SuperOps

Powered by aws | H2S

Building the Future of Agentic AI For IT Management

Team Name : BNSpace

Team Leader Name : Balaji Narayanamoorthy

Problem Statement : Operational Efficiency Improvement for MSPs and IT Teams

Brief about the Idea:

AutoOps AI is an **Agentic AI-powered platform** designed to make IT operations self-driving for MSPs and IT teams. It intelligently handles **patch management, alert management, and routine IT administrative tasks** by continuously monitoring systems, correlating alerts, prioritizing vulnerabilities, and triggering automated remediation — all within policy-defined guardrails.

Unlike traditional RMM tools that rely only on static scripts, AutoOps AI uses **LLM-driven decision-making** to analyze, decide, and act autonomously while learning from human feedback. This results in **faster resolution, fewer missed incidents, and lower manual workload** — directly improving operational efficiency and profitability for MSPs.

How different is it from any of the other existing ideas?

AutoOps AI goes beyond traditional RMM and automation tools by using **Agentic AI** to not just execute pre-defined scripts but to **analyze, decide, and act autonomously** within safe guardrails. Unlike rule-based systems, it continuously learns from human feedback, improving accuracy and efficiency over time.

How will it be able to solve the problem?

AutoOps AI directly addresses patch chaos, alert fatigue, and repetitive IT tasks by:

- **Prioritizing and scheduling patches** intelligently to minimize downtime and maximize compliance.
- **Correlating and deduplicating alerts** to reduce noise and focus on root causes.
- **Executing automated remediation and routine tasks** such as service restarts, cleanup jobs, and user lifecycle actions — freeing engineers from manual work.
- **Providing a natural language interface** so teams can query and control operations easily.

USP of the proposed solution:

- **Policy-driven autonomy:** IT admins stay in control while AI handles the heavy lifting.
- **End-to-end coverage:** Patch management, alert management, and task automation — all in one platform.
- **Continuous learning:** AI adapts to the MSP's unique environment, improving decision-making with every interaction.
- **Seamless integrations:** Works with existing RMM, ticketing, and alerting tools, reducing adoption friction.

✅ Patch Management:
• Endpoint discovery, patch prioritization, safe rollout & rollback

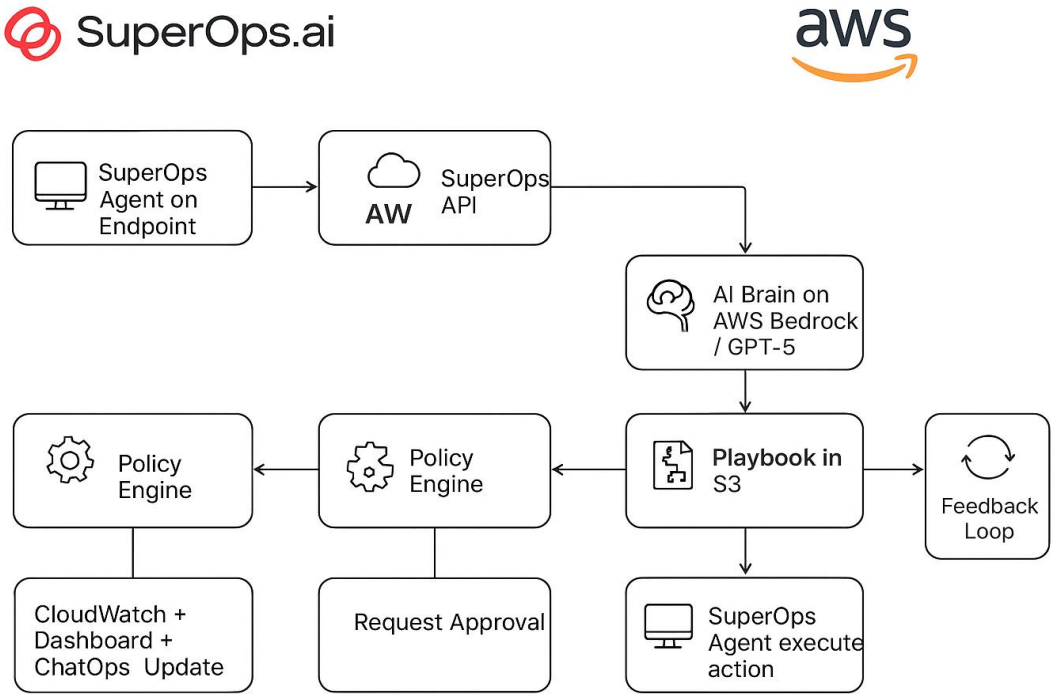
✅ Alert Management:
• Noise suppression, correlation, AI-driven root cause analysis

✅ Automated IT Tasks:
• Service restarts, log cleanup, compliance enforcement, user lifecycle automation

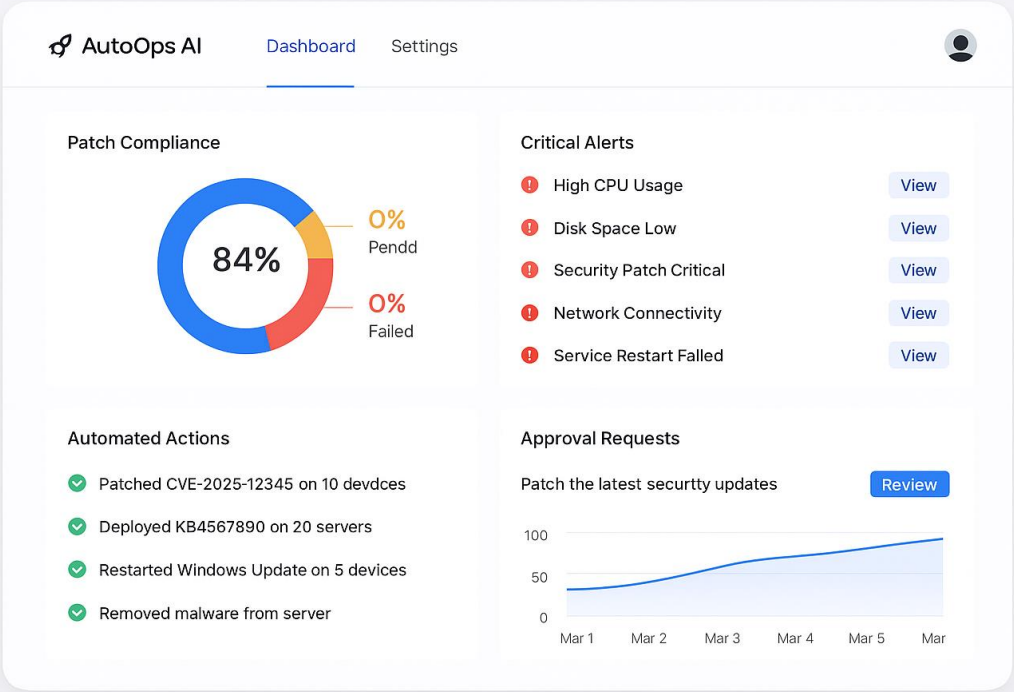
✅ Policy-Driven Autonomy:
• Admins define guardrails, AI takes safe automated actions

✅ ChatOps Interface:
• Slack/Teams integration for approvals, real-time insights

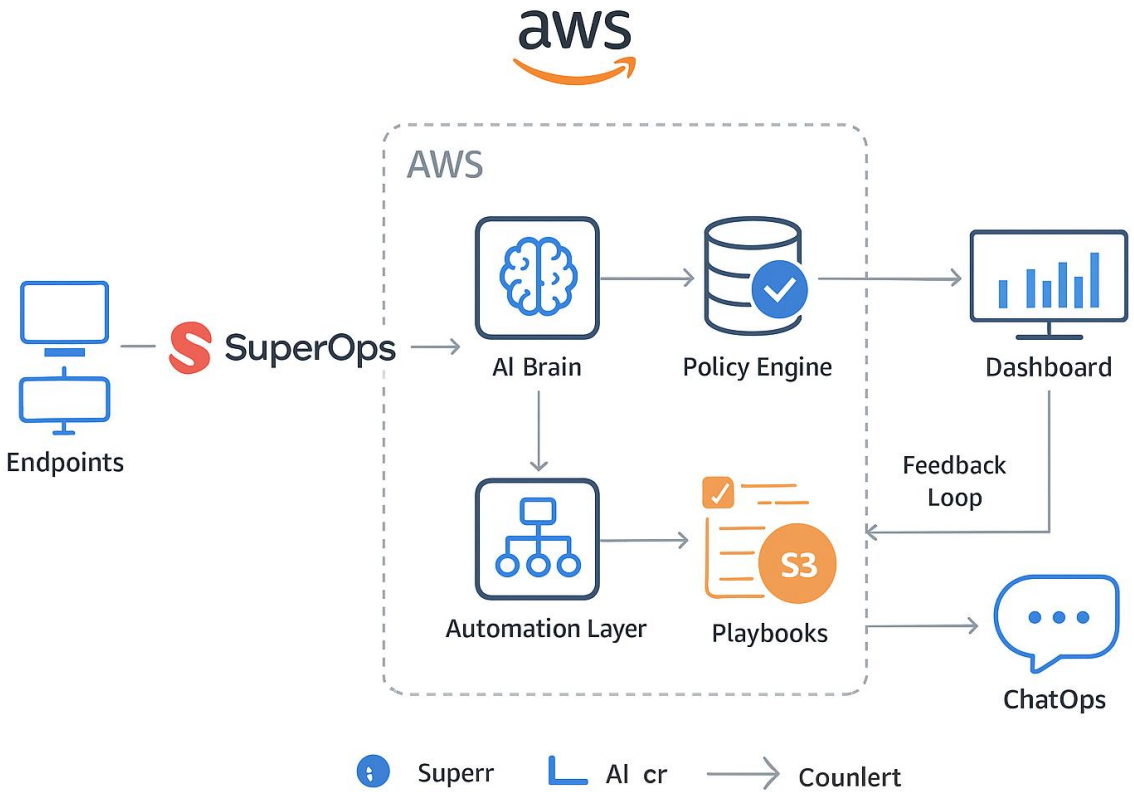
Process flow diagram or Use-case diagram



Wireframes/Mock diagrams of the proposed solution



Architecture diagram of the proposed solution



Technologies to be used in the solution:

Layer	Technology / Service	Purpose
Endpoint Management	SuperOps.ai Agent & APIs	Collect device inventory, patch status, alerts, and execute remote scripts or patches.
AI & Orchestration	AWS Bedrock (or GPT-5 API) + LangChain / CrewAI	Reasoning engine to prioritize patches, correlate alerts, decide remediation actions.
Policy Engine	AWS DynamoDB (storing JSON/YAML policies)	Stores automation guardrails, thresholds, and rules for safe auto-remediation.
Automation & Remediation	AWS Lambda + AWS Step Functions	Event-driven execution of playbooks, connecting AI decisions with SuperOps agent actions.
Playbook Repository	AWS S3	Stores patching scripts, remediation workflows, rollback scripts with version control.
Memory & Context	Amazon OpenSearch Service (Vector DB)	Maintains historical incidents, decisions, and user feedback for AI learning.
Monitoring & Logging	AWS CloudWatch + CloudTrail	Centralized logging, decision traceability, and auditing of AI actions.
Dashboard & UI	React + TailwindCSS (hosted on AWS Amplify)	Displays patch compliance, alerts, AI actions, and provides manual control.
ChatOps Interface	Slack/Teams Bot (AWS API Gateway + Lambda)	Approval workflow and natural language control of AutoOps AI actions.
Data Sources	NVD CVE API, Vendor Patch Feeds	Real-time vulnerability and patch information for prioritization.

Estimated implementation cost (optional):

Component	Tool/Service	Estimated Cost (1 Month)	Notes
LLM / AI Reasoning	AWS Bedrock (Claude/GPT-like) or GPT-5 API	~$100 – $150	Assume ~50k–100k tokens/day during dev & testing
Compute (Backend)	AWS Lambda, API Gateway	<$30	Pay-per-use, no servers required
Workflow Orchestration	AWS Step Functions	~$10	Low-cost for small number of flows
Data & Storage	DynamoDB (policy store), S3 (playbooks)	<$10	Minimal storage footprint
Monitoring & Logs	CloudWatch, CloudTrail	~$15	Audit + log retention for demo
SuperOps Agent + API	Existing RMM Agent	Included (SuperOps sandbox or trial)	No extra cost for hackathon prototype
Frontend Hosting	AWS Amplify (React dashboard)	~$5	For demo UI hosting
Slack/Teams Bot	AWS Lambda + API Gateway	~$5	Minimal execution cost
Miscellaneous	Testing, CI/CD, small EC2 for simulations	~$50	Optional if you spin up test endpoints

SuperOps

Powered by aws | H2S

Building the Future of Agentic AI For IT Management

THANK YOU