

SAKU GROSSARTH

Colorado Springs, CO • saku.grossarth@icloud.com • (719) 654-5161
github.com/Arknight38 • linkedin.com/in/saku-grossarth-9040083aa • arknight38.github.io

About Me

I'm a high school programmer focused on systems work: kernel drivers, hypervisor development, low level Rust and C++, and networked applications. I'm the third largest contributor to wplacer across a 2,500+ member open source community. I like building things that work close to the OS, and I have real world experience with SQL and data analysis from an R&D internship. Currently building tools around disassembly and decompilation, as well as anticheat and reversing. Portfolio available at arknight38.github.io.

TECHNICAL SKILLS

Languages: C++, Rust, C, Python, TypeScript, JavaScript, Java, x86/x64 Assembly

Systems & Low-Level: Windows kernel drivers, hypervisors (VMX/EPT), memory management, IPC, PE mapping, hooking, syscalls, OS security

Backend / Infrastructure: Rust, PostgreSQL, WebSockets, JWT, Argon2, Docker, AWS S3, Cloudflare

Tools: Git, Linux, Windows, Visual Studio, Snowflake, Zydys, LIEF, Winsock2

PROFESSIONAL EXPERIENCE

Data Analyst Intern | *Restaurant365* | *Reporting to Head of R&D* | Summer to Winter 2025

- Wrote SQL queries against datasets of ~500,000 rows in Snowflake to support analytics and internal decision making.
- Presented query results and data validation work directly to the Head of R&D. The platform serves thousands of restaurant clients.

OPEN SOURCE LEADERSHIP

wplacer – *Third Largest Contributor* – github.com/wplacer/wplacer

- wplacer is an auto drawing bot for wplace.live, a collaborative pixel canvas platform with 400+ stars and an active distributed contributor base.
- 40 commits and 5,306 lines added across core JavaScript components, becoming the #3 contributor out of the whole project.
- Helped run the contributor side of a 2,500+ member Discord by triaging issues, reviewing PRs, and walking new contributors through the codebase.
- Shipped features, bug fixes, and performance improvements while working within the project.

TECHNICAL PROJECTS

Flux — Rust — React — TypeScript

- High performance real time messaging platform built from scratch. Rust backend using Axum and Tokio handles hundreds of concurrent WebSocket connections with sub-100ms latency.
- Full feature set including servers, channels, direct messages, friend system, typing indicators, presence, reactions, and file uploads backed by AWS S3 and Cloudflare R2.
- JWT authentication with Argon2 password hashing, HTTP-only cookies, and compile-time checked SQL queries via SQLx against PostgreSQL. Containerized with Docker.

CS2 Extern — C/C++ — Closed Source — [read writeup](#)

- External process tool for CS2 with ESP, combat features, and a custom kernel-mode driver for memory operations. Built my own rendering library (zdraw) with FreeType font rendering and a Bounding Volume Hierarchy spatial system for processing entities.
- Driver communicates with the usermode process over shared memory and handles cross-process memory operations at the kernel level.

WdFilterDrv — C/C++ — Closed Source — [read writeup](#)

- Built a Windows kernel-mode driver with a shared memory IPC channel mapped into both kernel and user address spaces via MDL + MmMapLockedPagesSpecifyCache, chosen over IOCTL to eliminate syscall overhead for high-frequency memory access.
- Implemented a manual PE mapper from scratch: parsing DOS/NT headers, copying sections, fixing base relocations via delta calculation, and walking the IAT for import resolution, without touching the Windows loader.
- Exposed memory primitives via MmCopyVirtualMemory and direct page table walks for virtual-to-physical translation; used IRQL discipline, ProbeForRead/Write, and SEH blocks to prevent BSODs under load.
- Debugged iteratively with two-machine WinDbg kernel debugging over serial, analyzing crash dumps against _EPROCESS/_ETHREAD structures to diagnose race conditions and pool allocation bugs.

ArkVisor — C/C++ — ASM — Closed Source — [read writeup](#)

- Hypervisor built to learn how virtualization actually works at the hardware level. Implements VMX entry and exit handling, EPT for memory virtualization, hypercall dispatch, and VM exit routing.
- Includes stealth and anti-detection components to study how hypervisor based tools stay hidden from security software.
- Never fully finished, but still a %50 completed project.

Gif-Engine — Rust — github.com/Arknight38/Gif-Engine

- Desktop animation manager for Windows I built because I ran into performance issues with existing tools and wanted more control.
- Each animation runs in its own process so one bad GIF can't crash everything else. Renders via WS_EX_LAYERED windows and UpdateLayeredWindow for proper per-pixel transparency.
- Built with egui/eframe for the UI. Has a library manager, system tray integration, auto-start, and persistent settings.

Rune Editor — C++, React, TypeScript — Closed source — [read writeup](#)

- Memory editor for Elden Ring with a full-stack desktop architecture.
- C++ backend uses AOB scanning to resolve game memory addresses via the GameDataMan pointer chain, then exposes a REST API (cpp-httpplib) for read/write operations.
- React/TypeScript frontend runs inside the browser.
- Features process enumeration, live rune polling, value locking, and structured logging. Packaged as a standalone Windows app with static file serving from the same binary.

ServerShenanigans — C++ — github.com/Arknight38/ServerShenanigans

- Client-server file sharing app in C++ with resume support, zlib compression, and SHA-256.
- The server can theoretically have up to 50 concurrent connections, one thread per client, with stack-allocated 64KB buffers to keep heap usage low.

- Custom transfer protocol with RAW and COMPRESSED modes, offset-based resume, and real-time progress on the client side.

Additional Projects

- Fluid Physics Engine (C++): Real-time particle-based fluid simulation. Focused on stability, collision handling, and keeping it fast enough to actually run in close to real time.
- Raycasting Lighting Demo (C++): 2D raycasting renderer with dynamic shadow casting. Built to learn computational geometry and how rendering actually works with OpenGL.
- Atlus (In Progress, C++): A decompilation and binary diff analyzer I am currently developing. It employs different libraries and projects such as Zydis, LIEF, and rev.ng for decompilation.

EDUCATION

Pine Creek High School — *Colorado Springs, CO* – Expected Graduation: May 2027

GPA: 4.689 (Weighted) | Rank: 7

- 3 AP courses passed with 5s: AP Psychology, AP Computer Science Principles, AP Human Geography
- 1 AP course passed with a 4: AP Computer Science A
- 4 AP courses in progress: AP US History, AP Calc BC, AP Chemistry, AP Language

CERTIFICATIONS & ACHIEVEMENTS

Competitions

- FBLA National Competitor — Coding & Programming
- 1st Place — FBLA District & State
- Cyberpatriot Competitor

Certifications

- Python Professional Certification
- Java Professional Certification
- CS50 Graduate

Honors

- National Technical Honor Society
- Honor Roll (5 semesters)
- GitHub Achievement: Pull Shark