**MANISH CHINTHALAPALLY**

**Sr. DevOps Engineer**

**(469)-844-1901**

[**manishchinthalapally@gmail.com**](mailto:manishchinthalapally@gmail.com)

**LikedIn URL: https://www.linkedin.com/in/manish-chinthalapally-741a18191/**

**OBJECTIVE:**

I am a Senior DevOps Engineer with 10+ years of IT experience, specializing in DevOps. Proficient in AWS, Azure, and Linux administration. Skilled in team integration, infrastructure optimization, and CI/CD pipeline architecture. Committed to driving innovation and efficiency in dynamic tech environments.

**PROFESSIONAL SUMMARY:**

As a Sr. DevOps Engineer major focus on Azure, micro, Continuous Integration, Continuous Deployment, Configuration Management, Build/release Engineer, and Linux Systems Administrator with expertise in Private, Public and Hybrid Cloud platforms along with using various tools like **Docker**, **Kubernetes**, **Ansible**, **Puppet**, **Jenkins**, **Git**, **Maven** and experienced in Linux system Administrator.

* Over 8 years of professional experience in Build & Release, DevOps Engineering, and Site Reliability Engineering, specializing in Azure, coupled with mentoring capabilities and a solid grasp of foundational principles.
* Hands-on Experience Administrating Microsoft Azure Services like Azure App Services, Azure SQL Database, Azure AD, Azure Blob storage, Azure Functions, Virtual machines, Azure Fabric controller, Azure Data Factory, Azure web applications, Azure Service Bus, and Notification hub.
* Good in container technologies such as **Docker** and **Kubernetes**.
* Experi53 and utilizing Amazon AWS services like **EC2, S3, VPC, EBS, RDS, Elastic Beanstalk, DynamoDB, NFS Storage, Route 53**, and implementing Elastic Load Balancers for high-availability setups.
* Experienced in automating cloud infrastructure provisioning using **Terraform** by developing modular, reusable Terraform code for **AWS, Azure**, ensuring consistency and compliance in cloud deployments.
* Seasoned in Configuring Azure Automation Desired State Configuration (**DSC**) configuration management to assign permissions through Role-Based Access Control (**RBAC**), assign nodes to proper automation accounts and DSC configurations, to get alerted on any changes made to nodes and their configuration.
* Expertise in Designing and implementing Continuous Integration (**CI**) throughout many environments utilizing Azure DevOps tools to provide an agile development process that is automated and repeatable, allowing teams to safely deploy code several times per day while ensuring Azure Kubernetes Services (**AKS**) are supported.
* Well versed in using Azure Infrastructure as a Service (**IaaS**), Provisioning VMs, Virtual Networks, Deploying Web Apps, Microsoft **SQL** Server, using **ARM** Templates, and Azure DevOps CI/CD pipelines.
* Expertise in configuring the Monitoring and Alerting tools according to the requirements like **Prometheus** and **Grafana**, setting up alerts, and deploying multiple dashboards for individual applications in **Kubernetes**.
* Executed server, database, and application migrations from on-premises environments to AWS and Azure Cloud Platforms.
* Experience in building **Docker** images using **GitLab**-ci build automation runner.
* Experience in Setting up the build and deployment automation for **Terraform** scripts using **Jenkins**.
* Good exposure in managing various **LINUX** servers clustered environments.
* Led full-scale DevOps projects, skillfully blending automation, cloud infrastructure, and **CI/CD** practices to streamline software development processes.
* Provided training and mentorship to junior team members on **SRE** principles, practices, and tools, fostering their professional growth and development within the organization.
* Designed and implemented CI/CD pipelines on AWS Cloud Pipelines to automate the build, test, and deployment processes of applications.

**EDUCATION DETAILS:**

* Bachelor’s degree in computer science from JNTUK during 2007 to 2011.
* Master’s degree in computer and information science from Southern Arkansas University and graduated in December, 2012.

**CERTIFICATIONS:**

* Microsoft Certified Azure Administrator.
* AWS Developer – Associate

**TECHNICAL SKILLS:**

|  |  |
| --- | --- |
| **Cloud Environments** | Microsoft Azure, Amazon Web Services (AWS), GCP |
| **Configuration Management tools** | Ansible, Ansible Tower, Chef, Puppet, AWS Config |
| **Build Tools** | ANT, Maven, Gradle, SBT, MS Build, |
| **CI/CD Tools** | Jenkins, Bamboo, Git Hub Actions, Azure DevOps, Circle CI |
| **Monitoring Tools** | Splunk, Nagios, Cloud Watch, ELK, Grafana, Prometheus, Azure Monitoring |
| **Container Tools** | Kubernetes, Docker |
| **Scripting/Programming Languages** | Python, Shell (PowerShell/Bash), Ruby, YAML, JSON, Groovy, JavaScript, C/C++,  Shell Scripting |
| **Version Control Tools** | GIT, Subversion (SVN), Bit Bucket, Git Lab, TFS, Azure Repos |
| **Operating Systems** | UNIX, Linux, Windows Server, Ubuntu, Red Hat, CentOS |
| **Databases** | SQL Server, MYSQL, Oracle, NoSQL, MongoDB, Dynamo DB, Cassandra, IBM DB2, PostgreSQL, AWS Aurora |
| **Project Management Tools** | JIRA, Rally |
| **Testing / Code Quality** | Selenium, SonarQube |
| **Web/Application Servers** | Apache Tomcat, NGINX, httpd, JBOSS 4.x/5.x, IIS, WebSphere 4.0/5.0 |
| **Virtualization Tools** | Oracle Virtual Box, VMWare, vSphere, Vagrant, Windows Hyper-V, Power VM |
| **Infrastructure** | Terraform, Cloud formations, ARM |
| **AWS/Azure Services** | IAM, VPC, Route53, EC2, S3, CodeBuild, CodeDeploy, Redshift, RDS, CloudWatch, CloudFormation, Load Balancer, App Services, Key vault, function app, Blob storage, Azure Active Directory (Azure AD), Service Bus, Azure Container Registry (ACR) and Azure Kubernetes Services (AKS), Azure SQL, Azure Cosmos DB. |
| **Artifactory** | Jfrog, AzureArtifacts (ADO), ECR, Nexus |
| **SDLC** | Agile, Waterfall, Kanban |
| **Containerization/Orchestration Tools** | Docker, Kubernetes |

**WORK EXPERIENCE:**

**Client: Analytic Partners, New York JULY 2022 - PRESENT**

**Role: Sr. DevOps Engineer**

**Responsibilities:**

* As an Sr. Cloud/DevOps Engineer, I delivered **DevOps** pipelines that apply automation to the software development process and improve the efficiency, security, stability, and quality of applications being delivered by project and product teams.
* Leveraged Jenkins to orchestrate Python execution within secure Docker containers for enhanced financial data deployment on Databricks.
* Worked on **AWS** Cloud platform and its features which includes EC2, VPC, **S3**, AMI, SNS, RDS, **Cloud Watch**, Auto scaling, Cloud Front, IAM, S3 for Configuring and managing **IaaS**.
* Wrote Cloud Formation scripts to deploy different **AWS** Infrastructure components related to the respective services like created custom sized **VPC**, subnets, **NAT** to ensure successful deployment of Web applications and managed various resources and data elements defined in the scripts.
* Working with build Servers using **AWS** Importing volumes, Launching **EC2**, creating security groups, load balancers.
* Built the AWS Infrastructure using VPC, EC2, S3**, Route 53**, **EBS**, Security Group, Auto Scaling, and RDS in **Cloud Formation.**
* Configured the build jobs in **Jenkins** to automate the builds of **IOS**, Android and Java applications using **Groovy** and **Maven**

and deployed the successfully built artifacts into the Artifactory.

* Orchestrated Kubernetes clusters and managed various components such as pods, replication controllers, and deployments through **YAML** configuration files.
* Established robust application monitoring using **Kibana**, **Logstash**, **Prometheus**, and **Grafana**.
* Successfully deployed Docker-based applications on Azure Kubernetes Service (**AKS**) and Azure, ensuring seamless operation and scalability.
* Integrated AWS **Lambda** with other AWS services such as Amazon S3, Amazon SQS, and Amazon SNS to automate workflows, trigger functions, and orchestrate data flows between services.
* Set up and configure **Argo CD** on **Kubernetes clusters**, ensuring proper integration with version control systems like **Gi**t.
* Identified akibanand resolved issues using **Azure Monitor**, Log Analytics workspace, and Kubernetes logs analysis.
* Manage continuous delivery pipelines within **Argo CD**, including defining workflows, triggers, and automation.
* Utilize **Jira** for project planning, tracking, and management, including creating and managing tasks, epics, stories, and sprints.
* Implemented networking components like Subnets, Network Security Groups (**NSGs**), Internal Load Balancer (**ILB**), and App Service Environments (**ASE**) for efficient application deployment.
* Automated secrets management during application deployment by configuring **Azure Key Vaults**.
* Managed a diverse range of CI/CD pipelines for Java and .Net applications, utilizing **Jenkins**, **Azure DevOps**, and **AWS Cloud pipelines**.
* Integrating **Azure AD** with on-premises Active Directory environments.
* Set up monitoring and alerting for **AWS Aurora** databases using **AWS CloudWatch**, enabling proactive monitoring of database metrics, performance indicators, and alarms to detect and respond to potential issues.
* Configured Jenkins for seamless CI/CD processes, integrating with **Maven, Gradle**, **SonarQube**, and **Couchbase** for comprehensive code analysis.
* Utilized **Docker** and **Kubernetes** to optimize resource utilization and facilitate the deployment and scaling of microservices in the AWS environment.
* Developed custom pipeline stages and actions in AWS Cloud Pipelines using AWS Code Build for building artifacts and AWS Code Deploy for deploying applications.
* Provisioned the highly available servers using **Terraform** and cloud formation and wrote new plugins to support new functionality in Terraform.
* Utilized **Apache Kafka** as a distributed event streaming platform for building real-time data pipelines and stream processing applications.
* Played an active role in monitoring server performance, network traffic to reduce performance bottleneck by enhanced performance monitoring through Prometheus, **Grafana**, ELK/EFK and **Splunk**/SolarWinds/**Dynatrace**.
* Analyzing cloud usage data and cost metrics to optimize cloud resource utilization and costs with **Datadog**.
* Used **Groovy** scripts to query and process metrics from monitoring systems like **Prometheus** or to interact with logging tools like **ELK** stack.
* Built from scratch a development workflow, continuous integration and deployment pipeline using AWS **Code Commit**, Code Build, **Code Pipeline**, Terraform, **Docker**.
* Setup **Bugzilla** on Linux VM to keep track of bugs in deployment cycle and environment issues.
* Expertise in isolating and packaging application components in containers and managing horizontal scaling.
* Utilized **Terraform** console for testing, debugging, and addressing deployment and orchestration issues.
* Worked on **CI/CD** pipelines using **Jenkins** to build, test, deploy micro services containers on **Kubernetes** clusters using

**Ansible** on DEV, UAT, PROD environment.

**Environment**: Jenkins, Nexus, Checkmarks, SonarQube, Kubernetes, Kibana, Logstash, Prometheus, Grafana, Azure Monitor, Log Analytics, Virtual Networks (VNETs), DCS Hub, Subnets, Network Security Groups (NSGs), Internal Load Balancer (ILB), App Service Environments (ASE), Azure Key Vaults, CI/CD pipelines, Java, .Net, Azure DevOps, AWS CloudFormation, AWS CodePipeline, AWS CodeBuild, CloudWatch, Docker, Terraform, Bugzilla, Linux VM, RBAC, Helm, Backstage, Blob Storage.

**Client: Nokia, Irving, Texas APRL 2020 – JUNE 2022**

**Role: Azure DevOps Engineer**

**Responsibilities:**

* As a Sr. Azure DevOps Engineer, I have deployed and maintained web applications on **Azure** cloud technologies, utilizing Azure App Service for scalable web hosting. Converted a .Net Legacy to Core application and deployed it in the Microsoft Azure Cloud Service Project as part of cloud deployment.
* Dealt with Azure **IaaS & PaaS** - Virtual Machines, Cloud Services, Resource Groups, and Express Route, VPN, Load Balancing, and Application gateways, Auto-scaling, Traffic Manager, Azure Web Apps, Web Roles, Worker Roles, Azure SQL, and Azure Storage.
* Deploying and managing Kubernetes clusters using **RKE** across different environments like development, staging, production.
* Implement security best practices such as encryption, network policies, and authentication mechanisms to secure **Argo CD** deployments and data.
* Implementing infrastructure configurations using tools like **Terraform** or **Ansible** to automate **RKE** cluster setup.
* Map **Jira** project, issues, and configurations to corresponding Azure DevOps artifacts such as projects, work items, and boards to ensure data consistency and integrity during migration.
* Integrate **Argo CD** with existing **CI/CD** pipelines and tools like Jenkins, Travis CI, or CircleCI to automate the deployment process.
* Migrate test cases and test plans from **Jira** to **Azure DevOps** Test Plans, ensuring traceability between requirements, test cases, and test results for comprehensive test coverage.
* Wrote **Splunk** queries to extract the logs from the servers whenever there is an issue in the deployments.
* Configured **Nginx** as a load balancer to maintain the incoming traffic and configured mutual authentication using it.
* Experience in using Atlassian tools such as **JIRA**, Confluence, **bitbucket**, **bamboo** and **octopus**.
* Used bit **bucket and GitHub** as a source code management tool and integrated it with Jenkins.
* Written Python scripts to automate the manual stuff like Patching, Log rotation and daily jobs.
* Managed servers on the **Microsoft Azure** Platform, making use of Azure Virtual Machines instances. Implemented **Azure Automation** and Azure Policy to enforce compliance and governance policies across the infrastructure.
* Implemented Azure Availability Sets for high availability and conducted hardening of the **VMs**, along with disk encryption using **Azure Key Vault**.
* Proficient in creating and merging feature branches, hotfixes, release branches, and support branches in adherence to **Gitflow.**
* Experienced in version control systems such as **SVN** and **GIT**, ensuring code integrity, traceability, and collaboration across distributed teams.
* Experience with **Python** libraries and frameworks such as boto3 for interacting with cloud **APIs**, automating resource provisioning, and performing infrastructure orchestration.
* Familiarity with containerization technologies such as **Docker** and container orchestration platforms like Kubernetes, integrating them into Terraform workflows for deploying and managing containerized applications.
* Strong understanding of networking concepts, including TCP/IP, DNS, DHCP, VPN, firewalls, and load balancers, for designing and troubleshooting complex network architectures in Linux environments.
* Proficient in using **Git** version control for managing Terraform code repositories, implementing branching strategies, and collaborating effectively with development teams using Git workflows.
* Knowledge of cloud-native services and APIs for infrastructure provisioning and management, including AWS CloudFormation, Azure Resource Manager (ARM), and complementing Terraform workflows.
* Worked extensively on building and installing servers, creating multiple resources in Azure through Azure Resource Manager Templates (**ARM**), and provisioning them using **Terraform** templates. Also, worked on virtual networks, Azure Custom security, endpoint security, and firewall.
* Experienced in using **JIRA** for ticketing and defect tracking, configuring various workflows, customizations, and plugins for JIRA bug/issue tracker, and integrating Jenkins with JIRA, **GitHub**, **Maven**.
* Configured and managed **Azure Key Vault** to securely store and manage cryptographic keys, secrets, and certificates, ensuring the protection of sensitive data.
* Configured **AKS** clusters with **Azure CLI** to optimize performance, reliability, and security, leveraging features such as auto- scaling, load balancing, and network policies. Participated in code reviews for Kubernetes **YAML** definitions as part of team standards.
* Experienced in designing, implementing, and maintaining infrastructure as code (**IaC**) solutions using **Terraform**, automating the provisioning and management of cloud resources across multiple platforms.
* Expertise in Apache **Kafka** cluster design, deployment, and management for real-time data streaming applications.
* Skilled in setting up and automating CI/CD pipelines using Jenkins, GitLab CI/CD, CircleCI, integrating Kafka and APIM workflows.
* Knowledgeable in migrating and containerizing legacy and monolithic applications to **Kubernetes** on **AKS**.
* Strong understanding of **Kubernetes** design patterns for building scalable and resilient applications.
* Experienced in provisioning and managing infrastructure using **Terraform**, CloudFormation for **Kafka** and APIM environments.
* Familiar with Docker and **Kubernetes** for containerization and orchestration of Kafka and **APIM** deployments.
* Knowledgeable in monitoring solutions like Prometheus, Grafana, ELK stack, Splunk for Kafka and APIM clusters.
* Strong understanding of security best practices in Kafka and **APIM** environments, implementing authentication and encryption.

**Environment**: Azure, Terraform, Jenkins, Bamboo, Octopus, Puppet, WSO2, MongoDB, Python, Splunk, Nginx, Atlassian tools (JIRA, Confluence, Bitbucket), New Relic, Maven, Eclipse, Docker, Azure DevOps (formerly Visual Studio Team Services or VSTS), GitHub, Subversion (SVN), VMware vSphere, Red Hat Enterprise Linux (RHEL), SSL, TLS, Python, Linux.

**Client: Liberty Mutual, Boston, MA NOV 2018 – MAR 2020**

**Role: AWS/ DevOps Engineer**

**Responsibilities:**

* As an AWS Cloud/ DevOps Engineer, I delivered **DevOps** pipelines that apply automation to the software development process and improve the efficiency, security, stability, and quality of applications being delivered by project and product teams.
* Selecting and configuring the appropriate platform for the application (**Java, .NET, Node.js**) within **Elastic Beanstalk**.
* Drive continuous improvement initiatives for **GitHub** administration processes, tools, and policies based on feedback, metrics, and industry best practices.
* Established development, testing, and production environments for diverse applications on **AWS**, employing **Docker**, **Bash**, **Chef**, and provisioning **Kubernetes** clusters on **EC2** instances through **CloudFormation** and **Terraform**.
* Integrated **AWS** infrastructure using CloudFormation to enhance security by encrypting EBS volumes and automating routine operations with **Lambda** functions.
* Conducted deployments on **EKS**, **AWS**'s managed **Kubernetes** service, ensuring seamless orchestration.
* Managed **AWS** permissions using **IAM** roles, groups, and policies, customizing access for developers to specific AWS services and inter-service functionalities.
* Resolved infrastructure and pipeline issues involving **Docker**, **Ansible**, **Jenkins**, **Artifactory**, and **GitHub**, ensuring smooth operation of the deployment ecosystem.
* Automated **SonarQube** scans for multiple projects using configuration as code.
* Built **API** backends with **AWS Lambda** and API Gateway, creating serverless **RESTful APIs** or **GraphQL** endpoints to expose business logic and integrate with frontend applications or third-party services.
* Integrated Backstage with third-party tools and services such as **GitLab**, **Jira**, and **Jenkins** to create a seamless development and delivery ecosystem, enabling end-to-end visibility and automation.
* Integrating **Elastic Beanstalk** with CI/CD pipelines for automated deployment of application updates, ensuring rapid and reliable software delivery.
* Worked on **AWS** Cloud platform and its features which includes EC2, VPC, **S3**, AMI, SNS, RDS, **Cloud Watch**, Auto scaling, Cloud Front, IAM, S3 for Configuring and managing **IaaS**.
* Proficient in designing high availability and disaster recovery strategies for **Kafka** and APIM deployments.
* Configured the build jobs in **Jenkins** to automate the builds of **IOS**, Android and Java applications using **Groovy** and **Maven**
* and deployed the successfully built artifacts into the Artifactory.
* Played an active role in monitoring server performance, network traffic to reduce performance bottleneck by enhanced performance monitoring through Prometheus, **Grafana**, ELK/EFK and **Splunk**/SolarWinds/**Dynatrace**.
* Skilled in performance optimization of Kafka and **APIM** configurations for improved throughput and latency.
* Strong scripting skills in **Python**, Bash, PowerShell for automation of routine tasks related to Kafka and **APIM**.
* Experienced in capacity planning and scaling **Kafka** and APIM clusters based on workload requirements.
* Skilled in incident management processes and conducting root cause analysis for **Kafka** and APIM services.
* Implemented infrastructure as code (**IaC**) best practices, including code reviews, testing methodologies, and version control strategies, to ensure the reliability and maintainability of Terraform configurations.
* Worked on **CI/CD** pipelines using **Jenkins** to build, test, deploy micro services containers on **Kubernetes** clusters using

**Ansible** on DEV, UAT, PROD environment.

* Analyzing cloud usage data and cost metrics to optimize cloud resource utilization and costs with **Datadog**.
* Used **Groovy** scripts to query and process metrics from monitoring systems like **Prometheus** or to interact with logging tools like **ELK** stack.
* Built from scratch a development workflow, continuous integration and deployment pipeline using AWS **Code Commit**, Code Build, **Code Pipeline**, Terraform, **Docker**.
* In-depth understanding in Web Application Deployments, maintenance, and troubleshooting/monitoring.
* Working with build Servers using **AWS** Importing volumes, Launching **EC2**, creating security groups, load balancers.
* Built the AWS Infrastructure using VPC, EC2, S3**, Route 53**, **EBS**, Security Group, Auto Scaling, and RDS in **Cloud Formation**.

**Environment:** Chef, AWS (Amazon Web Services), Test Kitchen, ChefSpec, Docker, Jenkins, GitHub, SonarQube, Nagios, Python, Apache Web Server, WebSphere Application Server, Red Hat Linux, Sun Solaris, EC2, S3, IAM (Identity and Access Management), CodePipeline, Knife, Docker-Maven plugin, Maven, Git, Splunk, Ant, POM (Project Object Model), Nagios.

**Client: Zurich Insurance, Dallas, Texas AUG 2016 – OCT 2018**

**Role: SRE/ Build and Release Engineer**

**Responsibilities:**

* As a **SRE**, I have orchestrated the **CI/CD** process employing tools such as **GIT**, Nexus, SonarQube, Terraform, **Ansible** and

**Jenkins** for job creation.

* Established a robust CI/CD pipeline utilizing **Jenkins** and **Node CLI**, ensuring efficient builds and deployments.
* Orchestrated the containerization and deployment of multiple microservices onto **Kubernetes**, including modules for Purchase Order, User Profile, Payments, and Reports, to facilitate scalability and streamline deployment processes.
* Designed and implemented Azure Virtual Networks (**VNets**), subnets, and network configurations, along with DHCP blocks, DNS settings, and security policies.
* Maintain project documentation within **GitHub** repositories, including README files, contribution guidelines, and project wikis, ensuring they are up-to-date and comprehensive.
* Integrated **SVN** with **CI/CD** pipelines.
* Engineered BGP routing for ExpressRoute and site-to-site **VPN**, establishing connections between on-premises data centers and Azure.
* Conducted regular code quality analysis using **SonarQube** to identify and address code smells, bugs, vulnerabilities, and security hotspots in the codebase.
* Collaborated closely with cross-functional teams including Development, Operations, and Product Management to align SRE initiatives with business goals and priorities.
* Architected an event-driven architecture using **Apache Kafka** for real-time data processing across distributed systems.
* Managed **Azure VMs**, virtual networks, and crafted **ARM** templates/**Terraform** scripts to automate deployment processes.
* Automated tasks in private cloud environments using Ansible, leveraging **YAML** scripts to enhance automation.
* Utilized Terraform to provision services such as **AKS**, **ACR**, **VNET**, and **VM**s in Azure.
* Automated continuous deployment using **Ansible** playbooks and roles written in **YAML**.
* Developed **Docker** images for deploying microservices on Azure Kubernetes Services (**AKS**).
* Managed **Kubernetes cluster** creation, deployments, services, **RBAC**, and Ingress configurations.
* Implemented a structured **GIT** branching strategy for code management.
* Executed on-premises data migration to **Azure SQL**/**Cosmos DB** using **Azure Data Factory**.
* Managed Azure infrastructure using **Terraform**, Blueprints, Landing Zone, and Azure Stack services, integrating Azure Automation Scripts for application deployment.
* Created and maintained **Docker** images, managed **Docker Hub** repositories, and wrote **Docker** files.
* Executed Maven builds, managed **Elk**, Splunk, **Nagios**, Nginx, **Tomcat**, and created Docker images deployed in clusters.
* Architected, developed, and deployed microservices to **Kubernetes** and Docker Swarm through CI/CD automation.
* Monitoring resource and application monitoring on **AWS Cloud Watch**, establishing alarms for **EBS**, EC2, ELB, RDS, and **S3**

metrics, with configured notifications for events.

* Worked on configuration management tool using **Chef**, for automating server configurations and deployments across

Amazon **EC2** instances. Created custom dashboards in **Datadog** with aggregated metrics and automated emails with Datadog

reports using the Datadog **REST API**/web hooks.

* Automated infrastructure provisioning and configuration management utilizing **Terraform**, Ansible, and **Git**.
* Implemented an observability stack, including monitoring, logging, and analytics using ELK, **Splunk**, and Nagios.
* Installed and configured **SSH** server on **RedHat Linux** environments to ensure secure access.
* Utilized Prometheus, **Grafana**, and Alert manager for end-to-end monitoring of infrastructure, applications, and services.
* Designed custom Grafana dashboards to track crucial application and business **KPIs** for early problem detection.
* Established and administered **Kubernetes** clusters, deploying applications using **YAML**.
* Developed end-to-end **CI/CD** pipelines using Jenkins, **Groovy**, and plugins.
* Proficient in infrastructure as code tools such as Terraform, CloudFormation, and Ansible.
* Expertise in Nginx, **JBoss**, **MongoDB**, and microservices architecture.
* Demonstrated proficiency in **Python** for analytics, automation, and microservices development.

**Environment:** AzureDevOps, Terraform, AKS, Docker, Ansible, Prometheus, Grafana, Bash, Python, Jira, Bitbucket, Apache Tomcat, ARM, CRON, YAML, JSON, Splunk, AWS Elastic Beanstalk, AWS Lambda functions, AWS CodeBuild and CodeDeploy.

**Client: North Carolina Department of Insurance, Raleigh, NC OCT 2013 – JULY 2016**

**Role: Linux Administrator**

**Responsibilities:**

* Knowledge on **SaaS**, **PaaS**, and **IaaS** concepts of cloud computing architecture.
* Responsible for creating and managing a **Docker** deployment pipeline for custom application images in the cloud using Jenkins.
* Implemented and maintained the branching and build/release strategies utilizing **GIT** Administration of **Jenkins** server- Includes Setup of **Jenkins**, Configure Nightly builds, and parameterized builds.
* Wrote **Python** scripts for automating the deployments of applications.
* Used **Git** as source code management tool and integrated it with **Jenkins** for **CI/CD** pipeline, code quality tracking and user management with build tool **Maven**.
* Expertise in Installation and configuration of automated tool like **Puppet** which includes Puppet Master, Agent Nodes and an administration control workstation and tools like Hiera, Mcollective & **Puppet** Console. Used Bamboo and Octopus as **CI** (Continuous Integration) and **CD** (Continuous Deployment).
* Experience in using Atlassian tools such as **JIRA** (for work tracking) and Confluence as central repository for documentation.
* Troubleshooting and performance tuning issues with applications like Oracle 10.x, 11.x
* Installed, Upgraded, and configured **Solaris**, Red Hat **Linux** RHEL 3.x, 4.x and **Windows** server.
* Configured and administered **DNS** (Bind on Linux), **Web** (Apache), **Mail** (Exchange 2003), DHCP and **FTP** Servers.
* Upgraded Memory, CPU and Disk size on the existing **VMware** Sessions on ESX Servers.
* Aided in **TCP**/IP Networking troubleshooting and **Linux** Network Administration.
* Implemented and maintained server virtualization using VMware and **Oracle** Virtual Manager.
* Created virtual machine using **Oracle** Virtual Manager, creating server pool, **VM** cloning. Conceived, Designed, Installed and Implemented **CI/CD** automation system.
* Created users, Groups, and setup their home directories and appropriate access restrictions to software in the **Active Directory**

environment and **NIS** domains.

**Environment:** AWS CloudFormation, Amazon S3, Docker, Jenkins, Git, Maven, Puppet, Hiera, Mcollective, Puppet Console, Bamboo, Octopus, Python, Ruby, Oracle, WebLogic, JIRA, Confluence.

Top of Form

**Client: TSYS, Atlanta, GA FEB 2013 – AUG 2013**

**Role: Linux System Administrator**

**Responsibilities:**

* Installation, Configuration & Upgrade of Linux, Solaris and **HP-UX** Operating System.
* Excellent knowledge in Installation, Configuration and file system and RAID volume management through VXVM and Solaris Volume Manager **(SVM)** in Solaris and **LVM** in Linux and HP-UX.
* Excellent knowledge of Linux/Unix kernel tuning, building customized kernels.
* Experience in installing, configuring and maintaining WebLogic Application Server and WebSphere Server with **java** Application tools on **Linux** and **UNIX** servers’ environment.
* Created repositories, branches, and tags aligned with specified structures and managed permissions on **GIT** branches.
* Installation, integration and management of data backup/recovery solutions. Management and configuration of **VMWare** virtual machines running **Oracle**/Sun **Solaris** X86/64, **Red Hat** Enterprise.
* Created Zettabyte file system (ZFS) in Solaris 10. Created pools, snapshots and clones.
* Worked on maintaining DNS & NTP, MySQL database servers.
* Installed and configured system network monitoring tool used **Nagios** and troubleshooted virtual machine issues.
* Involved in design, configuration, installation, implementation, management, maintenance and support for the Corporate **Linux** servers **RHEL** 4.x, 5.x, SLES 9, **CENTOS** 5.x.
* Developed and maintained multiple build definitions and published profiles for automated builds in an **Agile**/**SCRUM**
* environment.
* Installed, administered, and configured **Jenkins**/**Bamboo** Continuous Integration tools.
* Leveraged **Ant** and **Maven** for daily and weekly software builds.
* Assisted developers and project teams in setting views and environments.
* Developed and deployed effective Software Configuration Management (**SCM**) processes and procedures.
* Installed, configured, and administered Web Servers like Apache **HTTP** Server, Apache **Tomcat** and Sun One Web Server

for WebLogic plug-ins.Installed and configured **SSH** server on **RedHat**/**CentOS** Linux environments to ensure secure access.

**Environment**: Centos, VMWare, GIT, Jenkins, Ant, Maven, Sun Solaris 8/9, Red Hat Linux 8,9.0, Samba, Oracle 8/9i/RAC, Apache 2.x/3.x, Nagios, Linux, LVM.