SoK: A Study of Using Hardware-assisted Isolated Execution Environments for Security

Fengwei Zhang

Wayne State University
Detroit, Michigan, USA

Overview Of The Talk

- Introduction
- Hardware-assisted Isolated Execution Environments (HIEEs)
- Use Cases of HIEEs
- Attacks against HIEEs
- Discussions and Conclusions

Overview Of The Talk

- Introduction
- Hardware-assisted Isolated Execution Environments (HIEEs)
- Use Cases of HIEEs
- Attacks against HIEEs
- Discussions and Conclusions

Introduction

- Isolating code execution is one of the fundamental approaches for achieving security
- Isolated execution environments
  - Software-based: Virtual machines
    - A large trusted computing base (e.g., Xen has 532K SLOC)
    - Failure to deal with hypervisor or firmware rootkits
    - Suffering from system overhead
- Hardware-assisted isolated execution environments (HIEEs)
  - Isolated execution concept: Trusted execution environment (TEE)
  - Hardware-assisted technologies
    - Excluding the hypervisors from TCB
    - Achieving a high level of privilege (i.e., hardware-level privilege)
    - Reducing performance overhead (e.g., context switches)

Overview Of The Talk

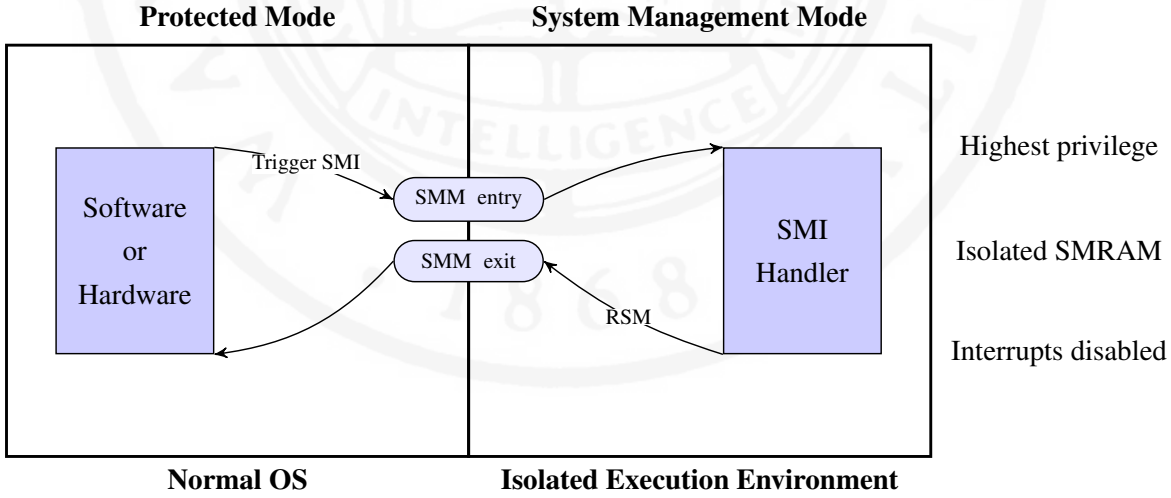
- Introduction
- Hardware-assisted Isolated Execution Environments (HIEEs)
- Use Cases of HIEEs
- Attacks against HIEEs
- Discussions and Conclusions

HIEEs

- A list of hardware-assisted isolated execution environments (HIEEs) that have been used for building security tools
  - System management mode (SMM) [24]
  - Intel management engine (ME) [36]
  - AMD platform security processor (PSP) [4]
  - Dynamic root of trust for measurements (DRTM) [52]
  - Intel software guard extension (SGX) [5, 23, 34]
  - ARM TrustZone technology [6]

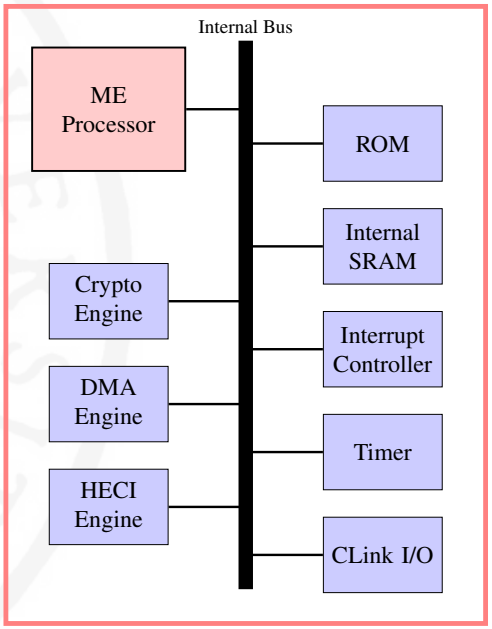
HIEE: System Management Mode

- A CPU mode similar to Real and Protected modes available on x86 architecture
- Initialized by the Basic Input/Output System (BIOS)
- Entering SMM by asserting the system management interrupt (SMI) pin
- System management RAM (SMRAM) that is inaccessible from the normal OS



HIEE: Intel Management Engine

Management Engine (ME) is a micro-computer embedded inside of all recent Intel processors; it is Introduced as an embedded processor, and Intel AMT is the first application running in ME [36]



Management Engine

Table 1: Main Hardware Components of ME

Hardware	Description
ME processor	Main master device that executes the firmware
ROM	Boot loader; cannot be modified; as the root of trust of ME
Internal SRAM	Storing the code and data at runtime
Crypto engine	Executing crypto algorithm to save the processor's cycles
DMA engine	Transmitting large amounts of data between host and ME
HECI engine	Moving small amounts of data; host can program it

HIEE: AMD Embedded Processors

- AMD secure processor [4]
  - Also called platform security processor (PSP)
  - Embedded inside of the main AMD CPU to enable running third-party applications
  - Partnership with ARM TrustZone
- System management unit (SMU) [30]
  - An embedded processor at Northbridge
  - Northbridge has been integrated into CPU
  - Responsible for a variety of system and power management tasks during boot and runtime

HIEE: Dynamic Root of Trust for Measurement

- TCG introduced DRTM, also called “late launch”, in the TPM v1.2 specification in 2005 [51, 52]

- SRTM v.s. DRTM
  - Static root of trust for measurement (SRTM) operates at boot time, DRTM allows the root of trust for measurement to be initialized at any points

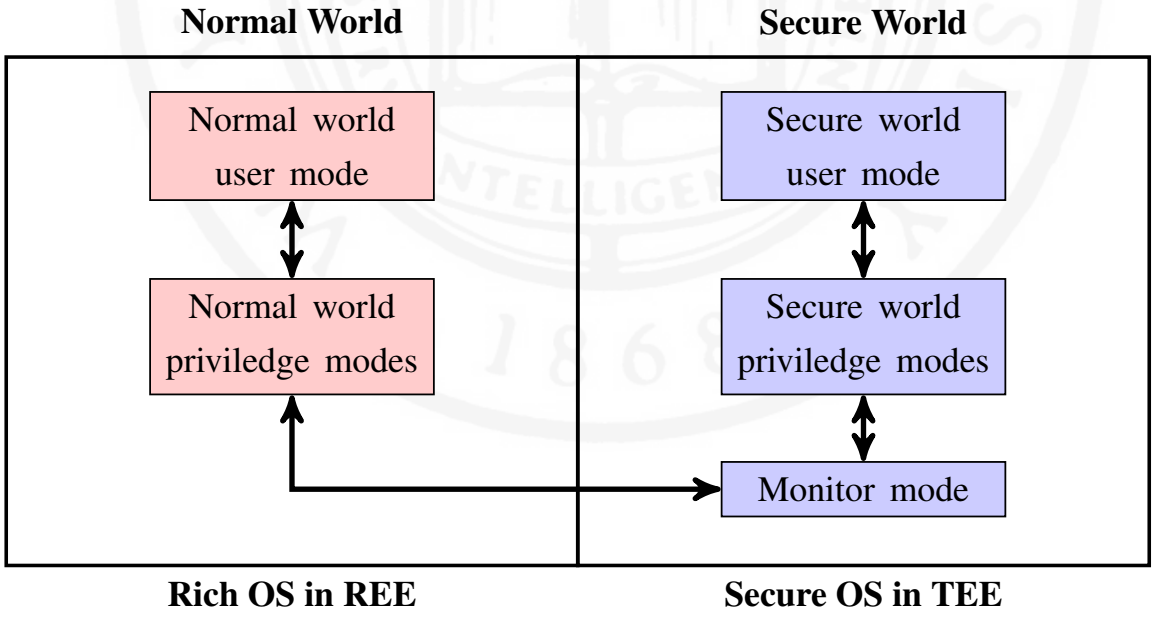
- Intel and AMD implementations
  - Intel trusted execution technology (TXT) [25]
  - AMD secure virtual machine (SVM) [2]
  - Overhead for late launch: SENTER v.s. SKINIT

HIEE: Intel Software Guard Extension

- Three introduction papers [5, 34, 23] about SGX presented at HASP 2013
- SGX is a set of instructions and mechanisms for memory accesses added to Intel architecture processors
- Allowing an user-level application to instantiate a protected container, called enclave
- Providing confidentiality and integrity even without trusting the BIOS, firmware, hypervisors, and OS
- OpenSGX [27]: An open-source platform that emulates Intel SGX at the instruction level by modifying QEMU

HIEE: ARM TrustZone

- ARM TrustZone technology is a hardware extension that creates a secure execution environment since ARMv6 [12]
- Two modes: Secure world and normal world
- Identified by the NS bit in the secure configuration register (SCR)



HIEEs

Table 2: Summary of HIEEs

	SMM	ME	PSP	DRTM	SGX	TrustZone
Timelines	~1993	~2007	~2013	~2005	~2013	~2002
Supported hardware	x86	Intel	AMD	Intel/AMD	Intel	ARM
Sharing main CPU	✓			✓	✓	✓
High privilege	✓	✓	✓			✓
Zero overhead		✓	✓			
Designed for security		✓	✓	✓	✓	✓

Overview Of The Talk

- Introduction
- Hardware-assisted Isolated Execution Environments (HIEEs)
- Use Cases of HIEEs
- Attacks against HIEEs
- Discussions and Conclusions

Use Cases of HIEEs

- System introspection
- Memory forensics
- Transparent malware analysis
- Execution sensitive workloads
- Rootkits and keyloggers

Use Case: System Introspection

- Running system introspection tools inside of HIEEs
  - Hypervisor/OS integrity checking
  - OS rootkits detection
  - Attacks detection (e.g., heap spray and heap overflows)
- SMM-based
  - Hypercheck [65], HyperGuard [41], HyperSentry [8], IOCheck [64], and Spectre [62]
- TrustZone-based
  - SPROBES [22] and TZ-RKP [7]
- DRTM-based
  - Flicker [31]

Use Case: Memory Forensics

- Using HIEEs to perform acquisition of volatile memory of a target system, and then transmit the memory contents to a remote machine for analysis

- Examples of existing systems
  - SMMDump [35] implemented by using SMM
  - TrustDump [48] used ARM TrustZone

Use Case: Transparent Malware Analysis

- Malware uses anti-debugging, anti-virtualization, anti-emulation techniques to evade traditional analysis suing virtualization or emulation technology
- Analyzing malware using HIEEs so that advanced malware can be debugged on bare metal
- Exposing the real behavior of malware with anti-debugging, anti-vm, and anti-emulation techniques
- Examples of exiting systems
  - MalT [61] using SMM
  - Other HIEEs like TrustZone and ME can be used for the same purpose

Use Case: Executing Sensitive Workloads

- Using HIEEs to run security sensitive operations
- DRTM-based
  - Flicker [31], TrustVisor [32], and Bumpy [33]
- TrustZone-based
  - TrustICE [49] and TrustOTP [47]
- SMM-based
  - SICE [9] and TrustLogin [63]
- SGX-based
  - Haven [10] and VC3 [43]

Use Case: Rootkits and Keyloggers

- Though researchers have used HIEEs for implementing defensive tools, attackers can also use them for malicious purposes due to their *high privilege* and *stealthiness*

- SMM rootkits
  - PS/2 [20] and USB [42] keyloggers
  - NSA: DEITYBOUNCE for Dell and IRONCHEF for HP Proliant servers [1]

- ME rootkits
  - Ring -3 rootkits [46, 50]

- DRTM, SGX, and TrustZone rootkits
  - We haven't seen any publicly available examples but attackers have the motivation to implement them due to their stealthiness

- HIEEs create ideal environments or infrastructures that attract attackers to implement super-powerful rootkits.

Overview Of The Talk

- Introduction
- Hardware-assisted Isolated Execution Environments (HIEEs)
- Use Cases of HIEEs
- Attacks against HIEEs
- Discussions and Conclusions

HIEE Attacks

- HIEE attacks: Bypassing the hardware protection mechanisms of HIEE isolation; not using HIEEs for malicious purposes

- SMM attacks

Table 3: Summary of SMM Attacks and Solutions

SMM Attacks	Solutions
Unlocked SMRAM [17, 20, 13]	Set D_LCK bit
SMRAM reclaiming [41]	Lock remapping and TOLUD registers
Cache poisoning [58, 19]	SMRR
Graphics aperture [18]	Lock TOLUD
TSEG location [18]	Lock TSEG base
Call/fetch outside of SMRAM [18, 60]	No call/fetch outside of SMRAM

HIEE Attacks (cont'd)

- ME attacks
  - In 2009, Tereshkin and Wojtczuk [50] demonstrated that they can implement ring -3 rootkits in ME by injecting the malicious code into the Intel AMT
  - DAGGER [46] bypasses the ME isolation using a similar technique in [50]
- DRTM attacks
  - Wojtczuk and Rutkowska from Invisible Things Lab demonstrate several attacks [57, 56, 59] against Intel TXT
- TrustZone attacks
  - Di [44] found vulnerabilities that are able to execute arbitrarily code in secure world using a user-level application in normal world on Huawei HiSilicon devices

HIEE Attacks (cont'd)

- SGX attacks
  - Cache timing attacks and software side-channel attacks including using performance counters from the study published by Costan and Devadas [15]
- Unclear if ME firmware is malicious
  - SGX for desktop-environments needs to establish a secure channel between I/O devices (e.g., key- board and video display) and an enclave to prevent sensitive data leakage [38, 27]
  - Protected Audio Video Path (PVAP) technology can securely display video frames and play audio to users; Identity Protection Technology (IPT) provides security features including Protected Transaction Display (e.g., entering a PIN by an user)
  - SGX needs Enhanced Privacy Identification (EPID) support for remote attestation [27]
  - PVAP, IPT, EPID are realized by ME [36]

Overview Of The Talk

- Introduction
- Hardware-assisted Isolated Execution Environments (HIEEs)
- Use Cases of HIEEs
- Attacks against HIEEs
- Discussions and Conclusions

Challenges of Using HIEEs for Security

- Ensuing trusted switching path
  - HIEE-based systems assume attackers have ring 0 privilege, so attackers can intercept the switching and create a fake one
  - Ad-hoc solutions using an external smartphone [33], keyboard LED lights [63], LED power lights [49]
  - Building a generic and user-friendly trusted path mechanism form HIEE-based system is an open research problem

- Verifying the trustworthiness of hardware
  - HIEE-based systems depend on the trustworthiness of hardware
  - Assuming hardware features are bug-free (e.g., isolation is graduated)
  - Hardware vendors tend not to release implementation details
  - How to reliably evaluate the trustworthiness of these mysterious hardware security technologies (e.g., ME)

Conclusions

- Main contributions of this SoK paper are:
  - Presenting a thorough study of six HIEEs including SMM, Intel ME, AMD PSP, DRTM, Intel SGX, and ARM TrustZone
  - Exploring both the defensive and offensive use scenarios of HIEEs and describe them with the state-of-the-art systems
  - Discussing all attacks against the computing environment of each HIEE (e.g., bypassing the isolation) and some mitigations

References

The reference numbers in the slides are the ones shown in the Section 8 of the paper.