KIJIN AN

https://kjproj84.github.io. kijin.an@gmail.com. Github: kjproj84. +82-010-3635-7845.

(a) Education

2015.8-2021.5: Ph.D., **Virginia Tech**, Computer Science, Blacksburg, VA
Advisor: Prof. Eli Tilevich, https://people.cs.vt.edu/~tilevich

2007.3–2009.2: M.S., **POSTECH**, Computer and Communication, Pohang, South Korea
Advisor: Prof. Hwangjun Song, http://mcnl.postech.ac.kr/professor

2003.3–2007.2: B.E., **University of Seoul**, Electrical and Computer Engineering, Seoul, South Korea

(b) Dissertation Topic and Interests

Research Interest: *Software Engineering, Refactoring, Web-based Systems, Code smell detection & Refactoring Recommendation, Machine learning, Distributed Systems, Software Security*

Dissertation Topic: I studied Software Engineering during my Ph.D., especially refactoring techniques for distributed apps. My approach has a novelty in that it creates a centralized variant of the distributed app to facilitate re-engineering. In other word, this variant works as a good proxy of the original version, thus perfective or architectural refactoring tasks could be applied to this version. My idea solved non-trivial tasks for evolving distributed apps: localizing bugs, optimizing distribution granularity, and replicating a subset of distributed apps into different machines for enabling edge-based/centralized computing. My approach is realized by using the latest software engineering techniques: fuzzing/checkpointing the HTTP executions (RESTful APIs) and declarative program analysis with the z3 constraints solver, and data replication techniques (i.g. relaxed consistency models). My dissertation research was presented in WWW 2020, ICWE 2019, SANER 2020, ICWE 2021 (Best Paper award), etc.

(c) Employment and Projects

2021.11–now: **Software Engineer/Researcher**, Software Engineering Team, Samsung Research, Seoul, South Korea
I am a member of Quality Tool Lab in Samsung Research, where I am developing AI-based refactoring tools to assist software developers in Samsung Electronics worldwide. I am developing web-based refactoring assistant tools by learning from refactorings applied to other projects.

2015.8–2021.9: **GTA/GRA**, Department of Computer Science, Virginia Tech, Blacksburg, VA, United States
tInferrer (1.5 years, Java): To port from Android to iOS (Java->Swift), this tool learns syntactical translation rules from the equivalent codebases of cross-platform mobile apps and stores them in the relational database. This work was presented in *MobileSoft'18* with Best Paper nomination.
Understanding Heap-Spraying Attacks (1.5 years): I was the main developer to build a core course project for the CS department, where I developed a victim server (C++) by extending JavaScript Virtual machine V8. I studied this work by employing pre-/post-surveys from 540 VT undergrads owning IRB approvals.
https://hosting.cs.vt.edu/CybersecurityEducation/gallery

RT-Trust (1.5 years): I developed LLVM C++ compiler tools to automate distributing embedded apps (drone firmware *PX4*) to support *optee-os*/*SGX* Environments. I'm the second author for *GPCE'18* and *COLA' 20*.

2012.9–2015.7: **Software Engineer/Researcher**, Robotics Research, KIST (Korea Institute of Science and Technology, *one of the top research institutions in South Korea*), Seoul, South Korea
SimonPiC: I developed a distributed system to process and fuse sensor data from multiple system units for real-time robot services. This system was designed not only as scaling to the number of sensor units but also but also being tolerant to a partial-failure of the unit. I implemented the core distributed system (Python/C++), Web-based UIs, 3D Simulator (Python/C++/Blender), and ML-based Leg detection (Python/C++). I annually led demonstrations and presented ten research papers.

(d)   Skills

Programming: JavaScript (4 years), Python (4 years), Java (**SCJP**, 5 years), C/C++ (5 years), SQL, wasm, golang
Package: Node.js, V8, z3py, Heroku (AWS), tensorflow.js, angular, Docker, H.264/AVC JM, optee-os, LLVM, CRDT, ROS, Blender, Ptolemy, ns-2, Linux/Unix systems

(e)   Publication (from 2018 -)

1. **Kijin An** and Eli Tilevich, to be submitted one work to a Software Engineering Conference.

2. **Kijin An** and Eli Tilevich, "Communicating Web Vessels: Improving the Responsiveness of Mobile Web Apps with Adaptive Redistribution," *21st International Conference on Web Engineering (ICWE 2021)*, May 2021. (**17%, 22/118**), **Best Paper Award**.

3. **Kijin An** and Eli Tilevich, "Client Insourcing: Bringing Ops In-House for Seamless Re-engineering of Full-Stack JavaScript Applications," *Proceedings of the Web Conference (WWW)*, April 2020. (**19%, 217/1129**).

4. **Kijin An**, "Enhancing Web App Execution with Automated Reengineering," *Proceedings of the Web Conference (Dotoral Symposium WWW)*, April 2020.

5. **Kijin An** and Eli Tilevich, "D-Goldilocks: Automatic Redistribution of Remote Functionalities for Performance and Efficiency," *Proceedings of the 27th IEEE International Conference on Software Analysis, Evolution and Reengineering (SANER 2020)*, February 2020. (**21%, 42/199**)

6. Yin Liu, **Kijin An**, and Eli Tilevich, "RT-Trust: Automated Refactoring for Different Trusted Execution Environments under Real-Time Constraints," *Journal of Computer Languages (COLA)*, Volume 56, 100939, 2020 *Journal Article*, **Nominated for the Best Paper Award**.

7. **Kijin An** and Eli Tilevich. "Catch & Release: An Approach to Debugging Distributed Full-Stack JavaScript Applications", *19th International Conference on Web Engineering (ICWE 2019)*, June 2019. (**25%, 26/106**)

8. **Kijin An**, "Facilitating the Evolutionary Modifications in Distributed Apps via Automated Refactoring," *19th International Conference on Web Engineering (Doctoral Symposium ICWE 2019)*, June 2019.

9. Yin Liu, **Kijin An**, and Eli Tilevich, "RT-Trust: Automated Refactoring for Trusted Execution Under Real-Time Constraints," *Proceedings of the 17th International Conference on Generative Programming: Concepts & Experience (GPCE 2018)*, Nov 2018.

10. **Kijin An**, Na Meng, and Eli Tilevich, "Automatic Inference of Java-to-Swift Translation Rules for Porting Mobile Applications," *MobileSoft*, 2018, **Nominated for the Best Paper Award**.(6%, 3/52)

(f)   Teaching and Services:

- GTA for CS2505 and CS2506 in CS@VT (Two lecture sessions for *Understanding Heap Spraying*)

- Co-Reviewer for TSE 2018, ECOOP 2020, RO-MAN 2020, MPLR 2020

- President for *Korean Computer Scientists* (KCS) in CS@VT (2019.6 - 2020.5)