Gabriel Sherman

Salt Lake City, UT | gabesherman6@gmail.com | +1 207-307-1490

linkedin.com/in/gabe-sherman | gabe-sherman.github.io

Summary

Security-focused Ph.D. researcher specializing in automated vulnerability discovery, fuzz testing, and large-scale tooling for secure systems. I'm particularly interested in expanding the scope and scale of software testing across an increasingly complex software ecosystem.

Education

Ph.D., Computer Science	University of Utah	2024-*current*
B.S., Computer Science	University of Utah	2020-2024

Publications

1. **No Harness, No Problem: Oracle-guided Harnessing for Auto-generating C API Fuzzing Harnesses**
   Gabriel Sherman, Stefan Nagy
   International Conference on Software Engineering **(ICSE '25)**

Experience

Ph.D. Research Assistant, University of Utah – Salt Lake City, UT — August 2024 – *current*

- Currently conducting research under Dr. Stefan Nagy to expand the capabilities of automatic harness generation through testing, development, and evaluation.
- Developed a novel automatic harness generation technique and published a top-tier conference paper detailing the approach.
- Found and reported 60+ bugs across major open-source libraries (40+ confirmed), including memory safety issues, logic flaws, and OOB vulnerabilities. Some of the bugs I have found can be found here: futures.cs.utah.edu/bugs/?search=gabe+sherman

Summer Research Intern, Trail of Bits – Salt Lake City, UT — Summer 2024, 2025

- Performed research and developed tooling for various security goals for two consecutive summers.
- *Summer 2025* — Built a browser-based checksec tool for cross-platform security analysis of ELF, PE, and Mach-O binaries, with a Rust backend for security checks and a client-side JavaScript interface to display results.
- *Summer 2024* — Developed and advanced an automatic harness generation approach for C-based libraries, integrating Trail of Bits' Multiplier tool for static analysis, generating harnesses for widely-used open-source libraries, and performing fuzzing and bug triage that led to the identification and reporting of 6 confirmed bugs.

Invited Talks & Articles

Building checksec without boundaries with Checksec Anywhere – The Trail of Bits Blog	11/2025
Introduction to Fuzzing – University of Utah Cybersecurity Club.	03/2025
Automated Bug Finding – Guest Lecture at Kahlert School of Computing.	09/2024
Automatic Harness Generation for C-based Libraries – Empire Hacking NYC.	08/2024
Automated Harness Generation – Mountain West Undergraduate Research Showcase.	11/2023

Technical Skills

Software Testing: Dynamic Analysis (AFL++), Static Analysis (LLVM passes, semgrep), Crash Triage (casr, ASan)

Software Development: Build tools (CMake, Make), Agentic Frameworks (LangChain), Containerization (Docker)

Languages: Python, C, C++, Java, Rust, Javascript

Projects & Activities

Prompt Injection Analysis: Assessed a commercial AI code-review tool's resilience to prompt-injection attacks such as instruction overrides, repeated-token attacks, and context manipulation.

University of Utah CTF Team Regular competitor in weekly challenges; led team workshops on fuzzing techniques and tooling.

Operating System Fundamentals Developed an xv6-based kernel prototype (booting, interrupts, user-space processes) and a reduced ELF linker that performs relocation.