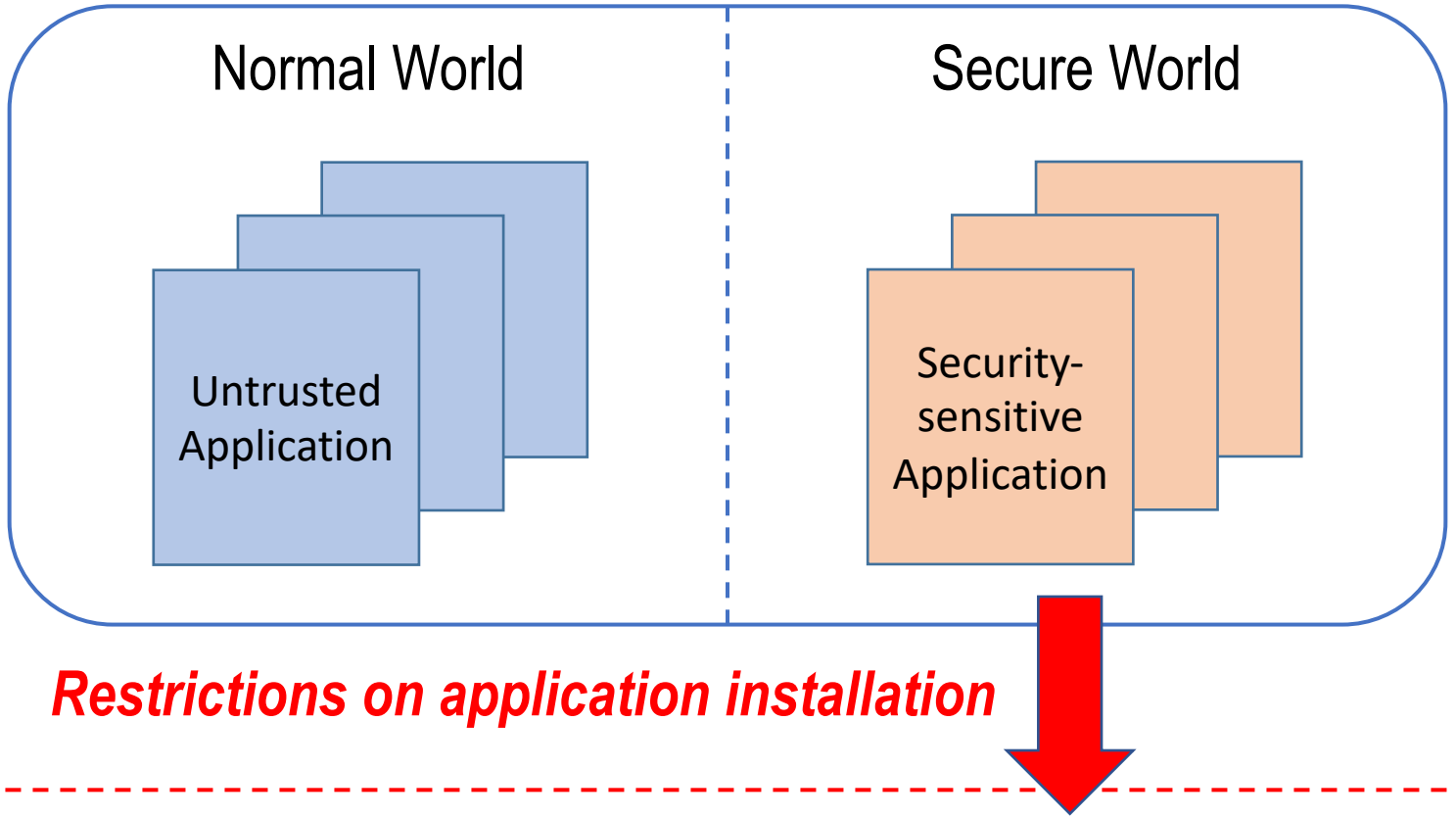
Cache-in-the-Middle (CITM) Attacks :

Manipulating Sensitive Data in Isolated Execution Environments

Jie Wang[1,2,3], Kun Sun[2], Lingguang Lei[1,3], Shengye Wan[2,4], Yuewu Wang[1,3], and Jiwu Jing[5]
[1]*SKLOIS, Institute of Information Engineering, CAS, China*
[2]*Department of Information Sciences and Technology, CSIS, George Mason University*
[3]*School of Cyber Security, University of Chinese Academy of Sciences, China*
[4]*Department of Computer Science, College of William and Mary*
[5]*School of Computer Science and Technology, University of Chinese Academy of Sciences*

ACM CCS, November 2020

ARM-based Trusted Execution Environment (TEE)



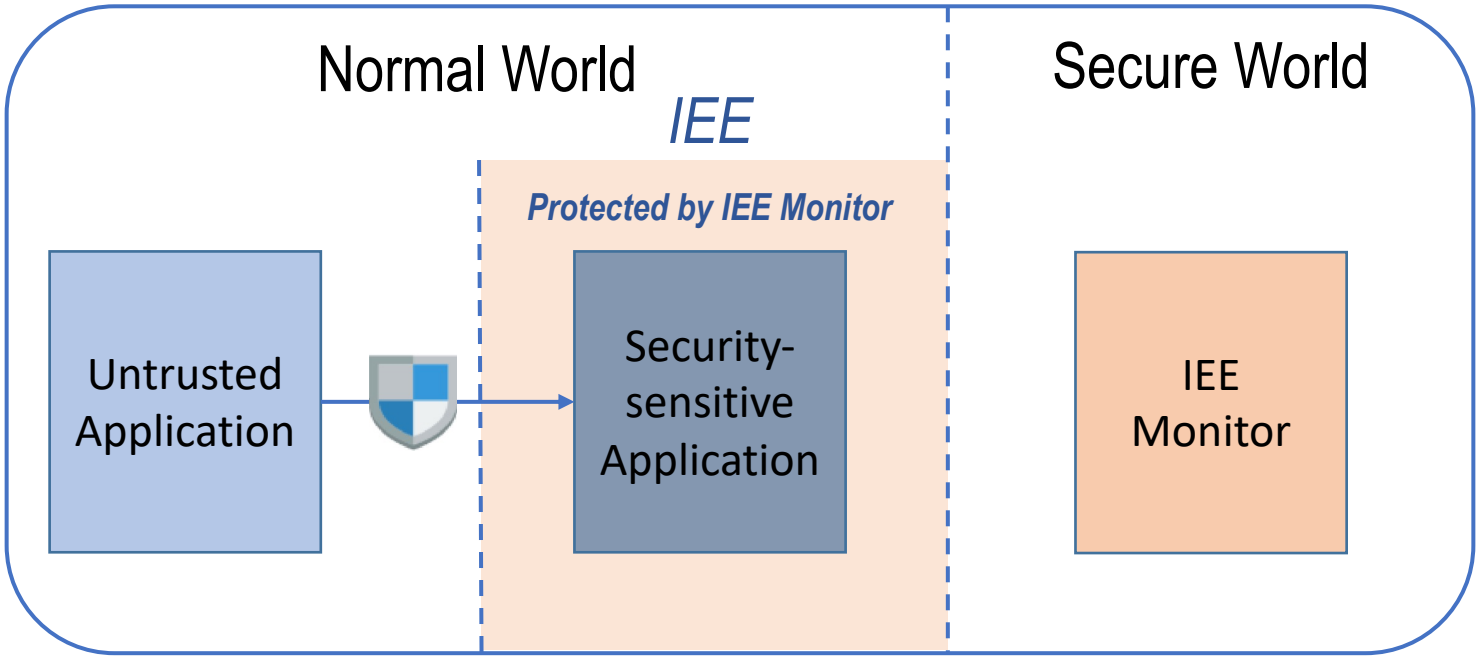
- Increased trusted computing base (TCB) in Secure World.
- Manufacturers prefer to only install their own applications with strict assessment.

Isolated Execution Environment (IEE)

TrustICE (DSN 2015), SANCTUARY (NDSS 2019), Ginseng (NDSS 2019), …….

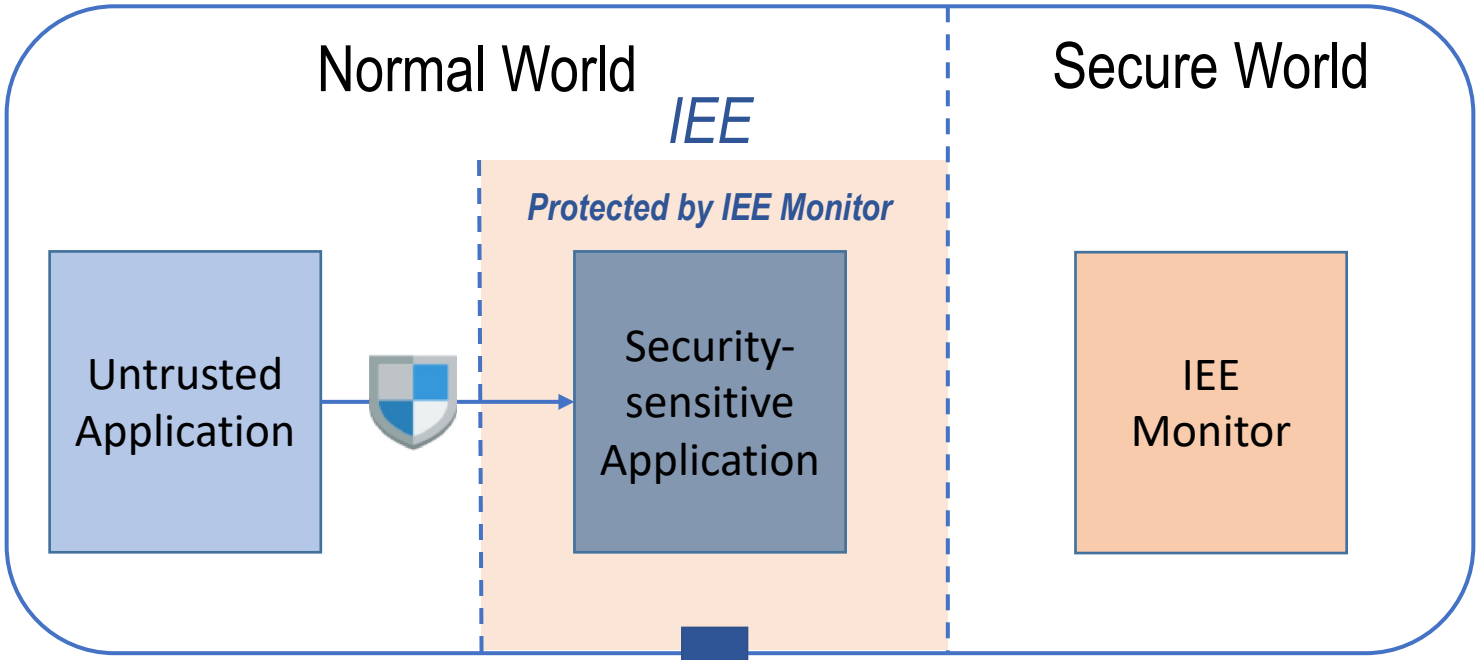
Introducing a new design: Isolated Execution Environment

Isolated Execution Environment (IEE)



- Creating Isolated Execution Environments (called IEEs) in the normal world.
- Using the IEE monitor in the secure world to ensure the security of IEEs.

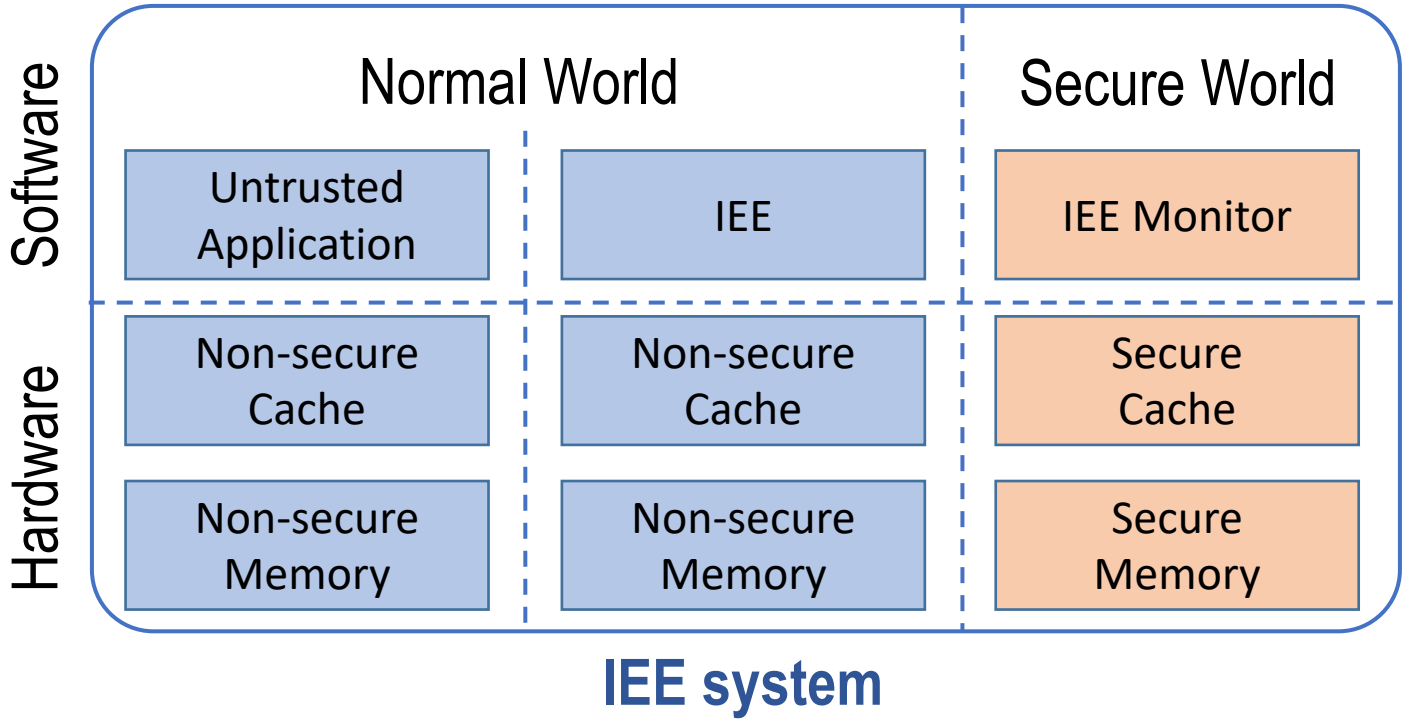
Isolated Execution Environment (IEE)



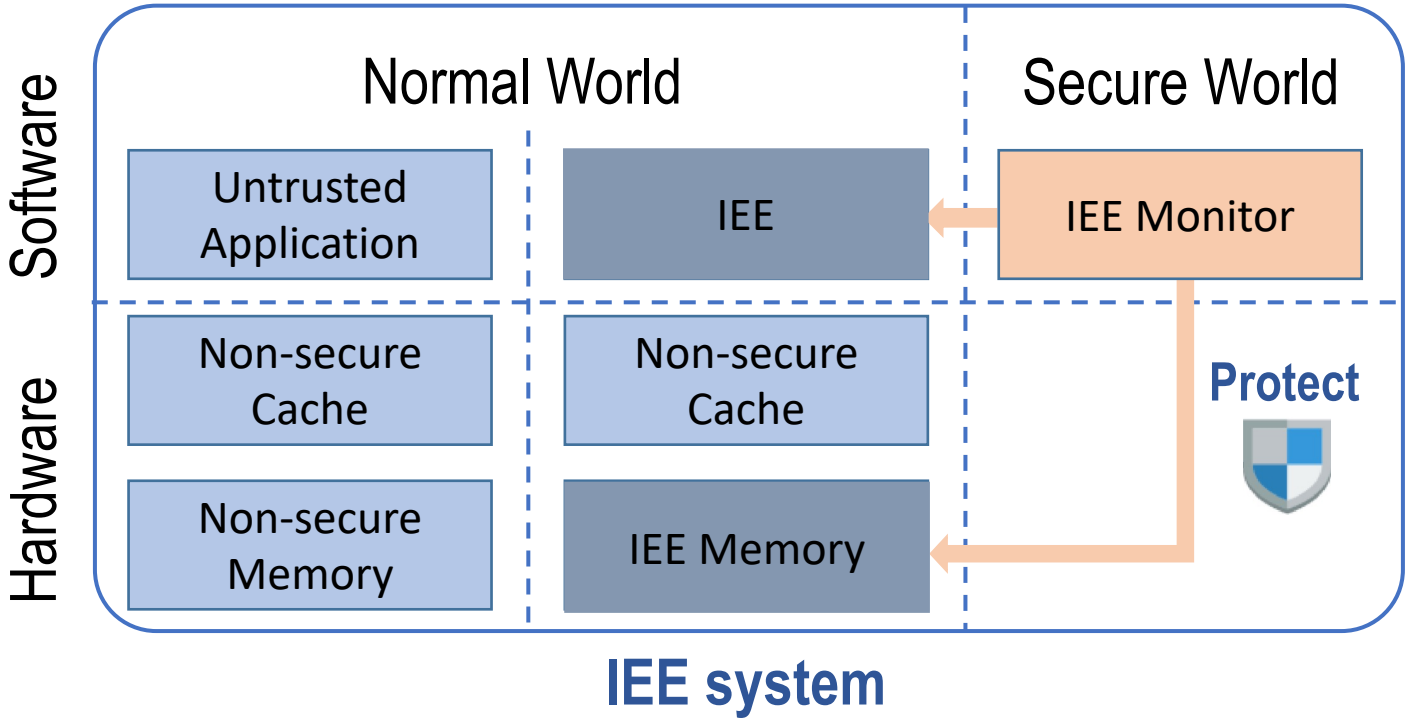
Improving the limitation of TEE systems

- Minimize the TCB of the secure world by only installing an IEE Monitor.
- More third-party applications can be imported for the enhanced security protection.

Cache-in-the-Middle (CITM) Attacks

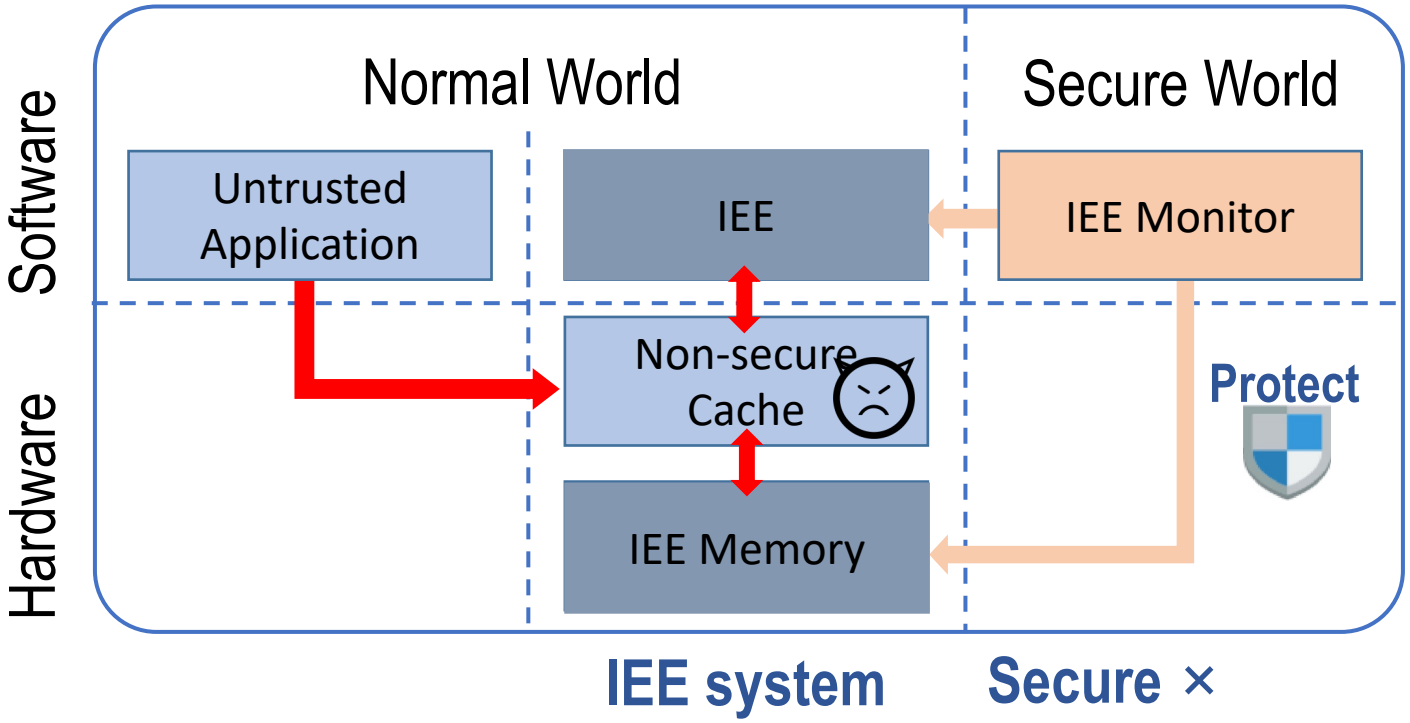


Cache-in-the-Middle (CITM) Attacks



Some existing systems ignore the security of data in the cache.

Cache-in-the-Middle (CITM) Attacks



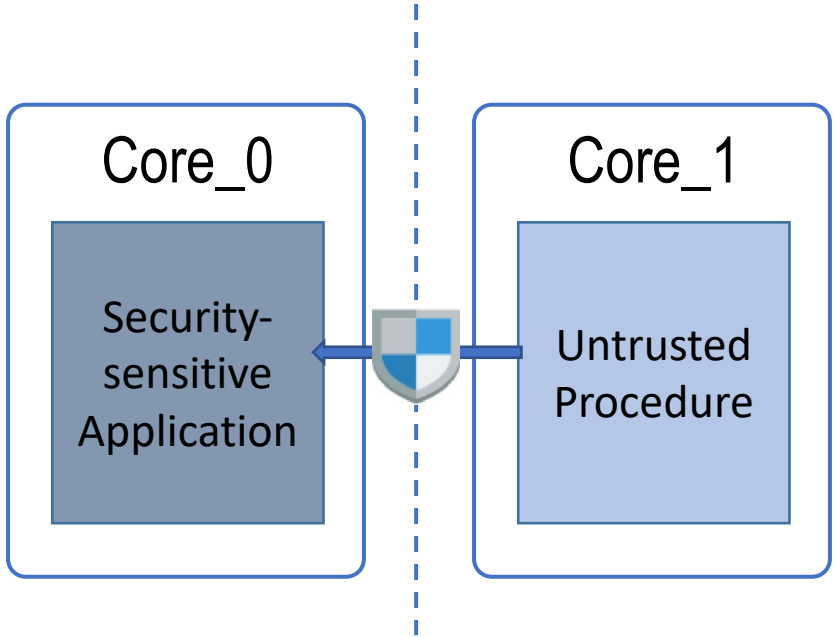
Some existing systems ignore the security of data in the cache.

Attackers can manipulate the cache to influence the protection of IEE systems.

Data Protection Model of IEE Systems

IEE systems are protected ① when they are running concurrently with untrusted procedures, ② when they are suspended or finished and ③ when they are resumed or started.

①



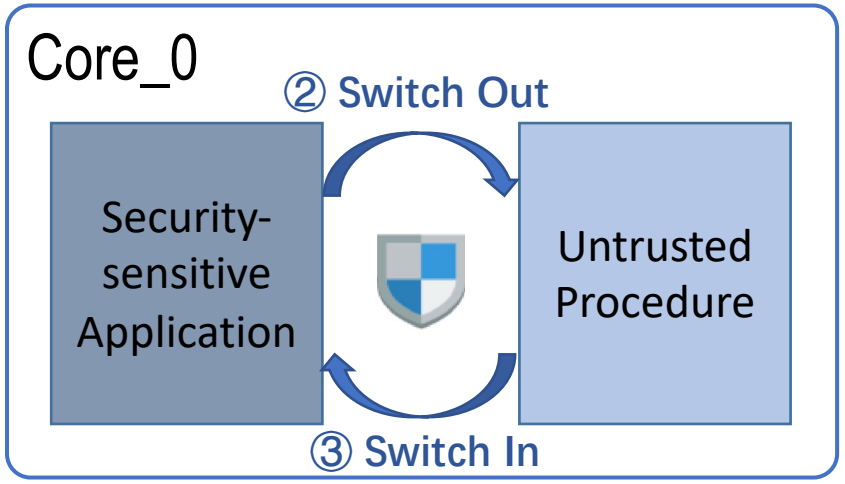
Core-isolated storage

The core-isolated storage can only be accessed by the core executing security-sensitive applications and is inaccessible to the other cores.

Data Protection Model of IEE Systems

IEE systems are protected ① when they are running concurrently with untrusted procedures, ② when they are suspended or finished and ③ when they are resumed or started.

②、③



Enforcing security measures during the context switching processes.

Preventing sensitive data leakage during switching out.

Restoring the sensitive data during switching in.

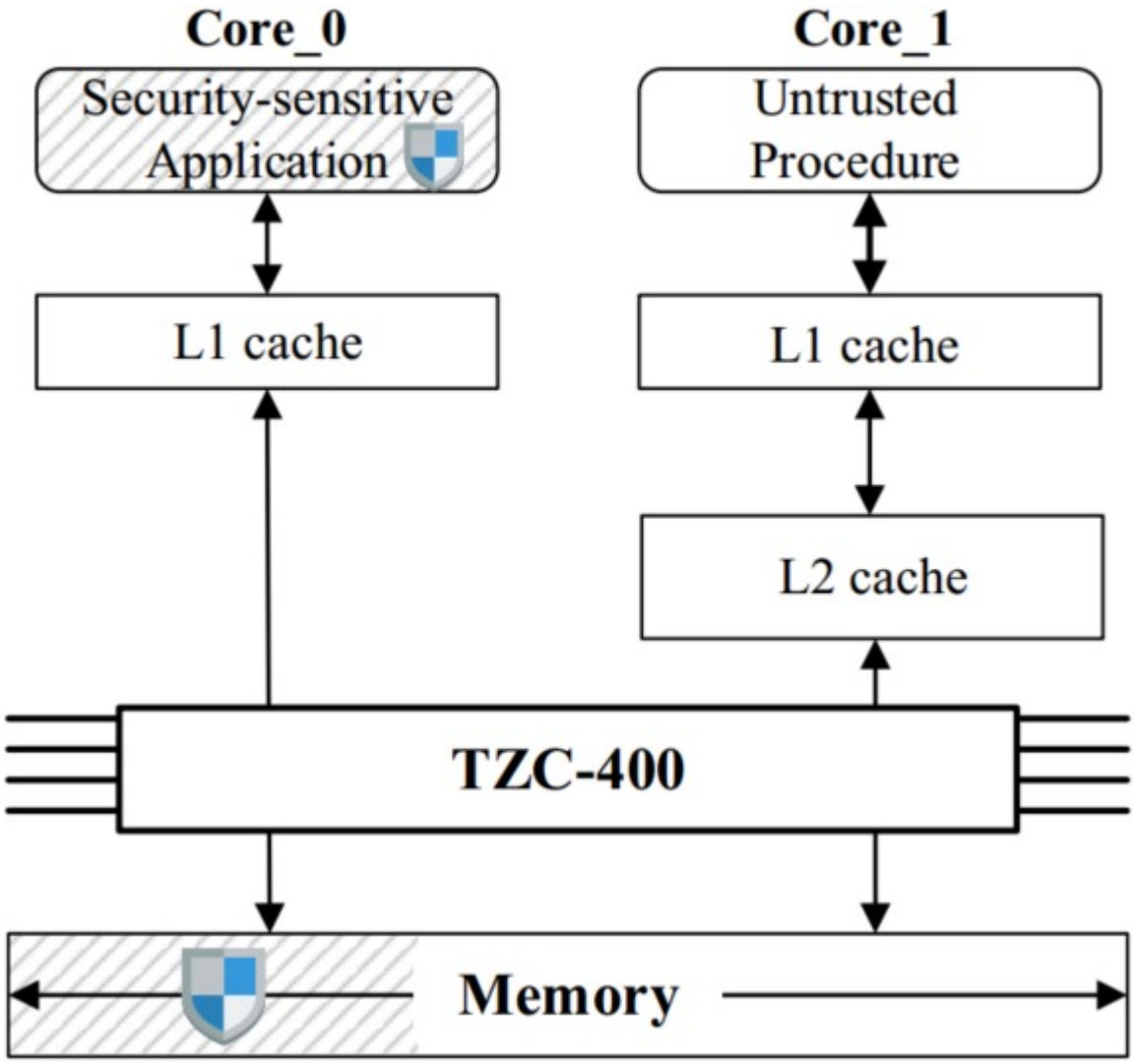
Data Protection Model of IEE Systems

- **Core-isolated storage**
  - Attack I: Manipulating data of core-isolated memory.
- **Security measures during the context switching processes**
  - Attack II: Bypassing security measures.
  - Attack III: Misusing incomplete security measures.

Data Protection Model of IEE Systems

- **Core-isolated storage**
  - Attack I: Manipulating data of core-isolated memory.
- **Security measures during the context switching processes**
  - Attack II: Bypassing security measures.
  - Attack III: Misusing incomplete security measures.

Attack I: Manipulating Data of Core-isolated Memory

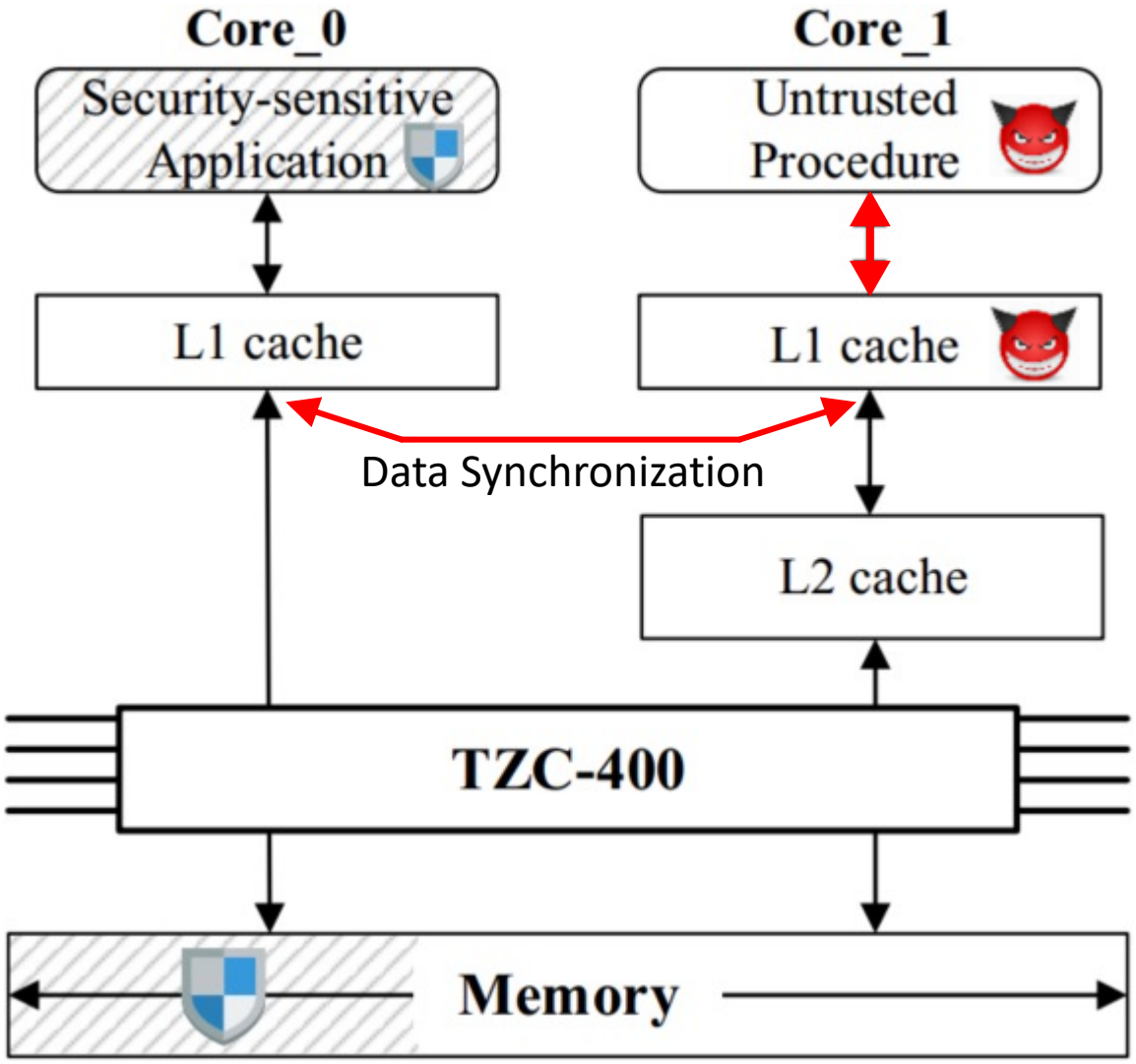


Configuration of core-isolated storage.

(e.g., SANCTUARY)

- Configuring core-isolated memory.
- Excluding the L2 shared cache.

Attack I: Manipulating Data of Core-isolated Memory



Utilizing the shareability attribute of L1 cache.

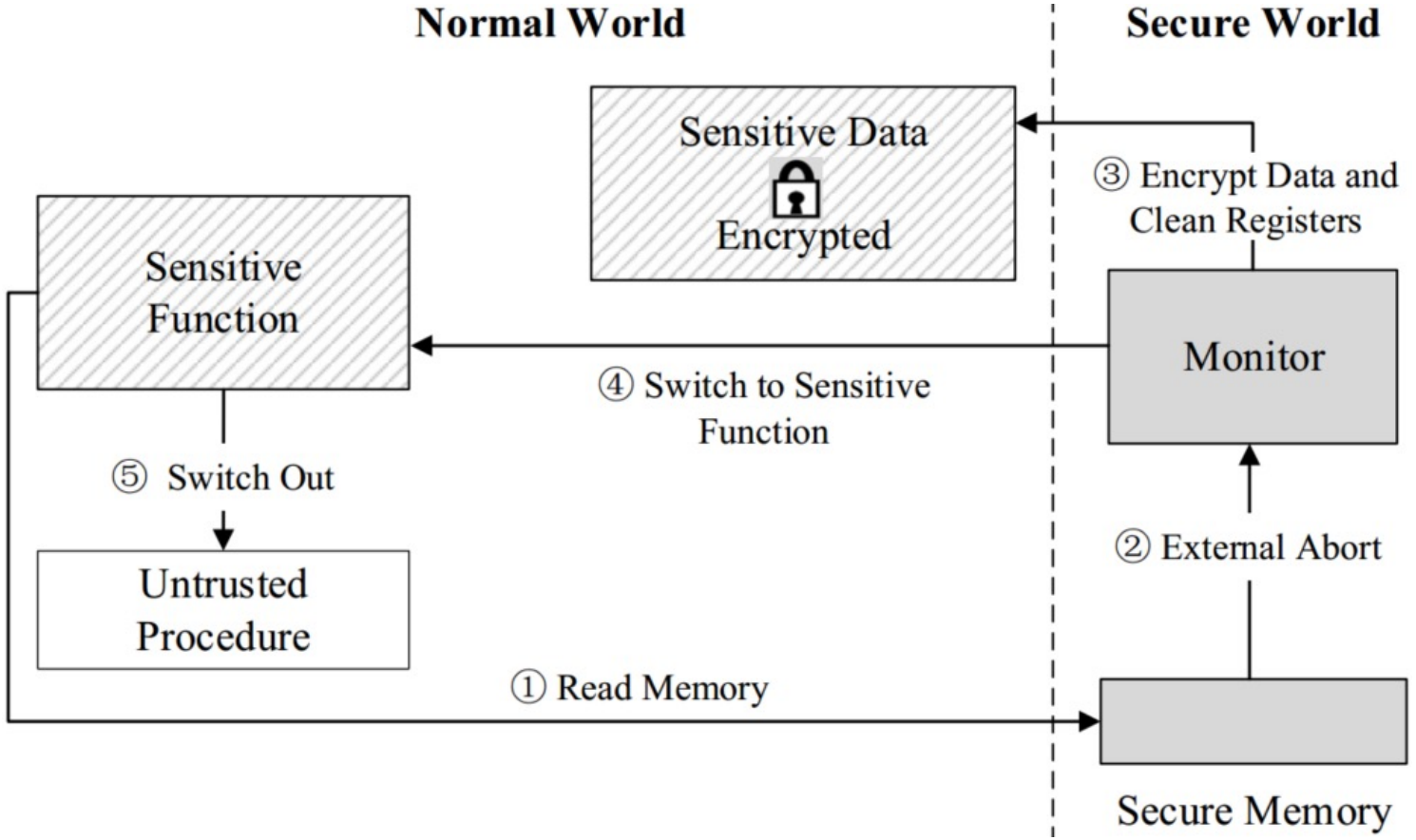
- Value coherency of L1 data cache.
- Manipulating L1 data cache to get data of core-isolated memory.

Data Protection Model of IEE Systems

- **Core-isolated storage**
  - Attack I: Manipulating data of core-isolated memory.
- **Security measures during the context switching processes**
  - Attack II: Bypassing security measures.
  - Attack III: Misusing incomplete security measures.

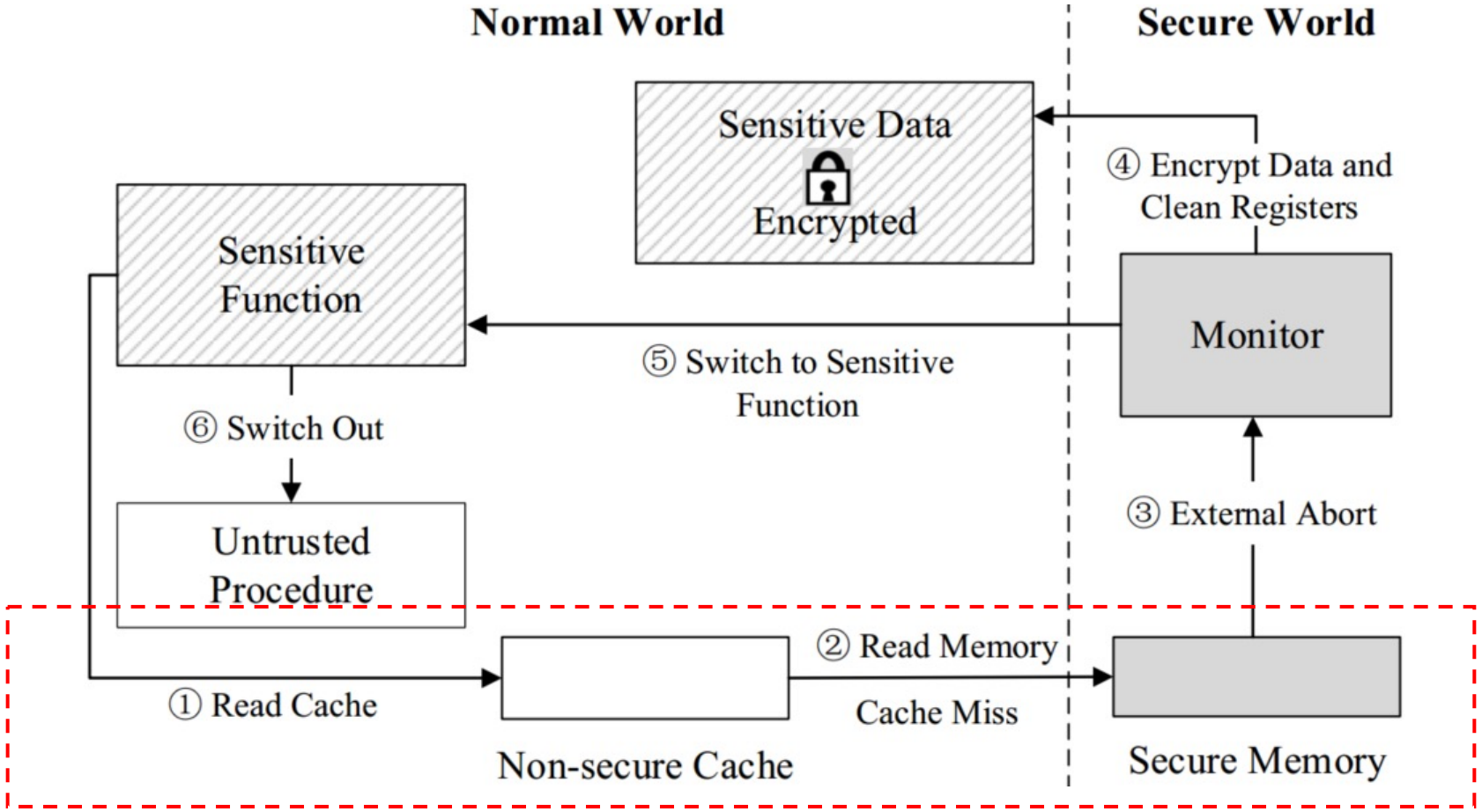
Attack II: Bypassing Security Measures

Accessing secure memory to trigger security measures when switching out (e.g., Ginseng)



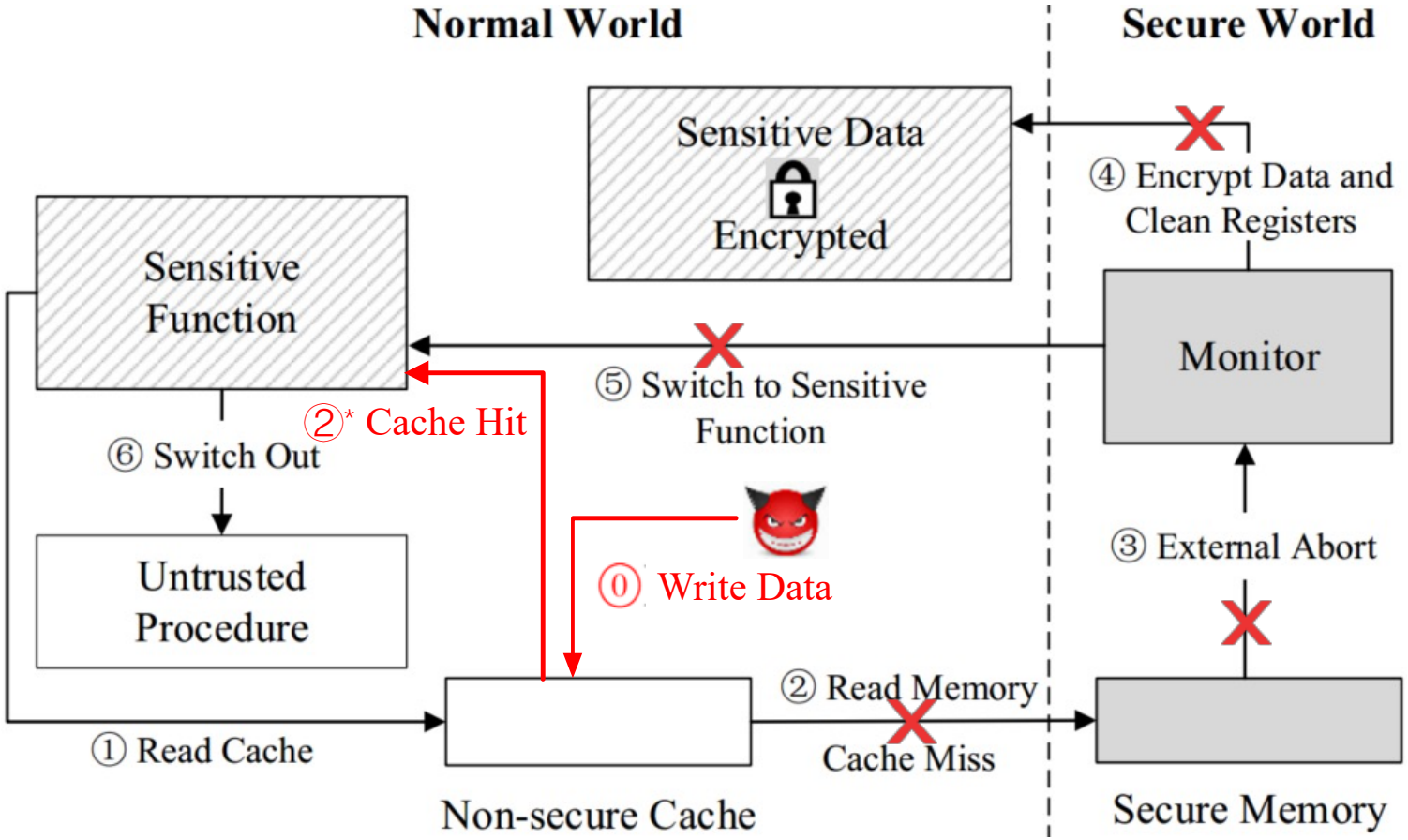
Attack II: Bypassing Security Measures

Analyzing the security measures with cache



Attack II: Bypassing Security Measures

Bypassing the security measures

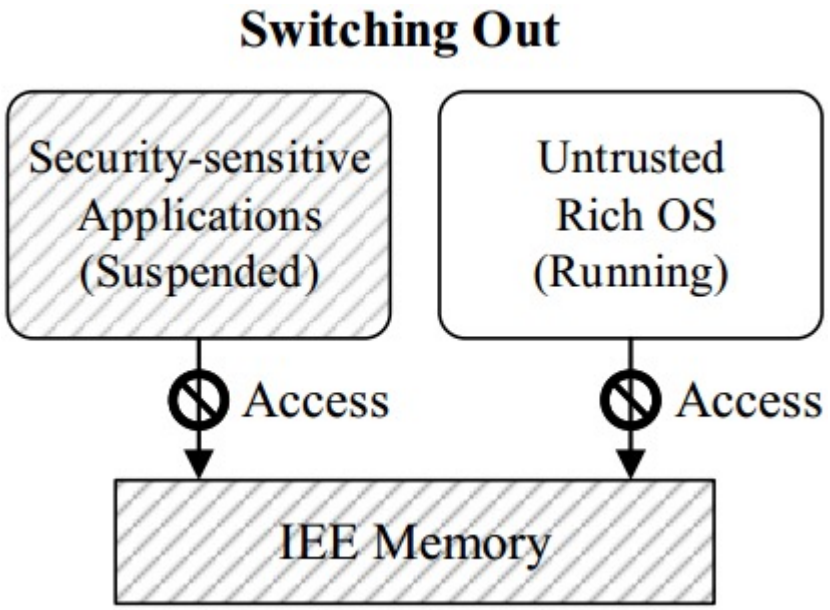


Data Protection Model of IEE Systems

- **Core-isolated storage**
  - Attack I: Manipulating data of core-isolated memory.
- **Security measures during the context switching processes**
  - Attack II: Bypassing security measures.
  - Attack III: Misusing incomplete security measures.

Attack III: Misusing Incomplete Security Measures

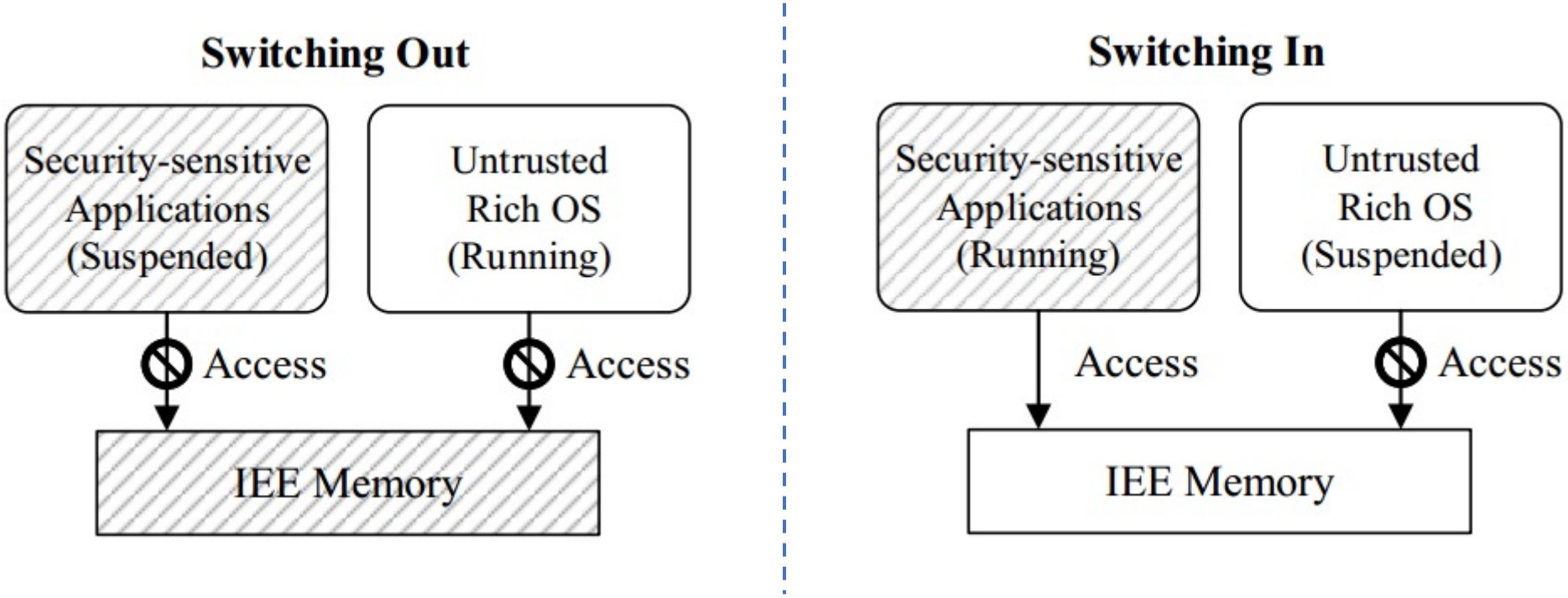
Configuring memory during the context switching processes (e.g., TrustICE)



- The switching out process configures the memory as secure.

Attack III: Misusing Incomplete Security Measures

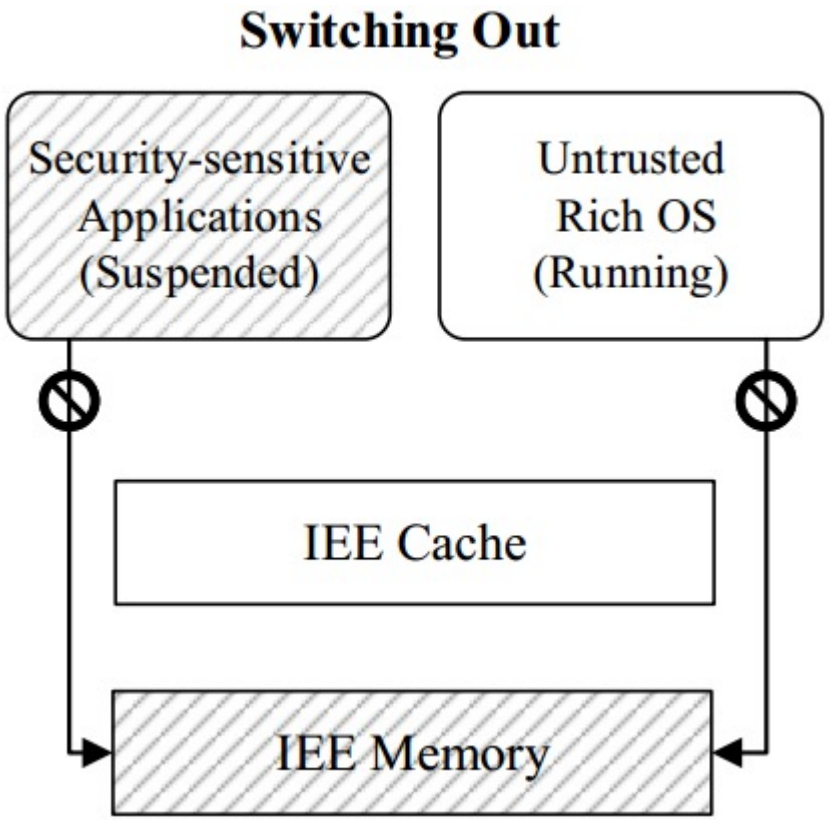
Configuring memory during the context switching processes (e.g., TrustICE)



- The switching out process configures the memory as secure.
- The switching in process configures the memory as non-secure and suspends the untrusted rich OS.

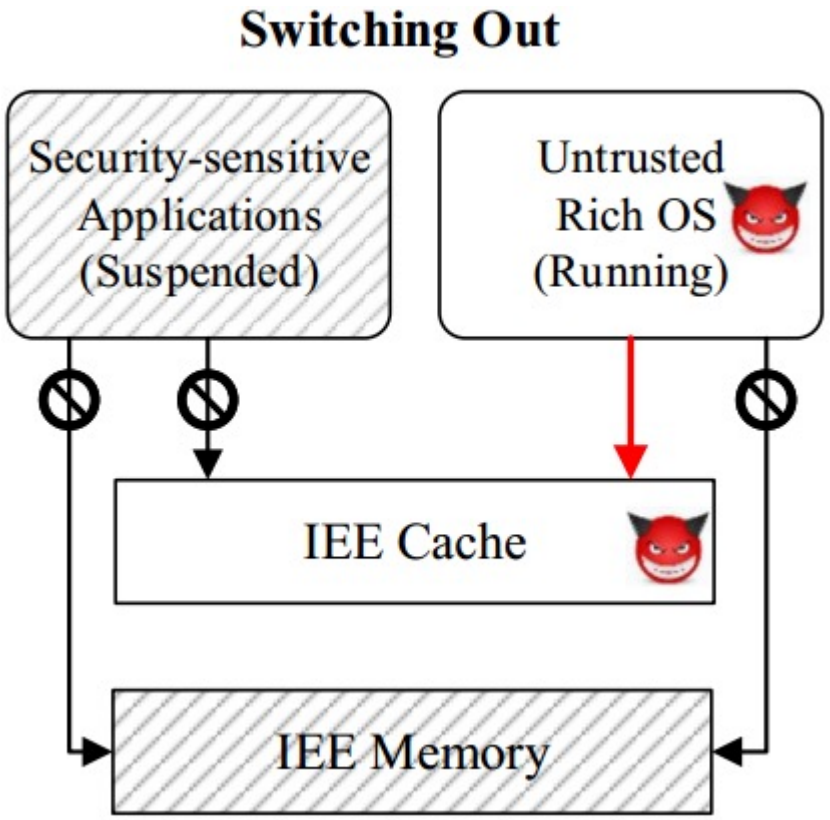
Attack III: Misusing Incomplete Security Measures

Memory configuration doesn't influence the security of cache



Attack III: Misusing Incomplete Security Measures

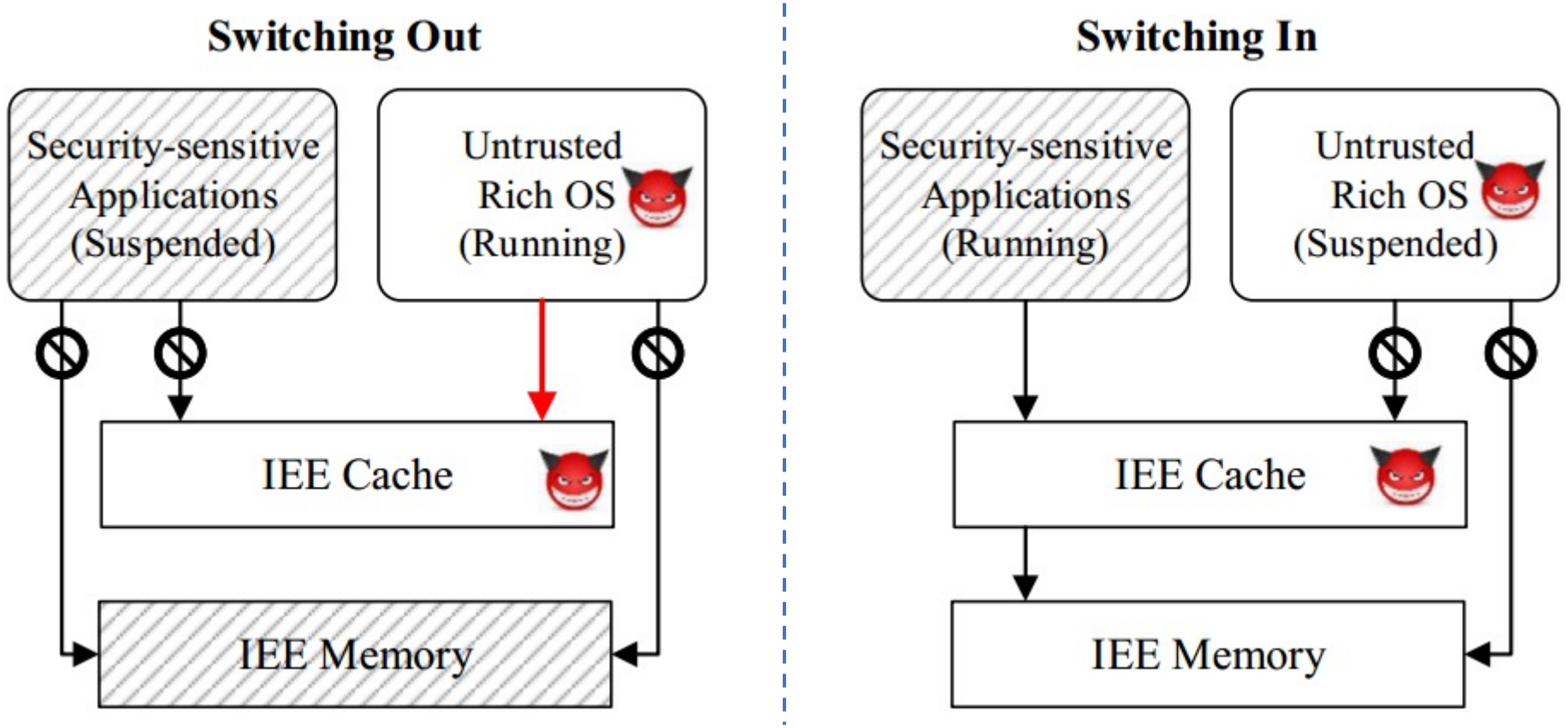
Memory configuration doesn't influence the security of cache



- Manipulating IEE cache when untrusted rich OS is running.

Attack III: Misusing Incomplete Security Measures

Memory configuration doesn't influence the security of cache



- Manipulating IEE cache when untrusted rich OS is running
- Reading polluted IEE cache when security-sensitive applications are running

Countermeasure

- **Secure cache attributes**
  - (e.g., write-through, non-shareable)
- **Cache cleaning operation**
- **Enforcing secure cache attributes**
  - Interposing all page table update operations

Conclusions

We must realize the importance of considering memory and cache together when designing IEE systems.

Thank you!

Questions ?