

Raj Patil

Backend / AI-Infrastructure Engineer

Pune, India | +91 89832 31242 | rpdinkar92260@gmail.com
github.com/Raj-glitch-max | linkedin.com/in/raj-patil-311b6b259 | rajpatil.dev

Summary

Backend engineer building the control plane for AI agents — authorization, evaluation, and incident response — in Go and Python. Shipped a live capability-delegation service bound to SPIFFE workload identity, and a Kubernetes benchmark that surfaced a named LLM diagnosis failure mode. AWS Solutions Architect Associate; ships own CI/CD and deploys.

Technical Skills

Languages: Python, SQL, Bash, GO

AI / Agent Infrastructure: Model Context Protocol (MCP), agent authorization, LLM evaluation harnesses, tool & function calling, structured outputs, RAG

Backend: gRPC, REST, JSON-RPC, FastAPI, Flask, PostgreSQL, Go concurrency

Cloud & DevOps: AWS (EC2, IAM, S3, CloudWatch), Docker, Kubernetes, GitHub Actions, Nginx, Linux

Security: SPIFFE/SPIRE workload identity, capability-based authorization, mTLS, JWT, least-privilege design

Projects

Atlas — Capability Delegation & Authorization for AI Agents

Go, SPIRE, gRPC, MCP | Code | Live

- Replaced static shared API keys with scoped, attenuable delegation tokens for AI agents — each bound to **SPIFFE workload identity** via SPIRE and verified fully offline, with no network hop on the authorization path.
- Passed a **28-vector conformance suite (18 adversarial)** — replay, scope escalation, clock skew, stale trust bundle — after an audit surfaced **12 deployment-phase defects** a fully passing unit test suite had hidden.
- Caught and fixed a signature time-unit defect (seconds vs. nanoseconds) and a trust-bundle refresh failure that silently rejected all traffic; hardened with fail-closed revocation on short-TTL refresh.
- Shipped an **MCP server plus SDKs in 3 languages** (Go, Python, TypeScript) and deployed live on Railway + Cloudflare Pages; now enforces least privilege on every tool call inside AISRE.

AISRE — Agentic Incident-Response Pipeline

Python, Flask, NVIDIA NIM (Nemotron), Docker | Code

- Built an end-to-end pipeline that detects a live service failure, investigates it with a **3-tool orchestrator agent**, and emits a structured-JSON root cause before any fix is applied.
- Gated every remediation behind human approval and automatic post-fix health re-verification, enforcing a hard **zero-financial-API-exposure** boundary on all agent tool calls.
- Verified **7 phases end-to-end in a live browser session** — detection, investigation, approval, remediation, re-verification — with Atlas gating agent capability and KLRB designed in as the RCA evaluator.

KLRB — Kubernetes LLM Root-Cause Benchmark

Python, Kubernetes, Chaos Engineering | Code

- Injected reproducible chaos faults into live Kubernetes clusters and scored how faithfully LLM agents diagnose root cause from real cluster evidence — events, logs, and metrics.
- Surfaced the “**Confident Liar**” effect via an evidence-ablation metric: models that keep their conclusion unchanged after the causal evidence is removed, separating real diagnosis from confident guessing.
- Built a multi-provider runner behind one adapter interface with deterministic scenario replay, making cross-model scores directly comparable instead of run-to-run noise.

Education

MIT ADT University, Pune

B.Tech, Electronics & Computer Engineering

Aug 2022 – May 2026

CGPA: 7.71/10

Certifications

AWS Certified Solutions Architect – Associate | AWS Certified AI Practitioner | AWS Certified Cloud Practitioner | GitHub Copilot Certified | Redis Certified Developer (Python)