Thimmesha S M

Sr.DevOps Engineer

 Bengaluru, KA 560037

 thimmeshasm64@gmail.com

 +91-7899441736

- Overall 6+ years of IT experience with strong expertise in DevOps, CI/CD Pipeline, AWS Cloud, Kubernetes, Terraform, Docker, Ansible and Linux distributions.
- Extensively worked on AWS Cloud platform and its features which includes EC2, VPC, EBS, AMI, CloudTrail, RDS, CloudWatch, SNS, WAF, Elastic binstack, Snapshot, EBS, ELB, EFS, Auto Scaling, S3, Route 53, Lambda, IAM, Code Pipeline, Code Deploy, Code Commit, EKS, ECR
- Worked on Kubernetes administration and managed the self-managed and managed clusters.
- Best experience with development of Kubernetes YAML files and Helm charts.
- Created the Kubernetes manifest file for deployment, services, Ingress, Diemonset, RBAC, SA, ConfigMaps, Secret
- Worked on Kubernetes deployment strategies like blue-green, recreate, rolling update and canary
- Integrated Vertical Pod Autoscaler and Horizontal Pod Autoscaler to ensure high availability and scalability of the application, resulting in improved performance and reduced downtime
- Integrated Grafana and Prometheus for monitoring and alerting purpose.
- Monitored the nodes, pods, VMs, Apps, Clusters, Services using Grafana and Prometheus
- Managed the AWS cloud infra using terraform.
- Experienced terraform modules, terraform null resources, and terraform workspace concepts.
- Secured the terraform state file.
- Managed the Docker services like Images, Containers, Volumes, and Networks
- Experienced with Docker compose to setup the entire application in docker host.
- Responsible for development of docker files for customer and internal team members requirement.
- Responsible for scanning the docker images for security reasons.
- Created the CICD pipeline from the scratch.
- Administrated the Jenkins, Managed the Jenkins agents also integrated docker agents.
- Automation of build process (Build Automation) using the Jenkins, Maven and Git.
- Provided production support and development environments. Ability to communicate requirements effectively to team members and manage applications.
- Experience in using ANT and Maven as a build tool for building deployable artifacts (WAR & EAR) from source code.
- DevOps/Infra Management lead” Architecture level understanding maintaining servers and applications- Making sure 100% up time with quick problem-solving techniques.
- Experience in creating the company's DevOps strategy in a mix environment of Linux (RHEL, Ubuntu) servers along with creating and implementing a cloud strategy based on Amazon Web Services.
- Created the ansible playbook and roles for deployment purpose and also cloud infra management
- Extensive knowledge in troubleshooting builds, deployment, configuration problems in development, QA and production environments.
- Skilled in understanding of Software Configuration Management requirements.
- Strong communication, collaboration & team building skills.
- Proven Track record in meeting the goals and delivering the deliverables within deadlines without slippages.

Skills

Cloud Computing: AWS

Configuration Management Tool: Ansible, Terraform
Continuous Integration/Continuous Deployment (CI/CD) Tools: Jenkins, GitHub Actions

Continuous Integration Tools: Jenkins

Build Tools: MS-Build, Maven, NPM, Gradle

Scripting Languages: Shell, Groovy

SCM Tools: GIT, GitHub, BitBucket

Application Servers: Tomcat Server, Nginx

Containerisation Tools: Docker, Kubernetes

File Transfer Application: Putty, WINSCP

Ticketing Tools: Jira, Redmine

Operating Systems: LINUX, Windows

Monitoring Tools: Grafana, Prometheus, Portainer, ELK stack

Other Devops Tools: SonarQube, Jfrog

Work History

- **Girmiti software pvt ltd** Sr.DevOps Engineer, Bengaluru, India, From june to till

Project #1

Bengaluru, India & Onsite (KSA) | July 2024 – Dcember 2025

Project: NamiPay

Period: June 24 to May 25

Environment: DevOps, Kubernetes, Terraform, Docker, AWS, Ansible, Jenkins, Shell, Grafana, Prometheus

Roles & Responsibilities :

- Created Kubernetes self-managed and EKS clusters from scratch
- Managed and monitored POD's, Nods, and deployments on the Kubernetes cluster
- Developed manifest files defining different Kubernetes components, including config maps, secrets, VPA, HPA, SA, PV, PVC, Ingress and a few other components
- Exposed Kubernetes deployment strategies like Blue-Green, Canary, Rolling Update, and Recreate in Kubernetes
- Created Helm charts and Kubernetes YAML for efficient scaling and deployment of applications. Streamlined the process, enhancing productivity and ensuring smooth operations.
- Exposed the Kubernetes application to the external world using Ingress.
- Created and managed EC2 instances, created AMIs, snapshots, modified the instance types, and created new instances from AMIs.
- Utilize expertise in storage management to effectively perform tasks such as adding, increasing, or decreasing disks using EBS and handling S3 buckets for data storage.

- Implement efficient S3 backups with versioning enabled and securely store objects in Amazon Glacier for extended storage.
- Managed and configuring VPCs, subnets, routing tables, security groups, IGW, NAT, SG, NACL and elastic IPs to ensure smooth network operations.
- Managed the containerized application with AWS EKS and AWS ECR services.
- Configured and managed ELB to evenly distribute traffic and ensure high availability of applications
- Implemented and maintained EFS for scalable and shared file storage across multiple EC2 instances
- Managed Auto Scaling to automatically adjust the number of EC2 instances based on demand, ensuring optimal performance and cost efficiency
- Managed S3 for secure and scalable object storage, enabling efficient data storage and retrieval also managed the S3 versioning
- Created and managed Route 53 for DNS routing and management of domain names and records
- Managed Lambda for serverless computing, allowing the execution of code without the need for infrastructure provisioning
- Implemented IAM to control and manage user access and permissions to AWS resources
- Implemented and managed Code Pipeline, Code Deploy, for continuous integration and delivery of software applications, ensuring efficient and automated deployment processes
- Implemented and managed EKS and ECR to orchestrate, store and manage containerized applications using Kubernetes on AWS
- Implemented and managed Cloud Trail for auditing and monitoring AWS API calls, ensuring compliance and security of AWS resources
- Created RDS for managed database instances, ensuring reliable and scalable database solutions on AWS
- Configured and managed Cloud Watch for monitoring and logging of AWS resources, allowing proactive identification and resolution of issues
- Managed SNS for notification and alerting purposes, ensuring timely communication of system events and status updates
- Identified and solved database-related problems, including connectivity issues, data corruption, and performance glitches with **MySql, RDS**.
- Configure Terraform for the creation of numerous AWS resources, including **VPC**, **EC2**instances, **S3**buckets, **IAM, EKS** and few other.
- Worked on **remote state locking**, **Life cycle rules**, **terraform states**, terraform **modules**, terraform **workspace** and few other concepts
- Engaged in collaborative efforts with teams spanning different functions to architect and implement cloud-based resources through Terraform, enabling seamless expansion and cost optimization.
- Enhanced the deployment process by designing reusable Terraform modules, thus boosting efficiency and ensuring uniformity across diverse projects.
- Carried out comprehensive testing and debugging of Terraform scripts to guarantee the highest level of reliability and stability in infrastructure deployments.
- Successfully migrated the **existing infrastructure** in DEV environment to the **QA** and **UAT** with terraform scripts.
- I successfully integrated **Grafana** and **Prometheus** to improve the monitoring dashboard for our application, clusters, servers, nodes, and pods and few other components.
- By integrating Grafana and Prometheus, I have simplified the **monitoring procedure**, resulting in better visibility and early detection of issues
- I have made use of Grafana and Prometheus to develop detailed monitoring dashboards, offering real-time insights into application performance and resource usage.

- I monitoring capabilities have been boosted through the integration of Grafana and Prometheus, allowing us to efficiently track important metrics and quickly identify bottlenecks.
- Working together with the testing team, I have integrated Grafana and Prometheus seamlessly, resulting in improved monitoring efficiency and easier troubleshooting processes

- **Capgemini** DevOps Consultant, Bengaluru, India From May 2018 to June 2024

Project #1

Project: Capital One
Period: May 18 to May 24
Environment: Kubernetes, Terraform, Docker, AWS, Ansible, Jenkins, Shell, Grafana, Prometheus

Roles & Responsibilities :

- I created the Jenkins pipelines to automate the **CICD** process from **scratch**.
- Automated the so many manual tasks in Jenkins.
- Integrated various tools with Jenkins like **Sonar, JFrog, Maven, git, MSBuild, NPM, X-ray, Ansible, Docker, Kubernetes**, and a few other tools.
- I created the Jenkins pipeline for automated testing scripts to guarantee the excellence of **software releases, leading to enhanced product dependability.**
- Working closely with the development and testing teams, I devised automation solutions to enhance the entire software development cycle.
- I increased team efficiency by producing and updating documentation for Jenkins pipelines and automation methods, making it simpler for new team members to get started.
- Streamlined the process of constructing and deploying applications by utilising Jenkins tasks
- Developed tasks within Jenkins to oversee the setup of devices, deployment settings, system management, backups, and more
- Created Docker Images for various requirements with the **development** and **testing** teams from **Dockerfile**.
- I developed **Docker file** to create docker images for various **microservices** including **front-end, back-end**.
- Implemented a **multistage Docker file** for streamlined deployment, optimizing image size for enhanced performance.
- Efficiently managed Docker services, including **Images**, **Containers**, **Volumes**, and **Networks** using the Portainer tool.
- Created action Ansible playbooks and roles for effective deployment and management of servers
- leading to more efficient procedures and enhanced server management.
- Created an ansible playbook and roles for deploying the **monolithic** and **microservices** into **Kubernetes clusters**, **Docker environments**, and **virtual machines** based on the requirements.
- Worked together with multiple teams to discover and implement automated solutions, decreasing manual work and boosting efficiency with ansible playbooks and roles.
- Carried out routine performance reviews and optimization of servers to improve overall system effectiveness and performance.
- Managed **MySQL** database, handling setup, settings, and regular maintenance
- Identified and resolved problems related to databases, like connection difficulties, data integrity issues, and performance concerns

- Implemented backup strategies and carried out regular backups of MySQL databases, while also creating and executing recovery strategies to reduce data loss and downtime
- Implemented security protocols, including user authentication, access controls, and database permissions to safeguard sensitive information
- Set up the Kubernetes cluster with Jenkins servers, overseeing both the master and worker nodes to maintain optimal cluster performance
- Configured Ingress regulations and routes to effectively direct traffic to backend services and microservices
- Created branches and tags via GIT for version control purposes
- Monitored the progress of builds, GIT check-ins, and access control
- Managed git and GitHub with various operations like Merge, Conflict, Cherry-pick, Rebase, Revert, Reset, Bi-sect, Amend, fork and few other.
- Provided support for product release management and patch set releases
- Responsible for devising plans for building and deploying applications
- I Created Jenkins Continuous Integration and Continuous Deployment (CICD) pipelines with Groovy scripting to automate the software development workflows
- Established monitoring and alert systems to proactively detect and address server issues, reducing downtime and enhancing reliability
- Successfully managed and maintained Linux servers, including CentOS, RedHat, and Ubuntu, ensuring smooth and efficient operation of VMs.
- Developed and implemented shell scripts to automate repetitive tasks, significantly improving productivity and reducing manual workload.
- Troubleshot and resolved Linux server issues promptly, minimizing downtime and ensuring uninterrupted business operations.
- Collaborated with cross-functional teams to plan and execute server upgrades and migrations, ensuring seamless transitions and minimal disruption.
- Implemented security measures and regularly performed system audits to ensure the integrity and confidentiality of data on Linux servers.
- Enforced version control best practices to maintain code integrity and promote smooth teamwork
- Resolved creation and deployment challenges promptly, reducing downtime and ensuring uninterrupted deployment
- Actively engaged in Agile development techniques, participating in sprint planning, daily stand-ups, and retrospectives
- Coordinate with the Development, Database Administration, QA, and IT Operations teams to ensure there are no resource conflicts.
- Worked closely with Project Managers to understand a code/configuration release scope and how to confirm a release was successful.
- Build, manage, and continuously improve the build infrastructure for software development engineering teams including implementation of build scripts, continuous integration infrastructure and deployment.
- Managed deployment automation using Packer-Docker Terraform to automate system operations. Handled work from initial stage of development to create branches, make developers follow standards creating build scripts, automating the build process and deploy process by using Jenkins plugins.

Education