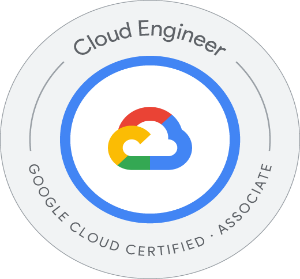
**Arun Acha** 

**Contact No: 737-255-6689**

**Email:**[aruncloud26@gmail.com](mailto:aruncloud26@gmail.com)

**Summary**

* Worked as DevOps Engineer in automating, building, deploying, managing, and releasing of code from one environment to other environment tightly maintaining Continuous Integration (**CI**), Continuous Delivery and Continuous Deployment (**CD**) in multiple environments like (DEV/FUT/STAGE & PROD).
* Hands on experience with **AWS** services like **EC2, S3, RDS, VPC, ELB, EBS, Cloud Watch,** and Auto scaling.
* Cloud Development and automation using Node.js. GO. Aws Lambda, AWS **CDK** (Cloud Deployment Kit) and AWS SAM (Serverless Application Model)
* Strong experience with UNIX/**Linux** in installing and configuring **LVM, RAID, NGINX, HTTPD, Tomcat, MySQL**, Oracle, patching, custom log metrics, configured custom Cloud Watch metrics.
* Expertise in **AWS** Identity and Access Management (**IAM**) such as creating users, groups, organizing IAM users to groups, assigning roles to groups.
* Good experience on **Linux** Administration installing, configuring, and troubleshooting webservers and application servers and been worked configuring **LVM** and **RAID**.
* Experience with DevOps essential tools like **Git, Jenkins, maven, Ant. Docker, EKS, Chef**, **Terraform** and **Linux**/Unix system administrator with RHEL, CentOS & Amazon **Linux**.
* Configured and monitored distributed and multi-platform servers using **Chef**. Defined **Chef Server** and workstation to manage and configure nodes.
* Good experience on containerization technology such as **Docker swarm** and **EKS**.
* Setup up and maintenance of automated environment using basic **Chef recipes** & **cookbooks** and mostly on Puppet manifest and modules within AWS environment.
* Writing **Kafka** playbooks, replacing the dependency on **Chef** Cookbooks and **Chef** Recipes to automate infrastructure as a code.
* Strong knowledge on **Kafka** Roles, inventory, **Kafka** Tower, **Kafka** Galaxy concepts and used **Kafka** Control server to deploy plays and playbooks to the machines and systems in the inventory.
* Worked on integrating **Kafka** Tower with cloud environment, provided role-based access control, setup job monitoring, email notifications, Scheduled jobs, multi-playbook workflow to chain playbooks.
* Good Experience with writing vagrant files and provisioning VMs on developers’ workstations.
* Expertise in the Installation, Configuration and Administration of Apache web server, Nginx and WebLogic.
* Setup SCM Polling for Immediate Build with Maven and Maven Repository (Nexus Artifactory) by installing **Jenkins** Plugins for GIT Repository.
* Providing 24 x7 on-call support in debugging and fixing **Linux** and Middleware related issues in **AWS** Cloud Environment.

**PROFESSIONAL SKILLS:**

**Operating Systems** :UNIX, **Linux**, Ubuntu, RHEL 5/6/7, Windows 7/8/10

**SCM Tools** : CVS, SVN, GIT, Bit Bucket, GitHub

**Build Tools** : ANT, Maven, Gradle, TFS

**CI Tools, CM Tools** : **Jenkins**/Hudson, Bamboo, Antilipo/: **CHEF**, Puppet, **Terraform**

**Database** :SQL Server, Oracle 9i/10g, PL/SQL, NoSQL’s

**Scripting Languages** : GO, Shell, Ruby, Perl, Groovy, JavaScript, XML

**Servers** : Apache, Tomcat, JBoss, WebLogic, WebSphere

**Cloud Computing** : AWS, GCP, Cloud Foundry, Salesforce

**Bug tracker & Testing Tools** : JIRA, Splunk, Nagios, Bugzilla, Junit, JMeter.

**Languages** : C, C#, Java, J2EE

**Professional Experience:**

**Client: DXC Technology Date:** Feb2019-Till Date

**Location: Austin Tx (Remote)**

**Role: GCP DevOps Engineer**

* Created recommendations on how to duplicate a subset of on-premises machines to the GCP Infrastructure as a Service (**IAAS**) offering which will be used for disaster recovery. This analysis included the specifics to synchronize on-premises data with SQL Server and SharePoint instances hosted in VMs.
* Worked on **GCP** Site Recovery and GCP Backup- Deployed Instances on GCP environments and in Data centers and migrating to GCP using GCP Site Recovery and collecting data from all GCP Resources using Log Analytics and analyzed the data to resolve issues.
* Configured **GCP** Multi-Factor Authentication (MFA) as a part of GCP AD Premium to securely authenticate users and worked on creating custom GCP templates for quick deployments and advanced PowerShell scripting. Deployed GCP **SQL DB** with GEO Replication, GCP SQL DB Sync to standby database in another region & Fail over configuration.
* Worked on services, created, and configured **HTTP** Triggers in the GCP Functions with application insights for monitoring and performing load testing on the applications using the Visual Studio Team Services (VSTS) also called as GCP DevOps Services.
* Created **GCP** Automation Assets, Graphical runbook, PowerShell runbook that will automate specific tasks, deployed GCP AD Connect, configuring Active Directory Federation Service (AD FS) authentication flow, **ADFS** installation using GCP AD Connect, and involved in administrative tasks that include Build, Design, Deploy of GCP environment.
* Configure Continuous Integration from source control, setting up build definition within Visual Studio Team Services (**VSTS**) and configure continuous delivery to automate the deployment of **ASP.NET MVC** applications to GCP web apps and managed GCP Active Directory, Office 365 and applied upgrades on a regular basis.
* Implemented a **CI/CD** pipeline with **Docker**, Jenkins (**TFS** Plugin installed), Team Foundation Server (TFS), GitHub and GCP Container Service, whenever a new TFS/GitHub branch gets started, Jenkins, our Continuous Integration (**CI**) server, automatically attempts to build a new **Docker** container from it.
* Worked with **Terraform** Templates to automate the GCP Iaas virtual machines using Terraform modules and deployed virtual machine scale sets in production environment.
* Managed **GCP** Infrastructure GCP Web Roles, Worker Roles, VM Role, GCP **SQL**, GCP Storage, GCP AD Licenses, Virtual Machine Backup and Recover from a Recovery Services Vault using GCP PowerShell and GCP Portal.
* Written Templates for **GCP** Infrastructure as code using Terraform to build staging and production environments. Integrated GCP Log Analytics with GCP VMs for monitoring the log files, store them and track metrics and used Terraform as a tool, Managed different infrastructure resources Cloud, VMware, and **Docker** containers.
* Worked on **Openshift** for container orchestration with **EKS** container storage, automation to enhance container platform multi-tenancy also worked on with **EKS** architecture and design troubleshooting issues and multi-regional deployment models and patterns for large-scale applications.
* Deploying windows **EKS** (K8s) cluster with GCP Container Service (ACS) from GCP CLI and Utilized **EKS** and **Docker** for the runtime environment of the CI/CD system to build, test and Octopus Deploy.
* Worked in container-based technologies like **Docker**, **EKS** and **Openshift**. Point team player on **Openshift** for creating new Projects, Services for load balancing and adding them to Routes to be accessible from outside, troubleshooting pods through ssh and logs, modification of Build configs, templates, Image streams, etc.
* Managing the **Openshift** cluster that includes scaling up and down the AWS app nodes.
* Working on implementing new OCR solution; Spring Boot. **Openshift**, microservices. Member of group developing containerized applications; **Docker**, Spring Boot,
* **EKS**, **Openshift**. Deployed Microservices to IBM Bluemix Cloud Foundry and later migrated to **Openshift**.
* The deployment model uses Atlassian development repository tools, Jenkins as the build engine, while execution deployments to container orchestration tools ranged over time from **Openshift** on EC2, AWS.
* Enterprise Container Services, and today using AWS Faregate Implemented Micro Services framework with Spring Boot, NODE.JS and **Openshift** containerization platform (OCP).
* Using **Kafka** created multiple playbooks for machine creations and SQL server, cluster server and my SQL installations.
* Integrated the f5 client VPN with GCP **Anthos** to create the secured tunnel between cloud to on-premises Automated the CI/CD pipeline and created a **Docker** images using Bitbucket, Maven, O Jenkins and push it to google container registry.
* Integrated the **Linux** environment with Active Directory providing a Single Sign On (SSO) solution.
* Using rsyslog & mysql, deployed a custom syslog reporting solution for system alerts and notifications.
* Deployed Redhat Satellite Server with custom repositories to provide a stable management solution for the **Linux** environment.
* Experience with Jumpstart & Kickstart Installations on Solaris & **Linux**
* Installed, maintained and finetuned the Apache-Tomcat serverand Websphere Application Server in **Linux** Platform
* Created Bash/Korn shell scripts to monitor system resources and system maintenance.
* Written Shell scripts for automating the deployment process to all environments in the public cloud.
* Automated the release process & **GKE** cluster upgrades to prod environment without any downtime using node affinity in **EKS**.
* Experience in rotating the application logs from persistent disk to google cloud storage using **GSutil** command line tool.
* Good experience in Nginx and HA Proxy as a reverse proxy & load balancer. administering the database on google cloud **MySQL** creating failover replicas and database backups.
* Shifted the **EKS** cluster workload between cloud to on-premises via Anthos
* Deployed the resources to Google **Anthos** by using the terraform code and Terraform backend in GCP buckets.
* Used Terraform to Setup/teardown of **ELK** stack (Elasticsearch, Log stash, Kibana) and troubleshoot the build issues with ELK and work towards the solution.
* Written **Terraform** handlers with multiple tasks to trigger multiple handlers and to decouple handlers from their names, making it easier to share handlers among Playbooks and Roles.
* Managed **EKS** charts using Helm, Created reproducible builds of the **EKS** applications, managed **EKS** manifest files and Managed releases of Helm packages.
* Worked on **GKE** Topology Diagram including masters, slave, RBAC, helm, kubecti, ingress controllers GKE Diagram including masters, slave, RBAC, helm, kubectl, ingress controllers
* Implemented **Docker** -maven-plugin in **Maven** pom.xml files to build **Docker** images for all microservices and later used **Docker** file to build the **Docker** images from the Java jar files.
* Virtualized the servers using **Docker** for the test environments and dev-environments needs, also configuration automation using **Docker** containers.
* Experience in creating **Docker** Containers leveraging existing **Linux** Containers and AMI's in addition to creating **Docker**
* Containers from scratch
* Managed local deployments in **EKS**, creating local cluster and deploying application containers.
* Configured MQ Series network using clustering, distributed queuing and remote administration.
* Container management using **Docker** by writing **Docker**files and set up the automated build on **Docker** HUB and installed and configured **EKS**.
* Designed, wrote and maintained systems in **GO** scripting for administering **GIT**, by using Jenkins as a full cycle continuous delivery tool involving package creation, distribution, and deployment onto Tomcat application servers via shell scripts embedded into **Jenkins** jobs.
* Worked with Nagios for GCP Active Directory & **LDAP** and Data consolidation for **LDAP** users. Monitored system performance using Nagios, maintained Nagios servers and added new services & servers.
* Built and managed a highly available monitoring infrastructure to monitor different application servers like **JBoss**, **Apache** Tomcat and its components using **Nagios**.

**Environment:** - GCP, Office 365, Terraform, Maven, Jenkins, Terraform, GCP ARM, GCP AD, GCP Site Recovery, **EKS**, GO, Ruby, XML, Shell Scripting, PowerShell, Nexus, JFrog Artifactory, Jenkins, Git, Jira, GitHub, Terraform, **Docker**, Windows Server, TFS, VSTS, LDAP, Nagios.

**Client: Synapse group Inc Date:** Oct2018-Feb2019

**Location: Connecticut**

**Role: AWS/DevOps Engineer**

**Responsibilities:**

* Develop installer scripts using **Ant, GO, Unix** for various products to be hosted on Application Servers
* Designed and implemented end-to-end DevOps solutions using AZURE DevOps, resulting in faster and more
* efficient deployments.
* Moved services from on premises to AZURE Cloud Environments. Using AZURE Portal and PowerShell scripts,
* AZURE Security groups were managed and linked to VMs and subnets.
* Automated deployments, operations, and scaling of application containers across clusters of hosts, provided
* container-centric infrastructure by Kubernetes.
* Installed and Configured the Ansible Configuration Management Tool for managing Web applications,
* Environments configuration Files, Users, Mount points, Packages. Used the Ansible Vault to secure the
* credentials used during deployments.
* Automated various manual tasks using PowerShell scripts, reducing deployment time by 30%.
* Implemented Infrastructure as Code (IaC) using Terraform and ARM templates, resulting in improved reliability
* and consistency of infrastructure deployments.
* Deploying windows Kubernetes (K8s) cluster with AZURE Container Service (ACS) from AZURE CLI and Utilized
* Kubernetes and Docker for the runtime environment of the CI/CD system to build, test and Octopus Deploy.
* Managed AZURE Infrastructure AZURE Web Roles, Worker Roles, SQL AZURE, AZURE Storage, AZURE AD
* Licenses. Virtual Machine Backup and Recover from a Recovery Services Vault using AZURE PowerShell and
* Portal.
* Developed Ansible playbooks, inventories and custom playbooks in YAML, and encrypted the data using
* Ansible Vault and maintained role-based access control by using Ansible Tower and implemented IT orchestration
* using Ansible to run tasks in a sequence which can work on different servers.
* Written Ansible handlers with multiple tasks to trigger multiple handlers and to decouple handlers from their
* names, making it easier to share handlers among Playbooks and Roles.
* Automated Installation and configuration of Apache, Tomcat servers using Python and supported
* implementations of Kafka and RabbitMQ message servers running Linux and Windows server.
* Worked with Terraform Templates to automate the AZURE IaaS virtual machines using terraform modules and
* deployed virtual machine scale sets in production environment.
* Implemented Docker -maven-plugin in Maven pom.xml files to build Docker images for all microservices and later
* used Docker file to build the Docker images from the Java jar files.
* Worked with Nagios for AZURE Active Directory &amp; LDAP and Data consolidation for LDAP users. Monitored system
* performance using Nagios, maintained Nagios servers and added new services &amp; servers.
* Environment: AZURE, Docker, Ansible, Kubernetes, Terraform, YAML, AZURE Container Service, AZURE CLI, AZURE
* Web Roles, IaaS, PaaS, Kafka, Nagios, Power Shell, Jenkins, GIT, Apache Tomcat Server, Maven, AZURE functions,
* Windows Server, Agile.

**Environment**: Terraform,Terraform, AWS, **Docker**, GIT, **Jenkins**, Red Hat Ubuntu, SQL, Ruby, Shell.

**Client: Kohl’s Innovation Centre Date:** Apr2017-Oct2018

**Location: Milwaukee**

**Role: DevOps Automation Engineer**

**Responsibilities:**

* Involved in migrating service from a managed hosting environment to **AWS** including overall plan, cost analysis, service design, network layout, data migration, automation, deployments and cutover, monitoring, documentation, and timeline.
* 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,** **Dynamo DB, Route53, EBS, Elastic** Load Balancer, Auto scaling groups.
* Launched and configured Amazon **EC2** Cloud Servers using AMI's (**Linux**/Ubuntu) and configured the servers for specified applications.
* Created **S3** buckets and maintained and utilized the policy management of S3 buckets and Glacier for storage and backup on AWS.
* Mass Build of **Linux**/Solaris OS servers using kickstart /jumpstart automation application.
* Building **Linux** servers (web-logic application, apache, DB servers etc) in large quantity as per EDC and Non EDC production requirement as well as App/Dev requirement.
* Worked with the development team to generate deployment profiles (jar, war, ear) using **MAVEN** Scripts and **Jenkins**.
* Responsible for Continuous Integration (CI) and Continuous Delivery (CD) process implementation using **Jenkins** along with Shell scripts to automate routine jobs.
* Involved in setting up the **CI/CD** pipeline using **Jenkins**, Maven, Nexus, GitHub, **Terraform**, **Terraform** and AWS.
* Configured **AWS** Identity Access Management (IAM) Group and users for improved login authentication.
* Deployed and monitored scalable infrastructure on Amazon Web Services (AWS) & configuration management using **Terraform**.
* Involved in writing various Custom **Terraform** Playbooks for deployment orchestration and developed **Terraform** Playbooks to simplify and automate day-to-day server administration tasks.
* Created **Jenkins** Automated Pipeline for **CI** and **CD** with Maven Scripts along with GIT Version control.
* Experienced in **GIT** for branching, tagging, and merging. Also, created branches and Tags for each release.
* Used **Jenkins** AWS Code Deploy plugin to deploy.
* Connected continuous integration systems with the Bitbucket version control repository and continually built as the check-ins came from the developer and managed **Maven** project dependencies by creating parent-child relationships between projects.
* Installed, configured and administered **Jenkins** CI for Gradle and Maven Builds of RDBMS and NoSQL tools such as Dynamo DB.
* Knowledge and expertise on all aspects of **Chef** concepts like **Chef** server, workstations, Nodes, **Chef** clients and various components like push jobs, supermarket etc.
* Responsible for building application full automation pipeline for deployment into AWS using Jenkins, Artifactory, **Chef** and Terraform.
* Writing Json templates for cloud formation and ruby scripts for **Chef** automation and contributing to our repository on GitHub
* Writing **Chef** Recipes, Cookbooks in **Chef** for Application Deployment automation, Cloud Infrastructure Automation.
* Worked in container-based technologies like **Docker**, **EKS** and **Openshift**. Point team player on **Openshift** for creating new Projects, Services for load balancing and adding them to Routes to be accessible from outside, troubleshooting pods through ssh and logs, modification of Build configs, templates, Image streams, etc.
* Managing the **Openshift** cluster that includes scaling up and down the AWS app nodes.
* Had very strong exposure using Kafka automation in replacing the different components of **Openshift** like ECTD, MASTER, APP, INFRA, Gluster.
* Used Splunk for configuration and installation process.
* Working on implementing new OCR solution; Spring Boot, **Openshift**, microservices. Member of group developing containerized applications; **Docker**, Spring Boot, **EKS**, **Openshift**.
* Implement cluster services using **Docker** and **EKS**.
* Developed Dev/Test/Prod environments of different applications on AWS by provisioning **EKS** clusters on EC2 instances using **Docker**. Bash, Chef, and Terraform.
* Administer, secure, and maintain multiple **EKS** clusters to support a global service.
* Used Jenkins pipelines to drive all micro services builds out to the **Docker** registry and then deployed to **EKS**, Created
* Pods and managed using **EKS**
* Building/Maintaining **Docker** container clusters managed by **EKS** **Linux**, Bash, GIT, **Docker**, on GCP (Google Cloud Platform) Utilized **EKS** and **Docker** for the runtime environment of the CI/CD system to build, test deploy.
* Involved in development of test environment on **Docker** containers and configuring the **Docker** containers using **EKS**.
* Developed microservice on boarding tools leveraging GO and Jenkins allowing for easy creation and maintenance of build jobs and **EKS** deploy and services.
* Refactoring of monolithic applications to a Microservices and Component based architectures.

worked on deployment automation of all the micro services to pull image from the private **Docker** registry and deploy to **Docker** swarm cluster using Kafka.

* Building/Maintaining **Docker** container clusters managed by **EKS** **Linux**, Bash, GIT, **Docker**, on GP O Worked with **Openshift** platform in managing **Docker** containers and **EKS** Clusters.
* Worked on infrastructure with **Docker** containerization and maintained **Docker** Images and containers.
* Implemented and managed Dynatrace on business social critical desktop and mobile applications.
* Resolved 30+ customer complaints everyday related to Jenkins, Git, Clearcase, Nexus, Sonar, Java.
* Created Jenkins CICD pipelines for continuous build & deployment and integrated Junit and SonarQube plugins in jenkins for automated testing and for Code quality check.
* Integrated SonarQube with Jenkins for continuous inspection of code quality and analysis with SonarQube scanner for Maven.

**Environment**: AWS, EC2, S3, EBS, ELB, Auto Scaling groups, VPC, IAM, Cloud Watch, Micro Services, Glacier, Bitbucket, Dynamo DB, RDBMS, Shell Scripting, GIT, **Terraform**, **Docker**, **EKS**, **Docker** Swarm, **Chef**, Maven, Gradle, **Jenkins**, GO, YAML.

**Client: PAREXEL India Limited Date:** Jan 2012 to Dec 2015

**Location: Hyderabad, India**

**Role: Linux/Windows Admin**

**Responsibilities:**

##### Experience with different flavors of Linux (RHEL, CentOS, Oracle Linux, Debian, and Ubuntu).

##### Installed and configured Red Hat Linux Kickstart and Solaris jumpstart Servers.

##### Setup a couple of hundred VM's running CentOS to be used for web, database, application, mail, ftp, monitoring and Git repositories.

##### Designs, configures, implements, and maintains UNIX hardware and operating system environments including AIX, HP-UX, and Solaris.

##### Worked on red hat 5.x, performed daily tasks using CRON scheduler, and automated it.

##### Discussed and planned the development issues with the team members and the lead.

##### Daily tasks such as taking data backups, cleaning junk content and updating software.

##### Configuration of Hardware and Software RAID on Digital & Sun Servers.

##### Installation of Oracle Patches and Troubleshooting, Creating and modifying application related objects, Creating Profiles, Users, Roles and maintaining system security.

##### Configuration and maintenance of NIS, NFS, DHCP and DNS Servers on Solaris.

##### Administrative tasks such as System Startup/shutdown, Backup strategy, Printing, Documentation, User Management, Security, Network management, dumb terminals and devices carried out.

##### Experienced in Installing and Configuring Veritas Storage foundation Suite (Veritas clustering and Veritas Volume Manager).

##### Installed and managed virtual Linux machines on VMWare ESXi.

##### Worked as part of Testing Team in Application testing using Manual methods and Scripts.

##### Installed and implemented NAS with RAID 1 and RAID 5 configurations.

##### Monitored load and performance on the infrastructure and added capacity as needed.

##### Installation, configuration and management of PostgreSQL & MySQL database servers.

##### Creates and manage LVM volumes and participate in root cause analysis.

##### Experience in writing UNIX Shell Scripts using commands like sed, bash, ksh and perl to automate system admin jobs.

##### Resolve network issues and skilled in network protocols such as TCP/IP, and DNS, DHCP.

##### Familiar with Disk Management Utility. Hands on experience in file system creation and file system management.

##### Mass Build of Linux/Solaris OS servers using kickstart /jumpstart automation application.

##### Building Linux servers (web-logic application, apache, DB servers etc) in large quantity as per EDC and Non EDC production requirement as well as App/Dev requirement.

##### 

##### Environment: RHEL, CentOS, Oracle Linux, Debian, and Ubuntu, VMWare, Veritas, LVM, Postgresql,Mysql

Education:

* Masters in Information Technology Atlantis University, Florida (2016-2017)
* Bachelors in Electrical and Electronics and Engineering, JNTU-H, India (2007-2011)