

Jinyeong Seo

Email: jinyeong.seo@protonmail.com

Homepage: <https://jin-yeong-seo.github.io/>

Overview

I am a Ph.D. student at Seoul National University, advised by Prof. Yongsoo Song. My research interest lies in (but is not limited to) the practical instantiation of cryptographic protocols using techniques from lattice-based cryptography. Specifically, my recent research focuses on improving the performance of lattice-based proof systems and homomorphic encryption schemes. I also have broad interests in the theoretical foundations of cryptographic proofs.

Education

Seoul National University	Seoul, South Korea
Ph.D. in Computer Science	Mar. 2022 – Present
Advisor: Prof. Yongsoo Song	
 KAIST	 Daejeon, South Korea
B.S. in Mathematical Science	Mar. 2016 – Aug. 2021
(double major: computer science)	

Publications

Authors are listed in alphabetical order by last name, unless an asterisk(*) is indicated.

Conferences

[C07] **Simpler and faster BFV Bootstrapping for Arbitrary Plaintext Modulus from CKKS**

Jaehyung Kim, Jinyeong Seo, Yongsoo Song
ACM CCS 2024

[C06] **Concretely Efficient Lattice-based Polynomial Commitment from Standard Assumptions**

Intak Hwang, Jinyeong Seo, Yongsoo Song.
CRYPTO 2024

[C05] **Optimizing HE operations via Level-aware Key-switching Framework**

Intak Hwang, Jinyeong Seo, Yongsoo Song.
WAHC 2023

[C04] **Asymptotically faster multi-key homomorphic encryption from homomorphic gadget decomposition**

Taechan Kim, Hyesun Kwak, Dongwon Lee, Jinyeong Seo, Yongsoo Song.
ACM CCS 2023

[C03] **Toward Practical Lattice-based Proof of Knowledge from Hint-MLWE**

Duhyeong Kim, Dongwon Lee, Jinyeong Seo, Yongsoo Song.

CRYPTO 2023

[C02] **Accelerating HE Operations from Key Decomposition Technique**

Miran Kim, Dongwon Lee, Jinyeong Seo, Yongsoo Song.

CRYPTO 2023

[C01] **Faster TFHE Bootstrapping with Block Binary Keys**

Changmin Lee, Seonhong Min, Jinyeong Seo, Yongsoo Song.

ACM ASIACCS 2023

Journals

[J01] ***HEaN-STAT: a privacy-preserving statistical analysis toolkit for large-scale numerical, ordinal, and categorical data**

Younho Lee, Jinyeong Seo, Yujin Nam, Jiseok Chae, Jung Hee Cheon

IEEE TDSC 2023

Preprints

[P02] **Practical Zero-Knowledge PIOP for Public Key and Ciphertext Generation in (Multi-Group) Homomorphic Encryption**

Intak Hwang, Hyeonbum Lee, Jinyeong Seo, Yongsoo Song

[P01] **TopGear 2.0: Accelerated Authenticated Matrix Triple Generation with Scalable Prime Fields via Optimized HE Packing**

Hyunho Cha, Intak Hwang, Seonhong Min, Jinyeong Seo, Yongsoo Song

Experiences

CryptoLab Inc.

Seoul, South Korea

- Researcher

Sep. 2019 – Mar. 2020

- Intern

Jun. 2019 – Aug. 2019

- Developed HEaN-STAT, homomorphic encryption-based statistical analysis toolkit.

eWBM Inc.

Seoul, South Korea

- Intern

Jun. 2018 – Aug. 2018

- Developed ECDH PKI protocols for secure communication on LoRa devices.

Presentations

Simpler and faster BFV Bootstrapping for Arbitrary Plaintext Modulus from CKKS

Oct. 2024

ACM CCS 2024

Concretely Efficient Lattice-based Polynomial Commitment from Standard Assumptions

Aug. 2024

CRYPTO 2024

	Practical Lattice-based Private Stream Aggregation and Application to Federated Learning		Aug. 2023
	<i>The 5th Privacy-Preserving Machine Learning Workshop 2023</i>		
Honors & Awards	Korea Cryptography Contest		Oct. 2024
	2nd Place (\$3, 000)	National Security Research Institute	
	Student Travel Grants		Oct. 2024
	Travel Grant (\$1,000)	ACM CCS 2024	
	Korea Cryptography Contest		Oct. 2023
	1st Place (\$10, 000)	National Security Research Institute	
	29th Samsung Humantech Paper Award		Feb. 2023
	Silver Award (\$7, 000)	Samsung Electronics	
	Korea Cryptography Contest		Oct. 2022
	3rd Place (\$2, 000)	National Security Research Institute	
GitHub Repositories	https://github.com/SNUCP/level-aware-ksw	PoC Implementation of [C05]	
	https://github.com/SNUCP/snu-mghe	PoC Implementation of [C04]	
	https://github.com/SNUCP/fast-ksw	PoC Implementation of [C02]	
	https://github.com/SNUCP/blockkey-tfhe	PoC Implementation of [C01]	
Skills	Programming : C, C++, Go, Python		