



Sicherheit von Prozessoren

Wie Seiteneffekte Geheimnisse Verraten

Michael Schwarz (@misc0110)

21.05.2022



Michael Schwarz

Faculty @ CISPA

Fokus auf Seitenkanal Angriffe

 @misc0110

 michael.schwarz@cispa.de

Seitenkanäle



- Systeme haben **verschiedene Kommunikationskanäle**



- Systeme haben **verschiedene Kommunikationskanäle**
→ Schnittstellen, Netzwerke, Dateien, Datenbanken, ...



- Systeme haben **verschiedene Kommunikationskanäle**
- Schnittstellen, Netzwerke, Dateien, Datenbanken, ...
- Es gibt auch **unbeabsichtigte Kanäle...**

Was ist ein Seitenkanal?



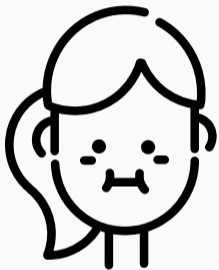
- Unbeabsichtigte Information über (geheime) Daten



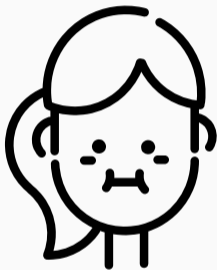
- Unbeabsichtigte Information über (geheime) Daten
- Verraten **Metadaten** über die Geheimnisse



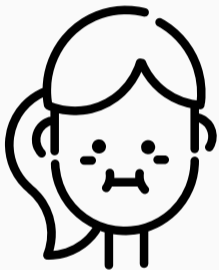
- Unbeabsichtigte Information über (geheime) Daten
- Verraten **Metadaten** über die Geheimnisse
- **Seitenkanalangriff**: Geheimnisse aus den Metadaten schließen



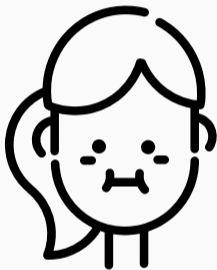
- Menschen **verraten** Seitenkanalinformationen



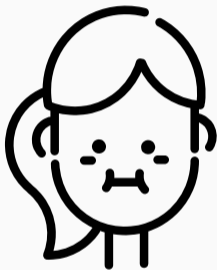
- Menschen **verraten** Seitenkanalinformationen
→ Gesichtsausdruck, Mikromimik



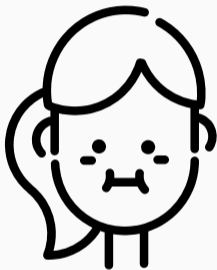
- Menschen **verraten** Seitenkanalinformationen
 - Gesichtsausdruck, Mikromimik
 - Gestik, Haltung



- Menschen **verraten** Seitenkanalinformationen
 - Gesichtsausdruck, Mikromimik
 - Gestik, Haltung
 - Atmung, Schwitzen



- Menschen **verraten** Seitenkanalinformationen
 - Gesichtsausdruck, Mikromimik
 - Gestik, Haltung
 - Atmung, Schwitzen
 - Artikulierung, Tonlage

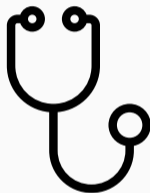


- Menschen **verraten** Seitenkanalinformationen
 - Gesichtsausdruck, Mikromimik
 - Gestik, Haltung
 - Atmung, Schwitzen
 - Artikulierung, Tonlage
- **Intuitiv** wahrnehmbar

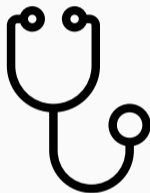


11 or 12 or 13 or 14 or 15 or 16 or 17 f
 21 w 22 s 23 v 24 in 25 in 26 s 27 g
 31 f 32 i 33 in 34 f 35 s 36 l 37 m
 41 w 42 o 43 s 44 p 45 q 46 x 47 f
 54 p 55 s 56 p 57 f
 67 s 65 m 66 q 67 s
 $\frac{2}{3} + \frac{3}{4} =$
 26743:8=
 12986 x 3 =

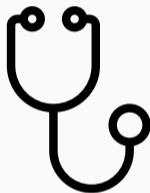




- Seitenkanäle gibt es auch in **Software**



- Seitenkanäle gibt es auch in **Software**
- Können auch für **Attacken** ausgenutzt werden



- Seitenkanäle gibt es auch in **Software**
- Können auch für **Attacken** ausgenutzt werden
- Statt **Geräusche** misst man meistens **Zeitunterschiede**



- Seitenkanäle nutzen **keine Fehler** aus



- Seitenkanäle nutzen **keine Fehler** aus
- Für diese Angriffe: **fehlerfreie** Software und Hardware

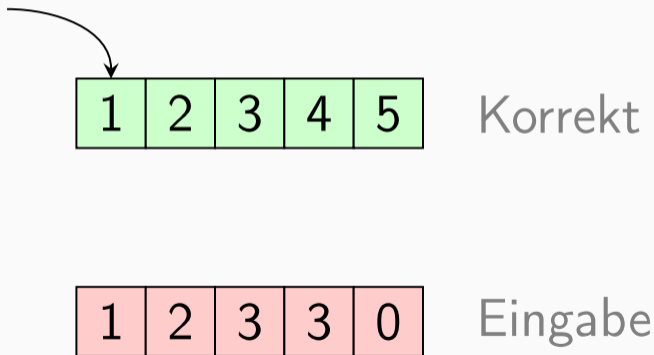


- Seitenkanäle nutzen **keine Fehler** aus
- Für diese Angriffe: **fehlerfreie** Software und Hardware
- Verwenden nur **Seiteneffekte** von Software oder Hardware

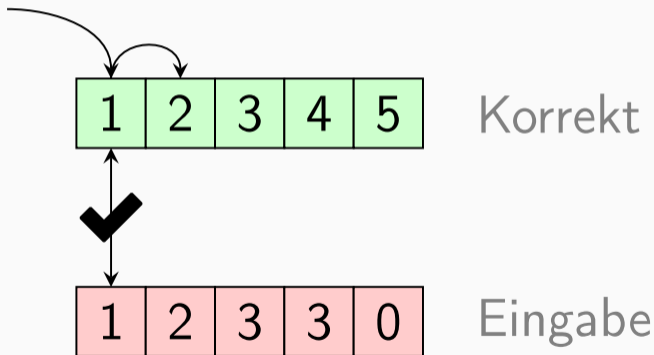


- Seitenkanäle nutzen **keine Fehler** aus
 - Für diese Angriffe: **fehlerfreie** Software und Hardware
 - Verwenden nur **Seiteneffekte** von Software oder Hardware
- Fehlerfreie Software bedeutet **nicht** sichere Software

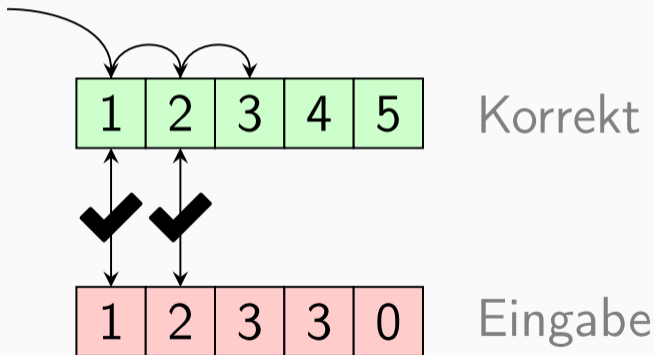
- Trivialer Ansatz:
 - Ziffer für Ziffer vergleichen
 - Sobald sich eine Ziffer unterscheidet, wird der Vergleich abgebrochen



- Trivialer Ansatz:
 - Ziffer für Ziffer vergleichen
 - Sobald sich eine Ziffer unterscheidet, wird der Vergleich abgebrochen

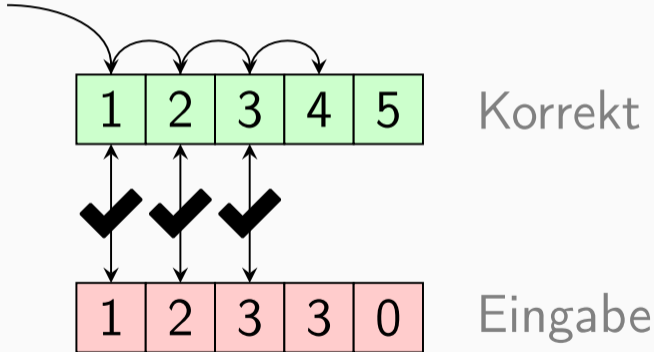


- Trivialer Ansatz:
 - Ziffer für Ziffer vergleichen
 - Sobald sich eine Ziffer unterscheidet, wird der Vergleich abgebrochen



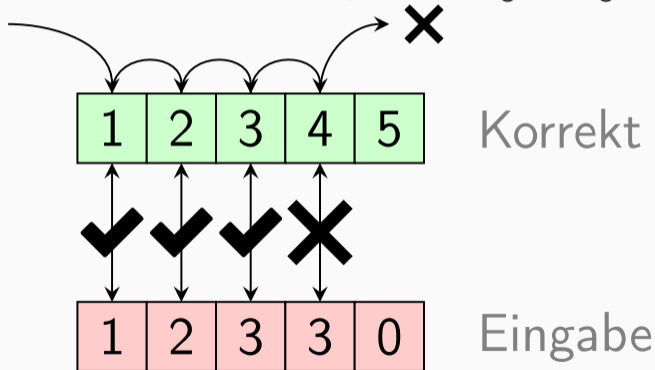
Beispiel: PIN Vergleich

- Trivialer Ansatz:
 - Ziffer für Ziffer vergleichen
 - Sobald sich eine Ziffer unterscheidet, wird der Vergleich abgebrochen

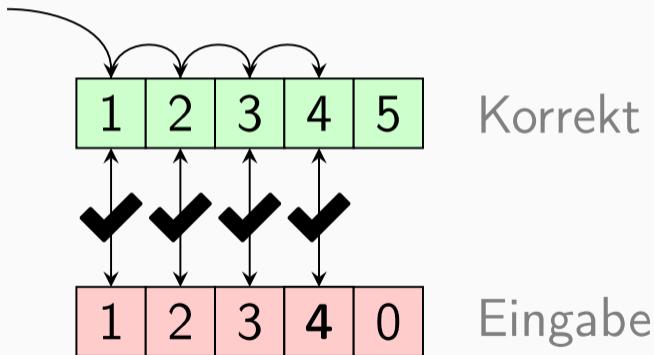


Beispiel: PIN Vergleich

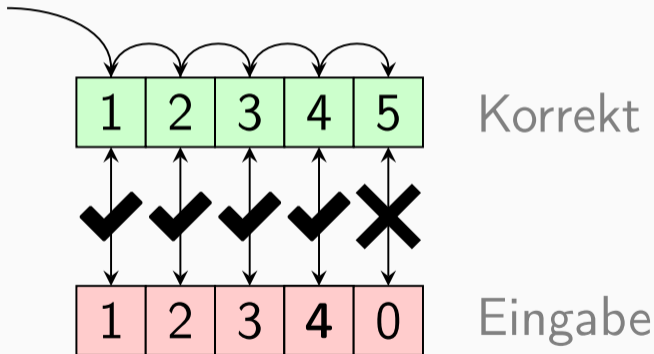
- Trivialer Ansatz:
 - Ziffer für Ziffer vergleichen
 - Sobald sich eine Ziffer unterscheidet, wird der Vergleich abgebrochen



- Trivialer Ansatz:
 - Ziffer für Ziffer vergleichen
 - Sobald sich eine Ziffer unterscheidet, wird der Vergleich abgebrochen

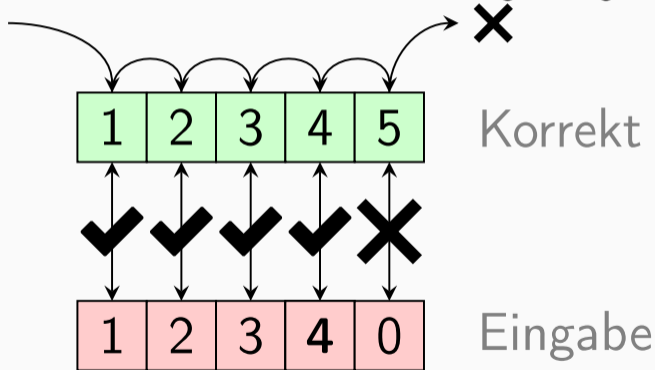


- Trivialer Ansatz:
 - Ziffer für Ziffer vergleichen
 - Sobald sich eine Ziffer unterscheidet, wird der Vergleich abgebrochen



Beispiel: PIN Vergleich

- Trivialer Ansatz:
 - Ziffer für Ziffer vergleichen
 - Sobald sich eine Ziffer unterscheidet, wird der Vergleich abgebrochen





- Messung der **Ausführungszeit** für verschiedene PINs

PIN Zeit




- Messung der **Ausführungszeit** für verschiedene PINs

PIN	Zeit
0000	





- Messung der **Ausführungszeit** für verschiedene PINs

PIN	Zeit
0000	
1000	





- Messung der **Ausführungszeit** für verschiedene PINs

PIN	Zeit
0000	
1000	
2000	





- Messung der **Ausführungszeit** für verschiedene PINs

PIN	Zeit
0000	
1000	
2000	
3000	

- Messung der **Ausführungszeit** für verschiedene PINs





PIN	Zeit
0000	
1000	
2000	
3000	
...	...

- Messung der **Ausführungszeit** für verschiedene PINs

PIN	Zeit
0000	
1000	
2000	
3000	
...	...

- Bei **korrekter** Ziffer wird die nächste Ziffer überprüft → **längere** Ausführungszeit

- Messung der **Ausführungszeit** für verschiedene PINs

PIN	Zeit
0000	
1000	
2000	
3000	
...	...

- Bei **korrekter** Ziffer wird die nächste Ziffer überprüft → **längere** Ausführungszeit
- Maximal 10 Versuche um die erste Stelle herauszufinden



- Messung der Ausführungszeit für verschiedene PINs

PIN Zeit

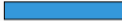


- Messung der Ausführungszeit für verschiedene PINs

PIN	Zeit
1000	





- Messung der Ausführungszeit für verschiedene PINs

PIN	Zeit
1000	
1100	





- Messung der Ausführungszeit für verschiedene PINs

PIN	Zeit
1000	
1100	
1200	





- Messung der Ausführungszeit für verschiedene PINs

PIN	Zeit
1000	
1100	
1200	
1300	

- Messung der Ausführungszeit für verschiedene PINs

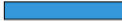



PIN	Zeit
1000	
1100	
1200	
1300	
...	...

- Messung der Ausführungszeit für verschiedene PINs

PIN	Zeit
1000	
1100	
1200	
1300	
...	...

- Für jede Ziffer wiederholen

- Messung der Ausführungszeit für verschiedene PINs

PIN	Zeit
1000	
1100	
1200	
1300	
...	...

- Für jede Ziffer wiederholen
- **Längste** Ausführungszeit verrät die Ziffer



- Für jede Ziffer maximal 10 Messungen



- Für jede Ziffer maximal 10 Messungen
- Bei 4 Stellen: 40 Versuche um PIN zu erraten



- Für jede Ziffer maximal 10 Messungen
- Bei 4 Stellen: 40 Versuche um PIN zu erraten
- Vergleich zu probieren: 10 000 Möglichkeiten



- Für jede Ziffer maximal 10 Messungen
- Bei 4 Stellen: 40 Versuche um PIN zu erraten
- Vergleich zu probieren: 10 000 Möglichkeiten
- Seitenkanal reduziert Anzahl Versuche um **Faktor 250**

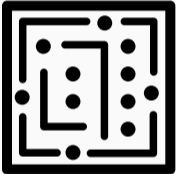
Mikroarchitekturelle Seitenkanäle



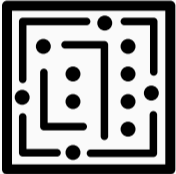




- Mikroarchitektur beschreibt die **interne** Art wie CPUs arbeiten

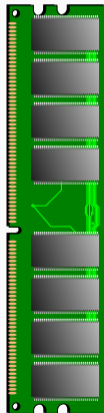


- Mikroarchitektur beschreibt die **interne** Art wie CPUs arbeiten
- Nicht sichtbar für Benutzer oder Programmierer



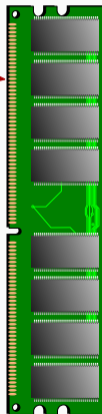
- Mikroarchitektur beschreibt die **interne** Art wie CPUs arbeiten
- Nicht sichtbar für Benutzer oder Programmierer
- Ist größtenteils **nicht dokumentiert** und kann nicht direkt beobachtet werden

```
val i = 42;  
print i;  
print i;
```



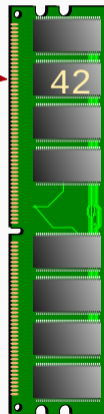
Beispiel: Daten im Speicher (Architektur)

```
val i = 42;  
print i;  
print i;
```



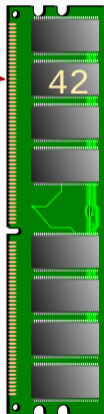
Beispiel: Daten im Speicher (Architektur)

```
val i = 42;  
print i;  
print i;
```



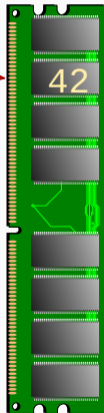
Beispiel: Daten im Speicher (Architektur)

```
val i = 42;  
print i;  
print i;
```



Beispiel: Daten im Speicher (Architektur)

```
val i = 42;  
print i;  
print i;
```











1337 4242

FOOD CACHE

Revolutionary concept!

Store your food at home,
never go to the grocery store
during cooking.

Can store **ALL** kinds of food.

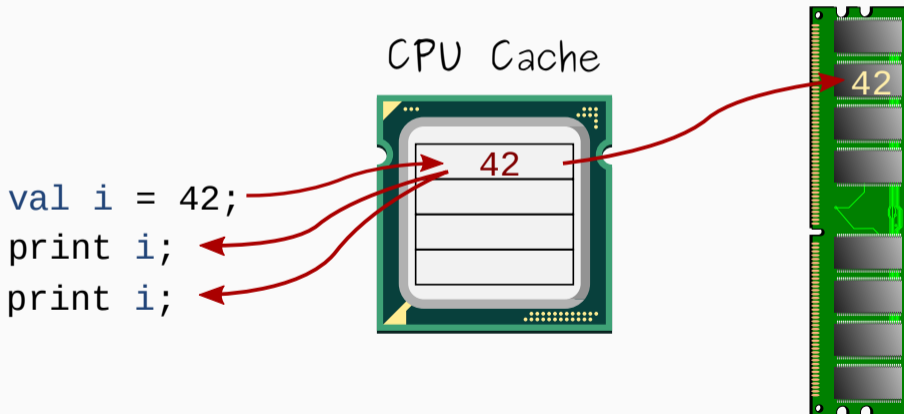
ONLY TODAY INSTEAD OF ~~\$1,300~~

\$1,299

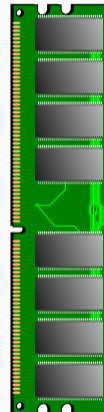
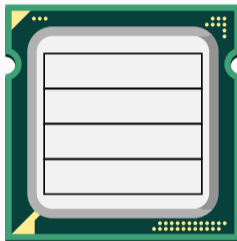
ORDER VIA PHONE: +555 12345



Beispiel: Daten im Speicher (Mikroarchitektur vereinfacht)

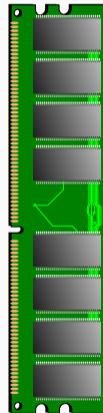
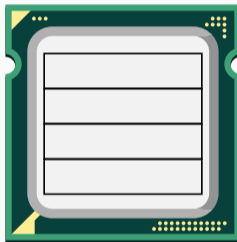



```
print i;  
print i;
```



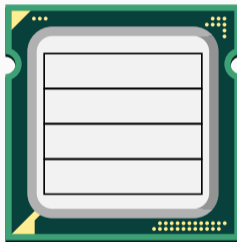
```
print i;  
print i;
```

Cache miss

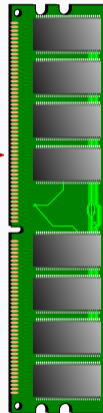


```
print i;  
print i;
```

Cache miss

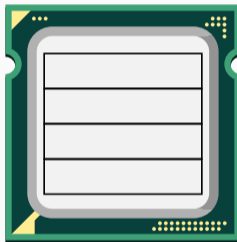


Request



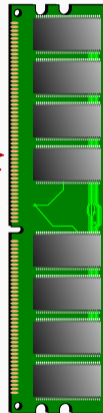
```
print i;  
print i;
```

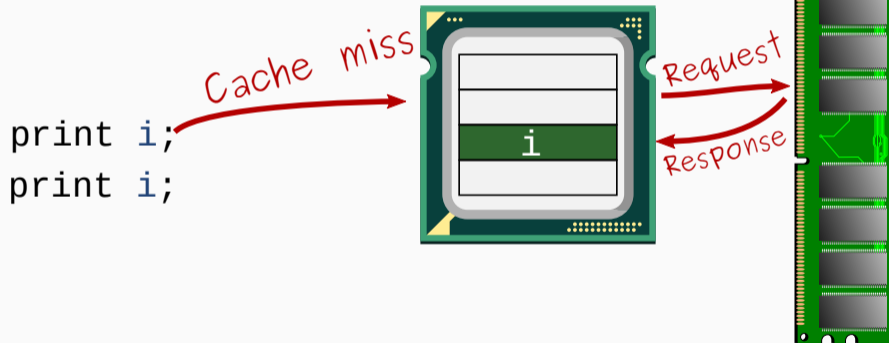
Cache miss

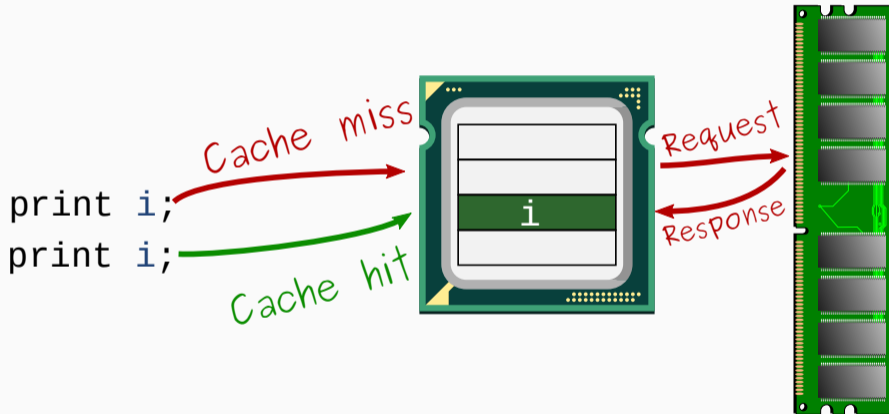


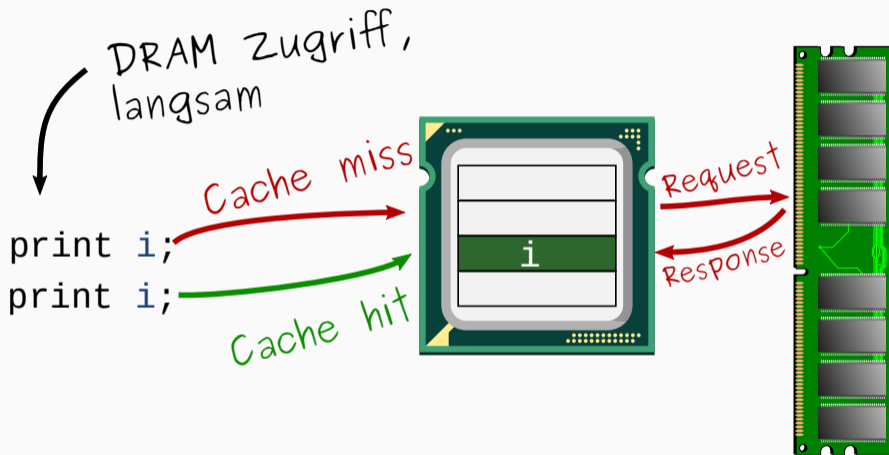
Request

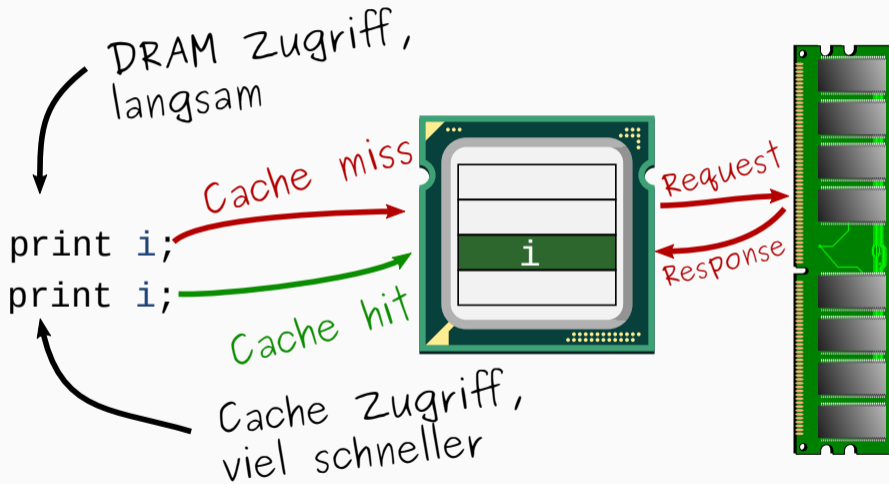
Response

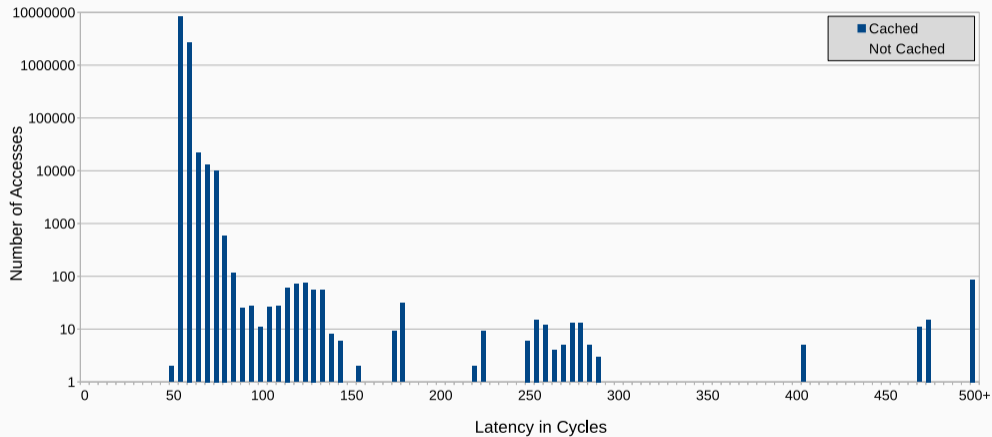


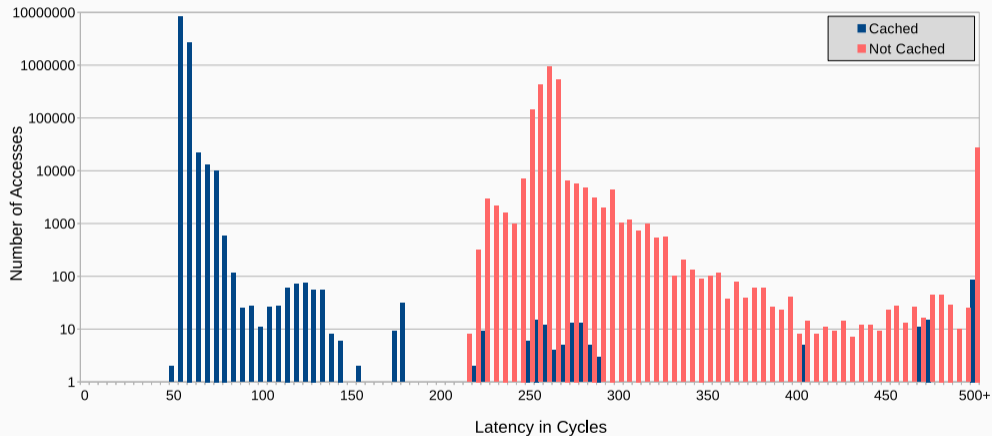














MELTDOWN

Ein Fehler in Prozessoren, so schwerwiegend, dass...





Ein Fehler in Prozessoren, so schwerwiegend, dass...

- er weltweit in den **Nachrichten** war

FOX
BUSINESS
WASHINGTON, D.C.

WASHINGTON, D.C.

**NEWS
ALERT**

**INTEL REVEALS DESIGN FLAW THAT
COULD ALLOW HACKERS TO ACCESS DATA**

WINTER STORM



FOX
BUSINESS
NETWORK



@FOXBUSINESS



DEVELOPING STORY

COMPUTER CHIP FLAWS IMPACT BILLIONS OF DEVICES

LIVE



DAX ▲ 164.69

NEWS STREAM





SECURITY FLAW REVEALED

Intel (Prev)
45.26 -1.59 [-3.39%]

Intel (After Hours)
44.85 -0.41 [-0.91%]

**CAPITAL
CONNECTION**

SHROUT: ISSUE NOT UNIQUE TO
INTEL, BUT IT'S AFFECTED THE MOST

CNBC



Ein Fehler in Prozessoren, so schwerwiegend, dass...

- er weltweit in den [Nachrichten](#) war
- es [Wikipedia](#) Artikel in mehreren Sprachen gibt



WIKIPEDIA
The Free Encyclopedia

[Main page](#)
[Contents](#)
[Featured content](#)
[Current events](#)
[Random article](#)
[Donate to Wikipedia](#)
[Wikipedia store](#)

[Interaction](#)

[Help](#)
[About Wikipedia](#)
[Community portal](#)
[Recent changes](#)
[Contact page](#)

[Tools](#)

[What links here](#)
[Related changes](#)
[Upload file](#)

Not logged in [Talk](#) [Contributions](#) [Create account](#) [Log in](#)

[Article](#) [Talk](#)

[Read](#) [Edit](#) [View history](#)

Meltdown (security vulnerability)

From Wikipedia, the free encyclopedia

Meltdown is a hardware [vulnerability](#) affecting [Intel x86 microprocessors](#) and some [ARM-based microprocessors](#).^{[1][2][3]} It allows a rogue process to read all [memory](#), even when it is not authorized to do so.

Meltdown affects a wide range of systems. At the time of disclosure, this included all devices running any but the most recent and [patched](#) versions of [iOS](#),^[4] [Linux](#)^{[5][6]}, [macOS](#),^[4] or [Windows](#). Accordingly, many servers and [cloud services](#) were impacted,^[7] as well as a potential majority of smart devices and [embedded devices](#) using ARM based processors (mobile devices, smart TVs and others), including a wide range of networking equipment. A purely software workaround to Meltdown has been assessed as slowing computers between 5 and 30 percent in certain specialized workloads,^[8] although companies responsible for software correction of the exploit are reporting minimal impact from general benchmark testing.^[9]

Meltdown was issued a [Common Vulnerabilities and Exposures](#) ID of [CVE-2017-5754](#)^[4], also known as *Rogue Data Cache Load*,^[2] in January 2018. It was disclosed in conjunction with another exploit, [Spectre](#), with which it shares some, but not all characteristics. The Meltdown and Spectre vulnerabilities are considered "catastrophic"



MELTDOWN

The logo used by the ^[5] team that discovered the vulnerability



WIKIPEDIA
The Free Encyclopedia

Main page
Contents
Featured content
Current events
Random article
Donate to Wikipedia
Wikipedia store

Interaction

Help
About Wikipedia
Community portal
Recent changes
Contact page

Tools

What links here
Related changes
Upload file

Not logged in Talk Contributions Create account Log in

Article Talk

Read Edit View history

Search Wikipedia

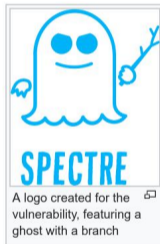
Spectre (security vulnerability)

From Wikipedia, the free encyclopedia

Spectre is a [vulnerability](#) that affects modern microprocessors that perform [branch prediction](#).^{[1][2][3]} On most processors, the [speculative execution](#) resulting from a branch misprediction may leave observable side effects that may reveal private data to attackers. For example, if the pattern of memory accesses performed by such speculative execution depends on private data, the resulting state of the data cache constitutes a [side channel](#) through which an attacker may be able to extract information about the private data using a [timing attack](#).^{[4][5][6]}

Two [Common Vulnerabilities and Exposures](#) IDs related to Spectre, [CVE-2017-5753](#)[ⓘ] (bounds check bypass) and [CVE-2017-5715](#)[ⓘ] (branch target injection), have been issued.^[7] [JIT engines](#) used for [JavaScript](#) were found vulnerable. A website can read data stored in the browser for another website, or the browser's memory itself.^[8]

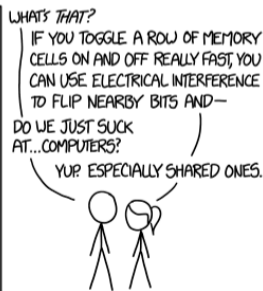
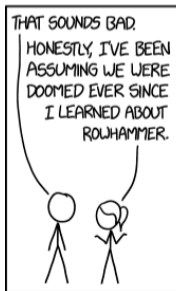
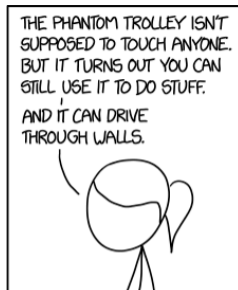
Several procedures to help protect home computers and related devices from the Spectre (and [Meltdown](#)) security vulnerabilities have been published.^{[9][10][11][12]} Spectre patches have been reported to significantly slow down performance, especially on older computers; on the newer 8th generation Core platforms, benchmark performance drops of 2–14 percent have been measured.^[13] Meltdown patches may also produce performance loss.^{[5][14][15]} On January 18, 2018, unwanted reboots, even for newer Intel chips, due to

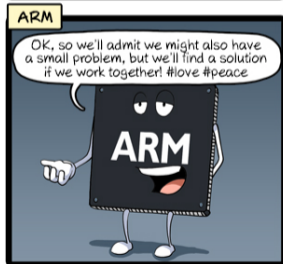
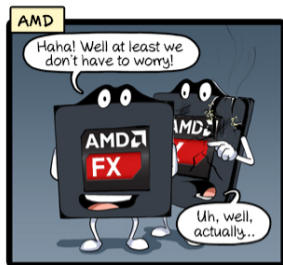
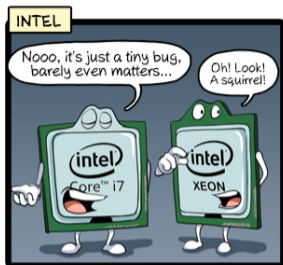




Ein Fehler in Prozessoren, so schwerwiegend, dass...

- er weltweit in den **Nachrichten** war
- es **Wikipedia** Artikel in mehreren Sprachen gibt
- es sogar **Comics** darüber gibt



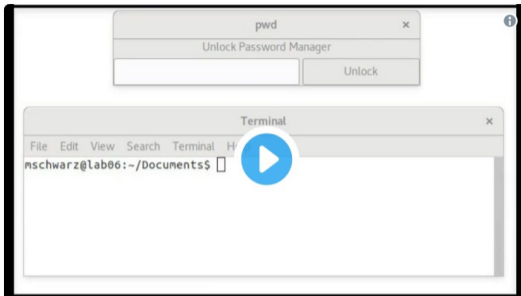


CommitStrip.com



Ein Fehler in Prozessoren, so schwerwiegend, dass...

- er weltweit in den [Nachrichten](#) war
- es [Wikipedia](#) Artikel in mehreren Sprachen gibt
- es sogar [Comics](#) darüber gibt
- sogar Snowden auf [Twitter](#) darüber schreibt



Edward Snowden ✓

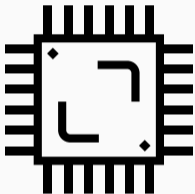
@Snowden



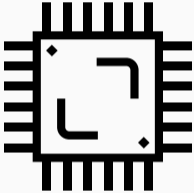
You may have heard about [@Intel](#)'s horrific [#Meltdown](#) bug. But have you watched it in action? When your computer asks you to apply updates this month, don't click "not now." (via [spectreattack.com](#) & [@misc0110](#))

23:37 - 4. Jan. 2018

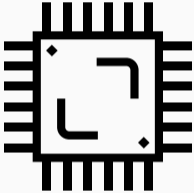
152 6.547 6.512



- Ein Fehler in Intel Prozessoren (und manchen ARM Prozessoren)

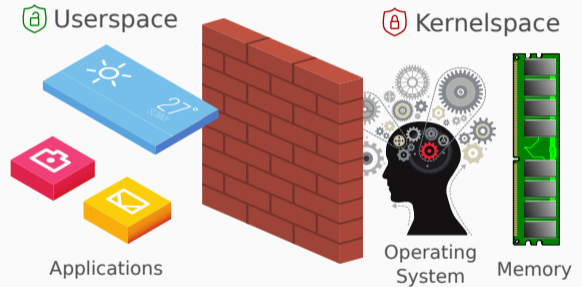


- Ein Fehler in Intel Prozessoren (und manchen ARM Prozessoren)
- Meltdown kann beliebigen Speicher auslesen

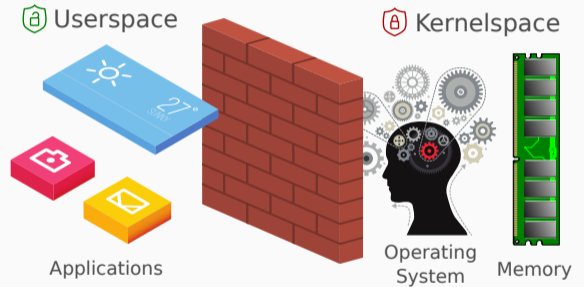


- Ein Fehler in Intel Prozessoren (und manchen ARM Prozessoren)
- Meltdown kann beliebigen **Speicher auslesen**
- Sowohl Speicher des Betriebssystems, als auch anderer Anwendungen

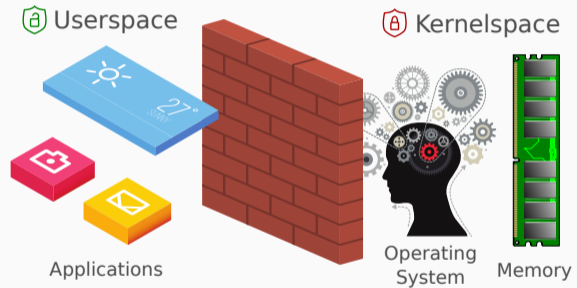
- **Isoliert** durch Software und Hardware



- **Isoliert** durch Software und Hardware
- Verwaltet Speicher



- **Isoliert** durch Software und Hardware
- Verwaltet Speicher
- Niemand kann direkt auf Betriebssystem zugreifen





- Speicher ist eine große Liste an Daten



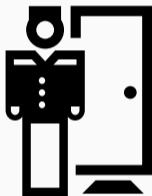
- Speicher ist eine große Liste an Daten
- CPU prüft, ob Zugriff auf Liste vom Betriebssystem kommt



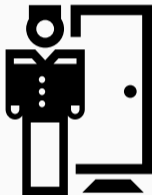
- Speicher ist eine große Liste an Daten
- CPU prüft, ob Zugriff auf Liste vom Betriebssystem kommt
- Wenn Zugriff von einer Anwendung kommt, wird diese beendet



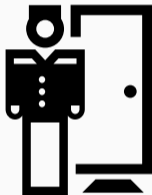
- Speicher ist eine große Liste an Daten
- CPU prüft, ob Zugriff auf Liste vom Betriebssystem kommt
- Wenn Zugriff von einer Anwendung kommt, wird diese beendet
- In Programmiersprachen wie C/C++ kann man dies probieren



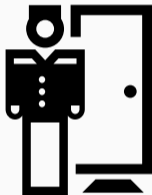
- Der Prozessor prüft, ob der Zugriff **erlaubt** ist



- Der Prozessor prüft, ob der Zugriff **erlaubt** ist
- Ja → Daten werden geladen



- Der Prozessor prüft, ob der Zugriff **erlaubt** ist
- Ja → Daten werden geladen
- Nein → Programm wird **beendet**



- Der Prozessor prüft, ob der Zugriff **erlaubt** ist
- Ja → Daten werden geladen
- Nein → Programm wird **beendet**
- Wirklich?

Out-of-order Ausführung

*6. Cook everything until
vegetables are soft*

*6. Add green to soup
and stir for 10 minutes*

*7. Serve with cooked
and peeled potatoes*





Eine Stunde warten



Eine Stunde warten



LATENZ

1. Wash and cut
vegetables

2. Pick the basil leaves
and set aside

3. Heat 2 tablespoons of
oil in a pan

4. Fry vegetables until
golden and softened



Abhängigkeit

1. Wash and cut vegetables

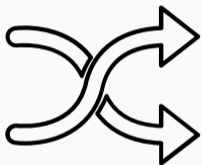
2. Pick the basil leaves and set aside

3. Heat 2 tablespoons of oil in a pan

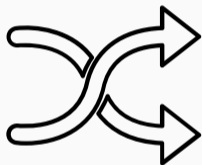
4. Fry vegetables until golden and softened

Parallelisierbar

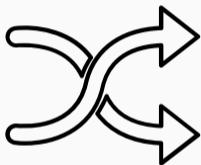




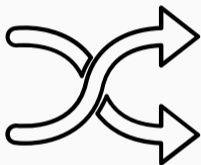
- Code darf in beliebiger Reihenfolge ausgeführt werden



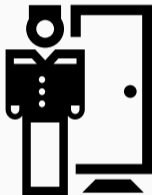
- Code darf in beliebiger Reihenfolge ausgeführt werden
- Ausführung muss jedoch immer korrekt sein



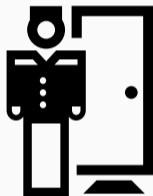
- Code darf in beliebiger Reihenfolge ausgeführt werden
- Ausführung muss jedoch immer korrekt sein
- Nicht nur der Fall auf Code-Ebene



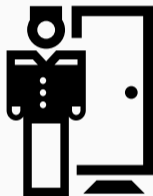
- Code darf in beliebiger Reihenfolge ausgeführt werden
- Ausführung muss jedoch immer korrekt sein
- Nicht nur der Fall auf Code-Ebene
- Einzelne Instruktionen können out-of-order ausgeführt werden



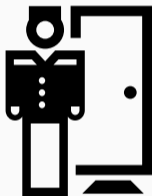
- Der Prozessor prüft, ob der Zugriff erlaubt ist



- Der Prozessor prüft, ob der Zugriff erlaubt ist
- Ja → Daten werden geladen



- Der Prozessor prüft, ob der Zugriff erlaubt ist
- Ja → Daten werden geladen
- Nein → Programm wird **beendet**



- Der Prozessor prüft, ob der Zugriff erlaubt ist
- Ja → Daten werden geladen
- Nein → Programm wird **beendet**
- Kann auch hier die **Reihenfolge umgedreht** werden?



- Zuerst laden, danach Berechtigung prüfen?



- Zuerst laden, danach Berechtigung prüfen?
- Wer hält das für eine gute Idee?



- Zuerst laden, danach Berechtigung prüfen?
- Wer hält das für eine gute Idee?
- Zumindest [Intel](#) hielt es für eine gute Idee



- Zuerst laden, danach Berechtigung prüfen?
- Wer hält das für eine gute Idee?
- Zumindest [Intel](#) hielt es für eine gute Idee
- Begründung: Programm wird danach so oder so beendet



CRIME SCENE DO NOT CROSS
CRIME SCENE DO NOT CROSS



- Annahme: Berechtigung wird **nach** dem Zugriff geprüft



- Annahme: Berechtigung wird **nach** dem Zugriff geprüft
- Resultat: Ein kleines Zeitfenster, in dem wir mit den Daten arbeiten können



- Annahme: Berechtigung wird **nach** dem Zugriff geprüft
- Resultat: Ein kleines Zeitfenster, in dem wir mit den Daten arbeiten können
- Alles Sichtbare (Variablen, Dateien, ...) wird **aufgeräumt**
- Daten müssen an einen "sicheren" Ort → Cache!



- Annahme: Berechtigung wird **nach** dem Zugriff geprüft
 - Resultat: Ein kleines Zeitfenster, in dem wir mit den Daten arbeiten können
 - Alles Sichtbare (Variablen, Dateien, ...) wird **aufgeräumt**
 - Daten müssen an einen "sicheren" Ort → Cache!
- **Zeitmessung** macht Cache "sichtbar"

SUPERMARKET



SUPERMARKET



SUPERMARKET

A B C D E F G

H I J K L N

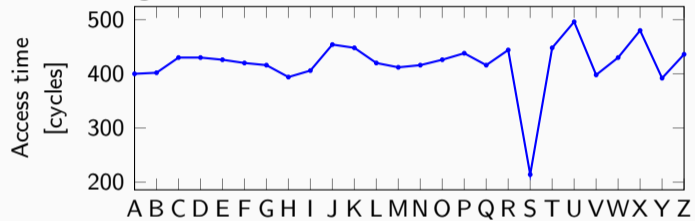
O P R S

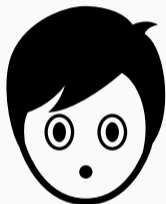
U V X Y Z



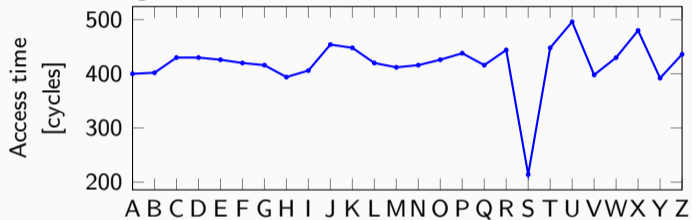


- Zeitmessung über alle Cache Bereich

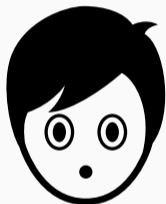




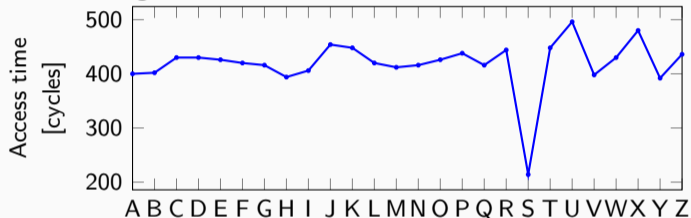
- Zeitmessung über alle Cache Bereich



- Zugriffener Bereich lädt **schneller...**

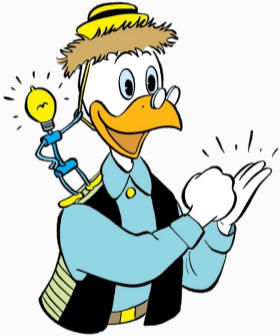


- Zeitmessung über alle Cache Bereich

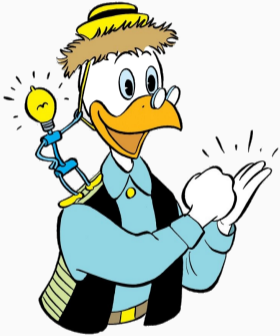


- Zugriffener Bereich lädt **schneller...**
 - ...und verrät damit den **Wert**
- Lesen von beliebigen Daten möglich!

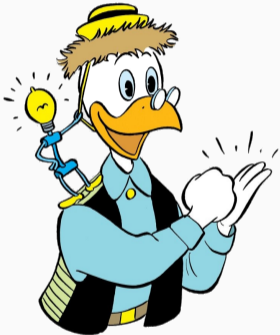
Und jetzt?...



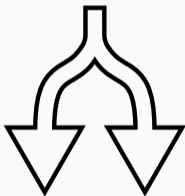
- Wir lassen das Betriebssystem **verschwinden**



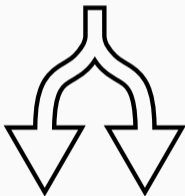
- Wir lassen das Betriebssystem **verschwinden**
- Geschützter Speicher ist unsichtbar



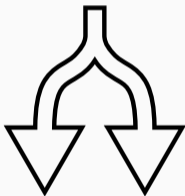
- Wir lassen das Betriebssystem **verschwinden**
- Geschützter Speicher ist unsichtbar
- Niemand kann auf geschützten Speicher zugreifen
- KAISER → Kernel Address Isolation to have Side channels Efficiently Removed



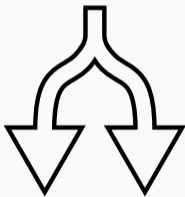
- Wir haben [KAISER](#) im Juli 2017 veröffentlicht



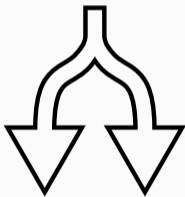
- Wir haben **KAISER** im Juli 2017 veröffentlicht
- Intel und andere haben es verbessert und in Linux als **KPTI** (Kernel Page Table Isolation) aufgenommen



- Wir haben **KAISER** im Juli 2017 veröffentlicht
- Intel und andere haben es verbessert und in Linux als **KPTI** (Kernel Page Table Isolation) aufgenommen
- Microsoft hat in Windows 10 etwas Ähnliches implementiert



- Wir haben **KAISER** im Juli 2017 veröffentlicht
- Intel und andere haben es verbessert und in Linux als **KPTI** (Kernel Page Table Isolation) aufgenommen
- Microsoft hat in Windows 10 etwas Ähnliches implementiert
- Apple in macOS 10.13.2 ebenfalls ("**Double Map**")



- Wir haben **KAISER** im Juli 2017 veröffentlicht
- Intel und andere haben es verbessert und in Linux als **KPTI** (Kernel Page Table Isolation) aufgenommen
- Microsoft hat in Windows 10 etwas Ähnliches implementiert
- Apple in macOS 10.13.2 ebenfalls (“**Double Map**”)
- Grundidee ist immer gleich: das Betriebssystem verstecken



- Software Seitenkanäle gibt es seit vielen Jahren



- Software Seitenkanäle gibt es seit vielen Jahren
- Waren nie “interessant”



- Software Seitenkanäle gibt es seit vielen Jahren
- Waren nie “interessant”
- Erst als wir gezeigt haben, dass man damit Daten lesen kann
(→ Meltdown, Spectre)



- Software Seitenkanäle gibt es seit vielen Jahren
- Waren nie “interessant”
- Erst als wir gezeigt haben, dass man damit Daten lesen kann (→ Meltdown, Spectre)
- Optimierungen führen neue Seitenkanäle ein



Die Entdeckung gibt uns die Chance

- neue Prozessorarchitekturen zu entwickeln
- mehr an Sicherheit zu denken
- Kompromisse zwischen Sicherheit und Performance zu finden



- Seitenkanal-Angriffe wurden zu lange unterschätzt
 - Grundlegende Konzepte gibt es schon lange (Zeitmessung)
- Es wird zu wenig Wert auf Sicherheit gelegt
 - Wir brauchen mehr Fokus auf Sicherheit
 - Performance darf nicht mehr das einzige Kriterium bei Prozessoren sein
- Noch viel Forschung notwendig

Fragen





Sicherheit von Prozessoren

Wie Seiteneffekte Geheimnisse Verraten

Michael Schwarz (@misc0110)

21.05.2022