Krish Chatwani
T11 16

THADOMAL SHAHANI TSEC ENGINEERING COLLEGE

SEPM

To understand DevOps: Principals, Practices & DevOps Engineer Role, & Responsibilities.

What is DevOps ?
DevOps is a collabarative approach where teams work together to build and deliver secure software efficiently It combines software development (dev) and operations (ops) to accurately deliver through automation, collaboration, fast ~~feets~~ feedback and iterative improvement. built on agile methodology, Devops creates a culture of accountability collaboration & shared responsibility for business outcomes.

Core Principals of DevOps:-
- Develop & Test in production - like environments
- Develop build frequently.
- Continuously validate operational quality.

Key Practices of DevOps.

1. Continuous Deployment
Continous delivery & deployment originate form continuous integration, a method to rapidly develop, build & Test new code with automation to That only code that is known to be good becomes part of software product.

2. Continoous Development.
This is the phase That involves planning & coding visioning and managing builds of The slw application functionality. Eg:- Git, Github, Maven.

3. Continous Testing.
Continoous testing is executed automated tests, continuously and repeated against the code base and the repeated various deployment environments. It is a software testing methodology which focuses on achieving continuous quality & improvement.
Eg: Appium, Banboo.

4. Continuous Integration.
Continuous Integration refers to the build & unit-testing stages of the software release process. Every revision that is commited triggers an automated build & test.
Eg:- Jenkins, Travis CI.

5. Infrastructure Management
Without automation, build & maintaining large scale modern without automation, IT system can be a resource intensive undertaking & can lead to increased risk due to manual error. Configuration & resource management is an automated method for maintaining computer systems & software in a known consistent attack.

6. Configuration Management:
Infrastructure as code is the practice of describing software runtime environment & networking setting & parameters in simple textual format, that can be stored in your version control systems and versioned on request. These test files are called manifests & are used by devops tools to automatically provision &

configure build servers, testing, staging & production environment.
Eg:- Chef, Saltstack.

DevOps Engineer Role
A DevOps engineer manages a company i.e IT infrastructure, bridging development & operation, the primary goal is to improve the process and efficiency throughout the software development lifecycle.

Key Role:-
1. Facilitator of Collaboration:-
Bridging the gap between development, operations, & QA teams to streamline communication.
2. Automation speacilist.
Automate repititive tasks like testing, deployment & monitoring.
3. Continuous Integration & Continuous Delivery (CI/CD).
Design, implement & maintain CI/CD pipelines to enable faster, reliable & repeatable software releases
4. Infrastructure as Code
use tools like Terraform, Ansible or cloud formation to define & provision infrastructure through code.
5. Monitoring & Incident Management.
Set up monitoring system to track application performance and troubleshoot issue in real time. It also encloses that systems are resistent and downtime is mimimized.

6. Cloud & Infrastructure Management.
Deploy, manage & optimize applications on cloud platform like Aws, Azure, or Google Cloud also handles container orchestration.

Key Responsibilities:-

1. Collaboration & Planning:-
Work with development & operations teams to plan & design scalable solution.
2. Configuration Management:
Uses tools like puppet, Chef on Ansible to manage server configuration & ensure consistency.
3. Pipeline Management
Maintain CI/CD pipelines to ensure consistency, test & deployment work flows.
4. Monitoring & logging.
Implement monitoring tools like Prometheus, Grafana or splunk to track system health & measurements performance.
5. Support & Troubleshooting:
Respond to incidents & resolve production issues promptly & identify root causes of failure & implement fines.
6. Documentation & Reporting:
Document system configurations, deployment process & troubleshooting guides.