**MD. MAHMOOD**

**Sr. DevOps Engineer**

**mm9092871351@gmail.com**

**­­Phone: (909) 287-1351**

**SUMMARY:**

* Sr Cloud DevOps engineer with over 8 years’ experience in **Cloud**(Azure, AWS, GCP), **DevOps, Configuration management, Infrastructure automation, Continuous Integration and Delivery**(CI/CD). I can implement effective strategies for N - Tier application development in both **Cloud and On-premises** environments. Experience in dealing with **Unix/Linux and Windows** server administration.
* Expertise in Architecting and Implementing **Azure Service** Offering, such as **Azure cloud services, Azure storage, IIS, Azure Active Directory (AD), Azure Resource Manager (ARM), Azure Storage, Azure, Blob Storage, Azure VMs, SQL Database, Azure Functions, Azure Service Fabric, Azure Monitor, and Azure Service Bus**.
* Hands on experience on Backup and restore **Azure services**and in Design and configure **Azure Virtual Networks (VNets), subnets, Azure network settings, DHCP address blocks, DNS settings, security policies and routing. Azure cloud services, Blob storage, Active directory, Azure Service Bus, Cosmos DB.**
* Expertise in Azure **Scalability**and Azure **Availability**- Build VMs availability sets using the **Azure**portal to provide resiliency for **IaaS**based solution and **Virtual Machine Scale Sets (VMSS)**using **Azure Resource Manager (ARM)**to manage **network traffic.**
* Experience in migrating on-premises applications to **Azure**and configured **VNETs**and **subnets**as per the project requirement also performed **PowerShell scripting**to do **Patching**, **Imaging**, and **Deploymen**ts in **Azure**.
* Knowledge of **Azure Site Recovery**and **Azure Backup**Installed and Configured the Azure Backup agent and virtual machine backup, Enabled **Azure Virtual machine**backup from the **Vault**and configured the **Azure Site Recovery (ASR).**
* Implemented a CI/CD pipeline using **Azure DevOps(VSTS,TFS)** in both cloud and on-premises with **GIT, MS Build, Docker, Maven**along with **Jenkins**plugins.
* Experience in writing **Infrastructure as a code**(IaC) in **Terraform, Azure resource management, AWS Cloud formation.**Created reusable **Terraform modules**in both Azure and AWS cloud environments.
* Experience in Infrastructure Development and Operations involving **AWS Cloud Services, EC2, EBS, VPC, RDS, SES, ELB, Auto scaling, CloudFront, Cloud Formation, Elastic Cache, API Gateway, Route 53, Cloud Watch, SNS.**
* e in managed kubernetes services such as **Google kubernetesengine(GKE), Azure kubernetes and AWS Elastic container service.**
* Hands-on in using **OpenShift**for container orchestration with **Kubernetes**, container storage, automation, to enhance container platform multi-tenancy. Experience with **OpenShift, Kubernetes**architecture and design, troubleshooting issues and multi-regional deployment models and patterns for large-scale applications.
* Proficiency in using **Docker Hub, Docker Engine, Docker images, Docker Weave, Docker Compose, Docker Swarm**, and **Docker Registry**and used **containerization**to make applications platform when moved into different environments.
* Proficient in creating **Docker images**using **Docker File**, worked on **Docker container**snapshots, removing images, and managing **Docker volumes**and implemented **Docker** automation solution for **CI/CD** model.
* individual applications in **Kubernetes**.
* Experience in **Configuration management tools**such as **Chef, Ansible,**and **Puppet.**Wrote Chef recipes and cookbooks in **Ruby**scripting.
* Hands-on experience in bootstrapping the nodes using **knife**and automated by **testing Chef Recipes, Cookbooks with test-kitchen and chef spec**. Refactored Chef and **Ops Works** in **AWS cloud** environment.
* **CloudWatch, Azure Monitor**and**ELK**.
* Skilled with **Python, Bash/Shell, PowerShell, Ruby**, **Perl**, **PowerShell**, **JSON**, **YAML**, **Groovy**. Developed **Shell** and **Python** Scripts used to automate day to day administrative tasks and automation of the **build** and **release process**.
* Experience in **Java, C, C++, C#, Net, and ASP.Net** to manage **Software Configurations** and builds in a **UNIX** environment.
* In-depth knowledge and experience in implementing **Service-Oriented Architecture (SOA)** using **XML** based **SOAP/RESTful API Web Services** .

**TECHNICAL SKILLS:**

**Public & Private Cloud Technologies:** Microsoft Azure, Amazon Web Services (AWS), Google Cloud Platform (GCP), OpenStack, Pivotal Cloud Foundry (PCF).

**Containerization Tools:** Docker, Docker Swarm, Kubernetes, AWS ECS, Apache Mesos, OpenShift

**Configuration Management:** Chef, Ansible, Puppet, Salt Stack, Terraform

**CI/CD Tools:** Jenkins, Bamboo, GitLab CI, Travis CI, Hudson, VSTS, TFS.

**Build & Testing Tools:** Maven, Ant, Gradle, Selenium, JUnit, NUnit, xUnit

**Version Control Tools:** Git, Subversion, GitHub, Bitbucket.

**Performing/Monitoring & Bug Tracking Tools:** ELK, Nagios, CloudWatch, Azure Monitor, New Relic, Splunk, Grafana, Prometheus, Confluence, Jira.VMware ESXI, Vagrant, KVM, Windows Hyper V, Power VM, vSphere 5Citrix

**Web Servers:** Apache Tomcat, Nginx, WebSphere, WebLogic, JBoss, Samba, SQL Server.

**Databases:** Dynamo DB, MySQL, RDBMS, NoSQL, Cassandra, PostgreSQL, Mongo DB, Oracle DB

**IDE Tools:** MS Visual Studio, NetBeans, Eclipse, IntelliJ IDEA, PyCharm, Oracle SQL Developer

**Web Technologies:** HTML5, CSS3, Bootstrap, JSON, jQuery, JavaScript, PHP, XML, XSD, Drupal, WordPress

**Networking/Protocols:** DNS, DHCP, FTP/TFTP, NFS, SMTP, TCP/IP, NIS, HTTP/HTTPS, WAN, LAN, Palo Alto, Cisco Routers/Switches

**Scripting/Programming Languages:** Python, Shell Scripting, Bash Shell, Korn Shell, Ruby, Groovy, PowerShell, YAML, Perl, C, C++, .NET, ASP.NET, Java, Java/J2EE, Go.

**Operating Systems:** RHEL, CentOS, Ubuntu, Fedora, Debian, Solaris, Windows, MacOS.

**PROFESSIONAL EXPERIENCE:**

**BNY Mellon. Woodland park, NJ October 2019 to till date**

**Sr Azure Cloud DevOps Engineer**

**Responsibilities:**

* Created recommendations on how to duplicate a subset of on-premise machines to the **Azure Infrastructure as a Service**(IAAS) offering which will be used for disaster recovery. This analysis included the specifics to synchronize on-premise data with SQL Server and SharePoint instances hosted in VMs.
* Worked on Azure Site Recovery and Azure Backup- Deployed Instances on Azure environments and in Data centers and migrating to Azure using **Azure Site Recovery**and collecting data from all Azure Resources using Log Analytics and analyzed the data to resolve issues.
* Configured Azure Multi-Factor Authentication (MFA) as a part of Azure AD Premium to securely authenticate users and worked on creating custom Azure templates for quick deployments and advanced **PowerShell**scripting. Deployed Azure SQL DB with GEO Replication, Azure SQL DB Sync to standby database in another region & Fail over configuration.
* Worked on **Serverless services**, created and configured **HTTP** Triggers in the **Azure Functions** with application insights for **monitoring** and **performing** load testing on the applications using the **Visual Studio Team Services** (**VSTS)**alsocalled as**Azure DevOps Services**.
* Created **Azure** Automation Assets, **Graphical runbook**, **PowerShell runbook** that will **automate** specific tasks, deployed **Azure AD** Connect, configuring **Active Directory Federation Service (AD FS)** authentication flow, **ADFS** installation using **Azure AD Connect**, and involved in administrative tasks that include **Build**, **Design**, **Deploy** of **Azure environment**.
* Configure **Continuous Integration** from source control, setting up build definition within **Visual Studio Team Services (VSTS)** and configure **continuous delivery**to automate the deployment of **ASP.NET MVC** applications to **Azure web apps** and managed **Azure Active Directory**, **Office 365** and applied upgrades on a regular basis.
* Implemented a **CI/CD** pipeline with **Docker**, **Jenkins** (**TFS** Plugin installed), **Team Foundation Server (TFS)**, **GitHub** and **Azure Container Service**, whenever a new **TFS/GitHub branch** gets started, **Jenkins**, our **Continuous Integration (CI)** server, automatically attempts to build a new **Docker container** from it.
* 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** and Recover from a Recovery Services **Vault** using **Azure PowerShell** and **Azure Portal**.
* 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.
* Worked on **OpenShift** for **container orchestration** with **Kubernetes** container storage, automation to enhance **container** platform multi-tenancy also worked on with **Kubernetes** architecture and design troubleshooting issues and **multi-regional** deployment models and patterns for **large-scale** applications.
* Deploying windows **Kubernetes (K8s)** cluster with **Azure Container Service (ACS)** from **Azure CLI** and Utilized **Kubernetes** and **Docker** for the runtime environment of the **CI/CD** system to **build**, **test** and **Octopus Deploy**.
* Using **Ansible** created multiple **playbooks** for machine creations and **SQL server, cluster server** and my SQL installations.
* Used **Ansible**to Setup/teardown of **ELK**stack ( **Elasticsearch, Log stash, Kibana**) and troubleshoot the build issues with **ELK**and work towards the solution.
* Written **Ansible handlers** with multiple tasks to trigger multiple handlers and to decouple handlers from their names, making it easier to share handlers among Playbooks and Roles.
* Managed **Kubernetes** charts using Helm, Created reproducible builds of the Kubernetes applications, managed Kubernetes manifest files and Managed releases of Helm packages.
* Implemented Docker -maven-plugin in **Maven**pom.xml files to build Docker images for all microservices and later used Docker file to build the Docker images from the **Java**jar files.
* Designed, wrote and maintained systems in **Python scripting** for administering **GIT**, by using **Jenkins** as a full cycle **continuous delivery** tool involving package creation, distribution, and deployment onto **Tomcat application** servers via shell scripts embedded into **Jenkins** jobs.
* Maintained **Artifacts** in binary **repositories** using **JFrog** Artifactory and pushed **new Artifacts** by configuring the **Jenkins** project **Jenkins Artifactory plugin**.
* Involved in **Jira** as defect **tracking** system and configure various workflows, customizations, and **plugins** for **Jira bug/issue tracker** integrated **Jenkins** with **Jira**, **GitHub**.
* Worked with **Nagios** for **Azure Active Directory** & **LDAP** and Data consolidation for **LDAP users** . **Monitored** system performance using **Nagios**, maintained **Nagios** servers and added new **services &servers** .
* Built and **managed** a highly available **monitoring infrastructure** to monitor different **application servers**like **JBoss, Apache Tomcat** and its components using **Nagios** .

**Environment:** s: - Azure, PCF, Office 365, Terraform, Maven, Jenkins, Ansible, Azure ARM, Azure AD, Azure Site Recovery, Kubernetes, Python, Ruby, XML, Shell Scripting, PowerShell, Nexus, JFrog Artifactory, Jenkins, Git, Jira, GitHub, Ansible, Docker, Windows Server, TFS, VSTS, LDAP, Nagios.

**West Corporation., Omaha, NE** **Sep 2018 to Oct 2019**

**Sr AWS Cloud / DevOps Engineer**

**Responsibilities:**

* Provisioned and administered **EC2** instances and configuring **EBS, Simple Storage(S3)**cross region replication, **Elastic Load Balancer, configure Auto scaling, setting up CloudWatch alarms, Virtual Private Cloud (VPC)**, mapping with multi **AZ VPC** instances and **RDS**, based on architecture.
* Worked on Amazon EC2 setting up instances, **virtual private cloud**(VPCs), and security groups and created AWS Route53 to route traffic between different regions and used BOTO3 and Fabric for launching and deploying instances in AWS.
* Configured **Amazon S3, Elastic Load Balancing, IAM**and **Security Groups**in Public and Private Subnets in VPC, created storage cached and storage volume gateways to store data and other services in the AWS.
* Architected and configured a virtual data center in the AWS cloud to support Enterprise Data Warehouse hosting including Virtual Private Cloud (VPC), Public and Private Subnets, Security Groups and Route Tables.
* Used Security Groups, Network ACLs, Internet Gateways, NAT instances and Route tables to ensure a secure zone for organizations in AWS public cloud.
* Worked on migration services like AWS **Server Migration Service**(SMS) to migrate on-premises workloads to AWS in easier and faster way using Rehost "lift and shift" methodology and AWS **Database Migration Service**(DMS), AWS Snowball to transfer large amounts of data and Amazon S3 Transfer Acceleration.
* Worked in an **Agile**development team to deliver an end to end continuous integration/continuous delivery product in an open source environment using tools like Puppet & Chef.
* Written **Terraform** scripts to automate **AWS services** which include **ELB**, CloudFront distribution, RDS, **EC2**, database security groups, **Route 53**, VPC, Subnets, Security Groups, and **S3** Bucket and converted existing AWS infrastructure to **AWS Lambda** deployed via **Terraform** and **AWS CloudFormation**.
* Implemented**AWS Elastic Container Service (ECS)** scheduler to automate application deployment in the **cloud** using **Docker** Automation techniques.
* Implemented **Docker** - **maven**-plugin in **Maven pom.xml** files to build **Docker images** for all **microservices** and later used **Docker File** to build the **Docker** images from the **Java jar files** also Created **Docker** images using a **Docker File**, worked on **Docker container** snapshots, **removing** images and managing **Docker volumes.**
* Wrote **Python** scripts using the **Boto3** library to automatically sign up the instances in **AWS EC2**and **OpsWorks stacks** and integrated with **Auto Scaling** with configured **AMI’s.**
* Used **Git** version control to manage the source code and integrating **Git** with **Jenkins** to support build **automation** and integrated with **Jira** to **monitor** the commits.
* Worked on **AWS costs**by writing the **Ansible Playbooks**for auto **Start/Stop**of AWS resources at the time of the day by triggering it from **Jenkins**and knowledge on **RHEL**on **Ansible Playbooks**, **Modules**and **Roles**.
* Designed various **Jenkins**jobs to continuously integrate the processes and executed **CI/CD**pipeline using **Jenkins**, **Ansible Playbooks**and **Ansible Tower.**
* Deployed and configured **Chef** server including bootstrapping of **Chef client** nodes for provisioning and experience in developing **Chef recipes** through **Knife** command-line tool, customized Ruby scripts to create Chef cookbooks to manage systems configuration on **Chef Server**to initiate **Chef Nodes**.
* AWS Cloud management using **Chef Automation**and Automated the cloud deployments using chef, python and AWS Cloud Formation Templates.
* Used **Jenkins**and pipelines to drive all micro services builds out to the **Docker-registry**and then deployed to Kubernetes, Created Pods and managed using **Kubernetes**.
* Set up and managed **ELK ( Elasticsearch, Logstash & Kibana)** Stack to collect, search and analyze **log files** across servers, log monitoring and created geo-mapping visualizations using Kibana in integration with **AWS CloudWatch** and **Lambda**. Evaluated system logs using ELK software stack.
* Created automation and deployment templates for relational and **NOSQL**databases including **MongoDB**and **Redis**.

**Environment:** - AWS, Terraform, Chef, Ansible, Docker, Jenkins, Git, Jira, Jenkins, Kubernetes, Maven, Nagios, ELK, Java, SonarQube, Shell, Bash, Python, DynamoDB, Cassandra.

**Care source, Dayton, OhioJul 2017 to Sep 2018**

**Cloud / DevOps Engineer**

**Responsibilities:**

* Involved in designing and deploying multitude applications utilizing almost all **AWS stack (Including EC2, S3, AMI, Route53, RDS, SNS, SQS, IAM)** focusing on high-availability, fault tolerance, and **Auto-Scaling** in **AWS Cloud Formation.**
* Maintained the user accounts**IAM** Roles, **Route 53**(**CNAME**),**VPC**, **RDB**, **MongoDB**, **SQS** & **SNS** services in **AWS** cloud.
* Created **Python scripts**to totally automate **AWS**services which includes web servers, **ELB, Cloud Front**distribution, database, **EC2**and database security groups and application configuration, this script creates stacks, single servers, or joins web servers to stacks.
* Involved in creating the company's DevOps strategy in a mixed environment of Linux (Ubuntu, CentOS, RHEL) servers.
* Experience in writing **Ansible playbooks** by using **YAML** script to launch **AWS instances** and used to manage web applications, configuration files, used mount points and packages.
* Wrote Ansible playbooks, inventories created custom **playbooks**written in **YAML**language, encrypted the data using Ansible Vault & maintained role-based access control by using **Ansible Tower**& Implemented IT orchestration using Ansible to run tasks in a sequence which can work on different servers.
* Virtualized the servers using the **Docker**for the test environments and development environment and performed configuration automation using Docker containers.
* Wrote **CI/CD**pipeline in **Groovy**scripts to enable end to end setup of **build & deployment**using **Cloud Bees Jenkins**and Developed an end to end **CI/CD Pipelines** in **Jenkins** to retrieve code, perform tests and push build **artifacts** to **Nexus**.
* Wrote scripts for the **build**, **deploy**, **maintain** using **Jenkins**, **Git**, **Maven**, **Python**, **Bash** and worked on **CI/CD** involving **Run Deck Jenkins** and **Ansible** to complete the automation from commit to deploy.
* Wrote **Python** scripts using the **Boto3** library to automatically sign up the instances in **AWS EC2**and **Ops Works stacks** and integrated with **Auto Scaling** with configured **AMI’s.**
* Used **Git** version control to manage the source code and integrating **Git** with **Jenkins** to support build **automation** and integrated with **Jira** to **monitor** the commits.
* Integrated **Docke**r container-based test infrastructure to **Jenkins CI** test flow and set up build environment **integrating** with **Git** and **Jira** to trigger builds using **Webhooks** and **Slave Machines**.
* Implemented **Git**for **Branching**, **Merging**, **Tagging** and maintaining the version across the environments on **Linux** platforms and, designed and implemented **Git Metadata** including elements, labels, attributes, **triggers**, hyperlinks.
* Installed and configured DHCP, DNS, web servers like Apache, IIS, mail servers like SMTP, IMAP, POP3, and file servers on **RHEL**servers.
* Implemented a **CI/CD** framework using Jenkins, Maven & Artifactory in Linux environment and Integration of Maven/Nexus, Jenkins, Git and JIRA.
* Maintained **Jenkins**continuous integration tool using **shell/Perl**and **ruby**scripts for automating releases to different environments.

**Environment:** - AWS, Chef, Ansible, Maven, Jenkins, Docker, Python, Linux, Bitbucket, GitHub, Nagios, Shell, Bash, groovy, RHEL, Apache, IIS, Linux, SMTP, IMAP, POP3, RHEL, VMWare vSphere, Jira.

**GEICO, Miami, FLMar 2016 to Jun 2017**

**Build and Release Engineer**

**Responsibilities:**

* Used **Jenkins** as continuous integration tool: creating new jobs, managing required plugins, configuring the jobs selecting required source code management tool, build trigger, build system and post build actions, scheduled automatic builds, notifying the build reports.
* Develop/Improve continuous integration scripts and automation scripts and perform database deployments, Responsible for troubleshooting environmental issues.
* Implementing Continuous Integration and Continuous Delivery framework using SVN, Bitbucket, ANT, Maven, Jenkins, Bamboo, Nexus, Control Tier, Make in Linux environment.
* Involved in the Consulting, Analysis, Design, Development and Implementation of automated Continuous Integration/Continuous Deployment pipelines and test automation solutions.
* Worked on in creating various modules and automation of various facts in **Puppet**, adding nodes to enterprise **Puppet Master**and managing **Puppet agents**. Creating **Puppet manifests**files and implementing **Puppet**to convert **IaC**.
* Integrated **Bitbucket** with**JIRA**for transition JIRA issues from within Bitbucket Server and monitored the JIRA issues in Bitbucket Server.
* Implemented and configured **Nagios** for continuous monitoring of applications and enabled notifications via emails and text messages.
* **Built** and managed a highly available **monitoring** infrastructure to monitor different application **servers** and its components using **Nagios**.
* Designed and scripting using **Ant** for **J2EE**, **Web Services**, Reusable **JARS**, **Web Client** and **Open Source** in creating Master **build.xml** build Properties and provided technical support to the **development team** for compilation problems.

**Environment:** - Linux, Windows, Tomcat, Jira, Confluence, Hudson, Bamboo, JBoss, Puppet, Puppet DB, Ant, Maven, SVN, Bitbucket, Nagios, Java, Shell Scripting, Python, Bash, Perl, Kickstart, Java, Agile, Scrum, Splunk**.**

**Northwestern Mutual Life-** **Milwaukee, WINov 2014 to Feb 2016**

**System Administrator**

**Responsibilities:**

* Worked on design, **configuration**, **installation, implementation**, **management**, maintenance and **support** for the Corporate Linux servers **RHEL 4x, 5.x, CENTOS 5.x.**
* Installed and configured **SSH, SFTP, NDM, EMC Power path, Veritas Dynamic multipathing, Oracle client, SAMBA** utility software.
* Implemented the Jumpstart in **Solaris**servers and **Kickstart**in Linux Servers to mechanize the server works for numerous profiles.
* Worked on Linux **Logical Volume (LVM),**created volume groups, Logical volumes, file systems, and troubleshooting.
* Performed package management using RPM, YUM, Apt-get and dpkg for maintaining stability and security of the servers.
* Set up the Linux Cron jobs for automating various build related jobs and application data synchronization jobs.
* Install, maintain and upgrade Drupal and Word press on LAMP stack and Configured LAMP Stack on Unix/Linux servers.
* **Plan**, **build, and support** new infrastructure including **VMware ESXi** and **vSphere** virtualization servers, **new file-server** and other resources and Implementing **Kickstart/PXE-boot configuration** for automating **Linux installations.**

**Environment:**  - Windows, Unix, Linux (RHEL 5.x/6.x), Solaris 9/10/11, SUSE, LVM, Tomcat, Cron Job, HTTP, SSH, Telnet, FTP, RSH, RAID 0/1/5/10, VMware, Kickstart, Shell, Bash Scripting.

**Sonata Software\_ India Aug 2012 to Oct 2014**

**Jr System administrator­**

**Responsibilities:**

* Created users, manage user permissions, maintain User & File System quota on **Red hat Linux.**
* Worked on Day to Day Server maintenance task using command **ns lookup, ping, top, vmstat, IOStat, tcp dump**etc.
* Worked on various **UNIX servers**(core dump and log file analysis) for configuring and troubleshooting along with analysis of hardware and software failures.
* Configure dynamic and static network settings for **IPv4** and **filter packets**, Review file system management concepts, removable media and configure NFS shares with Autofs.
* Installed and maintained Computer Software and setting of Workstations and managing of IT assets like routers, switches, printers.
* Managed team to provide solutions of users' daily problems and maintaining daily problem log sheet.

**Environment:** - Windows, Unix/Linux, DNS, RedHat, Networking, Windows OS

**EDUCATION:**Bachelor’s in Computer Science from JNTU India.