**SAI KAUSHIK**

SR. DEVOPS ENGINEER

[kaush910@gmail.com](mailto:kaush910@gmail.com)

+1 (737)-238-0530

[LinkedIn](https://www.linkedin.com/in/sai-k-a72725287/)

**Professional Summary**

A cloud enthusiast and an IT professional with **7+ years**of experience in the IT industry with major focus on **Cloud/ DevOps,**

**CI/CD**, **Linux Systems administration**, **Build and Release Management, Containerization, Monitoring** andsolving complex problems with creative solutions, supporting development and operations in major cloud platforms like **AWS, Azure and GCP.**

* Skillful about the stages of the **SDLC (Software Development Life Cycle),** and adept at putting the principles of software configuration management into practice using the **Agile, Scrum, Kanban, and Waterfall** approaches.
* Expertise in **Amazon EC2** for virtual machines, **AWS Lambda** for serverless computing, and **Amazon ECS/EKS** for container orchestration. Worked on **Elastic Container Service (ECS),** which gives high performance and scalability service in placing the containers across the cluster.
* Worked on various **AWS cloud services** like **AWS EC2, AWS S3**, **AWS Glacier, AWS ELB, AWS VPC, AWS AMI, SNS, RDS, IAM, ROUTE 53, AWS Auto-Scaling, CloudFront, AWS CloudWatch, AWS CloudTrail, and Security Groups.**
* Proficient in building different **subnets**, **routing tables, Network ACLs,** and **NAT gateways** to design and configure secure **Virtual Private Cloud (VPC)** over private/ public networks in **AWS**.
* Used **Bash** and **Python** included **Boto3** to supplement automation provided by **Ansible** and **Terraform** for tasks like encrypting **EBS volumes**, backing **AMIs** and scheduling **Lambda** **functions** for routine **AWS** **tasks**, worked on number of projects to migrate data from on-premises databases to **AWS Redshift, RDS, S3**, and performed folder management in **S3 buckets.**
* Knowledgeable in various **GCP platform services** like **GCP Compute Engine, GCP Functions, GCP Container Security, Graphic Processing Unit, App Engine Knative, Persistent Disk, Cloud Datastore**.
* Skilled in deploying and scaling web applications using the **Google Cloud Platform (GCP)** engine in various services like **Google Cloud Load Balancing, Google Cloud Storage, and Google Stack Driver, Cloud SQL.**
* Exposed to GCP services like**, GCP Dataproc, BigQuery, GCS bucket, cloud dataflow, Pub/sub, cloud shell, GSUTIL, BQ command line utilities, stack driver monitoring and involved in IaaS, resource graph, change automation using Terraform.**
* Proficient in various Azure services like **Azure storage**, **IIS**, **Azure Active Directory (AD)**, **Azure Resource Manager (ARM)**, **Azure Storage**, **Azure Blob Storage**, **Azure** **VMs**, **Azure** **SQL Database**, **Azure Functions**, Azure **Service Fabric**, Azure **Monitor**, **Application Gateway**, **Azure** **Migration** and **Azure** **Service Bus**.
* Knowledge of **Azure Scalability** and **Azure Availability,** creating **Virtual Machine Scale Sets** **(VMSS)** using Azure Resource Manager **(ARM)** to control network traffic and build VMs availability sets using the Azure portal.
* Experienced in constructing prototypes that consists of Azure resources like Azure **SQL Data warehouse**, Azure **Data Factory**, Azure **Cosmos Db**, Azure **Logic Apps**, Azure **API management** and Azure **Key Vault**.
* Proficient in using **Terraform** for **infrastructure as code (IaaC),** demonstrating expertise in defining, provisioning, and managing cloud resources efficiently and consistently across platforms.
* Skilled in containerization using **Docker**, with a proven ability to create, deploy, and **manage containerized applications**, designed **Docker images**, **optimizing resource utilization**, and **orchestrating containers** for efficient and scalable deployments.
* Experienced with **Docker Engine** and **Docker Machine** to deploy micro services-oriented environments for ship, run and deploy the application. For speeding up the build/release engineering with experience used **Docker** for **Virtualization**.
* Worked on **Docker cloud-based registry** such as **Docker-swarm** and **Docker Hub** and application-based servers such as **Apache** **Tomcat** and integrated it with **Amazon MySQL RDS database.**
* Expertise in creating pods using **Kubernetes** and rolling updates and deployment in **Kubernetes** utilizing **Blue-Green Deployment** strategy and **Canary** **Deployment** for handling less downtime.
* Experience in deploying, configuring, and scaling applications on **Kubernetes** clusters, utilizing tools like **kubectl**, **Helm,** and **Kubernetes** Operators to streamline application lifecycle management. Expertise in setting up **Kubernetes (k8s)** clusters for running microservices and pushed microservices into production with **Kubernetes** backed Infrastructure.
* Development of automation of **Kubernetes** clusters via playbooks in **Ansible**. Used **Kubernetes** to orchestrate deployment, clustering, and management of containerized applications as Pods and **Kubernetes** **Clusters** to support automatic high availability, load balancing, health checking and failover.
* Experienced with **OpenShift** application scaling, **OpenShift administration,** OpenShift handling Docker and Kubernetes and OpenShift **Security Context Constraints (SCC)**, and service account.
* Worked on configuring and maintaining the server using **Ansible**- configurational tool and used it in writing **Ansible playbooks** in **YAML** files for provisioning virtual and physical assistance, configuring, maintaining along with patching and **software deployment** on **cloud environments**.
* Created and maintained **SSH** and configured along with **Ansible Directory files** in **playbooks** and executed commands on **remote notes** using **SSH.** Used **Ansible control server** for deploying playbooks into control machines.
* Proficient in creating and managing **Cookbooks**, **Recipes** on a **Chef workstation** using the configuration management tool- **Chef.** Used **Chef server** as a central repository for storing and managing cookbooks, data bugs, roles, other data.
* Along with **Chef server**, used **Nodes** in communication for fetching the **chef configurations**. Used **Data bags, Knife CLI tool, Chef Supermarket and Chef automate** for configuration management, app-deployment, infrastructure management.
* Experienced in configuring, maintaining, and deploying using **Puppet**. Worked on **Puppet server** in controlling the access and authenticating the **SSL certificate.** Used **Puppet agent** in fetching the data from the **Puppet server** and applied them into the **node**. Worked on **Puppet DB** to store the data and for looking up the configured data used **Hiera** and added variables, parameters, and settings separately from **Puppet Manifests**.
* Expertise in setting up **Jenkins** for installation, security, and configuration in various **CI/CD services.** Knowledgeable in several **CI/CD tools** like **GIhub actions, Harness, Azure DevOps, GitLab CI, Jenkins, and TeamCity** for automation and for builds and deployments.
* Implemented **nightly builds** in **Jenkins** to build and deploying of **Java** **codes** on daily basis and **groovy scripts** to use multi branch pipelinesin Jenkins to configure as per the **CI/CD requirement**.
* Proficient in using a variety of build automation tools to streamline software development processes like **ANT, Gradle, and Maven**. Used declarative **POM files**, **Automating Dependency Management,** and **standardized project structures** for creation/managing the project in **Maven.**
* Worked on several **SCM tools,** implemented **XML-based build files** to define and execute build tasks for **Java applications**. Developed build scripts in **Gradle** using **Groovy/Kotlin DSL.**
* Used a variety of **SCM (Source Code Management)** tools to monitor the **version control**, and to manage changes to the source code and other files to the software development process. Extensive knowledge in various **SCM** **tools** like **GIT**, **Bit Bucket**, **GitLab**, **GitHub**, **AWS** **CodeCommit**, **SVN** and **IBM Rational ClearCase.**
* Experienced in several **Big data tools** like **Nagios and Splunk**. For collecting, visualizing, and analyzing the machine generated data using **Splunk**. For processing logs, data from servers used **Splunk Log and Data Analysis** and for analyzing security related data like logs from firewalls, used **Security Information and Event Management (SIEM).**
* Experienced in various monitoring toolslike **Prisma Cloud, Logic Monitor, Datadog, Dynatrace.**
* Experienced in querying and configuring the database using **PostgreSQL, Oracle, MySQL, and MS SQL, Cassandra, MongoDB, Cosmos DB, Snowflake Server Database.**
* Experienced in Performing application deployment on servers using the **JBoss, WebLogic, and WebSphere.** For Web application development used **Tomcat, Wildfly, and IIS.**
* Skilled in **Python, Bash/Shell, PowerShell, Ruby, Perl, PowerShell, JSON, YAML, Groovy**. For administrative tasks and automation of the **build and release,** developed **Shell and Python Scripts.**
* For all the issues related to that of Software Development Life Cycle used **bug tracking tools** like **Jira, q, Remedy and ServiceNow.**

**Skills Matrix:**

|  |  |
| --- | --- |
| Cloud Platforms | AWS, GCP, Microsoft Azure |
| IAAC Tools | Terraform, CloudFormation, ARM |
| CI/CD Tools | Jenkins, Bamboo, U-Deploy, Teamcity, Azure DevOps, Flux CD |
| Build Automation Tools | Maven, Ant, Gradle |
| Version Control Tools | GIT, Bitbucket, GitLab, GitHub, SVN |
| Containerization Tools | Docker, Kubernetes, OpenShift, EKS, AKS |
| Configuration Management Tools | Ansible, Chef, Puppet, Saltstack |
| Languages/Scripting | Python, Java, Shell, Bash, PowerShell, Perl, YAML, HTML, Groovy, Ruby |
| Monitoring Tools | Splunk, Nagios, Prometheus, Grafana, CloudWatch, Dynatrace, DataDog |
| Code Coverage and Security | SonarQube, Jacoco, Veracode |
| Testing Tools | Junit, Selenium, JMeter, Cucumber |
| Bug Tracking Tools | Jira, Bugzilla, ClearQuest |
| Databases | MySQL, MongoDB, BigQuery, Cassandra, PostgreSQL, Redis, NoSQL |
| Web Servers | Apache Tomcat, WebLogic, WebSphere, JBoss, Nginx |
| Operating Systems | Windows, MacOS, Red Hat Linux, Ubuntu, Unix |

**Professional Experience:**

Verizon Communications Inc., New York, NY. December 2022- Present

Sr. DevOps Engineer

**Description:** Verizon is a leading telecommunications company, providing a wide range of wireless and wired communication

services to customers globally. I worked on Infrastructure Migration from on-premises to cloud. I collaborated with the consumer content team to automate, maintain, and monitor the application. I’m also in charge of the payment and platform teams and created pipelines for automated build and deployments along with monitoring the infrastructure and applications.

**Responsibilities:**

* Developed and managed the Cloud infrastructures using the **AWS** including services like **EC2, S3, CloudFront, Elastic File System, RDS, Route53, CloudWatch, CloudTrail, CloudFormation and IAM** which allowed automated operations, also installed, and managed the **ELK Stack for Log Management** within **EC2 / Elastic Load balancer** for **Elastic Search**
* For ensuring the secure zone inside Verizon used **Network ACLs, Internet Gateways, NAT instances and Route Tables,** for setting up new servers on **AWS** and for setting up the lifecycle policies to back the data from **AWS S3 to AWS Glacier**, utilized **Terraform Migrate** Legacy and monolithic systems to AWS.
* For making the **Virtual firewalls** which supports the traffic flow for one or more **AWS EC2 instances**, have defined the **AWS Security Groups and ACL’s**. Used the **AWS VPC**, for configuring the **VPN tunnel** to communicate inside the Verizon Corporate Network.
* For creation, management and deploying of **VMs** on **Azure** used **Azure Infrastructure**, **Azure Web Roles**, **Worker** **Roles**, **VM Role**, **AzureSQL**, **Azure Storage**, **Azure AD Licenses**, **Virtual Machine Backup.**
* Composed **ARM templates** and designed, configured **Azure Virtual Networks (VNets)**, **Subnets**, **Azure network settings,** **DHCP address blocks**, **DNS settings**, and **security policies.**
* Used Azure Auto-scaling, API management and REST APIs for deploying the **Azure IaaS virtual machines (VMs**) and **Cloud services (PaaS role instances)** into secure **VNets** and **subnets, Azure Kubernetes service (AKS),** **Kubernetes Cluster** which are usually responsible for the **cluster management.**
* Used Azure Kubernetes for container-based applications with services for running the multi container applications and monitoring the health of **clusters and pods** used **Azure Kubernetes service clusters (AKS)** on the **Azure portal,** also designed the **Azure service mesh** on top of the **Kubernetes platform.**
* Used Virtual Network to deploy the **agent nodes,** **Ingress API Gateway**, **MySQL Databases** and **Cosmo DB** for stateless storage of external data. Responsible for setting up the **AWS and Microsoft Azure** with Databricks, **Databricks Workspace** for **Business Analytics**, Manage **Clusters** in **Databricks**, Managing the **Machine Learning Lifecycle**.
* By using **Terraform** wrote **Terraform Templates, Chef Cookbooks** and pushed them into **Chef** for configuring the **EC2** **instances**. Resolved the Gateway time issue on **ELB** and moved the logs to **S3** **bucket** using Terraform.
* For managing **Kubernetes Charts** used **Helm**, for creating builds used **Kubernetes apps** and for making it into a template used **Kubernetes Manifests,** used **Helm packages** for managing and deploying, and provision of the configuration parameters also created **Clusters** using **Kubernetes** and worked on creating many **pods**, **replication** **controllers**, **replica** **sets** **services**, deployments, labels, health check and ingress by writing the **YAML** **files**.
* Used **Kubernetes** to deploy, scale, load balance and manage **Docker** **containers** with different names, spaced versions and developed test environment on **Docker containers** and configured it using **Kubernetes.**
* Led the implementation of **Docker** technology to **containerize** **applications**, enhancing their portability and scalability, created **Docker images** and **Dockerfiles** to package applications and their dependencies, optimizing deployment and reducing overhead.
* Used Docker **snapshots** and volumes to ensure data integrity and efficient backup strategies within the containerized environment, managed **Docker registries** to store, distribute, and version **Docker images**, streamlining the deployment process.
* Collaborated with the team to integrate **Docker technology** into the **CI/CD pipeline** using the **Docker-maven-plugin.** Orchestrated **Jenkins jobs** and pipelines to automate the build, test, and deployment processes of containerized applications.
* Spearheaded the setup and configuration of **Kubernetes** for orchestrating **Docker containers**, allowing for efficient management of application clusters. Engineered a **node cluster** using **Kubernetes**, optimizing resource utilization and high availability of applications.
* Contributed to the development of the **Minimum Viable Product (MVP)** in the cloud environment, showcasing the successful **migration** and deployment of critical applications. Collaborated with cross-functional teams to ensure the **seamless** **integration** of **Docker containers with Kubernetes**, streamlining deployment and scaling processes.
* Used **Jenkins** for the weekly build, test, and deploy chain, Integrated Jenkins with **GIT** for **Dev**, **Test** and **Prod** branching models for weekly releases and implemented continuous build based on check-in for applications, Created **GitHub**
* Used **Maven, Nexus, Jenkins, Git and JIRA** and saved build artifacts in **Nexus.**  Created custom **Jenkins Jobs** and pipelines which has the **Bash shell scripts** using the infrastructure provisioning simultaneously implementing a **CI/CD framework** using **Jenkins, Maven and Artifactory**
* Used **GIT source control** to establish and apply the appropriate branching, labeling/naming conventions while coordinating with the developers. Resolved the conflicts and analyzed in merging the source code of **GIT.**
* Utilized **AppDynamics** in creating and managing the service availability, also deployed the **AppDynamics** for mission critical applications, alert configuration in **AppDynamics** for Service and Server unavailability.
* Captured thread snapshots and CPU samples and CPU samples to study the application behavior under the load and to uncover any CPU intensive method calls in **AppDynamics** along with waiting or blocked thread, also established the health rules to gather the health status of an entity by defining the levels of performance based on **AppDynamics** metrics.
* Installed, configured, maintained, and administrated the network servers **DNS, NIS, NFS**, **Sendmail** and **application** **server** **Apache** **Tomcat**, **WebSphere**, **JBOSS**, **WebLogic**, **Oracle** **E**-**business** **suite**, and **Samba** on **Linux**.
* Used scripting languages such a **Shell, Perl, Python** for build deployment, build scripts, and automated solutions, also scripted in **Groovy** to automate the collection analysis of **Cassandra.**
* Implemented the **SonarQube** integration with **Jenkins,** used Quality Analysis tools like **JUnit, SONAR**, to find bugs and check style, expert in automate the post-build integration including the Code Coverage.
* Designed and deployed the **ELK** **clusters** **Elasticsearch**, **Logstash**, **Kibana**, **beats**, **Kafka**, **zookeeper** etc., also used the **Elasticsearch** for Search and using **ELK** **stack** for logging and monitoring our systems using **Beats**.
* Worked on build deployment, build scripts and automated solutions using scripting languages such as **Shell**, **Perl**, **Python** and **Scripted** in **Groovy** to automate and collection analysis of various **DevOps** **tools**.
* Used **Kafka** to collect Website activity and Stream processing. Used **JIRA** in deploying and configuring along with that hosted and local instances for issue tracking, workflow collaboration, and tool-chain automation.

**Environment: AWS, ELK Stack, Azure, Terraform, Docker, Kubernetes GIT, GitHub, Bash, Python, Maven, SonarQube, Nexus, Jenkins, Chef, Linux, Unix, Apache Tomcat, AppDynamics VMware, Windows, Shell, Pearl, Jira etc.**

Unissant Inc., Herndon, VA April 2022- December 2022

DevOps Engineer

**Description:** Unissant Inc., is an information technology (IT) consulting company that provides a range of services and solutions to government agencies and commercial clients. I worked on developing and implementing the cloud security strategies, policies and guidelines based on industries practices. I have also deployed and configured security controls for cloud services, such as Zero Trust Network Access (ZTNA), micro-segmentation, and just-in-cases.

**Responsibilities:**

* Experienced in managing and configuring the **Google Cloud Platform (GCP)** services like **compute engine**, **cloud load balancing,** **cloud storage**, **cloud SQL,** **Stack driver monitoring** and **loud deployment manager** etc.
* Used **GCP cloud** services like **Compute, auto-scaling** and **VPC (Virtual Private Cloud)** for building and securing the highly scalable and flexible systems which handles expected and unexpected outbursts, Used **GCP App Engine** for deploying and scaling web apps and services developed with **Java.** Optimized the application for speed by using **Google cloud** and **app engine.**
* Set-up and build AWS infrastructure using resources **VPC, EC2, S3, RDS, Dynamo DB, IAM, EBS, Route53, SNS, SES, SQS, CloudWatch, CloudTrail, Security-Group, Autoscaling &amp; RDS** using **CloudFormation templates.**
* Set up **IAM policies** for **users, subnets, NAT** to ensure successful deployment of Web applications, database templates and security groups by creating a custom sized **VPC**, designed by **cloud formation templates**.
* Used **Chef** and created instances in **AWS** and migrated data to **AWS** from the data center, also deployed and monitored the scalable infrastructure on **AWS** and managed servers using **Chef**.
* Used **CloudWatch** in **AWS** for implementing and maintaining the monitoring and alerting of the production, corporate servers and stored in **AWS**. Assigned the **elastic IP addresses** to work around host and availability zone failures by remapping instance address to another running instance.
* Installed, Configured, and managed the **Docker Containers**, **Docker images** for webservers and applications. Implemented **Docker -maven- plugin** to build **docker images** for all the micro-services and used **Docker file** to build the **Docker images** from the **Java jar files.** Created **Docker images** using a **docker file** and worked on **Docker container snapshots**, removing the images and managed the **docker volumes**, virtualized the servers in **docker** QA, configured automation using the **docker containers**.
* Worked on **Docker-Compose, Docker-Machine** to create **Docker containers** for testing applications in theQA environment and automated the deployments, scaling, and management of containerized applications.
* For Orchestration of the deploying, scaling, and managing the **docker containers** used **GKE**. Managed local deployments in **Kubernetes**, created local cluster and deployed application containers. Used **GKE clusters** in the rollback and rollout methods in the deployment strategies. Integrated **Istio** and packages with **GKE clusters** for the **service mesh.**
* Experienced in setting up **GKE clusters** for running **micro-services** into production using **Kubernetes** backed infrastructure and worked on **auto-scaling** of the **cluster** by managing the **node**. Used **ingress controller** for providing the **reverse proxy,** configurable **traffic routing,** and **TLS termination** **for Kubernetes services**, used **K8 Ingress** to configure the **ingress routes** and rules for individual services.
* For scaling the cluster used **Horizontal Pod Auto-scalar (HPA)** and **Vertical Pod Autoscaler (VPA)** to scale the available resources in the container, for scaling the nodes inside of the **cluster** used **Cluster Auto-scaler.** Using **GKE**, created **clusters, pods, replica sets, services, deployments, labels, and health checks** using the **YAML files**. Managed multiple **Kubernetes pod containers** in **scaling and auto scaling. For API keys, Manager for API keys, Password and Certificates** used **Secret manager.**
* Used **Secret** **Manager** for **API** **keys**, **Password**, and **Certificates**. Also, used **Audit** **logging** with **Cloud** **Audit** for logs to generate an audit entry to interact **Secret Manager** and used those logs for abnormal access patterns.
* Engineered support scripts to enhance **Chef** **Recipes** and facilitated **AWS server** provisioning through **Chef Recipes,** optimizing server setup and configuration. Developed custom **Chef Recipes** to seamlessly configure, deploy, and maintain software components within the existing infrastructure, contributing to the robustness and reliability of the system.
* Employed **Chef configuration managemen**t to automate and standardize the deployment of **web applications,** manage **environment configuration** **files**, user configurations, mount points, and software packages. Created **YAML-based cookbooks** to streamline and automate development processes, effectively enhancing the efficiency and consistency of software deployments.
* Orchestrated end-to-end **DevOps** workflows by configuring **Jenkins** **pipelines**, seamlessly connecting code from **GitHub**, and driving build processes using **Maven**. Utilized **SonarQube** for code analysis, ensuring code quality, and streamlined artifact management through **Nexus** **artifactory** to facilitate efficient development and deployment lifecycles.
* Set up and managed **Jenkins** **pipelines** for automated builds and deployments, incorporating pre-release, release, and post-release stages. Integrated **Jenkins** with **GitHub**, incorporating key **plugins** like **SonarQube** for **code** **analysis** and **Maven** for transforming applications into **.jar and .war files**, while effectively maintaining **artifacts** in **Nexus** **artifactory**.
* Developed and customized **Splunk** **reports**, **dashboards**, and **visualizations** to provide actionable insights into system performance and application behavior, enhancing decision-making processes for the team. Configured and maintained **Splunk** **alerts** and **notifications** to proactively identify and resolve critical issues, resulting in a significant reduction in **mean time to resolution (MTTR)** and improved overall system reliability.
* Implemented an integrated monitoring architecture leveraging **Splunk** **Enterprise**, **Dynatrace**, and **Splunk** **forwarder**, effectively streamlining the collection, analysis of application and infrastructure data. Designed and deployed **dynamic** **Splunk** forms to enable real-time data interaction, empowering cross-functional teams to extract valuable insights and troubleshoot performance efficiently.
* Expert in using **MYSQL** server and writing the **SQL** queries for large scale web applications, indexing the database queries. Set up **MongoDB** **client** and wrote queries to validate the data against the **Mongo** **collections** and **MySQL**.
* By using **WebLogic** and **JBoss** servers on **Linux** and **Windows** environment configured, administrated, and troubleshooted on Linux and Windows Environment. Deployed application build for **java** and .**net** applications.
* Utilized **NodeJS** for synchronous and asynchronous file operation and for event driven programming, Installed and Configured **JBoss** server for developer and test environments.
* Expertise in using the **Bash, Python and Ruby** for creating, maintaining, and automating the scripts. Developed **Cron jobs** and **Shell Scripts (Shell, Python)** for automatic installations to extract logs and administrations tasks like file system management, process management, backup and restore.
* Used **ServiceNow** as a ticketing tool to create workflows for the IT service management. Created reports in **ServiceNow** for better tracking and process improvements.

**Environment: GCP, AWS, Terraform, Maven, Nexus, Docker, Kubernetes, Chef, Jenkins CI/CD, GitHub, SonarQube, Splunk, Dynatrace, MYSQL, SQL, MongoDB, WebLogic, Jboss, NodeJS, Bash, Python, Ruby, Shell, ServiceNow etc.**

AVIVA Health Sept 2018 – Aug 2021

DevOps Engineer

**Description:** I was part of the Data, Analytics Platforms (DAP) infrastructure team, where I was responsible for contributing to the development of centralized design patterns that can be used across Life, GI, and other Aviva business areas. I was experienced in integrating various cloud (SaaS and PaaS) solutions as well as on-premises applications.   
I was responsible for configuring and maintaining deployment automation tools.

**Responsibilities:**

* Spearheaded the implementation of sophisticated continuous deployment pipelines leveraging technologies including **AWS EC2, AWS S3, AWS VPC, AWS ELB, AWS RDS, DynamoDB, AWS Lambda, and Auto Scaling,** significantly expediting software releases and reducing manual interventions. Orchestrated seamless integration of deployment tools such as **Ansible, Terraform, and Jenkins**, enabling efficient infrastructure provisioning, configuration management, and automated build deployments.
* For infrastructure, Designed and managed **AWS-based architectures**, utilizing **VPCs**, **subnets**, **NACLs**, and **security** **groups** to establish robust network segmentation and security controls, while optimizing traffic routing with **Route53**. Leveraged **Terraform** and **Ansible** to code the **infrastructure as code (IaaC),** fostering consistency across environments and enabling repeatable deployments of resources such as **EC2** **instances**, **RDS** **databases**, and **Lambda** **functions**.
* For Containerization and orchestration, Utilized **Docker** to create portable and isolated containerized environments, expertly crafting **Docker** **images** and **Docker files** to encapsulate applications' dependencies and runtime environments. Managed **Docker** **registries** to facilitate **versioning** and distribution of **container** **images**, promoting seamless deployment across multiple environments.
* For Automation and scripting, developed automation scripts using **Python** and **Boto3** to interact with **AWS** **services**, achieving efficient management of resources like **EC2 instances, S3 buckets, and DynamoDB tables.** Automated routine tasks using **Bash scripts,** optimizing system administration processes on diverse **Linux distributions**, including **R.**
* Employed **Git** for **version** **control**, ensuring organized code management and effective collaboration within cross-functional teams by utilizing branching strategies and pull requests. Collaborated seamlessly with development teams, using tools like **JIRA** for issue tracking and **Agile** **methodologies** to drive project execution and facilitate timely software releases.
* Configuration Management and monitoring using the **Puppet** to ensure consistent system configurations and application deployments across multiple environments, enhancing stability and reliability. Integrated monitoring tools like **BigPanda** to proactively detect and address issues, ensuring optimal system performance and minimizing downtime.
* For database management, managed **MySQL** databases including provisioning **RDS instances** and optimizing query performance, contributing to data integrity and application efficiency. Designed, implemented, and maintained **Oracle databases** on various platforms, including **Solaris, HP-UX, and Linux**, ensuring robust data storage and retrieval capabilities.
* Automated build processes using **Jenkins and Maven (pom.xml),** promoting continuous integration and reliable build artifacts, for building automation, For Automation Management and for Artifact Management leveraged **Nexus** as a central repository for storing and managing build artifacts, streamlining application delivery and version control.

**Environment: AWS, Docker, Chef, Jenkins, Puppet, EC2, Lambda, Bash, Python, Java, Terraform, Ansible, Maven, Groovy, Git, Kubernetes, BigPanda, Jira, RedHat Linux, Nexus, MySQL etc.**

Santander Bank March 2016 – Sept 2018

DevOps Engineer/ System Admin

**Description:**

**Responsibilities:**

* Configured diverse **Azure Virtual Machine images** to align with the project requirements.
  + Implemented **Azure Load Balancers** with **Virtual Machine Scale Sets (VMSS)** to ensure robust application availability and scalability. Utilized **Azure Monitor** extensively to monitor and manage system performance, setting up proactive alerts for critical metrics such as CPU utilization and storage usage.
  + Managed user identities and access groups using **Azure Active Directory (AAD**), establishing, and maintaining a secure and streamlined access structure. Orchestrated the deployment, maintenance, and troubleshooting of various **Azure services**, including **Azure SQL Database, Azure DNS, Logic Apps, and Service Bus.**
  + Designed and deployed **virtual networks** and **subnets** using **Azure Virtual Network (VNet)**, enhancing security via **Network Security Groups (NSGs)** and **route tables**. Provisioned **Azure Synapse Analytics** for efficient large-scale data management, encompassing cluster configuration, data extraction from **Azure Storage**, and rigorous access controls.
  + Created and managed comprehensive **Network Security Groups (NSGs)**, increasing VM security through meticulous control of inbound and outbound traffic. Collaborated cross-functionally to seamlessly integrate services, optimizing communication across diverse **Azure components**.
* Utilized **Docker** to containerize microservices, streamlining deployment and ensuring consistency across various environments.
* Employed **Ansible** to automate configuration management tasks, simplifying the provisioning and maintenance of virtual machines and cloud resources. Developed custom **Python scripts** to automate operational processes, including log analysis, data extraction, and system health checks.
* Integrated **Grafana** with **Prometheus** to establish a robust monitoring framework, providing real-time insights into key metrics and facilitating proactive issue identification. Created tailored **Grafana dashboards**, offering clear visualizations of performance indicators, resource utilization, and allocation trends.
* Implemented alerting mechanisms in **Grafana** to promptly notify stakeholders about anomalies or performance degradation.

**Environment: Microsoft Azure, Azure Virtual Machines, Azure Load Balancer, Virtual Machine Scale Sets, Azure Monitor, Azure Active Directory, Azure SQL Database, Azure DNS, Logic Apps, Azure Service Bus, Azure Synapse Analytics, Azure Virtual Network, Azure Monitor Alerts, Performance Optimization, Backup and Recovery, Security Configuration, Azure Virtual Network Setup, Subnet Configuration, Network Security Groups (NSGs), Route Tables, Docker, Ansible, Python scripts, Grafana, Prometheus, Grafana dashboards.**