

**ANIL CHAVA**

**Sr. Cloud/Devops Engineer**

**Email: [anil.chava21@outlook.com](mailto:anil.chava21@outlook.com)**

**Phone: +1(412) 365-8486**



**MCID: 990909139**

### **PROFESSIONALSUMMARY**

- Overall 7+ years of extensive experience in “Automating, configuring and deploying” instances on cloud environments and experience in the areas of “DevOps, CI/CD Pipeline, Build and release management and Linux/Windows Administration”.
- Expertise in Architecting and Implementing Azure Service Offering, such as Azure cloud services, Azure storage, IIS, **Azure Active Directory (AD)**, **Azure Resource Manager (ARM)**, **Blob Storage**, Azure VMs, SQL Database, Azure Functions, Azure Service Fabric, Azure Monitor, and Azure Service Bus.
- Hands on experience on Backup and restore **Azure services** and in Design and **configure Azure Virtual Networks (VNETs)**, subnets, Azure network settings, DHCP address blocks, DNS settings, security policies and routing, Azure cloud services, Blob storage, Active directory, Azure Service Bus, Cosmos DB.
- Experience in migrating on premise to Windows “**Azure**” using Azure Site Recovery and Azure backups and having good knowledge on Azure Fabric, Microservices in Azure
- Experience in Private Cloud and Hybrid cloud configurations, patterns, and practices in Windows Azure and SQL Azure and in Azure web and database deployments
- Experienced in AWS Cloud platform and its features which includes **EC2, S3, VPC, EBS, ELB, RDS,RDB, Elastic Beanstalk, DynamoDB, RedShift** and Route 53,EC2 AMI, EBS Cloud watch, AWS Config, and Auto-scaling, IAM
- Experienced in Amazon EC2 setting up instances and setting up security groups and databases in AWS using S3 bucket and configuring instance backups to S3 bucket
- Expertise in building **CI/CD** on AWS environment using AWS Code Commit, Code Build, Code Deploy and Code Pipeline and experience in using **AWS CloudFormation**, API Gateway, and **AWS Lambda** in automation and securing the infrastructure on AWS
- Implemented multiple CI/CD pipelines as part of DevOps role for on-premises and cloud-based software using Jenkins, Chef and AWS.
- Excellent working experience in the **software development life cycle (SDLC)**, project management, configuration and release management plan
- Automate provisioning and repetitive tasks using **Terraform** and Python, **Docker** container, Service Orchestration
- Supported API development team in creating “**Ansible playbooks**” for deploying, integrating and monitoring their continuous integration environment
- Participated in configuring and monitoring distributed and multiple platform servers using **Puppet**. Used Puppet server and workstation to manage and configure nodes
- Extensively worked with Jenkins as Continuous Integration / Continuous Deployment Tool and strong experience with integration of Ant and Maven Build Frameworks.
- Research and implemented new tools such as **Kubernetes** with **Docker** to assist with auto-scaling, continuous integration, rolling updates with no downtime
- Experience writing **Ruby, Python, Shell (Bash), Perl and Batch/PowerShell** scripts to automate the deployments
- Deployed and configured Chef Server and Chef Solo including bootstrapping of chef client nodes for provisioning; Created roles, cookbooks, recipes, and data bags for server configuration, deployment, and app stack build outs
- Managed On-site OS/Applications/Services/Packages using Chef as well as AWS **for EC2/S3/Route53 and ELB** with Chef Cookbooks.
- Creating and Managing Virtual Machines in Windows Azure and setting up communication with the help of endpoints
- Creating fully automated CI build and deployment infrastructure and processes for multiple projects using Docker, Jenkins, **VMWare**, RHEL, Open Shift, **Kubernetes, Artifactory, Ansible & GIT** Lab

- Expertise in scripting for automation, and monitoring using Shell, Bash, **PowerShell**, PHP, Java, Python, YAML, Ruby & Perl scripts
- Have very strong expertise knowledge and experience with PowerShell, Perl and **Groovy scripting** in automating many Jenkins, **UrbanCode Deploy** activities with cloud foundry.
- Managed **GIT** version control system in creating branches, tags and merge.
- Hands on experience in configuring and maintaining various Webservers like **Apache** and App servers like **Tomcat**, **Tcserver**, **Weblogic**, **JBoss**.
- Expertise in Querying RDBMS and NoSQL such as Oracle, MYSQL, Dynamo DB, Mango DB and PostgreSQL by using SQL for Data integrity.
- Experience in designing for high availability, building multi-zone and multi-region architectures, and designing across appropriate SQL and NoSQL data layer technologies

## SKILLS

<b>Operating System:</b>	UNIX, Linux, Windows, Solaris, Ubuntu
<b>Containerization Tools:</b>	Docker, Kubernetes, Mesos, Openshift
<b>Configuration management:</b>	Chef, Puppet, Ansible
<b>CI/CD Tools:</b>	Jenkins, Hudson, Bamboo
<b>Build Tools:</b>	ANT, Maven, MS Build
<b>Cloud platforms:</b>	Microsoft Azure, Amazon Web Services (AWS), Google Cloud Platform (GCP), Open stack, PCF
<b>Application/Web Servers:</b>	Oracle Web logic Server 11g, Apache Tomcat, Oracle Application Server 10g BEA WebLogic 8.1/9.2, WebSphere, JBoss, Tomcat, IIS
<b>Scripting &amp; Programming Languages:</b>	Python, Shell Scripting, Bash, Perl, Ruby, Groovy, PowerShell, YAML, HTML, PHP, Java/J2EE, .Net, JSON, NodeJS, GO
<b>Logging &amp; Monitoring Tools</b>	SPLUNK, ELK, Nagios, Datadog, AppDynamics
<b>Security Scanning Tools:</b>	Fortify, NexusIQ, Blackduck, SonarQube, JFrog Xray
<b>Artifact Repository:</b>	JFrog, Nexus Repository Manager (NXRM)
<b>Infrastructure as a code:</b>	Terraform, Cloud Formation Templates, ARM Templates
<b>Virtualization Platforms:</b>	Virtual Box, Vagrant, VMware, vSphere Versions
<b>Databases:</b>	Oracle 10g/11g, Mongo DB, MySQL, SQL, NOSQL

## PROFESSIONAL EXPERIENCE

**Client: Lincoln Financial group, Radnor, PA**

**July 2019 - Present**

**Role: Sr. Cloud/DevOps Engineer**

### Responsibilities:

- Created **Azure** Automation Assets, Graphical runbooks, PowerShell runbooks that will automate specific tasks
- Deployed Azure AD Connect, configuring ADFS authentication flow, ADFS installation using Azure AD Connect
- Hands on experience in **Azure Development, worked on Azure web application, App services, Azure storage, Azure SQL Database, Virtual machines, Fabric controller, Azure AD, Azure search, and notification hub**
- Deployed Azure **IaaS virtual machines (VMs)** and Cloud services (PaaS role instances) into secure VNets with Azure Internal Load Balancer and subnets
- Provided support for Azure Cloud servers environment for project Code Deployments and Oracle DB installations and maintenance
- Created Azure **PowerShell script** for APIM backup and restore operations. Created several SQL scripts as part of bug fixes to provide immediate solutions for some of the common issues
- Managed Azure Infrastructure Azure Web Roles, Worker Roles, VM Role, Azure SQL, Azure Storage, Azure AD Licenses, Virtual Machine Backup and Recover from a Recovery Services Vault using Azure PowerShell and Azure Portal
- Implemented storage related to blobs/Azure files, SQL/Recovery Services, Managing the storage access, Implement, Integrate Azure Active Directory with AD Migration, Configure Application Access Panel

- **Terraform** to create, change and improve production infrastructure and maintained versioning infrastructure safely and efficiently by custom in-house solutions
- Create and maintain highly scalable and fault tolerant multi-tier AWS and Azure environments spanning across multiple availability zones using **Terraform** and **CloudFormation**
- Created **Docker** images using a Docker file, worked on Docker container snapshots, removing images and managing Docker volumes
- Written scripts to containerize using Dockers and orchestrate it using **Kubernetes** & Configured the services using modern DevOps tools, like **Ansible**
- Using **Ansible** as an automation engine for cloud provisioning, configuration management, application deployment, intra service orchestration and multi-tier deployments.
- Communication with team members for both Ansible Core and Ansible Tower teams to clarify requirements and overcome obstacles
- Implementing **Kubernetes** to manage containerized applications using its nodes, Config Maps, Selector, Services and deployed application containers as Pods
- Created private cloud using **Kubernetes** that supports DEV, TEST, and PROD environments
- Used Jenkins and pipelines to drive all **microservices** builds out to the Docker registry and then deployed to Kubernetes, Created Pods and managed using Kubernetes
- Used Jenkins as Continuous integration tool: creating new jobs, managing required plugins, configuring the jobs selecting required source code management tool, build trigger, build system and post build actions, scheduled builds, notifying the build reports etc
- Configured **Jenkins** Monitor, List and Build views to display the status and progress of selected jobs with build version information
- Researched and implemented code coverage and unit test plug-ins with **Maven/Jenkins**
- Worked on google cloud platform (**GCP**) services like compute engine, cloud load balancing, cloud storage, cloud SQL, stack driver monitoring and cloud deployment manager
- Maintain Chef Servers and management application that can use Service Now (CI) data to bring computers into a desired state by managing files, services, or packages installed on physical or virtual machines
- Provide general support, administration and maintenance of the ServiceNow platform, including ITSM, ITFM and other ServiceNow applications
- Automating the Build Infrastructure for deploying services in dockerized environment using Jenkins, SonarQube, Gradle, Groovy, Job DSL, Docker and Splunk
- Created and maintained build related scripts developed in ANT, Ruby and Python and worked with development team to migrate Ant scripts to **Maven**
- Implemented Unix/Linux Shell Scripting, Bash, Perl, and Python for monitoring and to automate the build and deployment process
- Worked with Relational (PL/SQL, Oracle), Non-relational databases (Mongo), and document-oriented databases (**MongoDB**) and Oracle RAC
- Developed Perl and shell scripts for automation of the build and release process. Developed automation scripting in Python to deploy some applications

**Client: Pointsbet, Denver, CO**

**Nov 2018 – July 2019**

**Role: Infrastructure /DevOps Engineer**

**Responsibilities:**

- Experienced in **AWS** Cloud platform and its features which includes **EC2, S3, VPC, EBS, ELB, RDS, RDB, Elastic Beanstalk, DynamoDB, RedShift and Route 53, EC2 AMI, EBS Cloud watch, AWS Config, and Auto-scaling, IAM**
- Deployed LAMP based applications in AWS environment, including provisioning MYSQL -RDS and establish connectivity between EC2instance and MySQL-RDS via security groups
- Provisioned **AWS S3** buckets for backup of the application and sync this content with remaining s3 backups, by creating entry for AWS S3 SYNC in crontab
- Created customized AMIs based on already existing **AWS EC2** instances by using create image functionality, hence using this snapshot for disaster recovery
- Architected and led migration to AWS and Implemented **AWS Redshift** for an automated media buying network including integration of Tableau for data visualizations

- Implemented and designed AWS virtual servers by **Ansible** roles to ensure deployment of web applications
- Used Ansible as an automation engine for cloud provisioning, configuration management, application deployment, intra service orchestration and multi-tier deployments
- Orchestrated and migrated CI/CD processes using Cloud Formation and **Terraform** Templates and Dockerized the infrastructure, which was setup in Vagrant, AWS and VPCs
- Used **Kubernetes** to deploy scale, load balance, scale and manage Docker containers with multiple namespace versions
- Created **Docker** images using a Docker file, worked on Docker container snapshots, removing images and managing Docker volumes
- Implemented a Continuous Delivery pipeline with Docker, Jenkins and **GitHub** and AWS AMI's, whenever a new GitHub branch gets started, Jenkins, our Continuous Integration server, automatically attempts to build a new Docker container from it, The Docker container leverages Linux containers and has the AMI baked in. Converted our staging and Production environment from a handful AMI's to a single bare metal host running Docker
- Created Clusters using Kubernetes and worked on creating many pods, replication controllers, services, deployments, labels, health checks and ingress by writing YAML files
- Worked with **RedHat OpenShift** Container Platform for Docker and Kubernetes, used Kubernetes to manage containerized applications using its nodes, ConfigMaps, node-selector, Services and deployed application containers as Pods
- Managed Kubernetes charts using Helm. Created reproducible builds of the Kubernetes applications, managed Kubernetes manifest files and managed releases of Helm packages
- Provisioned monitoring, metrics, and logging systems on AWS using tools such as SPLUNK, ELK, Sensu
- Managed Amazon Web Services (AWS) infrastructure with automation and configuration management tools Chef, and Ansible
- Worked on writing Jenkins build a pipeline with **Gradle script** and **Groovy DSL (Domain Specific Language)** and integrating **ANT/MAVEN** build scripts with Gradle for the sole purpose of continuous build
- Implemented multiple high-performance MongoDB replica sets on EC2 with robust reliability
- Implemented Jenkins Code Deploy plugin to deploy to AWS and used to automate the build process and deploy the application to Tomcat server
- Used Jenkins for Continuous Integration and deployment into Tomcat Application Server
- Extensive experience in designing and implementation of continuous integration, continuous delivery, continuous deployment through Jenkins.
- **Splunk** and Cloud Watch in the Amazon Web Services (AWS) environment
- Used CloudFront to deliver content from AWS edge locations to users, allowing for further reduction of load on front-end servers
- Configured an **AWS Virtual Private Cloud (VPC)** and Data Base subnet group for isolation of resources within AWS RDS
- Used Amazon Route53 to manage DNS zones and also give public DNS names to elastic load balancers ip's
- Used Amazon RDS Multi-AZ for automatic failover and high availability at the database tier for **MYSQL** workloads
- Set up Auto Scaling Groups based on memory and CPU to adapt to unforeseen spikes without having an outage or needing manual intervention
- Performed Branching, Tagging, Release Activities on Version Control (SVN, GIT)

**Client: Conning, Hartford, CT**

**April 2017 – Nov 2018**

**Role: DevOps Engineer**

**Responsibilities:**

- Created AWS Launch configurations based on customized AMI and use this launch configuration to configure auto scaling groups and Implemented AWS solutions using **EC2, S3, RDS, DynamoDB, Route53, EBS, Elastic Load Balancer, Auto scaling groups**
- Integrated AWS CloudWatch with AWS EC2 instances for monitoring the log files, store them and track metrics
- Created AWS S3 buckets, performed folder management in each bucket, Managed cloud trail logs and objects within each bucket
- Managing **Amazon Web Services (AWS)** infrastructure with automation and configuration management tools such as Udeploy, Puppet or custom-built designing cloud-hosted solutions, specific AWS product suite experience

- Lead many critical on-prem data migrations to AWS cloud, assisting the performance tuning and providing a successful path towards **Redshift** Cluster and RDS DB engines
- Well Versed with Configuring Access for inbound and outbound traffic RDS DB services, **DynamoDB** tables, EBS volumes to set alarms for notifications or automated actions
- Created Highly Available Environments using Auto-Scaling, Load Balancers, and SQS
- Administered and engineered **Jenkins** for managing weekly Build, Test and Deploy chain, GIT
- Integration of Automated Build with Deployment Pipeline. Currently installed Chef Server and clients to pick up the Build from Jenkins repository and deploy in target environments (Integration, QA, and Production)
- Worked **ChefDK** which takes care of creating cookbooks and recipes. Used **Ansible** for configuring and managing multi-node configuration management over SSH and **PowerShell**
- Experience supporting Chef Environment with 200+ servers and involved in developing manifests
- Implemented Chef Recipes for Deployment on build on internal Data Centre Servers. Also re-used and modified same Chef Recipes to create a Deployment directly into Amazon EC2 instances
- Managed **Ubuntu** Linux and Windows virtual servers on AWS EC2 using Open-Source Chef Server
- Proficient in writing Cloud Formation Templates (CFT) in YAML and JSON format to build the AWS services with the paradigm of Infrastructure as a Code
- Set up and built AWS infrastructure with various services available by writing cloud formation templates in json
- Managed AWS Cloud Formation templates to create custom sized VPC, subnets, NAT and Route53 to ensure successful deployment of Web applications and database templates
- Developed Cloud Formation scripts to build on demand EC2 instance formation
- Automated the cloud deployments using chef, python (boto& fabric) and AWS Cloud Formation Templates
- Created a microservice environment on the cloud by deploying services as docker containers. We used Amazon ECS as a container
- Experience with monitoring and logging tools like **Dynatrace**, **SPLUNK**
- Setting up SPLUNK monitoring on Linux and windows systems, monitored and tracked SPLUNK performance problems, administrations and open tickets with SPLUNK
- Management service to run microservices on a managed cluster of EC2 instances. Implemented Amazon API Gateway to manage, as an entry point for all the API's
- Actively involved in the DevOps streamlining Process through Jenkins CI and CD Release Automation

**Client: Inside Connect Cable, Louisville, KY**

**Feb 2016 - April 2017**

**Role: Java/Automation Engineer**

**Responsibilities:**

- Developed build and deployment scripts using **ANT** and **MAVEN** as build tools in Jenkins to move from one environment to other environments
- Perform Life Cycle Management for ESXi hosts. Worked on capacity planning and management of Virtual machines in the VMware environment
- Worked on installation and configuration of DevOps Automation Tool **Puppet**
- Installed/Configured/Managed Puppet Master/Agent. Wrote custom Modules and Manifests, downloaded pre-written modules from puppet-forge. Upgradation or Migration of Puppet Community and Enterprise
- Installed Jenkins/Plugins for GIT Repository, Setup SCM Polling for Immediate Build with Maven and Maven Repository (Nexus Artifactory) and Deployed Apps using custom modules through Puppet
- Experience in writing and Distributing modules to puppet agents
- Implemented a Continuous Delivery pipeline with Docker, **Jenkins and GitHub and AWS AMI's**. Implemented new projects builds framework using Jenkins & maven as build framework tools
- Worked on Integrated **JIRA** with subversion and Jenkins to automate the weekly build process
- Implemented the setup for Master slave architecture to improve the Performance of Jenkins
- Used the Eclipse as IDE, configured and deployed the application onto WebLogic application server using Maven build scripts to automate the build and deployment process
- Involved in a module where the development for front end included service side technology like **NodeJS**
- Designed a single page application with **AngularJS**
- Performed User Interface Design and coding using Java, Spring Boot framework and web technologies



- Streamlined the server implementation by moving to **JSON** and **MongoDB** as the backend data model
- Utilized **Cassandra**, NOSQL database management system for keeping user generated data by upgrading of Couch base and Cassandra cluster
- Used **Java Message Service (JMS)** for reliable and Asynchronous exchange of important information for status reports along with Apache Kafka as Queue Server
- Created SQL queries, PL/SQL Stored Procedures, Functions for the Database layer by studying the required business objects and validating them with Stored Procedures using DB2. Also used JPA with Hibernate provider
- Used Spring Data with JPA Repository and JPA, Hibernate annotations for handling ORM CRUD operations
- Used Junit for unit testing & Jenkins for automated build and deploy process

**Client: PRC Group, Hyderabad, India**

**Oct 2014 - Dec 2015**

**Role: Junior Java Developer**

**Responsibilities:**

- Involved in gathering business requirements directly from the end user and prepared System impact analysis document (SIA) and functionality documents
- Build **Microservices** for the delivery of software products across the enterprise
- Responsible for the full software development life cycle (SDLC) using **Agile Methodology**, participated in Daily stand-up meetings with Scrum Master, communicated with testers and resolved tickets
- Developed RESTful Microservices using Spring **REST** and **MVC** and used Swagger for manual testing and documentation of Microservices
- Created a platform as infrastructure with **AWS (EC2, RDS, ELB)** used Jenkins to run the automated deployments
- Implemented a continuous Delivery pipeline with **Dockers, Jenkins and GitHub and AWS AMI**
- Developed unit and integration tests for existing Micro services using **JUnit, Mockito and Dockers**
- Designing and deploying enterprise-wide scalable operations on AWS
- Performed unit testing of applications by developing and applying test cases in JUnit
- Used **GIT** as a version control

**Client: Choice Software Ltd, Hyderabad, India**

**Aug 2013 - Oct 2014**

**Role: System Engineer**

**Responsibilities:**

- Interacted with various business team members to gather and documented the requirements
- Level 2 Support for Applications Running on Production servers and ensure high availability to internal and external users/clients by responding to application related issues or requests
- Checking system and application logs for errors in various services and to troubleshoot the same
- Participate in **Enterprise Change Management process (ECM)** to install/upgrade patches for different applications running on production and test servers
- Responsible for configuring and connecting to SSH through SSH clients like **Putty**
- Created and modified users and groups with SUDO permission.
- Responsible for setting up CRON jobs scripts on production servers
- Monitoring the server for CPU usage, Memory usage, Network usage and disk utilization using netstat utilities.
- Running SQL update queries to fix production issues
- Interacting with Application developers while working on a production fix
- Testing new applications that are in pipeline which will replace the existing applications and to create documents and establish a process to do future troubleshooting
- Performing Linux System Administration routines, swap partition and file system changes, system booting and recovery

**Education**

**Master of Science, Informational Technology Management**

Campbellsville University, Campbellsville, KY

**Bachelor of Technology, Electronics and Communication Engineering**

Velagapudi Ramakrishna Siddhartha Engineering College, India