Nevin Shine

B.Tech Computer Science Student — Systems Security

German Citizen | Nürnberg, Germany (Available May–June 2026)
Currently Studying in India (Semester 4) | +49 157 54256832
Email: nevinshine05@outlook.com | GitHub: github.com/nevinshine

TECHNICAL FOCUS

Core Areas: Linux Kernel Security, eBPF Runtime Enforcement, Compiler–Kernel Co-Design.
Research Goal: Embedding compile-time intent (LLVM) into kernel-native enforcement (eBPF).

SYSTEMS SECURITY RESEARCH

Sentinel-CC — Compiler-Kernel Execution Integrity **2026 (Active)**

- **Architecture:** Architected a *Policy-Carrying Code (PCC)* system where security policies are generated during compilation and cryptographically bound to the ELF binary.
- **Compiler Engineering (C++):** Implemented a custom **LLVM Module Pass** using the C++ API to analyze Control Flow Graphs (CFG), extract syscall provenance, and inject a `.sentinel` section.
- **Trust Chain:** Built a root-of-trust pipeline using **SHA-256** hashing and **RSA-2048** signing, verified at runtime via the **Linux Kernel Keyring** to prevent supply-chain tampering.
- **Kernel Enforcement:** Engineered an **eBPF-LSM** monitor that intercepts `sys_enter` tracepoints, enforcing instruction-level offsets with $O(1)$ complexity.
- *Tech:* C++, C, LLVM IR, eBPF (CO-RE), LibELF, Linux Keyrings.
- *Artifact:* github.com/nevinshine/sentinel-cc

Sentinel Runtime — Kernel Intrusion Prevention **2025**

- **Objective:** Replaced legacy `ptrace` supervision with high-performance ring-0 hooks to neutralize "Living-off-the-Land" attacks and fileless malware.
- **Mechanism:** Implemented kernel-space process lineage tracking to enforce strict policy propagation across `fork`/`exec` chains.
- **Performance:** Benchmarked system overhead at **<5 μs per syscall** under stress tests exceeding 10,000 processes.
- *Tech:* C, Linux Kernel Internals, eBPF, Bash.
- *Artifact:* github.com/nevinshine/sentinel-runtime

Hyperion — Wire-Speed Network Containment **2025**

- **Design:** Architected an **XDP** firewall performing packet rejection at the NIC driver level for immediate Command-and-Control (C2) containment.
- **Resilience:** Achieved lockless, wire-speed packet filtering operational even if user-space is degraded.
- *Tech:* eBPF, XDP, C, TCP/IP Stack.
- *Artifact:* github.com/nevinshine/hyperion-xdp

Telos Runtime — Agentic Security Architecture **2026 (Ongoing)**

- **Vision:** Designing a closed-loop enforcement runtime correlating AI agent intent with kernel execution graphs.
- *Artifact:* github.com/nevinshine/telos-runtime

TECHNICAL SKILLS

Languages	C (System/Kernel), **C++ (STL, LLVM API)**, Python, SQL, x86 Assembly
Systems	Linux Kernel Internals, eBPF (LSM/XDP/TC), ELF Format, Linker Scripts
Security	Runtime Enforcement, Root of Trust Design, Cryptography (RSA/SHA)
Tools	Clang/LLVM, bpftool, GDB, Ghidra, Git, Make/CMake, Docker, QEMU

EDUCATION

Bachelor of Technology in Computer Science & Engineering **Expected 2028**
Amal Jyothi College of Engineering, India *Current: Semester 4*

Relevant Coursework: Operating Systems, Data Structures (C++), Computer Architecture, Networks.

ACHIEVEMENTS

Award	Best Concept Award: Mastermind 2025 Innovation Competition
Challenge	"100 Days of Systems Security" (Daily kernel research documentation)
Languages	**German (Native)**, English (Professional/Fluent)