

Darshana Venkatadasan

darshanav.vercel.app/about | 708 932 7447 | darshanav017@gmail.com

EDUCATION

UNIVERSITY OF CHICAGO

MS IN COMPUTER SCIENCE

Jun 2025 | Chicago, IL

NIT TRICHY

BTECH IN INSTRUMENTATION ENGG

May 2023 | Trichy, India

LINKS

Github:// [darshana-v](#)

LinkedIn:// [darshana-v](#)

Twitter:// [@DarshanaV5](#)

Lesswrong:// [@darshanav](#)

Medium:// [@darshanav](#)

RESEARCH INTERESTS

Verifiable AI inference (zkML)

Secure multi-party computation

Privacy-preserving ML

Distributed AI systems

AI safety & evaluation

SKILLS

LANGUAGES

Python • Rust • C++ • Go • JS • Bash

ML & CRYPTOGRAPHY

PyTorch • ONNX • EZKL • snarkjs /
circom • CrossHair / Z3 • Weights &
Biases

SYSTEMS & INFRASTRUCTURE

mTLS / X.509 • Cedar • Docker •
Kubernetes • AWS • Terraform

COURSEWORK

Cryptography

Machine Learning

Distributed Computing

Parallel Programming

Operating Systems

Databases

Compiler Design

Data Structures & Algorithms

ACTIVITIES

Google DSC Lead, NIT Trichy

Women in Graduate Science, UChicago

Mentor, Anita Borg

Spider RnD Club, NIT Trichy

EXPERIENCE

HIGHFLAME | AI SECURITY ENGINEER

Nov 2025 - Present | Remote

- Built multi-agent red-teaming orchestrator for adversarial evaluation of LLM systems; implemented single-responsibility attack agents across prompt injection, jailbreaking, and tool-use exploitation vectors with automated failure-mode classification.
- Architected real-time safety enforcement engine with Cedar policy evaluation and three-tier detection pipeline for monitoring agentic AI workflows.
- Developed zero-trust ML model security scanner and MCP protocol vulnerability analyzer.
- Designed session-based agentic safety framework with cumulative risk scoring, exponential decay, and cross-turn detection memory for real-time control of multi-step LLM agent sessions.

RESEARCH

INDEPENDENT RESEARCH | VERIFIABLE AI & SAFETY

Jan 2026 - Present

- zkAgent:** Built verifiable agent execution system using three ZK proof layers: EZKL (Halo2) for model inference, snarkjs/circom (Groth16) for tool schema compliance, and SP1 zkVM + TLSNotary for tool output authenticity. Trained a tiny GPT-2 model (2-layer, 64-dim) and proved end-to-end inference with EZKL. TypeScript + Rust + Python + Circom monorepo with formal trust model.
- LLMCodeProbe:** Applied symbolic execution (CrossHair/Z3) and property-based testing to LLM-generated code. Found 17.4% of HumanEval solutions contain formally discoverable bugs invisible to test suites. Introduced *deep@1* metric and the *correctness gap*. Published on Medium.
- SpecEval:** Multi-language formal specification inference benchmark (Java/JML, C/ACSL, Rust/Prusti, Solidity/SMTChecker). Introduced *minimality* and *strength* metrics measuring specification quality beyond consistency.

UCHICAGO | GRADUATE RESEARCHER

Sep 2023 - May 2025 | Chicago, IL

- Privacy-Preserving Q&A System:** Built schema-aware LLM→SQL pipeline with differential privacy guarantees, RBAC policy enforcement, privacy budget tracking, and audit logging to prevent statistical inference attacks.
- Transactional Storage Engine (Rust):** Implemented B-Tree-based key-value store from first principles with write-ahead logging, crash recovery, MVCC concurrency control, and workload benchmarking.

IIT DELHI, IISc BANGALORE | RESEARCH INTERN

Dec 2021 - Mar 2023 | India

- Extended multi-party computation framework **MOTION2NX** from 2-party to N-party settings; achieved 1.5x speedup in private set intersection benchmarks.
- Extended symbolic execution engine **SKLEE** for automated vulnerability detection in smart contracts; categorized 100+ real-world Solidity vulnerabilities with formal analysis.

PUBLICATIONS & WRITING

2026 Free will as a model parameter (LessWrong)

2026 The Correctness Gap (Medium)