



## Jayasree Keesari

Phone No: +1 (980)-216-4790

Email ID: jayasree3815@gmail.com

Education: Bachelor's in Computer Science (MVSREC, HYD-INDIA 2009-2013)

### Sr. Cloud DevOps Engineer #AWS #Azure

#### Summary:

- A professional DevOps Engineer with 9+ years of IT experience, in which 5+ years as a **Cloud DevOps** and **SRE** as well as a Build and Release engineer, on **Red hat Linux, CentOS, Ubuntu, and Windows** environments, Cloud computing such as **Azure DevOps** via Continuous Integration (**CI**) and Continuous Deployment (**CD**) tools such as Jenkins, Azure DevOps, GitHub Actions and AWS Code Pipeline and 4+ years of experience as Linux Administrator.
- Experience in Agile Scrum project methodology, working in cross-functional teams, and following Agile principles.
- Participation in Sprint planning, daily stand-ups, and retrospective meetings.
- Configured Continuous Integration from source control, setting up build definition within Visual Studio TeamServices (**VSTS**), and configuring continuous delivery to automate the deployment of ASP.NET MVC applications to Azure web apps, as well as regularly managing Azure Active Directory, Office 365, and applying upgrades.
- Configured and managed data quality tools and frameworks in **Azure DevOps** such as **Azure Data Factory, AzureDataLake Analytics, or AzureDatabricks** to execute data validation processes and compliant cloud adoption.
- Strong experience working on **AWS** services like **EC2, S3, IAM, EBS, Route 53, Cloud Front, Cloud Trail, Cloud Formation, RDS, DynamoDB, Elastic Bean Stalk, SNS, SQS VPC** and accessed data to cheaper storage using **AWS S3 Glacier**. Worked on Cloud automation using **AWS CloudFormation** templates.
- Experience in integrating Terraform with Ansible, and Packer to create and Version the **AWS** Infrastructure, designing, automating, implementing, and sustaining Amazon machine images (**AMI**) across the **AWS Cloud environment**.
- Good at scripting languages like **Python, Perl, and Bash** and configuration management tools Chef, Puppet, Ansible CF Engine, and Web Services like **AWS**.
- Experience with **Linux/UNIX** environment Sand scripting for Build and release automation. Experience in Branching, Merging, Tagging, and maintaining the versions across the environments using **SCM** tools like Subversion (**SVN**), and GIT (**GitHub, GitLab**).
- Experience in Continuous Integration tools like **Jenkins**, and TeamCity.
- Experience with and thorough understanding of automated implementation/deployment of a Cloud-Based Infrastructure (Web apps, Application Service Environments, firewalls, load balancers, storage, monitoring, security, AD, DNS).
- Expertise in Atlassian Tools Confluence, **Bamboo**, and **Jira**.

- Experience working with **Kubernetes**, a popular container orchestration platform, and using Gloo and Envoy to manage traffic in **Kubernetes** clusters.
- Strong Knowledge of **AWS IAM** policies and SCP Organizations, with experience in creating and managing IAM policies for **AWS** services and resources.
- Experience in creating and updating Puppet Master/Chef/Ansible etc., segregating the modules, cookbooks, recipes, writing manifests, and pushing them to all the puppet/chef agents.
- Proficient in creating and updating Ansible roles to ensure compatibility with the latest versions of critical products, including **Apache Kafka, Elasticsearch, Logstash, and Kafka**.
- Created alarms and trigger points in **Cloud Watch** based on thresholds and monitored the server's performance, CPU Utilization, and disk usage.
- Managed and orchestrated containerized applications on **OpenShift**, ensuring seamless deployment and scaling.
- Implemented a Continuous Delivery framework using Jenkins, Puppet, Maven & Nexus in a Linux environment. Deployed application updates using Jenkins. Installed, configured, and managed Jenkins and triggered the SIT environment build of client remotely through Jenkins.
- Implemented cost-saving strategies, such as **AWS** Reserved Instances, **EC2** Spot Instances, and resource tagging.
- Participated in configuring and monitoring distributed and multiple platform servers using Puppet. Used Puppet Server and workstation to manage and configure nodes.
- Deployed Puppet, Puppet dashboard for configuration management to existing infrastructure. Implemented continuous integration webhooks and workflows around **Jenkins** to automate the dev test deploy workflow around the Puppet codebase.
- Proficient in setting up, configuring, and managing an on-premises service mesh solution, with a preference for Istio, to enhance **Kubernetes microservices** communication, security, and observability.
- Experienced with Google Cloud Platform (**GCP**) services like Compute Engine, Cloud Functions, Cloud DNS, Cloud Storage, stack driver monitoring and cloud deployment manager, and **SaaS, PaaS, and IaaS** concepts of Cloud computing architecture and implementation using **GCP**.
- Experience with container-based deployments using **Docker**, working with Docker images, **Docker hub**, and Docker registries, installation and configuring **Kubernetes**, and clustering them.
- Used **Kubernetes** to orchestrate the deployment, scaling, and management of Docker Containers. To achieve the Continuous Delivery goal in a highly scalable environment, I used **Docker** coupled with the load-balancing tool **Nginx**.
- Setup puppet master, and client and wrote scripts to deploy applications on Dev, QA, and production environments.
- Created continuous integration system using Ant, Jenkins, and Puppet full automation, Continuous Integration, faster and flawless deployments.
- Experience in Installing, maintaining, administering, tuning, and troubleshooting WebSphere Application Server.
- Strong hands-on experience with scripting languages like **Python, PowerShell, and JavaScript**.
- Experience working on Docker components like **Docker Engine, Hub, Machine, Compose, and Docker Registry**.
- Creating Amazon **EC2** instances using command line calls troubleshooting the most common problems with instances and monitoring the health of Amazon **EC2** instances and other **AWS** services.
- Installed and administered **GIT** source code tool and ensured the reliability of the application as well as designed the branching strategies for **GIT**.
- Converted and automated builds using Maven and Ant. Scheduled automated nightly builds using **Jenkins**.
- Experience working with **AWS** Cloud computing and launching Instances and automation of administration roles.

## Technical Skills:

<b>AWS Services</b>	RDS, EC2, VPC, IAM, Cloud Formation, EBS, S3, ELB, Auto Scaling, Cloud Trail, SQS, SNS, SWF, Cloud Watch, AWS EKS, ECR
<b>CI/CD</b>	Jenkins, Azure Pipelines, Gitlab, VSTS, Bamboo
<b>Web Servers</b>	Nginx, Apache Httpd, WebLogic, Apache Tomcat, WebSphere
<b>Containerization</b>	Docker, Kubernetes, Helm and OpenShift
<b>Code Scanning</b>	SonarQube, Jfrog X ray, ECR Inspector
<b>Databases</b>	RDS, Cosmos DB, My SQL DB, Redshift, Mongo DB, Cassandra, SQL Server
<b>Logging</b>	Cloud Watch, Cloud Trail, Azure App Insights, Azure Monitor
<b>Monitoring Tools</b>	Nagios, Splunk, Kibana, Prometheus& Grafana, Dynatrace, New Relic
<b>Languages</b>	Python, Shell scripting.
<b>Cloud Platforms</b>	Microsoft Azure, Aws Cloud, GCP
<b>Azure Services</b>	App Services, Key vault, function app, Azure Active Directory (Azure AD), Service Bus, Azure Container Registry (ACR) and AKS, Azure SQL, Azure Cosmos DB.
<b>Version Control Tools</b>	GIT, Bit Bucket, GitHub, GitLab.

## Professional Experience

**Client: Freedom Mortgage, New Jersey**

**July 2021 - Present**

**Role: Sr. Cloud/DevOps 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 handle expected and unexpected load bursts and can quickly evolve during development iterations.
- Migrated Workloads (Applications, DB) from On-Prem to **AWS Public Cloud**.
- Migrated Containers from **OpenShift** which is running in Private Cloud to **EKS**.
- Proficient in using **ECS CLI or AWS Management Console** to interact with **ECS** clusters and tasks.
- Worked in provisioning of **AWS** services like **EC2, S3 Glacier, ELB, RDS, Redshift, IAM, Route 53, VPC, CloudFront, Cloud Watch, Cloud Trail, Cloud Formation, and Security Groups**.
- Worked with **AWS Terraform** in **AWS** Virtual Private Cloud to automatically set up and modify settings by interfacing with the control layer and was responsible for setup an Elastic search cluster using Terraform scripts to block Spam and Phishing attacks.
- Used **Terraform** to set up the **AWS** infrastructures such as launching the instances, **S3 buckets, VPC, and Subnets**.
- Used **AWS Beanstalk** for deploying and scaling web applications and services developed with **Java, .net, PHP, Node.js, Python**, and Docker on familiar servers such as **Apache**.

- Worked on Deployment Automation of all Microservices to pull images from the **Private Docker registry** and deploy to the **Kubernetes** Cluster. Integrated **Docker Container** Orchestration framework using **Kubernetes** by creating Pods, Configuration Maps, and Deployments.
- Integrated **Terraform** code and **Helm** charts with version control systems Git to manage changes and track infrastructure revisions.
- Configured (SQS) and (SNS) for seamless message queuing and notifications within applications.
- Designed and implemented **Kafka** scaling strategies using **AWS** Auto Scaling to accommodate fluctuating workloads while managing costs effectively.
- Wrote **Docker files** for **microservice** applications and had a good understanding of volumes and Networking for containers.
- Configuring '**nginx**' for proxy RESTful API calls to micro-services in Docker containers.
- Implemented scheduled downtime for non-prod servers for optimizing AWS pricing.
- Optimized **DynamoDB** for high-performance and cost-efficiency
- Worked in writing **Ansible** playbooks using Yaml scripting for **Ansible** and deploying applications using **Ansible**. Created and used **Ansible** templates to create customized configuration files for hosts and used **Ansible Vault** in playbooks to protect sensitive data.
- Configured **Ansible** along with Terraform on the Jenkins pipeline to deploy certain application-level dependencies on the Worker Nodes of the EKS Cluster
- Worked on **AWS IAM** service by creating users & groups, defining roles and policies, and Identity providers.
- Implemented Amazon **EC2** by setting up instances, virtual private cloud (**VPCs**), and security groups and by automating the process of launching and stopping/terminating **EC2** instances on **AWS** with **BOTO**.
- Written Cloud Formation Templates in **JSON** and **YAML** for automating the installation of services, packages, and deploying applications.
- Worked with **ELK** to debug production issues and log the **AWS S3** storage that stores the site's traffic activity.
- Implemented 'Serverless' architecture using **API Gateway**, **Lambda**, and **DynamoDB** and deployed **AWS Lambda** code from Amazon **S3** buckets.
- Created a **Lambda** Deployment function and configured it to receive events from your **S3** bucket.
- Used **Kafka** to ingest real-time data from various sources including **IoT Devices** and User Interactions.
- Implemented **Argo CD**, a declarative continuous delivery tool, for managing application deployments on AWS.
- Successfully implemented **Istio** as a service mesh solution within an **AWS** environment feature, such as load balancing, traffic shifting, and canary deployments, to optimize microservices communication and reliability.
- Created **IAM** role and fine-tuned permissions to ensure only authorized services and instances could interact with **Kafka** Cluster.
- Created and customized **Helm charts** to define the deployment of microservices within **Kubernetes** clusters.
- Developed **CI/CD** system with Jenkins on **Kubernetes** container environment, utilizing **Kubernetes** and **Docker** for the **CI/CD** system to build, test, deploy, and configure **Kubernetes (EKS)** to deploy scale, load balance, scale, and manage Docker containers with multiple names spaced versions.
- Developed installer scripts using Ant, and Python for various products to be hosted on Application Servers.
- Developed build and deployment processes for Pre-production environments. Advised open-source Kubernetes and **Docker** projects and design discussions in the Cloud Native Computing foundation.
- Used Amazon **RDS** Multi-AZ for automatic failover and high availability at the database tier for **MySQL** workloads.
- Creating an **AWS RDS MySQL DB** cluster and connecting to the database through an **Amazon RDS MySQL DB** Instance using the Amazon **RDS** console.

- Creating **snapshots** and Amazon machine images (**AMIs**) of the instances for backup and creating clone instances.
- Utilized **OpenShift's** integrated monitoring and logging tools, such as **Prometheus and Grafana, Dynatrace** to monitor application health and performance.
- Managed and monitored the server and network infrastructure using **NAGIOS, Datadog, and Kibana**.

**Environment:** AWS, Jenkins, GitHub, Jfrog, Aurora DB, AWS ECS, Apache Tomcat, MySQL, JIRA, Prometheus, Ansible, Docker, EKS, Kubernetes, Terraform, Maven, SonarQube, API Gateway, Kafka, SNS, SQS, Packer, Python, Shell, WebLogic, DynamoDB, Windows, ELK, Git, Linux/Unix, Service mesh, Grafana, Dynatrace, CloudWatch, Java, Angular.

**Client:** Department of Maryland Child Support, Lithium Heights, Maryland

**Sep 2019 – Jun 2021**

**Role:** Azure DevOps Engineer

#### **Responsibilities:**

- Implemented **Azure** infrastructure solutions, including **virtual networks, subnets, and security groups**, ensuring a secure and scalable environment. Improved security and scalability of **Azure infrastructure**, enabling deployment and management of applications with minimal downtime.
- Created **CICD** (VSTS) pipelines in **Azure DevOps** environments by providing their dependencies and tasks. Also, have experience in implementing and managing continuous delivery systems and methodologies in Azure.
- Configured servers to host **TFS** instance to set up and manage **CI** using Team Foundation Build service.
- Implemented **CICD** pipeline for **AKS** using **Helmcharts** in **Azure DevOps**.
- Implemented **CICD** pipeline for Webapps to deploy on app services using Azure DevOps Pipeline.
- Extensive experience in buildouts **using Azure DevOps for CI/CD** (Continuous Integration/Continuous Deployment) pipelines.
- Proficient in configuring and managing build agents and build definitions in **Azure DevOps** to automate the build process.
- Proficient in **cluster** creation, including provisioning **virtual machines**, networking components, and storage resources.
- Initiated automation of app settings configuration on Webapps using JSON files in **Azure DevOps** successfully.
- Created a **CICD** pipeline to deploy code in different environments (**Dev, Stage, QA, and Prod**) on a sequential basis.
- Scheduled CRON jobs using **CICD** on **VSTS** build servers to clean up the disk space and increase efficiency.
- Implemented unit testing and code coverage in **Azure DevOps** to validate the code efficiency.
- **HashiCorp Vault** was implemented for secret management, encryption, and secure data storage.
- Capable of delivering and managing containerized apps on **OpenShift** using its **Azure Kubernetes-based orchestration**.
- Developed **Ansible** playbooks for deploying service as pods and used **AKS** for orchestrating the pods.
- Written **Azure PowerShell** scripts to automate several routines like restarting **Azure app services**, backing up and restoring VM, setting up **RBAC** roles, and conditional access policies on Azure.
- Written **PowerShell** scripts to automate file share creation, SQL cluster creation, AD join process, and disk space increase process.
- Created ARM templates to provision infrastructure in Azure using PowerShell.

- Experience in implementing **AEM** for responsive web design, ensuring content is optimized for various devices and screen sizes.
- Created Python scripts to run **Ansible** templates from **Ansible** tower to update environment variables in the instances.
- Worked with **Terraform** scripts to automate the **Azure IAAS virtual** machines using Terraform modules and deployed virtual machine scale sets in a production environment.
- Developed and maintained Continuous Integration (**CI**) using tools in **Azure DevOps** (VSTS) spanning multiple environments, enabling teams to safely deploy code in Azure Kubernetes Services (**AKS**) using YAML scripts and HELM charts.
- Managing the **Azure Kubernetes Services (AKS)** policies, providing access to different Azure resources, and developing and improving the workflows that govern access.
- Monitoring and troubleshooting Kubernetes clusters using **Prometheus, and Grafana**.
- Developed SQL scripts to create users with specific DB roles, indexes for databases, and resolve application-related duplicate data entries.
- End-to-end Implementing **Azure Load Balancer** and Application Gateway for high availability.
- Responsible for patching, software upgrades, and maintenance of **Azure** components.
- Developing **Terraform** scripts for deploying Infrastructure in **Azure** cloud.

**Environment:** Azure, Java/J2EE, GitHub, Jenkins, Ansible, Maven, VSTS, Shell (bash), JSON schema, Django, F5 LTM, Apache Tomcat, WebLogic, Terraform, Azure app services, Azure Application Insights, Ariba, AEM, Azure application gateway, Azure DNS, Azure security groups, Azure functions, Active Directory services, Cockroach DB, bitbucket, private container registry, OpenShift, Splunk, GCP, Azure, PowerShell, Red hat LINUX, CentOS, UNIX, JIRA

**Client:** Northern Trust Co, Chicago, IL  
**Role:** DevOps Engineer

**Feb 2017 – Aug 2019**

#### **Responsibilities:**

- Created **Ansible** playbooks to automatically install packages from a repository, change the configuration of remotely configured machines, and deploy new builds and various automation purposes, file copy, permission changes, configuration changes, path specific folder creation.
- Installed **Nexus** repository tool to maintain the artifacts/dependencies jars.
- Responsible for **Continuous Integration (CI) and Continuous Delivery (CD)** process implementation using Jenkins along with PowerShell to automate routine jobs.
- Developed shell scripts for automation of the build and release process and developed Custom Scripts to monitor repositories and server storage.
- Designed and implemented Kubernetes resources, including deployments, services, and persistent volume claims, to deploy and manage applications on **OpenShift**.
- Configured **CI/CD** pipelines on Jenkins and integrated with **OpenShift**.
- Provided support for production issues related to **OpenShift**, identifying, and resolving critical incidents and working with **Red Hat** Support to escalate and resolve issues.
- Used **Terraform** scripts to Automate Instances for Manual Instances that were launched before.
- Developed environments of different applications on **AWS** by provisioning on **EC2** instances using Docker, Bash, and Terraform.

- Involved in the development of test environments on **Docker** containers and configuring the Docker containers using Kubernetes.
- Used Ticketing tool **JIRA** to track defects and changes for change management, monitoring tools like **New Relic** and **CloudWatch** in different work environments in real and container workspaces.
- Extensively involved in infrastructure as code, execution plans, resource graphs, and change automation using Terraform.
- Used Terraform scripts to Automate Instances for Manual Instances that were launched before.
- Deploy code updates into test and production environments and work to roll environments forward.
- Automated various infrastructure activities like Continuous Deployment using **Ansible** playbooks and Integrated **Ansible** with **Jenkins** on **AZURE**.
- Implemented a **CI/CD pipeline** using **Azure DevOps (VSTS, TFS)** in both cloud and on-premises with GIT, MS Build, Docker, and Maven along with Jenkins's plugins.
- Experience in working in **.Net applications** and branching, Tagging, and Release Activities on Version Control Tools like Bitbucket.
- Knowledge of Splunk content development with the ability to create queries and dashboards.

**Environment:** Ansible, Terraform, AWS, Maven, RDS, Kubernetes, OpenShift, Redshift, Bitbucket, Cloud Check, Shell, Python Scripting, JavaScript, Terraform, Jenkins, SonarQube, Docker, ECS, CloudWatch, New Relic, Nagios, Azure.

**Client:** Datafactz, India

**Jun 2015 – Jul 2016**

**Role:** DevOps Engineer

#### **Responsibilities:**

- Build and release software baselines, **code merges**, branch and label creation and interfaced between development and **infrastructure**.
- Involved in the Installation and configuration of an automation tool Puppet that included the installation and configuration of the **Puppet** master, agent nodes, and admin control workstation. Created **Puppet** Manifests and modules to automate system operations and for deploying, configuring, and managing collected for metric collection and monitoring.
- Responsible for delivering an end-to-end continuous integration - continuous delivery system for the products in an agile development approach using **Jenkins**.
- Continuous integration and continuous deployment (**CI/CD**) pipelines using **Azure DevOps** such as **Git**, and Azure DevOpsYAML pipelines.
- Implemented identity and access management using **Azure Active Directory** (AAD) for role-based access control (**RBAC**) and authentication.
- Used **Maven** to build rpms from source code checked out from the Subversion repository, with **Jenkins** being the Continuous Integration Server and **Nexus** Artifactory as repository manager.
- Created and Maintained Subversion Branching, Merging, and tagging across each Production release and perform builds with **Continuous Integration** using Scripts.
- Hands-on Experience in using **Tomcat** and **Apache web servers** for deployments and hosting tools.
- Responsible for Defining Mapping parameters and variables and Session parameters according to the requirements and usage of workflow variables for triggering emails in **QA** and **UAT** environments.
- Hands-on experience working with various Bug Tracking tools like **JIRA**, and Bugzilla and integrated with **Junit**, HPQC, and other software testing procedures.

- Extensively worked as an Informatica Designer and Workflow Manager. Designed and migrated various reusable mappings and mapplets in Mapping Designer, sessions, and workflows in Workflow Manager to extract data from various sources like Mainframes, Flat files, SQL servers, and Oracle.
- Managed and monitored the server and network infrastructure using **Nagios**.
- Developed and Maintained **UNIX/Perl/ANT** for Build and Release tasks.

**Environment:** UNIX, MQ, Azure, Maven, Ant, Jenkins, Shell, Java, JIRA, Service now, Apache tomcat, VPC, Elastic Beanstalk, Docker, Nginx, Stratus COBOL, GIT, File System, Forms, Macros, JCL, Mainframes

**Client:** IVY COMPTECH, India

**Jun 2013 – May 2015**

**Role:** System Administrator– Unix/Linux Administrator

#### **Responsibilities:**

- Entire front-end and back-end modules of the project are developed using Python with Django Framework.
- Installed, upgraded, and configured **Red Hat Enterprise Linux 4/5/6**.
- Configuration and administration of **DNS, LDAP, NFS, NIS, TCP/IP, and NIS+** on Red Hat Linux.
- Hands-on experience working with production servers at multiple data centers.
- Automated server building using Kick starting **RHEL 6** and Jumpstarting Sun Solaris 10.
- Troubleshooting production servers and configuring standalone production servers for testing.
- Configured, and managed ESX **VMs** with virtual center and VI client.
- Performed Kernel tuning with the systole and installed packages with yum and rpm.
- Performed Disk management with the help of **LVM** (Logical Volume Manager).
- Configuration and Administration of **Apache Web Server and SSL**.
- Network designing, implementing & managing **LAN** Networks and **WAN** Networks.
- Backup management using **CRONTAB**, running the backup based on client requirements.
- Password-less setup and agent-forwarding done for **SSH** login using **SSH-keygen** tool.
- Established and maintained **network users, user environment, directories, and security**.
- Documented testing procedures and steps involved in data migration on production servers.
- Created Multiple **Ansible** Playbooks to Manage Patching and software Updates in Multiple Host **Nodes**.
- Created Bash/shell scripts to monitor system resources and system maintenance.
- Created and maintained user accounts in **Red Hat Enterprise Linux (RHEL)** and other operating systems.

**Environment:** Linux, VMware, Red Hat, Veritas, PAAS, BASH, Crontab, Apache, JBOSS, IBM Server, MY SQL, Ansible.