Cloud & DevOps Engineer
Bala Krishna
Email: balakrishna09071995@gmail.com
Phone: +91-9182734703

PROFESSIONAL SUMMARY:

- Having around **4** years of experience as a **Cloud & DevOps, Build and Release engineer**.
- Experience in setting up project baselines, branching strategies, merging and taking regular backups of the source code.
- Experience in administration, maintenance of source control management systems, such as **Sub Version (SVN), Tortoise SVN and GIT**.
- Strong knowledge on Administration and configuration activities on **GIT**.
- Worked as **GIT** administrator as part of my role and monitored the repository for various issues.
- Handful experience on **continuous integration**(**CI**) and **continuous delivery**(**CD**) using **Jenkins**
- Experience in **Build, Package and Deploy** to all the environments by targeting to **Tomcat, Apache** and **WebLogic Servers.**
- Extensive experience in using Build Automation DevOps tools like **ANSIBLE, Maven, and Jenkins**.
- Worked on Amazon Web Services as **EC2, VPC, Cloud Watch, Auto Scaling, Lamda, ALB, SNS, SQS, S3, WAF, IAM, CloudFormation, EKS, ECR, AWS Config, SecretesManager, AWS SSM, Catalog, Athena**.
- Deployed and configured application on **AWS EC2** Instances and microservices on **EKS**.
- Hands on experience on remote tools **SSH, SCP, Winscp** and **Putty**.
- Excelled on creating **AMI** (AWS Machine Image) that utilizes **ELB** (Elastic Load Balancer) and **Auto Scaling**.
- Create and configure **S3** bucket with restricted policies.
- Installed and Setup **Web servers** (Apache and Tomcat), DB Server.
- Experience with administrative tasks with **CI tools** like maintaining configurations and backups and maintain high availability.
- Knowledge on configuring and Deploying Applications in the Client Servers using **Ansible** Master.
- Experience using **Nagios, ELK, CloudWatch, Prometheus, Grafana** monitoring system.
- Worked with different Bug Tracking Tools like **Jira**.
- Good experience on **Linux** and **Windows** environments.
- Ability to work successfully as part of small to medium sized team or independently.
- Implemented Maven-based unit testing and code coverage integrations within Jenkins CI pipelines to improve build quality and reliability.
- Experience in configuration and security using SSL/TLS certificates
- Experience in deploying artifacts such as **WAR** and **EAR** using build automation tools like **Maven**.
- Built and deployed various **Ansible playbooks** and modules in the dev, test and pre-production environments and having experience in using **Ansible** as **configuration, orchestration** tool.
- Having an Experience in managing infrastructure as code using **Ansible** to provisioning **AWS** services as **EC2, S3**, **VPC, Nexus Repo** etc. and knowledge on managing other cloud platforms using **Ansible playbooks.**
- Proficient in implementing and adapting to **Release/Change** Management Process.
- Having Experience in creating Custom **Docker images** using Docker file.
- Experience working on **Docker hub**, creating **Docker images** and handling multiple images primarily for middleware installations and domain configuration and container orchestration by **Kubernetes.**
- Having an Experience in managing infrastructure as code using **Terraform** to provisioning **AWS** services as **EC2, S3**, **VPC,** etc. and knowledge on managing other cloud platforms using Terraform.
- Adept at collaboration, documentation, and mentoring teams in DevOps best practices.

TECHNICAL SKILLS:

Cloud Platforms	AWS, Azure, IBM (OMS Sterling)
Container Tools	Docker, Docker Compose, Kubernetes (AKS,EKS, ECS), Helm

Version Control Tools	GIT, SVN, BitBucket, GitLab, Github,
Deployment Tools	Chef, Ansible,
Web/Application Servers	Apache 2.x,3.x, JBOSS, Tomcat, Web Logic (8/9/10), Nginx, Web Sphere 4.0/5.0, IIS.
Monitoring Tools	Nagios, Splunk, ELK , Grafana, Prometheus, NewRelic.
Testing & Ticketing Tools	SonarQube, Jira, ServiceNow,
CI & Build Tools	Jenkins, Maven, Ant,
Scripting Languages	Shell Scripting, Python, PowerShell, Yaml, JSON, Groovy
Languages	C, C++, C#, .Net, Java, J2EE, Node JS
Repository Management	Nexus, Jfrog Artifactory
Operating System	Unix, Linux, RHEL (4.x, 5.x, 6x), Ubuntu (16/14/13/12), CentOS, Windows Server 2003/2008
Networking/protocol	TCP/IP, DNS, NFS, ICMP, SMTP, DHCP, OSPF, BGP, LDAP and UDP.
Infrastructure as Code:	Terraform, CloudFormation
Scripting & Automation:	Shell scripting, Bash, YAML, JSON, Python

EDUCATIONAL QUALIFICATION:

- B. Tech in **CSE (**2014-2018**)** from Siddartha Institute of Technologies & Science- JNTUH

PROFESSIONAL EXPERIENCE:

Company: Barnes & Noble Education (BNED)
Duration: **23** may 2022 to Present
Role: Cloud & DevOps Engineer
Location: **Mumbai**

Project: Courseware Platform
Project Description: A digital platform for professors and students to share and collaborate on academic course materials and content.

Responsibilities:
- Built automated CI/CD pipelines with Jenkins, and IAC Automation with Terraform.
- Configured GIT with Jenkins and schedule jobs using POLL SCM option.
- Implemented Terraform modules for reusable infrastructure provisioning.
- Worked on Amazon Web Services (AWS) configuring various AWS services like EC2, S3, Glacier, ELB (Load Balancers),Auto Scaling Group, RDS,RDS Proxy, SNS, SQS, EFS, CloudFront, VPCs and EBS etc.
- Used AWS Cloud Front (Content delivery network) to deliver content from AWS edge locations drastically improving user experience and latency.
- Deployed Java applications on Apache Tomcat & Nginx.
- Configured CloudWatch alarms, logs, dashboards.
- Automated configuration using Ansible.
- Managed DNS, SSL, S3, snapshots, AMI lifecycle.
- Implementing and setting up Route 53 for AWS Web Instances ELB & Cloud Front on AWS Environment.
- Performed AWS cost optimization (unused AMIs, EBS, snapshots, EC2 right-sizing).
- Implemented DR strategy (backups, snapshots, multi-AZ)
- Performed extensive AWS cost optimization (unused AMIs, snapshots, idle EC2, EBS cleanup, resource
- right-sizing).
- Designed and implemented complete Disaster Recovery strategy, including automated backups, multi
- AZ failover, and cross-region snapshots.

- Created functions and assigned roles in AWS Lambda to run python scripts, and AWS Lambda using java to perform event driven processing. Created Lambda jobs and configured Roles using AWS CLI.
- Implemented and maintained the monitoring and alerting of production and servers/storage using AWS Cloud Watch.
- Created Cloud Watch dashboards for monitoring CPU utilization, Network In-Out, Packet In-Out and other parameters of the instances and notified those using SNS.

Impact & Achievements:
- Improved release delivery speed by 40% by automating end-to-end CI/CD pipelines using Jenkins & Terraform.
- Achieved 99.99% system uptime through Auto Scaling, load balancers, and CloudWatch-driven tuning.
- Reduced AWS monthly cost by 25% by cleaning unused AMIs, EBS volumes, snapshots, and right-sizing EC2.
- Implemented AMI & S3 lifecycle policies saving 30% storage overhead monthly.
- Strengthened platform reliability with automated backups, DR strategy, and multi-AZ failover.
- Reduced manual deployment effort by 80% using Ansible automation and reusable IaC modules.
- Delivered 25–30% cost reduction through automated cleanup and monitoring of unused resources.
- Ensured business continuity with a robust DR strategy improving RTO/RPO significantly.

Tools/Environment: AWS (EC2, ELB, CloudWatch, VPC), Terraform, Jenkins, Git, Ansible, Apache, Nginx, Tomcat, Nagios, RedHat, Shell scripting

Project: Direct Digital Platform **(DDP)**

Project Description: DDP is the central data hub and processing engine for BNED's digital education ecosystem. It acts as the "source of truth" for all institutional data, including courses, sections, enrollments, adoptions, and roster information.

Roles & Responsibilities:

- Experience in Build tools like Maven, Version control systems like **GIT** and **CI** Tools like **JENKINS**.
- Good experience in creating Branches, Tags and Release Tags activities and merging.
- Daily activities Included Force **builds**, daily patches.
- As per requests to Creating and maintaining the Development branches, Release branches, custom Branches and Tags.
- Performed **SCM** tasks, such as check-in, checkout, branching, tagging and merging code from development branches to main branch and main branch to development branches.
- Maintain the Access privileges of Jenkins, version control system to developers.
- Participated in all phases of **Build** and **Release** activities.
- Good experience in **Linux/Unix** environments.
- Monitoring the disk space and jobs in **CI** tools.
- Work with various development teams to build and support automated builds, supporting daily continuous integration.
- Designed EKS CI/CD pipelines using Jenkins and ArgoCD.
- Migrated services from ECS → EKS (hybrid architecture).
- Managed Amazon MSK (Kafka) and IBM MQ pipelines.
- Implemented autoscaling using AWS Karpenter.
- Managed Helm charts, Ingress, RBAC, namespaces, secrets.
- Built dashboards for MSK, ingestion health, EKS workloads.
- Conducted rolling/blue-green deployments.
- Implemented DR for EKS, RDS, MSK using cross-region replication.

Impact & Achievements:
- Reduced cluster cost by ~30% using Karpenter-driven autoscaling and optimized node provisioning.

- Increased pipeline stability by 60% through improved Jenkins/ArgoCD workflows and automated validation.

- Achieved 99.9% uptime for EKS microservices through scaling, failover, and GitOps automation.

- Improved data processing performance by 35% by optimizing MSK pipelines and ETL workflows.

- Enabled cross-region disaster recovery reducing RTO to under 15 minutes.

- Standardized Helm charts and GitOps structure, reducing deployment errors by 70%.

Tools/Environment: EKS, App Runner, API Gateway, Docker, Helm, Secrete manager, Git, Istio, Argo CD, AWS Karpenter

Project:**FDC** (First Day Complete) / **AIP** (Adoption & Insights Portal)
FDC /AIP is a comprehensive course material delivery program offered by BNED. Its primary goal is to ensure that every student receives all required course materials (digital and/or physical) automatically by the first day of class, typically bundled as part of tuition or fees. Adoption Information Platform. It is a system used primarily in educational institutions to manage and streamline the adoption of course materials

Roles & Responsibilities:
- Managed ECS microservices with Fargate, service discovery, blue/green deployments.
- Automated ETL & batch processing using Glue, Step Functions, EMR, AWS Batch, ECS, Step Functions.
- Implemented event-driven workflows (Lambda + MSK + MQ).
- Integrated MSK + MQ for event-driven flows
- Built Redis caching for real-time performance.
- Designed CodeBuild/CodeDeploy pipelines.
- Implemented multi-environment VPC deployments.
- Managed DR with multi-region backups and restore automation.

Impact & Achievements:
- Improved student material delivery reliability to 99.99% via ECS Fargate HA + blue/green deployments.
- Reduced compute costs significantly by optimizing EMR cluster sizing and Batch workloads.
- Reduced ingestion latency by 40% using Redis caching optimization.
- Achieved robust disaster recovery posture with multi-region replication and automated restore procedures.
- Decreased incident resolution time by 35% using centralized CloudWatch + Grafana dashboards.
- Reduced recovery time objective (RTO) by 50% through DR automation and cross-region backups.
- Tools/Environment: ECS, Lambda, EMR, Glue, Step Functions, API Gateway, AWS Batch, Redis, Amazon MSK, IBM MQ, VPC, CodeBuild, CodeDeploy, S3, SNS, SQS, IAM Roles.

Tools/Environment: ECS, Lambda, EMR, Glue, Step Functions, API Gateway, AWS Batch, Redis, Amazon MSK, IBM MQ, VPC, CodeBuild, CodeDeploy, S3,

Project:IBM Sterling Order Management System (OMS)Order orchestration and fulfillment platform for optimizing inventory and delivery processes.

Roles & Responsibilities:
- Built Jenkins CI/CD for OMS builds & releases.
- Automated JAR creation and artifact deployments.
- Managed SSL certs, staging tests, go-live, and rollbacks.
- Configured logging & monitoring for OMS nodes.
- Created SOPs & troubleshooting documentation.
- Designed DR with automated CDT backups.

Impact & Achievements:
- Reduced deployment cycle time by 70% through automated Jenkins CI/CD for OMS builds & releases.
- Reduced production issues by 30% by developing SOPs and structured troubleshooting

documentation.
- Achieved quicker rollbacks and safer releases through automated CDT backups.
- Improved cross-team productivity by providing training to L2/L3 teams and standardizing build workflows.

Tools/Environment: IBM Sterling OMS, Jenkins, Git, Linux, Shell, Apache, SSL, JIRA