### Image result for aws developer associate examAZHER M

### Sr. Cloud/DevOps Engineer | azhermoha77@gmail.com | 571-278-9164

PROFESSIONAL SUMMARY :

Dedicated to optimization, with 8+ years of experienced DevOps Engineer in architecting/automating and optimizing machine critical deployments over large infrastructure. Proficient with multi cloud (AWS, Azure, GCP), configuration management tools, containerization, orchestration, Infrastructure as a code, developing CI/CD pipelines, monitoring tools, security operations and source code management, configuring databases and application server configuration etc. Team oriented worker able to collaborate with coworkers in various developments to complete projects in a timely manner and heavily participated in on-call rotation.

EDUCATION**:**

* Bachelor of Science in Information Technology - Osmania University
* Master of Science in Information Technology - University of Potomac

CERTIFICATION**:**

* AWS – AWS Certified Developer- Associate
* Puppet – Puppet certified Professional
* Linux – Certified Linux Administrator

EXPERIENCE**:**

**Client name: Nutrien ag Solutions, Loveland, Co. Dec 18 - present**

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

* Creating the automated **build** and **deployment** process for **application**, **re-engineering** setup for better user experience, and leading up to building a **continuous integration** system for all our products.
* Deployed the initial Azure components like **AZURE Virtual Networks, Azure Application Gateway, Azure Storage** and Affinity groups and configured **AZURE** Multi-Factor Authentication as a part of **Azure** AD Premium to securely authenticate users.
* Implemented Azure cloud services, Azure storage, Azure active directory, Azure Service Bus. Create and manage Azure AD tenants and configure application integration with Azure AD.
* **Created Continuous Integration /Continuous Deployment Pipelines in Azure** DevOps environments by providing their dependencies and tasks. Configured and deployed **Azure Automation Scripts** utilizing Azure stack Services and Utilities focusing on Automation.
* Building and Installing servers through **Azure Resource Manager** (ARM) Templates. Have used service manager provider API which uses **Terraform** to create certificates on the **Azure server**.
* Devised **Terraform** to deploy the infrastructure necessary to formulate development, test, and production environments. Configured **Azure VPN gateway** and established Site-to-Site connectivity, Express Route with On Premise Networks and **Azure**.
* Customized **Kubernetes cluster** with objects like **Pods**, **Deployments**, **Services** and **Config Maps** and created reproducible builds of the Kubernetes applications, enhanced Kubernetes **manifest files** and **Helm packages** and implemented Kubernetes to deploy **scale**, **load balance** and developed **Docker containers**.
* Integrated additional **Docker Slave Nodes** for **CI/CD(Jenkins)** using custom Docker Images and Worked on all major components of Docker like **Docker Daemon**, **Hub**, **Images**, **Registry**, **Swarm** etc.
* Configured Ansible control machine, executed tasks using Ad-hoc commands, Ansible Playbooks (written in **YAML language**), **dynamic inventories**, roles, and modules. Used Ansible Vault and **Ansible Tower**, to secure sensitive data and control the access of playbooks. Implemented IT orchestration using Ansible to run tasks in a sequence which can work on different servers.
* Managing **IAM accounts (with MFA)** and IAM policies to meet security audit & compliance requirements and efficiently handled periodic exporting of SQL data into Elasticsearch.
* Deployed Azure IaaS virtual machines (VMs) and Cloud services (PaaS role instances) into secure VNets and subnets.
* Exposed Virtual machines and cloud services in the VNets to the Internet using Azure External Load Balancer.
* Set up **Continuous Integration** (CI) for major releases in **Jenkins** worked on Upstream and Downstream Jobs in Jenkins.
* Provisioned instances, storages, monitoring services and **CI/CD pipeline** through **Jenkins** Closely worked with developers, project managers and product owners to setup the road map and supported day to day build issues.
* **Python** allowing for easy creation and maintenance of build jobs and **Kubernetes** deploy and services.
* Building/Maintaining Docker container clusters managed by Kubernetes, Linux, Bash, GITlab, Docker on Azure and used **Kubernetes and Docker** for the runtime environment of the **CI/CD** system **to build, test deploy.**
* Deploying and maintaining multi-container applications through **Docker**. Worked on **Docker container snapshots**, attaching to a running container, removing images, managing directory structures, and managing containers.
* Worked on creation of **Docker** **containers** and **Docker** **consoles** for managing the application life cycle, worked on setting up the automated build on **Docker HUB.**
* Created pipelines for deploying code from **GitHub** to **Kubernetes** (**K8s**) cluster in the form of **Docker** **containers** using **Spinnaker** platform.
* Set up and maintained Logging and Monitoring subsystems using tools loke; Elasticsearch, **Kibana**, **Prometheus**, **Grafana** and Alert manager and Established infrastructure and service monitoring using **Prometheus** and **Grafana**.
* Worked with **Terraform** Templates to automate the Azure virtual machines using terraform modules and deployed virtual machine scale sets in **production environment.**
* Written Templates for Azure Infrastructure as code using **Terraform** to build staging and production environments. Integrated Azure Log Analytics with **Azure VMs** for monitoring the log files, store them and track metrics and used Terraform as a tool, Managed different infrastructure resources Cloud, VMware, and Docker containers.
* Build out server automation with **Continuous Integration – Continuous Deployment** tools like **Jenkins/Maven** using tools like **Git**, **Bit** **Bucket**, **Maven**, **Cucumber**, **JUnit**, **Nexus** **Repository**, **Ansible**, **Docker**, **Docker** **central**, **Kubernetes** and **Oracle** **DB** for deployment and build management system. Defined branching, tagging and merge strategies for all applications in **GIT**.
* Worked on google cloud platform (**GCP**) services like compute engine, cloud load balancing, cloud storage, cloud SQL, stack driver monitoring and cloud deployment manager.
* Setup **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.
* Configured and monitored infrastructure using **Nagios**, **Splunk**, **ELK** **stack** (Elasticsearch, Logstash, Kibana).
* Developed **Shell** and **python Scripts** for the manual deploying the code to the different environments and updating the team after the completion of the build and release successfully.
* Worked on **Google Cloud Components**, **Google container builders** and **GCP** client libraries and cloud **SDK’s**. Configured a Google Cloud Virtual Private Cloud (**VPC**) and **Subnet** Group for isolation of resources. Architecting the infrastructure on Google Cloud Platform using **GCP** **services** and **automated** **GCP** infrastructure using **GCP** **Cloud Deployment manager.**

**Client: Mind finder’s Baltimore, MD. Feb 18 – Nov 18**

**Role: AWS DevOps Engineer**

* Hands on experience in **Amazon Web Services** (AWS) provisioning and good knowledge of **AWS cloud services** such as **EC2, auto-scaling** and **VPC** to build secure, highly scalable, and flexible systems that handled expected and unexpected load bursts.
* Designing and implementation of public and private facing servers on **AWS Cloud**, Configured and managed various **AWS Services** including **EC2, RDS, VPC, S3, Glacier, Cloud Watch, Cloud Front, and Route 53** etc. Created users, roles and groups using **Identity Access Management (IAM)** and attached the required policies.
* Wrote playbooks for **Ansible** and deploying applications using **Ansible.** Worked with **AWS Cloud Formation Templates**, **terraform** along with **Ansible** to render templates with Orchestration templates in OpenStack Environment.
* Deploying and maintaining multi-container applications through **Docker.** Worked on **Docker** container snapshots, attaching to a running container, removing images, managing directory structures, and managing containers.
* Experience working on several **Docker** components like **Docker Engine, Hub, machine, compose and Docker registry**. Experience working on **Docker** Hub, creating **Docker** images, and handling multiple images primarily for middleware installations and domain configurations.
* Wrote Ansible Playbooks using **Python modules** to manage configurations of **Open stack** Nodes and Test Playbooks on **AWS instances** using **Python**.
* Configured **IAM** roles for **EC2** instances and assigns them policies granting specific level access to S3 buckets. Using Cloud Watch service, created alarms for monitoring the EC2 server’s performance like CPU Utilization, disk usage etc.
* Build out server automation with Continuous Integration – Continuous Deployment tools like **Jenkins/Maven** using tools like **Git, Bit Bucket, Maven, Cucumber, JUnit, Nexus Repository, Ansible, Docker, Docker central, Kubernetes**and **Oracle DB** for deployment and build management system. Defined **branching, tagging** and **merge** strategies for all applications in **GIT.**
* Analyzed the **ANT** buildprojects for conversion and converting to **Maven** build projects, installation, and configuration of **Jenkins** to support Java builds and Jenkins Plugins to automate continuous builds and publishing **Docker images** to **Nexus Repository**.
* Worked on creation of **Docker** containers and **Docker** consoles for managing the application life cycle, worked on setting up the automated build on **Docker** HUB.
* Created several pods using Master and Minion architecture of Kubernetes and developed microservice on boarding tools leveraging Python allowing for easy creation and maintenance of build jobs and Kubernetes deploy and services.
* Deployed **Kubernetes** clusters on top of Amazon EC2 instances using KOPS, Build Jenkins pipelines to drive all micro services builds out to the Docker registry and then deployed to Kubernetes, Created Pods and managed local deployments in Kubernetes creating local clusters and deploying application containers.
* Used Jenkins as Continuous Integration tools to deploy the Spring Boot Microservices to **Pivotal Cloud Foundry ( PCF)** using build pack.
* Building/Maintaining **Docker container clusters** managed by **Kubernetes, Linux, Bash, GIT, Docker** on **AWS** and used **Kubernetes** and **Docker** for the runtime environment of the CI/CD system to build, test deploy.
* Implemented multi-tier application provisioning in **OpenStack cloud**, integrating it with **Ansible**, Integrated **GIT** into **Jenkins** to automate the code checkout the process, configured plugins for integration tools to the **version control tools.**
* Administered and Engineered **Jenkins** for managing weekly **Build, Test, deploy chain** and **GIT** with Migrating the applications across environments such **DEV/Test/Prod** Branching Model for weekly releases. Created a fully Automated Build and Deployment Platform and coordinating code builds promotions and orchestrated deployments using **Jenkins/Bamboo**, and **GitHub**.
* Configured and monitored infrastructure using **Nagios, Splunk, ELK stack (Elasticsearch, Logstash, Kibana).**
* Created **Python** scripts to totally automate **AWS services**, which includes **Web servers, ELB, Cloud Front distribution, database, EC2** and database security groups, **S3 bucket** and application configuration.

**Client: Lands’ End, Dodgeville, WI Nov 16 - Feb 18**

**Role: Cloud**/**DevOps Engineer**

* Manage the **build activities** of multiple teams, delivering multiple **releases** of services and applications. Coordinate with the **Development, Database Administration, QA, and IT Operations teams** to ensure there are no resource conflicts.
* Build, manage, and continuously improved the build infrastructure for the **development, staging and production software environments** and interacted with client teams to understand client deployment requests.
* Installed **Pivotal Cloud Foundry (PCF)** on **EC2** to manage the containers created by PCF. Used **Docker** to virtualize deployment containers and push the code to EC2 cloud using PCF. Migrating On-Premises Infrastructure to **AWS** Cloud focusing on high-availability, fault tolerance, and auto scaling.
* Created **JIRA issues** to prioritize and act on what is important and stayed up to date with what is going on around the project.
* Continuously look to improve the scalability of the **build, release** and **deployment** processes and operation. Coordinate and support project branches merging process with **source code control**.
* Created resource group which includes **Virtual Machines, Network Interface, Virtual Networks, Sub-Nets, Load Balancers, Storage Accounts, Functions and SQL databases.**
* Evaluated **Chef** and **Puppet** framework and tools to automate the cloud deployment and operations. Created and implemented chef **cookbooks** for deployment and used Chef Recipes to create a Deployment directly into **Amazon EC2 instances**.
* Knowledge and expertise on all aspects of Chef concepts like**chef server, workstations, nodes, chef clients** and various components like push jobs, supermarket etc.
* Experience with container-based deployments using **Docker**, working with **Docker** images, Docker HUB and Docker registries.
* Worked with **AWS Code Pipeline**, Configured and Managed **EC2, Elastic Beanstalk, RDS, S3 Buckets, VPC, Security Groups, ELB, Auto scaling, Elastic Cache, Glacier, SNS, SQS, SES, CloudWatch, ECS, CloudTrail, CloudFormation.**
* Creating CloudFormation **JSON** templates to create custom **VPC**. Migrated a production infrastructure into AWS utilizing Code Deploy, Code Commit, Knowledge of Amazon ECS Clusters and able to manage containers in it.
* Used Ansible to Setup **ELK stack (Elasticsearch, Log stash, Kibana)** and troubleshoot the build issues with ELK and work towards the solution. Utilized **Ansible** and **AWS lambda**, **elastic cache** and **CloudWatch** logs to automate the creation of log aggregation pipeline with **Elasticsearch, Logstash, Kibana stack (ELK stack)** to send all our team's logs coming into CloudWatch, to process them and send them off to Elasticsearch.
* Configured **POM** file to manage Maven plugins such as Maven build plugins and Maven Reporting plugins and resolve Maven dependencies required for the project.
* Integrated **GIT** with **Jenkins** using the **GitHub** plugin to automate the process of source code check-out by providing the URL and credentials of the GIT repository.
* Installed, configured, and administrated all **LINUX** servers on **Amazon Web Services (AWS),** includes the design and selection of relevant hardware to Support the installation/upgrades of **Red Hat, CentOS** operating systems.
* Developed automated deployment scripts using **python** to deploy war files, properties file and database changes to development server or **QA server** and **Staging/Production server.** Developed the automated **build**and **deployment** using **ant** and **Jenkins** using **tomcat**.

**Client: Conning, Hartford, CT Sept 15 – Nov 16**

**Role: DevOps Engineer**

* Installed, Virtual **machine (KVM, VMware)** and configured **Red hat Enterprise** Linux (RHEL5/6/7), **Centos**5/6/7.
* Server racking, cabling patches on the operational and development system, which includes but is not limited to, Centos, Red Hat Linux and Solaris10.
* Worked on installation, configuration, and maintenance Red hat, CentOS and Suse Servers at multiple Data Centers.
* Managing **Amazon Web Services** (AWS) infrastructure with automation and configuration management tools such as Ansible, Puppet, or custom-built designing cloud-hosted solutions, specific AWS product suite experience.
* Managed **VMWARE vSphere HOSTS** in a HA/DRS Environment with vMotion/ Site **Recovery Manager**, Replication Manager vRealize Automation Center and Fiber SAN. Experience working with production servers at multiple datacenters.
* Implemented **Migration from VMWARE to AWS using VMDK Import tools**.
* Experience in migration of consumer data from one production server to another production server over the network with the help of Bash and Perl scripting.
* 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.
* Wrote script by using modules from **Puppet forge**.
* Installed Fuse to mount the keys on every **Debian Production** Server for **password-less authentication**.
* Manage **VM's** using **VMware** virtual center, Creating/Deploy VMs using templates/cloning.
* Deployment, configuration and administration of VMware vSphere 5.1 environments as well as Red Hat (5.x and 6.x) and Oracle Enterprise Linux (OEL) environment and installed required binaries and packages.
* Migrated**PhysicalServerstoVirtualServesusingVMware**ConvertersandmigrationofVirtualMachinesusingV-Motion.
* Installed & configured Guest operating system like Red hat on VMware ESX6.x.
* Worked with Server Installation & Customization of Jenkins for to atomize the build/deployment process for respective application/project.
* Created **Jenkins job**s for continuously building the projects. Scheduled jobs to build upon **every check** in into **subversion**.
* Integrated **Maven** with **Subversion** to **manage** and **deploy project related tags**.
* Implemented a Continuous Delivery pipeline with **puppet, Docker, Jenkins and GitHub** and **VMware.**
* Implemented detailed systems and services monitoring using **Nagios and Zabbix** services
* Involved installing and managing different automation and monitoring tools on **Red hat Linux Enterprise** like **Nagios and Puppet.**
* Developed shell scripts for automation of the build and release process, developed Custom Scripts to monitor repositories, Server storage.

 **Client: Magic Touch Manpower Supply, India. July 13 – Aug 15**

 **Role: Build & Release Engineer**

* Developed build and supported the **Software Release Management** procedures and deployment processes.
* Responsible for design and maintenance of the **Subversion/CVS** Repositories, views, and the access control strategies. Performed all necessary day-to-day **Subversion/CVS** support for different projects. Created the branches in Subversion and Perforce to implement the parallel development process.
* Used **Jenkins** innovatively to automate most of the build related tasks. Improved throughput and efficiency of build system by providing EO/managers rights to trigger required build. Configured and maintained **Jenkins** to implement the CI process and integrated the tool with Ant and Maven to schedule the builds.
* Merged release branches to the trunk after the production release and resolved the conflicts rose during the merge in both Subversion and Perforce.
* Implemented version control of puppet modules and process documentation, used **JIRA** for change control & ticketing.
* Installed and administered **Artifactory** repository to deploy the artifacts generated by **Maven** and to store the dependent jars, which are used during the build.
* Implemented **Puppet Version 3** for **Automation** in the Non-production environment and utilize Puppet for configuration management of hosted Instances within AWS.
* Handled Code reviews and merging Pull requests, resolving merging issues during build and release by conducting meetings with developers and managers. Responsible for designing and deploying best **SCM** processes and procedures.
* Worked on **ANT and MAVEN** Scripts to automate the build process. Implemented **Maven** builds to automate **JAR and WAR**.
* Developed build and deployment scripts using **ANT** and **MAVEN** as build tools in Jenkins to move from one environment to other environments.
* Worked closely with software developers and DevOps to debug software and system problems. Developed and maintained integration and release processes, create scripts for system administration and **AWS** using languages such as **BASH** and **Python**.
* Responsible for Connectivity issues among various Servers and various software components as well as setting up network environments using **TCP/IP, NIS, NFS, DNS, SNMP** agents, **DHCP** and **Proxy**.
* Deployed the applications to **Web sphere Application Server** and static content to **apache web servers.**

**Client: Macro pace Technologies, IndiaMay 12 – July 13**

**Role: Linux Administrator**

* Managed and administrated of all **UNIX** servers, includes **Linux** operating systems by applying relative patches and packages at regular maintenance periods using **Red Hat Satellite server**, **YUM, RPM** tools.
* Installation and administration of RHEL 6.x and Configured kickstart server and updating patches to the servers using Red Hat Satellite server. Tune the Servers and configuring networks for optimum performance
* Planned and performed the upgrades to **Linux (RHEL 5x, 6x, SUSE 10, 11, CENTOS 5, 6,** operating systems and hardware maintenance like changing memory modules, replacing disk drives.
* Setting up network environments using **TCP/IP, NIS, NFS, DNS, SNMP agents, DHCP and Proxy,** user account maintenance, backup & recovery, Auto-mounting, License Management, Printer configuration.
* Deployment and Configuration of application server **Tomcat** deploying **Java** and **Web Application**.
* Planned, scheduled and Implemented OS patches on **Linux** boxes as a part of proactive maintenance.
* Writing **Shell scripts** for automation of daily tasks, documenting the changes that happen in the environment and in each server, analyzing the **error logs**, analyzing the **User logs.**
* Identify, troubleshoot, and resolve problems with the OS build failures. Performing **kernel** and database configuration optimization such that it limits I/O resource utilization on disks.

**Technical Skills**

|  |  |
| --- | --- |
| **Configuration Management**  | Ansible, Chef and Puppet. |
| **Continuous Integration**  | Jenkins, Team city, Bamboo |
| **Version Control &****Centralized Repositories**  | Git, SVN, GitHub, Bitbucket and Subversion. |
| **Build Tools** | MAVEN, Gradle, MS-Build and ANT. |
| **Cloud Platforms &****Virtualization Platforms**  | AWS, Azure, Google Cloud, and PCF, VMWare. |
| **Package Management** | Nexus, Artifactory. |
| **Issue Tracking** | JIRA, Service-Now, Remedy. |
| **Containerization** | Docker, Kubernetes, OpenShift. |
| **Operating Systems** | Linux (Red Hat 5/6), Ubuntu, CentOS, Windows, and Unix. |
| **Databases** | MySQL, PostgreSQL, MongoDB, Oracle DB 10g, 11g. |
| **Programming Languages** | JavaScript, XML, HTML, Groovy, Shell script, Ruby and Python. |
| **Infrastructures spin up tools** | Terraform, CloudFormation, ARM templates. |
| **Web Application server** | Web logic, Web Sphere, Apache Tomcat and JBOSS. |
| **Logging & Monitoring Tools** | Nagios, Cloud watch, Splunk, ELK, Grafana, Prometheus. |