DevOps Interview Preparation

Key Concepts

- **CI/CD Pipelines**: Jenkins, GitHub Actions, GitLab CI, Azure DevOps
- **Infrastructure as Code (IaC)**: Terraform, Ansible, CloudFormation
- **Containers**: Docker, Podman
- **Container Orchestration**: Kubernetes, OpenShift
- **Monitoring & Logging**: Prometheus, Grafana, ELK/EFK stack
- **Version Control**: Git fundamentals (branching, merging, rebasing)
- **Configuration Management**: Ansible, Puppet, Chef
- **Cloud Platforms**: AWS, Azure, GCP basics
- **Networking**: Load balancers, DNS, VPC, service discovery
- **Security in DevOps**: Secrets management (Vault, KMS), shift-left security, vulnerability scanning
- **Agile & Collaboration**: Scrum, Kanban, DevOps culture

Example Questions

1. What is the difference between Continuous Integration, Continuous Delivery, and Continuous Deployment?
2. How does Docker differ from a virtual machine?
3. What are the key components of Kubernetes?
4. How do you handle secrets in CI/CD pipelines?
5. Explain Infrastructure as Code (IaC) and why it is important.
6. How do you monitor applications in production?
7. What’s the difference between blue-green and canary deployment?
8. How would you ensure high availability in a microservices architecture?

Scenario-Based Questions

Scenario 1: Deployment Failure - Your application deployment pipeline fails after pushing to production. Logs show container image not found. - **Approach:** - Check pipeline configuration for correct image reference. - Verify image exists in registry (e.g., Docker Hub, ECR). - Ensure credentials/secrets are configured for image pull. - Redeploy after fixing reference.

Scenario 2: High Latency in Application - After deploying new microservices, users report high response times. - **Approach:** - Use monitoring (Prometheus, Grafana, APM) to trace latency. - Identify if it’s database, network, or service-level issue. - Check Kubernetes HPA (Horizontal Pod Autoscaler) for scaling issues. - Rollback if new code introduced the latency.

Scenario 3: Node Failure in Kubernetes - One Kubernetes node goes down unexpectedly. - **Approach:** - Verify node status using `kubectl get nodes`. - Check logs/events for node health. - Confirm pods rescheduled on healthy nodes. - Investigate root cause (hardware issue, kubelet crash).

Scenario 4: Secrets Leaked in Repo - A developer accidentally commits AWS credentials to GitHub. - **Approach:** - Revoke exposed credentials immediately. - Rotate secrets. - Use Git history cleaning tools (e.g., `git filter-repo`). - Implement pre-commit hooks / secret scanners.

Technical Hands-On Tasks

1. Write a Jenkinsfile that builds, tests, and deploys a Dockerized app.
2. Deploy an Nginx container using Kubernetes with a ConfigMap.
3. Write a Terraform script to launch an EC2 instance with a security group.
4. Configure Prometheus and Grafana to monitor a sample application.
5. Implement blue-green deployment for a microservice.

Common Tools & Their Interview Angles

- **Jenkins**: Pipelines, plugins, scripted vs declarative pipelines.
- **Kubernetes**: Pods, services, deployments, configmaps, secrets, ingress.
- **Docker**: Images, volumes, networking.
- **Terraform**: State management, modules, workspaces.
- **Ansible**: Playbooks, roles, inventory.
- **Git**: Merge conflicts, branching strategies, GitOps.
- **Monitoring**: Alerts, dashboards, metrics vs logs.

Behavioral/Philosophy Questions

- What does DevOps mean to you?
- How do you handle conflicts between developers and operations teams?
- Share an example of how you reduced deployment time in a project.
- How do you ensure security in CI/CD pipelines?

✅This covers theory, scenarios, and hands-on practice for DevOps interviews.