Dr. JAEHYUK LEE, Ph.D.

470-861-6020 https://ruach.github.io jhyuk.lee@outlook.com Jaehyuk Lee

RESEARCH INTEREST

I am interested in computer systems security, especially in: software/hardware vulnerability research, fuzzing, exploit mitigations, hardware/software codesign for security, Trusted Execution Environment (TEE), secure IO design for TEE, side-channel attacks and its mitigations, and mobile security.

EDUCATION

Institution	Degree	Field of Study	Dates	GPA
Korea Advanced Institute of Science and Technology	*Ph.D.*	Information Security	Aug 2015 - Feb 2021	3.95
Korea Advanced Institute of Science and Technology	*M.S.*	Information Security	Sep 2013 - Aug 2015	4.1
Handong Global University	*B.S.*	Computer Science	Mar 2010 - Aug 2013	4.2

TECHNICAL SKILLS

Software: C, C++, Rust, Scala, Java, Python, CUDA, Fuzzing, GDB, Ghidra, IDA pro, Daikon, Linux Kernel, KVM, QEMU, ARM/X86 Assembly, Trusted Firmware (TF-A), EDK2, TF-RMM, TDX Module, SGX SDK

Hardware: GEM5 (Architecture Simulator), Processor Pipeline, Cache, TLB, Microcode, HW Side Channel, Verilog, System Verilog, Chisel (RISC-V), Intel SGX & TDX, ARM CCA & TrustZone, IOMMU, PCIe, USB

WORK EXPERIENCE

Postdoc Researcher: Georgia Institute of Technology, Atlanta, GA **Jan 2022 – Current**

- Developed an automated fuzzer selection tool that dynamically selects fuzzer(s) based on runtime analysis [2]. Anyone having no knowledge of fuzzers can simply utilize it for automate vulnerability discovery, irrespective of the target
- Experienced in vulnerability research on Intel and ARM's latest hardware feature designed for confidential VM (TDX & CCA). Identified new side-channel issues related to resource allocation in the construction of confidential VMs
- Implemented a secure and swift IO path between confidential Virtual Machines (VM) and peripherals such as GPUs by leveraging robust isolation guarantees of IOMMU (SMMU) and ARM CCA, avoiding encryption overhead [10]
- Volunteered as an IT admin, possessing hands-on experience on IPMI, python & shell scripting for multiple server management (30 servers), and troubleshooting software/hardware issues.

Postdoc Researcher: Korea Advanced Institute of Science and Technology, Korea **Mar 2021 – Dec 2021**

- Implemented novel cache side-channel attack on X86 and Apple M1 chips, unveiling the memory access pattern without evicting the victim's cache entries and completely bypassing side-channel defenses monitoring cache evictions [9]
- Actively engaged in mentoring and educating Ph.D. students in large group settings (20 students)

Research Intern: Georgia Institute of Technology, Atlanta, GA **Nov 2019 – May 2020**

- To address side-channel attacks, I implemented a hardware framework enabling user space applications to subscribe to micro-architecture events, including cache and TLB evictions, and demonstrated its effectiveness on GEM5 simulator [1]

Research Intern: Microsoft Research, Redmond, WA **May 2016 – August 2016**

- Conducted vulnerability research about the potential impact of a memory corruption within confidential computing (Intel SGX). Implemented practical ROP attack against encrypted binary, compromising its confidentiality guarantee [3]
- Developed secure Content Delivery Network (CDN) to safeguard user data residing on the server, allowing industry companies, to offload user data to a third-party CDN server without having to place full trust in the cloud provider

PUBLICATION & PATENT (546 citations)

top-tier conferences (USENIX Security, NDSS) and Journal (TDSC)

[1] **SENSE: Enhancing Microarchitectural Awareness via Subscription-Based Notification** | Paper **NDSS 2024**

- SENSE grants TEEs access to micro-architectural information, facilitating preemptive protection against side-channel attacks targeting caches and TLBs. SENSE is implemented using the GEM5 system-level processor simulator.
- Modified GEM5 out-of-order processor pipeline to redirect its execution to the pre-registered handler by implementing micro-code. I modified cache and TLB design to introduce additional bits tracking events with minimal cost.

[2] **autofz: Automated Fuzzer Composition at Runtime** | Paper | Hacker News **USENIX 2023**

- Autofz alleviates the need for users to manually configure the best fuzzer for each specific target. Similar to AutoML, it automatically and adaptively configures best-performing fuzzers at runtime and achieves the best performance.
- Experienced in multiple fuzzers such as LearnAFL, RedQueen, MOpt, AFLFast, AFL, LAF-Intel, Angora, FairFuzz, Radamsa, QSYM.

[3] **Hacking in Darkness: Return-oriented Programming against Secure Enclaves** | Paper | Talk | Demo **USENIX 2017**
- Dark-ROP demonstrates practical exploitation of memory corruption vulnerabilities within enclaves through Return-Oriented Programming (ROP), utilizing special oracles induced by the design of Intel SGX.
- Located ROP gadgets by leveraging the side effects of newly introduced instructions for Intel SGX, all without prior knowledge of the binary details of the target process. Implemented full ROP chain and demonstrated attack.

[4] **SGX-Bomb: Locking Down the Processor via Rowhammer Attack** | Paper | Hacker News | Demo **SysTEX 2017**
- SGX-Bomb showcases how Rowhammer attacks can jeopardize enclave integrity, posing a risk of denial-of-service (DoS) scenarios, which is especially concerning for cloud providers that host untrusted enclave programs.
- Implemented kernel driver to reverse engineer mapping between physical address to memory bank for locating two adjacent rows. Demonstrated rowhammer attack inside Intel SGX with X86 assembly and C++ coding.

[5] **PrivateZone: Providing a Private Execution Environment using ARM TrustZone** | Paper **TDSC 2016**
- PrivateZone provides a secure TEE for developers to deploy and execute user processes on mobile platforms. It delivers assurances similar to TrustZone without imposing additional costs on developers to deploy applications
- Implemented stage-2 page table for Privatezone and developed the logic for context switching, effectively managing the transition between trusted stage-2 page tables and untrusted ones. Experienced in ARM assembly coding.

[6] **OpenSGX: An Open Platform for SGX Research** | Paper | Wikipedia: SGX **NDSS 2015**
- OpenSGX enables Intel SGX exploration with instruction-level emulation, addressing limitations in TEE software development and offering a comprehensive ecosystem, including an OS, enclave handling, user libraries, and debugging
- Implemented X86 assembly facilitating context switching within the SGX environment, enabling the transition between entering and exiting the SGX. Developed stub-call allowing SGX to invoke system calls through a customized libc

[7] **A Rule Extraction Method Using Relevance Factor for FMM Neural Networks** | Paper **KTSDE 2013**
- We enhanced Fuzzy Min-Max (FMM) neural network by incorporating the frequency of features in training. Considering the relevance factor between features and pattern classes, we address ambiguity without overlapping and contraction.

[8] **Survey On DeFi Users' Perception of Security Breaches and Countermeasures** | Paper **Under Review**
- Investigated DeFi users' security perceptions and commonly adopted practices, and how those affected by previous scams or hacks (DeFi victims) respond and try to recover their losses.
- Conducted qualitative analysis through coding 14 interviewers' responses using Dedoose.

[9] **PRIME+RETOUCH: When Cache is Locked and Leaked** | Paper **Ready to submit**
- PRIME+RETOUCH attack circumvents cache-based side-channel defense monitoring eviction events by utilizing cache replacement policy metadata. The attack is demonstrated on both Intel x86 and Apple M1 architectures.
- Conducted reverse engineering to analyze the cache eviction policy of the most recent X86 processor and the Apple M1 chip. Developed PoC in X86 and ARM assembly, employing pointer chasing along with memory barriers

[10] **Portal: Secure IO for ARM CCA** | Paper **Ready to submit**
- TEEs designed for confidential VMs face limitations in directly interfacing with peripherals, necessitating encryption and decryption overhead when transmitting data to devices. Portal addresses this concern by implementing access control
- Redesign IOMMU subsystem within the Linux kernel and migrate the resource allocation component, including SMMU page table management, to TF-RMM. Introduce RMI interface calls between KVM and TF-RMM to enable confidential VM to have a dedicated, isolated DMA region.

[11] **Apparatus and method for open and private IoT gateway using Intel SGX** | Patent **KR102162018B1**
- The invention introduces a secure IoT gateway design, utilizing Intel SGX for protocol conversion and encrypted data transfer. It includes enclave designs for cross-platform communication, encrypted key management, and secure updates.

SOFTWARE ARTIFACTS

Autofz:	https://github.com/sslab-gatech/autofz	7	★54
OpenSGX:	https://github.com/sslab-gatech/opensgx	83	★276
SGXBomb:	https://github.com/sslab-gatech/sgx-bomb	5	★14

SELECTED PROJECTS

Enabling Rack-scale Disaggregation-aware Confidential Computing via Intel IPUs Intel Lab (On-going)
- Modern data centers disaggregate hardware for efficiency, but TEE lacks seamless communication with disaggregated resources. This proposal seeks to enhance TEE using Intel IPUs to establish secure channels within and across racks.

Exploring Possible Vulnerabilities in Intel TDX Intel Lab (On-going)
- TEEs for confidential VMs, create a unique threat model where trust cannot be placed in the host VMM and OS while still being essential for resource management. We investigate potential side channels due to this new threat model.

Establishing Secure and Swift IO Path for Confidential Computing Samsung Research (On-going)
- Develop a secure and efficient way for offloading LLM models on a mobile device with hardware-enforced access control. It achieves better performance and reduced energy consumption by eliminating the need for cryptographic operation

Research for Virtual Machine Introspection (VMI) for Linux KVM Nation Security Research Institute

- Developed VMI library for Linux KVM which allows an administrator to easily inspect Virtual Machine (VM) status (e.g., memory and registers). I showcased its effectiveness in detecting kernel malware.

OS Kernel Behavior Modeling for Introspection Agency for Defense Development

- To enforce Control Flow Integrity (CFI), it is essential to have an oracle that identifies which data can be considered benign during kernel execution. I utilized the Daikon static analysis tool to extract invariants from Linux kernel.

Introspection Platform for Trusted Execution Environment (TEE) National Security Research Institute (NSRI)

- Due to the confidentiality of TEE, even the system admin can't monitor the app in real-time, potentially allowing misuse as malware. To mitigate this risk, we've introduced a compiler-based patching generating a runtime status report.

PROFESSIONAL ACTIVITIES

External Reviewer

IEEE Symposium on Security and Privacy (Oakland)	*2016, 2017, 2018, 2020, 2021, 2022*
USENIX Security Symposium (Security)	*2015, 2021, 2022, 2023, 2024*
ACM Conference on Computer and Communications Security (CCS)	*2015, 2017, 2018, 2019, 2023*
Network and Distributed System Security Symposium (NDSS)	*2019, 2020, 2021*
ACM ASIA Conference on Computer and Communications Security (ASIACCS)	*2015, 2016, 2018*
ACM Symposium on Operating Systems Principles (SOSP)	*2021*
ACM International World Wide Web Conference (WWW)	*2018*

INVITED TALK

A Novel Cache Side Channel Attack without Attacker Eviction

Presented at Intel Side Channel Academic Program (SCAP) Virtual, Sep 2020

Hacking in Darkness: Return-oriented Programming against Secure Enclaves

Presented at the 26th Usenix Security | Presentation Link Vancouver, BC, Canada, Aug 2017

Presented at KimchiCon | Presentation Link Daejeon, South Korea, Aug 2017

Tutorial: Trusted Execution Environment: TrustZone, ITP and SGX

Korea institute of information security and cryptology Seoul, South Korea, Apr 2015

THESIS

Unintended Consequences of System Design: Unpremeditated Usage of Benign System Components Devastates Security Ph.D. Thesis.

Detecting Emulated Environment: Exploiting Behavioral Discrepancies In QEMU M.S. Thesis.