Looking Back at 10 Years of Rowhammer Exploits

hardwear.io
Hardware Security Conference and Training

hardwear.io NL

25th October 2024

Jonas Juffinger
CoreSec, Graz University of Technology

Andreas Kogler
CoreSec Alumni, Graz University of Technology

Who Are We?

Whoami

- **Jonas Juffinger**
- PhD Candidate, IAIK, Graz University of Technology
- soon™ on the job market
- @notimaginary_
- mail@jonasjuffinger.com

Whoami

- **Andreas Kogler**
- PhD Alumni, IAIK, Graz University of Technology
- @0xhilbert
- andreas.kogler.0x+hwio@gmail.com

2014

Flipping bits in memory without accessing them: An experimental study of DRAM disturbance errors

Kim, Y., Daly, R., Kim, J., Fallin, C., Lee, J.H., Lee, D., Wilkerson, C., Lai, K. and Mutlu, O.

SIGARCH, 2014 [13]

Rowhammer [13]

- Hardware **fault** of the DRAM
- Frequent accesses flip bits in neighboring rows


```
1 for (i = 0; i < N; ++i) {
2   for (j = 0; j < M; ++j) {
3     aggressor[j];
4     flush(&aggressor[j]);
5   }
6 }
```

- 2014

2014 2015

Exploiting the DRAM rowhammer bug to gain kernel privileges

Mark Seaborn and Thomas Dullien

Google Project Zero, 2015 [24]

Google Native Client (NaCL) Exploit [24]

- Native code browser sandbox
- Allows only 32-byte **aligned** indirect jumps
- Allows instruction only at 32-byte aligned locations

```
1 andl $~31, %eax   // Truncate to 32 bits and mask to be 32-byte-aligned.
2 addq %r15, %rax   // Add %r15, the sandbox base address.
3 jmp *%rax         // Indirect jump
```

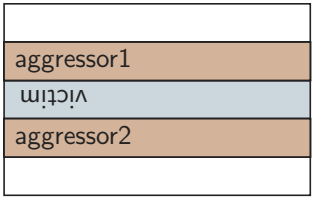
Google Native Client (NaCL) Exploit

```
1 20ea0:  48 b8 0f 05 eb 0c f4 f4 f4 f4  movabs $0xf4f4f4f40ceb050f,%rax
```

- Hide a SYSCALL instruction (`0f 05`) at address `0x20ea2`
- Spray 250MB of indirect jumps using `dyncode_create()`
- Disable jump alignment with Rowhammer
- **Check** if an exploitable bit flip happened

Double Sided Rowhammer

Double Sided Rowhammer [24]



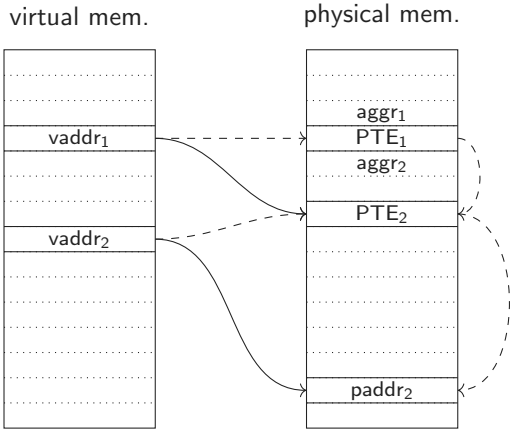

```
1 for (i = 0; i < N; ++i) {
2   *aggressor1;
3   *aggressor2;
4   flush(aggressor1);
5   flush(aggressor2);
6 }
7
```

Page Table Exploit [24]

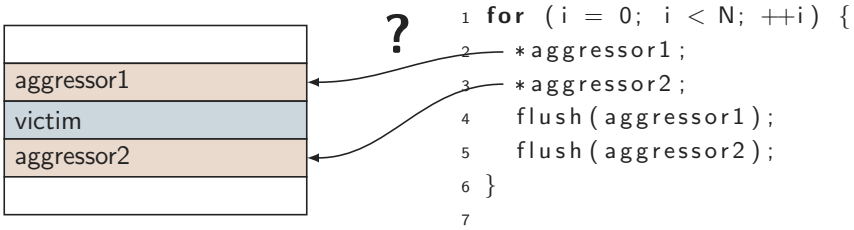
Page Table Exploit [24]

- Map **shared memory** to spray page tables
- Hammer page table
- Flip bit in page table entry
- Access to physical memory

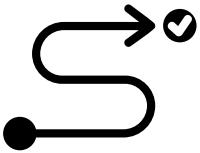
Page Table Exploit [24]



About Physical Address Information



About Physical Address Information



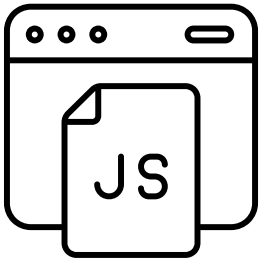
- Physical address → row mapping reverse engineered for many architectures [21]
- Until 2015 physical address through `/proc/self/pagemap`
- . . . but quickly closed
- new solutions in new exploits

Rowhammer.js: A Remote Software-Induced Fault Attack in JavaScript

Gruss, D., Maurice, C. and Mangard, S.

DIMVA, 2016 [7]

Rowhammer.js [7]



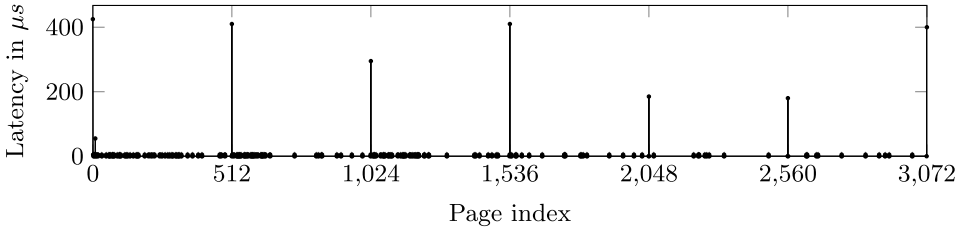
- No host physical address information

→ Browser maps arrays with 2MB pages

→ Double sided hammering

- Cache eviction

Rowhammer.js [7]



2014 2015

2014 2015 2016

Dedup Est Machina: Memory Deduplication as an Advanced Exploitation Vector

Bosman, E., Razavi, K., Bos, H. and Giuffrida, C.

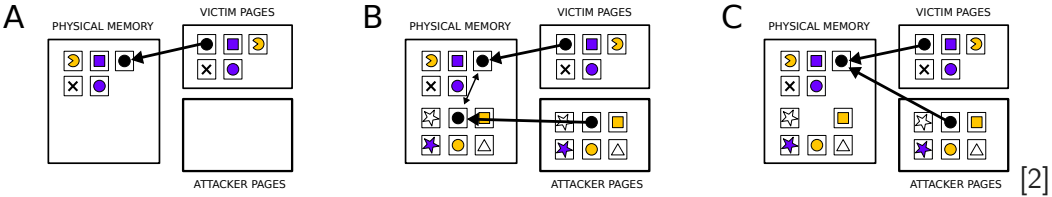
S&P, 2016 [2]

Dedup Est Machina [2]

JavaScript Browser (Edge) Exploit

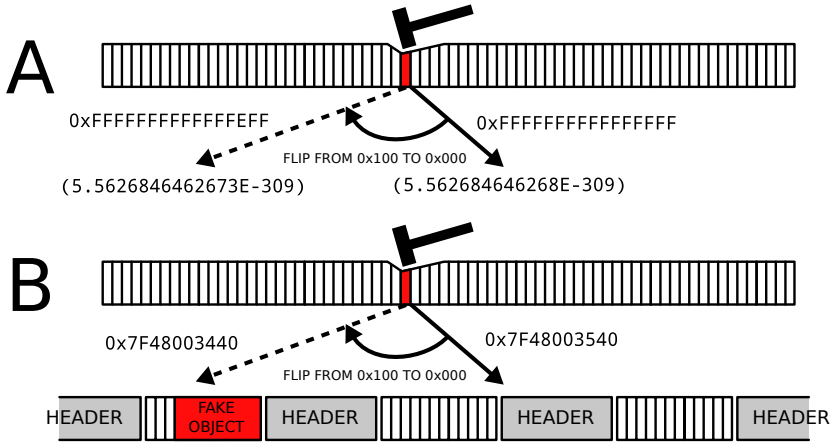
- Exploits page deduplication to leak pointers
- Create counterfeit `Uint8Array` object
- Pivot pointer to object using Rowhammer

Page Deduplication



[2]

Dedup Est Machina [2]

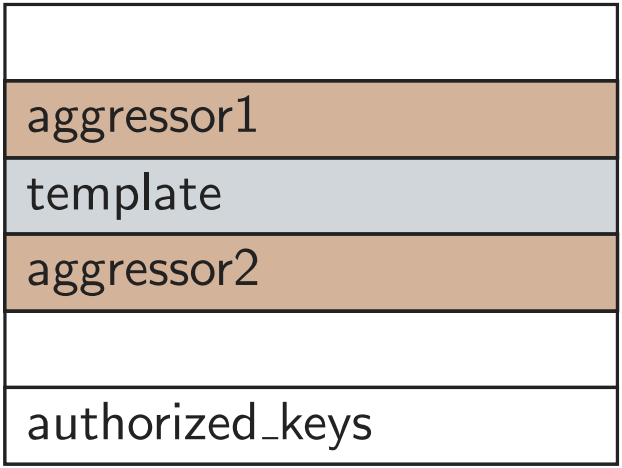


Flip Feng Shui: Hammering a Needle in the Software Stack

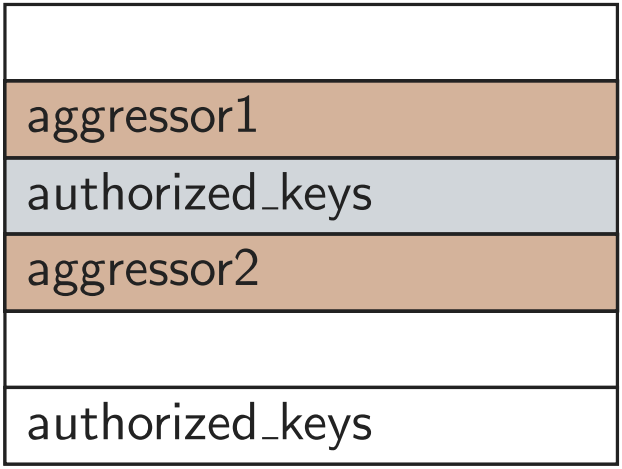
Razavi, K., Gras, B., Bosman, E., Preneel, B., Giuffrida, C. and Bos, H.

USENIX Security, 2016 [22]

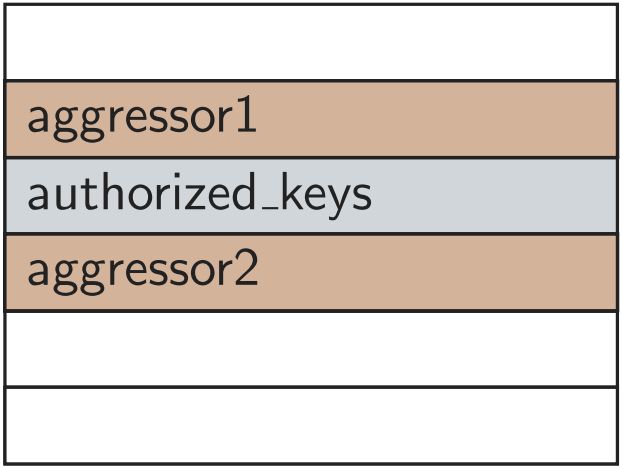
KSM Assisted Hammering [22]



KSM Assisted Hammering [22]



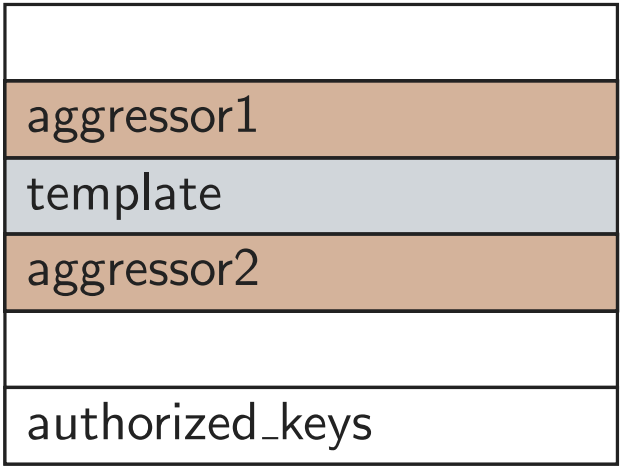
KSM Assisted Hammering [22]



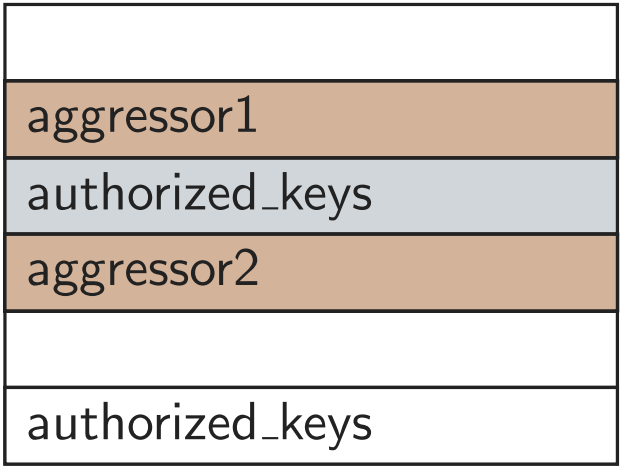
KSM Assisted Hammering [22]


Corporate needs you to find the differences
between this picture and this picture.
Kernel Samepage Merging
They're the same picture.
imgflip.com

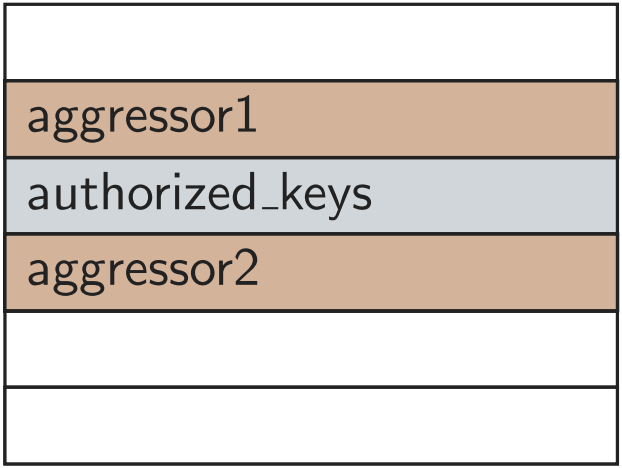
KSM Assisted Hammering [22]



KSM Assisted Hammering [22]



KSM Assisted Hammering [22]



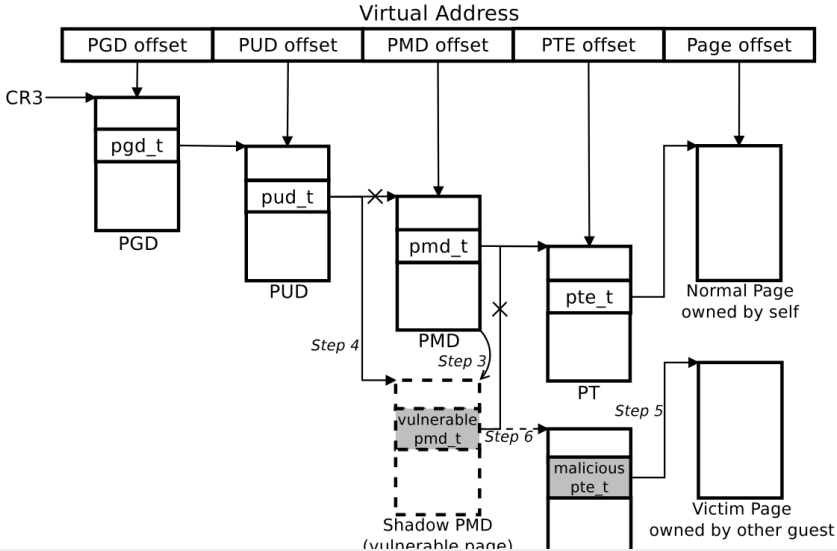
KSM Assisted Hammering [22]

One Bit Flips, One Cloud Flops: Cross-VM Row Hammer Attacks and Privilege Escalation

Xiao, Y., Zhang, X., Zhang, Y. and Teodorescu, R.

USENIX Security, 2016 [28]

One Bit Flips, One Cloud Flops [28]



Drammer: Deterministic Rowhammer Attacks on Mobile Platforms

Van Der Veen, V., Fratantonio, Y., Lindorfer, M., Gruss, D., Maurice, C.

CCS, 2016 [27]

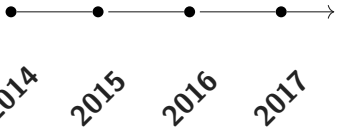
Attack ARM Android [27]

- Memory Templating via DMA (ION) uncached contiguous memory
- Phys Feng Shui on the buddy allocator

$\rightarrow$ Self referencing page table

2014 2015 2016


STAGEFRIGHT

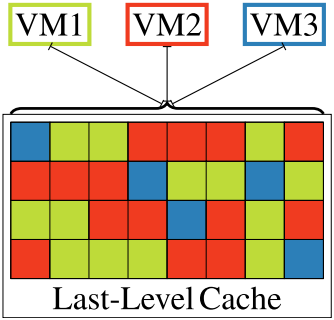

2014
2015
2016
2017

When Good Protections Go Bad: Exploiting Anti-DoS Measures to Accelerate Rowhammer Attacks

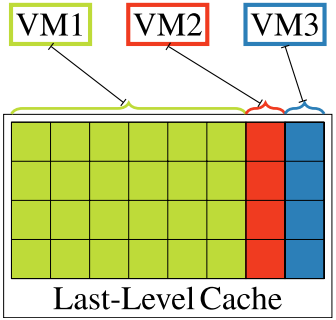
Aga, M.T., Aweke, Z.B. and Austin, T.

HOST, 2017 [1]

When Good Protections Go Bad [1]

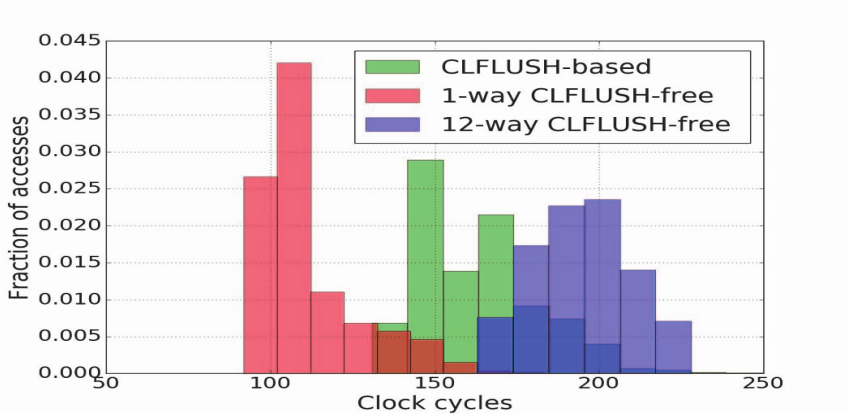


(a) CAT disabled

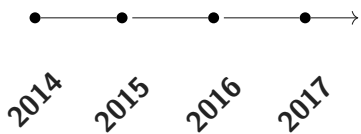


(b) CAT enabled

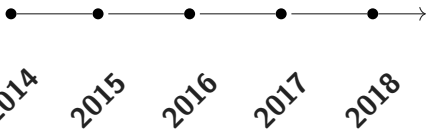
When Good Protections Go Bad [1]




STAGEFRIGHT


2014
2015
2016
2017


STAGEFRIGHT

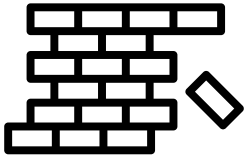

2014
2015
2016
2017
2018

Another Flip in the Wall of Rowhammer Defenses

Gruss, D., Lipp, M., Schwarz, M., Genkin, D., Juffinger, J., O'Connell, S., Schoechl, W. and Yarom, Y.

S&P, 2018 [6]

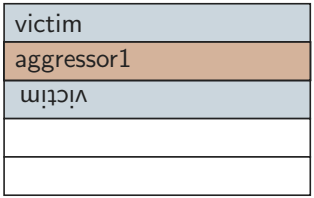
Another Flip in the Wall of Rowhammer Defenses [6]



- One Location Rowhammer
- Hammer from SGX
- Attack setuid binaries, e.g., sudo

One-Location Rowhammer

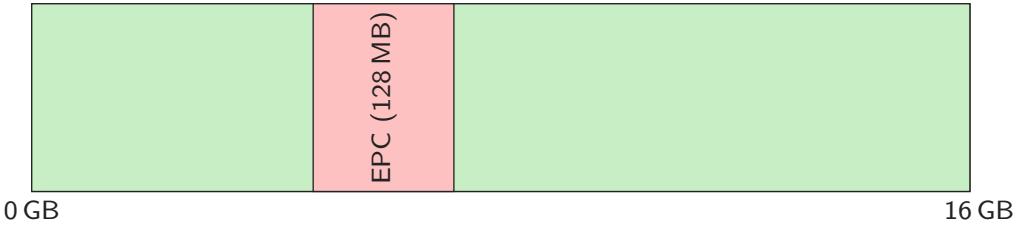
- Access only a single row
- Row immediately closed due to a closed-row policy [1]




```
1 for (i = 0; i < N; ++i) {
2   *aggressor1;
3   flush(aggressor1);
4 }
5
```

[1] To be continued in 2024 ...

SGX Encrypted Memory



Rowhammer + SGX = Cheap Denial of Service [6, 8]

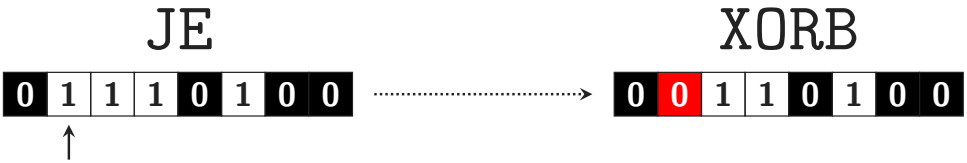
Hammer Non-EPC Memory

Opcode Flipping - setuid binaries

- Many applications perform actions as root
- They can be used by unprivileged users as well
- Implicitly: e.g., `ping` or `mount`
- Explicitly: `sudo`
- Target `sudo` (easy to exploit)

Opcode Flipping - Conditional Jump

Opcode Flipping - Conditional Jump

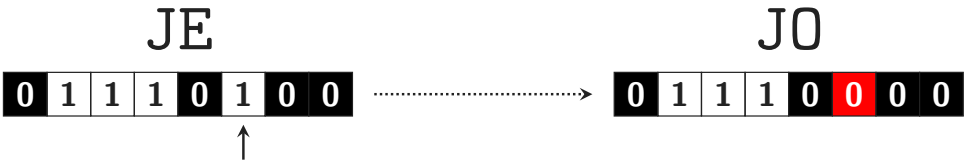


Opcode Flipping - Conditional Jump

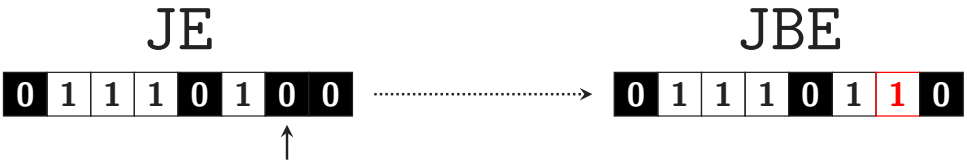
Opcode Flipping - Conditional Jump

Opcode Flipping - Conditional Jump

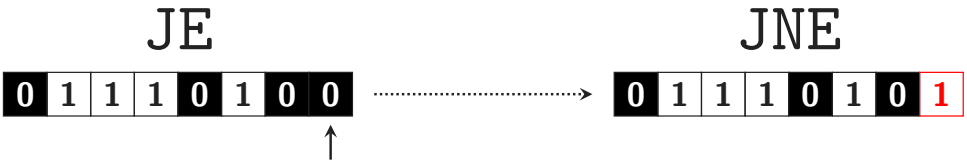
Opcode Flipping - Conditional Jump



Opcode Flipping - Conditional Jump



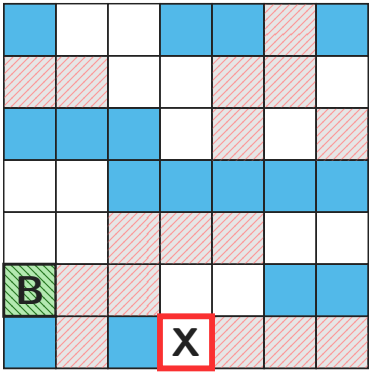
Opcode Flipping - Conditional Jump



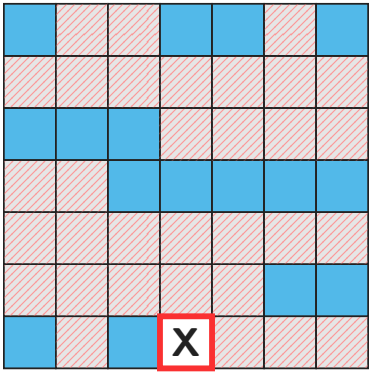
How to get the target virtual page to the target physical location?

Memory Waylaying

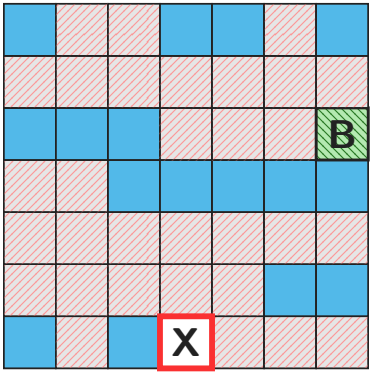
(1) Start



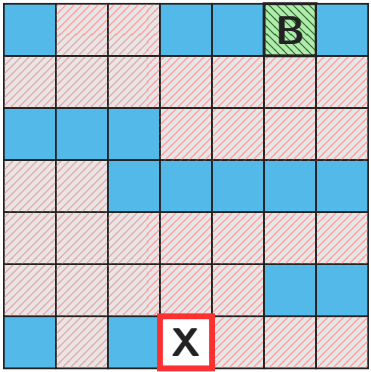
(2) Evict Page Cache



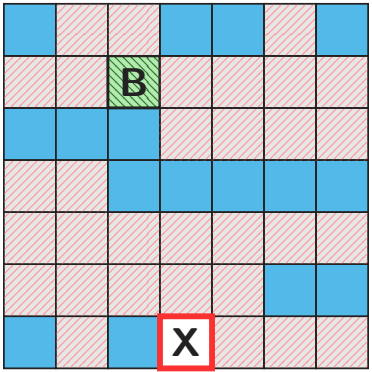
(3) Access Binary



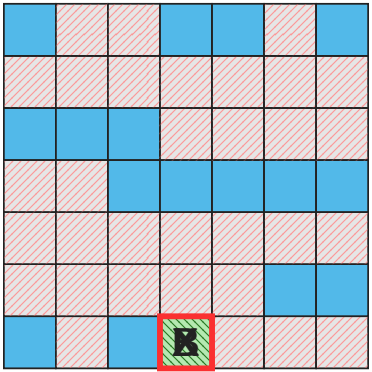
(4) Evict + Access



(5) Evict + Access



(6) Stop if target reached



SGX + One-location Hammering + Opcode Flipping = Undetectable Exploit

Grand Pwning Unit: Accelerating Microarchitectural Attacks with the GPU

Frigo, P., Giuffrida, C., Bos, H. and Razavi, K.

S&P, 2018 [6]

Attack the Web Browser from the Grand Pwning Unit (GPU) [4]

First web browser exploit on ARM using an integrated GPU

- Loading texture causes predictable memory accesses from the GPU
- Fast cache eviction because GPU has small deterministic caches
- Possible from JavaScript using WebGL
- And WebGL also supplies high resolution timers

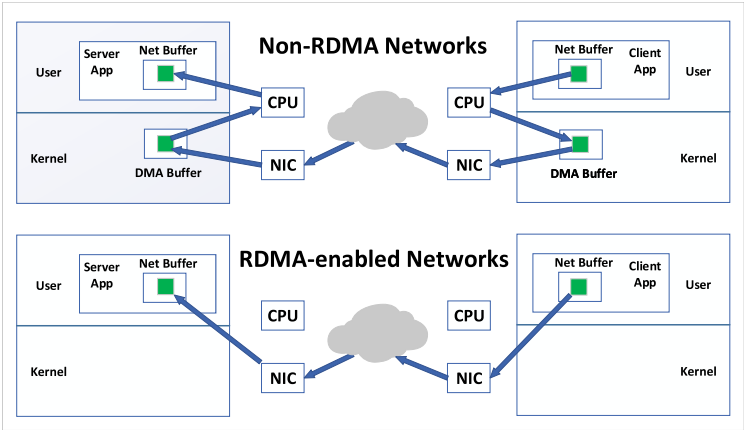
→ Arbitrary read/write

Throwhammer: Rowhammer Attacks over the Network and Defenses

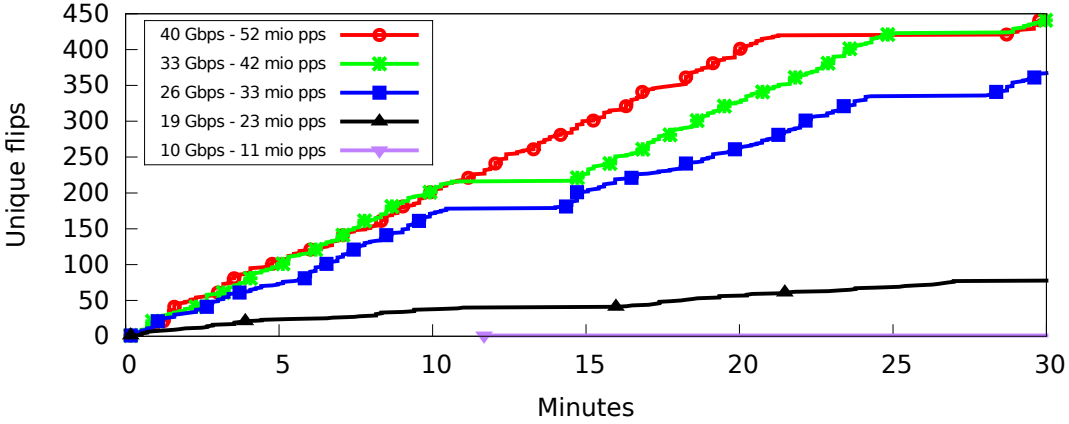
Tatar, A., Konoth, R.K., Athanasopoulos, E., Giuffrida, C., Bos, H. and Razavi, K.

USENIX ATC, 2018 [25]

Throwhammer [25]



Throwhammer [25]

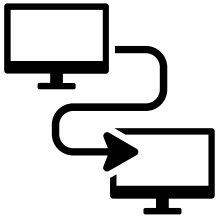


Nethammer: Inducing Rowhammer Faults through Network Requests

Lipp, M., Schwarz, M., Raab, L., Lamster, L., Aga, M.T., Maurice, C. and Gruss, D.

Euro S&P Workshop, 2020 [16]

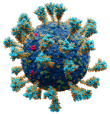
Nethammer [16]

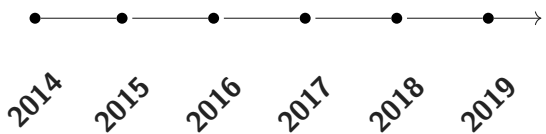


- Network handling code touches packages multiple times
- This hammers the memory if a single cache way is available (Intel CAT)
- $500\,\mathrm{Mbit\,s^{-1}}$ sufficient
- No targeted exploitation (DoS)

2014 2015 2016 2017 2018


STAGEFRIGHT



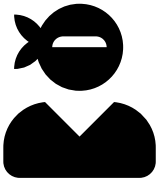

2014
2015
2016
2017
2018
2019

Exploiting Correcting Codes: On the Effectiveness of ECC Memory Against Rowhammer Attacks

Cojocar, L., Razavi, K., Giuffrida, C. and Bos, H.

S&P, 2019 [3]

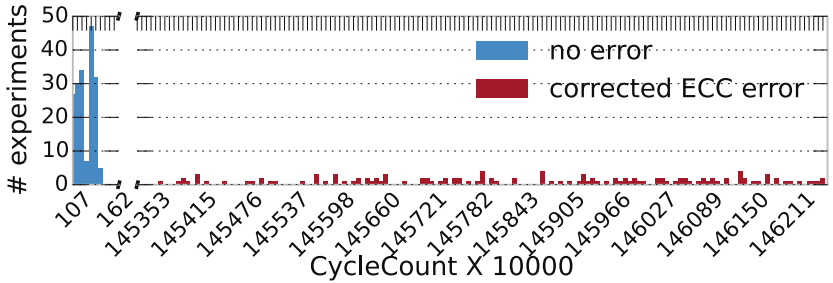
ECCploit [3]



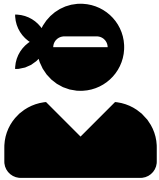
- Goal: Uncorrected and undetected Rowhammer bit flips

1. Reverse engineered ECC functions
2. Collect single bit flips using a correction time side-channel

ECCploit [3]


50
40
30
20
10
0
experiments
no error
corrected ECC error
107
162
145353
145415
145476
145537
145598
145660
145721
145782
145843
145905
145966
146027
146089
146150
146211
CycleCount X 10000

ECCploit [3]

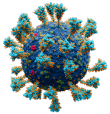


- Goal: Uncorrected and undetected Rowhammer bit flips

1. Reverse engineered ECC functions
2. Collect single bit flips using a correction time side-channel
3. Combine them for undetected bit flips

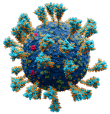
- Exploit page tables, RSA, sudo


STAGEFRIGHT




2014
2015
2016
2017
2018
2019


STAGEFRIGHT






zoom

2014
2015
2016
2017
2018
2019
2020

RAMBleed: Reading Bits in Memory Without Accessing Them

Kwong, A., Genkin, D., Gruss, D. and Yarom, Y.

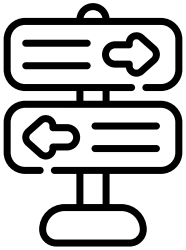
S&P, 2020 [15]

RAMbleed [15]

Exploit Rowhammer to directly leak data

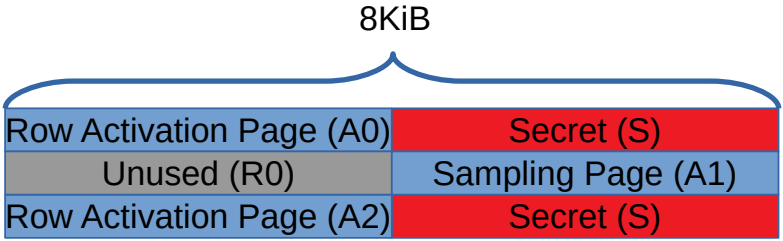
Rowhammer's Data Dependency

Rowhammer's Data Dependency



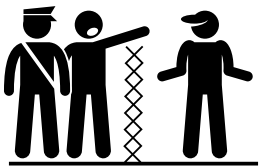
- Cells have only **one** flip direction
- Cells only flip to the value of neighboring cells
- $0 \rightarrow 1$ if neighbors are 1
- $1 \rightarrow 0$ if neighbors are 0

RAMbleed [15]

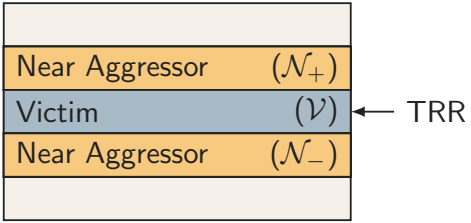

8KiB
Row Activation Page (A0)
Secret (S)
Unused (R0)
Sampling Page (A1)
Row Activation Page (A2)
Secret (S)

Target Row Refresh (TRR)

Target Row Refresh (TRR)



- Additional victim refreshes
- Standardized in LPDDR4x

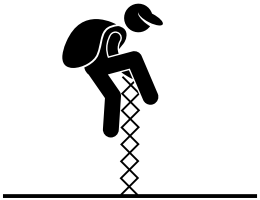


TRRespass: Exploiting the Many Sides of Target Row Refresh

Frigo, P., Vannacc, E., Hassan, H., Van Der Veen, V., Mutlu, O., Giuffrida, C., Bos, H. and Razavi, K.

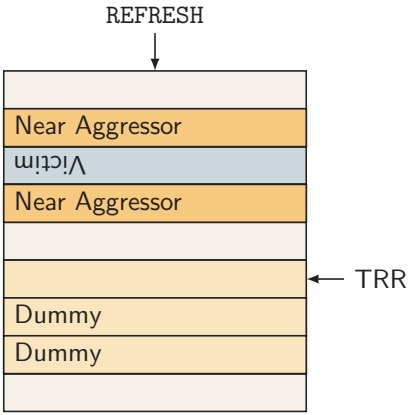
S&P, 2020 [5]

TRRespass [5]

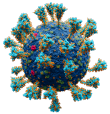


- TRR implementations are not perfect
- Can be distracted by dummy accesses

TRRespass [5]


REFRESH
Near Aggressor
Victim
Near Aggressor
TRR
Dummy
Dummy


STAGEFRIGHT

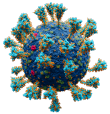





zoom

2014
2015
2016
2017
2018
2019
2020


STAGEFRIGHT



LOG4J




zoom

2014
2015
2016
2017
2018
2019
2020
2021

SMASH: Synchronized Many-sided Rowhammer Attacks from JavaScript

de Ridder, F., Frigo, P., Vannacci, E., Bos, H., Giuffrida, C. and Razavi, K.

Usenix Security, 2021 [23]

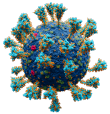
SMASH [23]

New browser JavaScript exploit

1. Evict the cache (no `clflush` instruction)
2. Circumvent TRR (TRRespass)

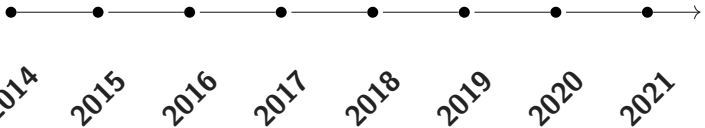
$\rightarrow$ Evict the cache with TRRespass dummy accesses


STAGEFRIGHT

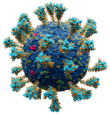


LOG4J




zoom

2014
2015
2016
2017
2018
2019
2020
2021


STAGEFRIGHT



LOG4J




zoom

ÆPIC
LEAK

2014
2015
2016
2017
2018
2019
2020
2021
2022

Half-Double: Hammering From the Next Row Over

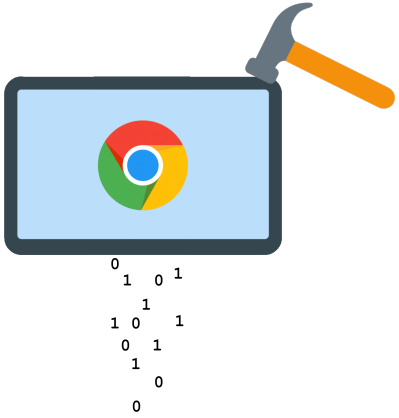
Kogler, A., Juffinger, J., Qazi, S., Kim, Y., Lipp, M., Boichat, N., Shiu, E., Nissler, M. and Gruss, D.

Usenix Security, 2022 [14]

Half-Double Rowhammer [14]

Far Aggressor	$(\mathcal{F}_+)$
Near Aggressor	$(\mathcal{N}_+)$ ← TRR
Victim	$(\mathcal{V})$
Near Aggressor	$(\mathcal{N}_-)$ ← TRR
Far Aggressor	$(\mathcal{F}_-)$

Half-Double Exploit [14]



Half-Double Exploit – Bit-Flip Verification [14]

- Corrupt page table entries can kill the attacker process

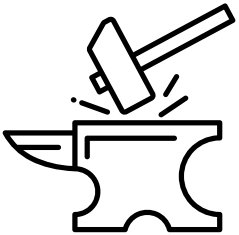
```
if ( /*misprediction*/ ) {
    access(probe + (*ptr & 1));
}
if ( is_cached(probe) ) {
    // ptr[0-4] valid
}
```

Blacksmith: Scalable Rowhammering in the Frequency Domain

Jattke, P., Van Der Veen, V., Frigo, P., Gunter, S. and Razavi, K.

S&P, 2022 [9]

Blacksmith [9]

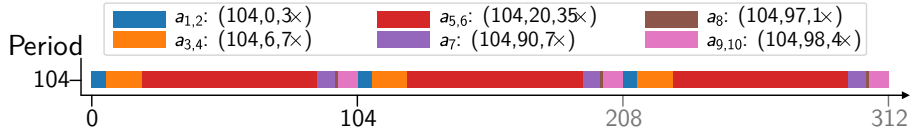


Rowhammer pattern is a combination of aggressor tuples with three characteristics:

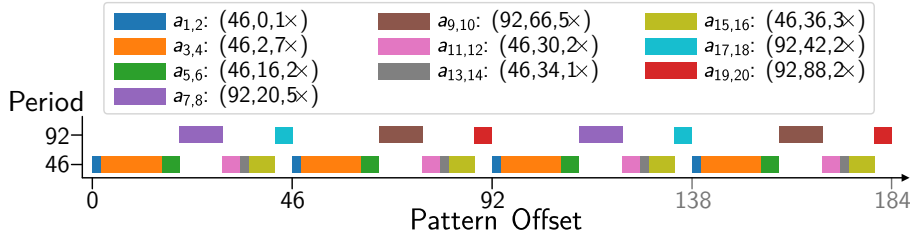
- **Frequency**: How often the tuple is accessed within the pattern
- **Phase**: First hammer after start of pattern
- **Amplitude**: How many consecutive hammers

A fuzzer finds effective patterns.

Blacksmith – Example Patterns [9]



Best pattern found on DIMM $\mathcal{B}_2$.



Best pattern found on DIMM $\mathcal{D}_1$.

Bit flips in all tested modules

SpecHammer: Combining Spectre and Rowhammer for New Speculative Attacks

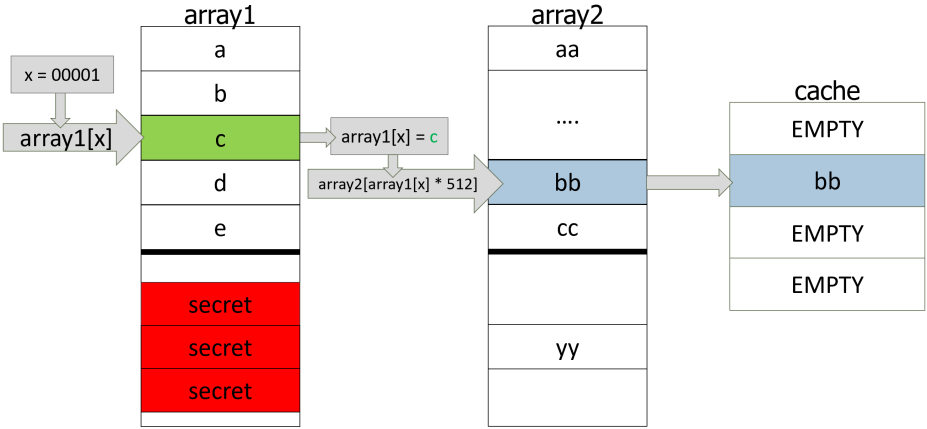
Tobah, Y., Kwong, A., Kang, I., Genkin, D. and Shin, K.G.

S&P, 2022 [26]

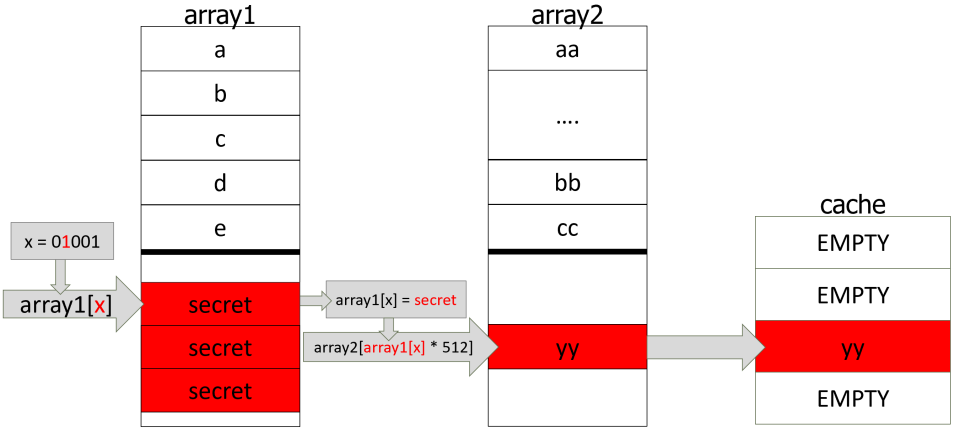
SpecHammer [26]

- No spectre gadgets in the Kernel anymore :(
- All branches with attacker controlled data are fenced
- Branches on “trusted” kernel data are not
- Hammer this kernel data for out of bounds spectre reads

SpecHammer [26]



SpecHammer [26]

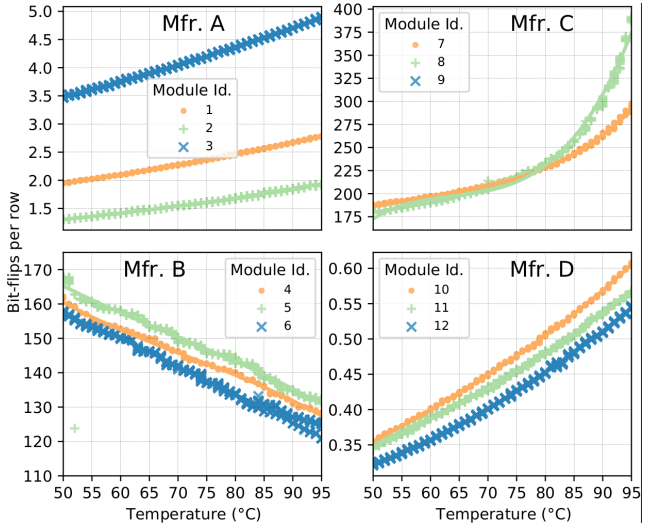


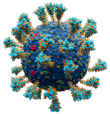
SpyHammer: Using RowHammer to Remotely Spy on Temperature

Orosa, L., Rührmair, U., Yağlikçi, A.G., Luo, H., Olgun, A., Jattke, P., Patel, M., Kim, J., Razavi, K. and Mutlu, O.

arXiv, 2022 [20]

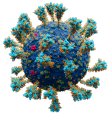
Measure temperature with Rowhammer [20]





2014 2015 2016 2017 2018 2019 2020 2021 2022


STAGEFRIGHT



LOG4J





zoom

ÆPIC
LEAK

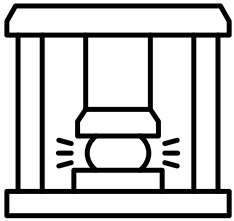
2014
2015
2016
2017
2018
2019
2020
2021
2022
2023

RowPress: Amplifying Read Disturbance in Modern DRAM Chips

Luo, H., Olgun, A., Yağlıkçı, A.G., Tuğrul, Y.C., Rhyner, S., Cavlak, M.B., Lindegger, J., Sadrosadati, M. and Mutlu, O.

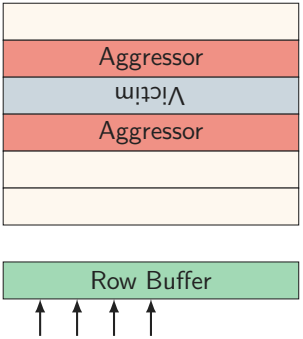
ISCA, 2023 [17]

RowPress [17]



- Different underlying physical effect than Rowhammer
- We unknowingly already saw this effect in 2018 [6, 11]
- Can be exploited [11]

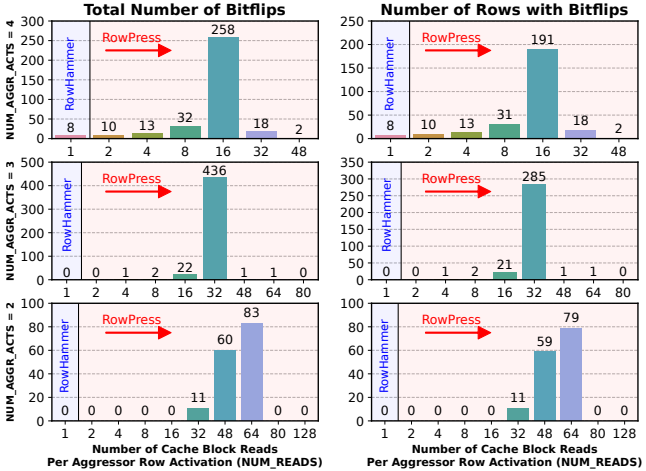
RowPress [17]



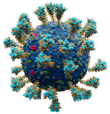
Keep rows open as long as possible.

Flips different bits than Rowhammer

Rowpress [17]


Total Number of Bitflips
Number of Rows with Bitflips
NUM_AGGR_ACTS = 4
NUM_AGGR_ACTS = 3
NUM_AGGR_ACTS = 2
RowHammer
RowPress
8 10 13 32 258 18 2
8 10 13 31 191 18 2
0 0 1 2 22 436 1 1 0
0 0 1 2 21 285 1 1 0
0 0 0 0 0 11 60 83 0 0
0 0 0 0 0 11 59 79 0 0
1 2 4 8 16 32 48 64 80 128
Number of Cache Block Reads
Per Aggressor Row Activation (NUM_READS)


STAGEFRIGHT



LOG4J



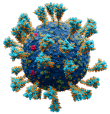


zoom

ÆPIC
LEAK

2014
2015
2016
2017
2018
2019
2020
2021
2022
2023


STAGEFRIGHT



LOG4J





zoom

ÆPIC
LEAK

CROWDSTRIKE

2014
2015
2016
2017
2018
2019
2020
2021
2022
2023
2024

ZenHammer: Rowhammer Attacks on AMD Zen-based Platforms

Jattke, P., Wipfli, M., Solt, F., Marazzi, M., Bölcskei, M. and Razavi, K.

ISCA, 2024 [10]

ZenHammer [10]

- First Rowhammer bit flips on AMD
- Some differences in DRAM mapping
- Better method to detect `REF` commands
- Optimized fencing strategy

RISC-H: Rowhammer Attacks on RISC-V

Marazzi, M. and Razavi, K.

DRAMSec, 2024 [18]

RISC-H [18]

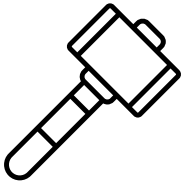
- First Rowhammer bit flips on RISC-V (SOPHON SG2042)
- Required access to multiple DRAM columns due to a hardware bottleneck
- Memory access ordering using `nop`s for small delays

SledgeHammer: Amplifying Rowhammer via Bank-level Parallelism

Kang, I., Wang, W., Kim, J., van Schaik, S., Tobah, Y., Genkin, D., Kwong, A. and Yarom, Y.

Usenix Security, 2024 [12]

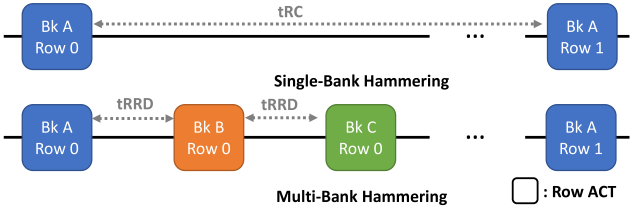
Sledgehammer [12]



- DRAM has multiple banks for parallelism

→ Hammer multiple banks in parallel

→ Less reordering leads to more bit flips

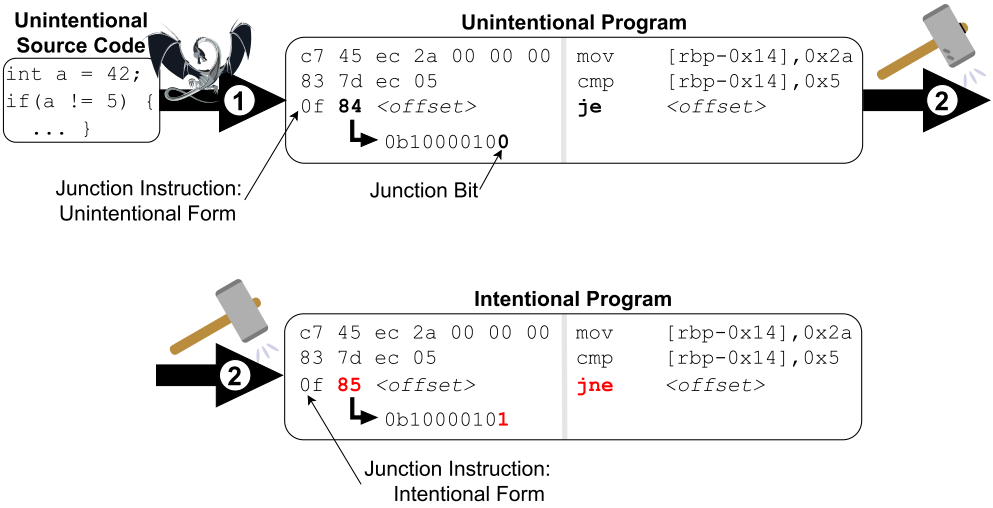


GlueZilla: Efficient and Scalable Software to Hardware Binding using Rowhammer

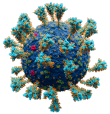
Mechelinck, R., Dorfmeister, D., Fischer, B., Volckaert, S. and Brunthaler, S.

DIMVA, 2024 [19]

Gluezilla [19]




STAGEFRIGHT



LOG4J





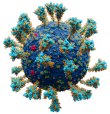
zoom

ÆPIC
LEAK

CROWDSTRIKE

2014
2015
2016
2017
2018
2019
2020
2021
2022
2023
2024


STAGEFRIGHT



LOG4J


?




zoom

ÆPIC
LEAK

CROWDSTRIKE

2014
2015
2016
2017
2018
2019
2020
2021
2022
2023
2024
future?

[1] Misiker Tadesse Aga, Zelalem Birhanu Aweke, and Todd Austin. When good protections go bad: Exploiting anti-DoS measures to accelerate Rowhammer attacks. In: HOST. 2017.

[2] Erik Bosman, Kaveh Razavi, Herbert Bos, and Cristiano Giuffrida. Dedup Est Machina: Memory Deduplication as an Advanced Exploitation Vector. In: S&P. 2016.

[3] Lucian Cojocar, Kaveh Razavi, Cristiano Giuffrida, and Herbert Bos. Exploiting Correcting Codes: On the Effectiveness of ECC Memory Against Rowhammer Attacks. In: S&P. 2019.

[4] Pietro Frigo, Cristiano Giuffrida, Herbert Bos, and Kaveh Razavi. Grand Pwning Unit: Accelerating Microarchitectural Attacks with the GPU. In: S&P. 2018.

[5] Pietro Frigo, Emanuele Vannacci, Hasan Hassan, Victor van der Veen, Onur Mutlu, Cristiano Giuffrida, Herbert Bos, and Kaveh Razavi. TRRespass: Exploiting the Many Sides of Target Row Refresh. In: S&P. 2020.

[6] Daniel Gruss, Moritz Lipp, Michael Schwarz, Daniel Genkin, Jonas Juffinger, Sioli O'Connell, Wolfgang Schoechl, and Yuval Yarom. Another Flip in the Wall of Rowhammer Defenses. In: S&P. 2018.

[7] Daniel Gruss, Clémentine Maurice, and Stefan Mangard. Rowhammer.js: A Remote Software-Induced Fault Attack in JavaScript. In: DIMVA. 2016.

[8] Yeongjin Jang, Jaehyuk Lee, Sangho Lee, and Taesoo Kim. SGX-Bomb: Locking Down the Processor via Rowhammer Attack. In: SysTEX. 2017.

[9] Patrick Jattke, Victor van der Veen, Pietro Frigo, Stijn Gunter, and Kaveh Razavi. BLACKSMITH: Rowhammering in the Frequency Domain. In: S&P. Nov. 2021.

[10] Patrick Jattke, Max Wipfli, Flavien Solt, Michele Marazzi, Matej Bölcskei, and Kaveh Razavi. ZenHammer: Rowhammer Attacks on AMD Zen-based Platforms. In: USENIX Security. 2024.

[11] Jonas Juffinger, Sudheendra Raghav Neela, Martin Heckel, Lukas Schwarz, Florian Adamsky, and Daniel Gruss. Presshammer: Rowhammer and Rowpress without Physical Address Information. In: DIMVA. 2024.

[12] Ingab Kang, Walter Wang, Jason Kim, Stephan van Schaik, Youssef Tobah, Daniel Genkin, Andrew Kwong, and Yuval Yarom. SledgeHammer: Amplifying Rowhammer via Bank-level Parallelism. In: USENIX Security. 2024.

[13] Yoongu Kim, Ross Daly, Jeremie Kim, Chris Fallin, Ji Hye Lee, Donghyuk Lee, Chris Wilkerson, Konrad Lai, and Onur Mutlu. Flipping Bits in Memory Without Accessing Them: An Experimental Study of DRAM Disturbance Errors. In: ISCA. 2014.

[14] Andreas Kogler, Jonas Juffinger, Salman Qazi, Yoongu Kim, Moritz Lipp, Nicolas Boichat, Eric Shiu, Mattias Nissler, and Daniel Gruss. Half-Double: Hammering From the Next Row Over. In: USENIX Security. 2022.

[15] Andrew Kwong, Daniel Genkin, Daniel Gruss, and Yuval Yarom. RAMBleed: Reading Bits in Memory Without Accessing Them. In: S&P. 2020.

[16] Moritz Lipp, Misiker Tadesse Aga, Michael Schwarz, Daniel Gruss, Clémentine Maurice, Lukas Raab, and Lukas Lamster. Nethammer: Inducing Rowhammer Faults through Network Requests. In: SILM Workshop. 2020.

[17] Haocong Luo, Ataberk Olgun, Abdullah Giray Yağlıkçı, Yahya Can Tuğrul, Steve Rhyner, Meryem Banu Cavlak, Joël Lindegger, Mohammad Sadrosadati, and Onur Mutlu. RowPress: Amplifying Read Disturbance in Modern DRAM Chips. In: ISCA. 2023.

[18] Michele Marazzi and Kaveh Razavi. RISC-H: Rowhammer Attacks on RISC-V. In: Workshop on DRAM Security (DRAMSec). 2024.

[19] Ruben Mechelinck, Daniel Dorfmeister, Bernhard Fischer, Stijn Volckaert, and Stefan Brunthaler. GlueZilla: Efficient and Scalable Software to Hardware Binding using Rowhammer. In: DIMVA. 2024.

[20] Lois Orosa, Ulrich Rührmair, A Giray Yaglikci, Haocong Luo, Ataberk Olgun, Patrick Jattke, Minesh Patel, Jeremie Kim, Kaveh Razavi, and Onur Mutlu. Spyhammer: Using rowhammer to remotely spy on temperature. In: arXiv:2210.04084 (2022).

[21] Peter Pessl, Daniel Gruss, Clémentine Maurice, Michael Schwarz, and Stefan Mangard. DRAMA: Exploiting DRAM Addressing for Cross-CPU Attacks. In: USENIX Security. 2016.

[22] Kaveh Razavi, Ben Gras, Erik Bosman, Bart Preneel, Cristiano Giuffrida, and Herbert Bos. Flip Feng Shui: Hammering a Needle in the Software Stack. In: USENIX Security. 2016.

[23] Finn de Ridder, Pietro Frigo, Emanuele Vannacci, Herbert Bos, Cristiano Giuffrida, and Kaveh Razavi. SMASH: Synchronized Many-sided Rowhammer Attacks From JavaScript. In: USENIX Security. 2021.

[24] Mark Seaborn. Exploiting the DRAM rowhammer bug to gain kernel privileges. Mar. 2015. URL: http://googleprojectzero.blogspot.com/2015/03/exploiting-dram-rowhammer-bug-to-gain.html.

[25] Andrei Tatar, Radhesh Krishnan, Elias Athanasopoulos, Cristiano Giuffrida, Herbert Bos, and Kaveh Razavi. Throwhammer: Rowhammer Attacks over the Network and Defenses. In: USENIX ATC. 2018.

[26] Youssef Tobah, Andrew Kwong, Ingab Kang, Daniel Genkin, and Kang G Shin. SpecHammer: Combining Spectre and Rowhammer for New Speculative Attacks. In: S&P. 2022.

[27] Victor van der Veen, Yanick Fratantonio, Martina Lindorfer, Daniel Gruss, Clémentine Maurice, Giovanni Vigna, Herbert Bos, Kaveh Razavi, and Cristiano Giuffrida. Drammer: Deterministic Rowhammer Attacks on Mobile Platforms. In: CCS. 2016.

[28] Yuan Xiao, Xiaokuan Zhang, Yinqian Zhang, and Radu Teodorescu. One Bit Flips, One Cloud Flops: Cross-VM Row Hammer Attacks and Privilege Escalation. In: USENIX Security. 2016.