SoftTRR: Protect Page Tables against Rowhammer Attacks using Software-only Target Row Refresh

Zhi Zhang*, Yueqiang Cheng*, Minghua Wang, Wei He, Wenhao Wang,

Nepal Surya, Yansong Gao, Kang Li, Zhe Wang, Chenggang Wu

(*: co-first authors)

Outline

- Background
- Motivation
- Overview
- Evaluation
- Conclusion

Background

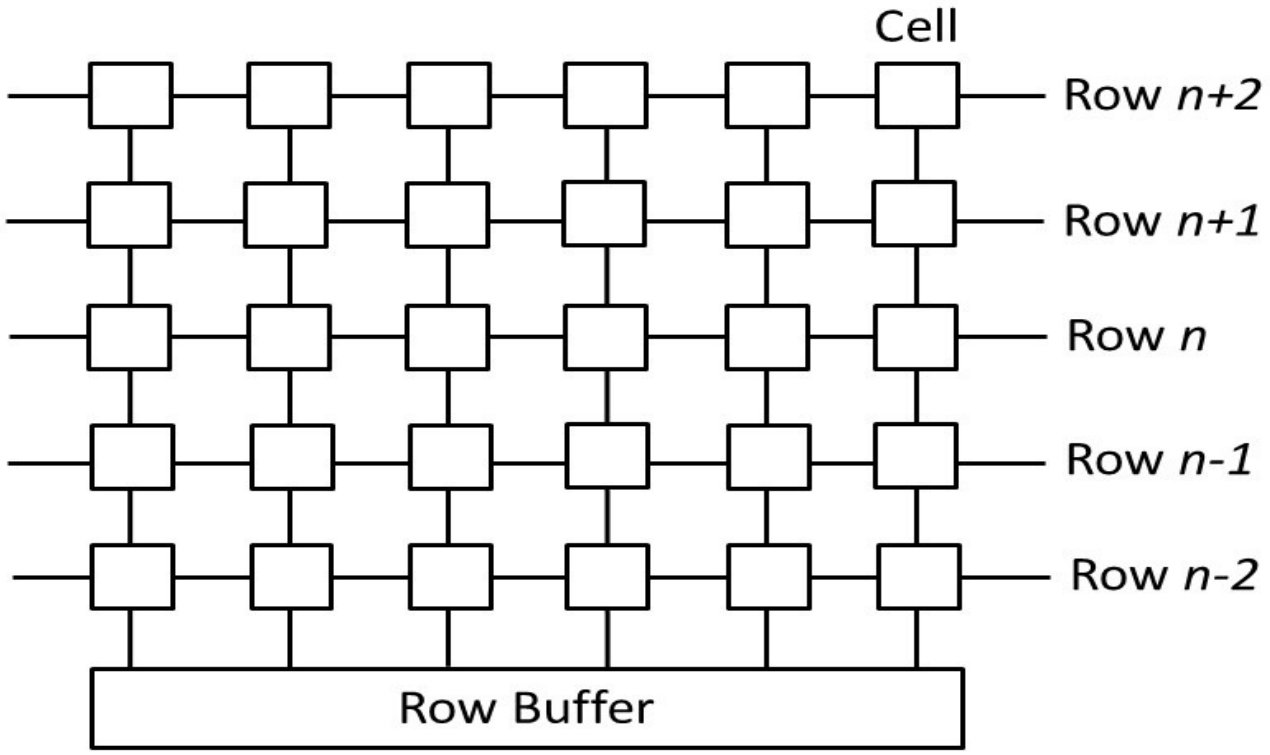
What is Rowhammer ?

What is Rowhammer ?

Rowhammer: Frequently accessing DRAM rows

DRAM bank

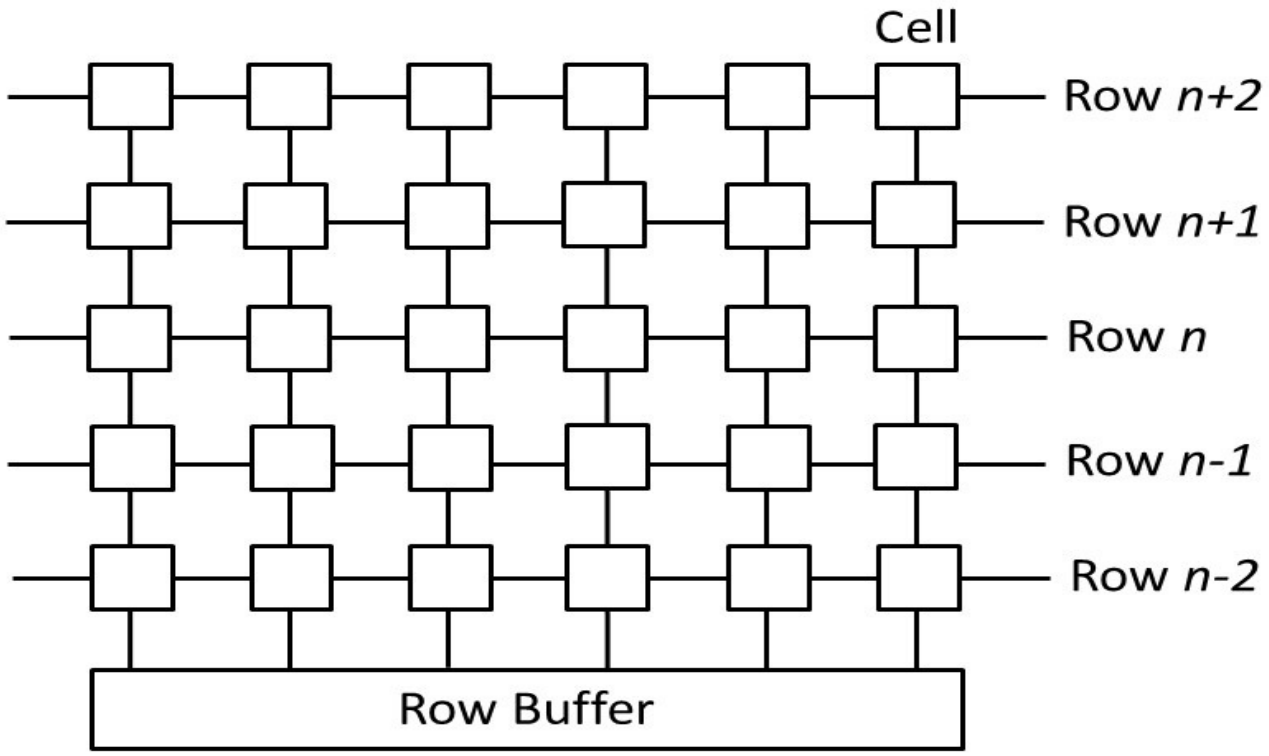
A bank has rows of cells



DRAM bank

A bank has rows of cells

A cell has a capacitor and an access-transistor



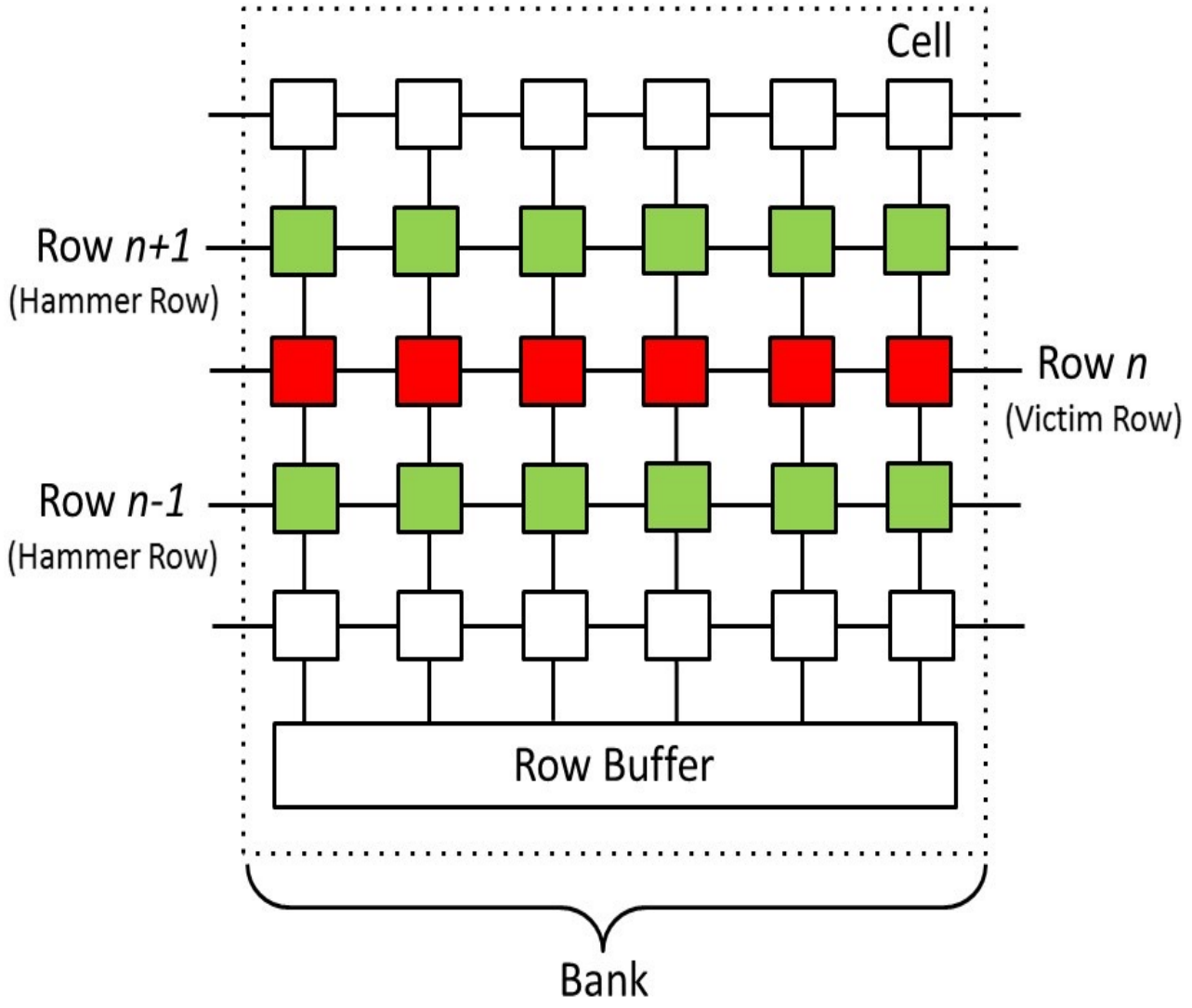
DRAM Refresh

- ★ capacitors of cells can lose charge over time
- ★ cells must be periodically refreshed
- ★ the refresh rate is typically 64 ms in DDR3 and DDR4

Rowhammer

Kim et al. (ISCA'14)

frequently opening rows *n+1* & *n-1* cause charge leakage (bit flips) in row *n*



Motivation

Rowhammer Attacks

- Rowhammer-induced page tables corruption is the most detrimental to system security and hard to mitigate (CTA *ASPLOS'19*)

- Mainstream rowhammer attacks target level-1 page table corruption

Limitations of the State-of-the-Art Works

Limitations of the State-of-the-Art Works

- Practicality
  - incurring modifications to kernel memory subsystem.

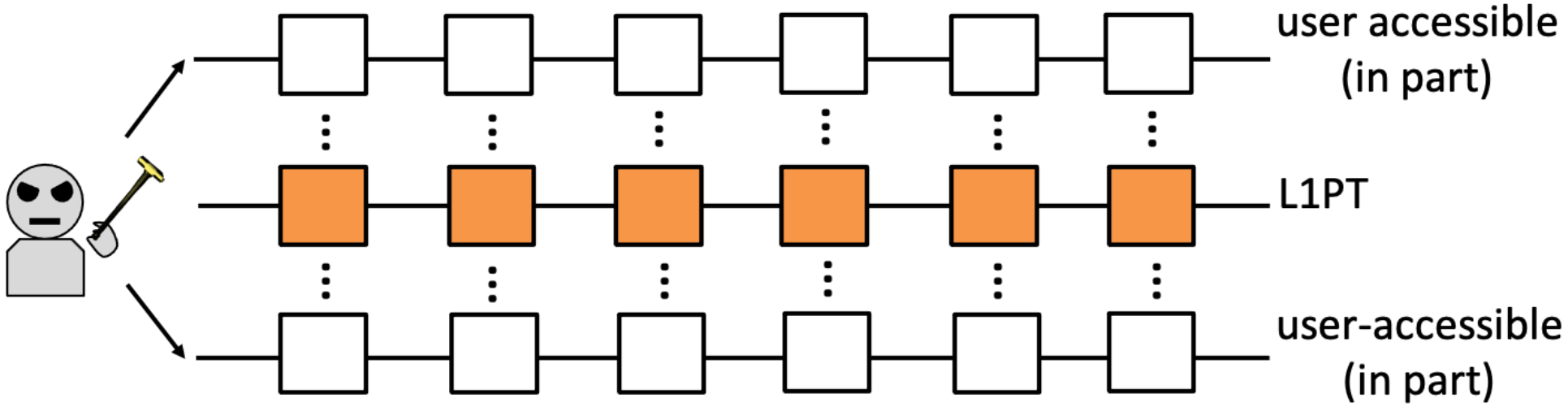
Limitations of the State-of-the-Art Works

- Practicality
  - incurring modifications to kernel memory subsystem

- Effectiveness
  - being ineffective against all existing rowhammer attacks targeting page tables (e.g., PThammer MICRO'20)

❖ Explicit Rowhammer Attacks

➢ Require access to part of rows adjacent to L1PT rows for explicit hammering

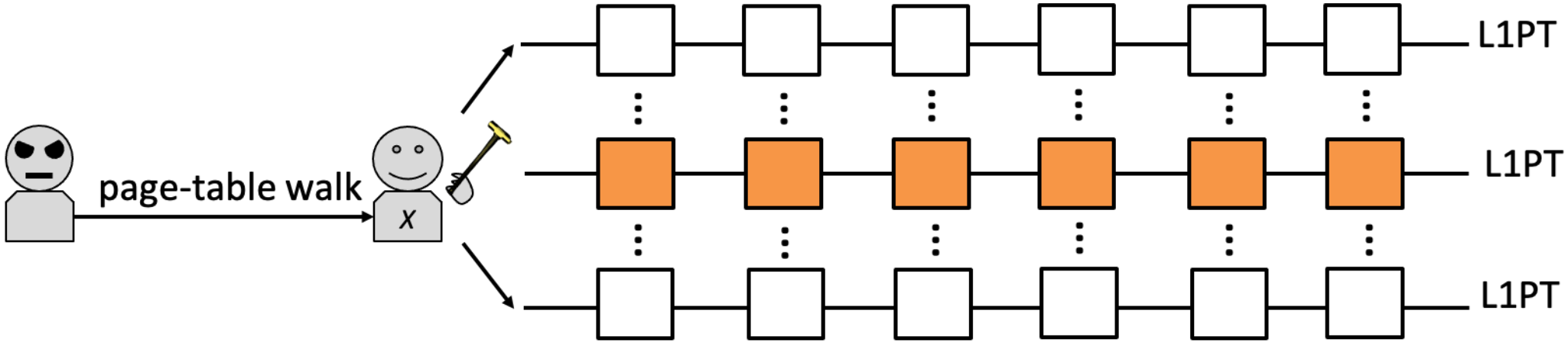


❖ Explicit Rowhammer Attacks

- ➢ Require access to part of rows adjacent to L1PT rows for explicit hammering

❖ Implicit Rowhammer Attacks

- ➢ PThammer, the only instance



Overview

Goal: protect page tables from rowhammer attacks

Goal: protect page tables from rowhammer attacks

Design Principles:

- effective in protecting page tables from explicit and implicit attacks

Goal: protect page tables from rowhammer attacks

Design Principles:

- effective in protecting page tables from explicit and implicit attacks
- compatible with OS kernels

Goal: protect page tables from rowhammer attacks

Design Principles:

- effective in protecting page tables from explicit and implicit attacks
- compatible with OS kernels
- small performance overhead to a commodity system

Key Insights

DRAM-chip-based TRR (ChipTRR), widely deployed in DDR4 modules.

- high-level idea: ChipTRR counts rows' activations and refreshes adjacent rows to suppress bit flips if the activation counts reach a pre-defined limit.

Key Insights

DRAM-chip-based TRR (ChipTRR), widely deployed in DDR4 modules.

- high-level idea: ChipTRR counts rows' activations and refreshes adjacent rows to suppress bit flips if the activation counts reach a pre-defined limit.
- security limitation: ChipTRR only tracks a limited number of rows, which renders its rowhammer-free guarantee broken by TRRespass IEEE S&P'20.

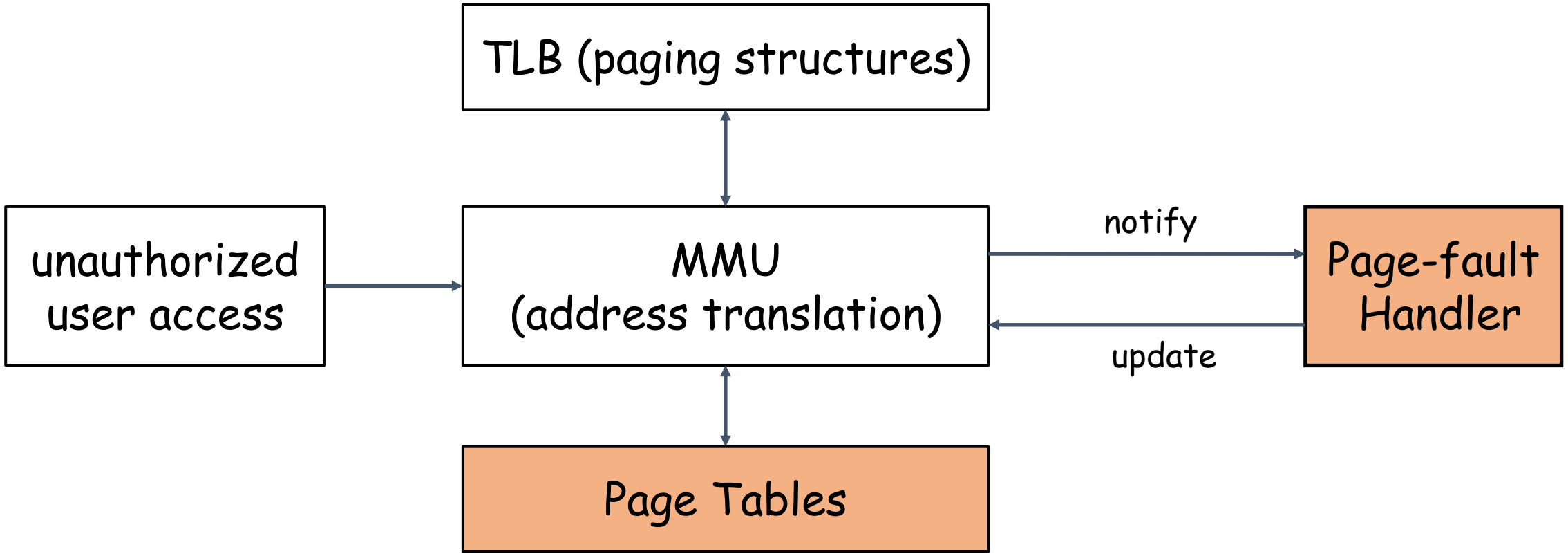
Key Insights

DRAM-chip-based TRR (ChipTRR), widely deployed in DDR4 modules.

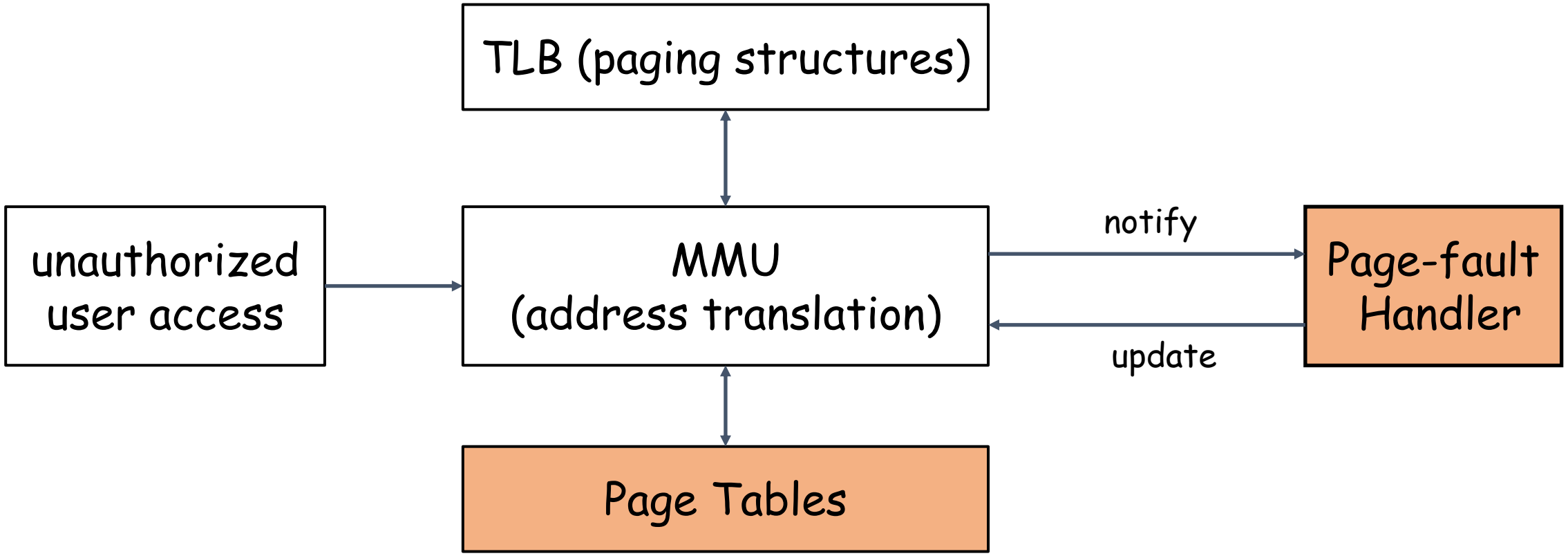
- high-level idea: ChipTRR counts rows' activations and refreshes adjacent rows to suppress bit flips if the activation counts reach a pre-defined limit.
- security limitation: ChipTRR only tracks a limited number of rows, which renders its rowhammer-free guarantee broken by TRRespass IEEE S&P'20.

Software-only TRR (SoftTRR): protects page-table integrity by adopting the above idea while addresses the security limitation by leveraging MMU and OS kernel features.

Memory-access Mediation

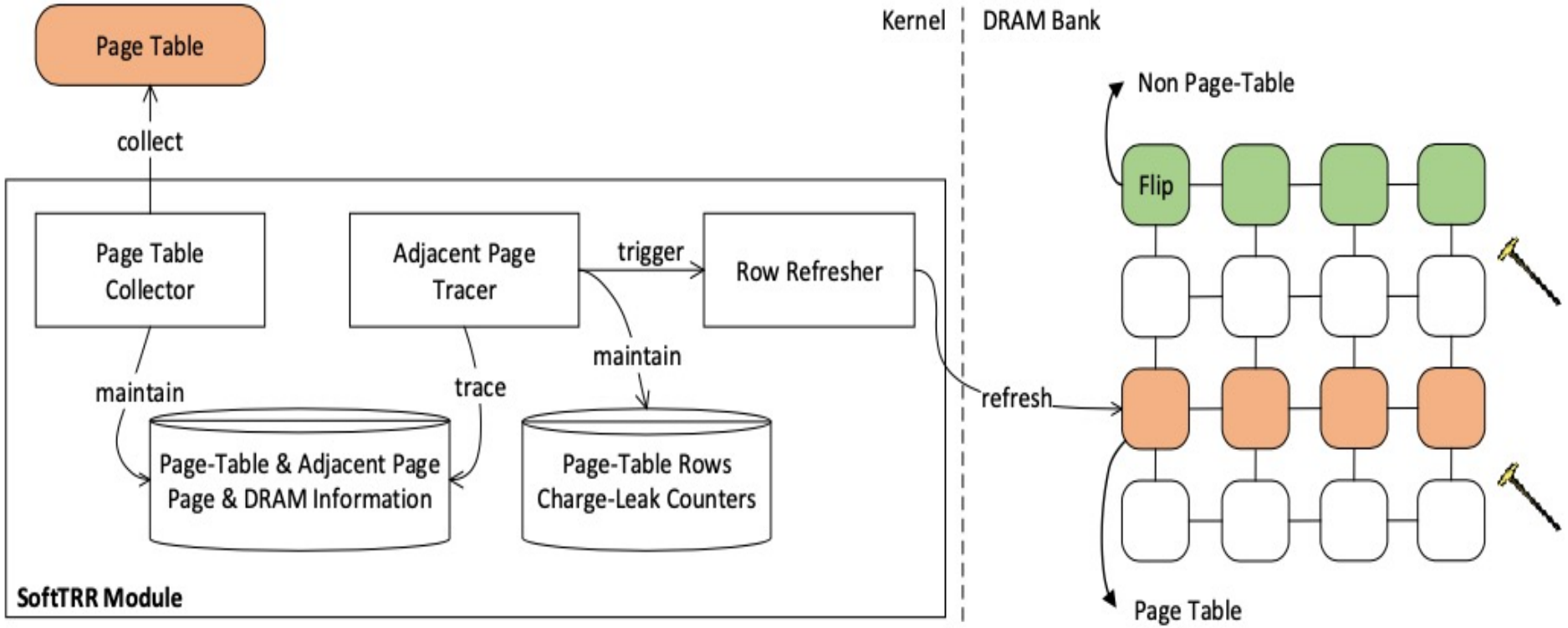


Memory-access Mediation

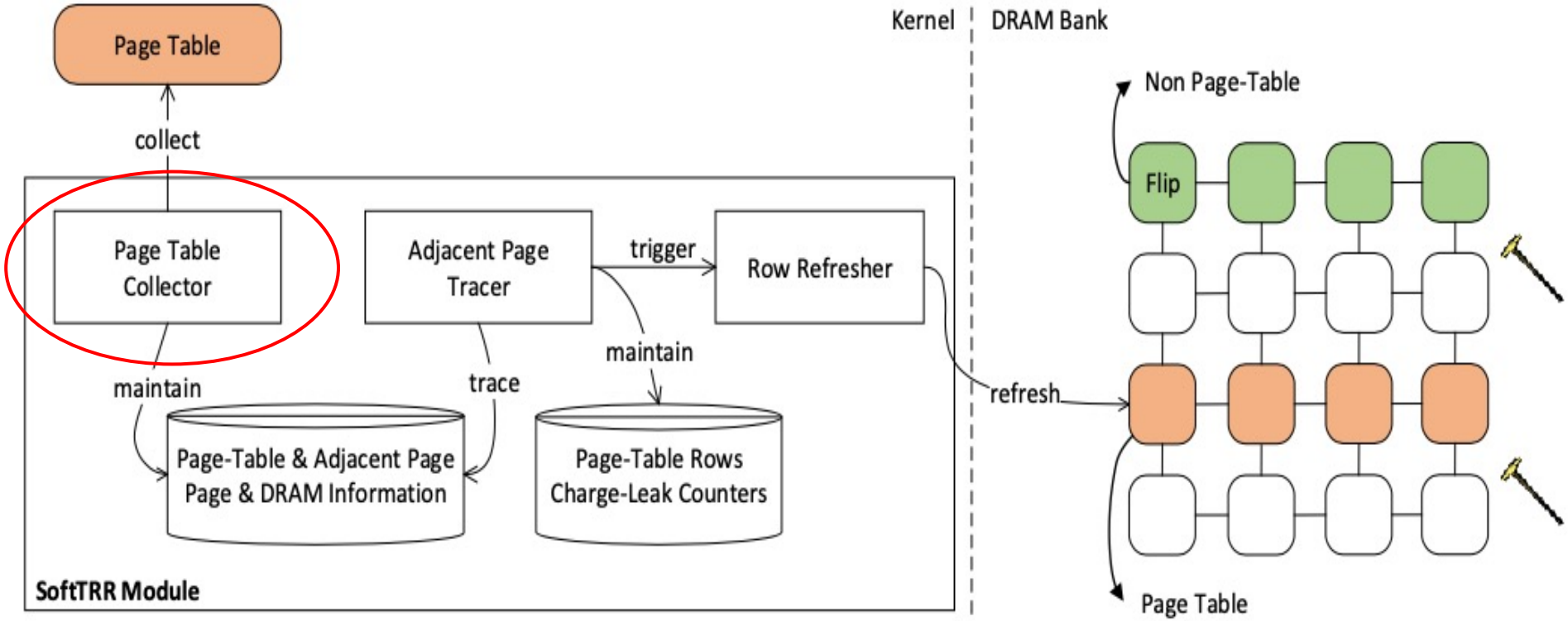


SoftTRR leverages page tables and page-fault handler to frequently trace memory accesses to any rows adjacent to rows hosting page-tables.

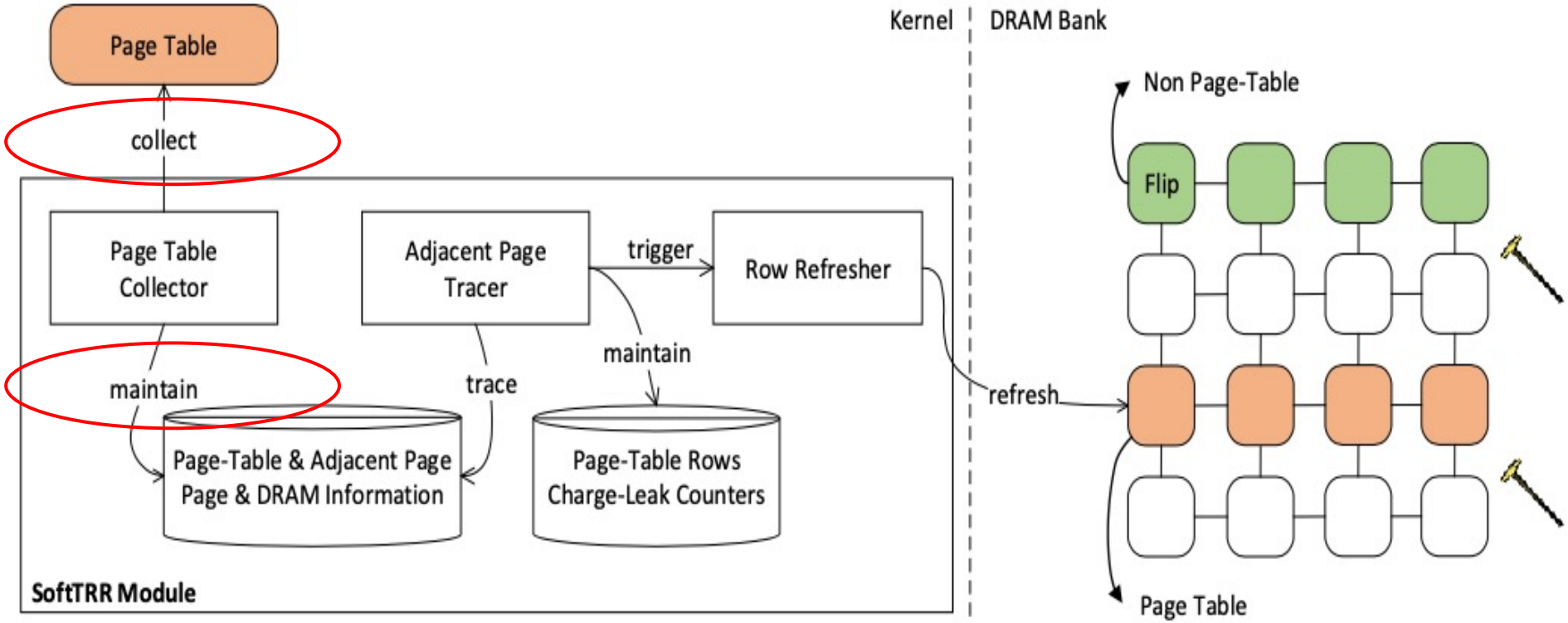
Overview


Page Table
collect
Page Table Collector
Adjacent Page Tracer
trigger
Row Refresher
maintain
trace
maintain
Page-Table & Adjacent Page Page & DRAM Information
Page-Table Rows Charge-Leak Counters
SoftTRR Module
Kernel
DRAM Bank
refresh
Non Page-Table
Flip
Page Table

Overview


Page Table
collect
Page Table Collector
Adjacent Page Tracer
trigger
Row Refresher
maintain
trace
maintain
Page-Table & Adjacent Page Page & DRAM Information
Page-Table Rows Charge-Leak Counters
SoftTRR Module
Kernel
DRAM Bank
Non Page-Table
Flip
refresh
Page Table

Overview


Page Table
collect
Page Table Collector
maintain
Adjacent Page Tracer
trigger
Row Refresher
maintain
trace
Page-Table & Adjacent Page Page & DRAM Information
Page-Table Rows Charge-Leak Counters
SoftTRR Module
Kernel
DRAM Bank
Non Page-Table
Flip
refresh
Page Table

Page Table Collector

- In our implementation, SoftTRR focuses on protecting level-1 page tables (L1PTs) that are targeted by both explicit and implicit rowhammer attacks.

Page Table Collector

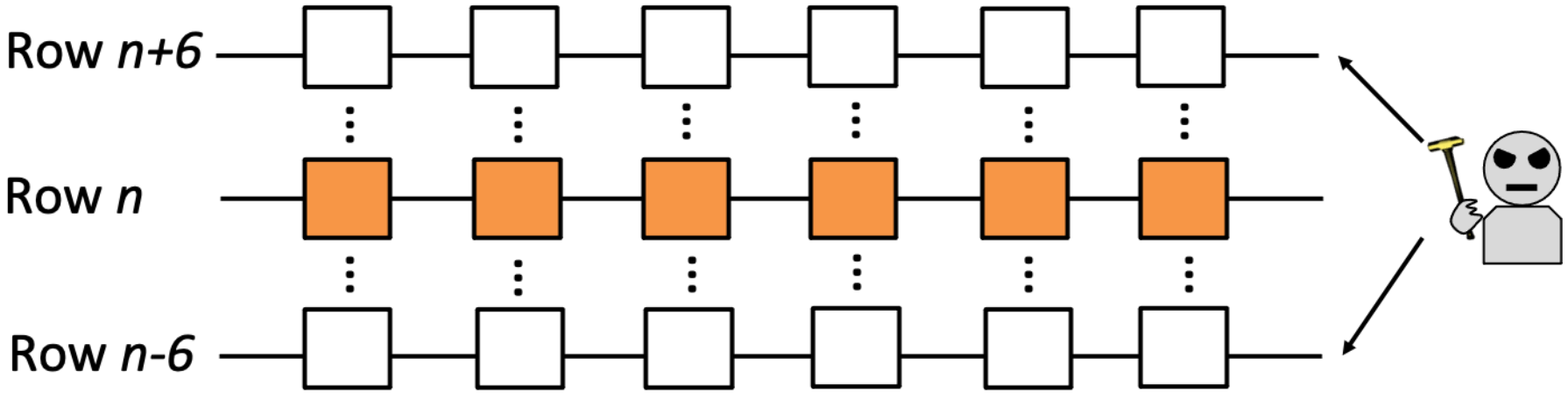
- In our implementation, SoftTRR focuses on protecting level-1 page tables (L1PTs) that are targeted by both explicit and implicit rowhammer attacks.
- Page table collector asks task_struct and hooks L1PT alloc and free functions for page collection
  - ✓ L1PT pages
  - ✓ DRAM-adjacent pages
  - ✓ their DRAM row locations

Page Table Collector

- In our implementation, SoftTRR focuses on protecting level-1 page tables (L1PTs) that are targeted by both explicit and implicit rowhammer attacks.
- Page table collector asks task_struct and hooks L1PT alloc and free functions for page collection
  - ✓ L1PT pages
  - ✓ DRAM-adjacent pages
  - ✓ their DRAM row locations
- DRAM-adjacent page

Page Table Collector

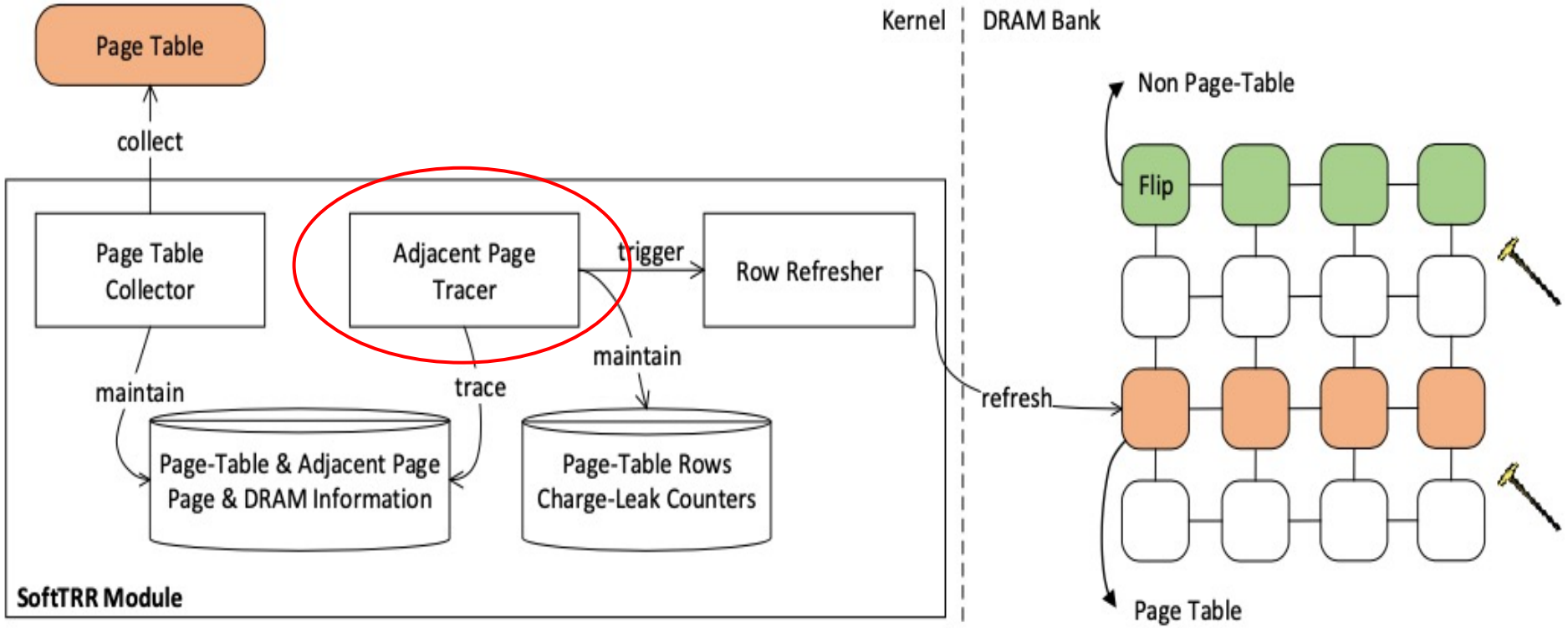
- In our implementation, SoftTRR focuses on protecting level-1 page tables (L1PTs) that are targeted by both explicit and implicit rowhammer attacks.
- Page table collector asks task_struct and hooks L1PT alloc and free functions for page collection
  - ✓ L1PT pages
  - ✓ DRAM-adjacent pages
  - ✓ their DRAM row locations
- DRAM-adjacent page: up to 6-row from a row hosting L1PTs (based on Kim et al. ISCA'20)



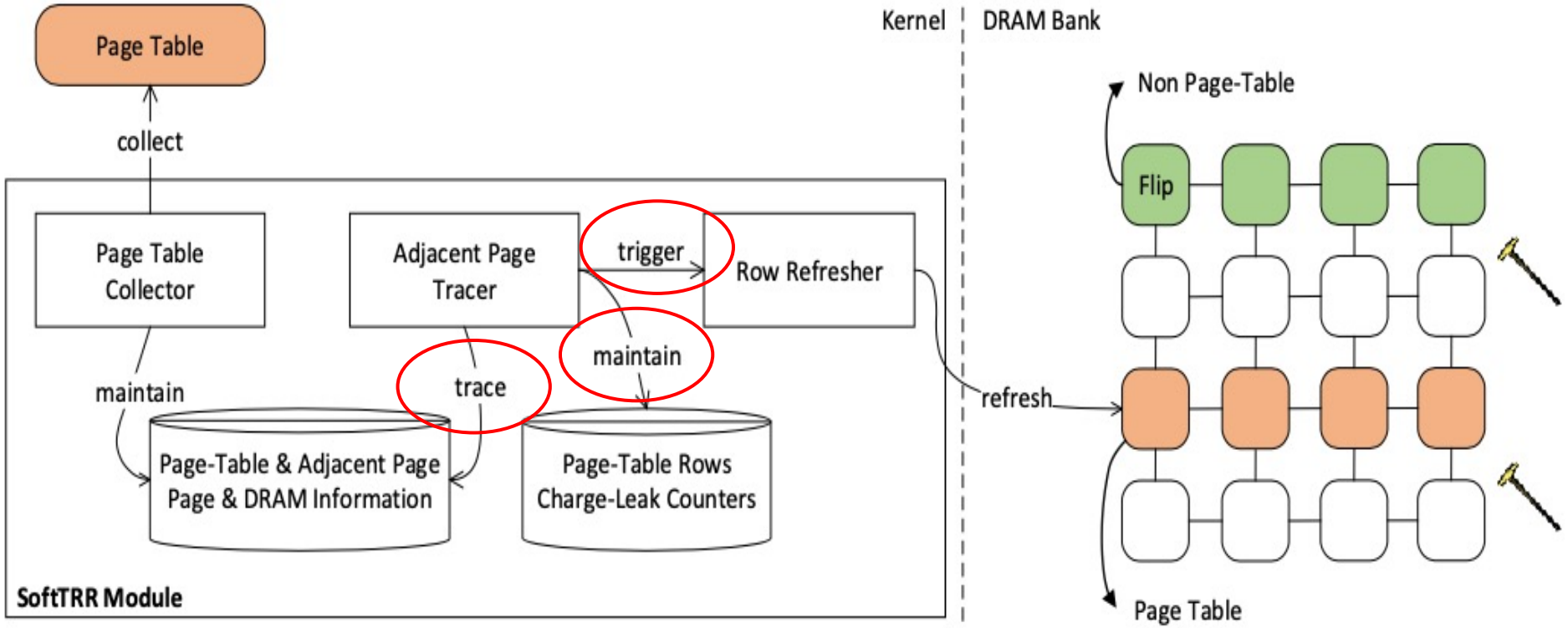
Page Table Collector

- In our implementation, SoftTRR focuses on protecting level-1 page tables (L1PTs) that are targeted by both explicit and implicit rowhammer attacks.
- Page table collector asks task_struct and hooks L1PT alloc and free functions for page collection
  - ✓ L1PT pages
  - ✓ DRAM-adjacent pages
  - ✓ their DRAM row locations
- DRAM-adjacent page: up to 6-row from a row hosting L1PTs (based on Kim et al. ISCA'20)
- An attacker can explicitly or implicitly hammer an adjacent page
- Page table collector maintains three red-black trees for the collected information
  - ✓ pt_rbtree
  - ✓ adj_rbtree
  - ✓ pt_row_rbtree

Overview

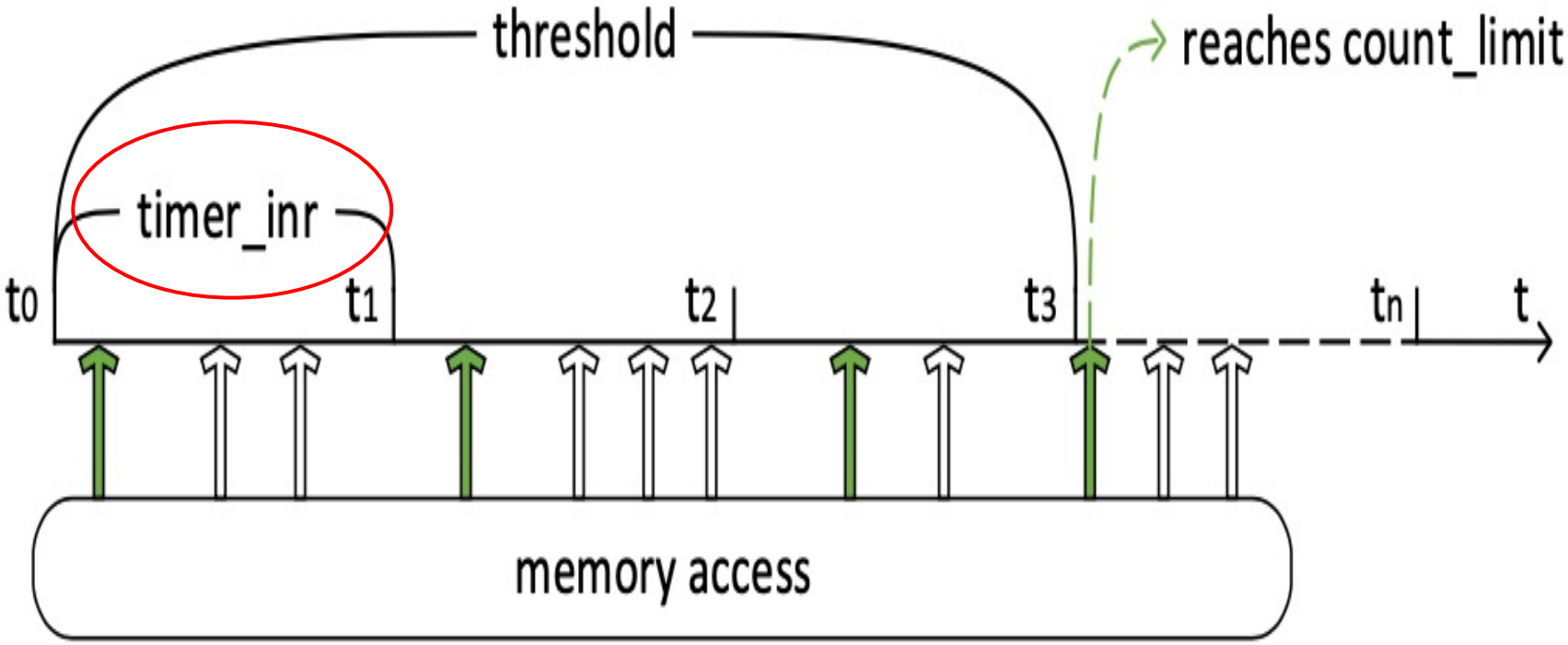

Page Table
collect
Kernel
DRAM Bank
Page Table Collector
Adjacent Page Tracer
trigger
Row Refresher
maintain
maintain
trace
refresh
Page-Table & Adjacent Page Page & DRAM Information
Page-Table Rows Charge-Leak Counters
SoftTRR Module
Non Page-Table
Flip
Page Table

Overview

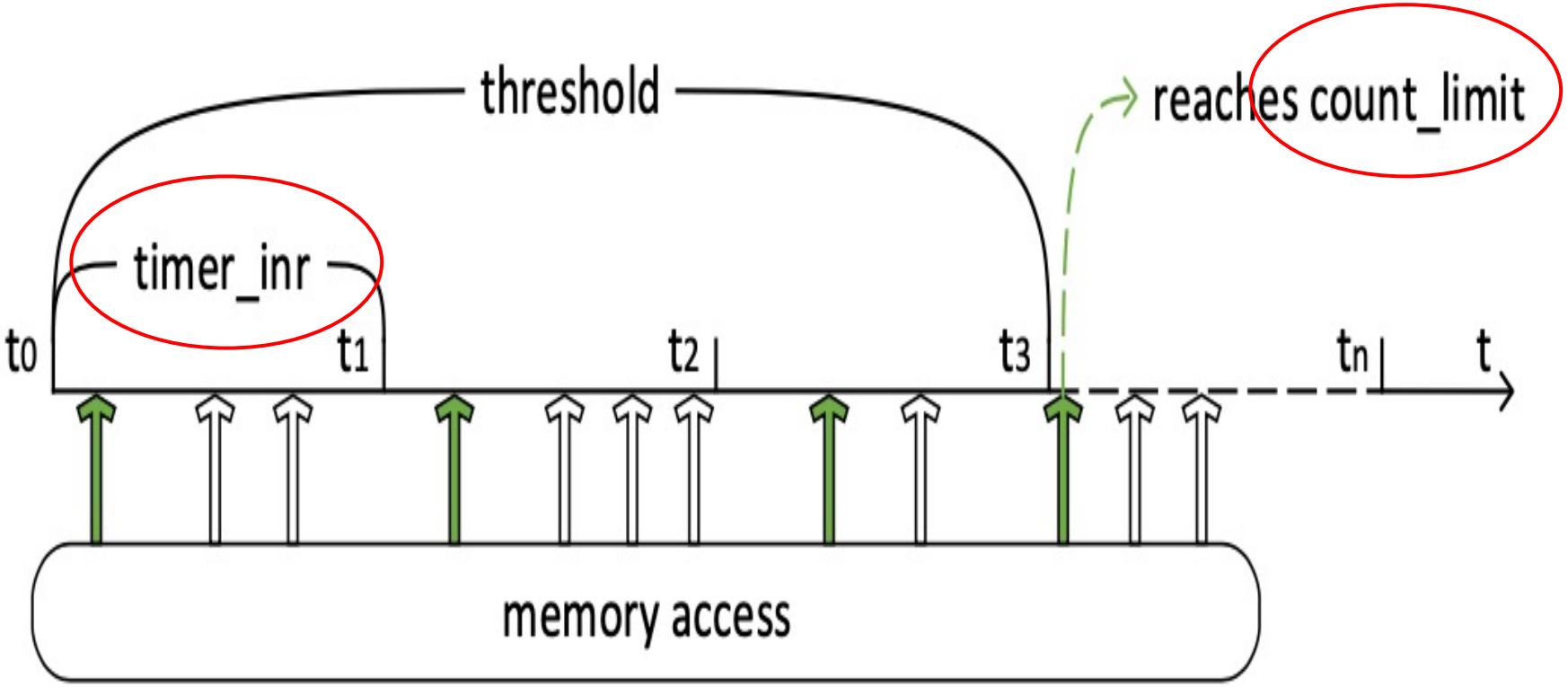


- Trace memory accesses to adjacent pages
- Maintain a counter for each page-table row
- Trigger row-refresher when the counter reaches a pre-defined limit, similar to ChipTRR

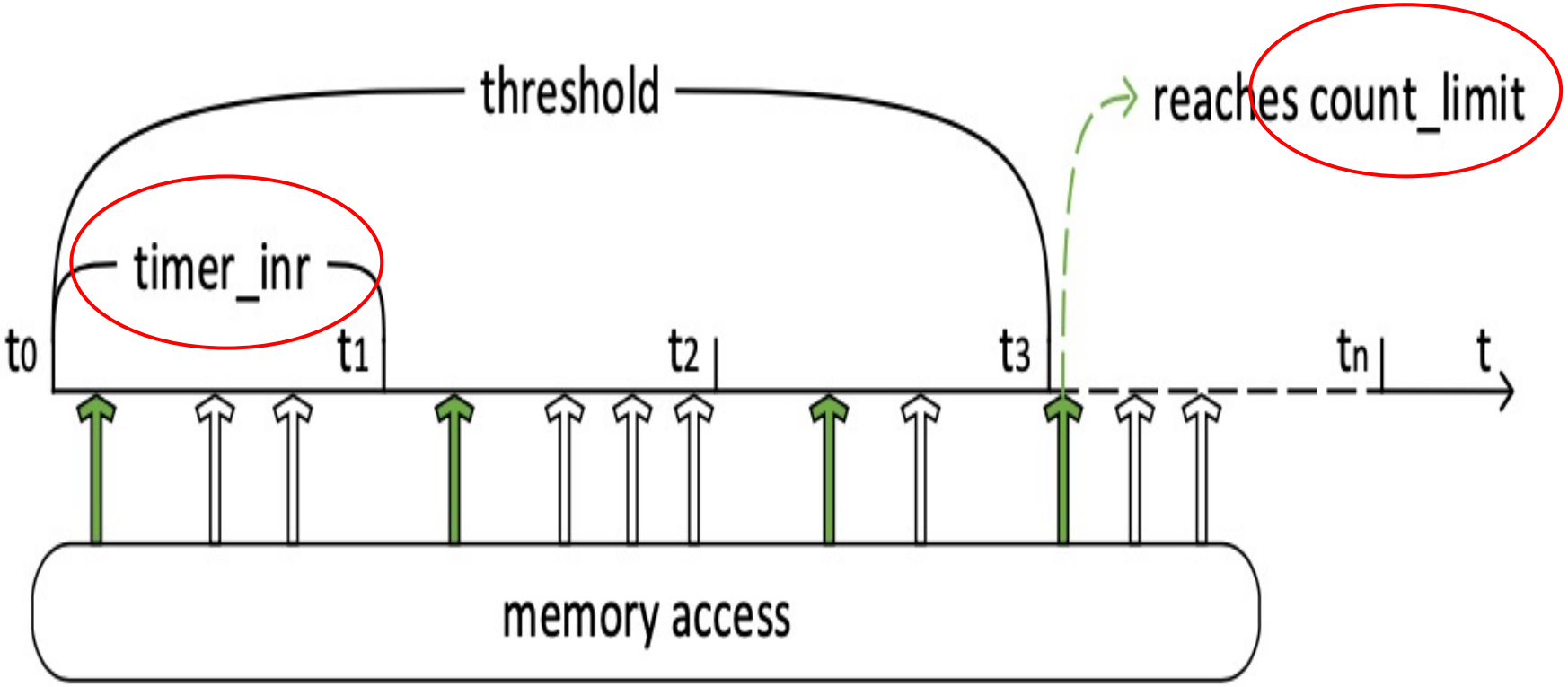
Adjacent Page Tracer



Adjacent Page Tracer

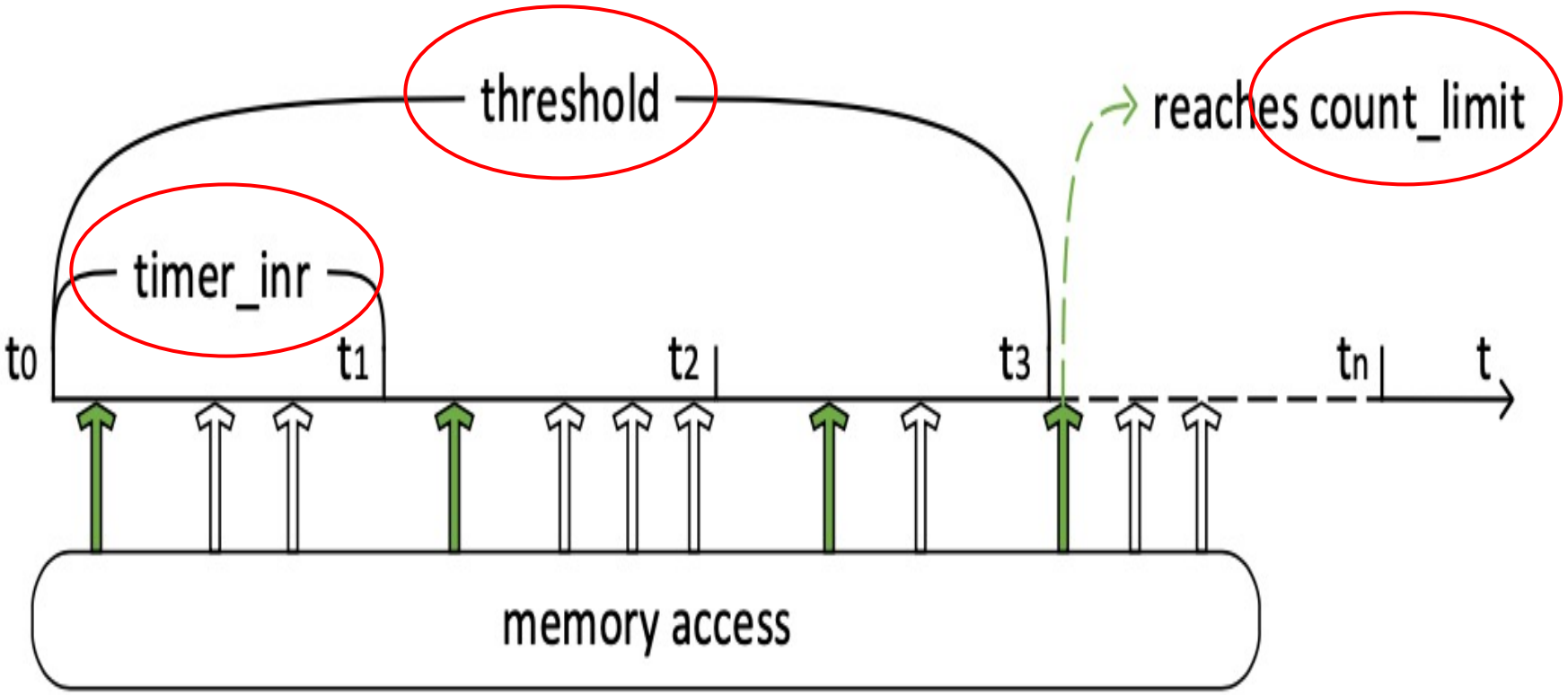


Adjacent Page Tracer



- Set-up tracing periodically
- Determine *timer_inr* and *count_limit*

Adjacent Page Tracer



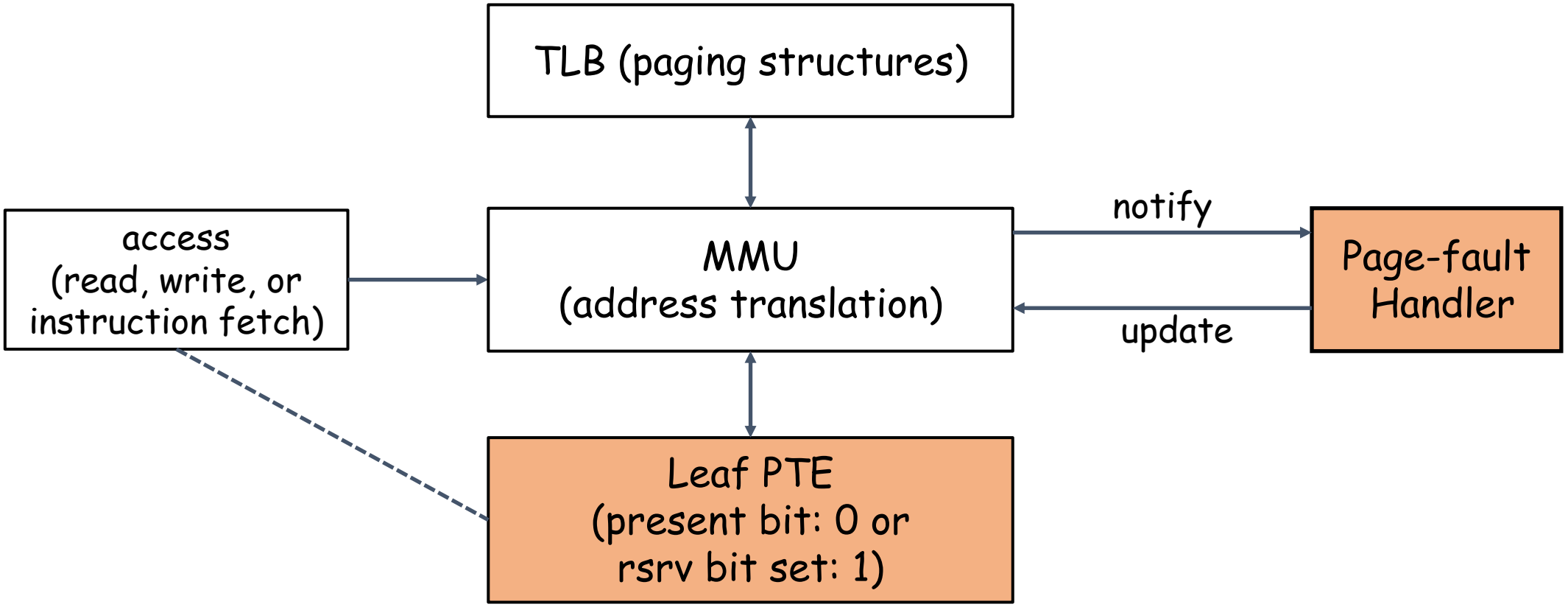
- Set-up tracing periodically
- Determine *timer_inr* and *count_limit*

Adjacent Page Tracer

- Set-up tracing periodically
  - ✓ Configuring *present* bit or *rsrv* bit in leaf PTEs (page table entries) can capture a memory access of *read*, *write* or *instruction fetch*.

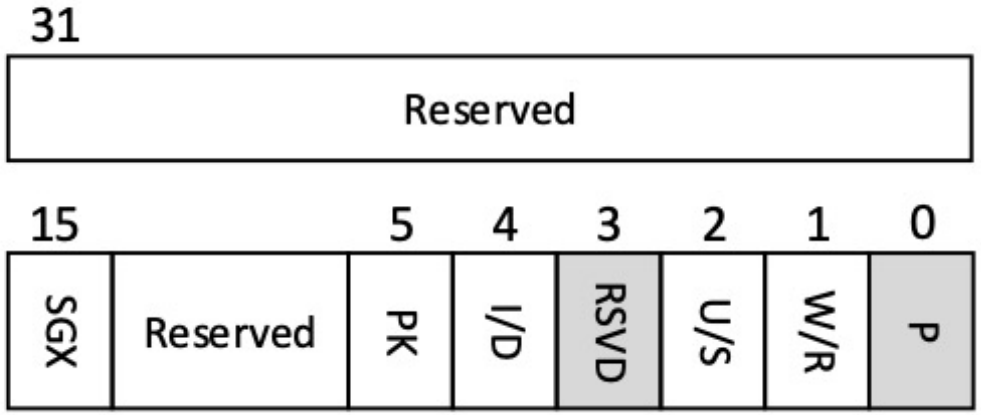
Adjacent Page Tracer

- Set-up tracing periodically
  - ✓ Configuring *present* bit or *rsrv* bit in leaf PTEs (page table entries) can capture a memory access of *read*, *write* or *instruction fetch*.



MMU-supported page-fault error code

- present set to 0 corresponds to P bit set to 0

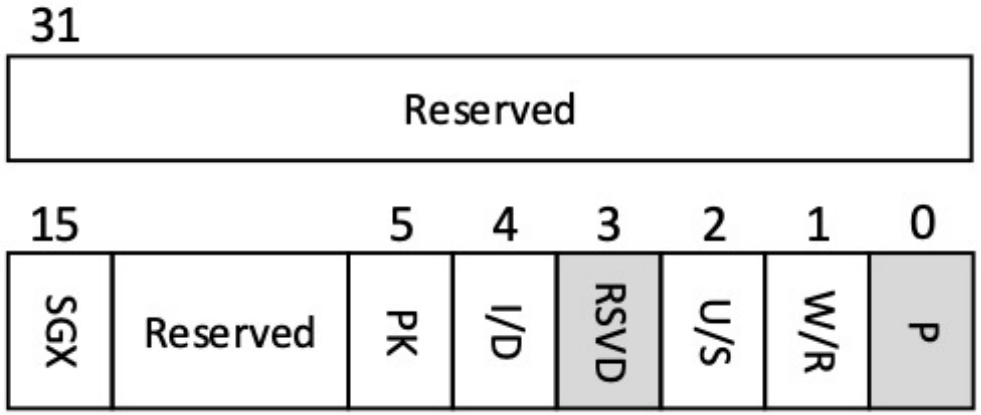


P — 0 means that the fault was caused by a non-present page.
1 means that the fault was caused by a page-level protection violation.

RSVD — 0 means that the fault was caused by reserved bit violation.
1 means that the fault was caused by a reserved bit set to 1 in a page-table entry

MMU-supported page-fault error code

- present set to 0 corresponds to P bit set to 0
- rsrv bit set to 1 corresponds to RSVD bit set to 1



P

0 means that the fault was caused by a non-present page.
1 means that the fault was caused by a page-level protection violation.

RSVD

0 means that the fault was caused by reserved bit violation.
1 means that the fault was caused by a reserved bit set to 1 in a page-table entry

Adjacent Page Tracer

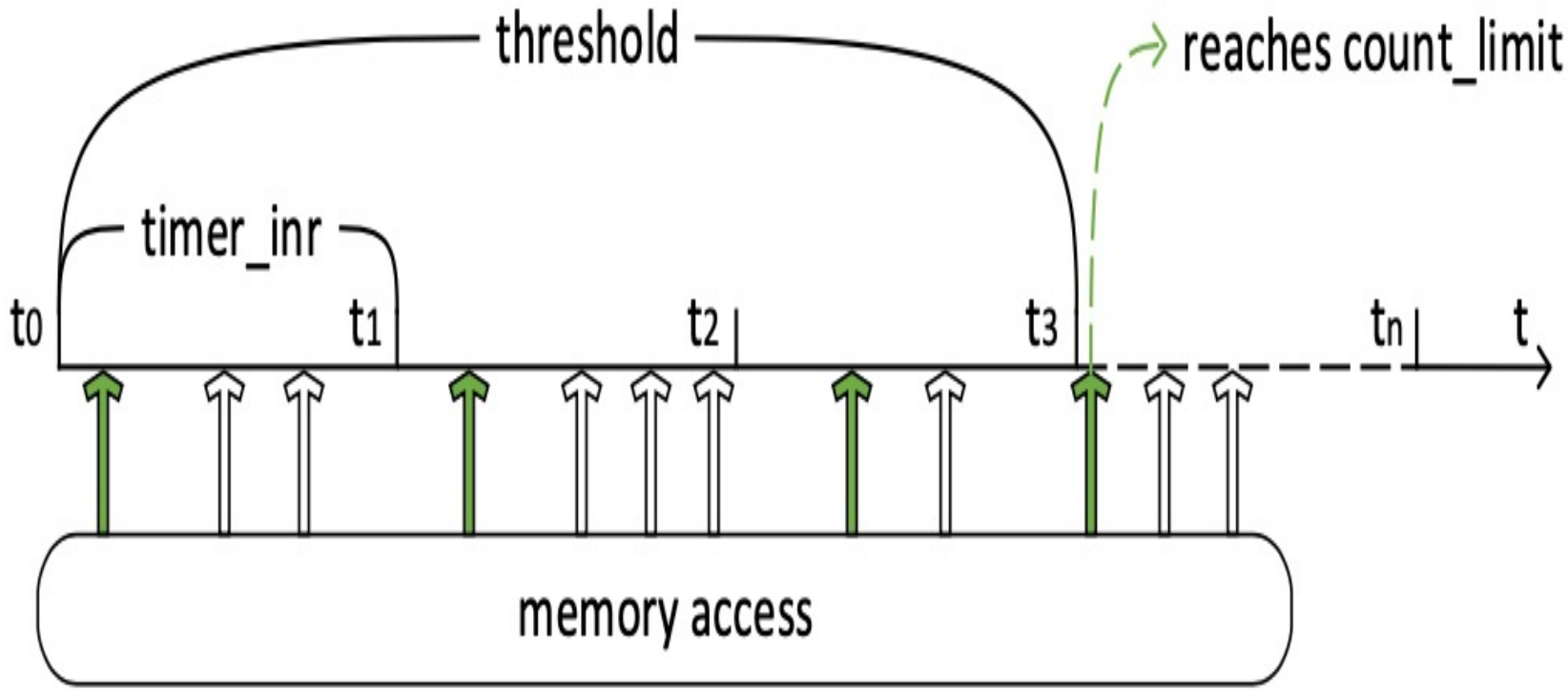
- Set-up tracing periodically
  - ✓ Configuring *present* bit or *rsrv* bit in leaf PTEs (page table entries) can capture memory access of *read*, *write* or *instruction fetch*.
  - ✓ Choose *rsrv* bit as configuring *present* bit causes kernel abort.

Adjacent Page Tracer

- Set-up tracing periodically
- Determine *timer_intr* and count_limit
  - ✓ *threshold = ?*

Adjacent Page Tracer

- Set-up tracing periodically
- Determine *timer_intr* and count_limit
  - ✓ *threshold = timer_inr × (count_limit – 1)* and means no bit flip will be caused by hammering



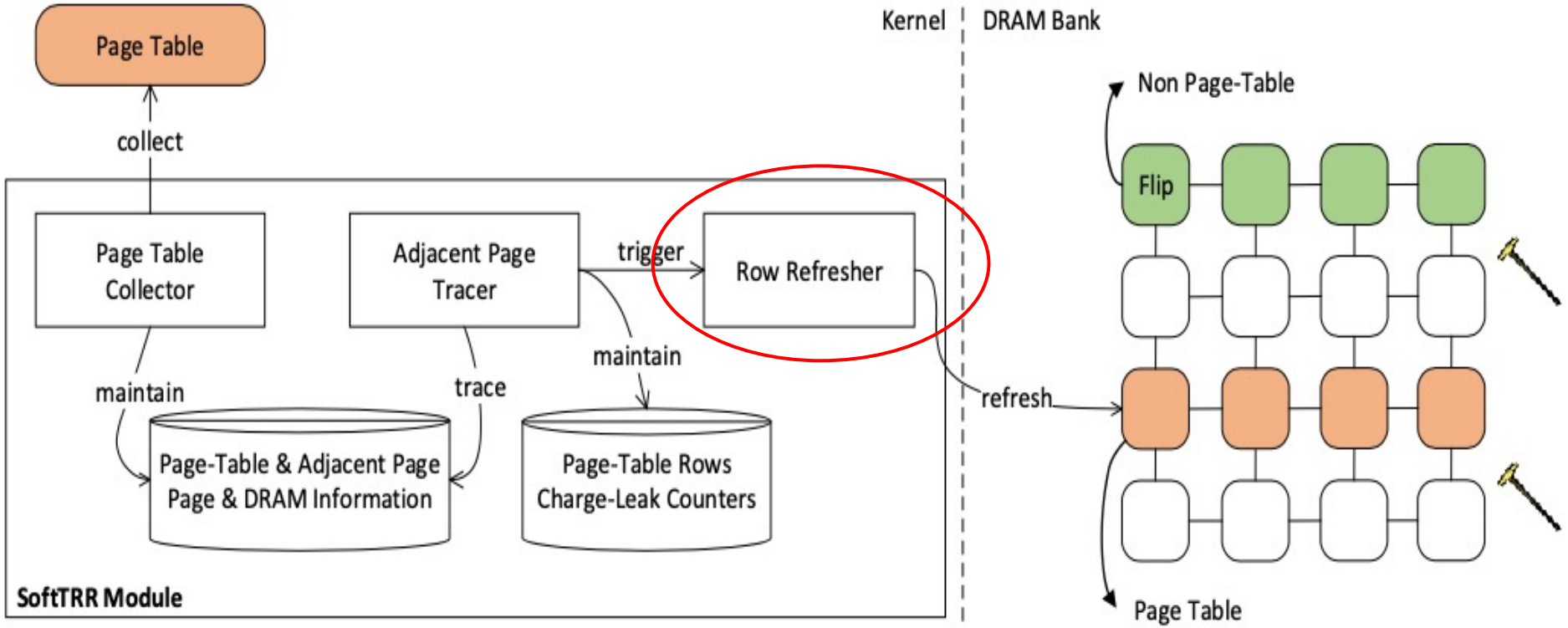
Adjacent Page Tracer

- Set-up tracing periodically
- Determine *timer_intr* and count_limit
  - ✓ *threshold = timer_inr × (count_limit – 1)* and means no bit flip will be caused by hammering
  - ✓ A safe threshold is 1 ms (based on Kim et al. ISCA'20)

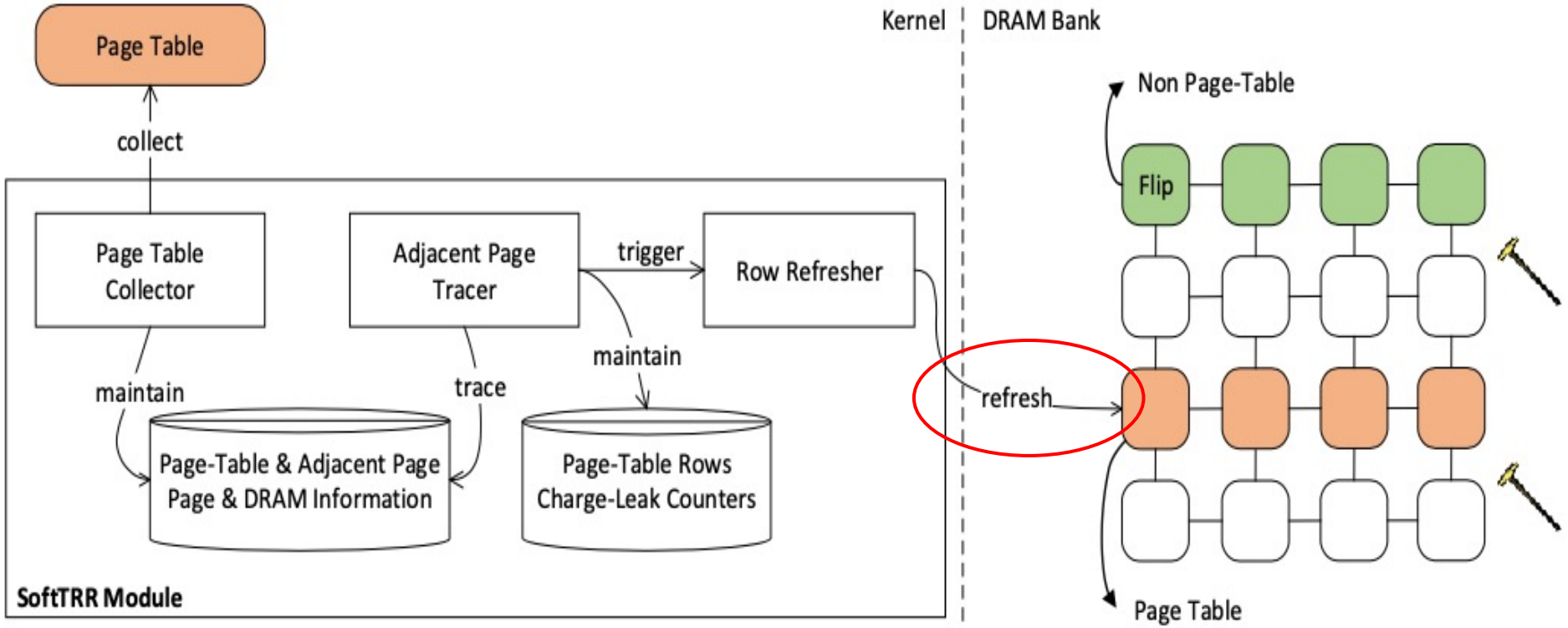
Adjacent Page Tracer

- Set-up tracing periodically
- Determine *timer_intr* and count_limit
  - ✓ *threshold = timer_inr × (count_limit – 1)* and means no bit flip will be caused by hammering
  - ✓ A safe threshold is 1 ms
  - ✓ *timer_inr* is set to 1 ms and *count_limit* is set to 2

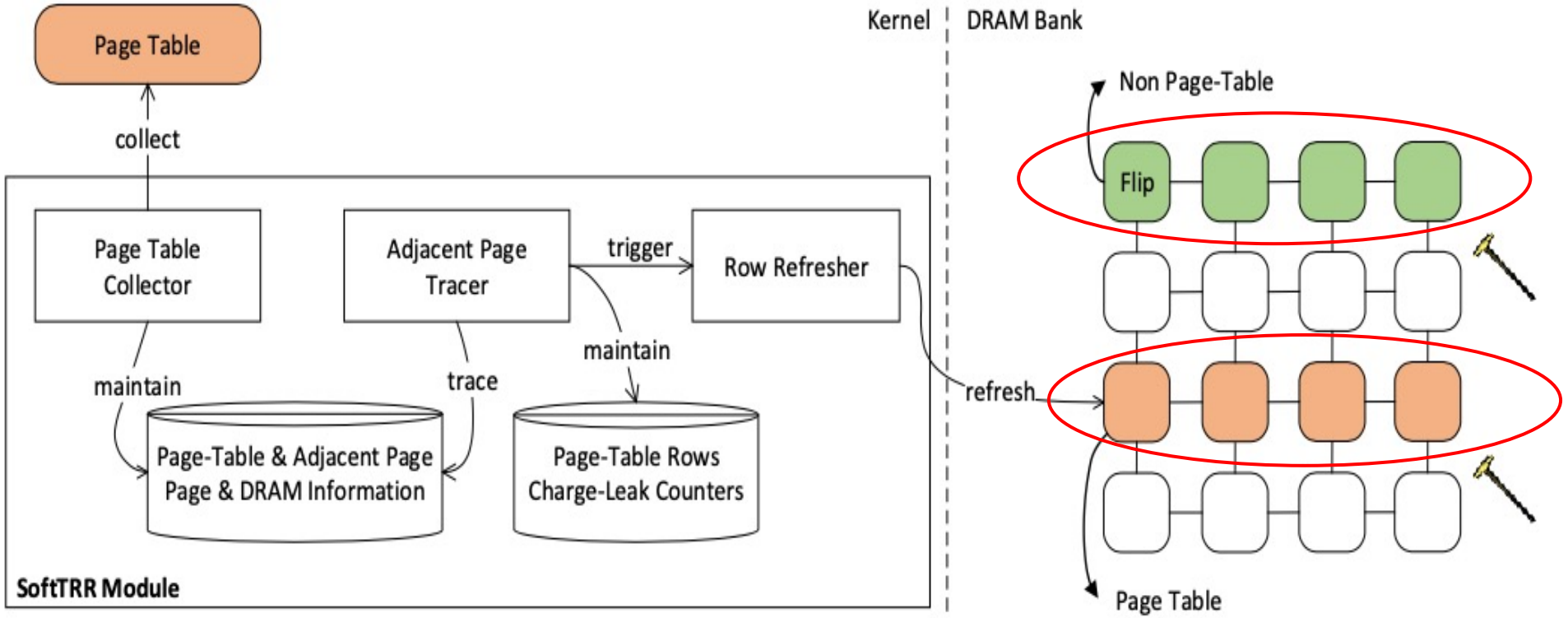
Overview


Page Table
collect
Kernel
DRAM Bank
Page Table Collector
Adjacent Page Tracer
trigger
Row Refresher
maintain
maintain
trace
refresh
Page-Table & Adjacent Page Page & DRAM Information
Page-Table Rows Charge-Leak Counters
SoftTRR Module
Non Page-Table
Flip
Page Table

Overview


Page Table
collect
Kernel
DRAM Bank
Page Table Collector
Adjacent Page Tracer
trigger
Row Refresher
maintain
maintain
trace
maintain
refresh
Page-Table & Adjacent Page Page & DRAM Information
Page-Table Rows Charge-Leak Counters
SoftTRR Module
Non Page-Table
Flip
Page Table

Overview


Page Table
collect
Kernel
DRAM Bank
Page Table Collector
Adjacent Page Tracer
trigger
Row Refresher
maintain
maintain
trace
refresh
Page-Table & Adjacent Page Page & DRAM Information
Page-Table Rows Charge-Leak Counters
SoftTRR Module
Non Page-Table
Flip
Page Table

Row Refresher

- Refresh A Specified Row
  - ✓ A simple read-access to a kernel virtual address can re-charge a specified row and prevent potential bit flips
  - ✓ A kernel virtual address should be mapped to the specified row

Row Refresher

- Refresh A Specified Row
  - ✓ A simple read-access to a kernel virtual address can re-charge a corresponding row and prevent potential bit flips
  - ✓ A kernel virtual address should be mapped to the specified row

- Direct-physical Map
  - ✓ Linux maps available physical memory into the kernel space
  - ✓ A kernel virtual address can be found based on the mapping between a physical address and a DRAM row location, and the direct-physical map

Evaluation

Security Evaluation

Three popular rowhammer attacks target corrupting level-1 page tables:

- Memory Spray (Blackhat'15): explicitly hammers user memory adjacent to L1PTEs
- CATTmew (IEEE TDSC'19): explicitly hammers device driver buffer adjacent to L1PTEs
- Pthammer (MICRO'20): implicitly hammers L1PTEs adjacent to other L1PTEs

Security Evaluation

Three popular rowhammer attacks target corrupting level-1 page tables:

- Memory Spray (Blackhat'15): explicitly hammers user memory adjacent to L1PTEs
- CATTmew (IEEE TDSC'19): explicitly hammers device driver buffer adjacent to L1PTEs
- Pthammer (MICRO'20): implicitly hammers L1PTEs adjacent to other L1PTEs

Machine Model	**Hardware Configuration**			**Attack**	**SoftTRR**
	CPU Arch.	CPU Model	DRAM (Part No.)	*n* Targeted Victim Pages	Bit Flip Failed?
Dell Optiplex 390	KabyLake	i7-7700k	Kingston DDR4 (99P5701-005.A00G)	Memory Spray [46]	✔
Dell Optiplex 990	SandyBridge	i5-2400	Samsung DDR3 (M378B5273DH0-CH9)	CATTmew [13]	✔
Thinkpad X230	IvyBridge	i5-3230M	Samsung DDR3 (M471B5273DH0-CH9)	PThammer [62]	✔

n = 50

Performance Evaluation

Three representative benchmarks:

- SPECspeed 2017 Integer: CPU-focused
- memcached: memory-focused
- Phoronix test suite: system as a whole

Performance Evaluation

Three representative benchmarks:

- SPECspeed 2017 Integer: CPU-focused
- memcached: memory-focused
- Phoronix test suite: system as a whole

Runtime overhead on benchmarks in two scenarios:

- $\Delta\pm1$: where an adjacent row is only 1-row from a row hosting level-1 page tables.
- $\Delta\pm6$: where an adjacent row is up to 6-row from a row hosting level-1 page tables.

Runtime Overhead

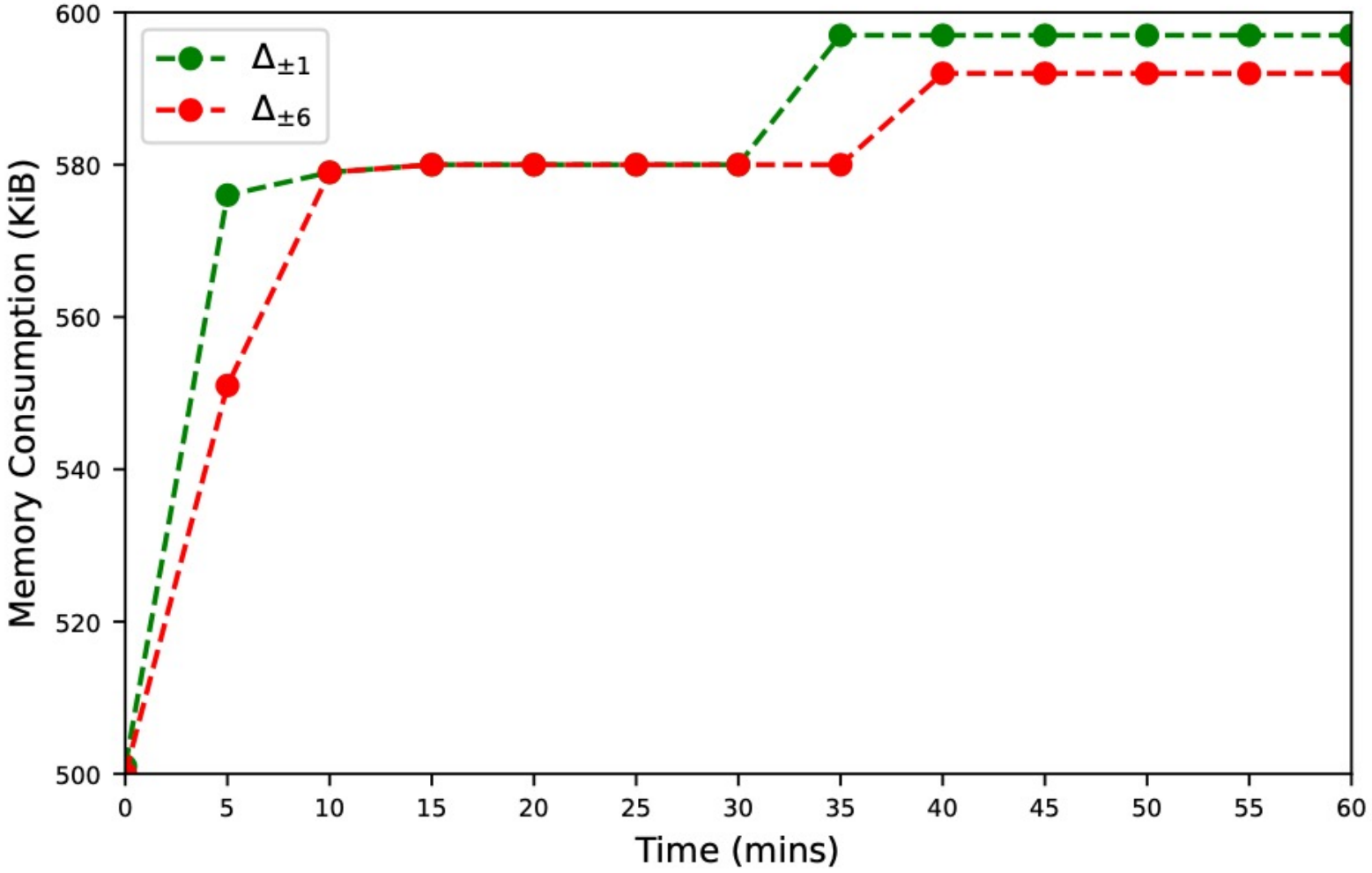
Benchmarks	Programs	SoftTRR Overhead	
		$\Delta_{\pm 1}$	$\Delta_{\pm 6}$ (default)
SPECspeed 2017 Integer	perlbench_s	0.67%	0.67%
	gcc_s	0.23%	0.92%
	mcf_s	-0.76%	0.30%
	omnetpp_s	-0.81%	1.82%
	xalancbmk_s	0.36%	2.50%
	x264_s	0.00%	0.61%
	deepsjeng_s	0.00%	0.28%
	leela_s	0.23%	0.46%
	exchange2_s	-0.70%	-0.23%
	xz_s	1.48%	0.93%
	Mean	0.07%	0.83%
Phoronix	Apache	-0.16%	0.32%
	unpack-linux	1.31%	1.84%
	iozone	0.89%	-1.15%
	postmark	0.89%	0.00%
	stream:Copy	0.01%	0.00%
	stream:Scale	0.60%	0.23%
	stream:Triad	0.07%	0.37%
	stream:Add	0.03%	0.35%
	compress-7zip	1.52%	2.24%
	openssl	0.14%	0.13%
	pybench	0.00%	0.52%
	phpbench	0.92%	0.01%
	cacheben:read	-0.38%	0.26%
	cacheben:write	-0.26%	-0.44%
	cacheben:modify	-0.01%	0.67%
	ramspeed:INT	-0.09%	-0.63%
	ramspeed:FP	-0.15%	-0.63%
	Mean	0.22%	0.24%
memcached	**Statistics**		
	Ops	0.39%	0.18%
	TPS	0.39%	0.15%
	Net_rate	0.46%	0.31%

Runtime Memory Consumption

- In a LAMP (Linux, Apache, MySQL and PHP) system with SoftTRR deployed
- Nikto stresses the LAMP system from another machine

Runtime Memory Consumption

- In a LAMP (Linux, Apache, MySQL and PHP) system with SoftTRR deployed
- Nikto stresses the LAMP system from another machine



System Robustness

Linux Test Project		Vanilla System	SoftTRR	
			$\Delta_{\pm 1}$	$\Delta_{\pm 6}$ (default)
File	open	✔	✔	✔
	close	✔	✔	✔
	ftruncate	✔	✔	✔
	rename	✔	✔	✔
Network	Listen	✔	✔	✔
	Socket	✔	✔	✔
	Send	✔	✔	✔
	Recv	✔	✔	✔
Memory	mmap	✔	✔	✔
	munmap	✔	✔	✔
	brk	✔	✔	✔
	mlock	✔	✔	✔
	munlock	✔	✔	✔
	mremap	✔	✔	✔
Process	getpid	✔	✔	✔
	exit	✔	✔	✔
	clone	✔	✔	✔
Misc.	ioctl	✔	✔	✔
	prctl	✔	✔	✔
	vhangup	✔	✔	✔

✔ the stress test does not report any problem

Conclusion

- ★ SoftTRR is a more effective and practical software-only mitigation, Compared to existing works

- ★ In its implementation, SoftTRR works as a loadable kernel module to defend against rowhammer attacks on L1PT pages. SoftTRR leverages MMU and OS kernel features to collect L1PT pages, track memory access, and refresh target L1PT pages

- ★ SoftTRR is evaluated to be effective against 3 representative rowhammer attacks and incur small overhead and memory footprints

Thanks & Questions?