

Kalyan Gopalam

Boston, MA · +1 (857) 339-9252 · gopalam.k@northeastern.edu

linkedin.com/in/gk-eh · github.com/Kalyan-Deva · kalyangopalam.com · HTB: @ScorpionXDev

Experience

Flixverse (Web3 Startup)

Application Security Analyst

Bengaluru, India

July 2020 — April 2022

- Tested web, auth, and API surfaces carrying 10K+ requests a day using Burp Suite and manual authorization checks, and caught broken access controls the scanners had marked low risk.
- Handled the full vulnerability lifecycle on a live platform, from triage and CVSS scoring through remediation tracking, and closed 40+ findings with every fix verified before sign-off.
- Reworked how findings were reported, adding proof-of-concept steps and clear fix criteria, so engineers could verify their own work and high-severity issues closed 35% faster.
- Wrote up secure coding and access-control standards, then walked each engineering team through them directly, which brought recurring vulnerability classes down 30%.

Projects

AI & Agentic Security Red Teaming — Prompt Injection, Jailbreaking, Tool Abuse

2026

- Tested agentic workflows for prompt injection, jailbreaking, and insecure output handling across the OWASP LLM Top 10, and treated the agent's tool access as attack surface rather than something to trust.
- Built a triage pipeline on the Claude API, then threat modeled the harness itself, because a system deciding what reaches an engineer's queue is worth attacking before somebody outside does it first.
- Worked out the prompts and workflow logic that made model output consistent enough to act on, which is where most of the real engineering in agentic systems turns out to live.

RuleRecoil — Detection Efficacy & Coverage Gap Validation

2026

- Built a Python harness that replays saved evidence against each detection rule and labels the result a true positive, false positive, or gap, so rule changes get checked before they ship.
- Put together a corpus of benign near-misses and wired it into a 125-test regression run, which catches the noisy rules that would otherwise end up in an analyst's queue.
- Added a per-detection explain view showing which condition fired and on what evidence, so deciding whether an alert is worth chasing stopped being guesswork.

MITRE eCTF 2026 — Team NEU-1 / NeuSense | Team Lead

2025 — 2026

- Threat modeled the design with STRIDE and made every permission check fail closed, which shut down replay, key-derivation, and memory-scraping attacks before they reached the evaluation round.
- Built a bare-metal hardware security module on ARM Cortex-M0+ with AES-CBC-256 via AESADV/DMA, HMAC-SHA256, and three-tier HKDF, and led 5 engineers to zero critical findings in adversarial review.
- Worked inside the limits of a TI MSPM0L2228 board with no operating system and a few kilobytes of RAM, so every design choice had to justify its memory and cycle cost before it went in.
- Added a secure flash filesystem and PIN rate limiting, then mapped all 20+ security requirements to tests reviewers could rerun themselves without a walkthrough.

BD Alaris 8015 Infusion Pump — Medical Device Security Research

- Took apart the configuration and authentication paths on a BD Alaris 8015 pump with Prof. Kevin Fu and Axel Wirth, and found an undocumented hardcoded default credential (CWE-798) alongside CVE-2016-9355.
- Walked 6+ entry points on the device, scored each one with CVSS v3.1 and STRIDE data-flow analysis, and worked out which findings needed disclosing first and which could wait for a firmware cycle.

Education

M.S. in Cybersecurity, Northeastern University

Boston, MA · Expected December 2026

coursework: Applied Cryptography, Network Security Practices, Software Vulnerabilities & Security, Linux Kernel Security

Leadership

NUIoTConnect — Northeastern IoT Security Club

Boston, MA

Technical Director | Security Partner for 100+ Student Engineers

2025 — Present

- Run hands-on workshops on offensive security, secure SDLC, and vulnerability triage for 100+ student engineers, and review project work early enough that teams can still change it.
- Put together shared threat-model and security-review templates for 8+ project teams, which cut the back-and-forth on reviews by about 40% and let teams catch the obvious problems before coming to me.

Technical Skills

Offensive Security & Red Teaming: Manual Pentesting, Exploit Development, Web & API Testing, Burp Suite

AI & LLM Security: Prompt Injection, Jailbreaking, Agentic Testing, Tool-Access Abuse, OWASP LLM Top 10

Detection & Purple Teaming: Detection-as-Code, Rule Validation, Coverage Gap Analysis, False-Positive Reduction

Vulnerability Management: Threat Modeling (STRIDE), CVSS v3.1, CWE Classification, Coordinated Disclosure

Firmware, Crypto & Languages: HKDF, AES-CBC-256, HMAC-SHA256, ARM Cortex-M0+, Python, C/C++, Bash, Linux