A close up of a sign

Description automatically generatedLIKHITHA ADULLA

Cloud/DevOps Engineer

Phone: (984) 212-3040

**Professional Summary**

Cloud DevOps Engineer with around 7 years of experience in IT industry with major focus on AWS, Azure and GCP. Experienced in working on DevOps/Agile operation processes and experience in Software Configuration, Code review, Build & Release management, Automation, Containerization.

* Experience in working with **Azure web applications**, **Azure SQL Database**, **Azure Virtual Machines**, **Azure Active Directory**, **Azure search**, **Azure DNS**, **Azure VPN Gateway**, and Notification Hub.
* Experience in designing, configuring, and deploying solutions on **Microsoft Azure** using **ARM Templates.**
* Hands-on experience in designing, configuring, and managing public/private cloud infrastructures utilizing **Amazon Web Services** (**AWS**) including **Auto Scaling**, **EC2 Instances**, **Elastic Load Balancer**, **Elastic Beanstalk**, **S3**, **Glacier**, **Cloud Front**, **RDS**, V**PC**, **Direct Connect**, **Route53**, **Cloud Watch**, **Cloud Formation**, **IAM**, **SNS**.
* Experience in writing **Terraform templates** to maintain and manage various cloud infrastructure like **AWS**/**Azure**/**GCP**.
* Experience in designing **Terraform templates** and deploying it in cloud deployment manager to spin up resources like **cloud virtual networks**, **compute Engines** in public and private subnets along with **autoscaling** in **Google Cloud Platform** (**GCP**).
* Expertise in creating **Kubernetes cluster** with **Terraform templates** and deploy them in different cloud environments and monitoring the health of pods using **Helm Charts**.
* Expertise in setting up **Kubernetes** (**k8s**) **clusters** for running **microservices** and pushed **microservices** into production with **Kubernetes** backed Infrastructure. Worked on automating the **Kubernetes** clusters via playbooks in **Ansible**.
* Experience in using **Docker** including **Docker Hub**, **Docker Engine**, **Docker Images** and **Docker Registry** and used **containerization** to make our application’s platform to be consistent when they are moved into different environments.
* Experience in the **orchestration** of **Docker Images** and **Containers** using **Kubernetes** by creating master and node configuration. Installed **Gluster FS** on-top of **Kubernetes** for dynamic PVC provisioning.
* Expertise in integrating **Terraform** with **Ansible**, **Packer** to create and version the Cloud Infrastructure also in designing, automating, and implementing **Amazon machine images** across the Cloud environment.
* Experience in building out **Server Automation** and **Maintenance** of **Chef** servers by **Cookbooks**, **Recipes**, Roles, environments, **Chef Automate**, **Chef Compliance**
* Hands-on experience in configuring the **Ansible control machines**, **Ansible Playbooks**, **roles**, **modules** and used **Ansible Vault** and **Ansible Tower** to secure sensitive data and control the access of playbooks.
* Experience in branching, tagging, and maintaining the version across the environments using **SCM** tools like **Git**, **GitLab**, **Bitbucket** and **TFS** on **Linux** and **Windows** platforms.
* Experience in **Continuous Integration/Deployment** pipeline for the automatic deployment of artifact applications to the servers or environments in **Jenkins CI** and maintained plugins, **webhooks**, SSH keys and controlling access to **Jenkins server** through matrix-based security option.
* Experience in authoring pom.xml, build.xml files, performing releases with the **Maven** release plugin, and managing artifacts in **Nexus**, **Jfrog Artifactory**.
* Experience in using different log monitoring tools like **Splunk**, **Dynatrace**, **ELK** (**Elasticsearch, Log Stash, and Kibana**) to monitor the logs information and to get the health & security notifications of nodes. Created monitors, alarms & notifications for EC2 hosts using **CloudWatch**.
* Hands on experience in writing **Python**, **PowerShell**, **YAML**, **JSON**, **Shell Scripting**, **Bash** to automate the deployments.
* Experience in Installing, Configuring and Managing the **RDBMS** and **NOSQL** tools such as **MySQL**, **DB2**, **PostgreSQL**, **MongoDB** & **Couchbase, Aurora DB**.
* Expertise in deploying and configuring **JIRA** on both hosted and local instances for issue tracking, workflow collaboration and tool-chain automation
* Experience in all phases of the **Software development life cycle** (**SDLC**) with focus on the build and release of the quality software, in-depth understanding of the principles and best practices of Software Configuration Management in **Waterfall model** and **Agile Methodology** (**Scrum** and **Kanban**).
* Hands on experience in handling services and applications like **DHCP**, **DNS** bind, **SSH**, **SSL**, **JVM**, **Apache**, **Load balancing** HA proxy on **Linux** machines.

**Certifications**

Amazon Web Services Certified Developer – Associate

**Professional Experience**

**Client:** Invitae, Colorado (Aug 2020 – Present)

**Role:** Sr.Cloud DevOps Engineer

**Responsibilities**

* Worked with **AWS** services using **S3**, **RDS**, **EBS**, **Elastic Load Balancer**, and **Auto-scaling groups**, **EC2** instances with optimized volumes and achieved **cloud automation** and deployments using **ansible**, **python**, and **Terraform templates**.
* Configured **AWS** Identity and Access Management (**IAM**) Groups and Users for improved login authentication. Created **AWS RDS**, **Aurora DB** cluster and connected to the database through an **Aurora DB** instance using the **Amazon RDS** Console.
* Worked with **AWS CLI** and **AWS SDK** to manage resources on **AWS** and created **python** script using **API** Calls to manage all resources deployed on **AWS**.
* Designed the data models to use in **AWS Lambda** applications which are aimed to do complex analysis creating analytical reports for end-to-end traceability and definition of Key Business elements from **Aurora**.
* Converted existing **AWS** infrastructure to **Serverless architecture** (**AWS Lambda**, **Kinesis**), deployed via **Terraform scripts**.
* Involved in various aspects and phases of architecting, designing, and implementing solutions in IT infrastructure with an emphasis on **Azure** and **AWS cloud**.
* Made use of **Azure Active Directory** for enabling a large number of policies like access policies, security policies, policies based on the cloud-based machines.
* Created **Azure** **Automation Asset Graphical runbooks**, **PowerShell** runbooks that will automate specific tasks, deployed **Azure AD Connect**, configured **ADFS authentication flow**, **ADFS installation** using **AD Connect**.
* Configured **continuous integration** from source control, setting up build definition within **Team Foundation Server** (**TFS**) and configured **continuous delivery** to automate the deployment of **ASP.NET** Model-View-Controller (**MVC**) applications to **Azure web apps**.
* Worked on various **Azure services** like **Compute** (**Web Roles**, **Worker Roles**), **AZURE Websites**, **Caching**, **SQL Azure**, **NoSQL**, **USQLS**, **Storage**, **Network services**, **Data Factory**, **Azure Active Directory**, **API Management**, **Scheduling**, **Auto Scaling**, and **PowerShell Automation**.
* Managed Clusters with various Servers in **Azure** Cloud Resource Groups and Monitored via remotely run scripts **Azure Data Factory** and **Blobs**.
* Built, implemented and maintained the Infrastructure as code using **ARM templates** and used it to create infrastructure in **Azure** and maintained using **Ansible** and implemented **Terraform** modules for deployment of applications across multiple cloud providers.
* Created **Terraform** **templates** for provisioning **virtual networks**, **subnets**, **VM Scale sets**, **Load balancers** and **NAT rules** and used **terraform graph** to visualize execution plan using the graph command.
* Configured **Kubernetes** provider with **Terraform** which is used to interact with resources supported by **Kubernetes** to create several services such as **Config Map**, **Namespace**, **Volume**, **Auto scaler**, etc.
* Deployed **microservices** based applications on **Azure Kubernetes** by using **AKS**, **Ingress API Gateway**, **MySQL** databases and **Cosmo DB** for stateless storage of external data, and setup reverse proxy Nginx servers.
* Managed **Kubernetes** using **Helm charts**. Created reproducible builds of the **Kubernetes** applications, managed **Kubernetes manifest files** and managed releases of **Helm packages**.
* Utilized **Kubernetes** for the runtime environment of the CI/CD system to build, test deploy.
* Created additional **Docker Slave** Nodes for **Jenkins**, using custom **Docker Images**, and pulled them to **ECR** and monitored using **Prometheus**.
* Implemented **Jenkins** and built pipelines to drive all **microservice** builds out to the **Docker registry** and then deployed to **Kubernetes**.
* Supported Data analytics team with **docker** by integrating with cloud, created custom **Docker Images**, and built various containers integrating **Docker engine** and **Docker Management Platform**, to deploy the code-services oriented environments for scalable applications.
* Experience in configuring and managing the container using **Docker**, run multiple applications by **Docker Compose** and used to **Docker Weave NET** for communication between the Containers.
* Configured **Jenkins** server and built jobs to provide Continuous Automated builds based on polling the **Git** source control system during the day and periodically scheduled builds overnight to support development by integrating **Jenkins**, **Git**, **Maven**, **SonarQube** and **Nexus**.
* Implemented **Chef recipes** to set up the **Continuous Delivery pipeline** with **Jenkins**, **SonarQube**, infrastructure to run these packages and various supporting software components such as **Maven**.
* Deployed & configured **Chef server** & **Chef solo** including bootstrapping of client nodes for provisioning, Managed & configured hundreds of servers using **Chef**. Written several **Chef Cookbooks** & **Recipes** with **Ruby scripting** to automate the installation of WebLogic domain & **JDK** & **customized Recipes** from **Chef Supermarket** to align.
* Involved in integrating **chef cookbooks** into **Jenkins** jobs for **CD** framework, and worked with various custom resources, created roles & environments, and using **chef** handlers for different auto Kickoff Requirement Jobs.
* Installed **Chef Server** on the workstation and bootstrapped the nodes using **Knife** and involved in writing **Chef Cookbooks** and **recipes** to automate the deployment process.
* Installed and configured **Nexus repository** manager for sharing **artifacts** between internal teams and created proxy **Nexus repositories** to speed up the build process.
* Replaced **Splunk** logging and analytics with an automated **ELK** cluster, increasing data capture capacity and reducing costs.
* Designed an **ELK** system to monitor and search enterprise alerts. Installed, configured, and managed the **ELK** Stack for Log management within EC2 / **Elastic Load balancer** for Elastic Search. Monitored performance of the applications and analyzed log information using **ELK** (**Elasticsearch**, **Logstash**, **Kibana**).
* Wrote **Python scripts** to apply the Integration label to all the files which needs manual labelling of files.
* Created and wrote shell scripts **Bash** and **PowerShell** for **automating tasks**.

**Environment:** Git, Maven, Kubernetes, Docker, Chef, Jenkins, AWS, AZURE, PowerShell, Python, Nexus, ELK, Splunk, Nginx, Jira, Maven, Aurora DB, Terraform, ARM

**Client:** Chateaux Software, CT

**Role:** GCP DevOps Engineer (Jan 2020- July 2020)

**Responsibilities:**

* Worked on optimizing infrastructure operations of **Google Cloud Platform** by designing target state architectures for the next generation of **GCP** infrastructure, leveraging the latest services and innovations from **GCP** Work closely with **GCP Platform Engineer** to incorporate solutions into custom designed cloud-based applications.
* Worked on the migration of mobile application from **on-premises** to **GCP** by making the chunk of code into **microservices** using **Google cloud SDK Shell**.
* Configured, supported, and maintained all network firewall, storage, load balancer, operating systems, and software in **Google Compute Engine Instances**.
* Configured **Cloud Storage versioning** and lifecycle policies to and backup files and archive files in Cloud File store which was fully managed by **Network Attached Storage** (**NAS**).
* Designed **Google Deployment Manager** templates to create custom sized **VPC**, setting up firewall rules to ensure successful deployment of **Web based applications** and **database templates**.
* Managed **Google infrastructure** and automation with **CLI** and **API**. Worked on **Inbound** and **Outbound** services with automation of **Ansible**. Deployed multiple resources simultaneously, using **Deployment Manager Templates** in **Google**.
* Worked on Setting up a **GCP** Firewall rules to allow or deny traffic to and from the VM's instances based on specified configuration and used **GCP** cloud **CDN** (**content delivery network**) to deliver content from **GCP** cache locations drastically improving user experience and latency.
* Created **Terraform templates** for provisioning virtual networks, subnets, **VM Scale sets**, **Load balancers** and **NAT rules** and used **Terraform graph** to visualize execution plan using the graph command.
* Managed **Kubernetes** using **Helm charts**. Created reproducible builds of the **Kubernetes applications**, templatize **Kubernetes manifests**, provide a set of configuration parameters to customize the deployment and managed releases of **Helm packages**.
* Responsible for installation & configuration of **Jenkins** to support various **Java builds** and **Jenkins plugins** to automate continuous builds and publishing **Docker Images** to the **JFrog Repository**.
* Implemented **Docker** for convenient environment setup for development and testing. Worked on **Docker container snapshots**, attaching to a running container, removing images, and managing containers.
* Implemented **Jenkins** and built pipelines to drive all microservice builds out to the **Docker registry** and then deployed to **Kubernetes**.
* Involved in **CI/CD** process using **Git**, **Jfrog**, **Jenkins** job creation, **Maven** build and create **Docker image** and use the **docker image** to deploy in **gcloud clusters**.
* Involved in writing **Ansible** **playbooks** for setup and configuration of tools including **Jenkins** and **Artifactory** on remote servers using **REST API** protocol and created **playbooks** for deployments and bug fixes with **Ansible**.
* Implemented **Ansible Tower** for scheduling playbooks and used **Bitbucket repository** to store playbooks and updated multiple **manifests** and customized facts.
* Worked with **Ansible** for **Orchestration** of deployments for various servers and managed **Ansible** **Playbooks** with **Ansible modules**, implemented **CD** automation using **Ansible**, also involved in managing existing **Tomcat Webservers** and automated the configuration of new servers.
* Automated configuration of infrastructure and deployment of web applications on **webservers** **Apache/httpd** and **Java** on **application servers** like **Tomcat** by using **Ansible playbooks**.
* Configured **Nagios** to monitor critical IT infrastructure components including system metrics, network protocols, applications, servers, services, and network infrastructure.
* Monitored servers using **Nagios**, **Dynatrace** also used Net scalar and **Nagios** for **Load Balancing** and Controlling flow of Traffic.
* Implemented **REST** services by **Golang** with **microservices** architecture and developed new **RESTful API** services that work as middleware between applications and third-party **API**s using **Golang**.
* Designed configurable **REST APIs** on demand using **Golang** and worked on numerous go frameworks like gin and revel.
* Involved in setting up **JIRA** as defect tracking system and configured various workflows, customizations, and plugins for the **JIRA** **bug/issue tracker**.
* Written **Jenkins Pipeline Groovy Scripts** for **Continuous integration** and build workflows and used Jenkins for uploading **Artifacts** into **JFrog Repository**.

**Environment:** Linux, Git, Maven, Kubernetes, Docker, Ansible, Jenkins, GCP, Golang, webservers Apache/httpd and Tomcat, Nagios, Jira, Bitbucket

**Client:** ValueLabs, HYD

**Role:** DevOps Engineer July 2017 – Nov 2019

**Responsibilities**

* Configured **AWS Cloud Formation templates** to deploy complete stack in multiple accounts.
* Designed **AWS Cloud Formation templates** to create custom sized **VPC**, **subnets**, **NAT** to ensure successful deployment of **Web applications** and **database templates**.
* Used multiple **EC2** instances at the same time and provided highly durable and available data by using **S3 data store**, **versioning**, **lifecycle policies**, & created **AMI**s using packer for mission-critical production server’s backup.
* Responsible for managing **Ubuntu**, **Linux** and **Windows virtual servers** on **AWS** **EC2** instances.
* Managed version control tools like **GIT** enterprise and automated current build process with **Jenkins** with proposed branching strategies to accommodate code in various testing cycles.
* Integrated **GIT** into **Jenkins** to automate the code check-out process and configured the components from **GitHub** into **Jenkins server**. Responsible for automated scheduled build/emergency build and release using **Maven** scripts for enterprise application (**J2EE**).
* Responsible for **Continuous Integration** (**CI**) and **Continuous Delivery** (**CD**) process implementation-using **Jenkins** along with **Python** and **Shell** scripts to automate routine jobs.
* Providing **continuous improvement** to **agile software development** teams by working with **Jenkins** under the **CI/CD pipeline**. Integrated **Maven**, **Nexus**, **Jenkins**, **Git**, **Confluence** and **JIRA**.
* Written and implemented **Perl** script to modify **LDAP** database.
* Configuring, maintaining network and troubleshooting such as **LDAP**, **DNS**, **NIS**, **NFS**, **Web**, **Mail**, **FTP** and **SNMP**.
* Provisioning and Configuring the **Docker containers** and creating **Docker files** for different environments.
* Implemented **Docker-maven-plugin** and **Maven POM** to build **Docker Images** for all **microservices** and later used **Docker file** to build the **Docker Images** from the java jar files.
* Created **JIRA** issues to prioritize and act on important events and stayed up to date with the updates of the project. Involved in setting up **JIRA** as defect tracking system and configured various workflows, customizations, and plugins for the **JIRA bug/issue tracker**.
* Worked on **Nginx webserver** and maintained access logs, error logs and maintained different vhosts.

**Environment:** Nginx webserver, Maven, Linux, AWS, Jira, AWS cloud Formation, Git, Jenkins, LDAP, J2EE,

**Client:** Apollo Healthcare, HYD

**Role:** Build and Release Engineer Sep 2015 – June 2017

**Responsibilities**

* Worked on **Release management**, **Environment Management**, **Continuous deployments**, **Continuous integration**.
* Provided support for more than 5 different applications for **Configuration Management** and build, designed and **deployed for production** and lower environments.
* Used **ANT** build tool for deploying scripts and deploy processes using **Jenkins** to move from one environment to other.
* Administration/Maintenance experience of **source control management** systems, such as **Git**. Managed project dependencies by creating parent-child relationships between projects.
* Installed and administered repository to deploy the **artifacts** generated by **ANT** and to store the dependent jars, which were used during the build.
* Used **Nexus repository manager** to share the artifacts by configuring the repository manager.
* Worked on **continuous integration** tool like **Jenkins** for **End-to-End automation** for all build and deployments.
* Built and deployed **Java/J2EE** to **Tomcat Application servers** in continuous integration process and automated the whole process implementing a **CI/CD** using **Jenkins**.
* Installed and configured the packages using YUM and RPM package managers.
* Developed automated processes that run daily to check disk usage and perform cleanup of file systems on **Unix/Linux** environments using **shell scripting** and **Cron**. Developed unit and functional tests in **Python** and **Ruby** and developed and maintained **Shell scripts** for **build and release tasks**.
* Assisted on monitoring servers and responding to event notifications (Service outages, Load Alerts etc.) using **Nagios**.

**Environment:** Git, Tomcat Application Servers, Jenkins, ANT, Linux, Nagios, Nexus, shell scripting

**Client:** ICICI Bank, HYD

**Role:** System administrator July 2014 – Aug 2015

**Responsibilities**

* Hands-on experience in Installation, configuration and administration of **RedHat/SuSE Linux** and **Solaris Operating System** and provided support of various applications on **SUN Solaris**, **Red Hat Linux** and **Windows environment**.
* Worked on **Vblock Infrastructure** to perform virtualization using **VMWare ESX**, **vSphere** on **RHEL/SUSE** servers.
* Responsible for Installing, configuring and administering **Application server** **WebLogic 8.1** on **Solaris** and Windows platform for **Deployment and Production environments**.
* Worked on **VMware ESXI 5.5** Host Server Installation, Configuration Maintenance, **VSphere** Suite with P2P and P2V, **Virtual Machines Installation**, **Configuration Maintenance**.
* Monitored the **Linux scripts** written in **Python** and **Perl** regularly and conducted troubleshooting steps, tested and installed the latest software on server for end-users.
* Experienced in working with **Network Engineers** to install, manage, and configure **NAS based storage** and insure overall system and **network security**.
* Troubleshooted **Linux network**, security related issues, capturing packets using tools such as **IP tables**, **firewall**, and **TCP wrapper** and **NMAP**
* Monitored and troubleshooted network like **TCP/IP**, **NFS**, **DNS** and **SMTP** in **Linux Servers**.

**Environment:** RedHat Linux, Solaris, Linux, Windows Apache Tomcat, TCP/IP, VSphere, VMware ESXI 5.5, IP addressing & Subnetting, routing

**Technical Skills**

|  |  |
| --- | --- |
| Cloud Services | Amazon Web Services, Microsoft Azure, Google Cloud Platform |
| Container Tools | Docker, Docker swarm, Kubernetes |
| Version Control Tools | GIT, GITHUB, SVN, Bitbucket. |
| Configuration Management tools | Puppet, Chef, Ansible. |
| Web/Application Servers | JBOSS 4.x/5.x, Apache Tomcat 5.x/7.x, Web Sphere (4.0, 5.0), Web Logic (8/9/10). |
| Monitoring Tools | Nagios, Splunk, ELK, CloudWatch, Dynatrace. |
| Ticketing Tools | SonarQube, Jira, ServiceNow, Junit, JMeter |
| CI & Build Tools | Bamboo, CircleCI, Hudson, Jenkins, Maven, Ant, Gradle |
| Scripting Languages | Bash, Shell, Ruby, Python, Golang, Perl, JavaScript, PowerShell, JSON, Groovy, YAML. |
| Database | My SQL, Oracle, Dynamo DB, Mongo DB, Cassandra. |
| Repository Management | Nexus, Jfrog |
| Virtualization | VMware ESX/ESXi, Windows Hyper-V, Power VM, Vagrant, Virtual box. |
| Operating System | Red Hat Linux 7/6/5/4, Ubuntu 18/17/16/14, Fedora, CentOS, Windows 10, Solaris 11/10/9/8 |
| Networking/protocol | TCP/IP, DNS, NFS, ICMP, SMTP, DHCP, OSPF, UDP, and RIP. |