

**Anish M**

Cloud Automation Engineer /AWS / DevOpsPH: **+1 805 815 7659| anishmullapudi18@gmail.com**

Linked IN URL: **https://www.linkedin.com/in/anish-m-0b489a205/**

**Professional Summary:**

A Certified AWS with **12+ years** of experience in IT industry alongwith primarily working on **DevOps**, **AWS** Cloud infrastructure, deep experience on **Cloud Implementation and Migration of Cloud services, Cloud security, Infrastructure as Code, Containerization/Orchestration, Configuration Management, CI/CD etc.**

* Extensive experience in strategy and practical implementation of AWS Cloud services which include EC2, EBS, S3, Glacier, Storage Gateway, VPC, Route 53, RDS, ELB, EMR, ECS, Cloud Front, Elastic Cache, Cloud Watch, Cloud Trail, RedShift, Lambda, AWS Elastic Beanstalk, SNS, Dynamo DB. And, also worked in migrating a production infrastructure into an Amazon Web Services utilizing **AWS Cloud formation**, Code Deploy, Code Commit, Chef, EBS.
* Developed Helm charts for the containerized version of the application running on-premise, deployed it to **EKS Kubernetes**.
* Developed **Terraform** code for a **Kubernetes** infrastructure build. Managed it using **Kops** for **Kubernetes.** Debugged issues and integrated Kubernetes with **ECR** and Amazon Route 53.
* Deployed OpenVPN into EKS with self-service certification authority using corporate central authentication.
* Experience in working with **Terraform** and **AWS Cloud Formation** templates for automating Infrastructure provisioning as Code – for example, VPC’s, ELB’s, Security groups, SQS queues, S3 buckets etc.
* Expertise in App Containerization technology **Docker**, creating Docker images, Containers, Docker Registry to store images, cloud-based registry Docker Hub to manage containers.
* Experience in Kubernetes to deploy scale, load balance and manage Docker containers with multiple name spaces versions and good understanding of Open Shift Platform in managing Docker Containers and Kubernetes Clusters.
* Hands-on experience in implementing **Ansible** as Configuration management tool, to automate repetitive tasks, quickly deploys critical applications, and proactively manage change by writing Python code by using Ansible Python API to automate Cloud Deployment Process.
* Created monitors, alarms & notifications for EC2 hosts using Cloud Watch. And Implemented Instances and Prometheus on Kubernetes.
* Experience with the Apache Kafka platform and other messaging buses and Knowledge in Security framework integration. Configured Kafka Cluster and enabled security and configured monitoring tools on messaging platforms. Remote login to Virtual Machines to troubleshoot, monitor and deploy applications.

**Technical Skill Set:**

|  |  |
| --- | --- |
| **Cloud Services** | AWS (EC2, S3, ELB, EBS, IAM, VPC, RDS, SNS, SQS, Glacier, Route53, Cloud Watch, Cloud Formation, Cloud Front, Auto Scaling, Elastic Cache, EMR, Red Shift). |
| **Configuration Management Tools** | Ansible, Chef, Puppet. |
| **CI/CD Tools** | Jenkins, Gitlab CI/CD, Maven. |
| **Containerization Tools** | Kubernetes, Docker, EKS, ECS. |
| **Application/Web Server** | Tomcat, Apache, NGINX  |
| **Testing and ticketing tools** | SonarQube, Selenium, JIRA. |
| **Monitoring and Logging Tools** | Nagios, Splunk, Cloud Watch, Data dog, Prometheus, Grafana. |
| **Version Control Tools** | GIT, GitHub, GitLab, Subversion, Bit bucket, Team Foundation Server (TFS). |
| **Scripting and****Programming Languages** | Python, Java, Bash/Shell, PowerShell. |
| **Operating System** | UNIX, Linux, Windows, Solaris, MacOS. |
| **Database Systems** | AWS RDS, Oracle, MySQL, MariaDB, NoSQL (Mongo DB, DynamoDB, Cassandra). |
| **Networking/Protocol** | DNS, DHCP, Cisco Routers/Switches, NIS, NFS, WAN, SMTP, LAN, FTP/TFTP, TCP/IP. |

**Education**

* Bachelors in Electronics and Communication Engineering, Satyabhama University, Chennai.

**Certifications**

* AWS Certified Solutions Architect Professional
* AWS Certified SysOps Administrator Associate
* AWS Certified Developer Associate
* AWS Certified Cloud Practitioner
* Six Sigma Green Belt

**PROFESSIONAL EXPERIENCE**

**Freddie Mac, McLean, VA**

**June 2021 – Present**

**AWS Cloud Automation Engineer /DevOps**

**Responsibilities:**

* **Provisioning servers/instances** using infrastructure as code such as **Terraform and Cloud Formation**. Code is stored in a private repository and constantly updated.
* Created a **POC** on testing the **CloudFormation templates** using **Taskcat**.
* Created a brand new **Python framework** to test the **CFTs** before deployment to Production.
* **Automation** of **CloudFormation** Testing Process. **Regression testing/Integration Testing.**
* Updated **Python 3.9 libraries** to the **GoldField Common layer.**
* Created the Account Provisioning Pipeline to automatically create the accounts.
* Designed and implemented Account Metadata solution using **AWS Serverless** stack with **Python**.
* Developed and Deployed the resources using **CloudFormation** templates.
* Implemented **Lambda functions** which when invoked creates a list of all metadata for accounts.
* Created **Multi Region KMS key** for encryption of all resource and resource data.
* Created **S3 Buckets and Bucket Policies** which allows Lambda to publish JSON objects.
* Created KMS to encrypt the data in S3 buckets.
* Created **CloudWatch** Events which invokes the Lambdas on Create, Update or Delete of an Account or **OU Tag.**
* Created **IAM Roles and Policies** with the **principle of least privilege** approach.
* Created SNS Topic which allows the S3 Bucket to Publish, and anybody in the **AWS Organization** to subscribe.
* Created and managed **SSM Parameter Store** for Organization metadata.
* Created less privileged IAM roles for EC2 and MDTC teams as replacement of their highly privileged Landing Zone Admin Role.
* Configured and maintained Jenkins to implement the CI process and integrated the tool with **Maven** to schedule the builds. Took the sole responsibility to maintain the CI server.
* Designed, developed, and managed infrastructure using Terraform to provision and manage resources across AWS environments.
* Modular Codebase - Created and maintained reusable Terraform modules to standardize and streamline infrastructure provisioning across multiple projects.
* State Management-Managed Terraform state files effectively, implementing remote state storage in S3 with state locking via DynamoDB to ensure safe and consistent state management.
* Gathered the teams LZA data off of Splunk and using this data tailored and designed the specific IAM roles to EC2 and MDTC teams to cater their daily activities in AWS.
* Deployed and tested the IAM roles in lower environments and deployed to **Production**.
* **Mitigated** and **analyzed** the risk of **broken lambda functions** in AWS Accounts by doing the root cause analysis and creating a **remediation plan**.

**Environment: Powershell, Shell scripting, BitBucket, PyCharm, VSCode, GITHUB, Jira, Ansible, Confluence, Jenkins, AWS (EC2, DynamoDB, VPC, ELB, S3, RDS, Cloud Trail and Route 53, EBS, IAM, RDS, ELB, Cloud watch, CloudFormation, LambdaFunctions, Auto Scaling), Terraform, Kubernetes.**

**CiscoSystems, SanJose CA**

**Dec 2019 – May 2021**

**DevOps/AWS Engineer**

**Responsibilities:**

* Developed highly available REST applications with DynamoDB.
* Designed and managed highly available and scalable Dynamo DB Architecture, with optimum hash keys and multiple global indexes.
* Designed an automated web application which will get data from third party API’s and updated the application on a timely basis.
* Implemented cluster services using Docker and Kubernetes to manage local deployments in **Kubernetes** by building a **self-hosted** Kubernetes cluster using **Terraform** and **Ansible** and deploying application containers.
* Used Kubernetes to manage containerized applications using its nodes, Config Maps, selector, Services & deployed application containers as Pods.
* Configured a private Docker Registry, authored Docker files to run apps in containerized environments and used Kubernetes to deploy scale, load balance and manage Docker containers with multiple namespace ids.
* Developed Ansible playbooks for managing the **application/OS configuration** files in GitHub, integrating with Jenkins, and Verifying with Jenkins plugins, deploying the application into the Linux environment.
* Worked with an agile Development team to deliver an end to end continuous integration/continuous delivery product in an open source environment using **Jenkins**. Created and executed test cases and test scenarios for various functionalities using Selenium
* Responsible for **CI/CD process** implementation using Jenkins along with Python and Shell scripts to automate routine jobs.
* Configured and maintained Jenkins to implement the CI process and integrated the tool with **Maven** to schedule the builds. Took the sole responsibility to maintain the CI server.
* Designed, developed, and managed infrastructure using Terraform to provision and manage resources across AWS environments.
* Modular Codebase - Created and maintained reusable Terraform modules to standardize and streamline infrastructure provisioning across multiple projects.
* State Management-Managed Terraform state files effectively, implementing remote state storage in S3 with state locking via DynamoDB to ensure safe and consistent state management.
* Performed all necessary day-to- day GIT support for different projects, Responsible for design and maintenance of the **GIT Repositories**, and the access control strategies. Led the migration of continuous integration and continuous deployment (CI/CD) pipelines from Jenkins to GitLab.
* Monitored the servers and Linux scripts regularly and performed troubleshooting steps like testing and installing the latest software on server for end-users.

 **Environment: Unix/Linux, Shell scripting, Subversion, GIT version Control, GITHUB, Maven, Ansible, Chef, Nagios, Jenkins, AWS (EC2, DynamoDB, VPC, ELB, S3, RDS, Cloud Trail and Route 53, EBS, IAM, ELB, Cloud watch, Cloud Formation, Auto Scaling), Terraform, Docker, Kubernetes.**

 **Actions:**

* Assessed existing Jenkins pipelines, identified key components, and mapped them to GitLab CI/CD.
* Rewrote Jenkins files into GitLab CI/CD configuration files (.gitlab-ci.yml).
* Implemented GitLab Runners and configured them to match Jenkins agent setups.
* Ensured seamless integration with existing build tools, repositories, and third-party services.

**Results:**

* Reduced build times by 20% due to optimized GitLab CI/CD processes.
* Improved pipeline reliability and maintainability.
* Enhanced developer productivity with GitLab’s integrated code review and merge request workflows.

**Charter Communications, Denver CO**

**Mar 2018 –     Nov 2019**

**Sr Cloud Engineer/Devops Engineer**

**Responsibilities:**

* Involved in building and maintaining **Highly Available secure multi-zone AWS cloud infrastructure** utilizing Ansible with AWS Cloud Formation and Jenkins for continuous integration.
* Worked for new **HashiCorp Vault** setup on **EKS** Cluster.
* Upgrading current Vault version to 1.7.2 (Manual and then automating). Move licenses from open source to licensed versions.
* Created **RBAC roles** for testing and designed templates.
* Used IAM to assign roles, to create and manage AWS users, groups, and permissions to use AWS resources.
* Created Amazon Machine Images (AMIs) of EC2 instances for backup and creating clone instances. Creation and Configuration of S3 buckets with restricted policies.
* Created EBS volumes to store persistent data and mitigate failure by using snapshots. Performed Data Back-up of Amazon EBS volumes to S3 by taking point-in-time snapshots.
* Utilized **Cloud Watch** to monitor resources like EC2, CPU memory, Amazon RDS DB services, and EBS volumes to set alarms for notification or automated actions to monitor logs for better understanding of the operations.
* Converted existing AWS infrastructure to server-less architecture (**AWS Lambda**) and also migrated data from Datacenters and on-premises to Cloud using AWS Import/Export Snowball service.
* Performed all necessary day-to- day **GIT** support for different projects, Responsible for design and maintenance of the GIT Repositories, and the access control strategies.
* Provisioned the highly available **EC2** instances using **Terraform** and **Cloud Formation** templates, wrote new plugins to support new functionality in Terraform and involved in using terraform **migrate** legacy and monolithic systems to **Amazon Web Services.**
* Utilized Kubernetes for the runtime environment of the CI/CD system to build, test deploy.
* Worked on implementing **Kubernetes** cluster with **Docker** to provide a platform for automating deployment, scaling, and operations of application containers across clusters of hosts.
* Design, build and manage the **ELK** (Elastic Search, Log stash, Kibana) cluster for centralized logging and search functionalities for the App.
* Configured **JIRA** workflows according to the needs to the CM team and integrated the project management features with build and release process.

**Environment**: **Unix/Linux, Shell scripting, GIT version Control, Maven, Ansible, Nagios, Jenkins, AWS (EC2, VPC, ELB, S3, RDS, Cloud Trail and Route 53, EBS, IAM, ELK, Cloud watch, Cloud Formation, Auto Scaling), Terraform, Docker, Kubernetes.**

**JPMorgan Chase, Columbus OH**

**Oct 2016 – Feb 2018**

**Cloud Engineer/ DevOps Engineer**

**Responsibilities:**

* Created **AWS** **Cloud Formation** templates to create custom-sized **VPC**, subnets, EC2 instances, **ELB**, security groups. Worked on tagging standards for proper identification and ownership of **EC2** instances and other AWS Services like **Cloud Front, cloud watch, RDS, S3, Route53, SNS, SQS, Cloud Trail**.
* Configured **AWS Virtual Private Cloud (VPC**) and Database Subnet Group for isolation of resources within the **Amazon RDS MySQL DB cluster, AWS-Dynamo DB and AWS-RedShift.**
* Virtualized the servers on **AWS** using the **Docker**, created the Docker files and version control to achieve Continuous Delivery goal on a highly scalable environment, using Docker coupled with load-balancing tool Nginx.
* Responsible for Administering and maintaining **Jenkins** and **Jenkins slaves** on windows and Linux (Debian/Ubuntu). Created many Jenkins slaves and setup jobs on master to run on slaves.
* Worked to setup **Jenkins** as a service inside the **Docker swarm** cluster to reduce the failover downtime to minutes and to automate the **Docker** containers deployment without using configuration management tool.
* Worked on Docker to containerize the Application and all its dependencies by writing Docker files, Docker-Compose files, Docker container snapshots, managing Docker volumes, and deployed Docker Swarm using **Ansible**.
* Worked with **Ansible** to manage Web Applications, Config Files, Data Base, Commands, Users Mount Points, and Packages. Ansible to assist in building automation policies.
* Utilized **Ansible** for configuration management of hosted Instances within AWS. Configuring and Networking of Virtual Private Cloud (VPC).
* Set up build environment integrating with **GIT** and **Jira** to trigger builds using **Web Hooks** and **Slave Machines** byintegrating **Docker** container-based test infrastructure to **Jenkins** CI test flow.
* Worked with **Ansible** for Orchestration of Deployments for various servers and managed **Ansible** Playbooks with **Ansible** modules, implemented CD automation using **Ansible**, also involved in managing existing Tomcat Web Servers and automated the configuration of new servers.
* Created **Kubernetes** cluster with objects like Pods, Deployments, Services and ConfigMaps and created reproducible builds of the Kubernetes applications, managed Kubernetes manifest files and Helm packages and implemented Kubernetes to deploy scale, load balance, scale and manage Docker containers with multiple namespace versions.
* Worked on the installation and configuration of the monitoring tool **Nagios**, updated monitoring parameters with active and passive check and monitored network devices and reviewed network logs to identify potential problems.
* Setup **SCM** Polling for Immediate Builds with **Maven** and **Maven Repository** (**JFrog** Artifactory) by installing **Jenkins Plugins** for **GIT** Repository.

**Environment: Linux, Shell scripting, Groovy, Apache, GIT, Maven, Jenkins, Vagrant, Puppet, Chef, WebLogic, AWS (ELB, VPC, EC2, S3, Route53, EBS, IAM, Cloud watch, Cloud Formation), Docker.**

**S&P Global, India**

**Jul 2012 –Apr 2016**

**Build and Release Engineer**

**Responsibilities:**

* Involved in **DevOps migration/automation** processes for build and deploy systems. Used **Bamboo** and implemented **CI/CD pipeline** using plugins like conditional build step, deploy to GIT.
* Setting up continuous integration and formal builds using **Bamboo**withthe **Nexus** repository and resolved update, merge and password authentication issues in **Bamboo** and **JIRA**.
* Analyzed frequently occurring system breaks and automated fixes with repetitive tasks using **Shell and Python** scripts and used to send notifications after each deployment.
* Implemented and configured **Nagios** for continuous monitoring of applications and enabled notifications via emails and text messages.
* Involved in setting up **JIRA** as defect tracking system and configured various workflows, customizations, and plugins for the JIRA bug/issue tracker.
* Integrated **Bit bucket** with **JIRA** for transition JIRA issues from within Bit bucket Server and monitored the JIRA issues in Bit bucket Server.
* Experience with **RHEL** Linux systems, virtualization in a large-scale environment, experience with Linux Containers.
* Design and extract various **reports** of Servers and Services.
* Prepared technical and non-technical **documents** for knowledge base.
* Extract various reports for interfaces from logs.
* Performing new releases and upgrades
* Linux – install, manage (Ubuntu, Red Hat, CentOS)
* Windows Server – install, manage (2008, 2012)
* Configuration and systems management for Linux and Windows.
* Developing and managing backup/recovery infrastructure
* Frequent, incremental code testing and deployment
* Managing build infrastructure (Jenkins, Travis, SVN, GIT)
* Administer and monitor the Applications configured on servers and fine tune or troubleshoot as per requirements from time to time.

**Environment: ANT, Jenkins, RHEL, Bamboo, JIRA, Web logic, Python, Java/J2EE, Shell Scripts, GIT, UNIX, Ubuntu LINUX, Windows Server 2008/2012.**