**S DHWANI**

New Jersey – (862) 223 8143

developer.dhwani@gmail.com

**PROFESSIONAL SUMMARY:**

* Over 7 years of IT experience as a Cloud Ops Engineer, DevOps Engineer, and System Administrator.
* Experienced in managing servers, building new servers.
* Quality experience in Amazon Web Services such as **EC2, VPC, S3, IAM, EBS, RDS, ELB, VPC, Route53, Dynamo DB, Auto Scaling, Cloud Trail, Cloud Watch, Cloud Formation, Elastic search, Elastic File system (EFS), Elastic Beanstalk, Code Commit, Code Pipeline, Code Deploy, Code builds, AWS SNS.**
* Having involvement in automating infrastructure IAC provisioning using **Terraform** on AWS.
* Experience with designing, building, and operating solutions using Virtualization by using private hybrid/public cloud technologies.
* Hands-on proficiency with Migrating a production infrastructure into an Amazon Web Services utilizing **AWS Cloud formation**.
* Proficiency in configuration and automation using **Chef, Jenkins, Ansible**, and **Docker**.
* Knowledge in configuring **Docker Containers** for Branching and deployed using **Elastic Beanstalk**.
* Ability to build utilities like **Maven, Ant** for the building of jar, war, and ear files.
* Expertise AWS **Lambada** function and **API Gateway**, to submit data via API Gateway that is accessible via Lambda function.
* Supervised configuration of Web App and Deploy to AWS cloud server through Chef.
* Developed AWS **Cloud Formation** templates and set up Autoscaling for EC2 instances.
* Designed **Kubernetes cluster** with objects like Pods, Deployments, Services, and Config Maps and created reproducible builds of the Kubernetes applications, managed Kubernetes manifest files, and Helm packages and implemented Kubernetes to deploy scale.
* Worked with **Kubernetes** for the runtime environment of the CI/CD system to build, test deploy.
* Operated on Installing, Configuring, and administering the **Jenkins CI** tool.
* Experienced in creating jobs and **pipeline jobs** in Jenkins.
* Familiarities with **Linux Administration** activities like troubleshooting of regular issues, configuration issues, applying patches, package management, and file system issues.
* Skilled in utilizing **Continuous Integration and Continuous Deployment** through Agile Methodology to deliver weekly software deployments through **Bamboo.**
* Efficient grasp in Virtualization Technologies **vSphere, VMware, Virtual Box.**
* Mastery in implementing Spark using **Scala**and **Spark SQL** for high-speed testing and processing of data responsible to manage data from different sources.
* Expanded an **Azure**-**based** high performance compute environment to help the massive computational specification of client congressional redistricting Azure application.
* Experience in Configuring and maintaining Networking services and protocols such as **LDAP, DNS, NFS, DHCP, and SSH.**
* Transformed existing **Terraform** modules that had version dispute to utilize **Cloud formation** templates during deployments, worked with Terraform to create stacks in AWS, and updated the Terraform scripts based on the requirement on a regular basis.

**TECHNICAL SKILLS:**

|  |  |
| --- | --- |
| **Cloud** | AWS, Microsoft Azure |
| **Containerization Tools** | Docker, Docker Swarm, Kubernetes |
| **Configuration Management Tools** | Chef, puppet, Ansible |
| **CI/CD Tools** | Jenkins, Circle CI, Bamboo |
| **Build Tools** | Maven, Ant, Gradle |
| **Version Control Tools** | Git, GitHub, Bitbucket, SVN |
| **Monitoring Tools** | Splunk, Cloud Watch, New Relic, Datadog, Nagios |
| **Languages & Scripting** | Python, Ruby, Bash, Shell scripting, Java, JavaScript, HTML, CSS |
| **Web Servers** | Apache Tomcat, Nginx, JBoss |
| **Artifact Repository** | Nexus |
| **Ticketing Tools** | JIRA, Rally |
| **Database** | MySQL, PostgreSQL, Aurora DB, MongoDB, Dynamo DB |
| **Networking Protocols** | TCP/IP, VPC, subnets, VPN |
| **Operating Systems** | Windows, Unix/Linux, CentOS, Ubuntu |

**EDUCATION:**

**Master of Science in Computer Science** August 2018 – May 2020

Texas A & M University – Kingsville  **GPA: 3.57**

**Bachelor of Engineering in Instrumentation & Control**  Jan 2017

Gujarat Technological University, Gujarat, India  **CGPA: 6.9**

**EXPERIENCE:**

**Senior AWS Engineer**

**PayPal – Timonium, MD Feb 2020 – Present**

**Responsibilities:**

* Analyzed, Designed, and Setup **IAM** Users, Roles, Policies as per the Organization Security Standard.
* Accomplished in **AWS** services like **VPC, EC2, S3, ELB, Autoscaling Groups (ASG), EBS, RDS, IAM, CloudFormation, Route 53, CloudWatch, CloudFront, CloudTrail.**
* Built CloudFormation templates for **SNS, SQS, Elasticsearch, DynamoDB, Lambda, EC2, VPC, RDS, S3, IAM, CloudWatch** services implementation, and integrated with Service Catlog.
* Professional in creating multiple **VPC**’s and public, private subnets as per demand and dispersed them as groups into various availability zones of the VPC.
* Featured **NAT** gateways and instances to permit communication from the private instances to the internet through bastion hosts.
* Configure the repository to issue an **SNS** notification upon changes. Support the pipeline to the Amazon SNS topic.
* Used **SonarQube** for uninterrupted inspection of code quality and to perform automatic reviews of code to discover bugs. Directing AWS infrastructure and automation with **CLI** and **API**.
* Building/Maintaining **Docker** container clusters managed by **Kubernetes**.
* Collaborated in the automation of AWS infrastructure via **Terraform**, including provisioning AWS environments using **Ansible Playbooks**.
* Hands-on with **Ansible Playbooks** to setup and configure Continuous Delivery Pipeline and Tomcat servers. Deployed Micro Services, with provisioning AWS environments using Ansible Playbooks.
* Set up an Amazon **CloudWatch** Events rule triggered by a Code Commit Repository State Change event for the master branch and add an Amazon SNS topic as a target.
* Formation of **RDS database** instances Postgres SQL in AWS cluster making use of EC2 and VPC and launched via Cloud Formation template.
* Used **AWS Beanstalk** for deploying and scaling web applications and services evolved with Python.
* Executed and preserved the monitoring and alerting of corporate servers/storage using **AWS Cloud Watch**.
* Integrated the **AWS S3** logs with Athena service, exported the data in csv format by leveraging the Lambda with python code.
* Build, Deploy, and Monitor **Microservices** in Amazon Cloud Services AWS.
* Worked in implementing the AWS organizations in the current environment, effectively leveraged the features in managing the centralized **billing** and **new account setups**.
* Deploying **Docker** container on AWS ECS.
* Hands-on experience on Container Orchestration like AWS ECS, **Kubernetes**.
* AWS ECS and EKS for docker image storage and deployment.

**Environment/Tools: EC2, Elastic Load Balancing, Cloud Front, Cloud Formation, Lambda, Elastic Cache, Cloud Watch, Route 53, Lambda, and Dynamo DB, Nagios, Terraform, Kubernetes, Docker, Jira**, **Ansible, Maven, Groovy Bash scripts.**

**DevOps AWS Engineer**

**Blink Health – New York Aug 2019 – Till Date**

**Responsibilities:**

* Involved in AWS Environment, Worked on Various Services like **EC2, IAM, VPC, S3, EBS, RDS, KMS, ELB, Route53, Auto Scaling, SQS.**
* Worked with Identify and Access Management **(IAM)** to allocate roles and to create and manage AWS users and groups, and user permissions to **AWS** resources.
* Worked on Setting up private networks and sub-networks using **Virtual Private Cloud** and created **Security Groups** to associate with the networks.
* Used **Terraform** Modules for Creating Users in **AWS.**
* Worked on Enabling **S3** Intelligence Tiering for Terraform and non- terraform managed buckets in Dev, Ops, Prod environments in **AWS.**
* Worked on Tooling, Building and improving **Infrastructure** to runs our Applications.
* Used Docker by writing **Docker files**and set up the automated build on Docker hub and configured Kubernetes.
* Created **Docker** Registry and handled multiple images primarily for domain configurations.
* Used **Kubernetes** to manage the containerized applications, deployments, secrets and services.
* Created **Ansible** playbook files for windows server builds and used yaml packages for installing, configuring push to make changes in time.
* Worked on end to end automation of infrastructure and continuous delivery of the applications by integrating **Jenkins.**
* Used **GitHub** repository to keep track of all changes in source code.
* Fabricated **Python** scripts to automateS3 data uploadand control instance operations with **AWS API.**
* Worked on **Shell scripting** for Automating the build and release process.
* Configured **Datadog** dashboards to monitor real time granular metrics of all the **AWS.**
* Worked on Automation by creating Users, teams, Services, Schedules and escalation policies in **PagerDuty** and integrated with slack for alerts.
* Created alerts, Schedule searches using **Splunk to** monitor application and server logs.
* Created AWS RDS **MySQL DB** cluster and connected to the database through an Instance using the Amazon RDS Console.
* Worked on **Packer** file creation and added docker daemon json file for **Ubuntu** 18.04.
* Worked on managing user accounts, security, disk space and process monitoring in **Linux.**
* **Troubleshoot** and debugged environment and resolved **Infrastructure** problems found in the production and non-production environments.
* Maintained **JIRA** for tracking and improving project defects and tasks.
* Included in **Sprints** and planned releases with the team.

**Environment: AWS (EC2, IAM, VPC, S3, EBS, RDS, KMS, ELB, Route53, Auto Scaling, SQS), Docker, Kubernetes, Ansible, Jenkins, Git, Python, Shell, Datadog, Splunk, PagerDuty, MySQL DB, PostgreSQL, Packer, Linux, Jira.**

**DevOps Engineer**

**Walmart – Reston, VA Jun 2016 – Apr 2017**

**Responsibilities:**

* Involved in the Study of the business logic and synchronized with the client to gather the requirements.
* Registered the **SCM** process and policies of the company and suggested some process changes.
* Configured internode conveyance between **Cassandra nodes** and client using SSL encryption.
* Installed and worked with Cassandra cluster and **CQL** on the cluster.
* Experience in containerized based deployments using **Docker**.
* Hands on experience in automating application deployment in the cloud using Docker technology with the help of **Amazon EC2.**
* Operated on developing Chef Recipes to perform deployments onto application servers like **Tomcat** and **Nginx**.
* Noted Chef Cookbooks, recipes and pushed them onto Chef Server for configuring EC2 Instances.
* Assembling and configuring of EC2 instance on AWS cloud platform, Configuration of Elastic Load Balancer (**ELB**) for traffic control for the EC2 and S3.
* Generated network architecture on **AWS VPS, subnets, Internet Gateway, Route Table** and **NAT** Setup.
* Utilized AWS lambda to run servers without managing them and to trigger to run code by S3 and SNS.
* Implemented, configured, and administrated Layer 7 **API Gateway** and Portal.
* Used groovy scripts to develop scripted continuous delivery pipelines in Jenkins and have integrated them with **Maven**.
* Defined dependencies and plugins in Maven pom.xml for various activities and integrated Maven with **GIT** to manage and deploy project related tags.
* Integrate existing performance tools with Jenkins for (**CI/CD**) as part of DevOps Chef and Packer to build **Jenkins** master AMI. This includes Groovy to configure plugins
* Performed system monitoring using **Nagios** and **Cloud Watch** in AWS Cloud.
* Involved in setting up **JIRA** as defect tracking system and configure various workflows, customizations and plugins for JIRA bug/ issue tracker.
* Produced **Bash, Shell & Python scripts** for various Systems Administration tasks to automate frequent processes.
* Configured Jenkins to implement nightly builds on daily basis and generated changes happened from last 24 hours.

**Environment:** **Maven, Docker, Jenkins, GIT, Chef, cloud watch, AWS (EC2, VPC, ELB, S3, RDS, Cloud Trail and Route 53), Python, Cassandra, Shell Scripting, Ruby, PUTTY.**

**DevOps AWS Engineer**

**GE Digital – San Ramon, CA Feb 2019 – Aug 2019**

  **Responsibilities:**

* Involved in **AWS** provisioning and managing services like **EC2, Elastic Load Balancers, VPC, Cloud Watch, Cloud Front, Elastic Container Service, S3, Dynamo DB, Cloud Trail, IAM, SQS.**
* Worked with the **GitHub repository for** storing Terraform files, and maintaining versioning.
* Responded to **Terraform scripts** to automate builds and deployments. Built Terraform script to register image to ECR.Wrote terraform scripts to create an **ECR Repository** and ECR policies and roles in AWS Environment.
* Made **Terraform modules** for AWS automation within the Production, Staging, and Development environments.
* Maintained user account **AWS IAM roles, Groups, Policies,** assuming roles cross-account, as well as maintaining the trust relationships between accounts.
* Wrote **shell scripts** to push the **Docker Images** into AWS ECR.
* Installed **Docker Registry** for local upload and download of Docker images and even from Docker hub.
* Facilitated with Development and Support teams to set up a **Continuous Delivery** environment with the use of **Docker,** continuous build, and delivery tools.
* Created two separate **Jenkins pipelines** for running Terraform scripts and the other to Push the Docker Images and Update the latest Image.
* Deployed Dockers in **AWS FARGATE** by using ECS Orchestrator.
* Configured backend to all Terraforms environments and directories using **S3** and **Dynamo DB.**
* Used **Parameter Store** which consists of information where **Rabbit MQ** to read IAM Role, Username, and Passwords.
* Created **Chef** Cookbooks to automate system operations. Made monitors, alarms, and **notifications** for **EC2 hosts** using Cloud Watch.
* Worked closely with the **Splunk** monitoring team to set up various **logging metrics** including **Cloud Watch, Kinesis Streams** all logging to **S3 destinations** using Terraform scripts.
* Implemented and maintained the **monitoring and alerting** of production and corporate servers or storage using the **AWS cloud watch**.
* Developed **UNIX** and **Shell Scripts** for the purpose of manual deployment of the code to the different environments.
* Defined and Implemented Software Configuration Management Standards based on **Agile/Scrum methodologies**, in line with organization.
* Handled daily urgent tasks using **Rally Dashboard.**

**Environment: AWS (EC2, S3, ECR, VPC, IAM, ELB, Fargate, Cloud Watch, Kinesis, Cloud Trail, Dynamo DB, Rabbit MQ), Git, Jenkins, Chef, Docker, Docker Registry, Terraform, Shell, Splunk, Unix, Rally.**

**DevOps AWS Engineer**

**WHOLE FOODS – Austin, TX Mar 2015 – Oct 2016**

**Responsibilities:**

* Designed, configured, and managed public, private cloud infrastructures utilizing AWS including **EC2, Auto-Scaling** in launching EC2 instances, **Elastic Load Balancer, Elastic Beanstalk, S3, Glacier, Cloud Front, RDS, VPC, IAM, SNS, Direct Connect, Route53, Cloud Watch, Cloud Formation.**
* Used **AWS CLI** command-line client and management console to interact with AWS resources and APIs.
* Implemented and maintained Monitoring and alerting of production and configuration such as EC2 and Storage as S3 buckets using AWS **Cloud watch**.
* Administered Amazon **Redshift** clusters such as launching the cluster and identifying the node type and used **AWS Beanstalk** for deploying and scaling web applications and services developed.
* Implemented multi-tier application provisioning in OpenStack cloud, integrated it with **Chef/Puppet**.
* Used Open-stack to download and install an OpenStack distribution within the internal network to see distributions.
* Designed **Terraform templates** to create custom-sized **VPC, subnets, NAT** to ensure successful deployment of Web applications and database templates and migration from traditional to cloud environment.
* Created **Jenkins** job to build the artifacts using maven and deploy the Terraform templates to create the stack.
* Worked on **Docker** open-source b the deployment of applications inside software containers.
* Implemented a new Docker container creation process for each **GitHub** branch to get started on **Jenkins** as a **CI server**.
* Worked on **Kubernetes** to deploy scale, load balance, and manage Docker containers with multiple namespaced versions.
* Implemented Chef best practices and introduced a test kitchen to facilitate a more natural cookbook development workflow. Setting up multiple cookbooks, recipes, templates, and attributes on workstations to be deployed to various **Chef Nodes**.
* Worked with **Ansible playbooks** for virtual and physical instance provisioning, configuration management, patching, and software deployment.
* Used Ansible to manage web applications, configuration files, used mount points, packages, and Worked on playbooks for Ansible in **YAML**.
* Composed Jenkins for any automation builds which are integrated with GIT as part of infrastructure automation under continuous integration (CI).
* Extensively apprehended on Jenkins, CI/CD pipeline jobs for end-to-end automation to build, test and deliver artifacts and troubleshoot the build issue during the Jenkins build process.
* Implemented a GIT mirror for **SVN** repository, which enables users to use both GIT and SVN and used **Maven** as a build tool on Java projects for the development of build artifacts on the source code.
* Worked with **New Relic** monitoring, and alerting services for servers, switches, applications and services.
* Used **JIRA** Confluence for handling documentation, running automated tests, JIRA bug tracking management.
* Migrated on-premise MySQL environments to **MySQL** on Amazon **EC2** and Amazon **Aurora MySQL** with ongoing Database replication.
* Installed, configured, and administrated all Linux servers, includes the design and selection of relevant hardware to Support the installation/upgrades of **Red Hat** (6/7), **CentOS** 6/7, operating systems.

**Environment: AWS, OpenStack, Terraform, Chef, Ansible, Docker, Kubernetes, Jenkins, Maven, Git, New Relic, Python, Apache Tomcat, JIRA, Java, Database, Linux.**