**Name: Jitender Gill** 

**Sr. Cloud/DevOps Engineer**

**Email: jitendergillnew@gmail.com**

**Phone: 732-314-0034**

**PROFESSIONAL SUMMARY**

*Highly motivated and results-oriented DevOps, Cloud, and DevSecOps Engineer with over 9 years of experience in designing, implementing, and automating secure and reliable software delivery pipelines. Possesses a strong understanding of Agile methodologies, CI/CD best practices, and expertise in cloud platforms like AWS & Azure. Proven ability to collaborate effectively with cross-functional teams to optimize efficiency, security, and performance across the software development lifecycle.*

**Reduced deployment time by 70% by implementing a CI/CD pipeline using deployment strategy in a fast-paced environment and saved million of dollars by creating cost optimization and sending reports to the different teams every month. Passionate about automation and improving software delivery efficiency.**

**SKILLS**

|  |  |
| --- | --- |
| **Operating System:** | UNIX, Linux, Windows, Solaris, Ubuntu |
| **Containerization Tools:** | Docker, Kubernetes, Mesos, Openshift |
| **Configuration management:** | Chef, Puppet, Ansible |
| **CI/CD Tools:** | Jenkins, Hudson, Bamboo |
| **Build Tools:** | ANT, Maven |
| **Cloud platforms:** | Microsoft Azure, Amazon Web Services (AWS), Google Cloud Platform (GCP), Open stack, PCF |
| **Application/Web Servers:** | Oracle Web logic Server 11g, Apache Tomcat, Oracle Application Server 10g BEA WebLogic 8.1/9.2, WebSphere, JBoss, Tomcat, IIS |
| **Scripting & Programming Languages:** | Python,Shell Scripting,Bash, Perl, Ruby, Groovy,PowerShell, YAML, HTML, PHP, Java/J2EE, .Net, JSON, NodeJS, GO |
| **Logging & Monitoring Tools** | SPLUNK, ELK, Nagios, Datadog, AppDynamics |
| **Security Scanning Tools:** | Fortify, NexusIQ, Blackduck, SonarQube, JFrog Xray |
| **Artifact Repository:** | JFrog, Nexus Repository Manager (NXRM) |
| **Infrastructure as a code:** | Terraform, Cloud Formation Templets, ARM Templets |
| **Virtualization Platforms:** | Virtual Box, Vagrant, VMware, vSphere Versions |
| **Databases:** | Oracle 10g/11g, Mongo DB, MySQL, SQL, NOSQL |

**PROFESSIONAL EXPERIENCE**

**Client: CarMax July 22 – Present**

**Role: Sr Cloud DevOps Engineer**

**Responsibilities:**

* Part Of the Highly Collaborative operations team to streamline the process of implementing the security confidential **Azure** cloud environment and introduced best practices for remediation.
* Gathering the requirements from the clients about existing applications to apply the security measures. Understand the latest features introduced by Microsoft **Azure** and utilized for existing Business applications.
* Implemented **Azure** Cloud products like Virtual Machines, Storage Accounts, Application. Insights, Automation Accounts, Databases, Networking, and IaaS/PaaS resources.
* Creating, Validating, and reviewing solutions and effort estimates of converting existing workload from classic to ARM based Azure cloud environment and created Azure Services Using **ARM** templates (JSON) and ensured no changes in the present infrastructure while doing incremental deployment.
* Setting up the build and deployment automation for terraform scripts using Jenkins. Used **Terraform** templates to automate the Azure Iaas virtual machines using terraform modules and deployed virtual machines scale sets in the production environment.
* Implemented the utilization of Azure offerings, including **Virtual Machines, Blob Storage, Azure Functions, and Azure Kubernetes Service** (**AKS**). Applied Azure DevOps to construct pipelines by integrating Azure Artifacts and Azure Pipelines. Additionally, configured **Azure Security Center, Network Security Groups, and Azure Active Directory**.
* Constructed and put into action disaster recovery and business continuity strategies through the employment of Azure services such as **Azure Site Recovery and Azure Backup**.
* Constructed **Azure AD** (Active Directory) Application Registration and service principles to enable release pipelines within Azure DevOps service connections, facilitating authentication to Azure resources.
* Executed the deployment of Azure Infrastructure as a Service (IaaS) virtual machines (VMs) and Platform as a Service (PaaS) role instances into protected VNets and subnets, integrated into the Software Development Life Cycle (SDLC) and Agile methodologies.
* Successfully integrated CostOps (Cost Optimization in Operations) into our DevOps practices, leading to significant cost savings and more efficient resource management. Conducted a comprehensive audit of existing cloud resources, implemented cost-saving measures such as right-sizing instances, and utilized reserved instances and spot instances where appropriate. Reduced overall cloud spending by 20% while maintaining high availability and performance of applications. Saved over **1.2million$.**
* Performed provisioning of **IAAS, PAAS** Virtual Machines and Web, Worker roles on Microsoft **AZURE Classic and Azure Resource Manager**. Deployed Web applications on Azure using PowerShell Workflow. And worked on **Azure ExpressRoute** to establish connection from Azure to On-premises datacenter. Working knowledge on Azure Fabric, Micro services, IoT & Docker containers in Azure.
* Installed Docker for setting Azure Container Registry with Docker and Docker-compose and actively involved in deployments on Docker using Kubernetes.
* Created **Azure Kubernetes service** to deploy a managed Kubernetes cluster in Azure and created an AKS cluster in the Azure portal, with the Azure CLI, also used template driven deployment options such as Resource Manager templates and Terraform.
* Developed Terraform along with the Packer and to create custom machine images, and Ansible was used to install the software dependencies once the infrastructure was provisioned.
* Implemented and maintained **security measures in Azure**, focusing on **access control, identity and access management (IAM)**, and **compliance**. I’ve configured **Azure Active Directory (AAD)** for secure user authentication, enforced **multi-factor authentication (MFA)**, and set up **role-based access control (RBAC)** to limit access based on least privilege.
* **Documented cloud best practices and guidelines**, fostering a culture of continuous improvement and collaboration within the team.
* Built and automated CI/CD pipelines in Azure DevOps for .NET and .NET Core applications, enabling faster deployments and minimizing manual interventions. Utilized ARM templates and Terraform to provision and manage Azure resources for the **.NET application**, ensuring consistent and reproducible environments.
* Set up **Azure Monitor** to track the health and performance of applications and infrastructure, setting up custom metrics, logs, and alerts to ensure proactive monitoring and troubleshooting. Mainly focuses on minimizing downtime, optimizing performance, and proactively preventing recurring issues through regular system checks and updates.
* Managed major incidents and escalated issues within Azure and server environments, ensuring rapid resolution and minimal business impact.
* Integrated SonarQube with Azure DevOps pipelines for code quality checks on .NET core codebases, improving code quality and maintainability.
* Advocated for the adoption of automation for repeatability, maintainability, and standard enforcement, successfully operationalizing solutions from proof-of-concept through to production.
* Designed, implemented, and maintained Azure-based solutions to address different business needs, ensuring scalability, reliability, and security.
* Provided architectural and engineering support for Azure infrastructure services, collaborating with OS, Middleware, Database, and Network teams.
* Troubleshoot issues and developed design documents and technical roadmaps for Azure projects.
* Translated business requirements into scalable Azure infrastructure solutions in collaboration with client stakeholders.
* Advised client teams on Azure cloud implementation best practices and developed findings and recommendations.
* Installed software that monitors systems and networks for security breaches and intrusions. Monitor systems for irregular behavior and set up preventive measures. Plan, develop, implement, and update the company's information security strategy.
* Using **Azure Boards**, I’ve set up work items, sprints, and custom workflows to streamline task tracking and agile project management. For **Azure Artifacts**, I’ve configured package management to share libraries and dependencies across teams efficiently..
* Used Azure Data Lake for scalable storage of structured and unstructured data, for big data analytics. Implemented data pipelines in ADF to orchestrate ETL workflows, moving and transforming data between on-prem and cloud, and SQL databases.
* Helped define **Service Level Objectives (SLOs)** by identifying key performance and reliability metrics, such as response times and uptime. We set **Service Level Agreements (SLAs)** based on business requirements, ensuring agreed performance guarantees.
* Worked with **Azure Functions** for serverless computing to run event-driven tasks like processing data streams, automating workflows, and triggering functions based on events in other Azure services.
* Automated alerts by setting up Azure Monitor to track key **performance metrics** and trigger alerts when thresholds are exceeded. For **cost monitoring**, configured alerts in Azure Cost Management to notify when spending approaches set limits.
* Making deployments of microservices as containers by using containerization tools like **Docker** and **Docker Compose**. **Docker** **Swarm** and **Kubernetes** for orchestration and MS Azure to ensure continuous deployments into the different environments.
* Containerized .NET applications using Docker and deployed to Azure Kubernetes Service (AKS) improving scalability and reducing deployment time.
* Used **ArgoCD** for managing and automating Kubernetes deployments, specifically for continuous delivery. It allowed the implementation of GitOps workflows, ensuring that the Kubernetes clusters stay in sync with Git repositories. I configured ArgoCD to automatically deploy applications, roll back in case of failures, and manage multiple environments. Its declarative approach made monitoring application state and maintaining consistency easier across clusters.
* Worked with the Operations team on specific **Ansible** Playbooks used in Jenkins for application installations and related config files to deploy the packages in production NoSQL Databases.

**Client: Precise Software Solutions Inc, Rockville, MD July 2019 – July 22 Role: Sr. Cloud/DevOps Engineer**

**Responsibilities:**

* Created **Azure** Automation Assets, Graphical runbooks, PowerShell runbooks that will automate specific tasks. Deployed Azure AD Connect, configuring ADFS authentication flow, ADFS installation using Azure AD Connect.
* Hands on experience in Azure Development, worked on Azure web application, App services, Azure storage, Azure SQL Database, Virtual machines, Fabric controller, Azure AD, Azure search, and notification hub.
* Deployed Azure **IaaS virtual machines (VMs)** and Cloud services (PaaS role instances) into secure VNets with Azure Internal Load Balancer and subnets
* Provided support for Azure Cloud servers environment for project Code Deployments and Oracle DB installations and maintenance
* Created Azure **PowerShell script** for APIM backup and restore operations. Created several SQL scripts as part of bug fixes to provide immediate solutions for some of the common issues
* Managed Azure Infrastructure Azure Web Roles**,**Worker Roles**,**VM Role, Azure SQL**,**Azure Storage**,**Azure AD Licenses, Virtual Machine Backup and Recover from a Recovery Services Vault using Azure PowerShell and Azure Portal
* Implemented TFS branching and merging operations for .NET Source Code in the Agile Development Methodologies.
* Implemented storage related to blobs/Azure files, SQL/Recovery Services, Managing the storage access, Implement, Integrate Azure Active Directory with AD Migration, Configure Application Access Panel.
* **Terraform** to create, change and improve production infrastructure and maintained versioning infrastructure safely and efficiently by custom in-house solutions.
* Create and maintain highly scalable and fault tolerant multi-tier AWS and Azure environments spanning across multiple availability zones using Terraform and CloudFormation.
* Created Docker images using a Docker file, worked on Docker container snapshots, removing images and managing Docker volumes
* Written scripts to containerize using Dockers and orchestrate it using Kubernetes & Configured the services using modern DevOps tools, like **Ansible**
* Using Ansible as an automation engine for cloud provisioning, configuration management, application deployment, intra service orchestration and multi-tier deployments.
* Communication with team members for both Ansible Core and Ansible Tower teams to clarify requirements and overcome obstacles
* Implementing **Kubernetes** to manage containerized applications using its nodes, Config Maps, Selector, Services and deployed application containers as Pods
* Created private cloud using Kubernetes that supports DEV, TEST, and PROD environments
* Configured Jenkins and pipelines to drive all **microservices** builds out to the Docker registry and then deployed to Kubernetes, Created Pods and managed using Kubernetes
* Wrote and modified MS build targets to publish to daily build location managed MS Build scripts for .Net core builds.
* Implemented Jenkins as Continuous integration tool: creating new jobs, managing required plugins, configuring the jobs selecting required source code management tool, build trigger, build system and post build actions, scheduled builds, notifying the build reports etc
* Worked with **Databricks** for big data processing and machine learning, using its distributed Spark environment for ETL tasks, data transformations, and model training, making large-scale data analytics more efficient.
* Configured Jenkins Monitor, List and Build views to display the status and progress of selected jobs with build version information
* Researched and implemented code coverage and unit test plug-ins with Maven/Jenkins
* Worked on google cloud platform (**GCP**) services like compute engine, cloud load balancing, cloud storage, cloud SQL, stack driver monitoring and cloud deployment manager.
* Used **Datadog** for real-time monitoring and alerting on cloud infrastructure, applications, and logs. It’s great for getting a unified view of system health and identifying performance issues.
* Maintain Chef Servers and management application that can use Service Now (CI) data to bring computers into a desired state by managing files, services, or packages installed on physical or virtual machines
* Provide general support, administration and maintenance of the ServiceNow platform, including ITSM, ITFM and other ServiceNow applications
* Automating the Build Infrastructure for deploying services in dockerized environment using Jenkins, SonarQube, Gradle, Groovy, Job DSL, Docker and Splunk.
* Created and maintained build related scripts developed in ANT, Ruby and Python and worked with development team to migrate Ant scripts to **Maven**
* Implemented Unix/Linux Shell Scripting, Bash, Perl, and Python for monitoring and to automate the build and deployment process
* Worked with Relational (PL/SQL, Oracle), Non-relational databases (Mongo), and document-oriented databases (**MongoDB**) and Oracle RAC
* Developed Perl and shell scripts for automation of the build and release process. Developed automation scripting in Python to deploy some applications

**Client: Travelport, Denver Colorado Nov 2017 – July 2019 Role: Site Reliability Engineer**

**Responsibilities:**

* Experienced in **AWS** Cloud platform and its features which includes EC2, S3, VPC, EBS, ELB, RDS, RDB, Elastic Beanstalk, DynamoDB, RedShift and Route 53, EC2 AMI, EBS Cloud watch, AWS Config, and Auto-scaling, IAM
* Deployed LAMP based applications in AWS environment, including provisioning MYSQL -RDS and establish connectivity between EC2instance and MySQL-RDS via security groups
* Provisioned **AWS S3** buckets for backup of the application and sync this content with remaining s3 backups, by creating entry for AWS S3 SYNC in crontab
* Created customized AMIs based on already existing **AWS EC2** instances by using create image functionality, hence using this snapshot for disaster recovery
* Architected and led migration to AWS and Implemented **AWS Redshift** for an automated media buying network including integration of Tableau for data visualizations
* Implemented and designed AWS virtual servers by **Ansible** roles to ensure deployment of web applications. **Openshift/Ansible** patching on both staging and production environment(clusters, certificates)
* Used Ansible as an automation engine for cloud provisioning, configuration management, application deployment, intra service orchestration and multi-tier deployments
* Orchestrated and migrated CI/CD processes using Cloud Formation and **Terraform** Templates and Dockerized the infrastructure, which was setup in Vagrant, AWS and VPCs
* Used **Kubernetes** to deploy scale, load balance, scale and manage Docker containers with multiple namespace versions.
* Created **Docker** images using a Docker file, worked on Docker container snapshots, removing images and managing Docker volumes
* **Grafana/Prometheus** to monitor the memory usage, memory saturation, CPU saturation, CPU usage of nodes. If there is any memory leak then do some test runs and setup some limits
* Implemented a Continuous Delivery pipeline with Docker, Jenkins and **GitHub** and AWS AMI's, whenever a new GitHub branch gets started, Jenkins, our Continuous Integration server, automatically attempts to build a new Docker container from it, The Docker container leverages Linux containers and has the AMI baked in. Converted our staging and Production environment from a handful AMI's to a single bare metal host running Docker
* Created Clusters using Kubernetes and worked on creating many pods, replication controllers, services, deployments, labels, health checks and ingress by writing YAML files.
* Worked with **RedHat OpenShift** Container Platform for Docker and Kubernetes, used Kubernetes to manage containerized applications using its nodes, ConfigMaps, node-selector, Services and deployed application containers as Pods.
* Managed Kubernetes charts using Helm. Created reproducible builds of the Kubernetes applications, managed Kubernetes manifest files and managed releases of Helm packages
* Provisioned monitoring, metrics, and logging systems on AWS using tools such as SPLUNK, ELK, Sensu
* Managed Amazon Web Services (AWS) infrastructure with automation and configuration management tools Chef, and Ansible
* Worked on writing Jenkins build a pipeline with **Gradle script** and **Groovy DSL (Domain Specific Language**) and integrating **ANT/MAVEN** build scripts with Gradle for the sole purpose of continuous build.
* Implemented multiple high-performance MongoDB replica sets on EC2 with robust reliability
* Implemented Jenkins Code Deploy plugin to deploy to AWS and used to automate the build process and deploy the application to Tomcat server
* Used Jenkins for Continuous Integration and deployment into Tomcat Application Server
* Extensive experience in designing and implementation of continuous integration, continuous delivery, continuous deployment through Jenkins.
* **Splunk** and Cloud Watch in the Amazon Web Services (AWS) environment
* Used CloudFront to deliver content from AWS edge locations to users, allowing for further reduction of load on front-end servers
* Configured an **AWS Virtual Private Cloud (VPC)** and Data Base subnet group for isolation of resources within AWS RDS
* Used Amazon Route53 to manage DNS zones and also give public DNS names to elastic load balancers ip’s
* Used Amazon RDS Multi-AZ for automatic failover and high availability at the database tier for **MYSQL** workloads
* Set up Auto Scaling Groups based on memory and CPU to adapt to unforeseen spikes without having an outage or needing manual intervention.
* Performed Branching, Tagging, Release Activities on Version Control (SVN, GIT)

**Client: Charter communications, Maryland heights, Missouri Oct 2016 – Nov 2017 Role: DevOps Engineer**

**Responsibilities:**

* Created AWS Launch configurations based on customized AMI and use this launch configuration to configure auto scaling groups and Implemented AWS solutions using EC2, S3, RDS, DynamoDB, Route53, EBS, Elastic Load Balancer, Auto scaling groups
* Integrated AWS CloudWatch with AWS EC2 instances for monitoring the log files, store them and track metrics
* Created AWS S3 buckets, performed folder management in each bucket, Managed cloud trail logs and objects within each bucket
* Managing Amazon Web Services (AWS) infrastructure with automation and configuration management tools such as Udeploy, Puppet or custom-built designing cloud-hosted solutions, specific AWS product suite experience
* Lead many critical on-prem data migrations to AWS cloud, assisting the performance tuning and providing a successful path towards Redshift Cluster and RDS DB engines
* Well Versed with Configuring Access for inbound and outbound traffic RDS DB services, DynamoDB tables, EBS volumes to set alarms for notifications or automated actions
* Created Highly Available Environments using Auto-Scaling, Load Balancers, and SQS
* Administered and engineered Jenkins for managing weekly Build, Test and Deploy chain, GIT
  + Integration of Automated Build with Deployment Pipeline. Currently installed Chef Server and clients to pick up the Build from Jenkins repository and deploy in target environments (Integration, QA, and Production)
* Worked ChefDK which takes care creating cookbooks and recipes. Used Ansible for configuring and managing multi-node configuration management over SSH and PowerShell
* Experience supporting Chef Environment with 200+ servers and involved in developing manifests
  + Implemented Chef Recipes for Deployment on build on internal Data Centre Servers. Also re-used and modified same Chef Recipes to create a Deployment directly into Amazon EC2 instances
* Managed Ubuntu Linux and Windows virtual servers on AWS EC2 using Open-Source Chef Server
* Proficient in writing Cloud Formation Templates (CFT) in YAML and JSON format to build the AWS services with the paradigm of Infrastructure as a Code
* Set up and built AWS infrastructure with various services available by writing cloud formation templates in json
* Managed AWS Cloud Formation templates to create custom sized VPC, subnets, NAT and Route53 to ensure successful deployment of Web applications and database templates
* Developed Cloud Formation scripts to build on demand EC2 instance formation
  + Automated the cloud deployments using chef, python (boto& fabric) and AWS Cloud Formation Templates
* Created a microservice environment on the cloud by deploying services as docker containers. We used Amazon ECS as a container.
* Implemented monitoring and logging tools like Dynatrace, SPLUNK
* Setting up SPLUNK monitoring on Linux and windows systems, monitored and tracked SPLUNK performance problems, administrations and open tickets with SPLUNK
* Management service to run microservices on a managed cluster of EC2 instances. Implemented Amazon API Gateway to manage, as an entry point for all the API's
* Actively involved in the DevOps streamlining Process through Jenkins CI and CD Release Automation

**Client: Inside Connect Cable, Louisville, KY Aug 2015 - Oct 2016**

**Role: Automation Engineer**

**Responsibilities:**

* Developed build and deployment scripts using ANT and MAVEN as build tools in Jenkins to move from one environment to other environments.
* Perform Life Cycle Management for ESXi hosts. Worked on capacity planning and management of Virtual machines in the VMware environment
* Worked on installation and configuration of DevOps Automation Tool Puppet
* Installed/Configured/Managed Puppet Master/Agent. Wrote custom Modules and Manifests, downloaded pre-written modules from puppet-forge. Upgradation or Migration of Puppet Community and Enterprise
* Installed Jenkins/Plugins for GIT Repository, Setup SCM Polling for Immediate Build with Maven and Maven Repository (Nexus Artifactory) and Deployed Apps using custom modules through Puppet
* Experience in writing and Distributing modules to puppet agents
* Implemented a Continuous Delivery pipeline with Docker, Jenkins and GitHub and AWS AMI's. Implemented new projects builds framework using Jenkins & maven as build framework tools
* Worked on Integrated JIRA with subversion and Jenkins to automate the weekly build process
* Implemented the setup for Master slave architecture to improve the Performance of Jenkins
* Used the Eclipse as IDE, configured and deployed the application onto WebLogic application server using Maven build scripts to automate the build and deployment process
* Involved in a module where the development for front end included service side technology like NodeJS
* Designed a single page application with AngularJS
* Performed User Interface Design and coding using Java, Spring Boot framework and web technologies
* Streamlined the server implementation by moving to JSON and MongoDB as the backend data model
* Utilized Cassandra, NOSQL database management system for keeping user generated data by upgrading of Couch base and Cassandra cluster.
* Used Java Message Service (JMS) for reliable and Asynchronous exchange of important information for status reports along with Apache Kafka as Queue Server
* Created SQL queries, PL/SQL Stored Procedures, Functions for the Database layer by studying the required business objects and validating them with Stored Procedures using DB2. Also used JPA with Hibernate provider
* Used Spring Data with JPA Repository and JPA, Hibernate annotations for handling ORM CRUD operations
* Used Junit for unit testing & Jenkins for automated build and deploy process

**Client: Vidya Softwares Nov 2014 - May 2015**

**Role: Junior Java Developer**

**Responsibilities:**

* Involved in gathering business requirements directly from the end user and prepared System impact analysis document (SIA) and functionality documents.
* Build Microservices for the delivery of software products across the enterprise
* Responsible for the full software development life cycle (SDLC) using Agile Methodology, participated in Daily stand-up meetings with Scrum Master, communicated with testers and resolved tickets
* Developed RESTful Microservices using Spring REST and MVC and used Swagger for manual testing and documentation of Microservices
* Created a platform as infrastructure with AWS (EC2, RDS, ELB) used Jenkins to run the automated deployments
* Implemented a continuous Delivery pipeline with Dockers, Jenkins and GitHub and AWS AMI
* Developed unit and integration tests for existing Micro services using JUnit, Mockito and Dockers
* Designing and deploying enterprise-wide scalable operations on AWS
* Performed unit testing of applications by developing and applying test cases in JUnit
* Used GIT as a version control

**Education**

Master’s in Science – Computer Science

Troy University (Summer 2017)

B.Tech – CSE

Punjab Technical University(Spring 2014)