**Project Title: DevOps AI Agent – Infrastructure Monitoring, Anomaly Detection & Remediation**

**Name: Lokesh Bheemagani**

**Date: July 2025**

**GitHub Link:** [**https://github.com/Lokesh-2003/DevOpsAgent-Task-Lokesh-Bheemagani**](https://github.com/Lokesh-2003/DevOpsAgent-Task-Lokesh-Bheemagani)

**Table of Contents**

1. Introduction
2. Tools & Technologies
3. Architecture
4. Setup Instructions
5. Slack Notifications
6. Screenshots
7. Conclusion

**1.Introduction**

This project implements an AI-powered DevOps Agent that automates infrastructure monitoring, anomaly detection ( CPU spikes), root cause analysis using LLMs, and automated remediation actions. The agent also sends real-time Slack notifications**.**

**2.Tools & Technologies**

|  |  |
| --- | --- |
| **Category** | **Tools Used** |
| Monitoring | Prometheus, Node Exporter |
| AI Log Analysis | OpenAI GPT API (LLM) |
| Container Management | Docker |
| Remediation | Python + Shell Commands |
| Notifications | Slack Bot API |
| Platform | AWS EC2 (Ubuntu 22.04) |

**3.System Architecture**

**[ EC2] --> [ Prometheus] --scrapes--> [ Node Exporter]**

**Python Agent (main.py)**

**Monitors Metrics**

**Analyzes Logs (GPT)**

**Remediates (e.g., Restart Service)**

**Sends Slack Alerts**

**4.Setup Instructions**

1. **SSH into EC2:**

chmod 400 DevOps-AI.pem

ssh -i "DevOps-AI.pem" ubuntu@ec2-13-203-220-150.ap-south-1.compute.amazonaws.com

1. **Update and Install Tools:**

sudo apt update && sudo apt upgrade -y

sudo apt install -y python3-pip python3-venv docker.io git curl unzip

sudo systemctl enable --now docker

1. **Clone the Repo**

git clone https://github.com/Lokesh-2003/DevOpsAgent-Task-Lokesh-Bheemagani.git

cd DevOpsAgent-Task-Lokesh-Bheemagani

1. **Create and Activate Virtual Environment**

python3 -m venv venv

source venv/bin/activate

pip install -r requirements.txt

1. **Set Environment Variables**

export SLACK\_BOT\_TOKEN='slack-token'

export OPENAI\_API\_KEY='openai-key'

1. **Install Prometheus + Node Exporter**
2. **Download and run Node Exporter**

cd ~

wget https://github.com/prometheus/node\_exporter/releases/download/v1.7.0/node\_exporter-1.7.0.linux-amd64.tar.gz

tar xvf node\_exporter-1.7.0.linux-amd64.tar.gz

cd node\_exporter-1.7.0.linux-amd64

./node\_exporter &

1. **Download and run Prometheus**

cd ~

wget https://github.com/prometheus/prometheus/releases/download/v2.52.0/prometheus-2.52.0.linux-amd64.tar.gz

tar xvf prometheus-2.52.0.linux-amd64.tar.gz

cd prometheus-2.52.0.linux-amd64

nano prometheus.yml

1. **Edit prometheus.yml**

global:

scrape\_interval: 15s

scrape\_configs:

- job\_name: 'node'

static\_configs:

- targets: ['localhost:9100']

1. **Start Prometheus:**

./prometheus --config.file=prometheus.yml &

1. **Verify manually**

curl -G http://localhost:9090/api/v1/query \

--data-urlencode 'query=100 - (avg by(instance)(irate(node\_cpu\_seconds\_total{mode="idle"}[2m])) \* 100)'

1. **Add Slack Bot and Permissions**
   1. Go to Slack Apps, create your app (e.g., DevOpsBot).
   2. Under OAuth & Permissions, add scopes: chat:write, chat:write.public, channels:read.
   3. Click Install to Workspace, authorize, and copy the Bot User OAuth Token.
   4. Ensure your bot appears in the #general channel:

In Slack: /invite @DevOpsBot

1. **Start Agent**

python3 main.py

1. **Final Execution**

cd ~/DevOpsAgent-Task-Lokesh-Bheemagani

source venv/bin/activate

python3 main.py

1. **Test Everything**

yes > /dev/null &

echo "Error: Infinite loop detected in test-app" | sudo tee -a /var/log/syslog

1. **Test the Slack Alert**

curl -X POST \

-H "Authorization: Bearer $SLACK\_BOT\_TOKEN" \

-H "Content-type: application/json; charset=utf-8" \

--data '{"channel":"#general", "text":"✅ Slack alert working from DevOpsAgent"}' \

https://slack.com/api/chat.postMessage

**5.Slack Notifications**

* Slack alerts are sent using chat.postMessage
* Format:

{

"channel": "#general",

"text": "⚠️ CPU Spike Detected on instance X",

}

**6.Screenshots**

**7.Conclusion**

This project demonstrates an automated DevOps agent capable of detecting anomalies, analyzing root causes via LLMs, and remediating issues without manual intervention. It integrates multiple technologies and is ideal for real-world SRE/DevOps scenarios.