RAJITHA BOLLA

AWS Devops Engineer

Email id: sv.rajitha627@gmail.com

Contact no: +44 7424393490

Professional Summary

Around **4+ years** of experience in implementing, delivering, integrating and maintaining infrastructure in **Linux/Windows** Environment. Focusing on configuration, Build and Release engineering also cloud infrastructure with automation following **Agile** methodology.

- Experience in configuring various **AWS services like VPC, EC2, IAM, ELB, RDS, S3, SNS, SQS, Route 53, Lambda, Cloud Front, Cloud Formation and Cloud Watch**.
- Experience in **AWS Administration Services like EC2, S3, EBS, VPC, ELB, RDS, EMR, Dynamo DB, Auto Scaling, Security Groups, AWS IAM Services in Users, Groups, Policies, Roles, AWS Access Keys.**
- Experience in creating and Implementing **DevOps** strategies in various distribution of **LINUX and Windows** using **DevOps** tool suites like Subversion **(SVN), GIT, CVS, ANT, Maven, Jenkins, Bamboo, Ansible, Chef, Puppet, Docker, Docker Swarm, Kubernetes** and **Splunk** in traditional and cloud environment like **AWS, Google Cloud (GCP) and Azure.**
- Experience in writing **Infrastructure as a Code (IaC) in Terraform** and creating its reusable modules in **Azure and AWS** Cloud environments.
- Extensive experience with Scheduling, deploying, and managing container replicas onto a node cluster using **Kubernetes** with building Kubernetes run time environment of the **CI/CD** system to build, test and deploy in an open source platform.
- Experience in creating custom **VPC's and IAM (Roles, Users, Groups**) and attached required policies.
- Created **S3 buckets** and enabled the versioning of each object and enabled CORS on multiple buckets to access the cross-regional objects.
- Implemented **Docker** containers to create images of the applications and dynamically provision slaves to **Jenkins CI/CD** pipelines
- Experienced in deploying and configuring **Chef** Server including Bootstrapping of **Chef** Client nodes for provisioning and created (Roles, Recipes, Cookbooks and Data Bags) for configuration
- Gained Knowledge on creation of **Puppet** manifest files to install tomcat instances and to manage configuration files for multiple applications.
- Experience in **Ansible** Playbook, **YAML** for Maintaining Roles, Inventory Files and Groups Variables.
- Proficient with **Shell, Python, Ruby, Perl, Power Shell, YAML, Groovy** scripting languages.
- Gained knowledge in **Azure** Development worked on **Azure** web application, **Azure Blob Storage, App Services, Azure SQL Database, Azure Virtual Machines, Azure AD, Azure search, Azure DNS, Azure VPN Gateway, and Notification Hub.**

- Setup and Implementing Continuous Integration and Continuous Delivery (**CI & CD**) process stack using **AWS, Cloud Foundry, GITHUB/GIT, Jenkins, SonarQube, Nexus, Docker, and Ansible.**
- Converted the **SVN** repository to a local **Git** repository and shared the repository with the developers via **Bitbucket**
- Have experience on utilizing **Elasticsearch ELK Stack** and other monitoring tools like **Nagios, Datadog, Prometheus, Grafana**, **and Cloud Watch**.
- Gained knowledge with operations of cloud services **PaaS/SaaS/IaaS** in designing and automating the infrastructures and deploying in cloud platforms such as **AWS, Azure.**
- Gained knowledge with installing and operating **My SQL** and **Mongo DB** servers, as well as supporting web and application servers like **Apache Tomcat and Nginx.**
- Experience in using build automation tools like **MAVEN, ANT** for the building of deployable artifacts such as **WAR, JAR & EAR** from source code
- Creation of custom **Docker** (Container Images, Tagging and Pushing) the images and creating the Docker Containers and **Docker** consoles for managing the application lifecycle
- Very Familiar with **Linux** environments like **CentOS, RHEL (OSE), Ubuntu and Windows**.
- Learned how to use the Atlassian technologies **JIRA, Confluence, Bit bucket, and Bamboo** for continuous integration and deployment procedures, team collaboration, and defect management.
- Experience on every stage of the Software Development Life Cycle (**SDLC**), including project analysis, planning, post-production development, testing, and implementing.
- Experience in using the **Agile/Scrum** and **Waterfall** testing lifecycle Methodologies.

Technical Skills

- **AWS, Azure**
- **Ansible, Puppet, Chef, Shell scripting, Python**
- **EC2, S3, Cloud Front, API Gateway, Document DB, VPC, Direct Connect, Route53, CloudWatch,**
- **Cloud Trail, SQS, SNS, Cloud Formation, IAM**
- **Kubernetes, Docker, Terraform**
- **Splunk, Confluence, Bamboo, Nagios**
- **S3 (Simple Storage Service), AWS Cloud Watch, Grafana, Splunk**
- **GIT, GitHub, Bit bucket, Mongo DB, Apache Tomcat**
- **ANT, Maven, Jenkins, GIT, SVN, Gitlab**
- **Windows, Linux, UNIX.**
- **Agile-SCRUM, Waterfall**

Professional Experience

Company: Sonata Software **Duration: June 2021 to till date**

Client: AT & T

Role: AWS DevOps Engineer

Description: The project's main goal is to effectively manage and customize AWS EC2 instances with CloudFormation while maintaining high availability. It also entails putting object lifecycle policies into place and efficiently managing S3. Creating CloudFormation templates is also necessary in order to quickly launch VPCs and instances in the AWS cloud. In addition, it involves effectively maintaining Glacier, S3, and ELB for quality control.

Responsibilities:

- Worked on **Terraform code** to build infrastructure-as-code solutions for Application pipelines and configuration management at scale.
- Hands-on experience with Amazon Web services (**AWS**) and implemented solutions using **EC2, S3,** and **RDS** in **cloud formation templates, EBS, Elastic Load Balancer, Auto Scaling Groups, Auto scaling Launch Configuration and Auto scaling** Lifecycle Hooks.
- Provided highly durable and available data by using **S3** data store, versioning, lifecycle policies, and create **AMI's** for mission-critical production servers for backup.
- Developing Dev, Test and production environments of different applications on **AWS** cloud by provisioning **Kubernetes** clusters on **EC2** instances using **Docker**, **Bash, and Chef.**
- Created **cloud watch** alarms through **Terraform** for **AWS lambdas** and monitoring them.
- Implemented **CI/CD** processes using **Cloud Formation** and Terraform templates and Containerized the entire infrastructure using **Docker** setup in **AWS Cloud** based environment.
- Responsible for creating different versions of **Docker images**, Compose **Docker Registry** from app source code pushing them to **Nexus** repository.
- Developed build and deployment scripts using **ANT** and **Apache MAVEN** as build tools in **Jenkins** to move from one environment to other environments.
- Configured **Grafana** and **Prometheus** to monitor databases, and use **Grafana** to create personalized dashboards.
- Extensive experience in using Version control systems includes **Subversion (SVN), GIT, and Clear Case**.
- Proficient with **Jenkins** and **Bamboo** for continuous integration and for End-to- End automation for application build and deployments.
- Worked on various **Azure services (PAAS & IAAS) like Function Apps, Logic Apps, Key Vault, Compute (Web Roles, Worker Roles), Azure Websites, Azure Active Directory, API Management, Signal R, Redis Cache, AKS, SQL Azure, Storage, Network services, Management, Scheduling, Auto Scaling, and PowerShell Automation.**
- Monitored and alerted **Nagios** in both passive and active modes. Time-based system/container metrics can be obtained via **CloudWatch**.
- Performed all necessary day-to-day **Git hub** support for different projects like Check-in, Checkouts, import, export, branching, tagging, and conflict resolution.

- Worked with **Ansible playbooks** for virtual and physical instance provisioning, configuration management, patching and software deployment on AWS environments through automated tools, Ansible / custom pipeline
- Authored **Chef Cookbooks** for setting up **Nginx** and **Tomcat** as well as for configuring load balancers and failover.
- Worked on **shell/python scripts** and implemented an auto deployment procedure while spending less time with **Python, Bash/Shell, Ruby, Perl and PowerShell**.
- Used **JIRA** tool to track all the defects and changes related to Build and Release team
- Developed **Linux, UNIX, Perl** and **Shell Scripts** for manual deployment of the code to various environments.

Environment: AWS, Azure. Devops, Linux, UNIX, Perl, Shell, Chef, Nginx, Tomcat, Bamboo, Python, SVN, GIT, Jira, s3, Grafana, Prometheus, CI/CD, EC2, Nexus, Nagios, Ant, Apache Maven, GitHub, PowerShell, Cloud watch.

Company: DXC **Duration: May 2019 to June 2021**
Client: Siemens
Role: DevOps Engineer

Description: In order to ensure that our applications were optimized for cloud deployment and that the cloud infrastructure is dependable, scalable, and secure, I worked closely with development and operations teams throughout the design, implementation, and management of cloud-based infrastructure and applications.

Responsibilities:

- Worked on **DevOps** methodologies and cloud infrastructure automation tools like **AWS** by using **Terraform**.
- Worked closely with the development teams to build the continuous integration and continuous Delivery Pipelines using **Git, Jenkins and Maven**.
- Involved in development of test environments on Docker containers and configuring the **Docker** containers using **Kubernetes**.
- Created **Docker images**, and handled multiple images in various Docker **hubs**. Created **Docker image** of the application and used **AWS ECS** container management service.
- Used **Elastic Stack ELK (Elastic search, Log stash, Kibana)** to validate the operational status of the resources by examining items such as metrics, Synthetic transactions, and log files.
- Automated **Ansible** to reduce manual intervention and speed up deployment procedures.
- Extensively worked on **Jenkins** by installing, configuring, and maintaining the purpose of **CI** and End-to- End automation for all build and deployments implementing **CI/CD** for the database using **Jenkins**.
- Written in scripting languages such as **PowerShell, Bash, and Python** as a part of writing scripts for the automation tasks.
- **Splunk** was utilized to set up infrastructure and service monitoring, while **AWS Cloud Watch, Nagios**, and other performance monitoring and analytics tools were used.
- Extensive experience in Implementing Configuration Management tools like **Puppet and Chef.**
- Installed and set up **Nexus** repository manager for internal teams to share artifacts. To improve the build process, proxy **Nexus** repositories were made.
- Implemented a **CI/CD** pipeline with **Docker, Jenkins** and **GitHub** by virtualizing the servers using **Docker** for the Dev and Test environments by achieving needs through configuring automation using Containerization.

- By using **Jenkins** and **Python libraries**, we automated the process of creating **Bitbucket** Server repositories.
- Experience in Installation of security features to **MySQL, Mongo DB** and **Cassandra** database on **Red Hat Linux** Servers.
- Involved in setting up **JIRA** as defect tracking system and configured various workflows, customizations, and plugins for bug tracker.

Environment: AWS, Devops, Nexus, Python, My SQL, MongoDB, Cassandra, Bitbucket, Splunk, NoSQL, Git, Maven, CI/CD, GitHub, Nagios, ELK, ECS, PowerShell, Bash, Puppet Chef, Jira, Linux.