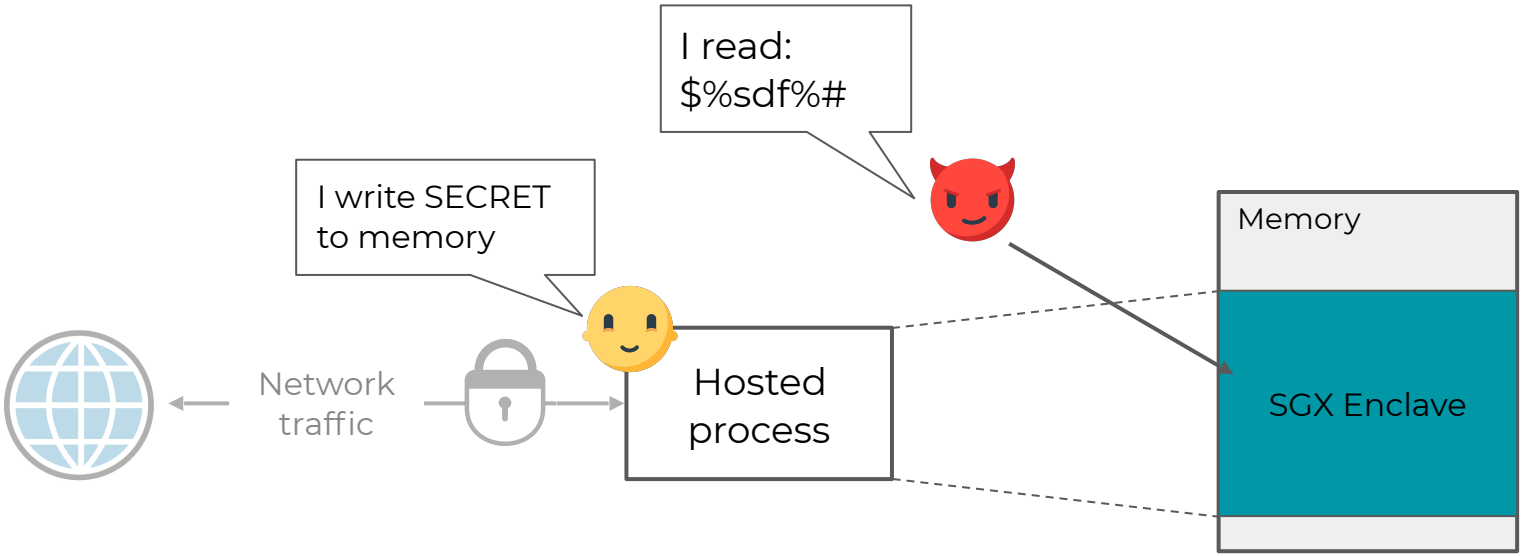
Privileged attack vectors: Memory



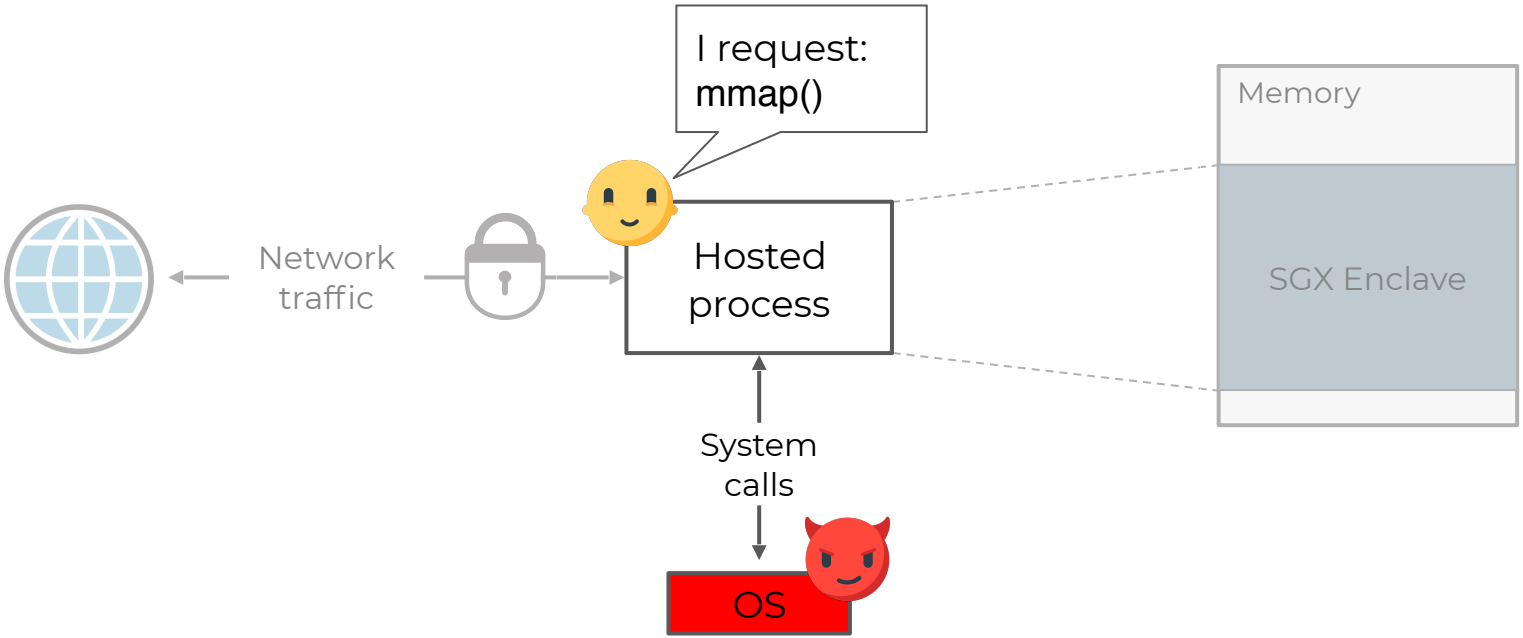
Privileged attack vectors: Memory



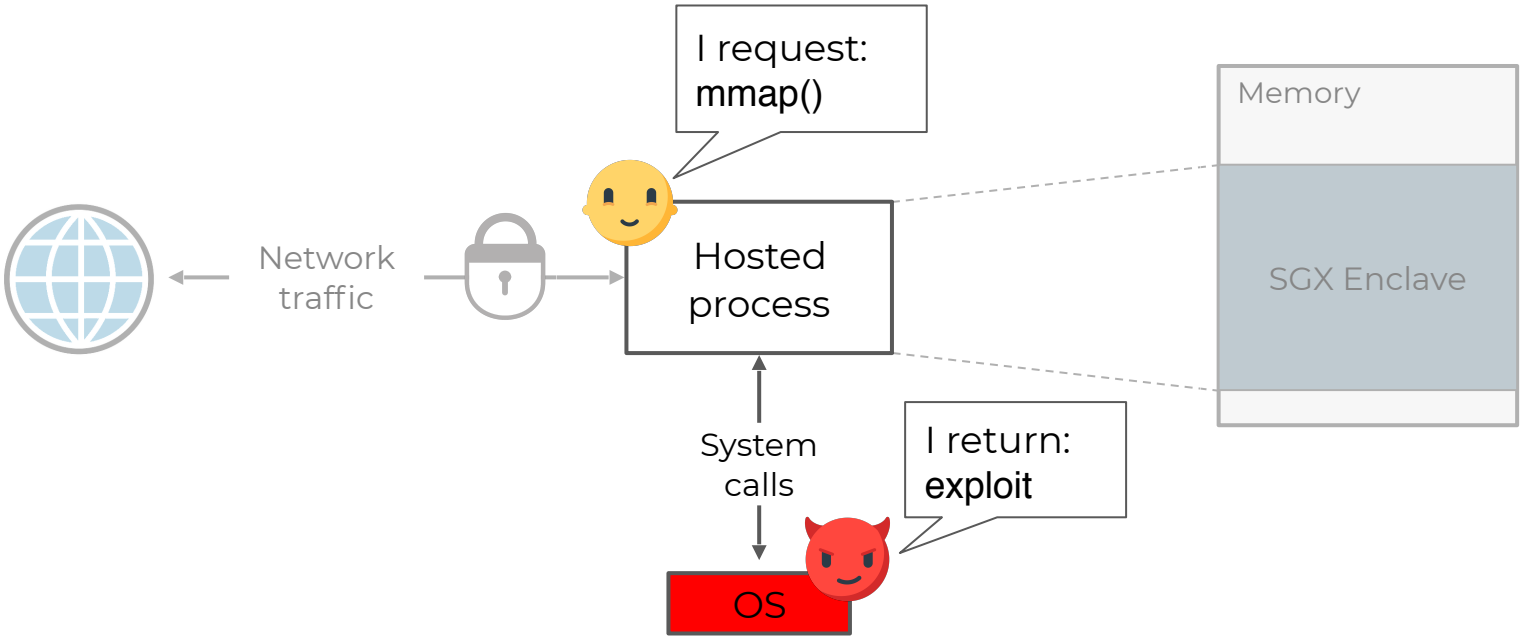
Privileged attack vectors: Memory



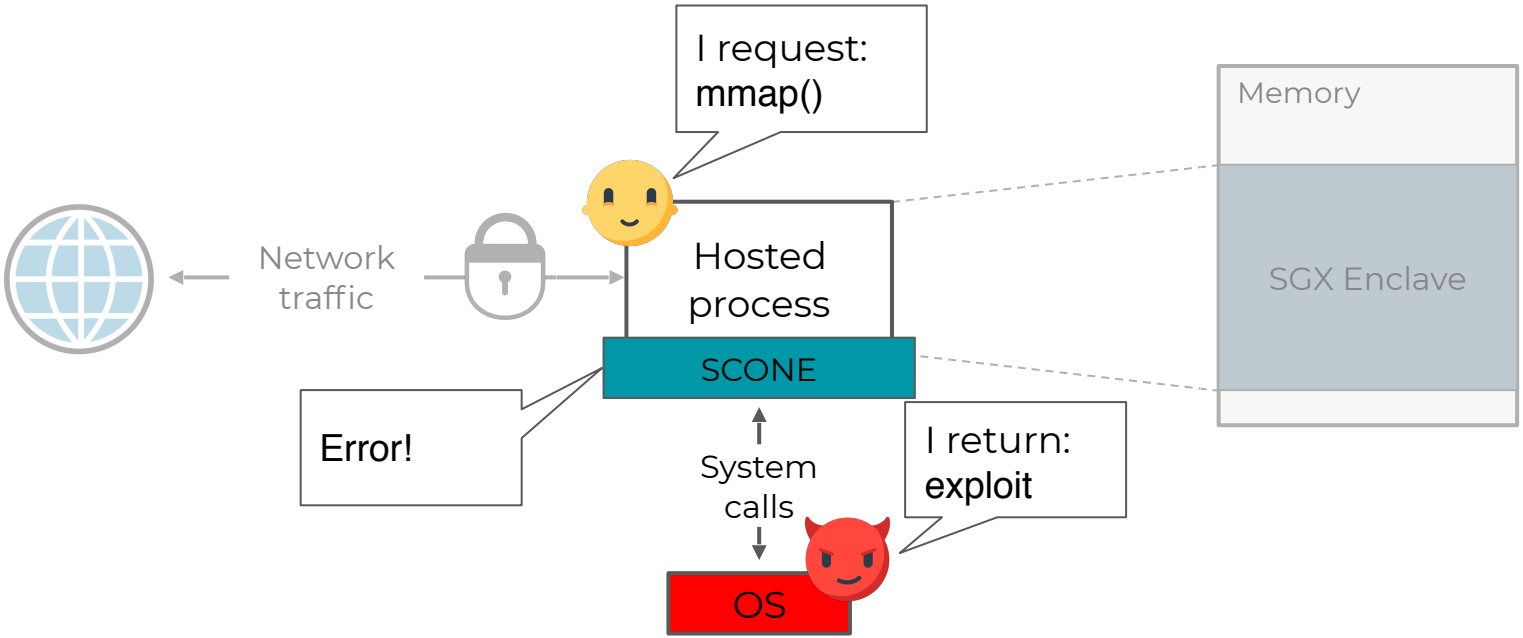
Privileged attack vectors: System Calls



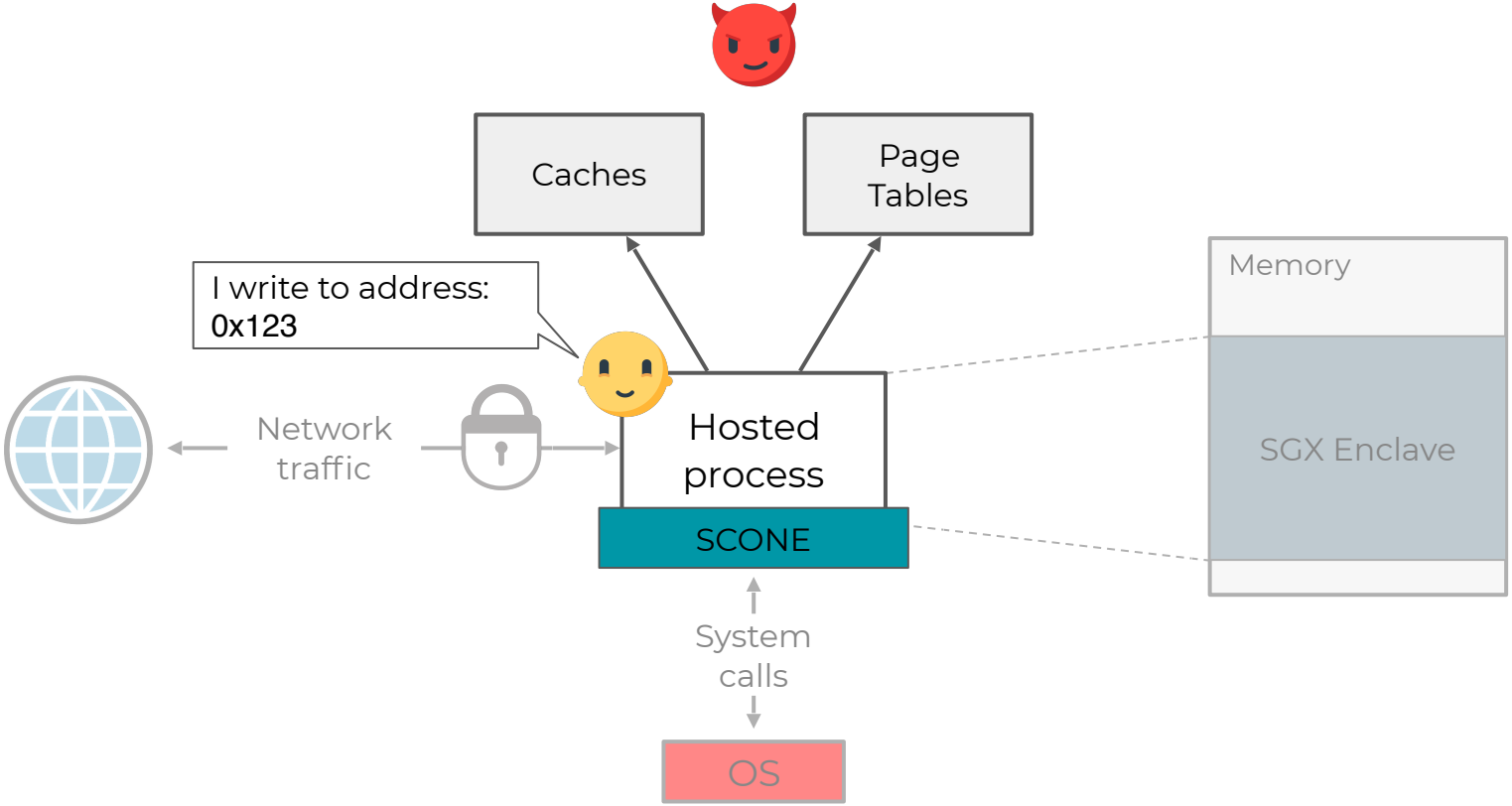
Privileged attack vectors: System Calls



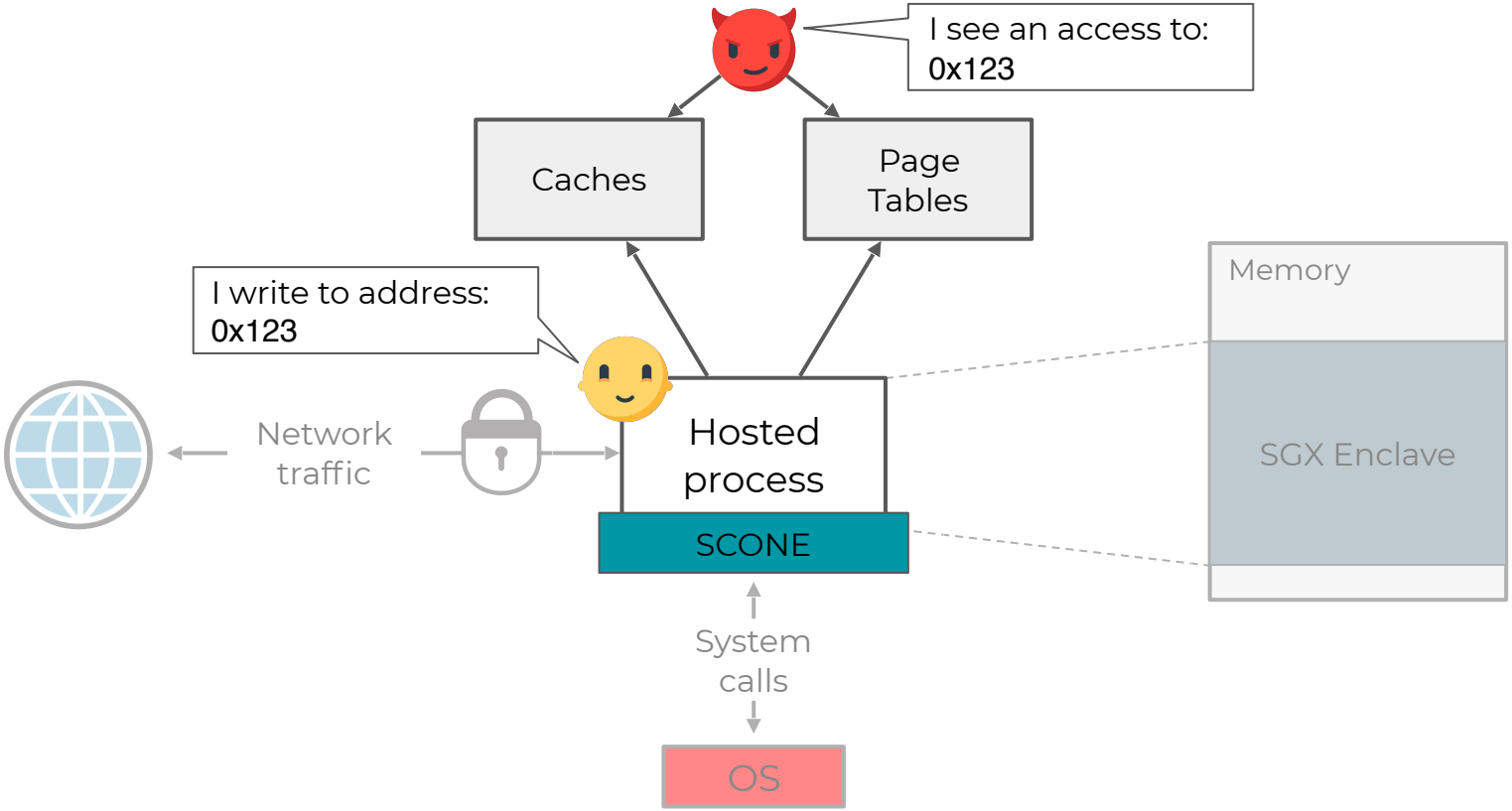
Privileged attack vectors: System Calls



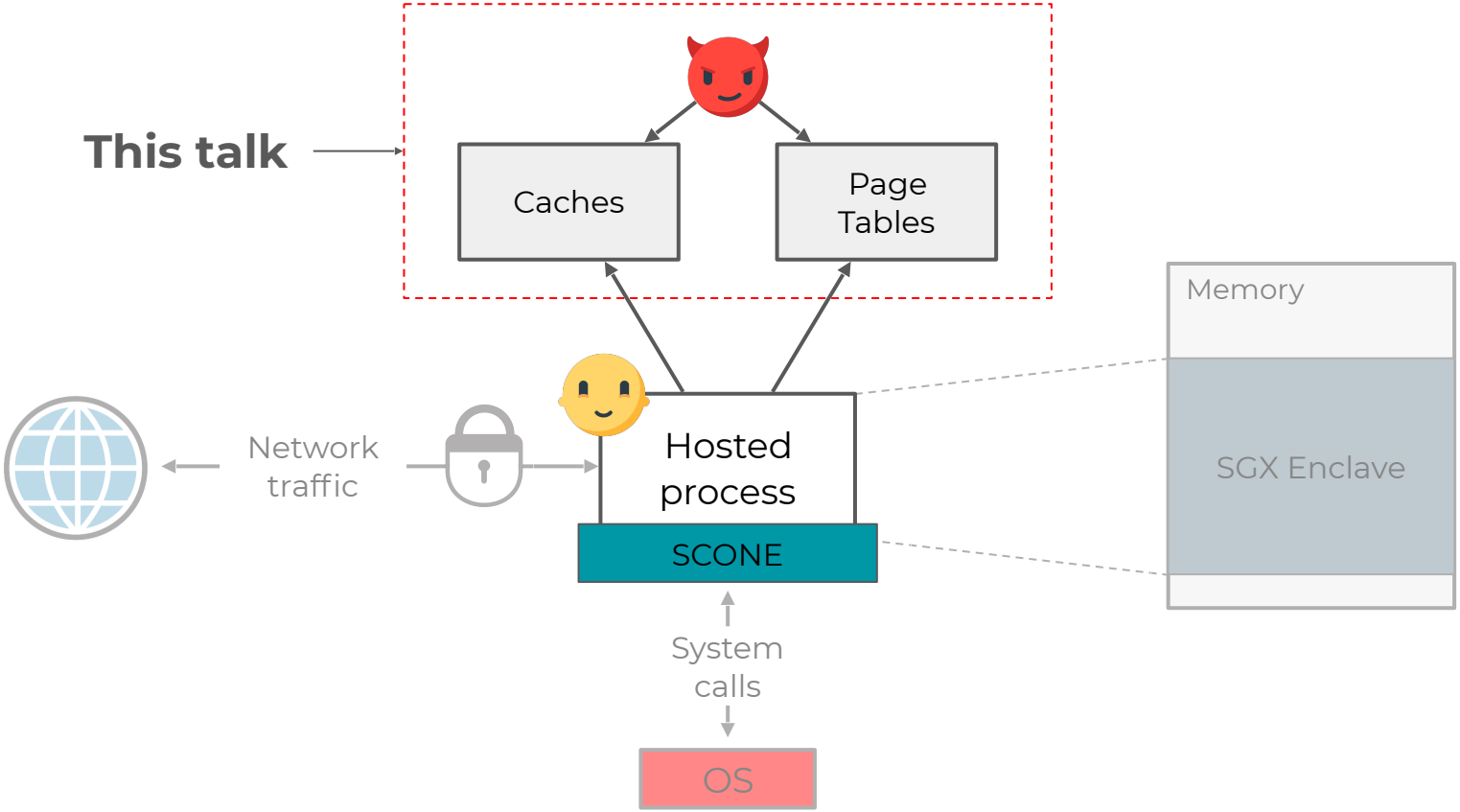
Privileged attack vectors: Shared Resources



Privileged attack vectors: Shared Resources

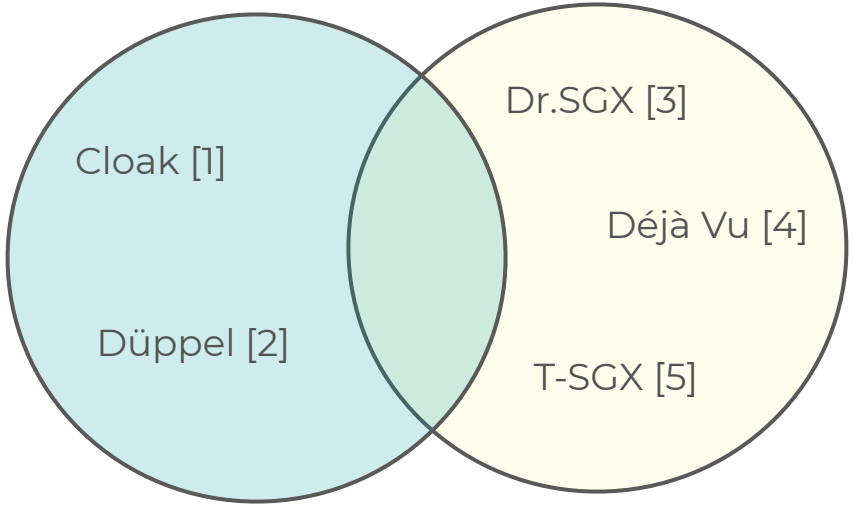


Privileged attack vectors: Shared Resources



Existing solutions

Low overhead



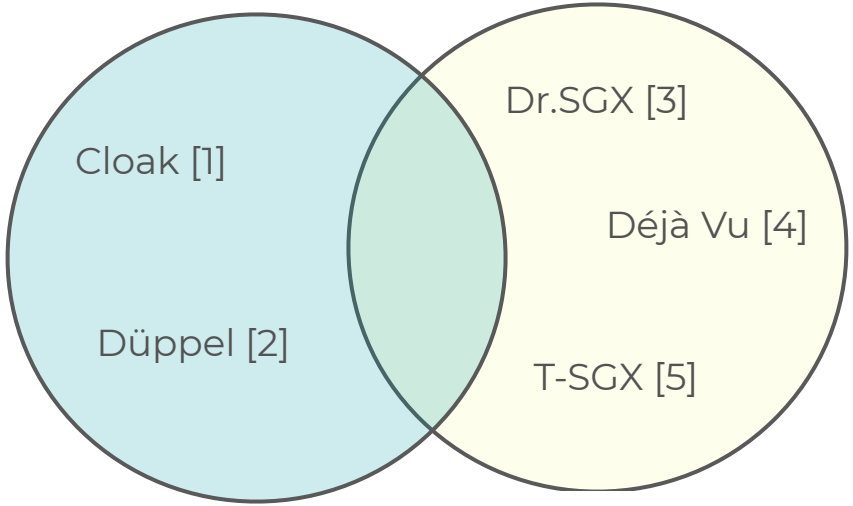
Low effort

(no code changes required)

[1] Gruss, D., Lettner, J., Schuster, F., Ohrimenko, O., Haller, I., & Costa, M. Strong and Efficient Cache Side-Channel Protection using Hardware Transactional Memory. In *Usenix Security 2017*.
[2] Zhang, Y., Reiter, M. K., Zhang, Y., & Reiter, M. K. Düppel: Retrofitting Commodity Operating Systems to Mitigate Cache Side Channels in the Cloud. *In CCS 2013*.
[3] Brasser, F., Capkun, S., Dmitrienko, A., Frassetto, T., Kostiainen, K., Müller, U., & Sadeghi, A.-R. DR.SGX: Hardening SGX Enclaves against Cache Attacks with Data Location Randomization. In arXiv 2017.
[4] Chen, S., Reiter, M. K., Zhang, X., & Zhang, Y. Detecting Privileged Side-Channel Attacks in Shielded Execution with Déjà Vu. In *ASIA CCS '17*.
[5] Shih, M., Lee, S., & Kim, T. T-SGX: Eradicating controlled-channel attacks against enclave programs. In *NDSS 2017*.

Existing solutions

Low overhead



Low effort

(no code changes required)

Varys

- 15% average slowdown
- No code changes

Approach

Rely but verify

Approach

Rely but verify

Request isolation from the untrusted OS

Approach

Rely but verify

Request isolation from the untrusted OS

Check within the enclave

Complete description

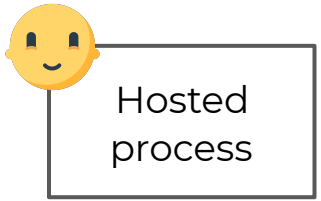
Varys implements a low-cost protection for Intel SGX enclaves against side-channel attacks by creating an isolated environment and verifying it at runtime.

Varys implements a low-cost protection for Intel SGX enclaves against side-channel attacks by creating an isolated environment and verifying it at runtime.

Rest of the talk explains this sentence

Varys implements a low-cost protection for Intel SGX enclaves against **side-channel attacks** by creating an isolated environment and verifying it at runtime.

Side-channel attacks



Side-channel attacks

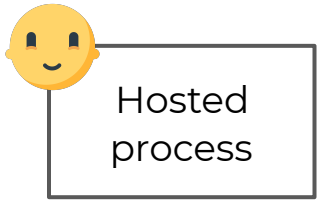
```
if (secret == 0)
    read(addr1)
else
    read(addr2)
```

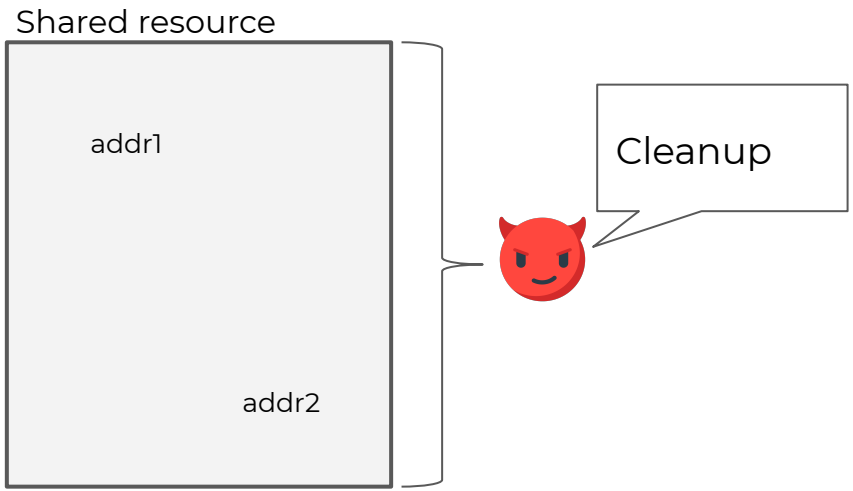
Side-channel attacks

```
if (secret == 0)
    read(addr1)
else
    read(addr2)
```

Side-channel attacks

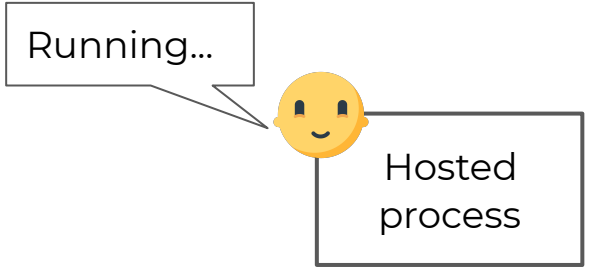
```
if (secret == 0)
    read(addr1)
else
    read(addr2)
```

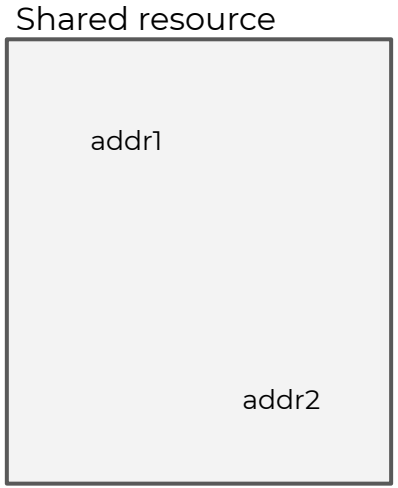




Side-channel attacks

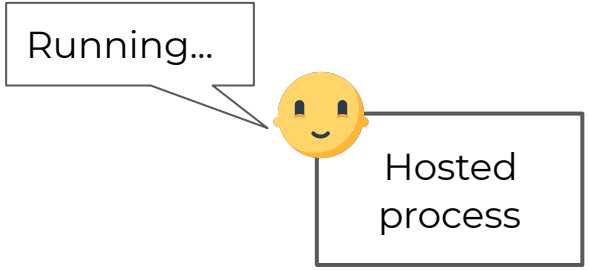
```
if (secret == 0)
    read(addr1)
else
    read(addr2)
```

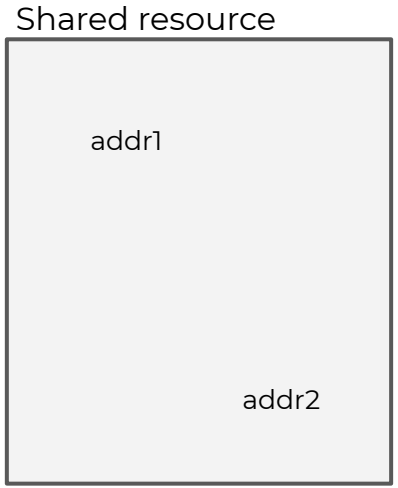




Side-channel attacks

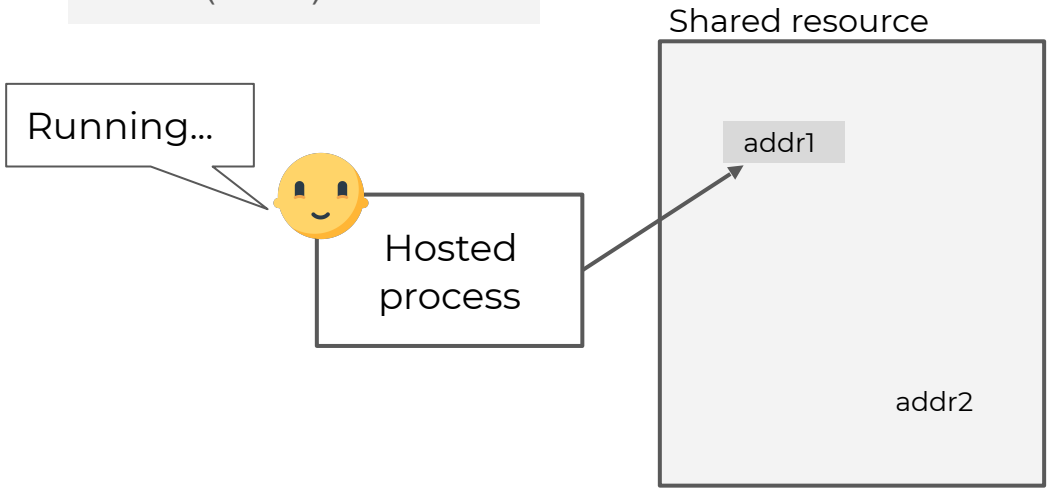
```
if (secret == 0)
    read(addr1)
else
    read(addr2)
```





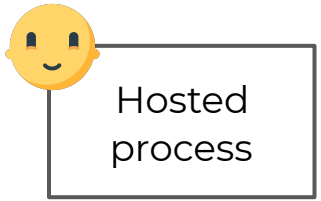
Side-channel attacks

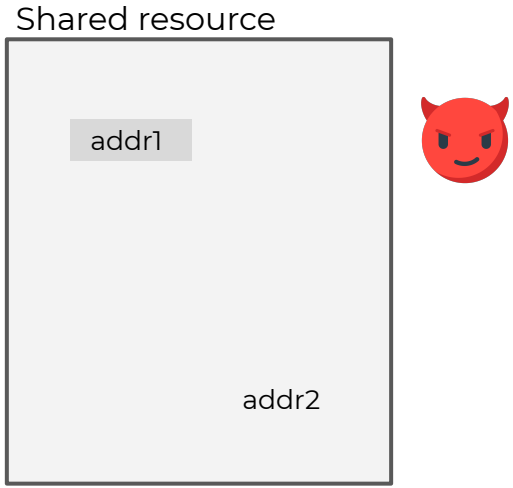
```
if (secret == 0)
    read(addr1)
else
    read(addr2)
```



Side-channel attacks

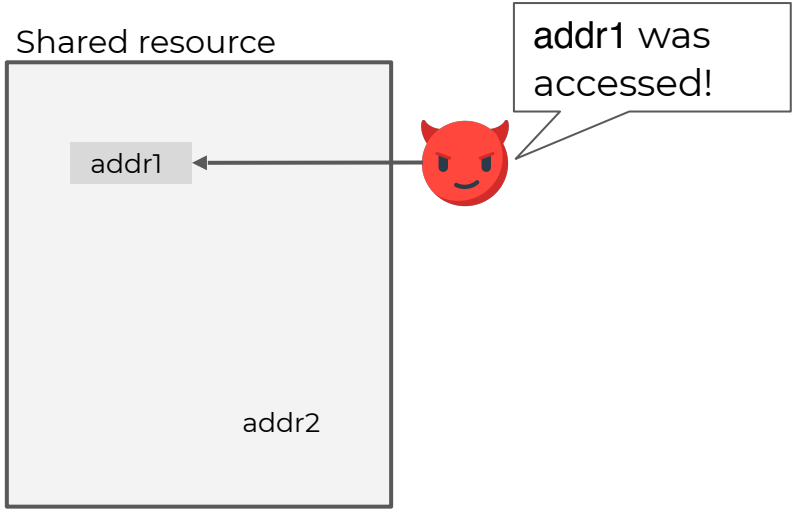
```
if (secret == 0)
    read(addr1)
else
    read(addr2)
```





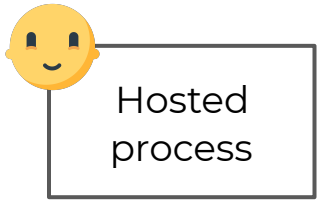
Side-channel attacks

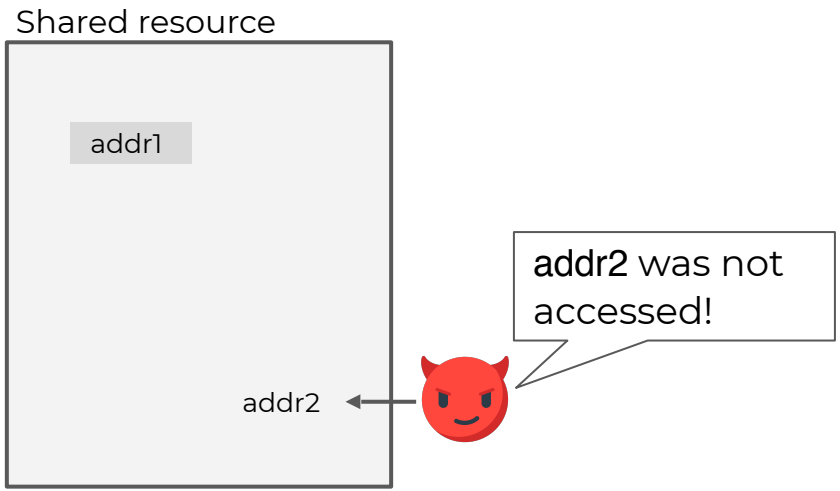
```
if (secret == 0)
    read(addr1)
else
    read(addr2)
```



Side-channel attacks

```
if (secret == 0)
    read(addr1)
else
    read(addr2)
```





Side-channel attacks

```
if (secret == 0)
    read(addr1)
else
    read(addr2)
```

This line was executed

Shared resource

addr1

addr2

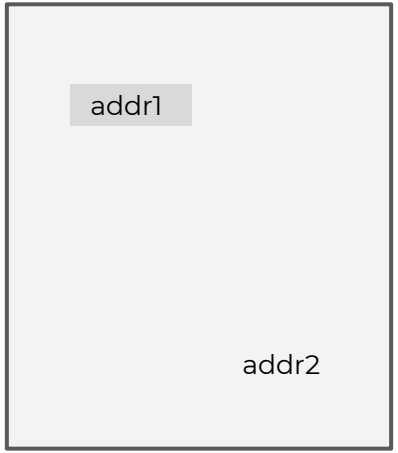
Hosted process

Side-channel attacks

```
if (secret == 0)
    read(addr1)
else
    read(addr2)
```

The secret is **0**

Shared resource



Hosted process

Vulnerable shared resources

- CPU caches
- Page tables
- FPU
- Memory bus
- …

Vulnerable shared resources

- CPU caches (L1, L2)
- Page tables
- FPU
- Memory bus
- ...

} **Varys**

Varys implements a low-cost protection for Intel SGX enclaves against side-channel attacks by creating an **isolated environment** and verifying it at runtime.

Attack requirements

- High interrupt rate
- Predefined cache state
- Shared core

Attack requirements

- ~~High interrupt rate~~
- ~~Predefined cache state~~
- ~~Shared core~~

} Isolated environment

Varys implements a low-cost protection for Intel SGX enclaves against side-channel attacks by **creating** an isolated environment and **verifying it at runtime**.

Design

- High preemption rate
- Predefined cache state
- Shared core

Restrict and terminate

Cache eviction

Trusted reservation

Design

- High preemption rate
- Predefined cache state
- Shared core

Restrict and terminate

Cache eviction

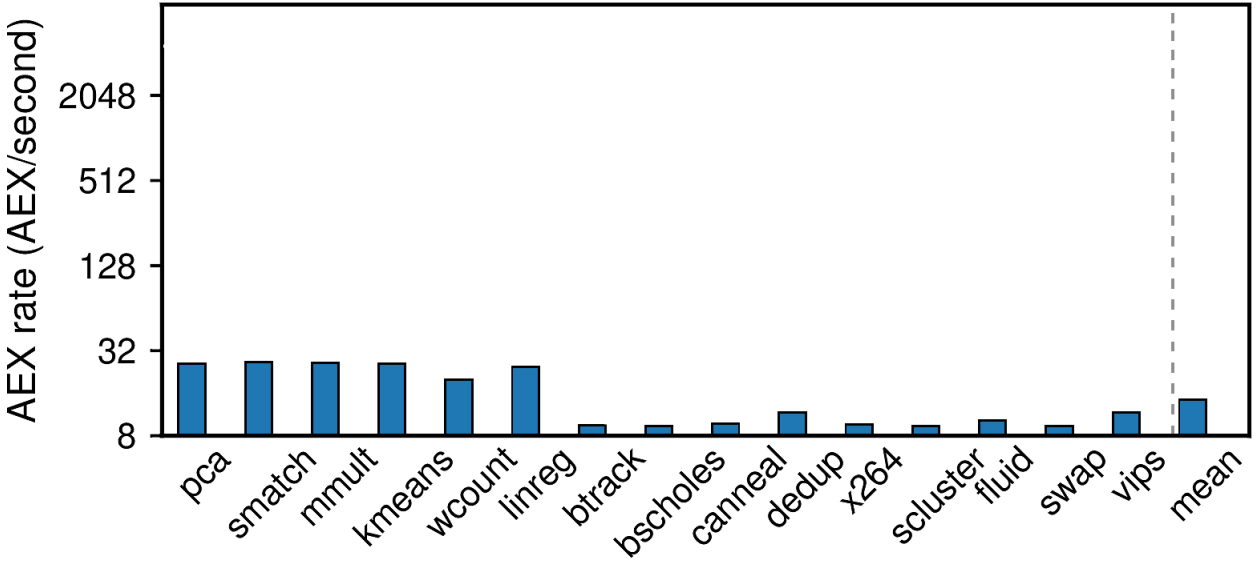
Trusted reservation

Restricting preemption rate

- Attack exit rate: ~ 5000 exits/s.

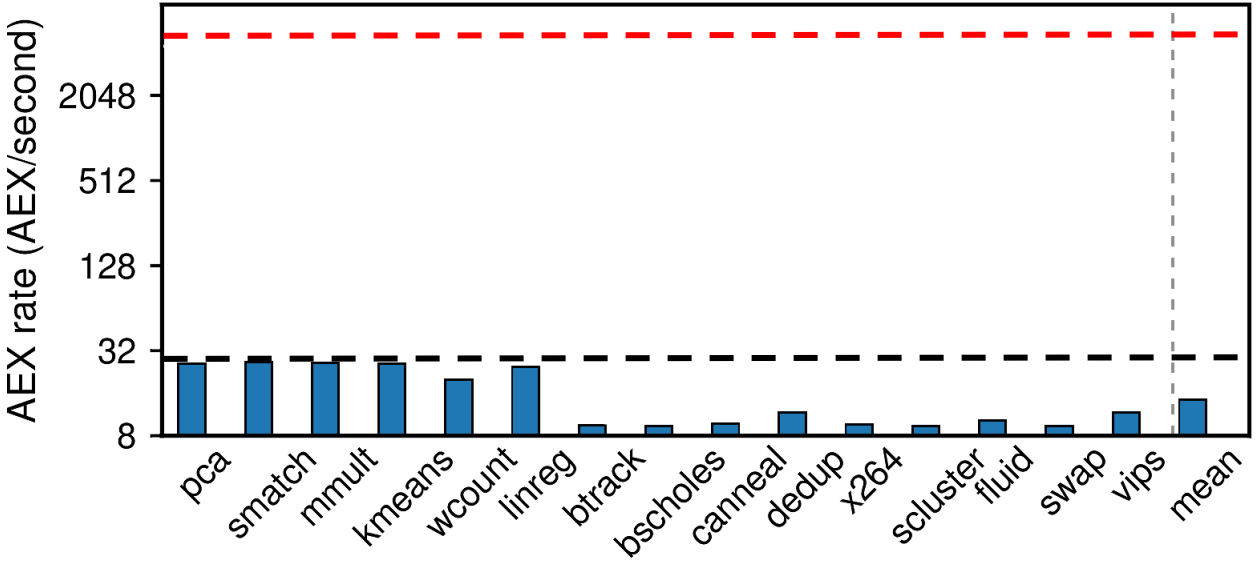
Restricting preemption rate

- Attack exit rate: ~ 5000 exits/s.
- Normal exit rate: ~ 30 exits/s.

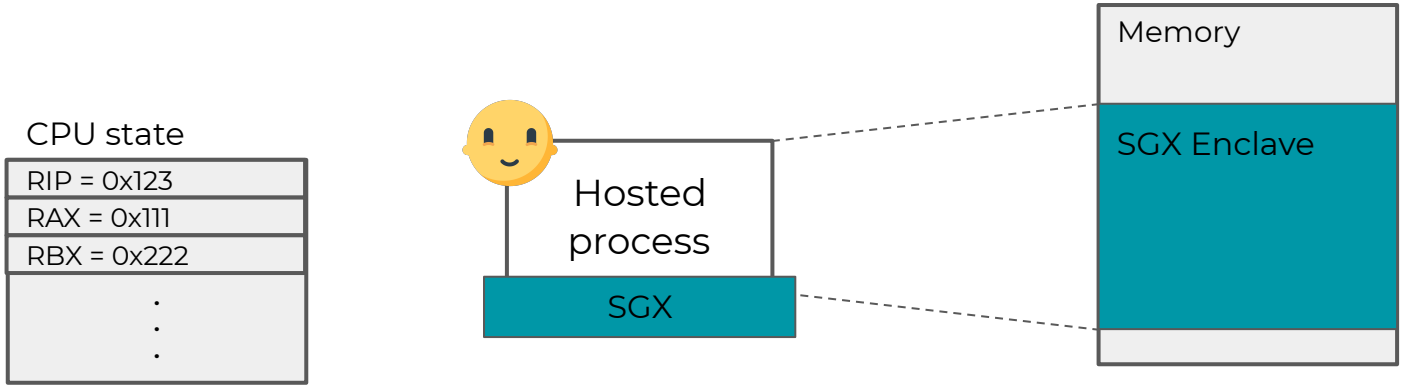


Restricting preemption rate

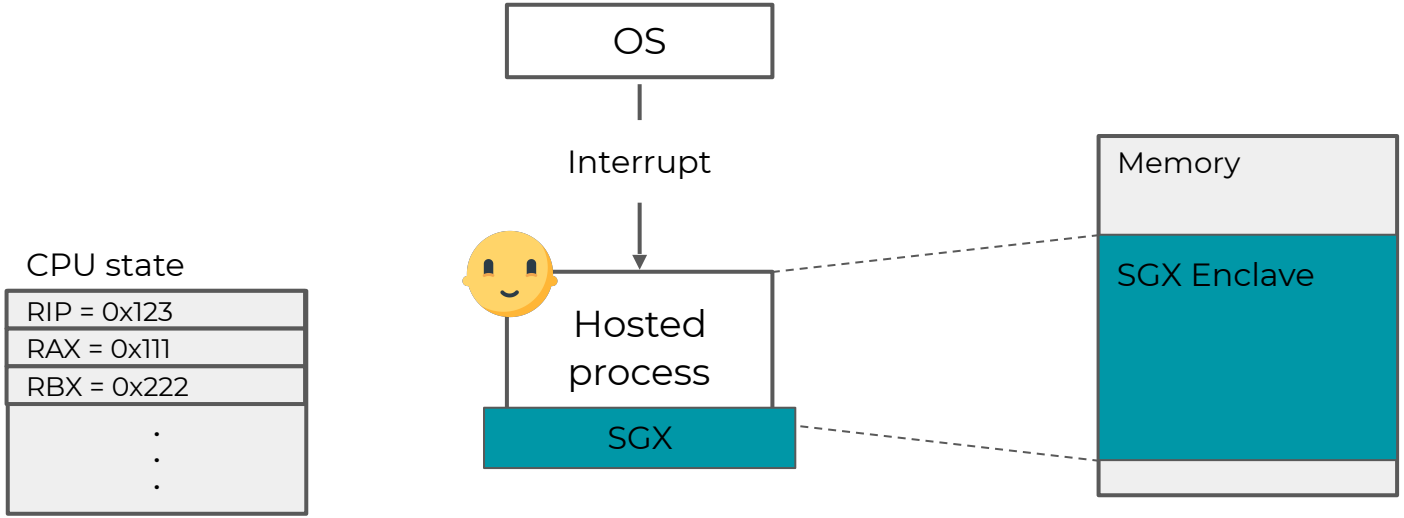
- Attack exit rate: ~ 5000 exits/s.
- Normal exit rate: ~ 30 exits/s.



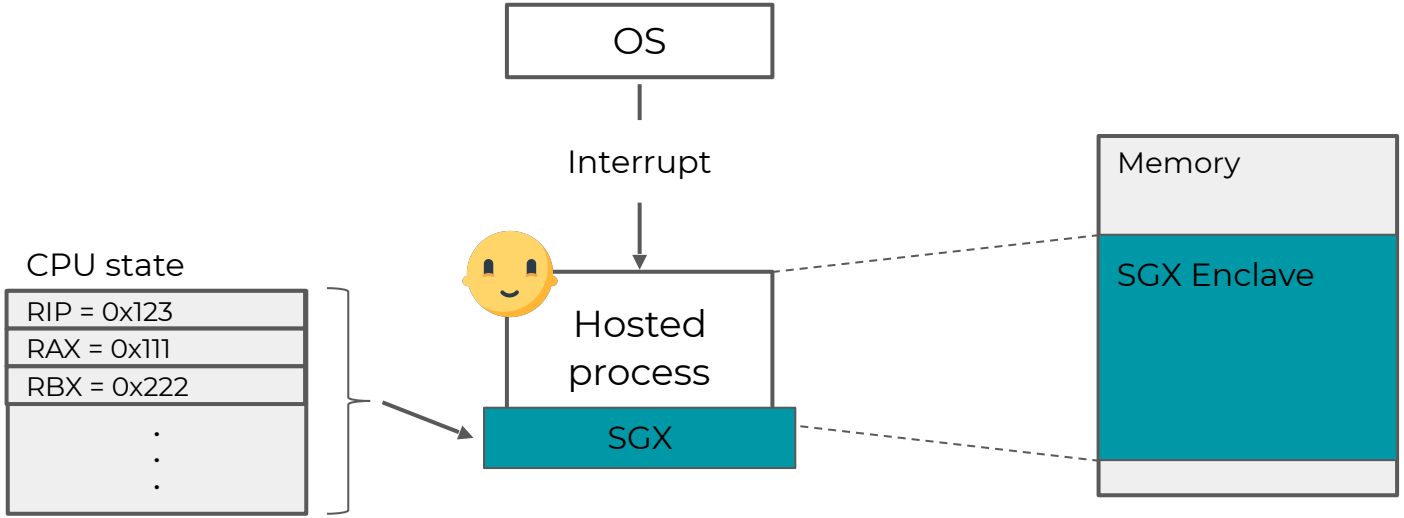
Asynchronous Enclave Exit (AEX)



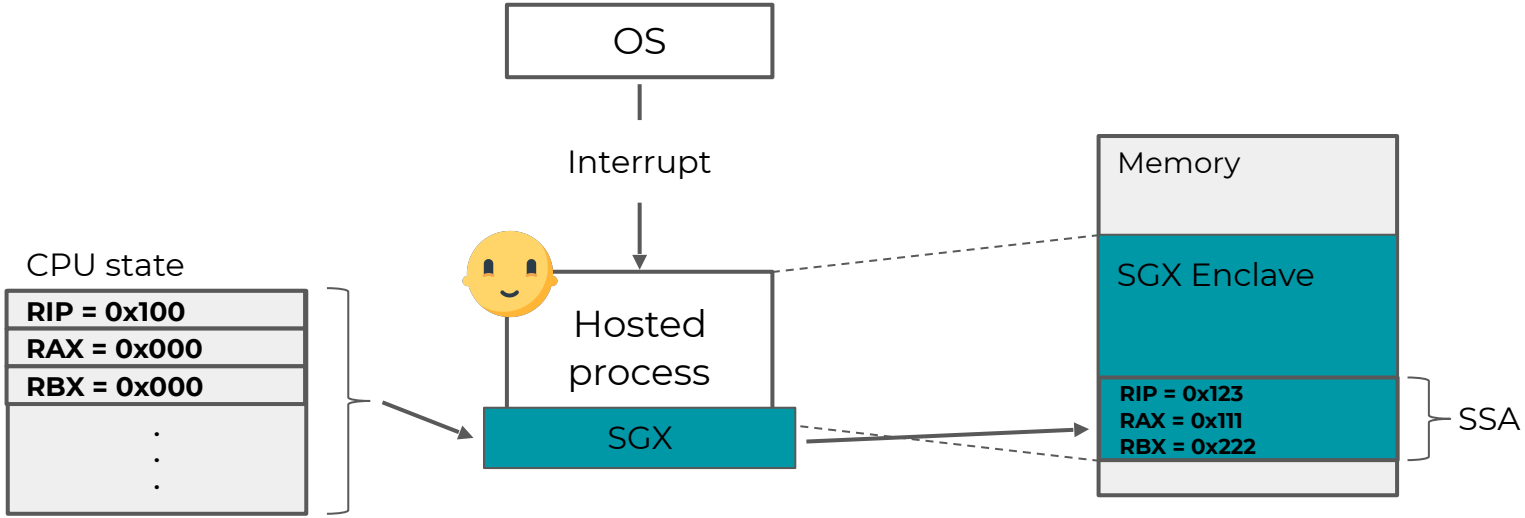
Asynchronous Enclave Exit (AEX)



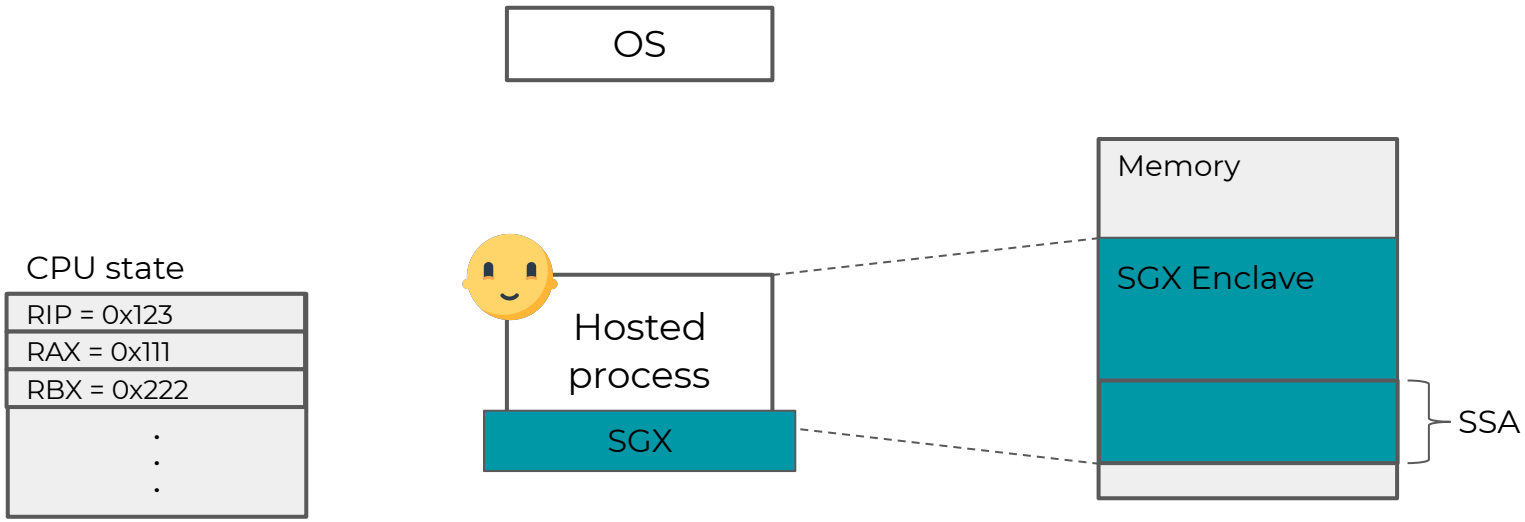
Asynchronous Enclave Exit (AEX)



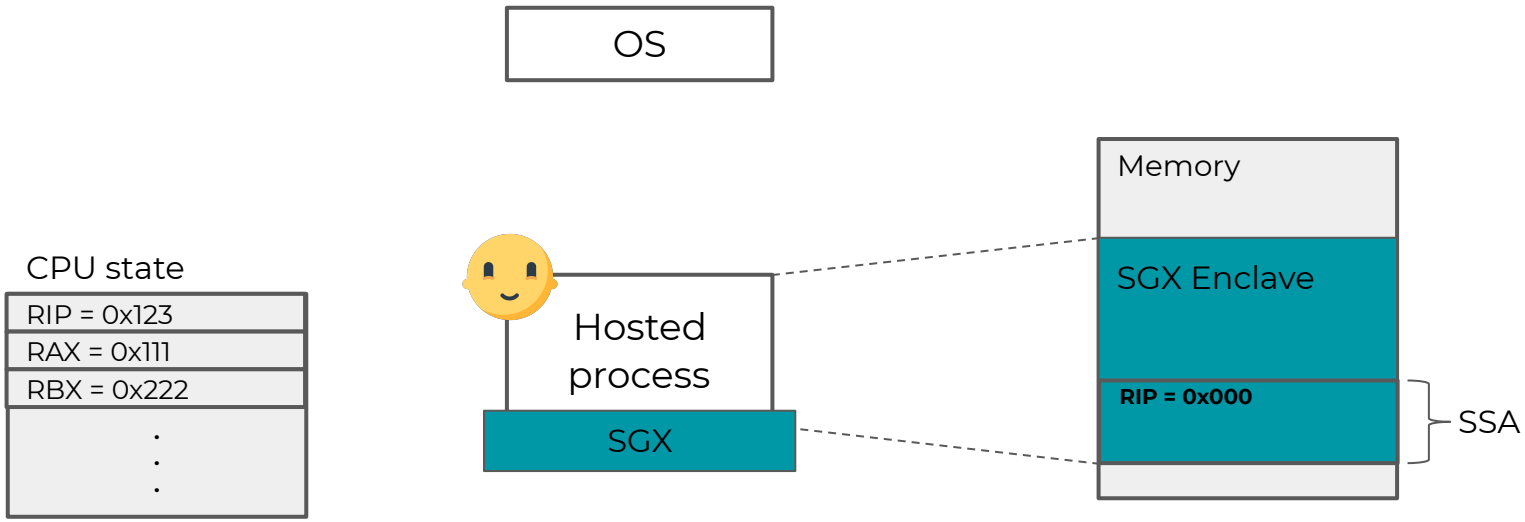
Asynchronous Enclave Exit (AEX)



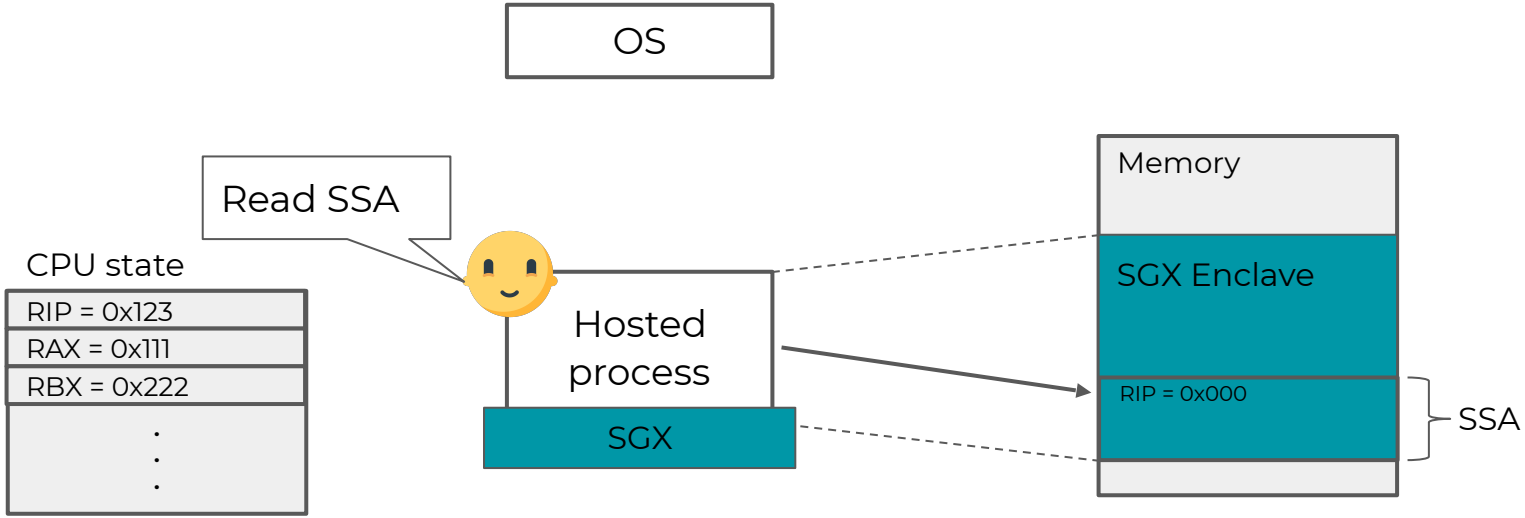
Detecting interrupts



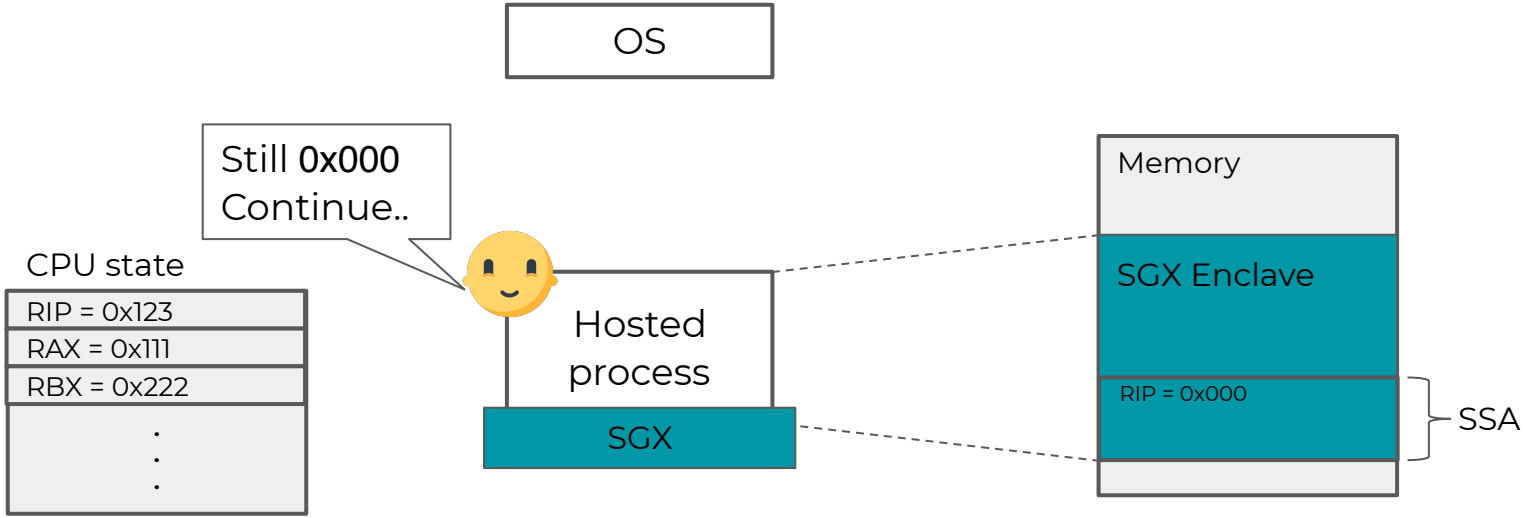
Detecting interrupts



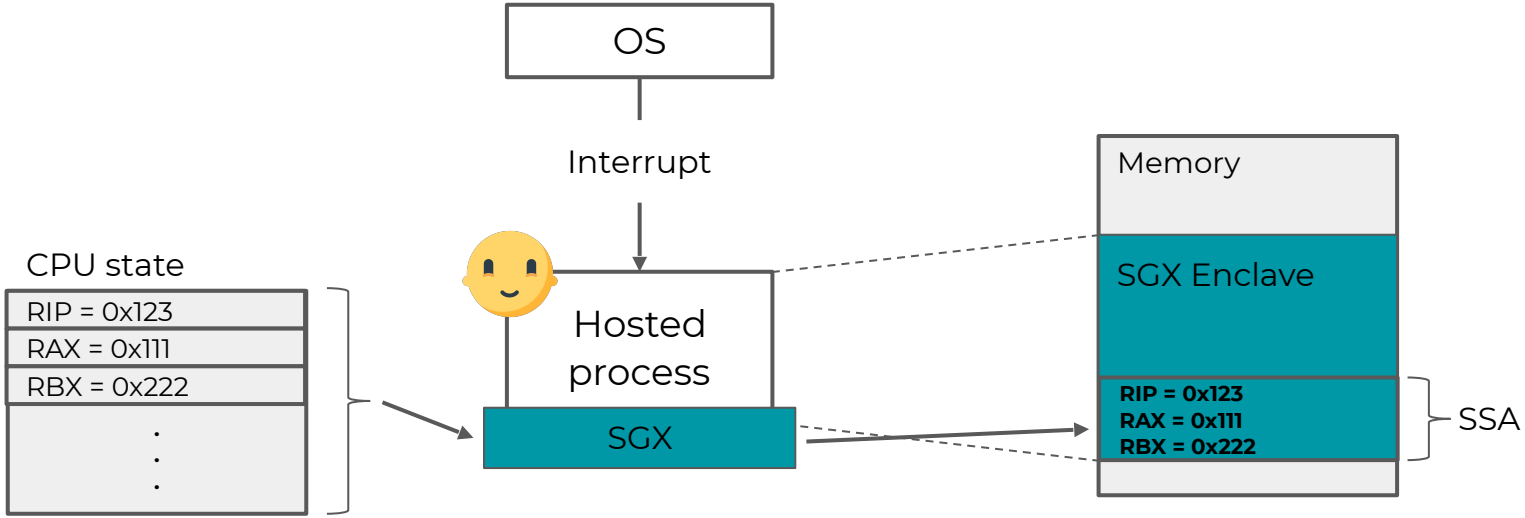
Detecting interrupts



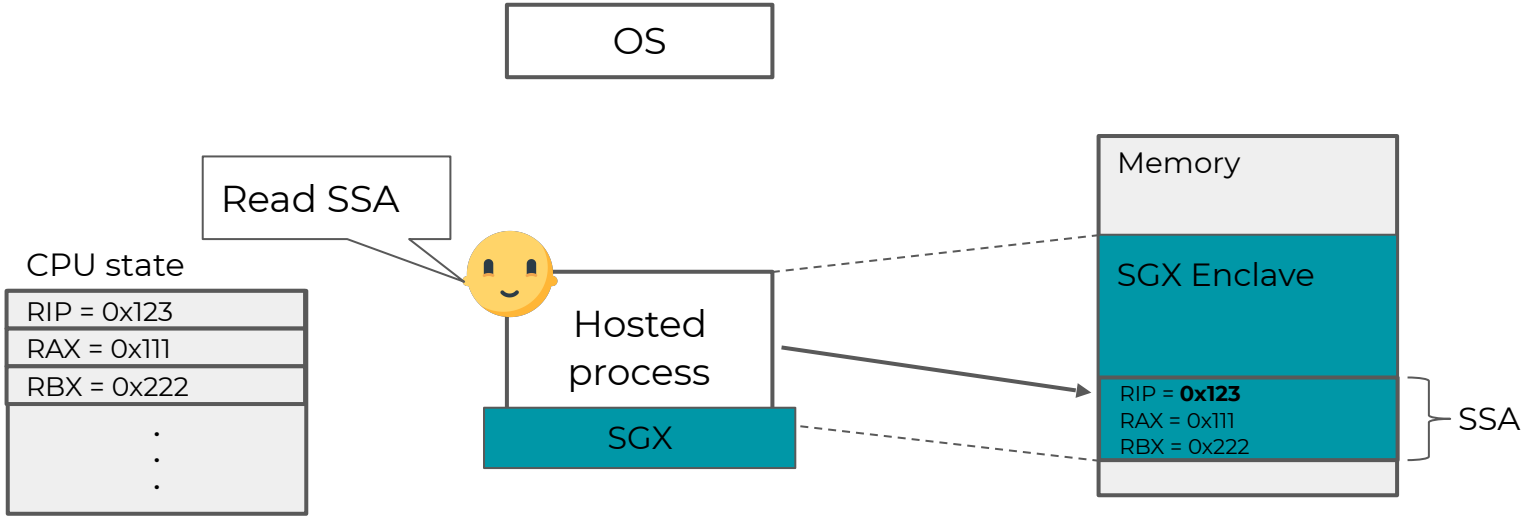
Detecting interrupts



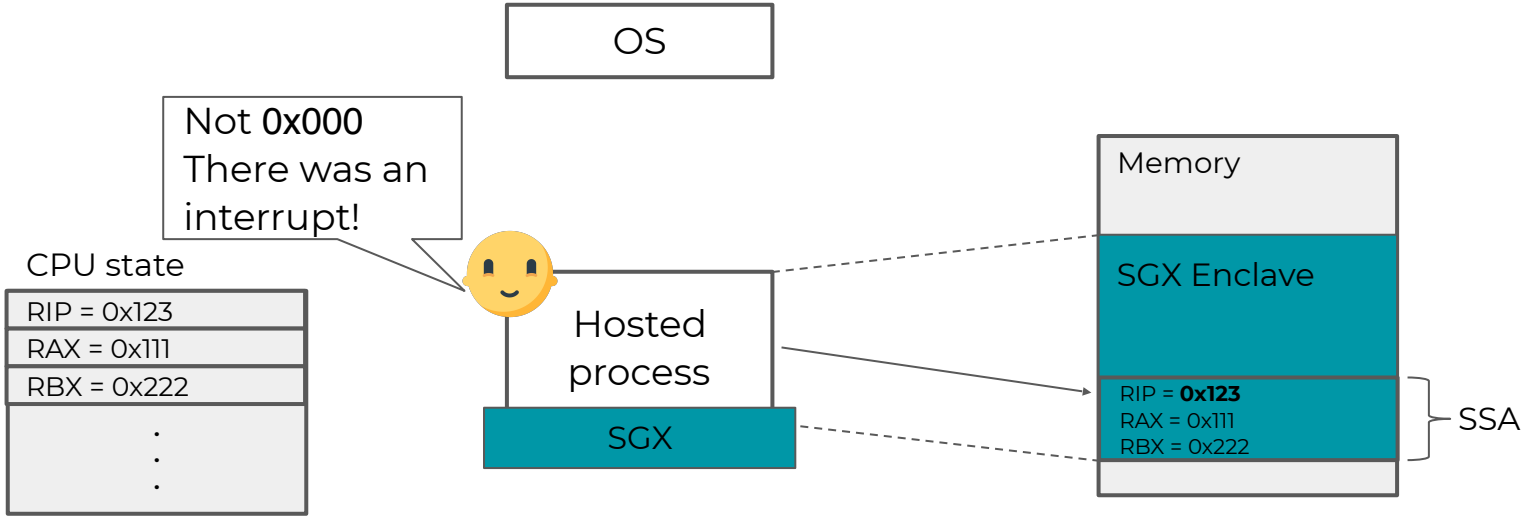
Detecting interrupts



Detecting interrupts



Detecting interrupts



Design

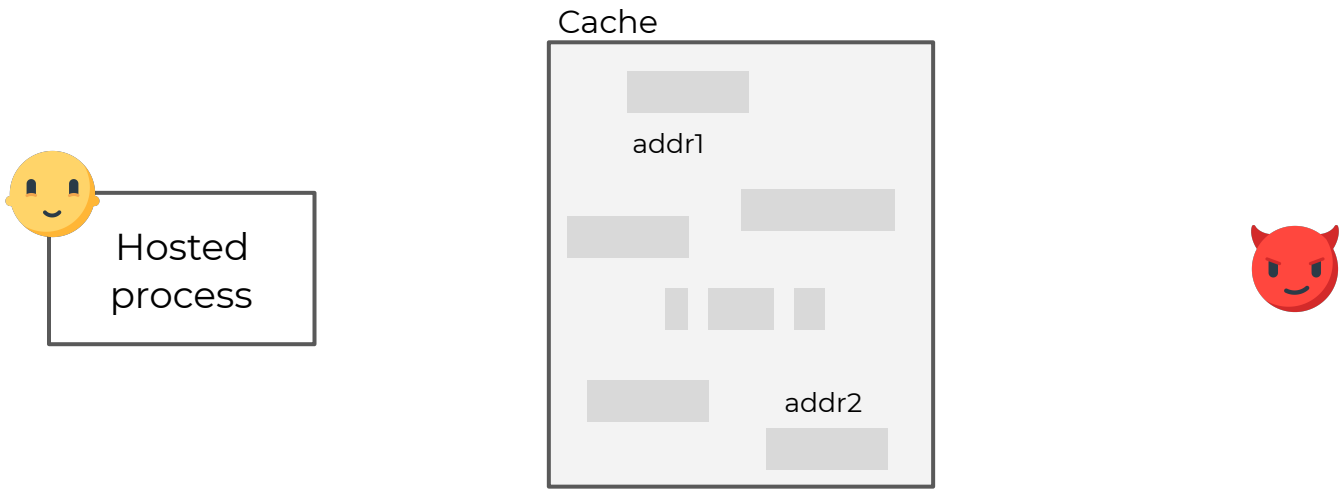
- High preemption rate
- Predefined cache state
- Shared core

Restrict and terminate

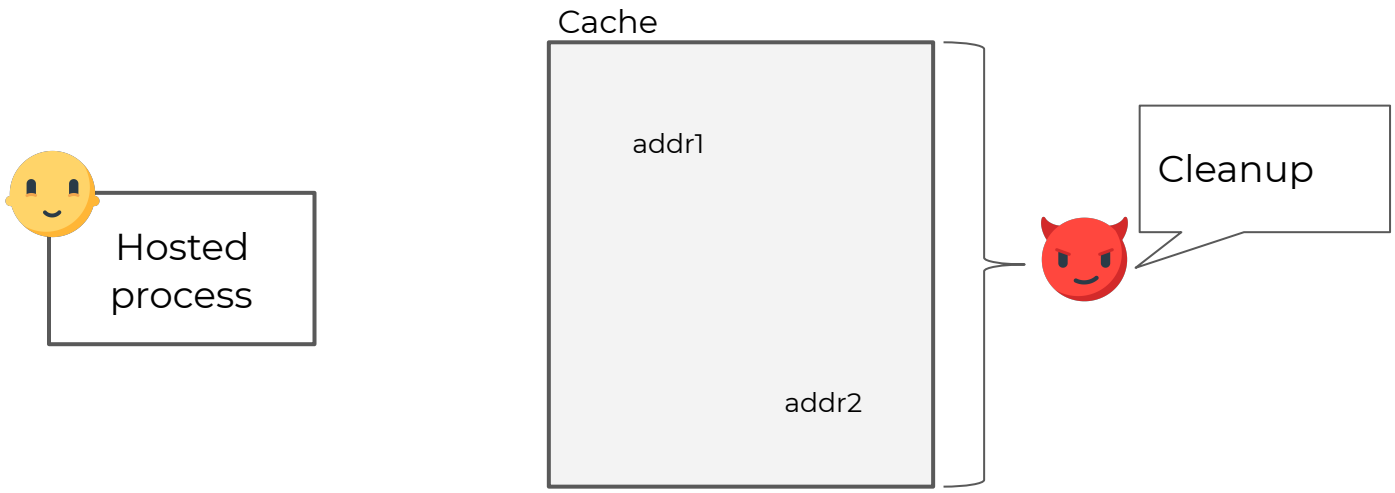
Cache eviction

Trusted reservation

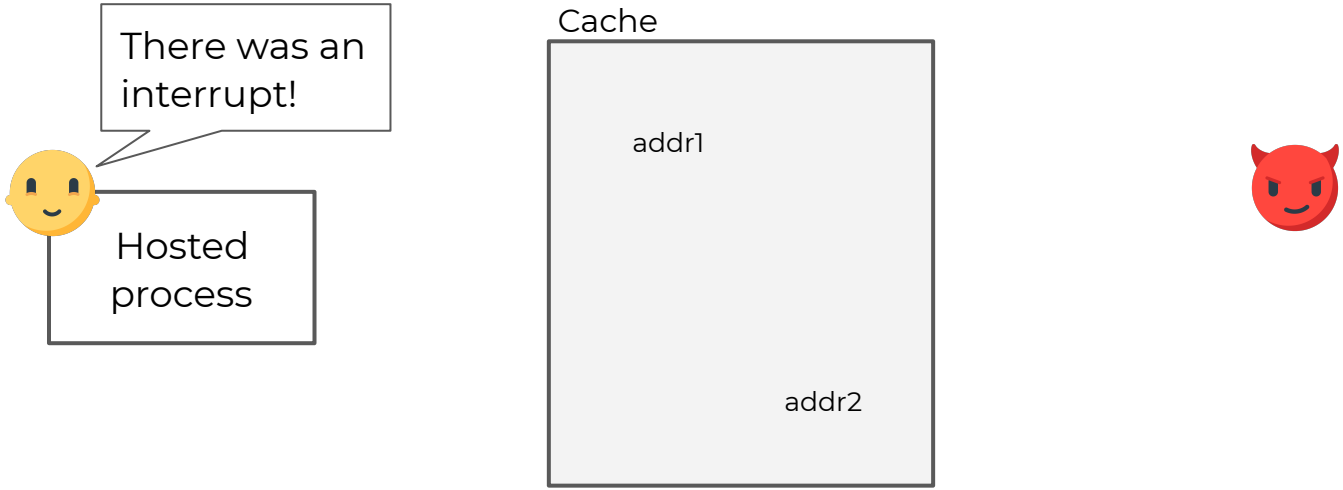
Hiding cache traces



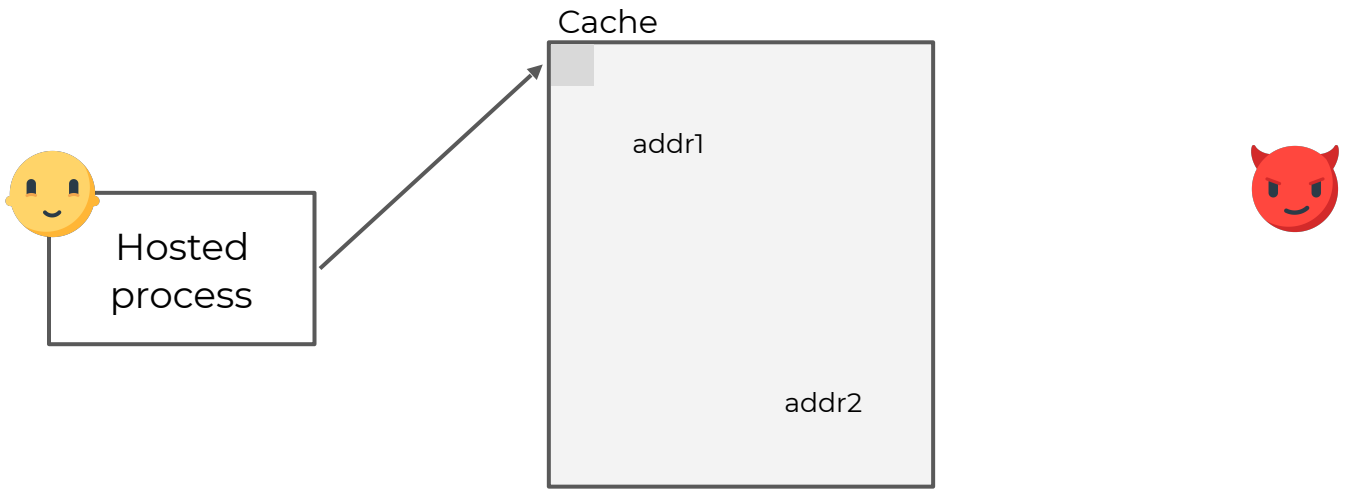
Hiding cache traces



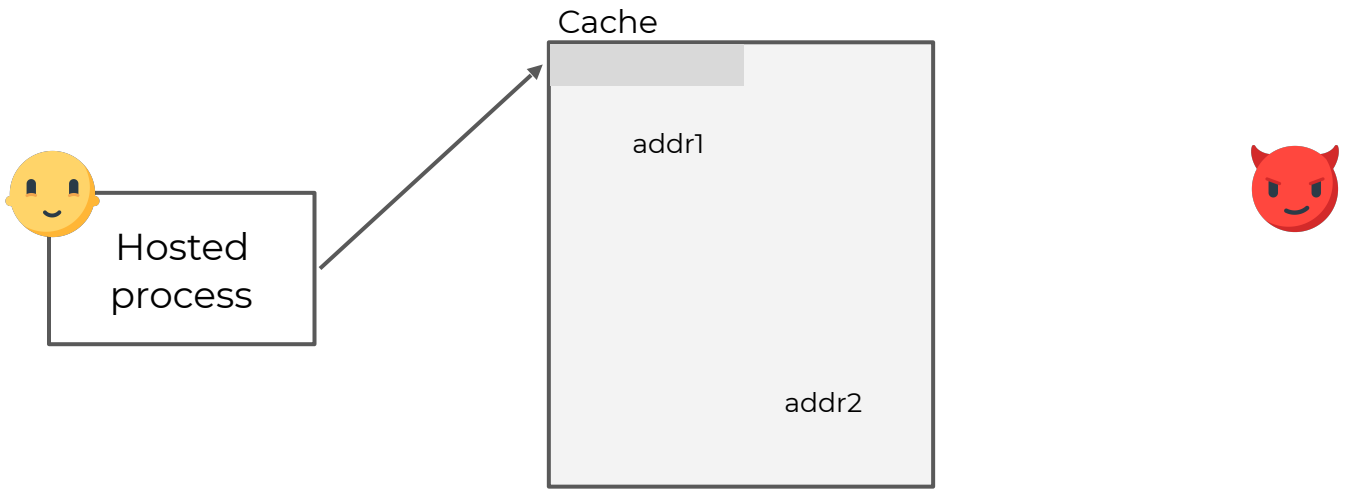
Hiding cache traces



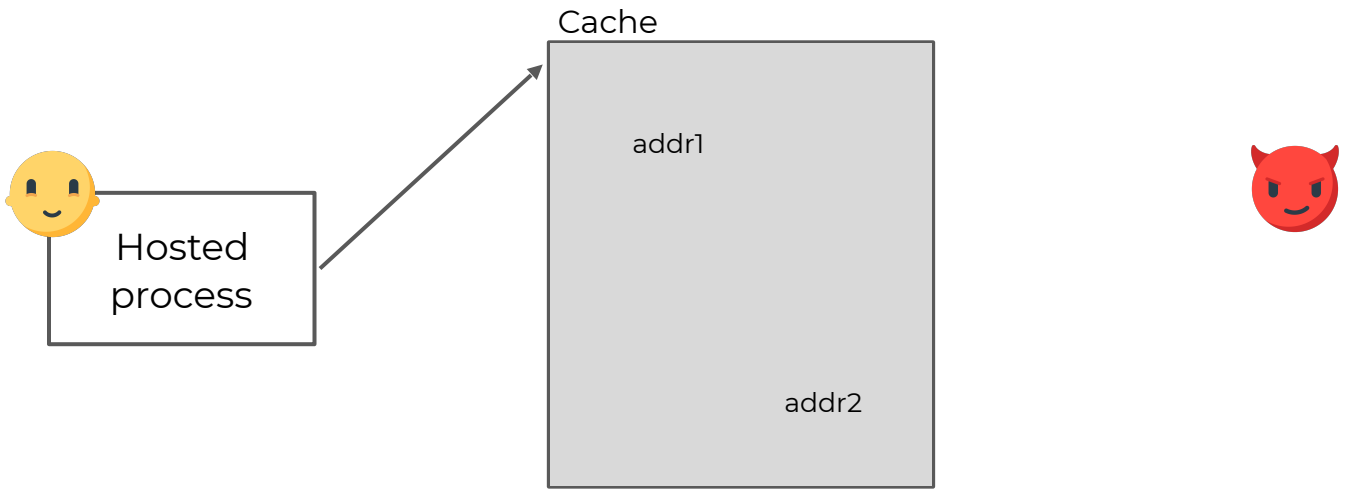
Hiding cache traces



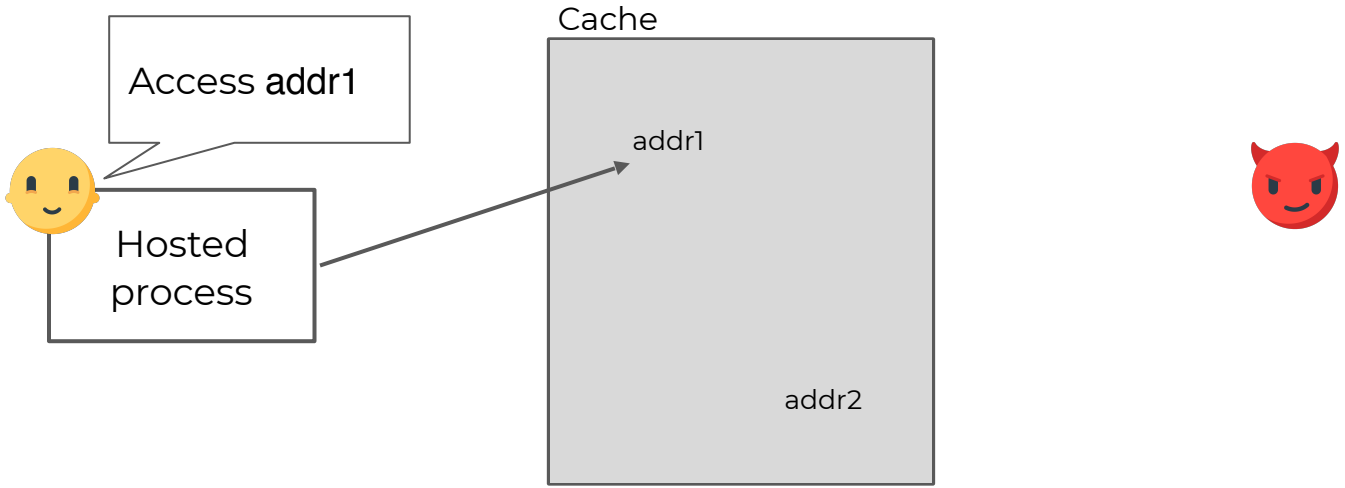
Hiding cache traces



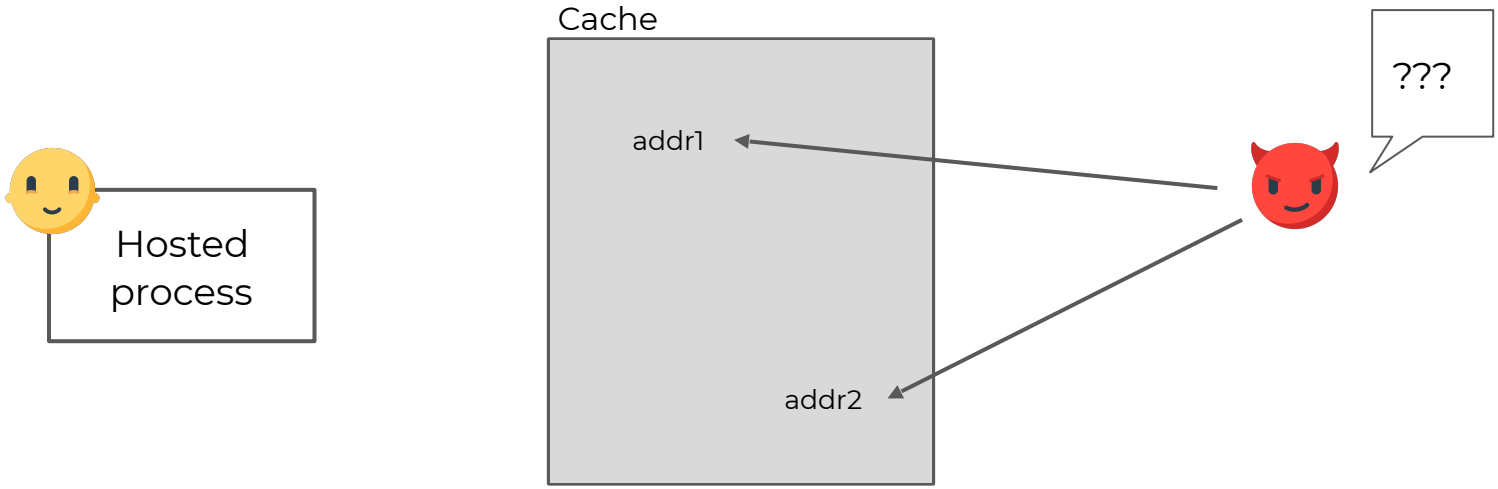
Hiding cache traces



Hiding cache traces



Hiding cache traces



Design

- High preemption rate
- Predefined cache state
- Shared core

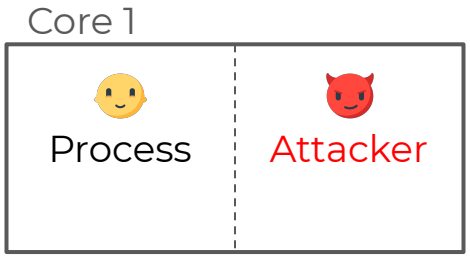
Restrict and terminate

Cache eviction

Trusted reservation

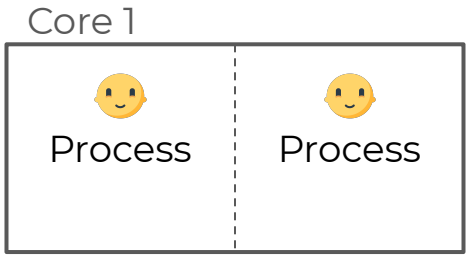
Preventing core sharing

- Occupy both hyperthreads

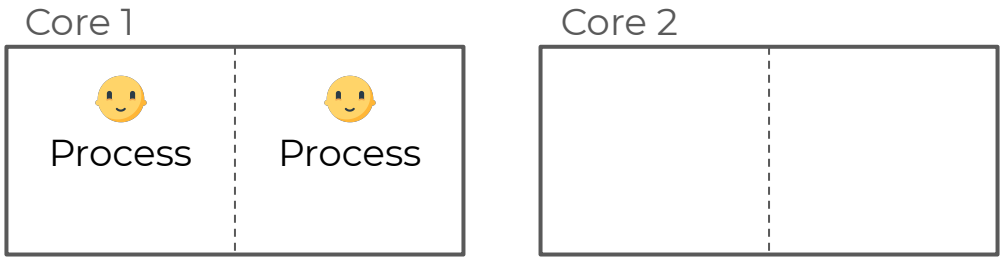


Preventing core sharing

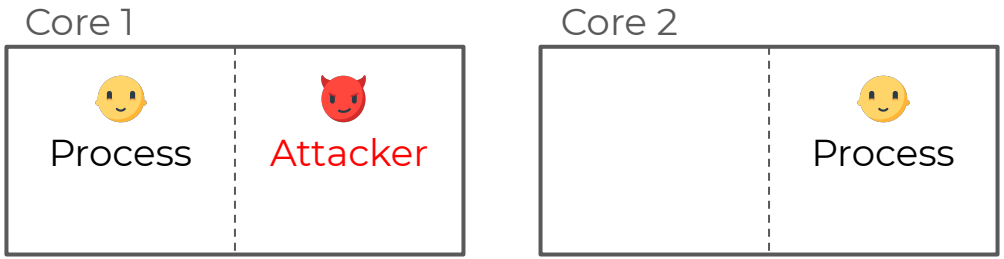
- Occupy both hyperthreads
  - Use process affinity



How do we ensure reservation?

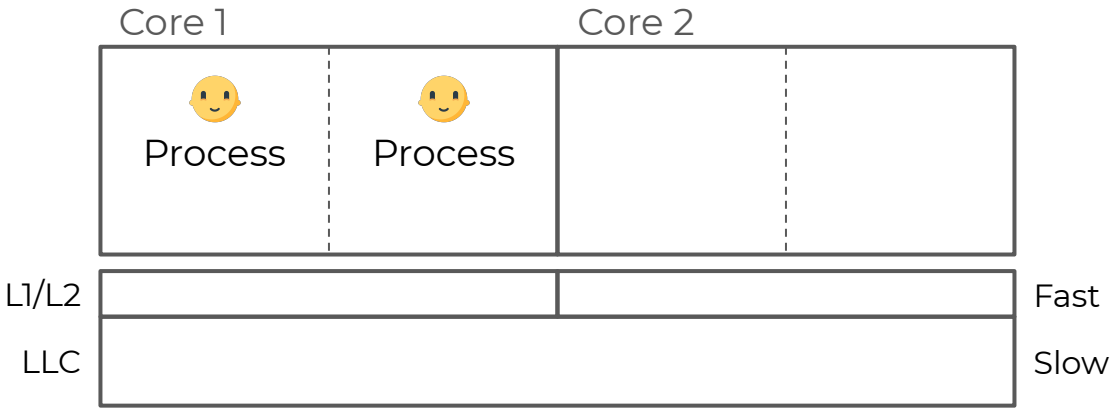


How do we ensure reservation?



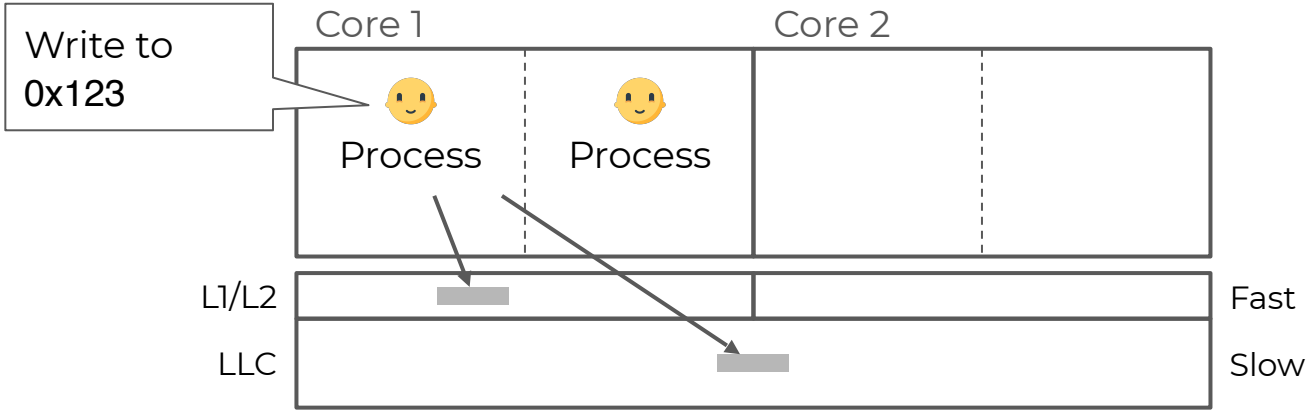
Handshake

- Use shared access timing



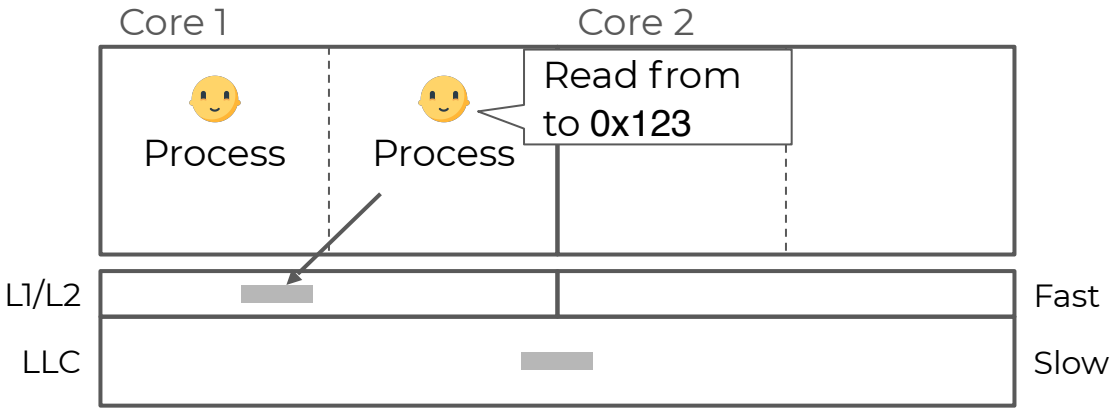
Handshake

- Use shared access timing



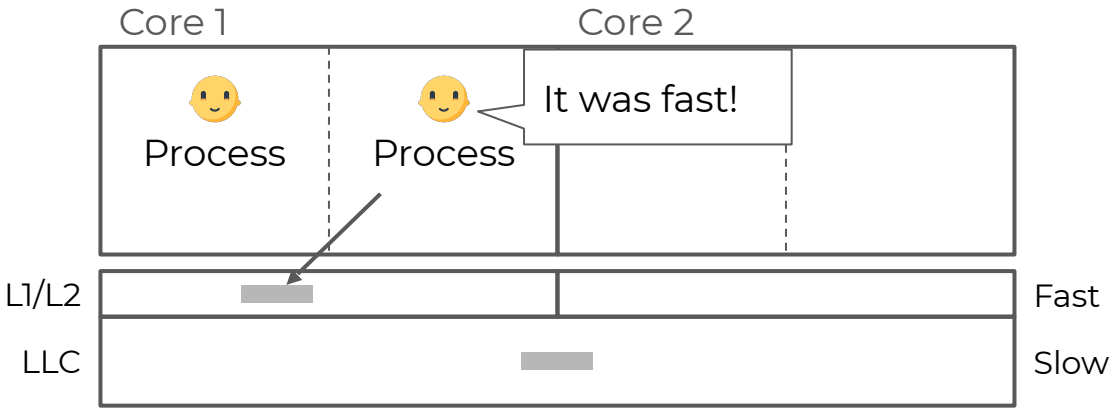
Handshake

- Use shared access timing



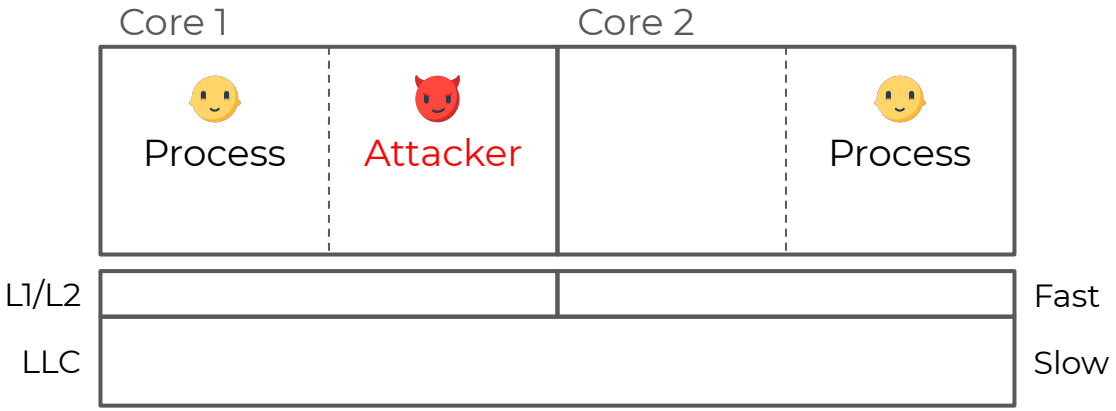
Handshake

- Use shared access timing



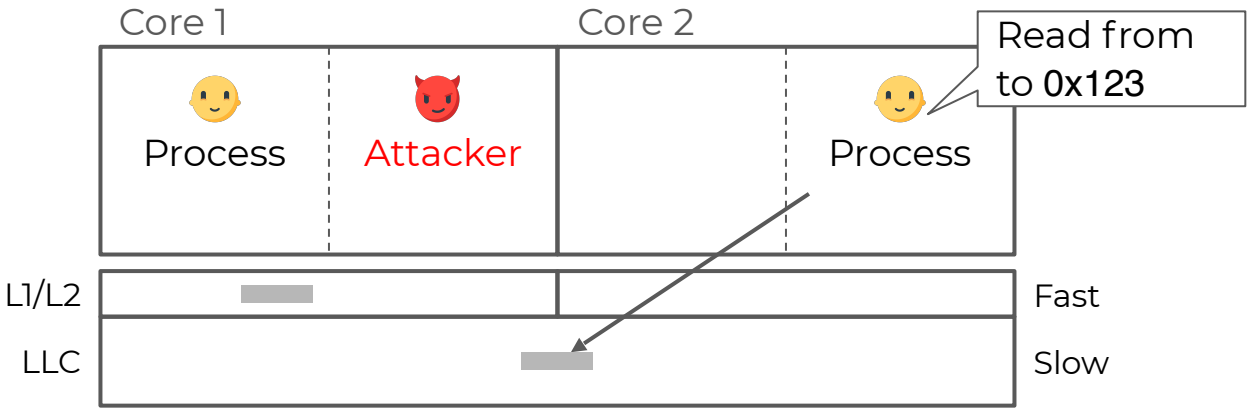
Handshake

- Use shared access timing



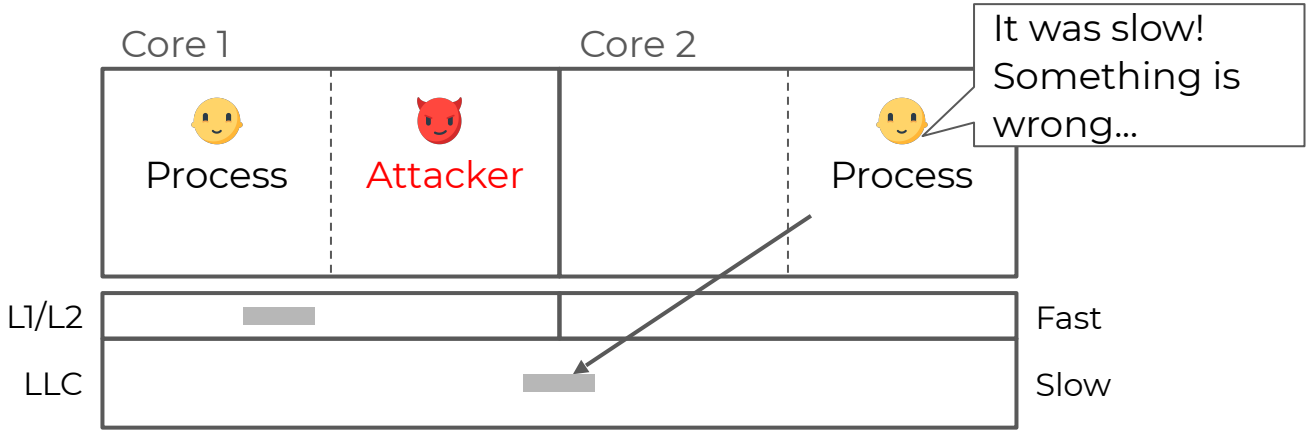
Handshake

- Use shared access timing



Handshake

- Use shared access timing



Design

- High preemption rate
- Predefined cache state
- Shared core

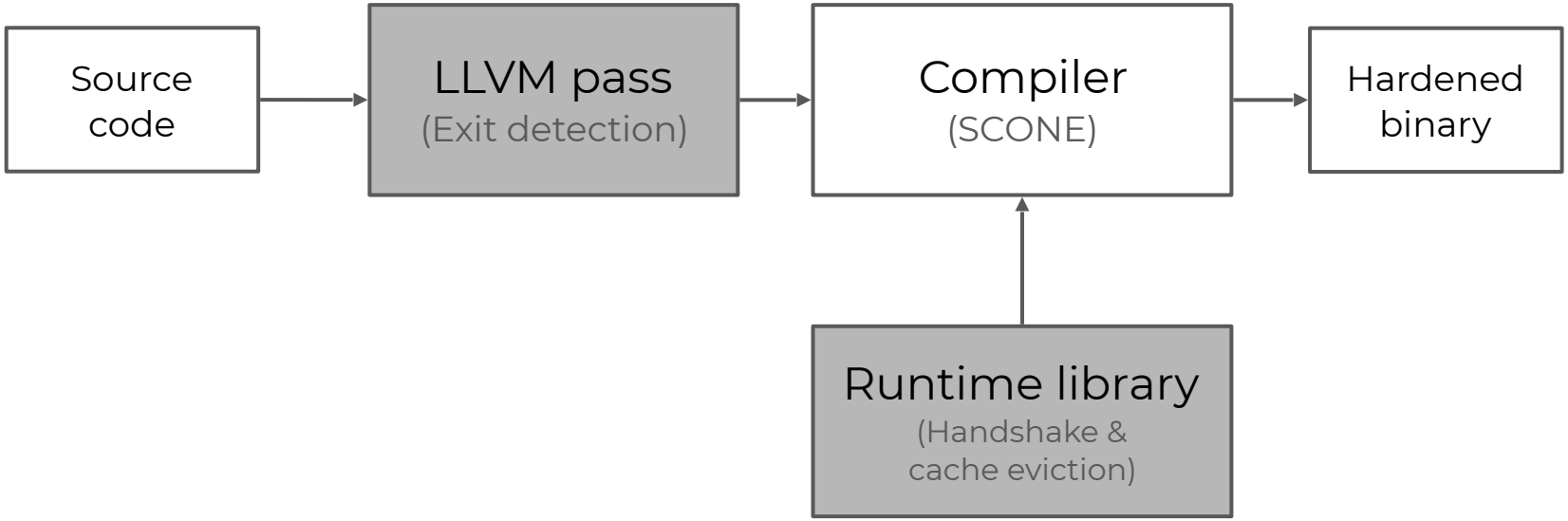
Restrict and terminate

Cache eviction

Trusted reservation

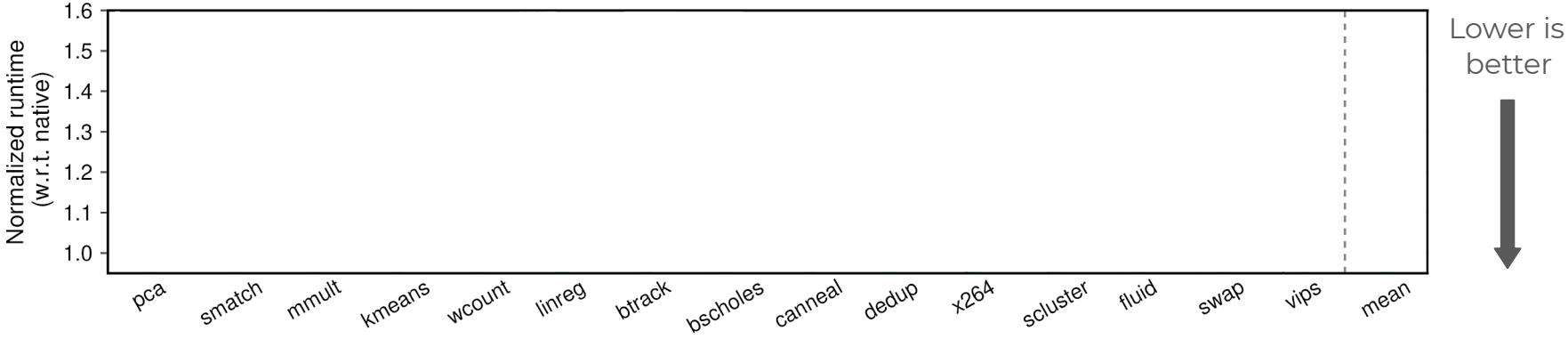
Varys **implements** a low-cost protection for Intel SGX enclaves against side-channel attacks by creating an isolated environment and verifying it at runtime.

Implementation

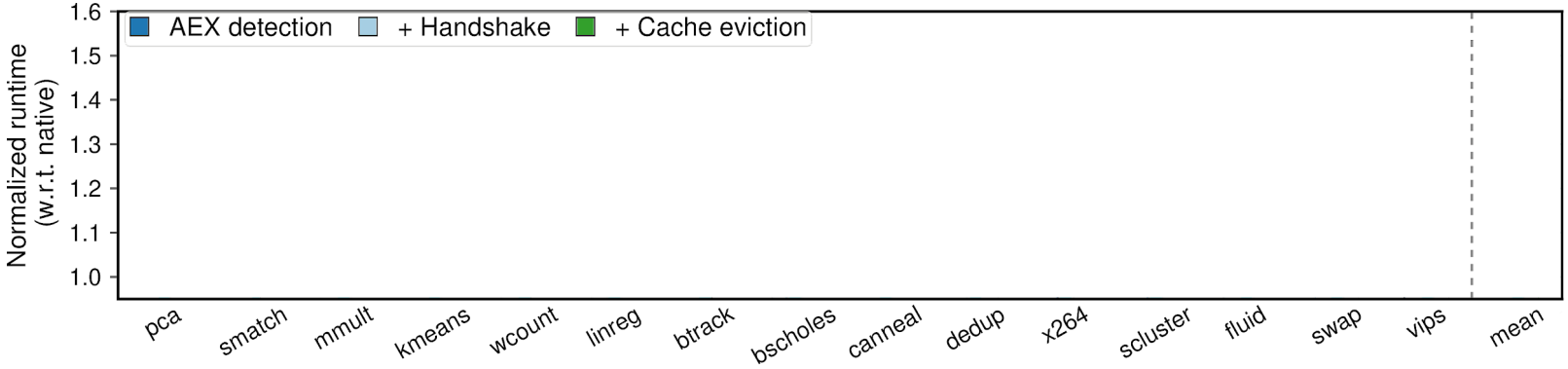


Varys implements a **low-cost** protection for Intel SGX enclaves against side-channel attacks by creating an isolated environment and verifying it at runtime.

Evaluation: performance

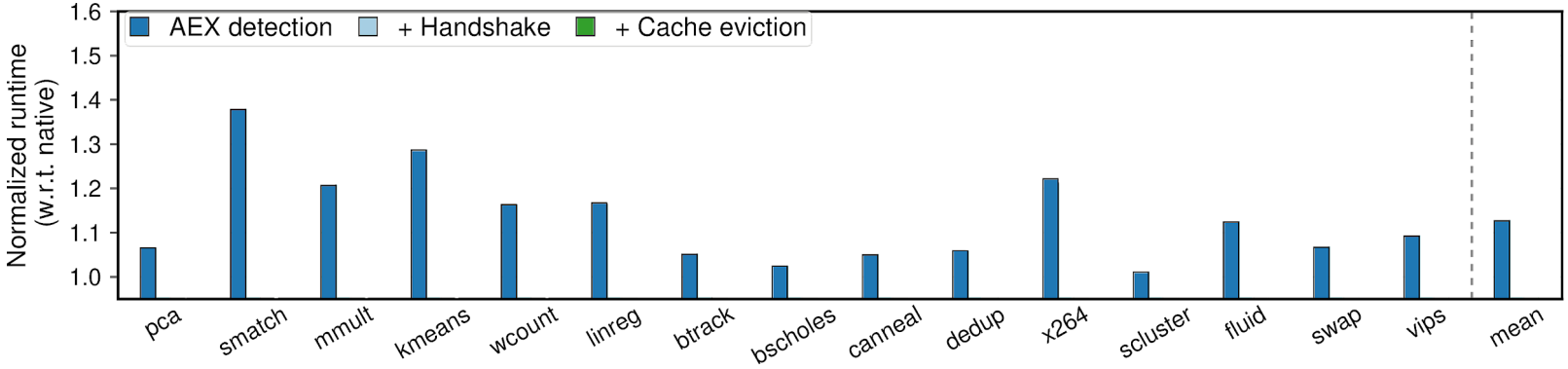


Evaluation: performance



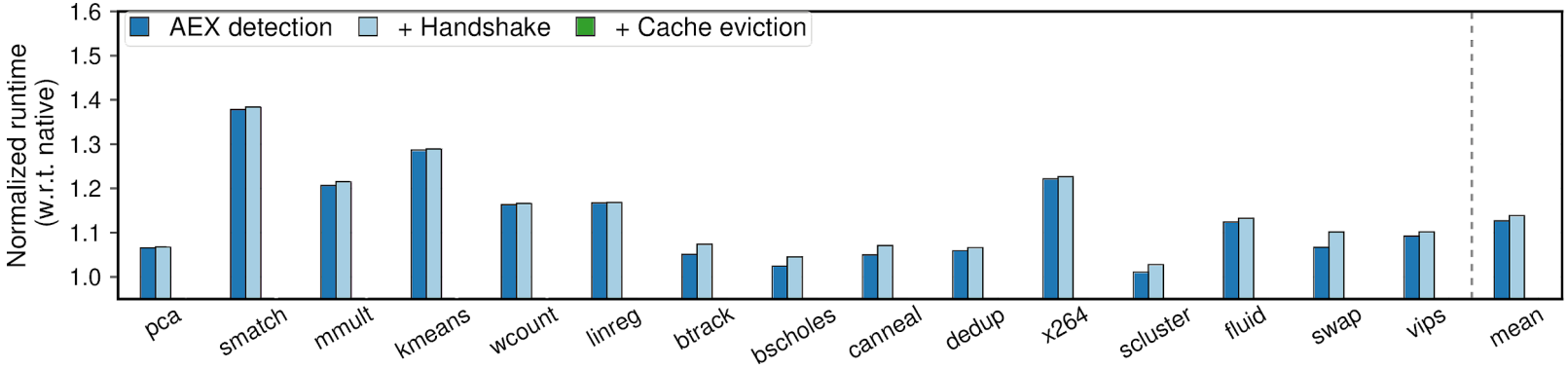
Lower is better

Evaluation: performance



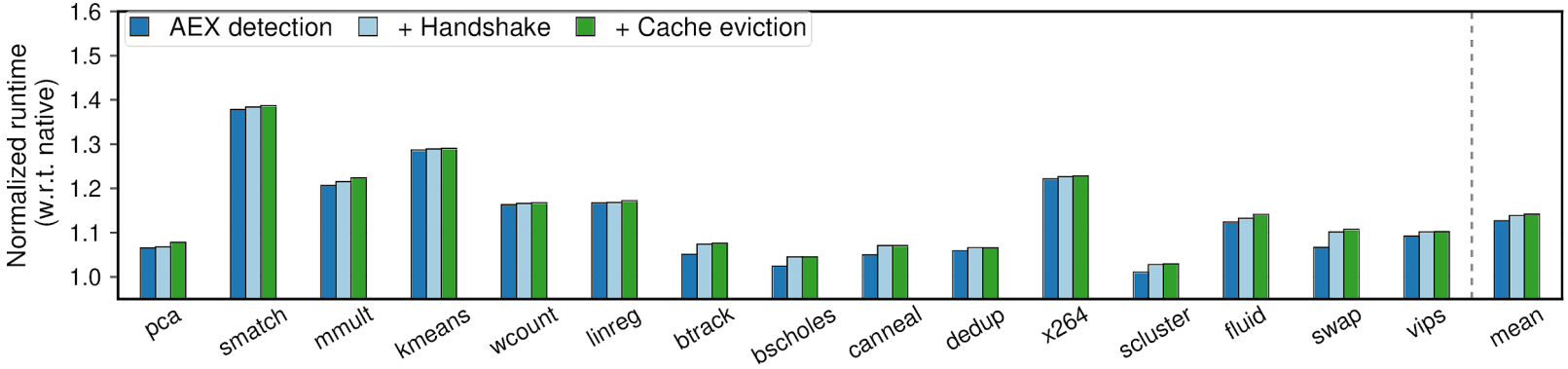
Lower is better

Evaluation: performance



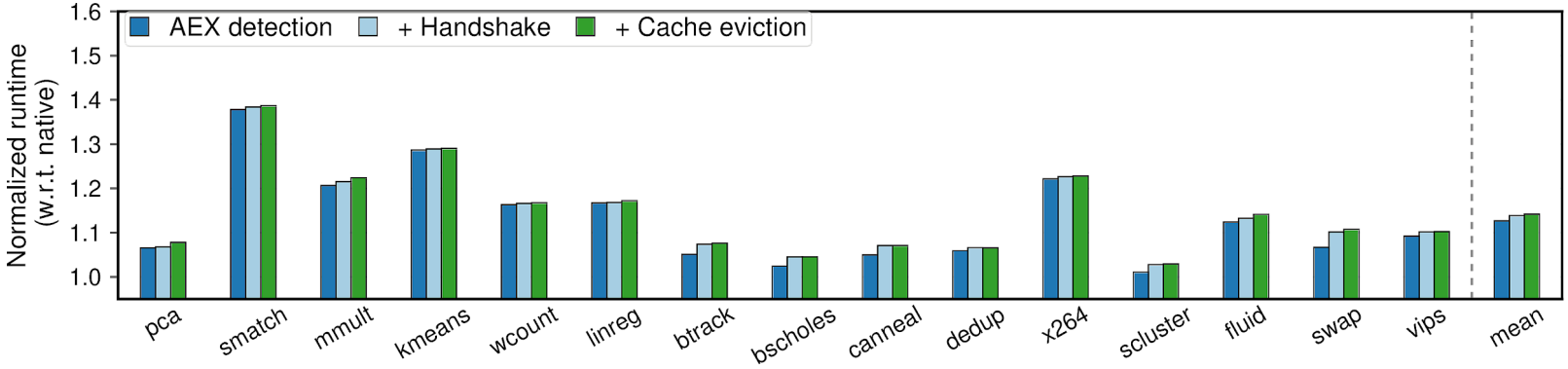
Lower is better

Evaluation: performance



Lower is better

Evaluation: performance



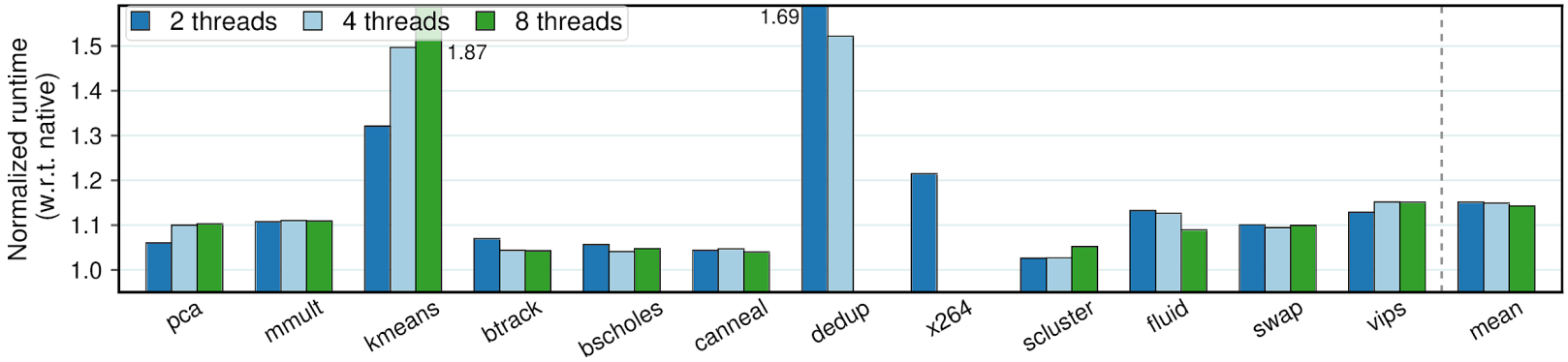
Lower is better

Handshake and eviction **only at enclave exits**

- 20-30 times per second

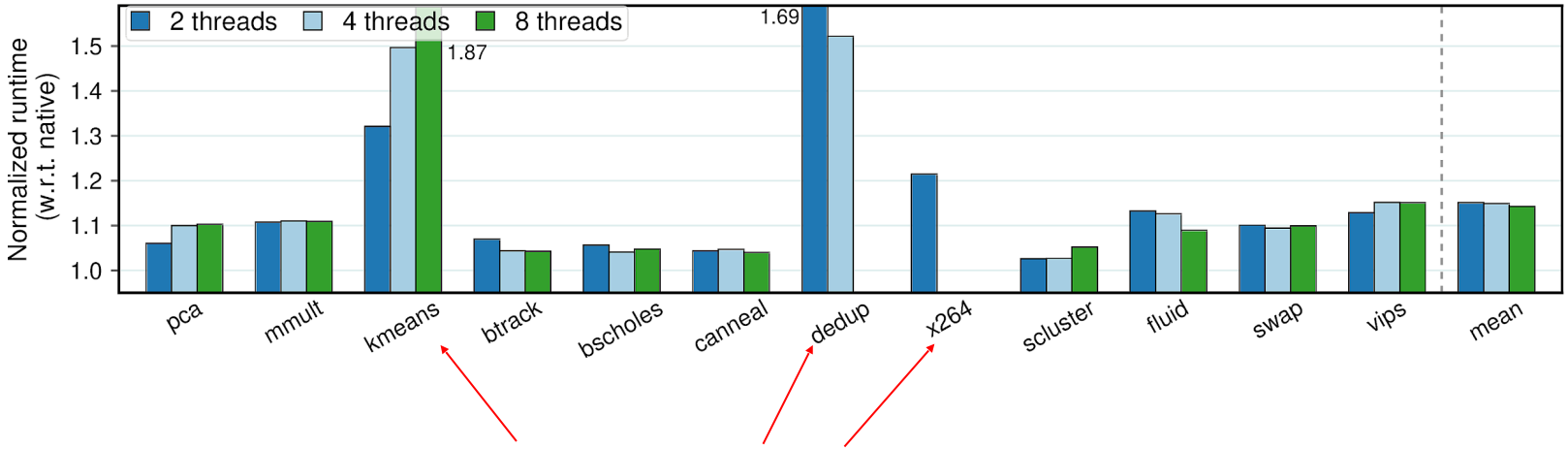
Evaluation: multithreading

Evaluation: multithreading



Lower is better

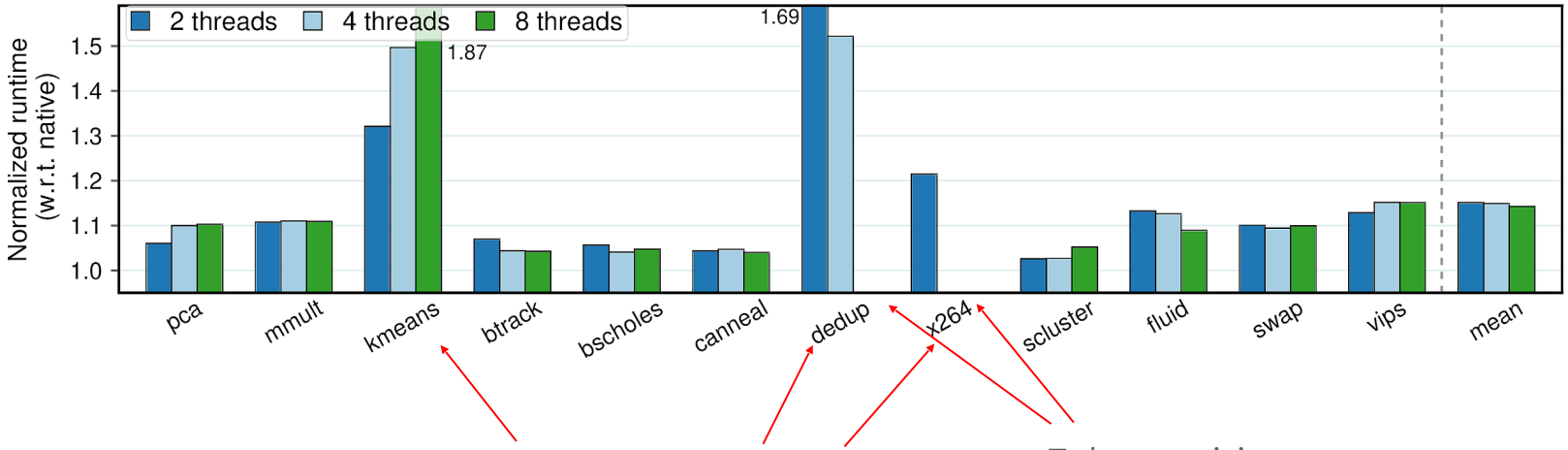
Evaluation: multithreading



Lower is better

EPC paging ⇒ higher exit rate

Evaluation: multithreading



Lower is better

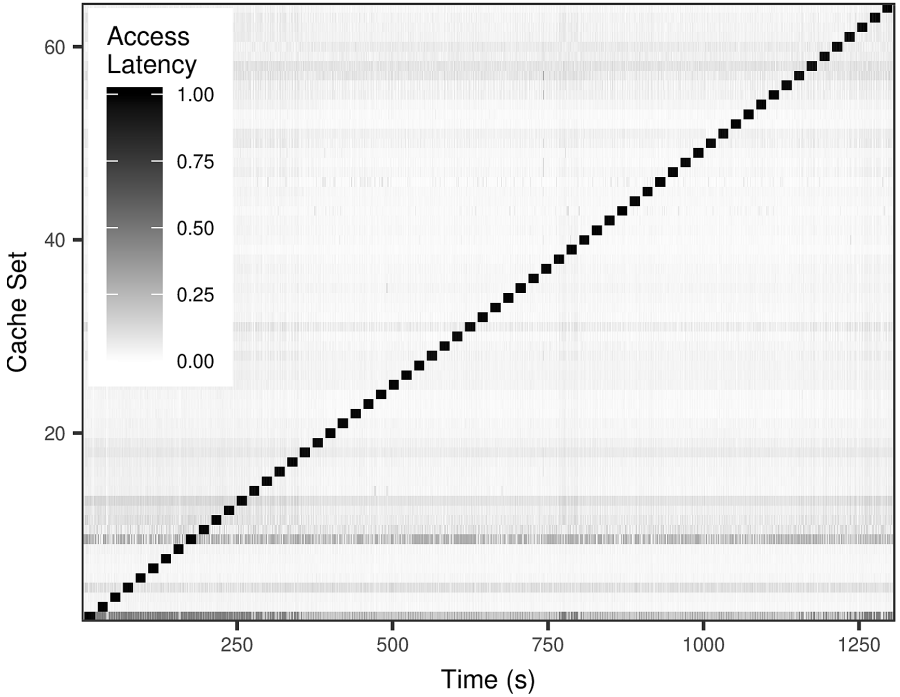
EPC paging ⇒ higher exit rate

False positives

Varys implements a low-cost **protection** for Intel SGX enclaves against side-channel attacks by creating an isolated environment and verifying it at runtime.

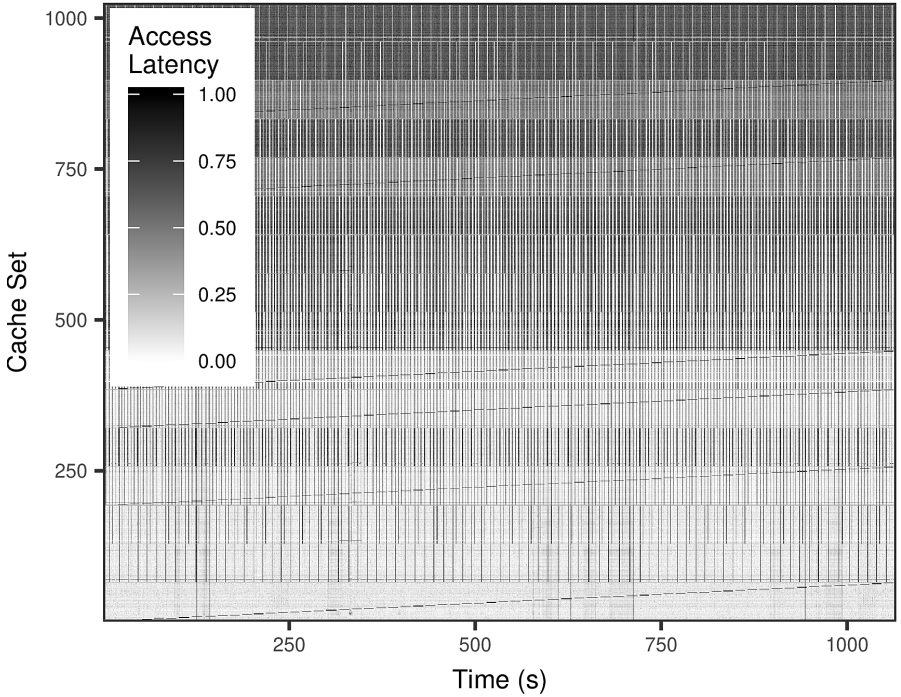
Evaluation: security

- Privileged cache SCA
  - Target: L1 cache
- No eviction



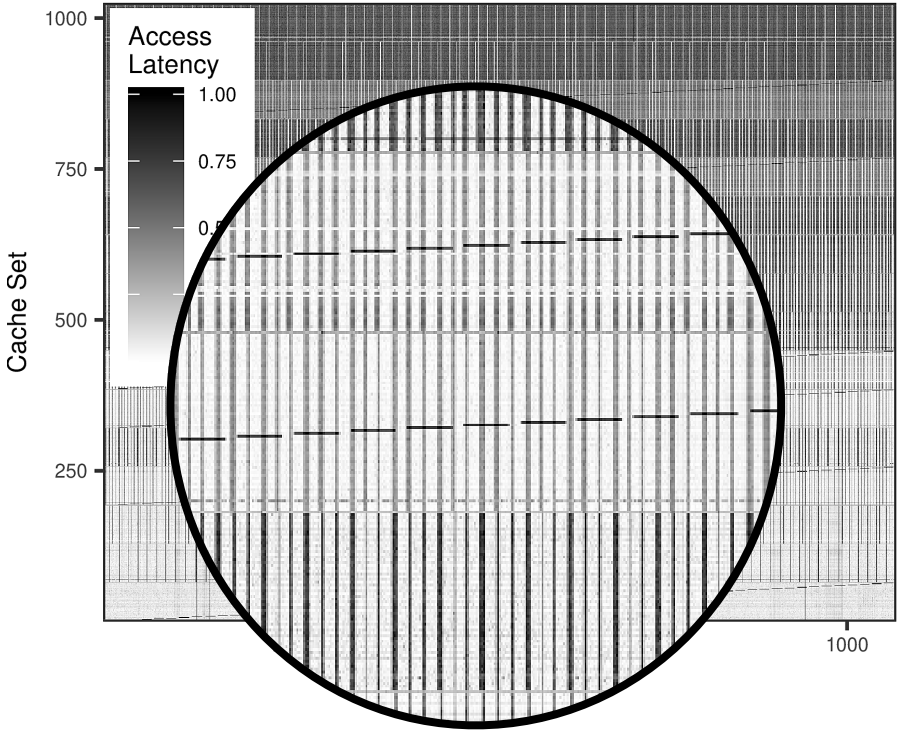
Evaluation: security

- Privileged cache SCA
  - Target: L2 cache
- No eviction



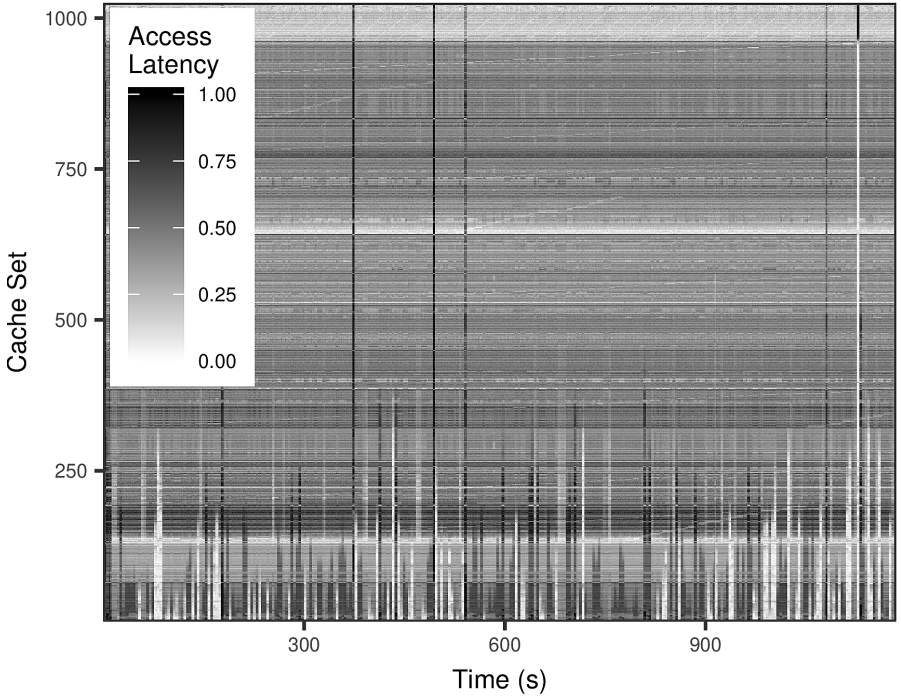
Evaluation: security

- Privileged cache SCA
  - Target: L2 cache
- No eviction



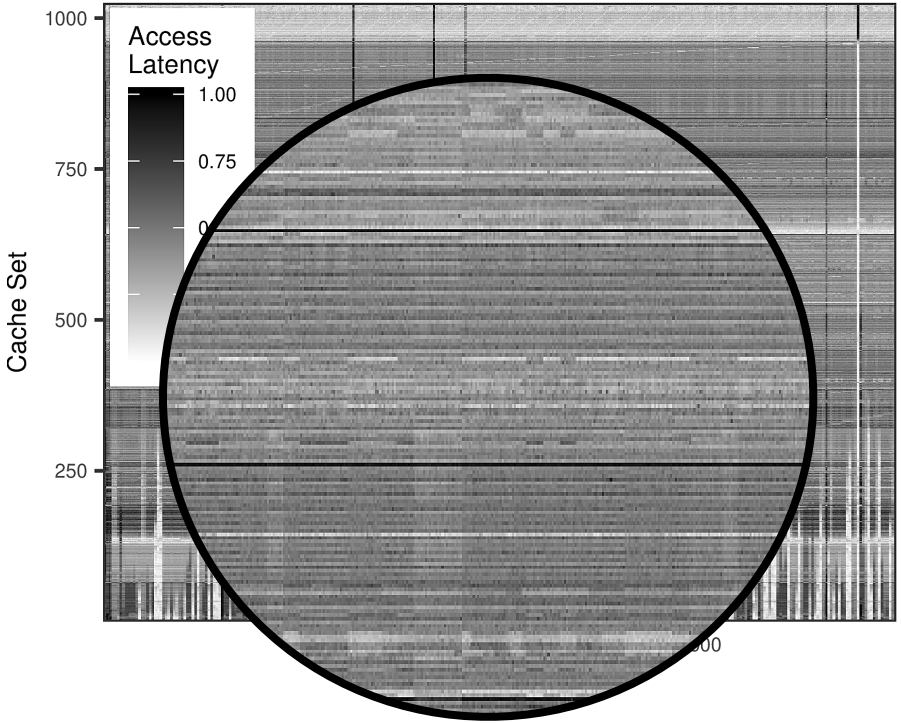
Evaluation: security

- Privileged cache SCA
  - Target: L2 cache
- Varys protection



Evaluation: security

- Privileged cache SCA
  - Target: L2 cache
- Varys protection



Summary

- Varys: side-channel protection for SGX enclaves
- "Rely but verify" approach
  - Ask OS for
    - Lower interrupt rate
    - Paired thread allocation
  - Verify the request
- Evict caches on enclave exits

Summary

- Varys: side-channel protection for SGX enclaves
- "Rely but verify" approach
  - Ask OS for
    - Lower interrupt rate
    - Paired thread allocation
  - Verify the request
- Evict caches on enclave exits

Thanks!

oleksii.oleksenko@tu-dresden.de

@oleksii_o