Kernel Samepage Merging (KSM)

Funktionsweise und Schwachstellen

Martin Heckel

13. Dezember 2021

Hochschule für angewandte Wissenschaften Hof

Übersicht

Grundlagen

Virtueller Speicher

- **Herausforderung**: Rechner verwenden verschiedene Arten von Speicher:
  - DRAM
  - „Auslagerungsspeicher“ (Swap) in Datei oder Partition (Sekundärspeicher)

Virtueller Speicher

- **Herausforderung**: Rechner verwenden verschiedene Arten von Speicher:
  - DRAM
  - „Auslagerungsspeicher“ (Swap) in Datei oder Partition (Sekundärspeicher)
- **Herausforderung**: Mehrere Prozesse gleichzeitig auf einem System, voneinander getrennt

Virtueller Speicher

- **Herausforderung**: Rechner verwenden verschiedene Arten von Speicher:
  - DRAM
  - „Auslagerungsspeicher“ (Swap) in Datei oder Partition (Sekundärspeicher)
- **Herausforderung**: Mehrere Prozesse gleichzeitig auf einem System, voneinander getrennt
- **Idee**: Jeder Prozess hat eigenen Adressraum, physischer Speicher wird in diesen Adressraum gemappt

Virtueller Speicher

- **Herausforderung**: Rechner verwenden verschiedene Arten von Speicher:
  - DRAM
  - „Auslagerungsspeicher“ (Swap) in Datei oder Partition (Sekundärspeicher)
- **Herausforderung**: Mehrere Prozesse gleichzeitig auf einem System, voneinander getrennt
- **Idee**: Jeder Prozess hat eigenen Adressraum, physischer Speicher wird in diesen Adressraum gemappt
- Um Zugriffe zu beschleunigen, werden Bereiche von 4 KiB Gemappt („Pages“)

Page Tables

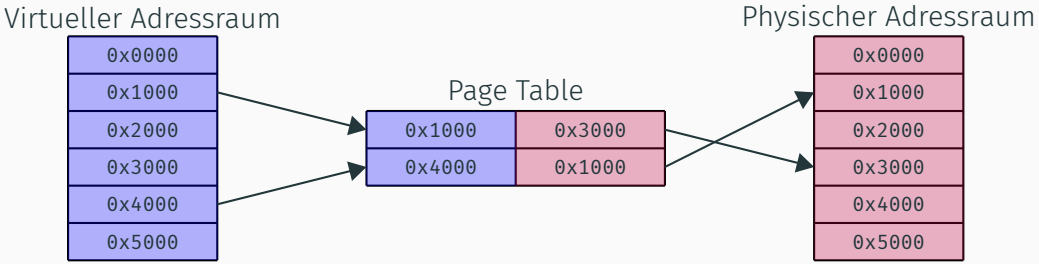


Abbildung basiert auf dem Foliensatz „0x0D Low-Level Fundamentals“ von Prof. Dr. Florian Adamsky

Physischer Aufbau von DRAM

DRAM Zelle

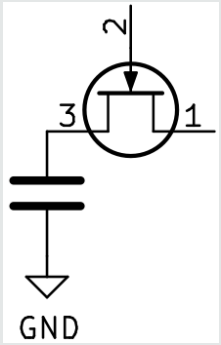


Abbildung aus den Slides zu [3]

Physischer Aufbau von DRAM

DRAM Array

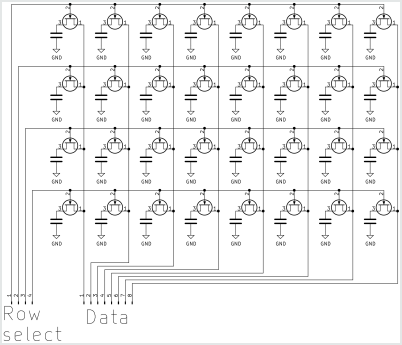


Abbildung aus den Slides zu [3]

Physischer Aufbau von DRAM

Innerhalb einer Bank

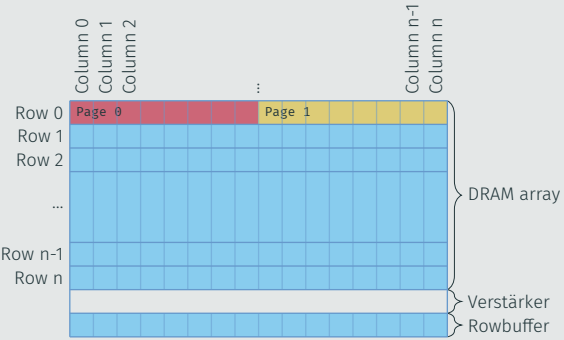


Abbildung aus den Slides zu [3]

Physischer Aufbau von DRAM

Bank

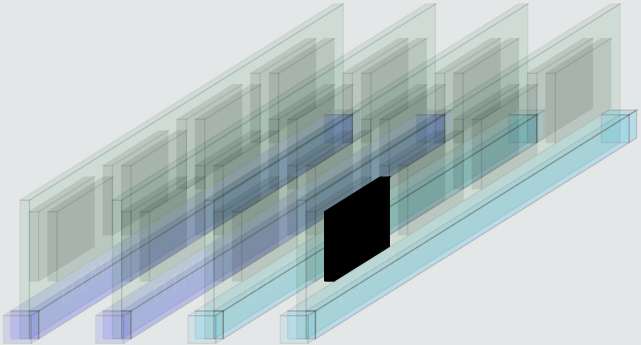


Abbildung aus den Slides zu [3]

Physischer Aufbau von DRAM

Rank

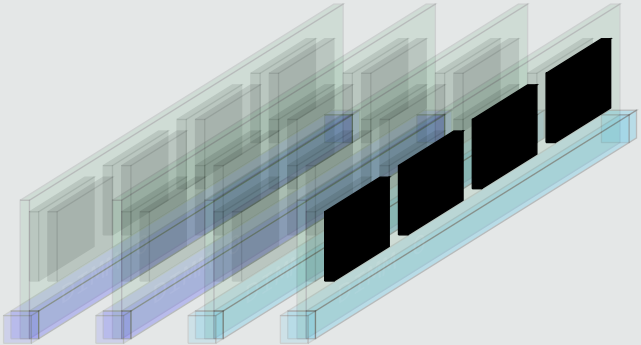


Abbildung aus den Slides zu [3]

Physischer Aufbau von DRAM

DIMM

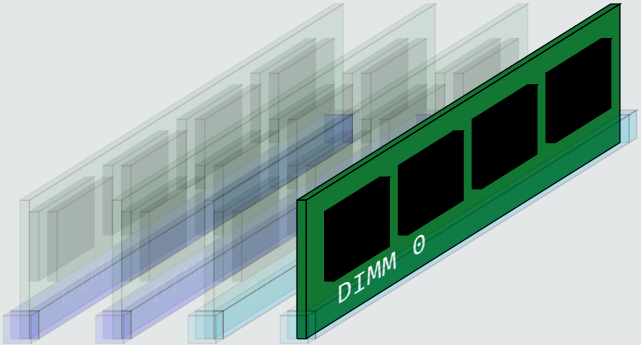


Abbildung aus den Slides zu [3]

Physischer Aufbau von DRAM

Channel

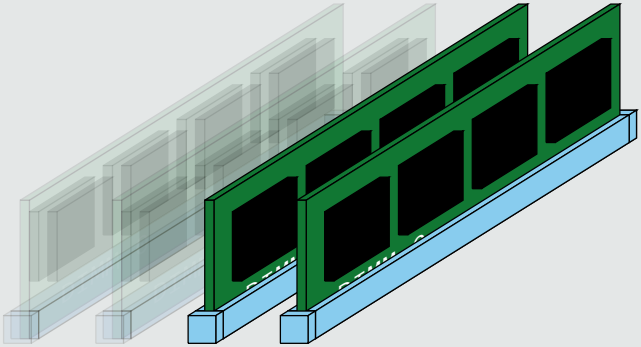


Abbildung aus den Slides zu [3]

Physischer Aufbau von DRAM

DRAM des Systems

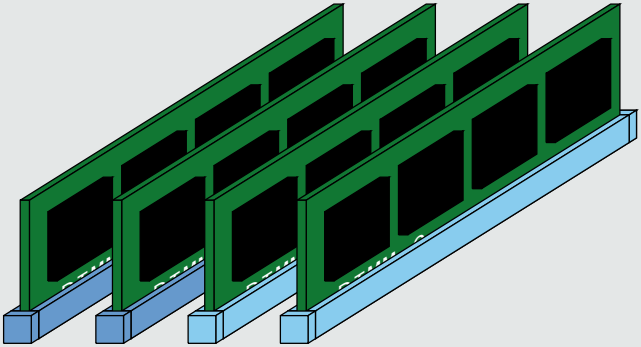


Abbildung aus den Slides zu [3]

Physischer Aufbau von DRAM

- Der Ladungszustand eines Kondensators wird einem logischen Zustand („1“ oder „0“) zugeordnet

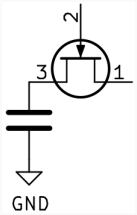


Abbildung aus den Slides zu [3]

Physischer Aufbau von DRAM

- Der Ladungszustand eines Kondensators wird einem logischen Zustand („1“ oder „0“) zugeordnet
- Kondensatoren entladen sich über die Zeit („Leckstrom“)

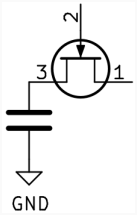


Abbildung aus den Slides zu [3]

Physischer Aufbau von DRAM

- Der Ladungszustand eines Kondensators wird einem logischen Zustand („1“ oder „0“) zugeordnet
- Kondensatoren entladen sich über die Zeit („Leckstrom“)
- **Herausforderung**: Aufladen eines vollen Kondensators bevor dieser als „leer“ interpretiert wird

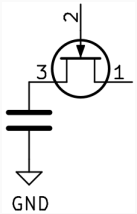


Abbildung aus den Slides zu [3]

Physischer Aufbau von DRAM

- Der Ladungszustand eines Kondensators wird einem logischen Zustand („1“ oder „0“) zugeordnet
- Kondensatoren entladen sich über die Zeit („Leckstrom“)
- **Herausforderung**: Aufladen eines vollen Kondensators bevor dieser als „leer“ interpretiert wird
- **Lösung**: Zyklisches Lesen und zurück Schreiben von Rows (typ. alle 64 ms), **Refresh**

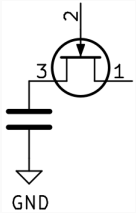
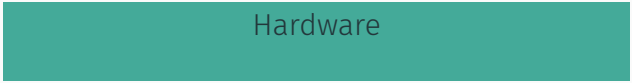


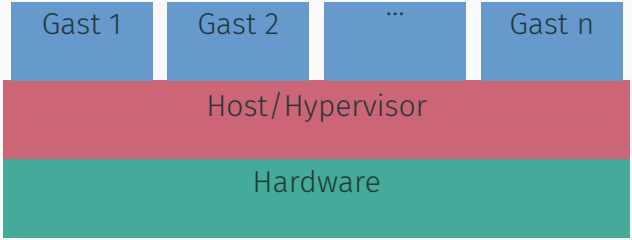
Abbildung aus den Slides zu [3]

Virtualisierung



Virtualisierung

Virtualisierung



Speicherverwaltung in virtualisierten Umgebungen

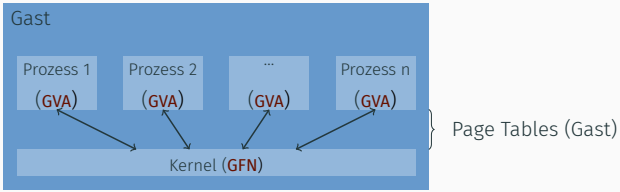


Abbildung aus den Slides zu [3]

Speicherverwaltung in virtualisierten Umgebungen

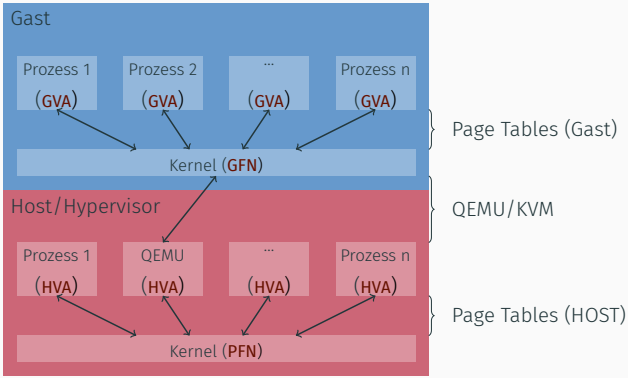


Abbildung aus den Slides zu [3]

Speicherverwaltung in virtualisierten Umgebungen

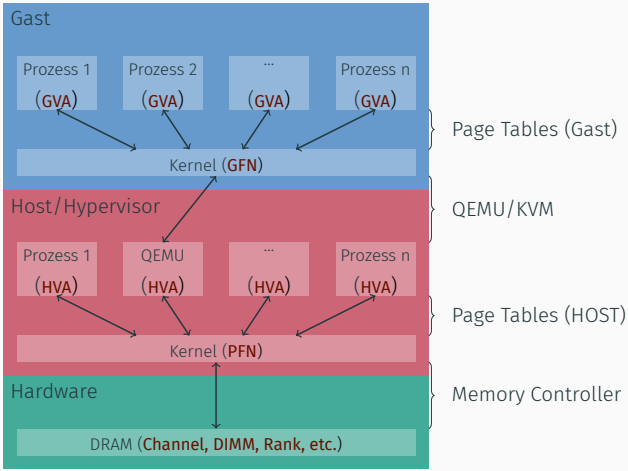


Abbildung aus den Slides zu [3]

Demo 1

Analyse virtueller Adressierung

Kernel Samepage Merging (KSM)

- Einige Speicherbereiche werden normalerweise nur lesend verwendet
- Einige Speicherbereiche haben identische Inhalte

Kernel Samepage Merging (KSM)

- Einige Speicherbereiche werden normalerweise nur lesend verwendet
- Einige Speicherbereiche haben identische Inhalte
- Anpassen der Page Tables, um mehrfach vorkommende Bereiche nur einmal im physischen Speicher zu halten (mergen von Speicherseiten mit identischen Inhalt)

Kernel Samepage Merging (KSM)

- Einige Speicherbereiche werden normalerweise nur lesend verwendet
- Einige Speicherbereiche haben identische Inhalte
- Anpassen der Page Tables, um mehrfach vorkommende Bereiche nur einmal im physischen Speicher zu halten (mergen von Speicherseiten mit identischen Inhalt)
- Copy-on-Write Policy, um Schreibzugriff auf deduplizierte Seiten zu verhindern

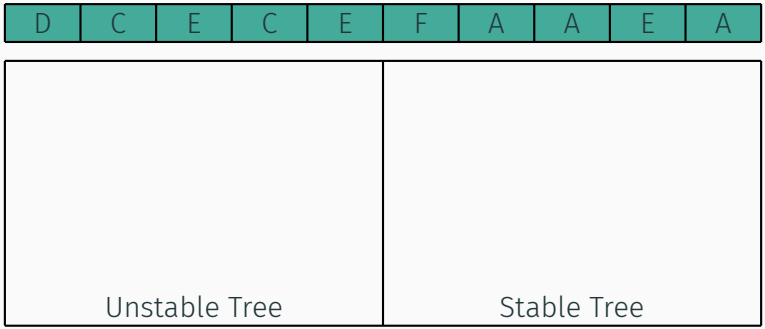
Kernel Samepage Merging (KSM)

- KSM berücksichtigt nur Seiten, die als *merge candidat* deklariert sind (s. MADVISE(2))

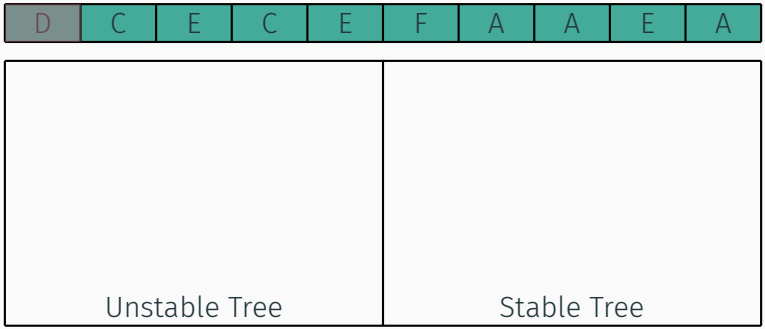
Kernel Samepage Merging (KSM)

- KSM berücksichtigt nur Seiten, die als *merge candidat* deklariert sind (s. MADVISE(2))
- Verwendung von zwei rot-schwarz Bäumen, um Seiten zu verwalten:
  - *Stable tree*: enthält Seiten, die bereits dedupliziert wurden
  - *Unstable tree*: enthält Seiten, die dedupliziert werden können (wenn eine zweite Seite mit identischen Inhalt gefunden wird)

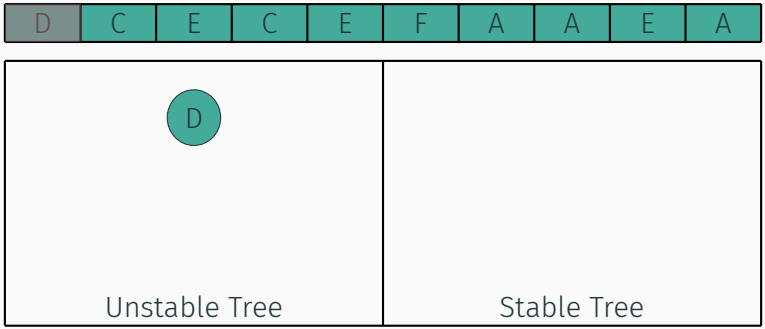
Kernel Samepage Merging (KSM)



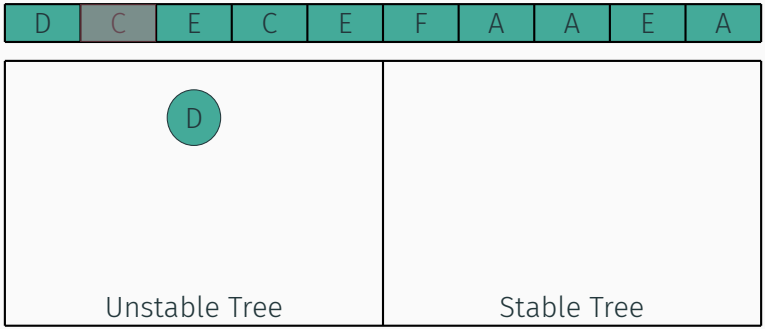
Kernel Samepage Merging (KSM)



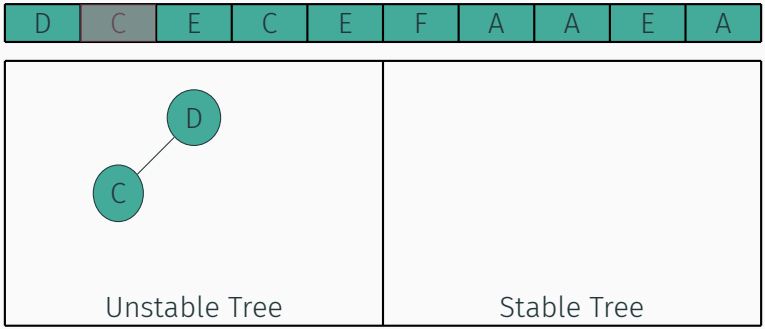
Kernel Samepage Merging (KSM)



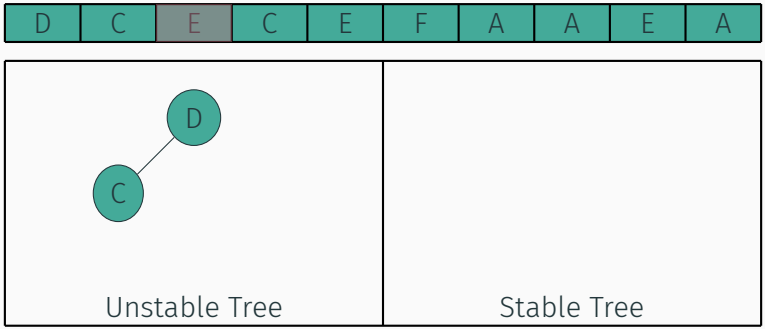
Kernel Samepage Merging (KSM)



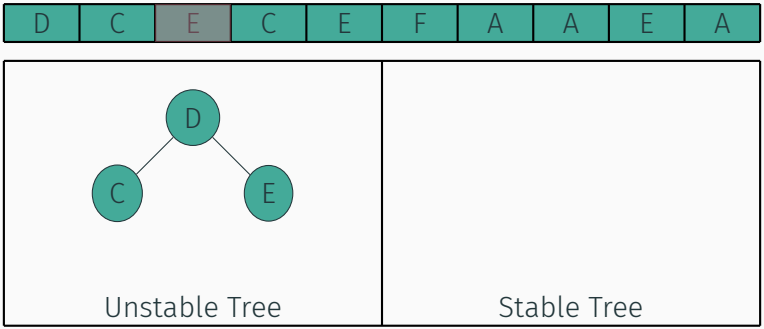
Kernel Samepage Merging (KSM)



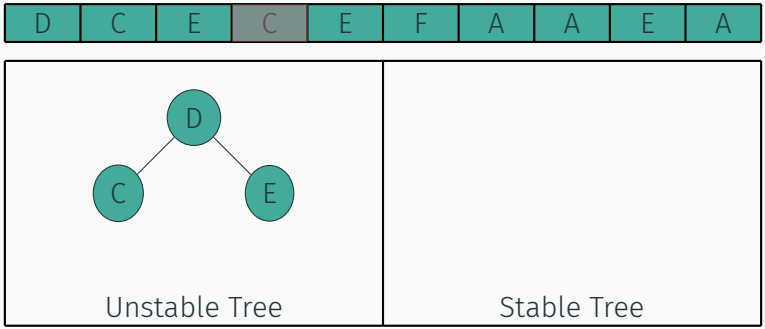
Kernel Samepage Merging (KSM)



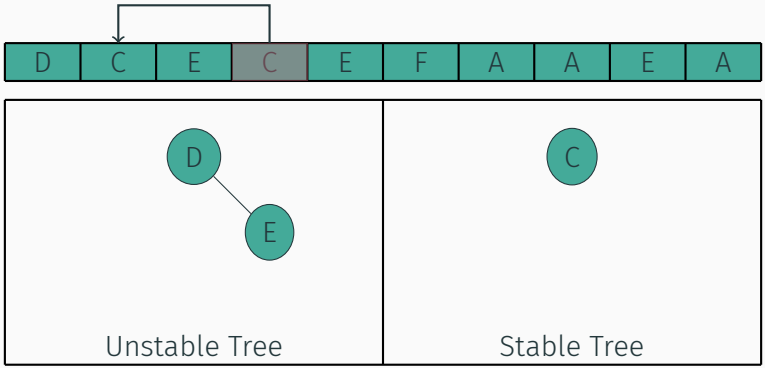
Kernel Samepage Merging (KSM)



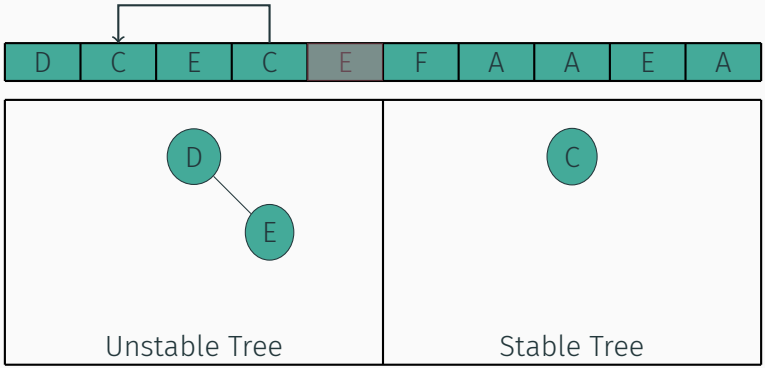
Kernel Samepage Merging (KSM)



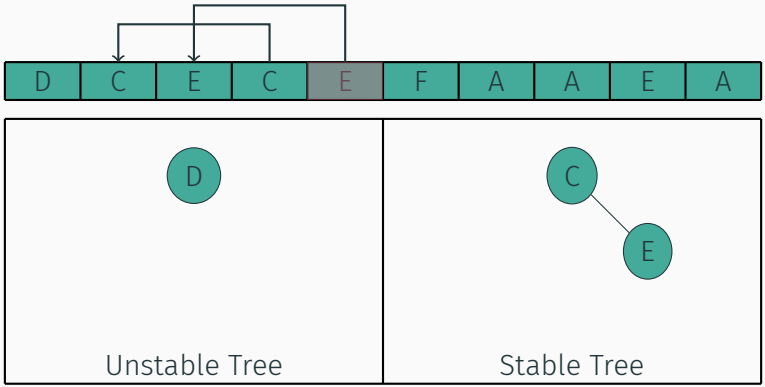
Kernel Samepage Merging (KSM)



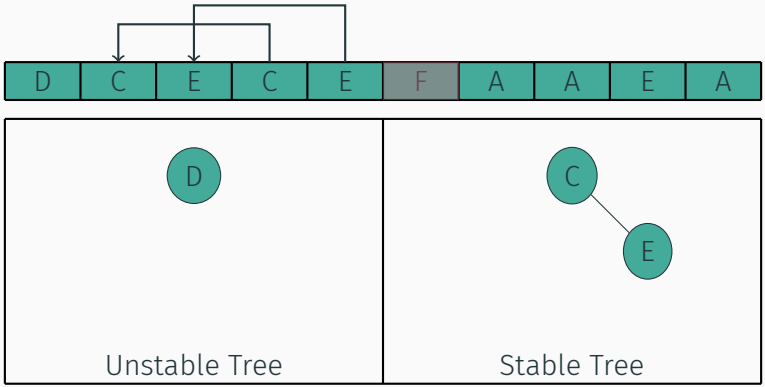
Kernel Samepage Merging (KSM)



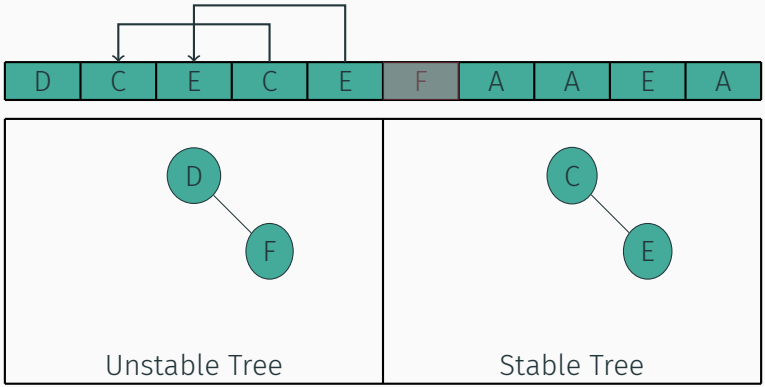
Kernel Samepage Merging (KSM)



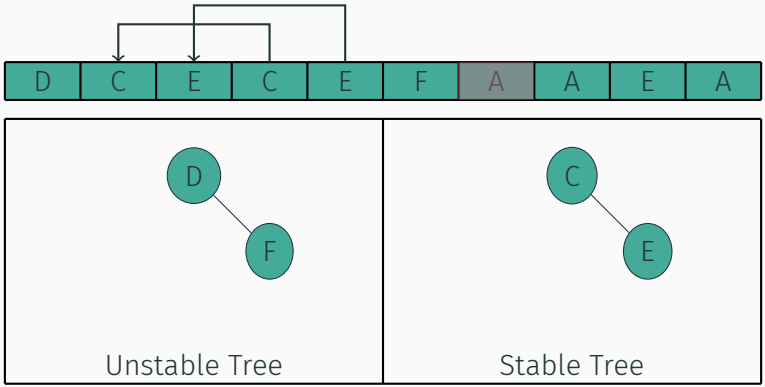
Kernel Samepage Merging (KSM)



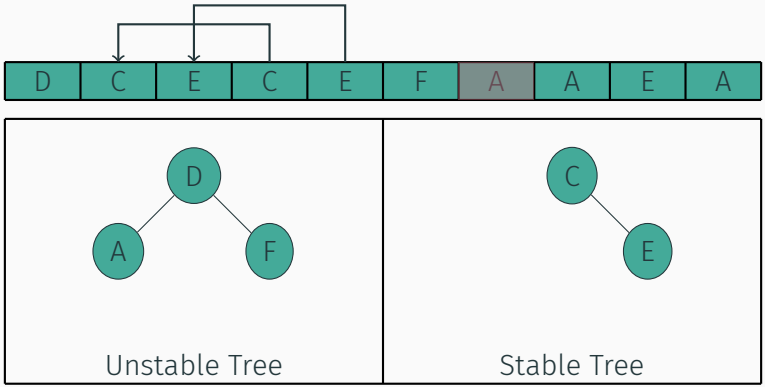
Kernel Samepage Merging (KSM)



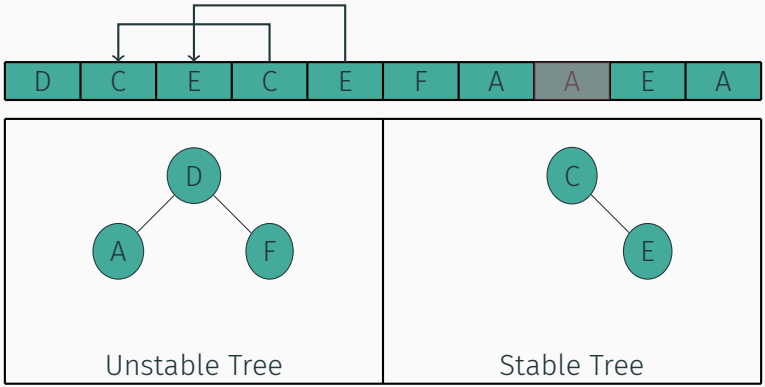
Kernel Samepage Merging (KSM)



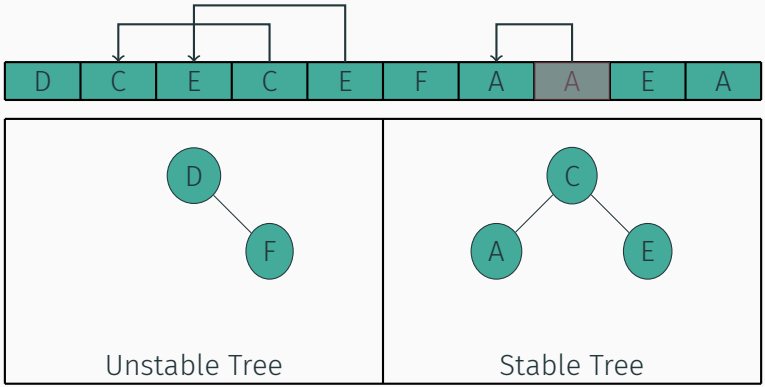
Kernel Samepage Merging (KSM)



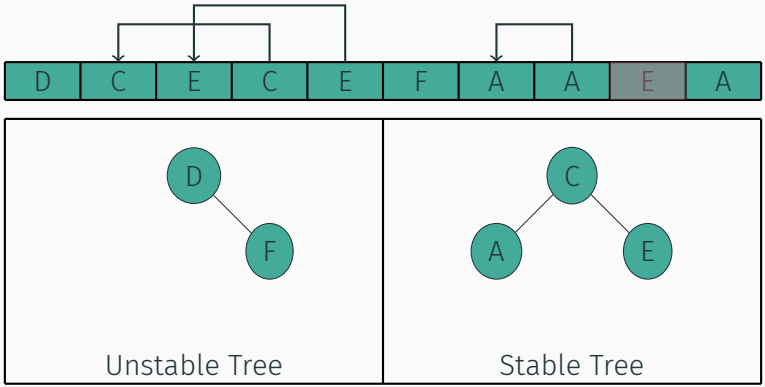
Kernel Samepage Merging (KSM)



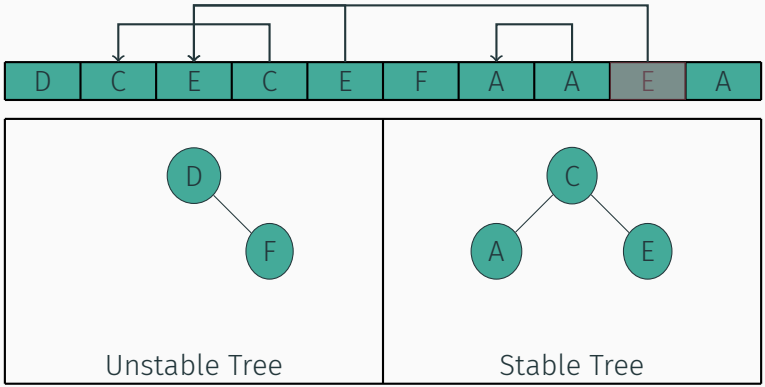
Kernel Samepage Merging (KSM)



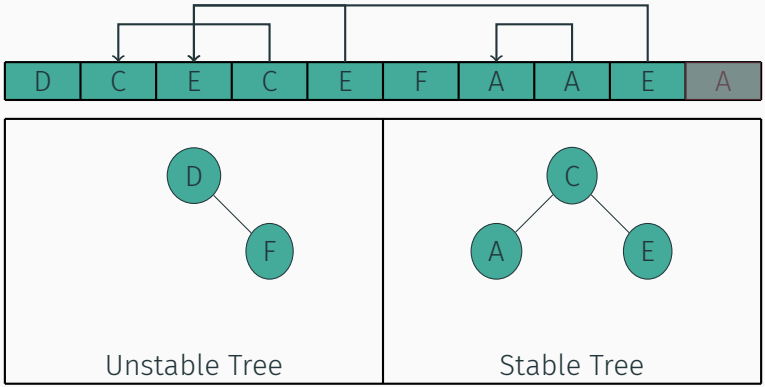
Kernel Samepage Merging (KSM)



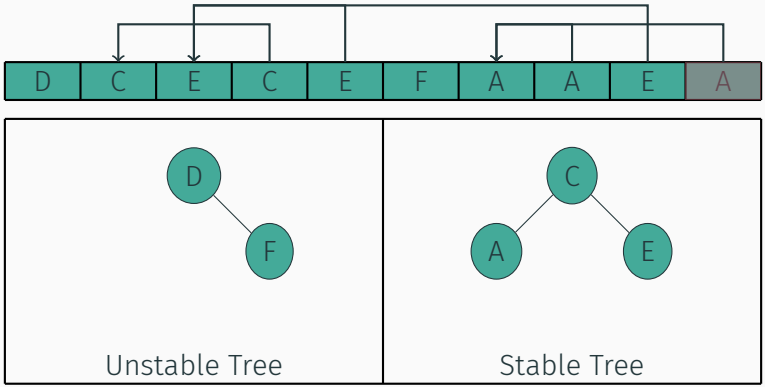
Kernel Samepage Merging (KSM)



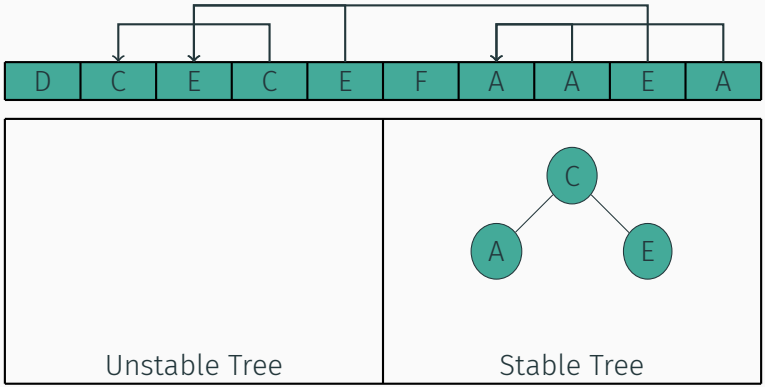
Kernel Samepage Merging (KSM)



Kernel Samepage Merging (KSM)



Kernel Samepage Merging (KSM)



Demo 2

Analyse virtueller Adressierung mit KSM

Rowhammer

- DRAM Zellen sind in einem Array aus Zeilen (*Rows*) und Spalten (*Columns*) angeordnet
- Zugriff erfolgt über den Rowbuffer gepuffert auf ganze Rows

Rowhammer

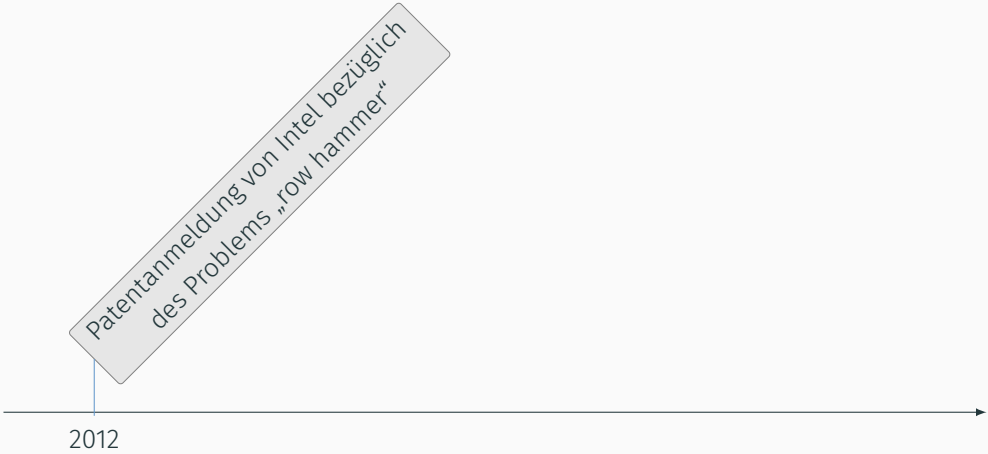
- DRAM Zellen sind in einem Array aus Zeilen (*Rows*) und Spalten (*Columns*) angeordnet
- Zugriff erfolgt über den Rowbuffer gepuffert auf ganze Rows
- Höhere Speicherkapazität erfordert höhere Integrationsdichte der DRAM Zellen

Rowhammer

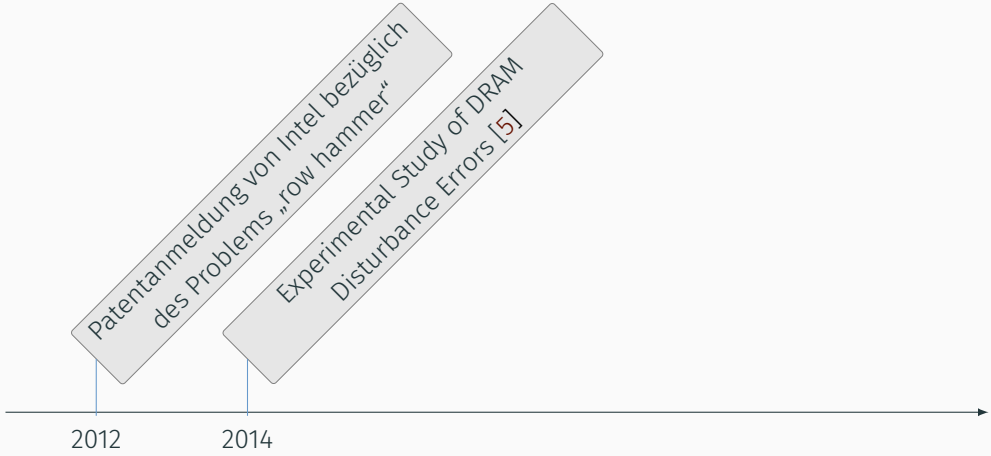
- DRAM Zellen sind in einem Array aus Zeilen (*Rows*) und Spalten (*Columns*) angeordnet
- Zugriff erfolgt über den Rowbuffer gepuffert auf ganze Rows
- Höhere Speicherkapazität erfordert höhere Integrationsdichte der DRAM Zellen
- Zellen sind so nah zusammen dass häufige Zugriffe auf Rows zu Speicherfehlern in benachbarten Rows führen

Rowhammer

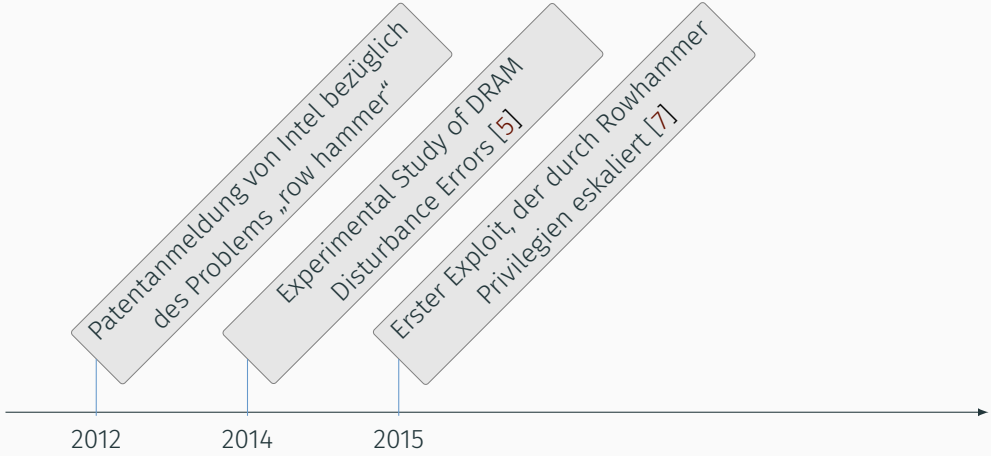
Rowhammer


Patentanmeldung von Intel bezüglich des Problems „row hammer“
2012

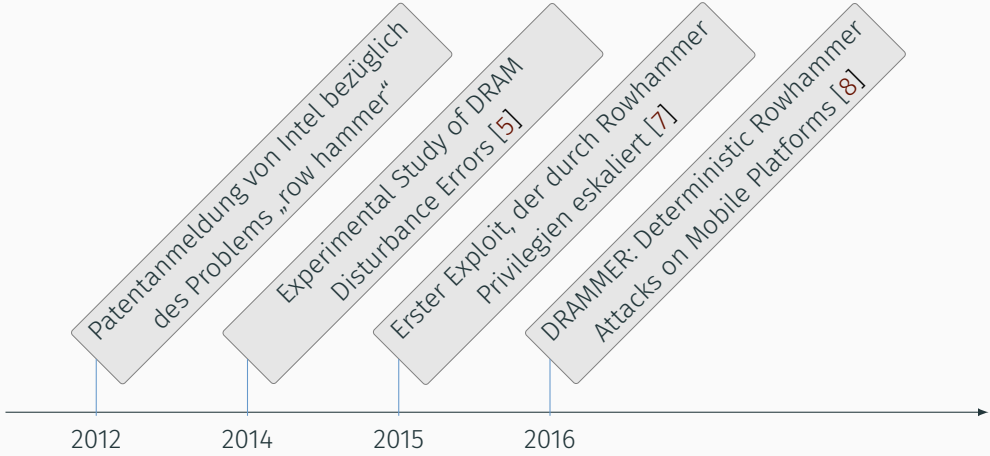
Rowhammer


Patentanmeldung von Intel bezüglich des Problems „row hammer“
Experimental Study of DRAM Disturbance Errors [5]
2012
2014

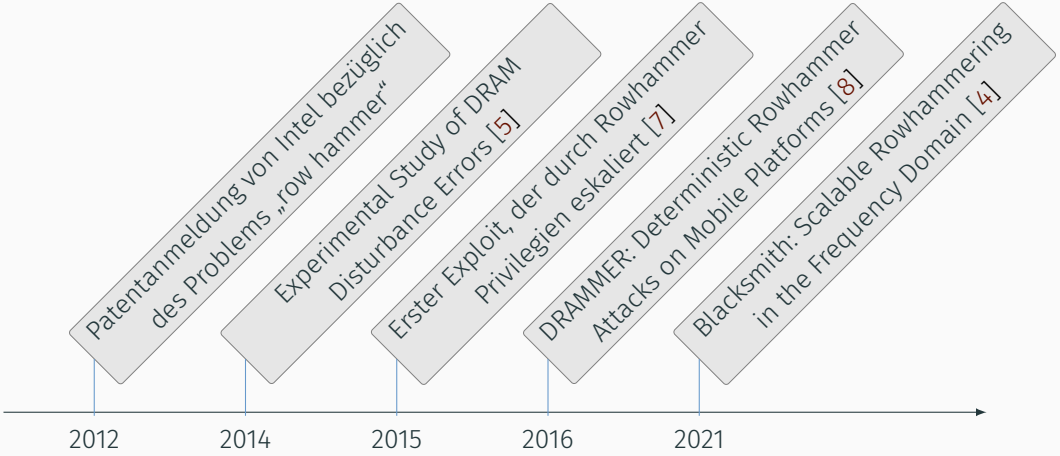
Rowhammer


Patentanmeldung von Intel bezüglich des Problems „row hammer“
Experimental Study of DRAM Disturbance Errors [5]
Erster Exploit, der durch Rowhammer Privilegien eskaliert [7]
2012
2014
2015

Rowhammer

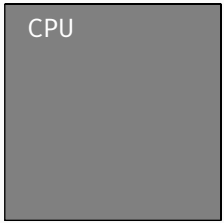

Patentanmeldung von Intel bezüglich des Problems „row hammer“
Experimental Study of DRAM Disturbance Errors [5]
Erster Exploit, der durch Rowhammer Privilegien eskaliert [7]
DRAMMER: Deterministic Rowhammer Attacks on Mobile Platforms [8]
2012
2014
2015
2016

Rowhammer


Patentanmeldung von Intel bezüglich des Problems „row hammer“
Experimental Study of DRAM Disturbance Errors [5]
Erster Exploit, der durch Rowhammer Privilegien eskaliert [7]
DRAMMER: Deterministic Rowhammer Attacks on Mobile Platforms [8]
Blacksmith: Scalable Rowhammering in the Frequency Domain [4]
2012
2014
2015
2016
2021

Rowhammer

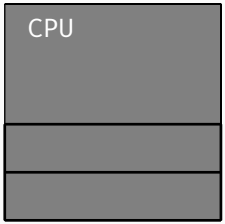
```
1  hammer:
2    mov eax, X
3    mov ebx, Y
4    clflush X
5    clflush Y
6    jmp hammer
```



Quellcode von Wikipedia, Abbildung aus den Slides zu [3]

Rowhammer

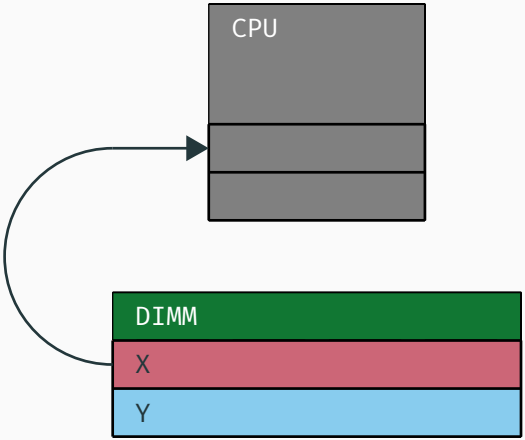
```
1  hammer:
2    mov eax, X
3    mov ebx, Y
4    clflush X
5    clflush Y
6    jmp hammer
```



Quellcode von Wikipedia, Abbildung aus den Slides zu [3]

Rowhammer

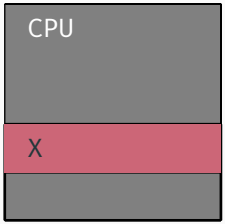
```
1  hammer:
2    mov eax, X
3    mov ebx, Y
4    clflush X
5    clflush Y
6    jmp hammer
```



Quellcode von Wikipedia, Abbildung aus den Slides zu [3]

Rowhammer

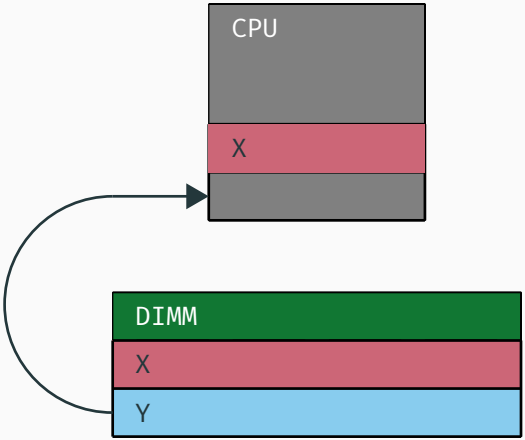
```
1  hammer:
2    mov eax, X
3    mov ebx, Y
4    clflush X
5    clflush Y
6    jmp hammer
```



Quellcode von Wikipedia, Abbildung aus den Slides zu [3]

Rowhammer

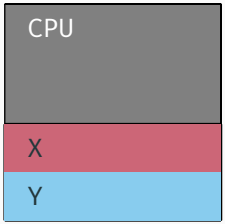
```
1  hammer:
2    mov eax, X
3    mov ebx, Y
4    clflush X
5    clflush Y
6    jmp hammer
```



Quellcode von Wikipedia, Abbildung aus den Slides zu [3]

Rowhammer

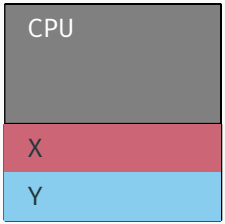
```
1  hammer:
2    mov eax, X
3    mov ebx, Y
4    clflush X
5    clflush Y
6    jmp hammer
```



Quellcode von Wikipedia, Abbildung aus den Slides zu [3]

Rowhammer

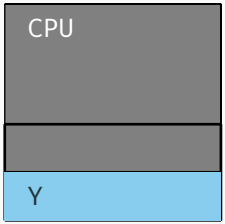
```
1  hammer:
2    mov eax, X
3    mov ebx, Y
4    clflush X
5    clflush Y
6    jmp hammer
```



Quellcode von Wikipedia, Abbildung aus den Slides zu [3]

Rowhammer

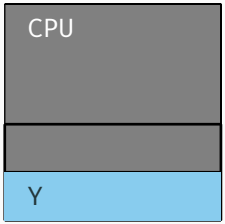
```
1  hammer:
2    mov eax, X
3    mov ebx, Y
4    clflush X
5    clflush Y
6    jmp hammer
```



Quellcode von Wikipedia, Abbildung aus den Slides zu [3]

Rowhammer

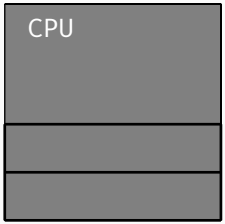
```
1  hammer:
2    mov eax, X
3    mov ebx, Y
4    clflush X
5    clflush Y
6    jmp hammer
```



Quellcode von Wikipedia, Abbildung aus den Slides zu [3]

Rowhammer

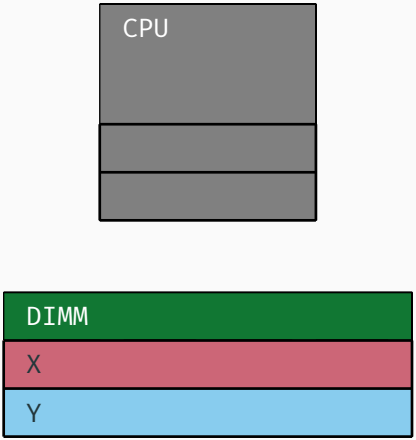
```
1  hammer:
2    mov eax, X
3    mov ebx, Y
4    clflush X
5    clflush Y
6    jmp hammer
```



Quellcode von Wikipedia, Abbildung aus den Slides zu [3]

Rowhammer

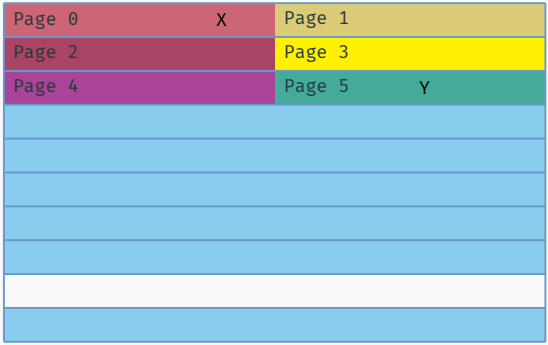
```
1  hammer:
2   mov eax, X
3   mov ebx, Y
4   clflush X
5   clflush Y
6   jmp hammer
```



Quellcode von Wikipedia, Abbildung aus den Slides zu [3]

Rowhammer

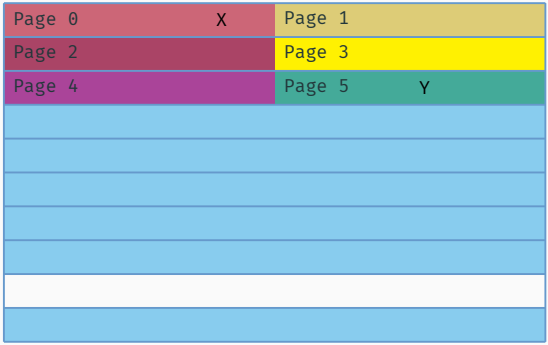
```
1  hammer:
2    mov eax, X
3    mov ebx, Y
4    clflush X
5    clflush Y
6    jmp hammer
```



Quellcode von Wikipedia, Abbildung aus den Slides zu [3]

Rowhammer

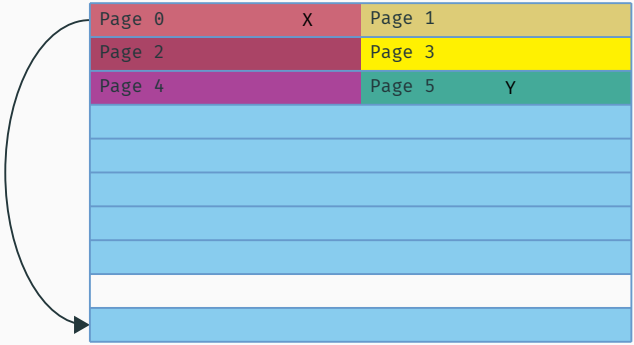
```
1  hammer:
2    mov eax, X
3    mov ebx, Y
4    clflush X
5    clflush Y
6    jmp hammer
```



Quellcode von Wikipedia, Abbildung aus den Slides zu [3]

Rowhammer

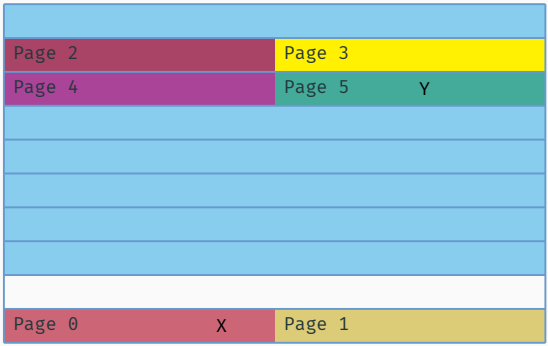
```
1  hammer:
2    mov eax, X
3    mov ebx, Y
4    clflush X
5    clflush Y
6    jmp hammer
```



Quellcode von Wikipedia, Abbildung aus den Slides zu [3]

Rowhammer

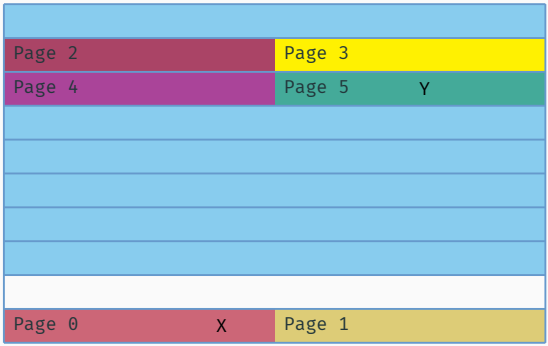
```
1  hammer:
2    mov eax, X
3    mov ebx, Y
4    clflush X
5    clflush Y
6    jmp hammer
```



Quellcode von Wikipedia, Abbildung aus den Slides zu [3]

Rowhammer

```
1  hammer:
2    mov eax, X
3    mov ebx, Y
4    clflush X
5    clflush Y
6    jmp hammer
```



Quellcode von Wikipedia, Abbildung aus den Slides zu [3]

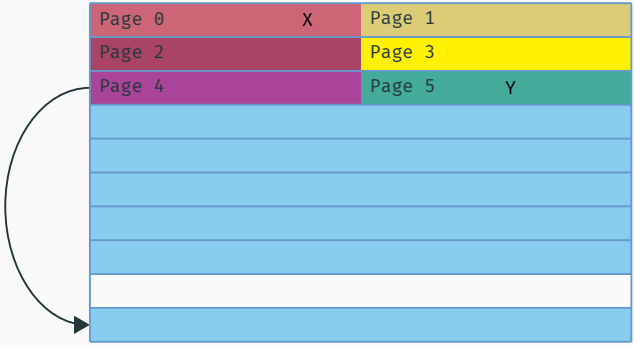
Rowhammer

```
1  hammer:
2    mov eax, X
3    mov ebx, Y
4    clflush X
5    clflush Y
6    jmp hammer
```

Quellcode von Wikipedia, Abbildung aus den Slides zu [3]

Rowhammer

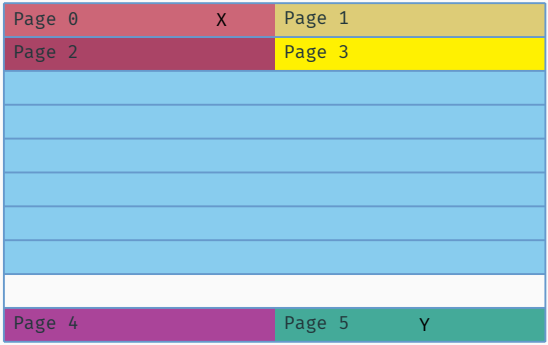
```
1  hammer:
2    mov eax, X
3    mov ebx, Y
4    clflush X
5    clflush Y
6    jmp hammer
```



Quellcode von Wikipedia, Abbildung aus den Slides zu [3]

Rowhammer

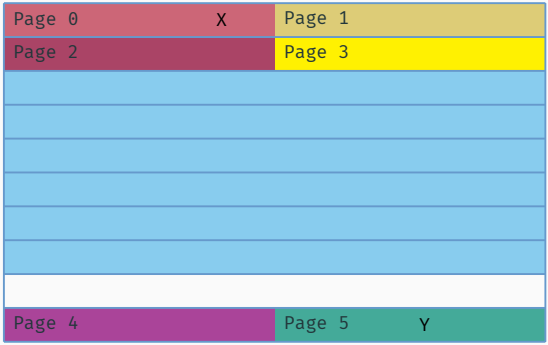
```
1  hammer:
2    mov eax, X
3    mov ebx, Y
4    clflush X
5    clflush Y
6    jmp hammer
```



Quellcode von Wikipedia, Abbildung aus den Slides zu [3]

Rowhammer

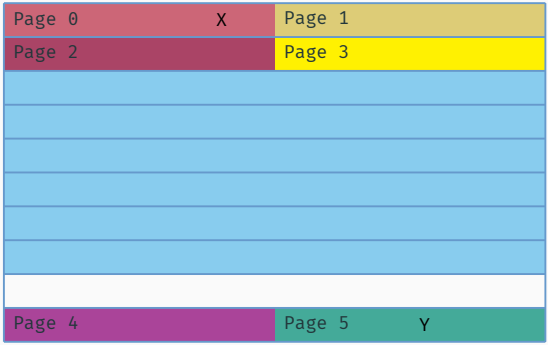
```
1  hammer:
2    mov eax, X
3    mov ebx, Y
4    clflush X
5    clflush Y
6    jmp hammer
```



Quellcode von Wikipedia, Abbildung aus den Slides zu [3]

Rowhammer

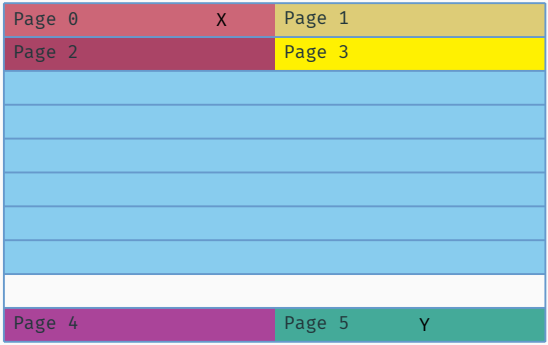
```
1  hammer:
2    mov eax, X
3    mov ebx, Y
4    clflush X
5    clflush Y
6    jmp hammer
```



Quellcode von Wikipedia, Abbildung aus den Slides zu [3]

Rowhammer

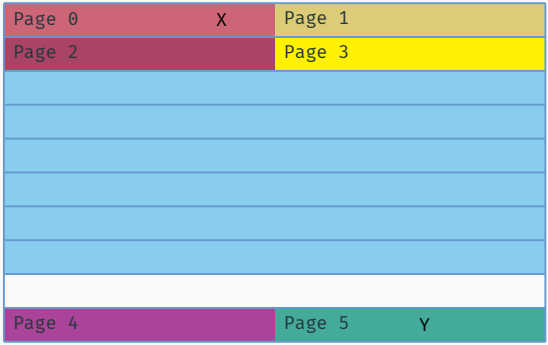
```
1  hammer:
2   mov eax, X
3   mov ebx, Y
4   clflush X
5   clflush Y
6   jmp hammer
```



Quellcode von Wikipedia, Abbildung aus den Slides zu [3]

Rowhammer

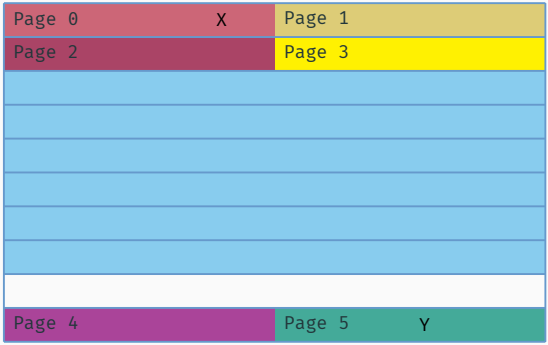
```
1  hammer:
2    mov eax, X
3    mov ebx, Y
4    clflush X
5    clflush Y
6    jmp hammer
```



Quellcode von Wikipedia, Abbildung aus den Slides zu [3]

Rowhammer

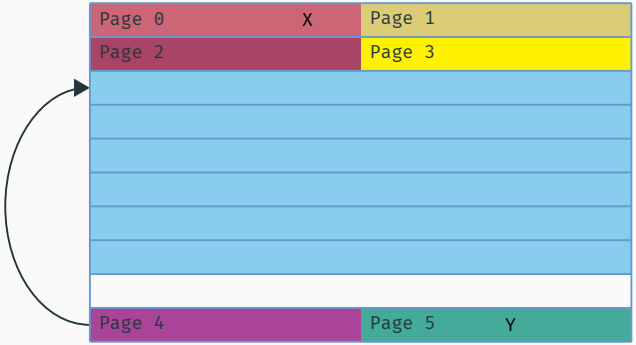
```
1  hammer:
2    mov eax, X
3    mov ebx, Y
4    clflush X
5    clflush Y
6    jmp hammer
```



Quellcode von Wikipedia, Abbildung aus den Slides zu [3]

Rowhammer

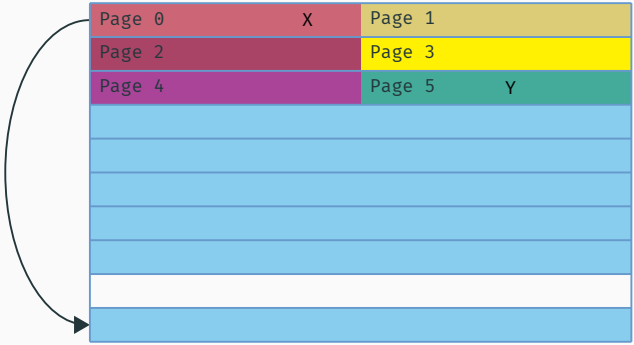
```
1  hammer:
2    mov eax, X
3    mov ebx, Y
4    clflush X
5    clflush Y
6    jmp hammer
```



Quellcode von Wikipedia, Abbildung aus den Slides zu [3]

Rowhammer

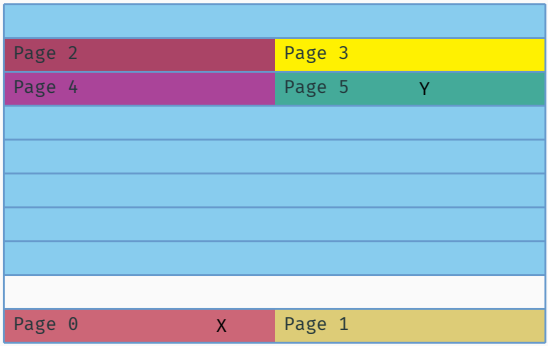
```
1  hammer:
2    mov eax, X
3    mov ebx, Y
4    clflush X
5    clflush Y
6    jmp hammer
```



Quellcode von Wikipedia, Abbildung aus den Slides zu [3]

Rowhammer

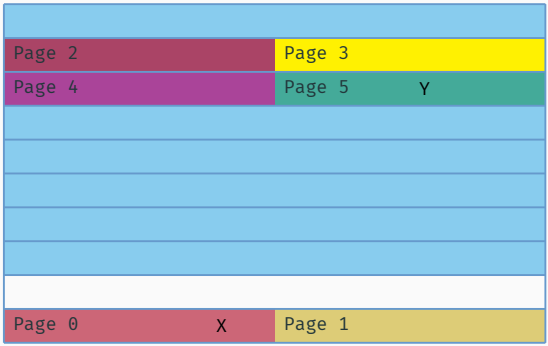
```
1  hammer:
2    mov eax, X
3    mov ebx, Y
4    clflush X
5    clflush Y
6    jmp hammer
```



Quellcode von Wikipedia, Abbildung aus den Slides zu [3]

Rowhammer

```
1  hammer:
2    mov eax, X
3    mov ebx, Y
4    clflush X
5    clflush Y
6    jmp hammer
```



Quellcode von Wikipedia, Abbildung aus den Slides zu [3]

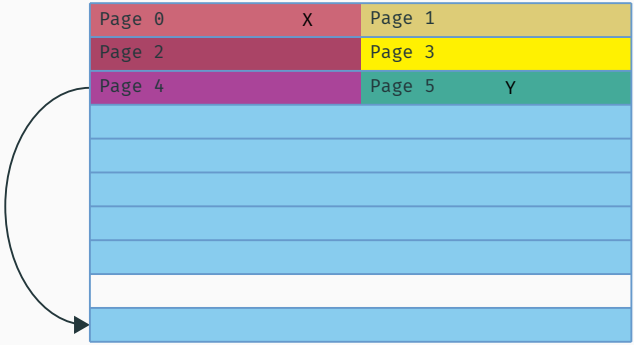
Rowhammer

```
1  hammer:
2    mov eax, X
3    mov ebx, Y
4    clflush X
5    clflush Y
6    jmp hammer
```

Quellcode von Wikipedia, Abbildung aus den Slides zu [3]

Rowhammer

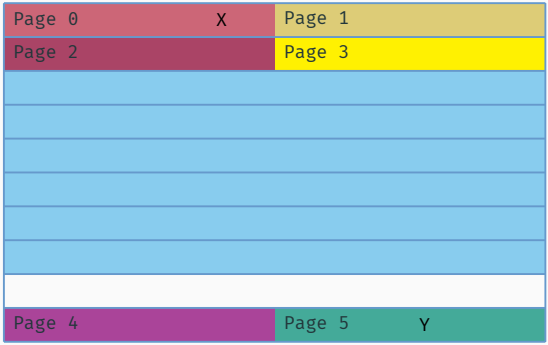
```
1  hammer:
2    mov eax, X
3    mov ebx, Y
4    clflush X
5    clflush Y
6    jmp hammer
```



Quellcode von Wikipedia, Abbildung aus den Slides zu [3]

Rowhammer

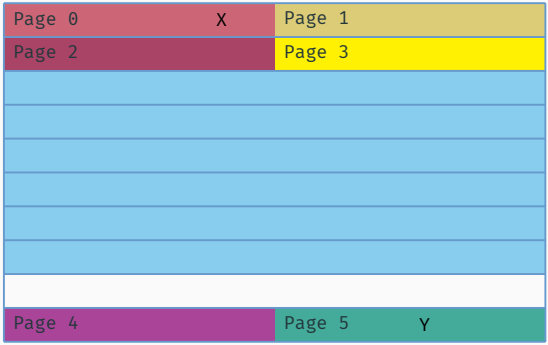
```
1  hammer:
2    mov eax, X
3    mov ebx, Y
4    clflush X
5    clflush Y
6    jmp hammer
```



Quellcode von Wikipedia, Abbildung aus den Slides zu [3]

Rowhammer

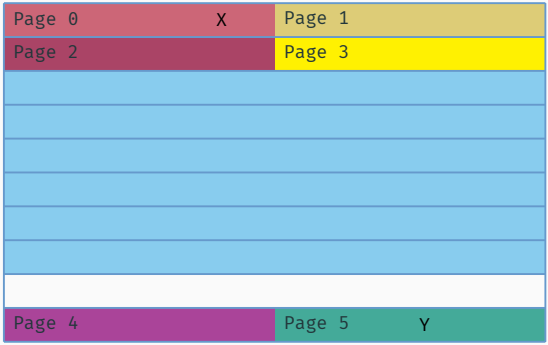
```
1  hammer:
2    mov eax, X
3    mov ebx, Y
4    clflush X
5    clflush Y
6    jmp hammer
```



Quellcode von Wikipedia, Abbildung aus den Slides zu [3]

Rowhammer

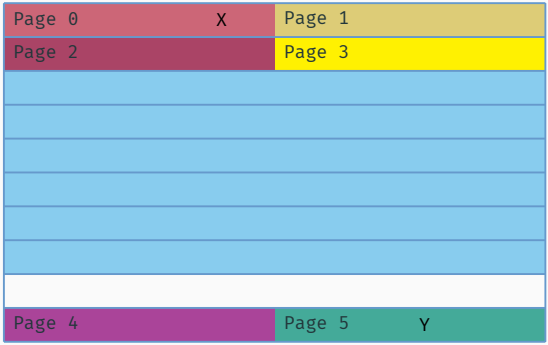
```
1  hammer:
2    mov eax, X
3    mov ebx, Y
4    clflush X
5    clflush Y
6    jmp hammer
```



Quellcode von Wikipedia, Abbildung aus den Slides zu [3]

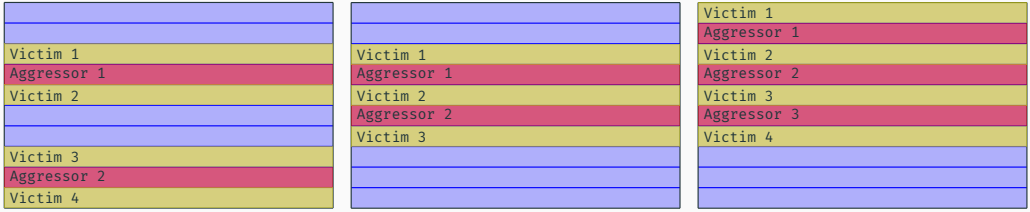
Rowhammer

```
1  hammer:
2   mov eax, X
3   mov ebx, Y
4   clflush X
5   clflush Y
6   jmp hammer
```



Quellcode von Wikipedia, Abbildung aus den Slides zu [3]

Rowhammer



(a) Single Side (b) Double Side (c) Three Side

Abbildung 1: Beispiele für Rowhammer-Muster

Abbildung aus den Slides zu [3]

Rowhammer

- **Hypothese**: Je höher die Speicherkapazität eines IC bei gleicher Bauart und gleicher Größe ist, desto höher ist die Integrationsdichte
- **Hypothese**: Je höher die Integrationsdichte eines DRAM IC, desto anfälliger ist dieser für Rowhammer [2]

Rowhammer

- **Hypothese**: Je höher die Speicherkapazität eines IC bei gleicher Bauart und gleicher Größe ist, desto höher ist die Integrationsdichte
- **Hypothese**: Je höher die Integrationsdichte eines DRAM IC, desto anfälliger ist dieser für Rowhammer [2]
- **Problem**: Integrationsdichte wird weiter steigen, Rowhammer wird schlimmer

Rowhammer

- **Hypothese**: Je höher die Speicherkapazität eines IC bei gleicher Bauart und gleicher Größe ist, desto höher ist die Integrationsdichte
- **Hypothese**: Je höher die Integrationsdichte eines DRAM IC, desto anfälliger ist dieser für Rowhammer [2]
- **Problem**: Integrationsdichte wird weiter steigen, Rowhammer wird schlimmer
- **Mitigationen**: Da das Problem nicht gelöst werden kann, kommen verschiedene Mitigationen zum Einsatz:

Rowhammer

- **Hypothese**: Je höher die Speicherkapazität eines IC bei gleicher Bauart und gleicher Größe ist, desto höher ist die Integrationsdichte
- **Hypothese**: Je höher die Integrationsdichte eines DRAM IC, desto anfälliger ist dieser für Rowhammer [2]
- **Problem**: Integrationsdichte wird weiter steigen, Rowhammer wird schlimmer
- **Mitigationen**: Da das Problem nicht gelöst werden kann, kommen verschiedene Mitigationen zum Einsatz:
  - **Allgemein**: Verdopplung der Refresh-Rate, Error Correction Code (ECC), ...

Rowhammer

- **Hypothese**: Je höher die Speicherkapazität eines IC bei gleicher Bauart und gleicher Größe ist, desto höher ist die Integrationsdichte
- **Hypothese**: Je höher die Integrationsdichte eines DRAM IC, desto anfälliger ist dieser für Rowhammer [2]
- **Problem**: Integrationsdichte wird weiter steigen, Rowhammer wird schlimmer
- **Mitigationen**: Da das Problem nicht gelöst werden kann, kommen verschiedene Mitigationen zum Einsatz:
  - **Allgemein**: Verdopplung der Refresh-Rate, Error Correction Code (ECC), ...
  - **Mustererkennung**: Target Row Refresh (TRR) und pseudo-TRR (pTRR), ...

- **Problem bei allgemeinen Mitigationen**: Hoher Overhead für effektive Mitigationen erforderlich (Overhead deutlich spürbar)

Rowhammer

- **Problem bei allgemeinen Mitigationen**: Hoher Overhead für effektive Mitigationen erforderlich (Overhead deutlich spürbar)
- **Problem bei Mustererkennung**: Muster müssen von „normalen“ Zugriffen unterschieden werden, müssen also bekannt sein

Rowhammer

- **Problem bei allgemeinen Mitigationen**: Hoher Overhead für effektive Mitigationen erforderlich (Overhead deutlich spürbar)
- **Problem bei Mustererkennung**: Muster müssen von „normalen“ Zugriffen unterschieden werden, müssen also bekannt sein
- **Offene Forschungsfrage**: Finden von allgemeinen Mitigationen mit geringen/tolerierbaren Overhead

Angriffe auf KSM

Information leaks

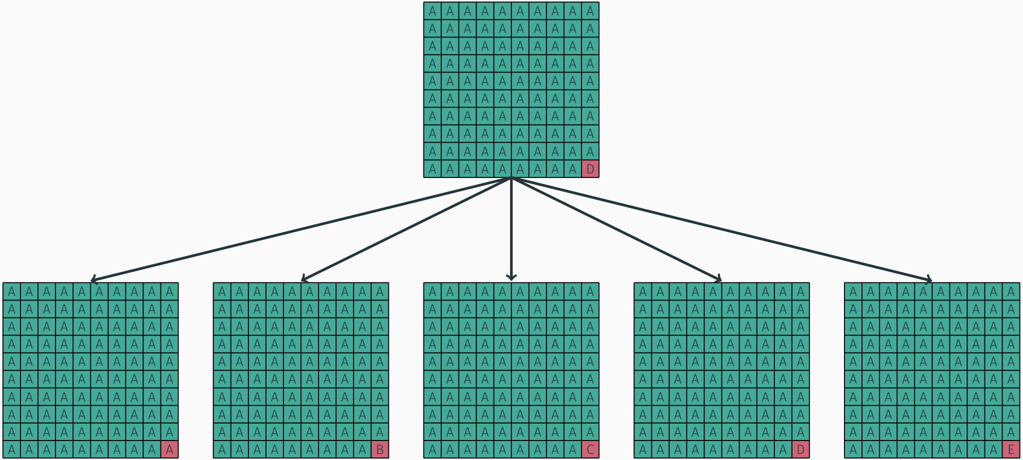
- Copy-On-Write Policy für gemergete Seiten
  - Schreiben auf einer dedupizierten Seite ist deutlich langsamer als auf einer nicht dedupizierten Seite

Information leaks

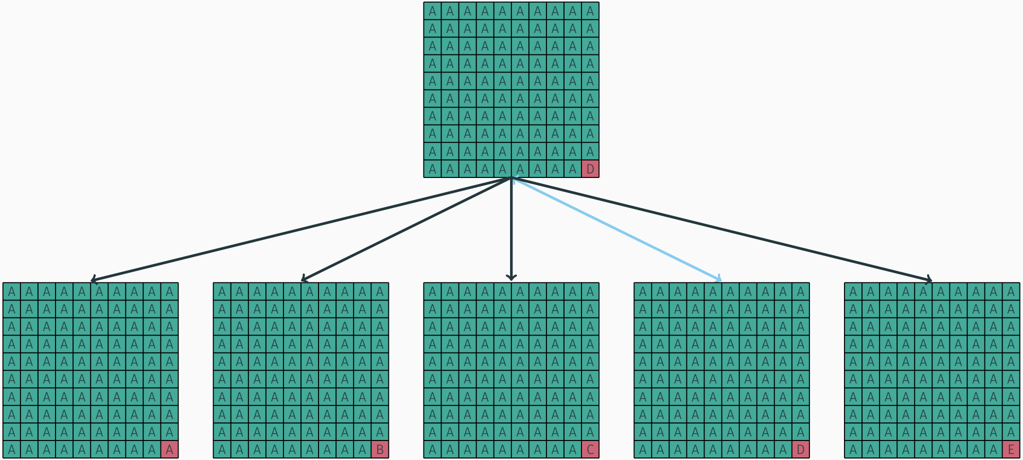
- Copy-On-Write Policy für gemergete Seiten
  - Schreiben auf einer dedupizierten Seite ist deutlich langsamer als auf einer nicht dedupizierten Seite
- **Idee**: Ausnutzen von KSM zum Raten von unbekannten Inhalten [1]

Information leaks

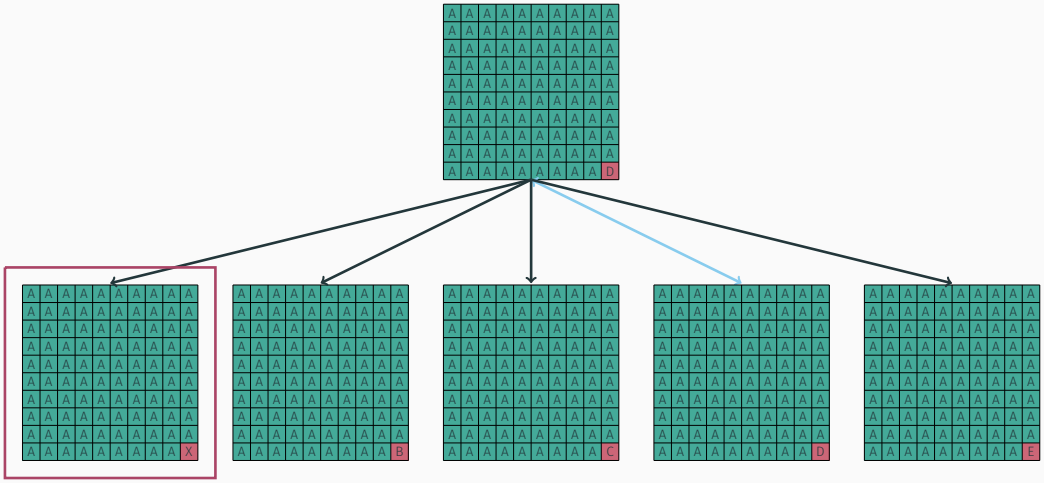
Information leaks



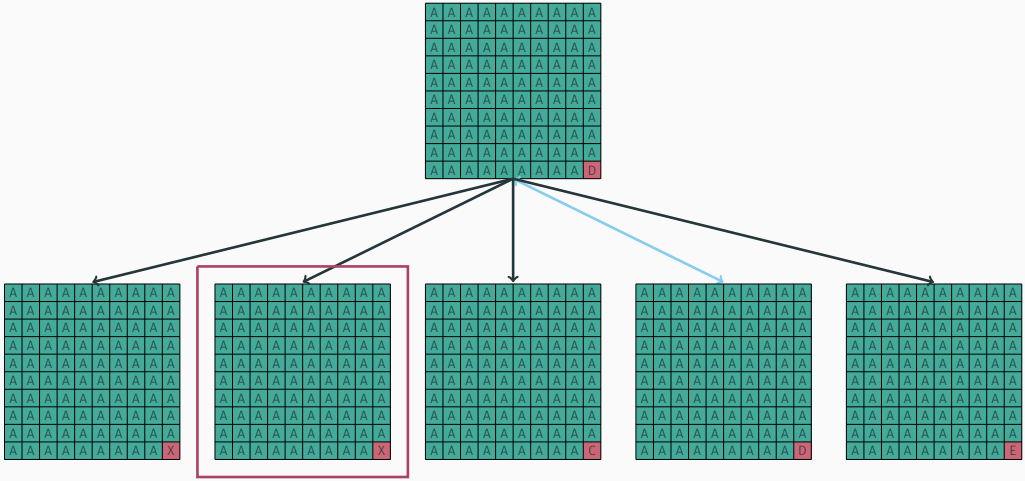
Information leaks



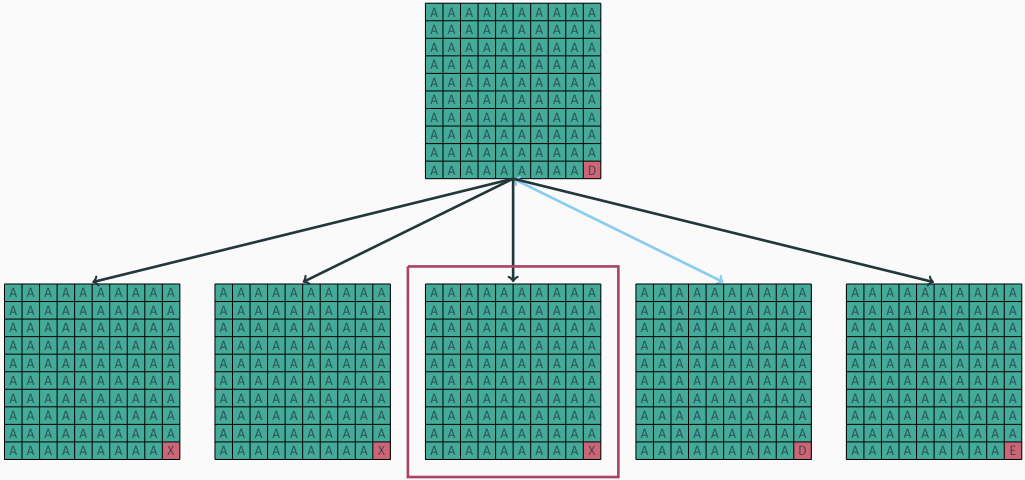
Information leaks



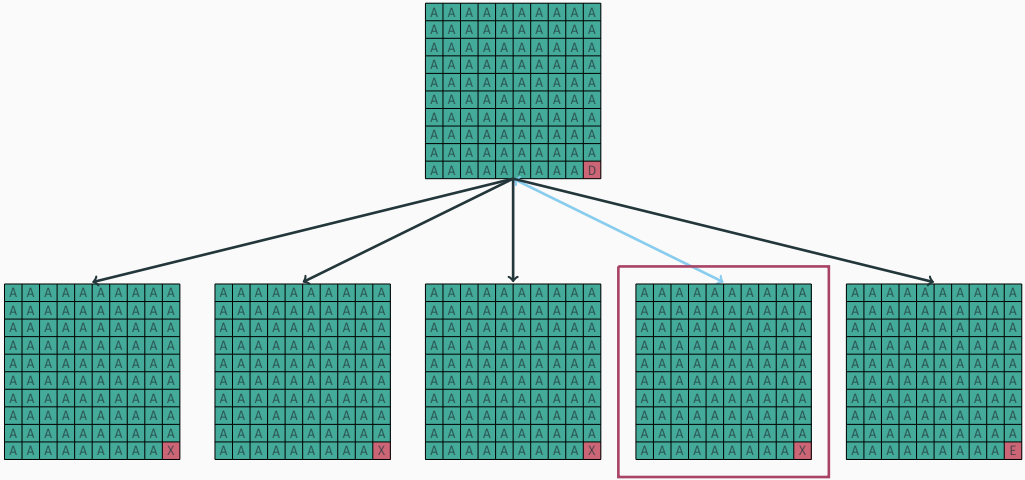
Information leaks



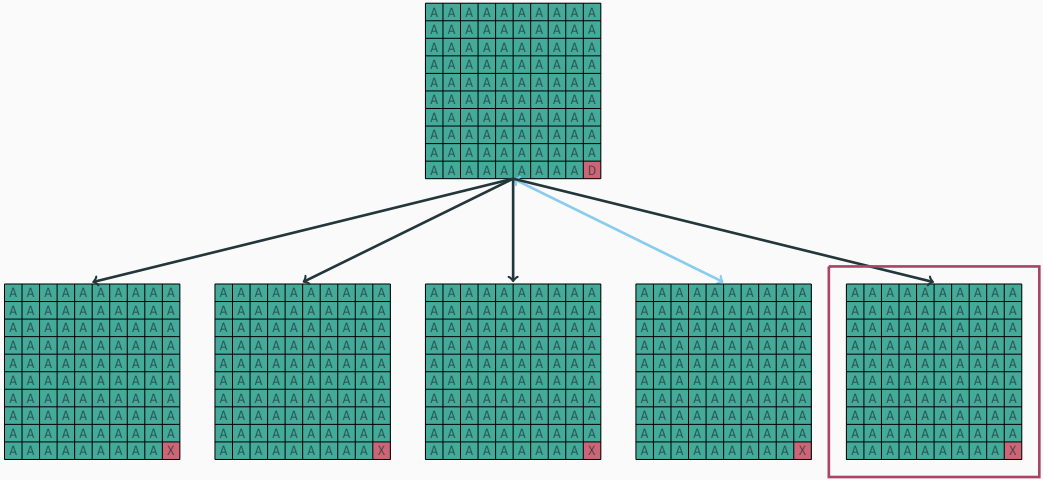
Information leaks



Information leaks



Information leaks



Covert Channel

- Kommunikation zwischen Prozessen ist zentraler Bestandteil der Informationstechnik

Covert Channel

- Kommunikation zwischen Prozessen ist zentraler Bestandteil der Informationstechnik
- In manchen Fällen ist die Kommunikation unerwünscht und wird unterbunden (z.B. Firewall)

Covert Channel

- Kommunikation zwischen Prozessen ist zentraler Bestandteil der Informationstechnik
- In manchen Fällen ist die Kommunikation unerwünscht und wird unterbunden (z.B. Firewall)
- Covert Channels ermöglichen Kommunikation ohne Nutzung „offizieller“ Kommunikationswege, umgehen damit entsprechende Einschränkungen

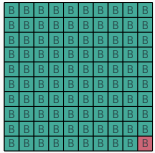
Covert Channel

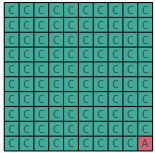
- Copy-On-Write Policy für gemergete Seiten
  - Schreiben auf einer dedupizierten Seite ist deutlich langsamer als auf einer nicht dedupizierten Seite

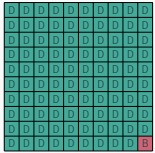
Covert Channel

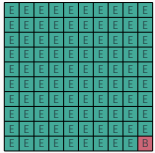
- Copy-On-Write Policy für gemergete Seiten
  - Schreiben auf einer deduplizierten Seite ist deutlich langsamer als auf einer nicht deduplizierten Seite
- **Idee**: Ausnutzen von KSM zur verdeckten Kommunikation [1]

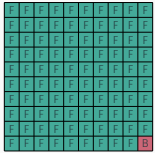
Covert Channel



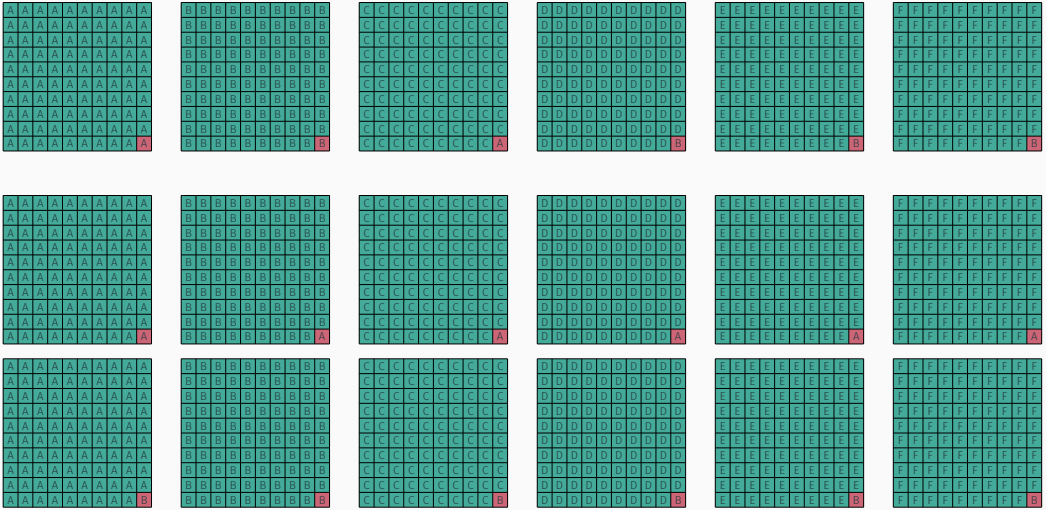




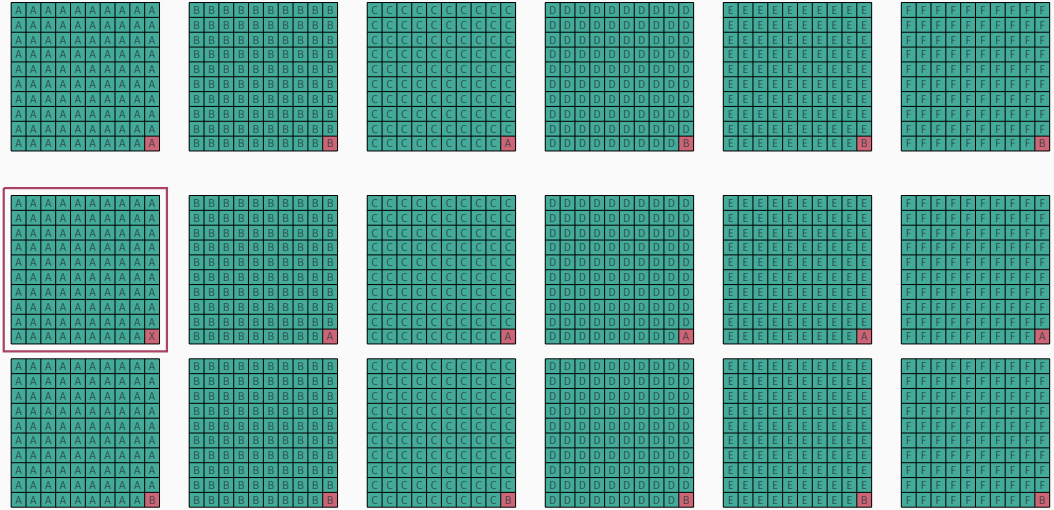




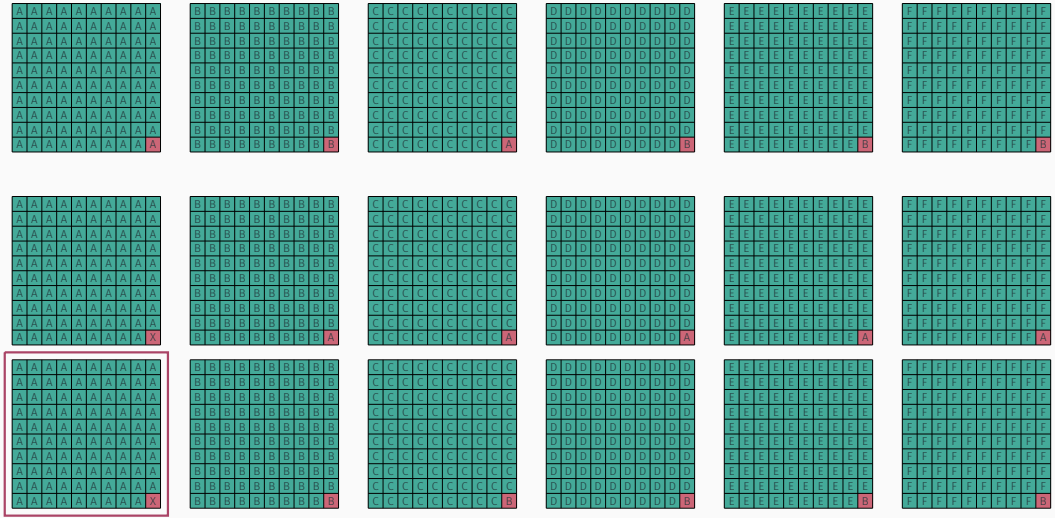
Covert Channel



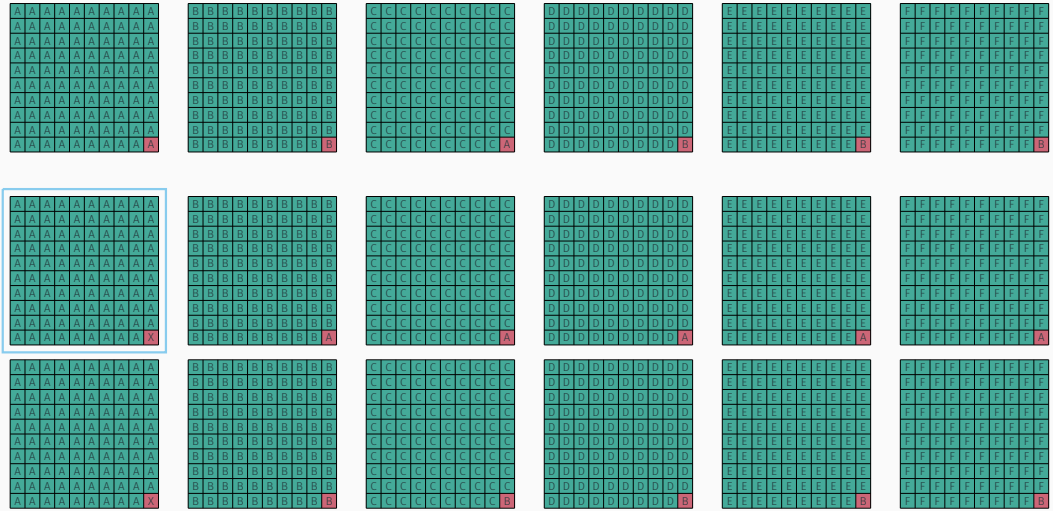
Covert Channel



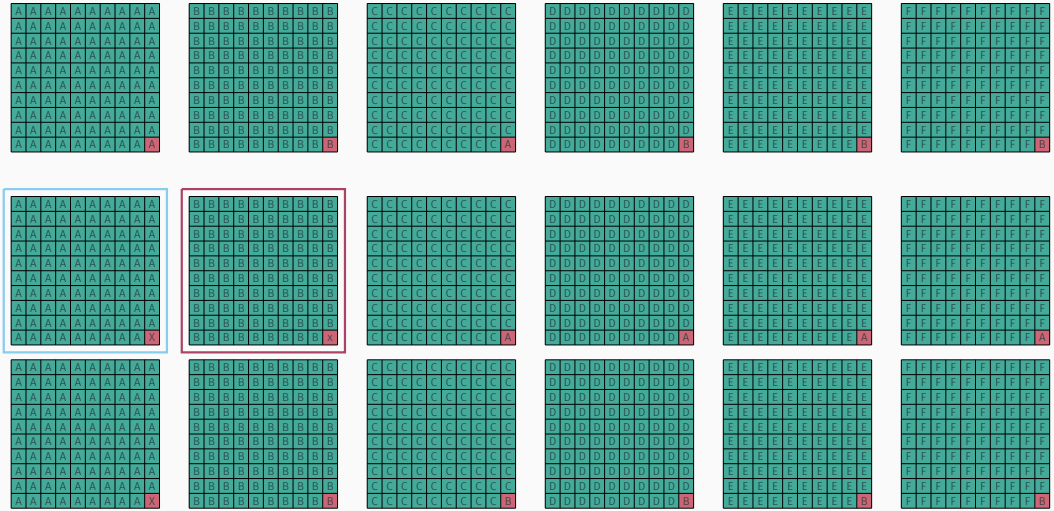
Covert Channel



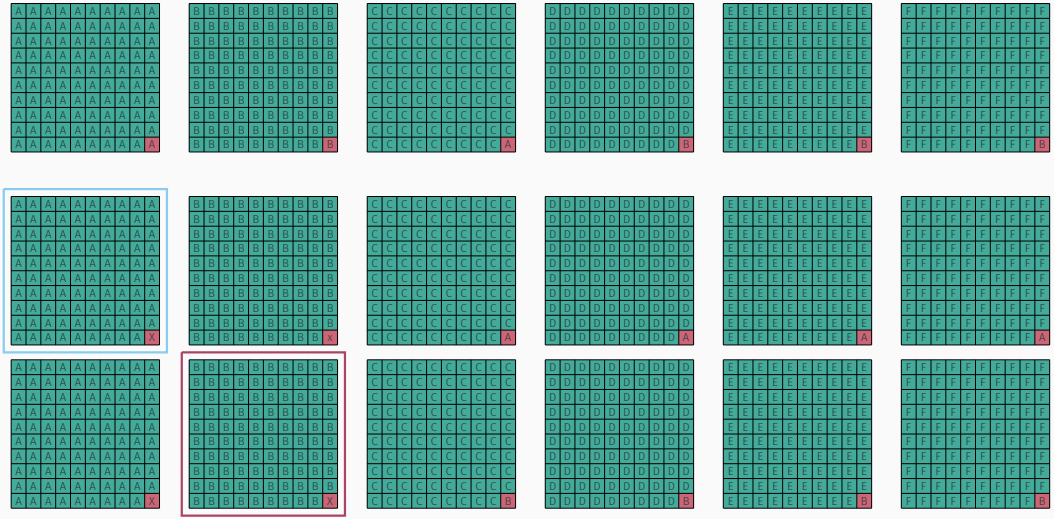
Covert Channel



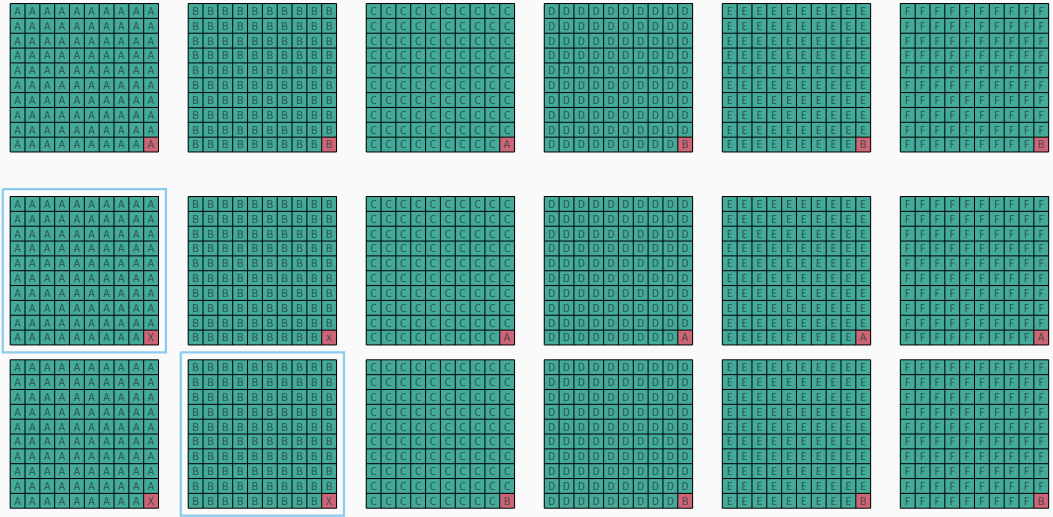
Covert Channel



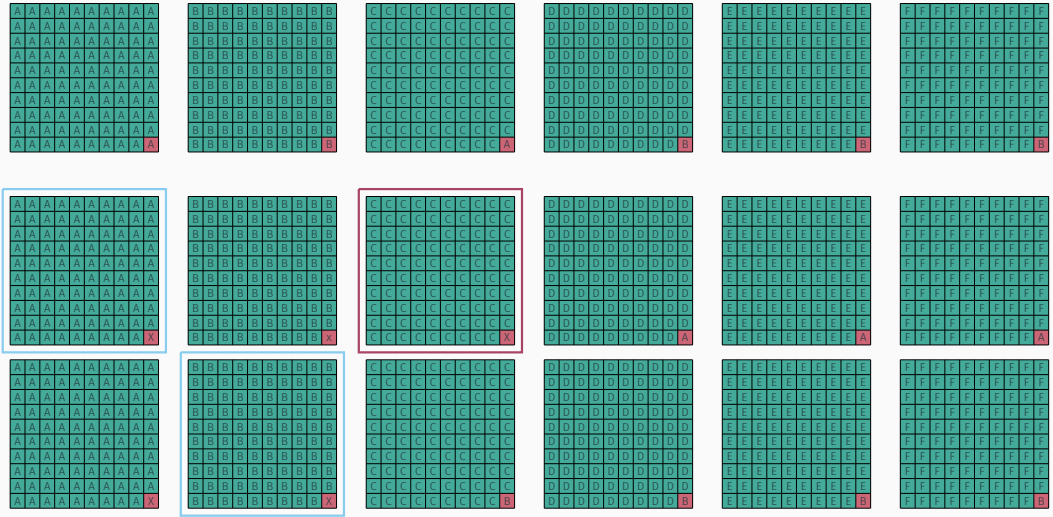
Covert Channel



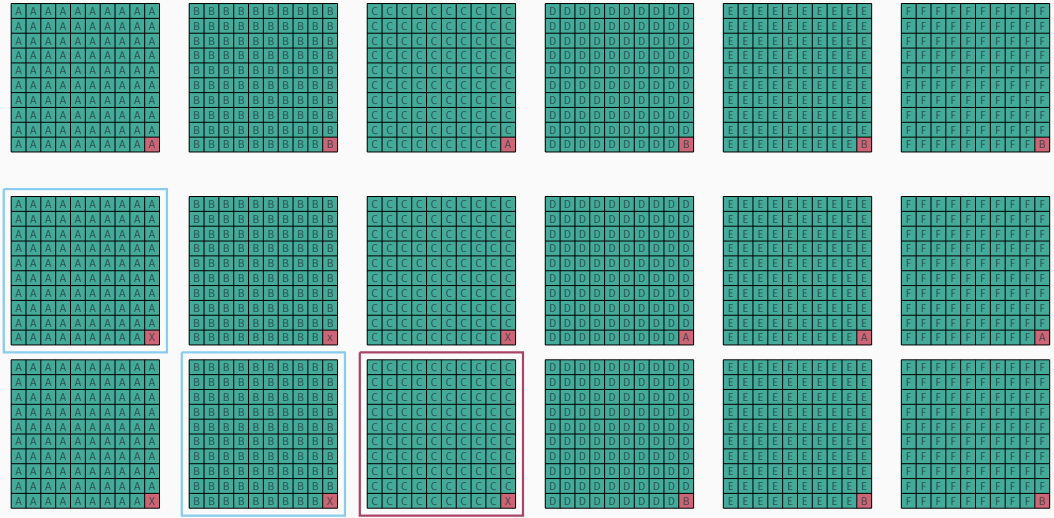
Covert Channel



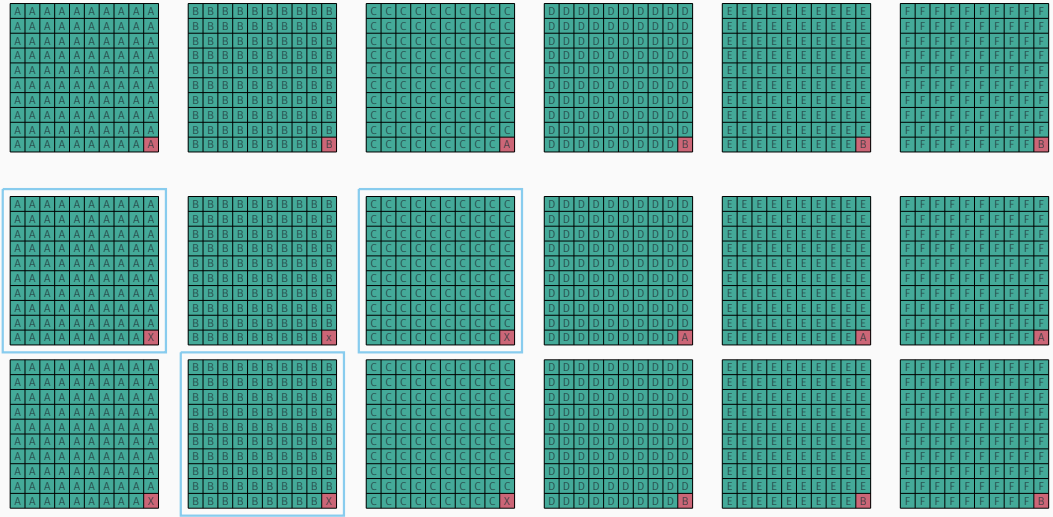
Covert Channel



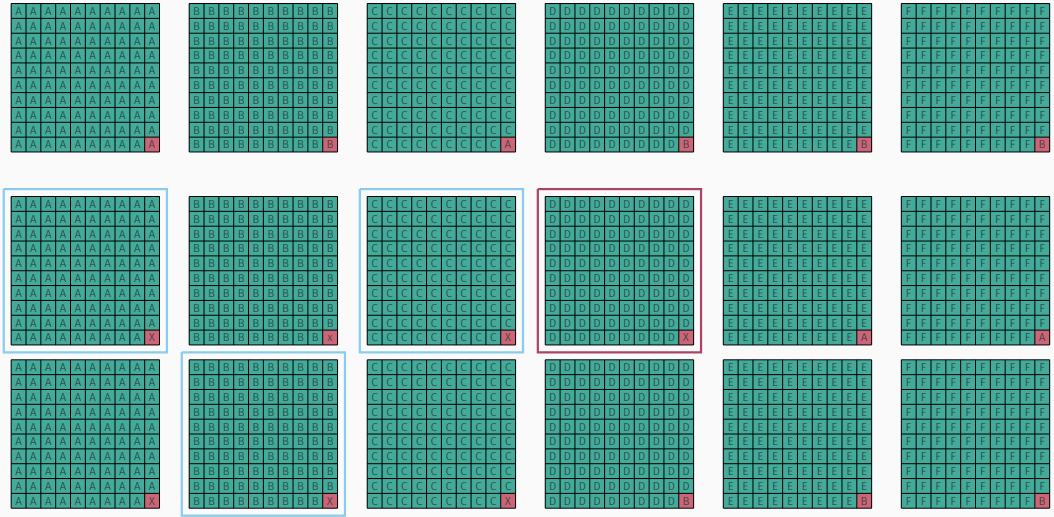
Covert Channel



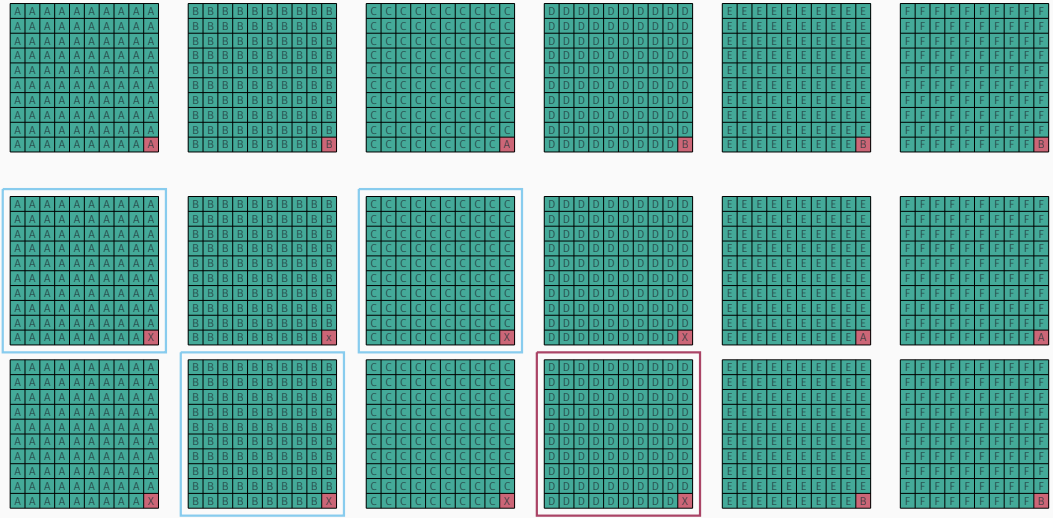
Covert Channel



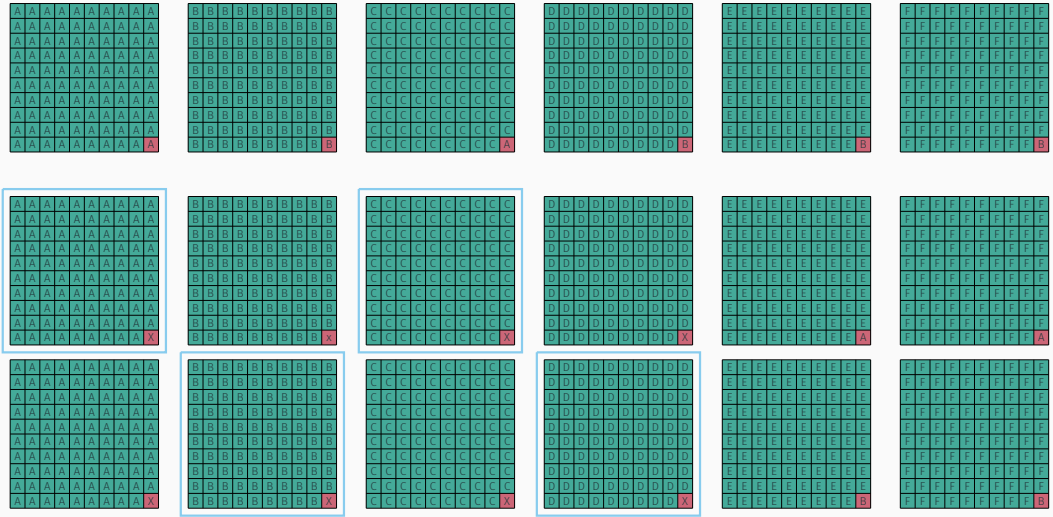
Covert Channel



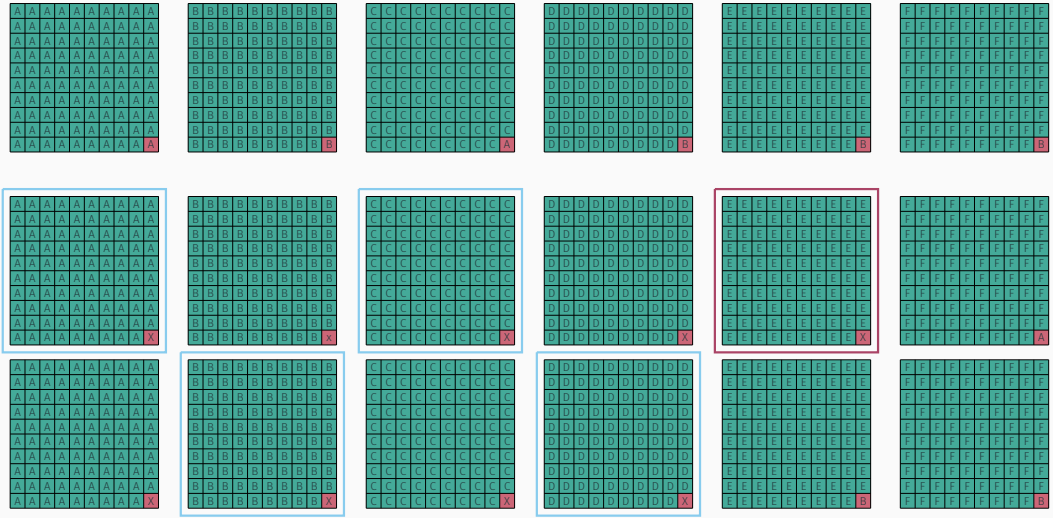
Covert Channel



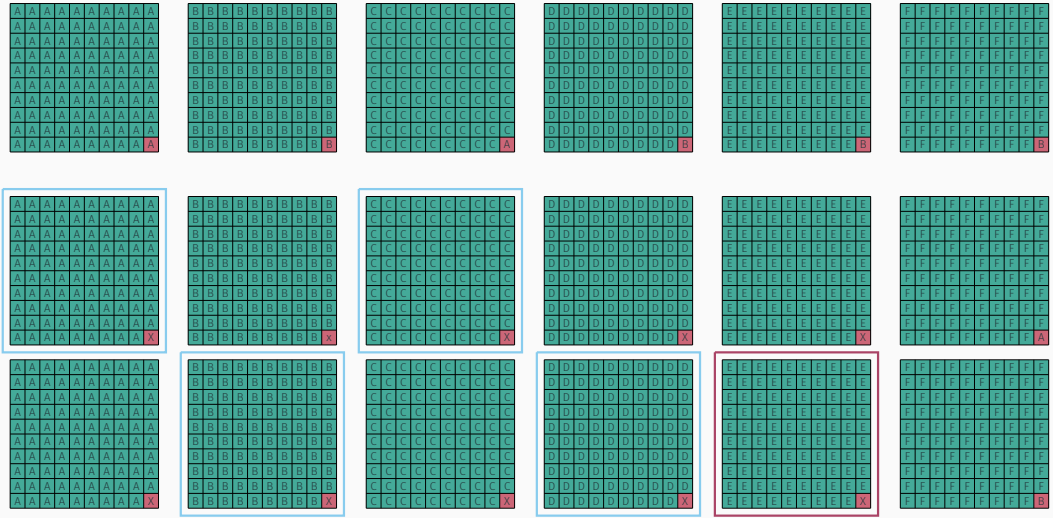
Covert Channel



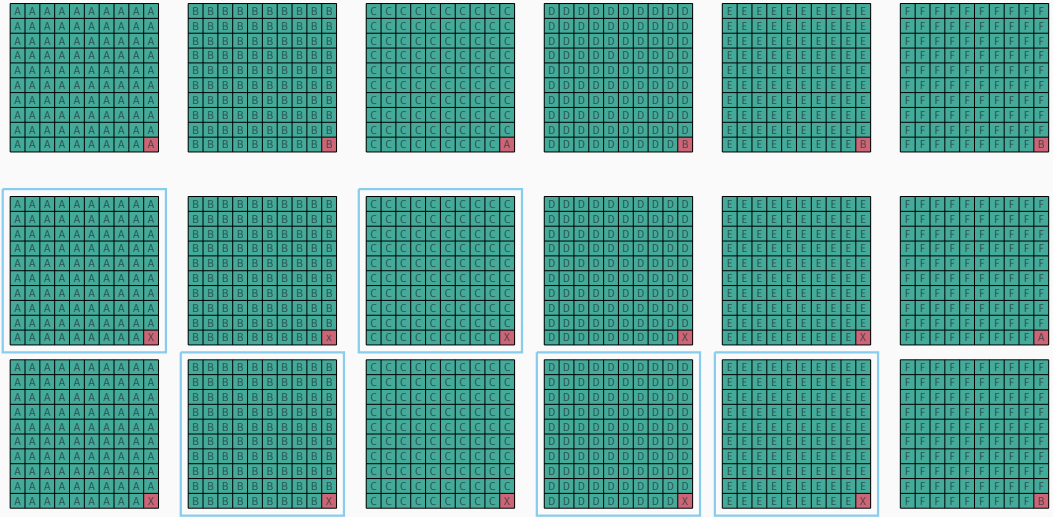
Covert Channel



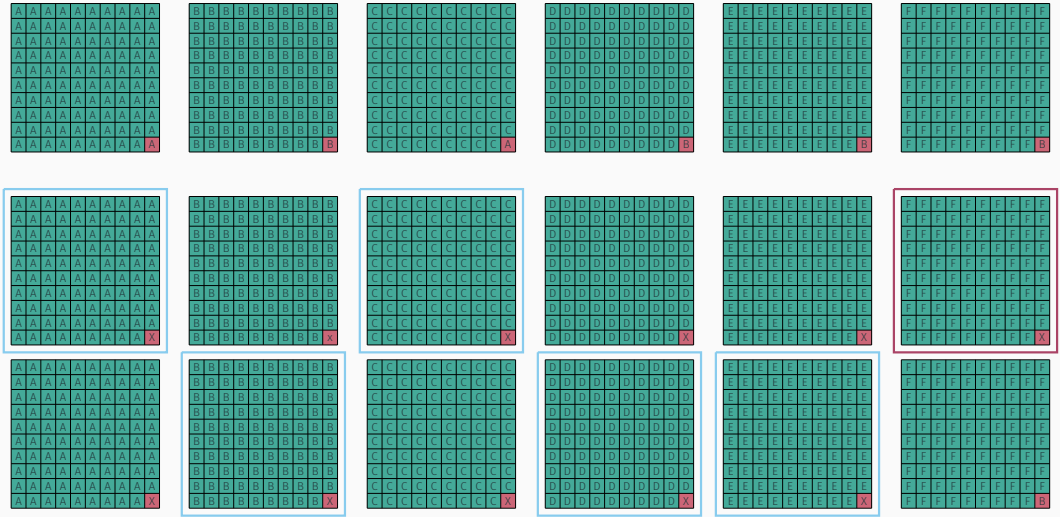
Covert Channel



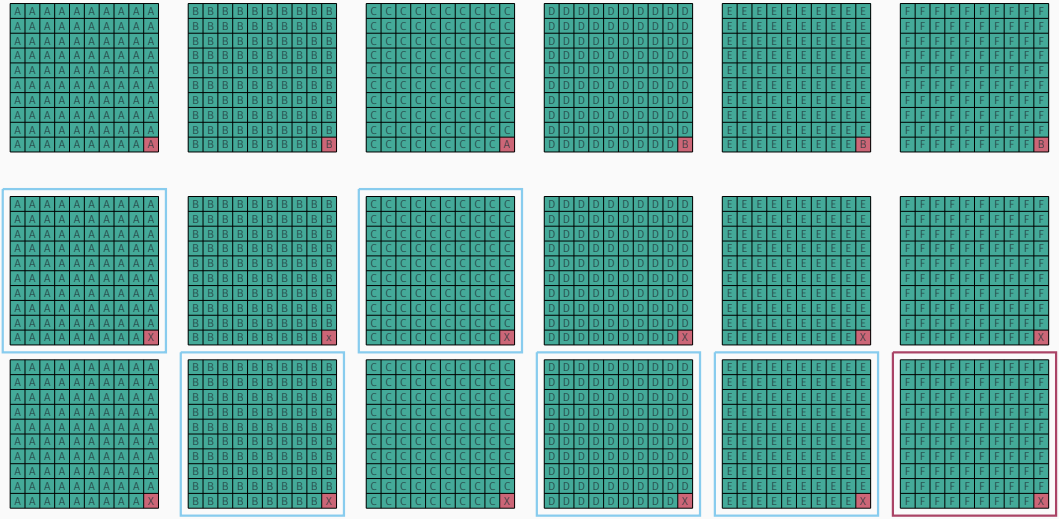
Covert Channel



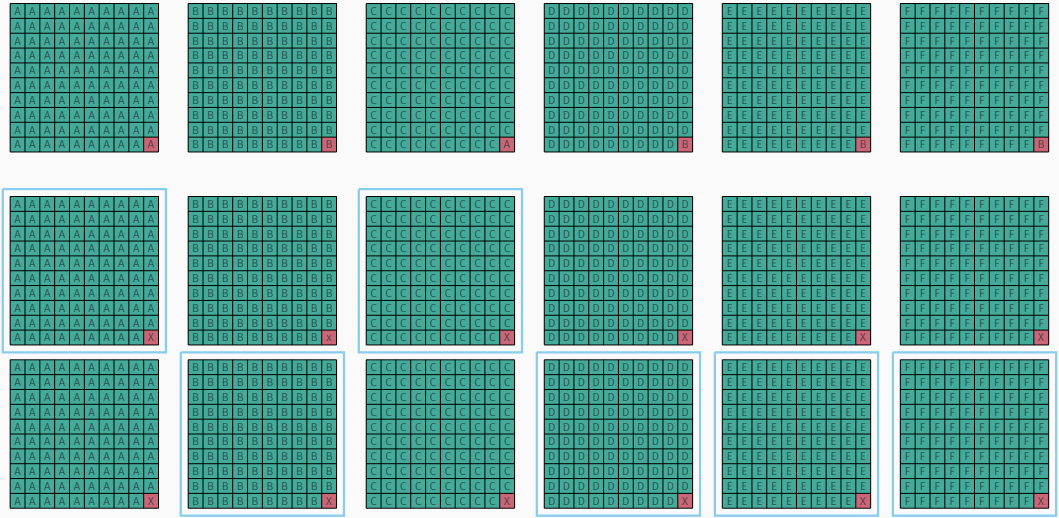
Covert Channel



Covert Channel



Covert Channel



Flip Feng Shui (FFS)

- Deduplizierte Pages liegen physisch an der gleichen Stelle
- Auftretende Speicherfehler betreffen alle Seiten, die physisch an der gleichen Stelle liegen

Flip Feng Shui (FFS)

- Deduplizierte Pages liegen physisch an der gleichen Stelle
- Auftretende Speicherfehler betreffen alle Seiten, die physisch an der gleichen Stelle liegen
- **Idee**: Provozieren von Speicherfehlern an bestimmten Stellen (offsets) in deduplizierten Seiten

Flip Feng Shui (FFS)

- Deduplizierte Pages liegen physisch an der gleichen Stelle
- Auftretende Speicherfehler betreffen alle Seiten, die physisch an der gleichen Stelle liegen
- **Idee**: Provozieren von Speicherfehlern an bestimmten Stellen (offsets) in deduplizierten Seiten
- **Konsequenz**: Umgehen der CoW Policy, effektives Schreiben beliebiger Pages (mit einigen Einschränkungen) [6]

Flip Feng Shui (FFS)

- **Annahme**: Bit Flips sind stabil, d.h. treten bei erneuten Zugriffen wieder an den gleichen Stellen auf
- **Annahme**: Der Inhalt der Page, sowie der Offset, an dem ein Bit Flip auftreten soll und dessen Richtung, sind bekannt

Flip Feng Shui (FFS)

- Suchen nach Speicherstellen, an denen Bits an der richtigen Stelle (Offset) flippen

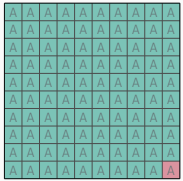
Flip Feng Shui (FFS)

- Suchen nach Speicherstellen, an denen Bits an der richtigen Stelle (Offset) flippen
- Schreiben des Inhalts der (bekannten) Page an die Stelle, bei der der gewünschte Flip auftritt
- Schreiben des Inhalts der (bekannten) Page an eine andere Stelle, sodass beide Seiten an der anfälligen Speicherstelle gemerged werden
- Warten, bis KSM die Seiten gemerged hat
- Erstellen der Zielseite triggern (z.B. Aufrufen eines Programms, etc.)
- Warten bis die Seiten gemerged wurden

Flip Feng Shui (FFS)

- Suchen nach Speicherstellen, an denen Bits an der richtigen Stelle (Offset) flippen
- Schreiben des Inhalts der (bekannten) Page an die Stelle, bei der der gewünschte Flip auftritt
- Schreiben des Inhalts der (bekannten) Page an eine andere Stelle, sodass beide Seiten an der anfälligen Speicherstelle gemerged werden
- Warten, bis KSM die Seiten gemerged hat
- Erstellen der Zielseite triggern (z.B. Aufrufen eines Programms, etc.)
- Warten bis die Seiten gemerged wurden
- Erneutes Triggern des Bit Flips

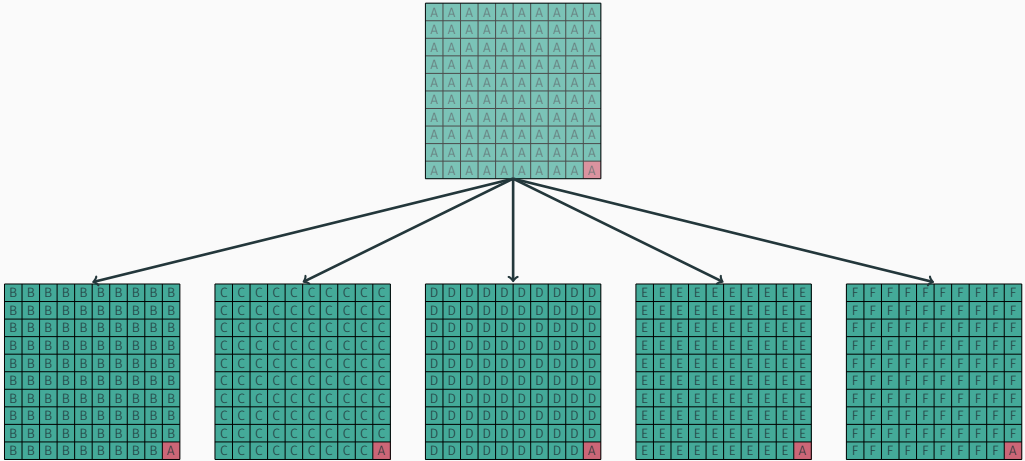
Flip Feng Shui (FFS) – Scannen des Speichers



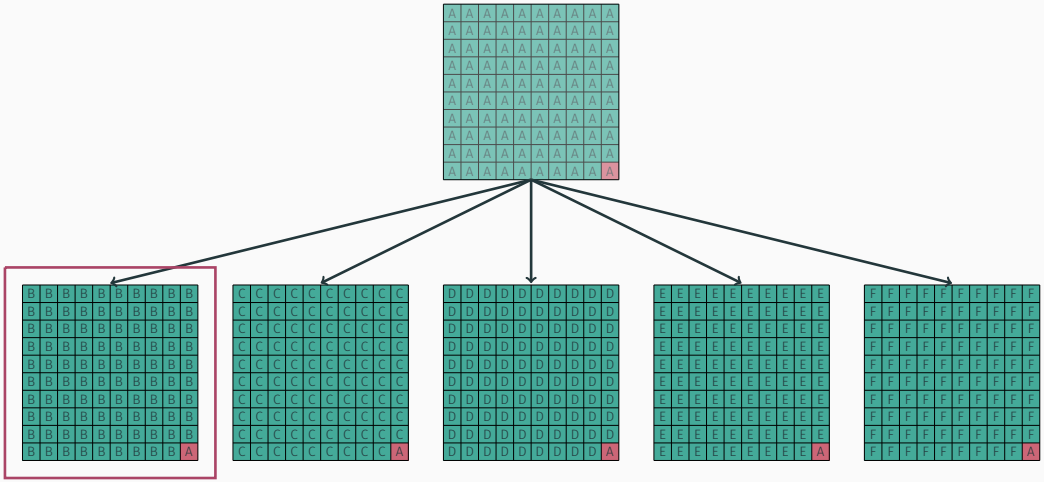
Binäre Darstellung von Zeichen

Zeichen	Hexadezimal	Binär
@	0x40	0010 0000
A	0x41	0010 0001
B	0x42	0010 0010
C	0x43	0010 0011
D	0x44	0010 0100
E	0x45	0010 0101
F	0x46	0010 0110
G	0x47	0010 0111

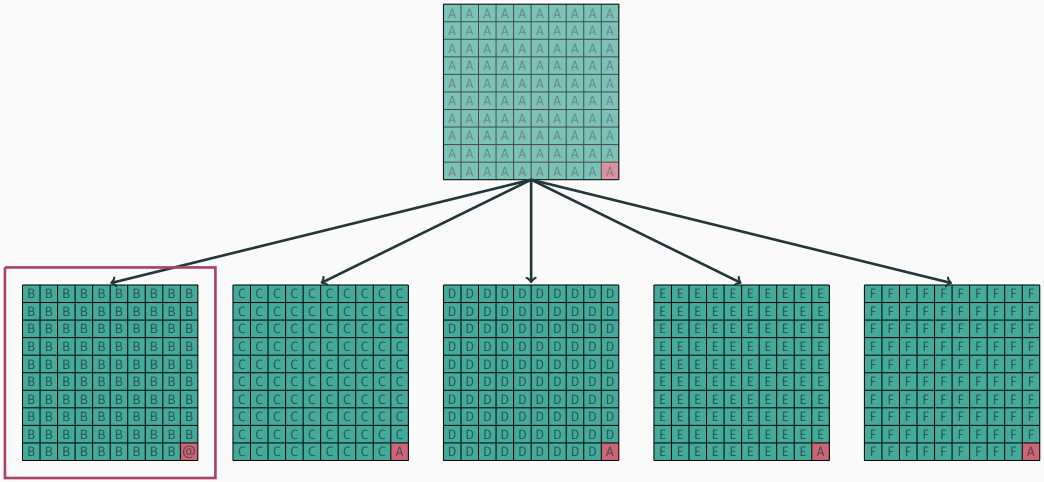
Flip Feng Shui (FFS) – Scannen des Speichers



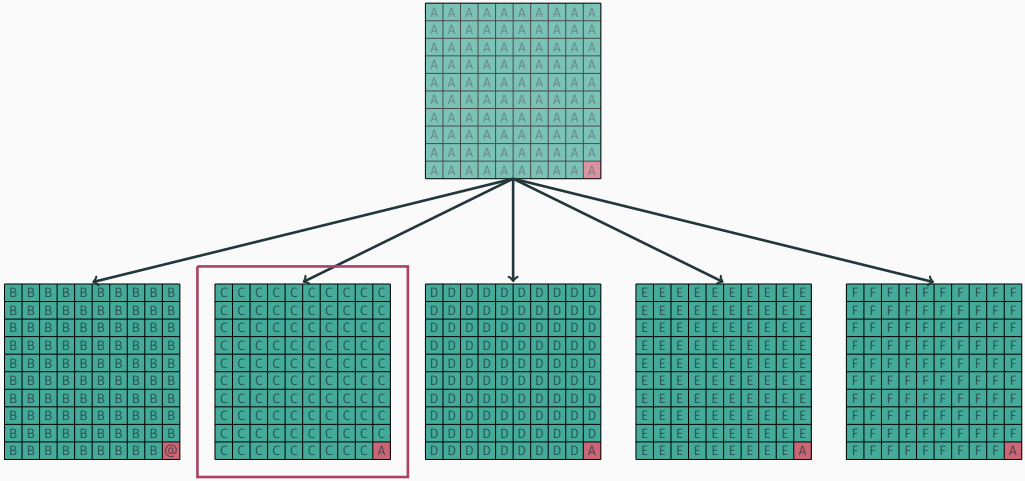
Flip Feng Shui (FFS) – Scannen des Speichers



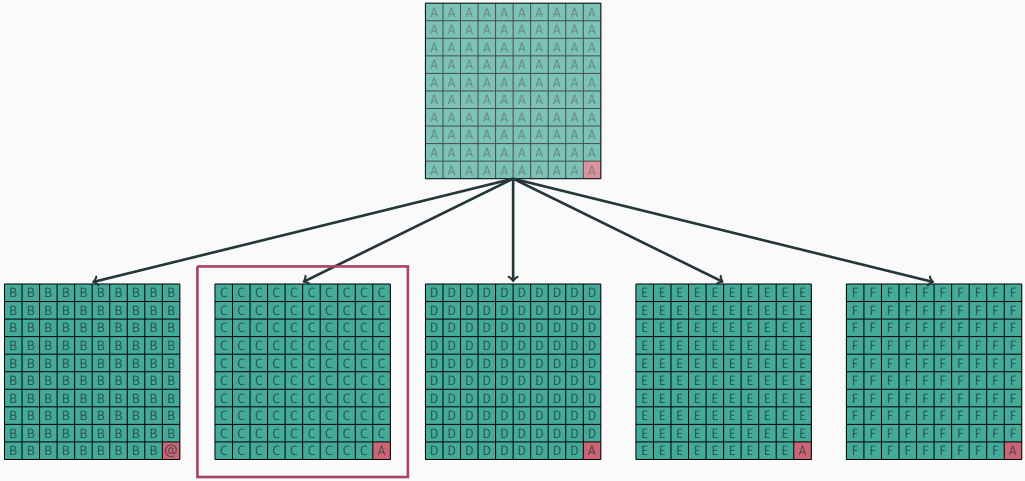
Flip Feng Shui (FFS) – Scannen des Speichers



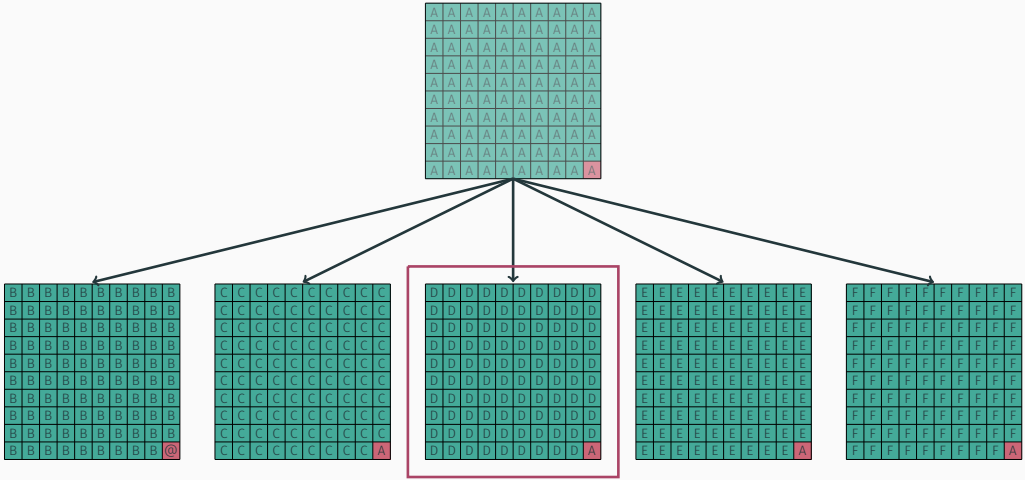
Flip Feng Shui (FFS) – Scannen des Speichers



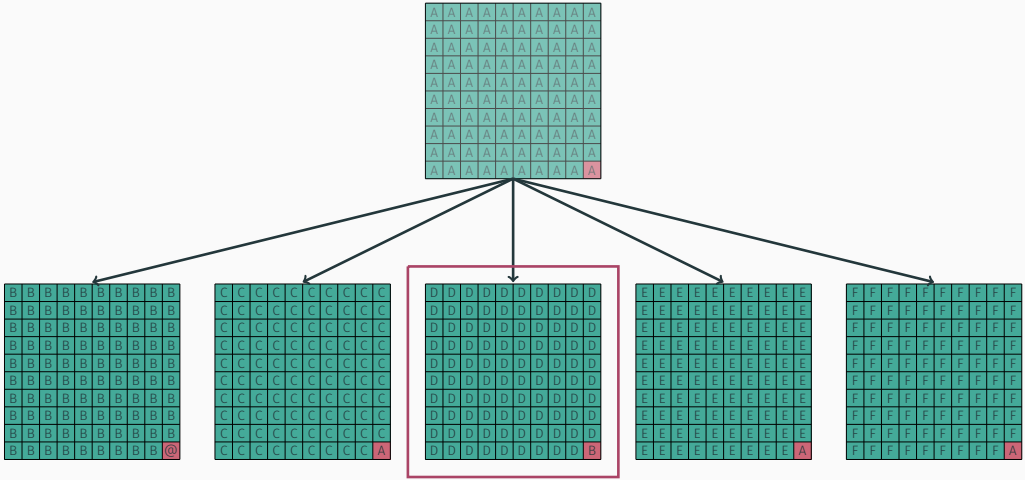
Flip Feng Shui (FFS) – Scannen des Speichers



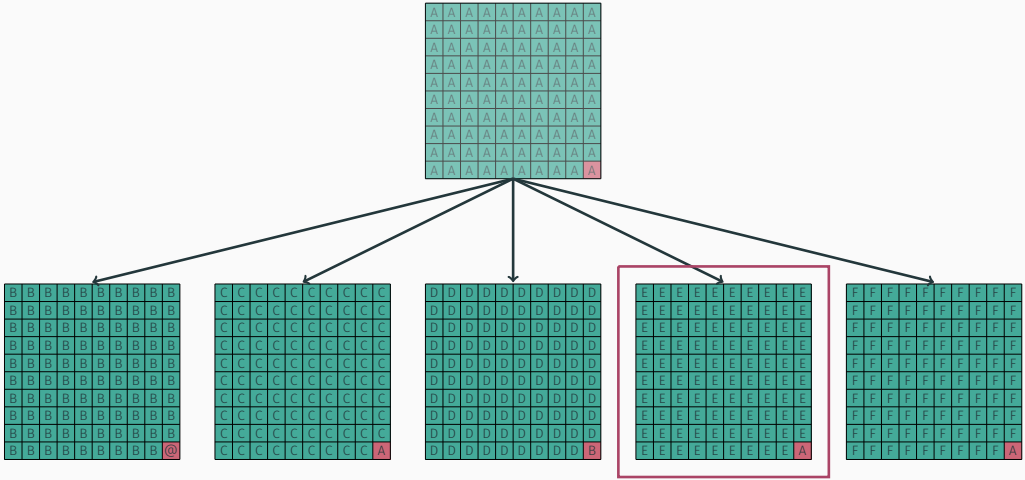
Flip Feng Shui (FFS) – Scannen des Speichers



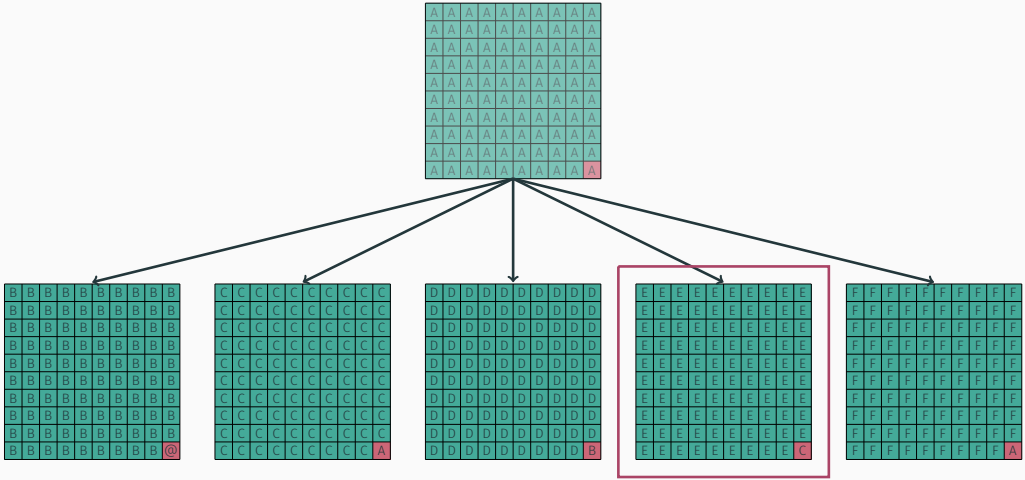
Flip Feng Shui (FFS) – Scannen des Speichers



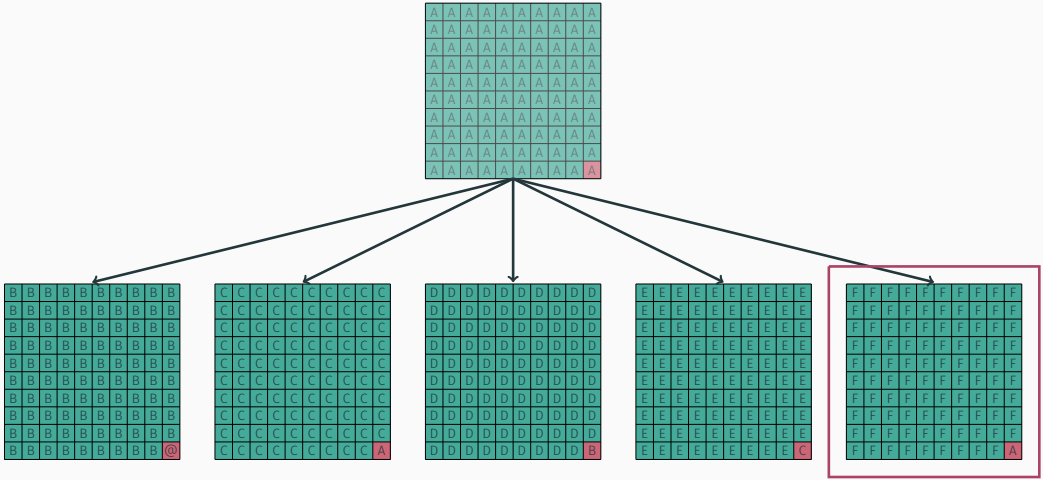
Flip Feng Shui (FFS) – Scannen des Speichers



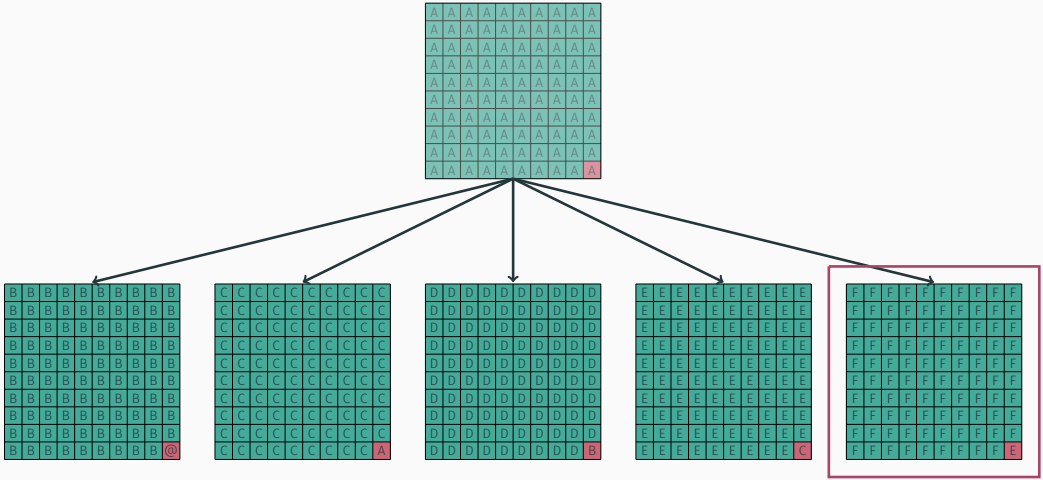
Flip Feng Shui (FFS) – Scannen des Speichers



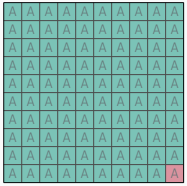
Flip Feng Shui (FFS) – Scannen des Speichers

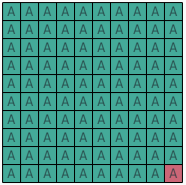


Flip Feng Shui (FFS) – Scannen des Speichers

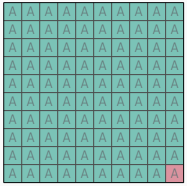


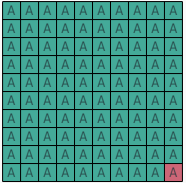
Flip Feng Shui (FFS) – Mergen der Seiten und Exploit



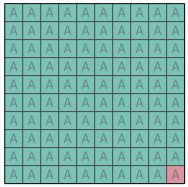


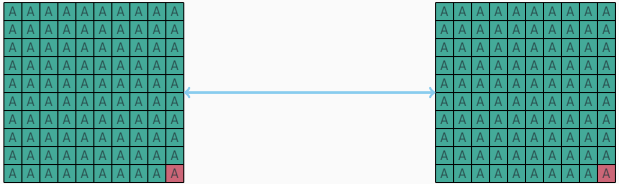
Flip Feng Shui (FFS) – Mergen der Seiten und Exploit



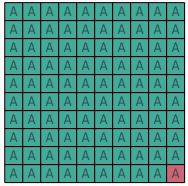


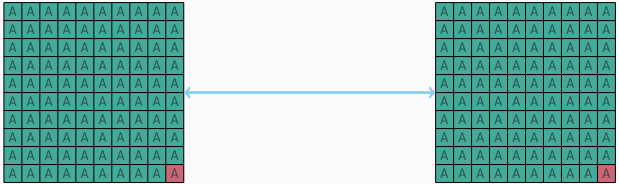
Flip Feng Shui (FFS) – Mergen der Seiten und Exploit



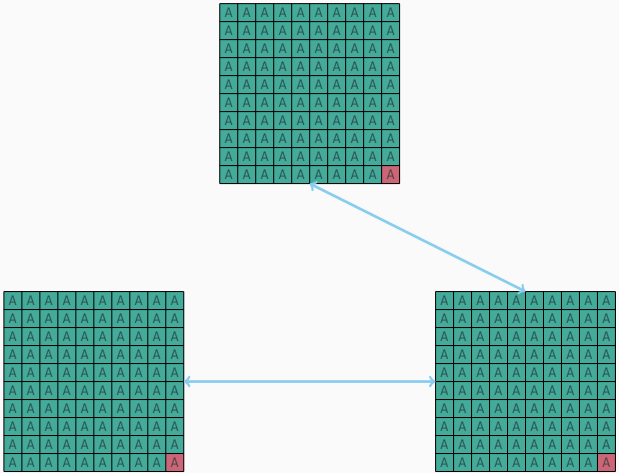


Flip Feng Shui (FFS) – Mergen der Seiten und Exploit

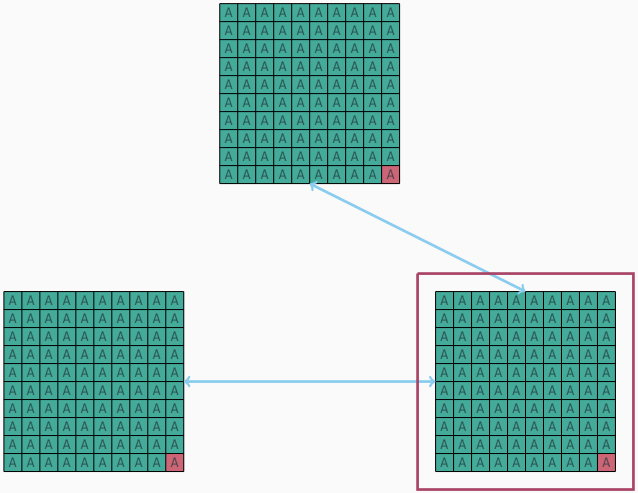




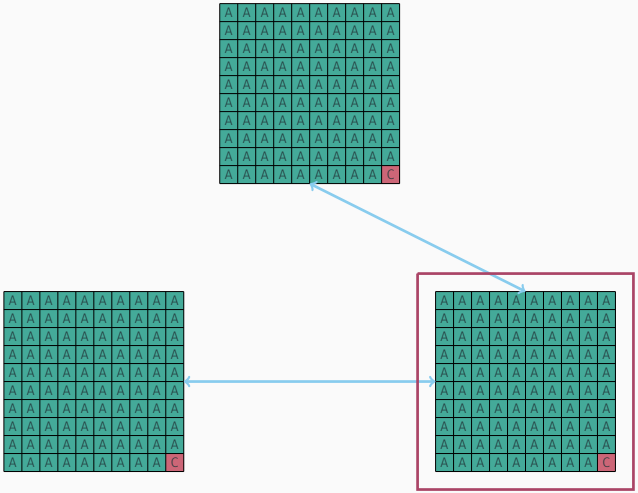
Flip Feng Shui (FFS) – Mergen der Seiten und Exploit



Flip Feng Shui (FFS) – Mergen der Seiten und Exploit



Flip Feng Shui (FFS) – Mergen der Seiten und Exploit



Demo 3

Flip Feng Shui (FFS) mit FFSTOOL

Fazit

Zusammenfassung

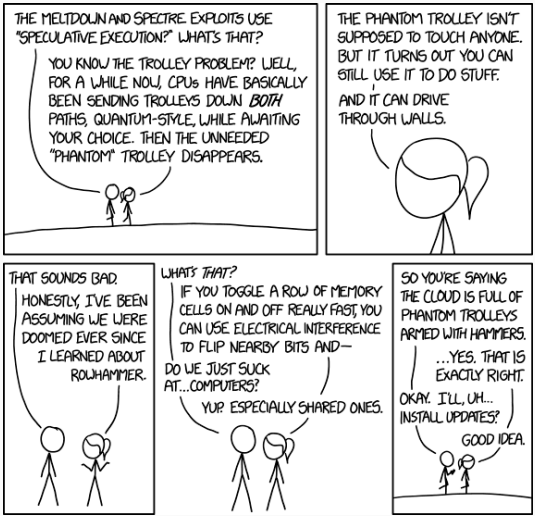


Abbildung von XKCD

Fragen?

Referenzen i

[1] Erik Bosman u. a. „Dedup Est Machina: Memory Deduplication as an Advanced Exploitation Vector“. In: *S&P*. Pwnie Award for Most Innovative Research. Mai 2016. URL: Paper=https://download.vusec.net/papers/dedup-est-machina_sp16.pdf%20Web=https://www.vusec.net/projects/dedup-est-machina%20Press=https://goo.gl/ogBXTm.

[2] Pietro Frigo u. a. „TRRespass: Exploiting the Many Sides of Target Row Refresh“. In: *S&P*. Best Paper Award. Mai 2020. URL: https://download.vusec.net/papers/trrespass_sp20.pdf.

[3] Martin Heckel. „RAEAX – Rowhammer Amplification by Execution of Additional X86 instructions“. Hof University of Applied Sciences, 2021, S. 1–40.

Referenzen ii

[4] Patrick Jattke u. a. „BLACKSMITH: Rowhammering in the Frequency Domain“. In: *IEEE S&P '22*. https://comsec.ethz.ch/wp-content/files/blacksmith_sp22.pdf. Nov. 2021.

[5] Yoongu Kim u. a. „Flipping Bits in Memory without Accessing Them: An Experimental Study of DRAM Disturbance Errors“. In: *SIGARCH Comput. Archit. News* 42.3 (Juni 2014), S. 361–372. ISSN: 0163-5964. DOI: 10.1145/2678373.2665726. URL: https://doi.org/10.1145/2678373.2665726.

Referenzen iii

[6] Kaveh Razavi u. a. „Flip Feng Shui: Hammering a Needle in the Software Stack“. In: *USENIX Security*. Juni 2016. URL: https://download.vusec.net/papers/flip-feng-shui_sec16.pdf.

[7] Mark Seaborn und Thomas Dullien. *Exploiting the DRAM rowhammer bug to gain kernel privileges*. 2015. URL: https://www.cs.umd.edu/class/fall2019/cmsc818O/papers/rowhammer-kernel.pdf (besucht am 16. 11. 2020).

[8] Victor van der Veen u. a. „Drammer: Deterministic Rowhammer Attacks on Mobile Platforms“. In: *CCS*. Pwnie Award for Best Privilege Escalation Bug, Android Security Reward, CSAW Best Paper Award, DCSR Paper Award. Okt. 2016. URL: https://vvdveen.com/publications/drammer.pdf.