 A picture containing text, clipart

Description automatically generated

***Thandarupalli Naveen Goud***

***Cloud DevOps Engineer***

***LinkedIn*:** [*https://www.linkedin.com/in/naveen-goud-thandarupalli-0ba02423b/*](https://www.linkedin.com/in/naveen-goud-thandarupalli-0ba02423b/)

***Email: naveentg1472@gmail.com | Phone: 402-227-4542***

**PROFESSIONAL SUMMARY:**

* Having **10+** years of experience in IT industry with **6** years as a Cloud DevOps Engineer, Build and Release Engineer which includes Software Configuration Management (SCM), Build/Release Management, Continuous Integration and Continuous Delivery using different tools and **4** years as a linux administrator.
* Expertise in DevOps tools such as **GIT, Docker, Kubernetes, AWS, Azure, Jenkins, Maven, Ant, Gradle, SonarQube, Nexus Repository, Jfrog, Ansible, Terraform, Splunk, New Relic, Prometheus, Grafana.**
* Experienced in Continuous Integration (CI) and Continuous Delivery (CD) process implementation using **Jenkins, Azure Pipelines, GitLab CICD pipelines and GitHub Actions.**
* Experienced in Configuration Management Tools such as **Ansible, Chef** Automated Linux production server’s setup using the **Ansible playbooks, roles, Cookbooks**.
* Experience with **Docker** containerized environment, hosting web servers on containers, building Docker images using Docker.
* Expertise working with **Kubernetes** to automate deployment, scaling and management of web Containerized applications.
* Worked on **Kubernetes** to orchestrate **Docker** containers of new and existing applications as well as deployment and management of complex run time environment. Setup RBAC Roles on Kubernetes Cluster.
* Automated deployment, configuration, scaling, and maintenance tasks using **Infrastructure-as-Code (IaC)** tools such as **Azure Resource Manager templates, Terraform**.
* Designed and distribution of data across all the nodes and Clusters on different availability zones in **AWS Redshift** and experienced in automating the infrastructure using **Terraform** in AWS console.
* Expertise in administering and automating operations across multiple platforms **(UNIX, Linux, Windows and MAC)**.
* Experienced in working on both On-Premise and Cloud (AWS/Azure) platforms utilizing **DevOps/Agile** operational processes.
* Expertise in AWS Cloud utilizing various services like **EC2, VPC, ELB, Auto-Scaling, Security Groups, Route53, IAM, EBS, Load Balancers, RDS, S3, SNS, SQS, CloudWatch, Elastic Beans Stalk, CloudFormation & EKS** and other services of AWS.
* Built and configured VPC in the AWS cloud to support Enterprise Data Warehouse hosting including **VPC, Public & Private Subnets, inbound & outbound Security Groups, Route Tables, and Elastic Load Balancer.**
* Designing, developing, testing, and deploying cloud-based applications and services using **Microsoft Azure** technologies and services.
* Understanding and implementing Azure services such as **Azure Functions, Azure Storage,** **Azure Container Registry (ACR),** **Azure Monitor, Azure App Service, Azure Logic Apps, Azure Kubernetes Service (AKS), Azure App Service, Azure Data Factory,** **Azure SQL** and others.
* Building, integrating, and deploying **Azure DevOps pipelines/ Jenkins** for continuous integration and continuous deployment **(CI/CD)** workflows.
* Responsible for setting up and configuring **Kafka clusters** on **Azure**, including provisioning **virtual machines**, **installing Kafka software, configuring networking, security, and authentication settings, and setting up backup and disaster recovery solutions.**
* Implemented monitoring and alerting solutions to proactively detect and resolve issues in **Kafka** clusters using **Azure Monitor**, **Apache Kafka** monitoring tools, and other third-party monitoring tools to collect and analyze metrics, logs, and other performance data, and setting up automated alerts and notifications.
* Worked closely with development teams to integrate **Kafka** into the DevOps pipeline, including implementing **CI/CD** processes, automated testing, and versioning.
* Used AWS **lambda** to run servers without managing them and configured to trigger, to run code using S3.
* Experience with **Linux/UNIX** environments and scripting for Build & Release automation.
* Hands-on experience in DevOps automation development for Linux and Windows Environments. Experience in **UNIX** environment and expertise in several flavors of **Linux** including **Red Hat, CentOS,** and **Ubuntu.**
* Experience in Configuring and Administrating **Nexus Repository Manager/JFrog** Artifactory
* Designed highly available, cost effective and fault tolerant systems using multiple **EC2 instances, Auto Scaling, Elastic Load Balancer and AMIs and S3 Glacier, EBS for ALL** environments as well as infrastructure.
* Performed application migration to **Docker** and containers for various projects and established the use of **Blue/Green Deployments, Load Balancers and AWS EC2** spot instances.
* Experience on Azure Site Recovery and Backup Installed and Configured the **Azure Backup agent** and **virtual machine backup**, Enabled Azure Virtual machine backup from the **Vault** and configured the **Azure Site Recovery (ASR)**.
* Experience in branching, tagging, and maintaining the version across the environments using SCM tools like **Subversion (SVN), Git, Bit Bucket**, and **GitHub**.
* Expertise in building **Kafka cluster**, cluster maintenance, trouble shooting, monitoring, commissioning and decommissioning Data nodes, Troubleshooting, Manage and review data backups, Manage & review log files.
* Expertise in working with diverse Database platforms for Installing, Configuring and Managing **RDBMS** tools like **MYSQL, Oracle and NOSQL Dynamo DB, and Mongo DB/Cosmos DB.**
* Worked on various network protocols **like HTTP, UDP, FTP, TCP/IP, SMTP, SSH, SFTP & DNS** and technologies **like load balancers (ELB), ACL, Firewalls**.
* Experienced with handling production incidents, queries, problems through remedy ticketing system and non -production issues/tasks/incidents through **JIRA** ticketing system and **ServiceNow**.
* Proficient in **Git version control system**, including branching, merging, and resolving conflicts.
* Experience with continuous integration tools such as **Jenkins**, **Travis CI,** or **CircleCI**.
* Familiarity with agile project management tools like **Jira** or **Trello**.
* Good understanding of web and application servers like **Apache Tomcat, WebSphere, Nginx and IIS**.
* Used build tools like **Maven, Gradle and MSBuild** for the building of deployable artifacts from source code.
* Experience in building and deploying **Java/.NET following SOA/Microservices** applications and troubleshooting the build and deploy failures.
* Proficient with **Terraform** Configuration files to spin up the infrastructure very easily and efficiently.
* Experienced using different log monitoring tools like **Splunk, New Relic, ELK (Elastic search, Logstash, Kibana)** for to see logs information, monitor, get the health & security notifications from nodes.

**EDUCATION:**

* Bachelors of Computer Science from JNTU in 2012

**CERTIFICATIONS**

* Microsoft Certified Azure Administrator
* Certified Kubernetes Administrator
* AWS Developer – Associate
* Certified Terraform Associate

**TECHNICAL SKILLS:**

|  |  |
| --- | --- |
| **AWS Services** | RDS, EC2, VPC, IAM, Cloud Formation, EKS, EBS, S3, ELB, Auto Scaling, Cloud Trial, SQS, SNS, SWF, Cloud Watch, Load Balancers, Elastic Beans Stalk, ECR. |
| **CI/CD** | Jenkins, Azure Pipelines. GitHub Actions |
| **Artifactory Storing Tools** | Jfrog and Nexus |
| **Servers** | Nginx, IIS, Httpd, Apache Tomcat |
| **Documentation Tools** | Confluence |
| **Operating Systems** | Microsoft Windows , Linux, UNIX. |
| **Tracking Tools** | Jira, Azure Devops |
| **Code Scanning** | Sonar Qube, Jfrog X ray, Trivy |
| **Databases** | RDS, Cosmos DB, My SQL DB. |
| **Monitoring Tools** | Cloud Watch, Cloud Trail, Azure App Insights, Azure Monitor |
| **Configuration & Automation Tools** | Ansible, Chef. |
| **Container Platforms** | Docker, Kubernetes, Open Shift. |
| **Monitoring Tools** | New Relic, Splunk, Datadog, Grafana, App Dynamics. |
| **Languages** | Python, Shell scripting. |
| **Cloud Platforms** | Azure, AWS. |
| **Azure Services** | App Services, Key vault, function app, Blob storage, Azure Active Directory (Azure AD), Service Bus, Azure Container Registry (ACR) and Azure Kubernetes service (AKS), Azure SQL. |
| **Version Control Tools** | GiHub, Git Lab, Bit Bucket. |
| **Infrastructure-as-code** | Terraform, CloudFormation |

**WORK EXPERIENCE**

**Client: VIR BIOTECHNOLOGY, San Francisco, CA**

**Role: Azure/DevOps Engineer | Oct 2022 - Till Date**

|  |
| --- |
|  |

**Responsibilities:**

* Implementing and managing the end-to-end DevOps practices and processes using **Azure DevOps** tools, including source control, continuous integration, continuous delivery, automated testing, and deployment.
* Responsible for managing source code repositories using version control systems such as **Git** in **Azure DevOps**, including creating and managing branches, merging code changes, resolving conflicts, and ensuring proper versioning and branching strategies are followed.
* Responsible for creating and managing infrastructure as code (IaC) templates using tools such as **Azure Resource Manager (ARM) templates or Terraform** to automate the provisioning and management of Azure resources required for application deployments.
* Implemented GitOps workflows using Argo CD, enabling declarative **configuration management** and version-controlled deployments.
* Configured Argo CD notifications and alerts to monitor **deployment status** and application health, ensuring timely response to issues.
* Developed **custom plugins** or extensions for Argo CD to extend its functionality and meet specific deployment requirements.
* Integrated Argo CD with **Helm charts** for managing complex application deployments and dependencies. Conducted performance tuning and optimization of Argo CD deployments for improved scalability and reliability.
* Contributed to the Argo CD **open-source** community by reporting bugs, submitting enhancements, or participating in discussions.
* Responsible for automating the deployment of applications and services to various environments using tools such as **Azure DevOps Pipelines**, **Jenkins**, or other deployment technologies, ensuring that deployments are consistent, repeatable, and reliable.
* Implemented Argo CD for **continuous delivery** of applications in Kubernetes clusters, providing automated deployment and rollback capabilities.
* Configured **Argo CD** settings for application-specific requirements, such as hooks, rollouts, and health checks, ensuring smooth deployment workflows.
* Integrated Argo CD with Git repositories to automate **synchronization** of application configurations and versions, improving traceability and auditability.
* Provided training and support to team members on **Argo CD** usage and best practices for managing Kubernetes applications.
* Responsible for managing **configuration files, settings, and secrets** for applications and services using tools such as **Azure Key Vault or** other configuration management solutions,ensuring that sensitive information is stored securely and properly managed.
* Experience working with several azure services such as **Azure Virtual Machines (VMs), Azure App Service, Azure Blob Storage, Azure SQL Database, Azure Cosmos DB, Azure Functions, Azure Virtual Network (VNet), Azure Active Directory (AD), Azure Key Vault, Azure Kubernetes Service (AKS), Azure Databricks, Azure Data Factory, Azure Firewall, Azure Load Balancer, Azure VPN Gateway, Azure Backup, Azure Monitor, Azure Policy, Azure Resource Manager (ARM), Azure Event Hubs**
* Responsible for monitoring the health and performance of applications and services in production environments using tools such as **Azure Monitor**, **Azure Application Insights** or other monitoring solutions.
* Managed, Configured and scheduled resources across the cluster using **Azure Kubernetes Service (AKS)/EKS/OpenShift.**
* Responsible for deploying and managing **Apache Kafka** clusters on **Azure**, including provisioning **virtual machines** or containers, configuring networking, and setting up authentication and authorization.
* Responsible for monitoring the health and performance of **Apache Kafka clusters** on Azure, including monitoring topics, partitions, and brokers, as well as tuning various Kafka configurations for optimal performance.
* Responsible for setting up data producers to ingest data into **Kafka** topics from various sources, such as applications, IoT devices, or external systems.
* Implemented data streaming solutions using **Kafka Streams** or other **Kafka clients** on Azure.
* Managed GitHub Enterprise and Azure DevOps environments, ensuring smooth operations and adherence to best practices.
* Implemented CI/CD pipelines using **GitHub** **Actions** and **Azure** **DevOps** workflows, leading to standardized development processes.
* Utilized **SonarQube** for code quality analysis and implemented integrations with monitoring tools for proactive issue identification.
* Responsible to integrate Apache Kafka with other Azure services, such as **Azure Event Hubs, Azure Blob Storage,** or Azure Stream Analytics, to enable data processing, analytics, or storage workflows.
* Worked on **Azure Data Factory/ Azure Synapse** to integrate data of both on-prem (**MY SQL, SQL**) **and cloud (Blob storage)** and applied transformations to load back to **Snowflake.**
* Performed the migration of large data sets to **Databricks (Spark),** create and administer cluster, load data, configure data pipelines, loading data from ADLS Gen2 to **Databricks** using **ADF** pipelines.
* Ingested data in mini-batches and performs RDD transformations on those mini-batches of data by using Spark Streaming to perform streaming analytics in **Data bricks**.
* Created various pipelines to load the data from **Azure data lake** into Staging SQLDB and followed by to **Azure SQL DB.**
* Worked on creating the infrastructure by using the terraform and initiated azure services like **AKS, ACR, VN, VM** ..etc.
* Utilized **Azure Logic Apps** to build workflows to schedule and automate batch jobs by integrating apps, **ADF pipelines**, and other services like **HTTP requests, email triggers** etc.
* Worked extensively on **Azure data factory** including **data transformations, Integration Runtimes, Azure Key Vaults**, **Triggers** and migrating **data factory** pipelines to higher environments using **ARM** Templates.
* Experienced in developing **Ansible** roles and **Ansible Playbooks** for the server configuration.
* Experience in working with **GIT** to store the code and integrated it to **Ansible** to deploy the playbook.
* Experienced in using **Tomcat, JBOSS, WebLogic and WebSphere Application servers** for deployment.
* Expertise in Querying RDBMS such as Oracle, MySQL and SQL Server by using PL/SQL for data integrity and proficient in multiple databases like **MongoDB, Cosmos DB, MySQL, ORACLE**.
* Experience using **CRONTAB** for job/task scheduling in Linux environment.

**Client:** **Comcast, Philadelphia, PA**

|  |
| --- |
| **Role: AWS/DevOps Engineer | April 2021 –Sep 2022** |

**Responsibilities:**

* Working with **AWS CLI** and **API** to manage resources on AWS for many services such as a **EC2, S3, VPC, Lambda, Cloud formation, Cloud Watch, RDS, Dynamo DB, ELB, Auto-scaling**, **Route53**, **IAM** created **python** script using **AWS API** Calls to manage all resources deployed on **AWS**.
* Used **AWS Beanstalk** for deploying and scaling web applications and services developed with **Java, .NET, PHP and Node.js** on familiar servers such as **Apache** and **IIS**.
* Experience in **Blue/green deployment** strategy by creating new applications which are identical to the existing production environment using **CloudFormation** templates & **Route53** weighted record sets to redirect traffic from the old environment to the new environment via **DNS**.
* Resolved complex issues related to **Argo CD deployments**, including troubleshooting connectivity, authentication, and configuration problems.
* Developed and maintained **runbooks** and documentation for common Argo CD issues and resolutions, reducing downtime and improving support efficiency.
* Participated in incident response and **root cause analysis (RCA)** for Argo CD-related incidents, implementing preventive measures to avoid future occurrences.
* Configured **EC2** instances by writing **Terraform** templates, **Chef** cookbooks, recipes and pushed them on to **Chef** server. Wrote **Lambda** functions in **Python** for **AWS Lambda** and invoked **Python** scripts for data transformations and analytics on large data sets in **AWS Glue** clusters and **AWS Kinesis** data streams.
* Utilized Argo CD with tools like **Terraform or CloudFormation** to manage infrastructure provisioning and configuration changes.
* Implemented **GitOps principles** for infrastructure management, ensuring that all changes are version-controlled and auditable.
* Automated the creation and management of **Kubernetes resources** using Argo CD, improving deployment speed and reliability.
* Used **Terraform** in AWS VPC to automatically setup and modify settings by interfacing with control layer.
* Created reproducible builds of the **Kubernetes** applications, managed **Kubernetes** manifest files and Managed Release of **Helm** packages and scheduled, deployed and managed Container replicas onto a node cluster using **Kubernetes.**
* Responsible for deploying and managing **Kubernetes** clusters using **Amazon EKS**, including cluster creation, scaling, upgrading, and monitoring. Have a deep understanding of **Kubernetes** concepts, such as **pods**, **services**, deployments, and **ingress**, and how they can be effectively used in **Amazon EKS**.
* Writing **Ansible** **Playbooks** with **Python** **Boto3** as a wrapper for configuration management of **AWS** nodes in the cloud environment and Tested **Ansible** playbooks on **AWS** instances using **Python** and involved in running **Ansible** Scripts to provision dev servers.
* Created **Docker file** for each **microservice** and changed some of the Tomcat configuration file which are required to deploy Java based application to the **Docker container.**
* Created clusters using **Kubernetes** and worked on creating many pods, replication controllers, deployments, labels, health checks and ingress by writing **YAML** files as well as managed Kubernetes charts using **Helm**.
* Responsible for **Continuous Integration (CI)** and **Continuous Delivery (CD)** process implementation-using **Jenkins** along with **Python** and **Shell** scripts to automate routine jobs. Migrated Legacy applications to a repeatable **Continuous** **Integration** platform.
* Designed and developed **Continuous Integration & Continuous Delivery** pipelines using Code Pipeline, Code Build and Code Deploy plugins in Jenkins. Developed **Jenkinsfile** and automated the **Jenkins** pipeline using pipeline **Groovy** script for the applications.
* Responsible for installing **Jenkins** master and slave nodes and configure **Jenkins** builds for **Continuous Integration (CI)** and **Continuous Delivery (CD).**
* Creating and Implementing **branching** & **merging** strategy with multiple branches. Used **Git** as source code management repository to keep track of version changes. Implemented and Maintained the Branching and **Build/Release** strategies utilizing **GIT** source code management.
* Installed and configured **GIT** and communicating with the repositories in **GITHUB** and Created and maintained **Subversion/GIT** repositories, branches and tags.
* Designed and scripted using **MAVEN** for (J2EE, Web Services, Reusable JARS, Web Client and Open Source) in creating **MasterBuild.xml**, **Build. Properties** &**Pom.xml** files and provided technical support to development team for compilation problems.
* Implemented and maintained the monitoring and alerting of production, corporate servers and storage using **AWS CloudWatch** for efficiency.

Created **Python** scripts to automate AWS Services which includes **ELB, Cloud front, Lambda**, database security and application configuration also Developed them to take backup of **EBS** volumes using **AWS Lambda, CloudWatch**.

|  |
| --- |
| **Client: Wells Fargo, Dallas, USA**  **Role:** Azure **DevOps Engineer | Nov 2019 –** Mar **2021** |

**Responsibilities:**

* Focused on improving **software development, deployment**, and operations processes. Worked in designing, implementing, and maintaining scalable and highly available infrastructure solutions.
* Collaborated with development teams to define infrastructure requirements and create infrastructure as code (IaC) using tools like **Terraform, CloudFormation.**
* Automated the **provisioning and configuration** of infrastructure components such as servers, networks, load balancers, and databases, also implemented and maintained scalable and highly available infrastructure architectures, leveraging cloud platform Azure.
* Implement and maintain CI/CD pipelines to automate the **build, test, and deployment** processes for applications and services.
* Evaluated and recommended improvements to the **CI/CD pipeline** using Argo CD, leading to faster and more reliable deployments.
* Implemented automated testing strategies within Argo CD pipelines to ensure **application quality** and reliability.
* Conducted regular audits and reviews of **Argo CD configurations** and policies to maintain security and compliance standards.
* Proposed and implemented enhancements to the Argo CD deployment **architecture** to improve scalability and performance.
* Integrate source control systems using Git and build automation tools **Jenkins, GitLab CI/CD** to enable continuous integration and delivery of software.
* Managed and optimized **release processes**, including version control, release coordination, and rollback strategies.
* Have used **ServiceNow** for incident management, change management, and problem management.
* Have set up monitoring and alerting systems like **Splunk, Nagios, Azure Monitor, CloudWatch, ELK** to track application and infrastructure performance, and ensure system availability.
* Implement proactive measures to anticipate and prevent **system failures** or **performance degradation**.
* Participated in incident response and **on-call rotations** to address system outages or performance incidents.
* Collaborated with development and operations teams to conduct **root cause analysis (RCAs)** and implement measures to prevent recurrence.
* Worked to improve system reliability, resilience, and fault tolerance, implementing practices like fault injection, chaos engineering, and **disaster recovery planning**.
* Orchestrated the creation of Pods with Kubernetes and implemented Jenkins pipelines to manage the builds of microservices, facilitating their deployment to the **Docker registry and Kubernetes**.
* Leveraged Docker **to virtualize servers**, creating isolated environments for testing, development, and production.
* Integrated the Kubernetes Container orchestration framework with Docker, streamlining container management by **creating Pods, Configuration Maps, and Deployments**.
* Engineered Azure infrastructure using Terraform, ensuring reliable versioning and utilizing Azure **Terraform modules** for automated resource provisioning.
* Implemented CI/CD pipelines for **AKS** using **Helm charts** in Azure DevOps, enabling automated build, testing, and deployment processes for efficient software delivery.

**Client: PNC, Pittsburgh, PA**

**Role: DevOps Engineer | July 2018 - Oct 2019**

**Responsibilities:**

* Experience in Software Integration, Configuration, building, automating, managing and releasing code from one environment to another environment and deploying to servers.
* Working on multiple **AWS** instances by setting services like **Route53, SES, SQS, SNS** and created and administered DNS system using **Route53**.
* Worked with full AWS stack (**Lambda server-less, IAM, Blue-Green deployment**).
* Running Container Workloads in OpenShift On-Prem Private Cloud.
* Provisioned the highly available EC2 instances using **Terraform** and **Cloud Formation** and wrote new plugins to support new functionality in Terraform.
* Researched and developed scalable Linux deployments on AWS via **CoreOS** and **Docker** and used **Docker** Compose to develop dependent instances to be deployed in the **EC2** platform.
* Experience with **S3** Bucket services like creating buckets, managing policies, configuring buckets with permissions, logging, versioning and tagging and used **S3** Bucket and **Glacier** for storage and backup on AWS.
* Worked on **AWS Cloud Watch** for monitoring the application infrastructure and used **AWS email services** for notifying & configured S3 versioning and lifecycle policies to and backup files and archive files in Glacier.
* Worked on migrating the on-premises application from its physical Storage Racks to **AWS** by using **EC2, S3**.
* Served application data using **Lambda** functions to store data in NOSQL database **Dynamo DB.** Configured REST API's using API Gateway that hit **lambda** which in turn invokes lambdas to do necessary operations.
* Installed Redshift on AWS and imported the legacy data from Oracle to Redshift. Participated in integration of applications with existing API’s.
* Implemented and maintained **Ansible** configuration management spanning several environments in **VMware** and AWS cloud.

|  |
| --- |
| **Client: Extra marks Education, Hyderabad, India.**  **Role: Linux Administrator | Aug 2012 to July 2017** |

**Responsibilities:**

* Installed and configured **RHEL, Solaris** and **Windows Servers.**
* Monitored, analyzed, and responded to security events utilizing security event management and reporting tools.
* Worked on **account, user and group administration, Permissions policy management and quota implementation**.
* Installed, configured, supported and security implementation on SSH, NIS, DNS and HTTPD.
* Implemented and maintained server virtualization using **VMware** and **Oracle Virtual Manager**.
* Coordinated with application team in installation, configuration and troubleshoot issues with **Tomcat server**.
* Created virtual machine using **Oracle Virtual Manager, creating server pool, VM cloning. Conceived, Designed, Installed and Implemented CI/ CD automation system**.
* Participated in setup **Storage Area Network (SAN)** and **NAS** and file system backup and restore.
* 24x7 on-call rotation for support of production systems.
* Developed Ansible Playbooks to manage multiple host nodes to Upgrade Software’s.