Flush+Reload

Verdeckte Kanäle mit cachebasierten Seitenkanalangriffen

Bundesamt
für Sicherheit in der
Informationstechnik

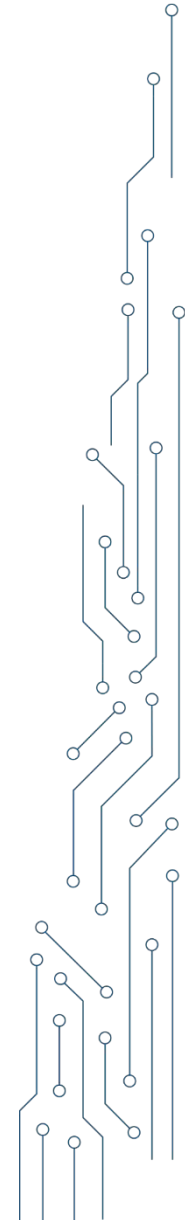
Ziele

- Verständnis von F+R
- Nachvollziehen der Timings
- Kommunikation über F+R

Verdeckte Kanäle mit cachebasierten Seitenkanalangriffen

Bitte was?

- **Verdeckte Kanäle**
- **Caches**
- **Seitenkanalangriffe**



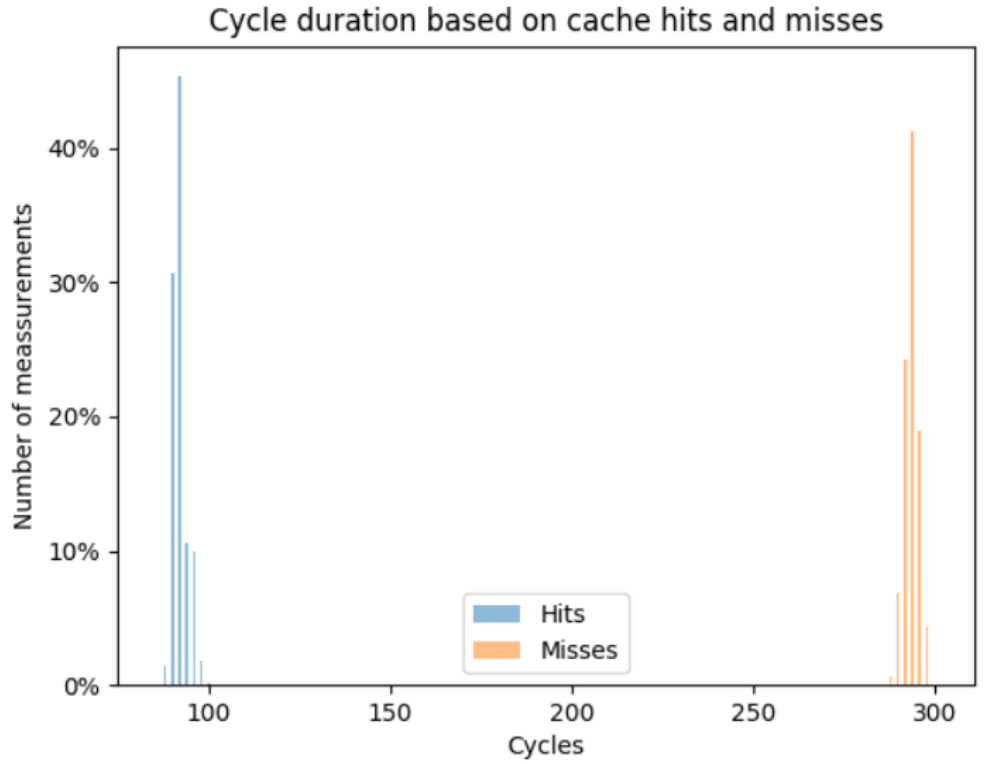
Cache-Angriffe

Seitenkanaleffekte von Caches

- **Cache wird von unabhängigen Prozessen verwendet**
- **Speicherisolation wird umgangen**
- **Seitenkanalinformationen über die Verwendung von Speicheradressen**
- **Shared-Memory vs Prime+Probe**

Reload Seitenkanaleffekt

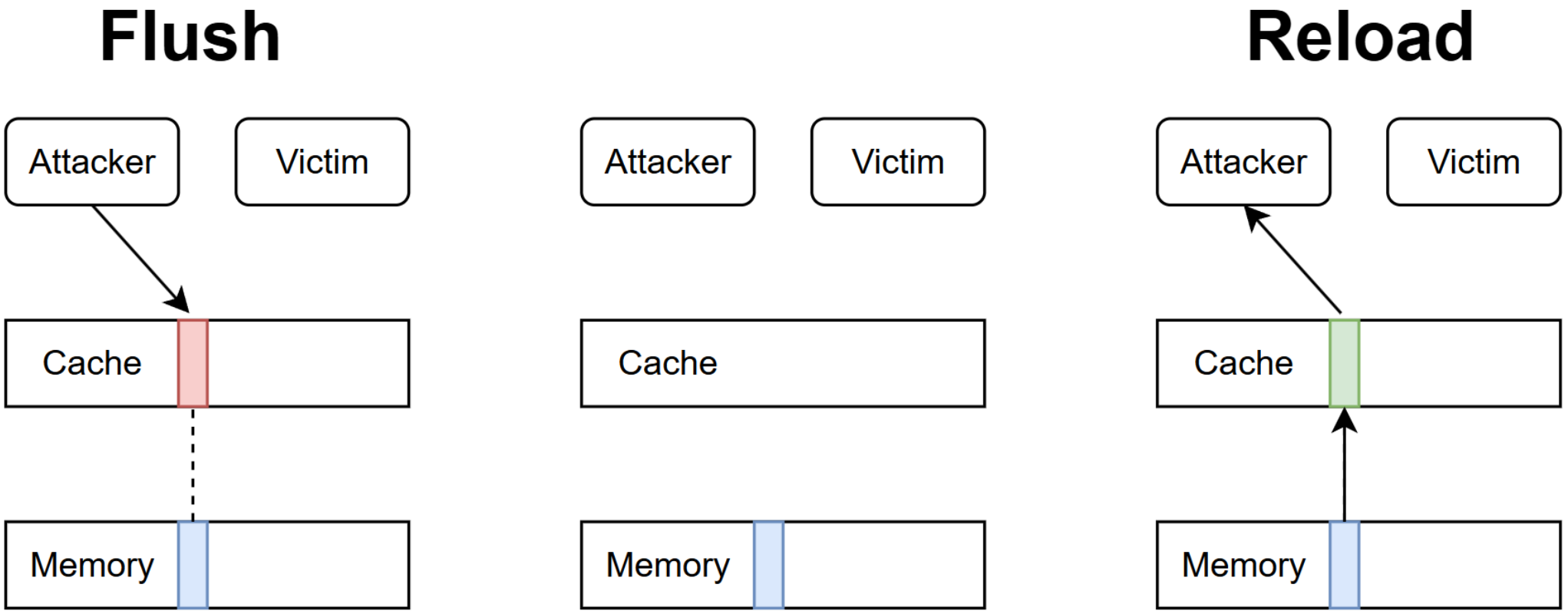
Informationen aus Timingdifferenz



Flush+Reload

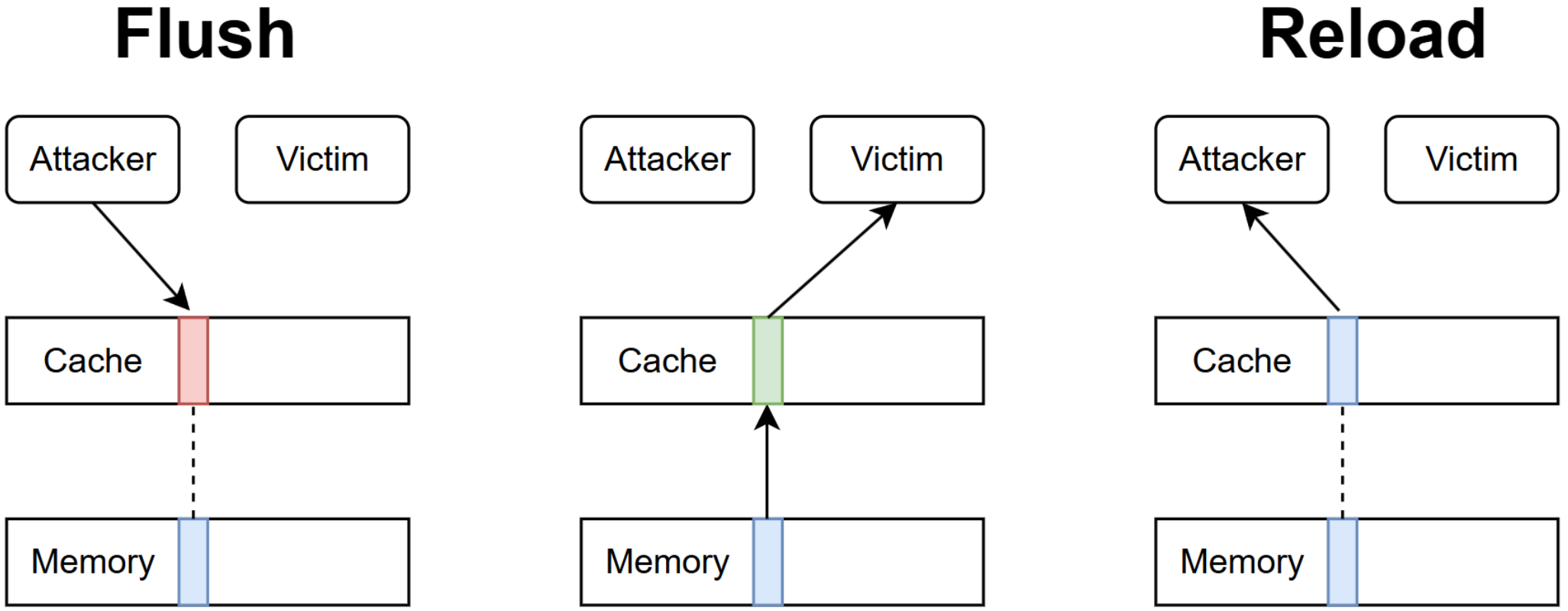
Flush+Reload

Kein Speicherzugriff des Opfers



Flush+Reload

Speicherzugriff des Opfers



Konstruktion eines Kanals

Bild: © AdobeStock/Azazul

Timings

Messung von Cache Hits und Cache Misses

- **RDTSC**
- **Memory Fence**

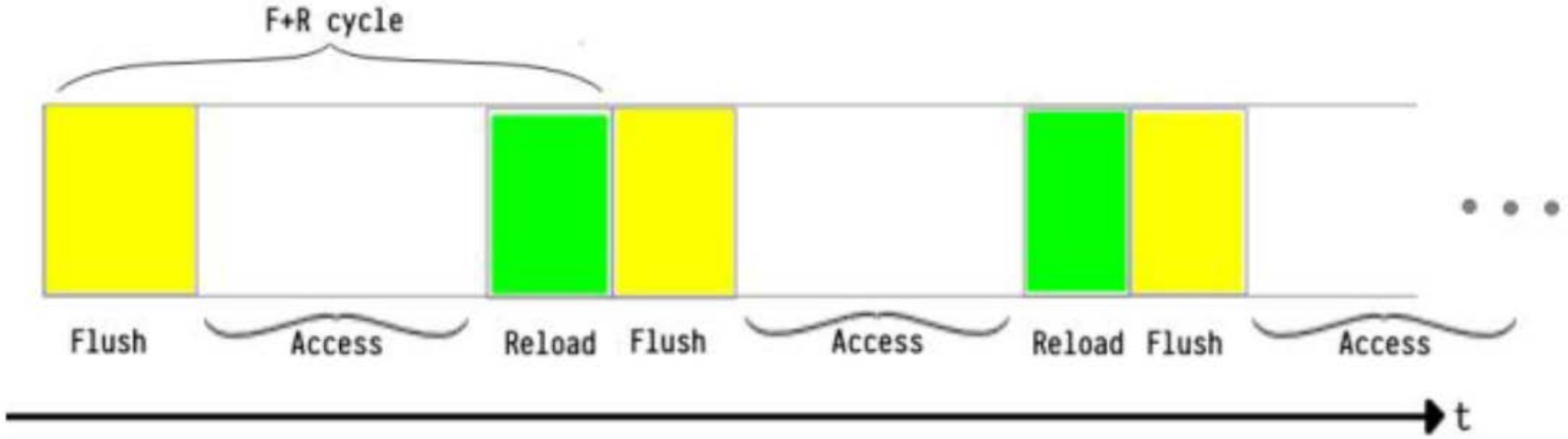
```
size_t time_maccess(void (* addr)(void))
{
    uint64_t start, end, delta;
    uint64_t lo, hi;
    asm volatile ("LFENCE");
    asm volatile ("RDTSC": "=a" (lo), "=d" (hi));
    start = (hi<<32) | lo;
    asm volatile ("LFENCE");

    asm volatile ("movq (%0), %%rax\n"
    :
    : "c" (addr)
    : "rax");

    asm volatile ("LFENCE");
    asm volatile ("RDTSC": "=a" (lo), "=d" (hi));
    end = (hi<<32) | lo;
    asm volatile ("LFENCE");
    delta = end - start;
    return delta;
}
```

Zyklus von F+R

Zeitlicher Ablauf



Synchronisierung

Mögliche Varianten

- **sleep**
- **nanosleep**
  - ABSTIME
- **ualarm**

Probleme

Noise

- **Verdrängung**
- **3rd Party Loads**

Probleme

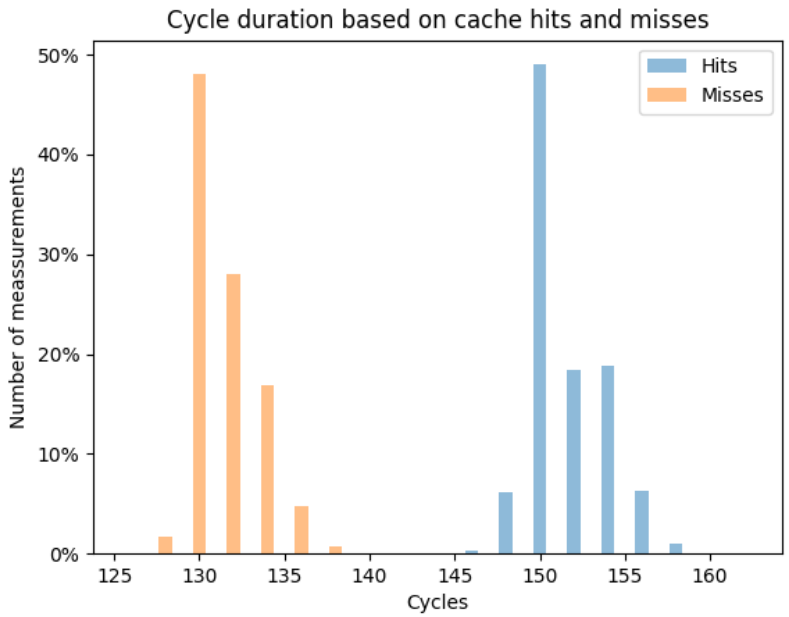
Abhilfe: Privater geteilter Speicher

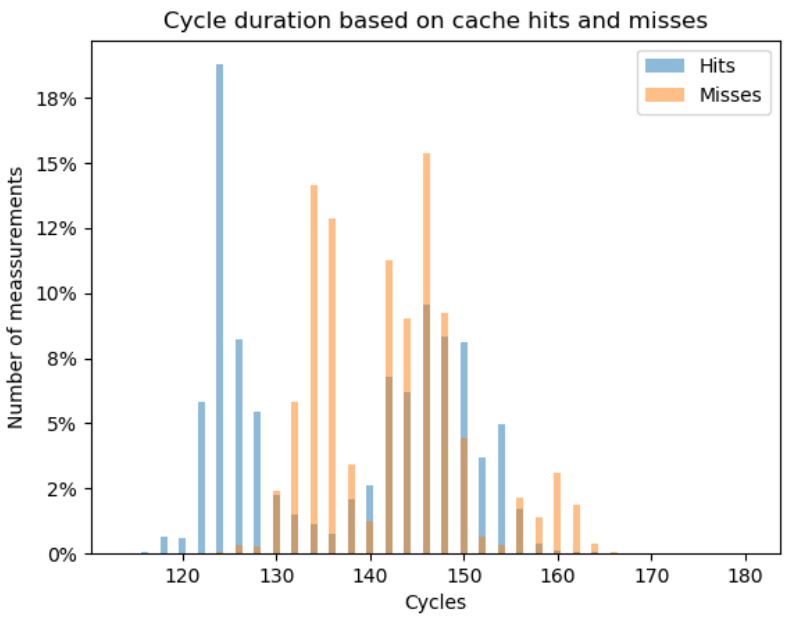
- **sharedlib.so**
- **Padding**

```
void myStupidNOPFunction(void)
{    asm volatile (
        "nop\n\t"
        "nop\n\t"
        "nop\n\t"
        "nop\n\t"
        "nop\n\t"
        "nop\n\t"
        "nop\n\t"
        "nop\n\t"
        "nop\n\t"
        "nop\n\t"
        "nop\n\t"
        "nop\n\t"
        ...
```

Probleme

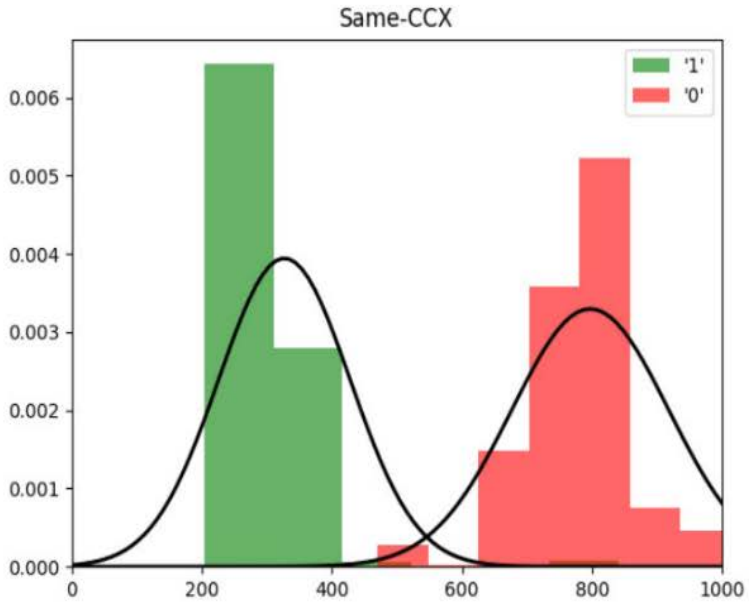
Unterschiedliche Systeme

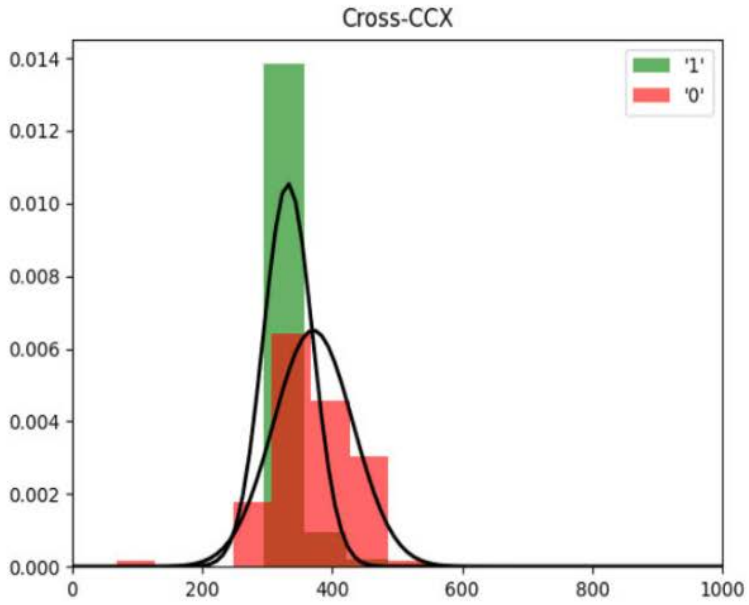




Probleme

Unterschiedliche Kerne

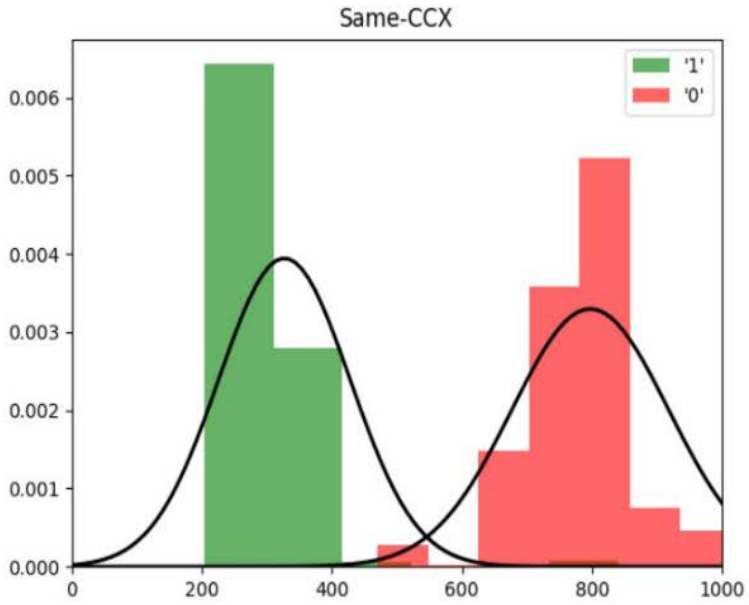


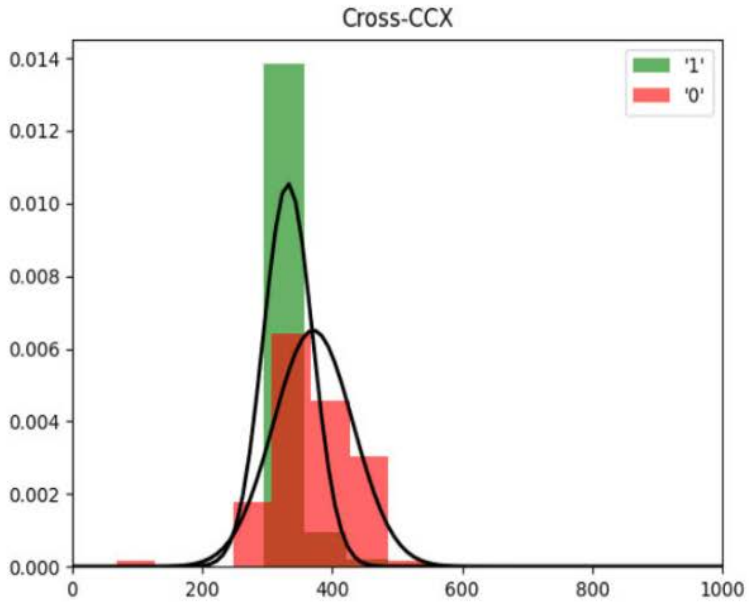


Probleme

Unterschiedliche Kerne

→ **taskset**





Demo

Vielen Dank für Ihre Aufmerksamkeit!

Noch Fragen?

Bundesamt
für Sicherheit in der
Informationstechnik

Follow us:

Bild: © AdobeStock/Nirut

Caches