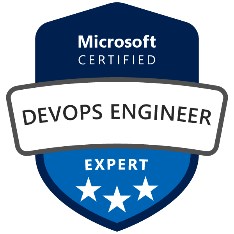
**Teja M**

Sr. Cloud Infrastructure Engineer

* (469)-844-7733
* tejamandapti@gmail.com

 [LinkedIn](https://www.linkedin.com/in/teja-m-9816b8262/)

**PROFESSIONAL SUMMARY**

Working in multi-cloud (Azure & AWS) settings as a Sr. Cloud Infrastructure Engineer and Cloud Engineer, certified Azure DevOps engineer with 8+ years of IT experience. An outstanding understanding of DevOps, configuration management, infrastructure automation, build and deployment automation, and environment management on Microsoft Azure and Amazon Web Services.

* Hands-on experience with **Microsoft Azure** services like **DNS, Azure Active Directory** **(AAD), Azure Storage, Log Analytics Workspaces, Azure Firewall, Key Vaults, Virtual Networks (VNET), Express Routes,** and **Application Gateways.**
* Expertise in creating and maintaining **Azure Key Vault keys**, **secrets**, and **certificates** as well as applying **access policies** for Service Principals, Users, and Groups to access these Key Vaults.
* Strong knowledge of writing **PowerShell scripts** and **Azure Resource Management (ARM)** templates to automate resource provisioning and deployment across different Azure subscriptions.
* Proficient in designing **Azure DevOps** pipelines that use **Docker** files to generate lightweight **Alpine Basic Images**, tag custom images, upload them to **Azure Container Registry (ACR),** and then use those images to deploy pods in an **Azure Kubernetes cluster (AKS).**
* Hand-on experience in establishing service connections to **Azure DevOps** and integrating **Azure DevOps** Boards with **Microsoft Teams** and Pipelines for Teams and Sprint Boards notification.
* Worked with **Terraform** modules to deploy infrastructure in **Microsoft Azure**, including **Virtual Networks, Load Balancers, Storage Accounts, Virtual Machines, Virtual Machine Scale Sets, Azure Kubernetes Service (AKS),** **Key Vaults,** and **Log Analytics Workspace**.
* Work experience in Amazon services including **ELB**, **CloudFront** distribution, **RDS**, **EC2**, database security groups, **Route53**, **VPC**, **Subnets**, **Security Groups**, and **S3 Bucket** was automated using Terraform scripts, and the current AWS architecture was changed to use AWS Lambda deployed using **Terraform** and **AWS CloudFormation.**
* Hands-on experience in setting up and configuring **Prometheus** to monitor **Kubernetes** resources and API using **Kube-state metrics**, and **Kubernetes** **nodes** using node-exporter.
* Hands-on experience in creating Ingress and Egress rules to control traffic entering and leaving the **Azure Kubernetes Cluster (AKS)** and writing various **RBAC** policies to define user permissions in the **AKS** Cluster.
* Configured **Kubernetes cluster** to scale up operations of clusters, maintain the cluster services, load balancing, network policies, and group **Docker** containers across different platforms. Configured cluster operations in **Amazon Kubernetes (EKS)** to deploy microservices with **CI/CD** pipeline.
* Expertise in creating services like **VPCs, Subnets, and EC2** using AWS Cloud Formation templates, as well as configuring the applications running on the launched instances make **Snapshots** and **use Amazon Machine Images (AMIs)** as backups in production environments.
* Implemented a serverless architecture using **API Gateway**, **Lambda** and **Dynamo DB**, and deployed **AWS Lambda** code from Amazon **S3** buckets and additionally created a Lambda deployment function and configured it to receive events from your **S3** bucket.
* Skilled in using **Jenkins** to deploy artifacts into Nexus Repository as well as writing **Groovy scripts** for Continuous Integration and build procedure.
* Experienced in utilizing **Jenkins pipelines** to deploy Docker images across DEV, QA, and PROD environments using the **Elastic Container Registry (ECR)** and **Docker Hub**.
* Integrated several technologies with **Cloud Bees Jenkins** to develop a **CI/CD** pipeline that generates and executes **Terraform** script templates for **Microsoft Azure** infrastructure creation.
* Experience in writing **Ansible Playbooks** to **Patch** and **Reboot** the Debian, RedHat, and Windows Virtual Machines across all environments and to send the final report to the team through email.

**CERTIFICATIONS:**

1. **Microsoft Azure Administrator**
2. **Microsoft DevOps Engineer Expert**

**TECHNICAL PROFICIENCY:**

|  |  |
| --- | --- |
| **Cloud Platforms** | Microsoft Azure, Amazon Web Services |
| **Container Orchestration Tools** | Docker Swarm, AKS, EKS |
| **Configuration Management** | Ansible, Chef |
| **Version Control Tools** | GIT, GITHUB, SVN, Bitbucket |
| **Web and Application Servers** | JBOSS, Apache Tomcat, Web Sphere, Web Logic, Nginx, Send Mail |
| **Monitoring Tools** | Prometheus, Datadog, Grafana |
| **CI/CD Tools** | Jenkins, Azure DevOps |
| **Scripting Languages** | Shell, Python, PowerShell, YAML, JSON |
| **Database** | MySQL, Oracle, Dynamo DB, Mongo DB |
| **Repository Management** | Nexus, JFrog |
| **Operating System** | Red Hat, Ubuntu, Debian, Fedora, CentOS, Windows, Mac OS |
| **Networking Protocols** | TCP/IP, DNS, NFS, ICMP, SMTP, DHCP, OSPF, BGP, UDP and RIP |

**PROFESSIONAL EXPERIENCE:**

**CLIENT: Vanguard, Malvern, PA DURATION: August 2021 - Present**

**ROLE: Sr. Cloud Infrastructure Engineer**

**Key Responsibilities:**

* Extensively worked on setting up and configuring **Microsoft Azure** services such as **Virtual Machines**, **Storage Accounts, App Services, Virtual Networks, Blob Storage, Application Gateways, Log Analytics Workspace, Function Apps, Application Insights,** and **Express Routes.**
* Well-versed in setting up **Express Route Circuit** in **Microsoft Azure** between the **Virtual Network Gateway** and on-premises to enable communication with cloud services.
* Played a significant role in creating **Kusto Query Language (KQL**) queries in Azure Log Analytics Workspace to get statistics and insights from Azure Resources**.**
* Worked extensively on setting **PagerDuty** integrations for **Alert** notifications to **monitor Heartbeat, CPU**, and **Memory Alerts** in **Azure Monitor**.
* Strong understanding of establishing **Azure DevOps** self-hosted agents on **Azure Kubernetes Cluster (AKS)** and implementing an auto-scaling mechanism to allow multiple pipelines to operate concurrently.
* Worked on maintaining the **Azure Container Registry (ACR)** which stores the private customized **Docker images** and makes them available to **Azure DevOps** pipelines for use in deploying **Docker images** across all environments.
* Expertise in utilizing reusable Terraform Modules to create resources in **Microsoft Azure**, including **VNETs**, **Virtual Machines**, **Application Gateways**, **Event Hubs,** **Storage Accounts**, **Azure Kubernetes Cluster**, **Key Vaults**, and **PostgreSQL** across all environments.
* Worked extensively in deploying Microsoft Azure's **Event Hub Name Space (EVNS)** and **Event Hubs** as well as working with Terraform modules to provide roles for users to access the **EVNS** and **Event Hubs**.
* Worked with **Helm** package manager in creating custom charts for various applications for implementing custom **Helm Charts**.
* Involved in implementing the concept of hardening the **Azure Kubernetes (AKS)** cluster to make it secure and resilient while administering and supporting the **Azure Kubernetes** infrastructure.
* Expertise in using **Prometheus** and **Grafana** to construct numerous dashboards for separate applications on **Azure Kubernetes (AKS)** clusters and set up alarms.
* Responsible for operating, shipping, and securely deploying the application using Docker to speed up the build and release engineering. Expertise in managing **Docker volumes** and **Docker container snapshots**.
* Hands-on experience removing unused **Docker containers** and **Dangling Docker images** across several nodes by setting up **CRON** jobs on them with **Ansible.**
* Well versed in installing, updating and managing packages like **azcli**, **python**, **kubectl**, **terraform**, **docker** and **git** on **Linux** and **Windows** Virtual Machines across DEV, QA and PROD environments using **Ansible Playbooks**.

**Environment:**  Azure, AAD, Azure DevOps, Terraform, AKS, Networking, Docker, Ansible, Prometheus, Grafana, Bash, Python, Linux, Jira, Bitbucket, Apache Tomcat, ARM, Virtualization, CRON.

**CLIENT: Capital One, McLean, VA DURATION: May 2019 – June 2021**

**ROLE: Cloud/DevOps Engineer**

**Key Responsibilities:**

* Hands-on experience in setting up **Users**, **Service Principals,** and **Groups** in **Azure Active Directory (AD)** and configuring Azure **Identity Access Management** (IAM) for enhanced login authentication.
* Extensive Experience of debugging **Virtual Machines**, **Virtual Machine Scale Sets**, and Connections using a **Serial Console** when SSH logging is malfunctioning.
* Responsible for setting up **OMS Agent** on **virtual machines** in various types of scenarios, tying them into **Microsoft Azure's Log Analytics Workspace**, monitoring metrics, and tracing down and fixing root causes of problems.
* Expert in utilizing **Packer** to update Azure Images in **Azure Compute Gallery** and **Terraform** to update these Image references for **Virtual Machines** and **Virtual Machine Scale Sets** across all environments.
* Experience managing infrastructure on **AWS** with a focus on high availability, fault tolerance, and auto-scaling using **Terraform** templates, and Continuous Integration and Continuous Deployment with **AWS lambda** and AWS code pipeline.
* Played a significant role in setting up and configuring **Prometheus** to monitor **Elastic Kubernetes Service (EKS)** **nodes** with node-exporter, resources with Kube-state-metrics, and the **Kubernetes API**.
* Responsible for running the **CI/CD** pipeline in **Jenkins** to deploy private Docker Images to the **Azure Kubernetes Cluster (AKS)** utilizing the **Azure Container Registry (ACR)** across various environments.
* Used master, node, and managed **Kubernetes orchestration** of **Docker containers** and **Images**. worked on taking container snapshots, managing Docker volumes, and deleting Images.
* Expert in creating **Groovy scripts** for **Jenkins** Pipelines and integrating **GIT Webhooks** and Poll SCM with Jenkins to automate Jenkins procedures.
* Played a significant role in using Amazon Web Services (AWS) including **Virtual Private Cloud (VPC)**, **Network Security Group (NSG)**, Public and Private **Subnets, Route Tables, Elastic Load Balancer (ELB)**, and **NAT Gateways**.
* Created a highly accessible and scalable **Domain Name System (DNS)** within the Amazon cloud to redirect traffic from customers connected via AWS Direct Connect for bigger deployments.
* Experience in **IAM policies** management for organizations in **AWS** to define groups, create users, assign roles and define rules for role-based access to **AWS resources**.
* Utilized **Ansible Tower** to manage complex network deployments, schedule tasks and add control and delegation to Ansible-powered environments. Automated infrastructure using Ansible roles that were downloaded and managed from Ansible Galaxy.

**Environment:**  Azure, AAD, PagerDuty, SSO, AWS, AWS Lambda, Jenkins, JIRA, Ansible, Terraform, Kubernetes, Prometheus, Docker, Python, Maven, Java, GitHub, Linux, Windows, PowerShell Scripts, Bash Script.

**CLIENT: Permanente Medical Group, Oakland, CA DURATION: June 2017 – May 2019**

**ROLE: AWS Cloud Engineer**

**Key Responsibilities:**

* Worked extensively with AWS Services such as **Elastic Compute Cloud (EC2), AWS Lambda, Amazon S3, Cloud Formation, Route53, Elastic Load Balancer (ELB), Elastic Block Store, Virtual Private Cloud (VPC), Security Groups, Cloud Watch,** and **Elastic Beanstalk.**
* Have worked on setting up **Virtual Private Clouds** (VPCs), **Security Groups** and instances on Amazon EC2, developed AWS Route53 to route traffic across regions; and used BOTO3 and Fabric to launch and deploy instances on AWS.
* Responsible for the successful deployment of Web apps and Database templates, custom-sized **VPCs**, **Subnets**, and **NAT** were created using **AWS Cloud Formation**.
* Created Build and Deployment Scripts using build technologies **like MS Build, and Maven** in **Jenkins**, and used **Nexus** for **artifacts** and **SonarQube** for code analysis to move across environments.
* Installed, removed and updated several **Jenkins plugins**. Created **Global credentials** in **Jenkins** and utilized them as environment variables throughout several pipelines and environments.
* Extensive experience in developing **Bash** scripts and **CRON** scheduling Logs older than 90 days will be deleted by tasks running as root on remote **Linux** systems.
* Responsible for creating scripts for cloud construction to host software on AWS cloud. using **PowerShell scripts** to automatically install the software.
* Worked extensively on **Source Code Management (SCM)** tools, such as SVN and **GitHub**, and ability to perform activities like branching, merging, tagging, and resolving disputes during mergers.

**Environment:** AWS, EC2, AWS Lambda, VPC, Cloud Formation, Jenkins, Maven, SonarQube, Bash, CRON, apt, yum, SVN, GIT, GitHub, Linux Administration, PowerShell.

**CLIENT: TCS, Hyderabad, India DURATION: June 2014 – November 2016**

**ROLE: System Administrator**

**Key Responsibilities:**

* Expertise in writing **Bash** scripts and deploying **CRON** Jobs on remote Linux servers to execute these scripts under the root user.
* Responsible for the upkeep of Linux operating systems such as **RHEL, CentOS, Ubuntu,** and **SUSE**.
* Involved in resolving problems that arose during code build, integration, and deployments and developed scripts for backing up essential **Linux** environment files and folders.
* Overseeing the installation and use of **SSH** clients such as **Putty** and Cyber-Duck, as well as remote file copying with **SCP, WINSCP,** and **FileZilla.**
* Worked extensively on installing and configuring the Linux system’s **logical volume manager (LVM)** for disk and file system management.
* Experience in using **rpm**, **yum**, **apt,** and **dpkg** for package management across multiple Linux hosts in different environments.

**Environment:** NFS, FTP, Linux, UNIX, RHEL, CentOS, Ubuntu, NIS, DHCP, FTP, Telnet, Nagios, Kickstart, SSH, VM Sphere, VMWare, Virtual Box, RPM, and YUM.

**EDUCATION:**

The University of Central Missouri, Warrensburg, MO.

Masters in Big Data Analytics and Information Technology.