**Teja Kanchi**Logo

Description automatically generated

**571–222-5869|** [**tejkdevops689@gmail.com**](mailto:tejkdevops689@gmail.com)

[**https://www.linkedin.com/in/teja-927b1a166/**](https://www.linkedin.com/in/teja-927b1a166/)

**Senior DevOps Engineer/AWS Cloud/ SRE/Platform Engineer**

**Professional SUMMARY:**

* Overall, 9 years of IT experience in Infrastructure, Build & release, Automating, configuring and deploying instances on cloud environments and Data centers.
* Experience in the full Software Development Life Cycle (**SDLC**) and Methodologies and validations to ensure Quality Assurance Control.
* Experienced in Infrastructure Development and Operations involving **AWS Cloud platforms like EC2, EBS, S3, VPC, RDS, SES, ELB, Auto scaling, Cloud Front, Cloud Formation, Elastic Cache, Cloud Watch, SNS, AWS Import / Export**.
* Experience in **Azure** Development, worked on **Azure** web application, App services, **Azure** storage, **Azure** **SQL Database**, **Virtual Machines, Fabric controller, Azure AD, Azure Search, and Notification hub**.
* Good experience in working with different Bug Tracking Tools like **JIRA, ServiceNow, Rally, Clear Quest, and Quality Center (QC)**.
* Design, build, secure, and manage clusters/workloads running on self-managed Kubernetes Amazon EKS (Amazon Kubernetes Service), Amazon ECS, and AWS Fargate.
* Installed and Configured the Chef Server, Workstation, Client servers, and nodes. Written several recipes and cookbooks in Chef to automate the environment provisioning, and Middleware Infrastructure Installations.
* Experience in Designing, Installing, and Implementing **Ansible** configuration management systems and in writing playbooks for Ansible and deploying applications.
* Designed **Puppet** Modules to manage configurations and automate installation process and developed automation scripting in Python using Puppet to deploy and manage Java applications across Linux servers.
* Skills in writing **Bash Scripts, Perl Scripts (hash and arrays), Groovy Scripts, and Python programming for deployment of Java applications on bare servers or Middleware tools**.
* Good knowledge and experience in using **Elasticsearch, log stash, Kibana and fluent dB, Cloud Watch, Nagios, Splunk, New Relic, Prometheus, and Grafana** for logging and monitoring.
* Strong experience creating **ANT/MAVEN** with Ansible scripts for deployment. Experience with Maven (Pom.xmls) and Ant (build. xmls).
* Good understanding of **Java 1.8 & JEE** design patterns have implemented custom frameworks for clients using a combination of industry practices.
* Experience with container-based deployments using **Docker**, working with Docker images, Docker hub, and Docker registries, installation and configuring Kubernetes, and clustering them.
* Good understanding of Design Patterns, Multithreading, and GO containers. Well-versed in using Channels and goroutines.
* Implemented **Kubernetes** to deploy scale, load balance, scale, and manage docker containers with multiple name-spaced versions.
* Good interaction with developers, managers, and team members to coordinate job tasks and a strong commitment to work.
* Experience in on-call support to support the environment 24/7 as required.

**TECHNICAL SKILLS:**

| * Integration Configuration Management tools | * Jenkins, Bamboo, Hudson, Ant, Maven, Chef, Ansible, Puppet, AzureDevOps |
| --- | --- |
| * Version Control Tool | * Git, GitHub, Bit Bucket, SVN (Subversion), Tortoise. |
| * Cloud Environment | * Amazon Web Services(AWS), Azure, Open Stack, Google Cloud Platform (GCP) |
| * Tools | * CHEF, PUPPET, HUDSON/JENKINS, XL Deploy, XL Release, MAVEN, CVS, SVN, GIT, GITHUB, TORTOISE SVN, NEXUS, NAGIOS, VAGRANT, DOCKER, ANSIBLE, SPLUNK. |
| * Operating Systems | * Red Hat Linux 4/5/6/7, CentOS, Fedora, SUSE LINUX, UNIX, Windows servers [2008, 2008 R2, 2012] Windows, Ubuntu, Sun Solaris, HPUX 10.x/11.x |
| * Scripting Languages | * Yaml, Python, ruby, shell, power, shell, J son |
| * Web/Application Servers | * Apache Tomcat, JBOSS, Web Logic 9, WebSphere, Apache 2.0.x, and Nginx, IIS |
| * Database Technologies | * SQL Server, MySQL, RDS, Mongo DB, Cassandra DB, DynamoDb |
| * Web Technologies | * HTML, CSS, J Query, Java Script, XML, JSON |
| * Monitoring Tools | * Nagios, Splunk, Elastic Search, Log stash and Kibana(ELK), Cloud Watch, Cloud Trial, Dynatrace |
| * Volume Manager | * VERITAS Volume manager, LVM |
| * Application Servers | * Web Logic Application Server 9, 10, Apache Tomcat 5/7, Red Hat JBOSS, |

**Work Experience:**

**Client: Leidos, Woodlawn, MD (Remote)                                       Sep 2022 -present**

**Role: SRE/Cloud Devops Engineer**

**Responsibilities:**

* Collaborated within **Agile** frameworks, actively participating in sprint planning, stand-ups, and retrospectives, fostering a culture of continuous improvement and iterative development.
* Designed **architecture diagrams** for code commit backups in AWS, automated backup infrastructures using CloudFormation, and managed log archives in S3 buckets for effective data retention and retrieval.
* Configured **Route 53** by using **CFT** templates, assigned the DNS mapping for the AWS servers, and troubleshooted the issues of the load balancer, auto-scaling groups, and Route 53.
* Built servers using AWS, including importing necessary volumes, launching **EC2** instances, S3 for object static WebPages, and creating security groups, auto-scaling, load balancer, Route 53, and **SNS** as per architecture using IAAC tools like Cloud formation JSON Templates.
* Built AWS Cloud Formation templates to create custom-sized **VPC**, subnets, and NAT to ensure successful deployment of Web applications and database templates and setting up EC2 instances, security groups, and databases in AWS using S3 bucket and configuring instance backups to an S3 bucket.
* Responsible for building AWS infrastructure **VPC, EC2, S3, IAM, EBS, Auto scaling and RDS** in **cloud formation** using **JSONtemplates**.
* Written various **AWS Lambda** services for automating the process of taking **EBS** Snapshots every day in a production environment on the cloud.
* Written templates for AWS infrastructure as code using Terraform to build staging and production environments. I also defined Terraform modules such as Compute, Network, Operations, and Users to reuse in different environments.
* Developed microservice onboarding tools leveraging Python and Jenkins, allowing for easy creation and maintenance of build jobs and deployment of services.
* Maintained over 40+ jobs in **Jenkins** from 8 different Application teams for over 4-5 releases in parallel.
* Responsible for the Batch team to handle the email daily connectivity issues, Create tickets, inquire about the existing tickets, and troubleshoot the connectivity and SFTP issues.
* Captured the nexus logs from the nexus server and replicated it to log archive and store them in S3 buckets
* Automated infrastructure setups and compliance checks using CloudFormation, worked on AWS **Config** rules development, and implemented Python scripts for S3 object encryption analysis, ensuring adherence to security standards.
* Developed **scripts** for analyzing S3 buckets' lifecycle policies, retention periods, and replication statuses across multiple accounts, ensuring compliance with storage policies.
* Update **PostgreSQL** client version 14 to 15 Across all multiple accounts on EC2.
* Prepared a document of the list of s3 buckets in all accounts which have or haven’t **lifecycle** **policy**, retention period and replication status.
* Played a pivotal role in providing **production support**, swiftly addressing critical issues, and ensuring minimal downtime by implementing proactive measures and troubleshooting solutions.

**Environment:** AWS EC2, Cloud watch, Cloud Formation, Git, Ansible Linux, Jenkins, EC2, python 2.7, Maven, SonarQube, Docker, AWS, Web Logic, Nexus, Splunk, Tomcat, RHEL, Shell script, Kubernetes, Docker**.**

**Client: Verizon, Dallas, TX (Remote)                                      Oct 21-Sep 2022**

**Role: SRE/Devops Prod Support**

**Responsibilities:**

* As a Level 3 Prod support Engineer, Orchestrated the full scope of day-to-day Production Support services, effectively resolving live incidents and conducting post-resolution problem management activities, ensuring minimal service disruption and maximizing system uptime..
* Actively contributed to continuous improvement initiatives, suggesting and implementing enhancements to existing processes, resulting in increased efficiency and reliability of systems.
* Maintained detailed documentation of **incidents, resolutions,** and best practices, facilitating knowledge sharing and ensuring the team's readiness for similar challenges in the future.
* Used **Terraform** for managing the infrastructure through the terminal sessions and executing scripts in creating alarms and notifications for **EC2 instances** using **AWS Cloud Watch.** Written Ansible Playbooks for various applications and deployed them in **AWS** using **Terraform.**
* Deployed application, which is containerized using Docker onto a Kubernetes cluster, which Amazon Elastic Container Service manages for Kubernetes (EKS).
* Utilized Ansible and Jenkins to automate the provisioning of our identity management solution which is used to implement Single Sign On for AWS. EKS authentication is integrated with SSO as well.
* Built/deployed custom Docker images from Artifactory into the EKS K8s cluster as part of a Gitlab CI pipeline.
* Initiating alarms in **CloudWatch** service for monitoring the server's performance, CPU Utilization, disk usage, etc., to take recommended actions for better performance.
* Utilized Amazon **Route53** to manage DNS zones and also assign public DNS names to elastic load balancers IPs.
* Built servers using **AWS** which included importing necessary volumes, launching **EC2** instances, **S3** for object static WebPages, and creating security groups, auto-scaling, **load balancer**, **Route 53,** and **SNS** as per architecture using **IAAC** tools like Cloud formation **JSON** Templates.
* Configured Tomcat and Apache web servers for reverse proxy redirection to UI application.
* Automated various infrastructure activities like Continuous Deployment, Application Server setup, and Stack monitoring using Ansible playbooks and has Integrated **Ansible with** Jenkins
* Wrote the Ansible playbooks which are the entry point for Ansible provisioning, where the automation is defined through tasks using YAML format to Run Ansible Scripts to provision Dev servers.
* Used Ansible to manage Web applications, Environments configuration Files, Users, Mount points, and Packages.
* Integrated Kubernetes with network, storage, and security to provide comprehensive infrastructure and orchestrated containers across multiple hosts.
* Created metrics and monitoring reports using Prometheus and Grafana dashboards.
* Engineered Python scripts facilitating efficient uploading and downloading of objects within AWS S3 buckets, streamlining data management processes.
* Utilized **Jenkins** pipelines to orchestrate the deployment of microservices onto Docker registries and Kubernetes clusters, managing Pods and leveraging Amazon EKS for efficient container orchestration.
* Engineered Python scripts for regular backup and restoration of database tables, enhancing data security and resilience through automated weekly backups.
* Leveraged Kubernetes as the runtime environment for the CI/CD system, enabling streamlined build, test, and deployment processes within a containerized ecosystem.

**Environment:** AWS, Windows 2008, Tomcat Apache, Cloud watch, Cloud Formation, Git, Ansible Linux, Jenkins, EC2, python 2.7, Maven, SonarQube, Docker, AWS, Web Logic,, Nexus, Splunk, UNIX, Tomcat, , Service-Now, Windows, RHEL, , LDAP, Shell script, Kubernetes, Docker**.**

**Client: Truckstop, Bosie, ID                                                    Nov 20-Oct 21**

**Role: Devops Cloud Engineer**

**Responsibilities:**

* Provided cross-functional support, collaborating closely with development, QA, and operations teams to ensure seamless integration of new features and enhancements into the production environment.
* Involved in designing and configuring Hybrid cloud, which enabled servers containing highly sensitive data on-premise while connecting with servers migrated to AWS Cloud.
* Managing the application deployments from GitLab ci to Kubernetes with the help of GitLab ci pipelines and setting up the **GitLab** repository and Runner for build automation and help the developing and deploying the services in both AWS and Azure.
* Efficient in Build and configuration of **infrastructure** in cloud development through the usage of the **Terraform** tool as **IAC**, and the ability to modify the **Terraform** modules based on the requirement of a development project.
* Create/Manage infrastructure automation in AWS cloud with terraform, including **EC2, EKS, ELASTIC SEARCH, ELASTICACHE, VPC, IAM, etc.**
* Daily monitoring production servers using **Grafana** and **Prometheus** which is integrated with Kubernetes, exceptions, and report to the team if something happens during standups.
* Worked with **AWS, lambda, Python,** and Big Query to create log solutions for up-to-date analytics On Contract to Infor Developed DevOps tools cross-platform utilities with Python with AWS Boto3.
* Created **users, indices, and aliases** in **Elastic Search APM** through the GITLAB ci pipeline by writing the scripts and integrating them into the service repo.
* Created Python scripts to totally automate AWS Services, which includes **ELB, Cloud Front distribution, database, EC2 and database security, S3 bucket,** and application configuration; this script creates stacks, single servers or joins web servers to stacks.
* Configured and managed Elastic Load Balancing (**ELB**) to implement fault tolerance and avoid single point of failure of applications, thus providing high reliability and efficient usage.
* Maintained and automated the scripts by using **Python**for automated deployments.
* Familiar with **helmcharts** for deployment manager to use  **charts** and templates for listed file names.
* Implemented **Pod security policies (PSP)**in AKS for required best practices and the ability to control what pods to be controlled, scheduled in the AKS cluster to prevent some possible security vulnerabilities or privilege Escalations.
* Built scripts on **Terraform** to launch the **Azure** VM instances to modify the resources and handle changes.
* Managed deployments in EKS managed Kubernetes, set up multi-node cluster, and deployed containerized applications.
* Implemented HTTPS **Ingress controller**and used TLS certificate on AKS to provide reverse proxy, configurable traffic routing for individual Kubernetes services.
* Managing the Kubernetes deployments with GitLab ci k8’s runner from the pipeline.
* Troubleshoot the CICD pipeline errors and monitor the CPU metrics, APM data, and Kafka logs.
* Creating **Kafka** topics, **API** keys in the confluent Kafka and store those keys in **AWS Secrets.**
* Extended support for existing product teams on how to integrate CICD into development life cycle.
* Gather new requirements with the product owner and create tasks in the backlog, help with groom prioritize and estimations. Once in every two weeks perform deployment to production environment.

**Environment: Gitlab, Terraform, CloudFormation, AWS, Azure Agile, Kubernetes, Docker, Jira, Confluence, Python, Spinnaker, , GitLab-Runner, Grafana, Prometheus.**

**Client: Capital One, Richmond, VA.                                     May 20-Nov 2020**

**Role: Cloud Engineer**

**Responsibilities:**

* Design, integration, and management of Amazon Web Services (AWS) cloud solutions. Provisioning of EC2 instances and integrated into local area offices in 5-time zones. Amazon RDS, VPC construction, Security Group policies, **IAM, APIs, Route 53, CloudFormation, S3, Glacier, OpsWorks.**
* SSL Security certificate management for enterprise, maintaining certificates across multiple ssl providers, and integrating certificates into products such as **nginx, apache, tomcat, AWS-ELB**.
* Used AWS Beanstalk for deploying and scaling web applications and services developed with Java, PHP, **Node.js, Python, Ruby, and Docker** on familiar servers such as Apache, and IIS.
* Expertise in docker compatibility testing on various Linux distros, Amazon Web Services (AWS) and Google cloud Engine (GCE).
* Used Vagrant and Ansible to install software and automate installation.
* Performed Automation Tests for the app using python Unit Test Framework.
* Having Knowledge on **AWS Lambda, Autoscaling, Cloud Front, RDS, Route53, AWS SNS, SQS, SES.**
* Developed AWS Lambda Functions using python boto3 library and configured the API Gateway to trigger those functions on respective API call.
* Created **IAM** roles for the lambda functions to access other AWS Resources.
* Used **Jenkins** and Pipelines which helped us drive all micro services builds out to the Docker registry and then deployed to Kubernetes.
* Installed and Implemented Ansible configuration management system. Used Ansible to manage Web applications, Environments configuration Files, Users, Mount points and Packages.
* Maintained high availability clustered and standalone server environments and refined automation components with scripting and configuration management (Ansible) and experienced in writing Ansible scripts.
* Experience in writing **Ansible** scripts to configure tomcat servers and maintain application deployments.
* Designed, Installed and Implemented Ansible configuration management system.
* Wrote Ansible Playbooks with Python **SSH** as the Wrapper to Manage Configurations of AWS Nodes and Test Playbooks on AWS instances using Python.
* Enhanced the User Experience through Alexa Skill Kit, hosted on AWS Lambda Function.
* Experience in Performance Tuning and Query Optimization in AWS Redshift. Managed Amazon Redshift clusters such as launching the cluster and specifying the node type.
* Hands-on experience with Docker containers to maintain Docker images, Docker hub & registries.
* Implemented Terraform modules for deployment of applications across multiple cloud providers.

**Client: Western union, Denver, CO.                                   May 19-May 2020**

**Role: AWS Devsecops Engineer**

**Responsibilities:**

* Automated way of creating entire infrastructure with the all required **AWS services** into Dev, QA, UAT and Production environments
* Modules written to automate **IAM roles, S3 bucket** and its policies, lambda function with dependencies, **DynamoDb**tables automation, **EC2, ASG**, Launch configuration and application load balancer with conditional rules.
* Written **terraform** scripts to fetch latest **AMI** from Amazon marketplace and install webserver with desired configuration using user data.
* Written **terraform** modules to reuse the code to spin up multiple environments with the same code.
* Designed an automated way to insert the data into **DynamoDB** tables, whenever a CSV file uploaded to S3, event triggers lambda function. Then lambda reads that CSV and split the data and then insert into the DynamoDB table.
* Worked closely with **security engineering and cloud engineering**teams for assessments of various technologies and mitigating the risks around it.
* Experienced working with **Java, Python** complementary programming languages, operating systems and supporting software’s. Performed software configuration/release management activities for three different Java applications
* Hands-on experience in performing **B/G deployments** with AWS services using **Jenkins** pipelines and shared libraries.
* Extensively worked on AWS Developer tools like **CodeBuild, CodeCommit, CodeDeploy,CodePipeline** to implement CICD Process using CFT’S to automate the infrastructure.
* Created and managed the security third parties tools**Tenable, Privx, Cyberark** Applications and deployed in AWS Service Catalog.
* Implemented CI pipelines for the customer and use **DevSecOps** principles and practices to get the most out of the tool chain.
* Experience with Front-end JavaScript frameworks like **Angular JS, Node JS**. And with the creation of RESTful Web service
* Setup and configure S3 buckets, S3 lifecycle policies, EFS and EBS volumes to store various application content.
* Manage **Kafka** consumer and producer deployments.
* Setting up the cluster environment for **WebLogic** Server integrated with multiple workflows.
* Created **Virtual Private Network, subnets, route tables and security groups** to control inbound and outbound traffic.
* Used Jenkins pipelines to drive all micro services builds out to the **Docker** registry and then deployed to **Kubernetes**, Created Pods.
* Manage large scale **Kafka** environment builds, including design, capacity planning, cluster setup, performance tuning and ongoing monitoring.
* Worked on automated ingestion of **VPC** flow logs, DNS query logs, **S3** bucket logs from multiple AWS accounts through **kinesis firehose.**
* Managed different infrastructure resources, like physical machines, VMs and even Docker containers using **Terraform** It supports different Cloud service providers like AWS.
* Setup and configure **CloudTrail** to collect audit logs and store it to relevant S3 bucket.
* Supported AWS Cloud environment with many AWS instances and configured Elastic IP and Elastic storage.
* Experience in creating **Docker** Hub, Docker Images and handling multiple images primarily for middleware installations and domain configurations.
* Wrote Python using **Ansible** Python API to Automate Cloud Deployment Process. Developed Python Modules for **Ansible** Customizations.
* Managed **Docker** networking subsystem using User-defined bridge networks, Host networks, overlay networks, and third-party network plugins. Created additional Docker Slave Nodes for Jenkins using custom Docker Images and pulled them to the Cloud. I worked on all major components of **Docker**, like Docker Daemon, Hub, Images, and Registry.
* Implemented Docker-Maven plugin and Maven POM to build Docker Images for all microservices and later used Docker file to build the Docker Images from the java jar files.
* Configuration and administration of Load Balancers, Network and Auto scaling for high availability.
* Worked with Amazon **IAM** console to create custom users and groups. Worked with various AWS EC2 and S3 CLI tools.
* Created Custom **Ansible** modules for finding facts about Cloud watch alarms and taking actions to pause/unpause those alarms during deployments.
* Virtualized the servers using the **Docker** for the test environments and dev-environments needs. And also, configuration automation using **Docker** containers.
* Experience in on-call support to support the environment 24/7 as required.

**Environment**: AWS, Kubernetes, Terraform, Jenkins, git, service now, docker, ansible, Kafka, Grafana,tenable,Privx

**Client: Caterpillar, Peoria, IL. Sep 18-April 2019**

**Role: Site Reliability and Release Engineer**

**Responsibilities:**

* Used the **Site Reliability Engineering** principles of change management, monitoring, emergency response, capacity planning, and production readiness reviews to run the platform.
* Test and deploy multiple **PAAS** services through **VSTS CI/CD** pipeline and create new **ARM** templates and artifacts to update the existing **PAAS** services as per the security standards. Extensively worked in converting the classic resources to **ARM** such as storage and Cloud service resources.
* Built the process of remediation for **Azure** resources, by automating using **JSON** Templates and PowerShell scripts and deploy it using build and release pipeline in **VSTS**.
* Manage code repo, improve code branching and merging processes and responsible member to troubleshoot the **build and release** issues.
* Have implemented backup methodologies by utilizing PowerShell Scripts for **Azure Services like Azure SQL Database, Key Vault, Storage blobs, App Services** etc.
* Worked on **POC** to implement the **selenium** and **JMETER** in **VSTS** for onboarding document.
* Worked with **Service Now** for handling incidents and raising change requests.
* Created **cloud formation** template for **IOT** topic rules in **IOT** Core.
* Written templates for **kinesis** stream and firehouse.
* Designed **Cloud formation** templates for launching complete application **stack** for non-prod and prod stacks with auto scaling feature.
* Expertise in writing new plugins to support new functionality in **Terraform**. Used Terraform scripts to Automate Instances for Manual Instances that were launched before.
* Experience in **Build and Release**(Installation**, Configuration, Tuning and Upgrades of Linux**Red Hat and Oracle).
* Azure cloud infrastructure provisioning and configuration activities for the project.
* Experience deploying applications onto their respective environments using **Elastic Beanstalk**.
* With the help of **IAM,** created roles, users, and groups and attached policies to provide minimum access to the resources.
* Worked with Automated deployments, Scaling, and operations of application containers across cluster of hosts, providing container-centric infrastructure by **Kubernetes**.
* Experienced with event-driven and scheduled **AWS Lambda functions** to trigger various AWS resources.
* Configuring **AppDynamics** and creating dashboards to monitor and gather required metrics.
* Experienced in creating **AppDynamics** dashboards, which helped in assessing business impacts of application outages.
* Deployment, troubleshooting, monitoring, and maintenance of applications in both **AWS/AZURE** environment.

**Environment:**Azure,AWS,Sonarqube,VSTS,ARM,Kubernetes,CICD,Ec2,VPC,ELB,Terraform,Python,JSON,Databases,Service Now, Key vault, KMS, CloudWatch, Jfrog, Docker.

**Client: FPT Software, Houston, TX Feb 2017- July 18**

**Role: Sr. Aws/DevOps Engineer**

**Responsibilities:**

* Build expertise in Workflow for Amazon Web services including Security Groups, **IAM, EC2, S3, ELB, EBS, Policy, VPC, Docker, Kubernetes, Cloud Formation, Chef, Jenkins Vagrant, Apache, Tomcat, Git and GitHub.**
* Design of Cloud architectures for customers looking to migrate or develop new **PaaS, IaaS, or hybrid solutions utilizing Amazon Web Services (AWS).**
* Creation and maintenance of building jobs and **Kubernetes** deployment and services.
* Building/Maintaining **Docker** container clusters managed by **Kubernetes Linux, Bash, GIT, and Docker.** Utilized Kubernetes and Docker for the runtime environment of the CI/CD system to build, test, and deploy.
* Used **Kubernetes** for creating new Projects and services for load balancing and adding them to Routes by accessing from outside, created Pods through new applications, and controlling, scaling, and troubleshooting pods through SSH.
* Integrated **Docker** container orchestration framework using **Kubernetes** by creating pods, config Maps, and deployments.
* Configuration management using Amazon **Cloud Formation**, Continuous integration with **Jenkins** and **Bamboo**.
* Used Jenkins pipelines to drive all microservices builds out to the **Docker** registry and then deployed to Kubernetes, Created Pods, and managed using Kubernetes.
* Provisioned the highly available **EC2** Instances using **Terraform** and **cloud formation** and wrote new plugins to support new functionality in **Terraform**.
* Deployed and managed many clusters on the Aws platform and on-prem to automate multiple tasks **using Terraform and Ansible playbooks**.
* Hands-on experience in **Terraform** for building, changing, and versioning of Infrastructure and wrote Templates for AWS infrastructure as a code using Terraform to build staging and production environments.
* Created Automation to create infrastructure for **Kafka clusters** different instances as per components in the cluster using **Terraform** for creating multiple EC2 instances & attaching ephemeral or EBS volumes as per
* Written a few **Python** (2.7) workers to automate the tasks, including taking backups.
* Created and transitioned the infrastructure to AWS using a weighted routing policy.
* Setting up the build and deployment automation for **Terraform** scripts using **Jenkins** and Used HCL scripts for writing Terraform Scripts.
* Used **Jenkins** as a continuous integration tool for managing required plugins, configuring the jobs using Git plugins, building system, and post-build actions, scheduling automatic builds, notifying the build reports, SonarQube, etc.

**Environment:** AWS, Git, Ansible, Ansible tower, Terraform, Kubernetes, Cloud Formation, Groovy, Python(2.7), Shell, Jenkins, Maven, Tomcat, Nexus, Artifactory (Jfrog), SonarQube**.**

**Client: Fox Communications, Washington, DC Dec 2015 – Jan 2017**

**Role: DevOps Engineer**

**Responsibilities:**

* Authored **Terraform** modules for infrastructure management. Authored and published a module to the Terraform registry for enterprise customers to deploy our product inside their AWS environment. And Set up build and deployment automation for **Terraform** scripts using Jenkins.
* Develop **Docker** images to support Development and Testing teams and their pipelines and Assisting in design micro services deployment using Docker.
* Orchestration of **Docker** images and Containers using Kubernetes by creating whole master and node.
* Maintaining the security groups in **EC2, VPC** and controlling the inbound and outbound traffic that can reach the instances.
* Tasks performed on performance tuning, capacity planning for AWS environments, monitoring, resource utilization, and alerts using CloudWatch; implementing hybrid architectures, routing private connections, peering and infrastructure zoning by **VPC; Cloud Formation** templates to deploy infrastructure for environment creation; Code Pipeline to design and implement a **CI/CD** pipeline.
* Optimized volumes and **EC2** instances &created multi-AZ VPC instances and used **IAM** to create new accounts, roles, and groups.
* Configured **S3** to host static web content, versioning, and lifecycle policies to and backup files and archive files in Glacier.
* Used Kinesis to aggregate the logs of web applications and then move those logs to AWS Redshift.
* Configured the **Kubernetes** cluster on **GCP** to establish communication between pods and MySQL instance on RDS of **AWS**.
* Deployed **Docker** containers by using images stored in the repository using Kubernetes, both on-premise and on **AWS** cloud.
* Prepared Java/J2EE development structure for Eclipse, maven
* Used Eclipse as IDE, configured and deployed the application onto the **WebLogic** application server using **Maven** build scripts to automate the build and deployment process.
* Troubleshooted emerging application issues from the **WebLogic** configuration side.
* Setting up the cluster environment for WebLogic Server integrated with multiple workflows.
* Developed various helper classes following Core Java multi-threaded programming and Collection classes.
* Implemented the use of the **Nagios** tool for monitoring and analyzing the network loads on the individual machines by enforcing custom Nagios monitoring, notifications, and dashboards to exhibit various metrics using **Shell** Scripting and add-ons.
* Branching, tagging, and maintaining the versions across the environments using **Git** on **Linux** and Windows
* Involved in **Jira** as a defect tracking system and configured various workflows, customizations, and plug-ins for Jira bug/issue tracker.

**Environment:** Amazon Web Services (AWS), Amazon EC2, Amazon EBS, Amazon Storage Services S3, CloudWatch, Auto Scaling, Route53, IAM, Glacier, AWS Opsworks, AWS Cloud front, PCF, Ansible, Java, J2ee, Oracle, Web logic, Jenkins, Nagios, Consul, Nexus, Vagrant, Linux.

**Client: Auxis - Houston, Texas Jan 2013 – Mar 2014**

**Role: DevOps Engineer**

**Responsibilities:**

* Developing automated processes for builds and deployments using Jenkins, Ant, Maven, Nexus, Shell, Python, and Java.
* Setup of Virtual Private Cloud (**VPC**), **Network ACLs, Security Groups, and route tables across Amazon Web Services.**
* Configuration and administration of Load Balancers, **Route53**, Network, and Auto scaling for high availability.
* Integrated GitHub web hooks into Jenkins to automate the code check-out process.
* Migrated various applications and services of teams from on-premises to **AWS cloud using AWS resources like EC2, VPC, S3, Security Groups, etc.,**
* Generated Ant, shell scripts for build activities in QA, Staging, and Production
* Worked on the transition project, which involves migration activities from Ant to Maven to standardize the build across all the applications.
* Migrated source code management tool from SVN to GitHub and ensured identical configuration.
* Managed Users and Groups in **GitHub** and involved in troubleshooting client spec issues and user issues.
* Configured local Maven repositories and multi-component Ant projects with Nexus repositories and scheduled projects in Jenkins for continuous integration.
* Maintained configuration files for each application for build purposes and installed them on different environments.
* Directed the Release Management Calls to synchronize with the Developers, Testers, and DBA teams for successful Release.
* Administered and integrated the Version Control Systems like Subversion and Git with **Jenkins**.
* Developed Ant and Maven scripts to automate the compilation, deployment, and testing of Web and J2EE applications.
* Developed **Python** (2.6) and shell scripts to automate the build and release process.

**Environment:** AWS, Git, putty, Linux, Windows, SVN, Java/J2EE, Ruby, Eclipse, Ant, Jenkins, Maven, Jira, Junit, Unix/Linux, Tomcat Apache Application Server.

**Education: Bachelor's in computer science, JNTUK, India,2012**

**Certifications: AWS Solutions architect -associate (AWS-ASA-34860)**