Self-Healing Infrastructure Project Report

Name: Amgoth Pravalika
Date: 26/04/2025

---

1. Introduction

This project implements a **self-healing infrastructure** using **Prometheus** (monitoring), **Alertmanager** (alerting), and **Ansible** (automated remediation). The system detects service failures (e.g., NGINX crashes) and automatically restores them without manual intervention, improving reliability and reducing downtime.

---

2. Abstract

- **Objective:** Automatically detect and recover failed services using monitoring and automation.
- **Approach:**
  - **Prometheus** scrapes metrics and triggers alerts when a service fails.
  - **Alertmanager** forwards alerts to a custom webhook.
  - **Ansible** executes a playbook to restart the failed service.
- **Outcome:** A fully automated system that maintains service availability.

---

3. Tools Used

Tool	Purpose
Prometheus	Monitoring & alert rule evaluation
Alertmanager	Alert routing & notification
Ansible	Automated remediation (restarting services)

Tool	Purpose
Docker	Containerization of services
EC2 (Ubuntu)	Hosting the infrastructure
GitHub	Version control & project hosting

4. Steps Involved

1. Infrastructure Setup

- Launched an **EC2 Ubuntu** instance with necessary ports open (80, 9090, 9093).
- Installed **Docker, Docker Compose, and Ansible**.

2. Monitoring with Prometheus

- Configured prometheus.yml to monitor:
  - **NGINX** (HTTP service)
  - **Node Exporter** (system metrics)
- Defined alert rules (alert.rules.yml) for:
  - **NGINX downtime** → Triggers if NGINX is down for 1 minute.
  - **High CPU usage** → Triggers if CPU > 90% for 5 minutes.

3. Alerting with Alertmanager

- Configured alertmanager.yml to send alerts to an **Ansible webhook**.

4. Self-Healing with Ansible

- Developed a **Flask-based webhook** (webhook.py) to receive alerts.
- Created an **Ansible playbook** (restart_nginx.yml) to restart NGINX when an alert fires.

5. Deployment

- Used **Docker Compose** to run all services:

```
docker-compose up -d --build
```

6. Testing

- **Simulated NGINX crash:**
  docker-compose stop nginx
- **Verified auto-recovery:**
  - Prometheus detected failure → Alertmanager triggered → Ansible restarted NGINX.
  - Confirmed NGINX was back online (curl http://localhost).

---

5. Conclusion

Successfully implemented a self-healing system that automatically recovers failed services.
Reduced manual intervention by automating remediation with Ansible.
Demonstrated DevOps principles (monitoring, alerting, automation).

GitHub Link: [https://github.com/Pravalika-27/self-healing-infrastructure.git]