

Vamshi

Sr. Cloud / DevOps Engineer



+1 (661) 221-5025

vamshi.0517@outlook.com

www.linkedin.com/in/vamshi-t0517

Professional Summary

A Techie with 7+ years of experience in IT industry in various roles as AWS/Azure DevOps engineer with proven expertise in designing and building highly-available, scalable, robust and cost effective cloud infrastructures, automating cloud with continuous integration and deployment (CI/CD) pipelines, orchestration, containerization and configuration management as per the client requirements.

- Highly proficient in designing, configuring and managing Public/Private cloud infrastructures utilizing **EC2, Elastic Container Service (ECS), S3, Elastic Beanstalk, CloudFront, Elastic File System (EFS), RDS, DynamoDB, Direct Connect, Route53, Redshift, CloudWatch, CloudTrail, CloudFormation, IAM, SQS, SNS, Elastic Transcoder and Elasticsearch.**
- Experience in Building and configuring a virtual data centers in the Amazon Web Services (AWS) cloud to support Enterprise Data Warehouse hosting including **Virtual Private Cloud (VPC), Public and Private Subnets, Security Groups, Autoscaling, Route Tables, Elastic Load Balancer(ELB).**
- Well-versed in building and converting existing AWS infrastructure to **serverless** by using (**AWS lambda, Kinesis, API Gateway, DynamoDB, and S3 bucket**) and deploying via **Terraform.**
- Expertise with Cloud automation technologies such as **Cloud Formation** and **Terraform** for building, changing, and versioning infrastructure and manage existing and cloud infrastructure through **Terraform scripts and Terraform modules.**
- Good Knowledge on integrating **Terraform** with **Ansible, Packer** to create and Version the AWS Infrastructure, designing, implementing and automating and sustainment of **Amazon machine images (AMI)** across the AWS Cloud environment
- Skillful experience in implementation, architecture, and design with a strong background in Cloud **IaaS/SaaS, Private/Public clouds.**
- Experience in Azure Deploying, Configuring, Automating Instances, analysis of Azure networks with **Azure network watcher, Azure VPN gateway, notification hub,** migration of on-premises environment to **Windows Azure using Azure Site Recovery and Azure backups.**
- Well-Versed in managing **Azure infrastructure, Azure storage accounts** and deploying workloads on **Azure virtual machines (VMs), virtual networks, cloud services, resource groups, express route, traffic manager, VPN, load balancing, application gateways, auto-scaling and cloud services (PaaS role instances)** into secure **V-Nets and subnets.**
- Familiar with toolset of **VSTS, AKS (Azure Kubernetes), Office 365, Application Insights, Log Analytics, DHCP address blocks, security policies and routing,** provisioning of **IaaS, PaaS Virtual Machines,** Worker roles on **AZURE Classic and Azure Resource Manager (ARM),** Deployed Web applications on Azure using **PowerShell Workflow.**
- Good Knowledge on **Docker components** like **Docker Engine, Docker-Compose, Docker-Hub, Docker Registry, Docker container snapshot** and familiar with setting up docker environment **Docker Daemon, Docker host and client.**
- Expertise in designing and deploying container-based production, **Orchestration** and Scheduling using tools such as **Kubernetes(k8's), Docker Swarm** and automation validation using **Test Kitchen, Chef, Ansible and Terraform.**
- Profound experience in building and deploying the application code using CLI of **K8s** called **KubectI, Kubelet, Kubeadm, Kubespray and Schedule the jobs using KubeScheduler.**
- Proficient in deploying **Kubernetes Cluster** with **master/slave architecture** and wrote **YAML** files to create many services like **Pods, deployments, auto-scaling, elastic load balancers, labels, health checks, Namespaces, Config Map.**
- Experience in managing **Kubernetes (k8s) charts using Helm.** Created reproducible builds of the **Kubernetes applications,** templated **Kubernetes manifests,** and provided a set of configuration parameters to customize the deployment.
- Hands-on experience on **Ansible server** to automate repetitive tasks, to deploy critical applications quickly, and proactively manage the changes.

- Familiar with **Ansible Vault**, orchestrate **zero downtime rolling updates** and used **Ansible Tower** to manage **Multiple Nodes** and Manage Inventory for different Environments through easy to use dashboard and **role based access control**.
- Good exposure on all areas of **Jenkins** like **Plugin Management, Securing Jenkins, Performance issues, Analytics, Scaling Jenkins, integrating Code Analysis and Test Phases to complete the CD pipelines within Jenkins**.
- Expertise in delivering **Agile** projects from end to end automation i.e. **Continuous Integration/Continuous Delivery (CI/CD) pipeline** by integration of tools like **Jenkins, Chef** and **AWS** for **VM provisioning**.
- Skilled in monitoring servers using **Nagios, Splunk, ELK, Datadog, AppDynamics, New Relic** and **CloudWatch** for Resource, **Network** and **Log Trace Monitoring**, created **geo-mapping** visualizations using Kibana in integration with **AWS CloudWatch** and **Lambda**. **Evaluated system logs** using **ELK software stack**.
- Extensively worked on source code repository **GitHub** and integrate it to **Jenkins** for automating the builds and have implemented a headless protocol for pushing and pulling changes from **Jenkins** to **JFrog repository** and **Nexus Repositories**
- Proficient in Linux/Unix(Ubuntu, Red hat, CentOS) utilities like **KICKSTART, SSH, FTP, AUTOFS, Quota Management, wrappers, User Management, Process Management, Memory Management, and Text processing tools, Log Management, Package Management, Disk Management and Swap Partition/Files Management**.
- Build **Automation** and **standardization tools** like **ANT, Maven, Gradle** and working experience in authoring **POM.xml** files and **automatically compile, package, test, build and deploy** applications to **Web Sphere** and **JBoss Application Server**.
- Skilled on Python, Bash/Shell, PowerShell, Ruby, Perl, JSON, YAML, Groovy. Developed Shell and Python Scripts used to automate day to day administrative tasks and automation of the build and release process.

Technical Skills

Cloud Technologies	Amazon Web Services (AWS), Azure, OpenStack, GCP
Containerization/Orchestration Tools	Docker, Docker Swarm, Kubernetes, Azure Kubernetes Services (AKS), Elastic Kubernetes Service (EKS), Amazon EC2 container service (ECS), Azure container service (ACS), OpenShift
CI/CD Tools	Jenkins, Hudson, Bamboo
Version Control	GIT, GitHub, GitLab, SVN, Bitbucket, TFS/VSTS
Build Tools	Maven, Ant, Gradle
Languages	Python, JAVA, XML, HTML, YAML, Perl, shell/Bash, Ruby
Database	MySQL, MS SQL server, Oracle, DynamoDB, RDS, Cassandra DB
Operating Systems	Windows, UNIX, Linux (Redhat, CentOS, Ubuntu), Solaris
Web Servers	Nginx, Apache HTTP, Apache Tomcat, JBoss, IIS
Incident Management tool	Jira, ServiceNow, Redmine, Rally
Repository Management	Nexus, JFrog, Artifactory
Monitoring Tools	Nagios, Splunk, ELK, Datadog, New Relic

Certifications

AWS Solution Architect – Associate

Validation Number S1376J62GB44QLGN

Professional Experience:

Sr. Cloud/DevOps Engineer

CLIENT: KPMG, TEXAS, USA

DEC 2019 – TILL PRESENT

RESPONSIBILITIES:

- Experienced in migrating the resources from on premises environment to cloud by using various AWS migration services such as import and export, **SMS (Server Migration service)**, **AWS DATABASE Migration service**.
- Converted existing AWS infrastructure to server-less architecture (**AWS Lambda**) and deployed it via **Terraform**. Migrated data from Datacenters and on-premises to Cloud using AWS Import/Export **Snowball** service.
- Provisioned infrastructure using **Cloud formation** templates for custom sized **VPC**, **Subnets**, **EC2 instances**, **ELB**, **Security groups**. Worked on tagging standard for proper identification and ownership of **EC2 instances** and other AWS Services like **Cloud Front**, **Cloud Watch**, **RDS**, **S3**, **Route53**, **SNS**, **SQS**, **Cloud Trail**.
- Deployed configuration management and provisioning to AWS using **Packer** and **Terraform**, involved in auto-mated deployment of **EBS** on to the AWS Cloud EC2 instance server and automated the complete deployment environment on AWS.
- Worked on **AWS CloudWatch**, **CloudFormation**, **Cloud Trail** and **CloudFront** to setup and manage cached content delivery. Created **Elastic load balancers (ELB)** and used **Route53** with failover and latency options for **high availability & fault tolerance**.
- Automate AWS infrastructure through **infrastructure as code** by writing various **Terraform modules, scripts** by creating AWS **IAM users, groups, roles, policies, custom policies, AWS Glue, Crawlers, Redshift clusters, snapshots of clusters, EC2, S3**.
- For Zero Down-Time on Application, used **blue-green** deployments with Aws **Elastic Bean Stalk** and worked on migration services like **AWS Server Migration Service (SMS)** to migrate on-premises workloads to AWS in easier and faster way using **Rehost & Lift** and **Shift & methodology** and **AWS Database Migration Service (DMS)**, **AWS Snowball** to transfer large amounts of data and **Amazon S3 Transfer Acceleration**.
- Configured AWS **Virtual Private Cloud (VPC)** with **Public and Private Subnets**, **Routing Tables**, **Internet Gateway**, **Network Access Control Lists** and **Security groups** appropriately for internet facing web servers and secure app and database servers
- Configured and Maintained **OpenStack** Single node and Multi Node Clusters, also Worked on OpenStack Services such as **Nova** for Compute, **Neutron** for networking, **Cinder** for Block Storage, **Keystone** for Identity, **Glance** for Image, **Swift** for Object Storage, **Horizon** for Dashboard and **Heat** for **Orchestration**.
- Configured AWS Cloud IAC using **Terraform** and continuous deployment through **Jenkins** and automated the **Cloud formation** using **Terraform**. And created a **REST API** in **API Gateway** to proxy **Amazon S3** for creating and configuring a **REST API** to expose the various **S3** operations.
- Configured **CloudWatch** and **Datadog** to monitor real-time granular metrics of all the AWS Services and configured individual dashboards for each resource agents.
- Implemented cluster services using **Docker** and **Kubernetes** to manage local deployments in **Kubernetes** by building a self-hosted Kubernetes cluster using **Terraform** and **Ansible** in deploying application containers.
- Experience with **Docker components** like **Docker Engine**, **Hub**, **Machine**, **Compose** and **Docker Registry** and experience with **Docker container service** and Docker applications by creating **Docker images** from **Docker file** and running them to create the Docker containers and Docker consoles for managing the application life cycle.
- Created **Kubernetes(k8's)** clusters with objects like **Pods**, **Deployments**, **Services** and **ConfigMaps** and created reproducible builds of the Kubernetes applications, managed Kubernetes **manifest files** and **Helm packages**. In order to provide secured network connectivity to the containers we used **Calico**.
- Worked on **K8s** for container orchestration with **Kubernetes container storage**, automation to enhance container platform **multi-tenancy** also worked on with Kubernetes architecture and design troubleshooting issues and multi-regional deployment models and patterns for large-scale applications. Deployed Kubernetes templates for various applications like **Cassandra**, **Grafana**, **Kafka**.
- Automated repetitive tasks, deployed applications, managed changes and automated software update and Verified functionality using configuration management tool **Ansible**. Created active-active disaster recovery model for **Ansible Tower**.
- Worked with **Ansible** to manage **Web Applications**, **Config Files**, **Database**, **Commands**, **Users Mount Points** and **Packages**.
- Wrote **Ansible playbooks**, inventories created custom playbooks written in YAML language, encrypted the data using **Ansible Vault** & maintained role-based access control by using **Ansible Tower**. Applied IT **orchestration** using **Ansible** to run tasks in a sequence which can work on different servers.

- Integrated **Ansible** with **Jenkins** to provide **automation, continuous integration(CI)** through **Jenkins** and Implemented **Jenkins Workflow** and Plugins for repeatable deployments of **multi-tier applications, artifacts and services to Docker**.
- Designed and developed **Continuous Integration & Continuous Delivery(CI/CD) pipelines** using **Code Pipeline, Code Build** and **Code Deploy plugins** in **Jenkins**. Worked on **Jenkins CLI** to communicate with other resources and to write scripts for automation.
- Worked and managed extensively with **Chef Workstation, Chef Attributes, Chef Templates, Chef Recipes** and **Chef Files** for managing and **bootstrapping** the configurations across various **nodes, setting up keys and Created Chef Cookbooks** to install and configured Infrastructure across environments and automated the process using **Python Script**.
- Developed **Python, Perl** and **shell/bash** scripts for **Tomcat web application servers** and automation of the build and release process and build tools like **Maven**. Also Configured and maintained **Jenkins** to implement the CI process and integrated the tool with **Maven** to schedule the builds.
- Installed and configured **ELK stack(Elasticsearch, Logstash, and Kibana)** and developed **pipeline** for application logs from App Server to Elastic search through **Logstash** and used logs aggregation tool **ELK configuration** and **administration** for monitoring **EC2 instances**.
- Managed **ELK (Elastic Search, Log Stash & Kibana)** Stack to collect, search and analyze logfiles across **servers, log monitoring** and created **geo-mapping visualizations** using **Kibana** in integration with **AWS Cloud watch** and **Lambda**.
- Performed **Branching, Labelling, Merging** and other **SCM POLL** work on various Centralized and Distributed VCS and migration of **SVN** repositories to **GIT**, responsible for design and maintenance of the GIT Repositories, views, and the access control strategies and used sub modules in **GIT**.
- Used **Git** for **source code version control** and integrated with **Jenkins** for **CI/CD pipeline**, code quality tracking and user management with build tools **Maven** and written Maven **pom.xml** build script. Worked with **Radius packages** for authentication.

Tools and Technologies: AWS (EC2, S3, EBS, VPC, Elasticsearch, Route53 (DNS), API Gateway, Lambda, CloudFormation, DynamoDB, RDS, RedShift, API's, REST API's, SMS, SES, SQS, OpsWorks), Radius Packages, Docker, Ansible, Chef, Maven, Jenkins, GIT, Bitbucket, Python, Linux, Jira, Elasticsearch, Logstash, Kibana (ELK).

Azure/DevOps Engineer

CLIENT: MAX FINKELSTEIN, NEW YORK

SEP 2018 –NOV 2019

RESPONSIBILITIES:

- Implemented **Azure Hybrid integration** to integrate the applications, data and processes using API's and Worked with **Azure AD, Azure Blob Storage and Data Factory** to compose data storage and processing micro-services into automated data pipelines and used **Azure compute services and Azure Web apps**.
- Configured and deployed **Azure Automation Scripts** for a multiple application utilizing the **Azure stack** (Including **Compute, Web & Mobile, Blobs, ADF, Resource Groups, Azure Data Lake, HDInsight Clusters, Azure Data Factory, Azure SQL, Cloud Services, and ARM**), Services and Utilities focusing on Automation.
- Configured **Site to Site VPN Connectivity**. Configured Window Failover Cluster by creating **Quorum** for File sharing in **Azure Cloud**. Configured **VNet-to-VNet** connections between deployment models.
- Designed capacity and architecture plan to create the **Azure Cloud environment** to host migrated **IaaS VMs** and **PaaS role instances** for refactored applications and databases. Deployed infrastructure as a code (IaC) application using **ARM Templates (JSON)**.
- Managed the **rate limit** by the requests which will quickly cause **Azure DevOps** Services to enforce rate limits as per organization. To reduce the risk of hitting the rate limits, we followed the best practices while using the **REST APIs** to integrate with **Azure DevOps Services**.
- Involved in Backing up and Recovering an **Azure Virtual Machine** from a **Recovery Services Vault** and **Azure Site Recovery** Services. Have a good knowledge on **Express Route, Network Security Groups, Storage Accounts, Storage Explorer** and **Azure Architecture**.
- Designed the data model and created the schema on **SQL Azure** and configured **Azure Cosmos DB** and exposed the service as Web API. Worked on Configuration **Internal Load Balancer, Load Balanced Sets** and **Azure Traffic Manager**.

- Managed **Nagios** as a **monitoring tool** to identify and resolve infrastructure problems before they effect critical processes and worked on **Nagios** Event handlers in case of an automatic restart of failed applications and servers.
- Designed templates for **Azure Infrastructure-as-a-code** using **Terraform** to build staging and production environments. Integrated **Azure Log Analytics** with **Azure 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 App **Containerization** technology **Docker**, creating **Docker images, Containers, Docker Registry** to store images, cloud-based registry like **Docker Hub, Docker Swarm** to manage containers **Registry** and handling multiple images primarily for middleware installations and domain configurations.
- Implemented cluster services using **Docker** and **Kubernetes** to manage local deployments in Kubernetes by building a selfhosted Kubernetes cluster using **Terraform** and **Ansible** in deploying application containers.
- Configured applications that run **multi-container Docker applications** by utilizing the **Docker-Compose** tool which uses a file configured in **YAML** format. Used **Kubernetes** to manage containerized applications using its **nodes, Config-Maps, selector, Services** and deployed application containers as **Pods**.
- Developed **Terraform Templates** for provisioning and automating **Azure IAAS virtual machines, Virtual Networks, VM scale sets, Load Balancer and NAT rules** using **terraform modules** and also used terraform graph to visualize execution plan using graph command.
- Played a crucial role in Migrating the Legacy application into **Google Cloud Platform(GCP)** and Responsible for Deploying **Artifacts** in the **GCP platform** by using **Packer**. Worked on **Stack Driver Monitoring** in **GCP** to check the alerts of the applications that run on the **Google Cloud Platform** and deploying on GCP using **Google Cloud Deployment Manager**.
- Designed **GCP Cloud Formation templates** to create custom sized **VPC, subnets, NAT** to ensure successful deployment of **Web applications** and database templates. Created **ETL scripts** that extracted data from **Mix panel (MP)** using MP's API scripted in **Google Cloud Shell** and converted to **Google BigQuery** tables for the Data Analytics teams.
- Implemented the **Chef Software** setup and configuration of **VM's** from the scratch and deployed the run-list into **chefserver** and **bootstrap** the chef clients remotely.
- Written multiple **cookbooks** in Chef. Implemented environments, roles, and data bags in Chef for better environment management. Worked on **Chef DK** which takes of care creating **cookbooks** and **recipes**.
- Developed **Ansible Playbooks** to manage **Web applications**, environment configuration files, Users, Mount points, and Packages. Customized **Ansible modules** for finding **facts** about **Azure Monitor alarms** and taking actions to manage those **alarms** during deployments.
- Worked profoundly on **Ansible Tower** to write custom templates for configuring the inventories and specifying the **playbooks** written in **YAML** to execute the jobs on destined nodes. Downloaded and managed **Ansible roles** from **Ansible Galaxy** to automate the infrastructure.
- Developed Build and Deployment Scripts using build tools like **MAVEN** and **JENKINS** for **Continuous integration/Deployment(CI/CD)** and **SonarQube** for performing continuous inspection of the code quality and detecting the bugs in code.
- Involved in **Jira** as **bug tracking system** and configured various workflows, customizations and plug-ins for **Jira issue tracker** and integration of **Jenkins with GitHub**.
- Administered on version control systems like **GIT** and used Source code management client tools like Source Tree, **GIT Bash, GITHUB, GIT GUI** and Managed **Git repository, code merging, and production deployments**.

Tools and Technologies: *Azure (Web Roles, Worker Roles, SQL Azure, Azure Storage, Azure AD, Resource Groups), GCP, Ansible, Docker, Kubernetes, Maven, Jenkins, GIT, Cassandra, AEM, Python, Jira, Dynatrace, Azure Kubernetes Services, Rest API, Azure Site Recovery, TFS, VSTS, Azure Terraform, Radius packages, Azure ARM, Azure RBAC, Azure Virtual Networks, Azure NSG, Azure Database, Azure app insights.*

CLIENT: MAGELLAN HEALTH, VIRGINIA, USA

FEB 2017 – JULY 2018

AWS/DevOps Engineer

RESPONSIBILITIES:

- Experienced in various AWS Cloud services such as **EC2, Auto-Scaling** and **VPC** to build secure, highly scalable and flexible instances that can handle expected and unexpected traffic. Managed various AWS services using **AWS console, AWS CLI** and using **Amazon API (boto sdk)** using **Python** and **Pip**.

- Migrated **On-premise** application to **AWS Private Cloud** for a highly scalable and cost-effective infrastructure with necessary backup and recovery strategy for the application.
- Configuration, administration, and troubleshooting of the **VPC networking in a cross-domain**, including **VPC peering**, limits and pricing and architects within AWS cloud populated by **infrastructure, platform, and application services** that share common security and interconnection and **AWS Lambda and AWS config**.
- Worked with **Chef Enterprise Hosted as well as On-Premise**, Installed Workstation, Bootstrapped Nodes, Wrote **Recipes and Cookbooks** and uploaded them to **Chef-server**, Managed On-site OS/Applications/Services /Packages using Chef as well as AWS for EC2/S3/Route53 & ELB with Chef Cookbooks.
- Enforcing **Test Driven Development (TDD)** approach for writing Chef Cookbook using **Test-kitchen, Vagrant, Berkshelf** (Berkshelf is a bundler-like dependency manager for Chef cookbooks.), **FoodCritic** and **ChefSpec**.
- Maintained and **automated** the environment using **Chef Recipes & Cookbooks** within AWS and involved in **Knife and Chef Bootstrap process, converted production support scripts to chef recipes** and AWS server provisioning using **chef recipes** and also configured **Splunk for monitoring Infrastructure**.
- Involved in writing **Maven** build scripts and **POM files** for generating java-based **artifacts** such as **JAR, WAR, EAR** and deployed artifacts to **Nexus Repository Manager**. Researched and implemented an **Agile** work flow for **continuous integration (CI)** and testing of applications using **Jenkins**. Administered and Engineered **Jenkins** to manage weekly Builds.
- Integrated **Jenkins CI** with **GIT** version control and implemented continuous build based on check-in for various across functional applications and created **GitHub Webhooks** to set up **triggers for commit, push, merge and pull request events**.
- Experienced in setup of **SCM Polling** for Immediate Builds with **Maven and Maven Repository (Nexus Artifactory)** by installing **Jenkins Plugins for GIT Repository**.
- Experienced in Releasing code to **testing regions** or **staging areas** according to the schedule.
- Extensively worked on **Splunk dashboard** creation and **advanced searching and reporting**
- Used **Splunk** for **monitor** and **analyze** large scale machine generated data and integrated with various applications and services through API.
- Wrote **Ansible playbooks** with **Python SSH** as a wrapper to manage configurations and the test **playbooks** on **AWS instances using Python**. Used the **Ansible Galaxy**, a shared repository for the roles to download, share and manage the roles.
- Developed Test Cases for **Unit Testing** and **Integration Testing**, by using **Python Script**, which included CAN messages for generating Incidents.
- Automated **Java builds** with **Maven** and implemented multiple plugins for Code analysis like **Junit, SonarQube**. Installed and administered **Artifactory repository** to deploy the artifacts generated by **Maven**.

Tools and Technologies: *AWS, Docker, Chef, Nexus, Jenkins, CI/CD, Maven, Git, Splunk, GitHub, WebLogic, Apache Tomcat, Linux MongoDB, VMWare Servers, PowerShell, Bash, Python, Ansible, Jenkins, Cassandra DB.*

Build and Release Engineer

CLIENT: NATSOFT TECHNOLOGIES, HYDERABAD, INDIA

DEC 2015 – JAN 2017

RESPONSIBILITIES:

- *Built **CI/CD Pipelines** in **Jenkins** to retrieve code, compile applications, perform tests and push build artifacts to Nexus.*
- Designed and implemented new project builds Framework using **Jenkins & Maven** as build Framework tools as **continuous delivery** framework using **Jenkins, Chef and Maven** in Linux environment.
- Used **Nexus Artifactory** to manage the repositories in **Maven** and used the same to share the snapshots and release of internal projects.
- Good exposure on all areas of Jenkins like **Plugin Management**, securing Jenkins, performance issues, Analytics, Scaling Jenkins. Integrating code Analysis and Test phases to complete the CD pipelines within **Jenkins**.
- Managed build and deployment using **Ant, Bamboo, SCM tools** and managed **maven** project dependencies by creating parent-child relationships between projects.
- Involved in **Python scripting** for deployment of Web applications and **implementation** and **automation** of **Shell scripts** for **release and build automation**.
- Used **Git** for **source code version control** and integrated with **Jenkins** for **CI/CD pipeline**, code quality tracking and user management with build tools **Maven and Ant** and written **Maven pom.xml** build script.

- Integrated **Docker** container-based test infrastructure to **Jenkins CI** test flow and set up build environment integrating with **GIT** and **JIRA** to trigger builds using **WebHooks** and **Slave Machines**.
- Deployed **Java/J2EE** applications to Application servers in an **Agile** CI environment and automated the whole process.
- Manage all Configuration management tools (**JIRA, Confluence, Artifactory, SVN, Maven, Jenkins, ANT, Git, GitHub, Visual Studio**) and their usage process ensuring **traceability, repeatability, quality, and support**.
- Managed tools like **Subversion, Jenkins, JIRA** and performed maintenance and troubleshooting of build/deployment systems.
- Incorporating **LINUX/UNIX** system builds, administration, installations, upgrades, troubleshooting on different distributions such as **Ubuntu, CentOS, Red Hat, RHEL** and performed User Administration like (setting up and maintaining accounts).

Tools and Technologies: Linux, Nexus, Web Logic, Artifactory, Jenkins, Bamboo, Java, PHP, Python, Ruby, JIRA, GIT, Sub Version, Confluence, Visual Studio, Vagrant, SVN, Maven, Ant, UNIX, LINUX (Ubuntu, CentOS, RedHat, RHEL).

System/Linux Administrator

CLIENT: ZENSAR TECHNOLOGIES, INDIA

AUG 2013 – DEC 2015

RESPONSIBILITIES:

- Administered **Linux servers** for several functions including managing **Apache/Tomcat Servers, Mail Server, MySQL** databases in both development and production.
- Experience in **Linux Administration** (Installation, Configuration, Tuning and Upgrades of Linux (RedHat and Oracle).
- Extensive experience in System Builds, Server Builds, Installations, Upgrades, Patches, Migration, Troubleshooting on **RHEL**. Experience in configuration of **OSI Model, TCP/IP protocol** suite (IP, ARP, TCP, UDP, SMTP, SAMBA, FTP and TFTP)
- Installed operating system on multiple machines using **KickStart** and Performed server updates, patching, upgrade and package installations using **RPM** and **YUM**. Installation of **SSH** and configuration of **keys-based authentication**.
- Worked with DBA team for **database performance issues**, network related issues on **Linux Servers**.
- Managed users and groups in **Solaris & Linux** servers to create, modify and delete users/groups and troubleshooting user related issues and User account management, worked with **Perl, Shell scripting (bash)** to automate administration tasks. Analyzed security- based events, risks, and reporting instances.
- Expertise in **UNIX shell scripting** and **Python scripts** used to automate day to day administrative tasks. Involved in writing **Python, Perl, and Shell scripts** for compilation and deployment process.
- Developed automated deployment scripts using **Ant and Python** to deploy **WAR** files, properties file and database changes to development server or QA server and Staging/Production server.
- Experienced in deploying **Puppet, Puppet Dashboard** and **Puppet B** for configuration management to existing infrastructure and created Modules for Protocols configuration and managing them using **Puppet automation**.
- Worked on **Jenkins, GitHub, Artifactory** and all internal build systems for the core development team on an enterprise-level Java-based cloud orchestration/automation tool.
- Hands-on experience in installation, configuration of **SSH** to enable secure access to servers, deployment of VMs, Snapshots, and templates.
- Worked with installation, configuration, and management of the various **Radius Packages**.
- Developed various **bash shell scripts** to automate resource, job monitoring and alerting.
- Created tickets to track, diagnose and resolve new and ongoing incidents using **Nagios** ticketing system.

Tools and Technologies: *Linux, Ruby, Nginx, Kickstart, Radius Packages, Nagios, Oracle, MySQL, Crontabs, Linux, LVM, LAMP, Apache Tomcat, Jira, VM Ware, Oracle, MySQL, RedHat, Sun-Solaris, RPM, SMTP, FTP, SSH, Fedora, GitHub, Perl, Python.*