# Nikhil Gupta

linkedin.com/in/nikhil-g99 | github.com/BitByNik

## Education

**Indian Institute of Technology Delhi** — New Delhi, India
*M.Tech in Cyber Security, CGPA 8.746/10* — *Jul 2024 – Jul 2026*

**Thapar Institute of Engineering and Technology** — Patiala, India
*B.E. in Computer Engineering, CGPA 8.50/10* — *Aug 2017 – Aug 2021*

## Experience

**Cerify** — New Delhi, India (Remote)
*Software Development Engineer (SDE 1)* — *Jun 2026 – Present*

- Built the core C++ analysis library from scratch, with static analysis passes for common Immunefi vulnerability classes
- Extended the symbolic execution engine to generate concrete exploit inputs, integrated with the static analysis passes
- Built a Solidity-to-LLVM bitcode toolchain, reusable across analyses
- Own server deployment and operations; lead interns and work across the team on roadmap and releases

**Cerify** — New Delhi, India (On-site)
*Intern* — *Aug 2025 – Jun 2026*

- Co-developed a neuro-symbolic detection framework (**0.98 reentrancy F1** on SmartBugs); same as my M.Tech thesis

**Piktorlabs** — Bengaluru, India (Remote)
*Software Engineer* — *Sep 2021 – Dec 2022*

- Developed full-stack features using ReactJS, Node.js (Express) and MongoDB
- Achieved **75%** unit test coverage (Jest) and resolved **70%+** of critical SonarQube issues
- Overhauled and optimized UIs, including the search system, improving performance, responsiveness and usability
- Designed and implemented REST APIs for user engagement analytics and recommendations

## Projects

**Neuro-Symbolic Smart Contract Auditing Framework** — Jul 2025 – Jul 2026
*M.Tech Thesis, "Symbolic Analysis of Smart Contracts", IIT Delhi | Prof. Subodh Vishnu Sharma*

- Co-developed a neuro-symbolic auditing framework that pairs AI-guided vulnerability scouting with symbolic execution to output concrete, solver-proven exploit inputs
- Co-built the symbolic engine: Solidity front end, program analysis, and detectors for reentrancy (read-write and read-only), arithmetic overflow and greedy contracts
- Highest reentrancy F1 of 11 tools on SmartBugs (**0.98**, no false positives); only tool to detect read-only reentrancy

**Coverage-Guided Greybox Fuzzing for TON Smart Contracts** — Jan 2025 – Jul 2025
*Prof. Subodh Vishnu Sharma | AFL++, C++, Python*

- Implemented a greybox fuzzer for TON using AFL++ with structure-aware Bag-of-Cells (BOC) mutations
- Built a C++ harness for the TVM emulator and a Python mutator to apply syntax- and semantics-preserving edits
- Designed a coverage proxy from execution signals (exit code, gas usage, actions), validated on vulnerable contracts

**JPEG-Compatible Grayscale Image Encryption with Adaptive XOR** — Jan 2025 – May 2025
*Prof. Huzur Saran | Cryptography, Image Compression*

- Implemented an encryption-then-compression scheme preserving JPEG compatibility for image sharing
- Enhanced security by adding adaptive XOR, improving entropy and robustness without affecting compression

**xv6 Kernel Enhancements** — Jan 2025 – May 2025
*Prof. Smruti Ranjan Sarangi | C, xv6*

- Extended xv6 with user login, history, chmod and system call controls in the shell
- Added signal handling, priority scheduling and adaptive page replacement for process and memory management

## Technical Skills

**Languages**: C, C++, Python, Java, JavaScript, HTML, CSS, Solidity
**Technologies**: ReactJS, Node.js, MongoDB, Android Development, LLVM, Static Analysis, Symbolic Execution
**Tools**: Git, GitHub Actions