Alexandra E. Michael

aemichael.github.io • linkedin.com/in/aemichael • aemichae@cs.washington.edu
PhD Student in Security, Programming Languages, and Compilers at the University of Washington

Objective

I am seeking a research internship for summer 2024, in the areas of programming languages, compilers, and/or security. I have experience in building languages and compilers, hardware security and side channels, formal methods including program verification and synthesis, and memory safety. I am well-suited for an internship related to one or more of these topics.

Languages & Tools

C • C++ • LLVM • Assembly • Bash • Rust • Java • Python • WebAssembly
Racket • Rosette • Haskell • Coq • JavaScript/TypeScript • Kotlin • HTML • CSS/SASS • Vega-Altair

Education

University of Washington, Paul G. Allen School of Computer Science & Engineering September 2022 - Present
Ph.D. in Computer Science & Engineering *Seattle, WA*
Advised by David Kohlbrenner and Dan Grossman.

University of California, San Diego August 2018 - December 2021
B.S. in Computer Science (GPA: 4.0) *La Jolla, CA*

Selected Coursework

University of Washington

- Computer-Aided Reasoning (in progress). *Automated reasoning with SAT and SMT solvers for bug finding, verification, and synthesis.*
- Graduate Programming Languages. *Functional programming and proofs with the Coq proof assistant.*
- Graduate Computer Systems. *Systems overview including operating systems, networks, databases, distributed systems, etc.*

University of California, San Diego

Graduate Computer Security • JIT Security • Compiler Construction • Modern Cryptography • Operating Systems • Algorithms
Computer Architecture: A Software Perspective • Programming Languages • Theory of Computation • Advanced Data Structures

Publications

Avoiding Instruction-Centric Microarchitectural Timing Channels Via Binary-Code Transformations. Michael Flanders, Reshabh K. Sharma, **Alexandra E. Michael**, Dan Grossman, and David Kohlbrenner. *Conditionally accepted to ASPLOS 2024.*

MSWasm: Soundly Enforcing Memory-Safe Execution of Unsafe Code. Alexandra E. Michael*, Anitha Gollamudi*, Jay Bosamiya, Evan Johnson, Aidan Denlinger, Craig Disselkoen, Conrad Watt, Bryan Parno, Marco Patrignani, Marco Vassena, and Deian Stefan. *Symposium on Principles of Programming Languages (POPL), January 2023.*

Research Experience

University of Washington
Compiler-based mitigations for hardware side channels *September 2022 - Present*

Recent work has found that previously proposed microarchitectural optimizations can open novel side channels, potentially leaking sensitive data to attackers. Some such optimizations are now being implemented in hardware, and we anticipate more appearing as time goes on. We are developing compiler-based approaches to mitigating the resulting vulnerabilities in security-critical code.

- Wrote binary code transformations to mitigate x86_64 assembly instructions against certain side-channel vulnerabilities.
- Formally verified that the transformations were semantically equivalent to the original instruction and protected against the required vulnerabilties, using the Rosette computer-aided reasoning tool.
- Collaborated to implement an LLVM pass to apply transformations when compiling cryptographic C code to x86_64 assembly.
- Built a pipeline to compile the libsodium cryptographic library with transformations applied, test the compiled library for correctness, and assess the run- and compile-time performance cost of the transformations.
- Coauthored a paper on our work so far, which has been conditonally accepted to appear at ASPLOS 2024.

University of California, San Diego
Memory-Safe WebAssembly (MSWasm) *September 2019 - September 2022*

MSWasm is a memory safety extension to the language WebAssembly (Wasm). MSWasm is motivated by the compilation of programs from unsafe languages like C to Wasm, and the resulting need to protect data those programs access from memory safety vulnerabilities.

- Collaborated with large, multi-institution research team to build MSWasm as a provably memory-safe extension to WebAssembly.
- Implemented MSWasm compiler in GraalVM, a Java VM-based platform and runtime engine that supports a variety of languages.
- Created and presented poster and short research paper on implementing the MSWasm compiler in GraalVM.
- Collaborated to write LLVM backend for compiling C code to MSWasm, based on existing Wasm backend for LLVM.
- Compiled subset of WASI Libc and the PolyBench-C benchmark suite to MSWasm and used to assess MSWasm's performance.
- Co-first authored a paper on MSWasm that appeared in the Principles of Programming Languages (POPL) 2023 conference.

Work Experience

Meta Menlo Park, CA
Software Engineer Intern, WhatsApp Android Platforms *May - July 2022*

- Investigated build speeds for WhatsApp Android application and ran experiments to identify sources of improvement.
- Designed and implemented proposed changes and assessed for effectiveness in improving build speeds.

Software Engineer Intern, WhatsApp Business Integrity *June - September 2021*

- Designed and implemented extensible internal API in Haskell and PHP for querying business integrity data.
- Implemented UI tool for querying the API; collaborated with other teams to improve on the tool for privacy and usability.
- Presented work on the API and tool to technical and non-technical stakeholders and iterated based on feedback.
- Project became the basis for a full customer-facing feature within a year of the internship.

Lytx, Inc. San Diego, CA
Software Engineer Intern *June - August 2020*

- Implemented backend API endpoints in C# and T-SQL and new frontend features for client-facing web application.

Teaching Experience

University of California, San Diego La Jolla, CA
Undergraduate Teaching Assistant (CSE Tutor) *Spring 2019 - Winter 2021*

CSE 20: Discrete Mathematics *Spring '19, Fall '19, Winter '20, Winter '21, Fall '21*
CSE 95: Tutor Apprenticeship *Spring '21*
CSE 105: Theory of Computation *Spring '20, Fall '20*

- Proofread, edited, and graded assignments and exams.
- Assisted students in office hours and in-class exercises.
- Led course organizational meetings, assigned grading duties to other TAs, and mentored newer TAs.
- Developed weekly assignments and exercises to achieve specific learning goals in Tutor Apprentice course.

The Harker School San Jose, CA
Teaching Assistant—Summer Programming & Advanced Programming *Summer 2016 - Summer 2018*

- Led small team of TAs and assisted course organization.
- Wrote and graded programming assignments and exams.
- Helped high school students with in-class exercises.

Honors & Awards

NSF Graduate Research Fellowship
National Science Foundation *March 2023 - Present*

Corin Anderson Endowed Fellowship
University of Washington, Paul G. Allen School of Computer Science & Engineering *September 2022 - June 2023*

Master's Honorable Mention for Excellence in Research
University of California, San Diego, Department of Computer Science & Engineering *June 2022*

Undergraduate Award for Excellence in Research
University of California, San Diego, Department of Computer Science & Engineering *June 2022*

Tau Beta Pi—The Engineering Honor Society, Member
Tau Beta Pi, California Psi Chapter *January 2021*

Summa Cum Laude
University of California, San Diego *December 2021*