

Synchronization Storage Channels (S²C): Timer-less Cache Side-Channel Attacks on the Apple M1 via Hardware Synchronization Instructions

Jiyong Yu, Aishani Dutta, Trent Jaeger*, David Kohlbrenner+, Chris Fletcher

University of Illinois Urbana-Champaign, *Penn State University, +University of Washington



UNIVERSITY OF
ILLINOIS
URBANA-CHAMPAIGN



PennState



UNIVERSITY *of*
WASHINGTON

Observing μ arch State is Crucial for Side-Channel Attacks

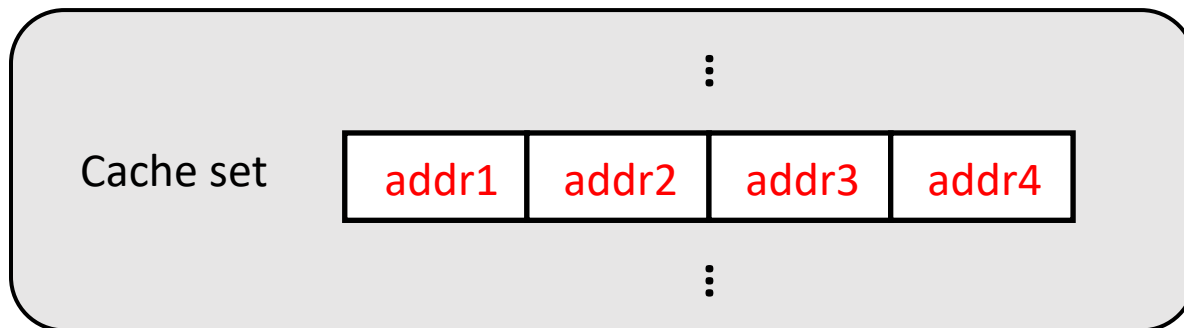


Victim Secret \rightarrow μ arch State Change



```
load [addr1]  
load [addr2]  
load [addr3]  
load [addr4]
```

Shared
cache



Observing μ arch State is Crucial for Side-Channel Attacks

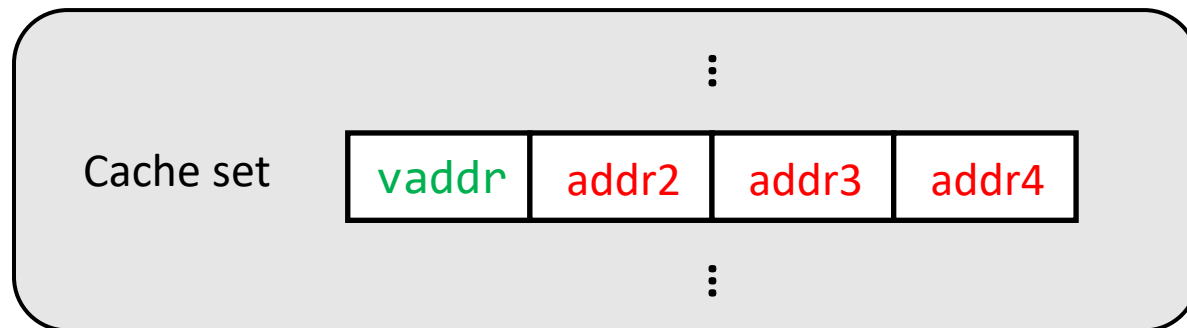


Victim Secret \rightarrow μ arch State Change



load [**vaddr**]

Shared
cache



Observing μ arch State is Crucial for Side-Channel Attacks

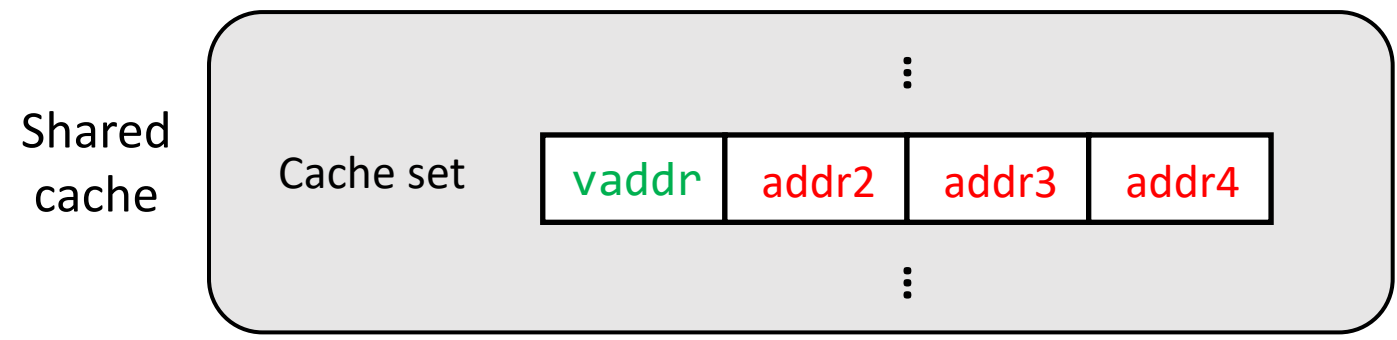


Timer



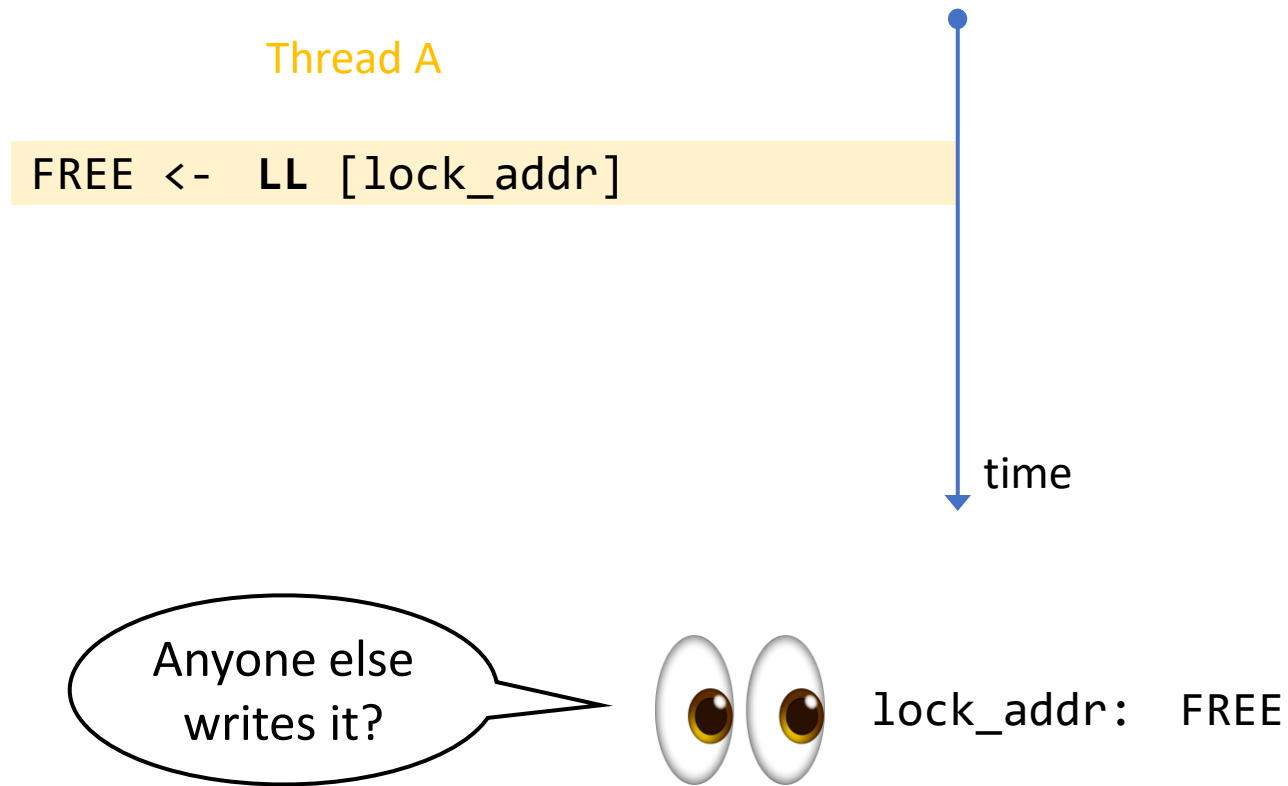
vs.

Timer-less (rare)

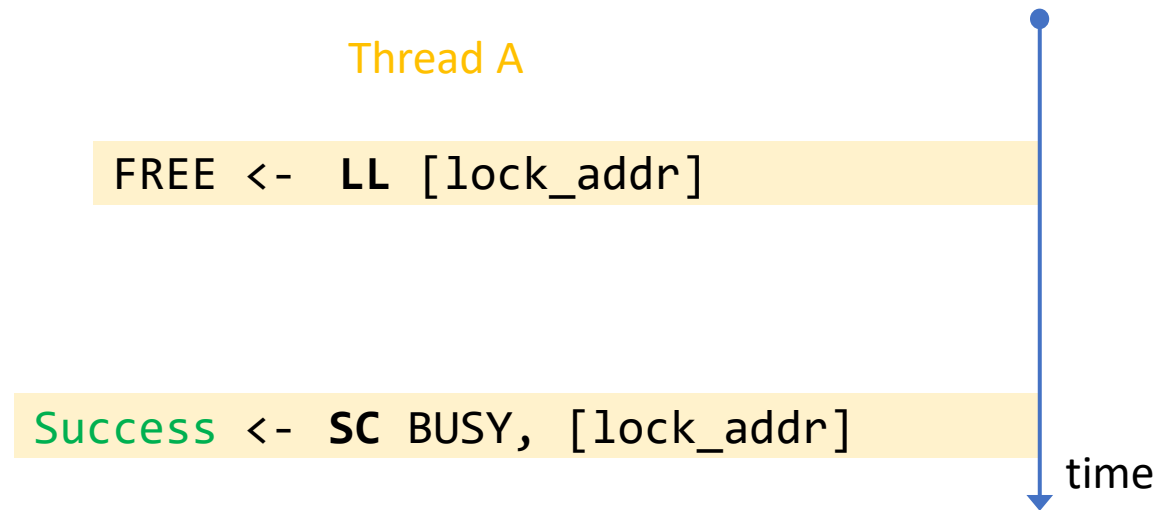


Uncacheable Memory [S&P'16]
Intel TSX [USENIX'17, HPCA'22]
This Work

Load-Linked (LL) / Store-Conditional (SC) on Apple M1



Load-Linked (LL) / Store-Conditional (SC) on Apple M1

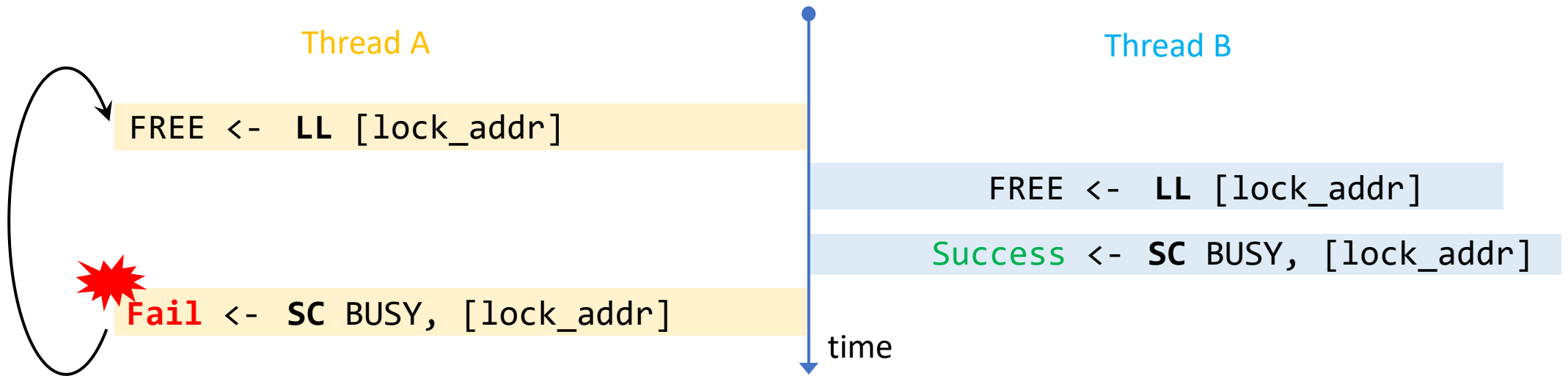


Anyone else writes it?



lock_addr: BUSY

Load-Linked (LL) / Store-Conditional (SC) on Apple M1



Anyone else writes it?



lock_addr: **BUSY** (lock taken by thread B first)



SC only fails on data race like this?



NO!

Observations of LL/SC on Apple M1

LL [**X**]

⋮

Success <- SC [**X**]

Private L1

X

Shared L2

Observations of LL/SC on Apple M1

LL [X]

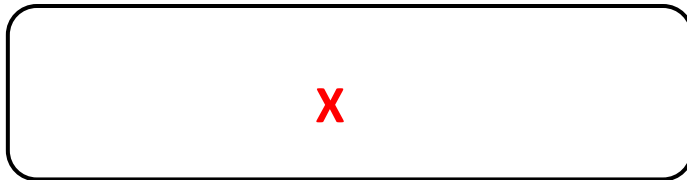
⋮

Fail ← SC [X]

Private L1



Shared L2



Key observations:

- LL/SC only monitors addresses in L1

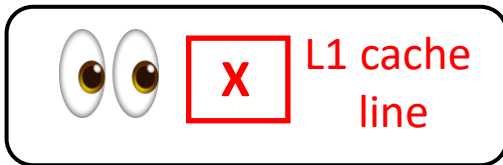
Observations of LL/SC on Apple M1

LL [X]

⋮

Success ← SC [X]

Private L1



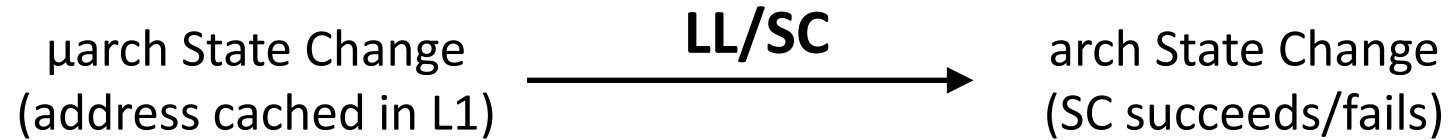
Shared L2



Key observations:

- LL/SC only monitors addresses in **L1**
- Monitoring granularity = **L1 cache line size**
- Each core only supports **1** outstanding LL/SC

Cross-core Cache Attack with LL/SC



But ...

LL/SC only monitors addresses in private L1.

Only 1 address is monitored at a time.

What cross-core cache attack needs ...


Observation over the shared L2.

Monitoring multiple addresses (ideally)

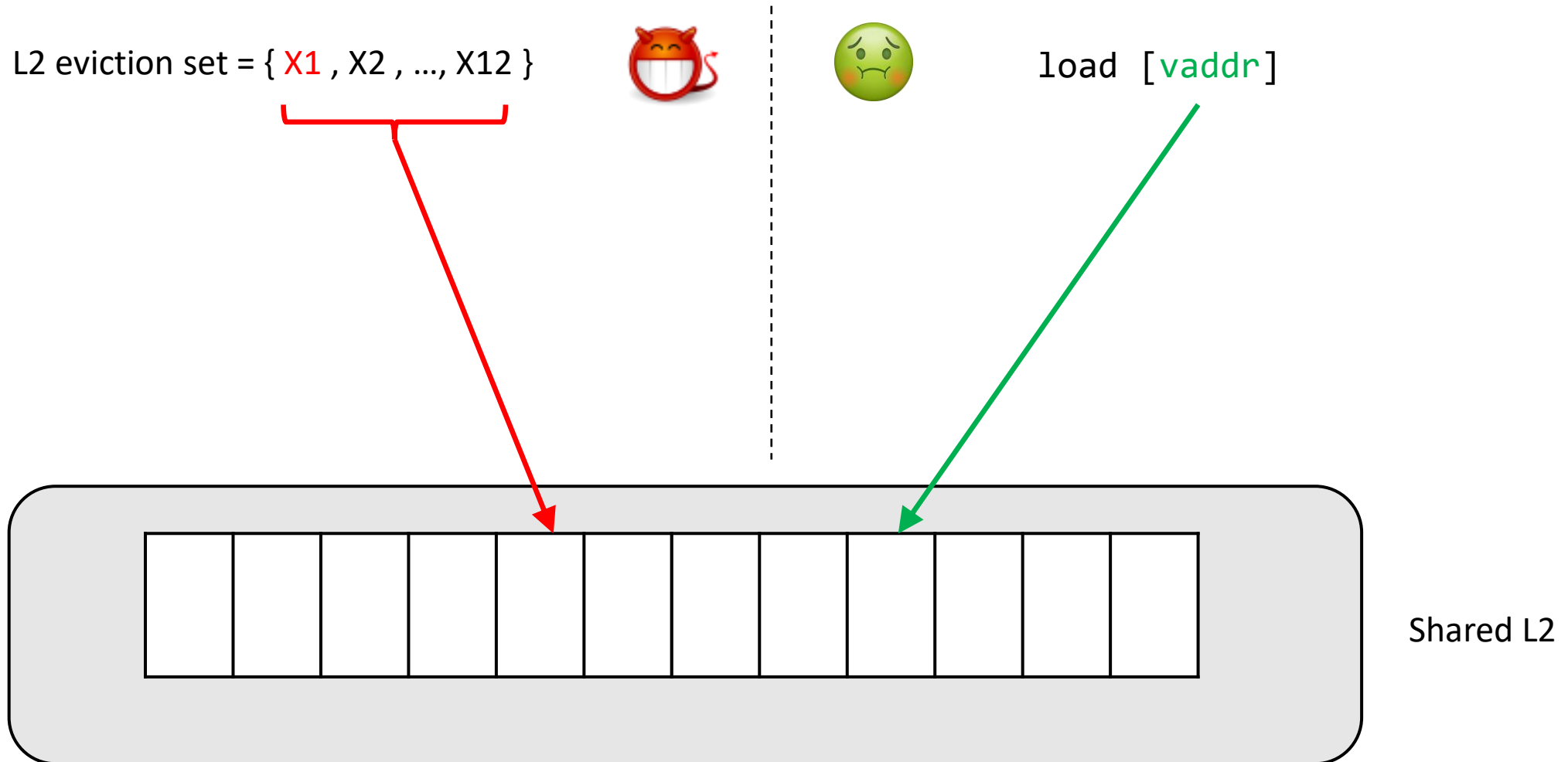


Synchronization Storage Channel (S²C)

M1 L2 Cache Reverse-Engineering

- Inclusion Policy
 - L1/L2 Replacement Policy
 - L1/L2 Cache Set Index Mapping
- Precisely control L1 / L2 evictions
- Generate eviction set
- 

Basic S²C: Monitor victim's access to a single L2 set

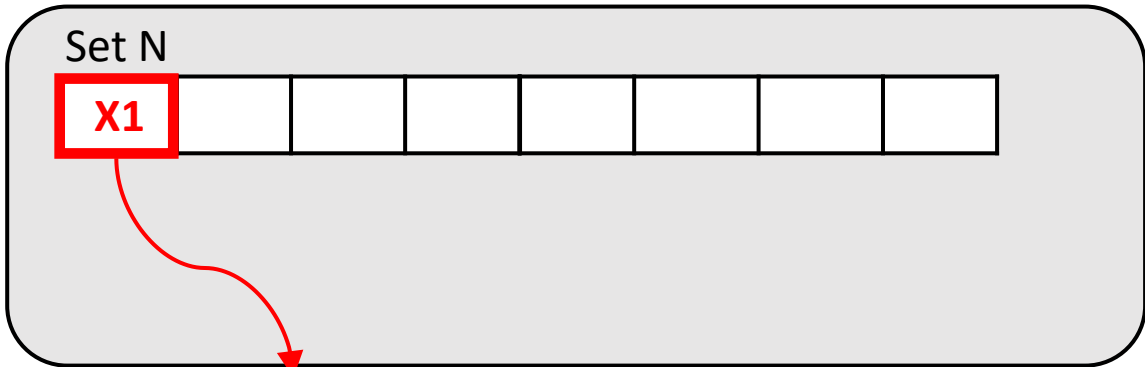


Basic S²C: Monitor victim's access to a single L2 set

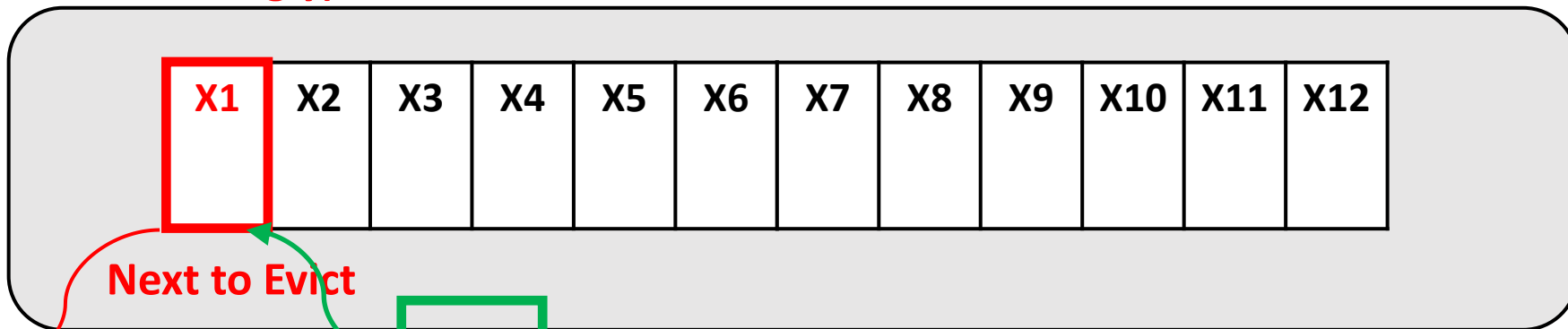
LL [X1]
Load X2, X3, ..., X12



load [vaddr]



(Evicted accordingly)



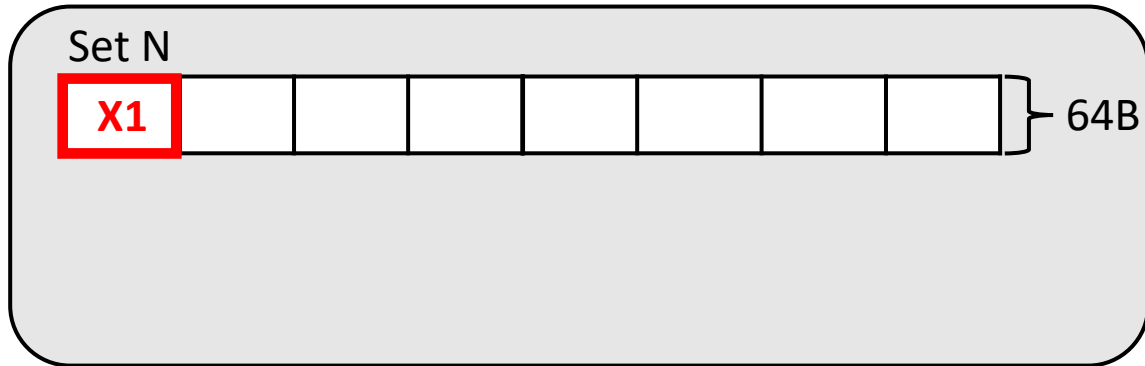
Shared L2

(Evicted)

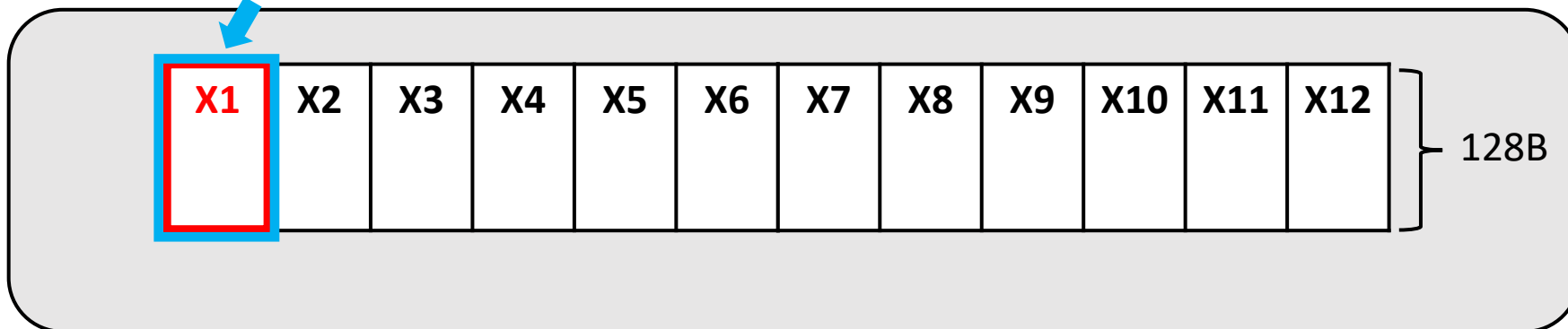
vaddr

Basic S²C: Monitor victim's access to a single L2 set

LL [X1]
Load X2, X3, ..., X12



“AutoLocked”

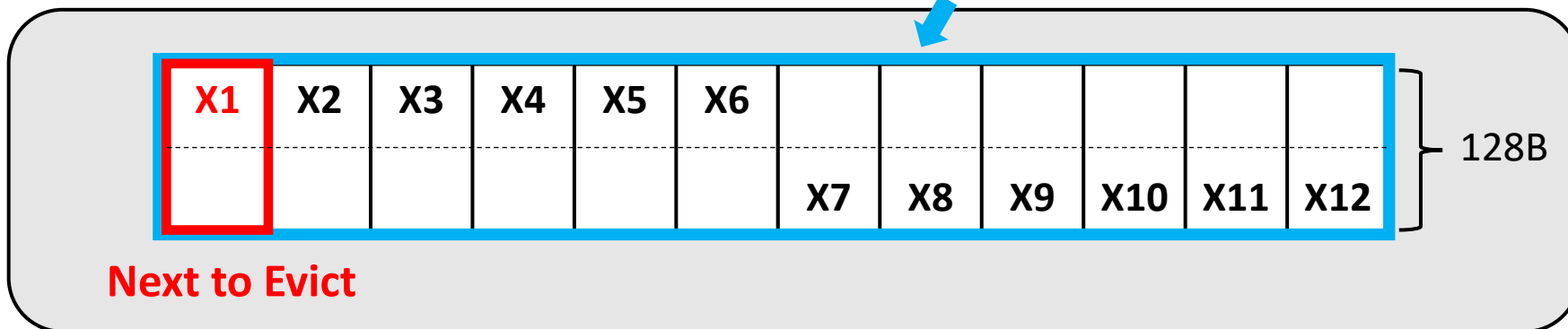


Basic S²C: Monitor victim's access to a single L2 set

LL [X1]
Load X2, X3, ..., X12



All "AutoLocked" <-> No one is "AutoLocked"

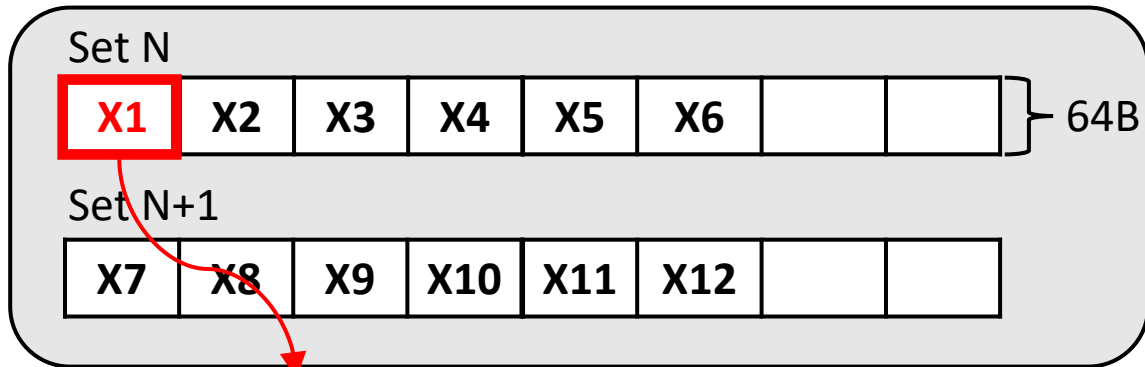


Basic S²C: Monitor victim's access to a single L2 set

Fail \leftarrow SC [X1]

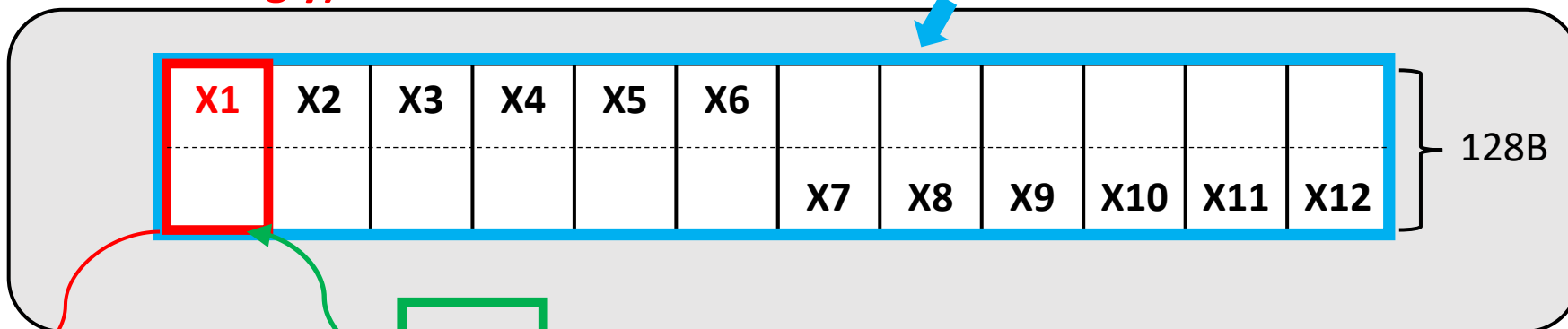


load [vaddr]



(Evicted accordingly)

All "AutoLocked" \leftrightarrow No one is "AutoLocked"



Shared L2

vaddr

(Evicted)

Advanced S²C: Monitor victim's access to multiple L2 sets

L2 cache set



X1

evicted

Fail ← SC [X1]

L2 cache set



X2

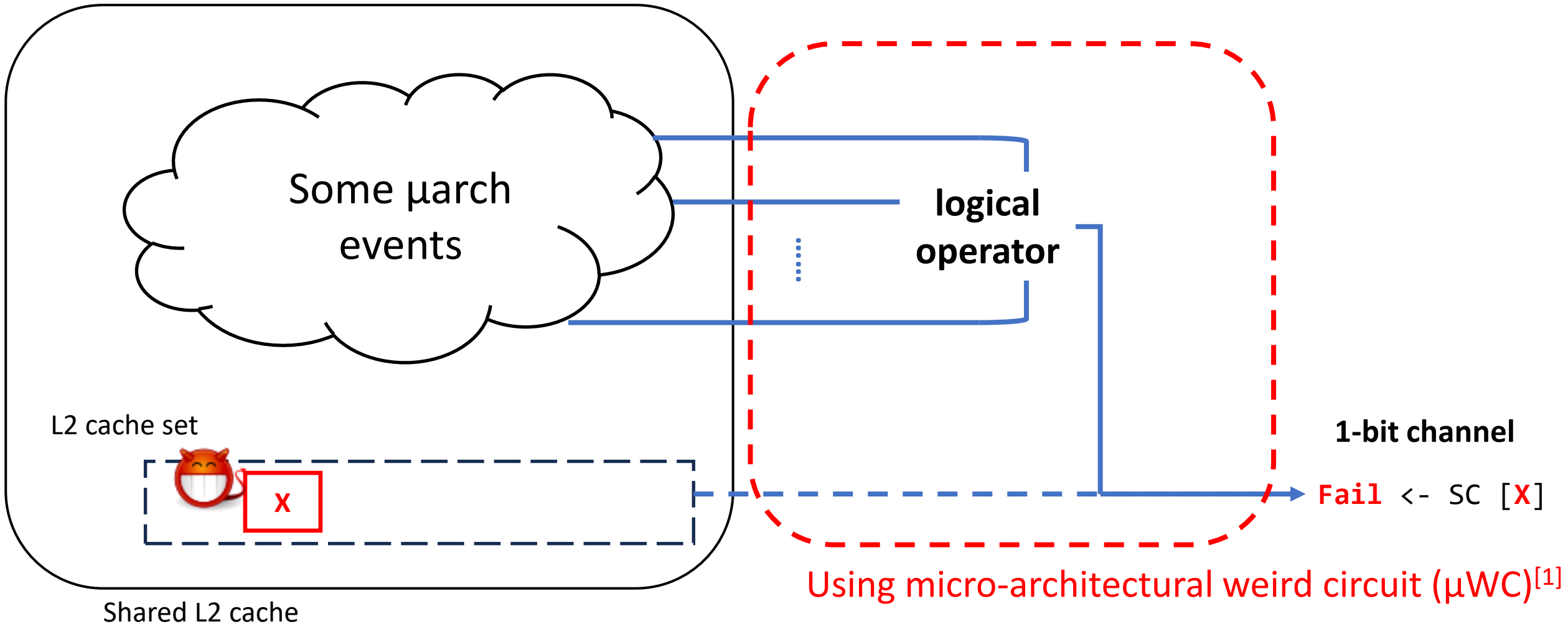
evicted

Fail ← SC [X2]

Impossible!

Shared L2 cache

Advanced S²C: Monitor victim's access to multiple L2 sets



Advanced S²C: Monitor victim's access to multiple L2 sets

L2 cache set



L2 cache set



L2 cache set

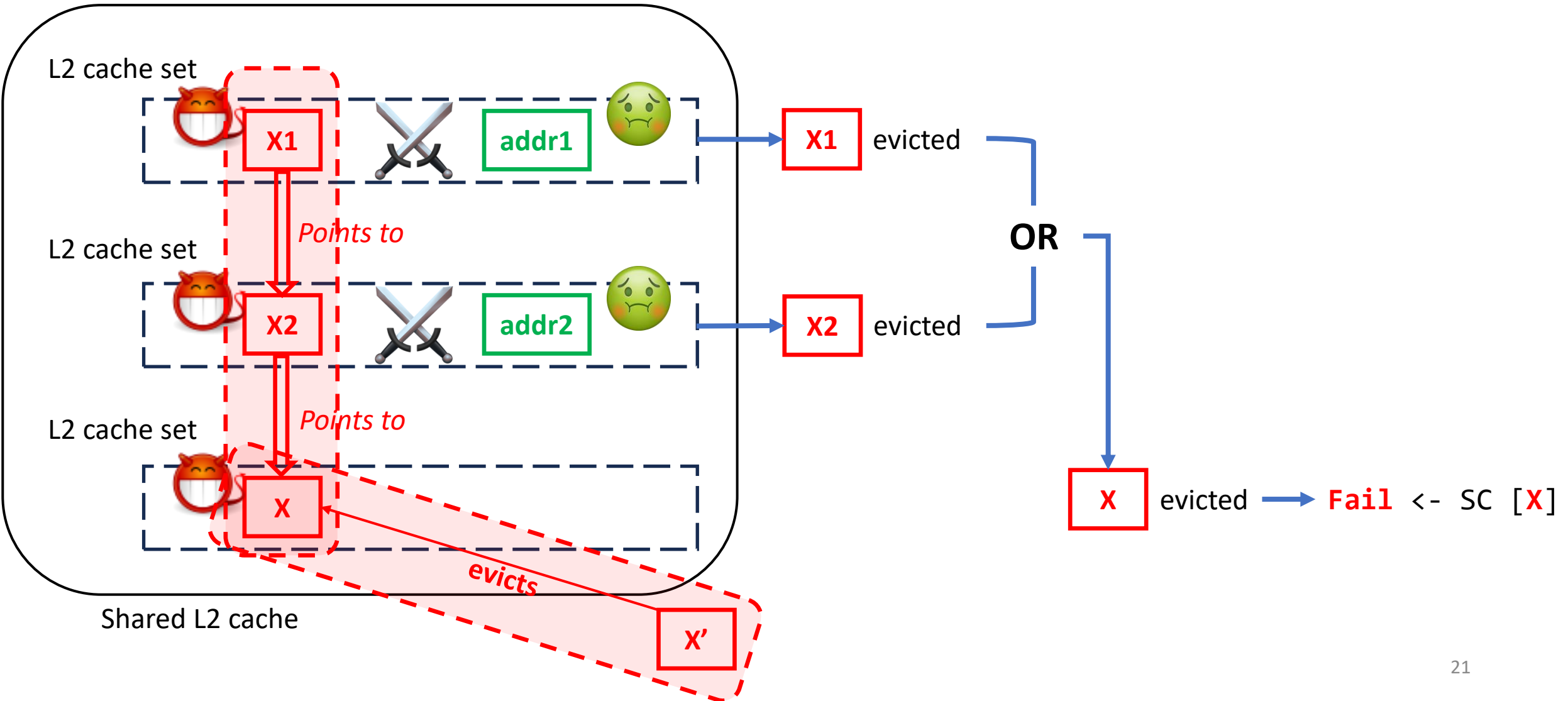


OR

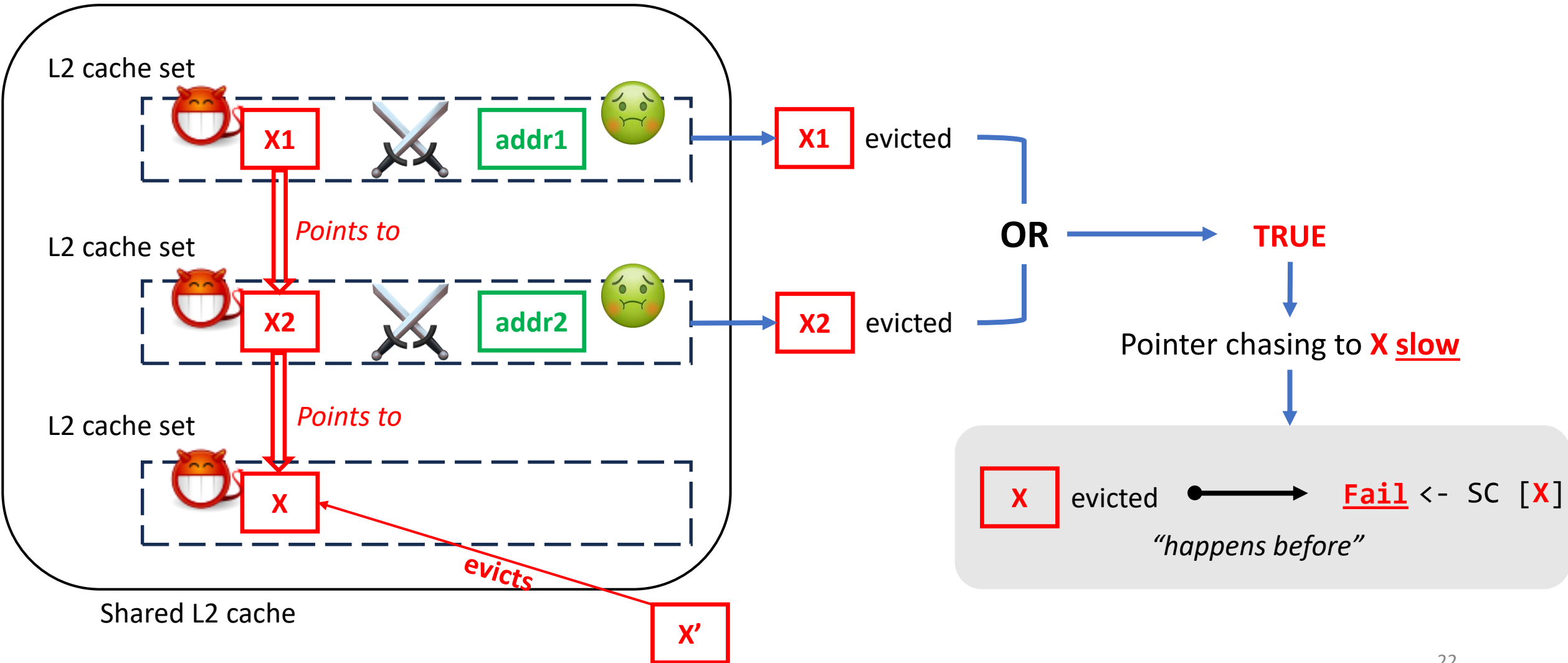
Fail <- SC [X]

Shared L2 cache

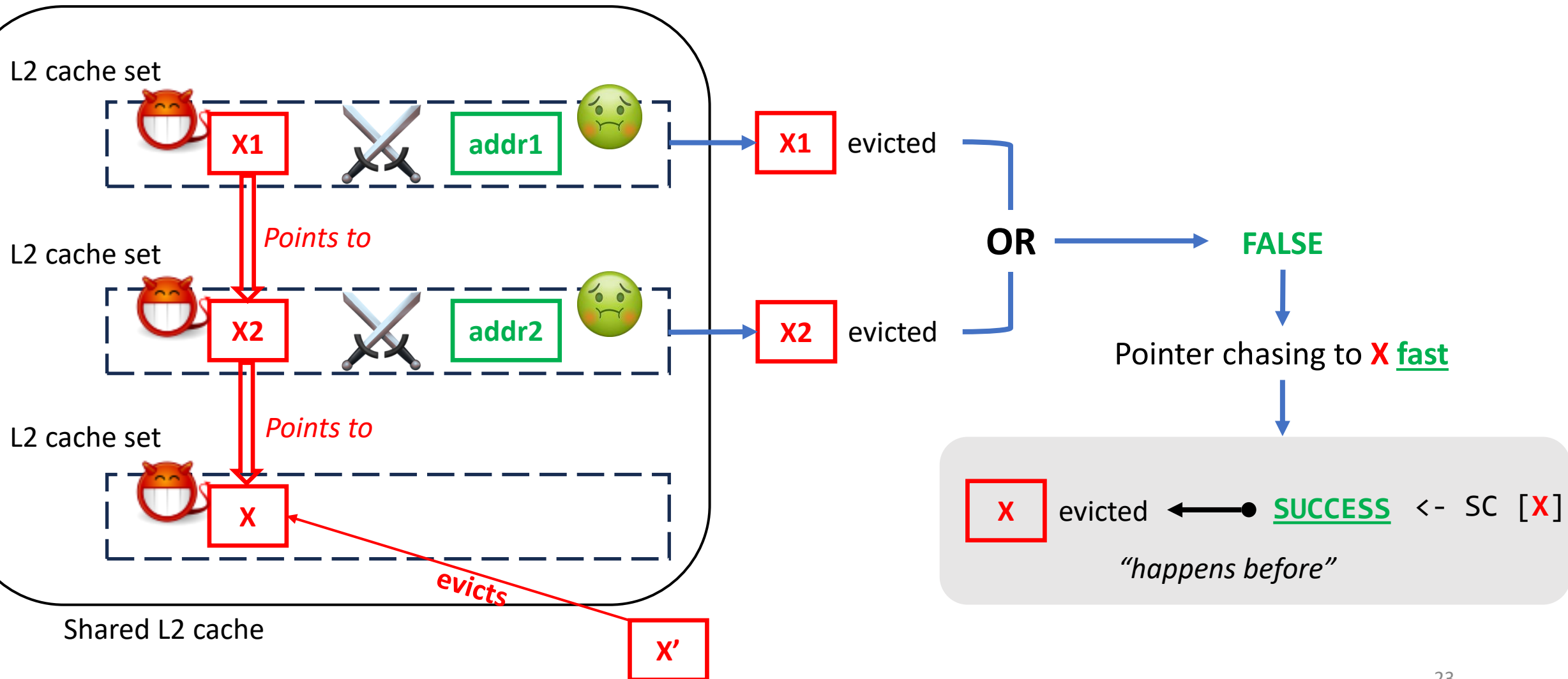
Advanced S²C: Monitor victim's access to multiple L2 sets



Advanced S²C: Monitor victim's access to multiple L2 sets



Advanced S²C: Monitor victim's access to multiple L2 sets



Evaluation

- μ WC method can monitor at most 11 different L2 cache sets
- Cross-core covert channel: 185Kb/s with 98.5% accuracy
- Full private key extraction in T-table AES

Conclusion

- **Synchronization Storage Channels (S²C):**
 - 1st timer-less, cross-core attack on Apple M1
 - 1st cache attack leveraging hardware synchronization instructions (LL/SC)
- Motivate future efforts in finding new “ μ arch-state-to-arch-state converters”

Synchronization Storage Channels (S²C): Timer-less Cache Side-Channel Attacks on the Apple M1 via Hardware Synchronization Instructions

Jiyong Yu, Aishani Dutta, Trent Jaeger*, David Kohlbrenner⁺, Chris Fletcher

University of Illinois Urbana-Champaign, *Penn State University, ⁺University of Washington



UNIVERSITY OF
ILLINOIS
URBANA-CHAMPAIGN



PennState



UNIVERSITY *of*
WASHINGTON

