BYEONGKYU HAN

byeongkyuhan@kookmin.ac.kr | Han-16.github.io | linkedin.com/in/byeongkyuhan | github.com/Han-16

Research Interests

My research focuses on leveraging applied cryptography to address real-world security and privacy issues.
I am especially interested in designing efficient zero-knowledge proof systems and advancing privacy-preserving AI.

Education

Kookmin University, Seoul, South Korea Mar 2024 – Present
M.S. in Electronics Engineering (Advisor: Jihye Kim, Hyunok Oh)

Kookmin University, Seoul, South Korea Mar 2019 – Feb 2024
B.B.A in AI, Big Data, and Management; Minor in Computer Science

Experience

Kookmin University & Hanyang University Sep 2023 - Feb 2024
Joint Research Intern — IT Security and Privacy & Security and Privacy Labs

- Built a core knowledge in modern cryptography through Katz's *Advanced Topics in Cryptography*
- Developed a solid foundation of zk-SNARKs through the *ZKP-MOOC*
- Designed and implemented a privacy-preserving Verifiable Credential system as a PoC

Publications and Manuscripts

[1] **Aegis: Scalable Privacy-preserving CBDC Framework with Dynamic Proof of Liabilities**
Gweonho Jeong, Jaewoong Lee*, Minhae Kim*, **Byeongkyu Han***, Jihye Kim, Hyunok Oh
[*Under Review*], [*paper*], [*code*] (** Equal contribution*)

Projects

Efficient Aggregation of Group Elements for zk-SNARK Circuits Sep 2025 - Present

- Proposed an efficient proof method for aggregating group elements in zk-SNARK circuit.
- Replaced naive MSM with a random permutation and small prime based accumulation scheme.
- Refined the Pianist protocol to shrink the CRS from $O(N)$ to $O(\frac{N}{M})$ (N: circuit size; M: # of machines).
- Applied the proposed Efficient Aggregate Scheme, mitigating the overhead from CRS reduction.
- Achieved prover work $O(M \log M)$ and verifier cost and proof size $O(1)$, as in Pianist.

A Privacy-Preserving STO system with Real-Time PoL Mar 2024 – Oct 2024

- Proposed a STO system providing privacy preserving, near-real-time PoL, and high-throughput trading.
- Extended **Aegis** to STO and designed the system to comply with Korean regulatory requirements.
- Achieved proof generation of 0.5s for 128 trades; reached 2,000 TPS through system optimizations.
- This system evolved into Fineapple by zkrypto
- Slides: [PDF]

Verifiable Voting Jun 2024 - Aug 2024

- Conducted as part of the Ethereum PSE Core Contribution Program 2024
- Implemented a Semaphore style voting circuit using Merkle tree membership and Groth16.
- Designed tally verification by integrating a Turbo Plonk custom gate with KZG.
- Achieved $O(1)$ verification time for both vote proofs and the final tally.
- Slides: [PDF]

Activity

Ethereum Core Contribution Program (Mentor) Jun 2025 - Aug 2025

ZK Education Team (ZKET), Ethereum Foundation

- Mentored participants in research and development of privacy and scaling technologies.
- Conducted sessions reviewing foundational papers on Groth16, KZG, PLONK and FRI.
- Facilitated special sessions on Circle STARK, FHE, and Privacy-Preserving AI.
- Provided hands-on sessions in building zk-SNARK circuits using Noir, Cairo, and Halo2.

Ethereum Core Contribution Program (Participant) Jun 2024 - Aug 2024

Privacy and Scaling Explorations (PSE), Ethereum Foundation

- Participated in an 8-week study with the PSE team at the Ethereum Foundation.
- Studied and analyzed foundational papers on Groth16, PLONK, and FRI.
- Implemented a privacy-preserving, verifiable voting system ensuring vote and tally integrity.

Honors and Awards

University Scholarships 2024 – 2025

- Professor-Nominated Merit Scholarship ($9,500 total for 4 semesters)

Special Prize, National Cryptography Paper Competition 2025

Korea Cryptography Forum, Korea Institute of Information Security & Cryptology, South Korea

- Category: Application and Utilization of Cryptographic Technologies.
- Related Manuscripts: [1]

Teaching Assistant

Embedded Systems for Next Generation Communications Fall 2025

- Course on cryptography using Katz and Lindell's *Introduction to Modern Cryptography*.

Data Structures Fall 2025

- Course on data structures in C: arrays, lists, stacks/queues, trees, and graphs.

Introduction to Programming Fall 2025

- Course on basic C language syntax.

Object-Oriented Programming Spring 2024

- Course on C++ OOP: classes, inheritance, core principles.

Programming Language Spring 2024

- Course on basic C language syntax.