**Harsha Vardhan**

**Sr. DevOps/Cloud Engineer**

**+1 469 277 7354**

**rishitha.23g@gmail.com**



**Professional Summary:**

* IT professional with 9+ years of experience in **DevOps, AWS, Azure, Google Cloud platforms (GCP)** and its features, **CI/CD** (Continuous Integration / Continuous Deployment) process, and robust background in **Linux Administration**.
* Proven exposure in Internet products Build and Release management, SDLC, Systems integration, and Environments Management.
* Hands-on experience creating **Chef Recipes, cookbooks,** and **Ansible Playbooks** and roles for cloud automation (AWS) and Application deployment automation.
* Ability to identify and gather requirements to define a solution to build and operate multiple applications on **AWS.**
* Experience in maintaining **AWS (Amazon Web Services) EC2 (Elastic Cloud Computing) resources like instances, EBS volumes, snapshots, Elastic Load Balancers (ELB), VPC, AMIs, Lamda, security Groups, elastic IPs,** Key pairs, and **Amazon Cloud watch** for different zones in different environments and **API Gateways.**
* Good Experience implementing Continuous Integration/Continuous Deployment Automation Processes using various tools like **Jenkins, Bamboo, and Ansible.**
* Experience writing **Infrastructure** as a code (**IaaC**) in **Terraform**, **Azure** resource management (ARM) Templates.
* Experienced in Windows Azure Services like PaaS, IaaS and worked on **Azure** **PaaS** storages like Blob (Page and Block), Azure SQL. Well-experienced in deployment & configuration management, and Virtualization.
* Skilled in implementing Linux features on Windows through VMs, UI’s, and cloud shell in Azure.
* Experience setting up the **CI/CD** pipelines using **Jenkins, Maven, Nexus, GitHub, Ansible, Terraform, and AWS.**
* Good Experience in Dockerized applications by creating **Docker** images from **Docker files** and handling large volumes of container clusters with **Docker Swarm, Kubernetes,** and **Mesos.**
* Expertise in building CI/CD on AWS environment using **AWS Code Commit**, Code Build, Code Deploy, and Code Pipeline and experience in using AWS Cloud Formation, **API Gateway**, and **AWS Lambda** in automation and securing the infrastructure on AWS.
* Experienced with building tools like **Jenkins, TeamCity, SonarQube, Maven, and ANT.**
* Experience setting up Baselines, Branching, Merging, and Automation Processes using **Shell, Perl,** **Python, and Bash Scripts.**
* Hands-on experience migrating Oracle, SQL Server instances to **AWS** **RDS** (Oracle to Oracle migration, Oracle to **Postgres** migration, **Oracle** to **MySQL**)
* Monitored and applied analysis using **Nagios** and **Splunk**. Used **JIRA** for bug tracking and creating the dashboard for issues.
* Hands-on experience setting up **Kubernetes** (**k8s**) Clusters for running **microservices**. Took several microservices into production with **Kubernetes-backed Infrastructure**
* Experienced in **Kubernetes** charts using **Helm** and created reproducible builds of the Kubernetes applications, managed Kubernetes manifest files, and managed releases of **Helm** **packages**
* knowledge of **IaC** tools like AWS **CloudFormation** and Terraform. Adept at utilizing **GitOps** principles for infrastructure provisioning and upgrades and using **Git** as the **version** **control** mechanism for infrastructure code.
* Hands-on experience with Azure web applications like **Azure App & Cloud Services**, **Azure Data Factory**, **Azure SQL Data** **Warehouse**, **Azure Blob Storage**, **Web API**, **VM creation**, Lift & Shift, Azure storage, Fabric controller, Azure AD, Azure Search, & Notification hub.
* Wrote **Ansible** **Playbooks** with Python **SSH** as the Wrapper to Manage Configurations of Open stack Nodes and Test Playbooks on AWS instances using Python
* Experience writing Infrastructure as a code (**IaaC**) in Terraform, Azure resource management (**ARM**) Templates.
* Experienced in Windows Azure **Services** like PaaS, IaaS and worked on Azure **PaaS** storages like **Blob** (Page and Block), **Azure** **SQL**. Well-experienced in deployment & configuration management, and Virtualization.
* Set up automated **build**, **test**, and release platform using **TeamCity**, **Jenkins** pipeline as a code, **SonarQube**, and **JFrog** **Artifactory** to be triggered on every code commit.
* Experience Upgrading **TFS** from the 2013 version to 2018 and migrating all the projects to the latest version.
* As part of cloud technologies, I worked on various clouds like AWS, where our **microservices** were frequently deployed in Dev, **STG**, and Prod environments.
* Hands-on Experience in deploying the **Maven** files into servers and storing the artifact files in the **Nexus** repository for controlling the version of the artifacts or builds.
* Experienced in troubleshooting, Configuring, and Deploying Enterprise Applications in **WebSphere,** BEA **WebLogic,** and **JBOSS/Tomcat**.
* Good Experience integrating JIRA with CI/CD Pipeline as a defect-tracking system and configuring workflows to automate deployment and issue tracking.
* Proficient in managing the source code control of multiple development efforts using **GIT,** **Subversion** control tools.
* Good interaction with developers, managers, and team members to coordinate job tasks and a solid commitment to work.
* Good analytical, problem-solving, and communication skills and the ability to work independently with little or no supervision or as a team member.

**Technical Skills:**

|  |  |
| --- | --- |
| **DevOps Tools** | Chef, Ansible, Puppet, SonarQube, Docker, Kubernetes, OpenShift, Hashicorp, Nexus, Terraform. |
| **Languages/Scripting** | Java, Python, Bash, PowerShell, Golang, PL/SQL |
| **Web Servers** | Apache Tomcat, WebSphere, Web logic, and JBOSS. |
| **CI/Build Tools** | Jenkins, Ant, Maven. |
| **Monitoring Tools** | Nagios, Splunk |
| **Version Controls** | SVN, CVS, GIT, Bitbucket. |
| **Databases** | Oracle, My-SQL, IBM DB2, MS SQL Server, Postgres. |
| **OS** | Windows XP/7/8, Linux, |
| **Other tools and technologies** | ELK (Elastic, Logstash, Kibana), Kafka, Grafana, AEM, Salesforce, Hadoop, Data Bricks, Prometheus, Argocd, WebSphere, Snowflake, Data Analysis and Visualization (tableau, power bi), Apache Pulsar, MS Office, and products |

**Education**

* Bachelors in Electronics and Communication Engineering- JNTUH - May 2014

**Professional Experience:**

**Client: Publix - McLean VA April 2022 – Till Date**

**Role: Sr Cloud DevOps Engineer**

**Description:**

DevOps Engineer with extensive experience in automating and optimizing mission-critical deployments over large infrastructures. Proficient in managing AWS services, deploying applications using CI/CD pipelines, and orchestrating Docker and Kubernetes environments. Adept at working with Pulsar/Kafka and Elasticsearch deployments, with a strong background in database management (Oracle, MySQL, PostgreSQL). Skilled in scripting with Python, Bash, and Shell for automation, and integrating various tools such as Jenkins, GitHub, and Ansible for seamless workflows. Proven ability in process improvement, infrastructure monitoring with Grafana, and maintaining high availability and fault tolerance in cloud and on-premises environments.

**Responsibilities:**

* Analyze and review the requirements, and identify process improvement and automation opportunities within the release process to continuously improve the overall product quality.
* Worked with Pulsar/KAFKA and ElasticSearch deployments and experience bringing up and administering a Pulsar.
* Created topics, set up redundancy cluster, deployed monitoring tools, and alerts with Apache Pulsar
* Experience with Grafana implementation and setting up dashboards.
* Implemented and managed release processes for code through development, test, and production environments.
* Designed and implemented multiple deployment strategies using **CI/CD** Pipeline through automation with enterprise **GitHub, Jenkins, Sonar Nexus, Docker,** and **AWS AMI** in a **Linux** environment.
* Good knowledge of AWS services like Glacier, **ELB** (**Load** **Balancers**), **RDS**, **SNS**, **SWF** and hands-on experience on Amazon Web Services (AWS) provision
* Developing the **Python** automated **scripting** using the **Boto3** library for **AWS** **Security** **audit** and reporting using **AWS** **Lambda** for multiple AWS Accounts
* Managed installing applications on **Docker** **containers**. Created **Docker** **volumes** to share data among multiple running **Dockers** **containers**
* Experienced in deploying applications into Kubernetes using Gitops Tools (Argo CD)
* Experienced in creating projects in Argo CD and deploying Argo CD into Kubernetes cluster from the scratch.
* Deployment into Kubernetes cluster directly from GitHub using the Argo CD tool
* Proven ability to manage database infrastructure in AWS (RDS and EC2)
* Strong hands-on background in database technologies (Oracle, Mysql, MS SQL, RDS, DynamoDB)
* Gathered data and presented a case to senior management to change direction and Migrate from Oracle to **EDB** **Postgres** Plus Advanced Server 9.3
* Automated the DBA activities for Oracle, PostgreSQL, and EDB Postgres Advanced Server.
* Involved in implementing the Amazon AWS RDS and Aurora.
* **Migrated** the on-premises **PostgreSQL**, Oracle & MYSQL databases to Amazon AWS RDS and Aurora Databases using Amazon AWS Data Migration Services (DMS) and AWS Schema Conversion Tool.
* Involved in Implementing and support of NoSQL (DynamoDB). Developed scripts for PostgreSQL, EDB Postgres Advanced Server databases for monitoring and tuning procedures.
* Worked on **docker** **hub**, **creating** **docker** **images** and handling multiple images primarily for **middleware** installations

and **domain** **configurations**.

* Create documentation for **Chef** best practices to be used by developers as a guide to **Chef** **Automation**.
* Implemented AWS solutions using **EC2**, **S3**, **AWS** **Lambda**, **RDS**, **IAM** (Identity and Access Management), **Route** **53**,

**EBS**, **Elastic** **Load** **Balancer**, Auto scaling groups, Optimized volumes, and EC2 instances using API Gateway.

* Involved in designing and deploying various applications utilizing the entire **AWS stack** **(EC2, Route53, S3, RDS, Dynamo DB, SNS, SQS, IAM**), focusing on high availability, fault tolerance, and auto-scaling.
* Deployed and operated Amazon MSK clusters and secured Amazon MSK clusters using IAM for Kafka.
* Used **Terraform** Plan to detect and analyze the changes inflicted after Terraform Apply meticulously**.**
* Extensively used **Ansible** Configuration management to deploy packages with the required build artifacts from Artifactory and facilitate a stable and scalable build and deploy infrastructure.
* Responsible for writing **Bash, Shell, Perl, and Python scripts** to ping servers, adding users to the boxes, and proficient in scripting languages to automate daily activities.
* Developed **Gradle and Maven** scripts to automate the compilation, deployment, and testing of Web and J2EE applications.
* Worked with Apache Kafka for High throughput for both publishing and subscribing, with disk structures that provide constant performance even with many terabytes of stored messages**.**
* Built end-to-end **CI/CD** Pipelines in **Jenkins** to retrieve code, compile applications, perform tests, and push build artifacts to **Nexus using Nexus** artifactory plugin.
* Configured security and system in **Jenkins**. Added multiple nodes to Jenkins and configured **SSH** for

**continuous** **deployments**.

* Building CI pipelines in Jenkins while fully scripting the creation and provisioning of Jenkins using Shell and Python.
* Shifted the **Kubernetes** **cluster** workload between clouds to **on**-**premises** via **Anthos**.
* Migrated all operations files and scripts to **GitHub** Enterprise installation to optimize deployment and allow tracking of configurations and scripts within GitHub
* Set up the build and deployment automation for **Java base** projects using **JENKINS** and **Maven.**
* Integrated **Docker container** orchestration framework with creating pods, config Maps, deployments, Replica sets, nodes, etc.
* Integrated **Jenkins** with various **DevOps tools** such as **Nexus, SonarQube** and used the **CI/CD** system of Jenkins on **Kubernetes container** environment, utilizing **Kubernetes and Docker** for the runtime environment for the **CI/CD** system to build and test and deploy.
* Implemented & maintained the branching and build/release strategies utilizing **BITBUCKET.**
* Working on **GIT** for data integrity and managed central repositories: Implemented **Atlassian Stash** with **GIT** to host **GIT** Central repositories for source code across products and facilitate code reviews and login audits for Security Compliance.
* Worked on proactive **database-level performance tuning** with an excellent working knowledge of **Oracle** internals.
* Integrated **AWS Connect** with various customer relationship management (CRM) systems to provide agents with relevant customer information during interactions, improving response times and customer satisfaction.
* Developed and implemented automated workflows using AWS Confect’s contact flows and AWS Lambda functions, resulting in streamlined processes and improved operational efficiency.
* Monitored **Elasticsearch** clusters using Prometheus and Grafana.
* Optimized **Elasticsearch** performance by tuning heap size, shard allocation, and replica settings.
* Worked on **Log Rotation** utility to set up the config file to migrate the logs from log producer servers to the Log stash server for parsing.
* Worked on **Atlassian** **Jira** to capture, organize, and prioritize Incident and Change tickets.
* Active participant in scrum meetings, reporting the progress and maintaining effective communication with each team member and manager.

**Environment:** Apache Pulsar, Grafana, Java/J2EE, Python, Ansible, Chef, Docker, Kubernetes, SonarQube, Nexus, Nagios, CI/CD, Jenkins, Terraform, AWS, Web GUI, SQL, Agile, Bash, GIT, Bitbucket, LINUX, UNIX, Confluence.

**Client: Insurance auto auctions, IL Feb 2020 – March 2022**

**Role: DevOps Engineer**

**Description:**

DevOps Engineer with extensive experience in building and deploying applications across various environments, including Azure and AWS. Proficient in creating CI/CD pipelines using Jenkins, Azure DevOps, and Argo CD. Skilled in container management with Docker and Kubernetes, and automation using Ansible and Terraform. Strong background in log analysis with Splunk, infrastructure scripting with Python and PowerShell, and database management. Implemented build and deployment scripts with Maven and ANT, and utilized Golang for API development. Collaborated with cross-functional teams to optimize performance, troubleshoot issues, and ensure continuous integration and delivery in cloud and on-premises environments.

**Responsibilities:**

* Performed and deployed Builds for various Environments like QA, Integration, UAT, and production environments.
* Developed build and deployment scripts **Maven** as build tools in **Jenkins** to move from one environment to another environments.
* Involved in automating Splunk setups and dashboard building can be automated using scripting languages like Python and PowerShell, as well as infrastructure-as-code tools like Terraform and Ansible.
* Worked on creating new **Azure Resource Manager** (ARM) Service connections for different teams.
* SPL queries were used to analyze the data and find patterns, abnormalities, and performance outliers. Analyzed data patterns and outliers using statistical computations, such as standard deviation.
* Integrated SPL queries into CI/CD pipelines and DevOps workflows in collaboration with cross-functional teams, enabling continuous monitoring and enhancing system performance.
* Designed and configured **Azure Virtual Networks** (VNets), subnets, Azure network settings, DHCP address blocks, DNS settings, security policies, and routing
* Worked on **Azure DevOps Multi-Stage YAML pipeline to Build and Deploy** .Net Core Applications into Azure APP Service Env.
* Handled **Docker** Engine, Docker-Hub, Docker Compose, Docker Registry, Docker Swarm, and Orchestration tools such as **Kubernetes**. Responsible for container creation and management for different applications on the Docker engine through writing and editing Docker files.
* Designed, configured, and deployed **Azure Automation Scripts** for many applications utilizing the Azure stack (Including Compute, Web & Mobile, Blobs, ADF, Resource Groups, HD Insight Clusters, Azure SQL, Cloud Services, and ARM), Services and Utilities focusing on Automation.
* competent with utilizing Splunk for log aggregation, analysis, and troubleshooting. Experienced in processing logs, building relevant insights, and establishing and managing log ingestion pipelines.
* used Splunk to analyze logs and troubleshoot issues to find the underlying causes of performance problems, system failures, and security events
* Configured existing **Azure DevOps pipelines** to **Continuous Integration (CI)** and **Continuous delivery (CD)** of applications
* Used Argo CD for continuous deployment and maintained release versions using tags in Gitlab. Worked on integration of Argo CD to Kubernetes for the management of Kubernetes resources.
* Developed the **PowerShell** scripts and **Azure Resource Management** (ARM) templates to automate the provisioning and deployment process, used **VSTS (Visual Studio Team Services)** for Continuous integration in Azure.
* Created **Shell & Python scripts** for various **Systems Administration** tasks to automate repeated processes.
* Worked closely with the development and operations organizations to implement the necessary tools and processes to support the automation of builds, deployments, testing, and infrastructure (infrastructure as code) using **Ansible.**
* Developed a **playbook** to configure, deploy and maintain software components of the existing Infrastructure.
* Used **Ansible** to automate Configuration management & Applications, manage Web Applications, Config Files, Data Base, Commands, Users Mount Points, and Packages.
* Using **Ansible with AWS** allowed me to reduce costs for the department and eliminate unwarranted resources.
* Used **Ansible** to automate Configuration management and to manage Web Applications, Config Files, Data Base, Users Mount Points, and Packages. Defined all servers in **Ansible** so the newly built server could be up and ready for production.
* Implemented New Relic to monitor the system and identify performance issues.
* Created **CI/CD pipeline** in **Jenkins** and run the build by integrating with the GitHub repository using **Ansible build** and deploy scripts.
* Built end-to-end **CI/CD Pipelines** in **Jenkins** to retrieve code, compile applications, perform tests, and push build artifacts to Nexus using **Nexus artifactory plugin.**
* Managed the deployment and maintenance of Adobe Experience Manager (AEM) on AWS.
* Customized the AEM operations dashboard and wrote health checks for monitoring and created workflow purge jobs.
* Worked on **Docker container** snapshots, attaching to a running container removing, image managing, and directory structures.
* Designed and implemented **Nexus binary repository** and process.
* Created **Bash Scripts** and cron jobs that observed and described security issues.
* Involved in building database Models, APIs, and Views utilizing **Python** to build an interactive web-based solution.
* Used **Docker, Kubernetes,** and Open Shift to manage microservices for continuous integration and delivery development.
* Utilized **Kubernetes** for the runtime environment of the CI/CD system to build, test deploy.
* Build and release **Cloud-based** products containing Linux and Windows environments using **Python Scripting.**
* Developed build and deployment scripts using **ANT and MAVEN** as build tools in Jenkins to move from one.
* Managed **SVN/Git** repositories for branching, merging, and tagging; resolved merging issues during rebasing and re-integrating branches by conducting meetings with the development team.
* Experience using Golang to design a new API and Microservices were developed using channels, functional interfaces, and a variety of contexts.
* Maintaining track of the number of deployments and the details of the specific environments using **JIRA.**
* Worked closely with various teams to resolve issues, including Business, QA, Build & Deployment, and Production Support teams.

**Environment:** Puppet, Jenkins, Maven, AWS, Bash Scripts, Python, CI/CD, GitHub, Ansible, Terraform, Docker, Nexus, Kubernetes, SVN, Jira, Agile and LINUX, Windows, Postgres.

**Client: Diversified, NJ Dec 2017 – Jan 2020**

**Role: DevOps/Cloud Engineer**

**Description:**

DevOps Engineer with expertise in GCP and AWS services, including compute, storage, and monitoring. Proficient in setting up CI/CD pipelines using Jenkins, Bitbucket, and Argo CD. Skilled in infrastructure automation with Terraform, CloudFormation, and Ansible. Experienced in container orchestration with Docker and Kubernetes (GKE). Adept at managing server configurations using Chef and Puppet, and log analysis with ELK stack. Hands-on experience in scripting with Python, Shell, and Perl for automation. Successfully managed software development lifecycle phases, release planning, and version control integrations, optimizing build and deployment processes for high availability and performance.

**Responsibilities:**

* Worked on google cloud platform (**GCP**) services like compute engine, cloud **load** **balancing**, **cloud** **storage**, **cloud** **SQL**, stack driver monitoring, and cloud deployment manager.
* Set up **GCP** **Firewall** rules to allow or deny traffic to and from the VM's instances based on specified configuration and used **GCP** cloud **CDN** (content delivery network) to deliver content from GCP cache locations, drastically improving user experience and latency.
* Worked on GKE Topology Diagram including masters, **slave**, **RBAC**, **helm**, **kubectl**, ingress controllers **GKE** Diagram including masters, slave, **RBAC**, helm, **kubectl**, **ingress** **controllers**
* Worked on Jenkins file with multiple stages like checkout a branch, building the application, testing, pushing the image into **GCR**, Deploying to **QA3**, Deploying to **QA9**, Acceptance testing, and finally Deploying to Prod
* Responsible for Setup and building AWS infrastructure using resources VPC, EC2, S3, RDS, Dynamo DB, IAM, EBS, Route53, SNS, SES, SQS, **CloudWatch**, CloudTrail, Security Group, Autoscaling, and RDS using CloudFormation templates.
* Hands-on experience using Argo CD to trigger fully automated builds from GitHub for the applications which often build releases. Utilized advantages of GITOPS in the build and release processes for automation using Bitbucket as Version Control
* Deploy and monitor scalable infrastructure on Amazon web services (**AWS**) and configuration management instances and Managed servers on the Amazon Web Services (**AWS**) platform using Chef configuration management tools and Created instances in AWS as well as migrated data to AWS from **EC2** Data Center
* Provisioned the highly available Instances using **Terraform** and cloud formation and wrote new **Python** scripts to support new functionality in Terraform.
* Designed fully automated server build management, monitoring, and deployment using Ansible playbooks. **Ansible** role created an ELK cluster for non-log purposes to search and analytics of product and pricing data
* Managed all phases of the software development life cycle (SDLC) by building customized Build and deployment Applications.
* Created plans and ensured Release management deliverables were met on time, risks were appropriately mitigated, and code implementation status was communicated regularly to Application Manager.
* Used **Ansible** for managing server application servers such as **Apache and MySQL.**
* Configured and set up activities of both environmental and functional and collected data for preparing monthly metrics of the migrated codes, monthly internal and yearly external audits.
* Developing automated processes for builds and deployments using **Jenkins, Maven, Sonar Type Nexus, Shell, Python,** and **Java.**
* Created **Puppet manifests** and modules to automate system operations. Created monitors, alarms, and notifications for EC2 hosts using **Cloud Watch.**
* Used **Puppet** for Monitoring system and automation.
* Design and document **CI/CD** tools configuration management.
* Involved in designing/automating proof of concept projects and version control tool migrations.
* Improving Continuous Integration workflow using Virtual Environments like **Docker** and automating the Build process using **Docker** and similar tools.
* Implemented a mirror for the **SVN repository,** which enables users to use both **SVN and GIT.**
* Worked on **Shell/Bash/Perl** scripts for automation purposes.
* Created and owned, Build, and Continuous Integration environment with **Maven and Jenkins.**
* Installed and Configured the **Nexus repository** manager for sharing the artifacts within the company.
* Configured Nightly Builds using SCM polling option to ensure an updated version of the artifactory generated in **NEXUS.**
* Administered and integrated the Version Control Systems like **Subversion** and **GIT with Jenkins.**
* Involved in migration activities using **Red Hat LVM, and Solaris LVM,** and in Research of the project application architecture to support/resolve build, compile, and test issues/problems.
* Involved in release planning and executed the release build request from the project team through **JIRA Ticket**

**Environment:** DevOps, AWS, EC2, S3, Cloud Watch, GCP, Shell, Jira, Terraform, K8’s, Puppet, Docker, GIT, Maven, JIRA, Tomcat, UNIX, Jenkins, Chef, Maven, Red Hat LVM, Solaris LVM, ANT, Apache, SVN, and Windows.

**Client: PB Technologies, India Aug 2015 – Sep 2017**

**Role: Cloud Engineer**

**Description:**

With extensive experience in administering SVN and GitHub Enterprise, I provided IaaS support on Microsoft Azure, deploying and managing virtual machines, storage, and networks. I worked on CI/CD pipelines for PostgreSQL and MySQL, set up monitoring in GCP, and managed Docker orchestration with Kubernetes. Additionally, I performed clinical study audits, created AWS scripts, and managed Linux servers, ensuring high-quality software builds and deployments.

**Responsibilities:**

* Administered and supported **SVN** and **GitHub** Enterprise version control tools.
* Worked on **Microsoft** **Azure** (Public) Cloud to provide IaaS support to clients. Create **Virtual** **Machines** through Power **Shell** **Script** and **Azure** **Portal**.
* Manage and Create **Storage** **Account** and Affinity Group in **Azure** Portal. Capture an image of a Virtual Machine. Attaching a Disk to a Virtual Machine. Manage and create **Virtual** **Networks** and End Points in Azure Portal.
* Deploying **VMs**, **Storage**, **Networks**, and Affinity Group through PowerShell Script. Creating Storage Pool and Stripping of Disk for Azure Virtual Machines. Backup, Configure, and Restore Azure Virtual Machine using Azure Backup.
* Taking Backup to cloud Storage Account using Cloudberry **Cloud** **Storage** Tools. Configure **Site**-**to-Site** **VPN** Connectivity.
* Configure Window Failover Cluster by creating Quorum for File sharing in Azure Cloud. Convert existing Virtual Machine from Standard to Premium Storage Account. Patching and Validating of Virtual Machine in Azure.
* **Monitor** Azure Infrastructure through System Center Operation Manager (SCOM). Coordinating with Microsoft for increasing subscription limits like- Core limit and Cloud Services.
* Assisted with developing the Development Priority List, working with the Product Management group to solve issues identified in the Problem Management process.
* Provide performance management, disaster recovery, monitoring, and access control solutions.
* Work with/support business users to understand issues, develop root cause analyses, and collaborate on enhancements/fixes with the team.
* Worked on CI/CD for **PostgreSQL** and **MySQL** Applications.
* Setup Alerting and monitoring using **Stack** **driver** in GCP.
* Performed requested/contracted **GCP/GLP** audits of clinical studies and related documentation (CSRs, protocols, TMFs, subject files, central labs, etc.) across various therapeutic areas (oncology, endocrinology, fertility, cardiovascular, etc.) for clinical studies conducted in Phases I-IV as contracted by the client/sponsor organization.
* Managing **Docker** orchestration and **Docker** containerization using Kubernetes.
* Utilized **Docker** for the runtime environment of the CI/CD system to build and test deploy.
* Utilized existing Deal Model in **Python** to inherit and create an object data structure for regulatory reporting.
* Created scripts for system administration and AWS using **BASH and Python.**
* Installation, configuration, and maintenance of different **Linux** Distribution servers like **Debian**, **RHEL,** and CentOS Servers at multiple Data Centers.
* Helped to establish the physical environments to perform effectively the software builds and deployments with an emphasis on quality, integrity, and security.
* All IT faults were diagnosed, investigated, and addressed; error budgets for technical and engineering teams were monitored.

**Environment**: Azure GCP, Git, Bamboo, Docker, Groovy, Chef, Puppet, Apache-Tomcat, RHEL, SonarQube, Nexus.

**Client: Genrapid Tech, India June 2014 - July 2015**

**Role: Build and Release Engineer**

**Description:**

DevOps Engineer with expertise in cloud infrastructure management, focusing on Azure and GCP. Experienced in VM deployment, storage management, and network configuration using PowerShell and Azure Portal. Proficient in CI/CD pipelines for PostgreSQL and MySQL, and container orchestration with Docker and Kubernetes. Skilled in version control administration with SVN and GitHub Enterprise. Adept at scripting with Python and Bash for automation and system administration across Linux distributions. Managed disaster recovery, performance monitoring, and access control solutions. Provided support for business users, performed clinical study audits, and ensured quality and security in software builds and deployments.

**Responsibilities:**

* Designed and implemented fully automated server build, management, monitoring, and deployment solutions spanning multiple platforms, tools, and technologies, including **Jenkins** Slave Nodes/Agent, Chef, Android, and VMware.
* Extensively worked on **Jenkins** by configuring and maintaining continuous integration and end-to-end automation for all builds and deployments.
* Supported **AWS** cloud services such as EC2, auto-scaling, and VPC to build secure, highly scalable, and flexible systems that handled expected and unexpected load bursts.
* Experience in release automation for **Java** technology to achieve continuous integration and continuous delivery (CI /CD).
* Automated Clear Case-based release management process, including monitoring changes between releases.
* Solid understanding of **Linux** shell scripting, **UNIX, Windows**, build and release engineering principles.
* Installed, configured, and administered **Jenkins** (CI Tool) on **Linux**/Windows machines, adding/updating plugins such as Build Pipeline, **SVN, SSH Agent, GIT Hub, Maven, ANT, SonarQube, etc.**
* Experience with build tools **ANT** and **Maven** for writing (build.xml and pom.xml) files, respectively.
* Build Automation and Build Pipeline development using **Jenkins** and **Maven**.
* Designed and created an automated Build and Deployment process using **Jenkins's** build pipeline.
* Used SVN as source code repository and experienced in cloning, pulling, and pushing the code.
* Provided periodic feedback on status and scheduling issues to the management.
* Used Subversion (SVN) in branching, merging, and tagging the Source Code and managed the source code repositories.
* Setting up private networks and sub-networks using a virtual private cloud (VPC) and creating security groups to associate with the networks.
* Setting up scalability for application servers using command line interface for Setting up and administering DNS system in **AWS** using Route53 Managing users and groups using the Amazon identity and access management (IAM).
* Managed all the bugs and changes in a production environment using the **Jira** tracking tool.

**Environment**: Linux, GIT, Jira, AWS, EC2, S3, VPC, CI/CD Management, Windows, CentOS, Shell script, SVN, SCM, Jenkins, Ant, Maven, Chef, RDS/DNS.