MIRAGE: Mitigating Conflict-Based Cache Attacks with a Practical Fully-Associative Design

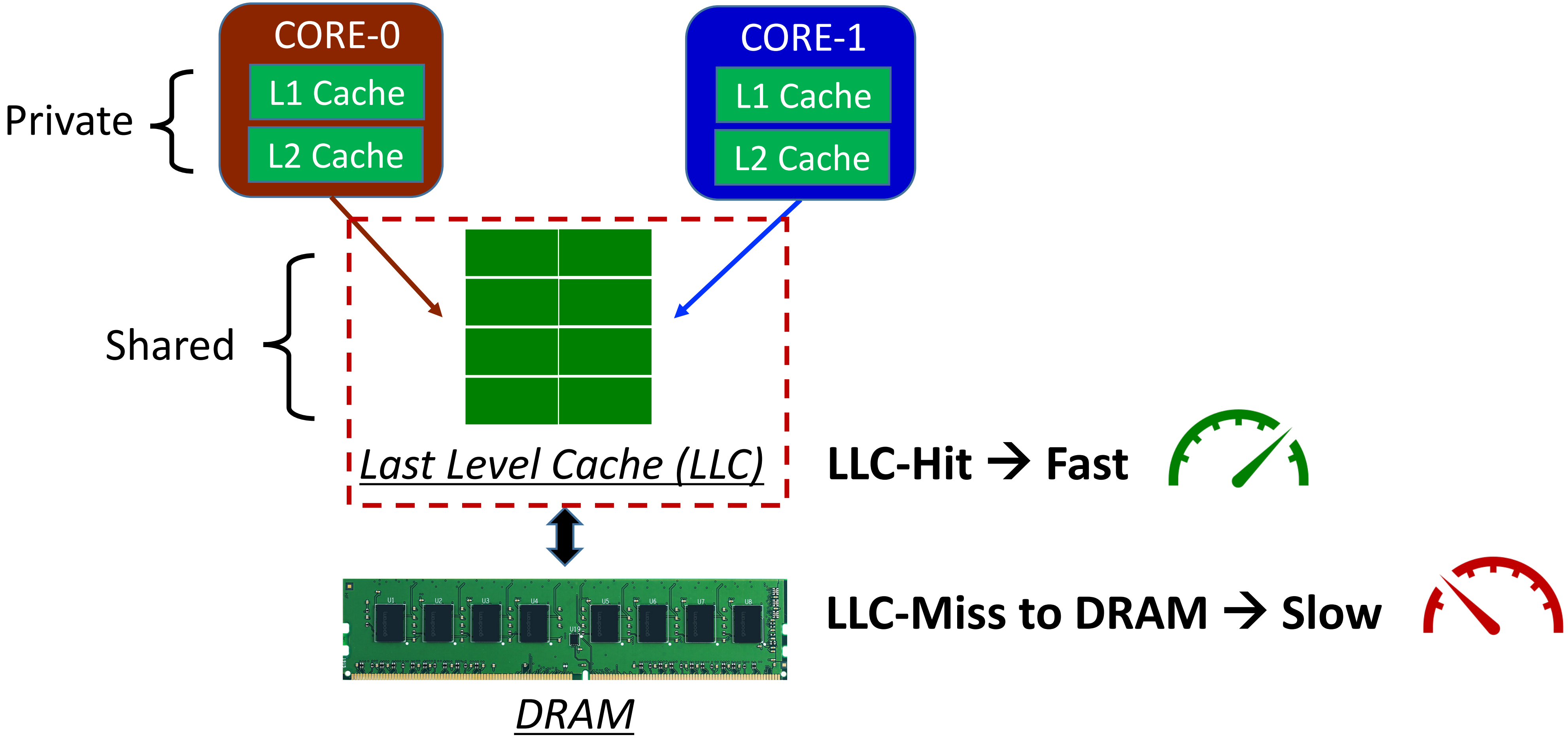
A design to eliminate eviction-sets & cache attacks

USENIX Security 2021

Gururaj Saileshwar and **Moinuddin Qureshi**

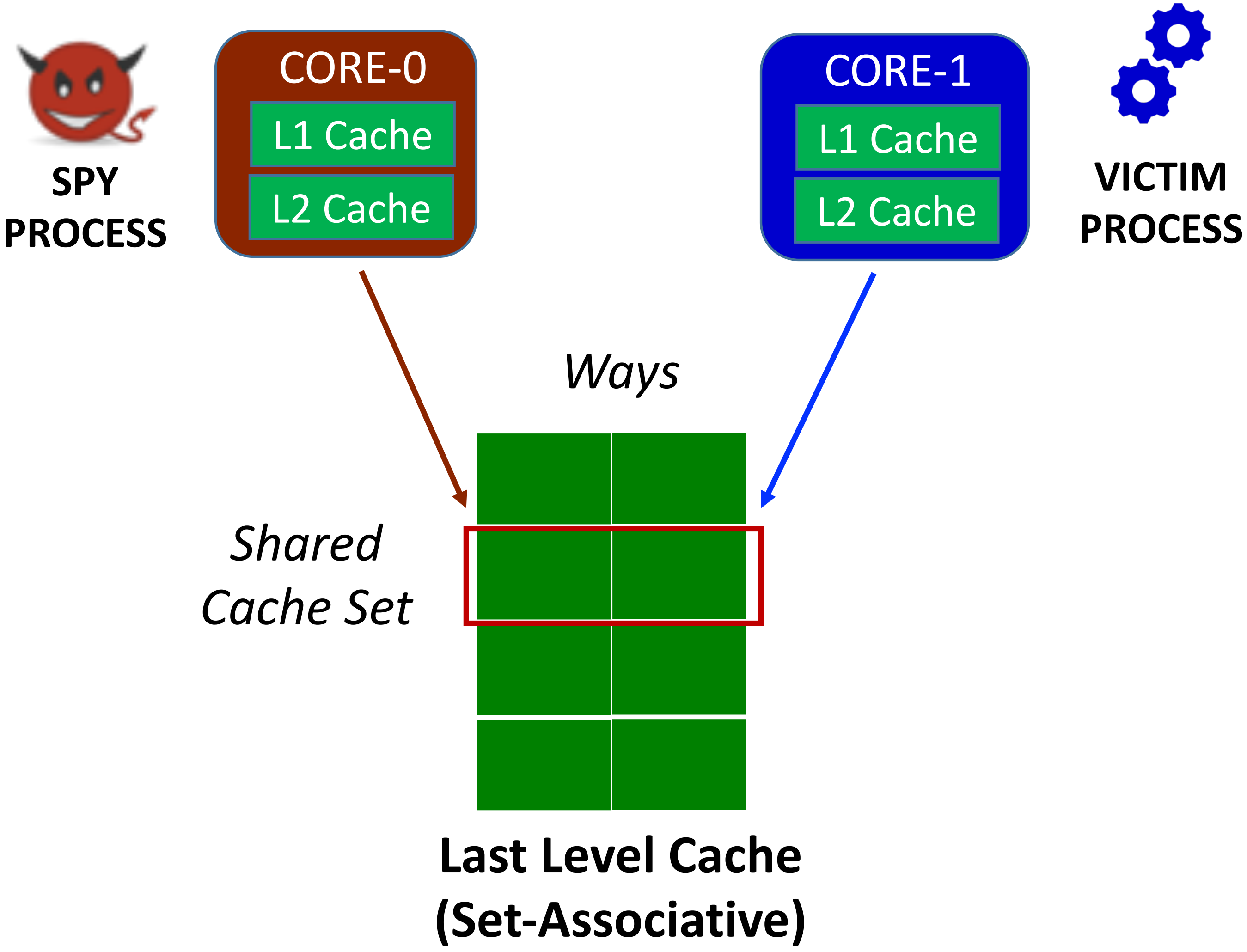
Georgia Tech

Problem: CPU Cache Side-Channels

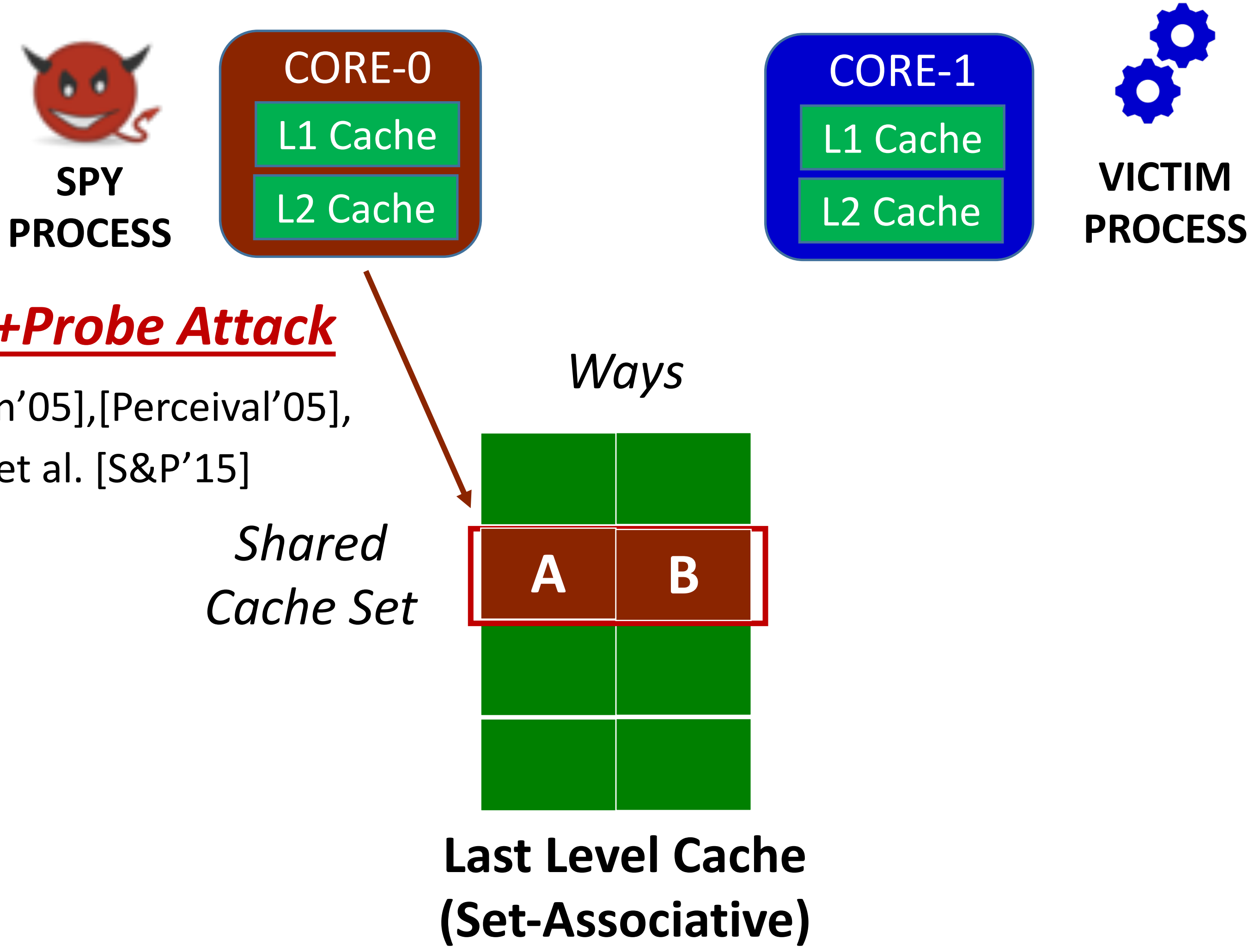


Shared LLC helps improve performance,
But timing difference (LLC Hit vs Miss) leads to side-channels!

Problem: CPU Cache Side-Channels



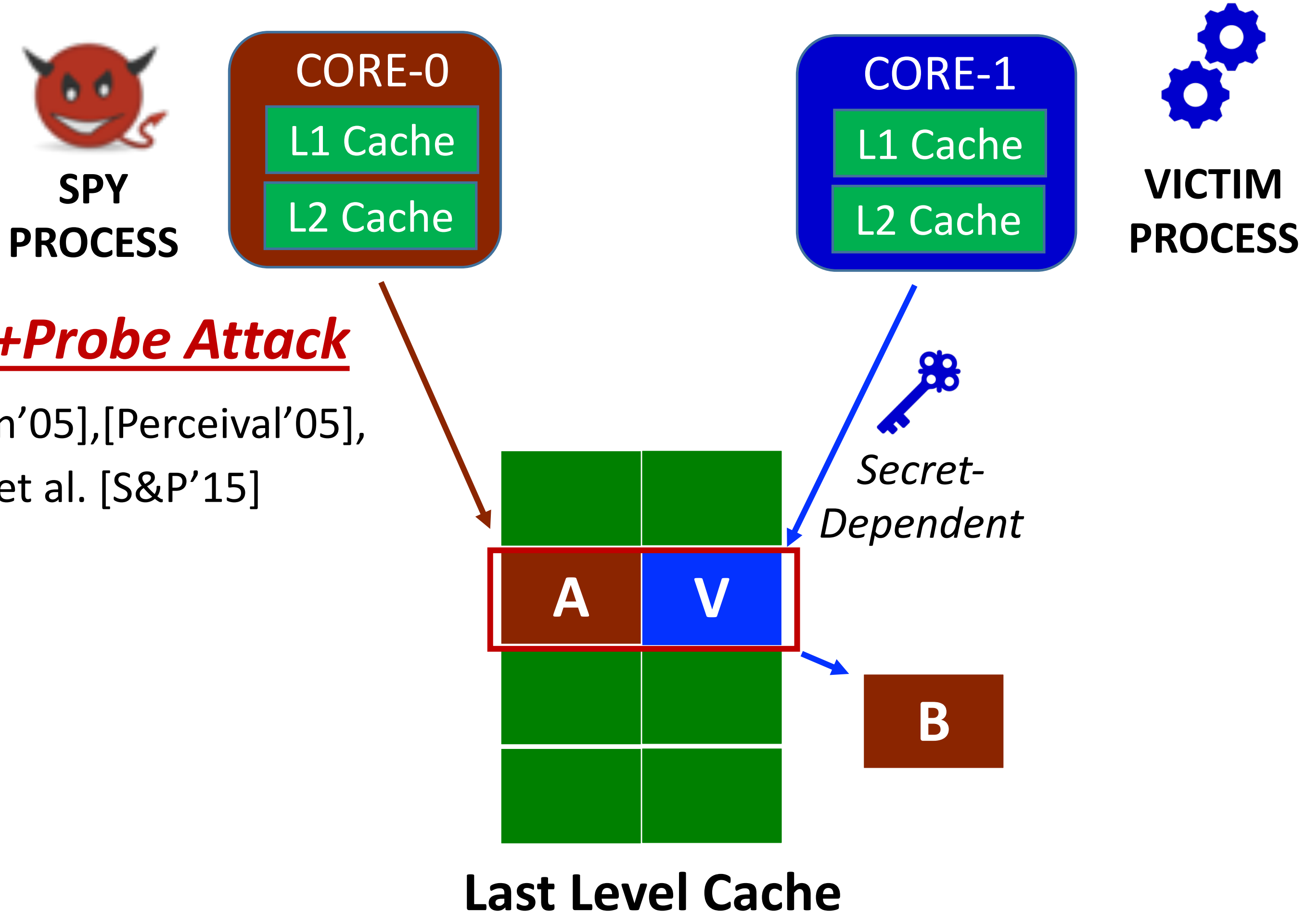
Problem: CPU Cache Side-Channels



Prime+Probe Attack

[Bernstein'05],[Perceival'05],
Liu et al. [S&P'15]

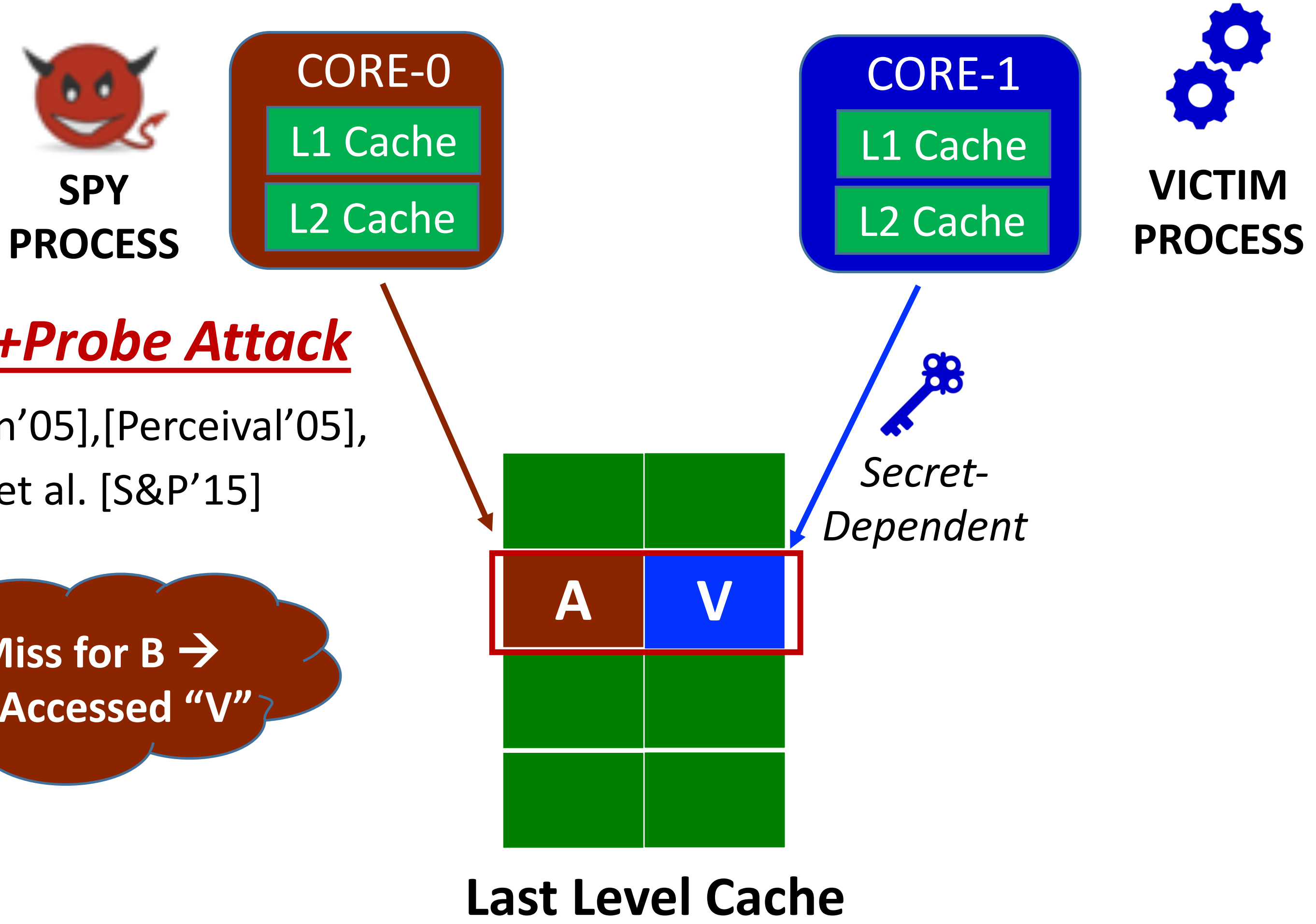
Problem: CPU Cache Side-Channels



Prime+Probe Attack

[Bernstein'05],[Perceival'05], Liu et al. [S&P'15]

Problem: CPU Cache Side-Channels



Prime+Probe Attack

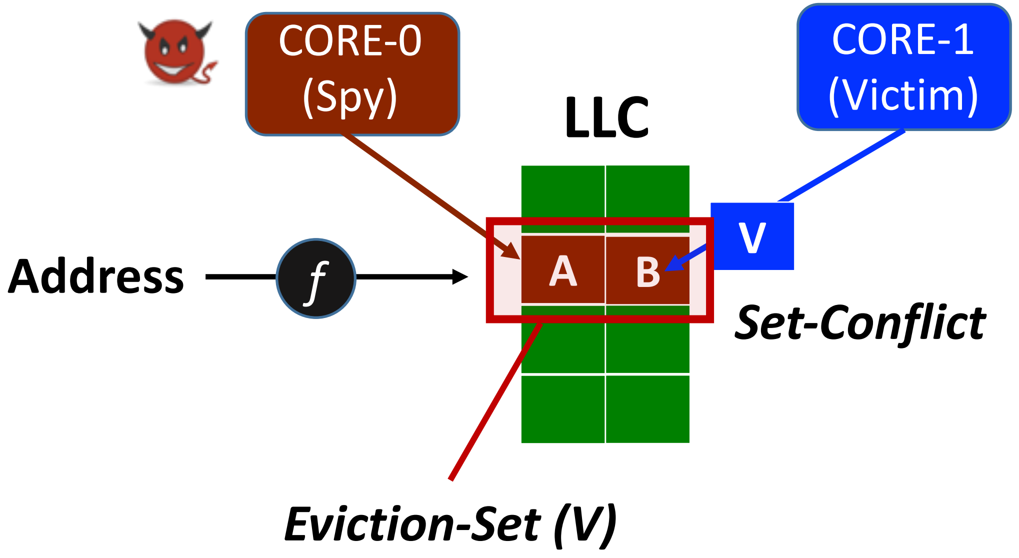
[Bernstein'05],[Perceival'05], Liu et al. [S&P'15]

LLC-Miss for B → Victim Accessed "V"

Spy can Leak Victim Secrets like AES/RSA Keys, User Key-Strokes, etc.

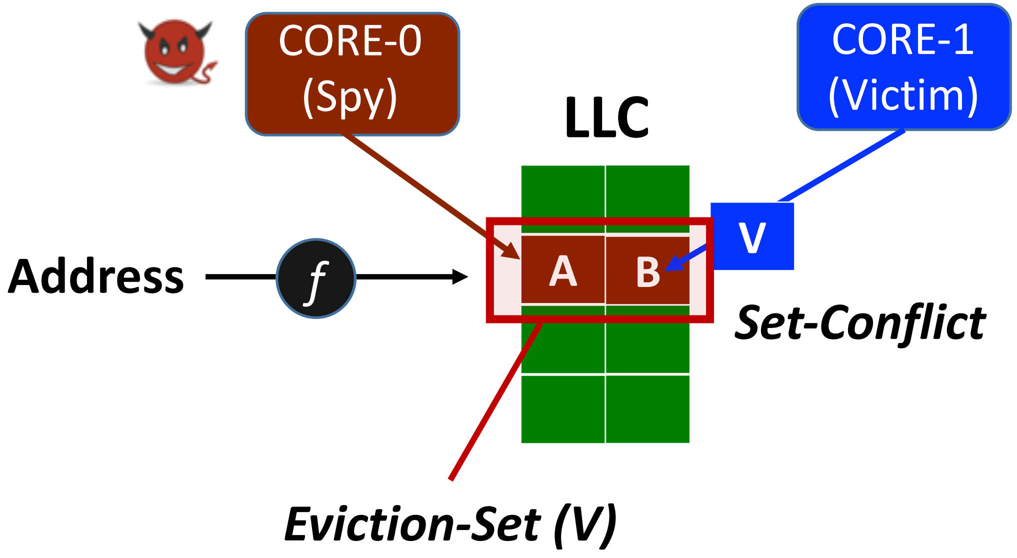
Randomized Cache Defenses

Prime+Probe Attack



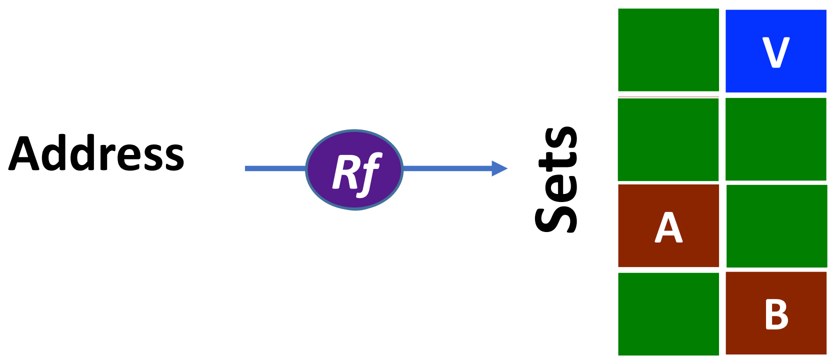
Randomized Cache Defenses

Prime+Probe Attack



Randomized Cache Defenses

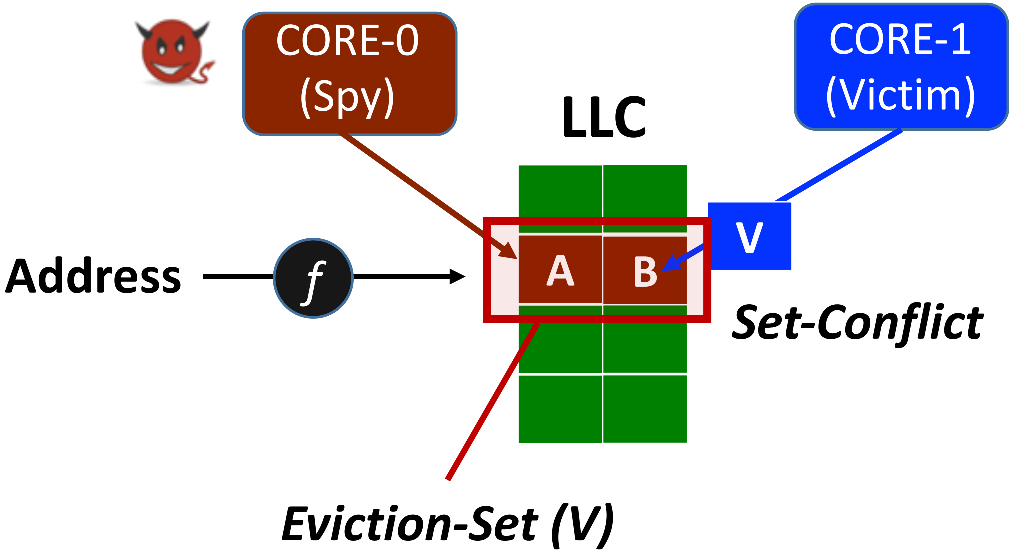
[MICRO'18], [ISCA'19], [SEC'19], [NDSS'20], [S&P'21]



Randomized Set-Indexing Obfuscates Set-Conflicts ➡ *Eviction-Sets harder to discover*

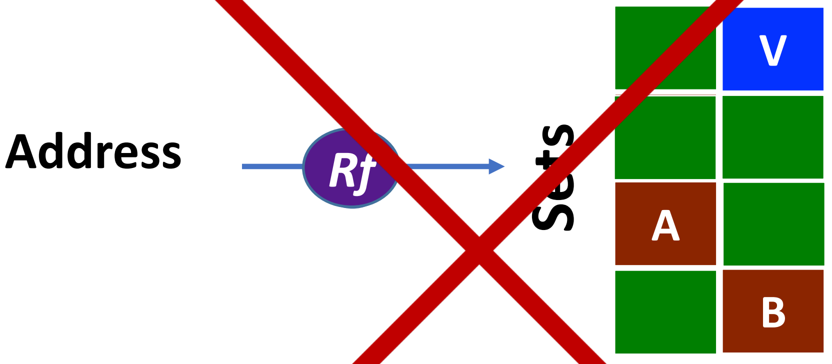
Randomized Cache Defenses

Prime+Probe Attack



Randomized Cache Defenses

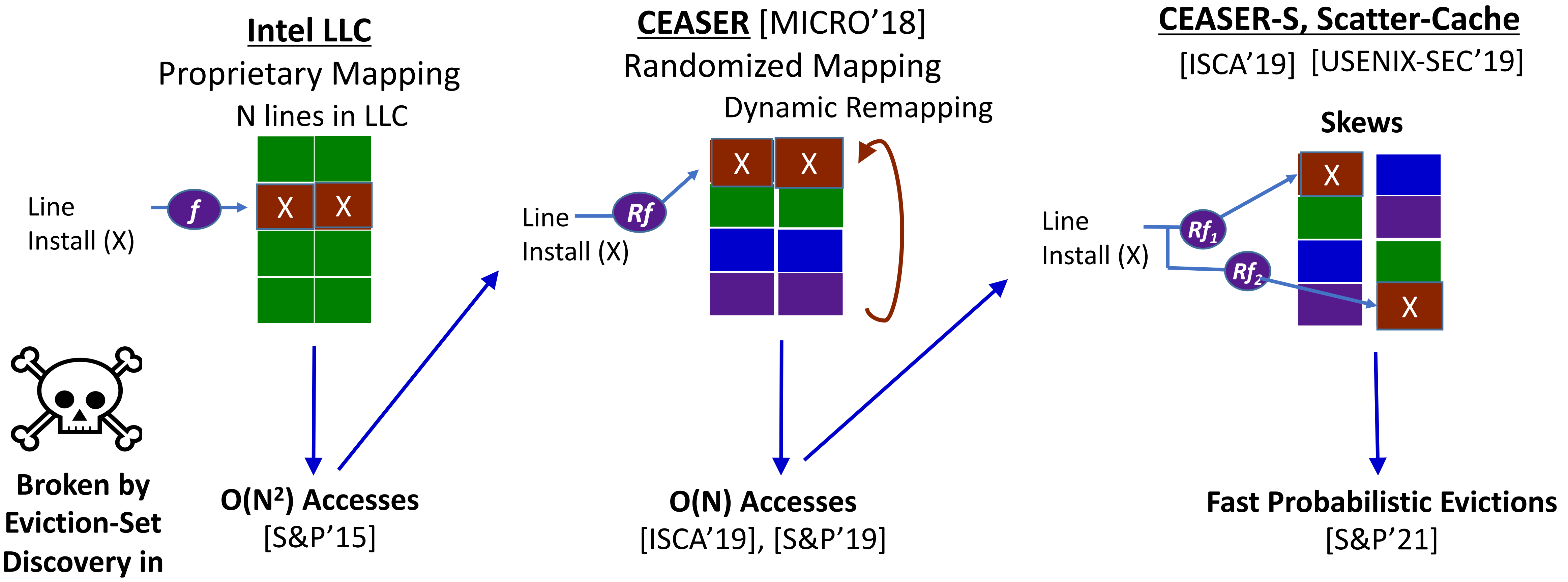
[MICRO'18], [ISCA'19], [SEC'19], [NDSS'20], [S&P'21]



Randomized Set-Indexing Obfuscates Set-Conflicts ➡ *Eviction-Sets harder to discover*

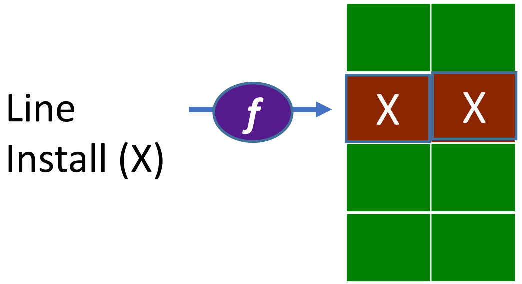
Successive Defenses Broken By Newer Attacks

Arms Race Between Attacks & Defenses

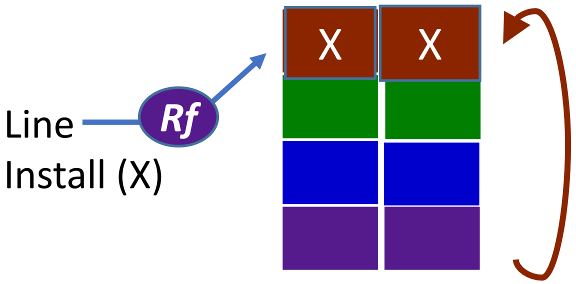


Arms Race Between Attacks & Defenses

Intel LLC
Proprietary Mapping
N lines in LLC

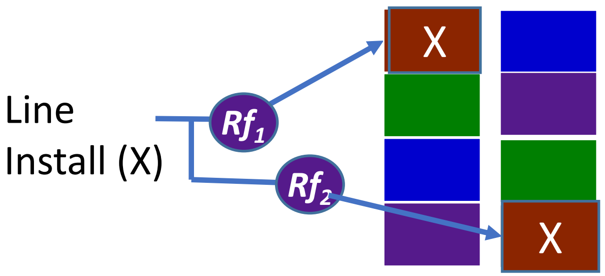


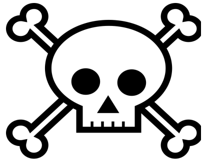
CEASER [MICRO'18]
Randomized Mapping
Dynamic Remapping



CEASER-S, Scatter-Cache
[ISCA'19] [USENIX-SEC'19]

Skews





Pitfall: Set-Conflicts Continue at Few Obfuscated Locations

Broken by Eviction-Set Discovery in

$O(N^2)$ Accesses
[S&P'15]

O(N) Accesses
[ISCA'19], [S&P'19]

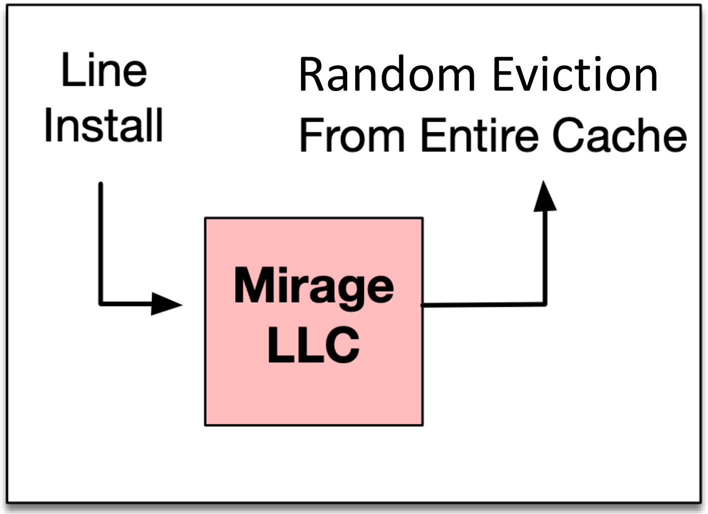
Fast Probabilistic Evictions
[S&P'21]

<u>Our Goal:</u> Eliminate Set-Conflicts to End the Arms Race

Goal: Fully-associative Randomized LLC

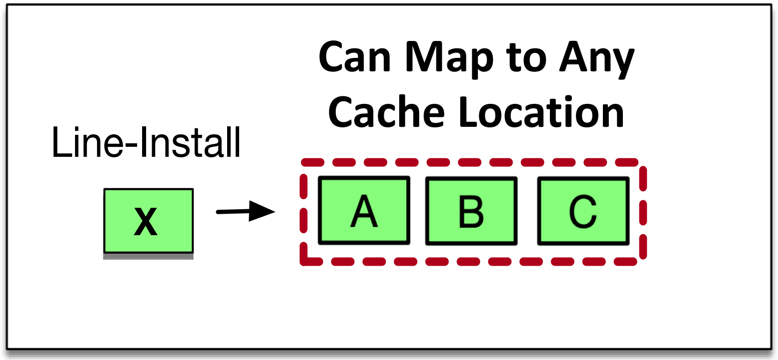
Abstraction to SW

Fully-Associative: No Set Conflicts



Principled Security

Challenge: Naive Fully-Associative Lookup Requires Checking 100,000+ LLC Locations

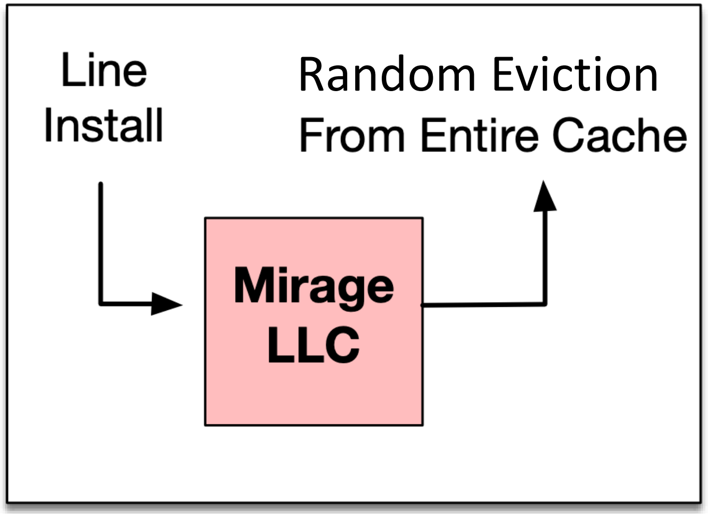


Impractical Lookup Latency & Power

Goal: Fully-associative Randomized LLC

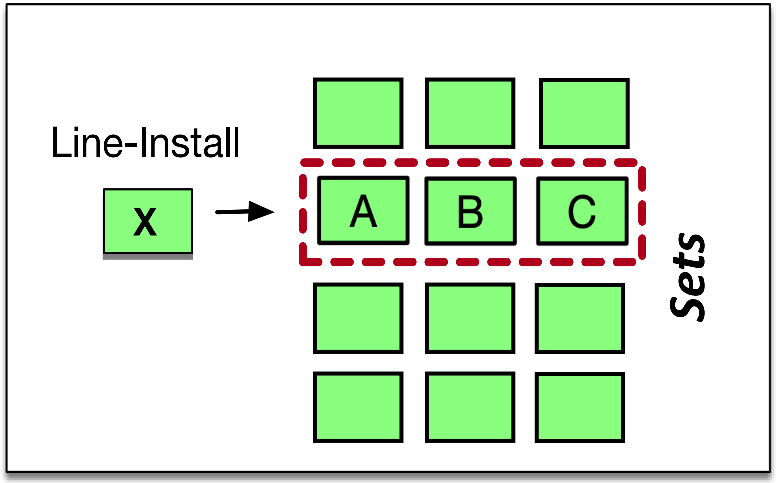
Abstraction to SW

Fully-Associative: No Set Conflicts



✔ **Principled Security**

Traditional Set-Associative Lookup

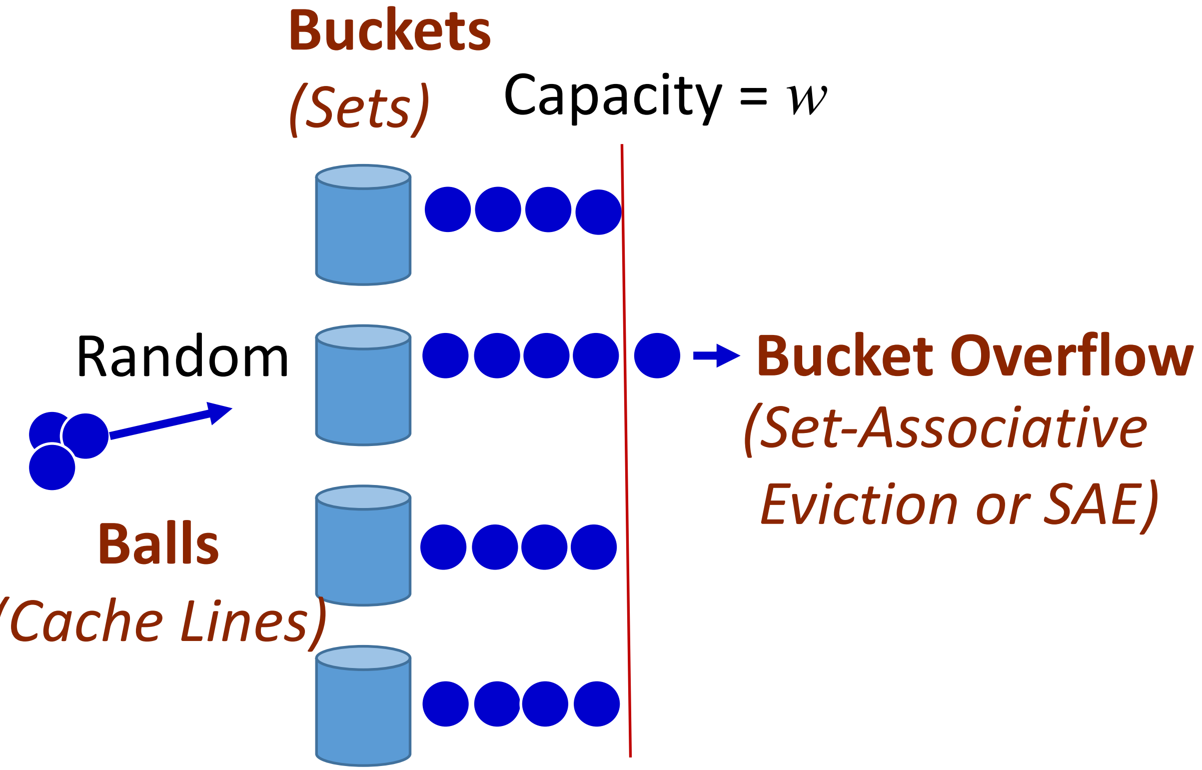


✔ **Practical Lookup within Set (16-32 Locations)**

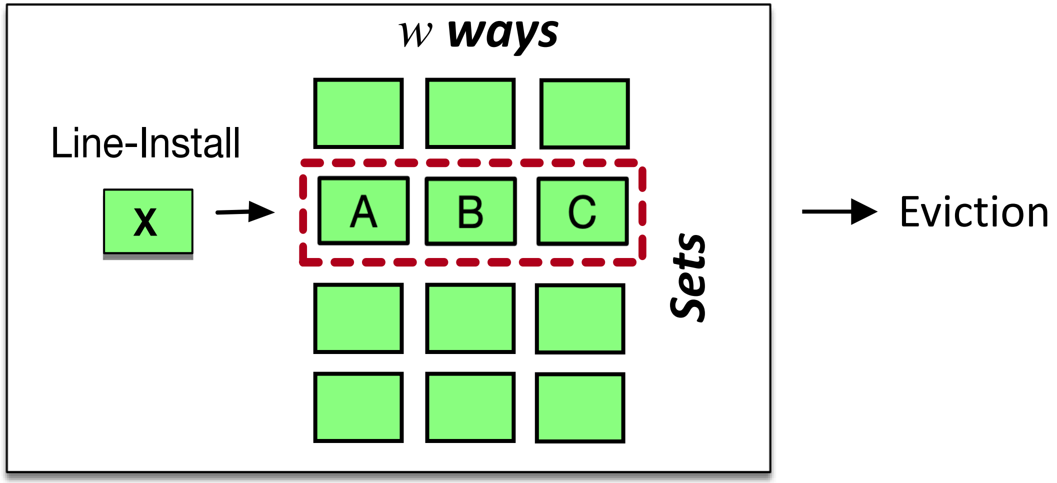
Key Challenge: How to get Security of Fully-Associative Design (No Set-Conflicts) with Set-Associative Lookups?

Insight: Use Load-Balancing to Eliminate Set-Conflicts

Buckets & Balls Problem

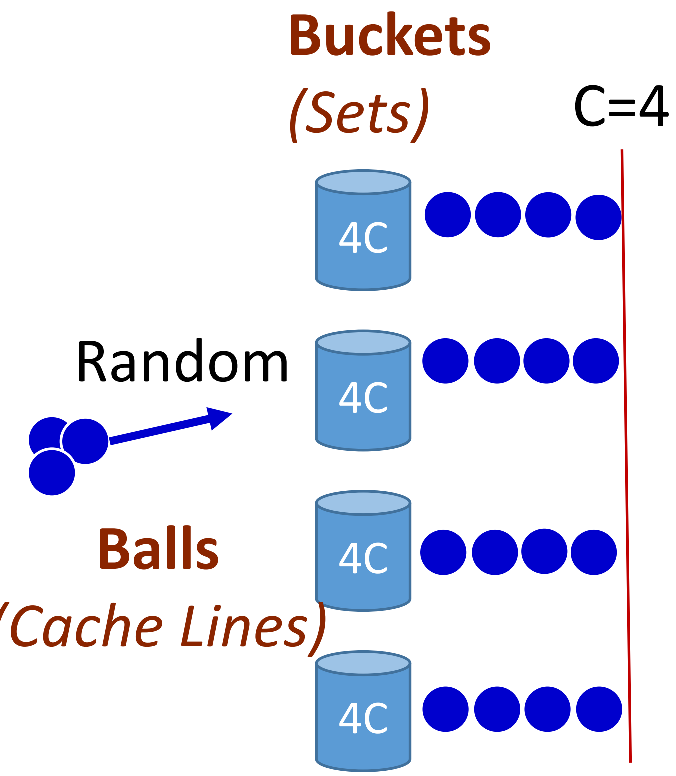


Set-Associative Randomized LLC



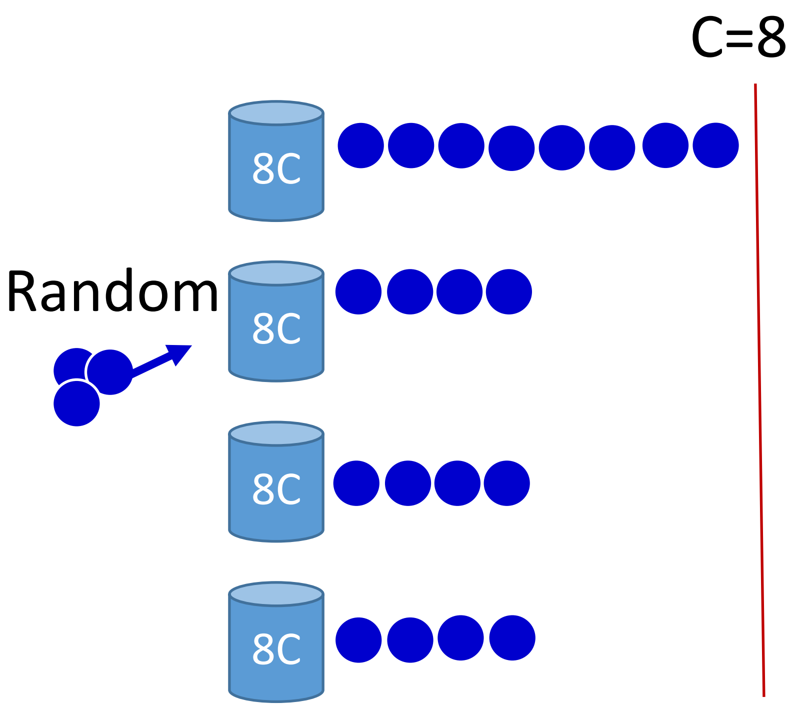
Insight: Use Load-Balancing to Eliminate Set-Conflicts

16 Balls in 4 Buckets (C=4)



Bucket Overflow Every Ball Throw

16 Balls in 4 Buckets (C=8)

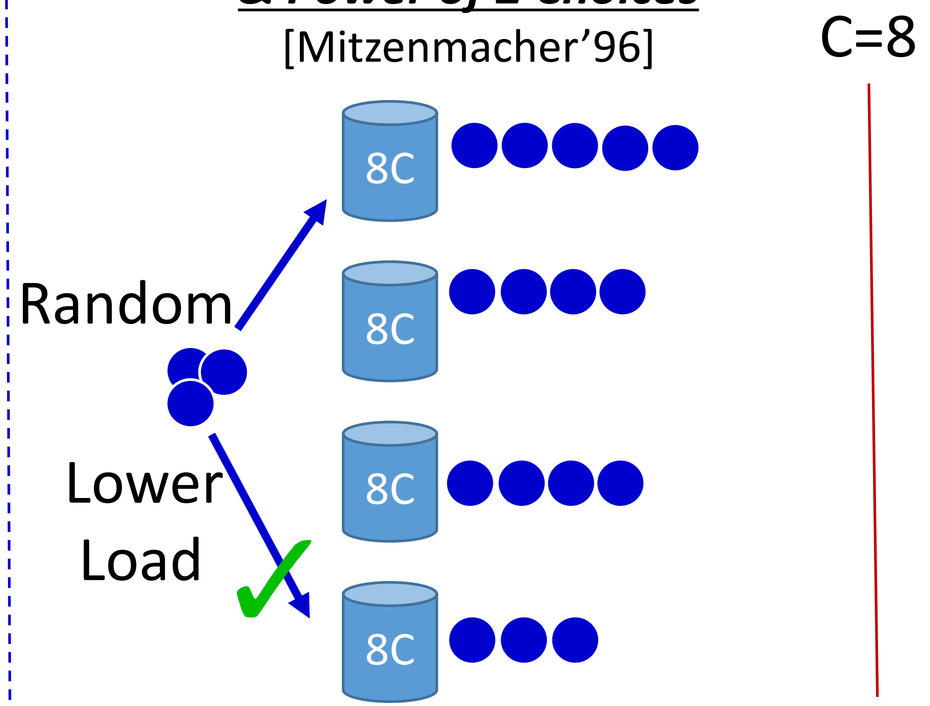


Bucket Overflow Likelihood Reduced, But Still Possible

16 Balls in 4 Buckets (C=8)

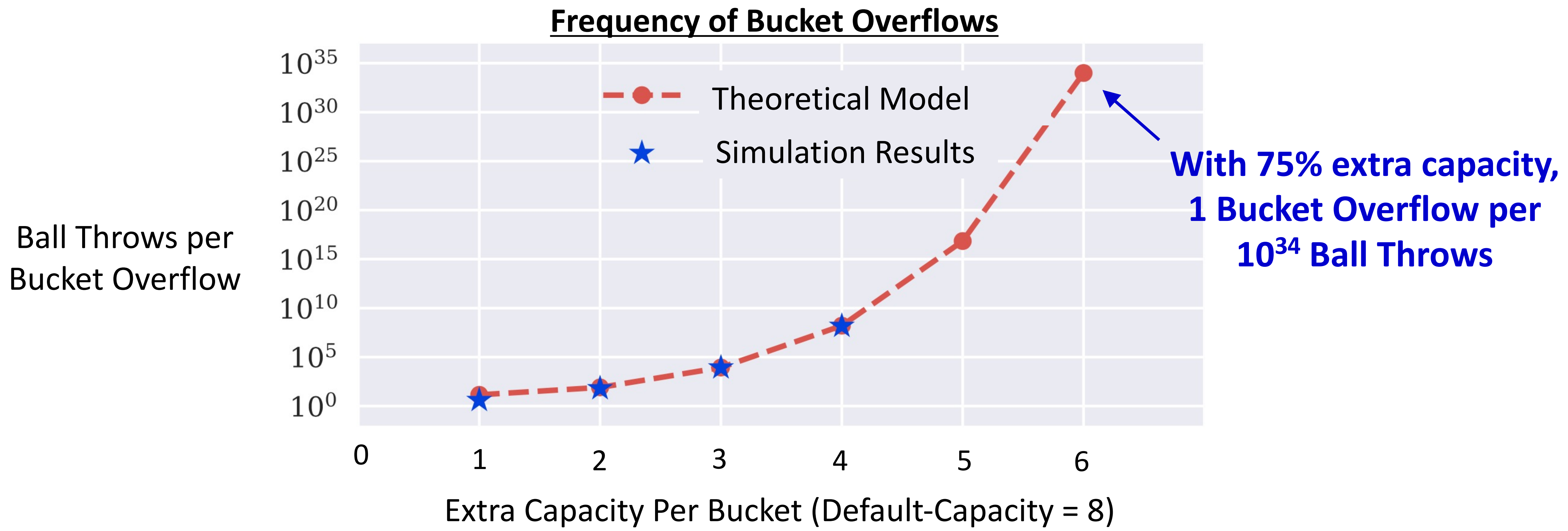
& Power of 2 Choices

[Mitzenmacher'96]



Bucket Overflow Improbable: Balanced Distribution

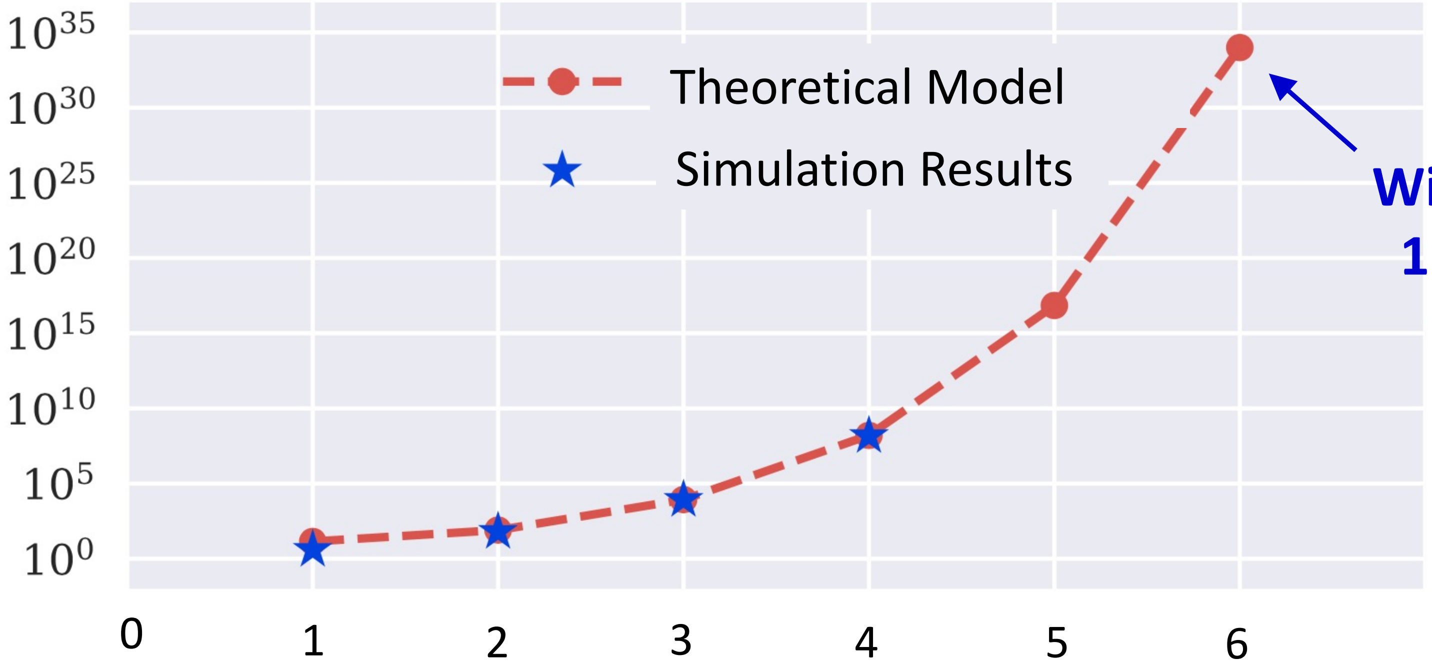
Security Guarantee With Power of 2 Choices



Security Guarantee With Power of 2 Choices

Set-Associative Eviction (SAE)

Frequency of ~~Bucket Overflows~~



Line Installs per SAE

~~Ball Throws per Bucket Overflow~~

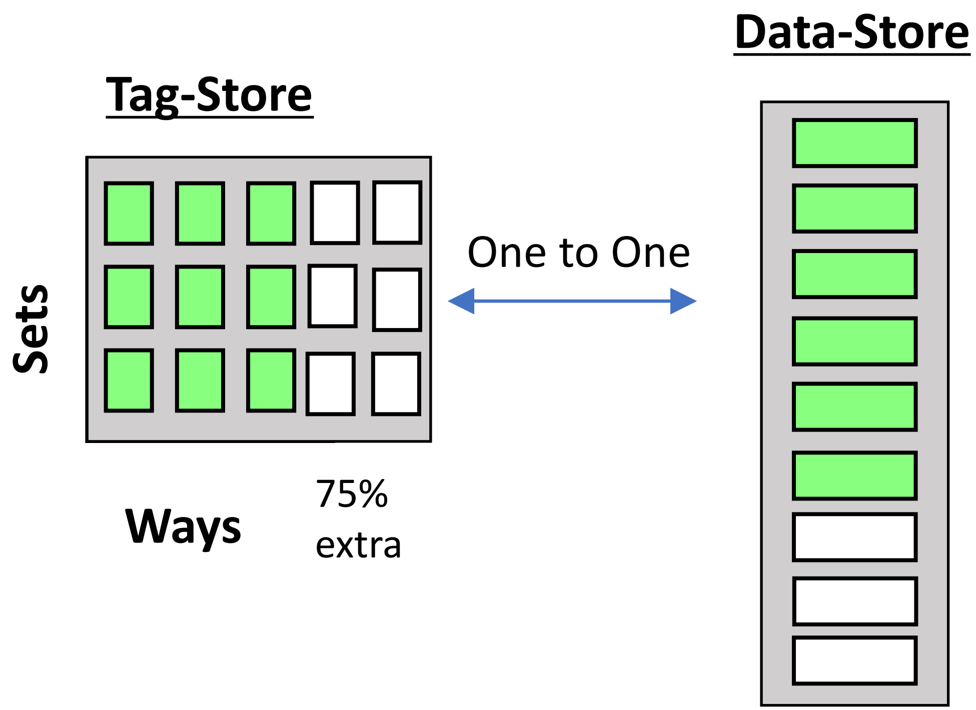
Extra Capacity Per ~~Bucket~~ Set (Default-Capacity = 8)

With 75% extra capacity, 1 Bucket Overflow per 10^{34} Ball Throws

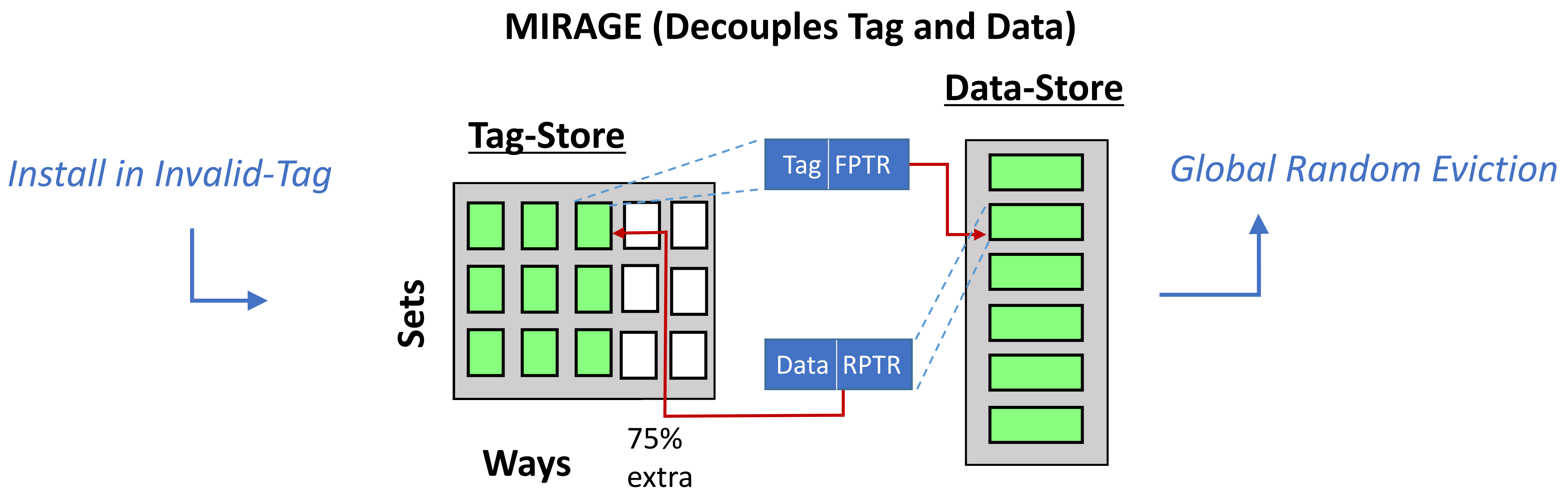
LLC with 75% extra capacity & Power of 2 Choices Indexing has Security Guarantee of 1 SAE Per 10^{34} LLC Installs (10^{17} years)

MIRAGE Design

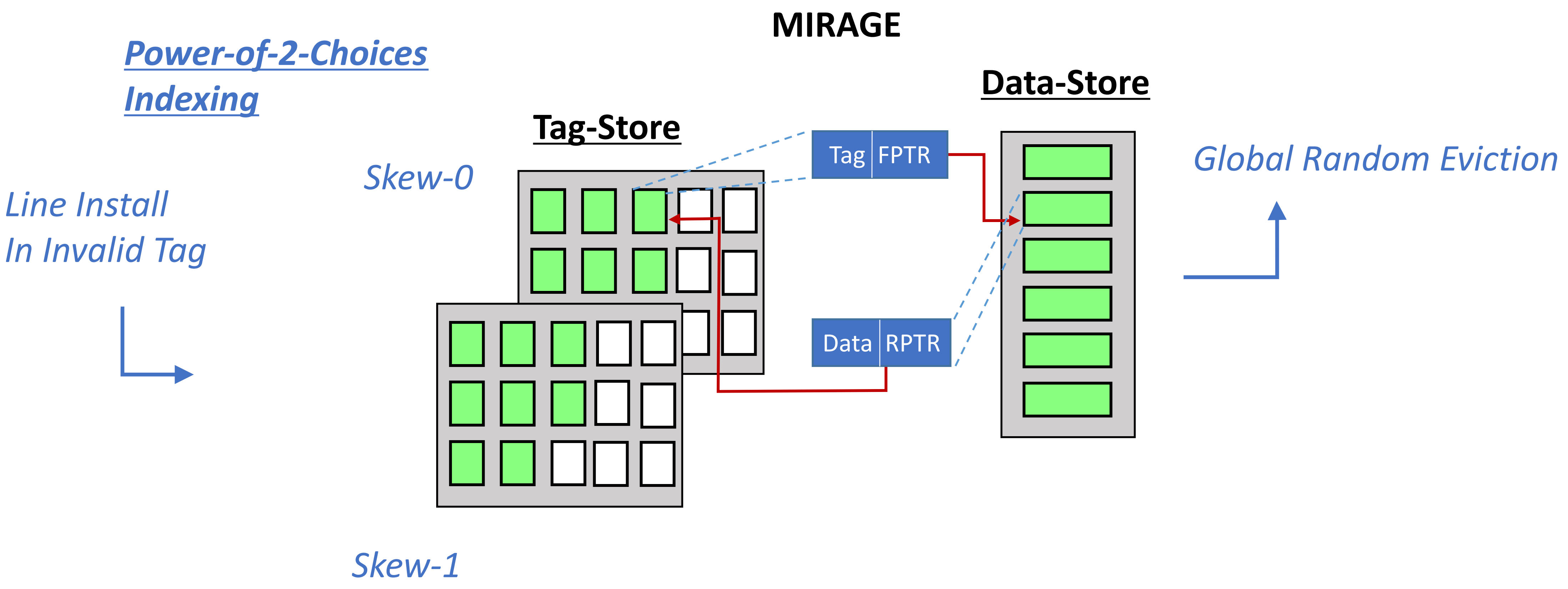
Extra Tags Cheap, Extra Data Expensive (1:10)



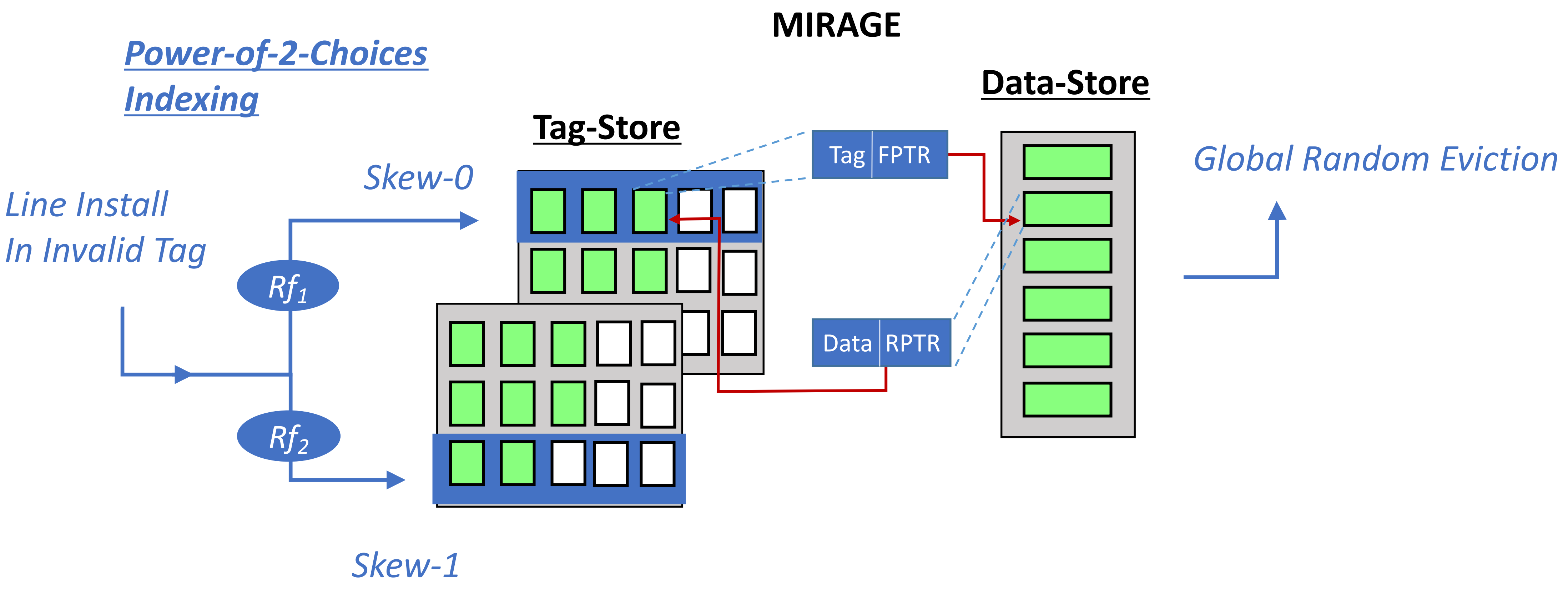
MIRAGE Design



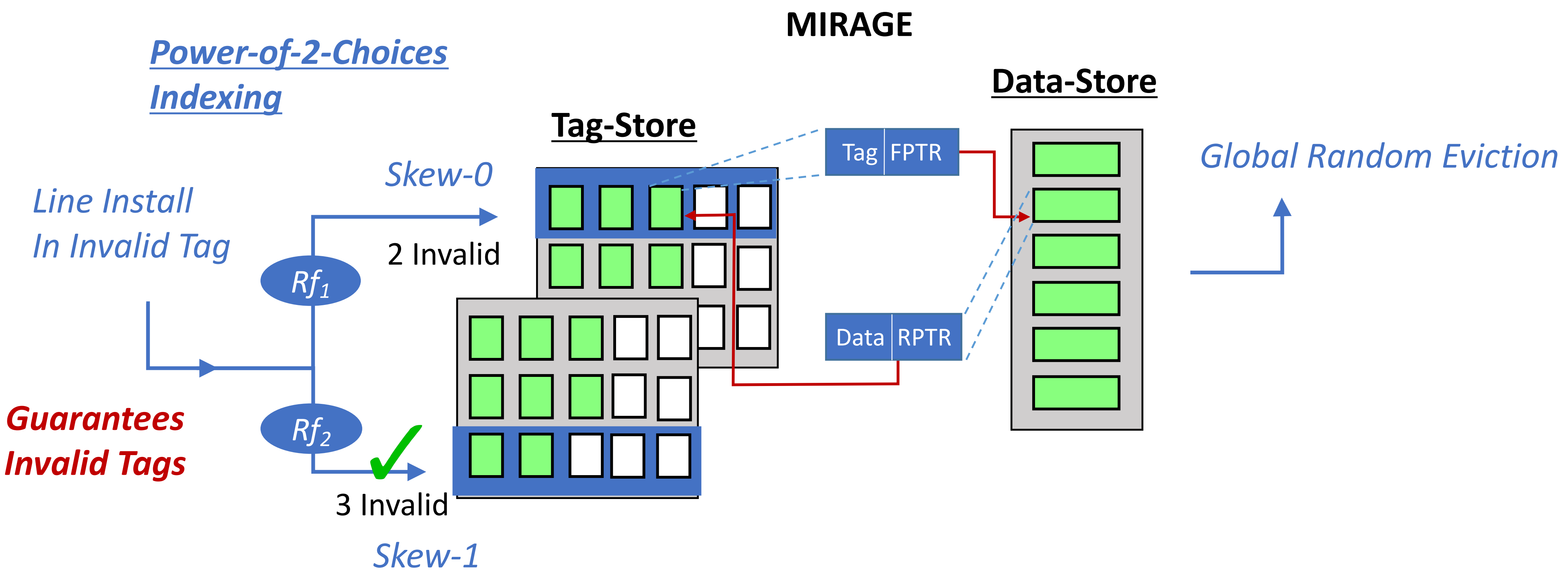
MIRAGE Design



MIRAGE Design



MIRAGE Design



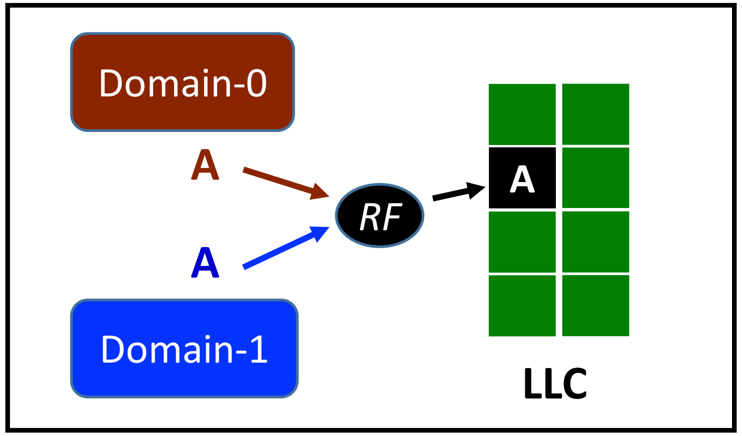
Security Guarantee: With 75% extra tags, MIRAGE ensures 1 Set-Associative Eviction that can leak information every 10^{34} LLC Installs (once in 10^{17} years)

Eliminates Conflict-Based Attacks

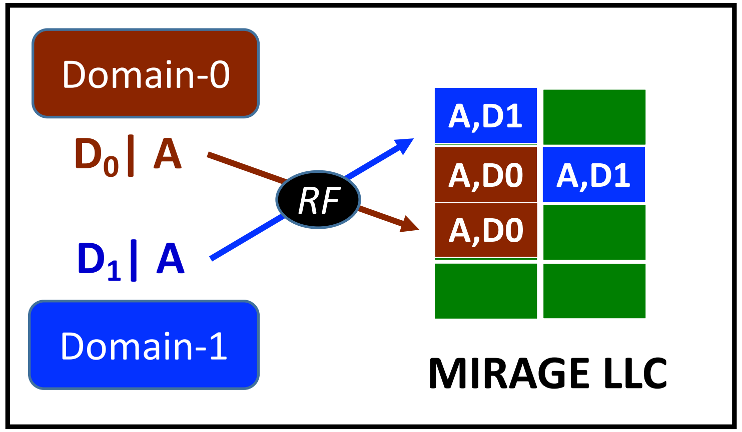
MIRAGE: Shared Memory Attacks

<u>Randomization Alone Cannot Mitigate Shared-Memory Attacks</u>

(e.g. Flush+Reload, Flush+Flush)



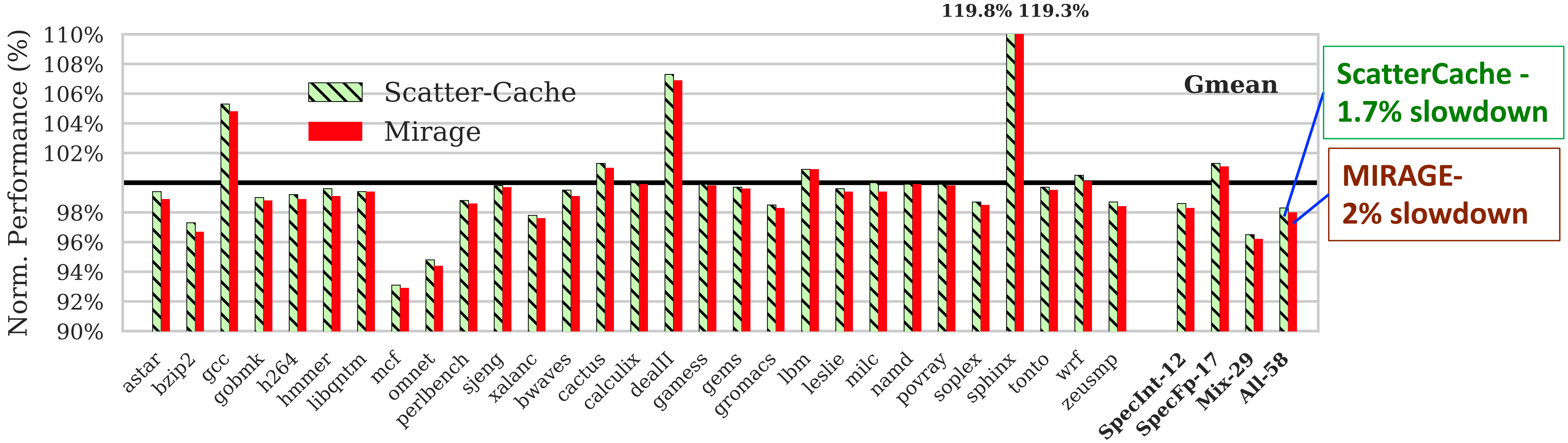
<u>MIRAGE uses Domain-ID for duplication of shared cache lines</u>



Eliminates Shared-Memory Based Attacks

Results - Performance

Simulation of 8-Core 16MB/16-way LLC system on Trace-Based Simulator

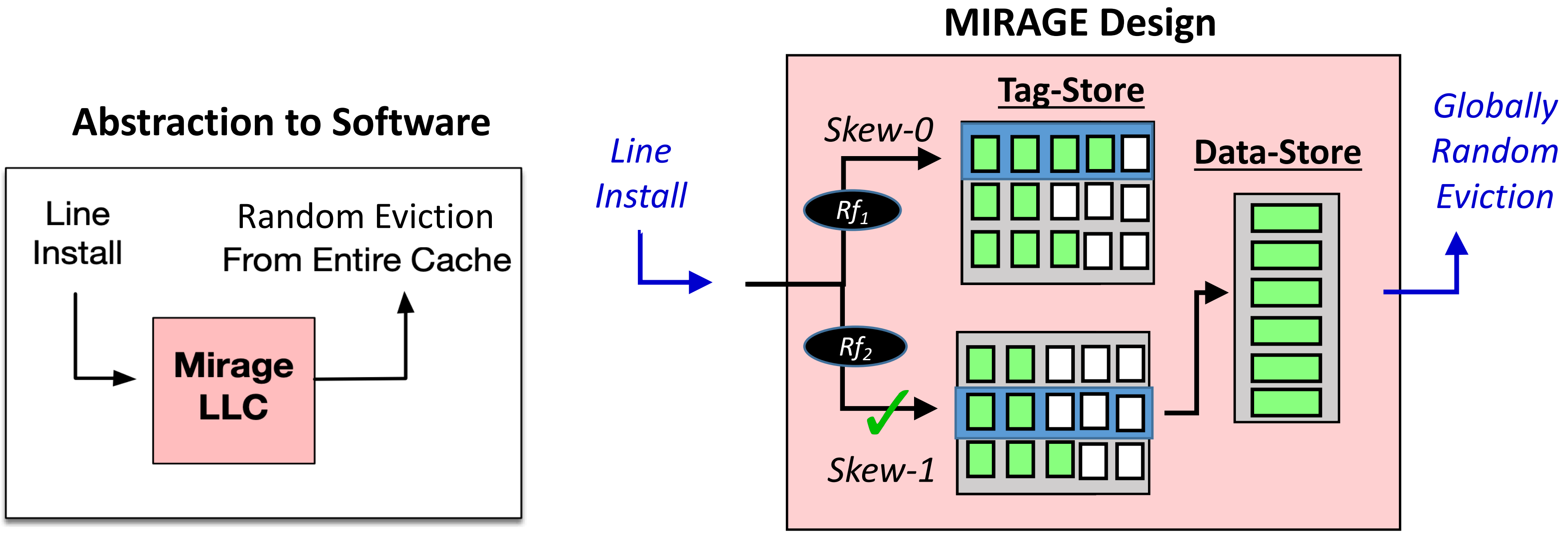


2% Slowdown, 20% Storage Overhead (75% extra tags).
Storage-Neutral Slowdown → 3.5%.

Paper includes MIRAGE-Lite with lower storage overheads (50% extra tags & similar security)

Additional Results in Paper: LLC Misses, Lookup Latency, Logic Overhead, RISC-V, Gem5 etc.

Takeaways from MIRAGE



Principled Security that Eliminates Cache-Attacks Leaking Victim Addresses

- **Strong Benefits:** Security of 1 SAE per 10^{17} Years
- **Modest Costs:** 2% Slowdown, 17% - 20% Storage Overhead

Thanks!

Code | **Slides** | **Paper**

ARTIFACT EVALUATED
usenix ASSOCIATION
PASSED

Code (Gem5 Artifact): https://github.com/gururaj-s/mirage

Slides: http://memlab.ece.gatech.edu/slides/SEC_2021_1_slides.pptx

Paper: https://www.usenix.org/system/files/sec21fall-saileshwar.pdf

Gururaj Saileshwar
gururaj.s@gatech.edu

Moinuddin Qureshi
moin@gatech.edu