Ankush Pandey

Cloud & DevOps Engineer
Email:pandeyankush8959@gmail.com | Phone: +91 9424017449 | Location: Indore, MP, India |
Portfolio: https://ankush.fun | linkedin.com/in/ankush-pandey-658824401 | github.com/Ankush-97

PROFESSIONAL SUMMARY

Cloud & DevOps Engineer with 10 months of hands-on experience in automating, containerizing, and scaling multi-tier production architectures. Proficient in architecting high-availability infrastructure on AWS using Terraform (IaC), orchestrating decoupled microservices with Docker Compose, and deploying secure CI/CD pipelines via Jenkins and SonarQube. Strong expertise in event-driven serverless automation using AWS Lambda and full-stack observability with Datadog, with a proven track record of optimizing system reliability and debugging complex cloud environment bottlenecks.

WORK EXPERIENCE

DevOps Engineer Intern | Hisan Lab Private Limited 01/2026 - Present

- Designed and deployed an end-to-end containerized **3-tier microservices web application** on AWS EC2, optimizing service communication and data isolation using secure Docker bridge networking and hardened **MariaDB** nodes.
- Automated production-ready AWS VPC infrastructure using modular Terraform (IaC), implementing high-availability architectures with Multi-AZ subnets, **Application Load Balancers (ALB)**, and **Auto Scaling Groups (ASG)**.
- Wrote optimized custom **Dockerfiles** using efficient base images e.g., eclipse-temurin **to reduce image footprints by 40%** and drastically accelerate deployment speeds across the decoupled stack.
- Implemented event-driven serverless workflows using **AWS Lambda** to automate critical cloud operations, including automated database snapshot triggers, custom log rotations, and S3 object event processing.
- Built multiple automated **multi-stage CI/CD pipelines** using **Jenkins Master-Worker architecture**, seamlessly integrating **SonarQube Quality Gates** to enforce coding standards, perform static analysis, and block security vulnerabilities.
- Established full-stack observability by engineering proactive **Datadog** and **AWS CloudWatch** dashboards, complemented by custom **Bash scripts** for automated host operations and log management.

TECHNICAL SKILLS

- **Cloud Platforms:** Amazon Web Services (AWS - EC2, VPC, S3, RDS, CloudFront, EFS, IAM, Security Groups, Lambda, AWS SES, AWS ACM, Route53) | Microsoft Azure |Google Cloud Platform (GCP)
- **Containerization & Orchestration:** Docker, Docker Hub, Kubernetes (K8s)
- **Infrastructure as Code (IaC):** Terraform
- **CI/CD Tools:** Jenkins
- **Version Control & Collaboration**: Git, GitHub
- **Monitoring & Observability:** AWS Cloud Watch, Datadog
- **Operating Systems & Scripting Languages:** Linux (Ubuntu/CentOS), Bash, Shell Scripting
- **Networking & Security:** TCP/IP, SSH Key Management, CORS, Custom VPC Architecture
- **Databases:** MySQL, MongoDB, MariaDB
- **Build Tools:** Maven, npm
- **Code Quality:** SonarQube

EDUCATION

Master of Science in Computer Science (M.Sc. CS) *06/2023 – 06/2025*
Awadhesh Pratap Singh University (APSU), Rewa

PROJECTS

ShopiTry – Microservices E-Commerce Platform | Github-Repo

- **Architecture**: Engineered a decoupled 6-service microservices ecosystem reducing backend API latency by 35% using Node.js and MongoDB Atlas.
- **Cloud Infrastructure**: Hosted production services on AWS EC2 using PM2, improving frontend asset delivery speed by 40% via S3 and CloudFront.
- **DevOps & Security**: Configured NGINX reverse proxy and Let's Encrypt SSL, reducing deployment rollout time by 25% across environments.
- **Hardening**: Integrated JWT and RBAC policies, blocking 99% of unauthorized endpoints via environment variable hardening and rate-limiting.

Technologies / Tools Used: Node.js, Express.js, MongoDB Atlas, React, Vite, AWS (EC2, S3, CloudFront), NGINX, PM2, Docker, Git, Let's Encrypt SSL

3-Tier Microservices Application Deployment using Docker Compose & AWS | Github-Repo

- **Architecture:** Designed and deployed a containerized 3-tier microservices application on AWS EC2, ensuring 99.9% environment consistency.
- **Orchestration:** Utilized Docker Compose to containerize Apache, Java 17, and MariaDB, reducing application startup times by 30%.
- **Networking:** Configured isolated bridge networks for secure container DNS communication, eliminating 100% of hard-coded IP addresses.
- **Security:** Implemented custom CORS policies and strict AWS Security Groups, achieving efficient cross-tier data routing with zero leakage.

Technologies / Tools Used: Docker, Docker Compose, AWS EC2, Apache, Java 17, MariaDB, Linux, AWS Security Groups

Production-Grade High Availability AWS Architecture via Terraform | Github-Repo

- **Automation:** Automated a highly available cloud architecture on AWS using Terraform modules, reducing environment setup times by 50%.
- **Custom Networking:** Structured Terraform modules to deploy custom VPCs and Multi-AZ subnets, ensuring 100% automated infrastructure provisioning.
- **Scaling:** Integrated an Application Load Balancer (ALB) with an Auto Scaling Group (ASG) to dynamically handle up to 3x sudden traffic spikes.
- **Resilience:** Enabled self-healing Nginx web instances to scale dynamically based on demand, maintaining zero downtime under peak load testing.

Technologies / Tools Used: Terraform (IaC), AWS (VPC, ALB, ASG, NAT Gateway, EC2), Nginx Web Server, Linux, AWS Security Groups

LANGUAGES

English | Hindi