Bhanu V

✉ bhanuv0507@gmail.com
☎ **+1 (919) 797-9651**
in www.linkedin.com/in/bhanuv7

Sr. Cloud/DevOps Engineer

Accomplished Cloud/DevOps Engineer with 8+ 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. Skilled problem identifier and troubleshooter. Comfortable in managing systems, projects, and teams in a wide range of IT environments.

Professional Summary:

- Experienced in designing, configuring and deploying solutions on Windows Azure for applications (Including Compute, Storage, IaaS, ARM templates) and in **Azure IaaS -** Virtual machines**, VM Scale Sets**, **Load Balancer**, **Traffic Manager**, Virtual Networks, SQL, Resource Groups and **Cloud Services**. Experienced in **Azure Backup**, **Azure Policies**, **Azure Key Vault**, **Salt Stack** and Tagging.
- Experienced in **migrating** on premise storage to **Microsoft Azure** using **Azure site recovery**, **Azure backups** and deployed **Azure IaaS virtual machines** and **cloud services (PaaS role instances)** into secure **Vnets and subnets** with **Azure Internal Load balancers.**
- Good Hands-on experience using **Azure service fabric** to package, deploy, and manage scalable and reliable **microservices** and **containers**. Also experience in developing different types of **Azure** Functions such as **Http trigger, Timer trigger, service bus trigger, Event-Hub trigger.**
- Expertise in deploying microservices, application development, and upgradation using AWS/Azure services such as **Azure DevOps, Kubernetes Service (AKS/EKS)**, **Container Services**, **MySQL DB**, and **Grafana, Azure pipelines**, **RBAC**, and monitoring clusters.
- Experienced in migrating on-premises, **.Net** apps, DevOps platform to Azure CI/CD processes by utilizing **ARM templates**, **Azure DevOps**, **Azure CLI**, **App services** such as Repos, Pipelines, **Web Apps**, **PowerShell** and to monitor applications with Application Insights.
- Expertise on **Amazon Web Services (AWS)** including **EC2**, **Elastic Load-balancers**, **Elastic Container Service**, **S3**, **Elastic Beanstalk**, **CloudFront**, **Elastic File system**, **Lambda, Elastic Cache, DMS, VPC, Direct Connect, Route53, CloudWatch, CloudTrail, CloudFormation, IAM, EMR, Elasticsearch.**
- Experienced in Integrating **Spinnaker** across **AWS EC2** instances and in multiple **AZ** of the **AWS VPC's** to ensure Continuous Delivery, monitor application deployments.
- Designed AWS **Cloud Formation** templates to create custom sized **VPC**, **Subnets**, and **NAT** to ensure successful deployment of Web applications, database templates and expertise in architecting **secure VPC** solutions in AWS with the help of **Network ACLs**, **Security groups**, **public** and **private network** configurations.
- Expertise 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.**
- Extensively used **Auto Scaling** for launching cloud instances while deploying **Microservices** and used **Terraform** to map more complex dependencies and identify network issues.
- Performed automation tasks on various Docker components like **Docker Hub**, **Docker Engine**, **Docker Machine** and **Docker Registry**. Worked with **Docker images & Docker Container**.
- Experienced in automating deployment of microservices on **Docker Swarm Cluster** by pulling **Docker images** from the private **Docker Registry,** Docker Trusted Registry. Proficient in **Networking Docker Containers** including Bridging, Routing, and troubleshooting networking issues in the Docker system.

- Experienced in Building and deploying the application code using **CLI** of **Kubernetes** called **kubectl**, **kubelet**, **kubeadm, kubespray** and Schedule the jobs using **Kube-scheduler**. Managed **K8s** charts using **Helm** and created reproducible builds of the **Kubernetes** applications.
- Hands on Experience in Troubleshooting the **Kubernetes Pod** logs, **Kube2IAM** logs and debugging pod restart and **CrashLoopBackoff** debug and resolving all those in production.
- Extensively involved in setting up **Kubernetes** (**k8s**) clusters for running microservices and implemented a

Kubernetes Container Orchestration solution within **OpenStack** allowing for easy management, creation & recovery of **OpenStack** assets and managed the container using **Docker,** set up the automated build on **Docker Hub** and run multiple Applications.

- Experienced in **Bluegreen Deployment and Canary Deployment, Octopus deployment** in the production. Managed Kubernetes cluster in the rollback and rollouts methods in the deployment strategies. Integrated **Istio** and **helm** packages with Kubernetes clusters for the service mesh.
- Good understanding of **OpenShift** platform in managing **Docker Containers, Kubernetes Clusters** and implemented a production ready, load balanced, highly available, fault tolerant **Kubernetes** infrastructure and created **Jenkins** jobs to deploy applications to **Kubernetes** Cluster.
- 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**.
- Proficient in configuration management tool **Ansible, Tower, Playbooks, Inventory files** to setup servers, deploying software, orchestrate fault-tolerance updates and delivery incorporating **Ansible Tower** for access control.
- Extensively worked on **Chef** Roles, **Cookbooks**, **Recipes**, Templates, Resources, Attributes & Data bags. Proficient in the setup of **Chef Servers**, **workstations** & **bootstrapping** infrastructure **Chef Nodes** for configuration management.
- Hands-on experience in using **Chef** for automation, provisioning and configuration management on enterprise Linux distributions, **Artifactory**, and **Nexus** repository by using ecosystem integrations with **AWS**, **Azure**, and **Docker.**
- Expertise in Installing, Configuring and managing the **Puppet Ecosystem** Components for both opensource and enterprise version of **Puppet** and involved in Designing and Implementing **Puppet** with **Orchestration** and code manager with **Git, GitLab** and **Git Hub**.
- Experience in working on version control systems like **GIT** and used Source code management client tools like **Git Bash**, **GitHub**, **BitBucket** and other command line applications.
- Implemented & maintained the branching and **build/release** strategies using **SCM** tools **GIT, GitLab, Subversion, Bitbucket** and experienced in **migrating Bitbucket repositories to GitHub.**
- Extensively experienced in using build **automation** tools like **Maven, Gradle** and **ANT** for the building of deployable artifacts such as jar, war and ear from source code to write Pom.xml, Build. Gradle and Build.xml respectively.
- Hands on experience in using **JIRA** as bug tracking system. Configured various workflows, customizations and plug-ins for **JIRA** bug/issue tracker and integration of **Jenkins** with **Jira/GitHub** to track change requests, bug fixes, manage tickets for corresponding **Sprints.**
- Proficient in Implementing relational Databases **NOSQL Database** Management systems like **MySQL, MSSQL, Oracle, PostgreSQL**, **Cassandra DB** and **Mongo DB.**
- Administered various flavours of Linux (RHEL, CentOS, Ubuntu, Solaris, Ubuntu, Fedora) and worked on Logical Volume Manager (LVM), Kickstart, Bonding, LAMP and LDAP.

Education Details:

- **Master's in Computer Science (MS)**
  University of Missouri
  Kansas City, USA.
- **Bachelor of Technology (B. Tech) in Mechanical Engineering**
  KL University, India.

Certifications:

- **AWS Certified Solutions Architect– Associate**
- **Certified Kubernetes Administrator**
- **Certified Azure Administrator**
- **Certified Terraform Associate**
- **Certified Azure DevOps Engineer**

Technical Skills:

Cloud	Amazon Web Services, Microsoft Azure, OpenStack
Container/ Orchestration Tools	Kubernetes, Docker, OpenShift
CM & Deployment Tools	Chef, Ansible, Puppet, Terraform.
CI/CD Tools	Jenkins, Bamboo, TFS, Azure DevOps.
Build Tools	Maven, Ant, Gradle, InstallShield, MS Build.
Repositories	Artifactory, Nexus.
Monitoring Tools	JIRA, Prometheus, Splunk, Grafana, Logstash and Kibana (ELK)
Version Control	GIT, Subversion, CVS, Clear Case, Bitbucket
Programming/ Scripting Languages	Shell, Python, Ruby, Groovy, Bash, YAML, JSON, XML, PowerShell.
Web/Application Servers	Tomcat, Web logic, Web Sphere.
Database	SQL Server, No-SQL Database, My SQL, DB2
Operating Systems	Windows, LINUX, RHEL, MAC.

Professional Experience:

Goldman Sachs, Dallas, TX ***Apr2022 - present***

Role: Sr. Cloud Infrastructure/DevOps Engineer

Responsibilities*:*

- Worked with **Terraform** for automating **VPC's, ELB's**, **Security groups**, **SQS queues**, **S3** buckets and continuing to replace the infrastructure as a code from scratch and Created **Terraform** Scripts for EC2 instances, **Elastic Load balancers** and **S3 buckets**.
- Built servers using AWS, importing volumes, launching **EC2**, **RDS,** creating **security groups, auto-scaling**, **load balancers (ELBs)** in the defined virtual private connection (VPC), provided high availability of applications of EC2 instances by taking the advantages of AWS **High availability** options.
- Managed **IAM policies**, providing access to different AWS resources, design and refine the workflows used to grant access.
- Created **AWS S3 Buckets**, Policies and on the IAM Role based polices customizing the **Terraform** templates.
- Implemented and maintained the monitoring and alerting of production and corporate servers/storage using **AWS Cloud watch**.
- Involved in setting up and operating **AWS RDS** & **NoSQL Database DynamoDB** services.
- Designed **Terraform templates** to create custom sized **VPC, subnets, Security groups** to ensure successful deployment of Web applications.
- Used AWS **Route53** to route the traffic between different availability zones. Deployed and supported Mem-cache/AWS Elastic-Cache and then configured **Elastic Load Balancing (ELB)** for routing traffic between zones.
- Implemented serverless applications on **CloudFront, API gateway, AWS Lambda, SNS** and created **AWS Lambda** functions to run scripts in response to events in Amazon DynamoDB table and S3 bucket to HTTP requests using Amazon **API gateway.**
- Established connectivity to **AWS** resources in another **VPC** in another region using **AWS VPC peering** and strong swan/IPSEC tunnel.
- Implemented a production ready, **load balanced**, highly available, **fault tolerant**, **auto scaling AWS infrastructure** and **Docker container orchestration.**
- Managed **GIT** repositories for branching, merging, and tagging for **DEV, QA UAT and Prod** environment.
- Extensively worked on **GitLab CI/CD pipeline** jobs for end-to-end automation to build, test and deliver **artifacts** and troubleshoot the build issues during the process.
- Defined and managed release planning and branching strategy of **GIT** servers. Coordinated with developers for establishing and applying appropriate branching, labelling/naming conventions using **GIT** repos.
- Written **Bash** and **Python** scripts, to supplement automation provided by **Terraform** for tasks such as encrypting **Amazon EBS (Elastic Block store)** volumes backing **AMI's** and Scheduling **Lambda functions** for routine AWS tasks.

- Created **JIRA** issues to **prioritize** and act on what's important and stayed up to date with what's going on around the **project**. Involved in setting up **JIRA** as defect **tracking** system and **configured** various **workflows**, customizations and plugins for the **JIRA bug/issue tracker.**

Environment: Amazon Web Services (AWS), Amazon S3, GitLab, Terraform, Amazon RDS, Chef, Docker, Ansible, Tomcat, Kubernetes, Jira, RHEL 5.x/6.x, YAML, shell, Python.

Johnson & Johnson IT Services, Raritan , NJ ***Apr2021 – Apr2022***

Role: Sr. Cloud Infrastructure Engineer

Responsibilities*:*

- Worked on various **Azure services like Compute (Web Roles, Worker Roles), Azure Websites, Caching, SQL Azure NoSQL, Storage, Network services, Azure Active Directory, API Management, Scheduling, Auto Scaling, and PowerShell Automation.**
- Responsible for creating and managing **Azure AD tenants**, manage users and groups, and configure application integration with **Azure AD**. Integrate on-premises **Windows AD with Azure AD**, configure **multi-factor authentication (MFA)** and federated single sign-on (SSO).
- Created **AZURE Resource Manager** (ARM) Templates to deploy multiple resources and extensive experience in designing custom build steps using **PowerShell** scripts and **ARM templates** to automate the provisioning and deployment process.
- Migrated data from on-premises **SQL Database servers** to **Azure SQL Database servers** sent by designing the **Azure Data Factory Pipelines** using the **Azure Data Factory** copy tool and Self-Hosted Runtimes.
- Configured and deployed **Azure Automation Scripts** for a multitude of applications utilizing the **Azure stack** for Compute, Web and Mobile, Blobs, Resource Groups, **Azure Data Lake**, **HDInsight Clusters**, **Azure Data Factory**, **Azure SQL**, Cloud Services, and **ARM** Services.
- Worked with **Version Control, Build** & **Release** Management and Deployments of the Solutions to the **DEV**, **QA & PROD** Environments leveraging Azur**e DevOps/VSTS** principles/process (CI/CD) and toolsets of Visual Studio, AKS **(Azure Kubernetes Service**), Application Insights, Log Analytics.
- 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**.
- Responsible for creating documentation of the **Terraform** infrastructure in Confluence, and used **Terraform.**
  to manage the cloud infrastructure and managed servers using configuration management tools
  like **Ansible** and **Chef.**
- Configured **Network Security Groups** to filter the traffic to and from **Azure** Resources by configuring the Inbound and Outbound traffic rules and associating them with **Subnet** and **Network Interfaces**, creating **Azure Key Vault** to store Certificates, Secrets.
- Developed **Ansible Playbooks** for automating the infrastructure, deployment process. Managed build and deployment scripts using **YAML** in Ansible, triggered the jobs using **Jenkins** to move from one environment to across all the environments.
- Involved in writing various custom **Ansible playbooks** for deployment **orchestration** and developed **Ansible Playbooks** to simplify and automate tasks. Protected encrypted data needed for tasks with **Ansible Vault.**
- Configured **Kubernetes** Replication controllers to allow multiple pods such as **Jenkins** Master server in multiple minions. Managed **Kubernetes** charts using **Helm**, managed **Kubernetes** manifest files and created reproducible builds of the **Kubernetes** application.
- Worked with Scheduling, deploying, managing container replicas onto a node cluster using **Kubernetes** and created **Kubernetes** clusters work with frameworks running on the same cluster resources.
- Worked on version control systems like **GIT** and used Source code management client tools like **Git Bash**, **GitHub**, **Bitbucket** and other command line applications.
- Administered and Engineered **Jenkins** for managing nightly Build, Test and Deploy chain, **GIT** with Development/Test/Production Branching Model for weekly releases.
- Used **JIRA** for ticketing and as well as defect tracking system and configure various workflows, customizations and plug-ins for **JIRA bug/issue tracker**, integrated Jenkins with JIRA, GitHub, Maven.

- Took lead on On-boarding, set-up, installation and maintenance for **Grafana** as a Monitoring tool and to collect the Metrics.
- Extensively worked on **Jenkins CI/CD pipeline** jobs for end-to-end automation to build, test and deliver **artifacts** and troubleshoot the build issue during the **Jenkins build** process.
- Created various dashboards, metrics, alarms and notifications for servers using **Grafana**, **Prometheus**.
- Performed installation, configuration, upgrades, Package administration and support for **Linux** systems on the client side using **RedHat satellite network server** and worked on patches installation, patch upgrades on **Red Hat Linux servers.**

Environment: Azure DevOps, Kubernetes, Docker ACS & AKS, Prometheus, Splunk, Terraform, Ansible, Jenkins, Git, Azure Boards, JIRA, Grafana, Python, PowerShell, YAML, Docker, Visual Studio Code, Shell, Nginx, Linux, Windows servers 2012/2016

CGI Inc, Raleigh, NC ***Dec 2018 – March 2021***

Role: Sr. DevOps Engineer

Responsibilities*:*

- Configured **Azure Virtual Networks**, **subnets, DHCP address blocks, Azure network settings, DNS settings, security policies and routing.** Also, deployed **Azure IaaS** virtual machines and Cloud services (PaaS role instances) into secure **Virtual Networks and** subnets**.**
- 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 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. Configured **Prometheus** to track performance and maintain security, and used **Grafana** tool to collect metrics, queries and logs.
- 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.
- Configured **Network Security Groups** to filter the traffic to and from **Azure** Resources by configuring the Inbound and Outbound traffic rules and associating them with **Subnet** and **Network Interfaces**, creating **Azure Key Vault** to store Certificates, Secrets.
- Deployed two tier **.NET**, Python applications to **Azure DevOps** to focus on development by using services such as **Repos** to commit codes, Build Pipelines using **PowerShell** scripts for CI/CD and Test plans for unit test, deploy to **App Service**, **Azure Application Insights** collects health performance and usage data of the process, stored artifacts in blob storages.
- Worked with **Version Control, Build** & **Release** Management and Deployments of the Solutions to the **DEV**, **QA & PROD** Environments leveraging **Azure DevOps/VSTS** principles/process (CI/CD) and toolsets of Visual Studio, AKS **(Azure Kubernetes Service**), Application Insights, Log Analytics.
- 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 project.
- 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.
- Evaluated **Kubernetes** for **Docker** container orchestration. Managed **Kubernetes** charts using Helm and created reproducible builds of the **Kubernetes** applications, templatize **Kubernetes manifests**, provide a set of configuration parameters to customize the deployment and Managed releases of **Helm** packages.
- Created **pods** and services using Master and Minion architecture of **Kubernetes** and developed microservices build by integrating **Jenkins** with **Docker** container using **Cloudbees, Docker** pipeline plugin, leveraging **Python** allowing for creation and maintenance build jobs and used **Kubernetes** to deploy services.
- Worked on **Kubernetes** to manage containerized applications using nodes, **Configuring Maps**, selector services and deployed application containers as **Pods**. In order to provide **secured network** connectivity to the containers used **Calico as CNI**. Responsible for build and deployment automation using **Docker**, **Kubernetes** containers.
- Configured **Ingress** and **Egress** rules to control inbound and outbound traffic on to the **Kubernetes** clusters.

- 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.
- Managed to 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.**
- Focused on architecting **NoSQL databases** like **Mongo**, **Cassandra** and **Cache** database. Worked on upgrading and migrating various versions of **Mongo** and **Cassandra databases** on different platforms.
- Created dashboards and visualizations using **Splunk, Grafana and Nagios** for performance and activity monitoring and setting up **Splunk** to capture and analyze data from various layers Load Balancers, Webservers and application servers.
- Involved in developing custom scripts using **Python, Shell** to automate the deployment process and for Task scheduling, Systems backups for **RHEL**.

Environment: Azure DevOps, Kubernetes, Docker ACS & AKS, Prometheus, Splunk, Terraform, Ansible, Jenkins, Git, Azure Boards, JIRA, Grafana, Python, PowerShell, YAML, Docker, Visual Studio Code, Shell, Nginx, Linux, Windows servers 2012/2016

Netfective Technology, Dallas, TX ***May 2017 – Dec 2018***

Role: DevOps/Cloud Engineer

Responsibilities:

- Involved in designing and deploying a multiple of applications utilizing almost all the **AWS** stack including **EC2**, **Route53, S3, RDS, Dynamo DB, SNS, SQS, Lambda, RedShift,** focusing on high-availability, fault tolerance and auto-scaling in **AWS** using **Cloud Formation**.
- Designed and implemented scalable, secure cloud architecture based on **Amazon Web Services**. Leveraged AWS cloud services such as **EC2**, **Auto-scaling Groups**, and **VPC (Virtual Private Cloud)** to build secure, highly scalable and flexible systems that handle expected and unexpected load bursts and can quickly evolve during development iterations.
- Designed roles and groups for users and resources using **AWS** Identity Access Management (**IAM**) and managed **IAM** account (with **MFA**) and **IAM** policies to meet security audit & compliance requirements and also managed network security using **Security Groups** and **IAM**.
- Created **AWS** launch configuration based on customized Amazon Machine Image (**AMI**) and used this launch configuration to configure auto scaling groups and created **AWS Route53** routing traffic between different AWS regions.
- Worked at optimizing volumes and **EC2 instances** and created multiple VPC instances, and on **IAM** to create new accounts, roles and groups and created alarms and notifications for **EC2** instances using **Cloud Watch**.
- Designed **AWS Cloud Formation** templates to create custom sized **VPC**, **subnets**, **NAT** to ensure successful deployment of Web applications and database templates, **Cloud Watch**, Cloud Trail and Cloud front to set up and manage the cached content delivery.
- Configured **S3** versioning and lifecycle policies and backup files and archive files in **Glacier** and Creating **Lambda** function to automate snapshot back up on AWS and set up the scheduled backup.
- Created **Ansible** Playbooks with Python SSH as the Wrapper to Manage Configurations of **AWS Nodes** and Test Playbooks on AWS instances using **Python**.
- Used **Docker microservice** and automated deployment **pipelines** and built jobs for web-mobile user application services through **CD/CI** pipelines on **AWS EC2** with **ECS, ECR**. Deployed web-mobile user application **microservices** using **Node.js, Ruby, Python** on **AWS Elastic Beanstalk** with **AWS S3**, **PostgreSQL RDS,** and **VPC**.
- Implemented a **Kubernetes Container Orchestration** solution within **AWS** leveraging **KOPS**, which allows easy management creation and recovery of **AWS assets.**
- Configured cluster operations in **AWS Kubernetes (EKS)** to deploy **microservices** with **CI/CD** system and used **Kubernetes** cluster to scale up operations of clusters, maintain the cluster services, load balancing, network policies, group **Docker** containers across different platform.

- Implemented **Docker -maven-plugin** in and **Maven POM** to build **Docker Images** for all **microservices** and later used **Docker** file to build the **Docker Images** from the **java jar files.**
- Used **Jenkins** for Continuous Integration and deployment into **Tomcat** Application Server and used **Jenkins** AWS Code Deploy plug-in to deploy to **AWS**.
- Used **Ansible** as the Configuration Management tool and written **Ansible** Playbooks to automate the process of creating the Master and Worker nodes in Kubernetes Environment and to install packages from repository, to change the configuration of remotely configured machines and to deploy new builds and configured **Ansible** playbooks with **Ansible** Tower.
- Used **Ansible** to **Setup/teardown** of **ELK** stack (Elasticsearch, Log stash, Kibana) and **troubleshoot** the **build issues** with **ELK** and work towards the solution.
- Written **Chef** Recipes to automate our build/deployment process and do an overall process improvement to any manual process. Setup and maintained automated environment using **Chef** Recipes within AWS environment. Good understanding of **Knife, Chef Bootstrap** process.
- Created **ELK Dashboards** and Parsing logs in Graylog and configured **Elastic search, Logstash and Kibana (ELK)** for **log analytics**, full text search, application monitoring in integration with **CloudWatch**.
- Used tracking tool **JIRA** and triaged the issues and maintained bugs using **JIRA tool**. Maintained JIRA for tracking and updating project defects and tasks ensuring the successful completion of tasks in sprint.
- Maintained the **RedHat** servers for infrastructure management to keep **RHEL** environments and other **RedHat** infrastructure running efficiently, properly secured, and compliant with various standards.

Environment: Amazon Web Services (AWS), Amazon RDS, GitHub, Terraform, Jenkins, Chef, Docker, Ansible, Tomcat, Kubernetes, Kibana, Jira, RHEL 5.x/6.x, YAML, shell, Python.

DBASE Infosolutions , AP, India ***May 2015 - Dec 2016***
Role: ETL / Power BI developer

Responsibilities:

- Created a data model which holds data from multiple business systems and created Tables, Views, Triggers, Stored Procedures using **T-SQL**.
- Created **ETL** packages to transform the data from multiple sources to align with current data model.
- Deployed the **ETL** packages to **Azure Data Factory (ADF)** and created pipelines to run on scheduled timings.
- To automate the process of uploading manual Excel files from **SharePoint** to **SQL Server**, connected Power BI to manual excel files and used the Data Model as a source in **SSIS**.
- Published the reports to Power BI Service and Created a Microsoft Flow to update the Power BI dataset whenever the manual file is updated and created a Common Data model for all the reports to reduce the Memory Usage and ability to run the reports faster within the Service.
- Created Power BI reports by connecting to Common Data Model with Power BI Dataset as source and used bookmarks to create an app like view within the reports.
- Created Parameters within the report to reduce the effort on changing the Data Source environment during deployment.
- Created Dataflows within the Power BI Service to refresh individual dataset or table within the Common Data Model based on the source data refresh.
- Created User **X-references tables** within the Data Model to support the dynamic Row Level Security **(RLS)** based in the User login.
- Maintained the Checking in and out of **ETL** Codes and reports within **VSTS** and monitored the performance of the reports by using variables within the **DAX** functions.
- Monitored the health metrics of the App and maintained the Capacity, Workspace and user accesses.
- Created a POC for Tabular **SSAS** cube model to present it to the stakeholders to show the reports efficiency and improved Row Level Security **(RLS).**

Environment: Excel, SharePoint Online, Power BI Desktop, Power BI Service, SQL Server Integration Service, SQL Server Management Studio, SQL Server Data Tools, SQL Server 2016, VSTS and Microsoft Flow.

Educomp Solutions Ltd, Bangalore , India ***May 2014 – Apr 2015***

Role: System Administrator

Responsibilities:

- Involved in configuration, installation, implementation, maintenance and support for the **Linux servers, RHEL**, **CentOS, Ubuntu, Solaris, Fedora**.
- Installed operating system on multiple machines using **Kick Start** and Performed server updates, patching, upgrade and package installations using **RPM** and **YUM.** Installation of **SSH** and configuration of keys bases authentication.
- Configured and maintained **NFS, LDAP, HTTP**, as well as **DNS** on Linux servers. Monitoring and managing performance of **ESX servers** and **Virtual Machines**.
- Worked with various storage volume managers such as **LVM, SVM, VERITAS** volume manager to create disk groups, volume groups and used **RAID** technology for backup and recovery.
- Administered Linux servers for several functions including **Apache/Tomcat server**, mail server and **MySQL** databases in both development and production.
- Wrote **bash shell scripts** for automation of daily tasks and involved in development of automation scripting to deploy some applications.
- Experience in using monitoring tools such as **Nagios** for general disk space usage and conduct systems performance monitoring and tuning.

Environment: Red hat Linux, Unix, Centos, Ubuntu, Solaris, Fedora, Veritas Volume, Manger, MySQL, Oracle, TCP/IP, Bash shell, NAGIOS, Python, Perl, LDAP.

.