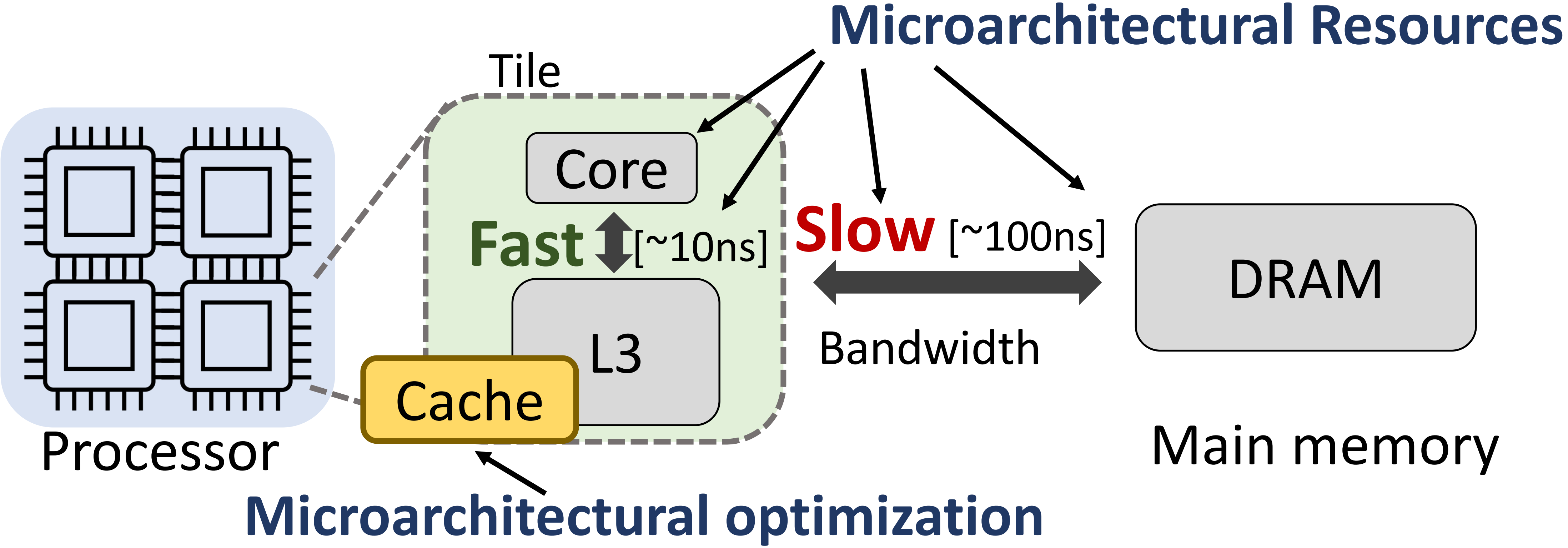
SoK: Analysis of Root Causes and Defense Strategies for Attacks on Microarchitectural Optimizations

Nadja Ramhöj Holtryd, Madhavan Manivannan and Per Stenström

Department of Computer Science and Engineering
Chalmers University of Technology

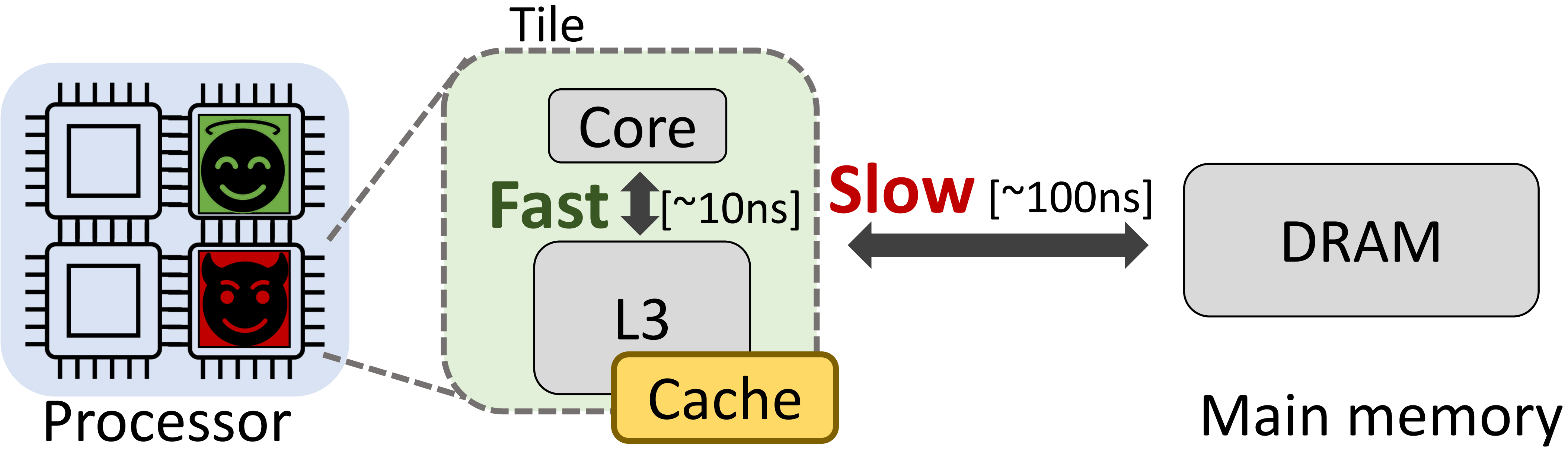
IEEE European Symposium on Security and Privacy 2023

Modern Multi-Core



Optimizations exploited in attacks

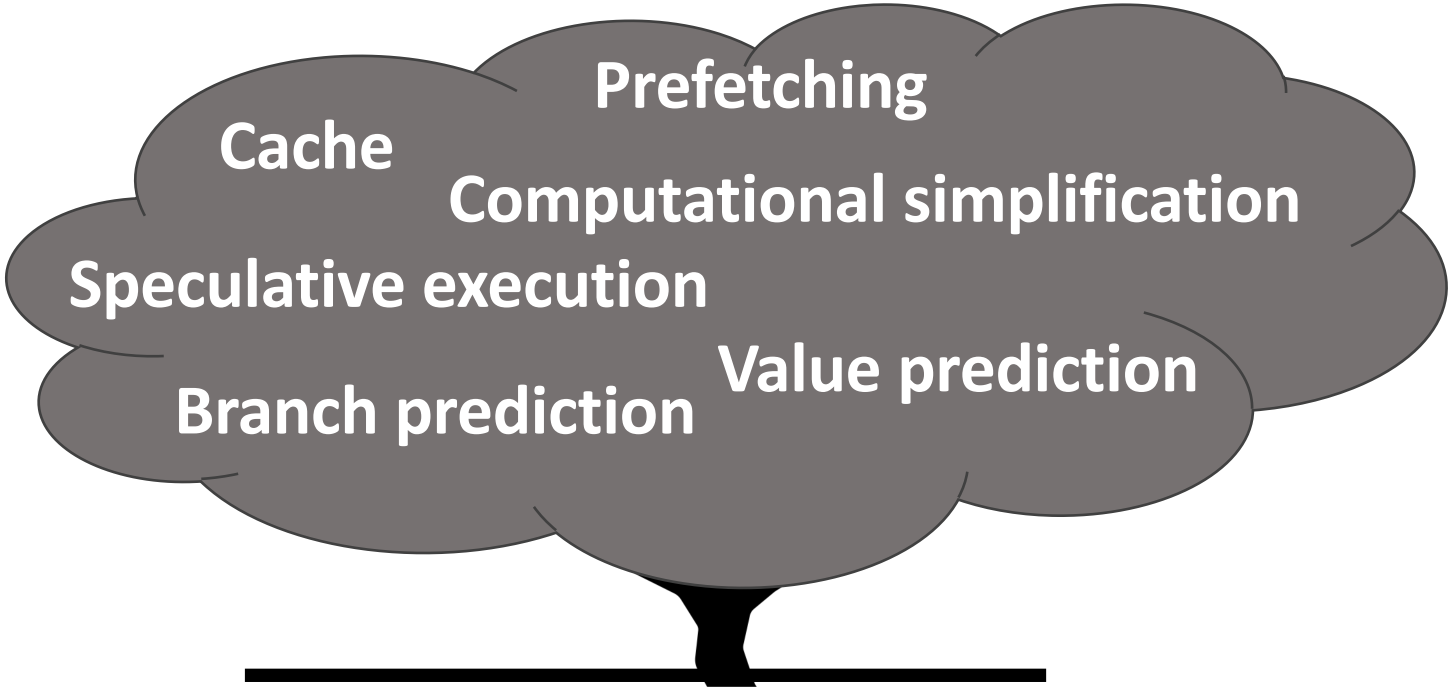
Attacks Exploit Optimizations



Attacks use timing variability: **Fast** 🐇 **Slow** 🐌

Can leak cryptographic keys

Why Common Root Causes?



Prefetching

Cache

Computational simplification

Speculative execution

Value prediction

Branch prediction

Problem

State-of-the-art:

- Only analyzed subset of optimizations […]
  - Only found root causes in context of individual optimizations
- Focused on quantifying leakage [Pandora]

What are the common ***root causes*** for timing-based attacks on microarchitectural optimizations?

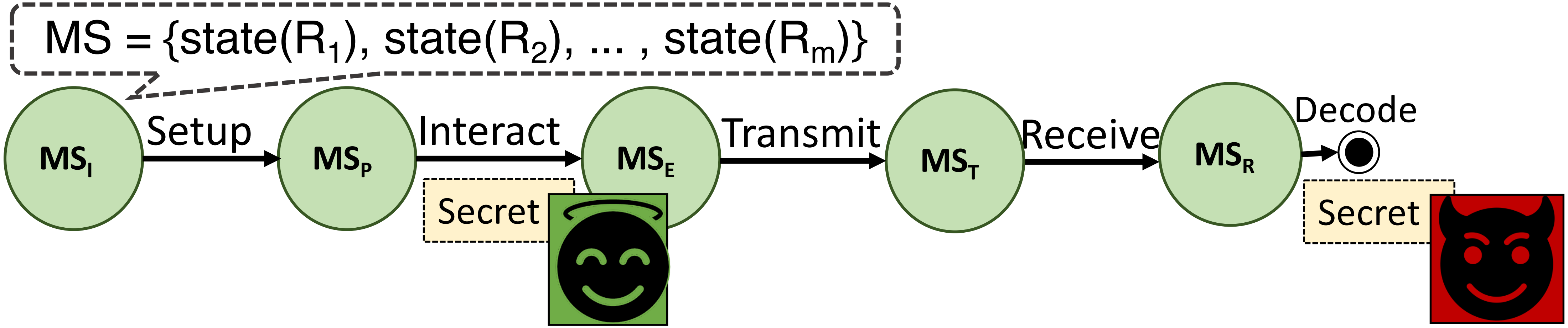
Systematization-of-Knowledge

Key idea: Abstract framework and identify the common root causes of timing-based side-channel attacks

- Unified and abstract framework
- Identification of the four root causes: ***determinism, sharing, access violation*** and ***information flow***.
- Systematic analysis of **attacks** and **defences** on a broad range of microarchitectural optimizations: Cache, Prefetching, Branch prediction, Computational simplification, Speculative execution and Value prediction

Framework: Model and Attack Steps

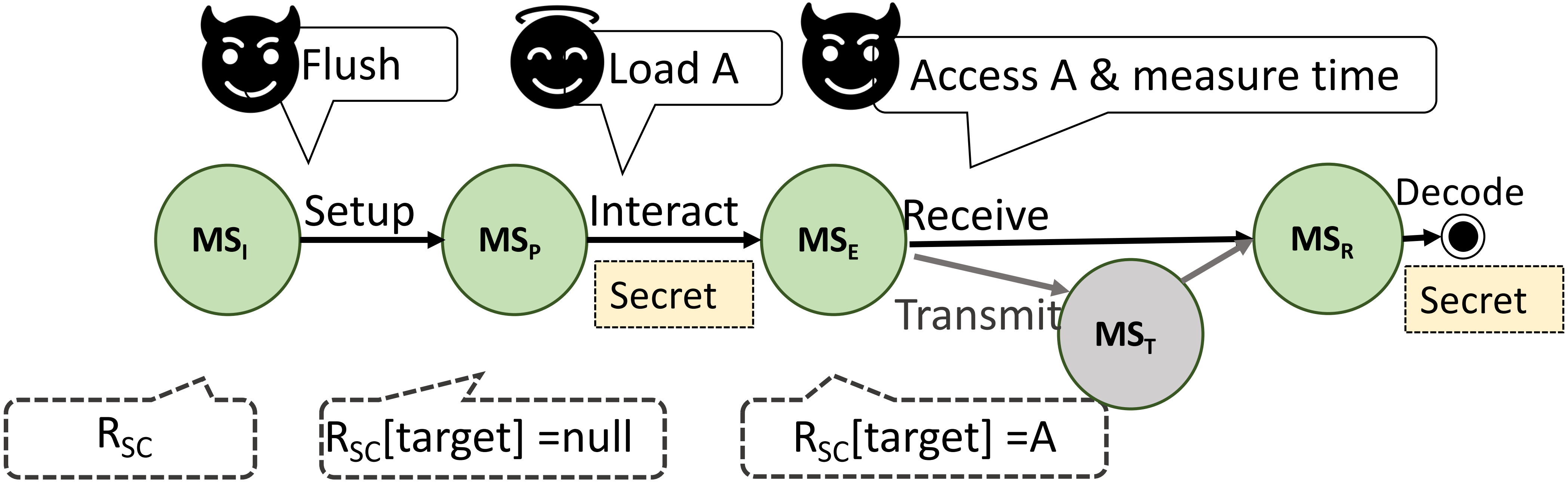
- The architectural model is a finite state machine
  - Architectural state (AS) is externally visible
- Many-to-one mapping between AS and MS
- Microarchitectural state (MS): snapshot of the state in microarchitectural resources depending on microarchitectural optimizations
- Attacker/victim actions modify state: {$MS_{current}$, action} → MS_{next}



Root Causes

- **Determinism** causes microarchitectural optimizations to be triggered in the same way under the same pre-conditions
  - Leads to predictable state transitions and timing variations.
- **Sharing** of microarchitectural state, between adversary and victim, enables the creation of a side-channel.
- **Access violation** enables access to a secret outside of the intended protection domain.
- **Information flow** refers to exchange of information through microarchitectural state.

Flush+Reload Attack on Shared Cache

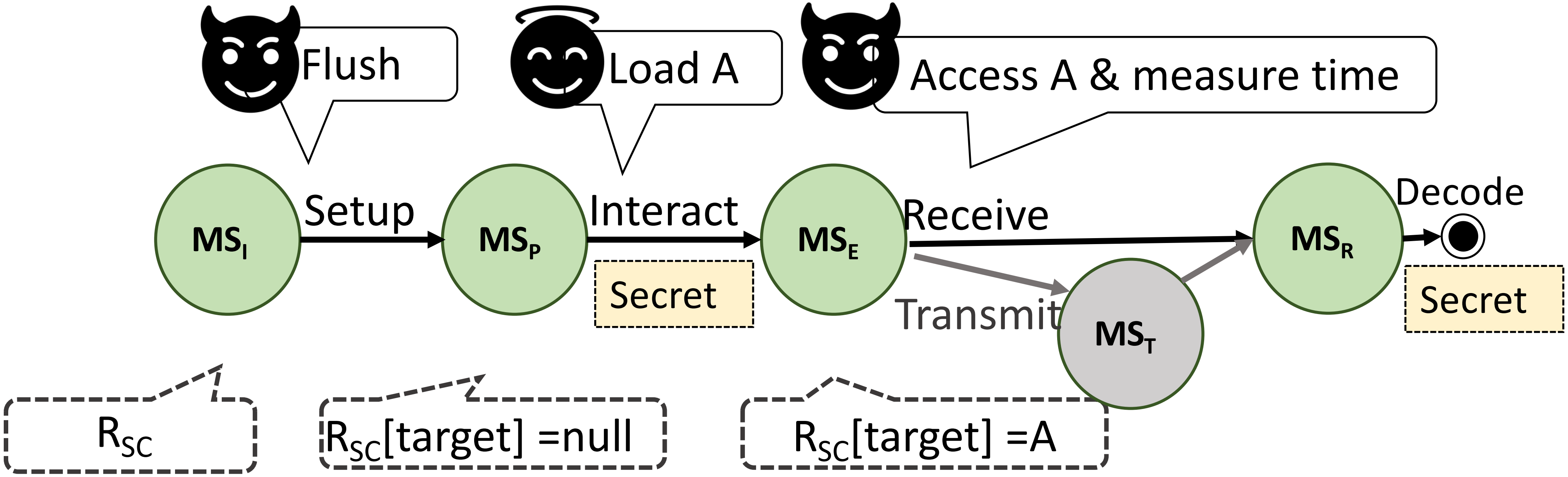


Determinism
Sharing
Information flow

Determinism
Sharing
Information flow

Determinism
Sharing
Information flow

Defense: Disable clflush

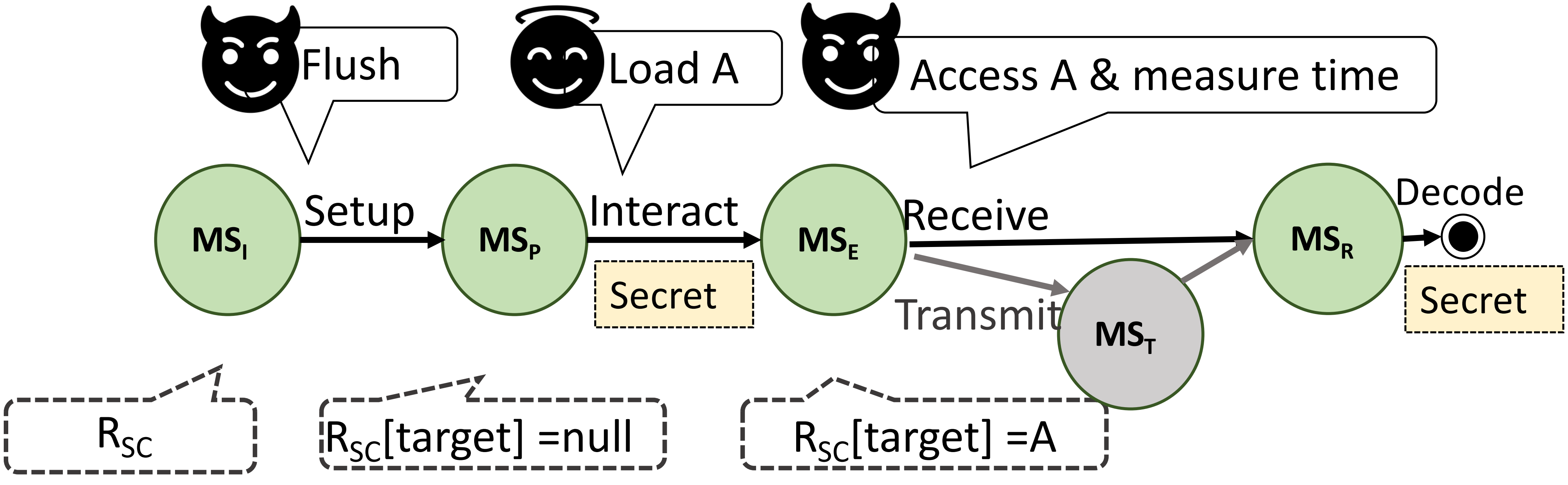


Determinism
~~Determinism~~
Sharing
~~Sharing~~
Information flow
~~Information flow~~

Determinism
Sharing
Information flow

Determinism
Sharing
Information flow

Defense: Randomization

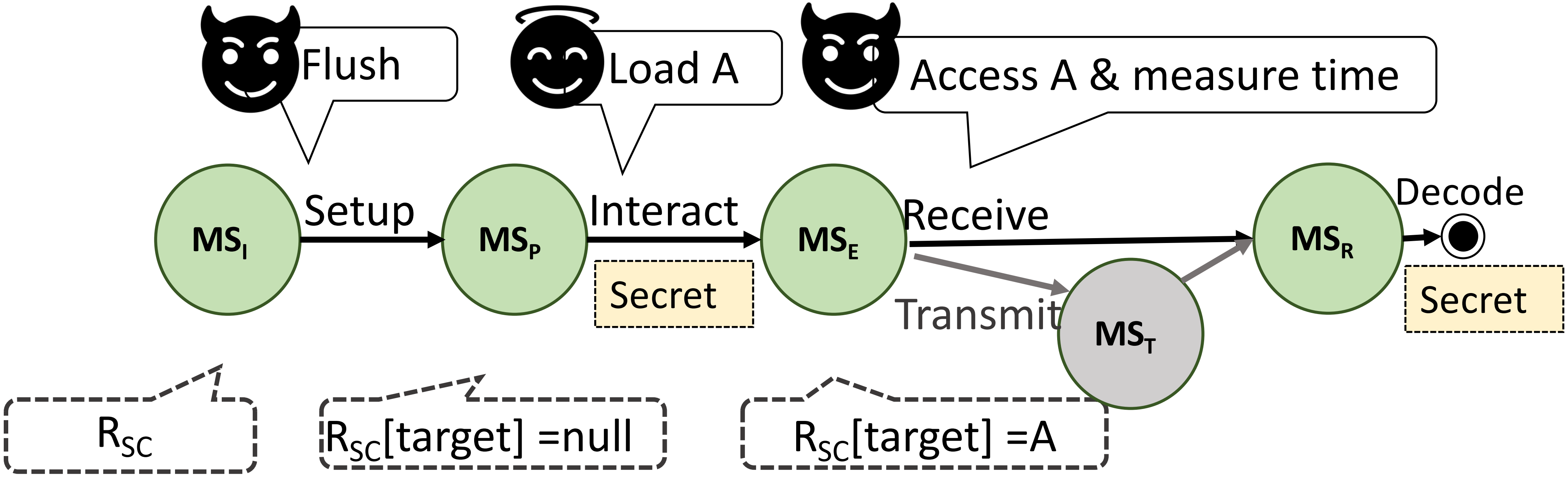


~~*Determinism*~~
Sharing
~~*Information flow*~~

~~*Determinism*~~
Sharing
~~*Information flow*~~

~~*Determinism*~~
Sharing
~~*Information flow*~~

Defense: Partitioning



~~Determinism~~
~~Sharing~~
~~Information flow~~

~~Determinism~~
~~Sharing~~
~~Information flow~~

~~Determinism~~
~~Sharing~~
~~Information flow~~

SoK: Attack and Defense Classification

- Both transient and non-transient attacks using these optimizations:
  - Cache
  - Prefetching
  - Branch prediction
  - Computational simplification
  - Speculative execution
  - Value prediction

- Threat model
- Performance overheads
- Protection level: Resources and threat model

Takeaways

1. The root causes are common
   - We have shown that the root causes for attacks are common, across a wide range of microarchitectural optimizations
2. Common root causes leads to common defense strategies
   - Partitioning, randomization, flushing etc.
3. New defense strategies for vulnerable optimizations
   - Apply common strategies to currently vulnerable optimizations
   - Combining strategies promising to decrease performance cost

Conclusions

- Increased importance of optimizations for performance
- Crucial to understand the root causes of attacks
- Our framework
  - Analyse attacks and defences on a wide range of microarchitectural optimizations.
  - Highlighting similarities and differences.
- Four root causes for timing-based side-channel attacks:
  - Determinism, sharing, access violation and information flow

Backup

Future work

- Implementation specific analysis
  - Focusing on specific resources/optimizations in Intel, AMD and ARM architectures.
- Use the framework to explored attacks and defenses on other optimizations and resources (such as NoC and DRAM)
- Extend the root cause framework to include microarchitectural optimizations for security, such as Intel SGX and performance degradation attacks.