

Julius Alexandre

(Looking for Compiler or Compiler-adjacent full time roles and/or Internship)

juliuswoosebert@gmail.com | github.com/wizardengineer | linkedin.com/in/julius-alexandre

EDUCATION

Western Governors University

Class of 2026

B.S. Computer Science

- Relevant coursework: Computer Architecture, Operating Systems, Data Structures & Algorithms, AI/ML, Network & Security

EXPERIENCE

Compiler Contributor | LLVM | C/C++, Rust, Compiler

Aug 2024 – Present

- Optimized SelectionDAG passes to reduce generated code size by 50% and improve runtime performance by 20% across Rust, C, and C++ on RISC-V, x86, and AArch64.
- Enhanced SandboxIR vectorizer transforms and added GoogleTest coverage for end-to-end profit-driven optimizations.
- Refactored LLDB on macOS—improved diagnostics and error reporting across all architectures, increasing maintainability.

Compiler Engineer Intern | Nestra | C/C++, Compiler Design, Luau

Jan 2025 – March 2025

- Designed CMake build system and developer tooling, cutting onboarding friction and build failures.
- Built a source-to-source compiler: deserialized Lua/Luau bytecode into SSA-form IR and constructed its CFG middle-end for obfuscation passes.
- Engineered IR obfuscation passes—control-flow flattening, dead-code injection, anti-analysis layers—to fortify embedded scripts.
- Debugged and optimized the compiler pipeline end-to-end, ensuring reliable, high-performance obfuscation across customers' codebases.

Compiler Engineer Intern | COI | C/C++, Assembly, Compiler Optimization

Sep 2024 – Jan 2025

- Built end-to-end deobfuscation toolchain—custom Compiler passes (opaque-predicate, folding, dead-code, CFG), plus Python symbolic execution and flow analysis—to reduce manual reverse-engineering by 40% and boost throughput 25%.
- Led development of a Type-2 hypervisor debugger for OS/hardware emulation, memory introspection, and syscall hooking—accelerating vulnerability discovery by 30%.
- Profiled and refactored C++ codebase and build system, improving performance by 15%.

PROJECTS

AMD-v Hypervisor KrakenSvm | C/C++, Assembly, Windows, VMWare, WinDbg, CPU Internals

- Engineered a C++ x64 Type-2 hypervisor using AMD-v nested page tables and VMEXIT hooks for seamless hardware and OS interception.
- Added Ring-0 hooks and a control plane for live guest OS introspection—tracking memory pages, CPU registers, cache behavior, and syscall/MMIO exits—to improve real-time debugging throughput by 30%. (90+ stars on GitHub).

OS Kernel Developer | C, Assembly, QEMU, Linux, GDB, Operating Systems

- Designed a 32-bit microkernel with a multistage bootloader, preemptive scheduler, and virtual memory paging.
- Integrated interrupt descriptor tables, SMEP/SMAP security protections, and basic VGA/Keyboard/Serial device drivers.

Vulnerability Scraper | Python, Windows, Kernel Drivers, Multithreading, Security

- Developed a Python driver-aware scraper that parses PE import tables in Windows kernel drivers to pinpoint high-risk APIs and identify potential exploit candidates.
- Automated triage workflows using Python multiprocessing for high-throughput vulnerability analysis.

TECHNICAL SKILLS

C, C++, Assembly, Arm, Python, Lua, TypeScript, SQL, Bash, Rust, Java, Git, LLDB, CMake, JSON, IDA Pro, CI/CD, QEMU, Linux, macOS, Windows, LLVM