

SEAN SINGH

Santa Cruz, CA ◊ (669) 286-6227 ◊ seansingh.041@gmail.com ◊ byseansingh.com ◊ github.com/Seanathan10 ◊ linkedin.com/in/seansingh04

EDUCATION

B.S., Computer Science, *University of California, Santa Cruz*

Sept. 2022 – June 2026

Coursework: Operating Systems, Computer Networks, Network Security, Distributed Systems, Compiler Design, Parallelism & Concurrency, Computer Architecture, Machine Learning, NLP, Data Structures & Algorithms

Google Cybersecurity Certificate

Dec. 2025

EXPERIENCE

Software Engineer (Contract)

Jan. 2026 – Mar. 2026

Influencer Marketing CRM Platform

Remote

- Extracted modules from a 260K+ line monolithic CRM (80M+ REST requests/week across a globally distributed deployment) into independently deployable FastAPI microservices, each shipped with unit and integration tests gating Docker deployment
- Built and owned the Authentication Service that fronts every other microservice; worked inside a private VPC data plane with network isolation, Cloudflare WAF, and disciplined secrets management across services
- Backed services with PostgreSQL/Supabase, Redis, Kafka, and Celery on Docker and Kubernetes with Datadog observability; validated each on a dev branch before promoting it to a v2 endpoint
- Contributed to a RAG-based AI chatbot trained on client conversation history and an automated message classification system filtering DMs across 800+ managed social accounts

Software Engineer / Undergraduate Researcher

Apr. 2024 – Jan. 2026

AIEA Research Lab, UC Santa Cruz

Santa Cruz, CA

- Built and operated an explainable-AI autograder serving 200+ students across multiple ML courses at 99.9% uptime; its new guidance system earned a 95% student preference rate over the prior system in a pilot study
- Developed tree-based Python code analysis to evaluate the correctness, behavior, and logic of student ML chess implementations, benchmarked against external UCI chess engine APIs for accuracy and performance
- Improved LLM accuracy and interpretability by 39% over baseline systems by integrating symbolic solvers and RAG
- Hardened sandboxing and security boundaries for executing untrusted student code; delivered on weekly Agile sprints with code review and standups

Computer Systems Tutor / Teaching Assistant

Sept. 2024 – Jan. 2026

Jack Baskin School of Engineering, UC Santa Cruz

Santa Cruz, CA

- Mentored 600+ students across 4 quarters in low-level systems (POSIX threads, sockets, HTTP servers, virtual memory, synchronization, memory management) and held a 4.8/5 average rating for helpfulness and clarity
- Built and maintained the course materials site (setup scripts, lab starters, benchmarks) for 800+ students at 99.9% uptime with a CI/CD pipeline, cutting setup time by 80% and basic support requests by 70%
- Taught debugging with GDB, Valgrind, and ThreadSanitizer; led weekly labs on segmentation faults, race conditions, and memory corruption, and coordinated with 4 TAs on grading and test scripts

Director of Instruction

Mar. 2024 – Aug. 2025

Google Developer Student Club, UC Santa Cruz

Santa Cruz, CA

- Designed and delivered 15+ technical workshops to 180+ students on networking, systems programming, APIs, distributed systems, Docker, and TensorFlow, plus 20+ weekly algorithm and code-review sessions
- Led development of the club portal (React frontend, Firebase backend, authentication, dashboards, event management, real-time updates) for 70+ active members; grew multi-club event turnout by 40%+

PROJECTS

Capsule – Encrypted, Self-Destructing File Sharing

Rust, Kotlin, TypeScript ◊ withcapsule.dev

- Launched publicly on ProductHunt and Hacker News in July 2026 to 150+ launch-day users; built end to end as a Rust/Axum server, Rust CLI, Kotlin/Jetpack Compose Android app, and static Web UI, with uploads auto-deleted after one hour
- Implemented interoperable client-side end-to-end encryption (**age** format with BIP39 mnemonic passphrases) across all three clients, including a dependency-free pure-JS port of **age** for the browser
- Hardened the Rust server with SQLite WAL, per-IP rate limiting and 2 GB/hour bandwidth caps via **DashMap**, **tower** concurrency limits and timeouts, and path-traversal-safe filename sanitization
- Shipped a zero-touch GitHub Actions release pipeline: one tag cross-compiles macOS, Linux, and Windows binaries (x86_64 + arm64) and auto-bumps the Homebrew tap, Fedora COPR spec, and Scoop manifest
- Wrote a 170+ test Android suite (Compose UI, Hilt, Room/DataStore) including an R8 guard that runs instrumented tests against a fully minified build to catch obfuscation regressions before shipping

Nikola – Tesla EV Companion App

Rust (Axum), React, TypeScript ◊ nikola.byseansingh.com

- Implemented Tesla's Vehicle Command Protocol signing pipeline in Rust, required by all 2024+ firmware: P-256 ECDH, SHA-1 KDF, HMAC-SHA256 subkey derivation, TLV-encoded metadata signing, and Protocol Buffer serialization
- Managed two independent signing domains with per-domain session caching, monotonic anti-replay counters, and automatic resync on epoch and counter faults; diagnosed undocumented edge cases the reference implementation omits
- Built OAuth 2.0 with PKCE and server-side-only token storage, and engineered RFC 8615 virtual key pairing through a `.well-known` endpoint verified against live vehicle data
- Surfaces live telemetry, remote vehicle commands, and charging analytics (cost per mile, cost per kWh, emissions) with charge-time recommendations derived from historical usage and electricity pricing

Beacon – Travel Discovery Platform

React, TypeScript, Node.js, AWS ◇ beaconapp.live

- Built and maintain a live full-stack map platform on AWS (Mapbox, WebSocket pin sharing for 10+ concurrent editors) plus a social layer of comments, follows, bookmarks, and leaderboards; ~20 active users with pins in all 50 states
- Integrated Gemini for a RAG-enhanced itinerary planner that folds in user preference history, plus Google Maps and Amadeus for location search and carbon emission tracking
- Refactored the Express/SQLite backend onto a repository pattern and hardened the frontend with typed service modules, a TypeScript typecheck build gate that resolved 25 pre-existing errors, and a Vitest harness

NSA Codebreaker Challenge 2025 – All 7 Tasks

Python, Ghidra, Volatility 3, Wireshark

- Performed digital, network, and memory forensics: reconstructed an attack timeline from an ext2 image, traced DNS poisoning through a 2,344-packet PCAP, and built a custom Volatility profile from an OpenWrt kernel image
- Reverse engineered an obfuscated dropper past ptrace and sigaction anti-debugging, extracted a nested ELF from `memfd`, and recovered the C2 server details and persistence mechanism
- Broke an AES-128-ECB + RSA + RC4 stack by finding 26-bit effective key entropy and brute-forcing the 2^{26} keyspace in 5 minutes; mapped a privilege escalation path into the C2 and achieved RCE via unsafe recursive ZIP extraction on Android

UCSC.app – AI-Powered Student Portal

React, TypeScript, FastAPI, GCP ◇ ucsc.app

- Built and deployed a campus information platform end to end in 48 hours at a hackathon, then grew it into a maintained product serving 50+ daily active users under a small open-source GitHub organization
- Rewrote course-data ingestion to consume the university's Pisa API directly instead of scraping HTML, archiving raw responses for reprocessing and cutting a full catalog refresh from roughly 50 minutes to 30
- Added Gemini summaries for menus, news, and course data, parallelized scraping, a class-location campus map, and sub-2s page loads

Simple-C Compiler and Web IDE

Python, FastAPI, React, TypeScript

- Built a compiler from scratch: maximal-munch scanner, recursive descent parser and AST, symbol tables with multi-pass type checking and scope resolution, error recovery, and three-address-code IR generation
- Implemented local value numbering and configurable loop unrolling as toggleable optimization passes, exposed via a FastAPI compile API and a React/TypeScript web IDE with a custom editor, command palette, and live recompilation

Mini Encrypted Network (Tor-Style Onion Router)

Python, Scapy, SSL/TLS, Flask

- Built a multi-hop circuit-based routing network using RSA PKCS1-OAEP for layered per-hop key wrapping, SSL/TLS at the exit node for real HTTPS relaying, and encrypted back-propagation of responses
- Developed protocol fingerprinting for VPN detection (OpenVPN/WireGuard), DNS-over-HTTPS analysis, and Tor exit identification, with a Flask dashboard streaming events over WebSockets

DIY Homelab

Proxmox, Docker, Ansible, Linux, iptables/firewalld

- Run a Proxmox hypervisor for 2+ years hosting 10+ containers and 3+ VMs of production services at 99%+ uptime, with Ansible automating provisioning, patching, backups, and health checks

SKILLS

Languages: Rust, Python, C, C++, TypeScript, JavaScript, Kotlin, Java, SQL, Bash, x86/RISC-V Assembly

Backend & Systems: Axum, FastAPI, Node.js/Express, Flask, PostgreSQL, Redis, SQLite, Kafka, Celery, POSIX threads

Frontend & Mobile: React, Vite, Tailwind, Jetpack Compose, WebSockets, Vitest, Compose UI testing

Infrastructure & DevOps: Docker, Kubernetes, Proxmox, Ansible, AWS, GCP, Cloudflare, GitHub Actions, CI/CD, Datadog

Security & AI: Ghidra, Binary Ninja, Volatility 3, Wireshark, Scapy, GDB, Valgrind, applied cryptography (ECDH, HMAC, AES, RSA); PyTorch, OpenCV, LLM APIs (Gemini, OpenAI), RAG, vector databases

AWARDS

NSA Codebreaker Challenge 2025 (all 7 tasks) ◇ 1st Place, SCAI “Is No Limit” AI Competition, 2023 ◇ Best SlugHack, CruzHacks 2023