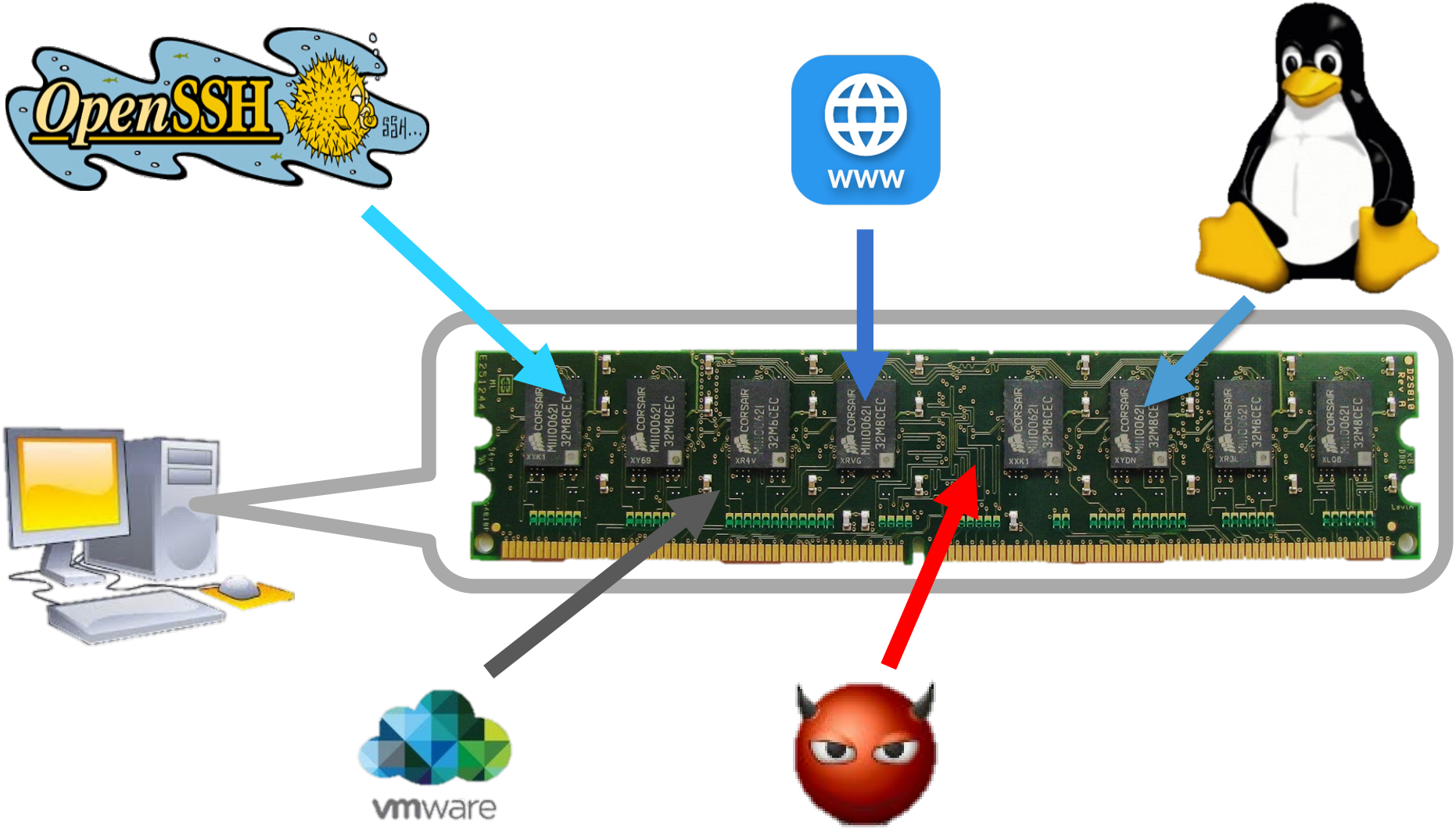
Comp 590-184: Hardware Security and Side-Channels

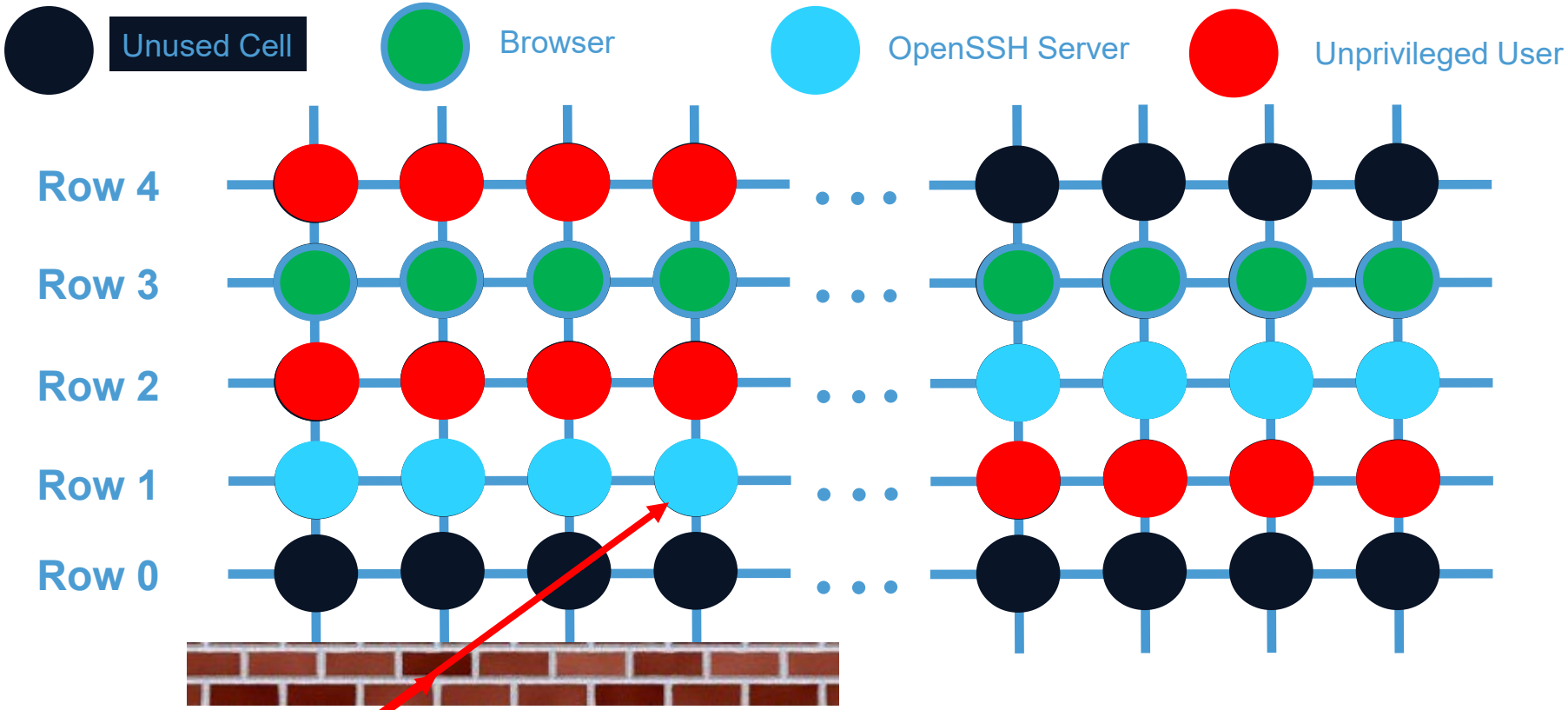
Lecture 18: Rowhammer

March 26, 2025

Andrew Kwong

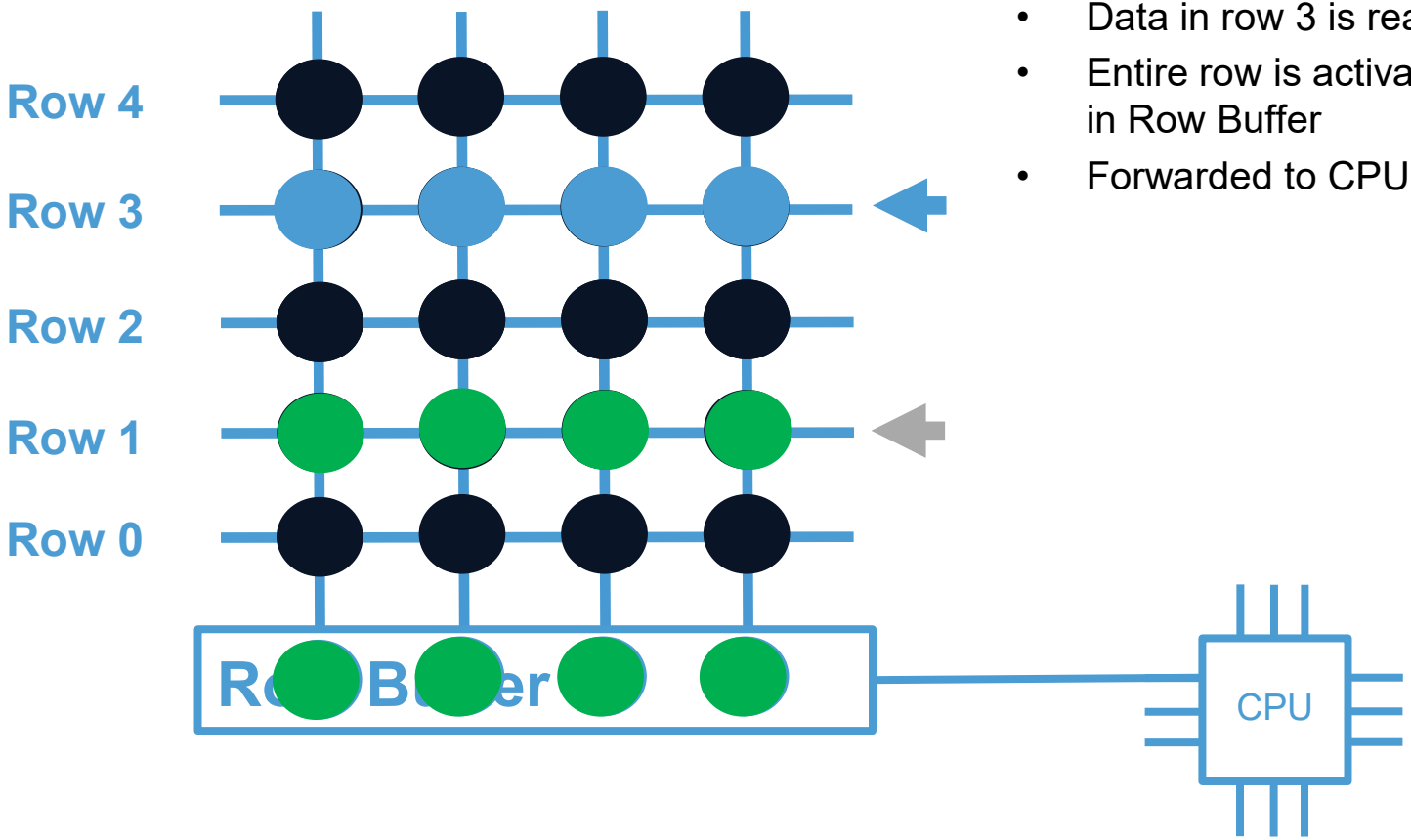
THE UNIVERSITY
of NORTH CAROLINA
at CHAPEL HILL


OpenSSH
www
vmware



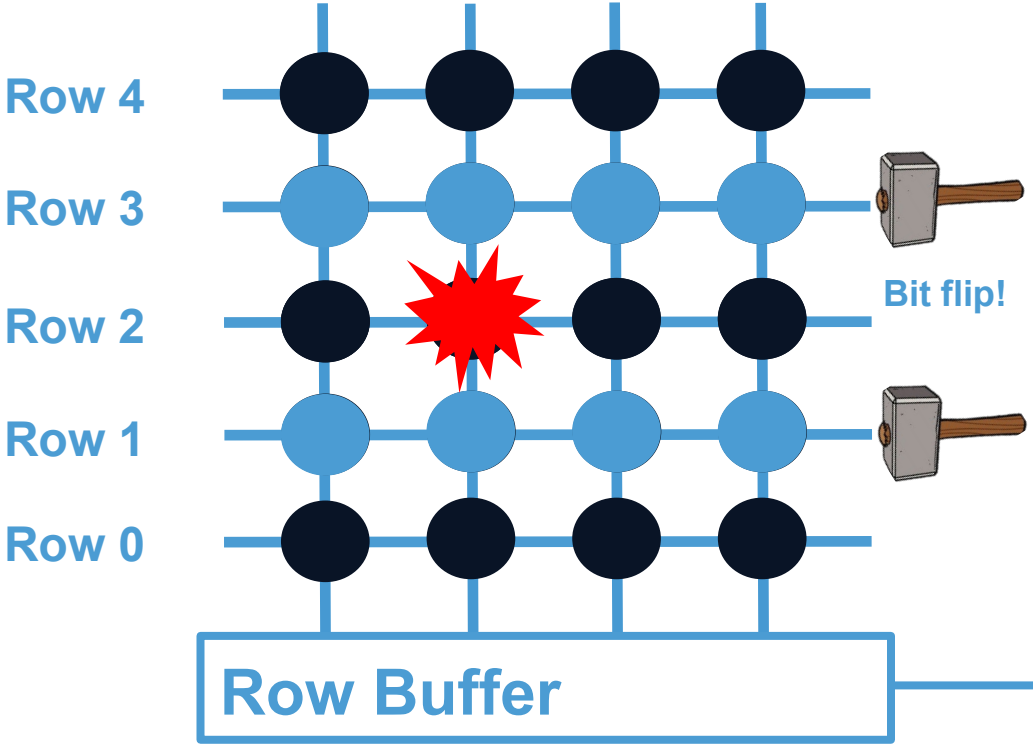
- OS enforces isolation
- **Can physics help us bypass OS and access memory across these boundaries?**

How DRAM works



- Data in row 3 is read
- Entire row is activated and stored in Row Buffer
- Forwarded to CPU

Rowhammer



- Activating a row drains charge from nearby capacitors
- Repeated activation of rows causes bit flips in nearby rows!
- Attacker that controls values in rows 1 and 3 writes to victim's memory in row 2

Why Should we Care About RowHammer?

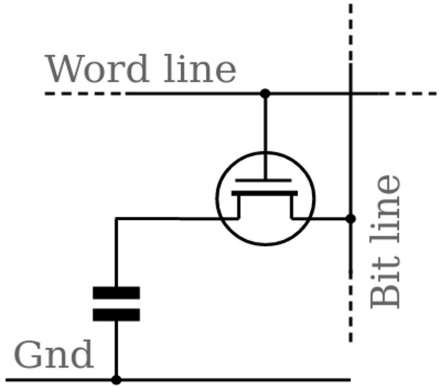
- One can predictably induce bit flips in commodity DRAM chips
- An example of how a simple hardware failure mechanism can create a widespread system security vulnerability

Outline

- Why does RowHammer happen? What is its working mechanism?
- How to perform the attack in practice? Challenges?
- Attack consequences? Mitigations?

DRAM Basics

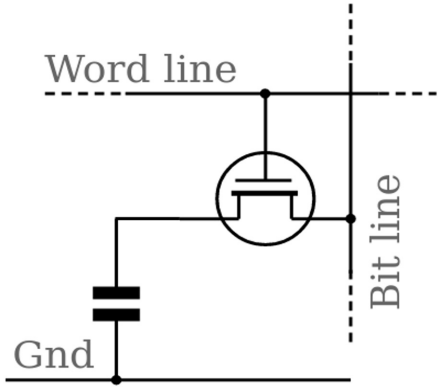
- Each bit in DRAM is stored in a “cell” using a *capacitor*
- *Read is destructive*
- DRAM cells lose their state over time (hence ***Dynamic*** RAM)
- Data stored in DRAM cells needs to be “refreshed” at a regular interval



https://www.electronics-notes.com/articles/electronic_components/semiconductor-ic-memory/dynamic-ram-how-does-dram-work-operation.php

DRAM Basics

- Each bit in DRAM is stored in a “cell” using a *capacitor*
- *Read is destructive*
- DRAM cells lose their state over time (hence ***Dynamic*** RAM)
- Data stored in DRAM cells needs to be “refreshed” at a regular interval

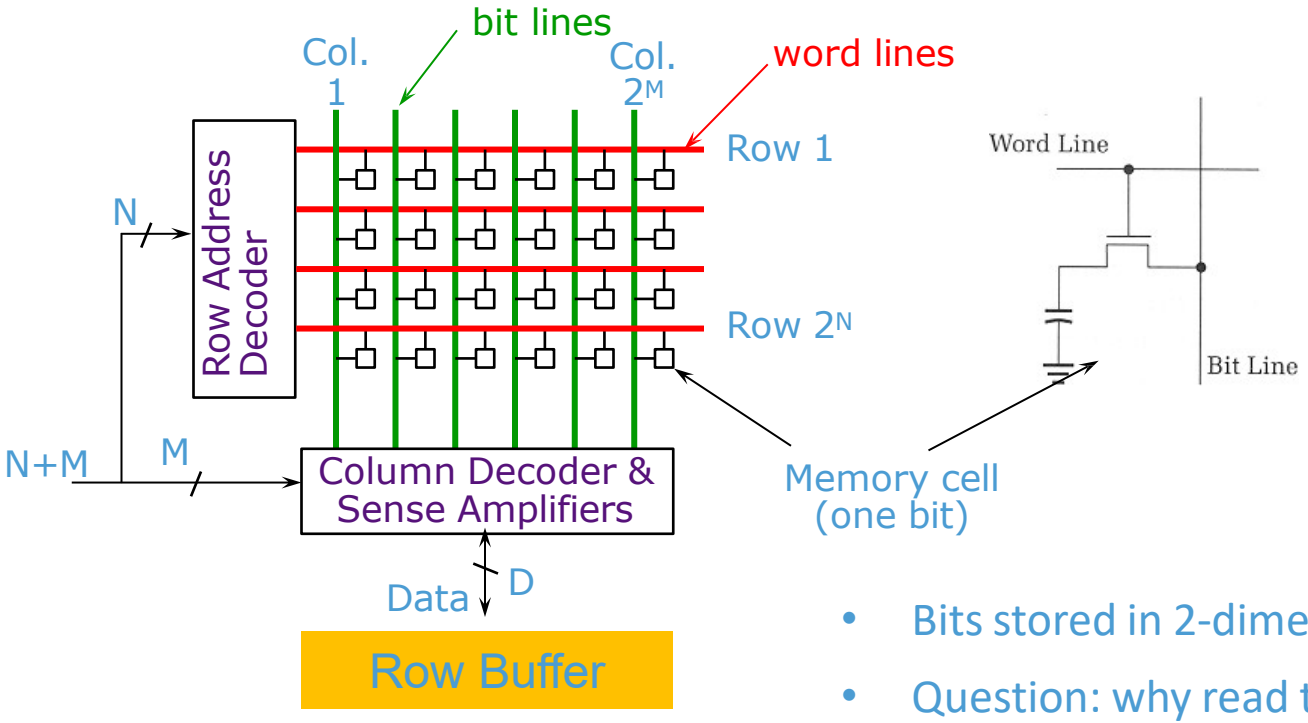


Why do we widely use DRAM given some of its unappealing properties?

- Speed (2-10x slower than SRAM)
- Density (20x denser than SRAM)
- Cost (~100x cheaper per MB)

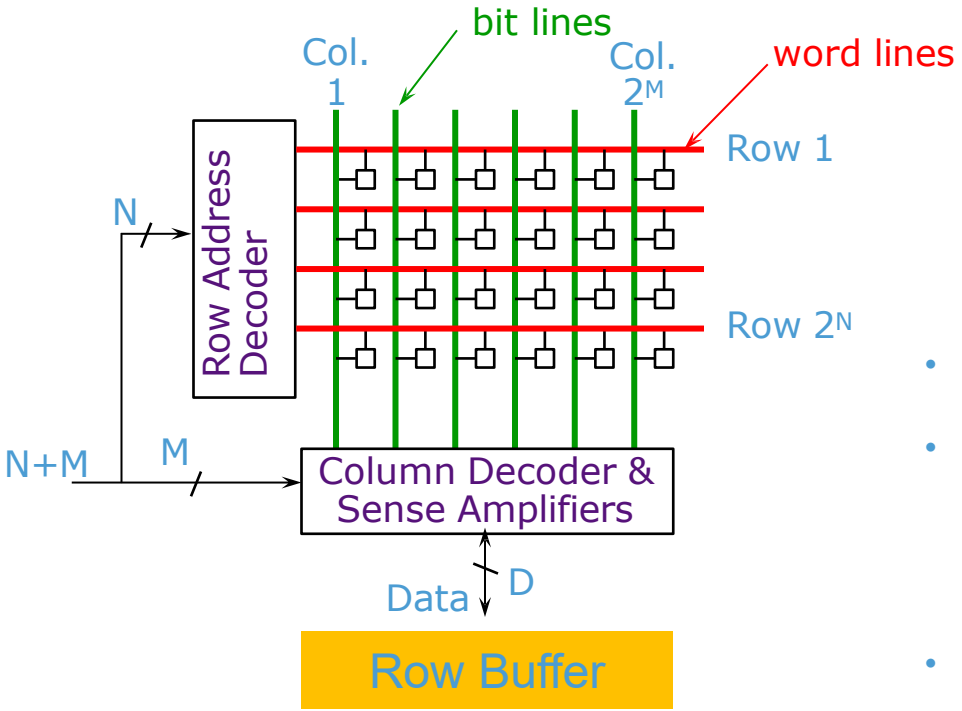
https://www.electronics-notes.com/articles/electronic_components/semiconductor-ic-memory/dynamic-ram-how-does-dram-work-operation.php

DRAM Architecture



- Bits stored in 2-dimensional arrays on chip
- Question: why read the entire row?

DRAM Refresh



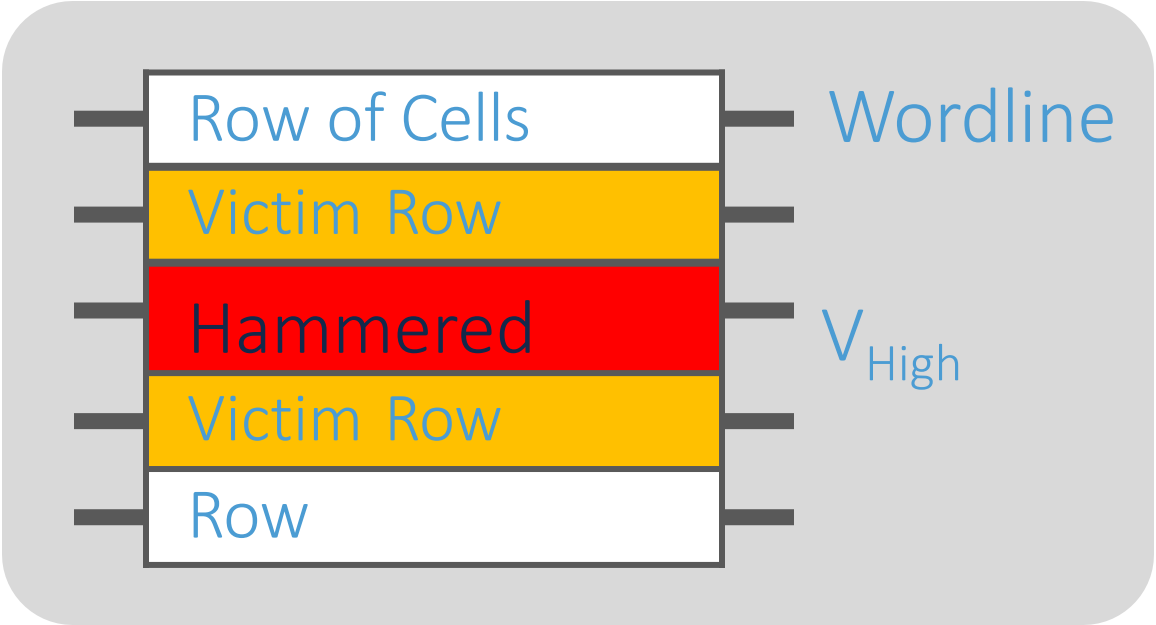
- How do we refresh?
- Performance penalty of refresh
  - In an 8Gb memory, upwards of 10% of time is spent in refresh!
- The common refresh interval: **64ms**

Aside: Cold Boot Attacks

	Seconds w/o power	Error % at operating temp.	Error % at −50°C
SDRAM (1999)	60	41	(no errors)
	300	50	0.000095
DDR (2001)	360	50	(no errors)
	600	50	0.000036
DDR (2003)	120	41	0.00105
	360	42	0.00144
DDR2 (2007)	40	50	0.025
	80	50	0.18

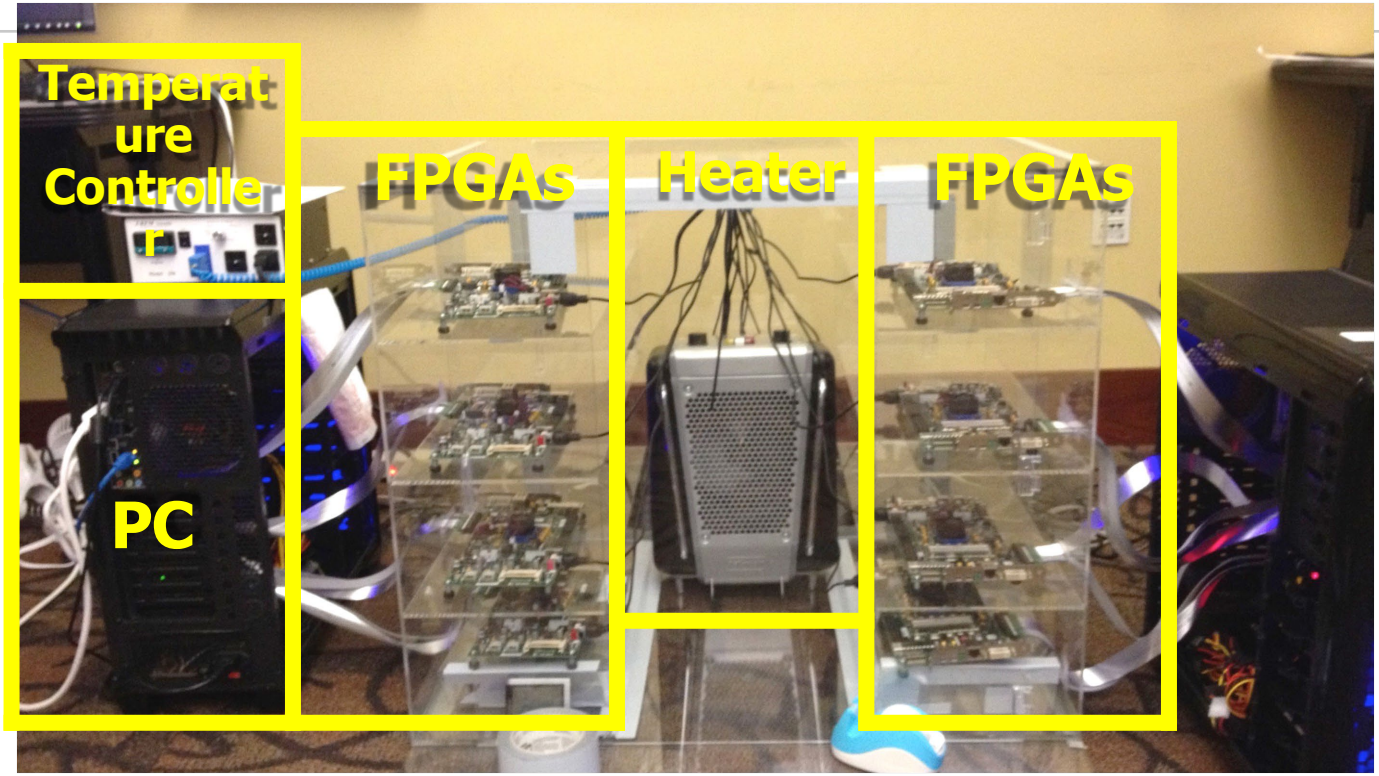
Halderman et al.; Lest We Remember: Cold Boot Attacks on Encryption Keys; USENIX Security'08

See RowHammer Again



Observation: Repeatedly accessing a row enough times **between refreshes** can cause disturbance errors in nearby rows

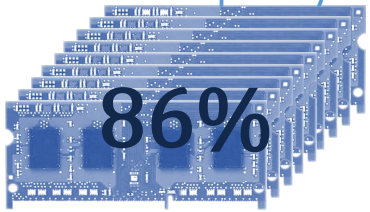
Infrastructures to Understand Rowhammer



Kim et al; Flipping Bits in Memory Without Accessing Them: An Experimental Study of DRAM Disturbance Errors; ISCA'14

Most DRAM Modules Are Vulnerable

A company

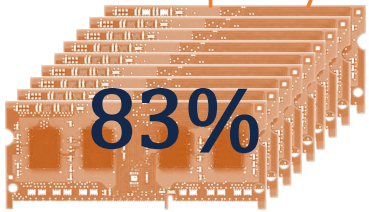


(37/43)

Up to

1.0×10^{7} errors

B company

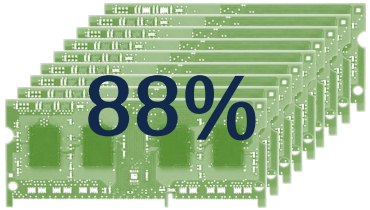


(45/54)

Up to

2.7×10^{6} errors

C company



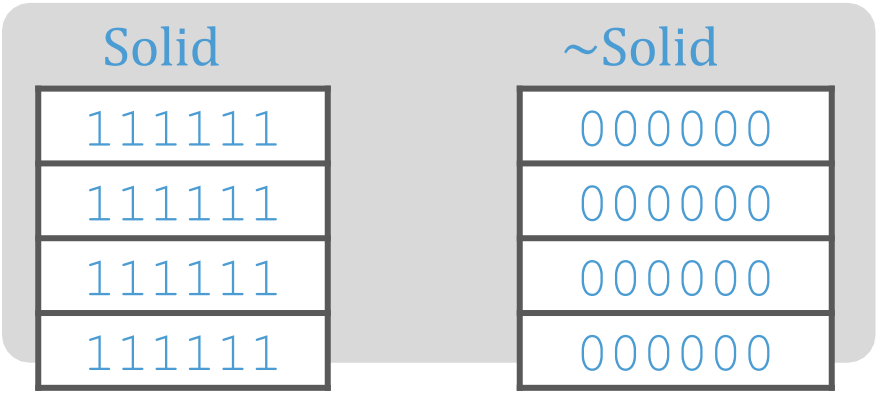
(28/32)

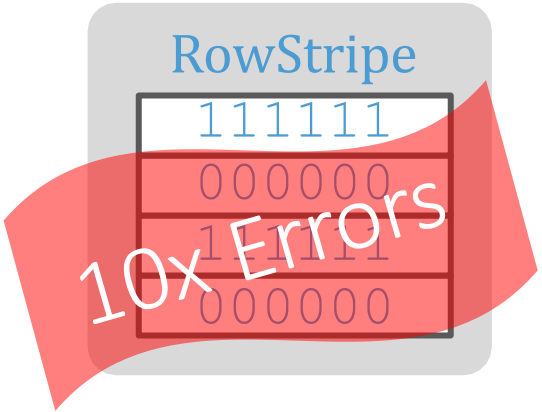
Up to

3.3×10^{5} errors

RowHammer Characteristics

- Highly local nature of the bit-flipping capability
- Bit flips are reproducible
- The probability of bitflips are data-dependent



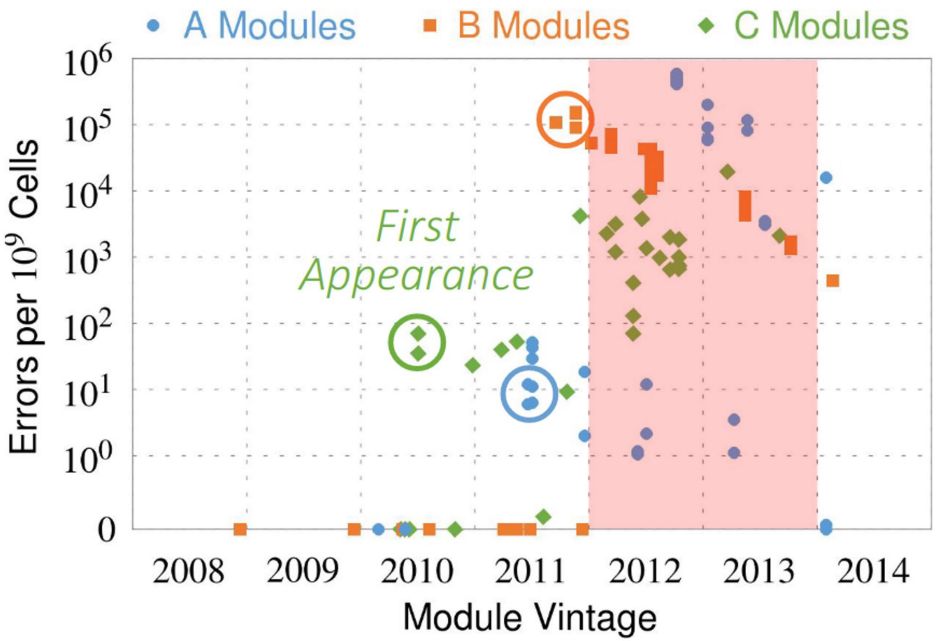


Study RowHammer Characteristics

- Highly local nature of the bit-flipping capability
- Bit flips are reproducible
- The probability of bitflips are data-dependent

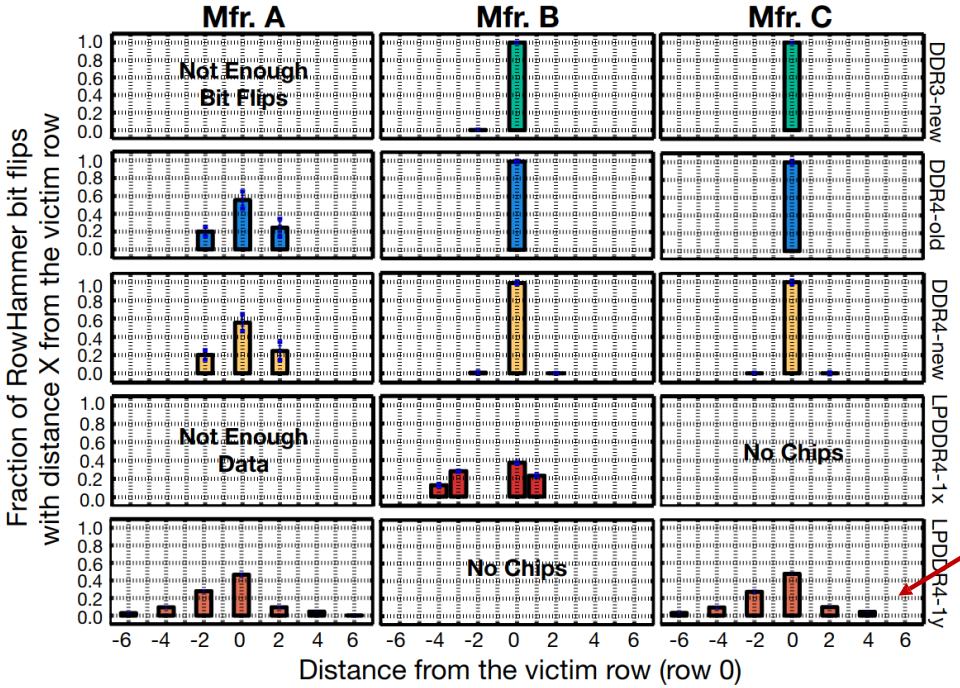
- More advanced DRAM technologies suffer more from this disturbance effect

Density Trends



- As DRAM gets physically denser, it becomes even **more vulnerable**!
  - Trend continues with DDR4
- Only a few thousand hammer iterations are required on modern DRAM to cause a bit-flip

Density Trends



Denser DRAM also can result in flips in rows which are not *directly* adjacent to the attacker

RowHammer is Getting Much Worse

- Jeremie S. Kim, Minesh Patel, A. Giray Yaglikci, Hasan Hassan, Roknoddin Azizi, Lois Orosa, and Onur Mutlu,
  "Revisiting RowHammer: An Experimental Analysis of Modern Devices and Mitigation Techniques"
  Proceedings of the 47th International Symposium on Computer Architecture (ISCA), Valencia, Spain, June 2020.
  [Slides (pptx) (pdf)]
  [Lightning Talk Slides (pptx) (pdf)]
  [Talk Video (20 minutes)]
  [Lightning Talk Video (3 minutes)]

Revisiting RowHammer: An Experimental Analysis of Modern DRAM Devices and Mitigation Techniques

Jeremie S. Kim[§†] Minesh Patel[§] A. Giray Yağlıkçı[§]
Hasan Hassan[§] Roknoddin Azizi[§] Lois Orosa[§] Onur Mutlu[§†]

[§]*ETH Zürich* [†]*Carnegie Mellon University*

Key Takeaways from 1580 Chips

- **Newer DRAM chips are much more vulnerable to RowHammer (more bit flips, happening earlier)**
- There are new chips whose weakest cells fail after **only 4800 hammers**
- Chips of newer DRAM technology nodes can exhibit RowHammer bit flips 1) in **more rows** and 2) **farther away** from the victim row.
- **Existing mitigation mechanisms are NOT effective at future technology nodes**

1580 DRAM Chips Tested

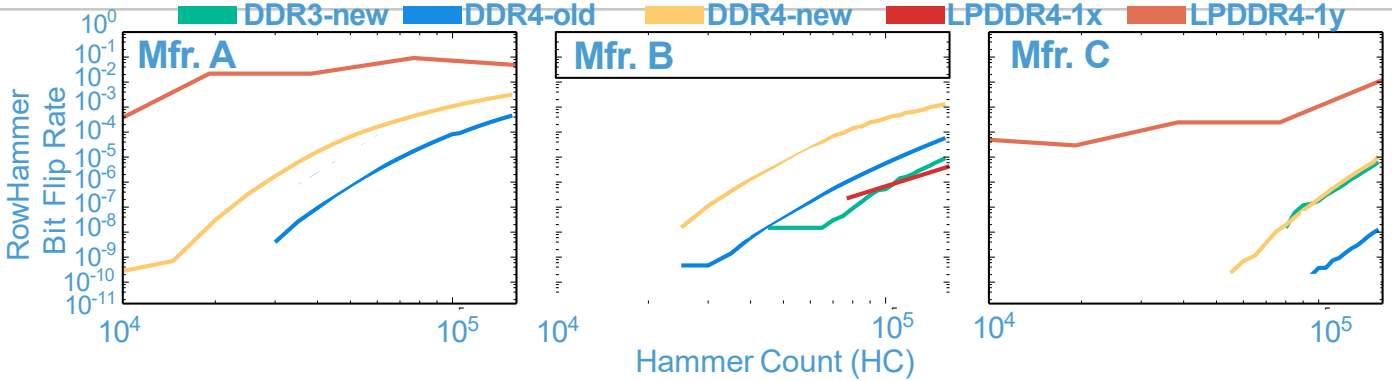
DRAM type-node	Number of Chips (Modules) Tested			
	Mfr. A	*Mfr. B*	*Mfr. C*	*Total*
DDR3-old	56 (10)	88 (11)	28 (7)	**172 (28)**
DDR3-new	80 (10)	52 (9)	104 (13)	**236 (32)**
DDR4-old	112 (16)	24 (3)	128 (18)	**264 (37)**
DDR4-new	264 (43)	16 (2)	108 (28)	**388 (73)**
LPDDR4-1x	12 (3)	180 (45)	N/A	**192 (48)**
LPDDR4-1y	184 (46)	N/A	144 (36)	**328 (82)**

1580 total DRAM chips tested from **300** DRAM modules

- **Three** major DRAM manufacturers {A, B, C}
- **Three** DRAM *types* or *standards* {DDR3, DDR4, LPDDR4}
  - LPDDR4 chips we test implement on-die ECC
- **Two** technology nodes per DRAM type {old/new, 1x/1y}
  - Categorized based on manufacturing date, datasheet publication date, purchase date, and characterization results

Type-node: configuration describing a chip's type and technology node generation: **DDR3-old/new, DDR4-old/new, LPDDR4-1x/1y**

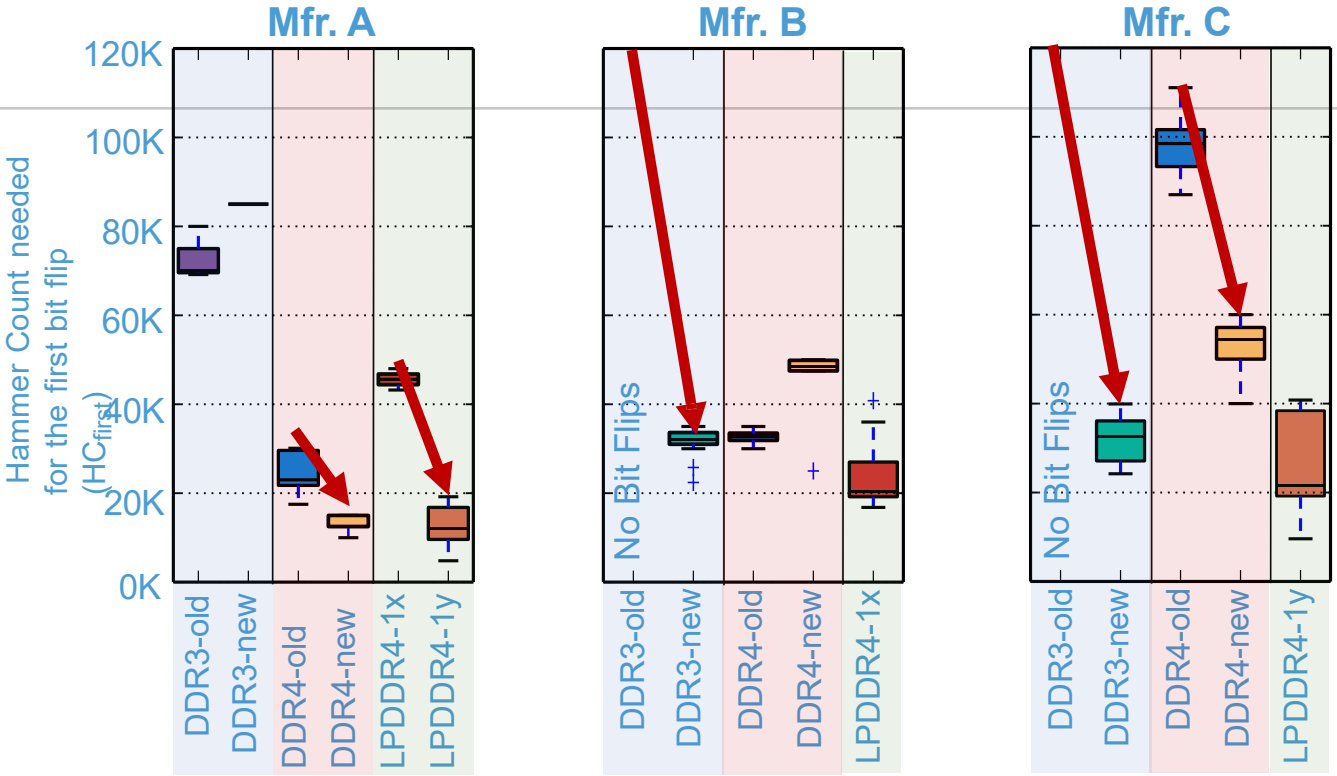
3. Hammer Count (HC) Effects



RowHammer bit flip rates **increase**
when going **from old to new** DDR4 technology node generations

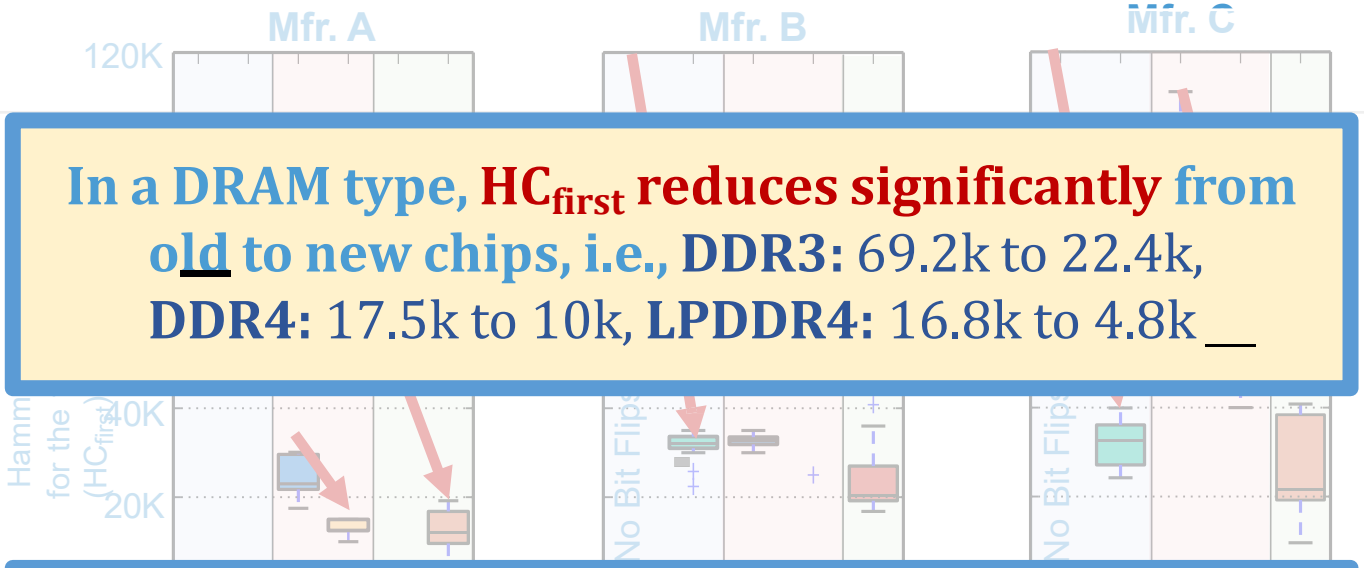
RowHammer bit flip rates (i.e., RowHammer vulnerability) increase with technology node generation

5. First RowHammer Bit Flips per Chip



Newer chips from each DRAM manufacturer are more vulnerable to RowHammer

5. First RowHammer Bit Flips per Chip



In a DRAM type, HC_{first} reduces significantly from old to new chips, i.e., **DDR3:** 69.2k to 22.4k, **DDR4:** 17.5k to 10k, **LPDDR4:** 16.8k to 4.8k

There are chips whose weakest cells fail after only **4800 hammers**

RowHammer is Getting Much Worse

- Jeremie S. Kim, Minesh Patel, A. Giray Yaglikci, Hasan Hassan, Roknoddin Azizi, Lois Orosa, and Onur Mutlu,
  "Revisiting RowHammer: An Experimental Analysis of Modern Devices and Mitigation Techniques"
  Proceedings of the 47th International Symposium on Computer Architecture (ISCA), Valencia, Spain, June 2020.
  [Slides (pptx) (pdf)]
  [Lightning Talk Slides (pptx) (pdf)]
  [Talk Video (20 minutes)]
  [Lightning Talk Video (3 minutes)]

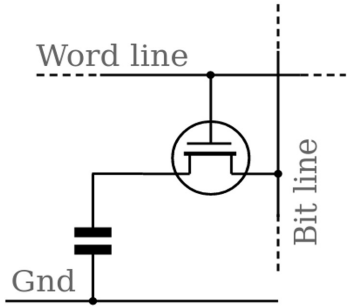
Revisiting RowHammer: An Experimental Analysis of Modern DRAM Devices and Mitigation Techniques

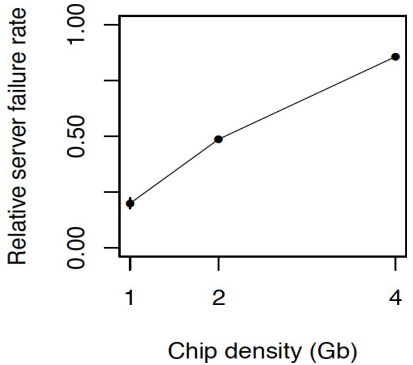
Jeremie S. Kim[§†] Minesh Patel[§] A. Giray Yağlıkçı[§]
Hasan Hassan[§] Roknoddin Azizi[§] Lois Orosa[§] Onur Mutlu[§†]

[§]*ETH Zürich* [†]*Carnegie Mellon University*

Technology Scaling

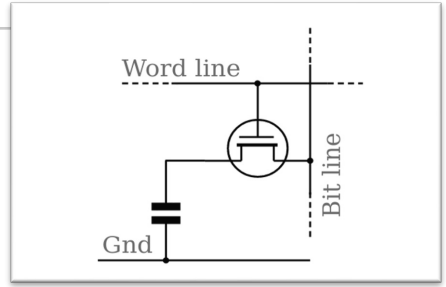
- Capacitor must be large enough for reliable sensing
- The access transistor should be large enough for low leakage and high retention time
- Scaling beyond 40-35nm (2013) is challenging [ITRS, 2009]





Data from all of Facebook's servers worldwide

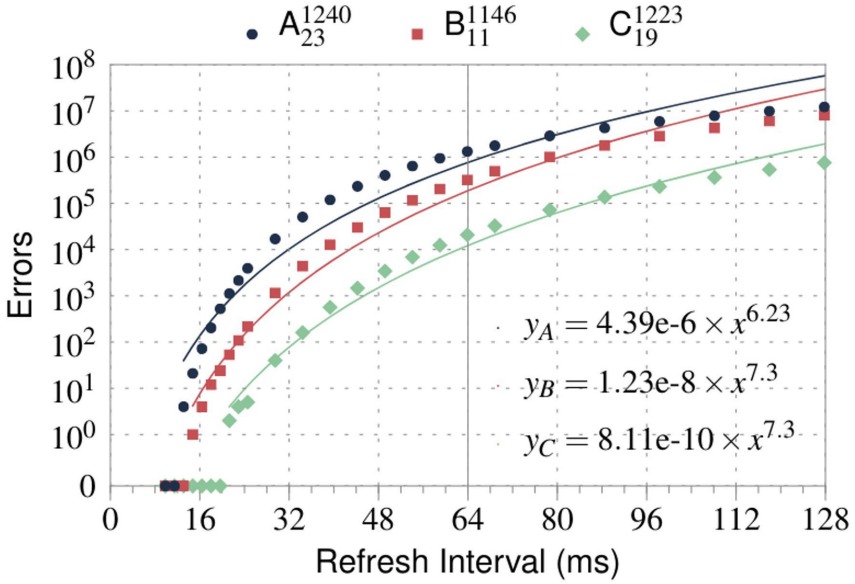
Why Is Rowhammer Happening?



- DRAM cells are too close to each other
  - They are not electrically isolated from each other
- Access to one cell affects the value in nearby cells
  - Due to electrical interference between the cells and wires used for accessing the cells
  - Also called cell-to-cell coupling/interference
- Example: When we activate (apply high voltage) to a row, an adjacent row gets slightly activated as well
  - Vulnerable cells in that slightly-activated row lose a little bit of charge
  - If row hammer happens enough times, capacitor's charge in such cells gets drained

Refresh + Hammering Interval Effects

Examining error rates for different refresh and hammering rates on DDR2 modules from 2011-2012



Flipping Bits in Memory Without Accessing Them: An Experimental Study of DRAM Disturbance Errors

Apple's Patch for RowHammer

- https://support.apple.com/en-gb/HT204934

  Available for: OS X Mountain Lion v10.8.5, OS X Mavericks v10.9.5

  Impact: A malicious application may induce memory corruption to escalate privileges

  Description: A disturbance error, also known as Rowhammer, exists with some DDR3 RAM that could have led to memory corruption. This issue was mitigated by increasing memory refresh rates.

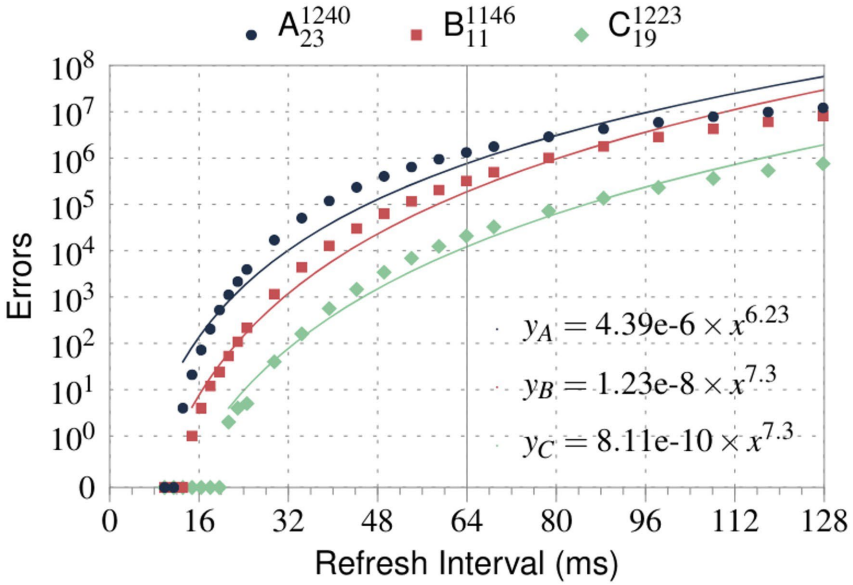
  CVE-ID

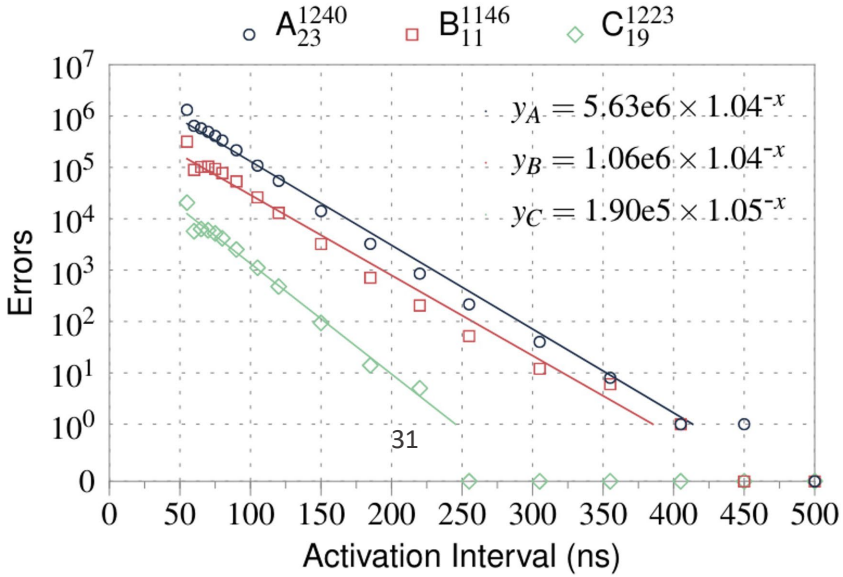
  CVE-2015-3693 : Mark Seaborn and Thomas Dullien of Google, working from original research by Yoongu Kim et al (2014)

HP, Lenovo, and many other vendors released similar patches

Refresh + Hammering Interval Effects

Examining error rates for different refresh and hammering rates on DDR2 modules from 2011-2012

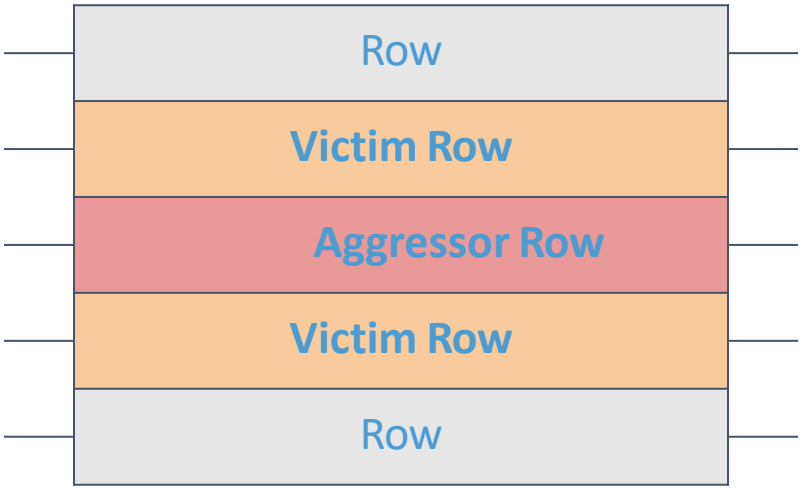




Flipping Bits in Memory Without Accessing Them: An Experimental Study of DRAM Disturbance Errors

RowHammer Attacks in Practice

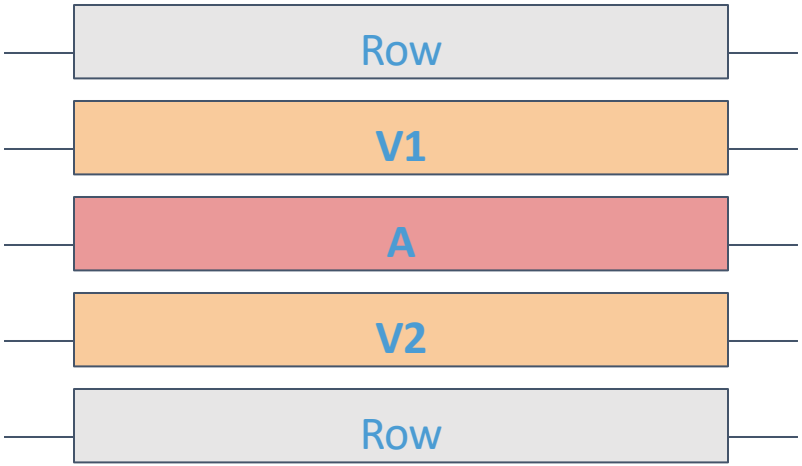
- Aggressor Row = Hammered Row



Challenges:

1. How to hammer? Need to access aggressor row enough times between refreshes.
2. Address mapping. How can we find addresses that map to neighboring rows?
3. How do we map victim's data to vulnerable cells?

Hammer Attempt #1: repeat accesses

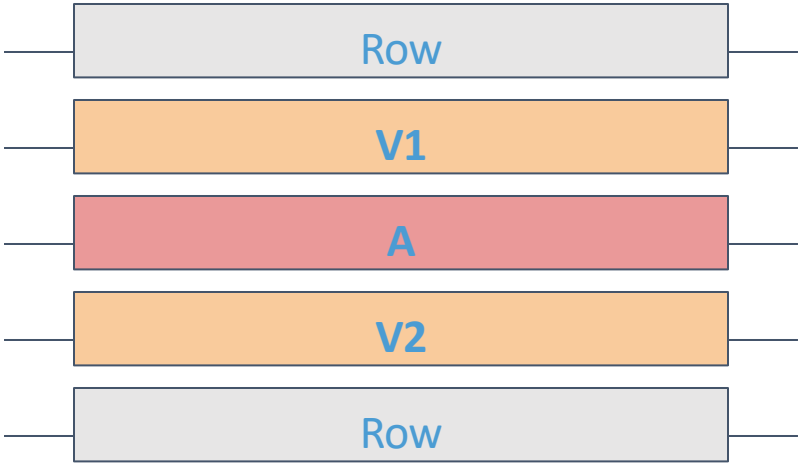



```
loop:
    mov (A), %eax

    mfence
    jmp loop
```

Will this work?
Why?

No. Because we will hit the cache.

Hammer Attempt #2: use clflush

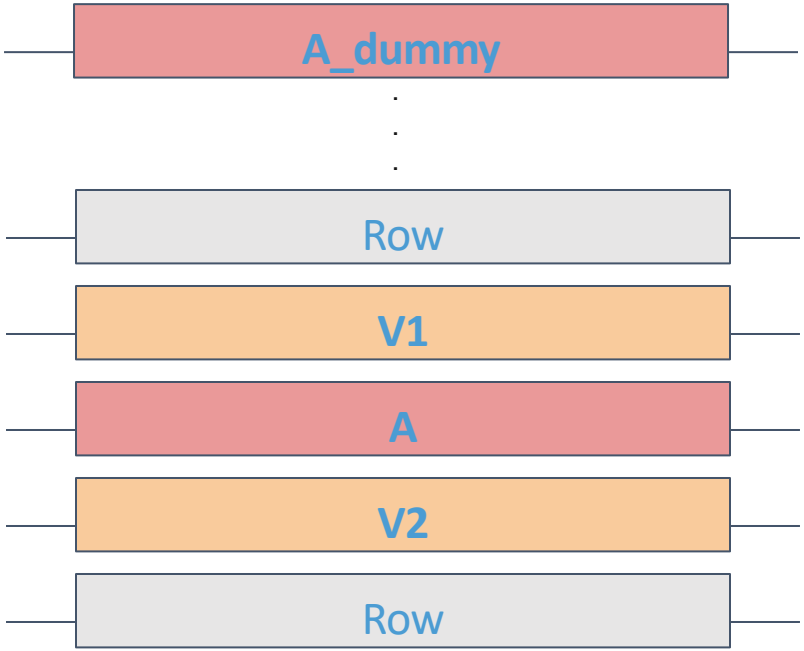



```
loop:
    mov (A), %eax

    clflush (A)

    mfence
    jmp loop
```

Will this work?
Why?

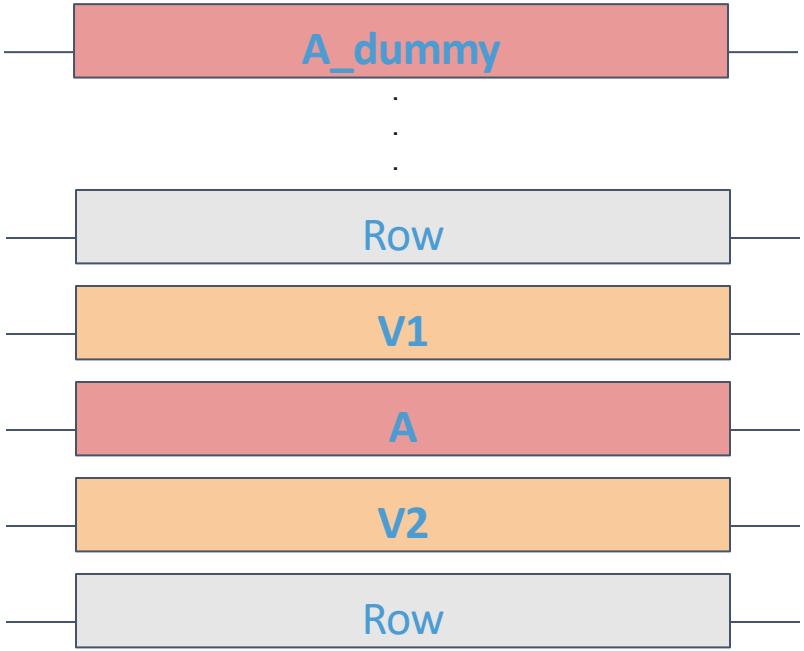
No. Because we will hit the row buffer.

Hammer Attempt #3: force row open/close



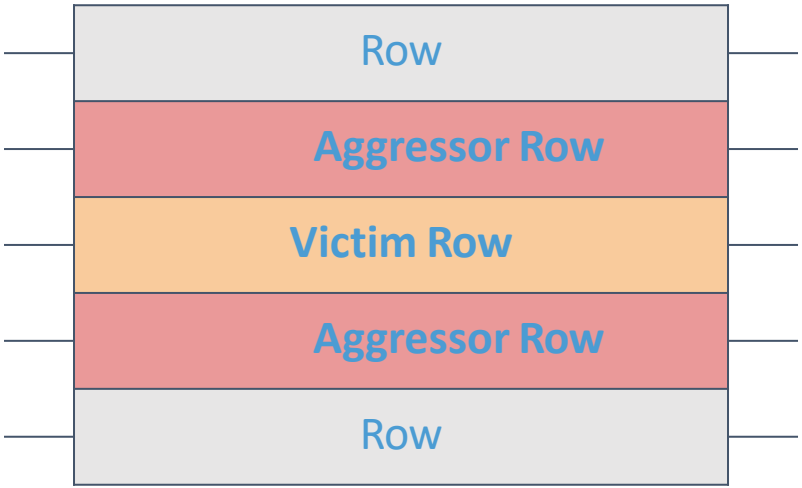

```
loop:
    mov (A), %eax
    mov (A_dummy), %ecx

    clflush (A)
    clflush (A_dummy)

    mfence
    jmp loop
```

“Single-Sided” Rowhammer




```
loop:
    mov (A), %eax
    mov (A_dummy), %ecx

    clflush (A)
    clflush (A_dummy)

    mfence
    jmp loop
```

“Double-Sided” Rowhammer



- Increase the stress:
- Repeatedly accessing both adjacent rows *dramatically* increases the error rate of the victim row

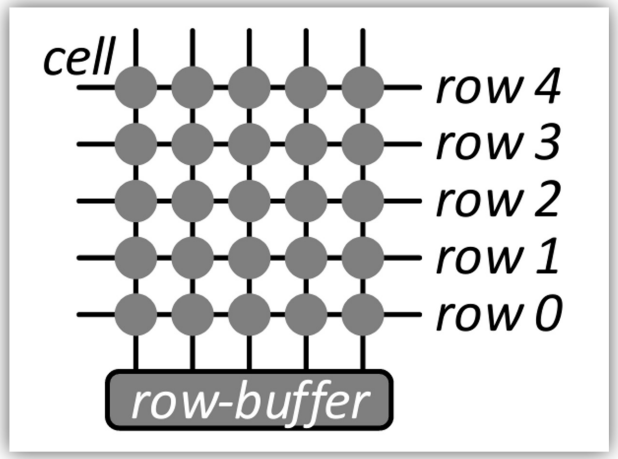
Challenge #2: DRAM Addressing

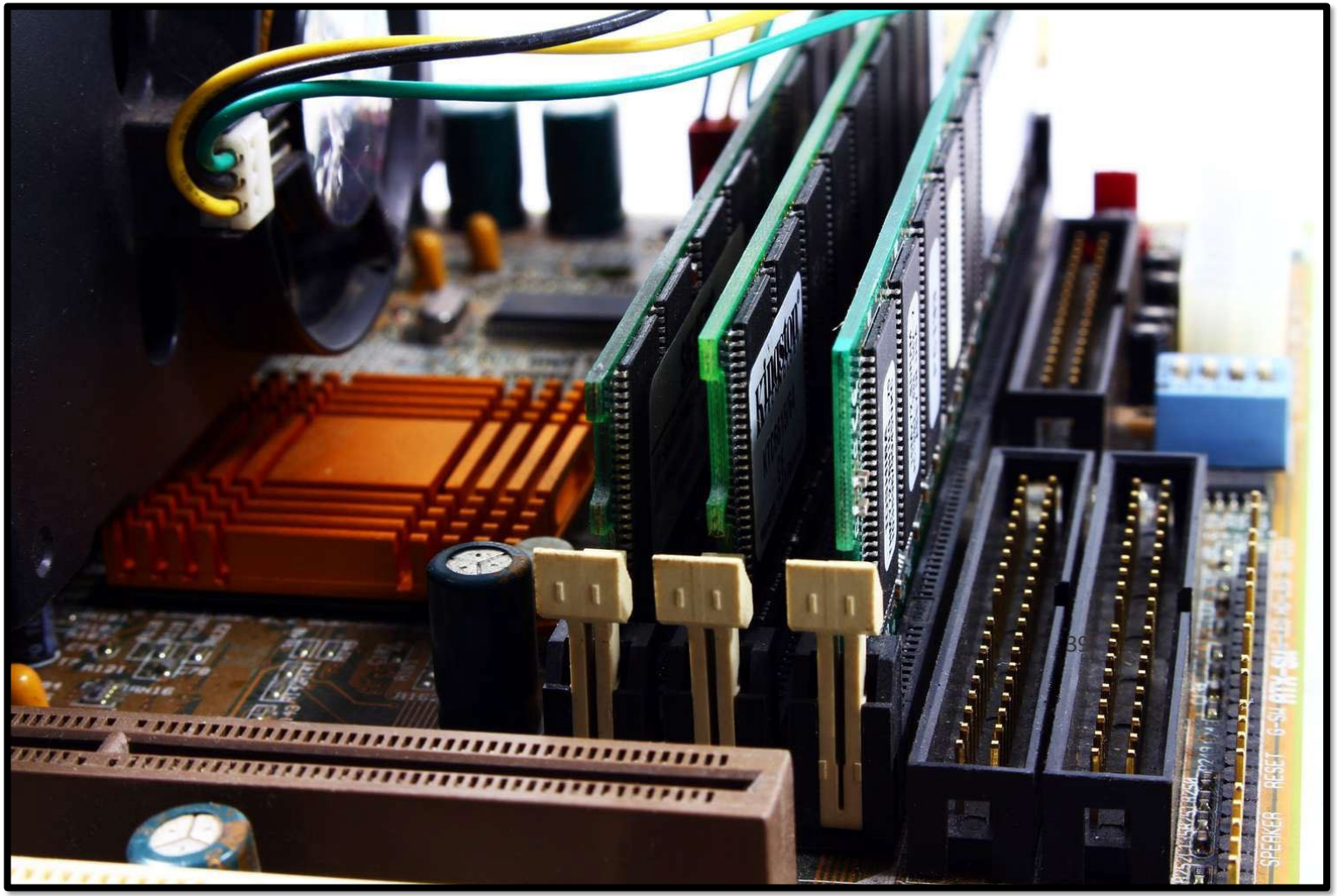
Virtual Address

Address translation
Using
Page table

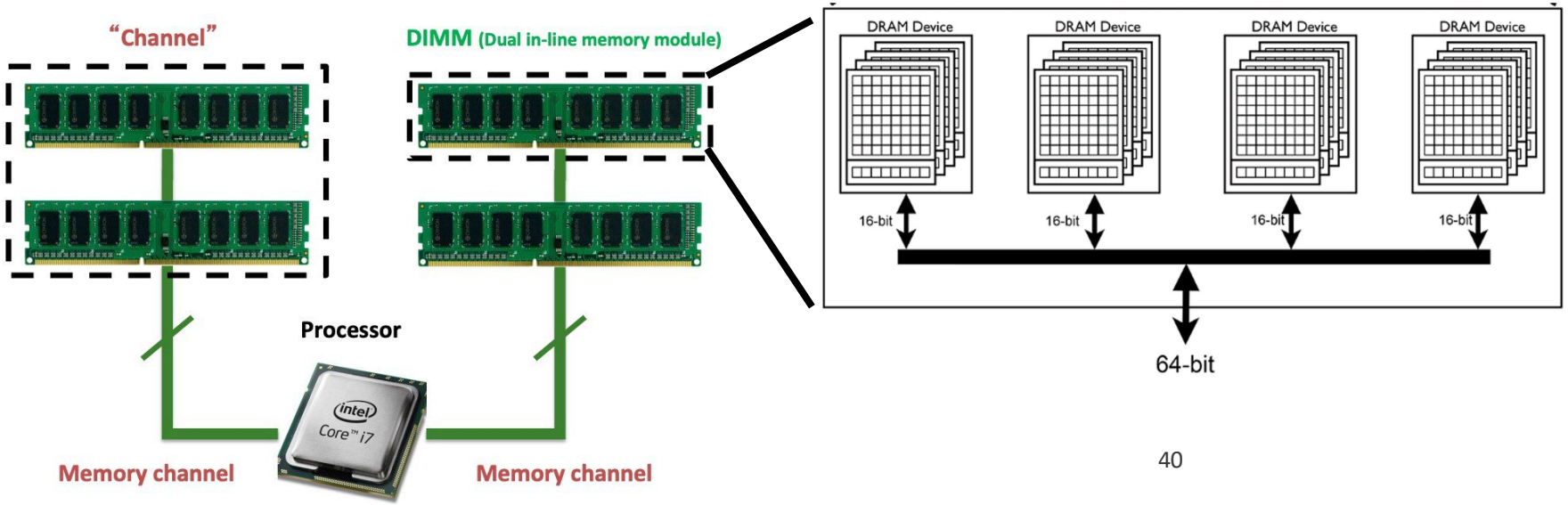
Physical Address

Map to cells?

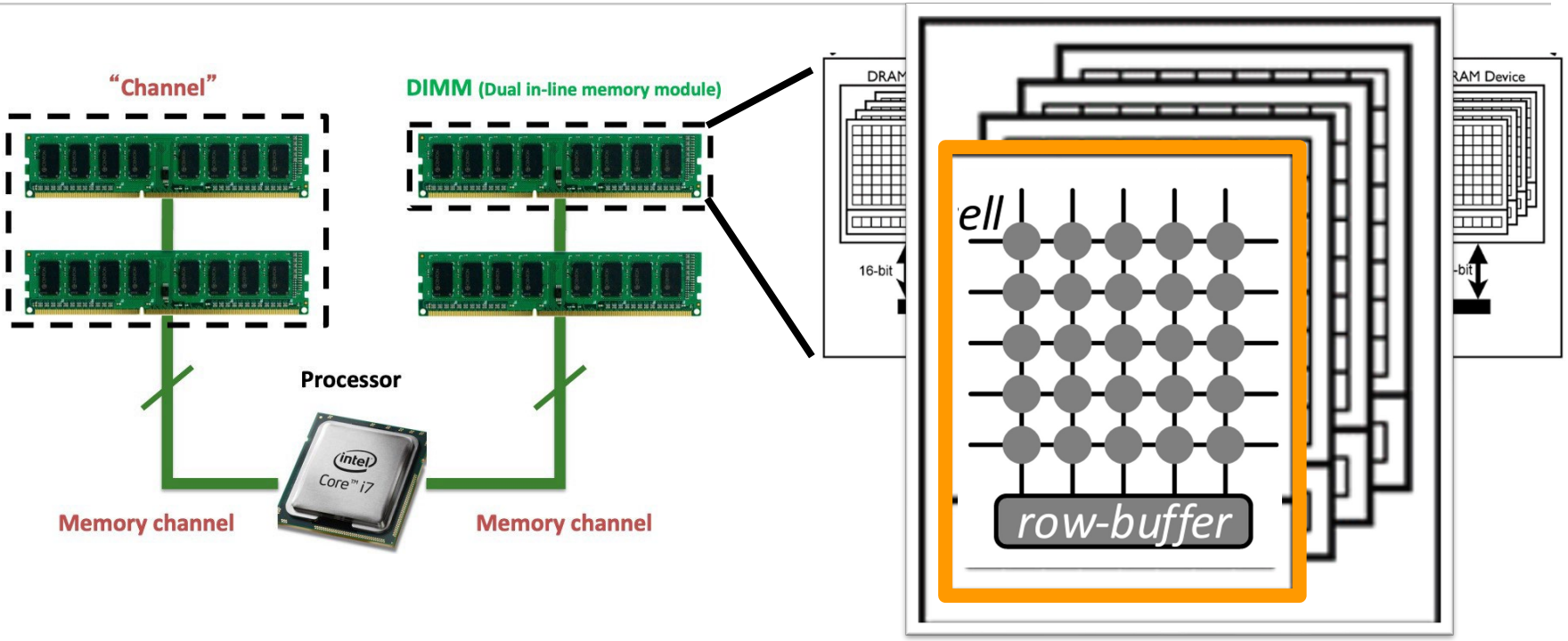




DRAM Organization: Top-down View



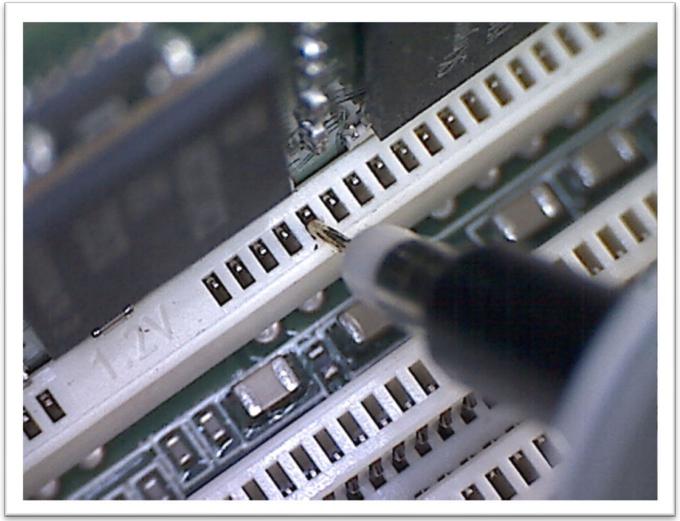
DRAM Organization: Top-down View

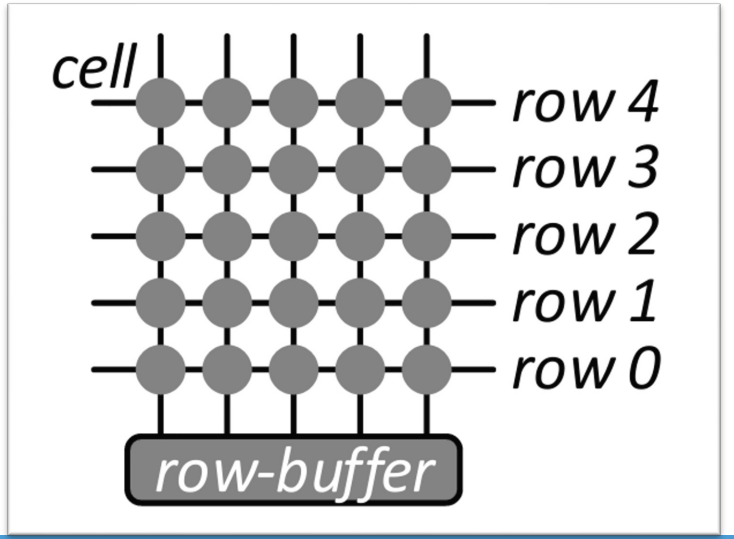


Channel -> DIMM -> Rank -> Bank -> Row/Column

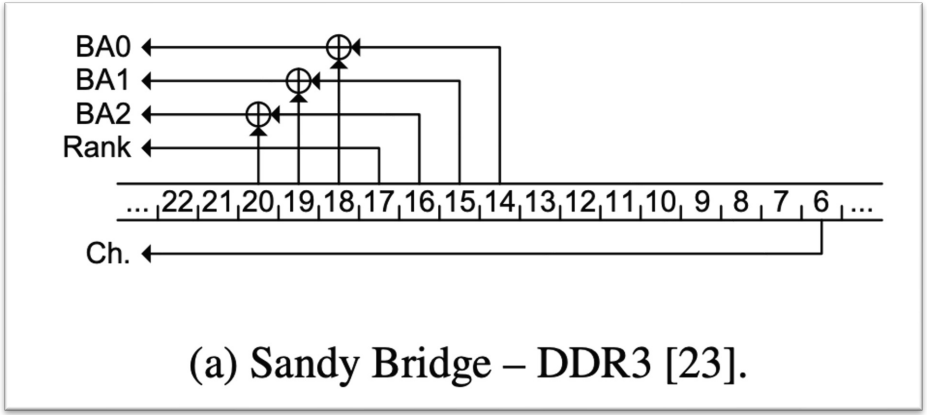
Reverse Engineer the Mapping

- Approach #1: Physical Probe
- Approach #2: Timing Side Channel via Row Buffer

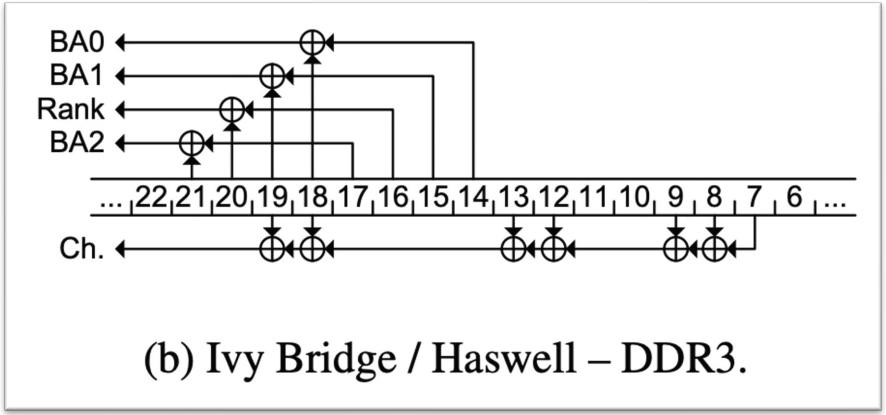




Address Mapping Examples



(a) Sandy Bridge – DDR3 [23].



(b) Ivy Bridge / Haswell – DDR3.

Pessl et al. DRAMA: Exploiting DRAM Addressing for Cross-CPU Attacks. USENIX'16

Rowhammer Attacks

Native Client (NaCl) Sandbox Escape

- NaCl is a sandbox for running native code (C/C++)
- Runs a “safe” subset of x86, statically verifying an executable
- Use bit flips to make an instruction sequence unsafe!

Example “Safe” Code:

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

Exploiting the DRAM rowhammer bug to gain kernel privileges (Seaborn and Dullien)

Native Client (NaCl) Sandbox Escape

We can flip bits to allow for (unsafe) non 32-byte-aligned jumps!

Exploited “Safe” Code:

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

Exploiting the DRAM rowhammer bug to gain kernel privileges (Seaborn and Dullien)

Kernel Privilege Escalation

What could happen if a user could gain direct write access to a page table?

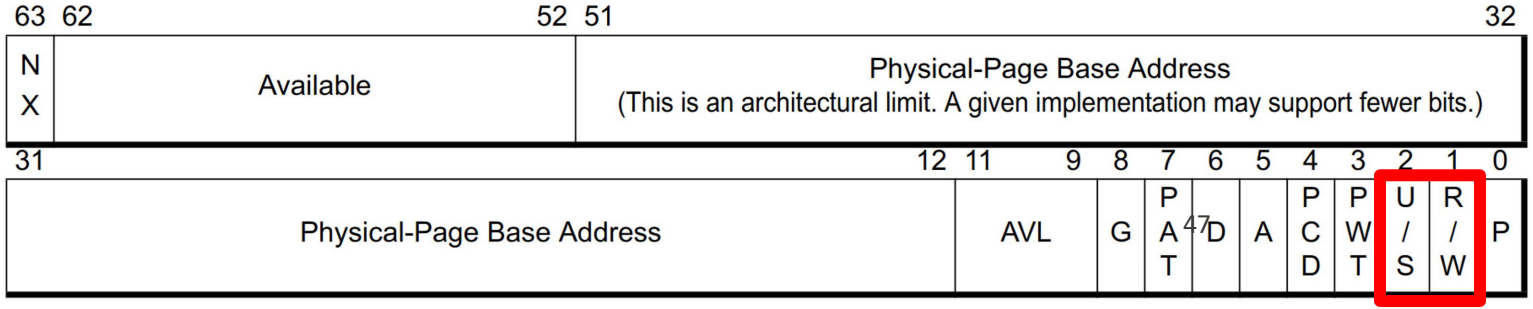
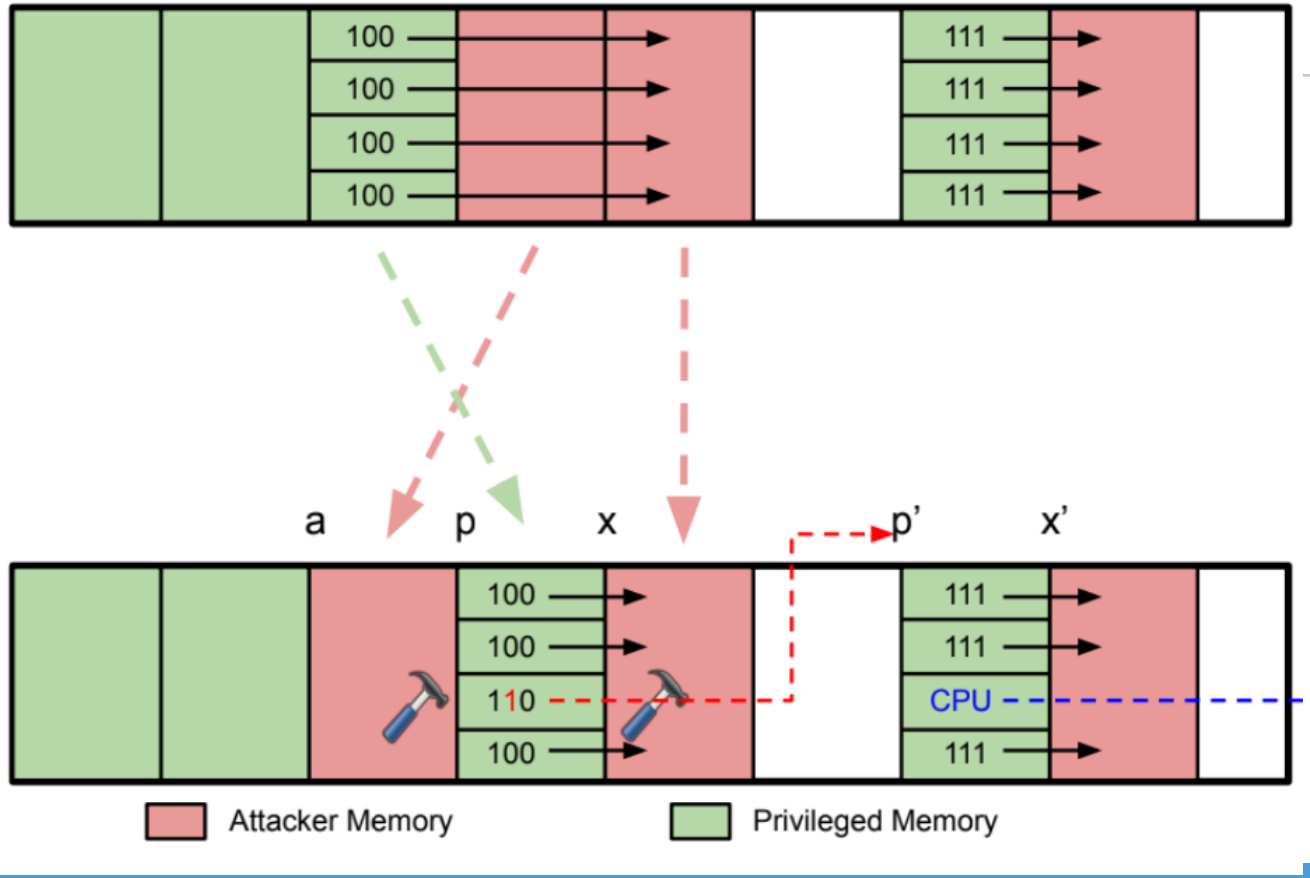


Figure 5-21. 4-Kbyte PTE—Long Mode


100
100
100
100
111
111
111
111
a
p
x
p'
x'
100
100
110
100
111
111
CPU
111
Attacker Memory
Privileged Memory

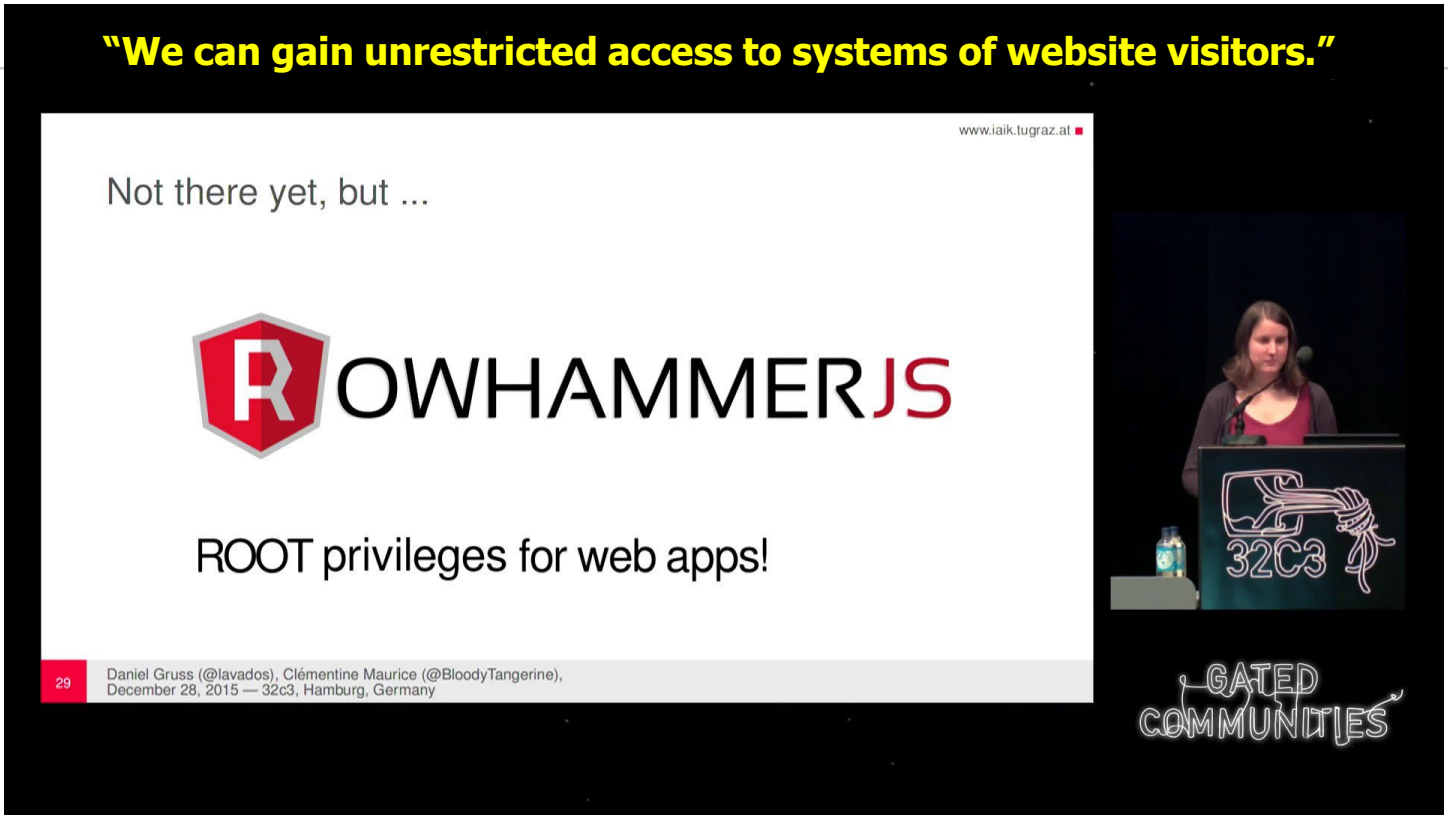
Other Attacks

- Virtual machine takeover
  - Use page de-duplication to corrupt host machine
- OpenSSH attacks
  - Overwrite internal public key with attacker controlled one
  - Read private key directly (RAMBleed)
- Drammer
  - Rowhammer privilege escalation on ARM
  - Utilizes determinism in page allocation to target vulnerable DRAM rows
- Rowhammer.js
  - Remote takeover of a server vulnerable to rowhammer

Without memory integrity, *any* software-based security mechanism is insecure!

More Security Implications (I)

"We can gain unrestricted access to systems of website visitors."



Rowhammer.js: A Remote Software-Induced Fault Attack in JavaScript (DIMVA'16)

https://lab.dsst.io/32c3-slides/7197.html

More Security Implications (II)

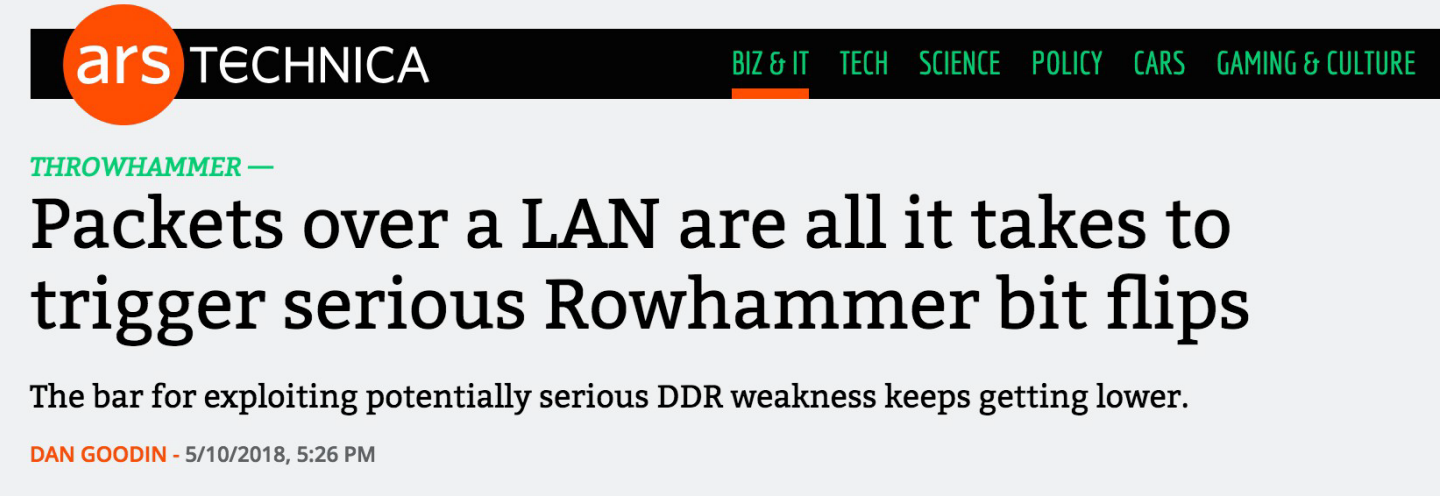
"Can gain control of a smart phone deterministically"

Drammer: Deterministic Rowhammer Attacks on Mobile Platforms, CCS'16

https://fossbytes.com/drammer-rowhammer-attack-android-root-devices/

More Security Implications (IV)

- Rowhammer over RDMA (I) USENIX ATC 2018



Throwhammer: Rowhammer Attacks over the Network and Defenses

Andrei Tatar
VU Amsterdam

Radhesh Krishnan
VU Amsterdam

Elias Athanasopoulos
University of Cyprus

Cristiano Giuffrida
VU Amsterdam

Herbert Bos
VU Amsterdam

Kaveh Razavi
VU Amsterdam

More Security Implications (V)

- Rowhammer over RDMA (II)

The Hacker News™
Security in a serious way

Nethammer—Exploiting DRAM Rowhammer Bug Through Network Requests

Nethammer:
Inducing Rowhammer Faults through Network Requests

Moritz Lipp
Graz University of Technology

Misiker Tadesse Aga
University of Michigan

Michael Schwarz
Graz University of Technology

Daniel Gruss
Graz University of Technology

Clémentine Maurice
Univ Rennes, CNRS, IRISA

Lukas Raab
Graz University of Technology

Lukas Lamster
Graz University of Technology

Rowhammer Mitigations?

- Manufacturing “better” chips — cost
- Increasing refresh rate — Performance, power
- Error Correcting Codes — cost, power
- Targeted row refresh (TRR) - Used in DDR4! — cost, power, complexity
- Retiring vulnerable cells — cost, power, complexity
- Static binary analysis — security
- User/kernel space isolation in physical memory

Error Correcting Codes (ECC)

- **Basic Idea:** Store extra *redundant* bits to be used in case of a flip!
- **Naive Implementation:** Store multiple copies and compare
- **Actual Implementation:** Hamming codes

Hamming codes allow for *single-error* correction, double error detection (aka **SECDED**)

How about more than 2-bit flips?

Takeaways

Reliability Concerns➔Security Implications

THE UNIVERSITY
of NORTH CAROLINA
at CHAPEL HILL