**A picture containing text, businesscard

Description automatically generated**Graphical user interface, text

Description automatically generated**Krishna Pinnamaraju**

**krishnapinnamaraju25@gmail.com**

**Phone: (813)-736-0527**

Certified Cloud/DevOps Engineer around 10 years of experience in DevOps operations, Build/Release, and Cloud Management with experience in managing server infrastructures and administration of mission-critical, high availability environments. Consistently high-performing individual with proven ability to create and deliver solutions tied to business growth, organizational development. Comfortable in managing systems, projects, and teams in a wide range of IT environments.

**Professional Summary**

* **Certified** **AWS- Solutions Architect** **Associate** with expertise in administration of AWS Cloud Services for Infrastructure Development and Operations involving EC2, ELASTIC BEAN STACK (EBS), S3, VPC, RDS, SES, ELB, Auto scaling, Cloud Front, Cloud Formation, Elastic ache, Cloud Watch, SNS, Kinesis, Lambda, Cloud Watch, Cloud trail and security groups*.*
* Expertise in **AWS-DevOps** platform building and deploying applications by using Code Commit, **Code Pipeline**, Code Build, Code Deploy, EC2, ECS, AWS CLI, and created platform for Chef OpsWorks with reusable stack.
* Expertise in writing **AWS Cloud Formation templates** in JSON to use them as blueprints for building & deploying multiple AWS resources. Worked on Kinesis Data Streams and integrated with AWS Lambda for serverless data collection
* **Certified DevOps Engineer – Expert and** **Azure Administrator -** **Associate** worked on Azure Web Apps, Azure App Services, Azure Virtual Network, Azure SQL Database, Azure Fabric controller, Azure Search, Azure Batch, Azure Autoscale, Azure Files Azure Blobs, Azure Queues, Azure Tables, Azure Data Lake Storage, and Azure Notification Hub.
* Configured deployment with Azure Iaas, Paas services and Azure Resources like Azure VM’s, Azure Storage Account, Virtual Account, Subnet along with Data pipelines using **ARM Templates** and **Azure DevOps.**
* **Certified ITIL Foundation -** Proficiency in implementing ITIL processes for service strategy, design, transition, operation, and continual service improvement.
* Proficient in implementing **Azure** Hybrid integration to integrate the applications, data and processes using **API's**. Worked with **Azure IaaS - Virtual Networks, Virtual Machines, Cloud Services, Resource Groups, Express Route, Traffic Manager, VPN, Load Balancing, Application Gateways, Auto-Scaling.** And Migrated existing Active Directory and Exchanged users to Microsoft Azure AD using Azure AD Connect, Active Directory Federation Service
* Worked with AWS SDK CLI, Google cloud SDK CLI and Azure CLI
* Proficient in troubleshooting the **Kubernetes** **Pod** logs, **Kube2IAM** logs and debugging pod restart and **CrashLoopBackoff** issues and resolving them. Worked on building, migrating and integration of **Istio** toautomate application network functions.
* Extensively involved in setting up **Kubernetes** (**k8s**) **clusters (EKS)** for running microservices and implemented a **Kubernetes** Container Orchestration solution within **AWS** allowing for easy management, creation & recovery of **AWS** assets and managed the container using **Docker,** set up the automated build on **Docker Hub** and run multiple applications.
* Experienced in deployment of **Kubernetes** to scale up **pods**, maintaining services, load balancing, network policies, group Docker containers across different platform and managed entire package with **Helm charts**.
* Experienced in **Bluegreen Deployment and Canary Deployment, Octopus deployment** in the production. Managed docker service rolling updates using these deployments to implement zero downtime PROD deployments
* Extensively deployed Restful APIs and **microservices** using spring on Azure Kubernetes by using **Azure Kubernetes service (AKS),** Ingress API Gateway, MySQL, SQL Databases and Cosmo DB for stateless storage of external data, and setup reverse proxy Nginx servers and encrypting with SSL and TLS.
* Experienced in creating additional **Docker Slave Nodes** for Jenkins using custom **Docker Images** and Worked on all major components of Docker like **Docker Daemon**, Hub, Images, Registry, **Docker Swarm**, configured docker containers with port and volume mapping.
* Experience in working on cloud AWS cloud EC2, S3, RDS, Load Balancer, Auto Scaling with AWS command line interface and AWS PYTHON SDK.
* Performed automation tasks on various Docker components like **Docker Hub**, **Docker Engine, Docker Machine and Docker Registry** for creating the **Docker images** in **Docker Hub** for creating the custom based deployments.
* Highly efficient in configuring and directing continuous delivery with **Docker** connected to **Jenkins** on virtual machines to build and integrate testing jobs while codes triggered from **GitHub**.
* Expert in **Terraform** to reliably version and create infrastructure on **Azure/AWS**. Created resources using **Azure/AWS Terraform** modules and automated infrastructure management and experienced with **Terraform** key features such as **Infrastructure as code**, **Execution plans, Resource Graphs**, **Change Automation.**
* Proficient in developing several **Terraform** modules for infrastructure management, **terraform scripts** to setup AWS **resources EKS, VPC, EC2, AMI, Images, Redshift cluster, ELB, and automate the deployment & infrastructure.**
* Extensive experience in installing, configuring, and administrating **Jenkins** **CI tool** on **Linux** machines. Worked on setting up the **Jenkins CI/CD pipeline** configurations for all the **microservices** build docker images and publish to the **Docker** **registry** and then deployed to **Kubernetes**, created **Pods** and managed using **Kubernetes**.
* Expert in configuring and managed source code using **GIT** and resolved code merging conflicts in collaboration with application developers and provided a consistent environment and implemented Continuous Integration using **Jenkins** and **GIT**. Configured **GIT** with **Jenkins** and schedule jobs using **POLL** **SCM** option and integrated to automate the code checkout process.
* Expertise in using **Ansible and Ansible Tower** as Configuration management tool, to automate repetitive tasks, quickly deploys critical applications, and proactively manages change.
* Extensive experience in writing **Ansible Playbooks** with **Python SSH** as the **Wrapper** to manage **Configurations of Azure Nodes** and Test Playbooks on Azure instances using Python SDK and Automated various infrastructure activities like continuous deployment, application server setup, stack monitoring using **Ansible playbooks.**
* Experienced in working with **Chef servers**, worker nodes, bootstrapping and **Chef nodes** for configuration management; also proficient in creating ***Chef Templates, Recipes, Cookbooks, Resources, Roles, Attributes & Data bags***.
* Implemented new deployment strategies for **Prometheus and Grafana.** Knowledge on monitoring application services using **Prometheus and Grafana**for Kubernetes clusters, also created alerts based on nodes, pods, containers and services.
* Involved in building and deploying Java code through Jenkins and done Code coverage analysis using **SonarQube**. Run quality checks using SonarQube & uploaded it to Nexus repository. Improved monitoring with **Nagios** and custom plugins. Developed utilities in Perl, including templates and configurations for Nagios and basic system configuration. Deployed and maintained deployments in clustering and high availability scenarios. Implemented scripting tools - Ruby as Chef DSL and Shell Scripting and on monitoring tools – **Nagios and Splunk**
* Experienced in working with **Nginx**, **Apache**, **Tomcat web servers** in the clusters in various operating systems such as Linux/Ubuntu and Windows systems. Troubleshooting **Linux/Ubuntu** servers and network related issues using log Configured NIC bonding/NIC Teaming configuration in **Linux/UNIX/Windows** systems to increase the performance based on the requirement by the application., **LAMP and LDAP**.
* Experienced in designing, developing and administrating transactional and analytical data structures and business reports from legacy to AWS and snowflake.

**Technical Skills**

|  |  |
| --- | --- |
| **Cloud** | Amazon Web Services, Microsoft Azure |
| **Container/ Orchestration Tools** | Kubernetes, Docker, Terraform, EKS, AKS |
| **CM & Deployment Tools** | Chef, Ansible, Puppet. |
| **CI/CD Tools** | Jenkins, GITLAB, GITHUB, Azure DevOps, Bamboo. |
| **Build Tools** | Maven, Gradle, ANT |
| **Repositories** | JFrog, Artifactory, Nexus. |
| **Monitoring Tools** | JIRA, Service Now, Splunk, ELK, Azure Monitoring, Grafana, Prometheus, Datadog |
| **Version Control** | GIT, Bitbucket |
| **Programming/Scripting Languages** | Shell, Python, YAML, JSON, PowerShell, Java |
| **Web/Application Servers** | Tomcat, Web logic, Web Sphere. |
| **Database** | SQL Server, No-SQL Database, My SQL, Mongo DB, Cassandra, Amazon RDS |
| **Operating Systems** | Windows, LINUX, UNIX, UBUNTU, MAC |

**Experience**

**Sr. Cloud Engineer *(Apr 2021 – present)***

***Client: Hudson's Bay Company, New York, NY***

***Key Responsibilities:***

* Deployed AWS application infrastructure using sources like **VPC, EC2, S3, MySQL,** Aurora DB**, IAM, EBS, Route53, SNS, ES, SQS, CloudWatch,** Security Group, Auto Scaling Group ASG, and RDS using **CloudFormation, Terraform templates.**
* Migrated classic **AWS infrastructure** to server less architecture **AWS Lambda** via **Terraform** and implemented **AWS Lambda functions** to run scripts in response to **CloudWatch** events in the **Amazon Dynamo DB table or S3 bucket to HTTP** requests using Amazon **API gateway** and invoked the code using **API** calls made using **AWS SDKs**.
* Worked **on Amazon Aurora database service** and implemented automatic machine disaster recovery on **AWS** cloud and also setting up **databases** using **RDS**, storage using **S3 bucket, Amazon** **Glacier** by configuring instance backups to the **S3** bucket and deployed instances multiple availability zones to ensure **fault tolerance and high availability.**
* Created **AWS Cloud formation Stack** using **VPC, subnets, EC2 instances, ELB, S3** and integrated it with **CloudTrail**. Configured **CloudFormation** **templates** are stored in **GIT**, visualized **CloudFormation templates** as diagrams and modified them with the **AWS CloudFormation Designer.**
* Designed and Implemented CI & CD Pipelines using Git, Jenkins, Maven, ECR and EKS from scratch achieving theend to end automation from code check into deploying docker images to EKS clusters
* Orchestrated **CI/CD processes** using **Cloud Formation**, **terraform** templates and containerized using **Docker** setup in **Vagrant.** Developed **microservices** on boarding tools leveraging **Python and Jenkins** allowing for easy creation, maintenance of **build jobs** and deploy services in **Kubernetes**.
* I was part of the team that designed and integrated capabilities to establish a DevSecOps pipeline, utilizing lab and cloud resources to design, build, test and evaluate functional components and technology.
* Experience in creating cloud instance and **Docker deployment** from tagged **docker** builds published to **Amazon ECR** through the **HashiCorp** **Terraform**. Built **Kubernetes** **POC** for **container orchestration for** performing **continuous Delivery** in a **Microservice infrastructure** with Amazon cloud, **Docker** and **Kubernetes.** Performed container management using **Docker** by writing **Docker files** and setting up the automated build on **Docker HUB**.
* Built and deployed the application code using **CLI** of **Kubernetes** using **kubectl**, **kubelet**, **kubeadm, Kubespray** and scheduled the jobs using **Kube-scheduler**. Managed **K8s** charts using **Helm** and created reproducible builds of the **Kubernetes** applications.
* **“DevSecOps Process & Regulatory” Plan & Execution for Cloud Migration Plan**
* **Build scripts on**AWS**cloud for scheduling EC2 auto scaling load balancer with**PYTHON**sdk**
* Created **Kubernetes YAML** files to deploy **SCM CI/CD** applications automatically and Idempotently to reduce time costs. Used **Kubernetes** to deploy, load balance, scale and manage **docker** containers with multiple name-spaced versions.
* Created **Terraform** **templates** for provisioning virtual networks, VM Scale sets, Load balancers, NAT rules and used **terraform graph** to visualize execution plan using the graph command. Used **Terraform** to deploy the infrastructure necessary to create development, test, and production environments for software development projects.
* Created dynamic routing/load balancing capability, allowing for large application scaling, used ingress rules, and a Network Load Balancer wif teh NGINX Ingress Controller on Amazon EKS
* “DevSecOps Change Control & ITIL” Plan & Execution for Cloud Migration Plan
* Designed **Terraform** templates to create custom sized **VPC**, subnets, **NAT** to ensure successful deployment of Web applications and database templates and migration from traditional to cloud environment.
* Set up **Jenkins** server and build jobs to provide Continuous Automated builds based on polling the **Git** source control system during the day and periodically scheduled builds overnight to support development needs using **Jenkins, GIT**.
* Configured **Bitbucket** with **Jenkins** & automated the build process through **SCM** polling. Created post-commit & pre-push hooks using **Python** in **Bitbucket** repositories. Resolved merging issues during rebasing & re-integrating branches.
* 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.
* Created a micro-Services to allow operators to create, update, and delete notes for their customers. The notes were stored in a secured LDAP using UnboundID LDAP SDK.
* Designed an **ELK (Elastic Logstash Kibana**) stack to monitor and search enterprise alerts and configured **ELK** stack in

conjunction with **AWS** and using Logstash to output data to **AWS S3**. Implemented centralized logging system using log stash configured as an **ELK** stack to monitor system logs, **AWS CloudWatch**, **VPC** Flow Logs, CloudTrail Events,

changes in S3 etc.

* Served the **ELK** stack community with use cases and Logstash plugin and Deployed applications using Jenkins’s server and Troubleshoot build & release job failures, resolve, work with developers on resolution.
* Worked with system health and performance monitoring tools like **Splunk, Prometheus, Grafana** and replaced **Splunk logging** and **analytics** with an automated **ELK cluster** for increasing data capture capacity and cost reduction.
* Configured **JIRA** as default tracking system for various workflows, customizations, and plugins for the JIRA bug/issue tracker.
* Collaborated in deployment strategies for **Prometheus**. Monitored application services using **Prometheus and Grafana**for Kubernetes clusters, also created alerts based on nodes, pods, containers and services.
* Implemented a production ready, load balanced, highly available, fault tolerant Kubernetes infrastructure with rancher, kops, EKS
* Worked on testing, evaluation and troubleshooting of **MongoDB** and **Cassandra NoSQL** database systems and cluster configurations to ensure high availability in various crash scenarios.
* Developed automated processes that run daily to check **disk usage** and perform cleanup of file systems on LINUX environments using **shell scripting**.
* Worked on converting process from Legacy (SAS, Teradata) to Cloud (SQL,Python,AWS,Snowflake).
* Implemented ITIL methodologies to streamline processes and enhance operational efficiency. Contributed to the improvement of service delivery through the application of ITIL frameworks.
* Directed and managed the Software Development Life Cycle (SDLC) processes, ensuring seamless integration of development, testing, deployment, and maintenance phases.
* Organized and maintained comprehensive schedules and calendars for project milestones, meetings, releases, and other critical events.
* Led release management activities, coordinating with development, testing, and operations teams to plan and execute successful software releases.

**Environment:** ***AWS (EC2, RDS, ELB, EBS, Auto Scaling, VPC, Route53, RDS, Code Commit, Code Build, Code Deploy, Code Pipeline, Glacier, Redshift, S3, IAM, SNS, SQS, SES, KMS, DynamoDB, Lambda, CloudFront, CloudFormation, ITIL Foundation, CloudTrail, Security Groups, Elasticsearch and CloudWatch), Hashicorp vault, Kubernetes, EKS, Docker, Terraform, Software Development Life Cycle (SDLC), GIT, GitHub, Jenkins, Ansible, ELK, Prometheus, Grafana, Mongo DB, Cassandra NoSQL, Shell and Ruby Script, Linux Systems.***

**Sr. Cloud Engineer (S*ep 2018 to Apr 2021)***

***Client: Travelport, CO***

***Key Responsibilities:***

* Implemented **Azure** Hybrid integration to integrate the applications, data and processes using **API's**. Worked with **Azure AD**, **Azure Blob Storage** and **Data Factory** to compose data storage, movement, and processing micro-services into automated data pipelines and have used **Azure compute** services and **Azure Web apps.**
* Involved in configuration for Web apps/Function apps, V-net integration, **HCM**, Application gateway, **App Insights, Active directory, Azure Key Vault**, Encryption and Security on Azure.
* Experience in Google Cloud components, Google container builders and GCP client libraries and cloud SDK’s
* Configured and implemented storage blobs and Azure files - Created Storage accounts, Configured the **Content Delivery Network (CDN)**, custom domain, Managing access and storage access key.
* Prepared an architecture plan to create the **Azure cloud** environment to host the migrated **IaaS VMs** and **PaaS** role instances for refactored applications and databases. Established connection from **Azure** to On-premises datacenter using **Azure ExpressRoute** for Single and Multi-subscription connectivity.
* Experience with Kubernetes in AWS with EKS.
* Customized EC2 instances using AWS SDKs for application deployment and authored security policies for providing access via AWS IAM.
* Deployed **Python** applications to **Azure DevOps** to focus on development by using services such as **Repos** to commit codes, Built Pipelines using **PowerShell** scripts for CI/CD and Test plans for unit test, deploy to **App Service**, **Azure Application Insights to** collect performance and usage data of the process, stored artifacts in blob storages.
* Created resource in **Azure using VDC** with **ARM Template**, PowerShell Script along with Azure DevOps CI/CD pipeline to automate the deployments.
* Standardized the EKS, trained and supported the application teams to move their applications from on premises to the highly scalable and secured containerized cloud environments.
* Deployed application which is containerized using Docker onto a Kubernetes cluster which is managed by Amazon Elastic Container Service for Kubernetes (EKS).
* Worked on creating clusters using **Kubernetes** and worked on creating many pods, replication controllers, deployments, labels, health checks and ingress by writing Yaml files.
* Deployed **microservices** based applications on Azure Kubernetes by using **Azure Kubernetes service (AKS),** Ingress API Gateway, MySQL, SQL Databases and Cosmo DB for stateless storage of external data, and setup reverse proxy Nginx servers and encrypting with SSL and TLS.
* Created several pods using Master and Minion architecture of **Kubernetes** and developed microservice on boarding tools leveraging **Power Shell** allowing for easy creation and maintenance of build jobs and **Kubernetes** deploy and services.
* Worked on implementation of Hadoop-based **Data lakes** in Microsoft Azure. Implemented the process of securing approvals to store data with PHI in the **Data lakes** in the cloud.
* **Terraform and Cloud Formation templates for deploying EKS cluster, Morpheus Data, IAM roles and policy creation, S3 bucket access, SQS event creation, Lambda Function etc. from scratch.**
* Configured and setup Cluster environment in **Kubernetes** with a master and 3 **minions**. Collaborated with development team for creating **Docker** and **Kubernetes** containers for each micro-service and installed all required dependencies for each container.
* Created additional **Docker Slave Nodes** for Jenkins using custom **Docker Images** and worked on all major components of Docker like **Docker Daemon, Hub, Images**, Registry, **Docker Swarm,** configured docker containers with port and volume mapping.
* Customized EC2 instances using AWS SDKs for application deployment and authored security policies for providing access via AWS IAM.
* Worked with **Terraform** features such as Infrastructure as code, Execution plans, Resource Graphs, Change Automation and Implemented **Terraform** modules for deployment of applications across **Azure**.
* Implemented Jenkins jobs to create **Azure** Infrastructure from **GitHub** repositories containing **Terraform code** and created on-premises active directory authentication.
* Worked on setting up continuous delivery for the **Pods** by running **Jenkins, Docker images**, in Kubernetes. Developed a pipeline using **Jenkins Pipeline** toautomate to push the PODs to the registry, and then deploying it in Kubernetes.
* Managed **Nexus** repository to download artifacts during builds in **Jenkins** and integrated **Ant** and **GIT** to deploy project related tags.
* Integrated **Jenkins CI** with **GIT** version control and implemented continues build based on check in for various process across functional applications and created **GitHub Webhooks** to setup triggers for commit, push, merge, and pull request events.
* Maintained **Bitbucket** Repositories which includes **Jenkins** for Integration, creating new repositories, enabling **GIT** to **ignore**, **branching**, **merging**, **creating** pull requests and the access control strategies from Bitbucket and **JIRA** for the collaboration.
* Designed and deployed open-source ELK stack on the EKS Cluster with the help of Helm Charts
* Administered **JFrog Artifactory**, which includes uploading the artifacts, third party binaries, creating repository and maintaining the Artifactory.
* Monitored and visualized VPC Flow Logs, Infrastructure Logs, Cross Account EKS and CloudWatch logs using the ELK stack
* Worked on setting up the configuration for projects to publish snapshot and release artifacts to **JFrog Artifactory** to share artifacts across projects and environments.
* Created Azure infrastructure using **ARM Templates**, **Azure pipelines** to build, test, and release, and managed **Azure Container Registry** to store private **Docker images**, which are deployed. Grafana Monitor to collect metrics and logs. Configured Monitor to track performance and maintain security, and used **Grafana** tool to collect metrics, queries.
* Created Domain Controller virtual machine with OMS Extensions that connects to Log Analytics using **ARM Template.** Created resource groups that contains (Tags, key vault, Automation Acct, RBAC assignment) using PowerShell/Arm Template/DevOps-pipeline).
* Worked on various components of **Chef** including Chef Server, workstations, Nodes, configurations of **chef-client** and written cookbooks and recipes for Chef and assigning roles to nodes.
* Worked with Chef Server in deploying and configuring **Chef Server** including bootstrapping of **Chef Client** nodes for provisioning. Created roles, recipes, cookbooks and **data bags** for server configuration.
* Developed **Splunk** dashboards with **HTML**, **JavaScript**, and **CSS** to provide a professional lookup as part of enterprise dashboard project for management and deployed **Java** applications to application servers in an agile continuous integration environment and automated the process using **Shell** & **Python** for **Linux**.
* Worked as part of Devops team working on Docker containerization efforts and migration of Java apps to **OpenShift** Container Platform.
* Utilized ITIL concepts to effectively resolve incidents and minimize downtime. Demonstrated expertise in problem-solving by applying ITIL incident management practices.
* Utilized JIRA extensively as the primary project management tool, proficiently managing tasks, workflows, and issue tracking throughout projects.
* Collaborated with cross-functional teams to implement risk mitigation strategies and foster a risk-aware culture within the organization.

***Environment: Azure, Azure Services, HCM, Application gateway, App Insights, Active directory, Azure Key Vault, Content Delivery Network (CDN), Azure DevOps, PowerShell Script, Kubernetes, Terraform, Jenkins, Microservices, AKS, HELM charts, ARM Templates, Jira, Azure pipelines, Docker, Docker Images, Docker Slave Nodes, Docker Daemon, Docker Swam, Docker Container, risk mitigation , ITIL Foundation, Jfrog, Bitbucket, Chef, Chef books, ANT, Grafana, data bags, Java Script, Data Lake, Python Scripts, Linux Systems.***

**Cloud Engineer (*Mar 2016 to sep 2018*)**

***Client: Quotient, CA***

***Key Responsibilities:***

* Implemented **AWS lambda** to run servers without managing them and to trigger to run code by **S3 and SNS** and Converted existing AWS infrastructure to server less architecture (**AWS Lambda, Kinesis**) and deployed via Terraform and AWS cloud formation.
* **Migrated** Linux environment to AWS by creating and executing a migration plan, deployed **EC2 instances** in VPC, configured security groups, attached profiles and roles using AWS Cloud Formation templates.
* Configured an **AWS Virtual Private Cloud (VPC)** and Database Subnet Group for isolation of resources within the Amazon **RDS Aurora DB cluster** and created notifications and alarms for EC2 instances using Cloud Watch.
* Involved in creating and documenting POC for helping migrate the current application to micro service architecture. The architecture included Docker as container technology with **Kubernetes** and worked on with REST API.
* Extensively involved in setting up **Kubernetes** (**k8s**) **clusters (EKS)** for running microservices and implemented a **Kubernetes** Container Orchestration solution within **AWS** allowing for easy management, creation & recovery of **AWS** assets and managed the container using **Docker,** set up the automated build on **Docker Hub** and run multiple applications.
* Used **Docker** for managing development environments through creation of Linux virtual containers.
* Built and maintained **Docker** infrastructure for **Service oriented architecture (SOA) applications**. Deployed Docker Engines in Virtualized Platforms for containerization of multiple applications.
* Experienced with **Terraform** key features such as **Infrastructure as code**, Execution plans, **Resource Graphs**, Change Automation and Used Auto scaling for launching Cloud instances while deploying microservices.
* Integrated Docker container-based test infrastructure to **Jenkins CI test flow** and set up build environment by

integrating with Git and Jira to trigger builds using Webhooks and Slave Machines.

* Installed and administrated Git source code tools and ensured the application's reliability and designed the branching strategies for GIT. Experienced in **GIT forks, tagging, handling merge requests** and notifications and setting up the GIT repos for Jenkins build jobs.
* Built a deployment pipeline for deploying tagged versions of applications to **AWS Beanstalk** using **Jenkins**.
* Worked on designing and deploying AWS Solutions using **EC2, S3, EBS, ELB, Auto Scaling groups**. Cloud as a code with **Ansible.**
* Managed the **Maven** repository using **Nexus** tool and used the same to share the snapshots and release of internal projects.
* Used **Bash** and **Python** and included **Boto3** to supplement automation provided with Terraform for tasks such as

encrypting EBS volumes backing AMIs and scheduling **Lambda functions** for routine AWS tasks

* Created Perl Scripts for complex **Data parsing, Analysis and Report generation** and created Shell scripts for

generation of Disk Space, Archive, system monitoring for staging environment.

* Used **Kibana** to visualize the Elasticsearch data to navigate the **Elastic stack.**
* Maintained CI environments with automation tools like **Jenkins and Bamboo.**
* Developed custom app configurations (deployment-apps) within **SPLUNK** to parse, index multiple types of log format across all application environments i.e., Deploying, configuring and administering Splunk clusters.
* Worked with different Bug Tracking and Ticketing Tools like **JIRA**.
* Implemented methodologies and best practices to streamline SDLC workflows, resulting in improved efficiency and timely project deliveries.
* Utilized advanced scheduling tools and techniques to coordinate tasks among teams and stakeholders, ensuring smooth project progression.

**Environment:** ***AWS (VPC, EC2, S3, IAM, EBS, Security Group, Auto Scaling, RDS) Kubernetes, Docker, Docker Hub, Terraform, GITLAB, AWS Beanstalk, Jenkins, Elastic Stack, Kibana, Nexus, Maven, Splunk, JIRA.***

**Infrastructure Data Platform Engineer *(May 2014 to July*** ***2015)***

***Acuvate Software pvt ltd india***

***Key Responsibilities:***

* Involved in the automation of **AWS** **infrastructure** with **Chef** by creating **OpsWork** platform for management and connected to **Jenkins** for continuous building and integration.
* Implemented **AWS Beanstalk** for **deploying web applications** based on **Java, Python** along with EC2 instances, Elastic Load Balancing **(ELB)**, which performed balance across instances in multiple availability zones. Launched Amazon EC2 instances using AWS **(Linux/ Unix/Red Hat/Windows)**.
* Created AWS **CloudFormation** template by using **JSON/YAML** and implemented **Tomcat** and **Apache** Web servers for deployment and coordinated with offshore and onshore teams for production, also created CRON job for recovering crucial data of production by using AWS REST **API** to store on Dynamo DB.
* Configured and managed Chef infrastructure, wrote cookbooks to automate the configuration setups, and created **Data bags** for implementation, coding standards, Cookbooks maintenance, testing documents.
* Designed **Kubernetes** cluster using **AWS (EKS)** and configuring and deploying the Kubernetes dashboard to access the cluster via its web-based user interface.
* Extensively used **Terraform** to a reliable version and created infrastructure on **AWS**. Also created resources, using **AWS** Terraform modules, and automated infrastructure management. Used Terraform to map more complex dependencies and identify network issues.
* Automating various infrastructure activities like **Continuous Deployment, Application Server setup**, Stack monitoring using Chef Cookbooks and integrated with **Jenkins**.
* Integrated **GIT** with Jenkins using the **GitHub** plugin to automate the process of source code check-out by providing the URL and credentials of the GIT repository.
* Involved in setting up the configuration for projects to publish snapshot and release artifacts to **JFrog Artifactory** to share artifacts across projects and environments.
* Deployed and configured **Chef Server** and Chef Solo including bootstrapping of Chef client nodes for provisioning using a knife and automated by testing **Chef recipes, cookbooks** with test-kitchen, chef spec.
* Managed **Chef Workstation,** Bootstrapped instances using Chef and integrating with Auto Scaling. Converted production support scripts to Chef Recipes, additionally, created **Chef Cookbooks** for virtual and physical instance provisioning, configuration management, patching and software deployment.
* Built and deployed Java code through **Jenkins** and done Code coverage analysis using **SonarQube**. Run quality checks using SonarQube & uploaded it to **Nexus** repository. Improved monitoring with **Nagios** and custom plugins. Developed utilities in Perl, including templates and configurations for Nagios and basic system configuration.
* Deployed and maintained deployments in clustering and high availability scenarios. Implemented scripting tools - Ruby as Chef DSL and Shell Scripting and on monitoring tools like **Nagios**.
* Worked on testing, evaluation and troubleshooting of **MongoDB** and **Cassandra NoSQL** database systems and cluster configurations to ensure high availability in various crash scenarios.

**Environment:** ***AWS, OpsWork, AWS Beanstalk, Kubernetes, Terraform, Chef, Chef Server, Chef Workstation, Chef Cookbooks, SonarQube, Nexus, Nagios, MongoDB, CassandraSQL.***

**AWS .NET Developer *(Apr 2013 to May*** ***2014)***

***Acuity Software Technologies, India***

**Key Responsibilities:**

* Involved in Requirement Gathering, Design and Deployment of the application using Scrum (Agile) as Development methodology.
* Developed & fixed issues in .NET batches. Changed Web UI code per business requirements.
* Involved in requirement analysis, impact analysis & design code fixes to maintain App.
* Implemented .NET framework, N-tier architecture, and Object-Oriented Methodologies for application development.
* Developed business objects and Data Transformation objects using Object Oriented Programming concepts and OOAD Principles.
* Implemented Web Services to send or retrieve data from different apps using SOAP protocol.
* Responsible for maintaining and expanding AWS (Cloud Services) S3 infrastructure using **AWS** (SNS, SQS)
* Involved in designing **Microservice architecture**, based services interacting through a combination of **REST** and leveraging **AWS** to build, test and deploy Identify Microservices.

**Environment: *Visual Studio, C#, WCF, ASP.NET, MVC, CSS3, HTML, XML, .NET Framework 3.5, MS Visual Studio 2008, AJAX, TFS, Kafka, WCF, web forms, JavaScript, XML, WPF, SQL Server 2005, T-SQL, Agile Methodology, AngularJS, DB, Entity framework, SQL Server 2012, T-SQL, N-Unit, Windows, Kafka.***