**S.Gopi**

**Email: gopibigdata45@gmail.com**

**Mobile: +91 8978552745**

**Professional Summary**

* A cloud enthusiast with **9+ years**of experience in the IT industry that includes **DevOps,Linux Systems administration, Build and Release Management, configuration Management, Containerization based** on various cloud platforms such as **Amazon Web Servers, Microsoft Azure** and **Google Cloud Platform**.
* Proficient knowledge on **SDLC phases** and implemented the principles of **Software Configuration Management** in **Agile, Scrum** and **Waterfall** methodologies and **server-side deployments** in application and middleware layers.
* Experienced in using **AWS Cloud** components like **EC2, VPC, ELB, Auto Scaling, Security Groups, Route53, IAM, EBS, AMI, RDS, S3, SNS, SQS, Cloud Watch, Cloud Formation, Cloud Front** and **Direct Connect**.
* Experienced in creating **ELK (ElasticSearch, Logstash** and **Kibana)** for centralized logging and used **AWS Lambda** **function** to store logs and metrics in **S3 buckets** and executed the code and managed **AWS servers.**
* Experienced in developing and maintaining **Infrastructure as a Code**, and worked with **Cloud Formation** templates and **Terraform** features like Execution plans and automated staging and production environments using **CI/CD**.
* Implemented various **Azure Services** such as **Azure storage, IIS, Azure Active Directory (AD), Azure Resource Manager (ARM), Azure Storage, Azure Blob Storage, Azure VMs, Azure SQL Database, Azure Functions, Azure Service Fabric, Azure Monitor, Application Gateway, Azure Migration and Azure Service Bus**.
* Designed and deployed **Microsoft Azure** solutions across variety of cloud models like **SaaS**, **PaaS**, **IaaS**, and integrated **on-premise** and **on-demand** workloads with the **Azure Public Cloud** and also, deployed **Azure IaaS virtual machines (VMs)** and **Cloud services (PaaS role instances)** into secure **VNets** and **subnets**.
* Worked on **Google Cloud Platform (GCP)** services like **Compute Engine, Google Cloud Load Balancing, Google Cloud Storage, Google Cloud SQL, Google Stack Driver Monitoring and Google Cloud Deployment Manager**.
* Experienced in **Kubernetes** to deploy **scale, load balance,** and **manage Docker containers** with multiple names spaced versions and created reproducible builds using **Helm charts**.
* Used **Kubernetes** to orchestrate deployment, scaling, clustering and management of containerized applications as Pods, **Docker swarm** in managing **Docker containers** and **Kubernetes** **Clusters** to support automatic high availability, load balancing, health checking and failover.
* Experience with **Docker** **Engine** and **Docker Machine** to deploy micro services-oriented environments for scalable application to Ship, Run, and deploy the application securely to speed up the Build/Release Engineering with experience utilizing **Docker** for **virtualization**.
* Involved in setting up **Jenkins Master**and multiple slaves for the **Continuous Integration and deployment process** of application and database servers.
* Worked on writing **Jenkins** pipeline with **Gradle** script and **Groovy DSL** **(Domain Specific Language)** and integrating ANT/MAVEN build scripts with **Gradle** for continuous build.
* Experienced in using unit test plugins like finding bugs, check style, code quality and code coverage by using **SonarQube scanner** and **runner**.
* Developed custom **Jenkins** **jobs/pipelines** that contained Bash shell scripts utilizing the **AWS** **Command Line Interface (CLI)** to automate infrastructure provisioning.
* Experienced in Integrating **Ansible** with **Jenkins** **CI/CD** pipelines to automatically populate **AWS Cloud formation** templates using **Jinja** Templating Script.
* Used **Ansible** for environment automation and configuration management by developing **Ansible** **playbooks** to deploy, manage, test and configure web/application servers onto the hosts.
* Worked with **Ansible tower**as **configuration management** tool to deploy the application to multiple servers and to manage deployments through a dashboard.
* Experience in Working on **Chef** using **Knife** commands to manage Nodes, **Cookbooks, Chef Recipes, Chef attributes**, **Chef Templates** and extensively used Ruby scripting on Chef Automation for creating cookbooks.
* Experience in working with **Puppet** **Enterprise** and **Puppet** **Open** **Source**. Installed, configured, upgraded, and managed **Puppet** masters, Agents, Databases and implemented custom Modules Manifests.
* Used build tools like **Maven** and **Ant** for building deployable artifacts such as **jar**, **war** & **ear** from source code and **Artifactory Repository** like **Sonatype**, **Nexus** and integrated them with **CI/CD** tools like **Jenkins**.
* Proficiency in Branching, Merging, Tagging and maintaining the version control tools across the environments by using SCM tools like **Subversion** **(SVN)**, **GIT**, **GitHub**, and **Bitbucket**.
* Experienced with **Splunk** data mining, log file reporting, Searching and Reporting modules, Knowledge Objects, Administration, Add-On's, Dashboards, Clustering and Forwarder Management.
* Experience with **Nagios**as a monitoring toolto identify and resolve infrastructure problems worked on **Nagios**event handlers in case of the automatic restart of failed applications and services
* Implemented **AWS cloud watch** logs for monitoring AWS resources and applications which includes **VPC flow logs, ELB logs**, and **RDS logs**. Created **AWS** customized Scripts to monitor various application and system level **EC2** **Instance** metrics and trigger alarms based on defined metrics.
* Managed **Monitoring Tools** such as **Prometheus, Splunk, Nagios and Azure Monitor** for Resource Monitoring, Network Monitoring and Log Trace Monitoring.
* Experience in writing various Automation Scripts to automate manual tasks, deploy applications, application build scripts/versioning etc. using open source libraries with **Python based scripting**.
* Developed **Bash**, **Python** **and PowerShell** scripts for Automation of Activities and builds in **Devops** lifecycle.
* Experience with Bug tracking tool like **JIRA**, **Remedy**. Experienced in Branching, Tagging, and maintaining the versions across different Source Code Management (SCM) tools like **GitHub**, **Subversion** (SVN), and **CVS**.
* Proficient in Querying **RDBMS** like **Oracle**, **MySQL, SQL**, **NoSQL,** **DB2**, **PostgreSQL**, **MongoDB** and **Cassandra**. Created Database, Tables, Views, Stored Procedures, Indexes (cluster/Non-cluster), and Constraints, Triggers, Buffer Pools, User Roles, and functions.

**Technical skills:**

|  |  |
| --- | --- |
| **Cloud Platforms** | AWS, Azure, GCP |
| **Source Code Management/Version**  **Control Tools** | SVN, GIT, Bitbucket, GitHub |
| **Build Automation Tools** | ANT, Maven, Gradle |
| **Continuous Integration and**  **Deployment Tools** | Jenkins, Azure TFS |
| **Artifact Repository Management Tools** | Azure artifactory, Nexus, JFrog |
| **Configuration Management Tools** | Chef, Ansible, Puppet, Terraform, Cloud Formation |
| **Containerization and Orchestration**  **Tools** | Docker, Docker Swarm , Kubernetes, EKS |
| **Bug Tracking Tools** | JIRA, Azure Boards |
| **Monitoring Tools** | Nagios, Splunk, ELK stack, Cloud watch |
| **Scripting Languages** | Shell (Bash), Python, PowerShell, Ruby, YAML/JSON |
| **Web Servers** | Apache, Tomcat, JBoss, WebSphere, WebLogic |
| **Operating systems** | Linux: RHEL 4.x/5.x/6.x/7, CentOS 5.x/6.x/7&Ubuntu 12.x, 14.x&  Windows 2008/2012, Debian |
| **Databases** | MS SQL, No SQL, DB2, SQL Server, Mongo DB, Cassandra |

**ORGANISATIONAL EXPERIENCE**

* Currently Working with **RIA ADVISORY** from June -2022 to till Date.
* Working with **Adecco** from Aug -2021 to June 2022.
* Working with **speqta technologies pvt ltd** from May-2020 to July 2021 .
* Working with **WishWorks** from Jan 2020 to April 2020**.**
* Working with **Infosys Limited** from November 2017 to Jan 2020

* Working with  **Atos Syntel** from july2014 to November 2017.

**Education Details**

* Bachelor of Technology (B.Tech) from JNTUK University in the year 2014.

**Professional Experience:**

**Project 1:**

**Client : AST**

**Sr. Cloud/DevOps Engineer**

**Description:** AST is the largest independent stock transfer agent in the United States. I am responsible for the migration of infrastructure from On-premise to the AWS and Azure Cloud Platforms. As cloud engineer I am responsible to create the infrastructure, manage build and deployment pipelines, Configuration management and Orchestration of Web/Application Servers.

**Responsibilities:**

* Created AWS Launch configurations based on customized **AMI** and used this configuration for auto scaling groups and Implemented AWS solutions using **EC2, S3, DynamoDB, Route53, EBS, Elastic Load Balancer, Auto scaling.**
* Configured **Elasticsearch**, **Logstash**, and **Kibana (ELK)** for log analytics, full-text search and application monitoring by integration with **AWS Lambda** and **Cloud Watch**.
* Designed and deployed a multitude of applications utilizing almost all the **AWS** **stack** include **SNS, SQS, LAMBDA, REDSHIFT** which focus on high-availability and also, Managed **Cloud Trail** logs and objects within each bucket.
* Managed solving manual redundant infrastructure issues by creating **CloudFormation** Templates using AWS's Server less application model and deploy **RESTFUL** **API's** using **API gateway** and triggering Lambda Functions.
* Managed security groups on **AWS**, focusing on high-availability, fault tolerance, and auto scaling using **Terraform** templates along with Continuous Integration and Deployment with **AWS Lambda using CloudFormation** templates.
* Developed VMs availability sets using the **Azure**portal for **IaaS**based solution and **Virtual Machine Scale Sets (VMSS)**using **Azure Resource Manager (ARM)**to manage **network traffic.**
* Managed **Azure** Infrastructure **Azure Web Roles**, **Worker Roles**, **VM Role, Azure SQL**, **Azure Storage**, **Azure AD** Licenses, **VM Backup** and Recover from a Recovery Services **Vault** using **Azure PowerShell** and **Portal**.
* Managed **Kubernetes K3s** for self-healing deployment objects for high availability and scalability of application servers using horizontal pods and resources management.
* Used **Kubernetes** to orchestrate deployment, scaling, clustering and management of containerized applications as Pods, **Kubernetes** in managing **Docker containers** and Kubernetes Clusters to support automatic high availability, load balancing, health checking and failover.
* Deployed **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**.
* Implemented **Docker** **Engine** and **Docker** **Machine** to deploy micro services-oriented environments for scalable applications. Ship, Run, and Deploy the application securely to speed up the Build/Release Engineering.
* Created **Docker** images using a **Docker** file, worked on **Docker** container snapshots, removing images and managing Directory structures for **Docker containers** and **Docker volumes.**
* Implemented **multi-stage** pipelines using **Jenkins** shared Library to ensure successful build of downstream dependencies and customized **CI/CD** processes with project parameters, build triggers and adding post build actions.
* Used **Jenkins** pipelines to drive all microservices builds out to the **Docker registry** and deployed on **Kubernetes clusters** for DEV, TEST, and PROD environments.
* Resolved **Merge Conflicts configured triggers** and queued **new builds and troubleshooted the build issues** within the released **Jenkins pipeline** to **monitor** and **track** the deployments.
* Created inventory in **Ansible** for automating the continuous deployment and developed playbooks using **YAML** scripting to provisioning the infrastructure onto the AWS Cloud.
* Developed **Ansible Playbooks** with **Python script** as the Wrapper to Manage Configurations of AWS Nodes and Test **Playbooks** on AWS instances using **Python**.
* Worked on various automation infrastructure activities like configuring resources, application server setup, stack monitoring using **Ansible playbooks** and integrated to **Ansible** using **Jenkins**.
* Used **MAVEN** as a build tools on java projects for the development of build artifacts on the source code and deployed artifacts into **Nexus** (Jar and War file) repository and defined dependencies and plugins in **Maven** **pom**.**xml**.
* Automated **Jenkins** **jobs** to automatically check out code from **Git** by **Poll** **SCM**, and scheduled jobs as night builds and configured Maven test, compile and build.
* Developed customized **Shell and python** scripts to support the existing **Jenkins** **pipelines** such as creating Virtual Environments, Configuring deployment logics and Automation.
* Maintained the monitoring and alerting of Non-production and production servers/storage using **AWS** **Cloudwatch**.
* Integrated **Splunk** Enterprise with Deployment pipelines to monitor the performance of the application and managed **Splunk** Forwarder on centralized deployment server.
* Created **python scripts** which Integrates with **Amazon API** to control instance operations within the infrastructure.
* Experienced in installing application dependencies like **Apache Tomcat, JBoss, and WebSphere** as part of executing user Data using Ansible Playbooks.
* Created detailed documentation of complex **build and release process**, post release activities process, **JIRA** workflow and Release notes.

**Environment:** AWS, Azure Cloud, Cloud Formation, Cloud Trail, Terraform, Kubernetes, Docket, Jenkins, Ansible, Maven, Cloudwatch, Splunk, Git, Shell, Python, YAML, Nexus, Nagios, Apache Tomcat, JBoss, WebSphere, Jira.

**Project 2:**

**Client :Syniverse**

**Role:Cloud/DevOps Engineer**

**Description:** Syniverse provides technology and business services for a number of telecommunications companies as well as a variety of other multinational enterprises. I am responsible for infrastructure creation to deploy micro services hosted on the Google Cloud. My day to day responsibilities include developing Terraform modules to support IaC, Troubleshooting Build and Deployment issues, monitoring the deployments and providing support to the existing infrastructure.

**Responsibilities:**

* Worked on **AZURE** (**IaaS**) by creating **AZURE** **Virtual Machines**, storage accounts, **VHDs**, storage pools, migrating on-premise servers to **AZURE** and creating availability sets in **AZURE.**
* Implemented Server-less services (**AZURE** Functions) created & configured **HTTP** triggers in **AZURE** Functions with Application insights for monitoring and performing load testing on the Applications using the **Visual Studio Team Service (VSTS)**.
* 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 and Sync to standby database in another region.
* Used **Google stack driver** for monitoring the logs of both **Google Kubernetes** **Engine** (**GKE)** and **GCP** instances and configured alerts from Stack driver.
* Set up Auto Scaling of the instance group for DEV/QA environments using **GCP CLI** (Command line tools) and **GCP** cloud environments, and provided proof of concept (POC) on creating server-less architecture with **GCP**.
* Worked on developing **Terraform** templates to automate the **Azure** **IaaS** infrastructure using **Terraform** modules and deployed virtual machine scale sets in production environment.
* Configured **AzureKubernetes** Service to deploy and managed **Kubernetes** cluster in **Azure** and created an **AKS** cluster in the **Azure** portal, with the **Azure** **CLI**, also used template driven deployment options such as Terraform.
* Containerization of applications with **Kubernetes cluster** and played a key role in building in- house **PaaS** environment for web servers. Automated **kubernetes** cluster for **Ansible Tower.**
* Worked on **Kubernetes** and **Docker** for build, test, deploy and containerizing the CI/CD application to deploy on cloud and managed container replicas on node clusters using **Kubernetes**.
* Worked on virtualizing the servers using the **Docker** for the test environments and dev-environments and also, configuration automation for test environment using **Docker** containers.
* Responsible for Installation and configuration of latest **Docker** version to Datacenter including **universal control plane (UCP)** and **Docker trusted registry (DTR)** on a cluster of host’s environment.
* Built code repository like **Team Foundation Server (TFS**) for **Git** to maintain different version of the application to keep track of changes and for peer code reviews.
* Developed **VSTS** **pipelines** to drive all **microservices** builds out to the Docker registry and deployed on **Kubernetes**, Created Pods and accelerated using **Kubernetes**. Created private cloud using that supports DEV, TEST, and PRODS.
* Responsible for Administering and Monitoring **Visual Studio Team System (VSTS),**taking backups and consolidating collections at the time of migration from one version of **VSTS**to another.
* Automated repetitive tasks and quick deployments for the critical applications and managed multiple nodes and inventories for different environments using **Ansible Tower**.
* Deployed and configured**Ansible Server,**experience in writing **Ansible Modules** to automate repetitive tasks, deploying critical applications, managing the changes in instances and managing multiple nodes.
* Worked on managing **Ansible Centralized Server**and created playbooks to support various middleware application servers and configuring the Ansible tower as a configuration management tool to automate repetitive tasks.
* Automated Weekly releases with **Maven** scripting for Compiling **Jar/War/Rar** files, Debugging and Placing Builds into **Maven** Repository.
* Used GIT Plugins in **Jenkins** to automate triggering builds for latest code commit in **GIT** and stored the .jar, .war files into **Nexus** and used **Kubernetes** as an Orchestration tool to deploy those files to desired environments.
* Maintained **Bitbucket** Repositories which includes **Jenkins** for Integration, creating new repositories, enabling Git to ignore, branching, merging and creating pull requests and the access control strategies from **Bitbucket.**
* Developed **Power Shell scripts** to automate the Azure cloud creation including end-to-end infrastructure and VMs.
* Involved in server side administration by deploying Web servers like **Apache HTTP**, **Nginx** and **Httpd** on-to the cloud based servers.
* Used **Nagios** to monitor the day-to-day performance and maintenance operations throughout the client’s network.

**Environment:** Azure, VSTS, TSL, GCP, VMware, SQL DB, Terraform, Kubernetes, Docker, Git, Ansible, Maven, Jenkins, MySQL, Nexus, Bitbucket, Powershell, HTTP, Apache, Nginx, Nagios, Splunk

**Microland, India**

**Build and Release Engineer**

**Description:** Microland is an IT infrastructure service based company**,** being a part of service team. I am responsible for configuring the build and release activities on web applications and responsible for maintaining and deploying builds in UAT and Production environments.

**Responsibilities:**

* Experience on **AWS** for deploying using Code commit and Code deploy of **EC2** instances consisting of various platforms like **Amazon Linux AMI, Red Hat Linux Enterprise, SUSE Linux, Ubuntu server**.
* Created a microservice environment on the cloud by deploying services as docker containers. We used **Amazon ECS** as a container management service to run **microservices** on a managed cluster of **EC2 instances**. Implemented **Amazon API** Gateway to manage, as an entry point for all the API's.
* Build servers using **AWS,** importing volumes, launching **EC2**, **RDS**, **creating security groups, auto-scaling, load balancers** ( **ELB** s) in the defined virtual private connection and open stack to provision new machines for clients.
* Designed **Cloud formation** templates for creating infrastructure for creating stacks in different Environments by just changing the Parameter Values in the **AWS** Environment.
* Consistently used **AWS** **CloudWatch** to monitor the cloud infrastructure and use generated metrics for analyzing the performance of the applications. Configured **SNS** for alerting the users on crossing the metric thresholds.
* Configured **GCP** Firewall rules in order to allow or deny traffic to and from the **VM's** instances based on specified configuration and used **GCP** content delivery network to deliver content from **GCP** cache locations.
* Implemented **Kubernetes (k8s)**clusters for running **microservices** and pushedmicroservices into **GCP** **Cloud** with **Kubernetes** backedInfrastructure. Automation of **Kubernetes**clusters via playbooks in **Ansible**.
* Containerized services in **Docker** for Confidential environment by Installing and Configured the **JFrog** repository manager for sharing the artifacts within the company.
* Virtualized the servers using the **Docker**for the test environments and development environment and performed configuration automation using **Docker** containers.
* Installed and Configured **Jenkins** pipelines to support **Groovy** script for various Java build. Also, installed & configured various **Jenkins** plug-in to automate Continuous build & deployments.
* Managed **Jenkins** as **CI/CD** tool to creating new jobs, plugins, configuring the management tool, build trigger, build system and post build actions, scheduled automatic builds, notifying the build reports.
* Created **Jenkins Workflows** for pre-deployment process (DB execution, Environment, configuration changes) on both QA and preproduction Environments.
* Worked on **bootstrapping** nodes with **Chef** **Server** for on-site application, cookbooks and wrote recipes and uploaded them to **Chef** **Server** to manage on-site application and services.
* **Developed various Chef Cookbooks, for Infrastructure Configuration and Deployment Automation using Roles, Environments, Attributes/resources, Secure Data bags with vault, and ERB templates.**
* Expertise 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.
* Collaborated with development teams with **Subversion**/**Maven** Installations, **Maven** dependency management strategies and branching, and Release Strategies for multiple releases.
* Worked on **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.
* Expertise in Querying **RDBMS** such as **Oracle** and **MYSQL** by using **SQL** for Data integrity, Installation, Configuration and Management of **NoSQL** tools such as **SQL Server**, **DB2**, **PostgreSQL**, **MongoDB** and **Cassandra**.
* Expertise in maintaining Infrastructure as a Code, AWS Cloud formation templates by developing **JSON/YAML** scripts, and python scripts to automate repetitive tasks.

**Environment:** AWS,GCP, Kubernetes, Ansible, Jenkins, Docker, JFrog, Groovy, Chef, Maven, Nexus, Redhat, RDBMS, Oracle, SQL, NoSQL, PostgreSQL, DB2, MangoDB, JSON, Ubuntu**,** Linux, YAML Ant, Cassandra, Jira, VMware, GIT.

**Landmark IT Solutions, India May 2013 - Dec 2014**

**Linux Administrator**

**Responsibilities:**

* Maintained the user accounts**IAM** Roles, configured  **Route 53**(**CNAME**) to access the privately exposed application on AWS ECS Clusters, Integrated **DynamoDB**, **SQS** and automated push notifications using AWS **SNS** service.
* Worked on configuring and deploying private and public subnets for the **VPC**, and attached Internet and **NAT** Gateways along-side with Security Groups with specific In-bound/Out-Bound Rules.
* Installed and configured an automated tool **Puppet** that included the installation and configuration of the **Puppet** master, agent nodes and an admin control workstation.
* Used **Puppet** to deploy **ELK** for automating Continuous Deployment (CD) and configured Slave Nodes and deployment failure reporting.
* Deployed **Puppet**, Puppet Dashboard and **Puppet DB** for configuration management to existing infrastructure.
* Automated the process of creating release packages for multiple configurations on **Unix** **Platforms**, using **ANT**.
* Involved in design, configuration, installation, implementation, management, maintain and support for the corporate Linux servers **RHEL** 3, 4, 5, **CENTOS** 5, **Ubuntu**.
* Installation, Configuration and upgrade of **Redhat Linux (2,3,4,5), CentOS, Oracle Linux, SUSE, Solaris, HP-UX & Windows 2000 / 2003/2008** Operating System.
* Developed automated processes that run daily to check disk usage and perform clean-up of file systems on UNIX environments using **shell scripting** and **CRON**.
* Automating the daily tasks in **Subversion** and streamlining the process of access to SVN for migrated Projects from **SVN**to **GIT**.
* Worked on configuring the **Jenkins** to use **MetaCase** Software to build Java code and also to do the whole C.I process on the java code generated by **MetaCase**
* Configured **LAN**, **WAN** and managed Network troubleshooting applications TCP/IP including Ethernet, IP addressing & Sub netting, routing.
* Worked on **RPM** package manager to **install, update, verify, query** and erase software packages on **Linux Servers** and worked on automation of activities using **bash** **scripting**.
* Managing multiple Active Directory domain controllers, user accounts and **DNS** and **DHCP** services.

**Environment:** AWS, Puppet,RHEL, Windows, HP-UX, CentOS, Ubuntu, Subversion, Git, Metacase, RPM, Oracle, Solaris, Shell Scripting, My SQL, CRON, Jenkins, UNIX, Linux.