Mihir Hrishikesh Naik Nethavath
mihir170204naik@gmail.com | (631)-882-5616 | LinkedIn | GitHub

EDUCATION

Stony Brook University — **Stony Brook, New York**

- *B.S. in Computer Science* - **GPA:** 3.5/4.00 — *Expected Graduation, Dec 2026*
- **Related Coursework:** Systems Programming, Computer Networks, Software Development, Web Security, Offensive Security

SKILLS

Languages: Python, Java, JavaScript, TypeScript, C, PHP, C++
Frameworks: React, Node.js, Express, Spring Boot, OpenAI API, pandas, Jupyter
Databases: MongoDB, PostgreSQL, MySQL, SQL
Security Tools: Kali Linux, Nmap, Scapy, Wireshark, Burp Suite
Tools & Other: Git, Apache, nginx, Linux, Microsoft Office
Security: CVE Analysis, OWASP Top 10, XSS/CSRF Prevention, TLS/SSL, AES-GCM, PKI/X.509, Network Sniffing, Packet Crafting, bcrypt

EXPERIENCE

Undergraduate Security Research Assistant — **Stony Brook, New York**
Stony Brook University - CSE Department — *Nov 2024 – Present*

- Engineered Python pipeline parsing compressed Debian repositories (lz4) and CVE databases, filtering across httpd, database, and web package categories to extract configuration-dependent security vulnerabilities
- Developed **GPT-4** powered classification system with structured prompting and **JSON schema validation**, analyzing configuration knobs that enable exploitation to reduce false positives in vulnerability scanning
- Authored technical documentation and delivered findings in sprint reviews, communicating results to faculty stakeholders and collaborating as part of a research team toward a publication-ready paper

Generative AI Engineer Intern — **Remote**
NeuralSeek (Remote) — *Aug 2025 – Nov 2025*

- Built AI agent demos for customer service automation across healthcare, finance, and education sectors using RAG platform, configuring knowledge base connections and prompt templates to reduce irrelevant responses
- Conducted **alpha testing** across **100+ queries**, documenting failures and developing a **Python** evaluation framework using semantic similarity scoring and **Jupyter** visualizations to track performance regressions

PROJECTS

Secure Web Application Deployment & Vulnerability Remediation — **Stony Brook, New York**
Apache, PHP, MySQL, Let's Encrypt, fail2ban, Linux (Ubuntu) — *Jun 2024 – Sep 2024*

- Deployed and hardened production web server on Ubuntu Linux with Apache, MySQL, and PHP stack, implementing TLS/SSL encryption via Let's Encrypt and HTTP Basic Authentication to protect against unauthorized access
- Identified and remediated critical OWASP Top 10 vulnerabilities in a PHP Twitter clone application, including XSS via output encoding, SQL injection via prepared statements, and CSRF via anti-CSRF tokens with SameSite cookie attributes
- Configured fail2ban intrusion prevention system with iptables integration to block brute-force SSH attacks, implementing automatic 20-minute IP bans after 5 failed authentication attempts

Argus – Passive Network Traffic Analyzer — **Stony Brook, New York**
Python, Scapy, Kali Linux — *Jan 2026 – Feb 2026*

- Built a passive network sniffer on Kali Linux using Scapy, implementing port-agnostic protocol detection for DNS, HTTP, and TLS traffic without relying on fixed port numbers
- Engineered protocol dissection logic to extract DNS query names, HTTP methods/URIs/hosts, and TLS SNI fields from raw packet captures across mixed network traffic
- Designed the tool to operate passively in promiscuous mode, capturing and classifying live traffic in real time without injecting packets or disrupting network sessions

Secure Password Vault — **Stony Brook, New York**
Python, AES-GCM, cryptography library — *Feb 2026 – Mar 2026*

- Implemented an encrypted password manager in Python using AES-256-GCM authenticated encryption, ensuring both confidentiality and integrity of stored credentials
- Derived encryption keys from a master password using PBKDF2 with a random salt, protecting against brute-force and dictionary attacks on the vault file
- Designed vault serialization with per-entry nonces to prevent IV reuse across encryption operations, following cryptographic best practices for GCM mode