|  |  |
| --- | --- |
| Saumya Gali **Cloud/DevOps Engineer** Email : [saumyagali@gmail.com](mailto:saumyagali@gmail.com)Phone no : (346) -327-4422 **LinkedIn :** https://www.linkedin.com/in/saumyagali/ | Microsoft Azure Fundamentals Online Training | K21 Academy |AWS Certified Developer – Associate |

**SUMMARY**

Certified **AWS Developer Associate** and **Azure Fundamentals** with around 7+ years of IT experience as **Cloud DevOps Engineer**, **Site Reliability Engineer,** **Build & Release Engineer**, **Linux Administrator** and **Systems Engineer** in SCM, Build/Release Management with Continuous Integration & Continuous Deployment/Delivery as an iterative process and Automation of infrastructure including end-to-end code configuration using different tools, and cloud platforms like AWS, Azure, OpenStack, PCF, Ansible, Jenkins, Azure DevOps.

* Experience in designing, deploy and managing mission critical applications and working on Windows, UNIX and Red Hat Linux platform executing DevOps principles to implement and manage CI/CD frameworks in on-premises and cloud.
* Knowledge on managing hosting plans for **Azure Infrastructure**, implementing & deploying workloads on **Azure virtual machines (VMs)**, design and implement VM storage (images/disks).
* Hands-on experience in **Azure** development, worked on **Azure web application**, **App services**, **Azure storage**, **Azure SQL Database**, **Virtual Machines**, **Fabric Controller**, **Azure AD**, **Azure search** and notification hub.
* Experience in migrating an On-premises Instances or **Azure Classic Instances** to **Azure ARM Subscription** with **Azure Site Recovery** and **Open Virtualization Format (OVF) templates**.
* Proficient in core **AWS services (S3, EC2, ELB, EBS, Route53, VPC, Auto scaling etc.)** and **deployment services (Elastic Beanstalk, Ops Works and Cloud Formation)** and **security practices (IAM, Cloud Watch and Cloud trail)**.
* Experienced in setting up **Amazon EC2 instances**, **virtual private cloud (VPCs)**, and **security groups**, **Data Pipeline** and **Lambda**. Setting up databases in **AWS** using **RDS**, storage using **S3 bucket** and configuring instance backups to S3 bucket.
* Proficient in **Kubernetes** to manage containerized applications using its **nodes**, **Config Maps**, **selector**, **Services** and deployed application containers as **Pods**.
* Worked with **Docker** and **Kubernetes** on multiple cloud providers, from helping developers build and containerize their application **(CI/CD)** to deploying either on public or private cloud.
* Expertise in integrating **Docker** containers infrastructure and Continuous Integration for building & deploying Docker containers and excelled on creating **AMI (Amazon Machine Images**) that utilizes **ELB (Elastic Load Balancer)** and **Auto Scaling**.
* Experience in writing **Ansible playbooks** to deploy automated tasks to multiple servers, managing configurations on **AWS** and test playbooks on AWS instances and written **Ansible Scripts** to assist **Dev servers**.
* Experience in managing Ansible Playbooks with **Ansible roles** and used **Ansible Tower** to manage Multiple Nodes and Manage Inventory for different Environments.
* Worked with **Ansible** playbooks for virtual and physical instance provisioning, configuration management, patching and software deployment on **OpenStack** environment.
* Experience in setting up Chef Workstation, boot strapping various enterprise nodes, setting up keys, and various other concepts of Chef like **Roles**, **Environments**, **Data Bags**, **Knife**, **Chef Server Admin/Organizations**.
* Hands on experience with **Chef** configuring the **chef-repo** and setting up multiple **Chef Workstations**; wrote **Chef Cookbooks** and **Recipes** to automate the deployment process and integrated with **Jenkins** jobs for **Continuous Delivery (CD) framework**.
* Experience in designing and implementing Cloud Automation and orchestration framework in private/public cloud environments involving **AWS APIs, OpenStack, VMware, Chef, Puppet, Python, Azure APIs, and Workflow Engine.**
* Experience in developing automation code using opscode **Chef** and **Python** to build **Azure** and **AWS** environments autonomously.
* Extensive knowledge on installing, configuring, and administering **Jenkins CI tool** in Linux machines and used **Jenkins Pipeline** to drive all microservices builds out to the Docker registry and then deployed to **Kubernetes**, Created **Pods** and managed using **Kubernetes**.
* Proficient in building **Jenkins jobs** to create **AWS infrastructure** from **GitHub repos** containing code and administered/engineered Jenkins for managing weekly builds.
* Experienced in writing scripts using **ANT**, **Maven** build tools and POM files for generating java-based artifacts such as JAR, WAR and EAR from source code.
* Proficient in the installation and configuration of web servers like **HTTP Web Server**, **Apache Web Server** and other servers on **Ubuntu**, **Red hat Linux**, **Fedora**, **CentOS,** and **Amazon Linux instances**.
* Expert in setting up SSH, SCP connectivity between Linux hosts and experienced in **DNS, NIS, NFS, CIFS, FTP, SAMBA, LDAP, remote access and security management**.
* Experience with Administration/Maintenance of Source Control Management Systems such as **GIT, Subversion (SVN)** on Linux and windows platform. Knowledge on setting up project Baselines, **Branching**, **Merging**, and **Tagging** source code in GIT.
* Exposed to all aspects of **Software Development Life Cycle (SDLC)** and In-depth understanding of the principles.
* Good understanding of the principles and best practices of **Software Configuration Management (SCM)** in **Agile** and **SAFE methodologies**.

**TECHNICAL SKILLS**

|  |  |
| --- | --- |
| **Cloud Technologies** | Azure, AWS, OpenStack |
| **Operating Systems** | Ubuntu, CentOS, Redhat Linux, Windows |
| **Build Tools** | Maven, Ant, Gradle |
| **CI/CD Tools** | Jenkins, Azure VSTS, GitLab CI |
| **Configuration Management Tools** | Ansible, Chef, Puppet, Terraform |
| **Containerization** | Docker, Docker Swarm, Kubernetes, Openshift |
| **Version Control Tools** | GIT, Bitbucket, SVN |
| **Scripting Languages** | Python, Ruby, JSON, YAML, Power Shell |
| **Virtualization Technologies** | VMWare ESX/ESXi, Windows Hyper-V, Virtual box, Vagrant |
| **Web Servers** | Apache HTTP 3.x, Apache Tomcat, Nginx |
| **Monitoring Tools** | Splunk, Nagios, Cloud Watch, ELK, App Dynamics, DataDog, SumoLogic |
| **Repository Management** | JFrog, Nexus, Artifactory |
| **Bug Tracking Tools** | Bugzilla, JIRA |
| **IDE Tools** | Eclipse IDE, Netbeans |
| **Databases** | MySQL, MS Access, NoSQL (MongoDB, DynamoDB), CosmosDB |

**PROFESSIONAL EXPERIENCE:**

**Cardinal Health, Dublin, Ohio March 2020 – Present**

**Sr. Cloud/DevOps Engineer**

* Experience in working with **Microsoft** **Azure**, **Azure Resource Management** templates, **Virtual Networks**, **Storage**, **Virtual Machines** and **Azure Active Directory**.
* Handled deployment workflow including **provisioning**, **load balancing**, **health monitoring** and was also responsible for creating and maintaining the **Resource Groups**.
* Provided support, assistance and guidance with post-migration related issues and be available as an escalation point for **Azure** related issues during project.
* Worked across the enterprise to coordinate and work with cross-functional team members for day-to-day activities.
* Created and managed Azure Virtual Machines and Virtual Machine Scale Set in **Azure portal** and setting up communication with the help of **Endpoints** and also setting up logs using **Log Analytics Workspace**.
* Involved in the implementation of **Azure Data Lake Storage** which enables analytics on the stored data and tuned for performance for data analytics.
* Implemented **Azure Disk Encryption** for **Managed disks** of the Virtual Machine and **Storage Accounts** using Azure portal and Azure CLI.
* Worked with Azure **ExpressRoute** to create private connections between Azure Datacenters and Infrastructure for on-premises and in colocation environment.
* Created Storage Pools and stripping of Disks for Azure Virtual Machines, backup, configure and restore Azure Virtual Machine using **Azure Site Recovery** (ASR).
* Involved in the implementation of **Akamai CDN** for domain names which are configured to point towards webapp URLs and creation of Edge key and CName for the domain names.
* Involved in the implementation of **SSL configuration** and **Certificate** creation and attached them to the approved domain name for **Tableau**.
* Involved in implementation of Tableau-Okta handshake using the SSL configuration files generated in the **Tableau server**.
* Handled promotion of Azure Pipelines of **Snowflake DB** tables to higher environment in **Azure VSTS** Organization.
* Involved in the implementation of **Azure Functions** integration in the deployment of **Snowflake** in Azure VSTS Pipeline.
* Involved in the implementation of Disaster recovery test for Microsoft Azure resource **Storage Accounts** using geo-replication with **Read Access – Geo Redundant Storage**(RA\_GRS).
* Worked on **Azure Site Recovery** (ASR) service in the implementation of Virtual Machine migration from one region to another and replication of data in the VMs to secondary region.
* Worked on **google cloud platform (GCP)** services like compute engine, cloud load balancing, cloud storage, cloud SQL, stack driver monitoring and cloud deployment manager.
* Dealt with setup of **GCP** Firewall rules to allow or deny traffic to and from the VM’s instances based on specified configuration and used **GCP** cloud CDN (content delivery network) to deliver content from **GCP** cache locations drastically improving user experience and latency.
* Worked on **GKE** Topology Diagram including masters, slave, RBAC, helm, kubectl, ingress controllers **GKE** Diagram including masters, slave, RBAC, helm, kubectl, ingress controllers. Created projects, VPC’s, Subnetworks, **GKE** Clusters for environments QA and prod using **Terraform**.
* Worked on **Google Dataproc** clusters and submit a job to an existing **Dataproc** cluster via a **Dataproc** API jobs and submitted HTTP or programmatic request, using the **Cloud SDK gcloud** command-line tool.
* Configured **Google Security Command Center** to have centralized posture for security and to apply CIS benchmarking.
* Involved in creating **Docker** **images** and **Docker containers** to be deployed in **Azure Kubernetes** cluster setup, configuring pods and mounting persistent volumes.
* Created **Drone.yml** files and Custom **Docker Images** to create CI pipelines on **Drone** and Used manifest files to build CD pipelines in spinnaker and also created **Istio rules** to manage traffic.
* Virtualized the servers using the **Docker** for the test environments and dev environments needs, and configuration automation using **Docker** containers.
* Automated the creation of **Azure Iaas** virtual machines using **Terraform modules** and deployed virtual machine scale sets in production environment.
* Implemented Infrastructure automation through **Terraform** and **Ansible** for auto provisioning, code deployments, software installation and configuration updates.
* Worked on **GitHub** management and merging to facilitate collaboration between development teams and configured webhooks for push events on **GitHub** to trigger a build automatically on **Azure VSTS**.
* Written **Ansible Playbooks** with **Python** SSH as the Wrapper to manage Configurations of **Azure Nodes** and Test Playbooks on Azure instances using **Python** SDK and Automated various infrastructure activities like continuous deployment, application server setup, stack monitoring using **Ansible** playbooks.
* Configuration Automation and Centralized Management with **Ansible** and **Cobbler**. Implemented **Ansible** to manage existing servers and automate the build/configuration of new servers.

**Environment:** ***GCP, Azure, Ansible, Cobbler, Docker, Akamai CDN, Terraform, Snowflake DB, Jenkins, Shell Scripts, Bash, PowerShell.***

**US Bank, Irving, TX Feb2019 – Feb 2020**

**Site Reliability Engineer**

* Created and managed Virtual Machines in Windows **Azure** and setting up communication with the help of Endpoints. Configured **Shared Access Signature (SAS)** tokens and Storage Access Policies in **Azure Cloud Infrastructure**.
* Setup **Azure Virtual Appliances (VMs)** to meet security requirements as software-based appliance functions (firewall, WAN optimization and intrusion detections) and Implemented **high availability** with **Azure Classic** and **Azure Resource Manager** deployment models.
* Designed and configured **Azure Virtual Networks (VNets), subnets, Azure network settings, DHCP address blocks, DNS settings, security policies and routing.**
* Provided consulting on **Azure cloud architecture** for training, assessments and internal projects running on MS Azure platform for high availability of services, low operational costs.
* Worked on various **Azure services** like Compute (**Web Roles, Worker Roles**), **AzureWebsites**, Caching, **SQL Azure**, NoSQL, USQLS, Storage, Network services, Data Factory, **Azure Active Directory**, API Management, Scheduling, Auto Scaling, and PowerShell Automation.
* Used **Python API** for uploading all the agent logs into **Azure blob** storage. Managed internal deployments of monitoring and alarm services for the Azure Infrastructure (OMS).
* Migrated in-house DB to **AWS RDS** and for **High Availability** clustering used read-replica data stream so that **RDS** automatically creates a primary DB instance and synchronously replicates the data to a standby instance in a different AZ.
* Wrote **Lambda** functions in **python** for AWS Lambda and invoked **python** scripts for data transformations and analytics on large data sets in **EMR clusters** and **AWS Kinesis data streams**.
* Created **Python scripts** to completely 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.
* Designed, architected cloud application deployment in **OpenStack** and worked on fixing issues related to **OpenStack** components like Nova, Neutron, Glance, Keystone command line clients.
* Created and managed Cinder volumes, instances and configured network in **OpenStack** Cloud by using both CLI and Horizon dashboard.
* Created **Terraform** scripts in AWS for services like **VPC** (Virtual Private Cloud) to automatically setup and modify settings by interfacing with control layer and responsible to setup **Elastic search cluster** using Terraform scripts to block all the **Spam and Possible attacks**.
* Replaced the use of CloudFormation with Terraform to not be fully locked into a single cloud provider.
* Written **Ansible** Playbooks with **Python** SSH as the Wrapper to Manage Configurations of **AWS** nodes and Tested Playbooks on **AWS** instances using **Python**.
* Installed, Configured and Managed **Docker** Containers, **Docker Images** for Web Servers and Applications servers such as Apache, Tomcat using Docker and integrated with **Amazon MySQL RDS database**.
* Used Docker to containerize custom web applications and deployed them on **EC2 instances** through SWARM Cluster and automated application deployment in cloud using **Docker HUB**, **Docker Swarm**, and **Vagrant**.
* Created **Kubernetes cluste**r with objects like **Pods**, Deployments, Services and ConfigMaps and created reproducible builds of the **Kubernetes applications**, managed **Kubernetes manifest files** and **Helm packages**.
* Involved in development of test environment on **Docker** containers and configuring the **Docker** containers using **Kubernetes**. Utilized Kubernetes and Docker for the runtime environment of the **CI/CD** system to build, test deploy.
* Created **Docker images** using a **Docker file**, worked on Docker container **snapshots**, removing images and managing **Docker volumes** and experienced with Docker container service.
* Deployed Kubernetes cluster in production using **Terraform** scripts and troubleshooted networking related issue.

Developed Continuous Deployment pipeline by integrating Jenkins with **Terraform**, developed **Modules** in **Terraform** to trigger the Infrastructure on the **Azure**.

* 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**.
* Collaborated with development and Support teams to setup a **Continuous Delivery environment** with the use of Docker, continuous build, and Delivery tools.
* Worked on **Ansible** as **Configuration management tool**, to automate repetitive tasks, quickly deploys critical applications, and proactively manage changes and written **Ansible playbooks** to perform **middleware binary Installation** and Configuration tasks involving JDK, Tomcat binaries installation.
* Written playbooks to deploy services on cloud, applications and to wrote modules in **Ansible** for implementing automation of continuous deployment.
* Automated the tasks with **Ansible Playbooks** by using **YAML** functions and migrating the servers with the required configuration changes and testing and deploying the machines using **Ansible** Commands.
* Developed custom **OpenShift templates** to deploy the applications and to create the OpenShift objects build, deployment configs, services, routes, and persistent volumes.
* Troubleshoot **OpenShift** containers for related issues like crash loop back off, pod restarts, out of memory issues and lack of CPU.
* Troubleshoot Jenkins pipelines using **JMeter and LoadRunner** to identify the issues and executed **load and stress tests** to identify the performance bottle necks.
* Created Log collection in **ELK** (Elasticsearch, Logstash, Kibana) installed File Beat on all nodes in cluster to send log data to Logstash. Applied filters on log data before sending to Elastic Search. In addition, configured **Splunk** for application monitoring.
* Deployed and configured **JIRA**, both hosted and local instances for issue tracking, workflow collaboration, and tool-chain automation.

**Environment:** ***AWS, Azure, Terraform, Ansible, OpenShift, ELK, Docker, Vagrant, Kubernetes, Python, JMeter, Splunk, Jenkins, AWS RDS, JIRA, GIT, PowerShell.***

**CHARTER COMMUNICATIONS, St. Louis, MO Nov 2016 – Dec 2018**

**Cloud Engineer**

* Involved in building highly available, cost effective and fault tolerant servers using **AWS services** which include using multiple **AWS EC2 instances, Auto Scaling, AWS Elastic Load Balance** and **AWS Amazon machine image (AMIs), DNS mapping** with subdomain using **Route 53**.
* 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**.
* Written **Chef Cookbooks** for various DB configurations to modularize and optimize product configuration, converting production support scripts to **Chef Recipes** and **AWS server** provisioning.
* Installed **Chef Server** and clients to pick up the integration of automated build with deployment pipeline from **Jenkins** repository and deploy in target environments.
* Built **Jenkins** jobs to create **AWS infrastructure** from **GitHub** repos containing code and administered/engineered Jenkins for managing weekly Builds and integrated Jenkins with various DevOps tools such as Nexus, SonarQube, Chef etc.
* Created **Jenkins jobs** for continuously building the projects. Scheduled jobs to build upon every check in into **subversion**. Wrote **Groovy script** in Jenkins jobs to do pre-and post-build operations.
* Involved in creating **Jenkins CI** pipelines and automate most of the pipeline build related tasks by deploying and validating the automated builds using pipeline **Groovy** and **Shell/Perl scripts**
* Used **Shell**, **Bash**, and **Python**, to supplement automation provided by **Chef** and **Terraform** for tasks such as encrypting EBS volumes backing AMIs and scheduling Lambda functions for routine AWS tasks.
* Developed UNIX shell scripts for auto deployment process and health monitoring of WebLogic servers and ran them as cron jobs.
* Configured JMS resources Queues, Topics, and JDBC connection pools and data sources for WebLogic/JBoss server instances.
* Created Log collection in **ELK** (ElasticSearch, Logstash, Kibana) installed File Beat on all nodes in cluster to send log data to Logstash. Applied filters on log data before sending to Elastic Search.
* Deployed and configured **JIRA**, for issue tracking, workflow collaboration, and tool-chain automation. Handled production incidents, queries, problems through remedy ticketing system.
* Created the file systems using **Redhat volume manager** and performing the health check on regular basis for all Linux servers.
* Worked with a complex environment on **Redhat Linux** and **Windows Servers** while ensuring that these systems adhere to organizational standards and policies.
* Used **Maven** to build java-based web applications and integrated it with **Nexus repository** and installed **Jenkins** plugin in **GIT** Repository.
* Assisted users of **Nagios** internally in designing and maintaining production-quality dashboard. Involved in installation, Configuration, and Managed **Nagios** for Resource Monitoring/Network Monitoring/Log Trace Monitoring.
* Dealt with databases like **SQL Server** and **Oracle Server** and was involved in creating stored procedures, functions, triggers and complex queries.
* Worked on branching, tagging and maintaining the version control and source code management tool **GIT** on Linux and windows platforms.

**Environment:** ***AWS, Jenkins, Redhat Linux, Maven, JIRA, Nexus, GIT, Apache Tomcat, Shell Scripts, Groovy, Chef, Terraform, ELK***

**Cyient Ltd, Hyderabad, India Mar 2015 – Nov 2016**

**Build and Release Engineer**

* Experience in setting up Jobs for Plugin Management, **Build/Deploy pipeline** setup and User management. Also involved in setting up the **backup server** and prepared disaster recovery plans for **Jenkins**.
* Integrated **GIT** and **Maven** with **Jenkins** to implement the continuous integration (CI) process.
* Configured Nightly Builds using **SCM** polling option to ensure an updated version of artifactory generated in **NEXUS**.
* Worked on building tools such as **Ant** and **Maven** to automate and enhance the overall operational environment. Automated testing builds and deployment by developing and maintaining the processes and associated scripts/tools.
* Configured **GITHUB** web hooks to Java code repositories hosted in **GITHUB** enabling automatic build triggers for a code commit with **JIRA** ticket preventing manual triggering options.
* Implemented and configured **Nagios** for continuous monitoring of applications and enabled notifications via emails and text messages.
* Performed troubleshooting and performance tuning of **firewalls**, load balancers, and virtual routers.

**Environment:** ***Jenkins, Artifactory, Linux, Maven, JIRA, Nexus, GIT, Apache Tomcat, Shell Scripts, Bash, PowerShell.***

**INTELLIGROUP, Hyderabad, India Apr 2014 – Mar 2015**

**Jr. Linux Engineer**

* Managed and administrated **UNIX servers**, including Linux operating systems by applying relative patches and packages at regular maintenance periods using **Red Hat Satellite server**, **YUM**, **RPM tools**.
* Planned and performed the upgrades to Linux **(RHEL 5x, 6x, SUSE 10, 11, CENTOS 5, 6)** operating systems and hardware maintenance like changing memory modules, replacing disk drives. Performing **kernel** and **database configuration optimization** such that it limits I/O resource utilization on disks.
* Managing **systems routine backup** by scheduling jobs like disabling and enabling cron jobs, enabling system logging, network logging of servers for maintenance, performance tuning, and testing.
* Deployment and Configuration of application server **Tomcat** deploying Java and Web Application.
* Worked with **Apache Web servers** and managed **JBoss**, **Apache Tomcat** and **Web Sphere Application servers**.

**Environment:** ***RedHat, Shell Scripts, Apache Tomcat, Virtualization, Linux, Oracle, DNS, FTP, WebSphere, SSH, Nagios.***