# Dhiraj Kumar Maddali

331 3306073

dhirajmaddali@gmail.com

PROFESSIONAL SUMMARY:

- Around 8 **years** of experience on DevOps essential tools like **Chef, Puppet, Ansible, Docker, Kubernetes, Jenkins, Ant, Maven, and GIT.**
- Work on an **agile** team developing **REST APIs** to support an agent internal operations tool.
- Experienced in all phases of the **software development life - cycle (SDLC)** with a specific focus on the build and release of the quality software.
- Experienced in Waterfall, Agile/Scrum and most recently **Continuous Integration (CI)** and **Continuous Deployment (CD)** practices.
- Experience in administering & automating **Solaris & Linux**. Experience in Automation, Installation, Configuration, Backup, Recovery, Maintenance, Support of **Linux (Ubuntu, CentOS, RHEL).**
- Excellent experience in documenting and automating the build and release process.
- Used Stack Driver and **AWS cloud** monitoring extensively to monitor and debug the cloud-based **AWS EC2 services.**
- Deployed Azure IaaS **virtual machines (VMs) and Cloud services** (PaaS role instances) into secure VNets and subnets.
- Implemented Security in Web Applications using **Azure and deployed Web Applications to Azure.**
- Worked with Client team to build flow diagram for existing Training environment.
- Worked on **Docker** container snapshots, attaching to a running container removing, image managing, directory structures along with **Kubernetes**.
- Hands on experience in using version control tools like **Subversion (SVN), GIT, GitHub and Bitbucket.**
- Extensively worked on Jenkins for continuous integration and for End-to-End automation for all build and deployments.
- Worked with **Terraform** to create stacks in AWS from scratch and deployed using commands.
- Had created Jenkins Pipelines using Groovy script to automate the process to deploy.
- Extensively used build utilities like **Maven, ANT and Gradle for the building of .jar, .war and .ear files.**
- Experience setting up **Azure Migrate** and ASR servers using OVF templates in the On-premise vCenter host.
- Skilled enough to build deployment, build scripts and automated solutions using scripting languages such as **Shell, Python, Ruby, Perl etc.**
- Experience in using **Nexus Repository Manager for Maven builds.**
- In-depth understanding of the principles and best practices of **Software Configuration Management (SCM).**
- Designed and implemented fully automated server build management, monitoring, and deployment by using **Ansible playbooks and Modules.**
- Worked with Ansible Playbooks for virtual and physical instance provisioning, Configuration management, patching, and software deployment.
- Good experience in automating CI and CD pipelines.
- Developed CI/CD system with **Jenkins** on **Kubernetes** container
- Worked on provisioning different environments using Chef, Puppet, and other Configuration management tools.
- Create **Terraform** State files and Cloud Formation templates to build the infrastructure, used chef stack files to deploy services.
- Server provisioning, automation (Puppet/Chef/Ruby), maintenance and performance tuning Hands-on experience in installing and administrating CI tools like Hudson/Jenkins, Team city, Bamboo and ALM tools like Anthill Pro and TFS.
- Implemented a production ready, load balanced, highly available, and fault tolerant **Kubernetes** infrastructure.
- Used Puppet and Chef for Configuration Management tool.
- Extensively involved in maintaining large amounts of structured, semi-structured, and unstructured data across multiple **data centres and the cloud using Cassandra.**

- Hands on experience in **GCP services like EC2, S3, ELB, RDS, SQS, EBS, VPC, EBS, AMI, SNS, RDS, EBS, Cloud Watch, Cloud Trail, Cloud Formation GCP Config, Autoscaling, Cloud Front, IAM, R53.**
- Integrated Jenkins with various **DevOps tools such as Nexus.**
- Have good experience in implementing **JIRA and HP Quality** centre as bug tracking and ticketing tool.
- Experience on search and analytic tools like **Splunk** and developing **Splunk queries, dashboards** targeted at understanding application performance and capacity.

TECHNICAL SKILLS:

| | |
|---|---|
| Cloud services | AWS, AZURE, GCP, Open Stack |
| Virtualization platforms | Virtual box, VMware, Docker |
| Infrastructure as code Tool | Terraform, Cloud Formation |
| Configuration tools | Chef, puppet, Ansible |
| Version control tools | Git, SVN, Beanstalk |
| Operating systems | Linux (RHEL, CentOS, Ubuntu), windows, Solaris |
| CI, build and test tools | Jenkins, Ant, Maven, Selenium, Junit, Bamboo, TeamCity. |
| Containerization tools | Kubernetes, Docker. |
| Monitoring Tools | Nagios, Prometheus, Splunk, Dynatrace |
| Databases | Oracle, Mogo DB, SQL server |
| Scripting languages | Python, ruby, Bash, JSON, Yaml, JavaScript. |
| Bug Tracking Tools | Jira, Scrum manager, Bugzilla |
| Data Visualization tool | Tableau |
| SDLC, Internet Protocols | Agile (Scrum), TCP/IP, DNS, SMTP, HTTP, waterfall |

Client: Apple    Remote                                                                April 2020 – Feb 2023
DevOps Engineer
Responsibilities:
- Designed and developed Backup Solution for an Enterprise Apps on **AWS** to serve RDS automated daily db backups and **EC2** with **EBS** snapshot and Amazon Glacier for individual file level backups.
- Created AWS Launch configurations based on customized AMI and use this launch configuration to configure auto scaling groups and Implemented AWS solutions using **EC2**, **S3**, **RDS**, **Dynamo DB**, Route53, **EBS**, Elastic Load Balancer, **Auto scaling** groups.
- Maintained user accounts (**IAM**), **RDS**, **Route53,VPC**, **RDB, DynamoDB,SQS** and **SNS** services in **AWS**.
- Implemented **Docker container** creation process for each **GitHub** branch gets started on **Jenkins** as Continuous Integration (**CI**) server.
- Experience on writing **Chef Cookbooks** and **recipes** to automate the deployment process and to integrating Chef Cookbooks into **Jenkins** jobs for a continuous delivery framework.
- Provisioned the highly available **EC2** Instances using **Terraform** and cloud formation
- Worked on **Jenkins** for continuous integration and continuous delivery (**CI/CD**).
- Implemented the **Docker Swarm** with **AWS CloudFormation Stack**.
- Evaluated **Kubernetes** for **Docker container orchestration.**
- Implemented Continuous Delivery framework using **Jenkins**, **CHEF**, **Maven** in **Linux** environment. Created virtual environments via **vagrant** with chef client provision.
- Hands-on experience networking, storage and database AWS services.
- Secured **Terraform** state file remotely in S3 buckets by encrypting and versioning. Configured Vault to store sensitive data. Defined AWS infrastructure as code by making use of various **Terraform AWS** modules to create VPC, subnets, EC2 instances, and RDS.
- Worked on **DynamoDB** for Database, Lambda functions for Querying Database, API Gateway to publish and deploy application, **S3** for file Storage and **AWS** Cognito for Authentication Services.
- Used **Terraform** to for building, changing, and versioning infrastructure safely and efficiently.
- Deploy and run applications using **Docker** containers.
- console and used **lambda** to run applications in AWS console.

- Regular Build jobs are initiated using the Continuous Integration tool with **Jenkins**.
- **AWS EC2** and **Cloud watch** services. **CI/CD** pipeline management through **Jenkins**. Automation of manual tasks using Shell **scripting**.
- Application Deployments & Environment configuration using **Chef, Ansible**.
- **Zabbix** monitoring with Grafana visualisation infrastructure in **Kubernetes** cluster.
- Implemented Atlassian Stash along with **GIT** to host GIT central repositories for source code across products, facilitate code reviews and login audits for Security Compliance.


**Environment:** AWS EC2, S3, VPC, SQL Server, Cloud Watch, Cloud formation, Git, ELB, Chef, Docker, Red Hat Linux 5/6, Bash Scripting, Jenkins, Terraform, chef, Docker.

**Epicor, India**                                                                                     **March 2018 – April 2020**
**DevOps AWS Engineer**


- Developed and implemented Software Release Management strategies for various applications according to the Agile Process.
- Configuring of monitoring and alerting tools according to the requirement like **AWS** CloudWatch, CloudTrail, Dynatrace, Nagios, Splunk Enterprise, SNMP monitoring for the VPN connections.
- Written CloudFormation Templates (CFT) in JSON and YAML format to build the AWS services with the paradigm of Infrastructure as a Code.
- Involved in AWS EC2, VPC, S3, SQS, SNS based automation through Terraform, Python, Bash Scripts. Adopted new features as they were released by Amazon, including ELB & EBS.
- DevOps role converting existing AWS infrastructure to Server-less architecture (AWS Lambda, Kinesis) deployed via **Terraform** template.
- Create automation and deployment templates for relational and NoSQL databases including MSSQL, MySQL, Cassandra, and MongoDB in AWS. Worked in setting up the CI/CD pipeline utilizing **Jenkins**, Chef, Git, **Bitbucket**, and **Terraform**.
- Worked on implementing **Terraform** modules to deploy application across multiple Cloud Provides.
- Worked on deploying web applications using Circle CI and controlled the deployment process through Circle CI to run Terraform configuration in automation. Worked on pipelines to drive all Microservices build out to the **Docker** registry using Circle CI and deployed it to Kubernetes.
- Worked on developing dev/test/prod environment on Docker container using **Kubernetes** and deployed the containerized applications to Kubernetes.
- Worked on reducing the downtime of APP team by migrating production workload from the old cluster to Kubernetes HA cluster.
- Used **Jenkins** pipelines to drive all micro services builds out to the Docker registry and then deployed to **Kubernetes**, Created Pods and managed using Kubernetes
- Worked on monitored, logging, and altering Kubernetes cluster by deploying Prometheus and Grafana using Helm chart.
- Worked on supporting Enterprise Data Warehouse consisting of Virtual Private cloud, Private and Public Subnets, Internet Gateway, Route Tables, Security groups and more services by building and configuring a virtual data center in AWS.
- Worked with migration of **AWS** cloud to **Azure cloud** and migrated various application to cloud.
- Worked with creation of S3 backups using versioning enable and moved objects to Amazon Glacier for archiving purpose and implemented auto-scaling for the instances to design cost-effective, fault-tolerant, and highly reliable systems.
- Worked with S3 as the data storage, performed versioning, created lifecycle policies, and maintained the user account's services in AWS to provide highly durable and available data.
- Worked on setting up the CI/CD automation environment for the Application team if necessary and helped them through the process of build and release automation.
- Solid understanding of **Linux** Shell and Perl Scripting, **UNIX,** Windows, build and Release Engineering principles.
- Coordinate/assist developers with establishing and applying appropriate branching, labelling/naming conventions using Subversion (SVN) and Git source control.

- Worked on Client-side hooks such as **GIT** committing and merging, server-side hooks run on network operations such as receiving pushed GIT commits.
- Create develop and test environments of different applications by provisioning Kubernetes clusters on AWS using **Docker**, Ansible, and Terraform. Actively involved in Architecting the puppet infrastructure to manage servers in different environments.
- Integration of Maven/Nexus, Jenkins, **Rundeck,** Urban Code Deploy with Patterns/Release, Git, Confluence, JIRA, and Cloud Foundry.
- Using Jenkins AWS Code Deploy plugin to deploy to AWS.
- Responsible for code build, release, and configuration on Amazon EC2.
- Optimized volumes and EC2 instances. Created multi-AZ VPC instances. Used IAM to create new accounts, roles, and groups.
- Using **Chef** and **AWS** allowed me to reduce costs for the department and eliminate unwarranted resources. Automated provisioning of cloud infrastructure with Chef. Configured S3 versioning and lifecycle policies to and backup files and archive files in Glacier.
- Configured Elastic Load Balancers (ELB) with EC2 Auto scaling groups.

**Environment:** AWS EC2, S3, VPC, SQL Server, Azure Cloud , Git, ELB, Chef, Docker, Red Hat Linux 5/6, Bash Scripting, Jenkins, Terraform, chef, Docker, UNIX.

**Client: Gainsight India**                                                               May 2015 - Feb 2018
**Role: Linux Engineer**
**Responsibilities**:

- Deployed releases to the production environment and assisted the project teams in troubleshooting the issues.
- Worked on setting up the build and deployment infrastructure using **Bamboo** and **Maven**.
- Administered Bamboo servers which include install, upgrade, backup, adding users, creating plans, installing the local/remote agent, adding capabilities, performance tuning, troubleshooting issues and maintenance.
- Build and deployment of scripts using **ANT**, Shell to set up development and QA environments.
- Worked with Version Control tools like **GIT**. Created branches in GIT to implement parallel development process.
- Created deployment request tickets in **Jira** for deploying the code to Production.
- Implemented a Continuous Delivery framework using Maven & Nexus in **Linux** environment.
- Collaborated with development and support team to setup a continuous delivery environment with the use of Docker.
- Created **Docker** images using a Docker file, worked on Docker container snapshots, removing images and managing Docker volumes. Expertise in setting up Docker Host, Docker Registry, RHEL/ Centos Atomic Host, Docker images (Centos, RHEL and Ubuntu instances).
- Worked on **Maven** creating artifacts through the source code and internal deployment in Nexus repository.
- Installed **Jenkins** on **Linux** machines and created master and slave configurations to implement multiple parallels builds.
- Documented the deployment process (Migration Doc) of code to production on an Excel Sheet.

**Environment:** GIT, CI/CD, Maven, ANT, Bamboo, SonarQube, Nexus, Oracle, Linux, Python/Shell script, Tomcat, Jira.