|  |  |
| --- | --- |
|  | **Sai Teja**  DevOps AWS Engineer saitejaavadootha98@gmail.com  **202 240 2324** |

**LinkedIn Profile here:**

# SYNOPSIS

* 8+ Years of experience in the **IT** industry comprising **Software Configuration, Linux Administration, Build** and **Release Management, Deployment Automation**, **Amazon Web Services,** and **Cloud implementation**.
* Strong Experience in setting up the enterprise infrastructure on Amazon Web Services (**AWS**) like **EC2** instance, **ELB, EBS, S3 Bucket, Security Groups, Auto Scaling, LAMBDA, KINESIS,,GLUE IAM, Cloud formation, Cloud Front & VPC services.**
* Experience in technologies and platforms including **JAVA**, **GIT**, **Unix/Linux**, windows server, **VMWare**, **Docker**, **Ansible**.
* Provisioning Web Servers in **AWS** using different services **EC2, VPC, ELB, Auto Scaling, Route53** using Cloud formation templates & AMI's using Infrastructure as a Service (**IAAS**).
* Experience in developing terraform scripts for the AWS Service to deploy on Prod Environment.
* Proficient with container systems like **Docker** and container orchestration like EC2 Container Service, **Kubernetes**, worked with Terraform.
* Managed Docker orchestration and Docker containerization using Kubernetes.
* Used **Kubernetes** to orchestrate the deployment, scaling, and management of Docker Containers.
* Implemented a Continuous Delivery framework using **Chef,Jenkins**, and **Maven** in a **Linux** environment on **AWS** Public cloud.
* Performed automation tasks on various **Docker** components like **Docker Hub, Docker Engine, Docker Machine, Compose,** and **Docker Registry.**
* **Docker** applications by creating **Docker images** from **Docker file**
* Extensively worked on **GIT, Jenkins** for continuous integration and End to End automation for all build and deployments.
* Experience in administrating, deploying, and managing **RedHat, Ubuntu** servers.
* Experience in **AWS**, **Kubernetes, Docker** andwriting scripts in **Shell** Scripts for managing various enterprise applications.
* Hands-on Experience in branching, tagging, and maintaining the version across the environments using **SCM** tools like **GitHub** on Windows platforms.
* Hands-on Experience in some monitoring tools like **Splunk,Grafana,Cloudera.**
* Automated setting up server infrastructure for the **DevOps** services, using **Ansible**, **shell,** and **python scripts**.
* Worked with different Bug tracking tools like **JIRA, ServiceNow.**

**EDUCATION:**

Master of Computer Applications in 2017 : University of Maryland Baltimore County

Bachelor of Computer Science in 2013 from : Guru Nanak Institutions Technical Campus, Hyderabad, Telangana

**Technical Skills*:***

|  |  |
| --- | --- |
| **Version Control** | GIT, Bitbucket |
| **Languages** | C, Java, XML, Python, Shell Scripting |
| **Scripting** | Shell, Python, Terraform |
| **Operating System** | Microsoft Windows,Ubuntu |
| **Virtualization** | VMware. |
| **Container Orchestration** | Docker, Kubernetes, EKS, ECS. |
| **Build Tools** | Maven |
| **CI Tools** | Jenkins. |
| **Configuration Management** | Ansible. |
| **Bug Tracking tools** | Jira |
| **Databases** | SQL, MySQL, NoSQL. |
| **Servers** | Tomcat |
| **Protocols/ Services** | Http, Https, SSL, DHCP, DNS, SSH, SFTP, TCP/IP, FTP/SFTP, SMTP. |
| **Cloud Tools** | AWS (EC2, EBS, ELB, Auto Scaling, S3, VPC, IAM, CloudFormation, CloudWatch, SNS, RDS, KINESIS,GLUE,LAMBDA, EKS,ECS,EFS) |
| **Others** | Splunk, Confluence, Jira. |

**Professional Experience**

**Client: AT&T Aug 2021-Present**

**Role: Sr.DevOps Engineer**

**Responsibilities:**

* Created **AWS** Launch configurations based on customized **AMI** and use this launch configuration to configure auto-scaling groups and Implemented **AWS** solutions using **EC2, S3, RDS, DynamoDB, Route53, EBS, Elastic Load Balancer, Auto Scaling groups.**
* Build servers using AWS, import volumes, launching **EC2, RDS**, creating **Security groups, Auto-Scaling, Loadbalancers** (**ELBs**) in the defined virtual private connection.
* Understanding secure-cloud configuration (**CloudTrail, AWS Configuration**), Cloud-security Technologies (**VPC, Security Groups**, etc.), and Cloud-permission systems (**IAM**).
* Used Jenkins pipelines to drive all micro services builds out to the Docker registry and then deployed to **Kubernetes**, Created Pods and managed using **Kubernetes**.
* Building/Maintaining **Docker** container clusters managed by **Kubernetes** Linux, Bash, GIT, Docker on AWS, ▪Utilized Kubernetes and **Docker** for the runtime environment of the CI/CD system to build, test deploy.
* Involved in development of test environment on Docker containers and configuring the **Docker** containers using **Kubernetes**. ▪Created alarms and trigger points in **Cloud Watch** based on thresholds and monitored logs via metric filters.
* Setup Log Analysis AWS Logs to **Elastic Search** and **Kibana** and Manage Searches, Dashboards, custom mapping, and Automation of data.
* DevOps Engineering in various technology stack including but not limited to **AWS, Azure, Docker, Containers, Python, Kubernetes, NodeJS, Angular, Terraform, Ansible, Jenkins, GIT, JIRA**, Monitoring in Cloud Environment.
* Converted **Pom.xml** into the build. **Gradle** to build the applications using **Gradle**.
* Managed the artifacts generated by Maven and **Gradle** in the Nexus repository.
* Deployed a Django application connected to Azure SQL using Docker and Azure App Service
* Involved in the migration of **CI** tool stack from **SVN** to **GIT**, Maven to **Gradle**, **Build a force** to **Jenkins**.
* Managed the artifacts generated by Maven and **Gradle** in the **Nexus** repository.
* Designing and implementing **CI** (Continuous Integration) system: configuring **Jenkins Servers, Jenkins nodes**, **TFS** creating required scripts (**Perl & Python**), and creating/configuring VMs (**Windows/Linux**).
* Monitoring apps servers using **ELK** - **Elasticsearch, Logstash,** and **Kibana** for debugging and logging.
* Experienced in monitoring servers using **Nagios, Splunk, Cloud watch,** and using **ELK**.
* Hands on Experience in Deploying **Splunk** systems and Monitor Splunk internal logs from the monitoring Console (MC) to identify and troubleshoot existing or potential issues
* Worked with **Terraform** for automating **VPCs, ELBs**, security groups, SQS queues, **S3** buckets, and continuing to replace the rest of our Infrastructure.
* Installed, configured, and administered CI tool **Jenkins** for automated builds.
* Analyzed and resolved conflicts related to the merging of source code for **SVN** and **Git**.
* Worked on creating and documenting POC for helping migrate the current Application to microservice architecture. The architecture included **Docker** as container technology with **Kubernetes** and worked on with **REST API**.
* Created **Docker** images using a **Dockerfile**, worked on **Docker** container snapshots, removing pictures, and managing **Docker** volumes.
* Virtualized the servers using the **Docker** for the test environments and dev-environments needs. And configuration automation using **Docker** containers.
* Created multiple **Python, Perl,** and **UNIX** shell Scripts for various application-level tasks.
* Used **Nagios** as a monitoring tool to identify and resolve infrastructure problems before they affect critical processes and worked on **Nagios** Event handlers in case of an automatic restart of failed applications and services.

**Environment**:Chef, Vagrant, AWS EC2, AWS S3, AWS Route53, AWS VPC,, Jenkins, Ruby, Shell, Apache Mesos, Marathon, Chronos, AWS API Gateway, Cloud formation, Docker, Kubernetes, Splunk

**Client: Comcast Jan 2020 - Jul 2021**

**Role: DevOps Engineer**

**Responsibilities:**

* Responsible for the continuous build and deployment of Java-based code using the **Jenkins** server.
* Extensively worked on Installation and configuration of **RHEL**5/6/7.x using Kickstart.
* Develop and maintain custom internal Bash &**Python** scripts to automate the management of **AWS** and cloud environments.
* Experience with Configuration Management - **Ansible**, which includes creating, modifying Playbooks for various applications.
* Worked on **Shell** and Python scripting to automate the tasks and passed Bash scripts to the Jenkins job for server **provisioning**.
* Installed **Nagios** on global servers, Managing Monitoring solution using Nagios Environment.
* Experience in **Monitoring** the system and **network stability** and performance, and tune performance as necessary.
* Day to day management/monitoring/troubleshooting of multi-tier applications using **Ansible** and Nagios in Dev, QA, Prod **environments**.
* Utilized **Kubernetes** for the runtime environment of the CI/CD system to build, test deploy.
* **Kubernetes** is being used to orchestrate the deployment, scaling and management of Docker Containers.
* Implemented docker-maven-plugin in maven pom to build **docker** images for all microservices and later used **Docker** file to build the docker images from the java jar files.
* Building/Maintaining **Docker** container clusters managed by **Kubernetes** Linux, Bash, GIT, Docker, on GCP (Google Cloud Platform). Utilized **Kubernetes** and Docker for the runtime environment of the CI / CD system to build, test deploy.
* Programmed several Cron jobs to **automate** data backups and server maintenance using Jenkins.
* Configured RHEL6 environments for the QA team and the development team.
* Developed **AWS** Cloud Formation templates to create custom sized VPC, subnets, NAT, EC2 instances, ELB, Security Groups.
* Developed and implemented Software **Release Management** strategies for various applications according to the agile process.
* Created scripts in **Python** which integrated with Amazon API to control instance operations.
* Installing, configuring, and **administering** the Jenkins CI tool on Linux machines.
* Experience in automated deployment of **EC2** instances in data centers and availability zones.
* Install/configure/maintain the Linux servers, NFS.

**Environment**:Chef, Vagrant, AWS EC2, AWS S3, AWS Route53, AWS VPC,, Jenkins, Ruby, Shell, Apache Mesos, Marathon, Chronos, AWS API Gateway, Cloud formation, Docker, Kubernetes, Splunk

**Client: Conoco Phillips Jan 2018 - Dec2019**

**Role: DevOps /AWS Engineer**

**Responsibilities:**

* Designed and implemented scalable, secure cloud architecture based on Amazon Web Services. Leveraged AWS cloud services such as EC2; auto-scaling; and VPC (Virtual Private Cloud) to build secure, highly scalable and flexible systems that handled expected and unexpected load bursts and can quickly evolve during development iterations. designing and deploying AWS Solutions using EC2, S3, EBS, Elastic Load balancer (ELB), auto-scaling groups and OpsWorks.
* Worked at optimizing volumes and EC2 instances and created multiple VPC instances and on IAM to create new accounts, roles and groups and Experience in creating alarms and notifications for EC2 instances using Cloud Watch.
* Experience involving configuring S3 versioning and lifecycle policies and backup files and archive files in glacier and Creating Lambda function to automate snapshot back up on AWS and set up the scheduled backup.
* Used the AWS-CLI to suspend an AWS Lambda function processing an Amazon Kinesis stream, then to resume it again and automated the cloud Utilizing Puppet for configuration management of hosted Instances within AWS.
* Wrote Ansible Playbooks with Python SSH as the Wrapper to Manage Configurations of AWS Nodes and Test Playbooks on AW Sinstances using Python.
* Utilized Oracle and MySQL DBMS, Amazon Elastic Cloud, Amazon’s Elastic Map Reduce, DataStaxSolr and DataStax Cassandra.
* Extensive worked for Oracle database and Cassandra, created and worked with various database objects like Packages, Stored Procedures, Functions, Triggers, Views, Materialized Views, DB links, external tables, Materialized Views tec.
* Expert increasing, maintaining, troubleshooting and monitoring the AWS, Azure Cloud Services.
* Created recipes in cookbooks to automate system operations and Created multi AZVPC instances.
* Used the MCollective, Facter, Hiera, Puppet DB to amend the automation pipeline build around Puppet.
* Designed Kubernetes Architecture and Google Container Engine (GKE).
* Container management using Docker by writing Docker files and set up the automated build on Docker HUB and installed and configured Kubernetes
* Created Docker Containers, Docker file, Docker Images, Docker Network, Docker Orchestration, Docker Swarm, Containers Clusters, Google Container Registry, Multi-Containers, Replication Controllers, Services, Pods, Endpoints, Rollbacks.
* Created web services, WSDL and web methods with Annotation in hibernate, used the spring container for data source and to load the Hibernate specific classes.
* ConfiguredRaid-0, Raid-1, Raid-5 for data redundancy using Veritas Volume Manager and SVM on multi-OS platform.
* Built a new CI pipeline, testing and deployment automation with Docker, Jenkins, and Chef.
* Built the source code using GIT and Jenkins and managed GitHub Projects and troubleshooted Git related problems.
* Monitored Log Analytics in Azure Security Center for Trouble shooting the Compute, Storage,Networking & Applications.
* Used Jenkins for Continuous Integration and deployment into Tomcat Application Server and used Jenkins AWS Code Deploy plug-in to deploy to AWS.
* Extensively used MAVEN and ANT as a Build Tool for the building of deployable artifacts (JAR, WAR & EAR) from source code.
* Involved in authoring pom.xml files, performing releases with the Maven release plugin, and managing artifacts in NEXUS repository.
* Defined and Developed plan for Problem & Incident management Process based on ITIL. Networking protocols such as TCP/IP, Telnet, FTP, NDM, SSH, rlogin.
* Installed, upgraded and configured RedHat Linux 5.x and 6.x using Kickstart installation and SUSE Linux Servers manually as well as using System Imager and updating using YUM and worked on upgrading and configuring SGI Servers.

**Environment**:AWS, Docker, Jenkins, JIRA, VPC, NAT, IAM, S3, AWS, Cloud Watch, GIT, Splunk, Web Servers, ELB, GKE, Cloud Front, Apache, Nginx, Groovy, Shell Scripts, Lambda, Cloud Watch, Cloud Formation, Cloud Trail, Cloud Front, Chef, Python, MySQL, Cassandra, Amazon RDS, DynamoDB, Elasticsearch, Logstash, Kibana, Kubernetes, VMware.

**Client: Celoris Tech Jan 2014 - Oct 2016**

**Role: DevOps & Cloud Engineer**

**Responsibilities:**

* Designed Created the automated build and deployment process for application, re-engineering setup for better user experience, and leading up to building a continuous integration system for all our products.
* Responsible for Design of different Release Environments for new projects.
* Implemented new projects builds framework using Jenkins & maven as build framework tools.
* Implemented a Continuous Integration and Continuous Deployment framework using Jenkins, Maven & Artifactory in Linux environment
* Managed deployment automation using Puppet, MCollective, Hiera, Custom Puppet modules, Ruby.
* Created puppet manifests and modules to automate system operations.
* Integrated Maven/Nexus, Jenkins, Urban Code Deploy with Patterns/Release, Git, Confluence, Jira and Cloud Foundry.
* Implemented AWS solutions using E2C, S3, RDS, EBS, Elastic Load Balancer, Auto scaling groups and optimized volumes, EC2 instances.
* Used IAM to create new accounts, roles and groups and Configured S3 versioning and lifecycle policies to backup files and archive files in Glacier
* Configured Elastic Load Balancers with EC2Autoscalling groups and Work with different team members for automation of Release components.
* Worked on installation and configurations of Puppet agent and Puppet Master and Expert in adding nodes to node groups, defining classes and maintain puppet infrastructure and worked on open source modules for puppet from puppet forge.
* Create a Virtual Network on Windows Azure to connect all the servers.
* Created customized modules from the existing modules based on client requirements.
* Scripting in multiple languages on UNIX, LINUX and Windows – Batch, Shell script etc.
* Troubleshoot the build issue during the Jenkins build process.
* Resolved system issues and inconsistencies in coordination with quality assurance and engineering teams.

**Environment**:GIT, SVN, JENKINS, PUPPET, MAVEN, AWS, EC2, WEBSPHERE, PUTTY, Java, ServiceNow.