

KALRA M

☎ (614) -733-9522

✉ [Kalra.cloudtech@outlook.com](mailto:Kalra.cloudtech@outlook.com)

in [www.linkedin.com/in/mkalram](http://www.linkedin.com/in/mkalram)

---

IT professional with **7+ years** of experience which includes **Cloud/DevOps Engineering**, Build/Release Management, System administration, Linux administration and **Cloud Management** in all phases of SDLC and worked for Various domains including financial, communications, health care and understanding the business needs of the applications.

### **Professional Summary:**

- Extensive experience on **AWS** services and its included components **ELB, EC2, S3, VPC, NACL, Cloud Trials, AMI, SNS, Cloud Formation, Route 53, Network Connectivity troubleshooting, Hybrid Environments and VPC peering.**
- Experience in working in an environment of Amazon Web Services (AWS) products gained good knowledge in Elastic Cloud Computing, **Glacier, Block Storage, Elastic Beanstalk, Amazon Lambda, Virtual Private cloud, Load balancing, Relational Database Service, and Cloud Watch.**
- Expertise in creating **AWS Multi-Factor Authentication (MFA)** for instance **RDP/SSO logon**, worked with teams to lock down security groups and setup specific **IAM profiles** per group utilizing newly released **APIs** for controlling resources within **AWS** based on group or user.
- Experience in provisioning servers/instances using infrastructure as code such as **Terraform**, Code is stored in a private repository and constantly updated by merging in the master node after the admin access.
- Expertise in working with **Terraform** key features such as Infrastructure as a code (**IaaS**), **Execution plans, Resource Graphs, Change Automation** and extensively used **Auto scaling** launch configuration **templates** for launching amazon EC2 instances while deploying **microservices.**
- Working experience on various **Azure** services like Compute (Web Roles, Worker Roles), **Azure NSG, Azure VM's, Azure Websites, Caching, SQL Azure, NoSQL, Storage, Network services, Azure Active Directory, API Management, Scheduling, Auto Scaling, and Power Shell Automation, Azure Office 365, Azure VPN's.**
- Experience in deploying a **Dynamics NAV** database that is exported as a data-tier application files into **Azure SQL** using **SQL Server Manager.**
- Extensive experience in setting up a **GCP(Google Cloud Platform)** Firewall rules in order to allow or deny traffic to and from the **VM's** instances based on specified configuration and used **GCP (Google Cloud Platform)** cloud **CDN** (content delivery network) to deliver content from **GCP(Google Cloud Platform)** cache locations drastically improving user experience and latency.
- Experience in creating **ARM** templates for **Azure** platform and in migrating on premise to **Windows Azure** using **Azure Site Recovery** and **Azure** backups and other **Azure** services.
- Expert in an Orchestration & Migration of **CI/CD** processes using **CloudFormation, Terraform Templates & Containerization** of Infrastructure using **Docker**, which was set up in **AWS**, and **VPC's.**
- Expertise in setting up **Docker** environments **Docker Daemon, Docker Client, Docker Hub, Docker Registries, Docker Compose** and handling multiple images by storing them in containers to deploy.
- Expertise in using **Chef** for server provisioning and infrastructure automation, release automation and deployment automation, Configure files and packages.
- Experience in working on **Jenkins** and **Bamboo** for continuous integration and for End to End automation for all build and deployments and deploy the Spring Boot Microservices to **Pivotal Cloud Foundry (PCF)** using build pack.
- Extensive experience in working on an **OpenStack** environment in creating, managing users, groups, roles, services, domains, managing instances, metadata, snapshots, creating new images, managing and creating networks, routers, subnets, allocation pools, managing **OpenStack swift** objects, time stamps, **OpenStack cinder** volumes, using **OpenStack** heat templates and troubleshooting.

- Experience in creating **Puppet Manifests** & modules to automate system operations. Installed configurations of **Puppet Agent**, **Puppet Master** and deployed **Puppet Dashboard** and **Puppet DB** for configuration management to existing infrastructure.
- Experience in working with **Ansible** Playbooks to automate various deployment tasks and working knowledge on **Ansible** Roles, **Ansible** inventory files and **Ansible** Galaxy.
- Experience in implementing **Ansible** and **Ansible** Tower as Configuration management tool, to automate repetitive tasks, quickly deploys critical applications, and proactively manage change by writing **Python** code by using **Ansible Python** API to automate **Cloud** Deployment Process.
- Experience in Installation and Configuration of **Nexus** repository manager for sharing the artifacts within the company and efficient in using **Nexus** and **Artifact Repository** managers for **Maven** builds.
- Expertise in using **Jenkins pipelines** to drive all microservices builds out to the **Docker registry** and then deployed to **Kubernetes** after the containerization is done, Created Pods and managed using **Kubernetes**.
- Experience in continuing architectural changes to move software system offerings to a distributed and service base architecture utilizing **Docker/Kubernetes** technologies.
- Experience in developing **microservice** onboarding tools leveraging **Python** and **Jenkins** allowing for easy creation and maintenance of build jobs and **Kubernetes** to deploy and services.
- Experienced using different log monitoring tools like **Splunk**, **ELK (Elasticsearch, Log Stash, and Kibana)** to see logs information. Monitor & get the health & security notifications from nodes. Created monitors, alarms & notifications for **EC2** hosts using **CloudWatch**.
- Experience in Monitoring server performance with tools like **Nagios**, **Splunk** and resolved network related issues with manual commands and built **Splunk Cluster** environment with High Availability resources.

### **Technical Skills:**

<b>Operating System</b>	Redhat Linux, Ubuntu, Windows, Mac OS X
<b>Automation and Configuration Management Tools</b>	Jenkins, Chef, Ansible, Ant, Maven, Puppet, Docker, Kubernetes, Terraform, Bamboo.
<b>Cloud Services</b>	Amazon Web Services, Microsoft Azure, GCP, OpenStack, PCF
<b>Version control tools</b>	GIT, GITHUB, Bit Buket, TFS, Subversion (SVN)
<b>Database System</b>	MS SQL Server, MySQL, MongoDB, IBM DB,IIS Server
<b>Application and Web servers</b>	Apache Tomcat, JBoss, Web Logic, Web Sphere
<b>Scripting</b>	C#, Perl, Python, Javascript, Bash shell, Power Shell scripting
<b>Virtualization Technologies</b>	VMware, Windows Hyper-V, Xen, Virtual Box, Power VM
<b>Monitoring Tools</b>	Nagios, Splunk, Cloudwatch, ELK
<b>Programming Languages</b>	C, C++, Java, Python
<b>Web Programming</b>	HTML, Javascript, JQuery, CSS, AJAX

### **Professional Experience:**

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

**Aug 19 – Present**

**U.P.S (Alpharetta, GA)**

#### **Responsibilities:**

As a Sr. Cloud/DevOps Engineer I have involved in providing technical strategic leadership and strategy development for the integration of requirements design and technology.

- Created **Terraform** modules to create custom-sized **VPC**, **subnets**, **EC2** instances, **ELB**, security groups. Worked on tagging standard for proper identification and ownership of **EC2** instances and other **AWS** Services like **Cloud Front**, **cloud watch**, **RDS**, **S3**, **Route53**, **SNS**, **SQS**, **Cloud Trail**.
- Experience on other **AWS** cloud services like **EBS**, **Auto scaling groups**, **Cloud watch**, **IAM** for installing **configuring** and **troubleshooting** on various **Amazon images** for server migration from physical into cloud.

- Worked with the **Terraform** key features like Execution plans, Resource Graphs, Change Automation and wrote **Terraform Templates** for AWS infrastructure to build staging and production environments for the testing department to get the work done in a flow.
- Created backup of **Amazon machine Images (AMIs)** of **EC2** instances using **Packer**, and critical business data for **Disaster Recovery (DR)** and upgrading to new instance types for better performance.
- Implemented Backup & Recovery, **S3 life cycle management**, reduced costs by creating Customized **Alarms**. Managed Billing across several **AWS accounts** and reduced costs by effectively utilizing the resources.
- Integrated **AWS Dynamo DB** using **AWS lambda** to store the values of the items and backup the **AWS Dynamo DB** streams and Automated backup of data in **EBS** and instance store to **AWS S3** buckets and created backup of AMI for mission critical production servers from **AWS CLI** and used AWS Data pipeline to configure data loads from **AWS S3** into **Redshift**.
- Used the blue/green deployment while migrating from the manual monitoring environment to automated monitoring environment and to implement zero downtime **PROD** deployments on **Docker** Service for rolling updates.
- Created and troubleshooted **Kubernetes YAML** files to deploy SCM **CI/CD** applications automatically and Idempotently to reduce time costs for the delivery of the project.
- Used **Kubernetes** to deploy, balance the load of the application in and out request by load balancing, scale, and manage docker containers with multiple name-spaced versions.
- Developed **Kubernetes** templates for various applications like **Cassandra**, **Grafana**, **Kafka** and setting up **Kubernetes Clusters** for running microservices and pushed microservices into production with **Kubernetes** backed Infrastructure.
- Implemented cluster services using **Docker** and **Kubernetes** to manage local deployments in **Kubernetes** by building a self-hosted **Kubernetes cluster** using **Terraform** and **Ansible** and deploying application containers.
- Configured **Kubernetes** Replication controllers to allow multiple pods such as **Jenkins** master server above all the slave nodes of the **Jenkins** slave system in multiple minions.
- Configured and maintained **Jenkins** to implement the **CI** process and integrated the tool with Maven to schedule the builds. Took the sole responsibility to maintain the **CI Jenkins** server.
- Experienced in branching and maintaining the version across the environments using **SCM** tools like **Git**, **GitHub** on **Linux** and **Windows** platforms.
- Virtualized the servers on **AWS** using the **Docker**, create the **Docker files** and version control to achieve Continuous Delivery goal on high scalable environment, used **Docker** coupled with load-balancing **Nginx**.
- Implemented a production ready, load balanced, highly available, fault tolerant, auto scaling **Kubernetes** AWS infrastructure and **Docker** container orchestration to define the workflow.
- Worked on **Ansible Playbooks** with **Ansible roles** and created inventory in **Ansible** for automating the continuous deployment and configured the servers, deploy software, and orchestrate continuous deployments or zero downtime rolling updates.
- Implemented **Ansible Tower** for managing the complex network deployments by addition of the control knowledge and delegation to **Ansible** powered environments.
- Wrote **Ansible** Playbooks with **Python** SSH as the Wrapper to Manage Configurations of **AWS** Nodes and Test Playbooks on **AWS** instances using Python. Run Ansible Scripts to provision Development servers.
- Used the **Ansible** Galaxy repository a shared repository for the roles to download, share and manage the roles so that the jobs can be accomplished according to the assignment of the role modules to each team member.
- Developed **Python**, **Perl** and **shell/bash** scripts for **TC Server/Tomcat** web application servers and automation of the build and release process and to automate certain processes for the smooth process of build tool like Maven.
- Created scripts in **Python** to automate log rotation of multiple logs from web servers. Worked with **Python Libraries** to create Web applications.

- Implemented a **GIT** mirror for **SVN** repository, which enables users to use both **SVN** and **GIT** because the team had the upgradation push of the stages of project in both the repository.
- Assisted developers with establishing and applying appropriate branching so that before the merge can take place the developer team should know the branch properly by labeling conventions using **GIT** source control.
- Set up customized monitoring to migrate from the manual monitoring with the help of tools like **Nagios** & **PNP4Nagios** Graphs for the legacy and the new environments.
- Generated reports using **JIRA** for creating projects, assigning permissions to users and groups for the project & created mail handlers and notifications schemes for **JIRA**.

**Technical Tools:** AWS EC2, Cloud Front, cloud watch, RDS, S3, VPC, SNS, SQS, Cloud Trail, Route 53, Docker, Kubernetes, Terraform, Maven, Jenkins, GIT, SVN, JIRA, Ansible, Nagios, Shell scripting, Python Scripting and Perl scripting.

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

**Aug 18 – July 19**

**Eli Lilly And Company (Indianapolis, IN)**

### **Responsibilities:**

My major duties include planning developing and assisting the migration of client's on-premises infrastructure to Microsoft Cloud (Azure). Design and implement hybrid on premise cloud migration and management of strategy for the new hybrid cloud solution in single and multiple data centers.

- Performed implementation of the **Azure** Operations dealing with **IAAS infrastructure** (**Azure** VMs, Virtual Networking, Azure services, Website Deployments) and deployed application as **PaaS** (Websites, Web Roles and Worker Roles).
- Created **Azure** automated assets, **Graphical runbooks**, **PowerShell** run books that will automate specific tasks. Expertise in deploying **Azure AD** connect, configuring ADFS installation using **Azure AD** connect.
- Created **ARM** templates for **Azure** platform and in migrating on premise to **Windows Azure** using **Azure** Site Recovery and **Azure** backups and other **Azure** services.
- Prepared capacity and architecture plan to create the **Azure Cloud environment** to host migrated **IaaS**, **VMs** and **PaaS role instances** for refactored applications and databases.
- Involved on migrating SQL server database to **SQL Azure** data base using **SQL Azure migration** wizard and used **Python API** to upload agent logs into Azure blob storage.
- Creation and Maintenance of **MS Azure Cloud Infrastructure** and **Virtual Network** between **MS Azure** Cloud and on-premises network for backend communication.
- Developed **Docker Images** to support Development and Testing Teams and their pipelines, **Jenkins** distributed builds, **Selenium** and **JMeter** images, Elasticsearch, Kibana and Logstash (**ELK** & **EFK**).
- Set up build environment integrating with **Git** and **Jira** to trigger builds using **Web Hooks** and **Slave Machines** by integrating **Docker** container-based test infrastructure to **Jenkins** CI test flow.
- Wrote **Chef Recipes** / **Cookbooks** in **Ruby** and automated the installation, build and deployment of Database servers, elastic search engines, and various other web servers.
- Implemented **Chef** recipes for the automated Orchestration of **Cassandra clusters** and worked on upgrading **Cassandra** from the old version 2.0 to 3.0 the newer version.
- Worked on Container management using **Docker** by writing **Docker files** and set up the automated build on Docker Hub and written **Docker Compose** file for **multi container** provisioning and to **build, run, tag** and **publish** a docker container to **Azure Container Registry**.
- Integrated **Jenkins** CI with **GIT** version control and implemented continuous build based on check-in for various cross functional applications and created **GitHub Web Hooks** to setup triggers for commit, push, merge and pull request events.

- Configuring and managing an **ELK stack**, setup the elastic search **ELK Stack** to collect search and analyze log files from across the servers and integration of Application with monitoring tool **New Relic** for complete insight and **proactive monitoring**.
- Experienced using different log monitoring tools like Splunk, **ELK (Elasticsearch, Log Stash, and Kibana)** to see logs information. Monitor & get the health & security notifications from nodes. Created monitors, alarms & notifications for EC2 hosts using CloudWatch.
- Replaced Splunk logging and analytics with an automated **ELK** cluster, increasing data capture capacity and reducing costs and installed and configured **ELK** stack in both legacy and Docker swarm mode and pipelined application logs from App Server to Elastic search through Logstash.

**Technical Tools:** Azure, Graphical runbooks, PowerShell, Python API, Docker, JMeter, ARM templates, Jenkins, GIT, Chef, Cassandra clusters SVN, GitHub, Shell scripting, Python Scripting.

**Role: DevOps Engineer**

**June 16 – June 18**

**Blue Umbrella (New Delhi, India)**

**Responsibilities:**

- Configured **AWS Route53** to manage DNS zones globally, create record sets, DNS failover and health checks of domains, assign domain names to ELB and **CloudFront**.
- Implemented AWS solutions using **EC2, S3, RDS, EBS, Elastic Load Balancer, Auto scaling groups** and maintained access to **AWS** resources through users and groups on an **IAM** console.
- Performed the automation deployments using **AWS** by creating the **IAMs** and used the code pipeline plugin to integrate **Jenkins** with **AWS** and created the **EC2** instances to provide the virtual servers.
- Written **Chef Cookbooks** for various packages. Implemented environments, **roles, data bags** in **Chef** for better environment management. Experience with setting up **Chef Infra Bootstrapping** nodes, creating and uploading recipes, node convergence in **Chef SCM**.
- Written the **Chef Cookbooks** for various packages in the various libraries. Implemented environments, **roles, data bags** in **Chef** for better environment management.
- Used **Chef** for server provisioning and infrastructure automation, release automation and deployment automation, Configure files and commands and packages.
- Implemented zero downtime deployment process in **WebLogic** using **python** and **shell script** and added it to the continuous automated process by using **Jenkins** as a tool.
- Experience in setting up all the **Upstream** and **Downstream** Jobs in **Jenkins**. Worked with the **Jenkins Pipelines** for putting the all the workable tasks in the continuous deployment.
- Developed build and deployment scripts using **ANT** as build tool and automated the build and deploy processes using **Jenkins** to move from one environment to other environments.
- Implemented the zero - downtime deployment of the workflow of the process in **WebLogic** using **python** scripting as the language and **shell script** and automated it using **Jenkins**.
- Developed automation scripting in **Python** (core) using **Jenkins** for the continuous deployment and management of the various applications across **Linux** servers.
- Used **Git** for source code version control and integrated with **Jenkins** for **CI/CD** pipeline, code quality tracking and user management with build tools **ANT** and written **pom.xml** build script.
- Wrote scripts using **ANT** tools and automated the build and deploy process using **Jenkins** to move from one environment to other environments. Also edited the existing **ANT** files in case of **errors**.
- Deployed **Splunk forwarders indexers** and **search heads** to monitor, analyze and visualize the AWS VM's on the **Splunk** dashboard that helps in increasing the cluster performance.
- Headed Proof-of-Concepts (POC) on **Splunk** implementation, mentored and guided other team members on Understanding the use case of **Splunk** and about the monitoring procedure.

**Technical Tools:** AWS Route53, ELB, CloudFront, Splunk, ANT, PowerShell, Python API, Terraform, Jenkins, GIT, Chef, Shell scripting, Python Scripting.

**Role: Build and Release Engineer**

**June 15 – June 16**

**Max Bupa Health Insurance Company Ltd (New Delhi, India)**

**Responsibilities:**

---

- Responsible for installation & configuration of **Jenkins** to support various **Java builds** and **Jenkins** plugins to automate continuous builds and publishing **Docker Images** to the **Nexus Repository**.
- Created artifact documents through the source code and internal deployment in **Nexus** repository. Implemented Disaster recovery project on **AWS** using various DevOps automation for CI/CD.
- Performed the automation deployments using **AWS** by creating the **AWS IAMs** and used the code pipeline plugin to integrate **Jenkins** with **AWS** and created the **EC2** instances to provide the virtual servers.
- Installed and configured **Jenkins** and created parameterized jobs to kick off builds for different environments. Managed the team's source repository through **GIT** and continuous integration system using **Jenkins**.
- Implemented a Continuous Delivery pipeline with **Docker**, **Jenkins**, and **GitHub**. Responsible for supporting various **Java builds** and **Jenkins** plugins to automate continuous builds and publishing **Docker Images** to the **Nexus Repository**.
- Developed automation scripting in **Python** (core) using **Puppet** to deploy and manage **Java applications** across **Linux** servers.
- Used **Git** for source code version control and integrated with **Jenkins** for **CI/CD** pipeline, code quality tracking and user management with build tools **Maven** and written **Maven pom.xml** build script.
- Used **MAVEN** as a build tool for **Java** projects and written Maven **POM** files from scratch for multi-module projects for the development of build artifacts on the source code.
- Designed and implemented **GIT** metadata including elements, labels, attributes trigger and hyperlinks and performed necessary day to day **GIT** support for different projects.
- As working on a **Linux/Unix system** maintained the various servers and Production Support of various applications in **Red Hat Enterprise Linux**, and Windows environment.

**Technical Tools:** Jenkins, Docker, Nexus, AWS, IAMs, GitHub, Python API, Maven, Linux, GIT, Puppet, Python Scripting.

**Role: Linux Administrator**

**May 13 – June 15**

**HDFC Bank (New Delhi, India)**

**Responsibilities:**

---

- Performed Administrative and Management tasks using shell Scripts written in **Bash**, **Python**, **C Shell** and **CRON** tab in Linux to automate the various scheduled tasks.
- Developed **Shell** scripts for automation of the build and release process. Developed automation scripting in **Python** to deploy some of the applications to progress towards the automation of the tasks.
- Administered Linux servers for several functions including managing **Apache/Tomcat** server, **mail server**, **MySQL** databases in both development and production environments.
- **Troubleshooting Backup and Restore** Problems Creation of **LVMs** on **SAN** using **Linux** utilities and troubleshooting Linux network, security related issues, capturing packets using tools such as IP tables, firewall, and **TCP** wrapper and **NMAP**.
- Did the deployment and configuration of the application **Server Tomcat** and deploying **JAVA** and **Web** Applications for the testing team to be tested on time so that the delivery of the product can be done on time to the client.

**Technical Tools:** C Shell, Python, Bash, Linux, MySQL, Apache, WebLogic, Java.