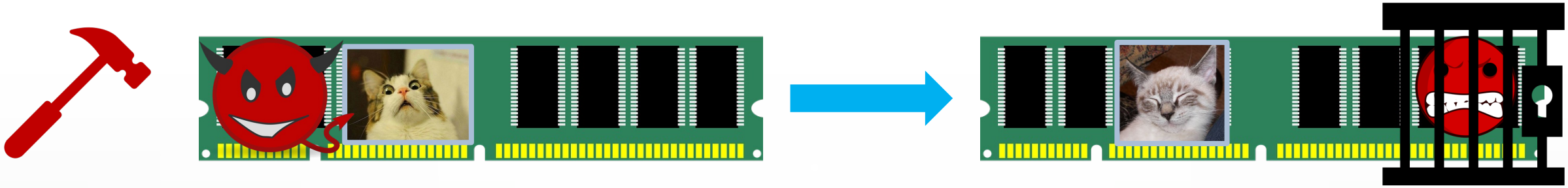


AQUA: SCALABLE ROWHAMMER MITIGATION

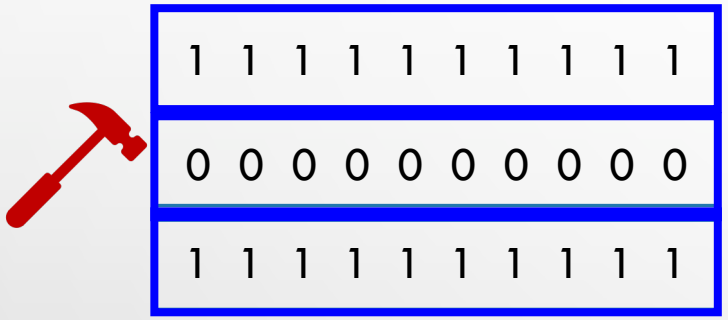
BY QUARANTINING AGGRESSOR ROWS AT RUNTIME

Anish Saxena†, Gururaj Saileshwar†, Prashant J. Nair‡, Moinuddin Qureshi†

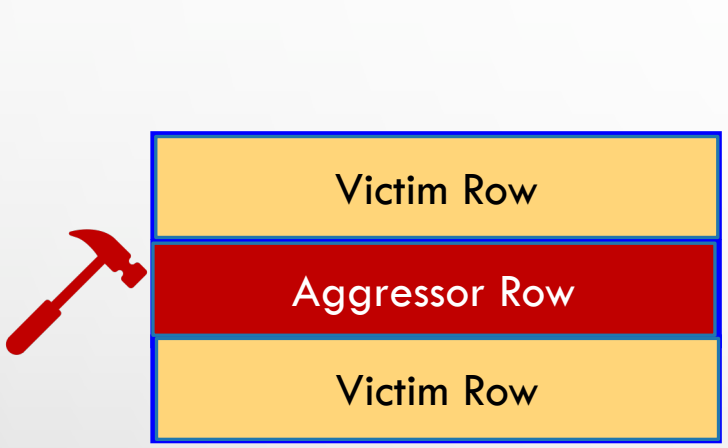
†Georgia Institute of Technology, USA

‡University of British Columbia, Canada

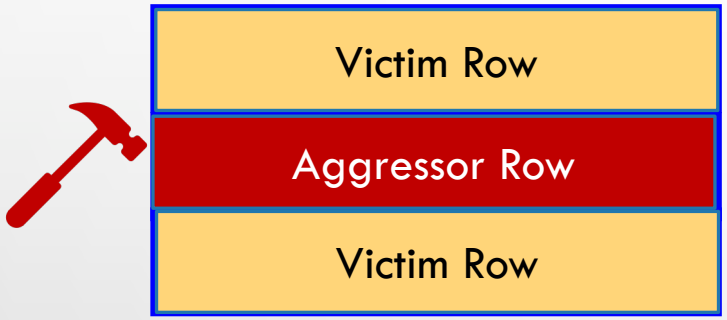
ROWHAMMER



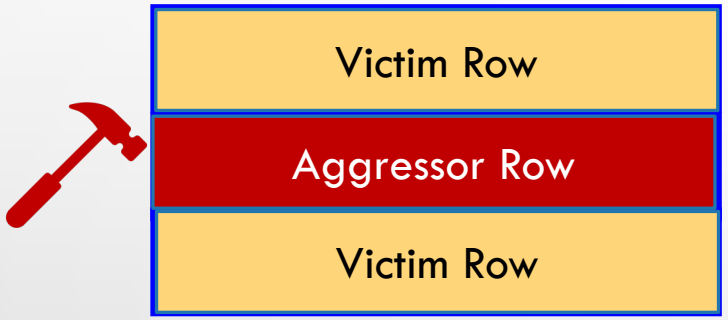
ROWHAMMER

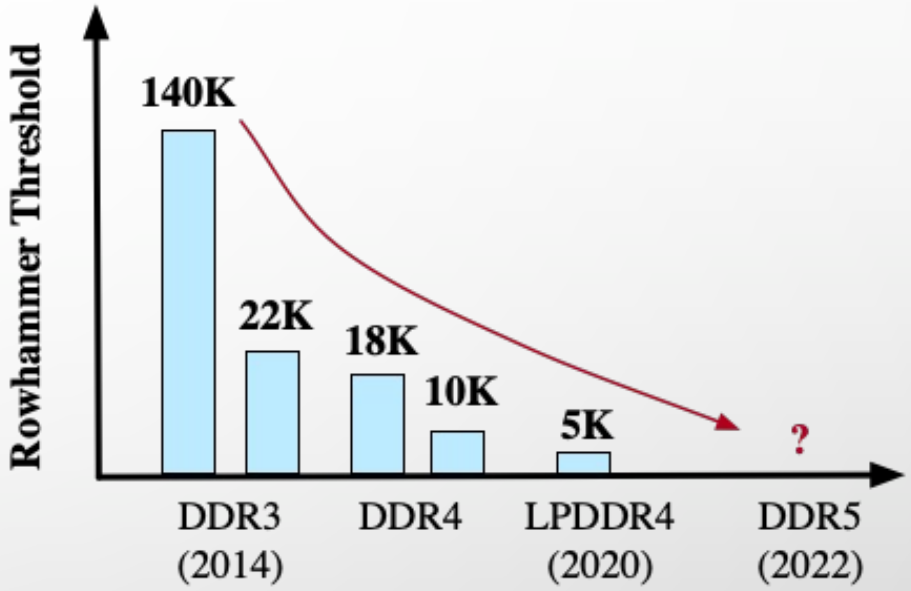


ROWHAMMER



ROWHAMMER

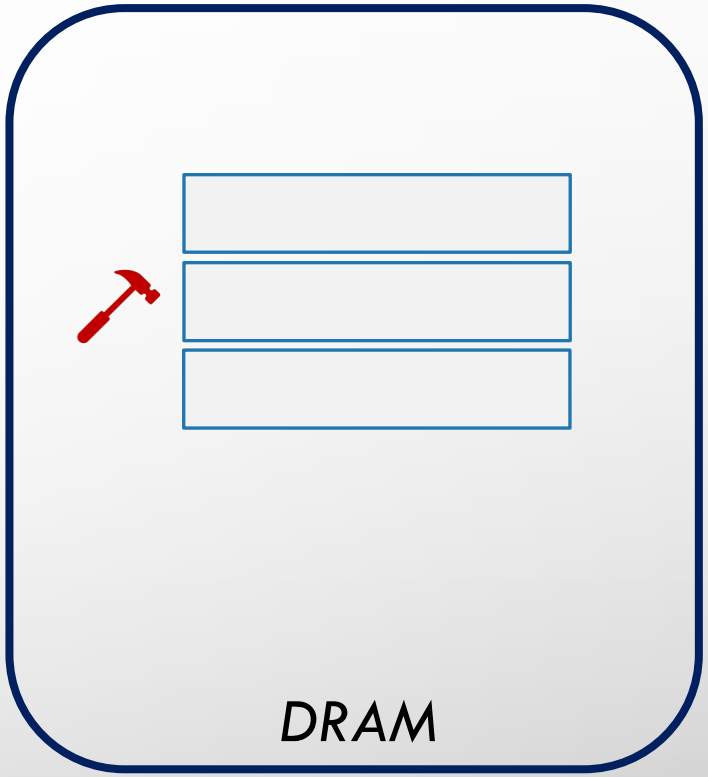




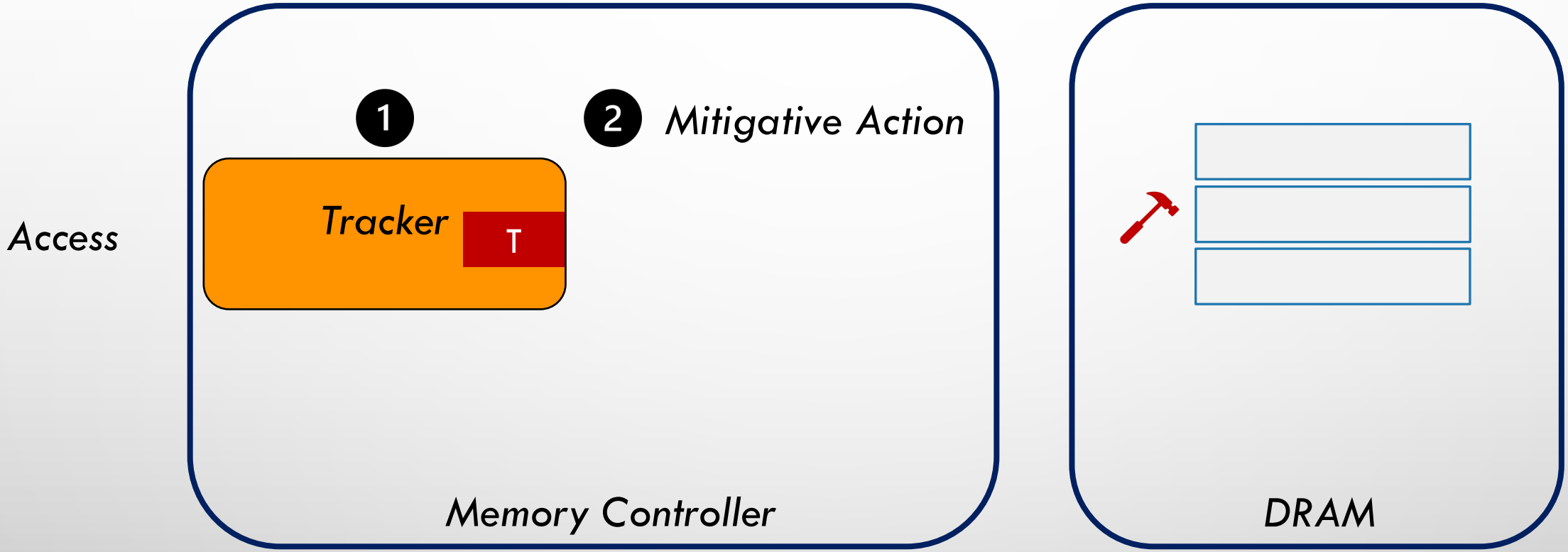
Defenses must scale to plummeting Rowhammer Threshold (TRH)

ROWHAMMER PROTECTION

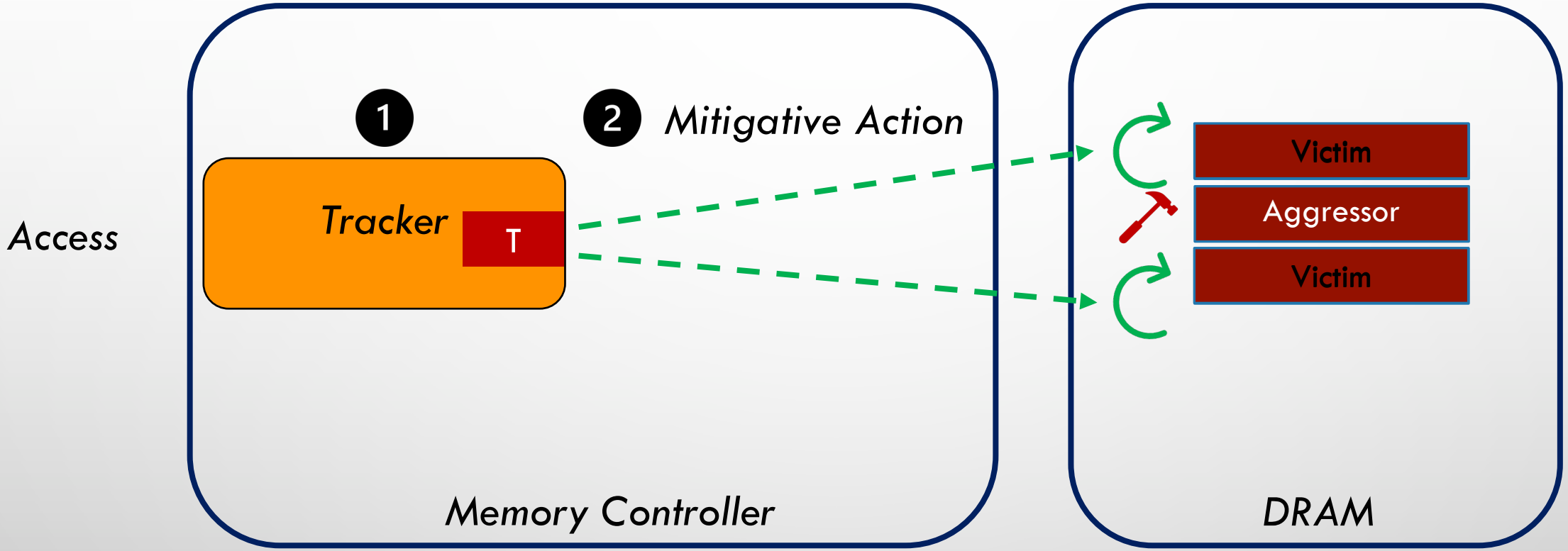
Access



ROWHAMMER PROTECTION

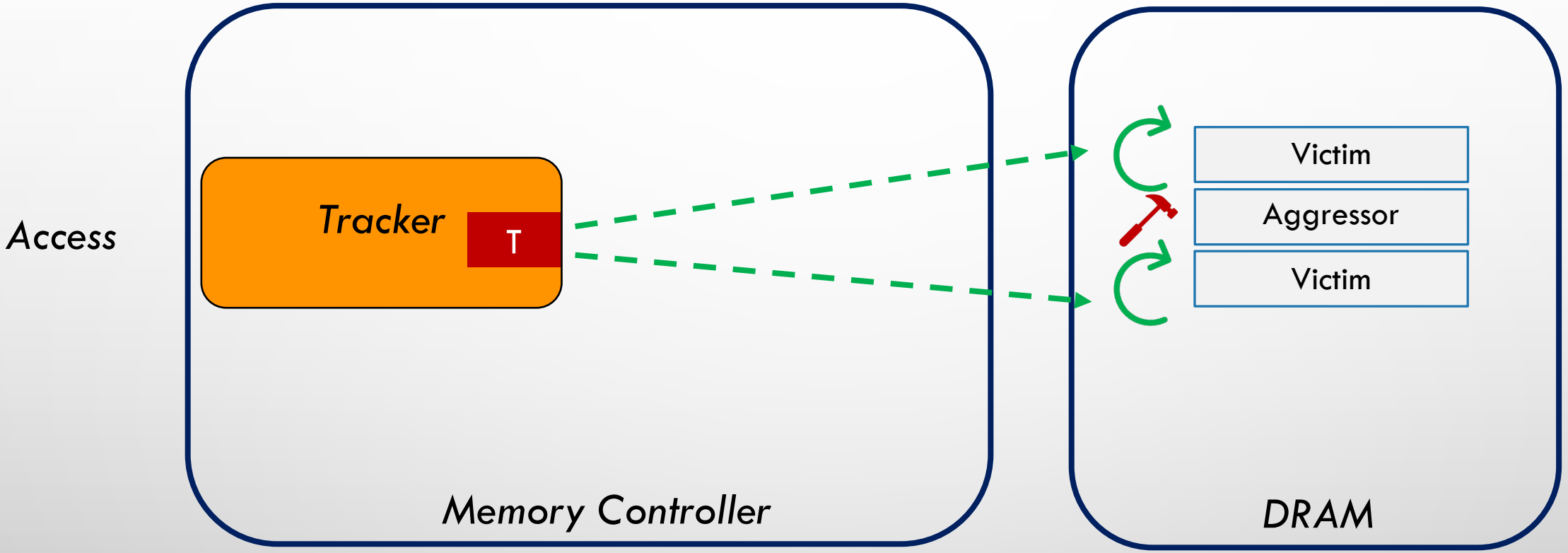


ROWHAMMER PROTECTION

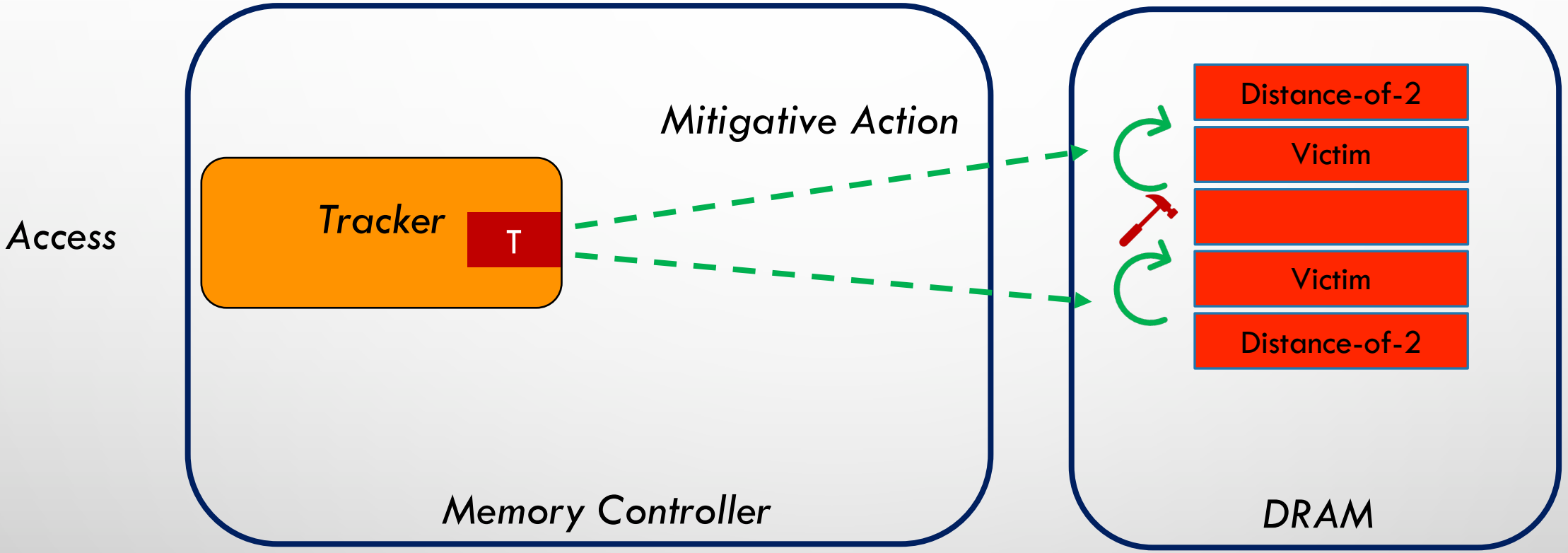


Commerical Rowhammer mitigations rely on victim refresh

HALF-DOUBLE ATTACK

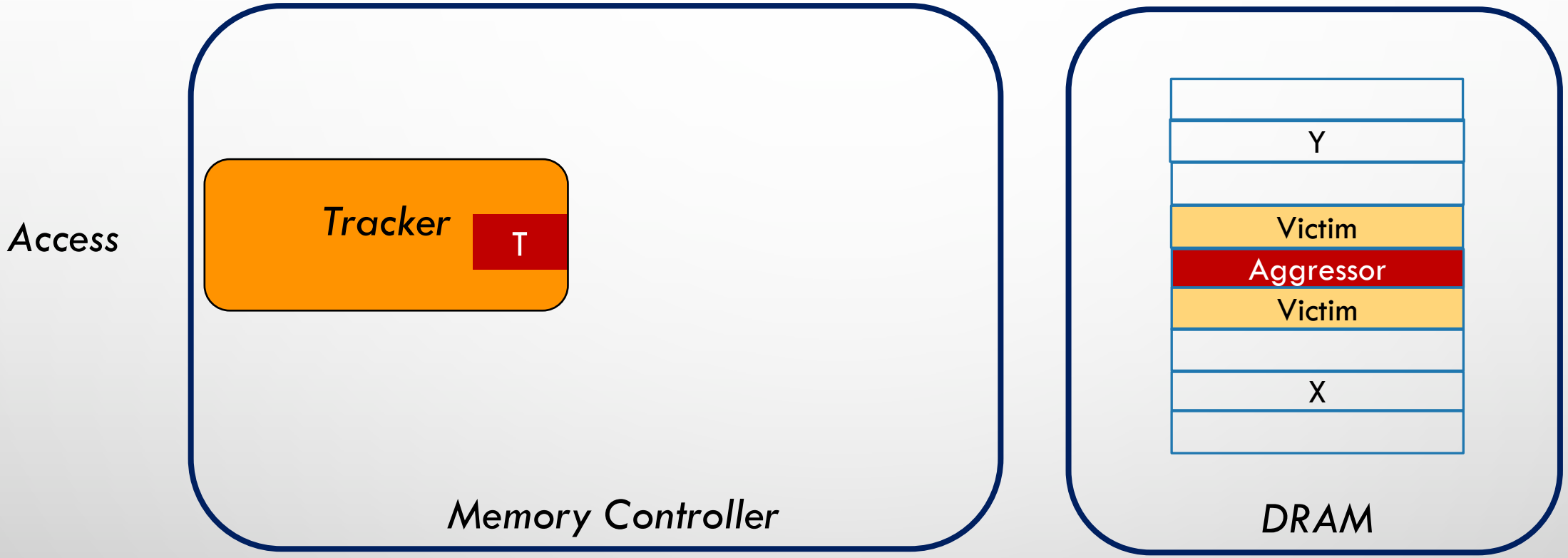


HALF-DOUBLE ATTACK

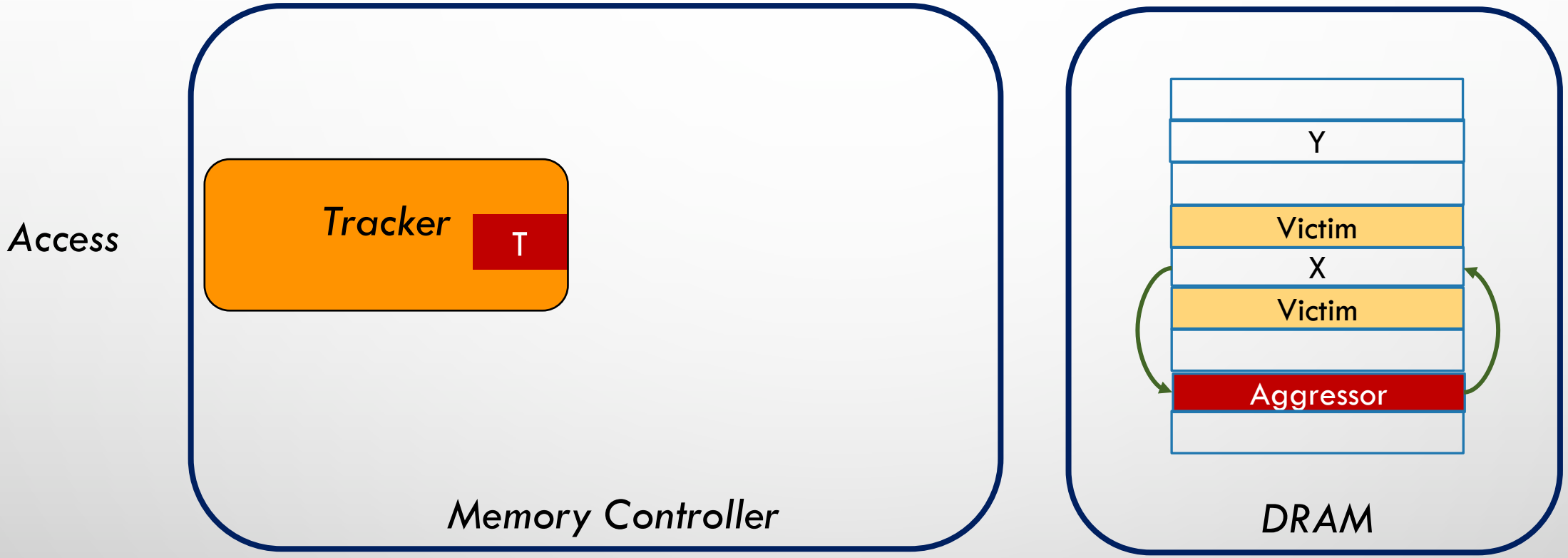


Half-Double breaks all commercial defenses by leveraging victim refreshes

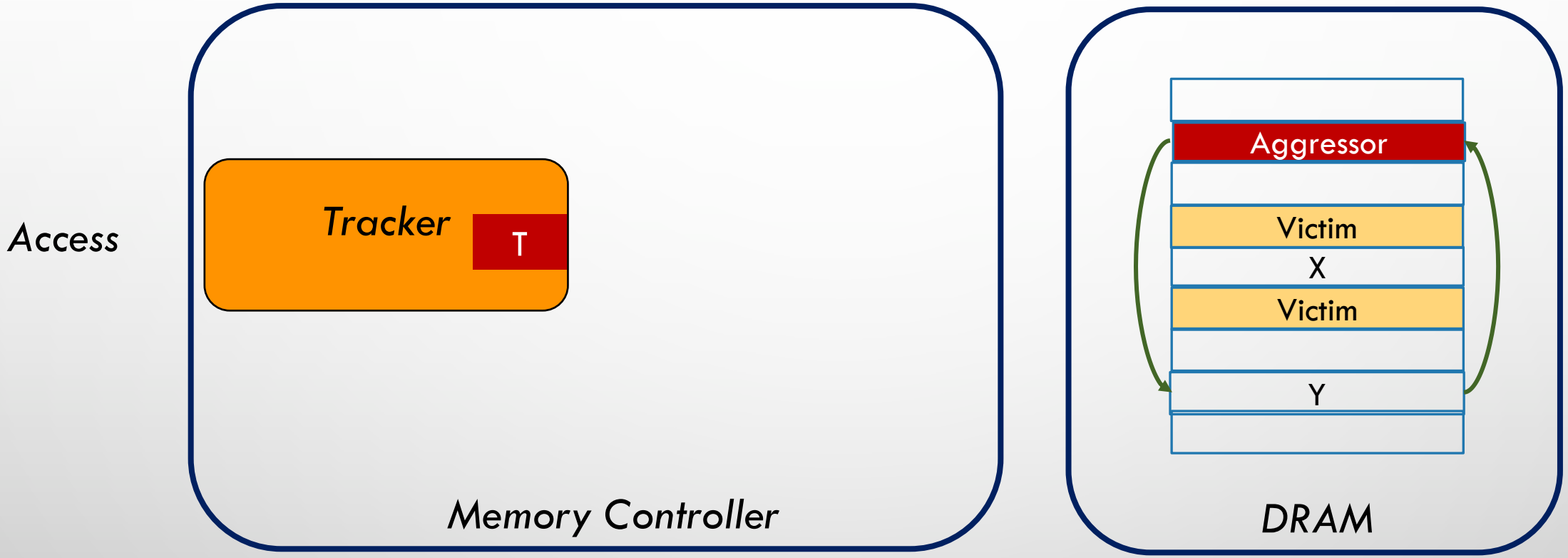
RANDOMIZED ROW-SWAP [ASPLOS'22]



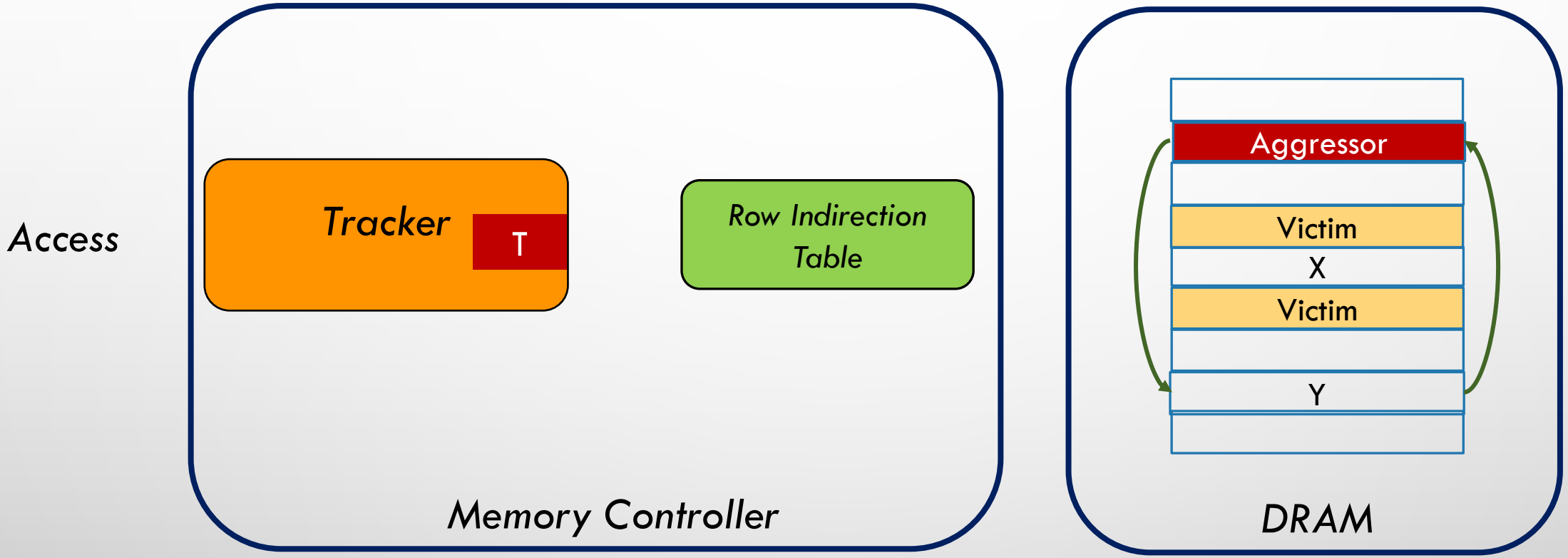
RANDOMIZED ROW-SWAP [ASPLOS'22]



RANDOMIZED ROW-SWAP [ASPLOS'22]



RANDOMIZED ROW-SWAP [ASPLOS'22]



RRS breaks spatial proximity by swapping the aggressor with a randomly selected row

LIMITATIONS OF RANDOMIZED SWAPS

Security via Randomization

LIMITATIONS OF RANDOMIZED SWAPS

Security via Randomization

$$T = TRH/6$$

Current TRH

TRH	T = TRH/6
4K	666

LIMITATIONS OF RANDOMIZED SWAPS

Security via Randomization

$$T = TRH/6$$

Current TRH

TRH	T = TRH/6
4K	666
1K	**166**

Our default TRH

LIMITATIONS OF RANDOMIZED SWAPS

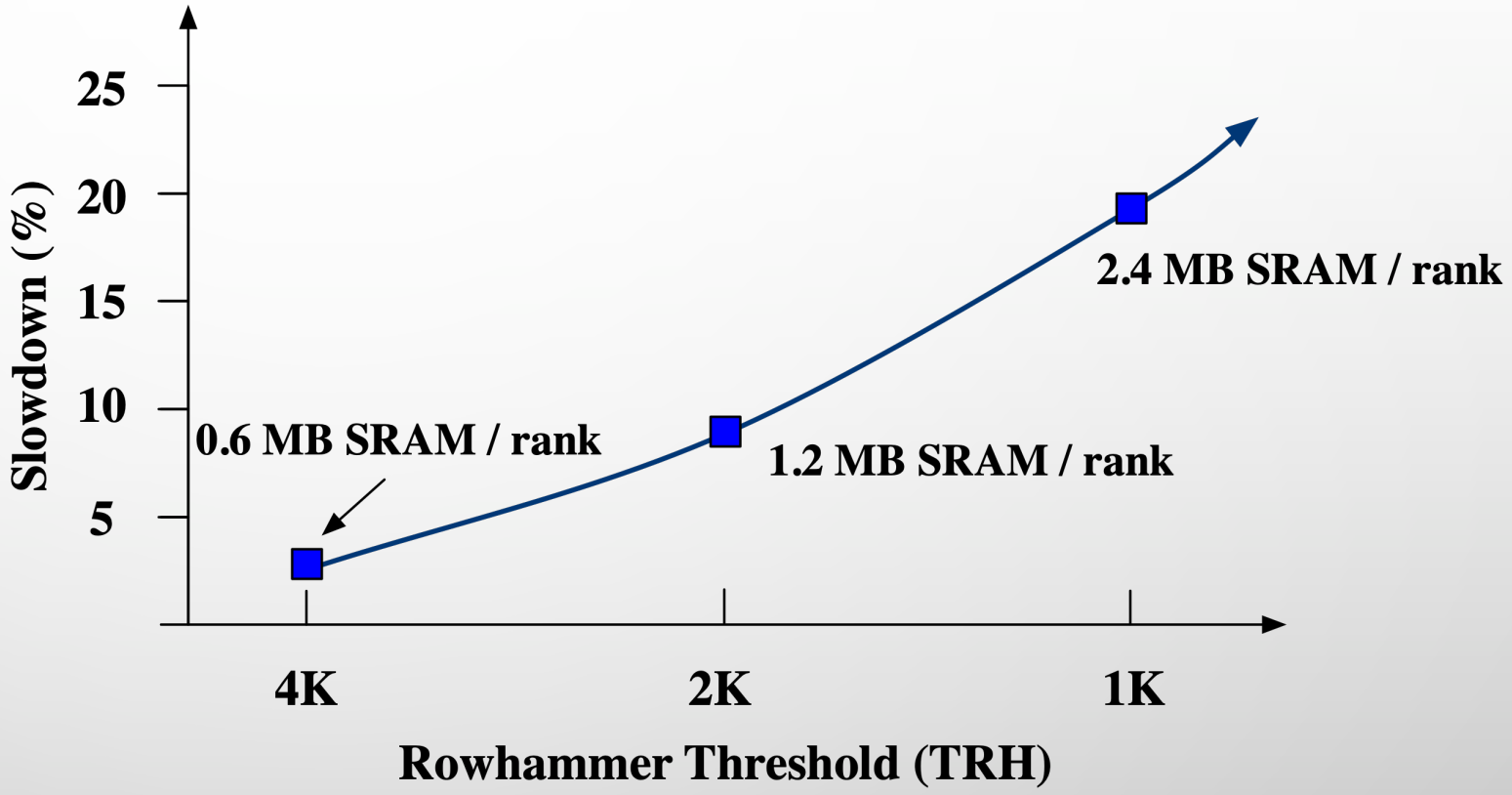
Security via Randomization

$$T = TRH/6$$

Current TRH

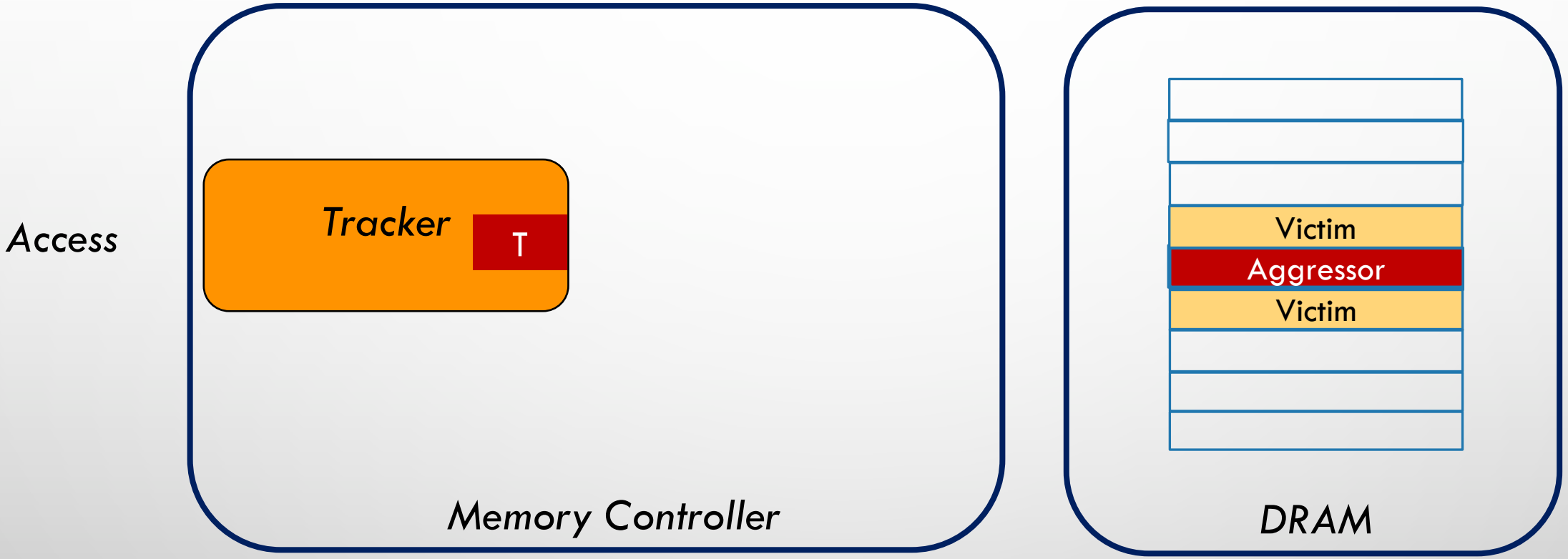
TRH	T = TRH/6
4K	666
1K	166

Our default TRH

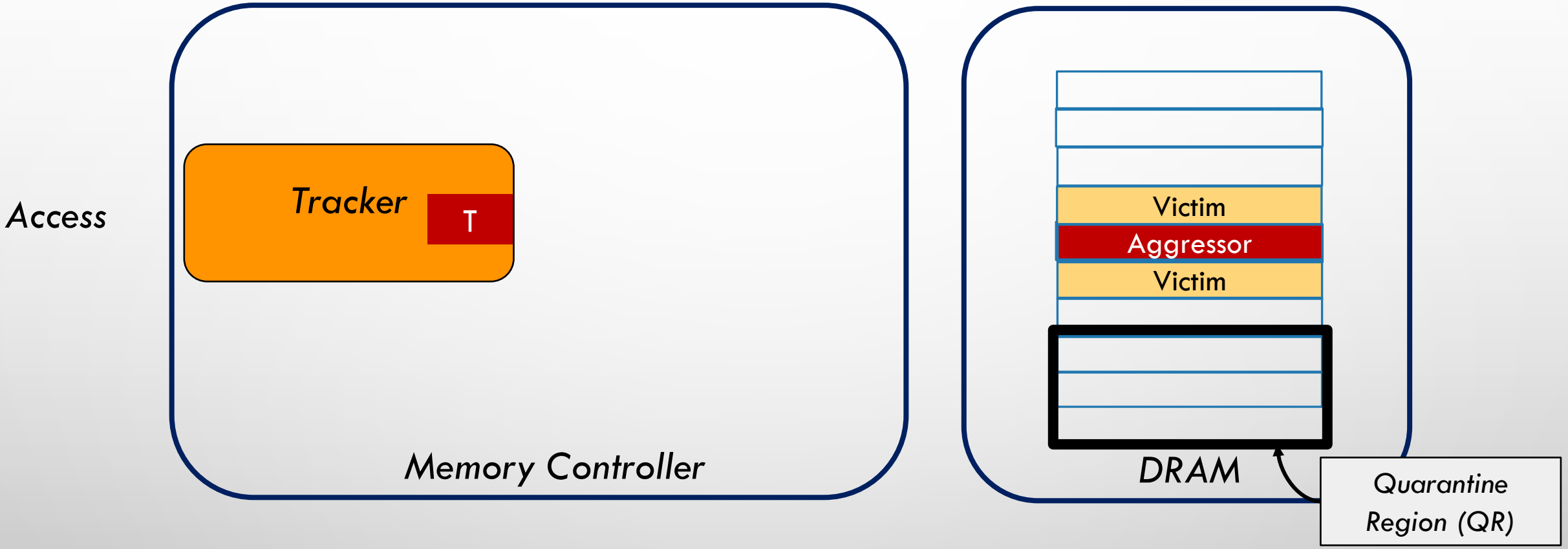


Our goal is to develop a scalable Rowhammer mitigation with low overheads

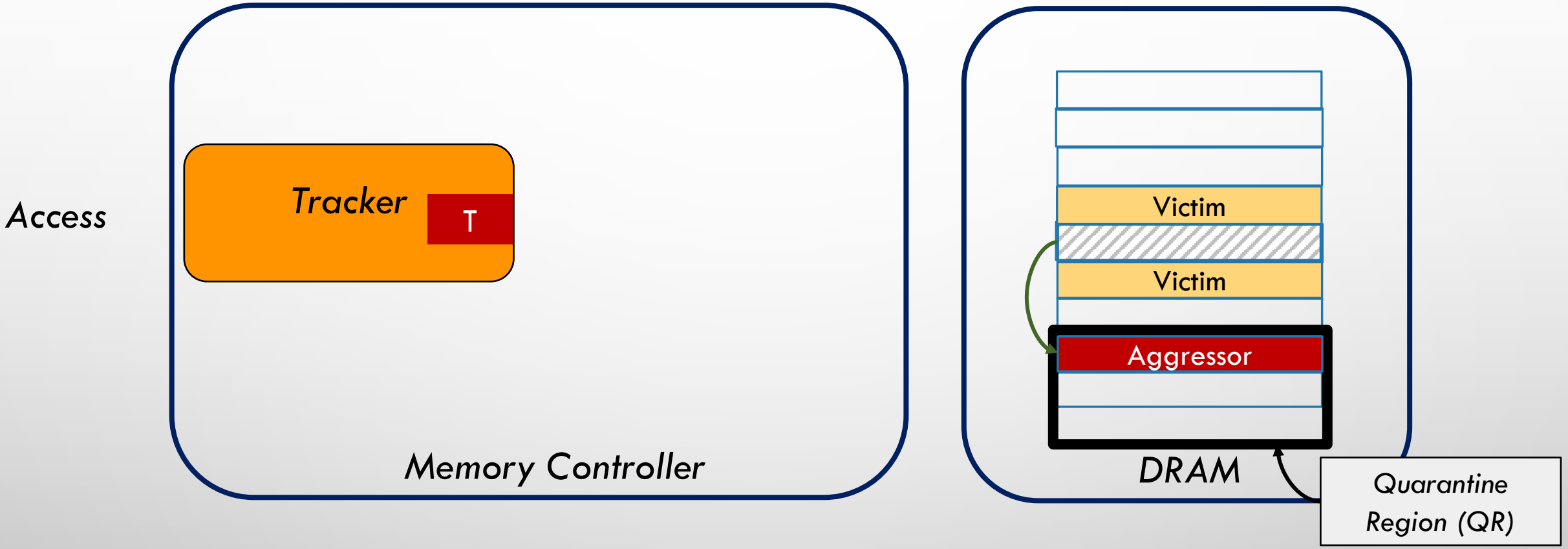
AQUA: QUARANTINING AGGRESSORS



AQUA: QUARANTINING AGGRESSORS

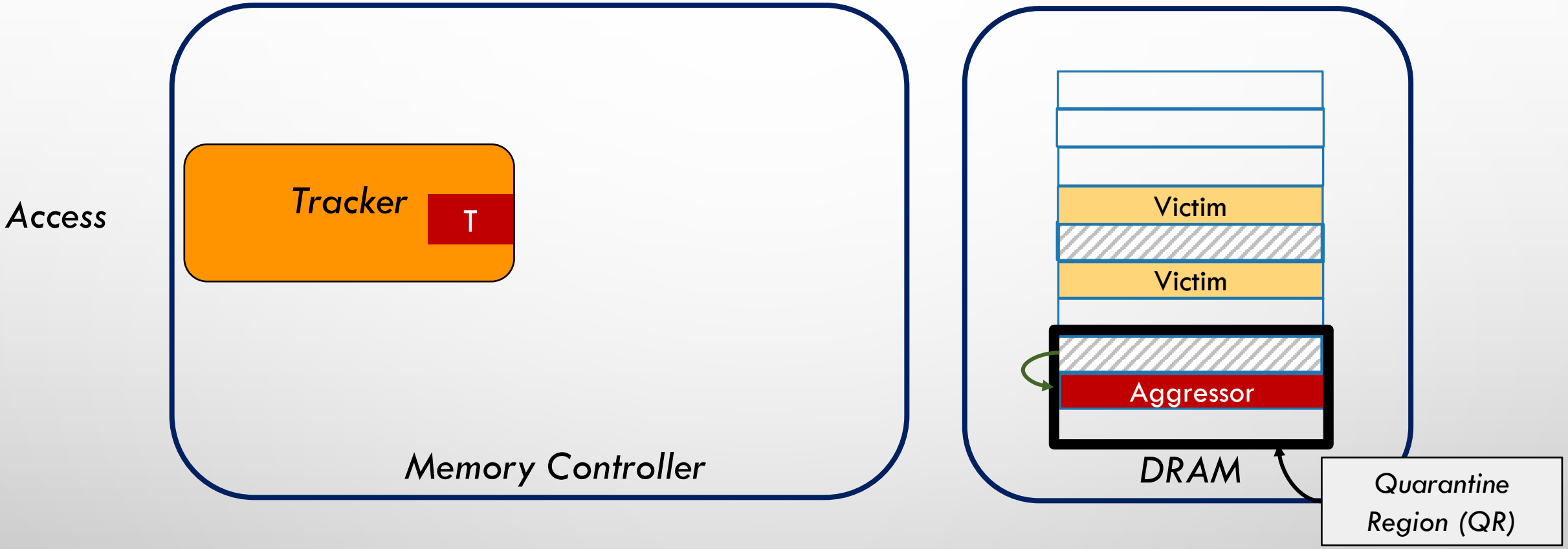


AQUA: QUARANTINING AGGRESSORS



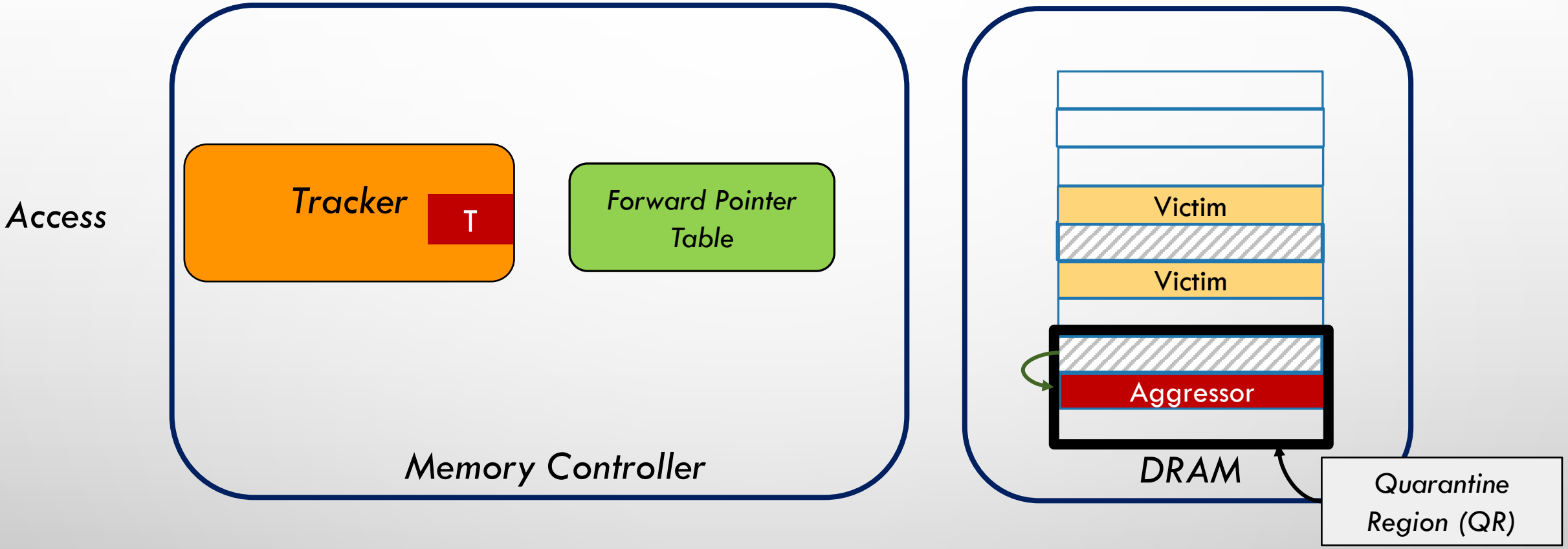
Security via Isolation: AQUA moves rows to a dedicated region of memory

AQUA: QUARANTINING AGGRESSORS



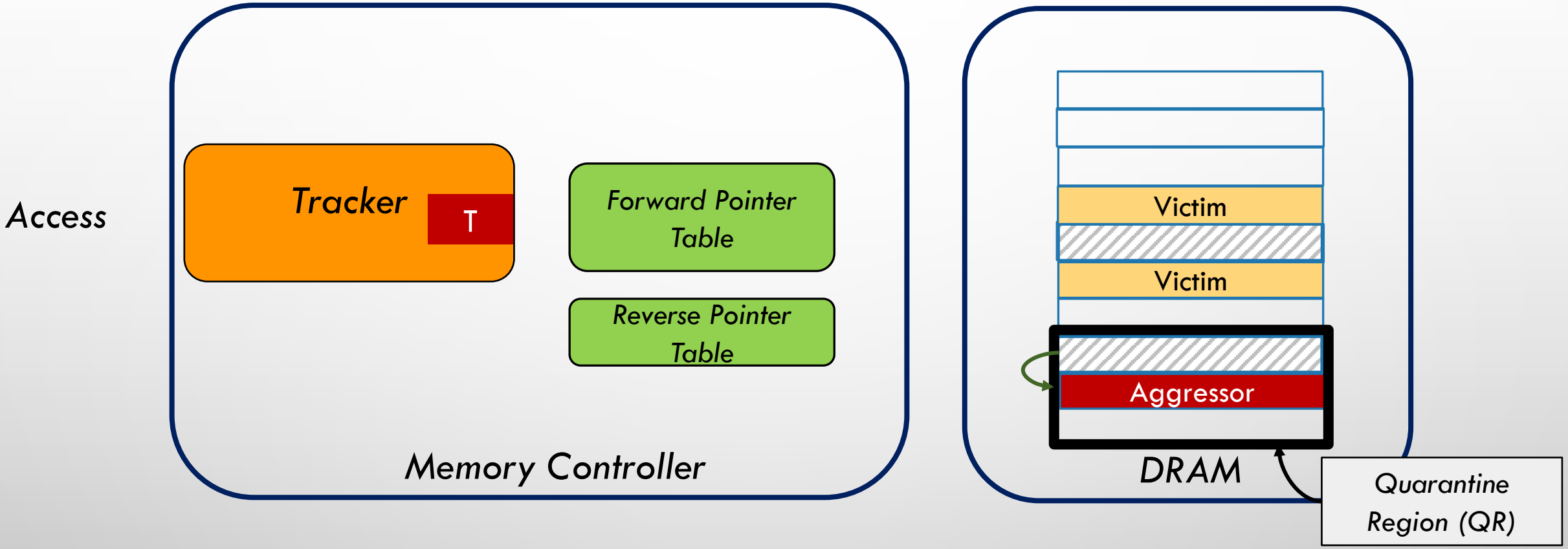
Security via Isolation: AQUA moves rows to a dedicated region of memory

AQUA: QUARANTINING AGGRESSORS



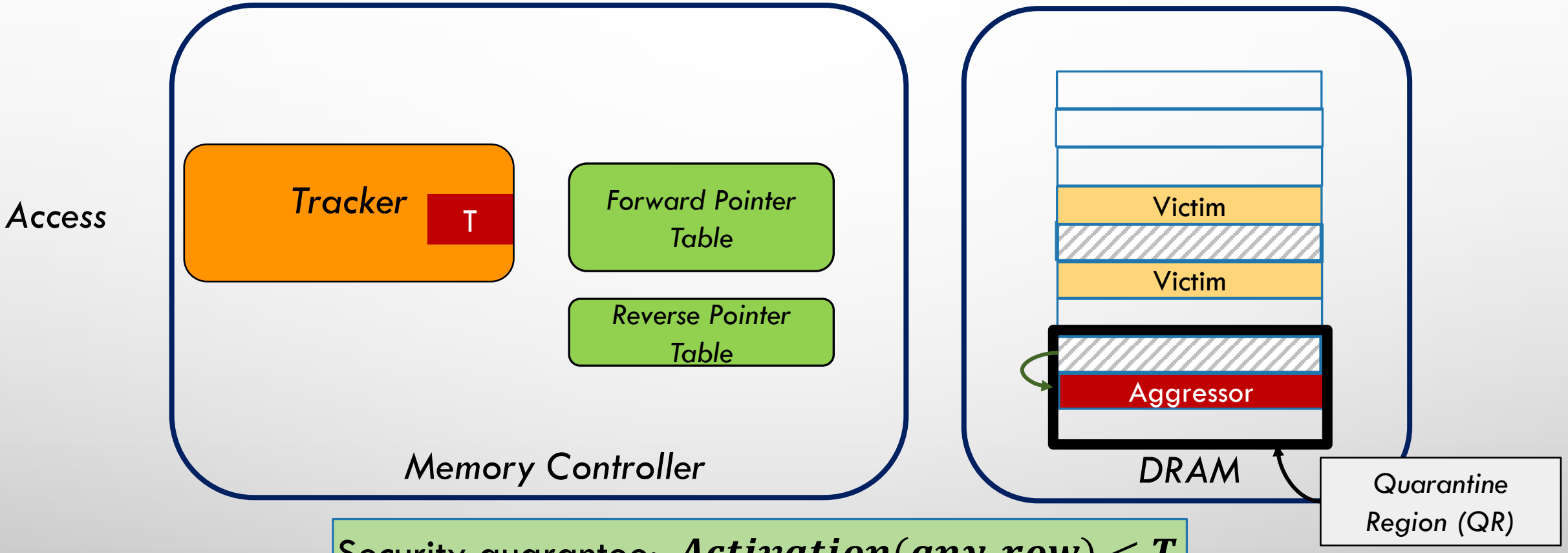
Security via Isolation: AQUA moves rows to a dedicated region of memory

AQUA: QUARANTINING AGGRESSORS



Security via Isolation: AQUA moves rows to a dedicated region of memory

AQUA: QUARANTINING AGGRESSORS



Security guarantee: $Activation(any\_row) < T$

Security via Isolation: AQUA moves rows to a dedicated region of memory

SIZING THE QUARANTINE REGION

SIZING THE QUARANTINE REGION

$$T = \frac{TRH}{2}$$

T	DRAM overhead (16GB memory)
500	1.1%
250	1.5%
125	1.8%

SIZING THE QUARANTINE REGION

$$T = \frac{TRH}{2}$$

T	DRAM overhead (16GB memory)
500	1.1%
250	1.5%
125	1.8%

Quarantine Region Size $\geq$ Max-Rows

SIZING THE QUARANTINE REGION

$$T = \frac{TRH}{2}$$

T	DRAM overhead (16GB memory)
500	1.1%
250	1.5%
125	1.8%

Quarantine Region Size $\geq$ Max-Rows

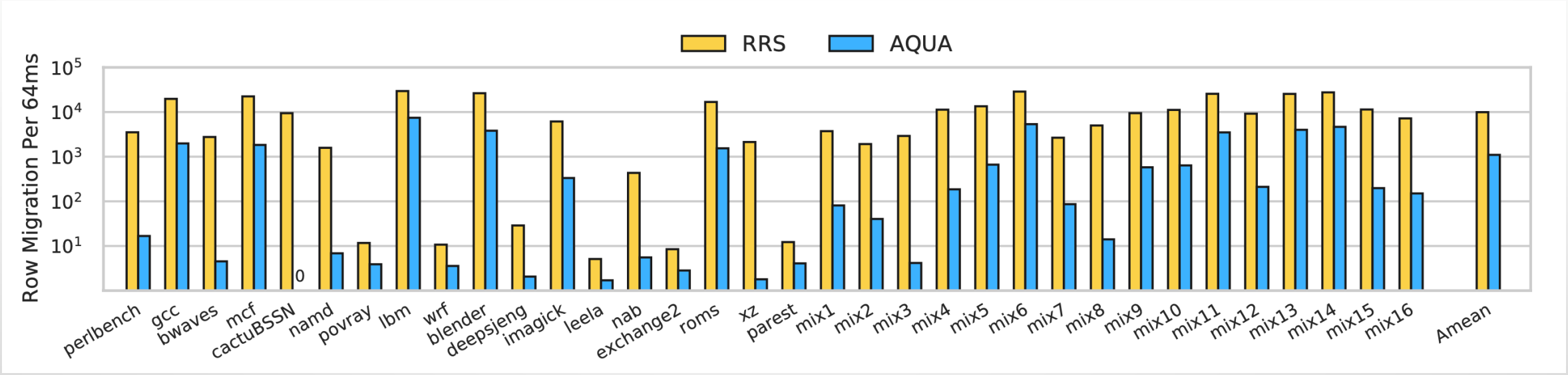
$$\text{Max-Rows} \propto \frac{1}{\text{T, time-to-move}}$$

Time spent performing row migration

Reserving 1-2% of DRAM for the quarantine region prevents Rowhammer

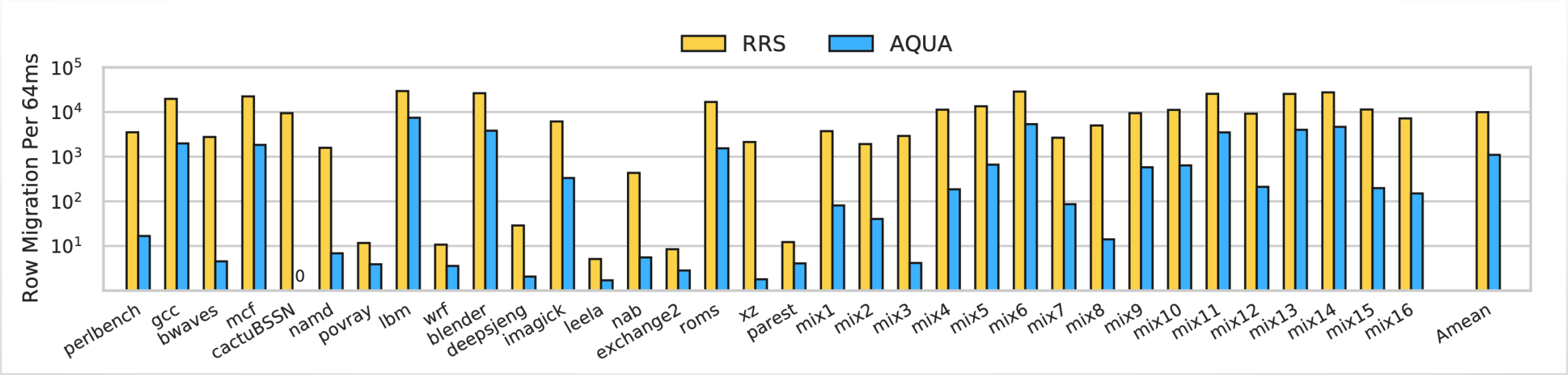
ROW MIGRATIONS

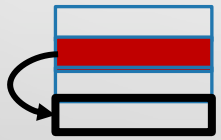
ROW MIGRATIONS



Row migrations reduce by 9X as less mitigations are performed

ROW MIGRATIONS





$$T = \frac{TRH}{2}$$

Row migrations reduce by 9X as less mitigations are performed

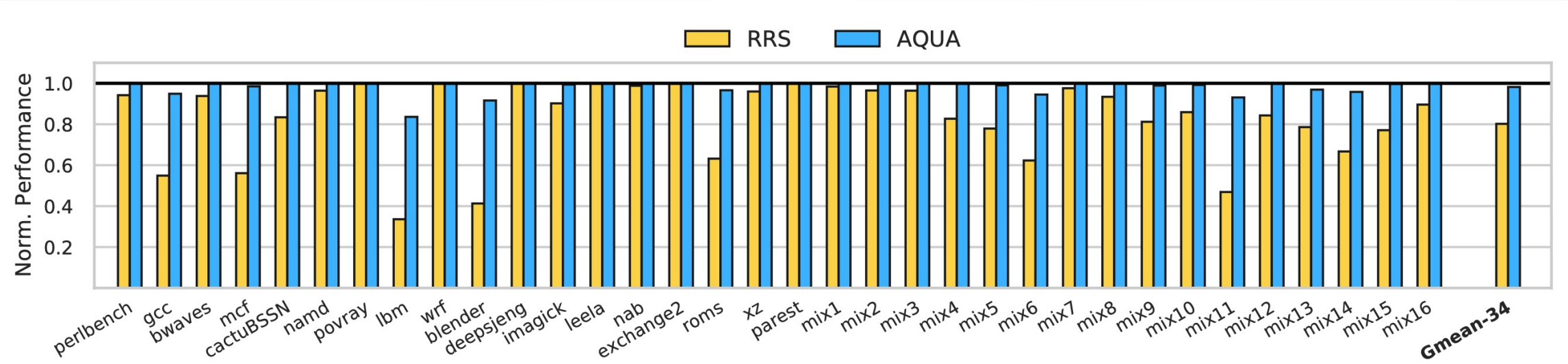
PERFORMANCE

PERFORMANCE

$$Slowdown \propto Migrations$$

PERFORMANCE

$$Slowdown \propto Migrations$$



Slowdown in AQUA: 1.8%

Slowdown in RRS: 19.8%

Less migrations leads to 11X reduction in slowdown

SRAM OVERHEAD

Indirection Tables

SRAM OVERHEAD

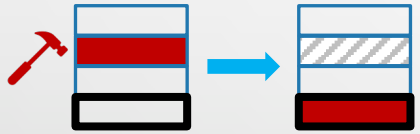
Indirection Tables

RRS	2.4 MB/rank
AQUA-SRAM	172 KB/rank

SRAM OVERHEAD

Indirection Tables

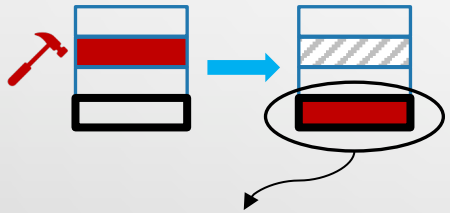
RRS	2.4 MB/rank
AQUA-SRAM	172 KB/rank



SRAM OVERHEAD

Indirection Tables

RRS	2.4 MB/rank
AQUA-SRAM	172 KB/rank

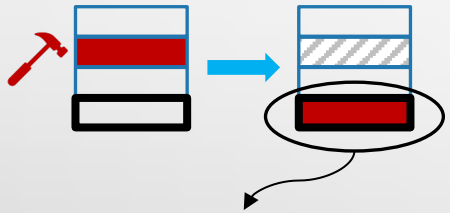


Not a secret

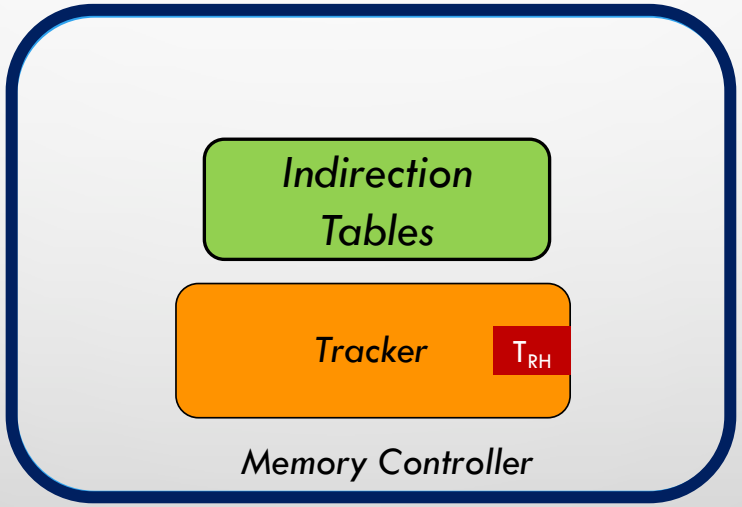
SRAM OVERHEAD

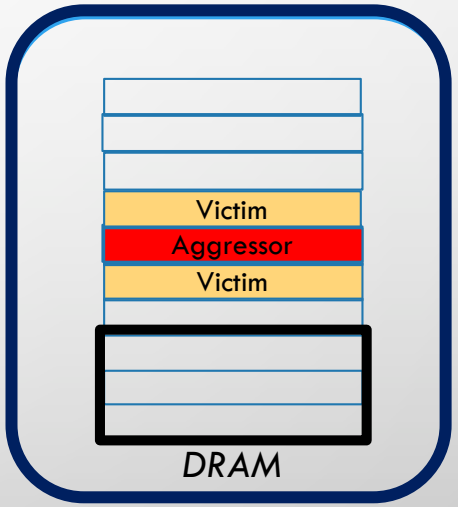
Indirection Tables

RRS	2.4 MB/rank
AQUA-SRAM	172 KB/rank



Not a secret



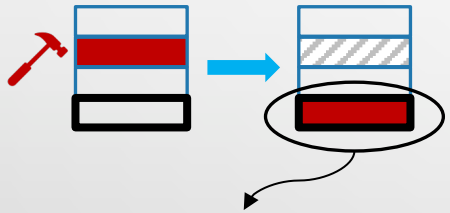


SRAM OVERHEAD

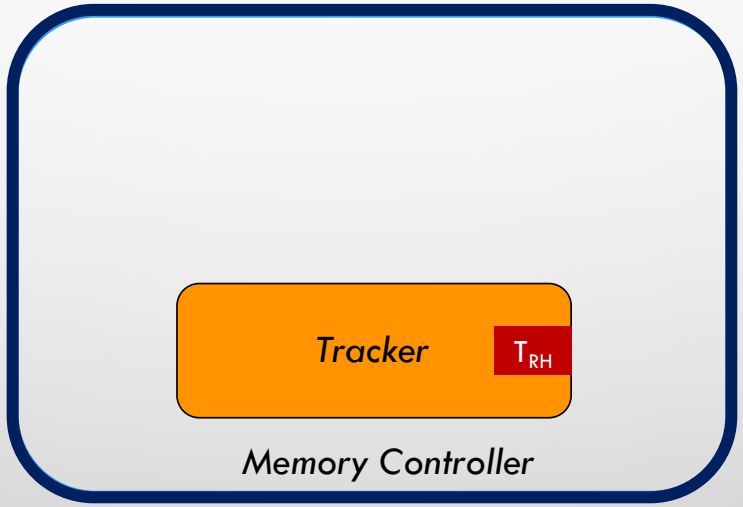
Indirection Tables

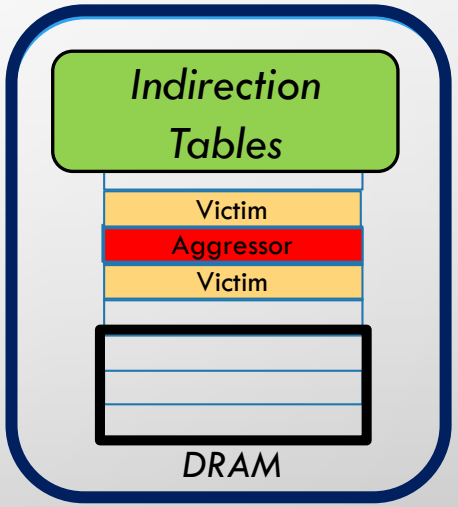
RRS	2.4 MB/rank
AQUA-SRAM	172 KB/rank
AQUA-DRAM	**32 KB /rank**

Slowdown
1.8% → 2.1%



Not a secret

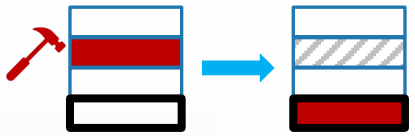




SRAM overhead for AQUA's mitigation is 41KB per rank, 60X lower than RRS

CONCLUSION

- ✓ Breaks spatial proximity by isolating aggressors in 1-2% DRAM
- ✓ Uses the same effective threshold as the tracker
- ✓ Fewer rows need mitigation and fewer mitigations per row
- ✓ 60X less SRAM and 11X less slowdown compared to RRS



CONCLUSION

- ✓ Breaks spatial proximity by isolating aggressors in 1-2% DRAM
- ✓ Uses the same effective threshold as the tracker
- ✓ Fewer rows need mitigation and fewer mitigations per row
- ✓ 60X less SRAM and 11X less slowdown compared to RRS

THANK YOU

https://tinyurl.com/AQUA-code

BACKUP SLIDES

TRACKER OVERHEADS

Structure	**RRS-MG**	**AQUA-MG**	**RRS-Hydra**	**AQUA-Hydra**
Tracker	396 KB	396 KB	28.3 KB	30.3 KB
Mapping Table(s)	2.4 MB	32.6 KB	2.4 MB	32.6 KB
Buffer(s)	16 KB	8 KB	16 KB	8 KB
Total	2,870 KB	437 KB	2,502 KB	71 KB

TRH = 1K

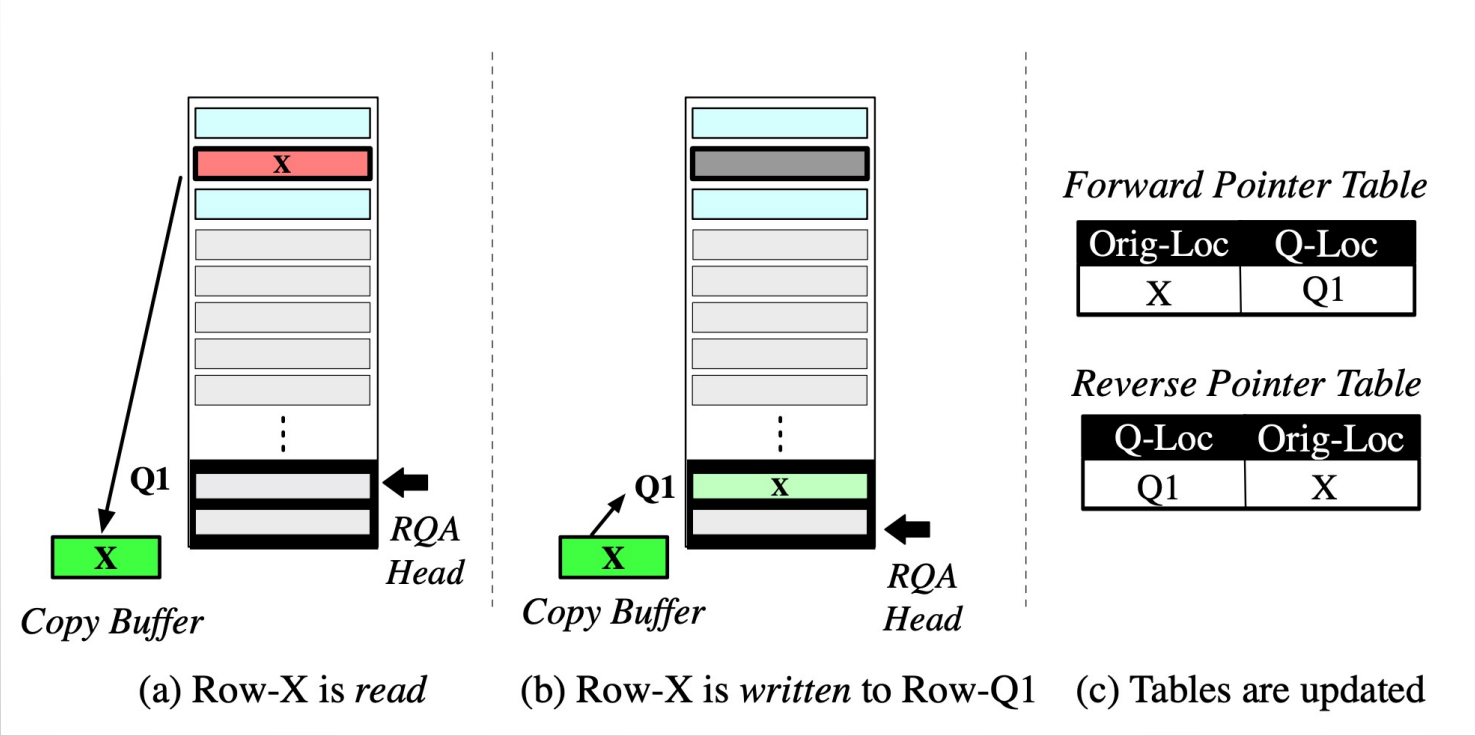
AQUA COMPARED TO VICTIM REFRESH

Attribute	Victim-Refresh	AQUA
Slowdown	<0.2%	2.1%
Mitigates Classic Rowhammer (Neighboring Row Bit Flips)	✓	✓
Mitigates Complex Patterns (Far Aggressors of Half-Double)	✗	✓
Works Without Knowing DRAM Mapping	✗	✓

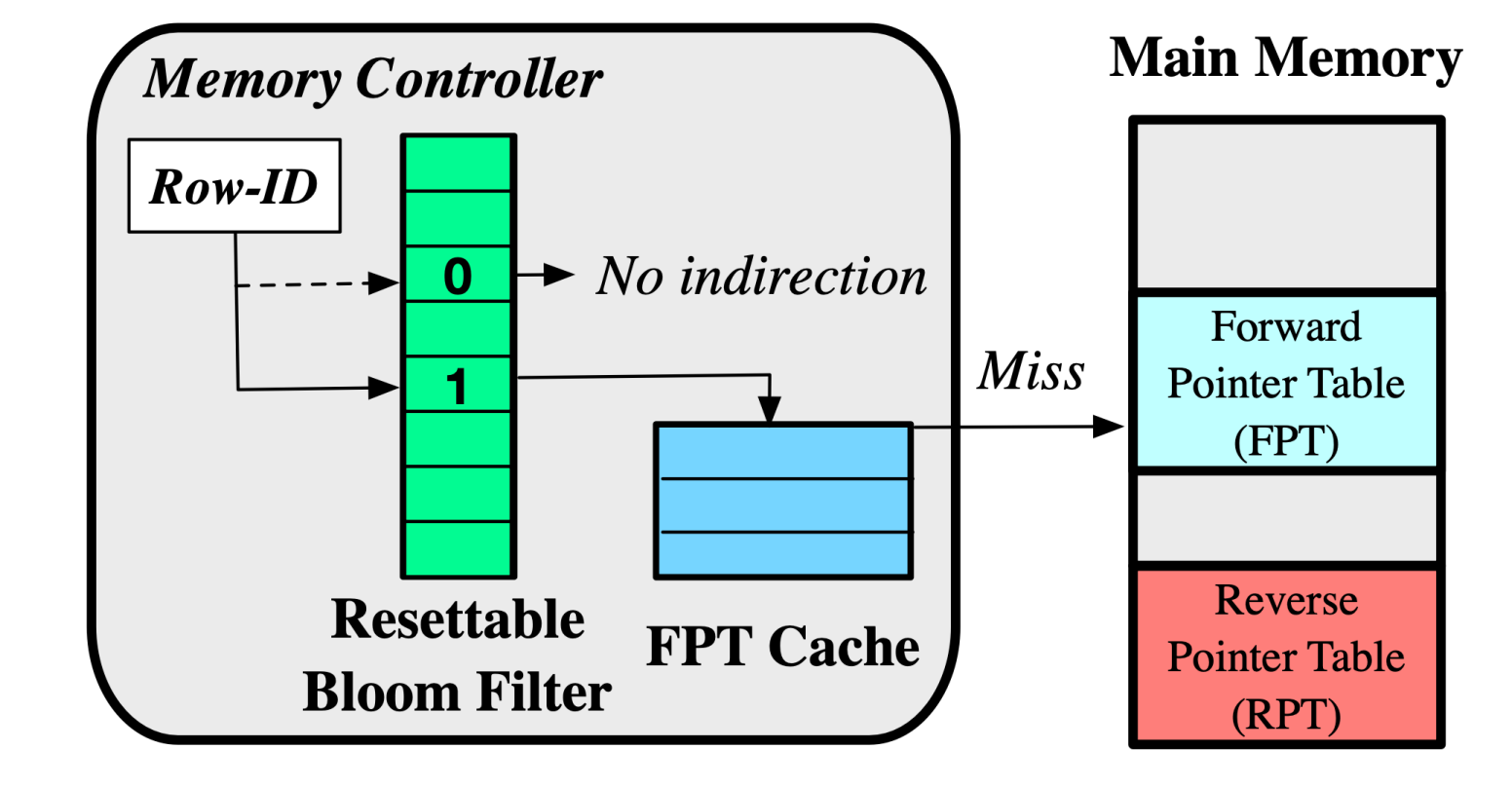
AQUA COMPARED TO PRIOR WORKS

Metric	Blockhammer	CROW	CROW-Agg	RRS	AQUA
SRAM for Mapping Tables	N/A	**26 MB**	32 KB	**2.4 MB**	41KB
DRAM Storage Overhead	0%	**1060%**	**530%**	0%	1.1%
Normalized Perf. Loss (Avg)	**36%**	<0.1%	<0.1%	**19.8%**	2.1%
Worst-Case Slowdown	**1280**×	<1%	<1%	11×	3×
Commodity DRAM	Yes	**NO**	**NO**	Yes	Yes

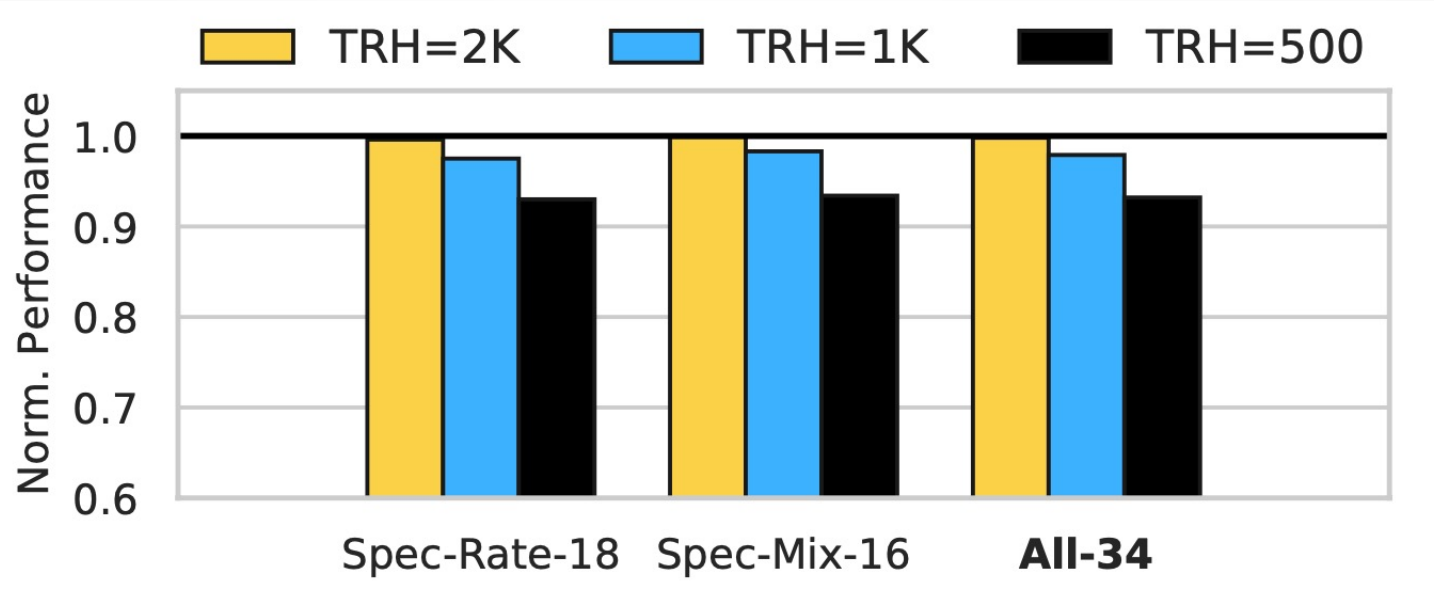
ROW MIGRATION WITH SRAM-BASED COPY BUFFER



AQUA WITH MEMORY-MAPPED TABLES



SENSITIVITY TO ROWHAMMER THRESHOLD

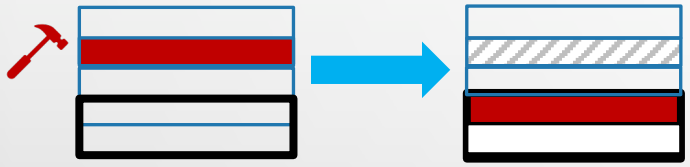


OVERHEADS OF CROW

Copy-Rows	DRAM Overhead	Aggressors	T_{RH} Tolerated
8 (default)	1.6%	4	340K
32	6.3%	16	85K
128	25%	64	21.3K
512	100%	256	5.3K

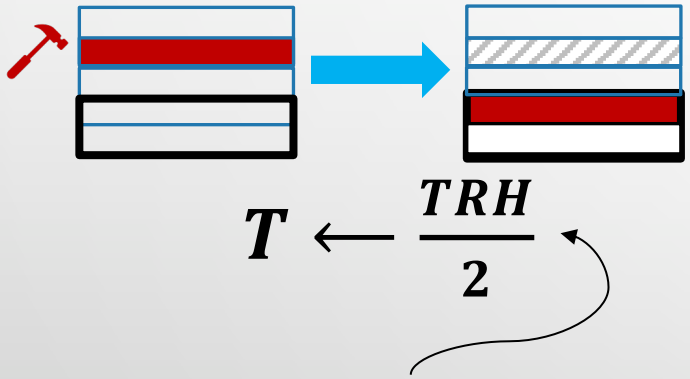
SIZING THE QUARANTINE REGION

Security via Isolation



SIZING THE QUARANTINE REGION

Security via Isolation



$$T \leftarrow \frac{TRH}{2}$$

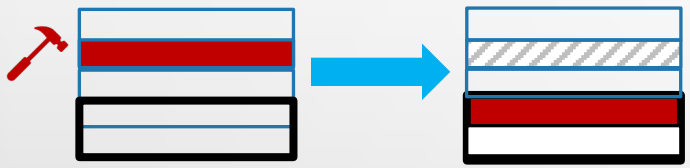
Due to tracker inefficiency

SIZING THE QUARANTINE REGION

Quarantine Region

$$QR\_size \geq Max\_Rows$$

Security via Isolation



$$T \leftarrow \frac{TRH}{2}$$

Due to tracker inefficiency

SIZING THE QUARANTINE REGION

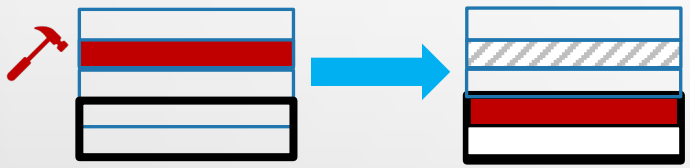
Quarantine Region

$$QR\_size \geq Max\_Rows$$

$$Max\_Rows \propto \frac{1}{\mathrm{T},\ time\_to\_move}$$

Time spent performing row migration

Security via Isolation

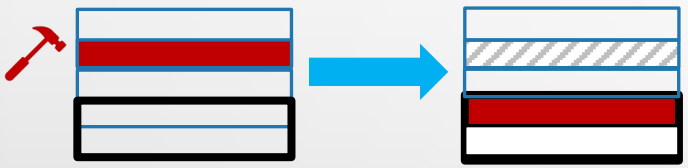


$$T \leftarrow \frac{TRH}{2}$$

Due to tracker inefficiency

SIZING THE QUARANTINE REGION

Security via Isolation



$$T \leftarrow \frac{TRH}{2}$$

Due to tracker inefficiency

Quarantine Region

$$QR\_size \geq Max\_Rows$$

$$Max\_Rows \propto \frac{1}{\mathrm{T},\ time\_to\_move}$$

Time spent performing row migration

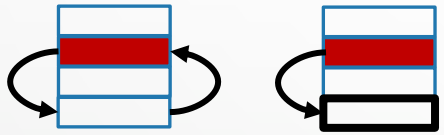
T	DRAM overhead (16GB memory)
500	1.1%
250	1.5%
125	1.8%

Reserving 1-2% of DRAM for the quarantine region is sufficient to prevent Rowhammer

ROW MIGRATIONS

ROW MIGRATIONS

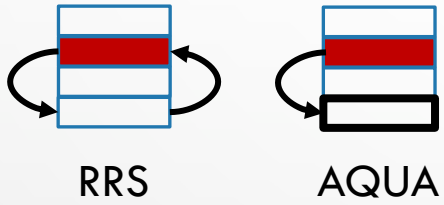
$$Row\text{-}Migrations \propto \frac{Row\text{—}Migrations}{Mitigation}$$



RRS AQUA

ROW MIGRATIONS

$$Row\text{-}Migrations \propto \frac{Row-Migrations}{Mitigation} \times \frac{Mitigations}{Rows-to-mitigate}$$

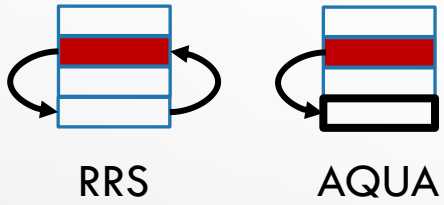


$$T_{RRS} \leftarrow \frac{T_{RH}}{6}$$

$$T_{AQUA} \leftarrow \frac{T_{RH}}{2}$$

ROW MIGRATIONS

$$Row\text{-}Migrations \propto \frac{Row-Migrations}{Mitigation} \times \frac{Mitigations}{Rows-to-mitigate} \times Rows-to-mitigate$$



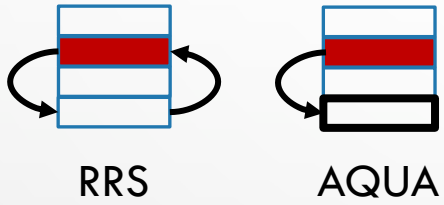
$$T_{RRS} \leftarrow \frac{T_{RH}}{6}$$

$$T_{AQUA} \leftarrow \frac{T_{RH}}{2}$$

Workload	Rows with ACT-166+	Rows with ACT-500+
Average	1,665 (2.4X)	694

ROW MIGRATIONS

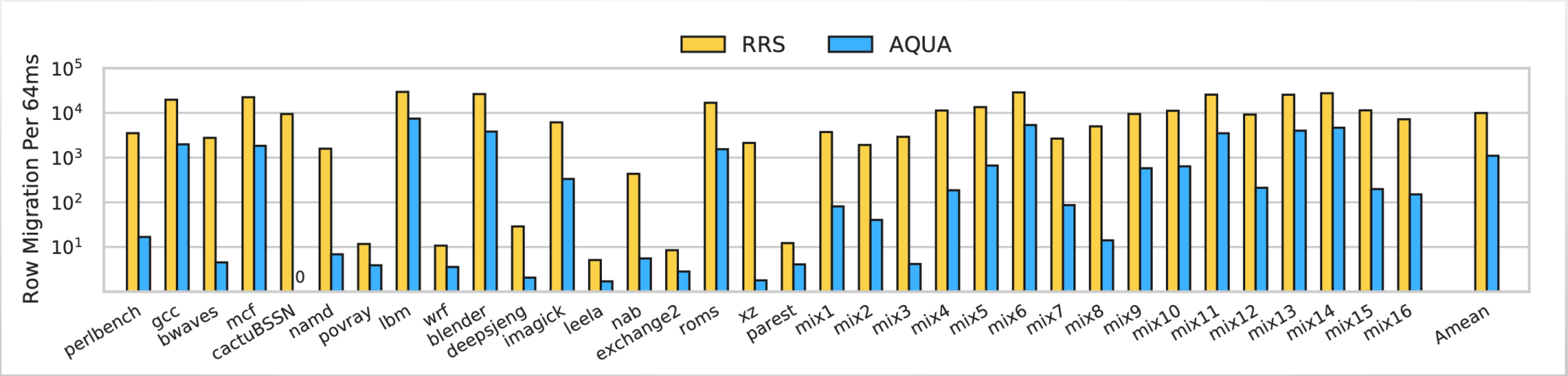
$$Row\text{-}Migrations \propto \frac{Row-Migrations}{Mitigation} \times \frac{Mitigations}{Rows-to-mitigate} \times Rows-to-mitigate$$



$$T_{RRS} \leftarrow \frac{T_{RH}}{6}$$

$$T_{AQUA} \leftarrow \frac{T_{RH}}{2}$$

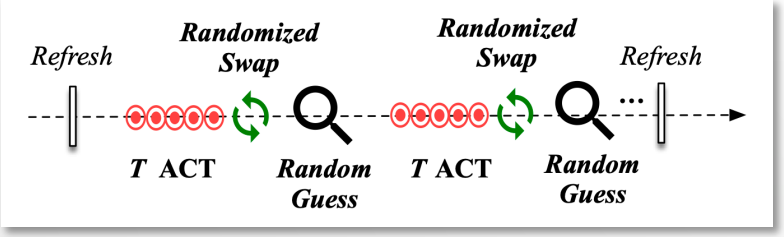
Workload	Rows with ACT-166+	Rows with ACT-500+
Average	1,665 (2.4X)	694



Row migrations reduce by 9X as less mitigations are performed

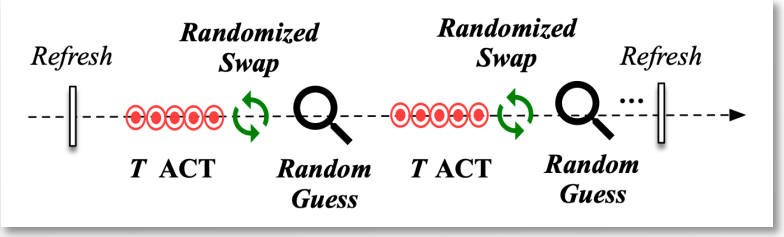
LIMITATIONS OF RANDOMIZED SWAPS

Security via Randomization



LIMITATIONS OF RANDOMIZED SWAPS

Security via Randomization



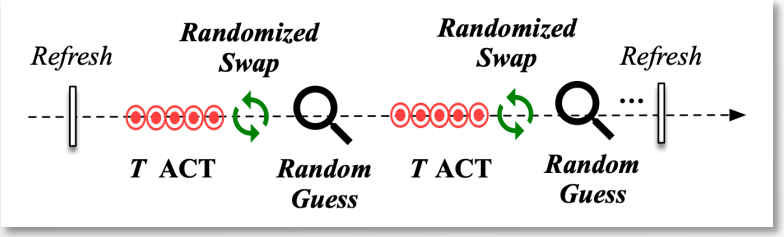
$$T \leftarrow \frac{TRH}{6}$$

Current TRH

$$666 \leftarrow 4K$$

LIMITATIONS OF RANDOMIZED SWAPS

Security via Randomization



$$T \leftarrow \frac{TRH}{6}$$

Current TRH

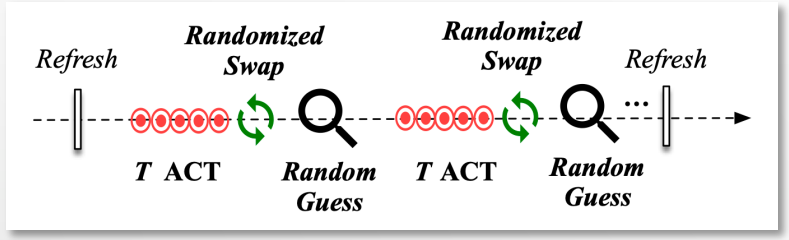
$$666 \leftarrow 4K$$

$$166 \leftarrow 1K$$

Our default TRH

LIMITATIONS OF RANDOMIZED SWAPS

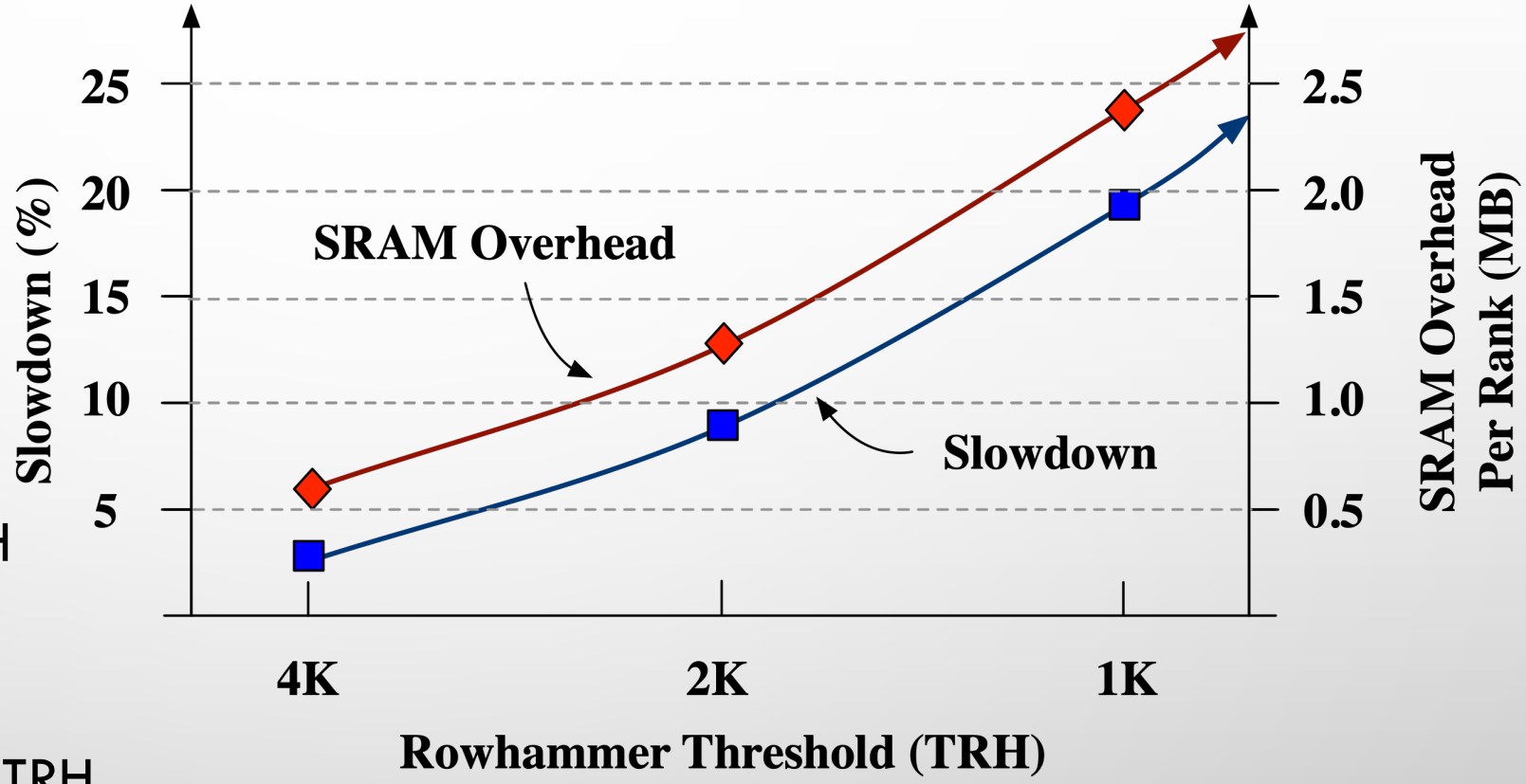
Security via Randomization



$$T \leftarrow \frac{TRH}{6}$$

$666 \leftarrow 4K$ Current TRH

$166 \leftarrow 1K$ Our default TRH



Our goal is to develop a scalable Rowhammer mitigation with low overheads

SPEC2017 ROW ACTIVATION CHARACTERISTICS

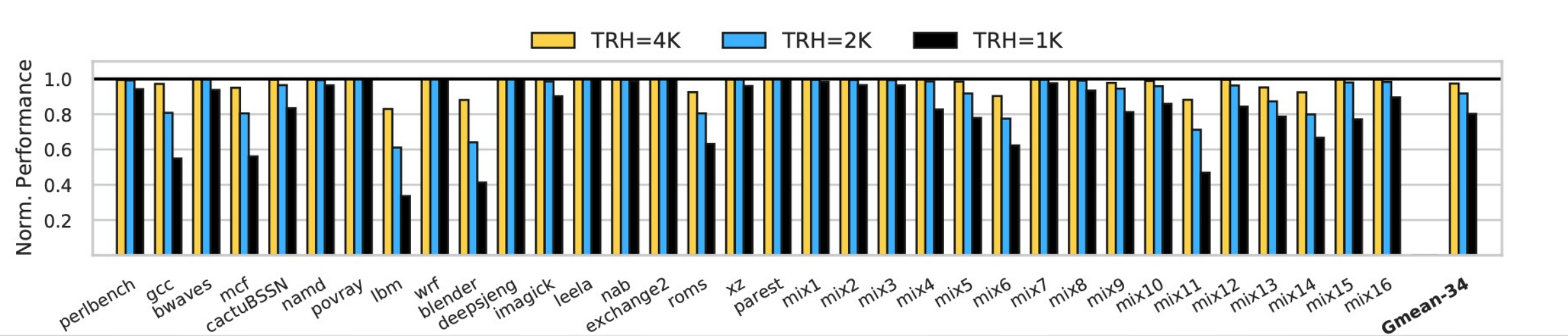
Workload	MPKI	Rows ACT-166+	Rows ACT-500+	Rows ACT-1K+
lbm	20.9	6794	5437	0
blender	14.8	6085	3021	572
gcc	6.32	4850	1836	111
mcf	7.02	4819	835	393
cactuBSSN	2.57	2515	0	0
roms	4.37	1150	191	11
xz	0.41	655	0	0
perlbench	0.74	0	0	0
bwaves	0.21	0	0	0
namd	0.38	0	0	0
povray	0.01	0	0	0
wrf	0.02	0	0	0
deepsjeng	0.25	0	0	0
imagick	0.27	0	0	0
leela	0.03	0	0	0
nab	0.54	0	0	0
exchange2	0.01	0	0	0
parest	0.1	0	0	0
Average	3.5	1665	694	57

EXPERIMENTAL SETUP

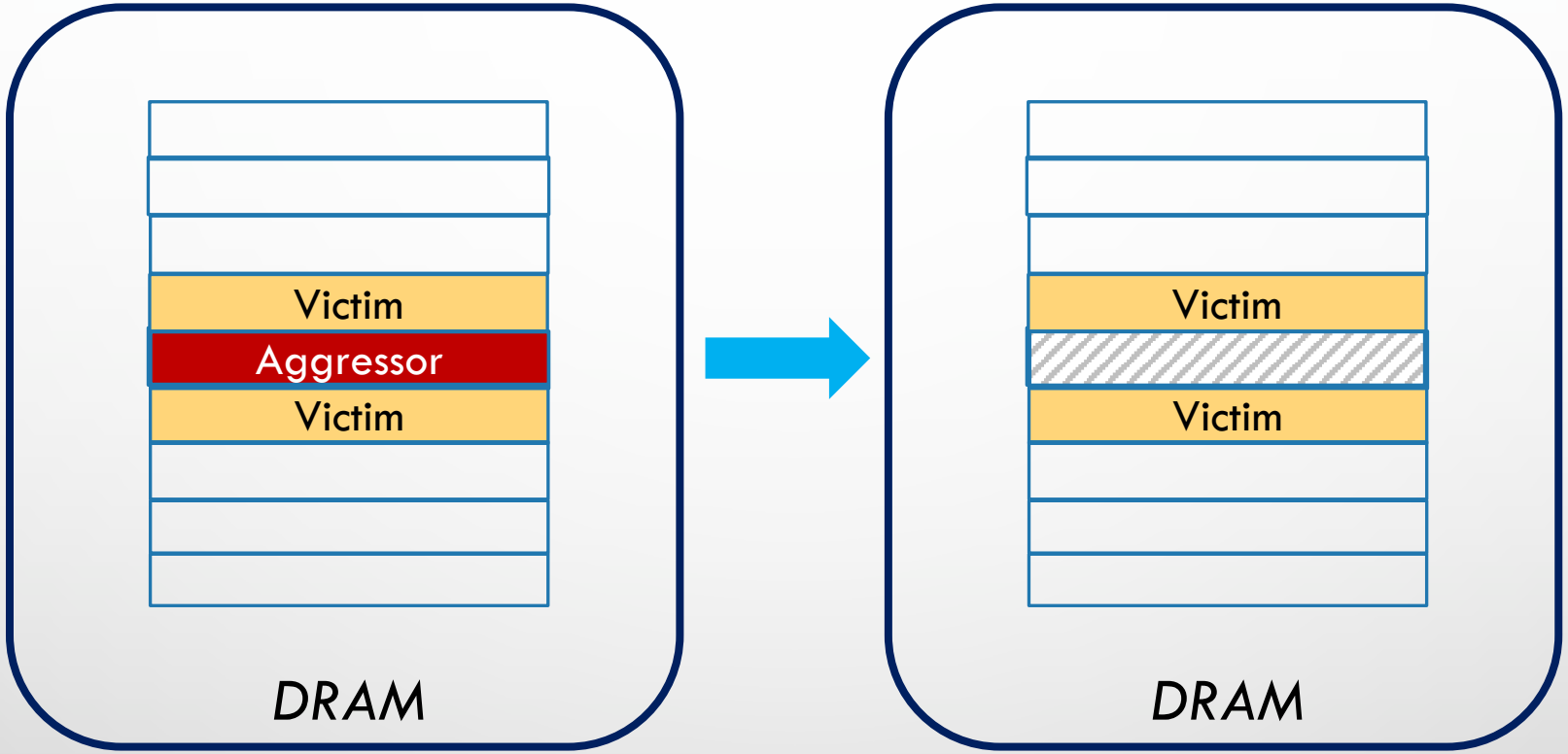
Out-of-Order Cores	4 cores at 3GHz
ROB size	192
Fetch and Retire width	8
Last Level Cache (Shared)	4MB, 16-Way, 64B lines
Memory size	16 GB – DDR4
Memory bus speed	1.2 GHz (2400 MT/s)
t_{RCD}-t_{CL}-t_{RP}-t_{RC}	14.2-14.2-14.2-45 ns
t_{CCD_S}, t_{CCD_L}	3.3 ns, 5 ns
Banks x Ranks x Channels	16×1×1
Rows per bank	128K
Size of row	8KB

- 18 SPEC2017 rate and 16 mixed workloads
- Fast-forward by 25 Bn instructions
- Simulate for 250 Mn instructions

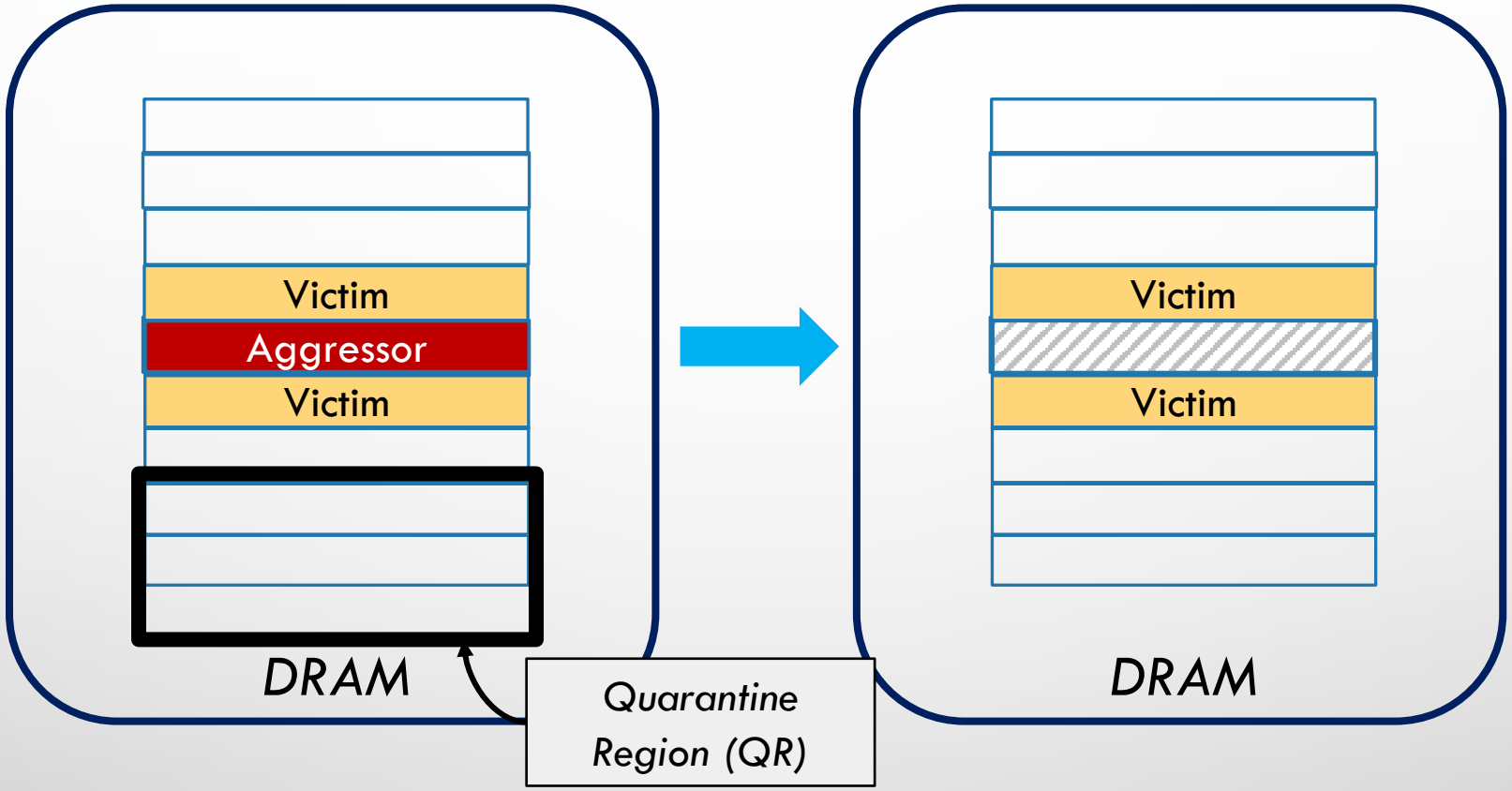
PERFORMANCE DEGRADATION OF RRS



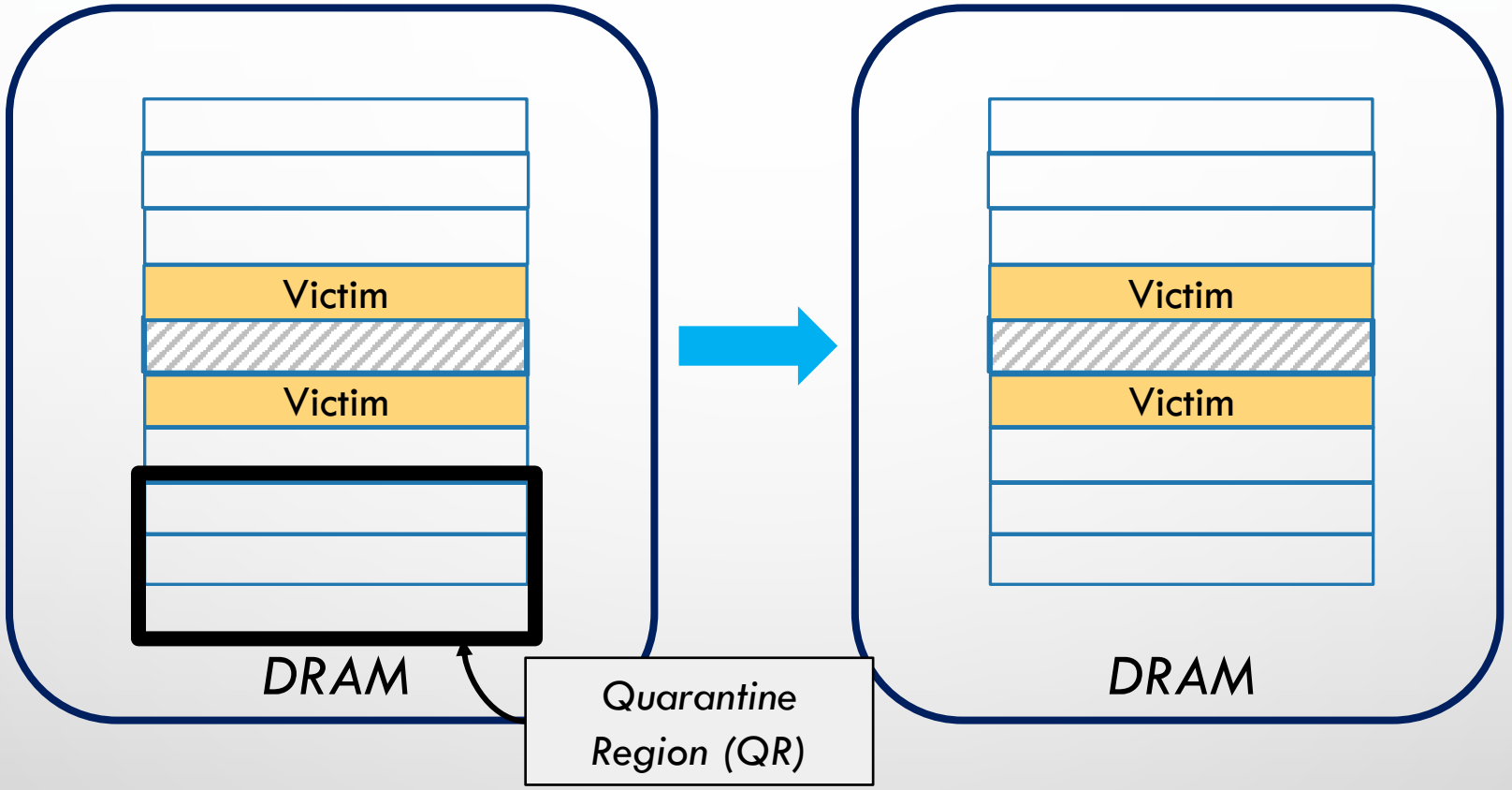
AQUA: QUARANTINING AGGRESSORS



AQUA: QUARANTINING AGGRESSORS

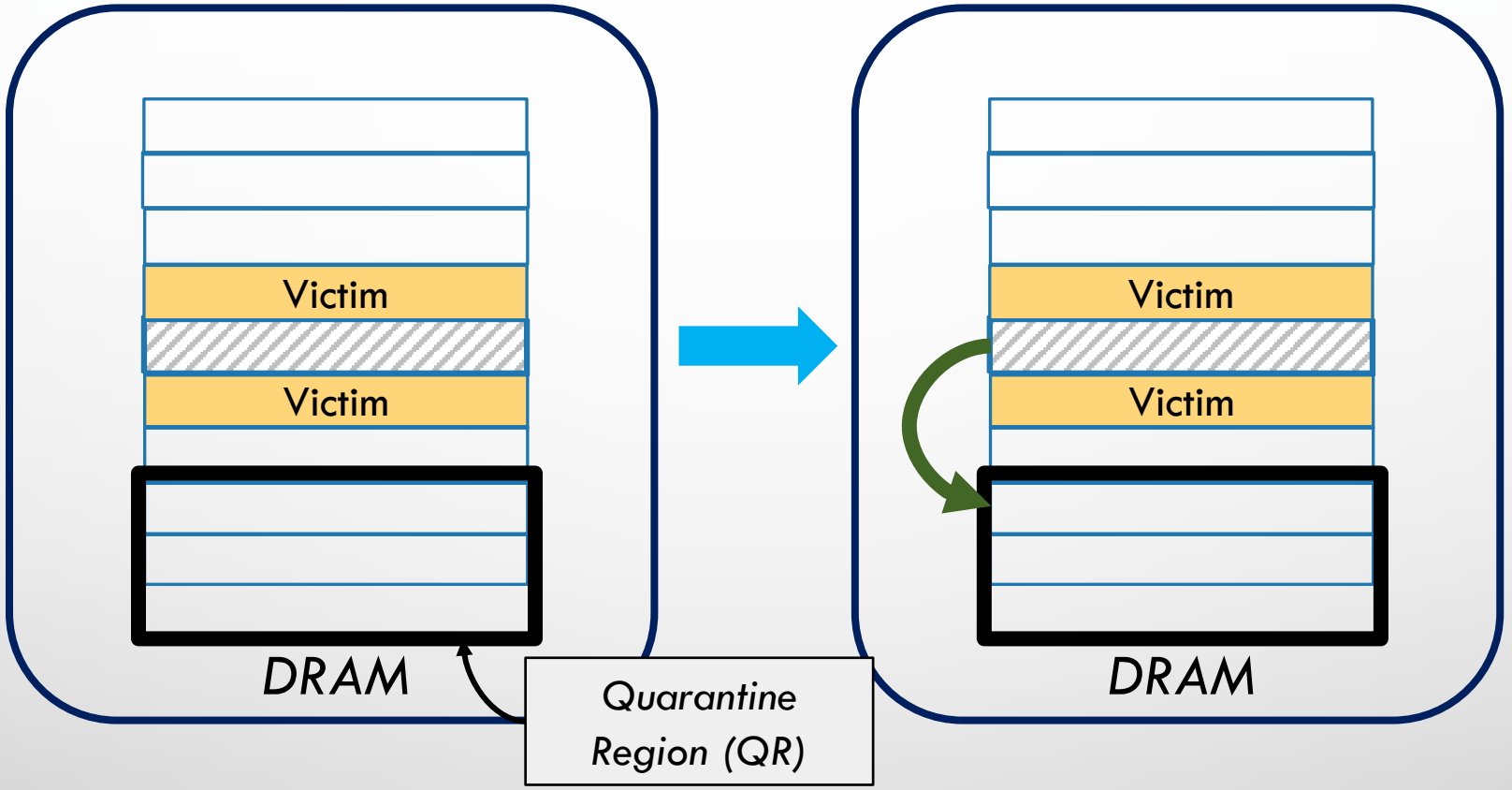


AQUA: QUARANTINING AGGRESSORS



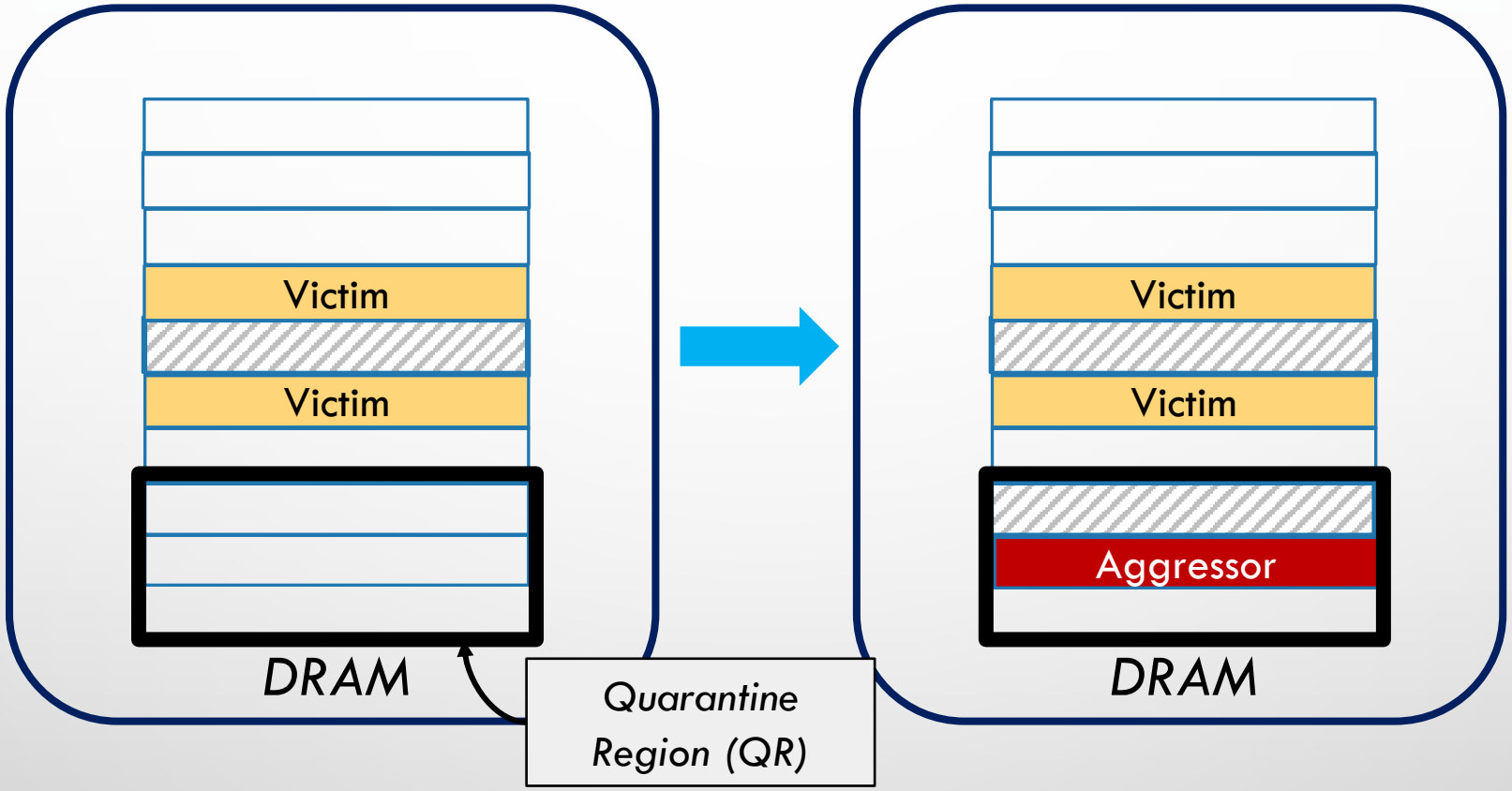
Security via Isolation: AQUA moves rows to a dedicated memory region

AQUA: QUARANTINING AGGRESSORS



Security via Isolation: AQUA moves rows to a dedicated memory region

AQUA: QUARANTINING AGGRESSORS



Security via Isolation: AQUA moves rows to a dedicated memory region