**Mohan Karki**

**Diagram

Description automatically generated**

**Professional Summary:**

* An **AWS Certified DevOps Engineer** having **8+ years** of Experience in IT industry with proven expertise in **CI/CD**, **Automation for Configuration and Provisioning, Build/Release Engineering, Security** and **Software Development** involving **cloud** computing platforms like **Amazon Web Services (AWS), Microsoft Azure,** and **Google Cloud Platform (GCP).**
* Experienced with full Software Development lifecycle (**SDLC),** architecting **scalable** platforms, **database design**, worked on **build** and **release** of quality software, and **Agile methodologies** (**Scrum** and **Kanban**).
* Design, Develop, capture, document architectural best practices for migrating on-premises servers to different Cloud environments mainly **AWS**, **Azure** and **Google Cloud.**
* Experience in building Cl/CD on **AWS** environment using **AWS** **CodeCommit**, **CodeBuild**, **CodeDeploy** and **CodePipeline.**
* Experience in designing and deploying AWS Solutions using other various AWS Services like **VPC, EC2, S3, CloudWatch, Auto-Scaling, EBS, SNS, SQS, Route 53, Elastic Load balancer (ELB), Security Groups, ECS, OpsWork, SSM, DynamoDB, CloudFormation, Step Functions, Cloud Trail**, and Disaster Recovery Services such as **Backup and Restore, Pilot Light, Warm Standby, and Multi-site approach.**
* Experience in implementing and configuring the application by utilizing the various Azure Services such as **Virtual Machines, SQL Database, Azure Storage, Azure Active Directory, Network Virtualization, Content Delivery Network, Azure Data Factory & Storage, Azure Site Recovery, Azure Stack.**
* Implemented and worked with the Azure best DevOps practices for continuous integration and deployment/delivery with available azure services like **Azure Pipelines, Azure Boards, Azure Monitor and Azure Kubernetes Service (AKS).**
* Expertise with working the different services provided by Google Cloud Platforms such as **Cloud Build,** **Compute Engine, Cloud Functions, App Engine, Cloud CDN, Cloud SDK, Cloud Storage, Cloud Bigtable, Cloud Logging, Cloud Monitoring** and **Kubernetes Engine Monitoring.**
* Experience with continuous delivery pipeline **Google Kubernetes Engine, Cloud Source Repositories, Cloud Build,** and **Spinnaker** that automatically triggers build, test and deployment processes.
* Worked on **GKE** **Topology** **diagram** including **master**, **slave**, **RBAC**, **helm**, **kubectl**, **ingress** controllers on GKE diagram including master salve.
* Provided consistent environment using **Kubernetes** for deployment scaling and load balancing to the application from development through **production**, easing the code development and deployment pipeli**n**e by implementing **Docker** containerization.
* Experience in Building end-to-end **CI/CD** pipelines in Jenkins groovy script to retrieve code and push build artifacts to **Nexus** and **JFrog** Artifactory's manages for **Maven** builds.
* Excellent experience on **J2EE design patterns** such as **Business Delegate**, **Service Locator**, **Transfer Object**, **Data** **Access Object**, **Session Facade**, **Singleton** **patterns**.
* Expertise in **configuring** and **administering**, Mail Servers and Web Servers like **Apache** **HTTP** & **Tomcat** in heterogeneous environments and working with **TCP/IP, DNS, DHCP, NFS, NIS, SAMBA, SSH, SSL, SFTP, SMTP, SNMP, SAN, NAS.**
* Experienced in **installing**, **configuring**, **modifying**, **testing** and **deploying** applications on **Apache** **Webserver**, **Nginx** and **Application** **servers** such as **Tomcat**, **JBoss**. Functional usage and deployment of applications in **WebLogic**, **WebSphere** and **Apache** **Tomcat** **Server**.
* Experience with working on Jenkins plug-in management areas and implemented several CI / CD plugins like **AWS EC2**, **build pipeline, Docker, Git, JUnit** and **pipeline** plug-ins.
* Experience in **Terraform** for building, changing, and versioning of Infrastructure. Wrote **Terraform** Templates for **AWS, Azure, and GCP** cloud infrastructure as a code to build from development to testing to production environments.
* Experience in writing **Jenkins** **Pipeline Groovy Scripts** for Continuous Integration and built workflows and used Jenkins for uploading Artifacts into the Artifactory and automated various day-to-day administration tasks by developing **Bash, Ruby, JSON, Perl, PowerShell, and Python Scripts**.
* Experienced in branching, tagging, and maintaining the version across the environments using **SCM** tools like **Git, GitHub, GitLab, Bitbucket, Subversion (SVN), Perforce, and TF**S on Linux and Windows platforms.
* Proficient in Containerization tools and solutions such as **Docker, Kubernetes, OpenShift, GKE, AWS ECS, AKS, Docker Enterprise, Mesos, Helm.**
* Experience with setup, configuration and maintain **ELK stack (Elasticsearch, Logstash and Kibana),** and **EFK stack (Elasticsearch, Fluentd/FluentBit and Kibana), Kafka, Zookeeper**.
* Experience working with the **Kibana, Grafana, and Prometheus** dashboard for the overall build status with drill down features.
* Handled several Docker components **like Docker Images, Docker Engine, Docker-Hub, Docker Compose, Docker Registry, Docker Swarm.**
* Proficient with the configuration automation tools such as **Ansible, Chef, Terraform, Puppet, Packer, Rudder, Salt.**
* Experience in building and deploying the application code using CLI of **Kubernetes** such as **kubectl, Kubespray**, and schedule the jobs using **Kube-scheduler**. Managed K8s charts using **Helm** and created reproducible builds of the Kubernetes applications.
* Hands-on experience on working with System health and performance Monitoring Tools like **Nagios, Splunk, AWS CloudWatch, New Relic, App Dynamics, Zenoss, Zabbix,** and using **EFK Stack (Elasticsearch Fluentd Kibana).**
* Hands-on experience with **Snowflake utilities, SnowSQL, SnowPipe, Big Data** model techniques using **Python**.
* Worked on real time data integration using **Kafka, Spark streaming and HBase**.
* Expertise in using Build Automation and continuous integration tools such as **Gradle, Ant, Maven, Jenkins, AWS CodeBuild, Cloud Build, Bamboo, Circle CI, Travis CI, and Codeship.**
* Experience in Deployment and configuration of Atlassian tool, **JIRA** for both hosting and issue tracking for local instances, workflow collaboration, and bug tracking. Also, have hand-on experience with other collaboration tools like **ServiceNow, Slack, OpsGenie, Stride, Trello, Remedy.**
* Experience with the various testing methods and tools such as **JUnit, SoapUI, Selenium, Cucumber, Mocha, Karma, and Sauce Labs.**
* Design, build and manage the **ELK (Elasticsearch, Logstash, and Kibana)** cluster for centralized logging and search functionalities.
* Proficient with release orchestration tools like **AWS CodePipeline, Spinnaker, UrbanCode Release, OpenMake, and CA CD Director.**
* Hands-on experience with the various analytics and AIOps tools such as **Kibana, New Relic, Datadog, AppDynanics, ElasticSearch, FluentD, Logstash, Sumo Logic, and ITRS.**
* Familiarity with frameworks such as **MLlib, scikit-learn, Torch, TensorFlow, Theano, Caffe**
* Proficienct with the different Security and Code Quality Analysis tools like **SonarQube, Snort, HashiCorp Vault, CyberArk, Veracode, BlackDuck, and Checkmarx SAST.**
* Experience in working with SQL databases like **MySQL, Oracle, SQL Serve**r, and NoSQL databases like **MongoDB, DynamoDB, and Cassandra.**
* Extensive experience in developing scalable, reliable and efficient enterprise applications using **Java**, **Spring**, **Web** **Services**, **RESTful**, **JSF**, **JDBC**, **JSP**, **Servlets**, **EJB**, **JMS**, **XML**, **PL/SQL**, **Unix** **scripting**.
* Strong experience in Spring Framework modules as **Spring core, MVC, AOP (Aspect Oriented Programming), IOC (Inversion of Control), DAO (Data Access Object), IO, JTA, JDBC.**
* Experience in administrating, configuring, managing Red Hat Linux distributions and Virtualization with **VMware ESX server 3.5, 4.0, virtual box, XEN and REDHAT VMWare vSphere ESXi 5.x/4.x, vCenter 5.x/4.x**.
* Excellent organizational and analytical abilities teamed with effective oral and communication skills and ability to work in complex environments.

**Technical Skills:**

|  |  |
| --- | --- |
| **Cloud Environment** | Amazon Web Services (AWS), Microsoft Azure, Google Cloud Platform (GCP), OpenStack |
| **Virtualization** | VMWare, Virtual box, Vagrant |
| **Operating Systems** | Ubuntu, IBM AIX, Windows NT, DEBIAN, Mac OSx, Linux (Red Hat, CENTOS & SUSE), Solaris, HP-UX |
| **Continuous Integration** | Jenkins, Codeship, AWS CodeBuild, Cloud Build, Travis CI, Circle CI, Bamboo |
| **Automation/Configuration Tools** | Kubernetes, Terraform, Ansible, Chef, Puppet, Docker, Kickstart, Vagrant, Packer, GKE, AWS ECS, AKS, Helm |
| **Monitoring & AIOps Tools** | CloudWatch, ELK (Elasticsearch, Logstash, Kibana), Nagios, Splunk, FluentD, Sumo Logic |
| **Source Control Management** | Github, Gitlab, SubVersion, Nexus, BitBucket, Artifactory. |
| **Version Control Tools** | Git, Subversion SVN, TFS, CVS |
| **Database Technologies** | Cassandra, Redis, Snowflake, Oracle MySQL, MongoDB, AWS RDS, DynamoDB, DB2, Sybase, SQL, Postgres, NoSQL. |
| **Security and Quality control tools** | SonarQube, Snort, Veracode, Fortify SCA, Checkmarx SAST |
| **Scripting languages** | Python, Ruby, Bash shell, Power shell scripting, Pearl |
| **Networking/Protocol** | TCP/IP, NIS, NFS, DNS, DHCP, SMTP, FTP/SFTP, HTTP/HTTPS, NDS, Cisco Routers/Switches, WAN, LAN |
| **Bug Tracking and Collabration Tools** | Jira, FindBugs, ServiceNow, Slack, Trello, Remedy, OpsGenie, Agile Central |
| **Application Servers** | Apache, Tomcat, IIS, Jboss, WebLogic, WebSphere |
| **Programming/Web Technologies** | Java, Python, C, C++, Go, .Net, PHP, Golang |
| **Frameworks** | Spring Boot 1.5.6/1.5.2/1.4.6 , Spring 4.3/4.0/3.2.5/3.1/2.0 , Hibernate 5.0.2/4.1.9 , Spring Security 3.2, Django, MLlib, scikit-learn, Torch, TensorFlow, Theano, Caffe |
| **Web Technologies** | IntelliJ 2017.1, STS (Spring Tool Suite) 3.8.3, Eclipse4.6, Edit plus, Notepad ++, TEXT PAD, JBuilder, Net Beans 8.0.2, GitHub 2.12.0 |
| **Design Patterns** | Singleton, Command, Iterator, Observer, Chain of Responsibility, Flyweight, Composite |

**Certifications:**

* AWS Certified **DevOps Engineer** – Professional
* AWS Certified **Solution Architect** – Associate
* AWS Certified **Develope**r – Associate

**Professional Experience:**

**Hilti Group November 2019 – Present**

**Plano, TX**

**Role: DevOps Engineer**

**Responsibilities:**

* Extensively worked **on Jenkins CI/CD** pipeline jobs for end-to-end automation to build, test and deliver artifacts and troubleshoot the build issue during the **Jenkins build process**.
* Worked on **Amazon ECS Cluster** is a group of ECS Container instances where to handles the logic of **scheduling**, **maintaining**, and **handling** scaling requests to these instances.
* Deployed **Kubernetes** clusters on top of **Amazon EC2** instances using **KOPS** and managed local deployments in **Kubernetes** creating local clusters and deploying application containers and building/maintaining Docker container clusters managed by **Kubernetes** and deployed Kubernetes using **helm** charts.
* Implemented **Blue/Green and In-place Deployment** based on the requirementswith **AWS Code Deploy** where new version of application is tested in deployment other than in-place deployment and then traffic is diverted into the latest deployment.
* Run **stateless** and **stateful** services on top of an **Auto Scaling Group**. Supports **zero**- **downtime**, **rolling deployment**, attaching **EBS** volumes and **ENIs**, **load balancing (ELB)**, **health checks, service discovery, and auto scaling.**
* Implemented **Terraform** to manage the **AWS infrastructure** and managed servers using configuration management tools **Ansible**. Execute and accomplish **Lambda** **functions** with **Terraform** and build **Serverless** **apps**.
* Deployed and configured **Elastic search, Log stash and Kibana (ELK**) for log analytics, full text search, application monitoring in integration with **AWS Lambda and X-Ray.**
* Managed **AWS EC2** instances utilizing **VPC, RDS, Auto Scaling, Service catalog, Code as pipeline, CDN, Elastic Load Balancing and Glacier for our Sandbox, Dev, QA, and Production** environments as well as infrastructure servers.
* Created alarms and trigger points in **CloudWatch** based on metric thresholds and monitored the application performance, **CPU** utilization, Disk usage etc.
* Designed AWS **CloudFormatio**n templates to create custom sized **VPC, Subnets, NAT** to ensure successful deployment of web applications and database templates.
* Extensively used Storage service **AWS S3, AWS EFS, AWS EBS, NAS Storage, Azure Blob Storage.**
* Extensively using **AWS Lambda** in integration with **API Gateway, Cognito** for serverless application design and development.
* Worked with **Terraform** to provision the infrastructure in AWS components like **EC2, IAM, VPC, ELB, Cognito, API Gateway, ECS, EKS.**
* Implemented **Kubernetes** to deploy, scale, load balance and manage **Docker** containers with multiple names spaced versions.
* Designed and Developed **ETL pipelines** in and out of data warehouse using combination of **Python and Snowflakes SnowSQL**.
* Extensively worked on the team to migrate **on premise SQL Server** data sets to **AWS Snowflake**.
* Setup the **Chef servers, Chef workstations, Chef nodes** and involved in bootstrapping new infrastructure **Chef nodes** for software configurations management.
* Worked on creation of custom **Docker container images**, tagging and pushing the images. Also, familiarity with **Docker Swarm.**
* Created and configured **HTTP** triggers in the **Azure Functions** with application insights for monitoring and preforming load testing on the application using the **Visual Studio Team Services (VSTS).**
* Deployed **Azure AD** **Connect**, configuring **Active Directory Federation Service (AD FS)** authentication flow, **ADFS** installation using **Azure AD Connect.**
* Installed monitoring tools like **Kafka manager, Kafka-tool** and confluent control center to get real time insights into **Kafka** **cluster**.
* Experience in administration of **Kafka** and Flume streaming using **Cloudera Distribution.**
* Experience using **Kafka** and **Flume** for streaming data transfers from different data sources into **HDFS and HBase**.
* Built real time data pipelines by developing **Kafka producers** and **Spark streaming** applications for consuming.
* Wrote **Kafka** **producers** to stream the data from external rest APIs to **Kafka topics.**
* Involved in administrative tasks that include **Build**, **Design**, **Deploy** of **Azure environment**.
* Worked with **Terraform** Templates to automate the **Azure IaaS** virtual machines using **terraform modules** and deployed virtual machine scale sets in production environment.
* Managed **Azure** Infrastructure **Azure Web Roles**, **Worker Roles**, **VM Role, Azure SQL**, **Azure Storage**, **Azure AD** **Licenses**, **Virtual Machine Backup.**
* Worked on Google cloud platform (**GCP**) services like **Compute Engine**, **Kubernetes Engine,** **Cloud load balancing, Cloud Storage, Cloud SQL, stack driver monitoring, Cloud IAM, Cloud Functions, Firestore, Cloud Armor, BigQuery** and **cloud deployment manager.**
* Worked on Jenkins file with multiple stages like checkout a branch, building the application, testing, pushing the image into **GCR**, Deploying to **QA3** and **QA9**, Acceptance testing and finally **Deploying to Production.**
* Created a continuous integration and delivery (CI/CD) pipeline on **Google Cloud Platform** using **Cloud Build**, and applied the best practices to deploy, canary, and automatically rollback applications to Kubernetes using **Spinnaker**.
* Experienced in designing a **Terraform** and deploying it in cloud deployment manager to spin up resources like Cloud virtual networks, **Compute Engines** in public and private subnets along with Auto Scaler in **Google Cloud Platform (GCP**).
* Used **Stack Driver Monitoring** in **GCP** to check the alerts of the applications that run on the Google Cloud Platform and deployed on **GCP** using Google Cloud Deployment Manager.
* Worked in **Kubernetes** charts using **Helm**. Created reproducible builds of the **Kubernetes applications**, **Kubernetes manifest** files, and releases of **Helm** packages.
* Worked on **Nagios** as a monitoring tool to identify and resolve infrastructure problems before they affect critical processes and worked on Nagios Event handlers in case of an automatic restart of failed applications and services.
* Created quality gates in **SonarQube** dashboard and ensured the pipelines build failed when conditions are not met.
* Worked on creating the **Docker containers**, **Docker container images**, **tagging** and **pushing** the images, and **Docker** **consoles** for managing the application life cycle.
* Integrated **Ansible** and **Ansible Tower** with cloud environment, provided role-based access control, setup job **monitoring**, **email notifications, Scheduled jobs, multi-playbook** workflow to chain playbooks.
* Developed & Implemented **OAuth2** functionality with **Spring** **Security** to secure the **REST** services/**REST** API for **authentication** & **authorization** and **generate** & **validate** the tokens for **REST** clients to protect the **REST services**.
* Used **JIRA** to capture, organize and prioritize issues.
* Worked on writing **Jenkins** build a **pipeline** with **Gradle script** and **Groovy DSL** (Domain Specific Language) and integrating **MAVEN** build scripts.
* Responsible for designing and deploying new **ELK clusters (Elasticsearch, Logstash, Kibana), beats, Kafka, Zookeeper.**
* Leveraged **Oracle Identity Manager** for Identity and Access Management within the company.
* Created **Chef** **Cookbooks** and wrote recipes in **Ruby** **Script** to install and configured infrastructure across environments and automated the process using **Python** **Script**.
* Automated the Build Infrastructure for deploying services in a **Docker** environment using **Jenkins**, **SonarQube, Maven, Groovy, Job DSL, Docker, and Splunk.**
* Sync Elasticsearch Data between the data centers using **Kafka** and **Logstash**. managing **Kafka** Cluster and integrated **Kafka** with **Elastic search.**
* Developed **Microservices** for Java Applications and enabled communication among them using **REST API**.
* Designed and developed **RESTful Web Service layer** and **WSLD.**
* Used Design Patterns like **Singleton, Business Delegate and DAOs** for Data Access layer using **Spring** **JDBC Template** and Stored procedures.
* Used **Spring AOP, Spring IOC frameworks** extensively during the development.
* Wrote unit test cases for Unit Level testing using **JUnit**, **TestSNG, Mockito**.
* Created and configured management reports and dashboards in **Splunk** for application log monitoring.
* Experience on **Splunk** query language and Monitored Database Connection Health by using **Splunk** DB connect health dashboards.
* Improved security by hardening servers like **Apache httpd** and **Tomcat**.
* Developed CI/CD pipeline with **Jenkins** for the **Kubernetes** environment and utilized Kubernetes for the runtime environment for the CI/CD system to Build, Test, and Deploy.
* Experienced with working the databases like **MySQL**, **DynamoDB**, **Cassandra, CosmosDB** and **MongoDB.**
* Set up a code review process using Version Control System (VCS) tools includes **Git, Subversion** and utilized the source code administration customer apparatuses like **GitHub, GitLab, Bitbucket**, and also configured a build system using **Gradle, Maven**.
* Worked with Data Scientists and Product Manager to frame a problem, both mathematically and within the business context.

**Environment**: AWS, GCP, CI/CD, EC2, S3, Terraform, AWS Lambda, Cloud-formation, Server-less, Python, Shell/Bash, JAVA, Jenkins, Git, GitHub, GitLab, Ansible, Ansible Tower, Docker, Kubernetes, Maven, ANT, Confluence, Linux, Cloud-watch, SQS, RDS, Code deploy, ELK, Jira, DynamoDB, RDS, SonarQube, Ansible Vault.

**Premier, Inc January 2019 – October 2019**

**Charlotte, NC**

**Role: Lead Cloud Engineer (AWS/Azure)**

**Responsibilities:**

* Engineered and fully automated **deployment**, **environment**, **network**, and **infrastructure** utilizing **Ansible, Packer, Docker, Kubernetes, Terraform, CloudFormation**, and a range of **AWS and Azure** tools.
* Functioned **Jenkins** pipelines to drive all Micro services builds out to the **Docker** **registry** and then deployed it to **Kubernetes**. Created **Pods** and managed using **Kubernetes**.
* Automated AWS components like **EC2 instances, Security groups, ELB, RDS, IAM** through **AWS CloudFormation** templates and **Terraform** based on the requirements.
* Implemented the various Disaster Recovery strategies provided by AWS such as **Backup and Restore, Pilot-Light, Warm Standby, and Multi-Site approach** based on the situations.
* Implemented continuous integration and deployment system with **Jenkins**, **AWS Code Deploy, Ops Works** and monitored the Infrastructure using **AWS Cloud Watch.**
* Extended On-premises infrastructure storage to **AWS** by using **Storage gateways** and created **S3** buckets, configuring buckets with **permissions, logging, versioning and tagging.**
* Created the different **CloudWatch Event Rules** in order to automate the various tasks and set the set targets to **Lambda, SNS, SQS, Step Functions.** Configured **monitoring**, **log aggregation**, and **alerting** using **CloudWatch, SNS, and S3.**
* Experienced with using the AWS Services like **AWS Config, SSM, Amazon Inspector, AWS Organizations**.
* Used **Kinesis Streams, Kinesis Data Firehose, Kinesis Analytics** for real data-processing especially customer click in website and store that data in **S3** too.
* Experienced in designing and implementing a fully operational solution on **Snowflake Data Warehouse**.
* Worked on version control systems like **Subversion, GIT** by using source code management client tools like **Tortoise SVN, GitBash, GitHub, Bitbucket,** and other command line applications.
* Hands on Experience in migrating Code Repositories from **TFVC** to **TFS-GIT** in **Azure DevOps** and handled code related issues. Migrated all the Build and Release Pipelines to use new **TFS-GIT as** Source Repositories and trained developers on it.
* Worked on Integrating Code scan tools **like SonaQube, Fortify Code Analyzer** on **Azure Devops** Build Server and integrating scan tasks in the Build Pipelines to resolve vulnerabilities and blocker issues related to code.
* Worked on **Azure Site Recovery** and **Azure Backup-Deployed Instances on Azure environments** and collecting data from all Azure Resources using **Log Analytics** and analyzed the data to resolve issues.
* Containerize existing applications into Docker images and running them on **Azure** **Kubernetes** **Service** (**AKS**). Integrating with **Azure** **Devops** for **CI**/**CD**.
* Executed the automation from commit to deployment directly into Azure VM using **Azure DevOps pipeline** for implementing a **CI/CD pipeline**.
* Have knowledge on partition of **Kafka** messages and setting up the replication factors in **Kafka Cluster.**
* Implemented to reprocess the failure messages in **Kafka** using offset id.
* Creating **ARM template**s for Azure Platform and Ability to create scripts using **Azure PowerShell** during for **automation** and **build** process and **Auto Deployments**.
* Installed, configured, and maintained **ELK stack systems**.
* Designed, built, and maintained **Log aggregation, Log parsing, Kibana and Grafana** dashboard.
* Monitored the **Splunk** infrastructure for capacity planning, scalability, and optimization.
* Installed, Configured, Maintained, Tuned and Supported **Splunk Enterprise Server 6.0** and **Splunk Universal Forwarder 6.0.**
* Used **Terraform** to write IAC (Infrastructure as code) to build the Development and Production environment in **Azure Cloud**. Extensively worked with **Active Directory Services, IIS Servers, DNS, and DHCP**. Configured Firewall and security using **SSL, IP chains, IP tables, TCP Wrappers, Tripwire, and port scanners.**
* Automated **Datadog** **Dashboards** with the stack through **Terraform Scripts** and used **CloudWatch** and **Datadog** to monitor real-time granular metrics of all the AWS Services and configured individual dashboards for each resource Agents.
* Efficient experience using build tools like **MAVEN** and **ANT** for the building of deployable artifacts such as **war**, **jar** & **ear** from source code.
* Creating clusters using **Kubernetes** and worked on creating many pods, replication controllers, deployments, labels, health checks and ingress by writing **YAML** files.
* Actively participated in deploying **Kubernetes** **Cluster** on **Azure cloud** with **master/minion architecture** and wrote many YAML files to create many services like **pods, deployments, auto-scaling, load balancers, labels, health checks, Namespaces, Config Map, etc**.
* Worked on **Tomcat, JBoss, Web Logic and Web Sphere Application servers** for deployments.
* Integrated delivery (CI and CD process) using **Jenkins, Kubernetes, Nexus, Chef, Yum, Groovy and puppet.**
* Performed automation tasks for test and dev environments on various Docker components like **Docker Hub, Docker Engine, Docker Machine, Compose and Docker Registry.**
* Wrote **Ansible** playbooks, inventories and created custom playbooks written in **YAML** language, encrypted the data using Ansible Vault, and maintained and troubleshot role-based access control by using **Ansible Tower**.
* Used **Splunk** **Dashboard** responsible for the end-to-end event **monitoring** and **log** **analytics** generated by **Fluentd** of business-aligned applications.
* Worked on **Continuous Integration (**CI) system i.e. Puppet for official nightly build, test and managing change list. Installed Multiple Plugins for smooth Build and Release of pipelines.
* Coordinated developers with establishing and applying appropriate branching, labeling/naming conventions using **Git** source control.
* Experienced in installing, configuring and management in **Ansible Centralized Server** (**Tower**) and creating the **playbooks** to support various middleware application servers, and involved in configuring the Ansible tower as a configuration management tool to automate repetitive tasks.
* Configuring ‘**nginx’** for proxy RESTful API calls to micro-services in **Docker** containers.
* Wrote **Chef** **Cookbook** to install **Tomcat** Server and deploying web application on Tomcat server from **Nexus** **antifactory** to maintain the **continuous** **Deployment**.
* Worked on setting up the **Chef repo, Chef workstations** and **Chef nodes** and Involved in chef-infra maintenance including backup/security fix on **Chef Server**.
* Built Docker container clusters accomplished **by Kubernetes Linux, Bash, GIT**, **Docker**, on **AWS**. Utilized **Kubernetes** and **Docker** for the runtime environment of the CI/CD system.
* Utilized Virtualization technologies like **VMWare, Virtual Box**, and worked with containerizing applications like **Docker, Kubernetes** and worked on **Apache and Firewalls** in both development and production.

**Environment**: AWS, AZURE, Chef, Docker, Ansible, Jenkins, Terraform, Kubernetes, ANT, Maven, Ruby, Shell, Python, WebLogic Server 11g, Load Balancers, WLST, Apache Tomcat 7.x, Virtualization, Configured plug-ins for Apache HTTP server 2.4, Nginx, LDAP, JDK1.7, XML, GitHub, Nagios, Splunk.

**LHC Group May 2017 – December 2018**

**Lafayette, LA**

**Role: Site Reliability Engineer**

**Responsibilities:**

* Automate the builds and deployments for several **AWS** Cloud Solutions in the Digital Platform using **CloudFormation** Templates and Cloud Formation Stacks.
* Used Jenkins to deploy applications and ran unit tests written in **Golang**.
* Experience writing Docker files for deploying **Golang** applications on amazon Linux Operating System and **AWS** cloud platform.
* Used **Ansible** and **Terraform** for creating subnets, security groups, route tables, **IAM** Groups, and **ACLs** for **VPC** creation in the AWS Cloud Environment.
* Wrote **Ansible** Playbooks with Python SSH as the Wrapper to Manage Configurations of AWS nodes and Tested **Playbooks** on **AWS** instances using **Python** and responsible for running Ansible Scripts to Provide Dev Servers.
* Troubleshooting Linux network, security related issues using tools such as **IP tables, firewall, TCP wrappers, NMAP.**
* Performed the deployment of release to, various QA & UAT in Linux environments.
* Integrate static code analysis and used different security scanning tools **SonarQube**, **Fortify** (SAST Scans) and **JFrog** X-**Ray** for code quality in all the builds using **Jenkins**.
* Used **Network Attached storage (NAS**) Storage service to allow storage and retrieval of data from a centralized location for authorized network users and heterogenous clients.
* Participated in problem resolving, change, release, and event management for **EFK stack**.
* Designed, built and managed the ELK (ElasticSearch, Logstash, Kibana) cluster for centralized logging.
* Implemented real time log analytics pipeline using Confluent **Kafka, storm, elastic search, Logstash Kibana, and Greenplum.**
* Used **Kafka** to transfer the data from different RDMS producers and **Greenplum** was the consumer of the **Kafka** data.
* Custom **Bash** **Scripts** to install and run the required software and configurations.
* Created topics on **Apache Kafka** distributed messaging system.
* Wrote **Python** script to send messages/metric from **AppD** to **Google Cloud Platform** (**Google** **PubSub** **Topic**) & from **PubSub** to SLO Dashboard
* Tasks involved OS installation, building and configuring from source packages such as **Apache**, **PHP**, and **MySQL** and setup of **Drupal**. In addition, was involved in configuring high availability cluster environments using **RHEL** Conga.
* Also worked on projects to deploy **DNS** master and slave servers, **HTTP** proxy servers, and **Squid** servers.
* Served as system administrator in a mixed **RHEL/CENTOS/Ubuntu** Linux environment supporting Confidential development infrastructure.
* Managed Docker orchestration and **Docker** containerization using **Kubernetes**. Used **Kubernetes** to orchestrate the deployment, scaling, and management of Docker Containers.
* Developed and integrated new monitoring and testing solutions, ensuring that all company policies and procedures were followed.
* Diagnosed, investigated, and resolved all IT incidents and improved the uptime for the services provided to the end-client.

**Environment**: AWS, RHEL, GCP, Docker, Kubernetes, Centos, Ubuntu, Artifactory, Terraform, Ansible, NMAP, TCP wrappers, Firewall, SonarQube, Jenkins, MySQL, PHP, Apache

**Eastern Illinois University January 2016 – March 2017**

**Charleston, IL**

**Role: Java/J2EE Developer**

**Responsibilities:**

* Designed and coded application components in an **Agile/TDD** environment utilizing a test-driven development and **SDET** approach and pair-programming.
* Experience in Design, Development and implementation of several J2EE frameworks like MVC **Struts1.1/1.2, Spring, Hibernate and EJB2.0.**
* Worked on Angular client-side **JavaScript** **framework**, or similar **Dojo**, **Backbone**, **Ember**.
* Developed DIS web application using **spring MVC**, **Spring IOC**, **Spring JDBC**.
* Experience in configuring and deploy in Web Application using **Web Logic, WebSphere, JBoss, Tomcat and Apache.**
* Utilized **Log4j** logging API for debugging and monitoring purposes.
* Used Core Java **Collections, Exception Handling, Multithreading, Serialization and Java 8 features like Lambda expressions, Stream API, Bulk data operations on Collections** which would increase the performance of the Application, facilitate pipeline processing and Method references.
* Used **J2EEDesignPatterns** like MVC (Model View Controller), Singleton, Factory**, DAO (Data Access Object), DTO(Data Transfer Object) and Service Locator**
* Designed **SPA (Single Page Applications**) using Angular JS concepts of Components, Directives, Services, Promises, Templating, Scope, Dependency Injection and two-way data binding.
* Used $**http** with Angular JS interceptors to make sure the requests were valid and used ng-router to handle nested views and avoid page redirection.
* Implemented the application using Spring Boot framework and handled security using Spring Security for implementing authentication, authorization and Access-Control features.
* Implemented Micro Services using **PCF** (**Pivotal Cloud Foundry**) platform build upon AWS.
* Responsible for creating and Launching Amazon EC2 cloud instances using Amazon Images, ELB, RDS, Auto Scaling groups, Security groups, EBS volume snapshot.
* Migrated complex **Map Reduce** programs **into Apache Spark RDD transformations**.
* Follow TTD (**Test Driven Development**) methodology to write unit tests with **JUnit**.
* Implemented **Hibernate** **Connection** **Pooling**, **HQL**, **Hibernate** **Caching**, **Hibernate** **Transactions**.
* Used Hibernate extensively to have database access mechanism with complex queries through Hibernate Query criteria interfaces and hibernate to deal with the database.
* Designed Android Mobile Applications, Used Android Content Providers for implementing features like sending E-mails, SMS, etc. from the app and designed GUIs for these applications.
* Used JIRA and supporting **CI/CD tools**, and **JIRA** for bug Tracking and Project Management.
* Extensive experience in using Build Automation DevOps tools like, **ANT**, **Maven**, **Artifactory**/**Nexus**, **Anthill** **Pro** and **Hudson**/**Jenkins**, **Bamboo**.

**Environment**: Java, J2EE, JavaScript, HTML, CSS, JSF, QA, DEV, MySQL, DAO, JDBC, JQuery, SQL, PL/SQLEclipse, JavaBeans, UML, JSP, JSON, Shell Scripting and WINDOWS 7, Jenkins, Git, J-unit, Selenium, Maven, QA, DEV, SOAP, WEB LOGIC, JDK

**EKbana Solutions June 2014 – December 2015**

**Kathmandu, Nepal**

**Role: Linux System Administrator**

**Responsibilities:**

* Performed configuration of standard Linux and network protocols, such as **SMTP**, **DHCP**, **DNS**, **LDAP**, **NFS**, **HTTP**, **SNMP** and others.
* Involved in hardening, Linux Server and Compiling, Building and installing Apache Server from sources with minimum modules.
* Build and Deployment of the Linux hardware Nodes to host several **KVM** templates.
* Created **RPM** packages using the **RPMBUILD** and checking the new build packages.
* Administration of **UNIX** servers like **AIX** and **Sun Solaris** in both test and production environment and applied patches
* Troubleshooting Linux network, security related issues, capturing packets using tools such as **IPtables**, **firewall**, **TCP** **wrappers**, **NMAP**.
* Created virtual servers on **VMware** **ESX**/**ESXi** and installed operating system on Guest servers.
* Monitor disk usage with **du**, **df**, **lsblk**, **lvscan, vgscan, pvscan.**
* Installed **Red Hat Enterprise Linux** 5/6/7.x on multiple Dell 2950 Servers and **HPE** **ProLiant** DL360 Gen10 Servers.
* Configured and optimized servers for maximum functionality and availability.
* Experience in troubleshooting, optimizing, and performing tuning of applications, databases, and Linux systems.
* Experience with Nagios monitoring software, writing scripts to extend monitoring capabilities.
* Experience in creating physical volume, logical volume, volume groups using **LVM (Logical Volume Management).**
* Works with advanced engineers on installing and configuring **VMware/ESXi 5.x/6.x Servers for virtualizations.**
* Installed and configured operating systems on physical hardware and a virtualize environment**; (i.e. RHEL5,6, 7/Centos 6,7 Linux**).
* Performing kernel and database configuration optimization, I/O resource utilization on disks.
* Monitoring CPU, memory, physical disk, hardware and software raid, multipath, file systems, network and automated repetitive tasks using Perl and Shell Scripting.
* Performed backup and restore of VMware host server configuration, Virtual Machine backup, snapshots, patch updating, monitoring/reporting, **SSH** and **SNMP** and notification of alerts.

**Environment**: RedHat 5.x, CentOS, Solaris 9,10, Windows, VMware vSphere 3.1/4.x, ESXi 3.1/4.x, TCP/IP, NFS, LVM, DNS, LDAP, Kickstart, Solaris Volume Manager (SVM), Shell, Perl.

**Bidhee Pvt. Ltd April 2012 – May 2014**

**Kathmandu, Nepal**

**Role: Jr. Java Developer**

**Responsibilities:**

* Worked as per **Agile Methodology** (Scrum) to meet customer expectation, timelines with quality deliverables.
* Optimized and cleaned up multiple levels of legacy back-end systems, yielding up to 30% increase in both space- and time- efficiency.
* Developed Single page application and structured the **JavaScript** code.
* Written **ANT** scripts for building application artifacts.
* Requirements gathering Analysis, Design, Business logic implementation and Unit testing, Deployment and Maintenance of the Application.
* Built mock-ups of user screens and prototypes for both web apps and built the pages using **HTML**, **CSS**, **JQuery** and **JSPs**.
* Maintained documentation on internal web applications, processes, and setting up local development environment and assisted in **CVS** to GitHub migration of Java web applications.
* Demonstrated strong knowledge of IT strategies and tactics by effectively maintaining and supporting applications and suggesting many new approaches adopted by top management.
* Used **Spring MVC** module to write **REST** controllers, map **URL's** for the request and present with different views for different use cases.
* Skilled in leading frameworks as **Node**.**js** to build high-quality, scalable, and reusable components and Front-end solution, and Performed client-side validation using JavaScript.
* Handled response data from **RESTFUL** webservices using **XML**, **JSON** and **JQuery** to update UI Interacted with **java** controllers
* Implemented Core Java **OOPS** technology along with Multithreaded environment to resolve the problem of concurrency.
* Also applied multi-threading with Java **Collections** and **Generics**.
* Used **JIRA** ticketing system to keep track of issues and tasks on individuals.
* Demonstrated ability to query large databases using **CRUD** commands within SQL.
* Built the Application using Design Patterns- **Singleton, Factory, DAO & Composite view.**
* Implemented with Core Java with **Multithreading**, **Concurrency**, **File Handling I/O, Generics, Data Structures and Collections, Garbage Collection.**
* Analyzed an existing java web application finding architectural flaws and explaining why it performs badly under some situations.

**Environment**: Core Java, Struts, JSP, Servlet2.5, HTML, Hibernate, JavaScript, JPA, Apache Tomcat, MySQL, JSON, jQuery, Singleton, Factory, DAO, Multithreading.