Pavan Kalyan Penchikalapati

Senior DevOps Engineer

pavanp.devopsengineer@gmail.com | +91-6309123731

Summary:

Experienced DevOps and Cloud Engineer with over 4+ years of expertise in designing, implementing, and managing scalable cloud solutions. Proficient in optimizing and automating deployments using AWS and Azure services. Skilled in Kubernetes, Docker, and Terraform to enhance application performance and reliability. Hands-on experience in building CI/CD pipelines with Jenkins, Gitlab Actions and Azure Pipelines for efficient software delivery. Adept in configuration management using Ansible and practicing infrastructure as code (IaC). Proven track record of collaborating effectively across teams to deliver high-quality software solutions on time. Familiar with monitoring tools like azure monitor, cloud watch, version control using Git, and proficient in Python and Shell scripting.

Highlights:

- 4 years of experience as DevOps and Cloud Engineer involving extensive work towards code compilation, automation, packaging, building, debugging, managing, tuning and deploying code across multiple Cloud environments.
- In-depth understanding of the principles and best practices of Software Configuration Management (**SCM**) in **Agile**, **SCRUM**, **Vmodel** methodologies.
- Worked on EKS, Kubernetes, ECS, AKS, Docker Swarm for microservices applications
- Wrote **Ansible** playbooks to manage configurations and automate installation process.
- Used **Ansible** and **Ansible roles** as Configuration management tool, to automate repetitive tasks, quickly deploys critical applications, and proactively manages change.
- Managing the configurations of multiple servers using **Ansible**
- **Worked on Terraform and Terraform modules to provide IAC on AWS**
- Managed environments DEV, QA, UAT and PROD for various releases and designed instance strategies.
- Good knowledge about **CI/CD** using Jenkins
- Managed **Multi Flavor of Linux** and **Windows** virtual servers with Ansible.
- Extensively worked on Jenkins and **Azure pipelines** for continuous integration (CI) and for End-to-End automation for all build and deployments.
- Performed Automation and Scaling of applications using **Kubernetes and EKS**
- Used **Ansible** to orchestrate software updates and verify functionality.
- Terraforming and Cloud Formation to code all infrastructures into Azure, AWS and GCP.
- Organized different infrastructure resources like physical machines, VMs and even containers using Terraform.
- Worked on Terra form to set up the AWS infrastructures such as launching theEC2 instances, S3buckets, VPC, Subnets, and created the module driven AWS Infrastructure with Terraform
- Used **GIT** as Version Control System for two applications. Managed development streams and Integration streams.
- Used **AWS** Bean Stalk for deploying and scaling web applications and services developed with Java, Node.js, Python and **Ruby** on familiar servers such as Apache, and IIS.
- Worked on **Docker** container snapshots, attaching to a running container, removing images, managing director structures and managing containers.
- Experience working on several Docker components like Docker Engine, Hub, Machine, **Compose** and Docker Registry.

- Experience working on **docker hub,** creating docker images and handling multiple images primarily for middleware installations and domain configurations.
- Used **Docker** for configuration management automation
- Setup Elastic Load Balancer for distributing **traffic** among multiple Web Logic servers and involved in deploying the content cloud platform on Amazon Web Services using **EC2, S3** and **EBS.**
- Creating the AWS Security groups to enable firewall.
- Automated **AWS** volumes snapshot backups for enterprise using Lambda.
- Created functions and assigned roles in AWS **Lambda** to run python scripts.
- Administration and maintenance of **Docker** runtime environment, Versioning and lifecycle management of **Docker images**, Experienced in Docker orchestration framework.
- Automated Compute Engine and Docker Image Builds with **Jenkins** and **Kubernetes.**
- Having good implementation experience with installation and configuration of **Kubernetes,** clustering them and managed local deployments in **Kubernetes.**
- Provided consistent environment using **Kubernetes** for deployment scaling and load balancing to the application from development through production, easing the code development and deployment **pipeline** by implementing **Docker** containerization.
- Focused on **containerization** and immutable infrastructure. Docker has been core to this experience, along with Kubernetes**, Docker Compose, AWS ECS Azure EKS & Mesos.**
- Integrated **Kubernetes** with network, storage, and security to provide comprehensive infrastructure and orchestrated container across multiple hosts.
- Created Docker images using a **Docker file**. Worked on Docker container **snapshots**, removing images and managing **Docker volumes** and experienced with Docker container service.
- **Ansible** playbooks to manage configurations and automate installation process.
- Used **Ansible** and **Ansible** Tower as Configuration management tool, to automate repetitive tasks, quickly deploys critical applications, and proactively manages change.

Technical Skills:

Operating Systems	Linux Red Hat (8.x, 7.x, 6.x), Linux Centos, Ubuntu, Windows 2016/2022/12/win 11/7,
Version Control Tools	Git, Github, Gitlab, Azure Repos, BitBucket, Github action
Web/Application Servers	Web Logic, Apache Tomcat, Web Sphere, JBOSS and Microsoft IIS
Automation Tools	Jenkins, Azure pipeline, Bamboo,CircleCI
Build Tools	Maven, gradle and MS Build.
Configuration Tools	Puppet and Ansible
Databases	MySQL, AWS RDS, MS Access and SQL Server 2012/2008R2/2005
Bug Tracking Tools	JIRA, Remedy, IBM Clear Quest.
Scripting	PowerShell, shell scripting, python scripting.
IaC	Terraform, ARM Template, cloud formation
Virtualization Tools	Docker, VM virtual Box and VMware.
Monitoring Tools	Cloud watch, Prometheus, Grafana, Azure Monitor, ELK
Cloud Platform	Amazon Web Services, Azure

Kubernetes	Azure Kubernetes Services (AKS), Elastic Kubernetes Service (EKS)
Languages	Python

Professional Experience:

- **Currently employed at Capgemini, Pune as a Cloud DevOps Engineer from Sep 2023 to till date**
- **Worked as a Cloud Engineer for FIS, Pune from Aug 2021 to July 2023**

Client: CITI BANK **Sep 2023 – Present**
Project: Citi Bank Commercial Banking System **Cloud DevOps Engineer**

Responsibilities:
Created and maintained continuous integration (CI) using tools Azure DevOps over different Environments to facilitate an agile development process, which is automated and repeatable, enabling Teams to safely deploy code many times a day while ensuring operational best practices are supported.

Azure Roles and Responsibilities:

- Configure Azure Repos to manage source code (e.g., application code for inventory management) and integrate with Git workflows (branching strategies like GitFlow).
- Create YAML-based Azure Pipelines to automate builds, unit tests, and artifact generation (e.g., Docker images stored in Azure Container Registry, ACR).
- Set up multi-stage pipelines with stages for development, staging, and production environments, ensuring zero-downtime deployments using blue-green or canary strategies.
- Integrate static code analysis (e.g., SonarQube) and security scanning tools (e.g., OWASP ZAP) to enforce quality gates.
- Collaborate with development teams to define pipeline triggers (e.g., pull requests, scheduled runs) and optimize build times.
- Break down the application into microservices (e.g., Inventory Service, Order Service, Auth Service) and define their boundaries using Domain-Driven Design (DDD) principles.
- Containerize each microservice with Docker, optimizing images for size and security (e.g., using Alpine base images, multi-stage builds).
- Create Kubernetes manifests or Helm charts for each microservice, specifying Pods, Deployments, and Services with labels for service discovery (e.g., app: inventory-service).
- Configure AKS namespaces to isolate microservices (e.g., inventory-ns, order-ns) for better resource management and security.
- Deploy a service mesh like Istio or Linkerd on AKS to manage traffic between microservices (e.g., HTTP/gRPC calls between Order Service and Inventory Service).
- Configure mTLS (mutual Transport Layer Security) for secure inter-service communication, aligning with BFSI-grade security requirements.
- Set up retry policies, circuit breakers, and timeouts in the service mesh to handle failures gracefully (e.g., retry failed inventory updates 3 times before failing over).
- Use Kubernetes Service objects (e.g., ClusterIP) and Istio Virtual Services to route traffic efficiently, supporting Tata's low-latency goals.
- Integrate with Azure Application Gateway or an AKS ingress controller for external traffic routing to microservices (e.g., /orders to Order Service).
- Containerize application components (e.g., frontend, backend APIs, database services) using Docker, ensuring compatibility with AKS runtime environments.

- Write Kubernetes manifests (e.g., Deployments, Services, ConfigMaps) or Helm charts to define application deployments, integrating with Terraform-provisioned AKS clusters.
- Configure AKS ingress controllers (e.g., NGINX or Azure Application Gateway) to manage traffic routing for the inventory management app, supporting Tata's low-latency goals.
- Set up Horizontal Pod Autoscalers (HPA) and Cluster Autoscalers to dynamically adjust resources based on demand (e.g., peak shopping seasons).
- Implement rolling updates and rollback strategies in pipelines to deploy application updates without downtime.
- Secure AKS clusters with RBAC (Role-Based Access Control), Azure AD integration, and network policies, aligning with BFSI-grade security standards.
- Work with Tata's network engineers and AT&T's cloud architects to ensure AKS clusters integrate with global connectivity (e.g., TGN-Pacific for Asia access).
- Troubleshoot pipeline failures, AKS pod crashes, or Terraform provisioning issues using tools like kubectl, Azure CLI, and pipeline logs.
- Conduct root cause analysis (RCA) for deployment incidents and implement preventive measures (e.g., stricter linting for Terraform code).
- Mentor junior engineers on best practices for Azure DevOps, Terraform, and Kubernetes, fostering a culture of knowledge sharing.
- Automate Terraform plan/apply steps within Azure Pipelines, integrating with approval gates for production deployments.
- Develop custom Azure DevOps extensions or scripts (e.g., PowerShell, Bash) to streamline environment setup and teardown.
- Optimize Docker image builds by implementing multi-stage builds and caching, reducing pipeline execution time.
- Set up automated testing frameworks (e.g., Selenium for UI, Postman for APIs) within pipelines to validate application functionality pre-deployment.

AWS Roles and Responsibilities:

- Deploying infrastructure on AWS utilizing services such as EC2, RDS, VPC and Managed Network and Security, Route 53, Direct Connect, IAM, Cloud Formation, AWS Ops Works (Automate operations), Elastic Beanstalk, AWS S3, Glacier, (Storage in the cloud) and Cloud Watch Monitoring Management.
- Handling and maintaining Azure DevOps Pipeline with ADO and Terraform.
- Maintaining Terraform for Infrastructure as Code for Azure and GCP.
- Experienced in creating AWS IAM and Security Group in Public and Private Subnets in VPC. Created AWS Route53 to route traffic between different regions.
- Launched AWS and Open Stack instances (SUSE/Ubuntu/CentOS) and Configured launched instances with respect to specific applications.
- Implementing AWS Lambda functions to run scripts in response to event in Amazon Dynamo DB table, S3 buckets, HTTP requests using Amazon API Gateway.
- Involved in creating, configuring AWS VPC services, installed EC2 instances for the new development team, and used AWS Route 53 to maintain the traffic and to create the DNS name.
- Create a custom image of an Azure VM with Azure PowerShell and create a VM scale set and deploy a high available app on Windows with Azure PowerShell.
- Experience in load balance Windows VM in Azure to create high available application with Azure PowerShell and manage Azure Virtual Network for Windows VM with Azure PowerShell.
- Experience in Backup and restore files for Windows VM in Azure cloud.
- Written Python scripts and PowerShell for setting up baselines, branching, merging, and automation processes across the environments using SCM tools like GIT on Linux and Windows platforms.

- Implementing VPC, Auto Scaling, S3, EBS, ELB, Cloud Formation templates and CloudWatch services from AWS.
- Used Maven dependency management system to deploy snapshot and release artifacts to Nexus to share artifacts across projects and environments. Built applications using Maven, Gradle scripts.
- Responsible for Installing, administering, repository management, User management in Nexus.
- Involved in checking the quality of the code by using SonarQube.
- Working knowledge/exposure in TOMCAT APACHE, WEB LOGIC & WEB SPHERE, formulated and executed designing standards for DNS servers.
- Involved in security testing, Windows Server Administration (WSUS), Infrastructure Solutions, ADD and DNS.
- Web application development using Web forms, C#, VB.net, JCL, Jscript and AJAX control.
- Used SVN for source code repository, code integration, Defect tracking in JIRA and HP QC.
- Deploy application code using CI/CD pipeline with Azure DevOps in Azure cloud, scale VM build automation using Azure DevOps in Azure VM agent plug-ins.
- Worked on Performance Monitoring, resolving network issues &tuning the system using tools.
- Developing plan for problem and incident response & management process based on Information Technology Infrastructure Library (ITIL).
- Improved team performance and accelerated release cycle accelerate delivery of new features and increase overall product qualities.
- Maintaining and developing Docker images for a tech stack including Cassandra, Kafka, Apache and several in house written in both OpenStack and AWS cloud on Kubernetes.
- Used Docker and Kubernetes to manage micro services deployment.
- Used Kubernetes to create Pods, Config Maps and deployments into the cluster.
- Hands on experience on Kubernetes to automate the deployment, scaling and operations of application containers across clusters of hosts.
- Worked with containerization tool Docker and running containers and services and utilizing docker swarm.
- API to provide lightweight containers that run processes isolation and worked on creation of customized Docker container images, tagged and pushed the images to the Docker Hub repository.
- Built and deployed a Ansible Server in Azure &AWS for infrastructure automation.
- Creating Chef Cookbooks and Recipes to maintain and automate various parts of infrastructure.
- Responsible for configuration changes, code building, code staging, build automation and deployment using tools like Octopus which also includes scripting in command line, Python, PowerShell, Perl.
- Configuration on WebSphere which includes writing Linux scripts- Shell scripting.
- Continuous Integration using Maven, Jenkins, Hudson.
- Involved in migration of Artifactory & GIT server.
- Created S3 bucket to host a static website with logging and versioning and customizing the template as per the client's requirement.
- DB administration and scripting in SQL, MySQL, Oracle databases for report generation and data management.
- Created lambda and wrote lambda function in Python script to stream the data from S3 bucket to Elasticsearch.
- Customized the log monitoring for a better and convenient reading of the logs by creating the indices and passing them on to the Elasticsearch which provides a URL for Kibana access.
- Responsible for designing & deploying new ELK clusters (Elasticsearch, Logstash, Kibana, beats, Kafka, zookeeper.
- Responsible for merging, tagging and conflict resolution in GIT and administered version control systems to create daily backups and checkpoint files.

Azure and Azure Devops Roles and Responsibilities:

- Implemented layered **security solutions** including **NSGs**, **ASGs**, **Azure Firewall**, and **Azure DDoS Protection** to enhance application security.
- Utilized **VPN Gateway**, **VNet-to-VNet VPN**, and **ExpressRoute** for secure network access and interconnectivity.
- Worked with **Azure Pipelines** to deploy applications into **Azure App Services** and created **Azure AKS clusters** with **Azure Container Registry** for container management.
- Managed **Azure AKS clusters** for container orchestration and application deployment, including **Azure Functions** and web applications.
- Developed and deployed infrastructure using **ARM templates** to automate resource provisioning in **Azure**.
- Developed **RESTful web APIs** for database communication and **Azure Function Apps** for **DB2 integration**.
- Created AWS **Infrastructure as Code** templates using **Terraform** for staging and production environments.
- Converted **CloudFormation JSON templates** to **Terraform** for improved infrastructure management.
- using **Python** and **PowerShell scripts**.
- Implemented **AWS Lambda functions** to automate scripts triggered by events from **DynamoDB**, **S3**, and **API Gateway**.
- Configured **Jenkins pipelines** for continuous deployments, ensuring security and managing parallel builds.
- Integrated **UFT** with **TFS** for test case automation and trained team members on migrating **UFT scripts** to **Azure DevOps GitHub**.

FIS **Pune**
Project: JPMC **Cloud Engineer**

Responsibilities:

- Setup and build **AWS** infrastructure various resources, **VPC EC2, S3, IAM, EBS,** Security Group, Auto Scaling, and RDS in Cloud Formation **JSON** templates.
- Working in implementation team to **build** and engineer servers for **Linux** & **AIX** operating systems.
- Working on AWS cloud to provision new instances. S3 Storage Services. AWS EC2 & **Cloud watch** services. **CI/CD** pipeline management through Jenkins.
- Automated Compute Engine and Docker Image Builds with **Jenkins** and **Kubernetes.**
- Having good implementation experience with installation and configuration of **Kubernetes,** clustering them and managed local deployments in **Kubernetes.**
- Provided consistent environment using **Kubernetes** for deployment scaling and load balancing to the application from development through production, easing the code development and deployment **pipeline** by implementing **Docker** containerization.
- Focused on **containerization** and immutable infrastructure. Docker has been core to this experience, along with **AWS ECS** and **Kubernetes.**
- Integrated **Kubernetes** with network, storage, and security to provide comprehensive infrastructure and orchestrated container across multiple hosts.
- Created Docker images using a **Docker file**. Worked on Docker container **snapshots**, removing images and managing **Docker volumes** and experienced with Docker container service.
- Wrote **Ansible** playbooks to manage configurations and automate installation process.
- Used **Ansible** and **Ansible** Tower as Configuration management tool, to automate repetitive tasks, quickly deploys critical applications, and proactively manages change.
- Managing the configurations of multiple servers using **Ansible.**

- Managed Artifactory repository for the current project created a new repository and provisioned it.
- Performed Automation and Scaling of applications using **Kubernetes.**
- **Setup** and build **AWS infrastructure** resources such as **VPC**, **EC2**, **S3**, **IAM**, **EBS**, **Security Groups**, **Auto Scaling**, **Lambda**, and **RDS** in **CloudFormation**.
- **Launched Amazon EC2 Cloud Instances**, configuring them for specific applications to meet business needs.
- Designed and deployed applications utilizing the **AWS stack**, focusing on **high availability**, **fault tolerance**, and **auto-scaling** in **CloudFormation**.
- Designed and developed **Terraform scripts** for effective infrastructure management and provisioning.
- Installed and configured **GIT**, writing pre-commit, post-commit, and post-receive hooks, handling merging, tagging, and conflict resolution.
- Coordinated with developers on **branching** and **naming conventions** using **GIT** and **SVN**.
- Installed, configured, and supported **CI/CD tools** like **Jenkins**, **Bamboo**, **Nexus**, and **SonarQube** for continuous integration and deployment.
- Configured **Jenkins** with **Maven** to generate **WAR files** and archived them efficiently.
- Configured local **Maven repositories** with **Nexus** and scheduled **Jenkins projects** for continuous integration workflows.
- Wrote **Ansible playbook** and roles to automate middleware installations and deployment activities, enhancing operational efficiency.
- Migrated **VMware VMs** to **AWS** using **Chef** for managing services like **EC2**, **S3 Bucket**, **Route**, **ELB**, and **EBS**.
- Implemented and maintained **Ansible Configuration Management** across **VMware** and **AWS environments** for consistent deployments.
- Worked with **Ansible** in conjunction with **Vagrant** in a **DevOps environment** to streamline development processes.
- Used **Ansible** as **Infrastructure as Code**, building and deploying applications with **YAML** and **Python functions** for automation.
- Orchestrated software updates and verified functionality with **Ansible**, ensuring system reliability.
- Managed containers using **Docker**, creating **Docker images** for isolation and faster deployments.
- Developed **Ansible scripts** for automated server provisioning and **Docker image management**.
- Utilized **Jenkins pipelines** to drive **microservices builds** to the **Docker registry** and deployed to **Kubernetes**.

Education:

- **Bachelor of Commerce (B.Com) in Computer Applications, Rayalaseema University, Andhra Pradesh — 2021**

Certifications:

- **Microsoft Certified: Azure Administrator Associate (AZ-104)**
- **Microsoft Certified: Azure DevOps Engineer Expert (AZ-400)**