**Icon

Description automatically generated with medium confidenceYaswanth G**

**AWS DevOps Engineer**

**Contact:** 3165303431

**Email:** [**gyaswanth499@gmail.com**](mailto:gyaswanth499@gmail.com)



\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**Professional Summary:**

* Skilled AWS DevOps engineer with over 8 years of hands-on experience automating, building, deploying, managing, and releasing code from one environment to other environments in AWS, leveraging configuration management, CI/CD, and DevOps processes. Experience in Build Engineering & Release Management process and as a Linux administrator.
* Good understanding of the principles and best practices of Software Configuration Management (SCM) in Agile (scrum) and Waterfall methodologies.
* Experienced in setting up the enterprise infrastructure on Amazon Web Services (AWS).
* Experienced in AWS Cloud Computing services, such as EC2, S3, API, Dynamo, EBS, VPC, ELB, Route53, Cloud Watch, Security Groups, Cloud Trail, IAM, Cloud Front, EMR, RDS, and Glacier also worked on DNS, SSL, and Firewalls.
* Implemented Amazon EC2 setting up instances, virtual private cloud (VPCs), and security groups.
* AWS CLI Auto Scaling and Cloud Watch Monitoring creation and update.
* Created databases in AWS using RDS, storage using S3 bucket, and configuring instance backups to S3 bucket.
* Excellent experience in documenting and automating the build and release process.
* Experienced in set up and maintenance of Auto-Scaling AWS stacks.
* Worked with IAM service creating new IAM users & and groups, defining roles and policies, and Identity providers.
* Managed application deployments into Kubernetes clusters across different environments like dev, test, staging, and production.
* Experience with container-based deployments using Docker, working with Docker images, Docker Hub, Docker registries, and Kubernetes.
* Skilled in maintaining Docker container clusters managed by Kubernetes and deployed Kubernetes applications using Helm charts.
* Configured Docker containers and created Docker files for different environments.
* Expertise in setting up Kubernetes clusters on AWS for production using Kops.
* Launched Docker containers on pods on top of multi-node Kubernetes cluster in the AWS environment using Kops.
* Good experience in writing Kubernetes yaml files for deployment of microservices into Kubernetes clusters.
* Used Packer and Ansible for creating AMI’s with the required configuration for the application.
* Implemented a CI/CD pipeline by creating a Cron job in Jenkins, which will integrate with the Git repository containing the Terraform code for any changes to modify the infrastructure.
* Templated AWS infrastructure as a code using Terraform to build staging and production environments.
* Deploy scalable infrastructure on Amazon Web Services (AWS) using Terraform.
* Experienced in Branching, Merging, Tagging, and maintaining the version across the environments using SCM tools like GIT and Subversion (SVN) on Linux platforms.
* Expertise in Source code control tools like SVN, and Bitbucket (Git) and good knowledge of Branching and merging code lines in the GIT.
* Developed integration workflows using the Mule ESB framework.
* integrated web services including SOAP as well as REST using Mule ESB
* Used Integration of Mule ESB system while utilizing AnypointMQ, HTTP, File system and SFTP transports.
* Deployed application in UNIX and to connect to see logs for fixing UAT/Production defects.
* Developing Mule ESB projects for the services with synchronous and asynchronous mule flows.
* Experience in Designing, Installing, and Implementing Ansible configuration management systems and in writing playbooks for Ansible and deploying applications.
* Hands-on experience on Ansible and Ansible Tower as a Configuration management tool, to automate repetitive tasks, quickly deploy critical applications, and proactively manage change.
* Expertise in Application Deployments and environment configuration using Ansible.
* Experience in writing playbooks and can manage thousands of servers by pushing the code from servers to different nodes in Ansible.
* Managing the configurations of multiple servers using Ansible.
* integrated the AWS CLI with Ansible to provision and manage AWS infrastructure.
* Expertise in Jenkins by installing, configuring, and maintaining for continuous integration (CI) and for Continuous Delivery (CD).
* Using Jenkins as an Amazon Web Services (AWS) Code Deploy plugin to deploy to AWS.
* Troubleshoot the build issue during the Jenkins build process.
* Installing, configuring, and administering Jenkins Continuous Integration tool on Linux machines along with adding/updating plugins such as SVN, Maven, and ANT.
* Having work experience in support of multi-platforms like UNIX, IOS, Linux, and Windows of production, test, and development servers.
* Involved in Linux Administration activities like troubleshooting of regular issues, configuration issues, applying patches, package management, and file system issues.
* Good Interpersonal Skills, team-working attitude, takes initiative, and very proactive in solving problems and providing the best solutions.
* Experience in Installing Firmware Upgrades, kernel patches, system configuration, and performance tuning on Unix/Linux systems.

**Technical Skills:**

|  |  |
| --- | --- |
| **Operating systems** | Linux (Red Hat 4/5/6/7, CENTOS & SUSE), Windows servers [2003, 2008, 2008 R2, 2012, 2012R2], Windows 2000, XP, Windows 7, Ubuntu 12/13/14. |
| **Cloud Platform** | Amazon Web Services (EC2, S3, EBS, RDS, ELB, Route 53, Auto scaling, Security Groups, CFT, SQS, SNS, Glacier, Lambda). |
| **Application Servers** | Web Logic Application Server 9.x, 10.x, Apache Tomcat 2.0.x, Red Hat, WebSphere 6.x/7.x/8.x |
| **CI CD tools** | Ant, Maven, Jenkins, Bamboo. |
| **Virtualization** | VMware Client, Windows Hyper-V, vSphere 5.x, Datacenters Virtualization, Virtual Box, KVM, Power VM |
| **Container tools** | Docker, Kubernetes. |
| **Infrastructure as Code tools** | Terraform. |
| **Configuration Management tools** | Chef, Puppet, Ansible, Salt |
| **Monitoring tools** | Splunk, Nagios,Cloud Watch, log Stash |
| **Scripting** | Python, Shell scripting, YAML, JSON |
| **Data base technologies** | Oracle, MySQL. |
| **Version control tool** | Git, SVN, Git Hub, Git Lab, Bitbucket, CVS. |
| **Logging tools** | Elastic search, Kibana |

**EDUCATION:**

**Master:** Master’s in computing / Wichita State University, Wichita, KS.

**Bachelor’s:** BTech in Computer Science and Engineering /JNTU Kakinada.

**Work Experience:**

**Sr. AWS/ DevOps Engineer**

**Carelon, Indianapolis, IN** **September 2021 – Till Date**

**Job Responsibilities:**

* Working on deploying, monitoring, and maintaining Amazon AWS cloud infrastructure consisting of multiple EC2 nodes across different environments.
* Created an S3 bucket for Storage and backup as well as Utilized AWS CLI to automate backups on ephemeral data.
* Created the security groups, Elastic Load Balancer and AMIs, and Auto scaling to design cost-effective, fault-tolerant, and highly available systems.
* Creating public and private subnets within the VPC and deploying EC2 instances based on the requirement.
* Designed and implemented Kubernetes clusters for scalable container orchestration, improving application deployment efficiency and reliability.
* Developed ansible playbooks to configure alerting strategies for machine configs, pv, PVC, and all of the openshift 4 components to notify support teams.
* Demonstrated expertise in container-based technologies, including Docker, Kubernetes, and OpenShift.
* Set up firewall rules and network load balancers for the Kubernetes OpenShift cluster's secure dmz segment.
* OpenShift Networks management for production, testing, and development.
* Applications should be containerized and moved to the OpenShift Container Platform and Kubernetes.
* Create an OpenShift cluster as part of a PaaS solution for TestNet, Development, DMZ, and live Production environments.
* Managed and maintained Kubernetes resources, including pods, services, and deployments, ensuring high availability and fault tolerance.
* Implemented container security measures within Kubernetes, utilizing features like PodSecurityPolicies and network policies.
* Configured and administered GitLab instances for version control and CI/CD pipelines, streamlining the development workflow.
* Automated build and deployment processes in GitLab CI/CD, integrating testing frameworks and ensuring code quality standards.
* Collaborated with development teams to optimize Git workflows and resolve version control challenges.
* Orchestrated complex application deployments on Kubernetes using Helm charts, enhancing consistency and repeatability.
* Customized Helm charts to accommodate project-specific requirements, enabling efficient application packaging and deployment.
* Managed end-to-end deployment processes for MuleSoft applications, ensuring seamless releases.
* Implemented efficient deployment strategies, reducing downtime and enhancing overall system reliability.
* Spearheaded the build and configuration of MuleSoft platforms, optimizing for performance and scalability.
* Customized Mule ESB configurations to meet specific project requirements.
* Established robust monitoring solutions for MuleSoft, promptly identifying and addressing performance issues.
* Conducted proactive ops management, ensuring the stability and health of MuleSoft environments.
* Conducted capacity planning to anticipate resource needs and ensure optimal MuleSoft platform performance.
* Collaborated with teams to implement performance improvements based on capacity planning insights.
* Led the planning and execution of MuleSoft upgrades and patching activities.
* Ensured the application of security patches and updates in a timely manner to mitigate potential vulnerabilities.
* Demonstrated deep expertise in Mule ESB and Anypoint Platform, implementing best practices for integration and API management.
* Ensured version control and rollback capabilities for Kubernetes applications through Helm releases.
* Automated routine tasks and system configurations through Bash scripting, improving operational efficiency.
* Developed and maintained a library of Bash scripts for system monitoring, log analysis, and error handling.
* Implemented end-to-end CI/CD automation using AWS DevOps services and Harness, streamlining software delivery pipelines.
* Utilized Infrastructure as Code (IaC) principles to automate infrastructure provisioning with hands-on experience in Terraform.
* Proficient in scripting languages such as PowerShell for Windows and Linux environments.
* Demonstrated strong understanding of both Windows and Linux operating systems.
* Applied Agile and DevSecOps methodologies to enhance collaboration, efficiency, and security throughout the software development lifecycle.
* Demonstrated proficiency with security tools such as Jenkins, GitLab CI/CD, SonarQube, OWASP ZAP, and other DevSecOps tools.
* Worked extensively on Linux/Unix platforms, demonstrating proficiency in basic commands and scripting for automation.
* Configured VPCs, Subnets, Security Groups, and NACLs to create secure network architectures.
* Hands-on experience with EC2 instances and ECS for container orchestration.
* Managed resources like S3, RDS, and DynamoDB for scalable and reliable storage solutions.
* Implemented serverless architecture using Lambdas and Lambda Triggers.
* Administered IAM for secure control and management of AWS resources.
* Configured Kinesis and SQS for efficient data streaming and messaging.
* Worked with Cloudfront, Cloudwatch, Cloudtrail, and Config for monitoring and governance.
* Integrated Bash scripts into CI/CD pipelines for automated testing and deployment processes.
* Created the AWS infrastructure on the cloud, from AWS CLI using terraform.
* Setting up IAM user roles with corresponding user and group policies using JSON
* Worked on creating the modules-driven AWS Infrastructure with Terraform. Created Infrastructure Git repositories for Terraform to launch the stacks.
* Proficient in CI/CD automation using AWS DevOps and Harness.
* Strong background in Infrastructure as Code (IaC) with hands-on experience in Terraform for automated infrastructure provisioning.
* Knowledge of other AWS services, including Cloudfront, Cloudwatch, Cloudtrail, and Config.
* Familiarity with Harness CI/CD tool and Ansible.
* Provided security and managed user access and quota using AWS Identity and Access Management (IAM), including creating new policies for user role management. Automated this process using Terraform.
* Expertise in API security, container security, and AWS cloud security, including S3, storage tiering, encryption, and pipeline management.
* Extensive experience in configuring and running code/binary scans using tools such as SonarQube and Veracode.
* Strong understanding of architecture principles and design methodologies, ensuring their effective application in system development.
* APIGEE is an API management platform that enables organizations to design, deploy, and scale APIs.
* APIGEE allows businesses to create APIs that connect with backend services and enable seamless communication between different software applications.
* The platform offers features like security, traffic management, and developer collaboration to ensure efficient API operations.
* APIGEE provides analytics and monitoring tools to track API performance, usage, and potential issues, allowing organizations to optimize their APIs continuously.
* Proven track record in maintaining and deploying highly available fault-tolerant systems at scale.
* Drive towards automating repetitive tasks through scripting (Bash, Python, Ruby), showcasing efficiency and process improvement.
* Implemented tools for continuous security integration, integrating security features into the development environment.
* Proficient in API gateway/load balancing and static analysis for comprehensive security measures.
* Hands-on experience with Docker containerization and clustering (Kubernetes/ECS), contributing to efficient deployment practices.
* In-depth knowledge of AWS services, including S3, storage tiering, encryption, and pipeline handling.
* Version control system expertise using Git, ensuring code collaboration and version management.
* Implemented CI/CD pipelines (e.g., Jenkins) to streamline development processes.
* Created Ansible playbooks to automatically install packages from a repository, to change the configuration of remotely configured machines.
* Created infrastructure as code (IaC) using Terraform to provision and manage cloud resources across AWS.
* Developed and maintained Python scripts for various DevOps tasks, such as deployment automation, configuration management, and infrastructure provisioning.
* Designed and implemented automation workflows using tools like Ansible, leveraging Python for custom modules and playbook development.
* Orchestrated complex tasks and processes, ensuring efficient and error-free execution through Python automation scripts.
* Designed and maintained modular and reusable Terraform configurations to ensure consistent and scalable infrastructure deployments.
* Orchestrated multi-tier application deployments, load balancers, and auto-scaling groups with Terraform.
* managed configuration, guaranteeing security and consistency across servers and environments using technologies like Ansible or Puppet.
* Set up alerting systems and implemented monitoring solutions utilizing software like Prometheus, Grafana, or Nagios to proactively discover and fix performance issues.
* Collaborated with development and operations teams to automate and streamline the application deployment process using Terraform and CI/CD pipelines.
* Actively participated in the development of Software Development Life Cycle (SDLC) processes, collaborating with cross-functional teams.
* Perform day-to-day operation and troubleshooting of EC2 instances.
* Used Jenkins with Ansible playbooks to run the deployment process.
* Used Ansible for configuration management of hosted Instances within AWS. Configuring and Networking of Virtual Private Cloud (VPC).
* Created Docker images using a Docker file, worked on Docker container snapshots, removed images, and managed Docker volumes.
* Created and managed a Docker deployment pipeline for custom application images in the cloud using Jenkins.
* Configure and ensure connection to the RDS database running on MySQL engines.
* Responsible for Plugin Management, User Management, regular incremental backups, and regular maintenance for recovery.
* Configure and deploy AWS Lambda service to run codes in response to events and automatically manage resources.
* Enabled AWS Cloud Watch to monitor major metrics like Network Packets, CPU utilization, and load balancer.
* Designed and implemented a data processing web service using Node.js to aggregate and calculate the uptime of thousands of servers from various data sources to Elasticsearch, displaying data in Kibana.
* Developed, constructed, and maintained CI/CD pipelines (Continuous Integration/Continuous Deployment), which automate the build, test, and deployment processes. This reduces manual error and speeds up release cycles.
* knowledgeable about utilizing Kubernetes and Docker to manage and deploy containerized applications, providing scalability and stability.
* Proficient in scripting languages such as Perl and Shell script, enabling automation of routine tasks and enhancing overall process efficiency.
* Conducted performance testing of Apache, Tomcat, and Nginx management services.

**Environment & Tools:** AWS (EC2, VPC, ELB, S3, RDS, IAM, Cloud Trail and Route 53), GIT, Bitbucket, VPC, Route53, EBS, SQL, ELB, Cloud watch, Cloud Formation, Ansible, Docker, Maven, Terraform, Helm, Kubernetes, AWS CLI, AWS Auto Scaling, Unix/Linux, Shell scripting.

**Sr. AWS DevOps Engineer**

**SEI Investment, Pennsylvania, PA January 2019 to August 2021**

**Job Responsibilities:**

* Worked on AWS cloud services like EC2, S3, VPC, IAM, RDS, ELB, and EBS for installing, configuring, and troubleshooting for server migration from physical to Amazon cloud.
* Utilized Cloud Watch to monitor resources such as EC2, CPU memory, Amazon RDS DB services, Dynamo DB tables, and EBS volumes; to set alarms for notification or automated actions; and to monitor logs for a better understanding and operation of the system.
* Strong understanding of infrastructure automation tools (AWS cloud formation, EBS).
* Configure and deploy AWS Lambda service to run codes in response to events and automatically manage resources.
* Enabled AWS Cloud Watch to monitor major metrics like Network Packets, CPU utilization, and load balancer.
* Designed and implemented a data processing web service using Node.js to aggregate and calculate the uptime of thousands of servers from various data sources to Elasticsearch, displaying data in Kibana.
* Used GIT Hub repository to run Jenkins job for Continuous Integration.
* Extensively involved in infrastructure as code, execution plans, resource graph, and change automation using Terraform. Managed AWS infrastructure as code using Terraform.
* Successfully implemented Kubernetes and OpenShift for container orchestration, enabling the seamless deployment of applications across clusters.
* Designed and maintained CI/CD pipelines for containerized applications on Kubernetes/OpenShift, leading to quicker releases and reduced downtime.
* Implemented security best practices for Kubernetes/OpenShift, including RBAC, network policies, and secrets management, to protect applications and data.
* Optimized MuleSoft workflows to enhance data flow and connectivity.
* Implemented and maintained robust CI/CD pipelines for MuleSoft projects, automating testing and deployment processes.
* Integrated CI/CD practices to achieve continuous integration and delivery.
* Proficient in scripting languages (e.g., Python, Bash) for automation of repetitive tasks in MuleSoft environments.
* Worked with cloud platforms (e.g., AWS, Azure) to deploy and manage MuleSoft solutions in a scalable and cost-effective manner.
* Utilized monitoring tools to track and analyze MuleSoft performance metrics.
* Implemented alerts and dashboards to provide real-time visibility into system health.
* Proficient in troubleshooting issues within Kubernetes/OpenShift environments, ensuring minimal application downtime and rapid issue resolution.
* Set up and configure monitoring tools (e.g., AWS CloudWatch) to track system performance and detect issues.
* Implement centralized logging for applications and infrastructure.
* Automate repetitive tasks and processes using scripting languages (e.g., Python, Shell) and automation tools.
* Implement auto-scaling and load-balancing solutions.
* Implement blue-green deployments and canary releases to minimize downtime during updates.
* Used Terraform scripts to automate instances for EC2 instances that were launched manually before.
* Increased efficiency by writing and reusing Terraform modules to create a full custom cloud infrastructure in minutes.
* Used Kubernetes for scaling the microservices deployment, created using Docker containers.
* Utilized Kubernetes for the runtime environment of the CI/CD system to build, test, and deploy.
* Experience working on several Docker components like Docker Engine, Hub, Machine, Compose, and Docker Registry.
* Experience working on Docker, creating Docker images, and handling multiple images primarily for middleware installations and domain configurations.
* Documented all build and release process-related items. Level one support for all the build and deploy issues encountered during the build process.
* Used Ansible server and workstation to manage and configure nodes.
* Developed build and deployment scripts using Maven as build tool, and integrated selenium in Jenkins, to perform the automated integration test.
* Created Docker images using the existing Maven-based images as the base image to compile and package the application.
* Configuring Jenkins for continuous integration (CI) for servers and Git lab for triggering automatic builds using Git in source code management.
* Experienced in the deployment of applications on Apache Web server, Nginx, and Application Servers such as Tomcat, and Oracle web logic server.
* Installing and configuring services such as Apache, DNS, SMTP, HTTPD, NTP, and DHCP and supporting them on Linux production servers.
* Comfortable and flexible with installing, updating, and configuring various flavors of UNIX and Windows.
* Involved in periodic archiving and storage of the source code for disaster recovery.

**Environment & Tools**: AWS (EC2, VPC, ELB, S3, RDS, IAM, Cloud Trail and Route 53), GIT, Bitbucket,

VPC, Route53, EBS, SQL, ELB, Cloud watch, Cloud Formation, Ansible, Docker, Maven, AWS CLI, AWS Auto Scaling, Unix/Linux, Shell scripting.

**AWS DevOps Engineer**

**3I Infotech, Hyderabad, IndiaJune 2016 to Dec 2017**

**Job Responsibilities:**

* Designed and implemented fully automated server build management, monitoring, and deployment solutions spanning multiple platforms, tools, and technologies including Jenkins Nodes/Agent, SSH, Amazon EC2, etc.

Build server deployment on Cloud (EC2) servers with the help of DevOps tools like Puppet.

* Created the naming strategy for branches and labels and involved a continuous integration system with the GIT version control repository and continued building as the check-ins came from the developer.
* Built Continuous Integration environment (Jenkins, Nexus) and Configuration management environment (puppet).
* Responsible for distributed applications across hybrid AWS and physical data centers
* Strengthening security by implementing and maintaining Network Address Translation in the company’s network
* Supported small to medium-sized projects using GIT.
* Worked on integrating GIT into the continuous Integration (CI) environment along with Anthill-Pro, and Jenkins.
* Developed a continuous deployment pipeline using Jenkins, shell scripts.
* Supporting Local System Administrators to troubleshoot Configuration Management and Network issues.
* Experience with containerization technologies such as Docker.
* Understanding of container orchestration tools like Kubernetes.
* Familiarity with configuration management tools like Ansible, Puppet, or Chef.
* Solid understanding of networking concepts in AWS, including VPCs, subnets, and security groups.
* Created and maintained continuous build and continuous integration environments in SCRUM and Agile projects.
* Importing the data in Splunk through inputs.conf, props. conf and transforms. conf.
* Automation/Simplification of Digital Guarding Process through SPLUNK.

**Environment:** GIT, Jira, Maven, ANT, Jenkins, Puppet, Unix Shell Scripting, Splunk, Anthill Pro, AWS, Nexus.

**Linux Administrator**

**Approsys, Hyderabad, India April 2014 to May 2016**

**Job Responsibilities:**

* Managed and administrated all UNIX servers, including Linux OS by applying relative patches and packages at regular maintenance periods using Red Hat Satellite server, YUM, and RPM tools.
* Managed Oracle Siebel applications on Linux Fedora.
* Maintained the performance by upgrading Linux (RHEL 5x, 6x, SUSE 10, 11, CENTOS 5, 6, OS and hardware maintenance like changing memory modules, and replacing disk drives.
* In-depth knowledge of NFS, DHCP, FTP, Auto Mount, DNS, LDAP related issues.
* Monitoring CPU, memory, physical disk, Hardware, and Software RAID, multipath, file systems, and network using the tools NAGIOS 4.0 monitoring.
* Performing failover and integrity tests on new servers before rolling out to production.
* Deployment and Configuration of application server Tomcat deploying Java and Web Applications.
* Automated daily tasks by writing Shell Scripts, documenting the changes that happen in the environment and in each server, analyzing the error logs, and analyzing the User logs.
* Planned, scheduled, and Implemented OS patches on Linux boxes as a part of proactive maintenance.
* Identify, troubleshoot, and resolve problems with the OS build failures.
* Used Chef for managing server application servers such as Apache, MySQL, and Tomcat.
* Installation, configuration, and customization of services Send Mail, Apache, and FTP servers to meet the user's needs and requirements.
* Worked on TCP/IP protocols and created on-premises static IPs.
* Performing kernel and database configuration optimization such that it limits I/O resource utilization on disks.

**Environment:** Red Hat Linux 5.x, 6.x, Autosys, NFS, DHCP, FTP, Auto Mount, DNS, LDAP CentOS, VMware vSphere 4.0, VMware ESX 3.5.0, GIT, Shell Scripting, MySQL, Tomcat, Nagios.