DevOps Lab

Assignment 1:

Aim: To understand DevOps principles, practices and roles and responsibilities of a DevOps Engineer.

Theory:

DevOps is a set of practices that combines software development (*Dev*) and IT operations (*Ops*). DevOps is a software development strategy which bridges the gap between the developers and the IT staff. With DevOps, organizations can release small features very quickly and incorporate the feedback which they receive, very quickly. Following are some other benefits:

1. Fewer Software Failure
2. Shortened lead time between fixes

It overcomes all the limitations of the traditional waterfall model. DevOps process involves a lot of development, testing and deployment technologies for developing automated CI/ CD pipelines. Following are some of the famous DevOps tools:

1. **Git and GitHub** — Source code management (Version Control System)
2. **Jenkins** — Automation server, with plugins built for developing CI/ CD pipelines
3. **Selenium** — Automation testing
4. **Docker** — Software Containerization Platform
5. **Kubernetes** — Container Orchestration tool
6. **Puppet** — Configuration Management and Deployment
7. **Chef** — Configuration Management and Deployment
8. **Ansible** — Configuration Management and Deployment
9. **Nagios** — Continuous Monitoring

Principles and Best Practices of DevOps:

1. Foster a Collaborative Environment: The main theory behind DevOps is to combine development and operations to create a unilateral team that focuses on delivering common objectives.
2. Implement Automation: Continuous integration and delivery allows developers to merge code regularly into the main repository. Instead of manually checking code, CI/CD automates this process, from batching in a specified window to frequent commits. In
3. Monitor the DevOps pipeline and applications: It's important to monitor the DevOps pipeline so a broken build or failed test doesn't cause unnecessary delays. Automation improves the speed of development tremendously, but if there is a failure in an

automated process and nobody knows about it, you're better off doing the work manually.

4. Gather continuous feedback: Continuous feedback ensures team members have all the information needed to do their jobs on a timely basis. From the development perspective this entails that the team is alerted to any pipeline failures immediately. It also means that clear, thorough code test results are made available to the developers as quickly as possible.

5. Build with the right tools: A DevOps toolchain requires the right tools for each phase of the DevOps lifecycle, with key capabilities to improve software quality and speed of delivery. Read more about how to choose DevOps tools and review functionality for each phase of the DevOps lifecycle.

6. Implement Cross-Functional autonomous teams: In product organizations with vertical, fully responsible teams, these teams need to be entirely independent throughout the whole lifecycle. That requires a balanced set of skills and also highlights the need for team members with T-shaped all-round profiles instead of old-school IT specialists who are only knowledgeable or skilled in for example testing, requirements analysis or coding. These teams become a hotbed of personal development and growth.

7. Follow a customer first approach: It is imperative nowadays to have short feedback loops with real customers and end-users, and that all activity in building IT products and services centres around these clients. To be able to meet these customers' requirements, DevOps organizations require the guts to act as lean start-ups that innovate continuously, pivot when an individual strategy is not (or no longer) working, and constantly invests in products and services that will receive a maximum level of customer delight.

Roles and Responsibilities of DevOps Engineer:

1. Documentation: Writes specifications and documentation for the server-side features.

2. Systems analysis: Analyses the technology currently being used and develops plans and processes for improvement and expansion. The DevOps engineer provides support for urgent analytic needs.

3. Development: Develops, codes, builds, installs, configures, and maintains IT solutions.

4. Project planning: Participates in project planning meetings to share their knowledge of system options, risk, impact, and costs vs. benefits. In addition, DevOps engineers communicate operational requirements and development forecasts.

5. Testing: Tests code, processes, and deployments to identify ways to streamline and minimize errors.

6. Deployment: Uses configuration management software to automatically deploy updates and fixes into the production environment.

7. Maintenance and troubleshooting: Performs routine application maintenance to ensure the production environment runs smoothly. Develops maintenance requirements and procedures.

8. Performance management: Recommends performance enhancements by performing gap analysis, identifying alternative solutions, and assisting with modifications.

Conclusion: Thus, successfully understood the principles and best practices of DevOps and also understood the roles and responsibilities of a DevOps Engineer.