**Premkumar Debbadi**

**Phone:** 9256033708

**Email:** Prem.Debbadi@outlook.com

**Linked In:** https://www.linkedin.com/in/premkumar-debbadi-8b482323a/

**Professional Summary:**

Skilled IT Professional with **12+** years of experience in **DevOps/Cloud Engineer** operations process involving **Automation**, **Build/Release Management**, Software Configuration Management, Service, Incident, and Change Management. Experience in Software Development Life Cycle (**SDLC**) & **Agile Methodologies**.

* Hands-on experience and good knowledge on **AWS** Platform Services like **EC2, S3, ELB, IAM, Route53, VPC, Auto Scaling, API Gateways, Lambda, Elastic Beanstalk, CloudFront, CloudTrail, CloudFormation, SNS, SQS** etc.,
* Experienced with **IAM** requirements for cloud - based services, including strategies, trust models, and open standards, encryption, etc. used to improve manageability, access governance, provisioning, information security, and privacy.
* Implemented the serverless architecture **(SAM)** using **API Gateway, Lambda**, **Dynamo DB** and triggered the **AWS Lambda** code from Amazon **S3 buckets**.
* **Elastic APM** helps in identifying and diagnosing performance bottlenecks, slow transactions, and resource-intensive components.
* Monitored individual transactions, API calls, and requests to understand response times, error rates, and more.
* Experience in building automated CI and CD pipeline with AWS CodePipeline, Jenkins and AWS Code Deploy for the deployment of serverless applications built using AWS Lambda, API Gateway, and other serverless services. It can manage the packaging, testing, and deployment of Lambda functions and API configurations.
* Experienced in writing **Terraform** modules for automation, Creation of **VPC-VPN** from data center to Production environment and Cross account **VPC Peering**.
* Redesigned the Views in snowflake to increase the performance.
* Unit tested the data between Redshift and Snowflake.
* Developed data warehouse model in snowflake for over 100 datasets using where Scape.
* Creating Reports in Looker based on Snowflake Connections
* Experienced in deploying multiple versions of **Java** with **Tomcat, WebLogic, IBM WebSphere, JBoss** servers on **AWS** cloud platform.
* Hands-on Experience in **Microsoft Azure** platform services like Virtual Machines **(VM), App Service, Virtual Machine scales sets, Logic Apps, Service Fabrics, container services, Batch, Cloud Services, Queue Storage, File Storage, Disk Storage, MS Build, MS Deploy** etc.,
* Experience in dealing with Windows Azure IaaS - Virtual Networks, Virtual Machines, Cloud Services, Resource Groups, Express Route, VPN, Load Balancing, Application Gateways, Auto-Scaling, and Traffic Manager.
* Major focus on Configuration, SCM, Build/Release Management, Infrastructure as a code (IAC) and as
* Azure DevOps operations Production and cross platform environments.
* Experience in writing Infrastructure as a code(IaC) in Terraform, Azure resource management, AWS
* Cloud formation. Created reusable Terraform modules in both Azure and AWS cloud environments.
* Responsible for installing **Jenkins** masterandslavenodes**, Bamboo**. Configured **Git with Jenkins** and scheduled jobs using **POLL SCM** option.
* Experienced in using **Jenkins** as continuous integration tool for creating new jobs, managing required plugins, configuring the jobs using source code management, build trigger, build system, post build actions, scheduling automatic builds and notifying the build report.
* Experienced and developed build and deployment scripts using **ANT** and **MAVEN** as build tools in Jenkins to move from one environment to other environments.
* Good knowledge in managing **Nexus repositories** for the **maven artifacts** and dependencies.
* Wrote several **Ansible playbooks** for the automation that was defined through tasks using **YAML** format and run **Ansible Scripts** to provision Development servers.
* Experience in Installing and using **Chef** Server Enterprise, workstation and bootstrapped the nodes using **knife** and automated by writing ruby scripts in **Chef Recipes** and **Cookbooks** with test-kitchen/chef spec.
* Experienced in configuration and automation with Ruby Script using Chef with Jenkins and **Docker**.
* Experienced in installing **Docker** using **Docker toolbox** Provisioning Environmental solutions. Source code management is performed using **GIT** from master repository and knowledge on **Container management** using **Docker** in **creating images**.
* Provided consistent environment using **Kubernetes** for deployment scaling and **load balancing** to the application from development through production, easing the **code development** and **deployment pipeline** by implementing **Docker containerization**.
* Orchestrated Rancher upgrades, ensuring compatibility with the latest Kubernetes versions and taking advantage of new features.
* Planned and executed cluster upgrades while minimizing downtime and maintaining application availability.
* Hands on experience with build tools like Hudson/Jenkins, Argo CD, Concourse, VMware v center GitHub, ANT, Maven and other products like Bamboo, JIRA, BitBucket for building of deployable artifacts.
* Hands on experience using Argo CD to trigger fully automated builds from GitHub for the applications which needs often build releases. Utilized advantages of GITOPS in the build and release processes for automation using Bitbucket as Version Control.
* Installed, Configured, Managed Monitoring Tools such as **Splunk, Nagio**s for Network, Resource Monitoring and Log Trace Monitoring.
* Responsible for User, Group creation, monitoring and maintaining log for system status, health status using **Linux commands and Splunk system monitor**.
* Developed **Bash, Shell, Perl, Python scripts** with focus on DevOps tools and **CI/CD architecture**.
* Expertise in Querying **RDBMS** such as **Oracle** and **MYSQL** by using **SQL** for Data integrity, Installation, Configuration and Management of **NoSQL** tools such as **SQL Server**, **DB2**, **PostgreSQL**, **MongoDB,** and **Cassandra**.

**Technical Skills:**

|  |  |
| --- | --- |
| Cloud Platform | Amazon Web Service, Microsoft Azure, Google Cloud Platform. |
| SCM/Version Control Tools | GIT, GitHub, SVN, Bitbucket. |
| Continuous Integration Tools | Jenkins, TeamCity, Bamboo, GitHub actions, GitLab |
| Configuration Management Tools | CHEF, Ansible, Puppet. |
| Containerization Tools | Docker, Kubernetes, OpenShift, Rancher |
| Build Tools | Maven, ANT, Gradle |
| Scripts/ Languages | HTML, Bash, Shell Scripting, Ruby, YAML, Python, Java, PL/SQL, Node.js, Java Script |
| Databases | Oracle, MongoDB, SQL Server, MS SQL, NOSQL,  Cosmos DB (Document DB), PostgreSQL, Cassandra. |
| Monitoring Tools | Splunk, CloudWatch, Prometheus, Grafana, Datadog |
| Operating System | UNIX, Linux (Ubuntu, RHEL, Centos), Windows. |
| Webservers | Apache Tomcat, WebLogic, IBM WebSphere, JBoss. |

**Work Experience:**

**Work Experience:**

**Client: Morgan Stanley, Alpharetta, GA Dec 2022 – Present**

**Role: Site Reliability Engineer**

* Involved in designing and deploying multiple applications utilizing all AWS services **EC2, AWS Lambda, Route 53, VPC, Security Groups, ELB, IAM, S3, Elastic Block Store, DynamoDB, Amazon RDS, Cloud Formation, Code Deploy, CloudWatch, and SNS.**
* Implemented **Security groups** for inbound/outbound access, **network ACLs** for controlling traffic through subnets, **Internet Gateways, NAT instances** and **Route Tables** to direct the network traffic and to ensure secure zones for organizations in AWS public cloud.
* Created **IAM** policies for administration in **AWS** to assist teams across local geographical and remote areas.
* Implemented various **EC2** services like Launching instances, attaching volumes and snapshots, Configuring security group and Monitoring.
* Configured **Auto-Scaling Group (AGS)** and **ELB** as per the business need to handle application traffic load during business hours.
* Responsible 24/7 on-call PagerDuty
* Created dashboards using Splunk , alerts creation and finding the logs to resolve the issue .
* Designed and converted existing **AWS** infrastructure to serverless architecture deployed via terraform and used those templates for modules to support other teams for flexibility to deploy their application into **AWS cloud**.
* Provided highly durable and available data by using **S3** data store, **versioning, lifecycle policies** and enabled **cross region replication** for disaster recovery.
* Written **Docker files** for Java projects to a single deployment method using **ECR** Docker Containers.
* Developing a deployment management system for Docker Containers in **AWS Elastic Container Service (ECS).**
* Worked on various teams as leading team member; successfully handling responsibilities such as setting up environments, knowledge transfer, code review, compliance to best practices & standards and deployment of applications.
* Experience of managing and coordinating the Requirement Gathering, Development and Deployment activities with offshore and onsite teams. Ability to learn and apply new technologies quickly along with strong leadership and problem-solving skills.
* Set up monitoring and logging solutions to track the performance and health of applications and infrastructure. This includes monitoring resource utilization, response times, and detecting potential issues proactively.
* Responsible for designing, implementing, and managing the cloud infrastructure. This includes choosing the appropriate cloud service provider (e.g., AWS, Azure, Google Cloud), setting up virtual machines, configuring networking, and managing storage solutions.
* Ensure that the team maintains proper documentation for infrastructure, processes, and best practices. Encourage knowledge sharing within the team and the broader organization.
* Lead incident response and resolution efforts in collaboration with development and operations teams. Establish procedures for identifying, diagnosing, and resolving incidents in a timely manner.
* Implemented **Amazon RDS** multi-AZ for automatic failover and high availability at the database tier and optimized the configuration of amazon **redshift clusters**, data distribution, and data processing.
* Utilized **Cloud Watch** to monitor resources such as **EC2**, **Amazon RDS DB services**, **Dynamo DB tables**, **EBS** volumes, to set alarms for notifications or automated actions, and to monitor logs for a better understanding and operation of the system.
* Created alarms and trigger points in **Cloud Watch** based on thresholds and monitored the **server performance, CPU Utilization, disk usage**.
* Orchestrated Rancher upgrades, ensuring compatibility with the latest Kubernetes versions and taking advantage of new features.
* Planned and executed cluster upgrades while minimizing downtime and maintaining application availability.
* Designed, configured, and managed end-to-end CI/CD pipelines using Azure Pipelines.
* Created and maintained Azure Resource Manager (ARM) templates for infrastructure provisioning and management.
* Orchestrated containerization of applications using Docker and deployment to Azure Kubernetes Service (AKS).
* Monitored application performance and implemented logging and alerting using Azure Monitor and Application Insights.
* Configured and managed HashiCorp Vault for secure storage and management of sensitive data, including API keys, passwords, and certificates.
* Designed and maintained Vault policies and access controls, ensuring appropriate permissions for different teams and roles.
* Integrated HashiCorp Vault with application workflows, enabling seamless retrieval of secrets at runtime while adhering to security best practices.
* Developed infrastructure as code (IaC) templates to provision HashiCorp Vault instances using tools like Terraform, enabling consistent deployments across environments.
* Worked on real-time insights into the performance of applications, allowing you to identify and address issues quickly and tracking and diagnosing errors in your application code.
* Responsible for receiving and processing APM data from different agents.
* Distributed tracing helps in tracing the root cause of errors and issues across different services.
* Responsible for creating and configuring the AWS CodePipeline.Integrate automated testing processes into the Codepipeline to validate code changes.
* Monitor the pipeline for any issues, such as failed builds, deployments, or performance problems.
* Define build specifications using AWS CodeBuild to compile, test, and package the code.Responsible for writing and committing code changes to the source code repository such as GitHub and AWS CodeCommit.
* Configured AWS CodePipeline to automate the testing and validation of feature branches when a pull requests or merge requests arised for merged into the main codebase.
* Utilized the AWS CodePipeline for the rollback and recovery processes in case of failed deployments by leveraging the Infrastructure as Code and automate the process of reverting to a previous version of the application.
* Responsible to designing and deploying new ELK clusters Elasticsearch, logstash, Kibana.Used Elasticsearch for powering not only Search but using ELK stack for logging and monitoring our systems end to end Using Beats.
* Automated Continuous Integration builds, nightly builds, deployments, and unit tests across multiple different environments (Development, QA, Training, Production) each constituting different types of servers (DB, App, Web) and different number of servers using AWS CLI, AWS Code Deploy, Git.
* As a member of, redefined processes and **Release Engineering group** implemented tools for software builds, patch creation, source control, and release tracking and reporting.
* Resolved Merge Conflicts, configured triggers and queued new builds within the release pipeline and monitored & tracked the deployments.
* Integrated various Version control tools, build tools, **Nexus**, and deployment methodologies (scripting) into Jenkins to create an end-to-end orchestration build cycles.
* Deployed dashboards in Dynatrace for both operations and various lines of business.
* Use Dynatrace to perform root cause analysis and quickly drill down to correct error fault path and hot spots.
* Worked with Dynatrace Guardian to obtain, debug, and update custom monitors and plug-ins.
* Orchestration improvements to Dynatrace deployment to reduce upgrade time.
* Maintained product release process, including generating and delivering release packages, generated various metrics for tracking issues against releases and the means of tracking compatibility among products.
* Supported engineering plans and schedules by providing Configuration Management / Release Engineering services to build, deploy, develop scripts, oversee branch, and build automated tools as necessary to offer services to engineering team.
* Extensive experience in developing applications Single Page (SPAs) using various JavaScript frameworks like Angular 2.0, React.JS and Backend(Server) like Node.js, Express.
* Implemented generating the verification hash code using crypto package on Node.JS.
* A web application for connecting care providers and members. This project includes server application with data fetching requirement. We
* have exposed the services as Rest Web Services. We developed backend system using NodeJS.
* Designed dynamic client-side JavaScript codes to simulate process for web application, page navigation and form validation.
* Developed data insertion forms and validated them using JavaScript.
* Attended daily Scrum meetings, kept up to date on best practices for JavaScript frameworks and techniques.
* Worked closely with QA Teams, Business Teams, DBA team and Product Operations teams to identify QA and UAT cycles release schedule to non-production and production environments.

**Environment:** AWS (EC2, EBS, S3, VPC, Code Deploy, Code Commit, Elastic search, API Gateway, Route 53 (DNS), ECS, Cloud Formation, RDS, Dynamo DB, CloudWatch, SNS, SES, SQS, IAM, RedShift)

**Client: Memorial Sloan Kettering Cancer center May 2022 – Dec 2022**

**Role: SRE Solution Architect**

* Designed and implemented end-to-end Site Reliability Engineering (SRE) strategies, driving improved system reliability and performance.
* Collaborated with cross-functional teams to define and implement best practices for application scalability, availability, and resilience.
* Architected solutions for high availability, disaster recovery, and fault tolerance using cloud-native technologies and practices.
* Led the design and implementation of monitoring and observability solutions, utilizing tools like Prometheus, Grafana, and Azure Monitor.
* Championed the adoption of Infrastructure as Code (IaC) principles, leading to faster provisioning and consistent configuration using tools like Terraform and Ansible.
* Designed and implemented containerization strategies using Kubernetes, enabling scalable and orchestrated application deployments.
* Provided highly durable and available data by using **S3** data store, **versioning, lifecycle policies** and enabled **cross region replication** for disaster recovery.
* Redesigned the Views in snowflake to increase the performance.
* Unit tested the data between Redshift and Snowflake.
* Developed data warehouse model in snowflake for over 100 datasets using whereScape.
* Creating Reports in Looker based on Snowflake Connections
* Written **Docker files** for Java projects to a single deployment method using **ECR** Docker Containers.
* Developing a deployment management system for Docker Containers in **AWS Elastic Container Service (ECS).**
* Implemented **Amazon RDS** multi-AZ for automatic failover and high availability at the database tier and optimized the configuration of amazon **redshift clusters**, data distribution, and data processing.
* Utilized **Cloud Watch** to monitor resources such as **EC2**, **Amazon RDS DB services**, **Dynamo DB tables**, **EBS** volumes, to set alarms for notifications or automated actions, and to monitor logs for a better understanding and operation of the system.
* Created alarms and trigger points in **Cloud Watch** based on thresholds and monitored the **server performance, CPU Utilization, disk usage**.
* Written **Docker files** for Java projects to a single deployment method using **ECR** Docker Containers.
* Developing a deployment management system for Docker Containers in **AWS Elastic Container Service (ECS).**
* Implemented **Amazon RDS** multi-AZ for automatic failover and high availability at the database tier and optimized the configuration of amazon **redshift clusters**, data distribution, and data processing.
* Utilized **Cloud Watch** to monitor resources such as **EC2**, **Amazon RDS DB services**, **Dynamo DB tables**, **EBS** volumes, to set alarms for notifications or automated actions, and to monitor logs for a better understanding and operation of the system.
* Created alarms and trigger points in **Cloud Watch** based on thresholds and monitored the **server performance, CPU Utilization, disk usage**.
* Worked on setting up Rancher orchestrator, to manage Kubernetes everywhere it runs, worked with
* Rancher CLI.
* Developed microservice on boarding tools leveraging Python and Jenkins allowing for easy creation and
* maintenance of build jobs and Rancher/Kubernetes deploy and services.
* Configured memory usage and network protocols on multiple Amazon EC2 instances for Rancher
* Master and Worker nodes.
* Created AWS CodePipeline pipelines for containerized applications using Docker and Amazon ECS, automating image building, testing, and deployment to multiple clusters.
* Implemented multi-environment deployment pipelines, streamlining the process of promoting code changes from development to production stages while maintaining rigorous testing and validation.
* Integrated AWS CloudFormation into the pipeline to automate the provisioning and updating of infrastructure resources, ensuring consistency across environments.
* Automated Continuous Integration builds, nightly builds, deployments, and unit tests across multiple different environments (Development, QA, Training, Production) each constituting different types of servers (DB, App, Web) and different number of servers using AWS CLI, AWS Code Deploy, Git.
* Configure Continuous Integration from source control, setting up build definition within Visual Studio
* Team Services (VSTS) and configure continuous delivery to automate the deployment of ASP.NET MVC
* applications to Azure web apps and managed Azure Active Directory, Office 365 and applied upgrades
* on a regular basis
* Written Templates for Azure Infrastructure as code using Terraform to build staging and production
* environments. Integrated Azure Log Analytics with Azure VMs for monitoring the log files, store them
* and track metrics and used Terraform as a tool, Managed different infrastructure resources Cloud,
* VMware, and Docker containers.
* Maintained product release process, including generating and delivering release packages, generated various metrics for tracking issues against releases and the means of tracking compatibility among products.
* Supported engineering plans and schedules by providing Configuration Management / Release Engineering services to build, deploy, develop scripts, oversee branch, and build automated tools as necessary to offer services to engineering team.
* Worked closely with QA Teams, Business Teams, DBA team and Product Operations teams to identify QA and UAT cycles release schedule to non-production and production environments.

**Environment:** AWS (EC2, EBS, S3, VPC, Code Deploy, Code Commit, Elastic search, API Gateway, Route 53 (DNS), ECS, Cloud Formation, RDS, Dynamo DB, CloudWatch, SNS, SES, SQS, IAM, RedShift)

**Client: PG&E, San Francisco, CA Sep 2021 – May 2022**

**Role: DevSecOps/Release Engineer**

* Involved in designing and deploying multiple applications utilizing all AWS services **EC2, AWS Lambda, Route 53, VPC, Security Groups, ELB, IAM, S3, Elastic Block Store, DynamoDB, Amazon RDS, Cloud Formation, Code Deploy, CloudWatch, and SNS.**
* Implemented **Security groups** for inbound/outbound access, **network ACLs** for controlling traffic through subnets, **Internet Gateways, NAT instances** and **Route Tables** to direct the network traffic and to ensure secure zones for organizations in AWS public cloud.
* Created **IAM** policies for administration in **AWS** to assist teams across local geographical and remote areas.
* Implemented various **EC2** services like Launching instances, attaching volumes and snapshots, Configuring security group and Monitoring.
* Configured **Auto-Scaling Group (AGS)** and **ELB** as per the business need to handle application traffic load during business hours.
* Designed and converted existing **AWS** infrastructure to serverless architecture deployed via terraform and used those templates for modules to support other teams for flexibility to deploy their application into **AWS cloud**.
* Provided highly durable and available data by using **S3** data store, **versioning, lifecycle policies** and enabled **cross region replication** for disaster recovery.
* Written **Docker files** for Java projects to a single deployment method using **ECR** Docker Containers.
* Developing a deployment management system for Docker Containers in **AWS Elastic Container Service (ECS).**
* Implemented **Amazon RDS** multi-AZ for automatic failover and high availability at the database tier and optimized the configuration of amazon **redshift clusters**, data distribution, and data processing.
* Utilized **Cloud Watch** to monitor resources such as **EC2**, **Amazon RDS DB services**, **Dynamo DB tables**, **EBS** volumes, to set alarms for notifications or automated actions, and to monitor logs for a better understanding and operation of the system.
* Created alarms and trigger points in **Cloud Watch** based on thresholds and monitored the **server performance, CPU Utilization, disk usage**. And attached Grafana for Dashboard view for the metrics.
* Written **terraform** reusable modules to provision cloud resources, used **s3** **buckets** as a backend file storage and implemented **dynamo** **dB** for state locking.
* Worked on various Azure services like Compute (Web Roles, Worker Roles), **Azure Websites, Caching, SQL Azure, NoSQL, Storage, Network services, Azure Active Directory, API Management, Scheduling, Auto Scaling,** and **PowerShell Automation.**
* Provided high availability for **IaaS VMs** and **PaaS role instances** for access from other services in the VNet with **Azure internal Load Balancer.**
* Used **Azure** **terraform** to deploy the infrastructure necessary to create development, test and production environments for software development process.
* Configured and installed **Docker** using **Docker toolbox**, creation of custom docker container images, tagging and pushing the images, removing images, and managing docker volumes.
* Utilized **Docker** containers to create images of the applications and dynamically provision slaves to **Jenkins CI/CD** pipelines.
* Configured **Jenkins** to setting up **CICD** for new branches, build automation, plugin management and securing **Jenkins** and setting up **master/slave** configurations.
* Created and configured Kubernetes clusters using Rancher, tailoring them to match application requirements and workload demands.
* Set up multi-cluster management to efficiently oversee multiple Kubernetes clusters from a single Rancher interface.
* Customized Rancher configurations to align with specific organizational needs, optimizing performance and security.
* Created detailed documentation and guidelines for Rancher system configuration, contributing to an internal knowledge base for team members.
* Presented Rancher system configuration best practices in team meetings and training sessions.
* Creating scripts in **DSL Groovy** which integrate with Jenkins for Automation to create seed jobs.
* Integrated **Jenkins** with **GitHub** for continuous integration and deployment of the code by enabling **Git webhooks**, the build jobs are automatically created once the changes are made to the code by dev team.
* Defined dependencies and plugins in **Maven** pom.xml for various activities and integrated **Maven** with **Git** to manage and deploy project related tags.
* Maintenance of source code in **Git** and branching, creating labels, merging of codes on git for testing and release.
* Developed customized applications configurations in **Splunk** to parse, index multiple types of log format across all application environments.
* Experience in building and architecting multiple Data pipelines, end to end ETL and ELT process for Data ingestion and transformation in **GCP** and coordinate task among the team.
* Write a Python program to maintain raw file archival in GCS bucket.
* Worked on build scripts and automated solutions using various scripting languages like **Python**, **Groovy, BOTO3**, **Shell/Bash** and **YAML.**
* Debugging the failure issues by capturing the array and register dumps using Python scripts, traces and performing several experiments by interacting with the design team.
* Developed **Bash, Shell, Perl, Python scripts** with focus on DevOps tools and **CI/CD architecture**.
* Provisioned an EKS cluster for different deployment groups (Lower environments: Development and QA. Also, in the Production environment) with 3 node groups deployed in 3 availability zones in two different regions.
* Experienced in writing **Terraform** modules for automation, Creation of **VPC-VPN** from data center to Production environment and Cross account **VPC Peering**.
* Managed **Kubernetes** **charts** using **Helm**. Created reproducible builds of the Kubernetes applications, managed Kubernetes manifest files and Managed releases of **Helm** **packages**.

**Environment:** AWS (EC2, EBS, S3, VPC, Code Deploy, Code Commit, Elastic search, API Gateway, Route 53 (DNS), ECS, Cloud Formation, RDS, Dynamo DB, CloudWatch, SNS, SES, SQS, IAM, RedShift), GCP, Docker, ELK, Jenkins, Groovy, GIT, Python.

**Client: Novartis Jul 2020 – Aug 2021**

**Role:** **Site Reliability Engineer**

Its multinational pharmaceutical corporation based in Basel, its mainly focuses on the manufacturing and marketing of drugs pharmaceutical products and formulations for consumer healthcare and animal healthcare. The purpose is to reimagine medicine to improve and extend people's lives. Uses innovative science and technology to address some of society's most challenging healthcare issues.

* Managed multiple AWS accounts with multiple VPC's for both production and non-production where primary objectives included automation, build out, integration and cost control.
* Worked Intensely with AWS and its infrastructure, including EC2, ECS, Elastic Cache, Elasticsearch, RDS, VPC implementation, IAM, KMS, CloudTrail, CloudWatch, S3, CloudFront, AWS CLI scripting, ELB, Route 53, Lambda, DynamoDB, Elastic Beanstalk, SQS, SNS and Security group management.
* Templated AWS infrastructure as a code using **Terraform** to build staging at different production environments.
* Created and configured **S3** **buckets** versioning and managing policies for S3 buckets and Utilized S3 buckets for storage and backup on AWS.
* Monitor Service Level Indicators (SLI) for application performance issues and analyze poor performing SQL queries with Dynatrace and coordinate with DBAs and Dev teams to meet Service-Level Objectives (SLO).
* Monitored Service Level Indicators (**SLI**) for application performance issues and analyze poor performing SQL queries with Dynatrace and coordinate with DBAs and Dev teams to meet Service-Level Objectives (**SLO**).
* Involved in source control management with **Code Commit** and **GitLab** Enterprise level repositories. Regular activities include configure user’s access levels, monitor logs, identifying merge conflicts and managing master repository.
* Maintained and administered source code in GitLab. Created Branches, Labels and performed Merges in Stash.
* Integrated GIT into Jenkins to automate the code check-out process. Used **Jenkins** for automating Builds and Automating Deployments.
* Designed and created multiple deployment strategies using Continuous Integration and Continuous Development (CI/CD) Pipelines and configuration management tools with remote execution. Ensured zero downtime using **blue**/**green** **deployment** **strategy** and shortened deployment cycles through Jenkins automation.
* Redesigned the Views in snowflake to increase the performance.
* Unit tested the data between Redshift and Snowflake.
* Developed data warehouse model in snowflake for over 100 datasets using whereScape.
* Creating Reports in Looker based on Snowflake Connections
* Managed the **Maven** Repository to create JAR, WAR, EAR packages using **Nexus** and used the same to share the snapshots and releases of internal projects.
* Using **Ansible** as Configuration management tool, to automate repetitive tasks, quickly deploys critical applications, and used **Ansible** **Tower** to monitor automation activity and manage cloud infrastructure.
* Written Ansible playbook & modules for Ansible customizations and to configure, deploy and maintain software components of the existing/new infrastructure.
* Working with Docker, pulling images from AWS ECS, running containers based on an image, creating **Docker** **file** to manage customized containers, exposing a container with port redirect, container volume management, docker inspect, docker commit to capture the file changes/settings into a new image and pushing the image to ECS.
* Created **Docker** **images** and worked on **Docker** container snapshots, removing images, and managing docker volumes and experienced with **Docker** **container** **service**.
* Used **Kubernetes** to deploy scale, load balance, scale and manage docker containers with multiple versions.
* Setting up the complete **Kubernetes** Dev, QA, Prod Environments from scratch to deployed java and python applications into clusters and managed using **helm** **charts.**
* Configured **CloudWatch** for real-time Monitoring logs, Metrics, Notifications, an Alerting system for deployed applications and Health checks in AWS Infrastructure.
* Implemented and maintained the monitoring and alerting of Non-production and Production servers/storage using AWS CloudWatch.
* Involved in using **Datadog** to monitor cloud-scale applications, bringing together data from servers, databases, tools, and services to present a unified view of an entire stack.
* Using JIRA to track all issues related to the software development lifecycle and integration of JIRA with repositories to track all code changes.

**Environment:** AWS (EC2, VPC, ELB, S3, Amazon RDS, DynamoDB, EBS, ECS, SQL, CloudWatch, AWS Auto Scaling, AWS CLI, Elastic Beanstalk and Route53), GIT, Maven, Jenkins, Docker, Kubernetes, Ansible, Jira, Datadog, Splunk, Terraform, Bash Shell script, SQS, SNS, Lambda, Python.

**Client: Novartis Dec 2019 – Jul 2020**

**Role:** **Sr. Cloud/Devops Engineer**

* Automated Continuous Integration builds, nightly builds, deployments, and unit tests across multiple different
* environments (Development, QA, Training, Production) each constituting different types of servers (DB, App, Web) and different number of servers using AWS CLI, AWS Code Deploy, Git.
* As a member of Release Engineering group, redefined processes and implemented tools for software builds, patch creation, source control, and release tracking and reporting.
* Worked in all areas of Jenkins setting up CI for new branches, build automation, plugin management and securing Jenkins and setting up master/slave configurations.
* Resolved Merge Conflicts, configured triggers and queued new builds within the release pipeline and monitored & tracked the deployments.
* Integrated various Version control tools, build tools, Nexus, and deployment methodologies (scripting) into
* Jenkins to create an end-to-end orchestration build cycles.
* Troubleshooted build issues in Jenkins, performance and generating metrics on master's performance along with jobs usage.
* Maintained product release process, including generating and delivering release packages, generated various metrics for tracking issues against releases and the means of tracking compatibility among products.
* Supported engineering plans and schedules by providing Configuration Management / Release Engineering
* services to build, deploy, develop scripts, oversee branch, and merge strategies, and build automated tools as necessary to offer services to engineering team.
* Installed and used Chef Server Enterprise on premise, workstation and bootstrapped the nodes using knife and automated by writing scripts in Chef Recipes and Cookbooks with test-kitchen/chef spec.
* Managed the configurations of nodes using Chef, configured Jenkins builds for Continuous Integration and Delivery for automated deployments.
* Worked on Docker container snapshots, attaching to a running container, removing images, managing Directory structures, and managing containers. Built Docker files to create containers using Docker engine and Docker Machine environments to deploy the microservices oriented environments for scalable applications.
* Worked on Docker and deploying and maintaining Micro services in Development & QA environments. Creating the containers and maintained the clusters in docker.
* Configured elastic kubernetes clusters and deployed application into kubernetes, setup autoscaling with HPA and CA (cluster autoscaler).
* Extensive experience in developing applications Single Page (SPAs) using various JavaScript frameworks like Angular 2.0, React.JS and Backend(Server) like Node.js, Express.
* A web application for connecting care providers and members. This project includes server application with data fetching requirement. We
* have exposed the services as Rest Web Services. We developed backend system using NodeJS.
* Designed dynamic client-side JavaScript codes to simulate process for web application, page navigation and form validation.
* Developed data insertion forms and validated them using JavaScript
* Attended daily Scrum meetings, kept up-to-date on best practices for JavaScript frameworks and techniques.
* Provisioned **Elastic Kubernetes clusters** to deploy scale, load balance, scale and manage docker containers with multiple names spaced versions.
* Configured CloudWatch to monitor resources such as **EC2**, CPU memory, Amazon RDS DB services, Dynamo DB tables, Elastic Block Store (**EBS**) volumes to set alarms for notifications or automated actions; and to monitor logs for a better understanding and operation of the system.
* Worked closely with QA Teams, Business Teams, DBA team and Product Operations teams to identify QA and UAT cycles release schedule to non-production and production environments.

**Environment:** AWS (EC2, VPC, ELB, S3, Amazon RDS, DynamoDB, Lambda, CloudTrail, VPC, EBS, SQL, SQS, SNS, ELB, Cloud Watch, CloudFormation, AWS Auto Scaling, Elastic Beanstalk and Route 53), CHEF, Jenkins, Docker, Kubernetes, AWS CLI.

**Client: Merck Aug2015 – Dec2019**

**Role:** **Cloud/Build and Release engineer**

It’s an American based its engaged with manufacturing and marketing of pharmaceuticals and chemicals. Responsible for delivering high quality products and services. Aspire to improve the health and wellness of people and animals worldwide.

* Automated Continuous Integration builds, nightly builds, deployments, and unit tests across multiple different environments (Development, QA, Training, Production) each constituting different types of servers (DB, App, Web) and different number of servers using AWS CLI, AWS Code Deploy, Git.
* As a member of, redefined processes and **Release Engineering group** implemented tools for software builds, patch creation, source control, and release tracking and reporting.
* Worked in all areas of **Jenkins** setting up **CI** for new branches, build automation, plugin management and securing Jenkins and setting up master/slave configurations.
* Resolved Merge Conflicts, configured triggers and queued new builds within the release pipeline and monitored & tracked the deployments.
* Integrated various Version control tools, build tools, **Nexus**, and deployment methodologies (scripting) into Jenkins to create an end-to-end orchestration build cycles.
* Troubleshooted build issues in **Jenkins**, performance and generating metrics on master's performance along with jobs usage.
* Maintained product release process, including generating and delivering release packages, generated various metrics for tracking issues against releases and the means of tracking compatibility among products.
* Supported engineering plans and schedules by providing Configuration Management / Release Engineering services to build, deploy, develop scripts, oversee branch, and build automated tools as necessary to offer services to engineering team.
* Installed and used **Chef** **Server** **Enterprise** on premise, workstation and bootstrapped the nodes using knife and automated by writing scripts in Chef Recipes and **Cookbooks** with test-kitchen/chef spec.
* Managed the configurations of nodes using **Chef**, configured **Jenkins** builds for Continuous Integration and Delivery for automated deployments. Worked on **Docker** container snapshots, attaching to a running container, removing images, managing Directory structures, and managing containers.
* Built **Docker** **files** to create containers using Docker engine and Docker Machine environments to deploy the microservices oriented environments for scalable applications.
* Utilized **Splunk** to monitor resources such as EC2, CPU memory, Amazon **RDS** **DB** services, **Dynamo** **DB** tables, **Elastic** **Block** **Store** (**EBS**) volumes to set alarms for notifications or automated actions; and to monitor logs for a better understanding and operation of the system.
* Worked closely with QA Teams, Business Teams, DBA team and Product Operations teams to identify QA and UAT cycles release schedule to non-production and production environments.

**Environment:** AWS (EC2, S3, Amazon RDS, DynamoDB, Lambda, CloudTrail, VPC, EBS, SQL, SQS, SNS, ELB, Cloud Watch, CloudFormation, AWS Auto Scaling, Elastic Beanstalk and Route 53), Jenkins, Docker AWS CLI, Unix/Linux, Cookbooks.

**Client: Anthem April 2012 – Aug 2015**

**Role: Linux Administrator**

Is one of the leading US based healthcare companies dedicated to improving the lives and healthcare Wellpoint is a leader in providing organizations with a full suite of services to meet all their occupational health and safety needs with different types of health plans.

* Responsible for System Administration, System Build, server builds, installs, upgrades, Patches, Migration, Troubleshooting, Security, Backup,
* Disaster Recovery, performance Monitoring and Fine-tuning on Unix Red hat Linux systems.
* NIC bonding/NIC teaming configuration in Linux/Unix/Windows/VMware systems to increase the bandwidth or redundancy based on requirement by the application
* Used MAVEN and Ant as build tools for building deployable artifacts through Jenkins (jar, war& ear) from source code.
* performance tuning and management for Linux server and working with the application/database team to resolve issues.
* Creating the file systems using Red Hat volume manager and performing the health check on regular basis for all Linux servers.
* Planned and performed the upgraded-on Linux operating systems and hardware maintenance on HP and POWER servers like increasing memory, disk, replacing failed hardware.
* Configuration and administration for LDAP, NFS, FTP, SAMBA, and POSTFIX servers in Red Hat Enterprise Linux.
* Assembled Puppet Master, Agent, and Database servers on multiple Red Hat Enterprise.

**Environment:** ANT, Perl, MS Build, Linux, Unix, ShellScript, AWS, Python, RedHat, NFS, FTP, Apache Tomcat.

**Education Details:**

* Bachelor’s Computer science – 2008 -2011 from Osmania University
* Master’s Cyber Security – 2021 – 2023 from Saint Leo University