# Manikanta

***Phone : +1(919)5216206 Email:manig9108@gmai.com***

***LinkedIn:*** [www.linkedin.com/in/manikanta-24g](http://www.linkedin.com/in/manikanta-24g)





***PROFESSIONAL SUMMARY***

***Certified Azure DevOps Engineer with 8+ years of IT experience with excellent knowledge in Configuration Management, Continuous Integration and Continuous delivery (CI/CD), Build and Release, Linux, and System Administration with a major focus on Cloud Platforms Azure, Amazon Web services (AWS) and includes great knowledge on the principles and best practices of software configuration Management (SCM) in agile, scrum and waterfall methodologies.***

## Proficient in Microsoft Azure IAAS, SAAS and PAAS services such as Virtual Networks, Virtual Machines, Azure Blob storage, Azure Data Factory, Cloud Services, Azure Express Route, Traffic Manager, VPN, Load Balancing, Application Gateways, Auto-Scaling, deploying Web Apps, monitoring by Azure Application Insights, and creating Web-Jobs.

* Experience in using **Azure Resource Manager** (ARM) to deploy, update and delete multiple resources on Azure and in migrating the on-premises resources to Azure with **Azure site recovery (ASR)** and Azure backups.
* Expertise in deploying **AZURE AD Connect**, configuring using **AZURE AD Connect**.
* I have experience in **Azure Network Security Groups**, **Azure Operational Insight and** OMS, **Azure Premium Storage**, **Site Recovery, Network Virtual Appliance, Client-Side** encryption fo**r Azure storage objects.**
* Experience in setting up the infrastructure using **AWS** services including **ELB, EC2, Elastic container services (ECS), Auto-scaling, S3, IAM, VPC, Red Shift, DynamoDB, Cloud Trail, Cloud Watch, Elastic Cache, Lambda, SNS, Glacier, Cloud Formation, SQS, EFS, and Storage Gateway.**
* Designed **AWS Cloud Formation** templates to create custom sized **VPC, Subnets**, and **NAT** to ensure successful deployment of **Web applications**, database templates and expertise in architecting secure **VPC solutions** in **AWS** with the help of **Network ACLs**, **Security groups, public and private network configurations**.
* Experience in maintaining **Amazon Web Services** and have hands on experience in architecting Legacy Data Migration projects such as **Teradata** to **AWS Redshift, migration, Teradata to Snowflake Migration, AWS Cloud** from **on- premises**.
* Experience in **Kubernetes** to deploy **scale**, **load balance**, and manage **Docker** containers with multiple names spaced versions using **Helm charts.**
* Deploying **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.
* Creating Docker **containers** and **Docker consoles** for managing the application life cycle. Responsible for Build and Install Schedules using **Unix Cron jobs.**
* Worked on **Docker-Compose, Docker-Machine** to create **Docker** containers for testing applications in the QA environment and automated the deployment to **Nexus, scaling**, and management of containerized applications.
* Configured applications that run multi-container **Docker** applications by utilizing the **Docker-Compose** tool which uses a file configured in **YAML** format.
* Implemented microservices, application development, and migration using **AWS/Azure services** such as **Azure DevOps, Kubernetes Service (AKS)**, **Container Registry**, **Cosmos DB**, and Grafana, Azure **Pipelines**, **Monitor**, **RBAC, AWS Kubernetes, EKS and Kubernetes API** to run workloads on **EKS Clusters**.
* Installed and configured **Nagios** as monitoring tool for disk usage**, CPU utilization**, list of open files etc. and **sending alerts** to our teams based on the same. Setup threshold alerts in **Nagios XI** and updated contact definitions to notify potential service or host interruptions.
* Experience in **Configuration management** tools such as **Chef, Ansible**, and **Puppet**.
* Wrote **Chef recipes** and **cookbooks** in Ruby scripting
* Extensive hands-on experience with **Ansible** and **Ansible Tower** as Configuration management tool, to automate repetitive tasks, quick deploys of critical applications, and proactive management of changes.
* Used **GIT** as a source code management tool for creating local repo, cloning the repo, adding, committing, pushing the changes in the local repo, saving changes for later (Stash), recovering files, branching, creating tags.
* Manage and maintain source code management with **GIT** including writing hooks, resolving issues with repositories, ensuring accessibility of environment
* Expertise in dealing with Linux Utilities**: KICKSTART, SSH, FTP, autofs, Quota Management, wrappers, User Management, Process Management, Memory Management, and Text processing tools. Log Management, Package Management, Disk Management and Swap Partition/Files Management.**

# Certifications:

* 1. [***Certified Azure Administrator***](https://www.credly.com/badges/0d44334f-0157-4f21-918a-85b6fab264a1/public_url)
  2. [***Certified Azure DevOps Engineer***](https://www.credly.com/badges/b12884e1-a45e-4407-acaa-60a77479e8da/public_url)
  3. [***Certified Kubernetes Administrator***](https://ti-user-certificates.s3.amazonaws.com/e0df7fbf-a057-42af-8a1f-590912be5460/b742be7f-1c04-4449-9135-db201aa64ff6-manikanta-gurram-1bed1eb9-040b-4f7e-b3f8-cca14c1074c5-certificate.pdf)

***Technical Skills:***

|  |  |
| --- | --- |
| **Cloud Platforms** | Microsoft Azure, Amazon Web Services, OpenStack |
| **Infrastructure Provisioning Tools** | Terraform, Azure Resource Manager, Cloud Formation |
| **Configuration Management Tools** | Ansible, Chef, Puppet |
| **CI /CD Tools** | Jenkins, Bamboo, Azure Pipelines, Gitlab Pipelines |
| **Build Tools** | Maven, ANT, Gradle |
| **Containerization Tools** | Docker, Kubernetes, Mesos, OpenShift |
| **Source Code Management** | GIT, GitHub, Gitlab, Bitbucket, Azure Repos. |
| **Logging & Monitoring Tools** | Nagios, Splunk, ELK, Data Dog, Dynatrace, Prometheus, Grafana, New Relic  Azure Service Health, Azure Network Monitor. |
| **Scripting & Programming Languages** | Python, PowerShell, Bash/Shell, JavaScript, Perl, ASP .Net VBScript, Ruby, C, PHP, Java/ J2EE. |
| **Databases** | Data Lake, Data Bricks, Cosmos DB Oracle 10g/11g/12c, DB2, MySQL, MongoDB, DynamoDB, Cassandra |
| **Application/Web Servers** | Apache Tomcat, NGINX, IIS, JBoss4.1, WebSphere, Web Logic 9.x/10.x, Gerrit |
| **Operating Systems** | UNIX, Linux, Windows |

|  |  |
| --- | --- |
| **Virtualization Platforms** | Oracle VirtualBox, VMware Workstation, Vagrant, VMware vSphere ESXi 5.x/4.X, ESX /3.x, Hyper-V |
| **Bug Tracking Tools** | JIRA, Bugzilla, Remedy, HP Quality Center, IBM Clear Quest, Mingle, VersionOne. |

**EXPERIENCE:**

***Client: Microsoft , Atlanta, Georgia Jan 2023 –Present***

***Role****:* ***Sr. Azure Cloud Infrastructure Engineer***

## Worked on Azure services including Virtual Networks, Virtual Machines, Resource Groups, VPN, Azure DNS, Network Peering, Load Balancing, Application Gateways, Azure web application, App organizations, Auto-Scaling and Traffic Manager.

* Created Azure Automation Assets, Graphical runbooks, **PowerShell** runbooks that will automate specific tasks. Expertise in deploying **Azure AD** Connect, configuring **ADFS** authentication flow, **ADFS installation** using **Azure AD Connect.**
* Worked on **Azure Automation** through **Runbooks** Creation, Migration of existing .PS1 scripts, Authorizing, Configuring and Scheduling.
* Built **CI/CD** pipeline with **Azure DevOps** for Kubernetes container environment, utilizing **Kubernetes** and **Docker images** for the runtime environment for the **CI/CD** system to build test and deploy on dev and prod environments
* Worked on **Azure DevOps**, creation of **Pipeline** for various **ARM** template deployments using Pipeline variables and variable groups and publish **Pipeline Artifacts** creations and Release **Pipeline** creations for variables stages like Dev, Test and Prod which kicks-in when the build is created.
* Implemented **Terraform** modules to automate **Azure IAAS vm’s** using **Terraform** modules and deployed **Virtual Machine Scale Sets** in production environment.
* Created and deployed Azure Automation Scripts for a multitude of applications utilizing **the Azure stack services** including Compute, Web &amp**; Mobile, Blobs, ADF, Resource Groups, Azure Data Lake, HDInsight Clusters, Azure SQL and ARM** focusing on automation, created quality gates **in SonarQube** dashboard and enforced in the Azure **DevOps YAML pipelines** to fail the builds when conditions not met.
* Setting **up CI/CD pipelines** for Code Migration within the **Azure DevOps** to migrate code from **Azure DevOps projects to Enterprise GitHub.**
* Worked on setting up Kubernetes Sandbox on **Azure Kubernetes Services (AKS)** for testing the different features and manage containerized applications by using its features include **Config Maps, Selectors, Services,** and deployed application containers as **Pods.**
* Configured an **AKS cluster** to integrate with **Azure AD**. With **Azure AD integration**, configured **Kubernetes** access based on existing identity and group membership only existing **Azure AD Users** and **Groups** can be provided with an integrated sign-on experience and access to **AKS** resources.
* Used **Azure Monitor** to create visualizations with **Azure Dashboards and Workbooks** and Drill into monitoring data with **Log Analytics** for troubleshooting and deep diagnostics.
* Utilized **Ansible** and **Ansible Tower as a Configuration Management Tool**, to robotize repetitive assignments, rapidly deploy critical applications, and proactively oversees change.
* Experience with escalated manual reporting within the **ServiceNow Architecture**, as well as offering the finest product for **present and future analytics.**
* Developed build and deploy scripts using **MAVEN** and initiated them using **Jenkins** to move from one environment to other environments. Used **Maven** to build **java-based web applications** and integrated it with **Nexus repository**.
* Coordinated and assisted developers with establishing and applying **appropriate branching, labelling/naming**
* conventions using **GIT** source control.

***Client: Conga, USA March 2022 –Nov 2022***

***Role****:* ***SR. Azure DevOps Engineer***

* Working as **Azure Devops Engineer**, involved in configuration for **Web apps/Function apps, V-net integration, HCM,** **Application gateway, App Insights, Active directory, Azure Key Vault**, Encryption and **Security on Azure**.
* Configured and implemented **storage blobs** and **Azure files** - Created **Storage accounts**, Configured the Content Delivery Network (CDN), **custom domain**, Managing access and **storage access key**.
* Experience in **Windows Azure Services** like **PaaS, IaaS** and worked on storages like **Blob** (Page and Block), **Sql Azure.** Well experienced in deployment & **configuration management** and **Virtualization.**
* Worked on the **PowerShell Script** to automate the **Windows patching** and created the release in **Azure DevOps** **pipelines**. Deployed Change Management requests and hotfixes for Microsoft Azure cloud services in all regions across the world using **PowerShell scripts**.
* Build, deploy and migrate infrastructure to Azure utilizing **ARM templates** and **Terraform**. Work with **Azure DevOps** to create all-inclusive **CI/CD** process.
* Used **VSTS** (Azure DevOps) to view project description, add members and to check the latest activity. Used  **Terraform** for **building**, **changing**, and versioning **Microsoft Azure infrastructure** safely and efficiently.
* Create and Maintain Subversion Branching, Merging and Tagging across each production releases and perform builds using **Jenkins** Continuous Integration using **Maven Build scripts**. Installed and administered **Artifactory repository** to deploy the artifacts generated by **Maven.**
* Evaluated **Kubernetes** for **Docker container orchestration** and managed **Kubernetes** charts using **Helm** and 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.**
* 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.**
* Deployed **Kubernetes Clusters** on **cloud/on-premises** environments with **master/minion architecture** and wrote many **YAML files** to create many services like **pods,** **deployments, auto scaling, load balancers**, labels, health checks etc.
* Configured applications that run **multi-container Docker applications** by utilizing the **Docker**-Compose tool which uses a file configured in **YAML** format and used **Kubernetes** to manage containerized applications using its **nodes,** Config-**Maps, selector, Services**, and deployed application containers as **Pods**.
* Created **CI/CD pipelines** for **.NET**, **Java apps** in **Azure DevOps** by integrating source codes **GitHub, VSTS**, and artifacts. Created deployment areas such as testing, **pre-production**, and **production** environment in **Kubernetes cluster**.
* Built **CI/CD** pipeline with **Azure DevOps** for **Kubernetes container environment,** utilizing **Kubernetes** and **docker** for the runtime environment for the **CI/CD** system to build test and deploy on **DEV** and **PROD** environments.
* Setup **Kubernetes** local cluster and made it up and running with **Docker Container, Minikube,** and **Kubectl.** Development of automation of **Kubernetes clusters** via playbooks in **Ansible.** Set up continuous delivery for the Pods by running **Jenkins, Docker images,** in **Kubernetes.**
* Proficient in **configuration management** tool **Ansible**, Playbooks, Inventory files to setup servers, deploying software, orchestrate fault-tolerance updates and delivery incorporating **Ansible** Tower for **access control.**
* Developed a pipeline using **Jenkins Pipeline** to automate to push **the PODs** to the **registry**, and then deploying it in **Kubernetes.** Created looper pipelines from the **GitHub** source code for creating **Docker images.**
* Associated in assisting the **Unix** and **Splunk** administrators to deploy **Splunk** across the **UNIX** and **Windows** environment. Worked on setting up **Splunk** to capture and analyze data from various **layers Load Balancers, Web** servers.
* **JIRA** has been used as **a bug tracking** tool and created, maintained and customized complex **JIRA** project configurations including **workflows, custom fields, permissions** and **notifications** & **managed** all the **bugs** and changes into a production environment using the **JIRA tracking tool.**
* Worked on implement **major changes** in development processes and procedures, customizing **ServiceNow** to take the forefront of the new systems. Created **Workflows, permission Schema**, Screens and Custom Field Schemes for multiple projects and ticketing projects using **ServiceNow.**

***Client: Edward Jones St. Louis, USA Jan 2020 -Feb 2022***

***Role****:* ***Sr. Azure Cloud Infrastructure Engineer***

***Roles & Responsibilities:***

* Utilized **Azure** **Active Directory** for single sign-on access to thousands of cloud SaaS applications and setup privileged Identity Access Management for Azure AD roles such as Application administrator and RBAC roles such as Contributor for subscription.
* Built an application environment to operate at high availability in different Azure regions by using **CDN, Azure Traffic Manager**, **App Services-Web Apps**, **Active Directory**, Storage Queues, Cognitive Services, Azure Search, Storage Blobs, Redis cache, and SQL Database.
* Employed **Azure Resource Manager (ARM)** to deploy, update and delete multiple resources on Azure and in migrating the on-premises resources to Azure with **Azure site recovery (ASR)** and **Azure backups** and created **resource group** of **virtual machines** with deployment parameters.
* Deployed and configured **AZURE AD Connect**. Also configured **Azure Network Security Groups**, **Azure Operational Insight** and **OMS,** **AZURE Premium Storage**, Network Virtual Appliance, **Client-Side encryption** for Azure storage objects.
* Configured **VNETs** and **subnets** as per the project requirement also performed PowerShell scripting to do **Patching, Imaging, and Deployments in Azure**.
* Implemented a CI/CD pipeline using **Azure DevOps (VSTS, TFS)** in both cloud and on-premises with **GIT, MS Build, Docker, and Maven.**
* Provisioned the high availability of Azure VMs using Terraform and **ARM** templates. Introduced plugins to support new functionality in **Terraform** and implemented terraform to migrate legacy and monolithic systems to **AZURE**.
* Worked as cloud Engineer, involved in configuration for **Web apps/Function apps, V-net integration, HCM, Application gateway, App Insights, Azure Key Vault, Encryption and Security on Azure.**
* Implemented **Azure Role-based Access Control (RBAC)**to segregate duties within our team.
* Responsible for implementing containerized based applications on Azure Kubernetes by using **Azure Kubernetes service (AKS),** **Kubernetes** Cluster, which are responsible for cluster management.
* Managed Virtual Networks to deploy agent nodes, **Ingress API Gateway**, **MySQL** Databases and **Cosmo DB** for stateless storage of external data, and setup reverse proxy Nginx in the AKS cluster.
* Setup and integrated **Prometheus** with **Grafana** for Monitoring. Used **Elasticsearch** (ELK stack) for centralized logging and analytics in the continuous delivery pipeline to store logs and metrics into Azure storage using storage function.
* Responsible for writing Ansible playbooks to deploy managed AKS clusters and setup microservice applications using helm manifest files, including deployment, service, secrets, and config maps.
* Provisioned **load balancer**, **auto-scaling** group and launch configuration for microservices using **Ansible** and configured **AppDynamics** to monitor **AKS** pod failures and restarts, node starvation, missing dependencies such as Services, Config maps and Secrets.
* Supervised two tier **Java, Python**web applications to **Azure DevOps CI/CD** to focus on development by using **Repos**, **Test Plans**, deployed **App Service**, **Azure Application Insights,** collects health performance and usage data of the process, stored artifacts in blob storages.
* Implemented microservices, application development, and migration using Azure services such as **Azure DevOps, Kubernetes Service (AKS)**, Container Registry, **Cosmos DB**, and **Grafana**, Azure **pipelines**.
* Supervised AWS to **Azure migration**, Jira to **Azure DevOps**Migration, **GitHub** to Azure DevOps Migration along with configuring and using PaaS offerings (like **Azure Database services, Azure Kubernetes, Azure Containers, DNS, Application gateway rerouting, Application Insights, Azure Container Monitors, etc.).**
* Utilized **Azure DevOps pipelines** to create **Azure** Infrastructure from **GitHub**repositories containing **Terraform code** and created on-premises active directory authentication using automation with **ansible** **playbooks**.
* Configured and Integrated **Docker** container orchestration framework using Kubernetes deployments and used Python Scripts to schedule, deploy and manage container replicas onto a node cluster using Kubernetes.
* Developed **Kubernetes** templates for various applications like **Cassandra**, **Grafana**, **Kafka** and setting up Kubernetes Clusters for running micro services, pushed micro services into production with Kubernetes Infrastructure.
* Managed **Kubernetes** charts using **Helm** and created reproducible builds of **Kubernetes** applications, templatized **Kubernetes** **manifests**, configured parameters to customize the deployment and releases of Helm packages.
* Created dynamic routing/load balancing capability on Kubernetes clusters enabling large application scaling, used ingress rules and **Ingress Controllers - Nginx.**
* Configured **IngressAPI** to give services externally reachable URLs, **load-balance** traffic, terminate SSL/TLS, and offer name-based virtual hosting.
* Configured a **Horizontal Pod Autoscaler** in K8 cluster that maintains replicas controlled by deployment and it increases and decreases the number of **replicas**, via the deployment, to maintain average CPU utilization.
* Defined permissions on namespace and cluster-scoped resources using **Role** and **ClusterRole** and granted access within a specific namespace and cluster-wide using **RoleBinding** and **ClusterRoleBinding** in Kubernetes Clusters.
* Utilized **Terraform** to Setup infrastructure in **Azure** Environments. Converted existing **terraform modules** that had version conflicts to utilize cloud formation during terraform deployments to enable more control and missing capabilities.
* Implemented **Terraform** Templates automating **Azure IAAS** VM's using terraform modules and deployed virtual machine scale sets in production environment.
* Configured **VNET Peering** to enable communication between resources across multiple Virtual Networks and configured **NSGs** for two tiers and three tier applications to filter network traffic using **Terraform modules**.
* 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.
* IManaged **software artifacts** required for development using repository managers **JFrog Artifactory** and integrating it with **Azure DevOps pipelines** for Continuous Integration and also built custom workflows.

***Client: Memorial Sloan Kettering, Newyork, USA Nov 2018 – Dec 2019 Role: .Azure DevOps Engineer***

***Roles & Responsibilities:***

* **Designed, Configured**, and deployed **Microsoft Azure** for a multitude of applications utilizing the **Azure stack** (including **Compute, Web & Mobile, Blobs, Resource Groups, Azure SQL, Cloud Services and ARM**), focusing on high – Availability (HA), **Fault Tolerance** (FT), and **Auto-Scaling**.
* Experienced in Designing **Azure Resource Manager** Template and in designing custom build steps using **PowerShell**.
* Experienced in configuring **Cloud platform** such as **Virtual Networks Multi-site**, **Cross-site, VMs, Azure Active** **Directory, Load Balancers, Azure SQL**, Cloud Services etc., and Created **Azure Backup vault** and protecting requires **VMs** to take the **VM** level backups in **Azure.**
* Worked on integrating **Azure Log Analytics** with **Azure Virtual Machines** for **Monitoring** the **log files**, store them and**Track Metrics**. Experience in **Azure AD Domain Service, SSO, MFA, RBAC** and **Security**.
* Designed, **configured**, and deployed **Azure Automation** **Scripts** for a **multitude** of applications utilizing the

**Azure stack** (Including **Compute,** **Web Mobile, Blobs, ADF, Resource Groups, HDInsight Clusters, Azure SQL,Cloud** **Services**, and **ARM),** Services and Utilities **focusing** on **Automation**.

* Deployed **Kubernetes clusters** using **AKS** managed local deployments in **Kubernetes** creating **local clusters** and deploying application containers and created **Microservices applications** with **integration** to **Azure services** by using **Azure AKS** while providing access to the full suite of **Kubernetes** functionality.
* Provided consulting and **cloud architecture** for premier customers and internal projects running on **Microsoft Azure** Platform for High Availability of services, **low operational costs.**
* Worked on Server less services **(Azure Functions**) created and configured **HTTP** Triggers in the **Azure Functions** with

Application Insights for **Monitoring** and performing **load testing** on the **Applications** using the **VSTS.**

* Created and maintained **Continuous Integration** (CI) using tools **Azure DevOps (VSTS**) over multiple environments to facilitate an **agile** development process which is automated and Repeatable, enabling teams to safely deploy code in **Azure Kubernetes Services (AKS) by YAML Scripting**.
* Worked on deploying the **AKS (Azure Kubernetes services**) cluster by installing the **Kubernetes through CLI** using **ACR** authentication and then configured and connected it to the **cluster**.
* Configured cluster operations in **Azure Kubernetes (AKS**) to deploy **microservices** with **CI/CD** system and created different namespaces service for each environment. Deployment in **Kubernetes** **scale up pods**, **maintain services, load** **balancing, network policies, group Docker containers** across different platform and managed entire package with **Helm charts.**
* Used **Ansible** **Galaxy** to pull the reusable modules for **ansible** and performed the **installation** and **configuration** of middleware. Integrated **Ansible** with **Jenkins** to provide automation**, continuous integration** through **Jenkins** and Implemented **Jenkins** Workflow and **Plugins** for repeatable deployments of **multi-tier applications**, **artifacts**, and services **Docker.**
* Experience in creating a **Log analytics** workspace and enabling **Cluster addon.** Leveraged and integrated it with **Azure** **Kubernetes** Service to **monitor** and figure out if requests are processed, and to inspect **Kubernetes** event or **logs**. Also monitored **Kubernetes Clusters** health by using **Prometheus**.
* Configured **Kubernetes** Replication controllers to allow **multiple pods** such as the master server in multiple minions and managed **Kubernetes charts using Helm.**
* Deployed **Windows** and **Linux** based **Docker containers** with popular Container Orchestrator tools like **Kubernetes** and **DCOS** provisioned by the **Azure** Container Service and responsible for the integration of **Container Registries**, specifically **Docker Hub** and the **Azure Container** Registry into **DevOps Workflows**.
* Developed different modules in **Terraform** to trigger the Infrastructure on the **Azure Platform** by Integrating **Selenium** Automation regression test suite with Build Pipeline and Developed **Continuous Deployment.**

***Client: BJC Health Care, St Louis, USA 2017– Oct 2018***

***Role: AWS Cloud Engineer***

***Roles & Responsibilities:***

* Designed **AWS Cloud** Formation templates to create custom-sized **VPC**, to set up **IAM** policies for **users, subnets**, and **NAT** to ensure successful deployment of **Web applications, database templates**, and **Security groups.**
* Designed **AWS Cloud** Formation templates to create custom-sized **VPC,** **subnets**, and **NAT** to ensure successful deployment of **Web applications** and **database templates**.
* Installed the application on **AWS EC2 AMI Instances** and configured the storage on **S3 Buckets.**
* Involved in migration of **on-premises applications** to **AWS cloud** environments and **AWS** to **AZURE**. Developed automation scripting in **Python**(core) using **Puppet** to deploy and manage **Java applications** across **Linux servers**.
* Experience working with **IAM** (Identity Access Management) in order to create **new accounts, roles**, and **groups.**
* Professional with expertise in **VPC, Route 53,** and **Direct Connect**. Expertise in providing **virtual private networks**, domain name system services, and dedicated network connections for **on-premises data centers.**
* Configured **Route 53** by using **CFT templates**, assigned the **DNS mapping** for the **AWS servers**, and **troubleshoot** the issues of the **Load Balancer**, **auto-scaling groups**, and **Route 53.**
* Managed **AWS infrastructure** as code (IaaS) using **Terraform**. Expertise in writing new **Python scripts** in order to support new functionality in **Terraform**. Provisioned the highly available **EC2 Instances** using **Terraform** and cloud formation and set up the **build** and **deployment** automation for **Terraform scripts** using **Jenkins**.
* Experienced with a strong background in computing services such as **EC2, Lambda, EKS**, and **ECS**. Skilled in managing scalable computing capacity and running code without managing servers.
* Involved in the design and deployment of a multitude of Cloud services on **AWS stack** such as **EC2, Route53, S3, RDS,** **Dynamo DB, SNS, SQS**, and **IAM**, while focusing on **high-availability, fault tolerance**, and **auto-scaling** in **AWS** **CloudFormation**.
* Deployed and hosted the application on **AWS,** automated single click deployment onto **AWS** using **Ansible**.
* Worked on **Docker container** snapshots, attaching to a running container, removing images, managing Directory structures, and managing containers.
* Installed **Workstation**, Wrote **Recipes** and **Cookbooks** and uploaded them to **Chef** Server, Managed On-site **OS Applications, Services**, and Packages using **Chef** as well as **AWS** for **EC2**, **S3 ELB** with **Chef** Cookbooks.
* Design and setup of **CI** tool **Jenkins** to integrate **SCM** tool **Git**, and automated the build process. Working with the Build Verification team to make sure builds are delivered within deadlines.
* Storage specialist with expertise in **S3, EBS,** and **EFS.** Proficient in providing object storage**, block-level storage**, and file storage services for **EC2 instances**.
* **Database administrator** with a deep understanding of **RDS, DynamoDB**, and **Aurora**. Proven ability to manage and maintain managed relational and **NoSQL database** services.
* Repository management in **Maven** to share snapshots and releases of internal projects using the **Nexus tool.**
* Worked in a group running **Jenkins** in a **Docker container** with **EC2 slaves** in **Amazon AWS** **cloud configuration**. Also, gain familiarity with surrounding technologies such as **Mesos (Mesosphere**) and **Kubernetes**.
* Integrated Testing Tools like **Junit,** and Coverity in **Jenkins** which provide **unit** testing and Code coverage with detailed graphical reports.
* Configured and Managed **AWS EC2 instances** utilizing **Auto Scaling, Elastic Load Balancing**, and Glacier for our **QA** and **UAT environments** as well as infrastructure servers.
* Used **Chef** cookbooks to provision and manage the container's host and use **Chef** Provisioning with **Docke**r to deploy a new running container. Deployed **microservices** with **Chef cookbooks** in **Docker containers**.
* Built **Puppet** enterprise modules using **Puppet DSL** to automate infrastructure provisioning and configuration management to existing infrastructure by deploying **Puppet**, and **Puppet Dashboard**.
* Deployed and configured **Elasticsearch, Logstash**, and **Kibana** (ELK) for **log analytics**, **full-text search**, and application monitoring in integration with **AWS Lambda** and **CloudWatch**.
* Expertise in **logging IAM**, **CloudWatch**, and **CloudTrail**. Proficient in managing access to **AWS resources** and providing monitoring and audit logging capabilities.
* Used **Splunk f**or monitoring server’s stats**, issue identification**, and **security analysis**. Used **Jira** as a **ticket tracking, bug** **tracking**, and **workflow tool.**
* Created **Docker** images from scratch and customized them as per the requirement, worked on **Docker containers**, Snapshots, managing **Docker** volumes, directory structure and removing **Docker images**.
* Built end-to-end **CI/CD** Pipelines in **Jenkins** to retrieve code, compile applications, perform tests and push build artifacts to **Nexus** and Orchestrate changes across servers and components and configured **SonarQube** for code analysis in **Jenkins**.
* Created **Docker** images from scratch and customized them as per the requirement, worked on **Docker containers**, Snapshots, managing **Docker** volumes, directory structure and removing **Docker images**.
* Developed microservices on tools leveraging **Python** and **Jenkins** allowing for easy creation and maintenance of build jobs using **Jenkins pipeline** and deployed the services on **Kubernetes**.

***Client: Cerner Corporation, St Louis USA June 2016 – May 2017 Role: AWS DevOps Engineer***

***Roles & Responsibilities:***

* Designed highly available, cost-effective, and fault-tolerant systems using multiple **AWS EC2 instances, Auto Scaling, Elastic Load Balance** and **AWS Amazon machine images, DNS mapping** with subdomain using **Route 53**.
* Managed multiple **AWS** accounts with multiple **VPC's** for both production and non-production where primary objectives included automation, build-out, integration and cost control.
* Hands-on experience in migration of one Business from a physical data center environment to **AWS**. Worked at optimizing volumes and **EC2** instances and created multiple **VPC** instances, deployed and monitored scalable infrastructure on **Amazon web services (AWS).**
* Authored **Terraform** modules for infrastructure management. Authored and published a module to the **Terraform** registry for enterprise customers to deploy our product inside their **AWS** environment.
* Designed roles and groups for users and resources using **AWS Identity Access Management (IAM)** and implement **Multi-Factor Authentication (MFA)** on externally available servers and on AWS Console, Monitoring and alerting leveraging **Cloud Watch** and **Cloud Trail**.
* Experienced in implementing Data warehouse solutions in **AWS Redshift**, worked on various projects to migrate data from one database to **AWS** **Redshift**, **RDS**, **ELB**, **EMR**, **Dynamo** **DB** and **S3**.
* Involved in analysis, architecture design and creation of **AWS Virtual Private Cloud** (VPC), **Elastic compute cloud** (EC2 instance), **FTP server with Elastic IP**, **AWS Load Balancer**, sending email and pager duty services when new instances launch using cloud watch.
* Expertise in **Terraform** for creating **stacks** of **VPCs, ELBs**, Security Groups, **SQS queues, S3 buckets** in **AWS** and updated the **Terraform Scripts** based on the requirement on regular basis.
* Implemented **Terraform** modules for the deployment of applications across multiple cloud providers. Worked with **Terraform** key features such as Infrastructure as code, Execution plans, Resource Graphs, Change Automation. Hands -on experience in writing **Terraform modules** to manage **AWS** infrastructure, for automatic creation of **RDS** instances, **VPCs, Autoscaling groups, Load balancers, SQS, S3 buckets***.*
* Experience in working with **AWS EC2** Container Service plugin in **Jenkins** which automates the **Jenkins** master-slave configuration by creating temporary slaves.
* Designed end-to-end automation of infrastructure and continuous delivery of the applications by integrating cloud formation scripts**, Jenkins**, **AWS** & **Ansible** playbooks.
* Configured and maintained the **AWS Lambda function** to trigger when there is a **Jenkins** build kicked off, which internally gets stored on **AWS S3 bucket** for everybody to access.
* Built end-to-end **CI/CD** Pipelines in **Jenkins** to retrieve code, compile applications, perform tests and push build artifacts to **Nexus** and Orchestrate changes across servers and components and configured **SonarQube** for code analysis in **Jenkins**.
* Created **Docker** images from scratch and customized them as per the requirement, worked on **Docker containers**, Snapshots, managing **Docker** volumes, directory structure and removing **Docker images**.
* Developed microservices on tools leveraging **Python** and **Jenkins** allowing for easy creation and maintenance of build jobs using **Jenkins pipeline** and deployed the services on **Kubernetes**.

***Client: Capgemini, Hyderabad, India Feb 2015– May 2016 Role: Build and Release Engineer***

***Roles & Responsibilities:***

* Configured and **maintained stress servers in different geographical locations and provide setup in every release to perform stress** testing.
* Involve in **building, configuring**, and provisioning **red hat Linux servers** using **Kickstart** server as required for the project.
* Hands-on working with **LDAP configuration**. Configuration of **Groups, Roles**, and **Policies**. Assigning **users** to **Groups** and **Roles**.
* Able to create **scripts** for system administration, **AWS** using languages such as **BASH** and **Python.**
* Created a **continuous integration** system using **Ant, Jenkins**, and **Puppet full automation**, **Continuous Integration**, faster and flawless deployments and worked with **ANT tool** to do the **builds**, integrated **ANT** to **Eclipse**, and did local builds. Automated the **release pipeline** to achieve zero-touch deployments using **Jenkins**, and **GIT.**
* Wrote **ANT** and **MAVEN Scripts** to automate the build process. Written **Maven scripts**, **Installed Jenkins**, written shell script for end-to-end build and **deployment automation.**
* Experience on **Hudson/Jenkins** and **Bamboo** for **continuous integration** and for end-to-end Automation for all the **builds** and **deployments**.
* Expert in using **Jenkins** and **Bamboo** to create pipelines involving several **Downstream/Upstream** job configurations based on dependencies from other **applications** and **Release methodologies**. Configured Source Code Management tools with **Bamboo** and executed triggers in **SVN.**
* Managed **TCP/IP packets** & **DHCP servers**, resolved **TCP/IP network access** problems for the clients, and worked with various **TCP/IP implementations** like **NFSv4, NIS, DNS**, and **DHCP**.
* Written **Chef Cookbook**, recipes to automate the installation of Middleware Infrastructure and configuration tasks for **new environments**. Experience in using **Cruise Control** and **Bamboo** as CI **Tools**. Wrote **Chef Recipes** to manage configurations and automate the installation process.
* Performed installation and managed **Grafana** to visualize the metrics collected by **Prometheus**.
* Developed **Splunk dashboards** with **HTML, JavaScript,** and **CSS** to provide a professional lookup as part of an enterprise dashboard project for management.
* Hands-on with **GIT / GIT Hub** for code check-ins/checkouts and branching and various other functions.
* Configured **Git /Stash, Maven,** and **Bamboo**-based software configuration and release activities in central **Git** repositories and **Bamboo** Build, Branching, Tagging, Building and Release.