

Challenge: DevOps Workflow Implementation

Task Overview:

You will create a simple web application and set up a complete DevOps workflow to deploy, monitor, and manage it. The tasks will include using Linux commands, version control with Git, containerization with Docker, orchestration with Kubernetes, setting up a CI/CD pipeline, and implementing logging and monitoring.

Steps:

1. Setup and Version Control:

- Create a simple web application (e.g., a basic Flask app for Python or a simple Node.js app).
- Initialize a Git repository and commit your code.
- Create a remote repository on a platform like GitHub or GitLab and push your local repository to it.

2. Dockerization:

- Write a Dockerfile to containerize your application.
- Build and run your Docker container locally to ensure it works.

3. Kubernetes Deployment:

- Set up a local Kubernetes cluster using Minikube.
- Write Kubernetes manifests (Deployment, Service) to deploy your Dockerized application to the cluster.
- Deploy your application to the Kubernetes cluster and ensure it is accessible.

4. CI/CD Pipeline:

- Set up Jenkins on your local machine or Minikube cluster.
- Create a Jenkins pipeline script (Jenkinsfile) to automate the build, test, and deployment of your application.
- Integrate static code analysis and security scans using tools like SonarQube and Trivy in your pipeline. (Bonus)

5. Monitoring:

- Set up Prometheus for monitoring your Kubernetes cluster.
- Configure Prometheus to scrape metrics from your application.
- Set up Grafana and create a dashboard to visualize the metrics collected by Prometheus.
- Define basic monitoring thresholds and alerts in Prometheus.