

# Alexandra E. Michael

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

## EDUCATION

---

### University of Washington, Paul G. Allen School of Computer Science & Engineering

*Ph.D. in Computer Science & Engineering (in progress)*

*M.S. in Computer Science & Engineering*

Advised by David Kohlbrenner and Dan Grossman.

Seattle, WA, USA

September 2022 - Present

June 2024

### University of California, San Diego

*B.S. in Computer Science (GPA: 4.0)*

La Jolla, CA, USA

December 2021

## SELECTED COURSEWORK

---

### University of Washington

- Computer-Aided Reasoning. *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 • Programming Languages • Operating Systems  
Computer Architecture: A Software Perspective • Software Engineering • Modern Cryptography • Theory of Computation

## PUBLICATIONS

---

### Avoiding Instruction-Centric Microarchitectural Timing Channels Via Binary-Code Transformations.

Michael Flanders, Reshabh K. Sharma, *Alexandra E. Michael*, Dan Grossman, and David Kohlbrenner.

*International Conference on Architectural Support for Programming Languages and Operating Systems (ASPLOS), April 2024.*

<https://doi.org/10.1145/3620665.3640400>

### MSWasm: Soundly Enforcing Memory-Safe Execution of Unsafe Code.

*Alexandra E. Michael*<sup>\*</sup>, Anitha Gollamudi<sup>\*</sup>, 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.*

<https://doi.org/10.1145/3571208>

## RESEARCH EXPERIENCE

---

### University of Washington

*Compiler-based mitigations for hardware side channels*

Seattle, WA, USA

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 specific timing side channels.
- Formally verified transformations for semantic equivalence to the original instruction and safety against the considered vulnerabilities, using the Rosette computer-aided reasoning tool.
- Collaborated on 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, published at ASPLOS 2024.

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

La Jolla, CA, USA  
 September 2019 - September 2022

MSWasm is a memory safety extension to the language WebAssembly (Wasm). MSWasm protects programs compiled to Wasm from unsafe languages like C and C++ from memory safety vulnerabilities in the original source code.

- Worked with researchers from several institutions to build MSWasm, a provably memory-safe WebAssembly extension.
- Implemented MSWasm frontend for GraalVM, a JVM-based runtime engine with several frontends, including Wasm.
- Created and presented a poster and short research paper on implementing the MSWasm compiler in GraalVM.
- Collaborated on an 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 to assess MSWasm's performance.
- Co-first authored a paper on MSWasm that appeared in Principles of Programming Languages (POPL) 2023.

TEACHING EXPERIENCE

---

**University of California, San Diego**  
*Undergraduate Teaching Assistant ("CSE Tutor")*

La Jolla, CA, USA  
 Spring 2019 - Fall 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

- 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.
- Proofread, edited, and graded assignments and exams.
- Assisted students in office hours and in-class exercises.

**The Harker School**  
*Teaching Assistant—Summer Programming & Advanced Programming*

San Jose, CA, USA  
 Summers 2016 - 2018

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

WORK EXPERIENCE

---

**Meta**  
*Software Engineer Intern, WhatsApp Android Platforms*

Menlo Park, CA, USA  
 Summer 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*

Summer 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 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.**  
*Software Engineer Intern*

San Diego, CA, USA  
 Summer 2020

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

## HONORS & AWARDS

---

### **NSF Graduate Research Fellowship**

*National Science Foundation*

*March 2023*

### **Corin Anderson Endowed Fellowship**

*University of Washington, Paul G. Allen School of Computer Science & Engineering*

*September 2022*

### **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*

### **Summa Cum Laude**

*University of California, San Diego*

*December 2021*

### **Tau Beta Pi—The Engineering Honor Society, Member**

*Tau Beta Pi, California Psi Chapter*

*January 2021*

### **Provost Honors**

*University of California, San Diego*

*Fall 2018 - Spring 2020, Winter - Spring 2021*

## SERVICE

---

### **University of Washington Department Service**

Programming Languages & Software Engineering (PLSE) Blog Co-coordinator

*Fall 2023 - Present*

New Grad Mentoring Program Mentor

*Fall 2023 - Present*

Security & Privacy Lab Outreach Co-coordinator

*Winter 2023 - Present*

PLSE Visit Days Co-coordinator

*Winters 2023 - 2024*

Pre-Application Mentorship Service (PAMS) Mentor

*Fall 2023*

PhD Applications Reader

*Fall 2022*

### **Other Service**

Student Volunteer, Principles of Programming Languages (POPL)

*2022 (remote), 2023*

## LANGUAGES & TOOLS

---

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

---

\*Formerly known as Coq