# Hemanth Pogula

# Dallas, TX | 4693249141| hemanthp0696@gmail.com |

# Professional Summary:

* Having around **9+ years** of experience in the IT industry comprised of **Development**, **Systems** **Administration**, Infrastructure Management, Installation, Configuration, Tuning, and Upgrades of Linux (**Red Hat and Oracle**).
* Software Configuration Management (**SCM**) experience includes **Build**/Release **Management**, Change/Incident Management implementing duties such as monitoring, Automation, **deployment**, **documenting**, **support**, and troubleshooting along with **Amazon** **Web Services** and **Cloud Implementation**.
* Well experienced in **Cloud/DevOps**, Continuous Integration and Continuous Delivery (**CI/CD pipeline**), **SDLC**, Site **Reliability**, **Build** and Release management, and **Windows**/**Linux** administration.
* Participating in **Agile/Scrum** methodologies and driving continuous improvement.
* Dedicated DevOps engineer with a strong background in AWS resources, Kubernetes, and **CI/CD pipelines**.
* Skilled in automating infrastructure, containerizing applications, and troubleshooting Linux systems.
* Unique experience working with AWS services like **VPC** with scalable infrastructure, knowledge of transferring petabytes of data between the on-premises data center and **S3** using **Snowball**, and written **CloudFormation** templates to provision and configure many resources and services running on **AWS EC2**.
* Proficient in designing REST APIs adhering to best practices and industry standards (like REST, JSON, Swagger/OpenAPI).
* Expertise in automating Aurora cluster provisioning and configuration using AWS **CloudFormation or Terraform**.
* Proficient in designing, implementing, and managing infrastructure as code using tools like **Terraform**, **CloudFormation**, or Ansible.
* Skilled in DevOps principles including continuous integration (CI) and continuous delivery (CD). Familiar with automating builds, tests, and deployments using tools like Jenkins, GitLab CI/CD, or similar. Knowledgeable in infrastructure automation and configuration management tools (e.g., Terraform, Ansible).
* Implemented automated CI/CD pipelines using Azure DevOps or other tools for deploying applications on AKS, reducing deployment time by [percentage].
* Demonstrated proficiency in using **AWS** **SageMaker** for building, training, and deploying machine learning models at scale
* Integrated AI-ops into CI/CD pipelines to automate testing, validate code quality, and optimize deployment processes, accelerating release cycles and improving efficiency
* Experience with **AWS** services **EC2**, managed **Docker** containers on a cluster hosted on a serverless infrastructure using **ECS** and distributed the application traffic in **ELB**, CloudFront to distribute content to edge locations, **Cloud** watch to set alarms and notifications. Also worked with Glacier storage, IAM policies for different users, RDS, Route53, and SNS.
* Experienced in designing and developing applications using Go programming language (Golang).
* Skilled in orchestrating complex workflows using Ansible Tower or AWX to manage inventory, scheduling, and notifications.
* Proficient in configuring, and managing Amazon Aurora database clusters for MySQL and PostgreSQL
* Experience in Designing Azure Resource Manager Template and in designing custom build steps using PowerShell. Written Power shell scripts to perform Patching, Imaging, and Deployments in Azure.
* Proficient in Google Cloud services such as Compute Engine, Kubernetes Engine (GKE), Cloud Storage, BigQuery, and Cloud Functions
* Designed and deployed comprehensive monitoring strategies using **Open Telemetry** to collect, process, and export telemetry data for applications and infrastructure.
* Proficient in Agile practices including sprint planning, daily stand-ups, sprint reviews, and retrospectives to foster iterative development and team collaboration.
* Good Knowledge of Azure cloud services, Azure storage, Azure active directory, and Azure Service Bus. Managed Clients' Microsoft Azure-based PaaS and IaaS environments.
* Knowledgeable in **Kafka**, **Confluent Cloud**, and writing **Ansible** playbooks for automating deployment and management tasks.
* Proficient in provisioning, configuring, and managing EMR clusters on AWS for scalable big data processing.
* Worked on Configuration, SCM, Build/Release Management, Infrastructure as a code (**IAC**), and **Azure DevOps** operations Production and cross-platform environments.
* Hands-on experience with containerization technologies like **Docker** and orchestration tools such **as Kubernetes** for deploying and managing containerized applications.
* Having good experience in implementing **CI/CD** pipelines end to end including version control in **GitHub**. Creating jar, and war files in Maven, and Gradle, later integrated to Jenkins using webhooks in **GitHub** and deployed the artifacts into **Nexus**, and **Frog**.
* Leveraged NLP techniques in AI-ops to analyze logs and monitoring data, improving root cause analysis and operational insights
* Designed filesystems to install 3 instances of Jenkins on a single server with different port numbers and deployed Jenkins war file onto Apache Tomcat with different ports and integrated them into load balancer to perform round-robin distribution of builds in **Jenkins**.
* Experience in database schema design, optimization, and performance tuning for Aurora instances.
* Designed and implemented modern web architectures leveraging microservices and cloud-native principles on AWS cloud infrastructure.
* Monitored **Kafka** clusters using Confluent Control Center, Prometheus, and Grafana dashboards, proactively identifying performance bottlenecks, resource utilization issues, and other operational concerns. Conducted root cause analysis and implemented remediation actions to resolve incidents and prevent future issues.
* Utilized **New Relic** to monitor application performance, infrastructure metrics, and distributed tracing, enabling proactive issue detection and resolution
* Experienced in writing Ansible playbooks to define tasks, roles, and dependencies for consistent and repeatable infrastructure deployments.
* Hands-on experience deploying and configuring applications like Hadoop, Spark, Hive, and HBase on EMR clusters.
* Identifying areas for process improvement within **Agile** practices and DevOps workflows.
* Creating scripts in **DSL** Groovy which integrates with **Jenkins** for Automation to create seed jobs.
* Experienced with Google Cloud Console, Cloud SDK, and GCP APIs for managing cloud resources
* Implemented a **CI/CD** pipeline using Azure DevOps (**VSTS/TFS**) in both cloud and on-premises with **GIT, MS Build**, **Docker**, and **Maven** along with Jenkins pipeline builds & YAML/JSON.
* Familiar with integrating GitHub repositories with CI/CD pipelines (e.g., GitHub Actions) to automate build, test, and deployment workflows.
* Experienced in leveraging AWS services such as AWS Code Deploy and Elastic Load Balancer to orchestrate seamless Blue/Green deployments.
* Experienced in deploying and managing containerized applications on AKS, leveraging Kubernetes concepts such as pods, deployments, services, and ingress controllers.
* Designed, developed, and deployed applications onto several cloud-based solutions using configuration Management tools like **Chef**, **Puppet**, and **Ansible**.
* Experience in developing Rest APIs using frameworks and languages such as Node.js, Python Flask, or Java Spring Boot.
* Implemented CI/CD pipelines using tools like **Jenkins or GitLab CI** to automate builds, tests, and deployments of containerized applications.
* Utilized Golang to build robust automation tools and scripts for CI/CD pipelines, deployment, and infrastructure management
* Orchestrated Oracle EBS deployments using Docker containers for improved scalability and environment consistency across development, testing, and production stages.
* Experienced in using Helm charts and Helmfile to manage Kubernetes applications, ensuring smooth deployment and scaling.
* Developed and maintained ROS packages for robotic systems, including nodes, libraries, and interfaces, to facilitate robot control, perception, and communication.
* Maintained multiple Applications that run in Multi-Tenant Environments, Such as Load Balancer to Web server to App Server, and Databases which are SQL databases like **PostgreSQL**, **MySQL**, and **NoSQL** Databases.
* Expertise in configuring and administering network services such as **DNS**, **NFS**, **SMTP**, **NTP**, **DHCP**, **LDAP**, **SAMBA**, and Mail Servers. Knowledge of using various routed protocols like TCP, FTP, SFTP, SSH, HTTP, HTTPS, and Connect Direct.
* Experience in setting up alerts and notifications in New Relic to proactively monitor service-level objectives (SLOs) and key performance indicators (KPIs).
* Implemented Ansible playbooks for automating infrastructure provisioning and configuration updates across AWS environments.
* Demonstrated a proactive approach to learning new technologies and methodologies related to NixOS, Nix package manager, and ROS to stay updated with industry trends and best practices.
* Skilled in using GitHub for collaboration through issues, wikis, and project boards, enhancing team communication and project documentation.
* Proficient in Ansible for automating infrastructure provisioning and configuration management.
* Managed all the bugs and changes in a production environment using the **JIRA**/**TFS**/**HP** **ALM** tracking tool and configured various workflows, customizations, and plug-ins for the JIRA bug/issue tracker.
* Worked on web servers like Apache, and Nginx and application servers like Web **Logic**, **Tomcat**, **WebSphere**, **JBOSS**, and **IIS Server** to deploy code.
* Ability to independently write Terraform code for infrastructure as code (IaC) deployments and configuration management.
* Expertise in fine-tuning configurations for optimal performance and resource utilization on EMR instances
* Adapting Agile/Scrum methodologies to suit the team’s specific needs and project requirements.
* Experience in keeping up and looking at log archives using monitoring tools like **Nagios**, **Splunk**, **CloudWatch**, **ELK** **Stack**, **Dynatrace**, **Datadog**, **Prometheus**, **Grafana**, and **AppDynamics**.
* Identified and resolved software bugs, performance bottlenecks, and compatibility issues within NixOS deployments and ROS applications through systematic debugging and testing.
* Built end-to-end machine learning pipelines using **GCP Vertex AI** to streamline model development and deployment.
* Skilled in documenting Rest APIs comprehensively using tools like Swagger UI or Postman, ensuring clear and accessible API documentation.
* Expertise in **Git and GitHub** for version control, collaboration, and managing code repositories.
* Implemented Agile practices to enable continuous delivery, ensuring that software updates are consistently delivered to stakeholders with minimal lead time.

**TECHNICAL SKILLS:**

| Cloud Platform | Amazon Web Services, Azure Cloud |
| --- | --- |
| CI/CD | Jenkins, CircleCI, Azure DevOps and AWS CICD tools. |
| Containerization & Orchestration Tools | Docker, AWS Fargate, EKS, Kubernetes Clusters, AWS ECS. |
| Configuration Management Tools | Ansible, Puppet, and Chef. |
| Version Control Tools | GIT, Bit bucket and Code Commit |
| Build Tools | Maven, ANT, and Gradle |
| Tracking Tools | JIRA |
| Artifact Repositories | Nexus and Artifactory |
| Logging & Monitoring Tools | Prometheus, Splunk, Cloud Watch, and Grafana |
| Application Servers | Apache Tomcat, Nginx, Web Sphere, Web Logic, JBoss, Samba, SQL Server |
| Database | Amazon Aurora, Dynamo DB, MongoDB, Oracle, SQL Server, MySQL |
| Operating Systems | Windows, Linux/Unix, and MAC OS |
| Network Service & Topologies | LDAP, DNS, Web, FTP, Sub netting, LAN, VPC |
| Scripting Languages | Python, SQL, Groovy, HTML, CSS3, JSON, and YAML |
| Infrastructure Spin-up Tools | Terraform, AWS Cloud Formation, AWS CDK and AWS CLI |

**Professional Experience:**

**Client: Freddie Mac US state Oct 2022 – Present**

**Role: Sr. AWS DevOps Engineer**

**Responsibilities:**

* Developed multiple **AWS** lambdas using python scripting language.
* Worked on various kinds of **Continuous Integration and Continuous Delivery** using tools such as **Jenkins, AWS CICD s**ervices**.**
* Creating and managing **EKS** Cluster and Node-groups using **Terraform** scripts and deploying stateless applications on cluster using **kubectl** command-line tool.
* Hands-on experience in setting up Aurora clusters for high availability (multi-AZ deployments) and scalability (auto-scaling read replicas).
* Integrated and deployed applications using various **AWS** server less services like **Lambda, DynamoDB, Event Bridge, SQS, SNS, S3** to maintain robust and to decouple the application to reduce the application failures.
* Worked on lifting and shifting the applications from on-prem to **AWS** Infrastructure to reduce the server maintenance and utilizes the scalability.
* Implemented CI/CD pipelines for Oracle EBS using tools like Jenkins, Ansible, or Terraform to automate provisioning, configuration, and deployment processes.
* Designed and developed **Helm charts** for packaging and deploying microservices-based applications in Kubernetes, incorporating best practices for modularity, reusability, and configurability.
* Designed and implemented microservices in Golang, emphasizing scalability, reliability, and performance.
* Proficient in writing Groovy scripts to support Jenkins pipelines and automating routine tasks. Skilled in **Bash/Shell** scripting for Linux system administration and automation.
* Wrote **Cloud-Formation** scripts to automate AWS services which include **EC2**, **VPC**, **subnets**, **security groups** and **EKS** cluster
* Automated model training workflows using **AWS** **SageMaker** capabilities, improving efficiency and reducing time-to-deployment.
* Implemented ROS nodes in Python or C++ for real-time data processing, sensor integration, and robot behavior coordination.
* Implemented AI-ops solutions for proactive monitoring and anomaly detection, reducing incident response time and enhancing system reliability.
* Writing **Kubernetes manifest files** which describes the object and its attributes for resources like **pod**, **replica**-**set**, **configmap**, **secret** and **Deployment**.
* Demonstrated expertise in implementing and configuring monitoring and observability solutions, with a focus on **Open Telemetry** and **New Relic.**
* Proficient in implementing **Trunk Based Development** strategies to promote collaboration and reduce integration issues.
* Experienced in Agile principles to visualize workflow, limit work in progress (WIP), and optimize cycle time for continuous delivery.
* Utilized **Google Cloud Deployment** Manager/Terraform to automate provisioning of GCP resources, improving deployment efficiency by [metric].
* Implemented Release practice and responsible for pushing builds into Develop, QA, UAT and **Production** environments.
* Configured and administered NixOS-based systems, leveraging declarative configuration with Nix expressions to manage system packages, services, and configurations
* Responsible **for Build and Packaging** process of software. Responsible for maintaining source code repositories using **Git** and to track documents and control changes to source code.
* Hands-on experience with Rest API gateway solutions (e.g., AWS API Gateway, Kong, Apigee) for API routing, authentication, and rate limiting.
* Skilled in utilizing Infrastructure as Code (IaC) tools like AWS CloudFormation or Terraform to manage and provision the infrastructure required for **Blue/Green** environments.
* Set up monitoring and logging solutions (e.g., ELK stack, Prometheus/Grafana) for Oracle EBS instances to ensure system health, performance optimization, and proactive issue resolution
* Developed AI-ops systems to autonomously detect, prioritize, and resolve incidents, improving uptime and reducing manual intervention.
* Integrated Golang applications with cloud services (e.g., AWS SDK, Google Cloud APIs) for provisioning, monitoring, and managing cloud resources
* Skilled in monitoring EMR cluster performance using AWS CloudWatch and other monitoring tools to ensure efficient data processing.
* Active participant in GitHub community, contributing to open-source projects, and leveraging GitHub's features for transparency and engagement in software development.
* Schedule **execution** of **builds** and ensure that all build failures are resolved in **timely manner**, support **SCM** and Build environment.
* Integrated **Neptune** with other AWS services like IAM, CloudWatch, and AWS CloudTrail for security, monitoring, and auditing
* Integrated ROS-based applications with robotic hardware components (sensors, actuators) and middleware (e.g., ROS 2, Gazebo) to simulate and test robotic behaviors.
* Implemented Kafka security features such as SSL/TLS encryption, SASL authentication, and ACL-based authorization to ensure data security and compliance with regulatory requirements.
* Leveraged **Terraform** to provision and manage AWS resources, ensuring infrastructure scalability and reliability
* Used Jenkins pipelines to drive all micro services build out to **AWS Elastic Container Registry (ECR)** and then deployed to **EKS**.
* Defined and maintained **Helmfile** configurations to manage Helm releases across different environments (e.g., development, staging, production), specifying chart versions, values, and overrides for each environment
* Wrote and maintained custom scripts to increase system efficiency and performance time.
* Orchestrated **blue-green deployments** and canary releases on GKE to minimize downtime and mitigate risks during application updates.
* Created RESTful APIs in Golang for internal tools and services, ensuring efficient communication and data exchange.
* Implemented customizations and optimizations using NixOS modules and options, ensuring system stability and reproducibility.
* Familiarity with distributed tracing using **Open Telemetry** and New Relic Distributed Tracing to analyze latency and performance bottlenecks across microservices
* Experience setting up monitoring and analytics for REST APIs using tools like Prometheus, Grafana, or AWS CloudWatch.
* Creation and renewal of **SSL** certificates using **AWS Certificate Manager** (**ACM**) and update **CNAME** name and **CNAME** value in Route 53 records.
* Integrated **AWS** **SageMaker** such as S3 for data storage, IAM for access management, and CloudWatch for monitoring
* Implemented **Blue-Green Deployment** and Canary Release strategies to minimize downtime and mitigate risks during deployment.
* Proficient in Agile methodologies such as Scrum and Kanban. Experienced in iterative development, sprint planning, daily stand-ups, and sprint retrospectives. Able to facilitate cross-functional team collaboration and deliver incremental software improvements.
* Configuring/ updating the Name Servers of **Route53** hosted zone in **Go Daddy** **DNS** **service**. Experience in Branching, Merging, Tagging, and maintaining the version across the environments using **SCM** tools **GIT (GitHub, GitLab), Clear Case, Harvest, Subversion** and **VSS.**
* Knowledge of the **12 Factors** methodology for building scalable and maintainable cloud-native applications.
* Experience with **Feature Toggles (Feature Flags)** for controlled feature releases and experimentation in production environments.
* Experience in optimizing EMR workflows and jobs to improve job completion times and reduce costs.
* Able to integrate Ansible with CI/CD pipelines (e.g., Jenkins, GitLab CI/CD) to automate infrastructure changes as part of the deployment process.
* Knowledgeable in implementing security mechanisms (e.g., OAuth, JWT) and authentication protocols to secure REST APIs.
* Implemented **InfluxDB**, **Grafana** runs as **docker** containers which track the status and percentage of success for Jenkins and all web apps. Implemented **Terraform** modules for the deployment of applications in mixed cloud environments.
* Implemented Agile principles of continuous improvement (Kaizen) through retrospectives and feedback loops to enhance team performance and delivery predictability.
* Proficient in designing and implementing CI/CD pipelines using tools like Jenkins, GitLab CI/CD, or CircleCI to automate build, test, and deployment processes.
* Implemented Git workflows and branching strategies in GitHub, enhancing code collaboration and version control management
* Implemented Agile metrics (e.g., velocity, burn-down charts) to track team progress and forecast delivery timelines, facilitating data-driven decision-making and transparency
* Designed migration strategies tailored to business requirements, including lift-and-shift, re-platforming, and refactoring approaches based on application dependencies and architecture.
* Utilized **GCP Vertex AI's** AutoML capabilities to automate model selection, hyperparameter tuning, and feature engineering
* Managed Kubernetes clusters on **GKE,** handling [number] of nodes and optimizing workload deployment strategies.
* Integration of Aurora deployments into CI/CD pipelines for automated testing, deployment, and version control.
* Ensured **AWS** **Neptune** clusters scaled seamlessly to handle growing datasets and query loads.
* Experience in using Mesosphere for clustering to provide efficient resource isolation and sharing across distributed applications or frameworks.
* Plan and execute **RPM** and **YUM** packages and update installations necessary for optimal system performance
* Experience in configuration of **DNS**, **LDAP**, **NFS**, **DHCP** **Server**, **Samba**, and **TCP**/**IP** and have experience in process automation and system monitoring using Shell Scripts.
* Experienced in integrating CI/CD pipelines with version control systems (e.g., Git, GitHub, Bitbucket) to trigger automated builds and deployments upon code commits.
* Experience in designing and executing automated tests (integration tests, endpoint testing) for REST APIs to validate functionality and performance.
* Collaborated with robotics engineers and developers to troubleshoot integration issues and optimize system performance in ROS environments.
* Facilitated Agile ceremonies, including backlog grooming and sprint planning, to align team goals with business objectives.
* Assist in **LDAP** server configuration for user authentication on network
* Experience in maintaining Atlassian products like **JIRA**, **Confluence**, **Fisheye**, **Bamboo**, **Crowd** etc.
* Experience on **JIRA** administration and **JIRA** workflow design and JIRA service desk.

**Client: TIAA, New York United States Aug 2019 to Sep 2022 Role: Sr. AWS DevOps Engineer**

**Responsibilities:**

* Developed and tested enterprise products, with current focus on cloud-based application and giving solution to challenges imposed by multi data center deployed **SaaS** products and their needs for DevOps tools and automation.
* Proficient in setting up and configuring CI/CD pipelines in **OpenShift** using tools like Jenkins, Tekton, or ArgoCD for automating application builds, tests, and deployments
* Created CloudFormation Templates for different environments (**DEV**/stage/prod) to automate infrastructure (**ELB**, **CloudWatch** alarms, **ASGs**, **SNS**, **RDS** etc.) on click of a button.
* Configure, monitor, and automate Amazon Web Services as well as involved in deploying the content cloud platform on Amazon Web Services using **EC2**, **S3** and **EBS**.
* Leveraged IaC principles with tools like Terraform or CloudFormation to manage Oracle EBS infrastructure on cloud platforms (e.g., AWS, Azure, Oracle Cloud Infrastructure).
* Migrated data from Datacenters and on-premises to Cloud using **AWS** Import/Export Snowball service.
* Designed **AWS** Cloud Formation templates to create custom sized **VPC**, Subnets, **NAT** to ensure successful deployment of Web applications and database templates.
* Utilized AI-ops for predictive maintenance, predicting infrastructure failures and optimizing maintenance schedules to minimize downtime and increase reliability
* Implemented robust backup and recovery strategies for **AWS Neptune** databases, ensuring data integrity and availability.
* Developed Nix packages for various software applications, adhering to best practices in package definition, dependency management, and version pinning.
* Promoted Agile principles to enhance team collaboration, transparency, and delivery efficiency.
* Created and maintained Amazon Virtual Private Cloud (**VPC**) resources such as Subnets, Network Access Control Lists (**NACL**), and Security Groups.
* Skilled in setting up monitoring and alerting for Aurora clusters using AWS CloudWatch, custom metrics, and alarms.
* Proficient in automating EMR cluster management tasks using AWS SDKs, CLI, and infrastructure-as-code tools like CloudFormation or Terraform.
* Used Terraform in **AWS** Virtual Private Cloud (**VPC**) to automatically setup and modify settings by interfacing with control layer.
* Developed **Kafka** Streams applications for real-time data processing, including event-driven microservices and complex event processing (CEP) pipelines, to extract insights and derive actionable intelligence from streaming data.
* Responsible for managing the Chef Client nodes and upload the cookbooks to chef-server from Workstation.
* Written Chef Cookbooks and recipes in Ruby to Provision several pre prod environments consisting of Cassandra DB installations.
* Utilized Kubernetes and **OpenShift** to containerize applications and manage containerized workloads effectively.
* Wrote Ansible Playbooks with **Python** SSH as the Wrapper to Manage Configurations of AWS Nodes and Test.
* Worked with **Docker** components like **Docker** Engine, **Docker** -**Hub**, **Docker**-**Compose**, **Docker** Registry and **Docker** **Swarm**.
* Integration of REST APIs into CI/CD pipelines using tools like Jenkins, GitLab CI/CD, or AWS CodePipeline.
* Leveraging kops implemented a Kubernetes Container Orchestration solution within AWS allowing for easy management, creation, and recovery of **AWS** assets.
* Implemented custom algorithms and scripts in **AWS** **SageMaker** environments, leveraging its flexibility for specific project requirements.
* Managed package builds, deployments, and upgrades using Nix package manager, ensuring consistency across development, testing, and production environments.
* Implemented Azure Monitor for monitoring AKS clusters and applications, configuring alerts and dashboards to monitor performance metrics and detect issues proactively
* Advanced knowledge of **New Relic APM (**Application Performance Monitoring), Infrastructure Monitoring, and Insights for deep dive analysis and troubleshooting.
* Collaborated with development teams to define GKE architecture and deployment strategies aligned with application requirements and scalability goals.
* Knowledgeable in implementing security best practices for Aurora databases, including encryption at rest and in transit.
* Monitored Applications and Servers through Nagios, Web Servers Administration on Apache and Nginx.
* Implemented **Feature Toggles** to enable phased feature releases and A/B testing, improving user experience and feature adoption
* Documented software designs, APIs, and technical specifications related to ROS packages and Nix configurations for knowledge sharing and team alignment
* Proficient in Agile development practices, delivering incremental value and adapting to changing requirements throughout project lifecycles.
* Knowledgeable in implementing security best practices and ensuring compliance through Ansible automation, including configuration drift detection and remediation
* Re-architected the storage designs and the Devops team to re-vamp Cisco's **WebEx 11** cloud platform
* Deployed machine learning models on **GCP** **Vertex** **AI**, utilizing containerization for scalable and reliable model serving.
* Wrote Bourne and bash scripts to automate hammer kickstarts with expect to suppress passwords being echoed in history/cli. Proceed moving our platform to Amazon (**AWS**) and Google (**GCE**).
* Scripting experience (e.g., Python, Bash) to automate routine EMR operations, job scheduling, and data pipeline orchestration.
* Installed, tested, and deployed monitoring solutions with **Splunk**.
* Experience in implementing SAFe principles and practices for large-scale Agile projects, ensuring alignment across multiple teams and departments.

**Client: Charls Schwab Corporation, Austin, TX Sep 20****17 – July 2019**

**Role: Sr. AWS DevOps Engineer**

**Responsibilities:**

* Install, configure and troubleshoot Linux **OS** on development servers.
* Troubleshooting performance related issues (**CPU**, Memory, Network, permissions) on **EC2** Servers.
* Proficient in using Ansible to automate provisioning, configuration management, and deployment tasks across heterogeneous environments.
* Led or participated in Agile transformation initiatives within teams or organizations, driving adoption and improving team efficiency.
* Implemented migrations using **AWS** Server Migration Service (**SMS**) using "lift and shift" methodology to migrate on-premises workloads to **AWS** in a reliable manner along with AWS Database Migration Service (**DMS**), **AWS** Snowball to transfer large amounts of data and between on-premises and cloud services.
* Provisioning and configuring the Logical Volume Management, User Management and File system management.
* Perform and Implement security base lining on **Linux** **Systems**.
* Troubleshooting skills to hunt down the root causes of issues and persistence in preventing them from happening again in **AWS** environments.
* Automated application deployment workflows using **Helm and Helmfile in** CI/CD pipelines, enabling continuous delivery and seamless rollouts of application updates and releases in Kubernetes clusters.
* Perform troubleshooting of system related issues including connectivity issues; identify root cause and resolutions on computer services.
* Designed and deployed **Kafka clusters** in production environments, ensuring scalability, reliability, and high performance for real-time data processing and analytics
* Used message queuing service to decuple the application using Amazon Simple Queue Service (**SQS**), Azure Queue Storage, event routing service like **AWS** Event Bridge, **Azure** Event Grid to reduce the possibility of event failures.
* Proficiency in configuring and managing **Open Telemetry** agents, collectors, and exporters to gather telemetry data from distributed systems.
* Knowledgeable in implementing security best practices for EMR clusters, including encryption, IAM roles, and network security configurations.
* Advanced proficiency in **OpenShift** Container Platform, including cluster administration, application deployment, and resource management using the OpenShift CLI (oc) and web console
* Wrote Groovy scripts to support **Jenkins** pipelines and automate build, test, and deployment tasks.
* Integrated **Vertex AI** with other GCP services such as BigQuery for data analytics and Cloud Storage for data storage.
* Configured multiple lambdas with DynamoDB, Event Bridge, **SQS**, **SNS**, **S3**.
* Deployed multiple **AWS** services along with code pipeline using Cloud Development Kit (CDK) using python, and Git as repository.
* Implemented a **CI/CD** pipeline with **Docker**, **Jenkins**, Team Foundation Server (**TFS**), **GitHub**, and **AWS** Container Service.
* Configured and deployed applications using AWS Elastic Bean Stack environment.
* Provisioning of **AWS** resources like Lambda, **ELB**, **ECS**, **EC2**, **ASG**, **EKS**, **SQS**, **Route53** using Cloud Formation and Terraform.
* Design and implementation of backup and restore strategies for Aurora clusters, including automated snapshots and point-in-time recovery
* Schedule the execution of builds and ensure that all build failures are resolved in a timely manner, support **SCM** and Build environment.
* Managed infrastructure configuration using Terraform and Ansible, ensuring consistency and compliance with best practices.
* Managed environment-specific configurations using **Helm** values files and **Helmfile** environments, allowing for fine-grained control over application settings and resource allocations in different environments
* Troubleshooting of build errors in Jenkins for both the Software Configuration Management team and the development team.
* Setting up **CI/CD** pipeline using Jenkins pipelines and GitHub.
* Following proper industry standards for naming and tagging resources.
* Ample experience in customizing and setting up the helm charts.
* Scaling the number of **EC2** instances horizontally and vertically using the auto-scaling fronted by Elastic Load Balancer which optimizes the cost.
* Implemented migration methodologies to optimize performance, scalability, and cost-efficiency in cloud environments.
* Created and maintained tables in both **AWS** and **Azure** using **DynamoDB** and **Cosmos** **DB**.
* **Automated manual**, long-running, error prone deployments of **Azure** **services** using **Azure** **Automation**.

**Client: Quick Heal, Pune, India July 2014 – Aug 2017**

**Role: AWS DevOps Engineer**

**Responsibilities:**

* Created continuous integration/deployment with Application Release Automation by integrating and improving the client's existing infrastructure and building pipelines, largely aiming for autonomous automation when possible.
* Worked on Multiple **AWS** instances, set the security groups, Elastic Load Balancer and **AMIs**, and Auto scaling to design cost-effective, fault-tolerant, and highly available systems.
* Worked extensively with AWS services like **EC2**, **S3**, **VPC**, **ELB**, **Autoscaling** Groups, **IAM**, **CND**, **Cloud Trail**, **CloudWatch**, **NAT** and **RDS**.
* Administered databases using **RDS** and DynamoDB in **AWS** and executed the **DML** and **DDL** scripts.
* Used **JIRA** as a defect tracking system and configured various workflows, customizations, and plugins for **JIRA** **bug**/issue tracker, integrated Jenkins with **JIRA**, **GitHub**.
* Performed **JIRA** administration, setting up new projects, creating new workflows, and adding users to roles.
* Worked with the layout of project workflow/pipelines using Jenkins as a **CI tool**, performed a cleanup of the file system using Shell scripting, and configured various Cron jobs.
* Created the branches in **Git** to perform the simultaneous development process to support quarterly releases.
* Developed and maintained Cloud Formation **JSON** Templates and automated cloud deployments using Chef.
* Involved in setting up builds using Chef as a configuration management tool and managed the configurations of servers and Installed Chef Server on the workstation and bootstrapped the nodes using **Knife** and was involved in writing Chef **Cookbooks** and recipes to automate the deployment process.
* Hands-on knowledge with the **ChefDK** for storing cookbooks and in documenting the pattern and use of Delivery **CLI** to automate the adeptness in Chef.
* Prepared projects, dashboards, reports, and questions for all JIRA related services and generated scripts for effective integration of **JIRA** applications with other tools.
* Worked with **Ant** and Maven build tools and implemented them with **AWS** **services**.
* Provisioned the highly available **EC2** Instances using Terraform and cloud formation and wrote new **plugins** to support new functionality in **Terraform**.
* Installed, configured, and administered Bamboo on Linux machines along with adding/updating plugins like **GIT**, **ANT**, **Sonar**, Check style, **Build Pipeline**, etc.
* Implemented a Continuous Delivery Pipeline with Bamboo and **GitHub** and worked on designing and implementing a continuous integration system using **Bamboo** by creating **Python** and **Perl** **scripts**.
* Worked with **Terraform** key features such as Infrastructure as code, **Execution** **plans**, **Resource** **Graphs**, and **Remote** **State**.
* Used **Bamboo**, Build forge for Continuous Integration and deployment into **Tomcat** **Application** **Server**.
* Developed installer scripts using **Python**, **Groovy**, and **Linux** for various products to be hosted on Application servers.