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

Yuan Xiao, Xiaokuan Zhang, Yinqian Zhang,
Radu Teodorescu

The Ohio State University

SOFTPEDIA® DESKTOP MOBILE WEB NEWS

Softpedia > News > Security

Row Hammer DRAM Bug Now Exploitable
JavaScript, Most DDR3 Memory Chips Vuln

Row Hammer bug that affects memory chips can now be exploited via JavaS
unpatachable at software-level

Jul 29, 2

In Mar
with s
any co

Row Hammer can be launched from any website

The three researchers used Rowhammer.js to test out their theory and observed
"attack runs in [a] sandboxed JavaScript which is present and enabled by defaul
modern browser."

"Although implemented in JavaScript, the attack technique is independent of th

SOFTPEDIA® DESK

Softpedia > News > Security

Row Hammer DRAM
Access to Physical M

ome

obvi
one
ect.

f-co

An experimental stu
Without Accessing
by Carnegie Mellon University.

A process' address space in the memory is isolated from that of other processes, but the researchers demonstrated that repeated access to two memory locations in a process' virtual address space leads to flipping the bit in a third location in a different DRAM row. However, no attack was attempted at the time.

Starting from this paper, security researchers at Google's Project Zero picked up from

ars technica

MAIN MENU MY STORIES: 25 FORUMS SUBSCRIBE JOBS

RISK ASSESSMENT / SECURITY & HACKTIVISM

Once thought safe, DDR4 memory shown to be vulnerable to "Rowhammer"

New research finds "bitflipping" attacks may pose more risk than many admit.

by Dan Goodin - Mar 17, 2016 7:16pm EDT

Share Tweet Email

"Based on the analysis by Third I/O, we believe that this problem is significantly worse than what is being reported," the paper warned. "And it is still visible on some DDR4 memory modules."

vider

Once thought safe, DDR4 memory shown to be

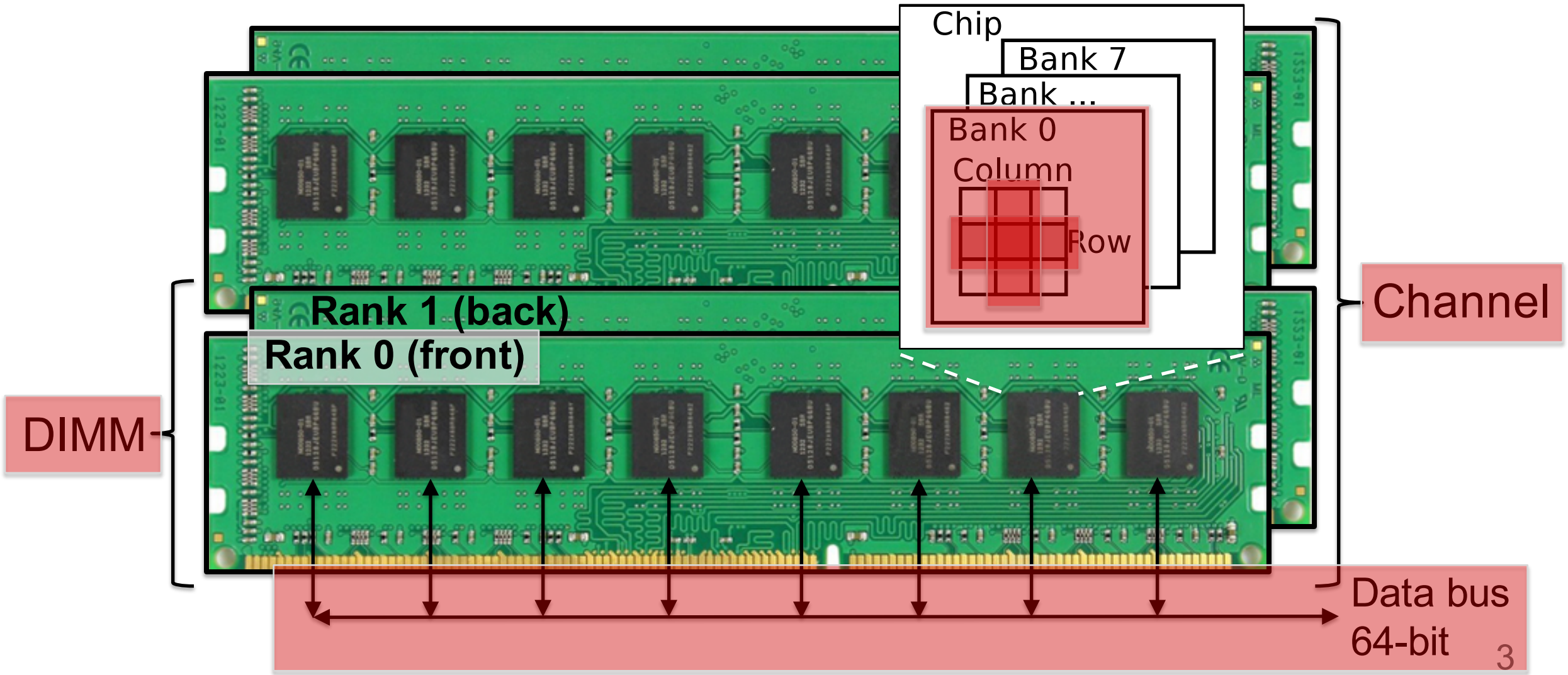
Row Hammer DRAM Bug Exploited, Unlocks Access to Physical Memory, **March 9 2015**

Row Hammer DRAM Bug Now E
JavaS
Memo

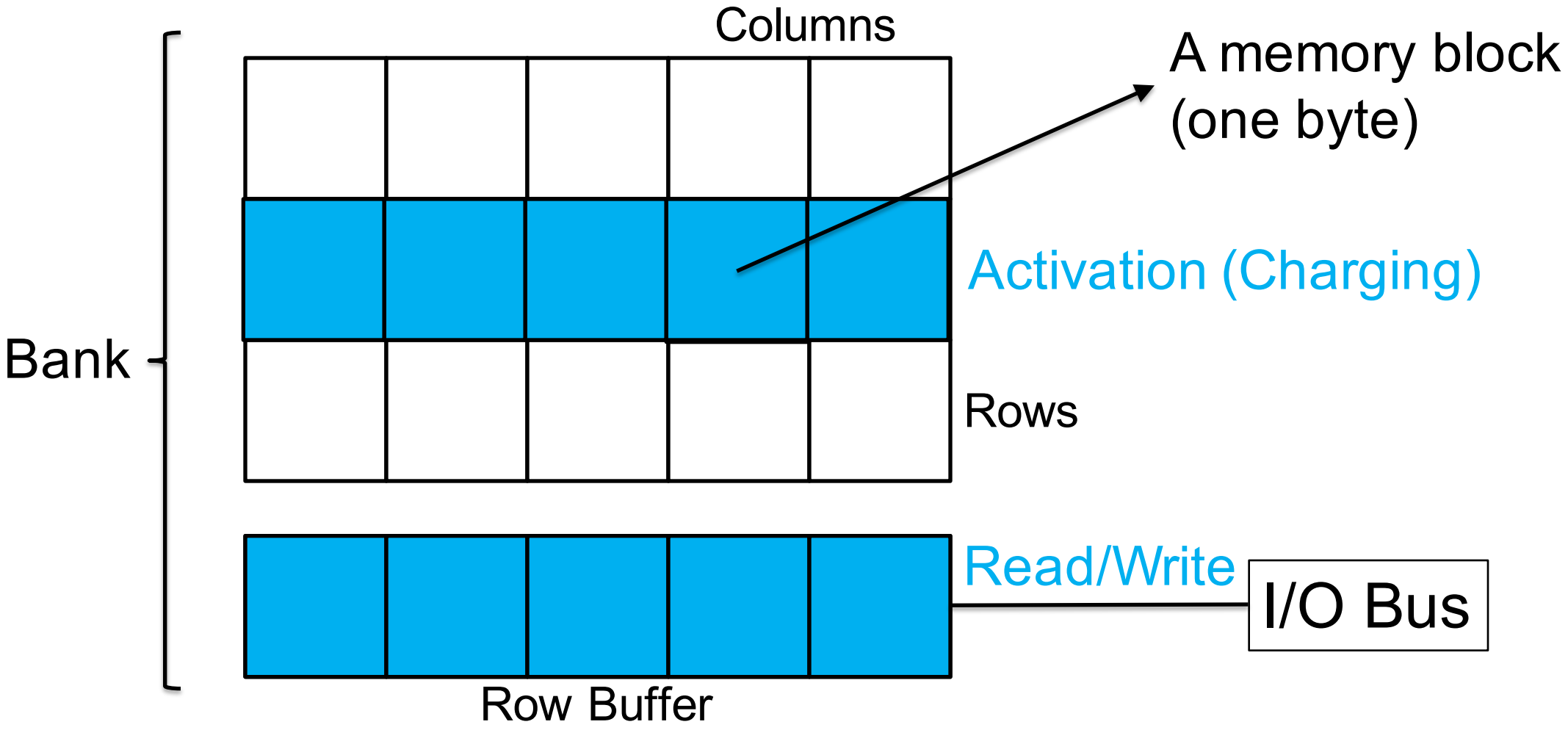
Flipping DRAM bits - maliciously, **December 29, 2014**

July 29 2015

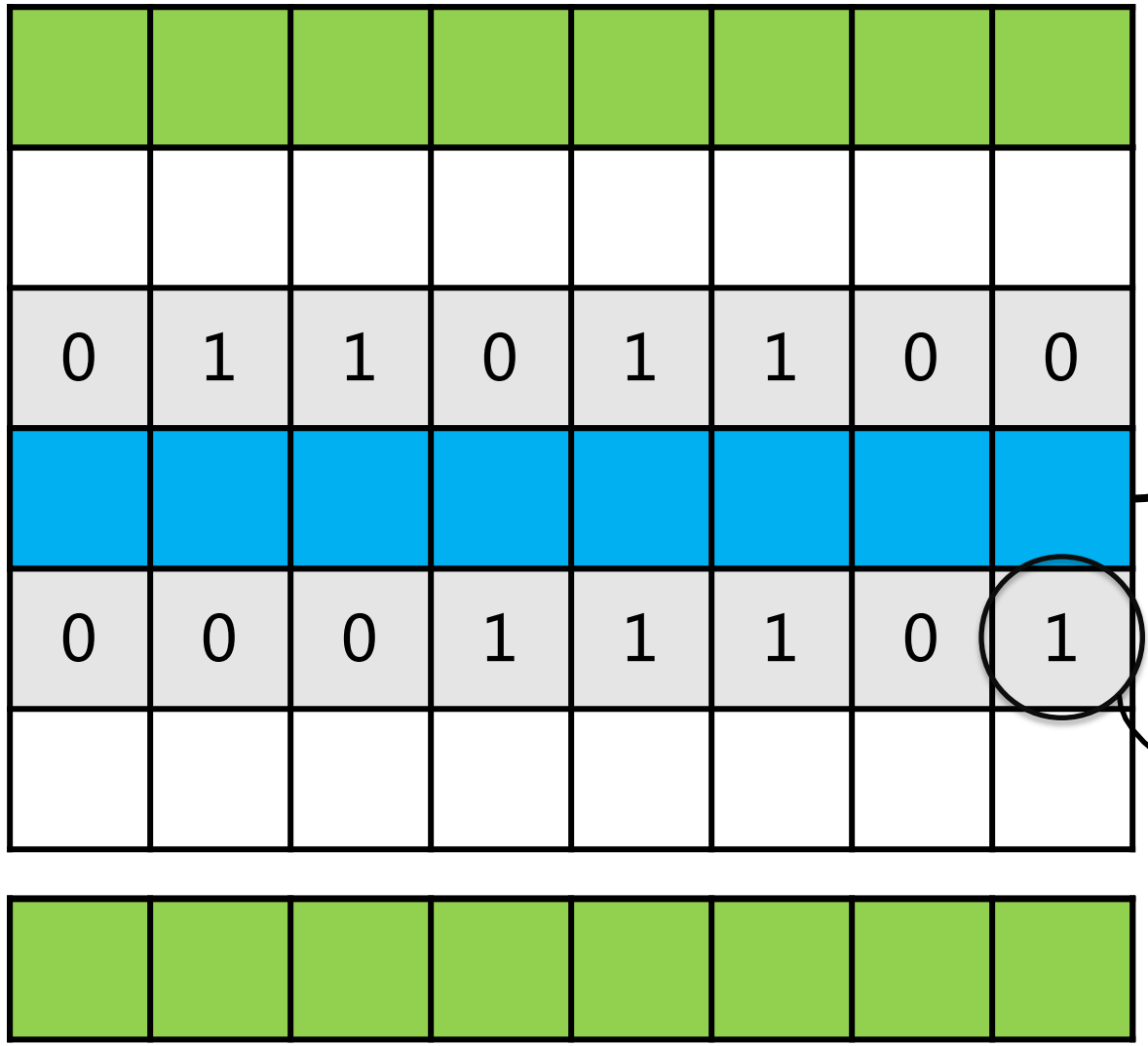
DRAM Architecture



Row Buffer and Row Activation

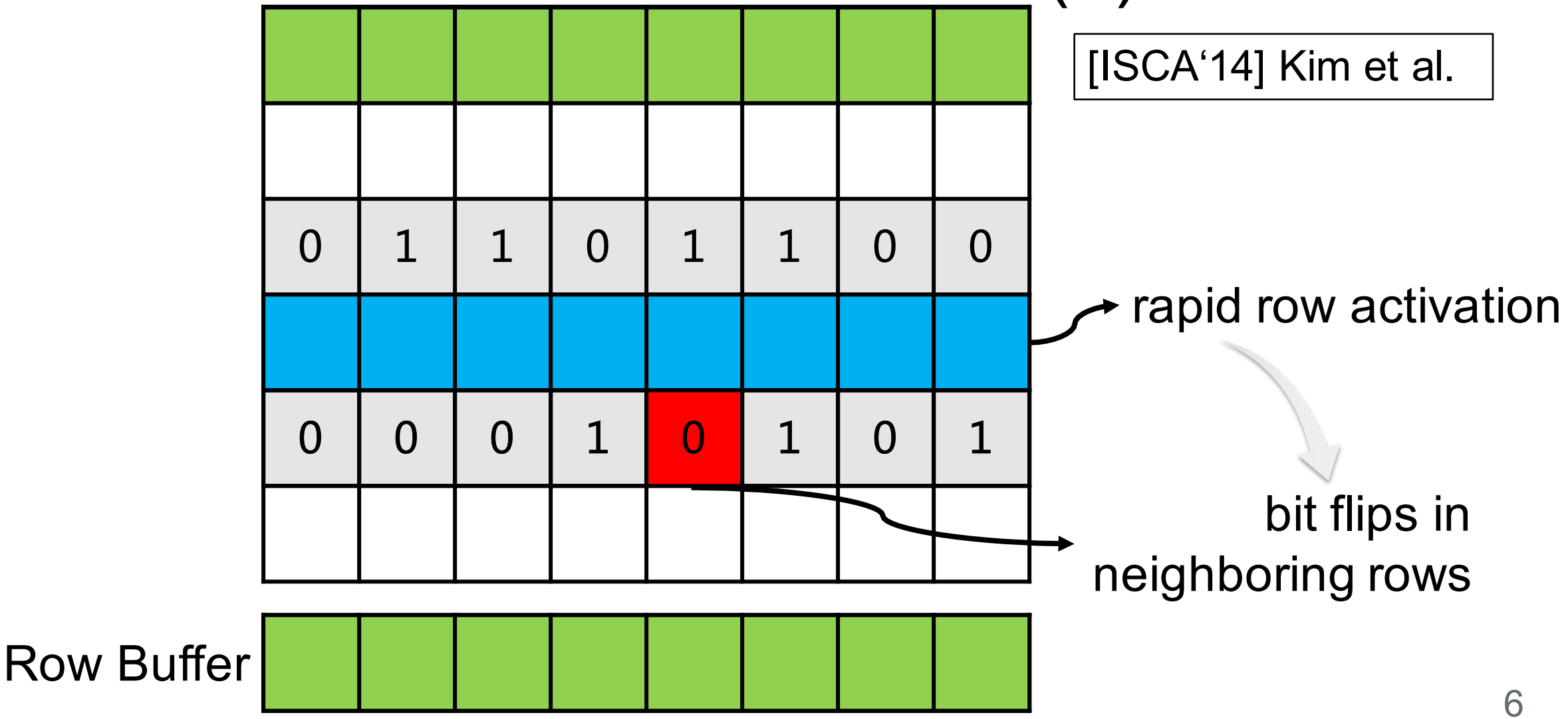


Disturbance Errors (1)

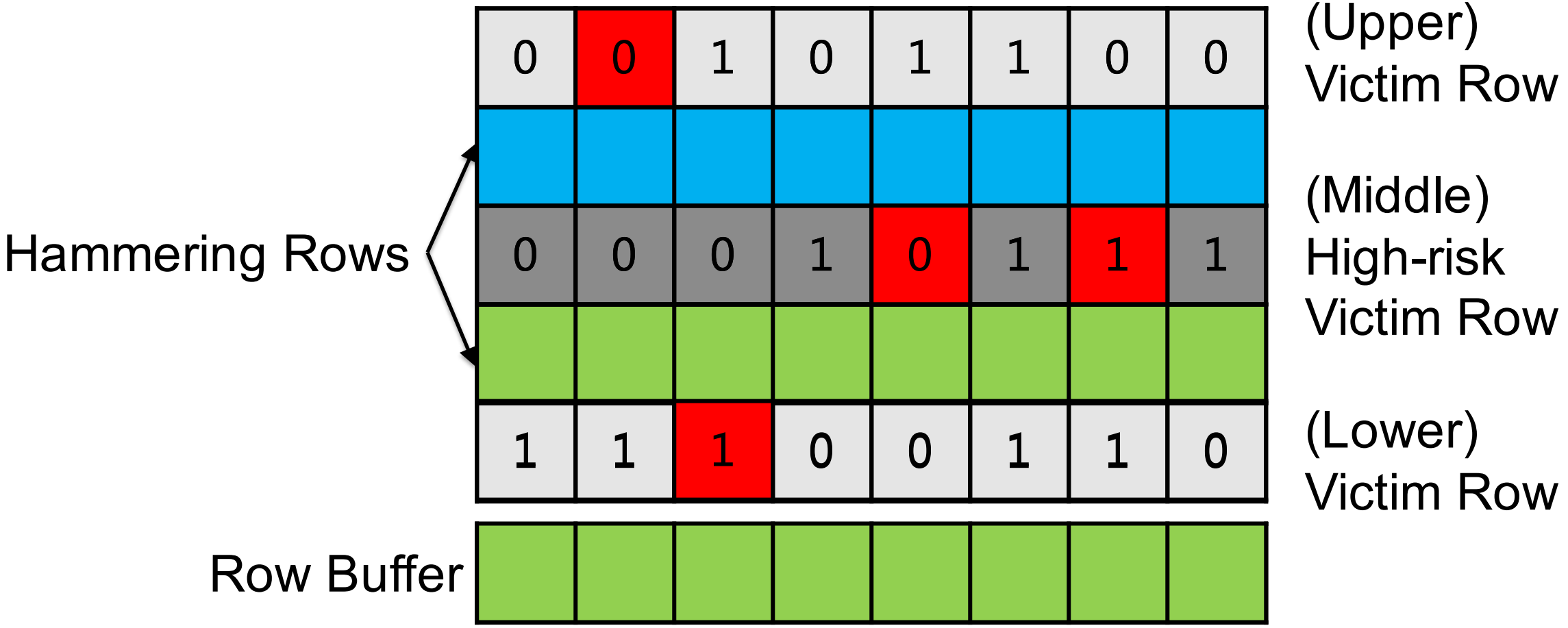


Disturbance Errors (2)

[ISCA'14] Kim et al.



Double-sided Row Hammer Attack



Related Work

- Kim, et al. [ISCA '14]
- Seaborn (Google), **2015**
- R3dF09 (Tencent), **2015**
- Bosman, et al. [SOSP '16]
- Gruss, et al. [DIMVA '16]

- Single-sided vs. double-sided row hammer
- Exploitation scenarios

How to conduct double-sided row hammer attacks?



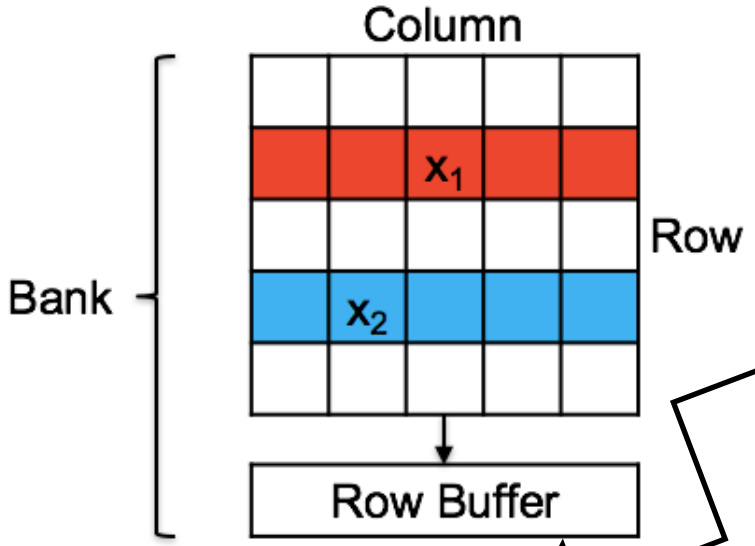


How to exploit row hammer vulnerabilities in cross-VM settings?

Outline

1. Reverse-engineer DRAM Mapping
2. Cross-VM Row Hammer Exploitation
3. Case studies

Difficulties in Double-sided Row Hammer

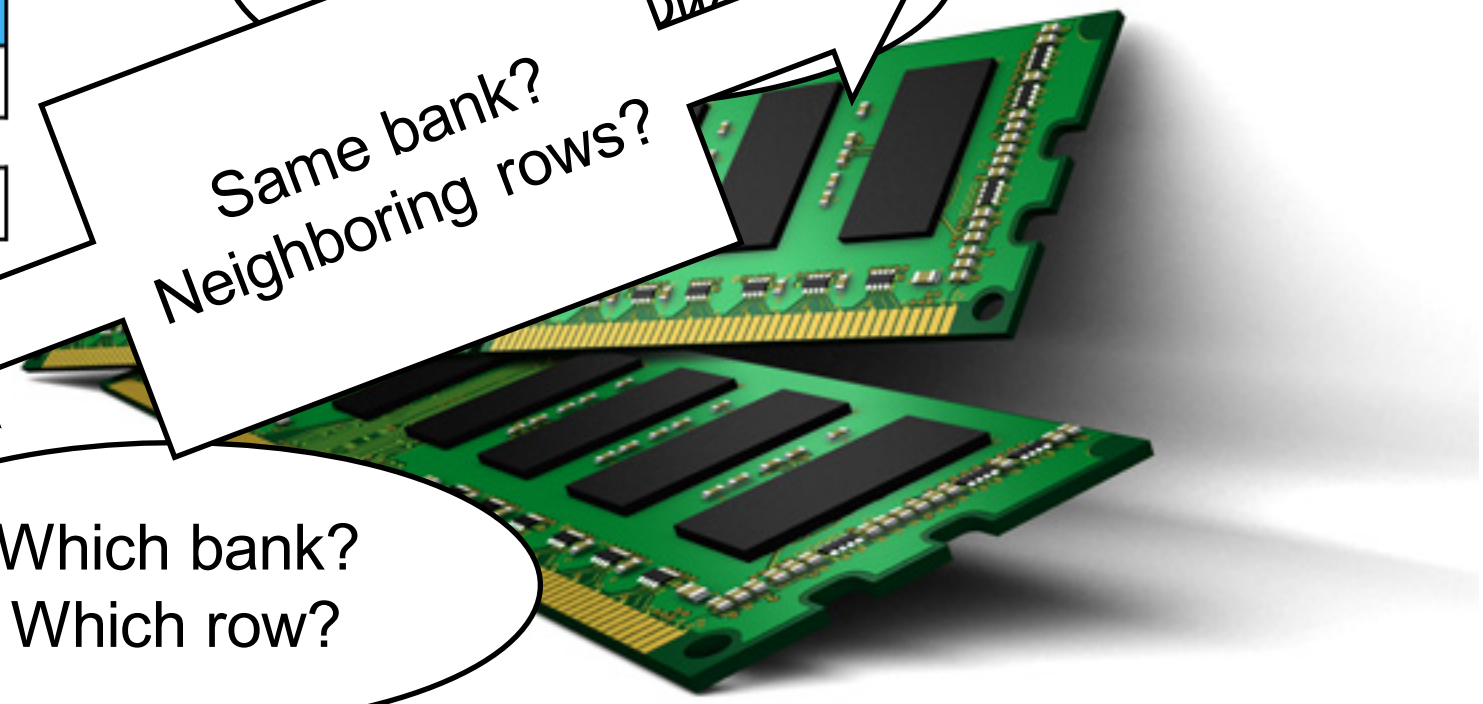


Physical Address
x_1

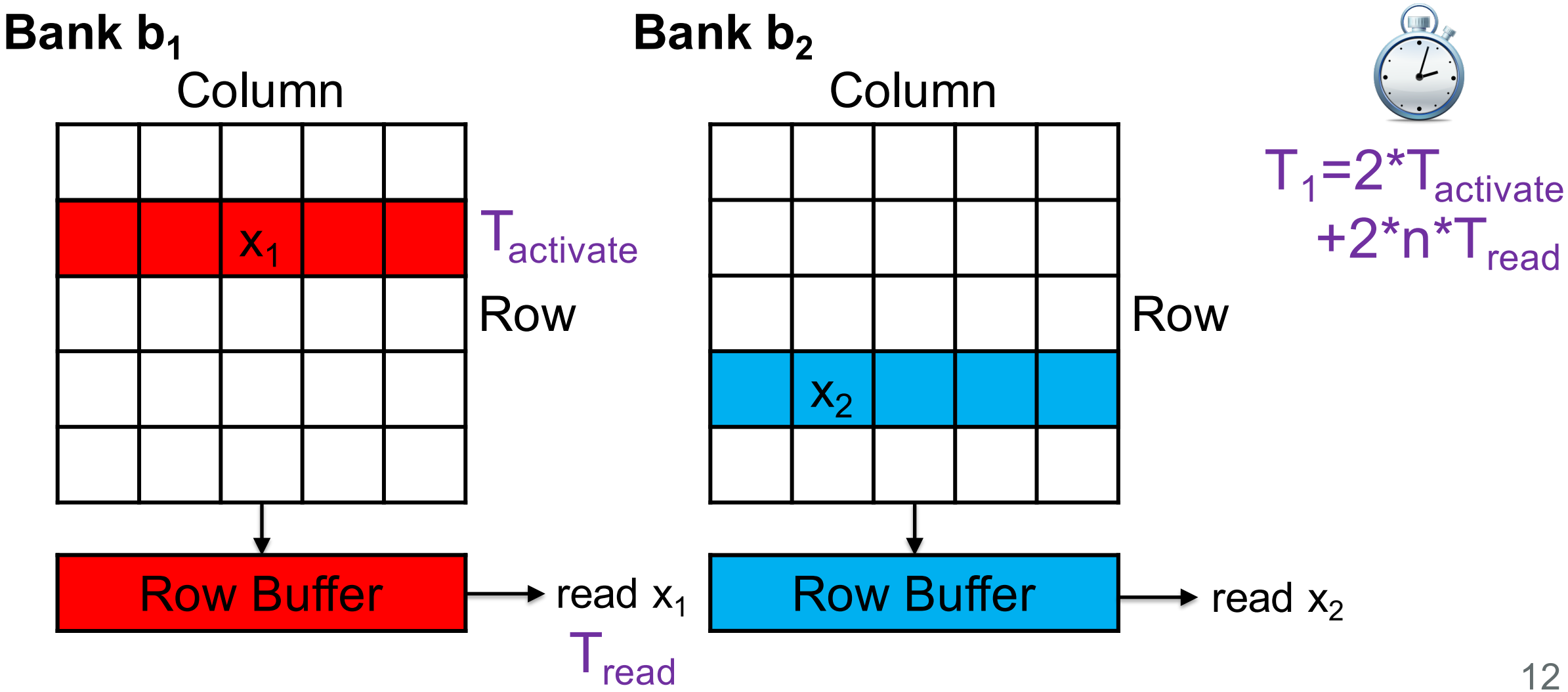
Same bank?
Neighboring rows?

Physical Address
x_2

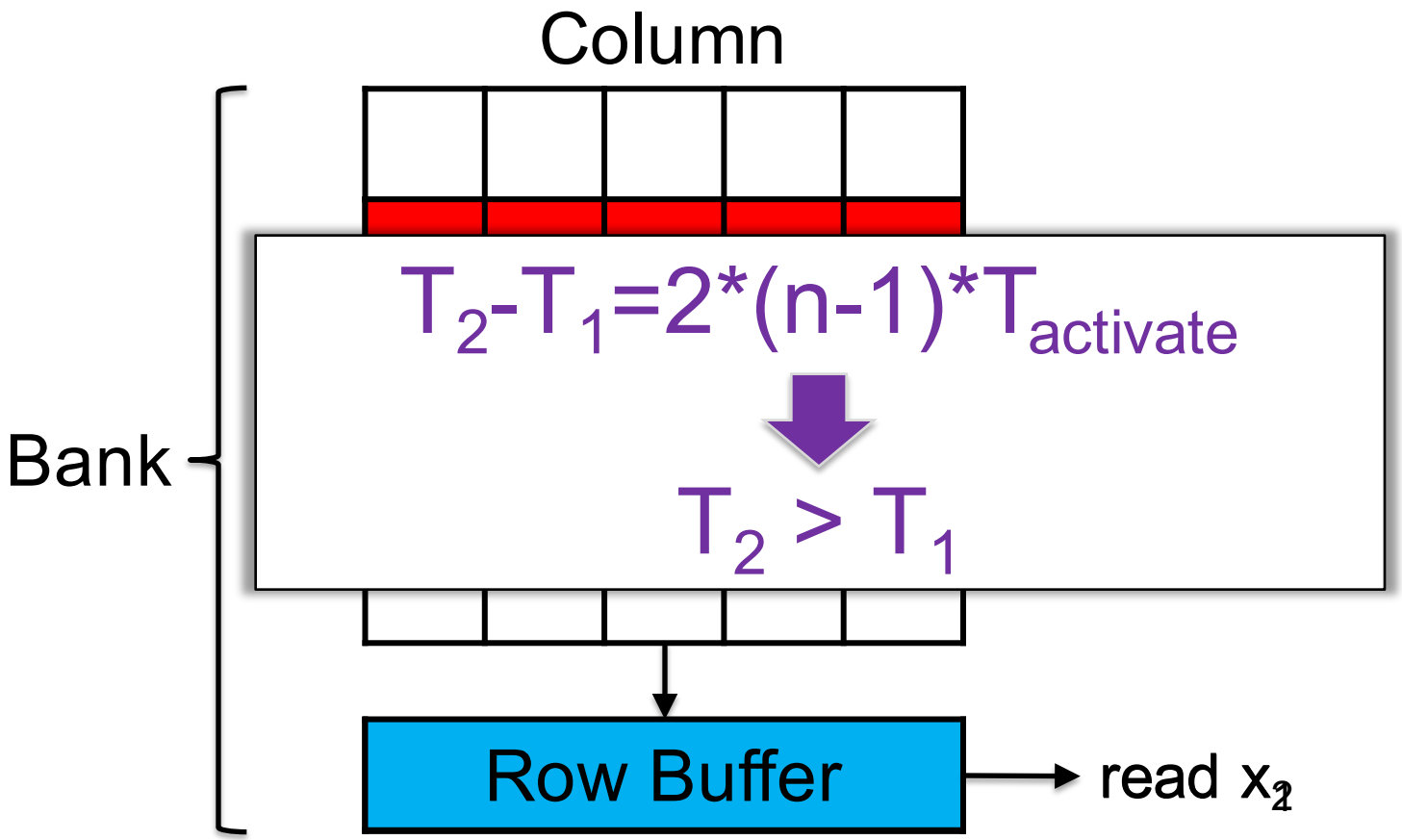
Which bank?
Which row?



A Basic Timing Channel Primitive



Basic timing channel primitive





T_1=2*$T_{activate}$ +2*n*T_{read}



T_2=2*n*$T_{activate}$ +2*n*T_{read}

Basic timing channel primitive



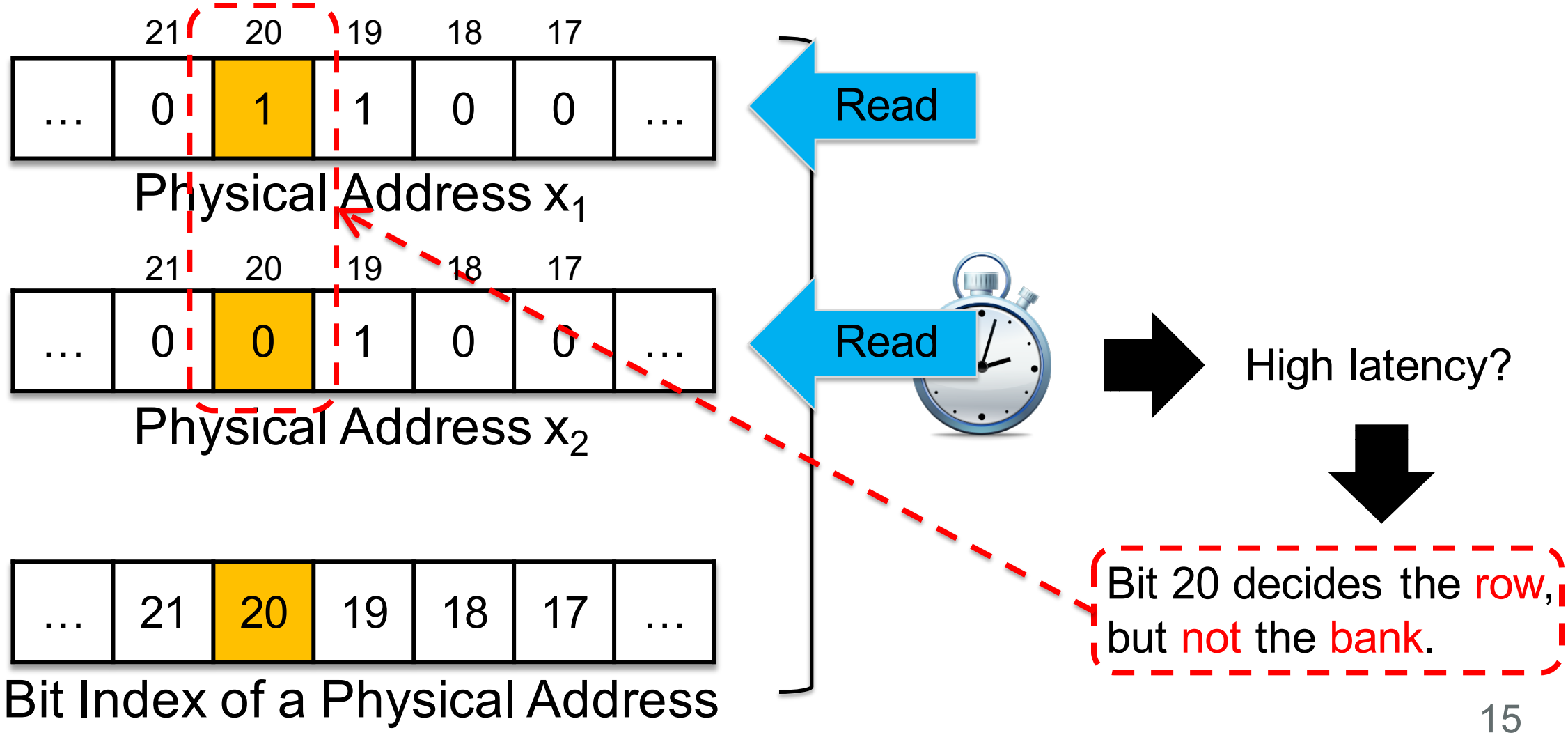
High latency

- same bank
- different row

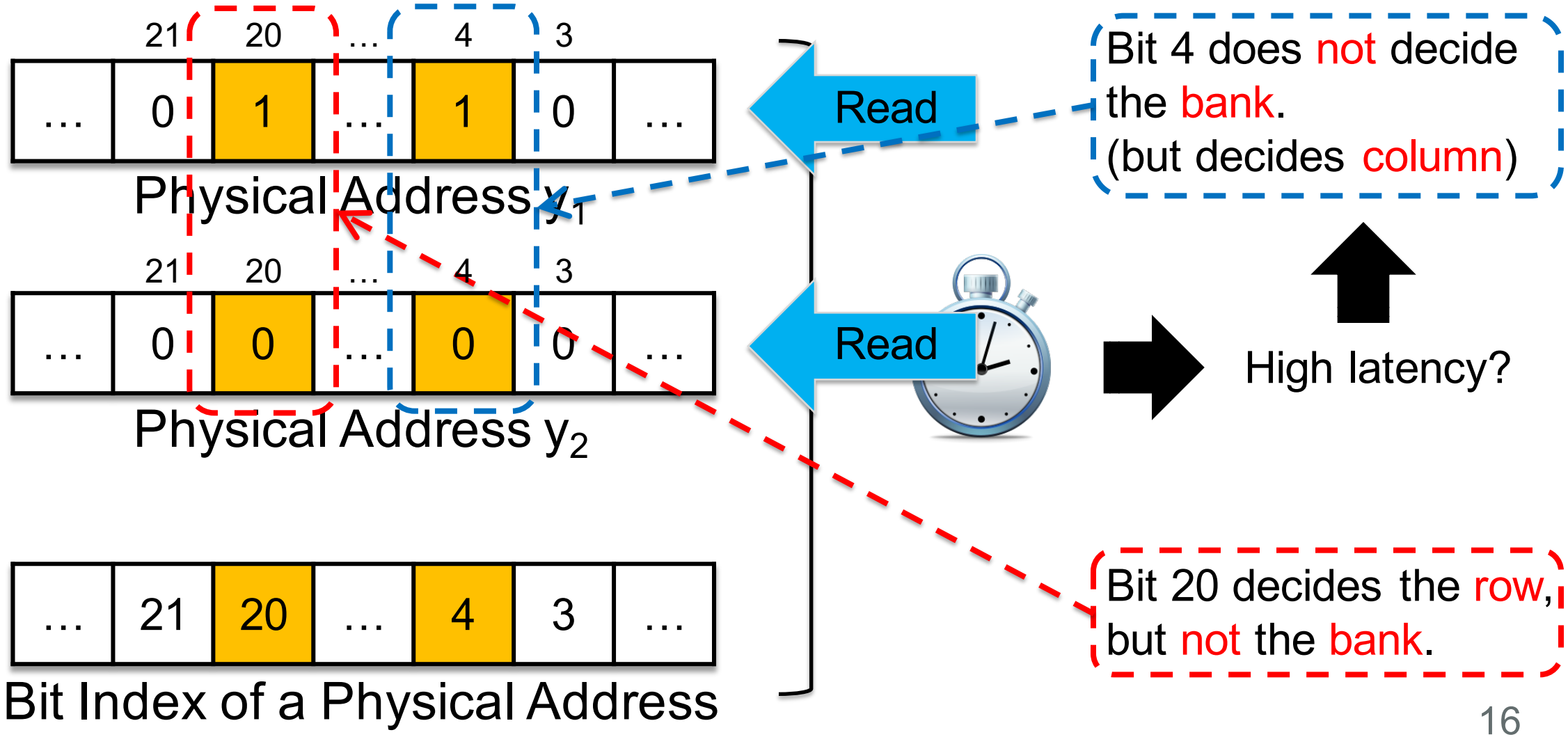


What if the two physical addresses differ only in specific bit positions?

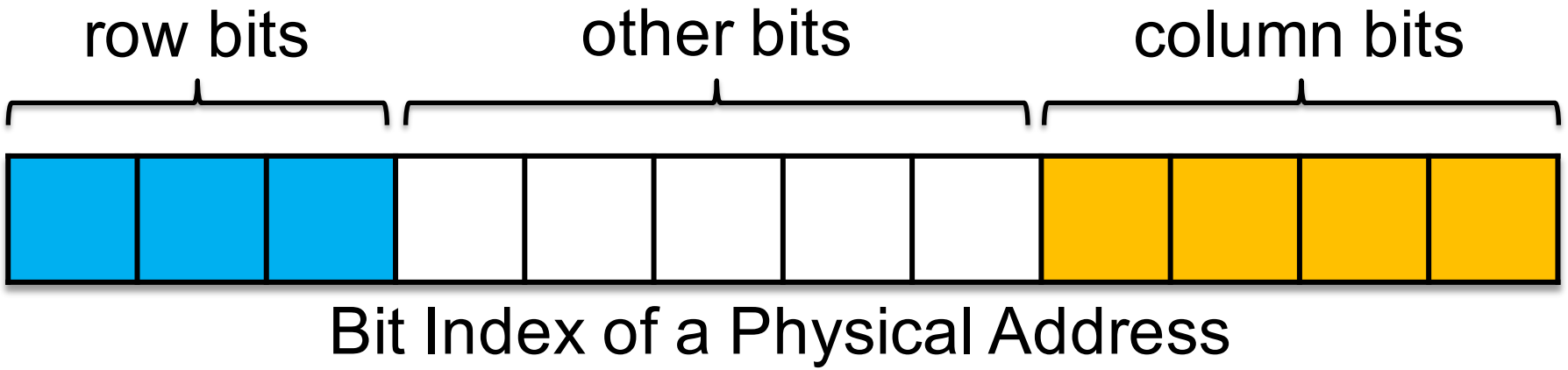
Row Bit Detection



Column Bit Detection

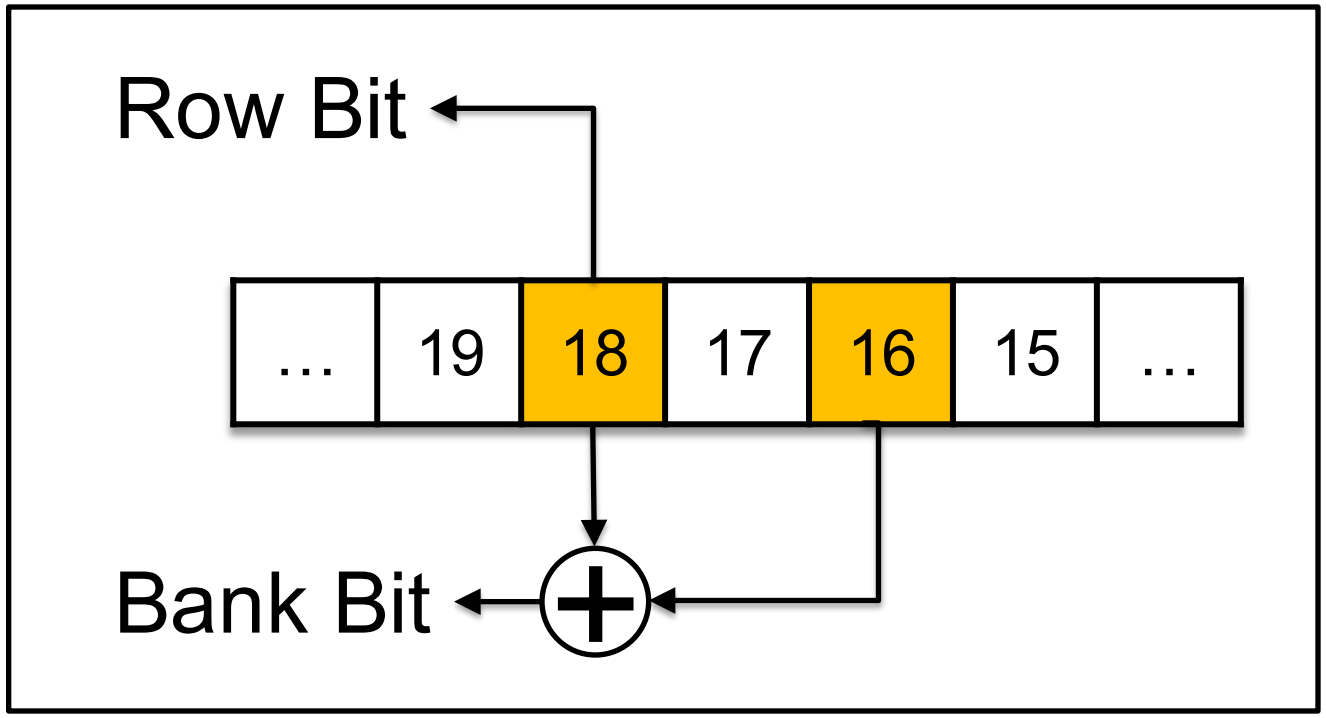


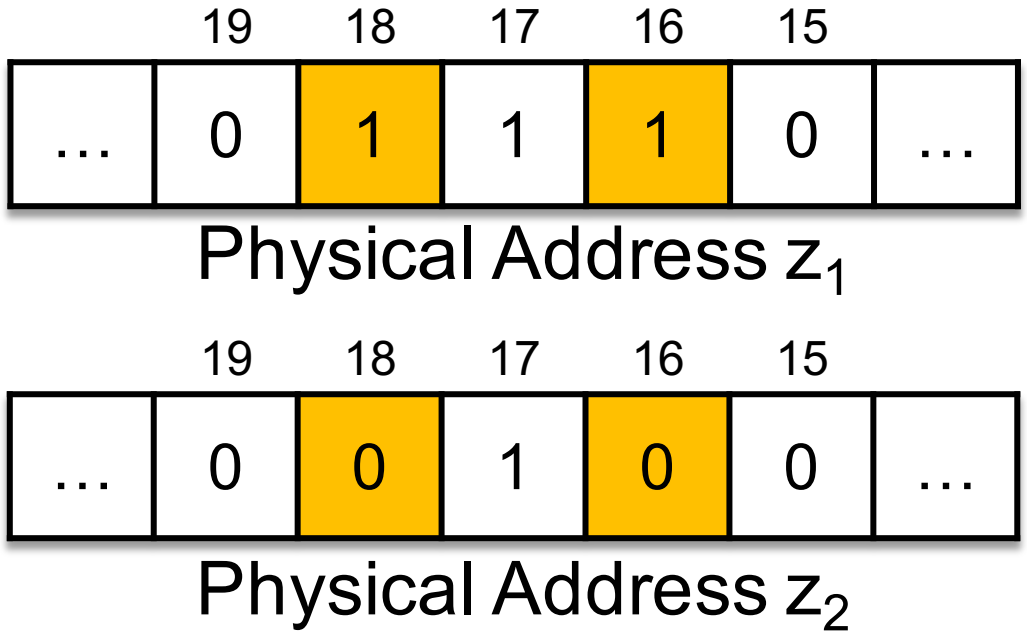
Row/Column Bit Summary



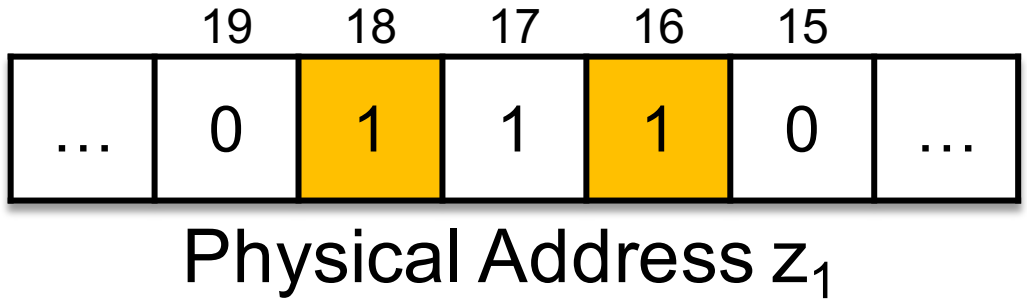
White bits: always show low latency in previous two kinds of tests.

XOR Scheme Detection (1)





XOR Scheme Detection (2)



Read

Read

Bit 18 (higher bit) should be row bit



High latency?

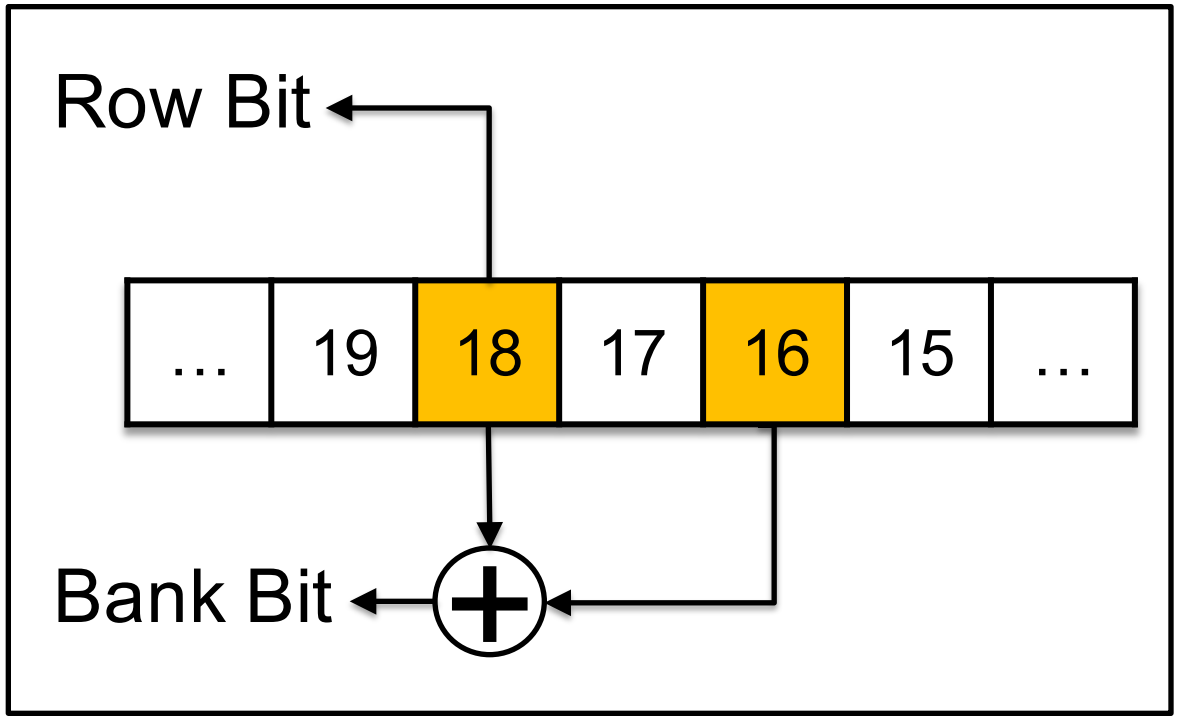
- different row
- same bank

Either 18 or 16 decides the row

XOR Scheme Detection (3)

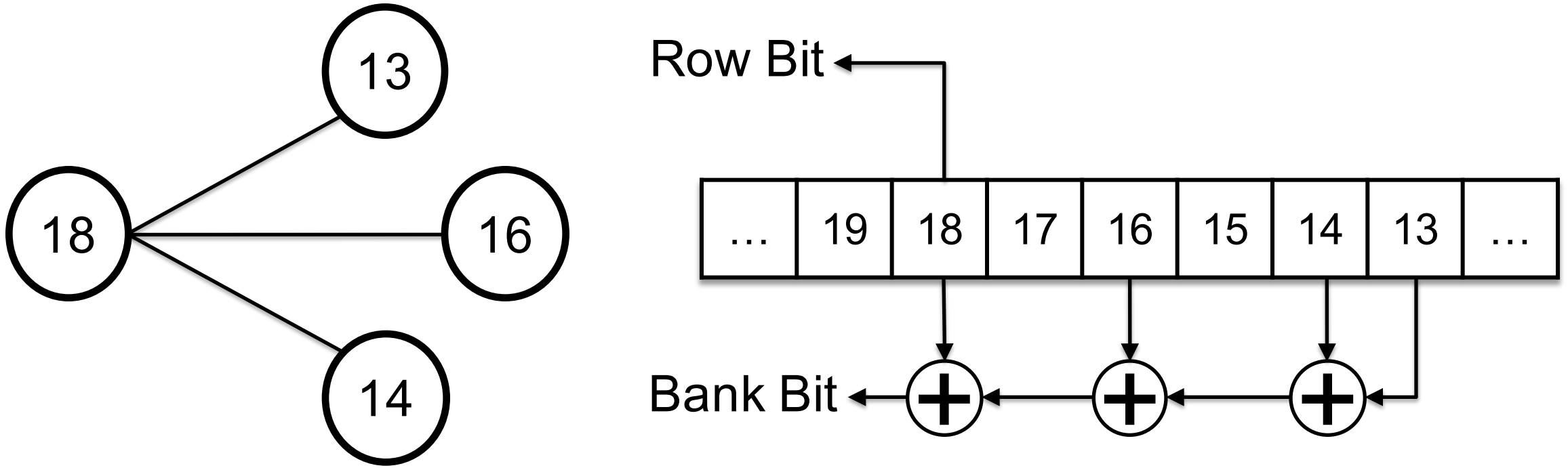
test	18	16	18 ⊕ 16	latency	bank	row
1	1 0	1 1	0 1	low	different	different
2	1 0	1 0	0 0	high	same	different
3	1 0	0 1	1 1	high	same	different

XOR Scheme Detection (4)

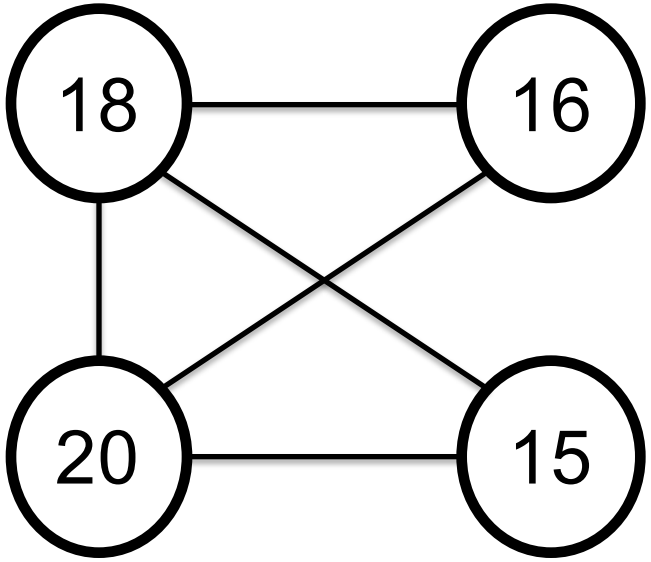


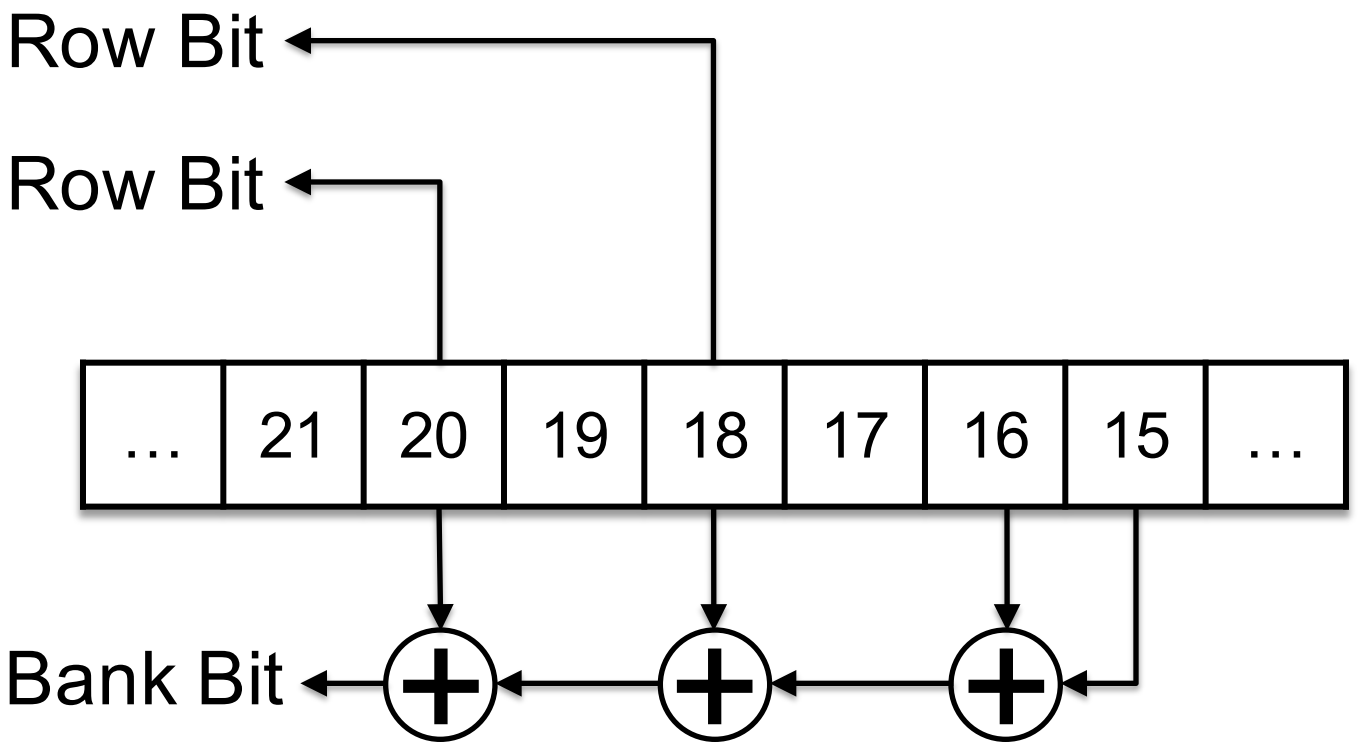
18 ⊕ 16	bank
0 1	different
0 0	same
1 1	same

Graph-based Bit Detection Algorithm

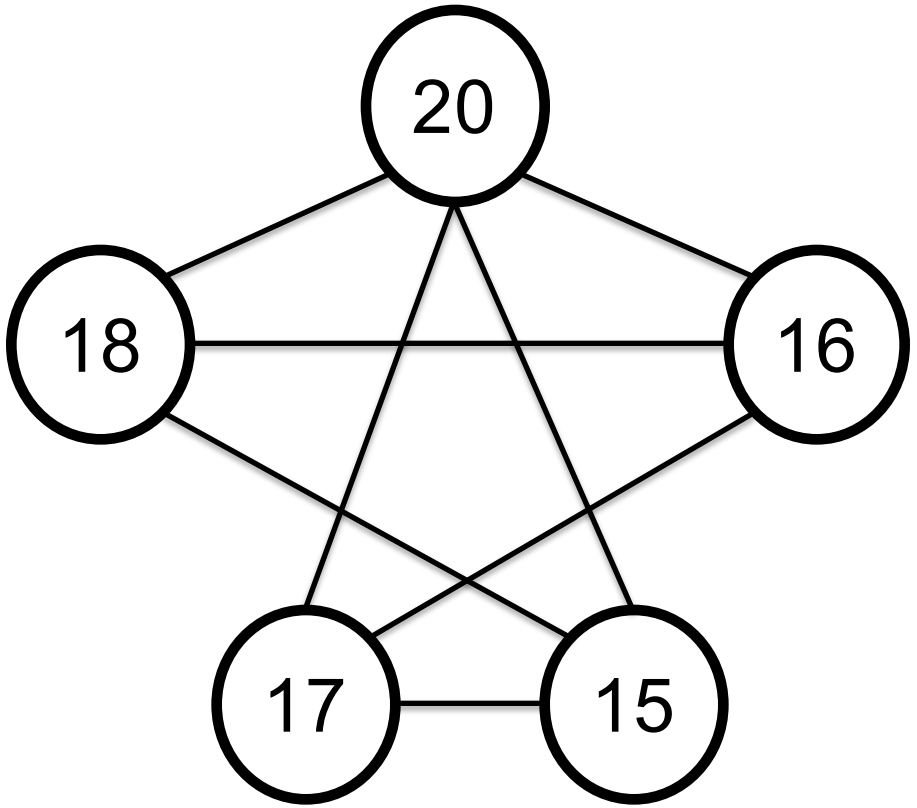


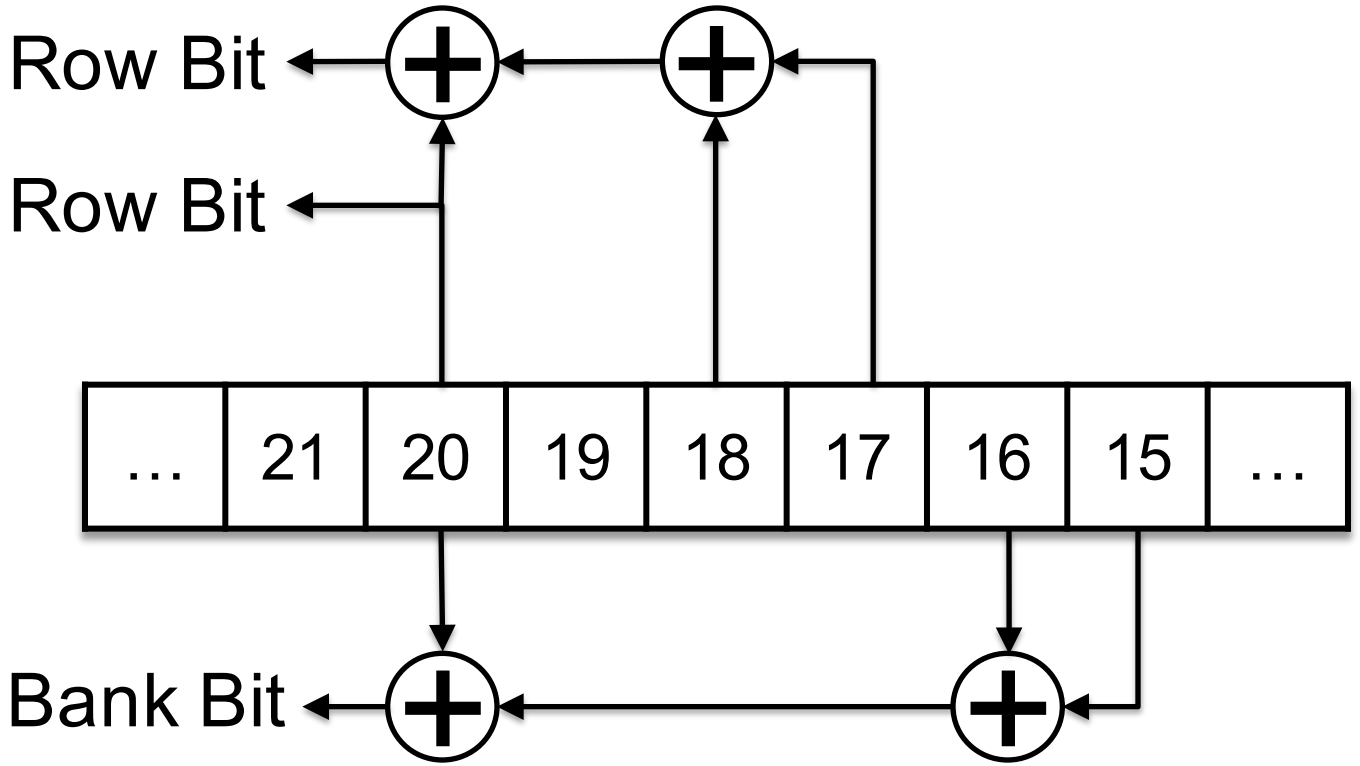
Graph-based Bit Detection Algorithm





Graph-based Bit Detection Algorithm

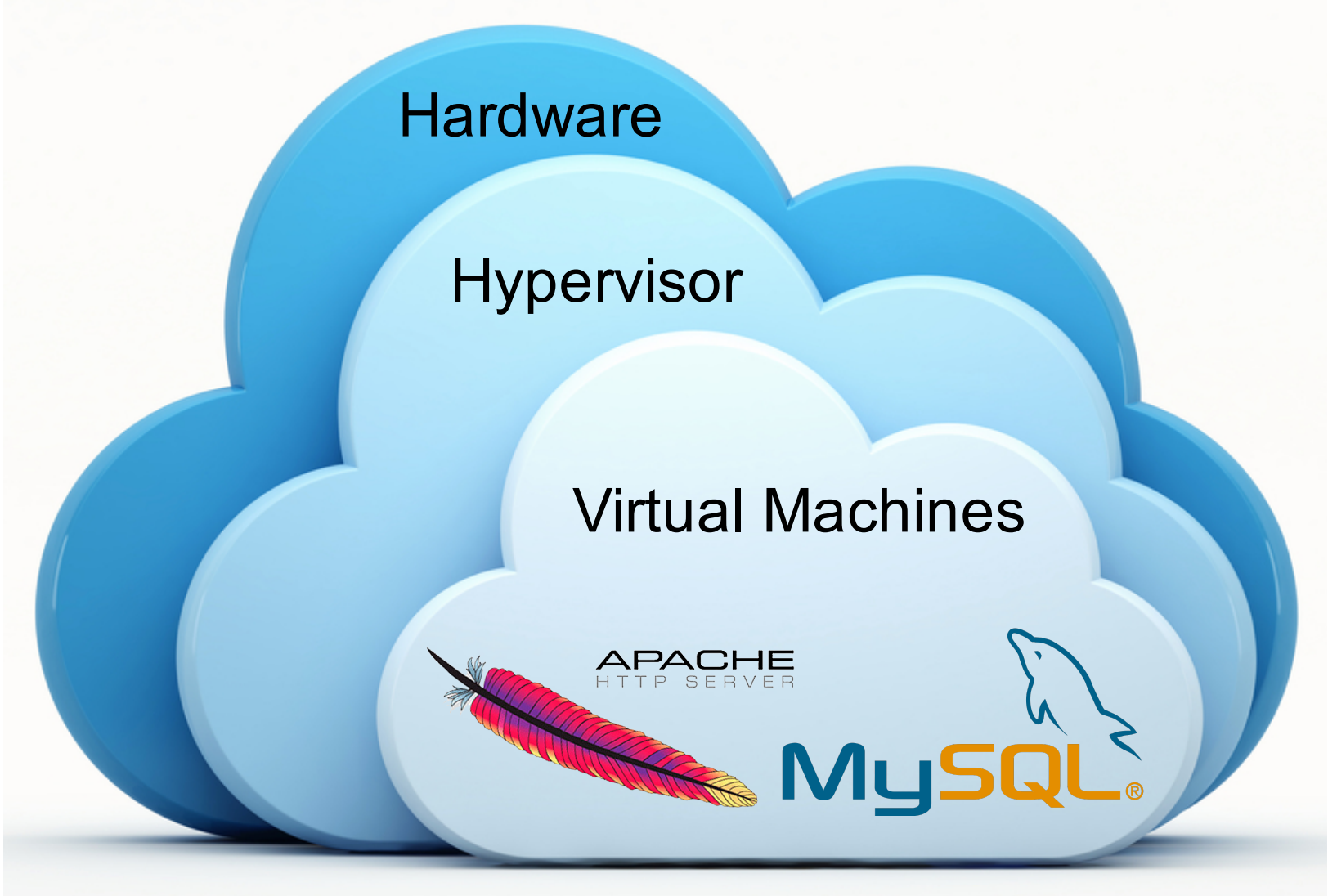




Outline

1. Reverse-engineer DRAM Mapping
2. Cross-VM Row Hammer Exploitation
3. Case studies

Virtualization and Cloud Computing



Xen Para-Virtualized instances in Real-world Public Clouds

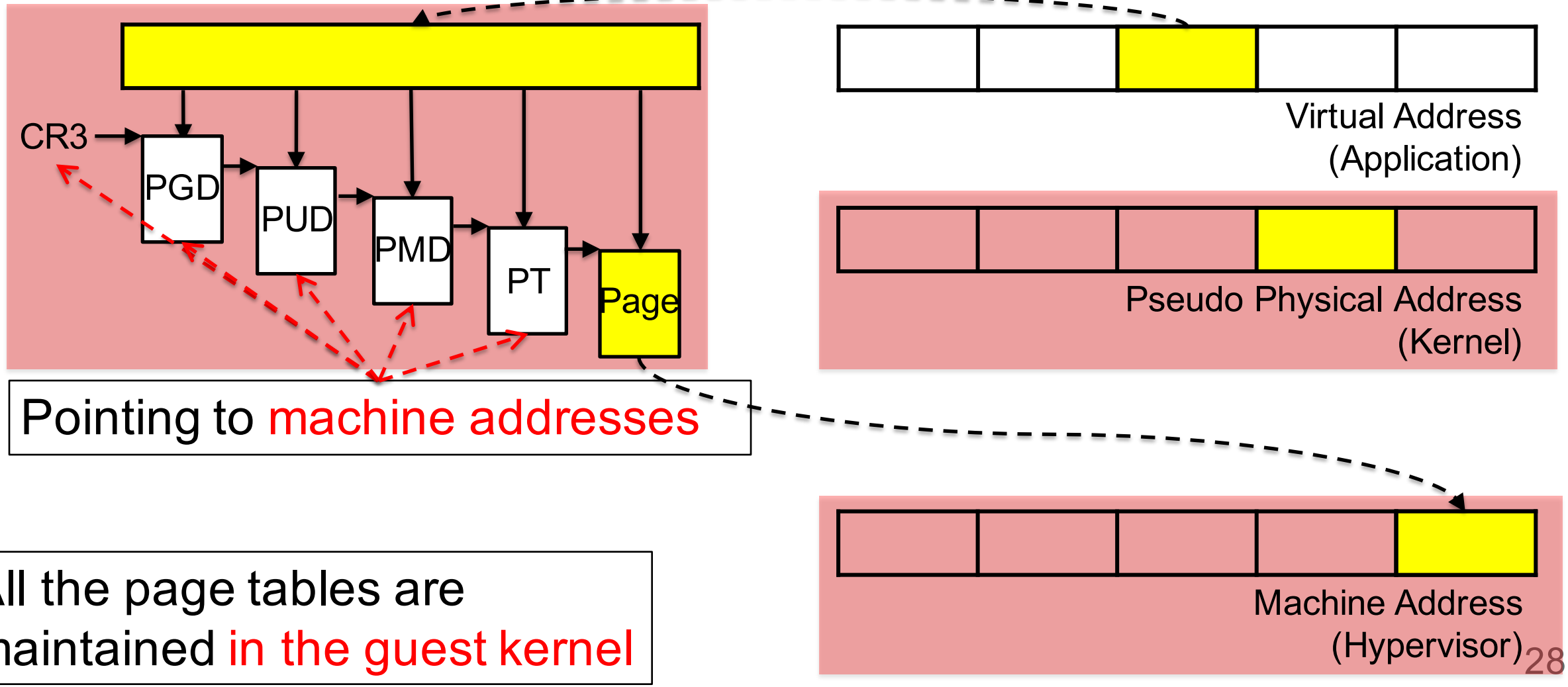
amazon web services™ | EC2

SOFTLAYER® an IBM Company

rackspace®

CloudLab

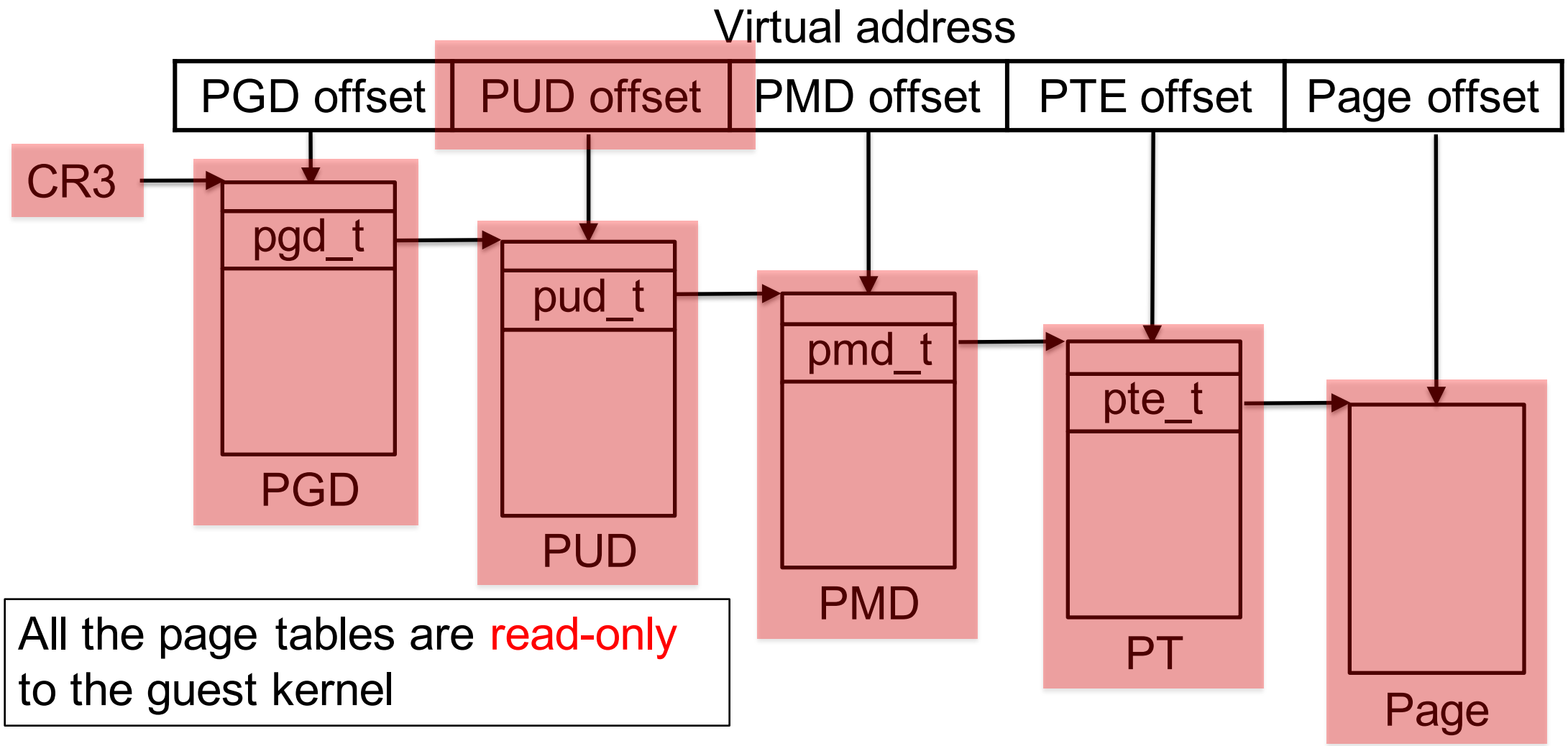
Xen Para-Virtualized Memory Management



Pointing to machine addresses

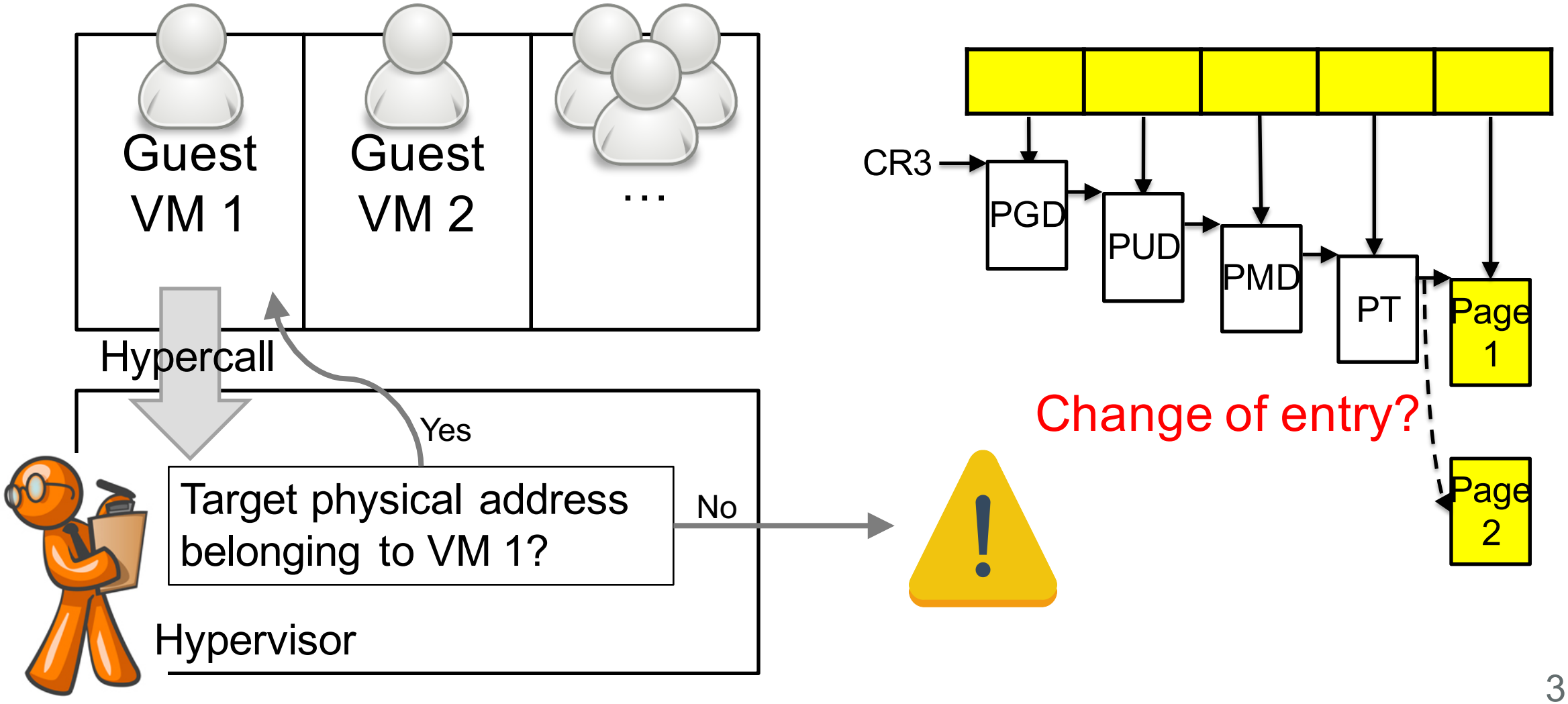
All the page tables are maintained in the guest kernel

Page Table Management

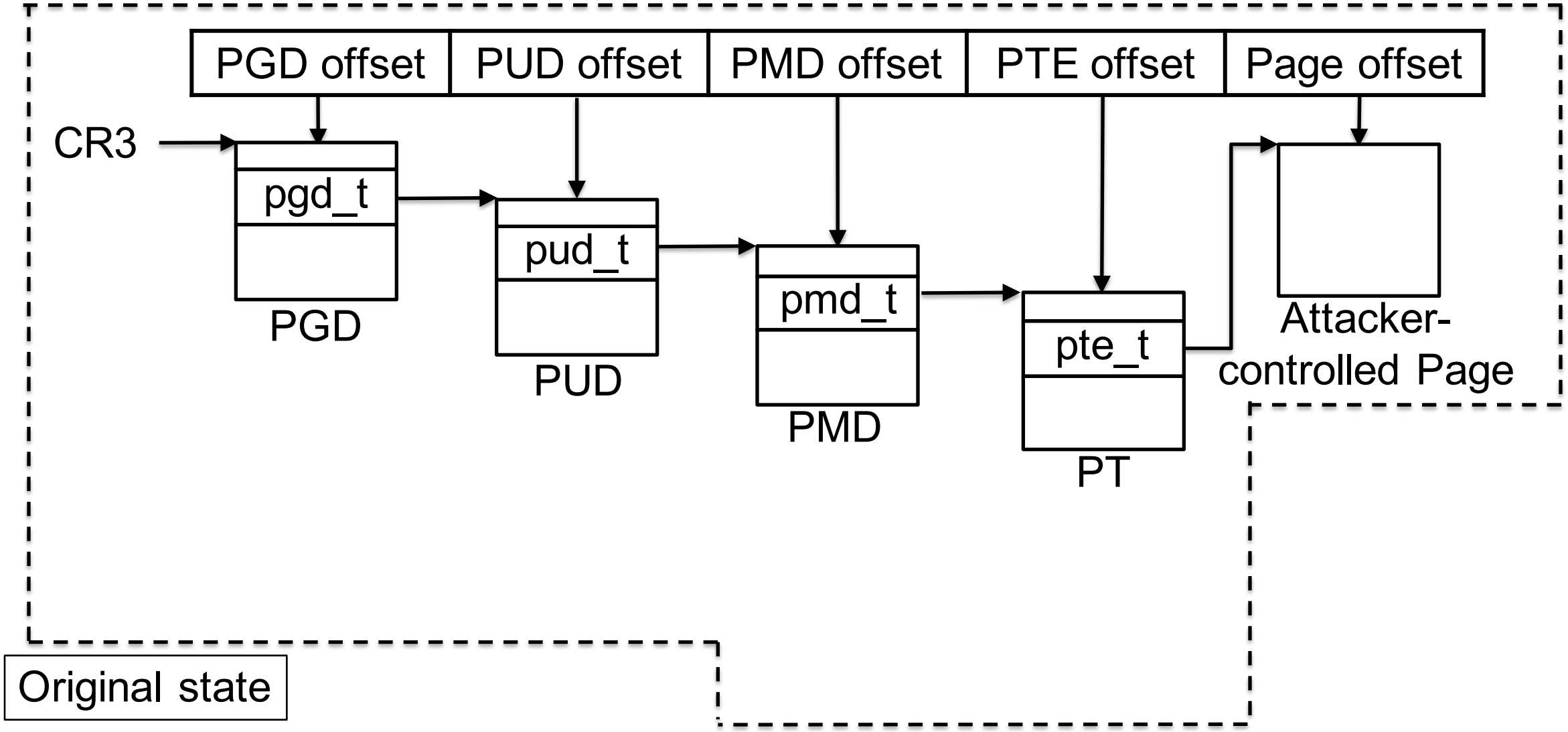


All the page tables are read-only to the guest kernel

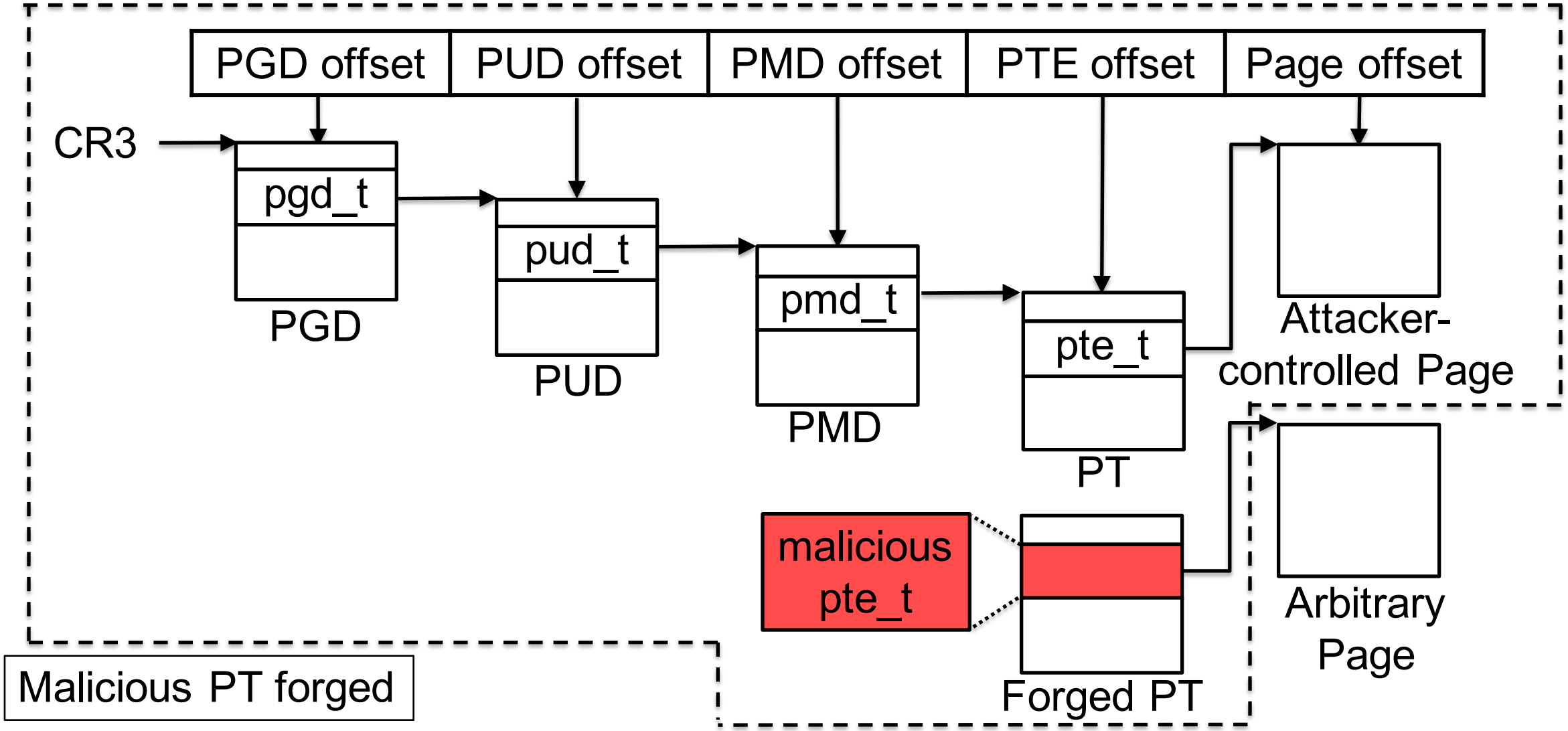
Xen PV memory management



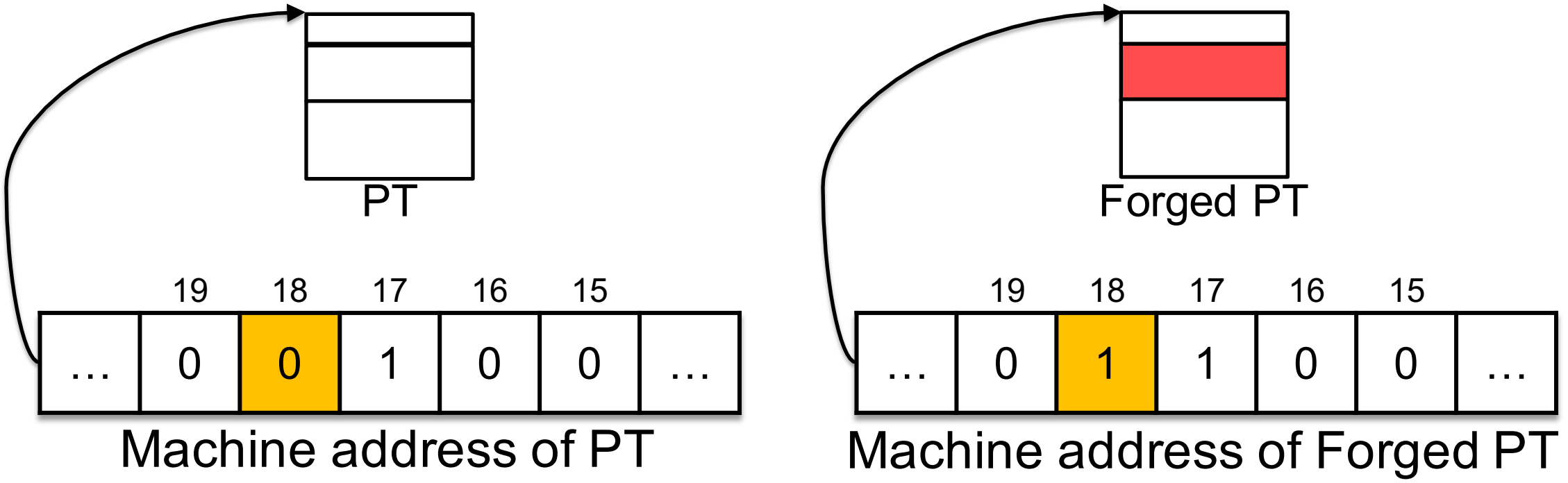
Page Table Replacement Attack (1)



Page Table Replacement Attack (2)

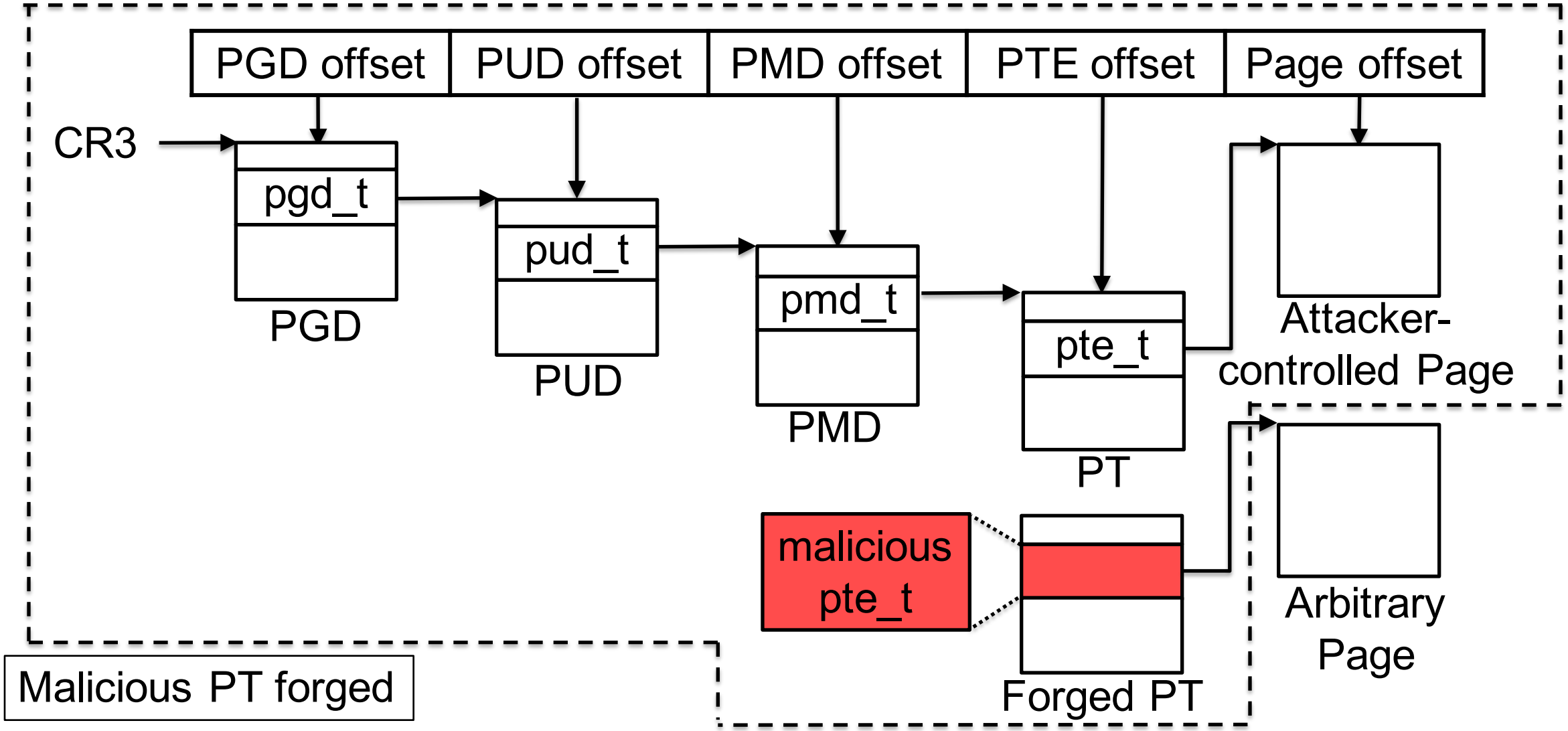


Page Table Replacement Attack (2)

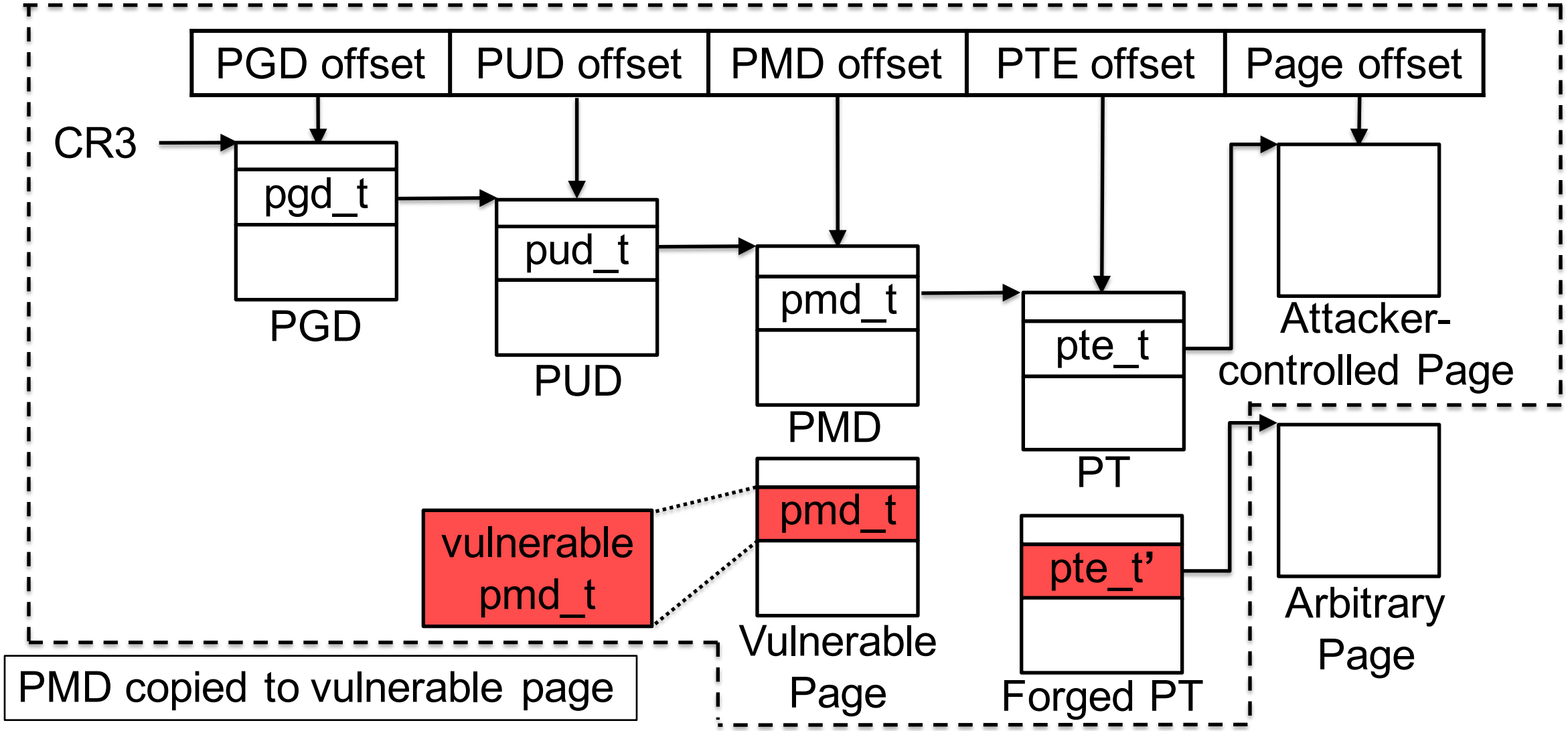


The addresses of PT and Forged PT only differ in one particular bit.

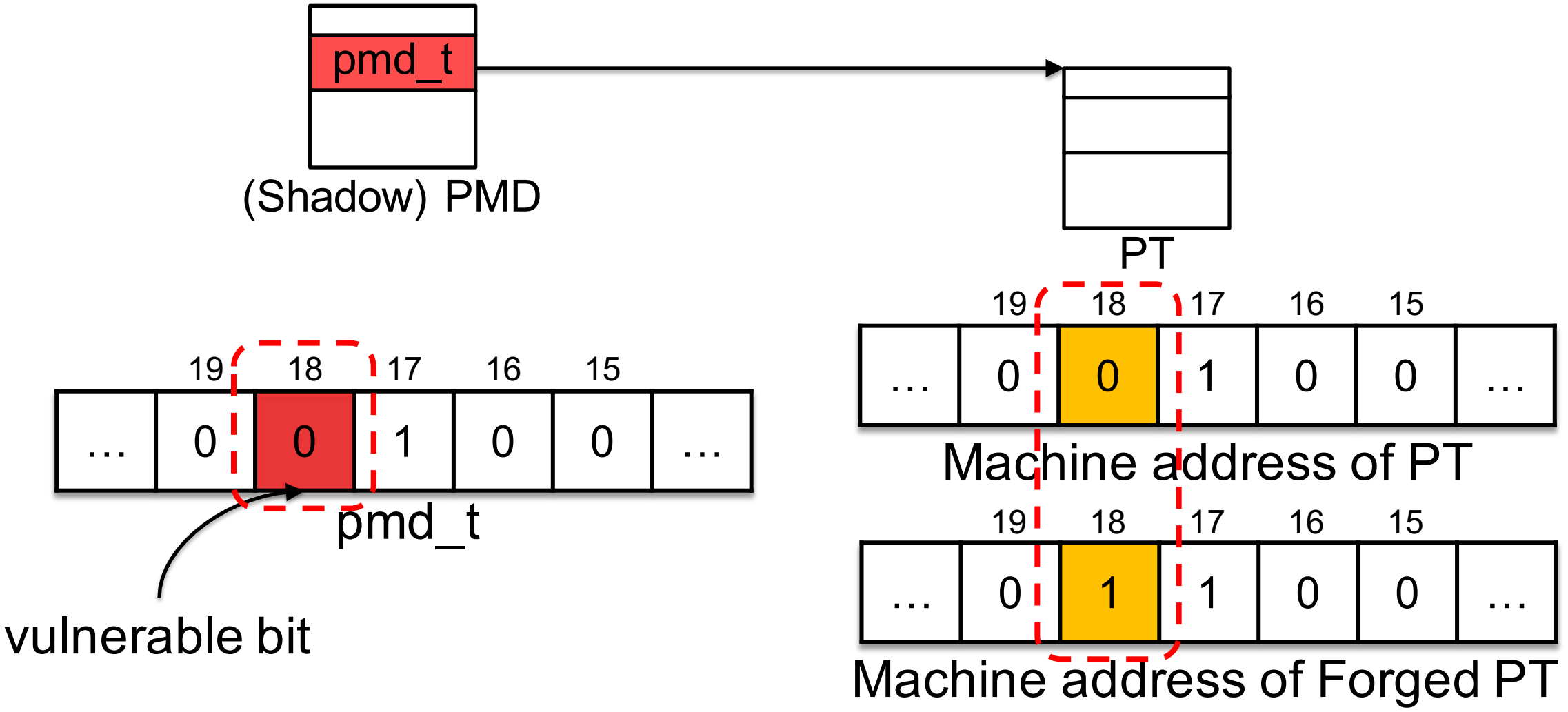
Page Table Replacement Attack (2)



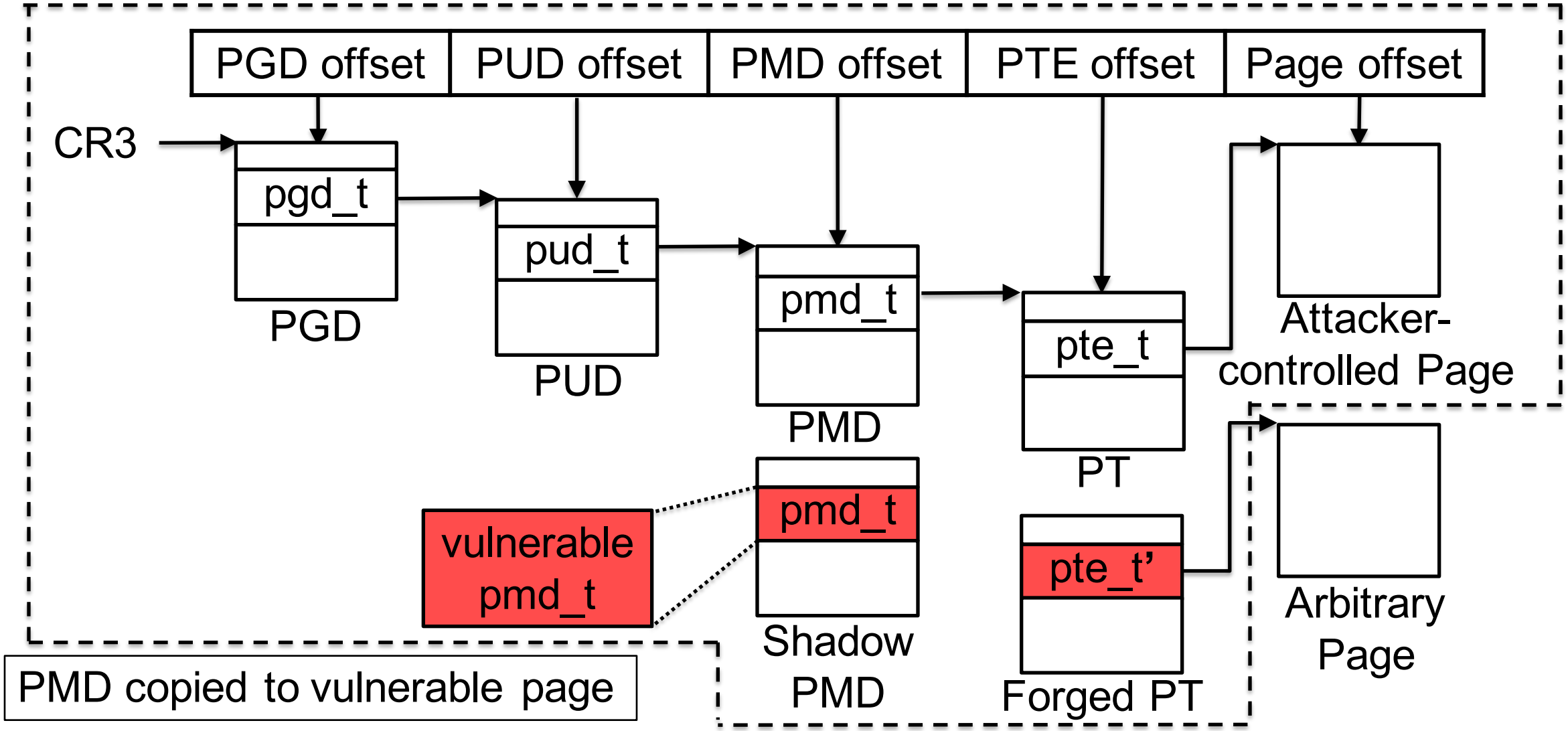
Page Table Replacement Attack (3)



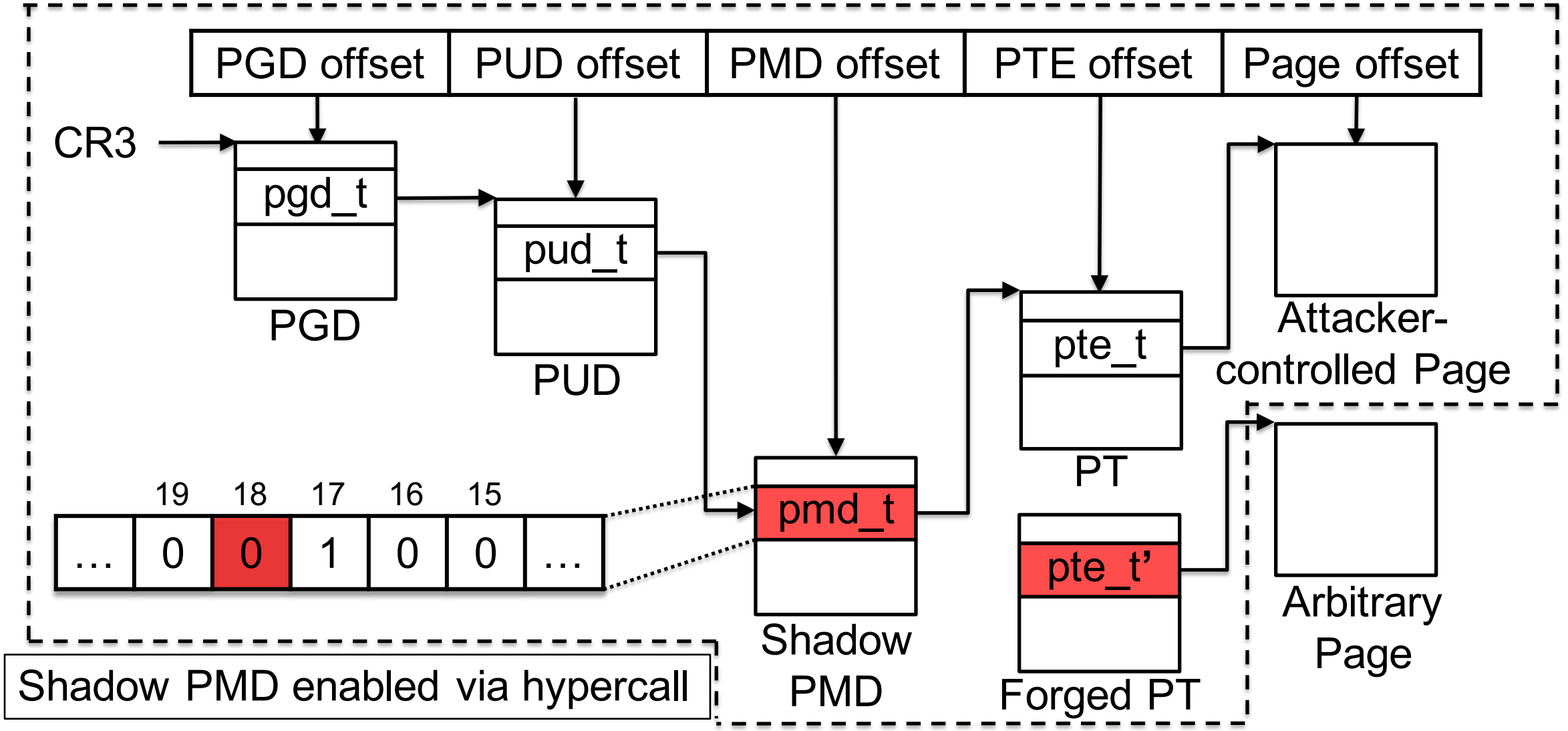
Page Table Replacement Attack (3)



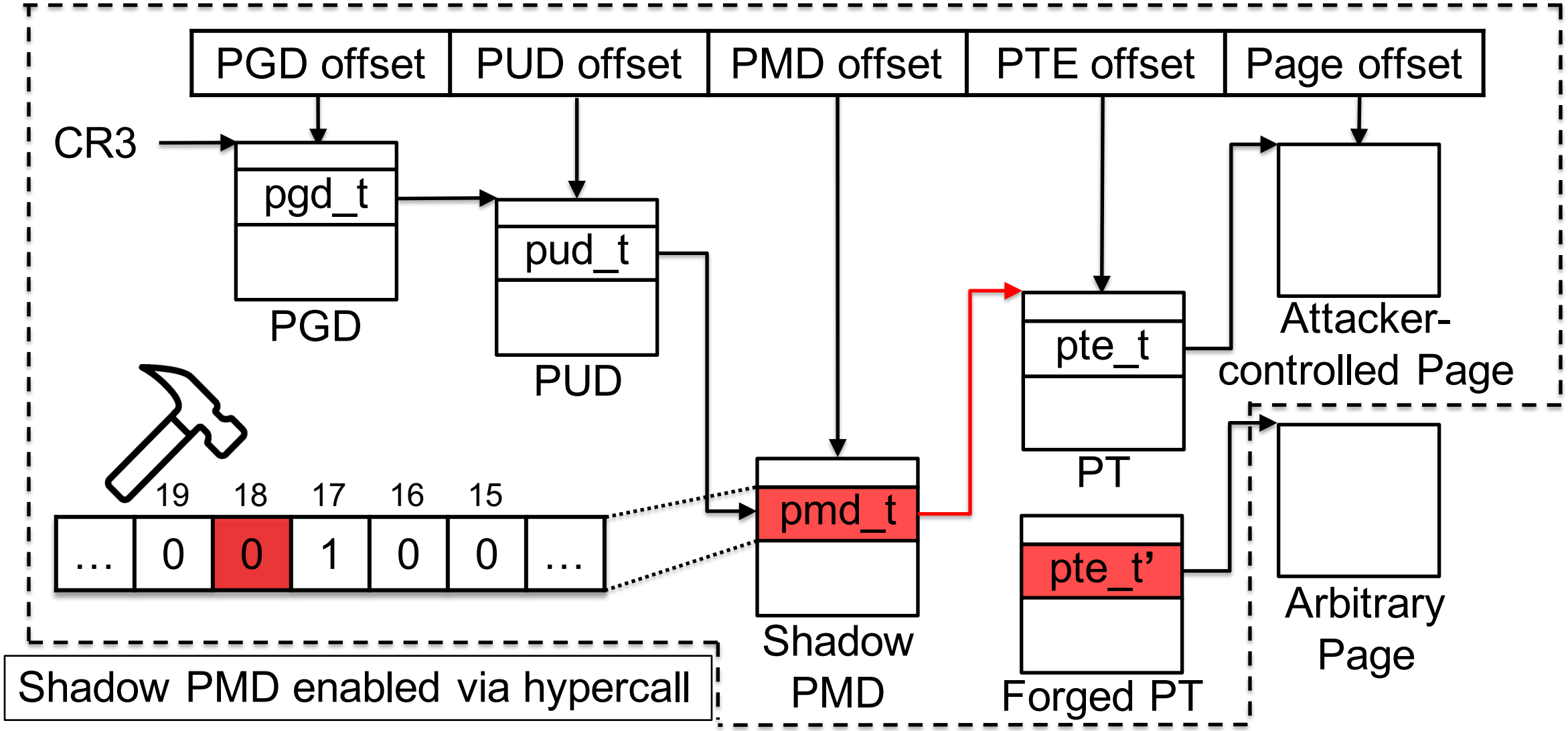
Page Table Replacement Attack (3)



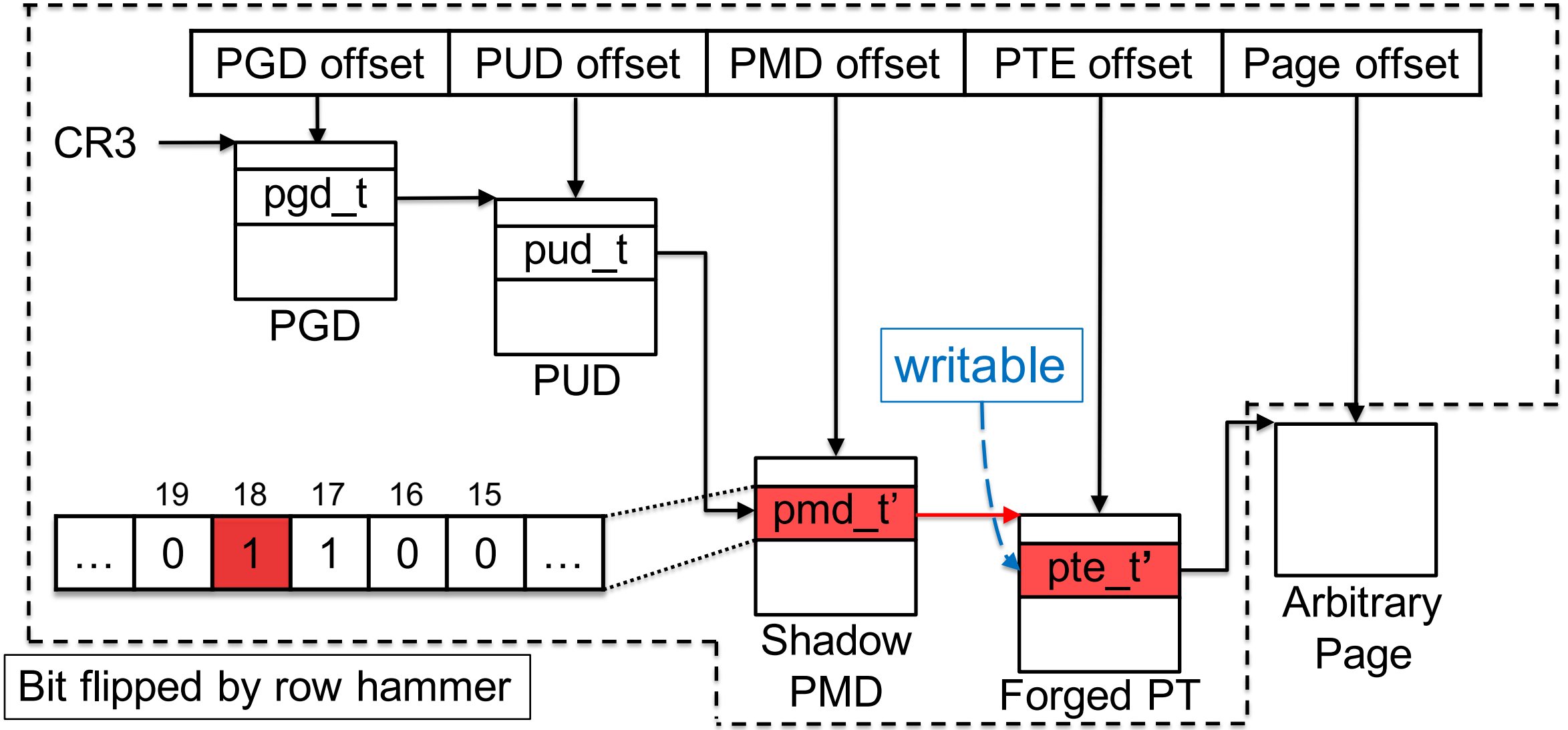
Page Table Replacement Attack (4)



Page Table Replacement Attack (4)



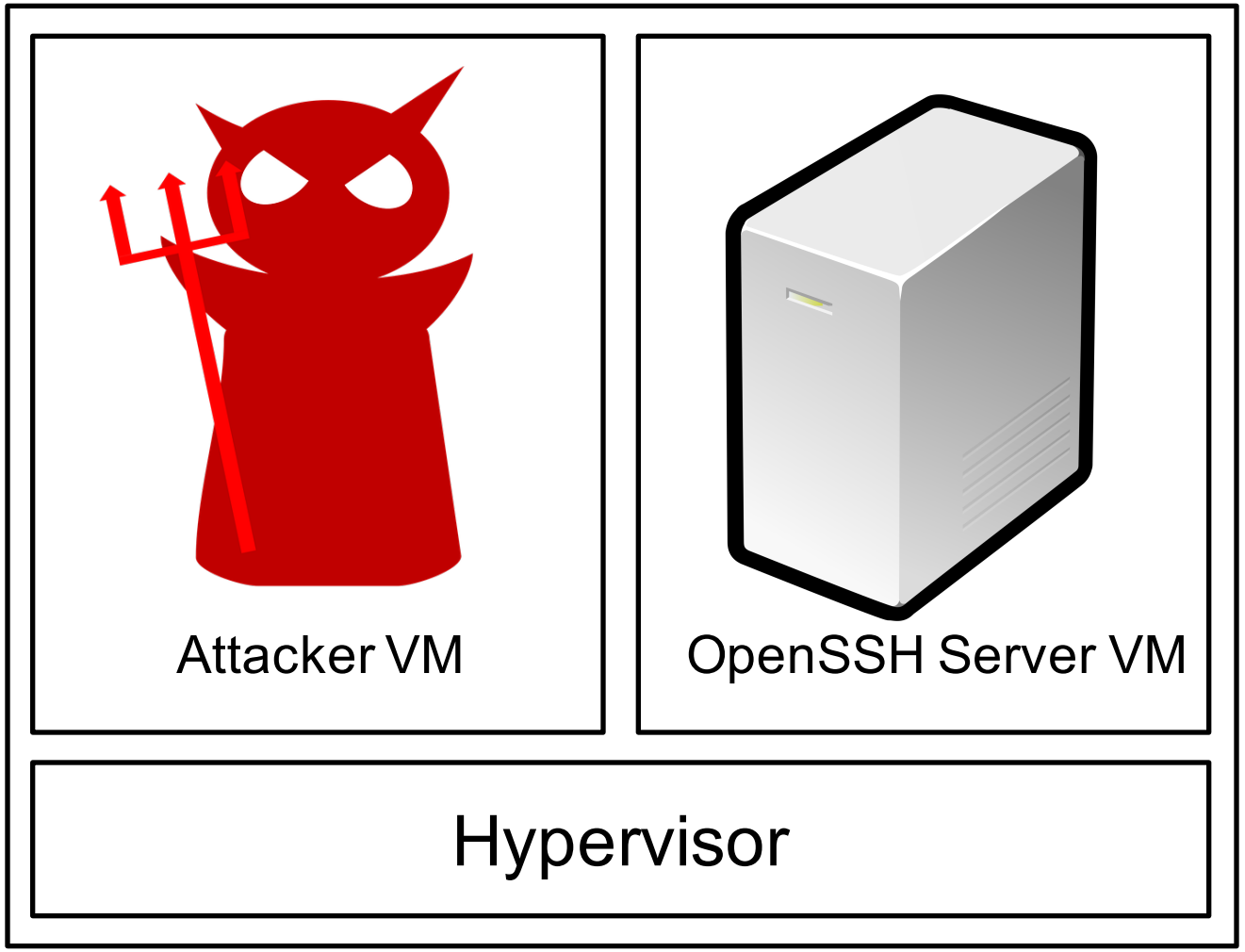
Page Table Replacement Attack (5)



Outline

1. Reverse-engineer DRAM Mapping
2. Cross-VM Row Hammer Exploitation
3. Case studies

OpenSSH Server Authentication Bypass



OpenSSH Server: Attack Target

Machine code: E8 1B 74 FD FF

```
callq pam_authenticate
test %eax, %eax
jne <error handling>
```

eax equals to 0?

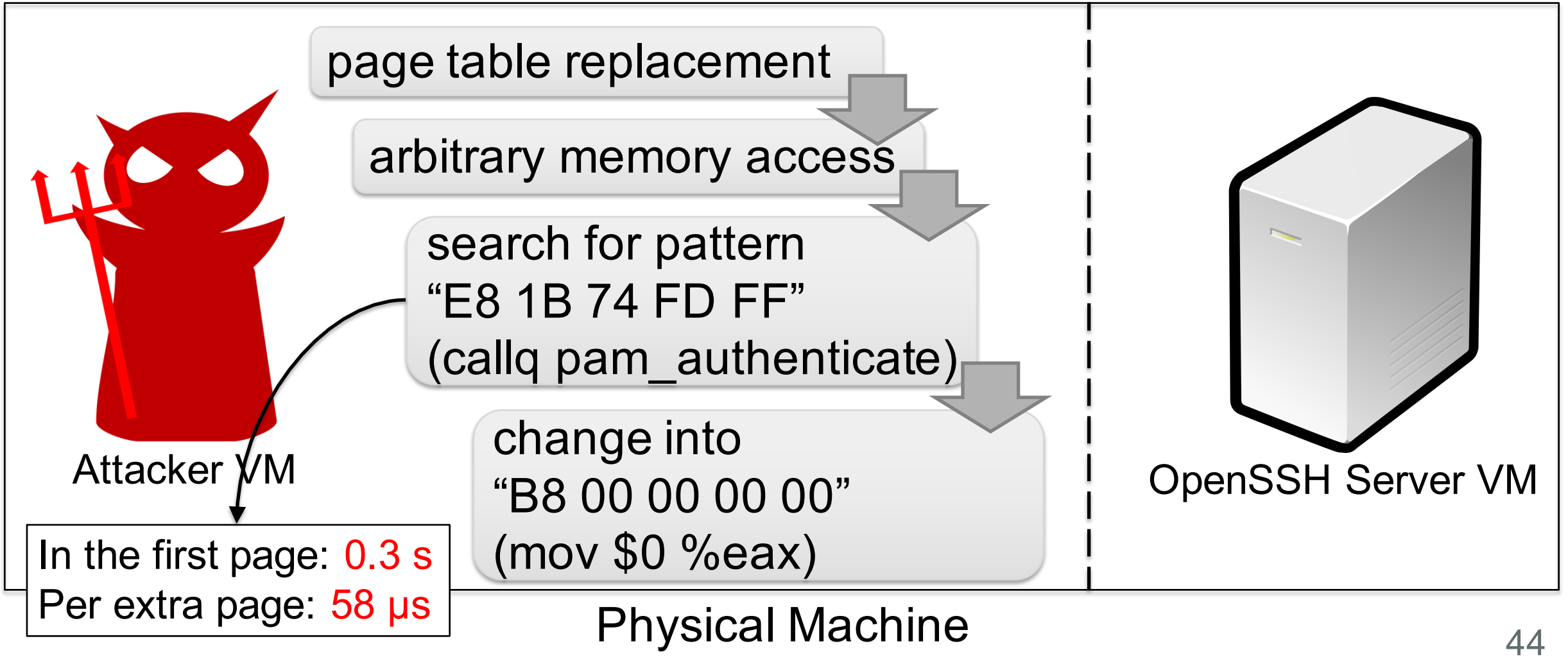
Machine code: B8 00 00 00 00

```
mov $0, %eax
test %eax, %eax
jne <error handling>
```

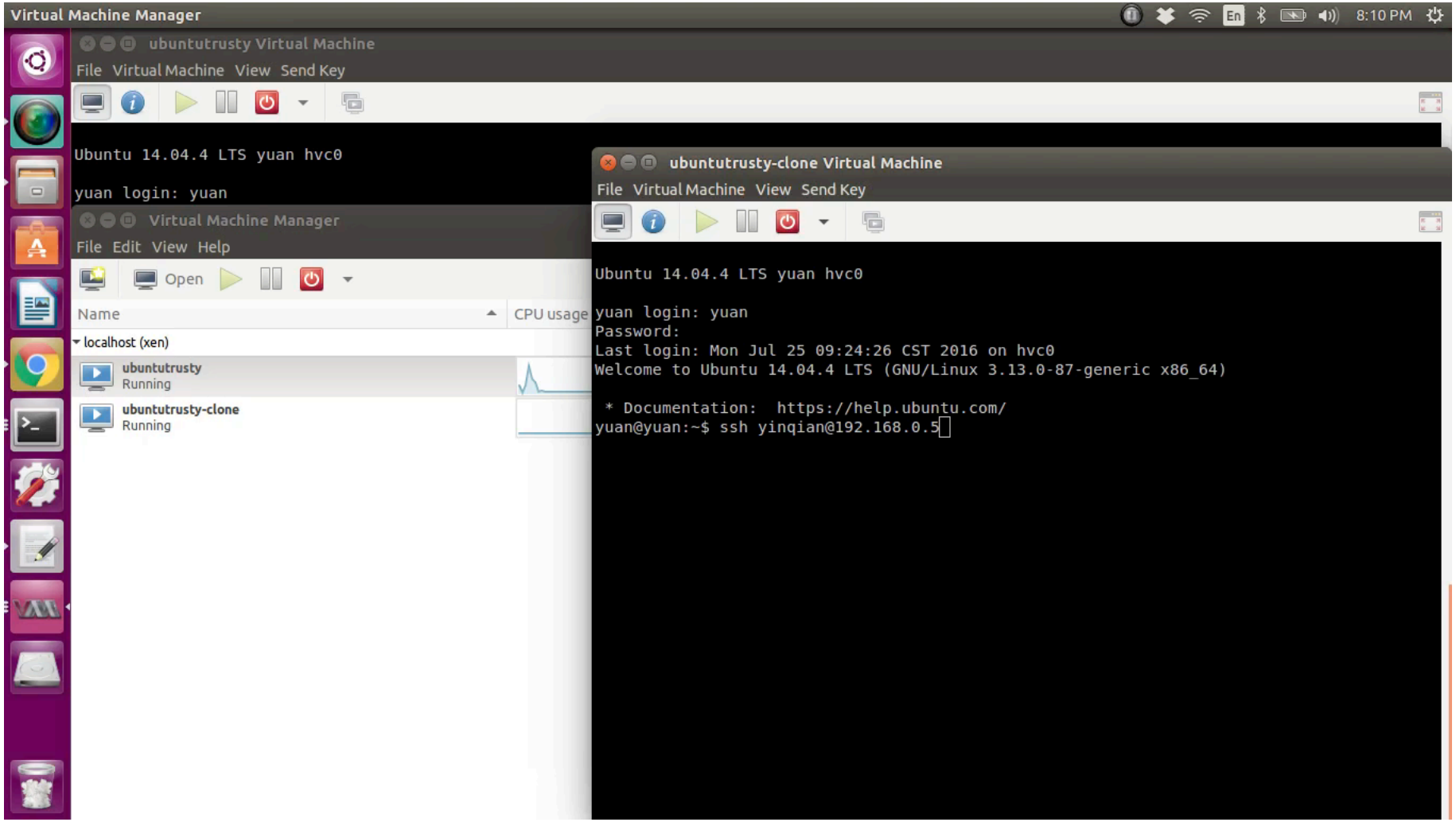
Primary goal: code modification



OpenSSH Authentication Bypass (2)



OpenSSH Authentication Bypass (3)

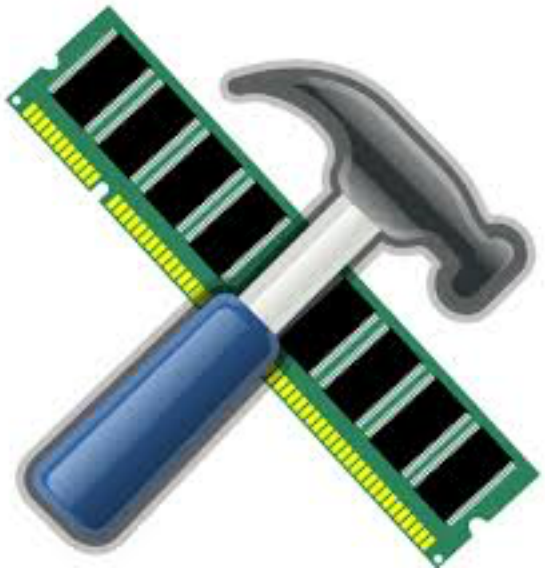


Existing Countermeasures

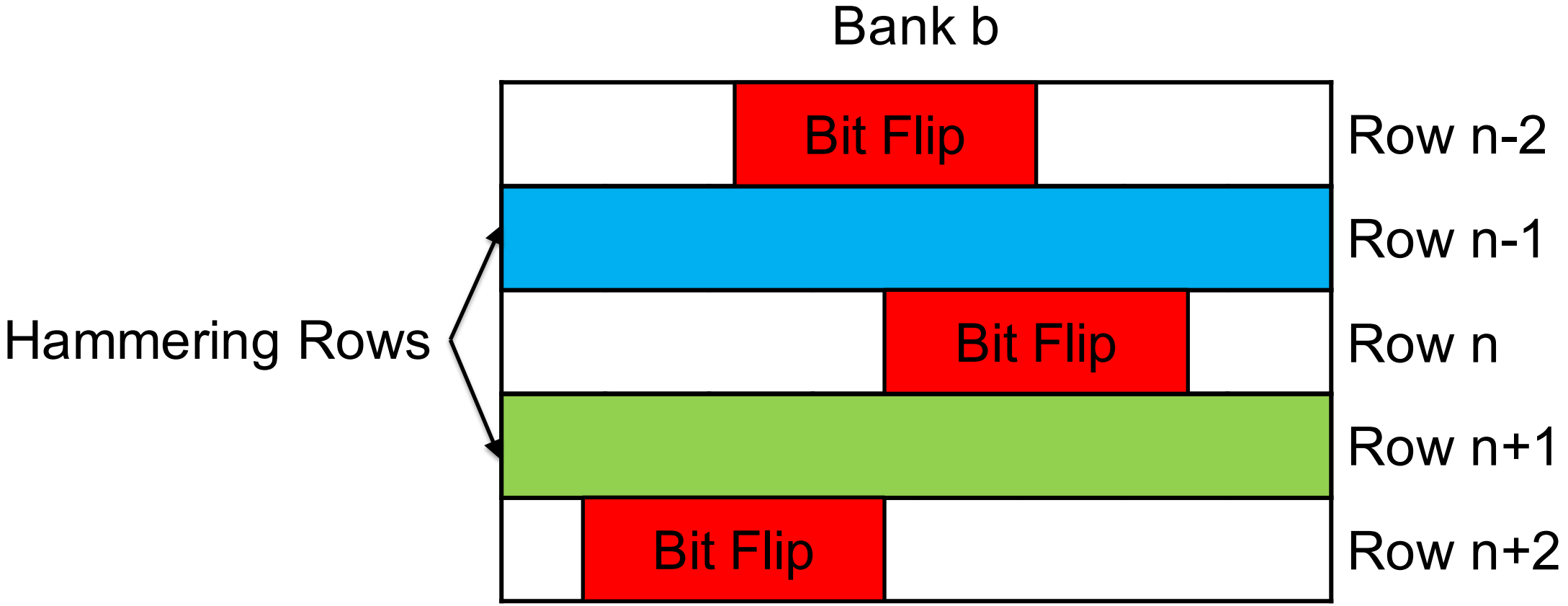
- ECC (Error Correcting Code)
- DDR4 (TRR - Target Row Refresh)
- HVM (Hardware-assisted Virtualization)

Conclusion

- We use timing channel to reverse-engineer the physical address mapping to the DRAM.
- We conduct efficient double-sided row hammer attacks.
- Xen PV can be exploited by row hammer and allow cross-VM access.

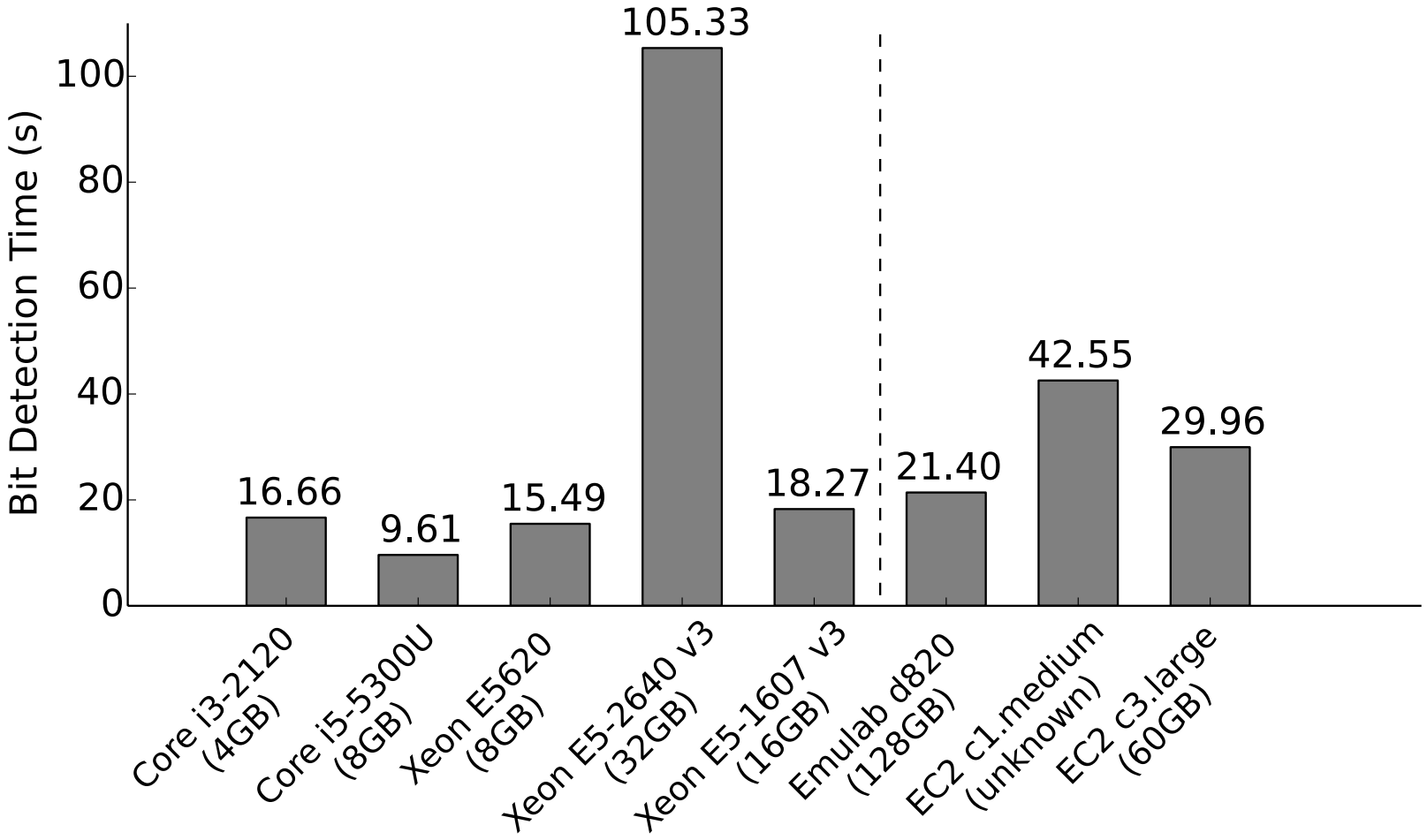


Row Hammer in Safe Mode

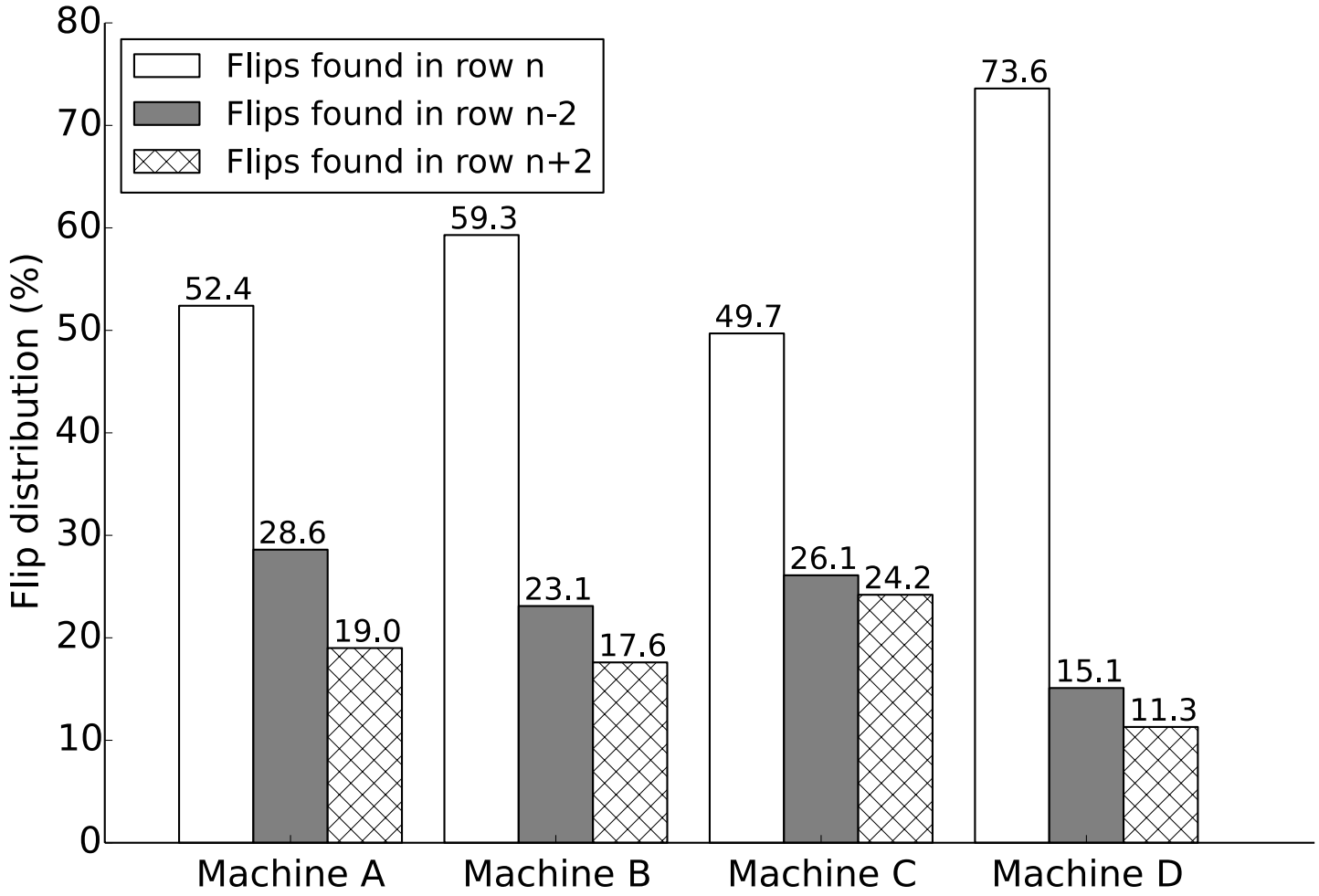


Safe: Row n-2, n, n+2 are mapped in the memory buffer of program

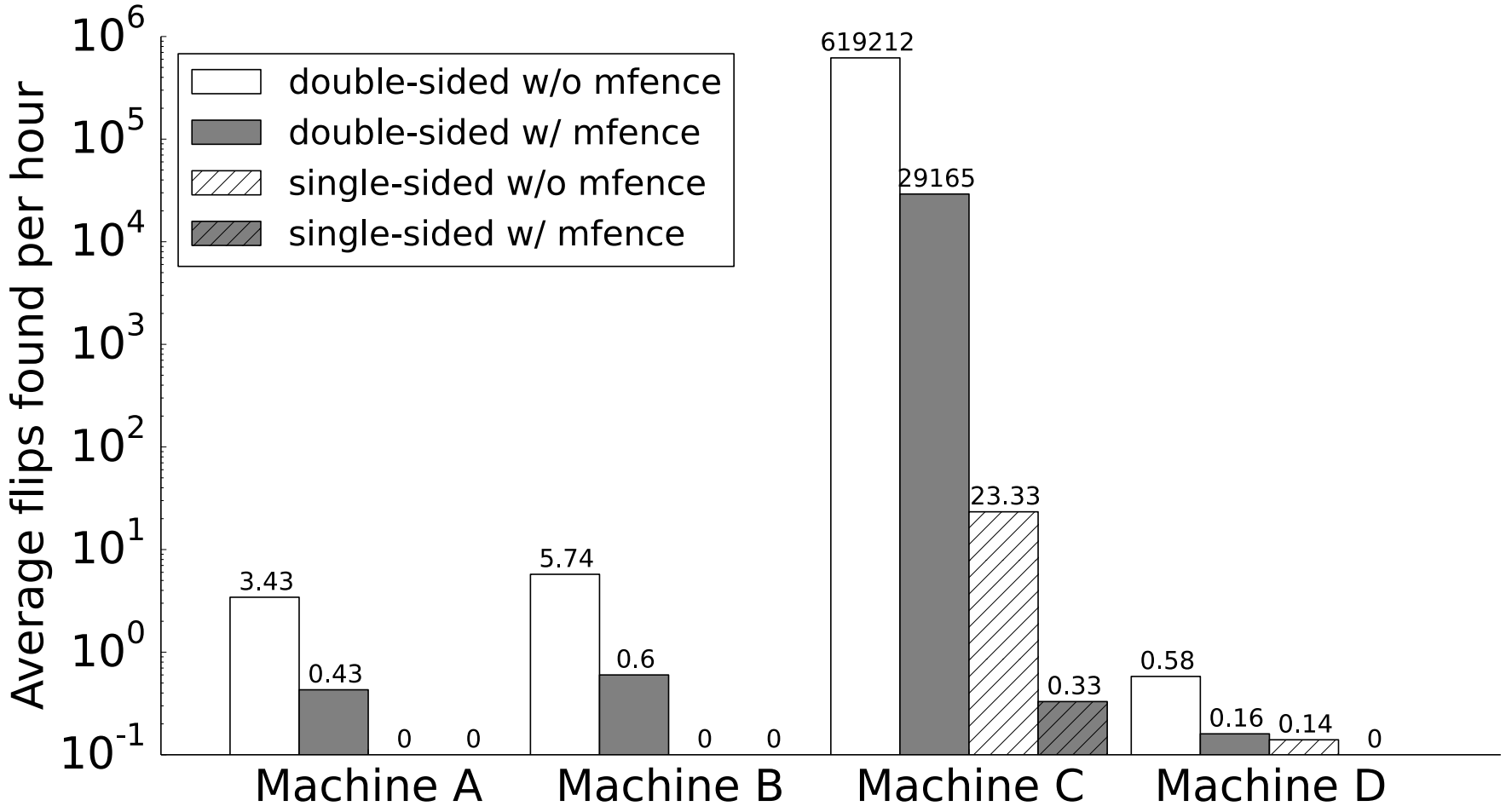
Bit Detection Efficiency



Flippable Bit Distribution



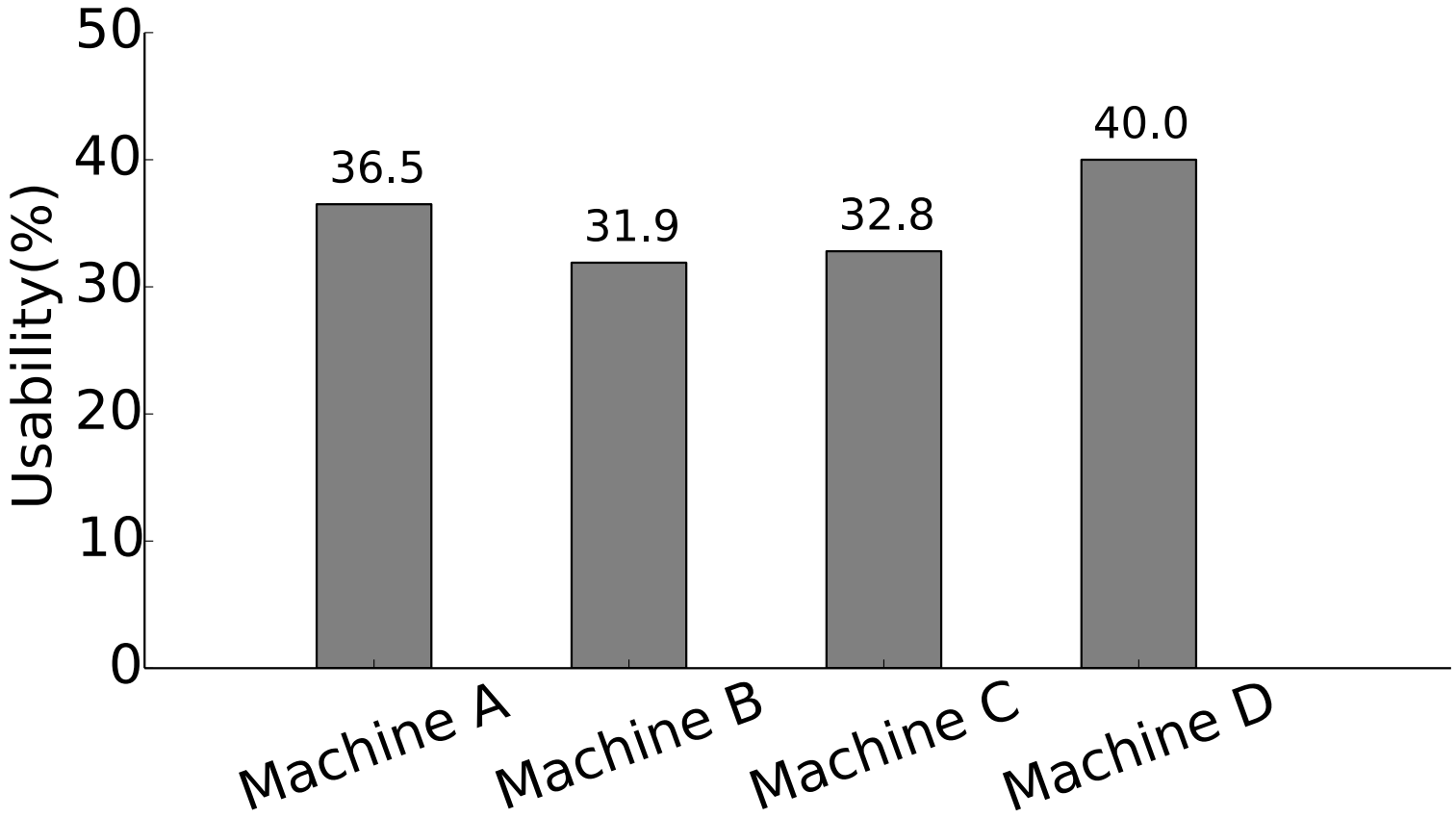
Effectiveness of Different Row Hammer



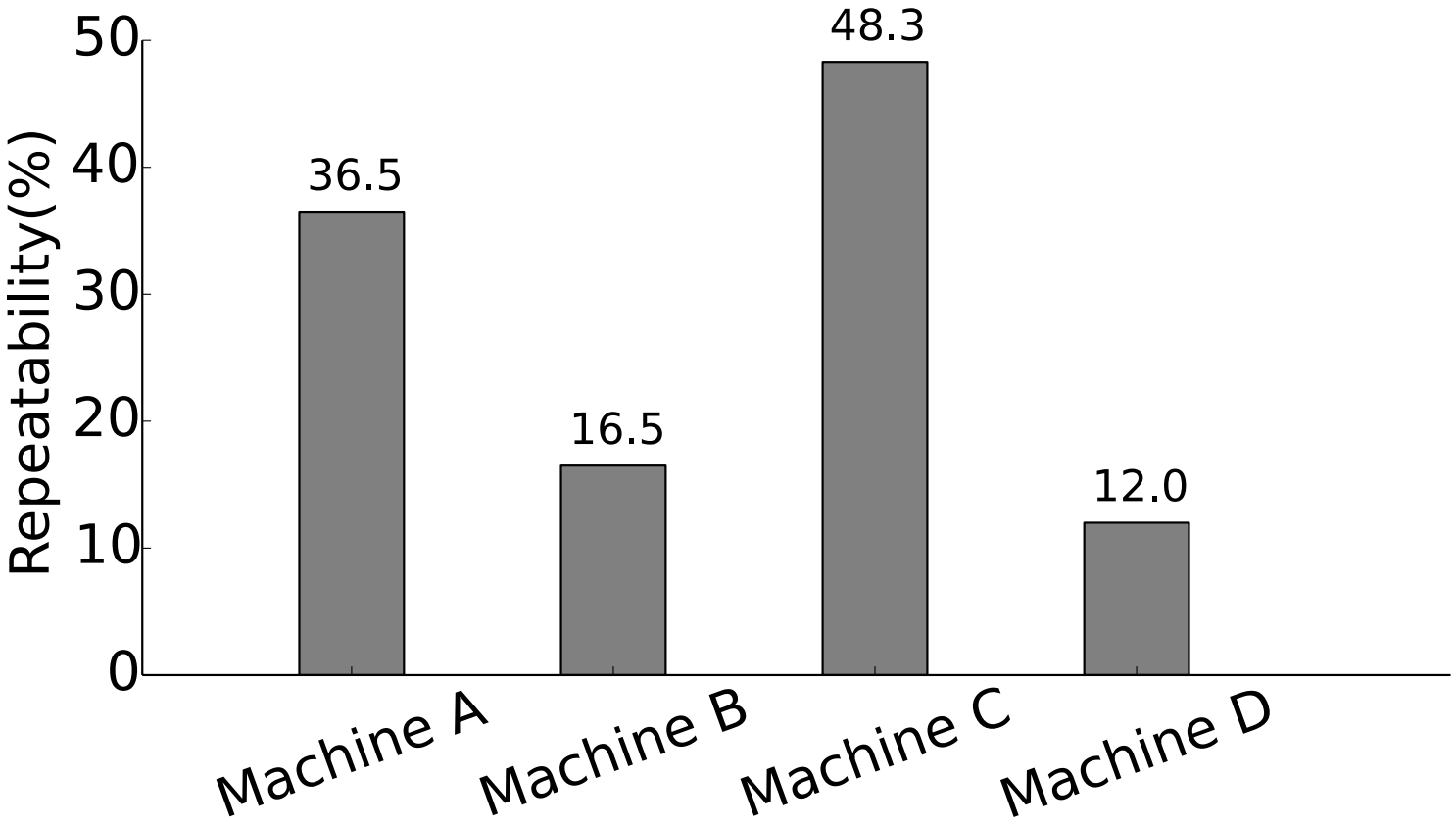
Vulnerability of Test Machines

Machine configuration	Execution time (hrs)	Vulnerable bits
(Machine A) Sandy Bridge i3-2120 (4GB)	18.37	63
(Machine B) Sandy Bridge i3-2120 (4GB)	15.85	91
(Machine C) Sandy Bridge i5-2500 (4GB)	9.08	5622445
(Machine D) Broadwell i5-5300U (8GB)	42.88	25

Usablility


Usability(%)
50
40
30
20
10
0
36.5
31.9
32.8
40.0
Machine A
Machine B
Machine C
Machine D

Repeatability



Flips within 64-bit Block

