

Spencer Christensen

Milwaukee, WI | 612-222-4286 | contact@spencer-christensen.com | spencer-christensen.com
U.S. Citizen | Available: May 2027

CyberCorps: Scholarship for Service recipient pursuing an M.S. in Information Assurance and Cyber Defense, specializing in operating-system and firmware security. Developed RISC-V kernel subsystems in C; currently porting Embedded Xinu to Rust for memory safety. Hands-on experience in embedded hardware security assessment.

EDUCATION

Honors & Affiliations

<i>CyberCorps: Scholarship for Service (SFS) – Scholar</i>	<i>01/2026 – Present</i>
<i>Upsilon Pi Epsilon Honor Society – Member</i>	<i>04/2026 – Present</i>
<i>Association for Computing Machinery (ACM) Student Chapter – Treasurer</i>	<i>08/2025 – 05/2026</i>

M.S. Information Assurance and Cyber Defense

08/2025 – 05/2027 (Expected)

Marquette University, Milwaukee, WI

- Cumulative GPA: 3.91
- Thesis Topic: Porting Embedded Xinu to Rust while redesigning the memory-protection and privilege-level architecture.
- Coursework: Ethical Hacking, Network Design and Security, Compiler Construction, Advanced Computer Security

B.S. Computer Science, Minor in Data Science

08/2022 – 05/2026

Marquette University, Milwaukee, WI

- Cumulative GPA: 3.15 | Dean's List (3 semesters)
- Coursework: Operating Systems, Programming Languages, Regression Analysis, Statistical Methods, Capstone Project (*Team Lead*)

EXPERIENCE

Embedded Operating Systems Research Fellow

06/2025 – 08/2025; 06/2026 – 08/2026

Embedded Xinu Lab – Marquette University, Milwaukee, WI

40 hrs/week

- Selected for two 10-week NSF Research Experience for Undergraduates (REU) appointments to design and implement core subsystems of the 64-bit RISC-V port of the Embedded Xinu operating system.
- Implemented Sv39 virtual memory with per-process isolated address spaces and safe user data transfer across the kernel boundary.
- Designed the user process system: POSIX-style threads sharing heap and stack views, child processes with parent tracing and protection, and virtualized device descriptors with stdin/stdout/stderr inheritance.
- Redesigned the kernel's interrupt and exception handling to support preemptible and blocking kernel operations, with stackable kernel interrupts.
- Built interrupt-driven asynchronous device drivers (serial UART, TTY) on top of a device-tree-driven abstraction layer, with the device table, system-call table, and platform configuration generated at build time.
- Hardened the kernel's privilege architecture by enforcing RISC-V privilege separation that the prior port bypassed, reworking the U-Boot bootloader to boot exclusively in supervisor mode per the RISC-V privilege specification and cutting boot time from 40 to 20 seconds.

Operating Systems Teaching Assistant

01/2026 – 05/2026

Marquette University, Milwaukee, WI

15 hrs/week

- Led weekly labs (20 students) and open office hours for a 75-student Embedded Xinu-based operating systems course; co-designed and graded written exams, lab practicals, and weekly kernel projects.
- Integrated GitHub-based submission into the course's nightly test suite, creating access-controlled student repositories with per-assignment branches enabling commit-history reviews for academic-integrity audits and debugging assistance.
- Replaced the course's manual setuid submission system with an automated script that pulled and repackaged student code for nightly hidden-testcase feedback and final graded submissions.

IT Service Desk Student Lead and Analyst

09/2023 – 01/2026

Information Technology Services – Marquette University, Milwaukee, WI

15–20 hrs/week

- Promoted to Student Lead (09/2024); trained and oversaw 8 new staff and served as the escalation point for university-wide technical issues and policy rollouts.
- Resolved 10–25 support tickets per week spanning software installation, hardware issues, and Active Directory account administration and password resets.

PUBLICATIONS

“Improving Privilege Levels and Memory Protection for Teaching in Embedded Xinu”

02/2026

ACM SIGCSE Technical Symposium 2026, Student Research Competition, St. Louis, MO

- S. Christensen and S. O'Brien. In Proc. 57th ACM SIGCSE Technical Symposium (SIGCSE TS '26), pp. 1653–1654, 2026. doi:10.1145/3770761.3777159

PROJECTS

IoT Device Penetration Testing

spencer-christensen.com/p/8d09f

- Conducted a simulated security audit of a purpose-built SoC security-training board: cracked its password-protected serial debug shell with John the Ripper, dumped internal and external flash, identified an undocumented data stream, and recovered all embedded secrets.
- Reverse-engineered the board's obfuscated salt-and-hash verification mechanism in Ghidra, locating a device-specific secret that standard analysis tooling missed.

Lab Practical Exam Server

spencer-christensen.com/p/fad38

- Built and deployed a dependency-free lab-practical exam system in Go that administered live practical exams.
- A single server binary distributes time-boxed materials and accepts submissions over self-signed HTTPS, with PIN-based access control (constant-time checks), late bins, JSON-configured sessions, and query-able audit logs.
- Student binaries embed the TLS certificate for pinned connections, and submissions are tar-archived with upload caps and path-traversal protections.

Embedded Firmware Analysis and Automation

spencer-christensen.com/p/bb5cd

- With a three-person team, performed static security assessments of a consumer router, an IP camera, and bare-metal firmware, finding hard-coded root credentials, embedded private keys and cloud tokens, a vulnerable BusyBox, and an insecure U-Boot chain allowing unsigned code execution; co-authored the mitigation report.
- Built a modular, portable Bash triage scanner with pluggable analysis modules and dependency verification, and used it to independently reproduce the assessment's findings.

System Administration Course Assignment Redesign

Spring 2026

- Redesigned all 10 assignment specifications for a graduate system administration course (Linux server administration, networking services, service deployment, shell automation) as an enrolled student for course credit; designs adopted as the official course assignments for the 25-student class.

SOZO Community Platform (Technical Co-founder)

sozo-creative.app

- Designed the authorization model around PostgreSQL row-level security policies, with signed-URL Supabase Storage access, layered input validation, and open-redirect protection in auth flows.
- Implemented a three-stage CI pipeline running unit, integration, and access-control policy tests against a containerized database.

TECHNICAL SKILLS

Security: vulnerability assessment, network and firmware security, static/binary analysis, reverse engineering, penetration testing, access control, secure coding, memory safety, Ghidra, Wireshark, Nmap, John the Ripper, Hashcat

Languages: C, Rust, Python, Go, Bash, TypeScript, RISC-V assembly

Systems: OS kernel development, virtual memory (Sv39 paging), privilege/memory-protection architecture, interrupt and device-driver design, embedded systems

Platforms & Practices: Linux/Unix, Git, GitHub Actions (CI/CD), test-driven development, AI-assisted development (agentic and chat-based workflows), SQL, Docker, Excel/VBA