**PRIYANKA K**

**:** 732 810 1291

**:** [vasu@cloudoneinc.com](mailto:vasu@cloudoneinc.com)

***An IT expert with over 8 years of professional experience, and well-versed in agile methodologies and implementations. Proficient in Site Reliability, System Administration, Software Configuration Management (SCM) that spans across Build/Release Management, CI/CD pipelines and virtualization using VMware and Cloud Computing utilizing Amazon Web Services, Microsoft Azure and Google Cloud Platform.***

***PROFESSIONAL SUMMARY:***

* Experience on **AWS**, focusing on high availability, **fault tolerance SaaS** **applications,** and auto-scaling using **Terraform** templates. Along with **(CI/CD)** with **AWS Lambda** and **AWS Code Pipeline** and using **Python** included **Boto3** to supplement automation provided by **Ansible** and **Terraform** for tasks such as encrypting **EBS volumes** backing **AMIs** and scheduling **Lambda** functions for routine **AWS** tasks,
* Experience in **Azure** Development in **Azure net applications, Azure storages, Azure SQL Database, Virtual machines, Fabric Controller, Azure AD,** Azure search and notification hub Aure logic apps.
* Expertise in **Google Cloud Platform (GCP)** cloud by **provisioning compute engine, cloud load balancing, cloud storage, cloud SQL, stack driver monitoring** components using the **Terraform GCP** Foundation modules.
* Specialized in provisioning the **GCP Global HTTPS load balancer routes** the traffic to the **GCP GKE cluster** via **Terraform** modules.
* Leveraged in designing and implementing **Site Reliability Engineering (SRE) principles** in a **cloud** environment to ensure high availability and reliability of critical applications and services.
* Experience in implementing and configuring a comprehensive monitoring stack, including **Prometheus, Grafana, and Alert manager**, to provide real-time visibility into system performance, **Splunk** for comprehensive monitoring and alerting, ensuring real-time visibility into reliability.
* Expertise in detecting network outages and protocol failures by using **Nagios** **monitoring system** and experienced in configuring other **monitoring tools** like **Splunk** and **ELK stack**.
* Proficient **troubleshooter** with the ability to handle live production incidents, debug and troubleshoot complex issues encompassing OS, networking, database, and application challenges in **a cloud-based SaaS environment**.
* Expertise in working with **Terraform** key features such as **Infrastructure as a code,** Execution plans, Resource Graphs, Change Automation and used **Auto**-**scaling** launch configuration templates for launching **Amazon EC2 instances** while deploying **microservices** and also implemented in writing **Terraform** **scripts**, **performing validation, deploying, managing** **AWS** **cloud infrastructure** and involved in setting up **Terraform** **continuous build integration system**.
* Expertise in **DevOps** culture through **CI/CD** tools like source code repositories **GIT, Bit Bucket** Continuous Integration **Jenkins/Bamboo**, binary code repositories for artifacts **Nexus, JFrog** configuration management **Chef**, **Ansible**, **Puppet**.
* Proficient in writing **Helm charts from scratch**, encompassing **Kubernetes deployment** configurations, services, and other resources.
* Experience in **Docker** tools like **Docker** **Swarm** and **Compose**. Manage clustering functionality for **Docker containers** using **Docker Swarm**, running the multi-Docker container applications using **Docker Compose**.
* Expertise in Clusters using **Kubernetes** and worked on creating many **pods**, replication controllers, services, deployments, labels and **health checks**. Proficient knowledge of **Mesos** providing fine-grained resource allocations for **pods** across nodes in a cluster.
* Experience in **migrating** infrastructure and applications from **on-premise** to **Azure, AWS** and **GCP**. Has Experience in automation of code deployment across **AWS**, **Azure**, **Google Cloud**, **VMWare** and **OpenStack**.
* Experience in **Orchestrating** and **migrating** **CI/CD** processes using **Cloud** **Formation** and **Terraform** **Templates** and **containerized** the **infrastructure** using **Docker**, which was set up in **Vagrant**, **AWS** and **VPCs**.
* Experience in integrating **Jenkins** with various **DevOps** **tools** such as **Nexus**, **SonarQube** and used **CI/CD** system of **Jenkins** on **Kubernetes** **container** environment, utilizing **Kubernetes** and **Docker** for the runtime environment for the **CI/CD** system to build, test and deploy.
* Demonstrated expertise in using **DevSecOps** tools such as **Jenkins, GitLab CI/CD, Travis CI, and Circle CI.**
* Implemented **DevSecOps** practices by seamlessly integrating **security checks** and controls into the **CI/CD pipeline,** ensuring that security is considered at every stage of the development process.
* Experience in working with **version control systems** like **GIT, Subversion (SVN)** and using **Source code management** tools **GitHub, GitLab, Bitbucket** including command line applications.
* Extensive experience in using **Maven, ANT, Gradle** as a build tool for the building of deployable **artifacts** (war and jar) from source code.
* Extensively worked on **Chef Roles, Cookbooks, Recipes**, Templates, Resources**,** Attributes and Data bags. Proficient in the setup of **Chef Servers,** workstations and **bootstrapping** infrastructure **Chef Nodes** for configuration management.
* Proficient in **Scripting** languages like **Bash, Ruby, Perl, Python, Power shell, Java and XML Scripting** for automating administrative tasks.
* Experience in installing, configuring, supporting, and troubleshooting **Unix/Linux** **Networking services** and **protocols** like **TCP, SMTP, HTTP, SNMP, NIS, NIS+, OSPF, LDAP, DNS, NFS, DHCP, NAS, FTP and SSH**.
* Expertise in utilizing collaboration platforms such as **JIRA, Confluence, Wiki, and VersionOne** to streamline project management, documentation, and **agile** development processes, enhancing team productivity and communication.

***TECHNICAL SKILLS:***

|  |  |  |
| --- | --- | --- |
| **Cloud Technologies** | **:** | AWS, Azure, GCP, OpenStack |
| * **Containerization/orchestration** | **:** | Docker, Kubernetes, OpenShift, EKS, AKS, Docker Swarm. |
| **CI/CD** | **:** | Jenkins, Bamboo, TeamCity, Hudson |
| **Configuration Management** | **:** | Terraform, Ansible, Chef, Puppet |
| * **Version Control Tools** | **:** | GitHub, GitLab, Bitbucket, SVN, TFS |
| **Scripting** | **:** | BigQuery, Python, Java, NodeJS, Ruby, GoLang, YAML, Bash, shell |
| * **Monitoring Tools** | **:** | Splunk, ELK, Nagios, Dynatrace, CloudWatch, Datadog |
| * **Virtualization Technologies** | **:** | VMware, Windows Hyper-V, Virtual box, Vagrant |
| * **Operating Systems** | **:** | Linux, Solaris, Ubuntu, Windows 2003/2008/2012, CentOS |
| * **Databases** | **:** | Oracle/SQL, MySQL, PostgreSQL, NoSQL, MongoDB, DynamoDB,  AWS RDS, Azure SQL |
| * **Web Servers** | **:** | Apache HTTP 3.x, Nginx, Apache Tomcat |
| * **Build Tools** | **:** | Maven, Ant, Gradle, MS Build |
| * **Code Quality Analysis** | **:** | SonarQube, Veracode |
| * **Ticketing Tools / Project Tracking** | **:** | JIRA, Azure DevOps, ServiceNow |
| * **Repository Management** | **:** | JFrog, Nexus, Artifactory |
| * **API** | **:** | RESTAPI, SOAP |
| **Application Servers** | **:** | WebLogic Application Server 9.x, 10.x, Apache Tomcat 5.x/7.x,  JBoss 4.22.GA, WebSphere 6.x/7.x/8.x |
| * **Networking** | **:** | TCP, SMTP, HTTP, SNMP, NIS, NIS+, OSPF, LDAP, DNS,  NFS, DHCP, NAS, FTP, SSH, AWS Route 53, LDAP |

***WORK EXPERIENCE:***

**Site Reliability Engineer (Lead) January 2023** - **Present**

***VERIZON – Basking Ridge, NJ***

***Responsibilities:***

* Led a team of SRE and DevOps engineers, providing guidance, mentoring, and coaching to foster professional growth, improve team efficiency, and achieve organizational goals.
* Implemented **Site Reliability Engineering (SRE)** best practices to ensure the reliability, availability, and performance of critical systems and applications. Collaborated with cross-functional teams to define and measure **service-level objectives (SLOs)** and **service-level indicators (SLIs)**.
* Ensured the reliability and scalability of the **SaaS platform** by effectively utilizing **Amazon RDS** for database management.
* Designed roles and groups for users and resources using AWS **Identity Access Management (IAM**) and implemented **Multi-Factor Authentication (MFA)** on externally available servers and **AWS Console, Monitoring** and alerting leveraging **Cloud Watch and Cloud Trail.**
* Integrated external APIs and web services into **Groovy** applications, ensuring seamless data exchange and interaction with external systems.
* Managed a **production infrastructure** on **Amazon Web Services** **VPC** utilizing **AWS Code Deploy, S3, EC2, Container Service and CloudFormation including Auto Scaling.**
* Created and managed **subnets** within cloud environments, ensuring efficient resource allocation, IP management, and access controls.
* Implemented customizable configuration values and parameters within **Helm charts** to allow for dynamic deployment and easy customization.
* Developed **AWS** **Lambda** functions in **Python** to automate AWS Cloud Watch Scheduled Events between services like **AWS SNS** and **AWS SES** to deliver notifications regarding **AWS EC2** instances to the team.
* Created functions and assigned roles in **AWS** **Lambda** to run **Python** scripts, and **AWS** **Lambda** using **Java** to perform event-driven processing.
* Created alerting and notification rules within **Prometheus and Alert manager** to proactively detect and respond to anomalies and incidents, ensuring minimal downtime.
* Implemented **migrating** and **deploying** workloads on **Google Cloud Platform (GCP)**. Involved in migrating infrastructure and applications from on-premise to **GCP** i.e., from **AWS** to **GCP** using **Google Cloud Migrate**for **Compute Engine**.
* Used **GCP** **App Engine** for deploying and scaling web applications and services developed with Java.
* Implemented **Infrastructure as Code** using tools like **Terraform** or **Cloud Deployment Manager** to automate and manage **App Engine** environments. Created reusable templates for provisioning and scaling **App Engine** services, reducing deployment time and ensuring consistency.
* Provisioned the **GCP Global HTTPS load balancer routes** the traffic to the **GCP GKE cluster** via **Terraform** modules.
* Used **Terraform** in **AWS Virtual Private Cloud** to automatically set up and modify settings by interfacing with the control layer and responsible for setting up an **Elastic search cluster** using **Terraform** scripts to block all the **Spam** and **Phishing** **attacks**.
* Implemented **NGINX** for load balancing, distributing traffic across multiple servers to ensure optimal resource utilization and high availability.
* Conducted **performance monitoring** and tuning activities on **Apache Tomcat**, optimizing JVM settings and server configurations.
* Diagnosed and resolved issues related to **Apache Tomcat**, collaborating with development and operations teams to address incidents promptly.
* Orchestrated the deployment and configuration of **Splunk** for log management and analysis, ensuring optimal system performance. Developed custom alerts and dashboards to proactively identify potential issues and minimize downtime.
* Automated various infrastructure activities like Continuous **Deployment, Application Server setup, Stack monitoring** using **Ansible playbooks** and Integrated Ansible.
* Created **Docker images** using a **Docker file**, worked on Docker container **snapshots**, **removed images** and **managed Docker volumes.**
* Maintained and developed **Docker images** for a tech stack including **Cassandra**, **Kafka**, **Apache**, and several in-house written **Java** services running in **Google Cloud Platform (GCP)** on **Kubernetes**.
* Managed **Amazon Elastic container registry (ECR)** for storing **docker images** and integrated with **Amazon ECS** for simplifying development to production workflow.
* Managed Container management using **Docker** by writing **Docker files** and setting up the automated build on **Docker** **HUB**. Installed and configured **Kubernetes**, deployed cluster on **AWS** with **Jenkins**, **Docker** pipeline implementation.
* Integrated **Amazon Elastic container registry (ECR)** with **IAM** to get resource-level control of each repository.
* Wrote **groovy scripts** to set up job DSL within **Jenkins** and to develop **Jenkins Pipeline** jobs and to set up upstream and downstreamprojects.
* Involved in **DevOps migration** and **automation** processes for building and deploying systems. Used **Jenkins/Azure DevOps** and implemented **CI/CD pipeline** using **plugins** like conditional build step, deploy to **Git**.
* Integrated **Splunk** into automated workflows to enhance incident response times and minimize manual intervention.
* Developed and documented incident response playbooks, incorporating **Splunk** queries to streamline troubleshooting procedures. Integrated **Big Panda** into incident response workflows, automating the correlation and prioritization of alerts.
* Implemented **version control systems (e.g., Git)** to track changes in **Ab Initio code**, ensuring a collaborative and organized development process in a **DevOps** environment.
* Integrated **OAuth (Open Authorization)** authentication and authorization framework into **DevOps** processes to enhance security and streamline user authentication for cloud and web applications.
* Orchestrated **Single Sign-On** using **OAuth-based protocols and IBM ISAM**, allowing users to access multiple applications with a single set of credentials, enhancing user experience and reducing password fatigue.
* Automated **Java** builds with **Maven** and implemented multiple plugins for Code analysis like Junit, SonarQube, etc. Installed and administered **Artifactory** repository to deploy the artifacts generated by Maven.
* Orchestrated **Istio** upgrades and updates to ensure the platform's stability and compatibility with the evolving microservices ecosystem.
* Used **Ansible** to deploy **ELK** for automating continuous deployment (**CD**) and configured Slave Nodes and deployment failure reporting.
* Deployed **AppDynamics** to monitor and analyze the performance of critical applications, identifying and resolving performance bottlenecks.
* Used **SonarQube** in build system for continuously inspecting the **code quality**, **Datadog**, **Nagios** for monitoring and performed log analysis using **Splunk**, **ELK** stack and created monitoring charts.
* Implemented monitoring tools to track the performance of **RPA bots** and proactively addressed issues to minimize downtime. Conducted regular reviews of **RPA** processes, optimizing scripts to improve execution speed and resource utilization.
* Utilized **Vault** feature to **encrypt** any structured **data file** by using **Ansible**. Also encrypted arbitrary files, binary files using **Ansible.**
* Facilitated seamless integration between **Windows and Unix** environments, ensuring interoperability and data exchange. Administered **Unix-based servers**, performing tasks such as user management, system monitoring, and package installations.
* Used **Shell**, **Python** and **Perl** scripting languages to design the **automation process** in the build process.
* Involved in setting up **JIRA** as a defect tracking system and configured various workflows, customizations and plugins for the **JIRA bug/issue tracker.** Resolved update, merge and password authentication issues in **JIRA**. Monitor scalable infrastructure on **AWS** and configuration management using **Ansible**.
* Integrated **Apache Tomcat** into the middleware stack, collaborating with developers to ensure seamless interaction with other components. Facilitated the configuration of Tomcat connectors for improved connectivity with databases and other backend services.
* Implemented advanced **routing** strategies to streamline **data traffic** and minimize latency in **cloud-based systems**, resulting in enhanced user experiences.

***Environments:*** *DevOps, AWS, GCP, Site Reliability Engineering (SRE), GKE, ECR, EKS, EC2, AWS Lambda, S3, Google*Cloud Migrate, Cloud Formation, Cloud Watch, *Compute Engine, CI/CD, IAM, Docker, Docker HUB, GitHub, GitLab, SCM, Kubernetes, Jenkins, Groovy, SonarQube, JIRA, Cassandra, Kafka, Apache, Maven, Nexus, JFrog Ansible, Nginx, MySQL, Dynamo DB, Elastic Cache, Splunk, ELK, Nagios, Python, Shell, Perl, Windows, Linux.*

**Sr Cloud DevOps Engineer (Lead)****January 2022 - December 2022**

***TD SYNNEX – Clearwater, FL***

***Responsibilities:***

* Built and installed servers through **Azure Resource Manager Templates (ARM) or Azure Portal**. Worked on **Azure VPN-Point to Site, Virtual networks, Azure Custom security, endpoint security and firewall.**
* Implemented **Azure Cloud Services (PaaS and IaaS), Storage, Web Apps, Active Directory, Application Insights, Logic Apps, Data Factory, Service Bus, Traffic Manager, Azure Monitoring, OMS, Key Vault, Visual Studio Online (VSO), Cognitive Services (LUIS) and Azure SQL Database.**
* Led a team for designing, planning and implementing existing **on-premise** applications to **Azure Cloud** (ARM). Configuring and deploying Azure Automation Scripts utilizing **Azure stack** **(Compute, Web and Mobile, Blobs, ADF, Resource Groups, Azure Data Lake, HDInsight Clusters, Azure Data Factory, Azure SQL, Cloud Services and ARM),** Services and Utilities focusing on Automation.
* Used **Azure Cloud principles,** including the configuration of **virtual networks (VNets),** management of service principals, setup of **private endpoints**, management of **route tables**, and utilization of **Azure Container Registry** for efficient container deployments.
* Designed and implemented **Microsoft Azure Solutions** across a range of cloud models like **SaaS, PaaS, IaaS**, and integrated on-premise, on-demand workloads with the **Azure** **public cloud**.
* Managed **SaaS platform security** and compliance, conducting regular audits and assessments to ensure data protection and adherence to industry regulations.
* Involved in setting up **Azure Virtual Appliances (VMs)** to meet security requirements as software-based appliance functions(firewall).
* Configured VM's availability sets using **AZURE** portal to provide resiliency for **IaaS** based solutions and scale sets using **AZURE** **Resource Manager** to manage network traffic.
* Created multiple Jenkins and **Azure** **DevOps** CI/CD Pipelines for our Infrastructure code validation and verification (e.g., **Puppet, Terraform**).
* Created and maintained **Azure DevOps** organizations, self-hosted Build agents and agent pools. Configured security policies and strategies for resources.
* Used **Azure Kubernetes** to provide a platform for automating deployment, scaling, and operations of application containers across clusters of hosts.
* Implemented **Terraform** modules that had version conflicts to utilize during deployments to enable more control or missing capabilities. Managed different **infrastructure resources**, like physical machines, **VMs** and **Docker** containers using **Terraform**.
* Automated thedeployment, configuration, and scaling of **Cassandra clusters** using **Infrastructure as Code (IAC)** tools such as **Ansible, Terraform, or Puppet**.
* Implemented **Blue/Green** deployment strategy by creating new applications that are identical to the existing production environment by using automation frameworks such as **ARM** and **terraform templates.**
* Utilized **Mesos Marathon** and **Zookeeper** cluster environment for application deployments.
* Configured and Integrated **Docker** container orchestration framework using Kubernetes by creating pods, deployments and used Python Scripts to schedule, deploy and manage container replicas onto a node cluster using Kubernetes. Configured **Docker Swarm Cluster** and set up Jenkins inside it to reduce the failover downtime.
* Configured **Kubernetes** (k8s) Clusters for running microservices and **pushed microservices** into production with **Kubernetes** Infrastructure. Development of automation of Kubernetes clusters via playbooks in **Ansible.**
* Managed **Kubernetes** charts using **Helm.** Created reproducible builds of the Kubernetes applications, managed Kubernetes manifest files and managed releases of **Helm** packages.
* Used **OpenShift** platform in managing **Docker containers Kubernetes Cluster**.
* Created a **Jenkins** plan with integration of **Terraform** for the automation of **migration** for **LINUX/WINDOWS** **servers**. Built end-to-end **CI/CD** Pipelines in **Jenkins** to retrieve code, compile applications, perform tests and push build artifacts to **Nexus, JFrog** to orchestrate changes across servers and components.
* Used **Jenkins** as a continuous integration tool to create new jobs, manage required plug-ins, configure the jobs select required source code management tools, **build triggers**, build system and post-build actions, schedule automatic builds and notify the build reports.
* Developed build and deployment scripts using **MAVEN** as build tools in **Jenkins** to move from one environment to other environments for **Continuous integration**.
* Performed upgrades of all the **CICD** applications including **SonarQube**, **nexus**, **JFrog**, **nexus-pro**, **Jenkins**, **Cloud bees**, **SCM** Manager, **Gitlab**.
* Used **GitLab** as **SCM** tool to create repositories and manage them among other users. Stored the entire work in the repositories.
* Involved in integration of **DevSecOps** **pipeline** components to include, using a code repository, an **artifact** repository, security assessment platform, and an **orchestrated integration** and delivery platform to enable automated application building, testing, securing and deployment.
* Worked on **Istio's** integration with other tools and technologies to streamline service mesh operations and monitoring, such as custom dashboards and alerting systems.
* Involved in writing **initiate**, **auth, verify, QueryScan** test cases for **Inspector, SAST, DAST, API** security scanning solutions and deployment with Jenkins pipelines. Integrated **Azure Key vault and Azure Redis** cache into the application.
* Managed **NGINX** configurations for SSL/TLS termination, enhancing security and offloading encryption processes from application servers.
* Integrated **RPA** solutions with external systems and APIs to facilitate seamless data exchange and enhance end-to-end automation capabilities.
* Deployed and configured **Splunk**, **Elastic** **search**, **Log** **stash** and **Kibana** (**ELK**) for log analytics, full-text search, and **application** **monitoring**.
* Converted **.Net application** to **Microsoft Azure Cloud** service project as part of cloud deployment.
* Managed **Apache Kafka** using **Confluent** libraries for **.NET (C#)** and also setting up **Kafka consumers** and producers in .NET environments.

***Environment:*** *Azure Cloud, Azure DevOps, Azure Kubernetes, ARM, Azure Portal, Site Reliability Engineering SRE, CI/CD, VMs, SaaS, PaaS, IaaS, Jenkins, Groovy, Kubernetes, Docker, Docker Swarm, Kafka, Ansible, Terraform, Puppet, Microservices, Helm, Mesos Marathon, Zookeeper*, *Blue/Green, Cassandra, Maven, Nginx, Nexus, JFrog, Splunk, Log Stash, Elastic Search, ELK, SCM, GitLab, Git, SVN, Linux, Windows.*

**Sr. Cloud DevOps Engineer****July 2020 - December 2021**

***MEDTRONIC – Northridge, CA***

***Responsibilities:***

* As a **Sr.** **Cloud** Engineer, I am responsible for **continuous Delivery, Deployment and monitoring**. Activities include build issues, deployment issues for multiple releases like maintenance release, enhancement release, bug fixes, emergency releases.
* Engaged in **AWS** environment, involved in utilizing **Compute Services** (EC2, ECS, EKS), **Storage Services** (S3, Glacier, Block Storage, Lifecycle Management policies), **CloudFormation** (JSON Templates), **Elastic Beanstalk**, SNS, IAM, Route53, **Autoscaling**, **Lambda**, VPC, RDS, Trusted Advisor and **CloudWatch**.
* Managed Cloud Services using **AWS Cloud Formation templates**, which helped developers and businesses an easy way to create a collection of related **AWS** resources provision them in orderly and predictable fashion.
* Configured virtual data center in the AWS cloud to support Enterprise Data Warehouse hosting including **Virtual Private Cloud (VPC)**, **Public and Private Subnets, inbound and outbound Security Groups, Route Tables, Elastic Load Balancer.**
* Designed and orchestrated complex data workflows using **Google Cloud Composer** to automate and schedule data pipelines and tasks.
* Setting up the build and deployment automation for **Terraform** scripts using **Jenkins**.
* Used **Terraform** for automating **VPC's, ELB's**, **Security groups**, **SQS queues**, **S3** buckets and continuing to replace the infrastructure as a code from scratch and Created **Terraform** Scripts for EC2 instances, **Elastic Load balancers** and **S3 buckets**. Also **Terraform scripts** to Automate Instances for Manual Instances that were launched before.
* As part of continuously delivering **Agile** team, develop, test, and deploy Data platform features develop ongoing test Automation using **Chef**, **Python** based framework using **Ansible** to **Setup**/**teardown** of **ELK** stack (**Elastic Search, Log stash, Kibana**)
* Used **chef**, **Data** bags implementation, Coding standards, **Cookbook** documents, testing docs, **chef** **Infra**, **bootstrapping** nodes, creating, and uploading **Chef** **Recipes**, **Chef** **Node** convergence in **Chef** **SCM**.
* Worked with **service mesh** like **Istio** for advanced service discovery of **microservices** running in the cluster.
* Implemented **Micro-services** using **AWS** platform build upon **Spring Boot Services** and created workflows on **Jenkins** for setting up automated pipelines for **CI/CD** with AWS.
* **Integrated Helm charts** **into CI/CD pipelines** for automated deployment and updates.
* Configured **Istio's** traffic management features, including **routing, load balancing, and fault injection**, to improve service resilience and reliability.
* Worked on **HASHICORP** **Vagrant, Terraform, Packer, Consul** to achieve scalable **microservices** infrastructure.
* Designed and Developed **Jenkins** Builds deployments on **Docker** containers. Managing and optimize the Continuous Integration using Jenkins and troubleshoot the deployment build issues using the **trigged logs.**
* Used **OpenShift** to improve application security by putting admin consoles on different **Docker** **containers** accessible only from unusual port numbers.
* Installed **Maven** and implemented Maven builds to **automate artifacts** like jar, war and ear and **managed Nexus, JFrog** to download the **artifacts** during the **build**. Integration of automated build with deployment pipeline.
* Implemented **Stack driver monitoring** to ensure the robustness and performance of the deployed resources.
* Built end to end **CI/CD** Pipelines in **Jenkins** to retrieve code, compile applications, perform tests and push build artifacts to **Nexus** and **Orchestrate** changes across servers and components and configured **SonarQube** for code analysis in **Jenkins**.
* Responsible for building/deploying consistently repeatable deployments to production and non-production environments using **JENKINS** and **BUILD** Pipelines, **MAVEN**, **PowerShell**.
* Set up **CI/CD** pipelines for **Ab Initio** jobs, enhancing the efficiency of code deployment, testing, and **monitoring**.
* Written **Chef Cookbooks** and recipes to provision several pre-production environments consisting of **Cassandra** database installations and several proprietary middleware installations.
* Integrated **Cassandra** schema changes and data migrations into **CI/CD pipelines**, automating database updates and versioning.
* Managed and developed custom **ETL scripts** to automate **data** **extraction** from source systems, transformation within **BigQuery**, and loading into destination data stores. Automated **ETL processes** with **Talend**, integrating them into CI/CD pipelines for efficient data processing and deployment.
* Continuously improved **ETL pipelines**, optimizing data transformation and loading processes for enhanced performance and efficiency.
* Developed build and deployment scripts using **MAVEN** as build tools in Jenkins to move from one environment to other environments and also collaborated in Web designing technologies like **HTML5**, **XHTML**, **JavaScript**, **CSS3**, .**Net**, **jQuery**, **Ajax** and **JSON**.
* Designed and implemented **Site Reliability Engineering (SRE**) principles in a cloud environment to ensure high availability and reliability of critical applications and services.
* Involved with security teams to **integrate** security best practices into **SRE** processes, ensuring compliance with industry regulations (e.g., **GDPR, HIPAA**) and security standards.
* Implemented monitoring and observability solutions to track the health and performance of **GitOps**-driven deployments, using tools like **Prometheus and Grafana**.
* Used **security groups**, network ACL's, internet gateways and route tables to ensure a secure zone for organization in **AWS** Public Cloud and also used **IAM** for creating, managing and deleting roles, users, groups and implemented **MFA** to provide additional security to **AWS** accounts and its resources.
* Involved in **version control systems** like **GIT, Subversion (SVN)** and used **Source code management tools** **GitHub, GitLab.**
* Ensured **Infrastructure as a code** best practice by storing Terraform configurations in version control systems and adhering to coding standards.
* Involved in diagnosing and resolving issues with web servers and Java applications, such as **JBoss** and **Spring Boot**, optimizing performance and enhancing reliability.
* Used **Shell, Python** scripting languages to **automate** deployments.
* Developed **Splunk** queries and dashboards targeted at understanding application performance and capacity analysis and increased data costs capacity and reducing cost.
* Orchestrated the provisioning of a **Google Kubernetes Engine (GKE)** cluster using **Terraform** modules, optimizing the cluster's configuration for scalability and performance.
* Involved in monitoring and troubleshooting Network like **TCP/IP, NFS, DNS and SMTP** in **Linux servers** and System Activities like **CPU**, **Memory**, **Disk** and **Swap space** usage to avoid any performance issues.

***Environment:*** *DevOps, AWS, GCP, Site Reliability Engineering (SRE), Agile, CI/CD, Terraform, Cloud Formation, VPC, Jenkins, Groovy, Ansible, Chef, Microservices, OpenShift, Docker, ELK, IAM, Maven,**Google Kubernetes Engine (GKE), GCP Cloud Engine, GCP App Engine, Kafka, Nexus, JFrog, SonarQube, Cassandra, HTML5, XHTML, JavaScript, CSS3, .Net, jQuery, Ajax, JSON, SCM, GIT, SVN, GitHub, GitLab, Splunk, Unix/Linux, Power Shell.*

**DevSecOps Engineer**  **September 2018 - June 2020**

***KOHL’S – Milpitas, CA***

***Responsibilities:***

* Created **S3 buckets** and managing policies for S3 buckets and Utilized S3 bucket and **Glacier** for storage and backup on **AWS** and provided security and managed user access and quota using **AWS** **Identity and Access Management** **(IAM)**, including creating new Policies for user management.
* Configured **AWS EC2 instances**, **security groups**, and databases in **AWS** using **S3 buckets** that are configured to keep machine backups in **S3 buckets, Glacier**.
* Used multiple **EC2 instances** at the same time and provided highly durable and available data by using **S3 data store**, versioning, lifecycle policies, and create **AMIs** for mission-critical production server’s backup.
* Designed **AWS Cloud formation templates** to create custom sized **VPC, subnets,** **NAT** to ensure successful deployment of **web applications** and **database templates**.
* Collaborated in creating **cloud formation templates** to **provision instances**.
* Involved in creation of the custom **Docker** **container** **images**, tagging and pushing the images and **Docker** consoles for managing the application of life cycle. Conducted auto **deployment** of **software applications** in secure and managed environment using **Docker Container** technology.
* Built **REST APIs** using .NET, facilitating seamless data communication and integration of Apache Kafka.
* Developed and implemented **Software Release Management** strategies for various applications according to the **agile** process.  Installed, Configured and Administered **Hudson/Jenkins** Continuous Integration Tool.
* Debugged and resolved issues in **Groovy code**, pinpointing and fixing problems efficiently.
* Developed **automation framework** for Application Deployments to the **cloud environments**.
* Managed the Private Cloud Environment using **Chef**.
* Performed **Branching**, **Tagging**, **Release** **Activities** on Version Control Tools: **SVN**, **GIT**.
* Used **Maven** as build tool on Java projects for the development of build **artifacts** on the source code. Developed build and deployment scripts using **MAVEN** as build tools in **Jenkins** to move from one environment to other environments.
* Performed and deployed Builds for various Environments like **QA**, **Integration**, **UAT** and Productions Environments.
* **Troubleshoot** and resolved **Build** failures due to infrastructure issues reduced by 95% stabilizing the build process. Setup and executed process to code review system effectively.
* Responsible for defining **branching and merging** strategy, check-in policies, improving code quality, automated Gated Check-ins, defining backup and archival plans.
* **Troubleshoot** Build and Deployment Issues, with little downtime.
* Documented release **metrics**, software configuration process. Used **Maven scripts** to build the source code. Supported and helped to create Dynamic Views and **Snapshot** views for end users.
* Utilized **Okta Multi-Factor Authentication (MFA)** to add an extra layer of security by requiring users to provide multiple forms of verification before granting access to applications.
* Designed and implemented proactive **monitoring** and **alerting systems** using tools like **CloudWatch, Prometheus, Grafana and Datadog** to identify and address performance bottlenecks and service disruptions.
* Developed **Perl** and **shell** **scripts** for automation of the build and release process, developed Custom Scripts to monitor repositories, **Server** **storage**.
* Developing scripts to be executed as checks by **Nagios** for execution in both **Windows** and **Linux** environments.

***Environment:*** *AWS, Cloud Formation Templates, S3 Buckets, Glacier, Jenkins, Groovy, Hudson, Docker, Maven, Chef, Python, GIT, SVN, JBoss, Meta Case, GIT, SVN, CloudWatch, Prometheus, Grafana, Datadog, Perl, Shell, Windows, Linux.*

**DevOps Engineer May 2016 - August 2018**

***HDFC BANK – Hyderabad, India***

***Responsibilities:***

* Developed **Puppet** Modules for installation and auto healing of various tools like **Bamboo**, **Nolio** agents, **MSSQL**, **Nexus** etc., these modules are designed to work on both **windows** and **Linux** platforms.
* Installed and configured an automated tool **Puppet** that included the installation and configuration of the **Puppet master, agent nodes,** and an admin control workstation.
* Created and updated **Puppet manifests and modules, files**, and packages stored in the **GIT** repository.
* Created **Cloud formation template** which will spin up EC2 instance and install **Jenkins** on the EC2 instance.
* Automated the cloud deployments using **Puppet, python** (boto and fabric) and AWS **Cloud Formation Templates.**
* Deployed and managed **Okta’s** identity management platform, ensuring high availability and scalability for customers.
* Installed, configured and administrated **Jenkins CI** tool on **Linux** machines and built Continuous Integration and Continuous delivery environment and used **Nginx** as reverse proxy for securing **Jenkins** using **OPEN SSL**.
* Implemented security measures to protect **Groovy** applications from common vulnerabilities, including input validation and access control.
* Implemented groovy shared libraries in **Jenkins** for making the Jenkins process easily and also implemented the **webhooks**.
* Configured and managed the **SSH**, **SSL**, **NFS**, **FTP** and **SMTP** servers on **LINUX**.
* Involved in implementation team to build and engineer servers for **Linux** and **AIX** operating systems.
* Performed **Linux** **OS** management tasks which include installing packages using rpm and **zypper**, creating soft and hard links, **YaST** for installing and configuration on **SUSE** operating systems.
* Created and wrote **Shell** **Scripts** (**Bash**), **Ruby** and **Power** **Shell** for setting up branching, merging, and automation processes across the environments using **SCM** tools like **GIT**, **SVN** and **TFS** on **Linux** and windows platforms.
* Integrated **Okta's authentication and authorization** services into customer applications, enhancing security and user experience.
* Involved in installing and managing different automation using **Puppet** and monitoring tools on **RedHat Linux** like **Nagios and Splunk**.
* Installed, Configured, Administrated and Supported **RHEL 4, 5.x, 6.x SUSE** **Linux** Enterprise Server (**SLES**) **11.x, 10.x, 11i Solaris 8, 9, 10, 11**, **Windows** Server NT, 2002, 2003, 2007, 2008, 2010.

***Environment:*** *AWS EC2, AWS Cloud Formation,**Puppet, Python, Jenkins, Groovy, Linux, MSSQL, Bamboo, Nolio, Nexus, Open SSL, Nginx, AIX, SUSE, SCM, GIT, SVN, RedHat, Nagios, Splunk, RHEL, Bash, Shell, Ruby, Windows.*

**Linux Administrator**  **September 2015 - April 2016**

***CYIENT – Hyderabad, India***

***Responsibilities:***

* Managed and administered **Windows and Linux-based systems**, ensuring optimal performance, **security**, and **system uptime**.
* Installed operating system on multiple machines using **Kickstart** andModified Kernel parameters to improve the server performance in **Linux**.
* Established configuration files for the **Kickstart** environment, enabling streamlined setup for both **Linux production** and test systems.
* Worked on **Linux Kernel** performance management and optimization.
* Installation, integration and management of data **backup/recovery** solutions. Management and configuration of **VMWare** virtual machines running **Oracle/Sun Solaris X86/64, RHEL, vCenter Server, Dell iDRAC, Client iLO, vSphere ESXi** **and Oracle Linux** server.
* Creating the file systems using **RedHat** volume manager and performing the health check on regular basis for all **Linux** servers.
* Built environments utilizing **TCP/IP, NIS, NFS, DNS, VSFTP,** and **DHCP**.
* Management of **RedHat Linux** user accounts, groups, directories, file permissions and Sudo rules. Packaged software and maintained repositories using **Yum** or **RPM** and **Apt** or **DEB** distribution systems.
* **Monitored** System Activities like CPU, Memory, Disk and Swap space usage to avoid any **performance** issues.
* Responsible for writing/modifying scripts using **bash, SSH** for Day-to-Day administration.
* Utilized **Shell scripting** to automate routine tasks, streamline deployments, and enhance operational efficiency.
* Did a health check of servers after patching / rebooting and **checked** all the **services** of **Linux and Windows server's** whenever we had reboots and code drops.

***Environment:*** *Windows Server 2008/2008R2, Linux, RedHat Linux 5,6.1/6.2, Kickstart, Kernel, Oracle, Sun Solaris X86/64, RHEL, VMware, kickstart, Python, Shell, bash.*