**Ram P**

**857-529-8768**

**rampothireddy7@gmail.com**

<https://www.linkedin.com/in/ram-dev/>



**PROFESSIONAL SUMMARY:**

* Skilled IT Professional with **7+** years of experience in **DevOps/Cloud Engineer** operations process involving **Automation**, **Build/Release Management**, Software Configuration Management, Service, Incident, and Change Management. Experience in Software Development Life Cycle (**SDLC**) & **Agile Methodologies**.
* In depth understanding in cloud computing strategies **SaaS, PaaS, IaaS** solutions and skilled at Software development Life Cycles like **Waterfall, Agile and Scrum programming methodologies**.
* Working with **Azure Active Directory** service creating Azure AD authentication, Roles, Users and groups.
* Also working on Application registration using Azure Active Directory to create Service Principal connections between **Azure DevOps** and **Azure Cloud** for linking the web app services to make deployments.
* Experience in migrating from on-premises to **Windows Azure** and build **Azure Disaster Recovery** Environment and Azure backups from the scratch using **PowerShell** script.
* Creating and Managing Virtual Machines in Windows **Azure** and setting up communication with the help of Endpoints. Configured **Shared Access Signature (SAS)** tokens and Storage Access Policies in **Azure Cloud Infrastructure**.
* Experience on **Azure Services** like **VM, App Service, Websites, SQL Azure, Storage, Blob Service, Cosmos DB, Network Services, VNets, Load Balancer, Resource Manager, Scheduling** and **Power Shell** Automation.
* Managing and Monitoring **Kubernetes** clusters using **Prometheus** as a data aggregator and **Grafana** as a data visualization platform.
* Expertise in **AWS**services Like **EC2, Load Balancer, Lambda, S3, RDS, DynamoDB, Aurora, SQL, VPC, Auto-Scaling, Route53, IAM, WAF, CloudFormation, Elastic Container Service (ECS), Elastic Kubernetes Service(EKS), Kinesis, EBS, Elastic Beanstalk, Elastic Cache, Snowball, LightSail, Glacier CloudWatch, CloudTrail, Security Groups, SNS, SQS, Direct Connect.**
* Implemented infrastructure on new **AWS** systems and migration of existing infrastructure to AWS Cloud platforms and automated using **Terraform, Cloud Formation** Templates and modules.
* Hands on experience in writing **Terraform API modules** to manage infrastructure, for automatic creation of AWS services like **RDS**instances, **VPCs, Autoscaling groups, Load balancers, SQS, S3 buckets**.
* Experienced in creating the AWS cloud Infrastructure by using **Terraform** and Deployed IaC code using Jenkins CI/CD pipelines and configure **AWS EC2** with **Ansible Playbooks and Roles**.
* Experienced in querying and extracting data from AWS **RDS** using Oracle SQL, MySQL, SQLServer, MongoDB and deploying in the AWS Redshift.
* Provisioned the **AWS App Mesh**for network traffic controls on EC2 instances, **ECS, EKS**, **AWS Fargate**and integrated monitoring tool cloud watch to automatically export the data to **Splunk**and **Kibana**.
* Implemented **AWS Beanstalk**for deploying web applications based on **.Net, Node.js, 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/ Red Hat/Windows**).
* Set up a **GCP** Firewall rules in order to allow or deny traffic to and from the **VM** instances based on specified configuration and used **GCP** cloud **CDN** (content delivery network) to deliver content from **GCP** cache locations drastically improving user experience and latency.
* Worked on version control and source code tools like **GIT**, **Subversion (SVN)** and used Source code client tools like **Bitbucket, GitLab, Git Bash, GitHub** and **Git GUI**.
* Experienced in and demonstrated understanding of source control management concepts such as **Branching, Merging, Labeling/Tagging** and **Integration**.
* Experienced in setup and maintained entire **Jenkins, Bitbucket** and **Bamboo** for **continuous integration** and **continuous deployment** (**CI/CD**) for End-to-End automation for all build and deployments.
* Integrated **Jenkins** with tools like **Git** (Repository), **Maven** (Build tool), **SonarQube** (code verification), **Nexus** (Artifactory) and implementing **CI/CD** automation for creating **Jenkins** pipelines programmatically.
* Managed servers on AWS platform instances using **Ansible** and **Chef** Configuration management and wrote **Ansible** **playbooks** using YAML scripting and created inventory to automate the build and deployment process on servers using **SSH** connections and reduced overall processing time.
* Experienced in setting up **Puppet** Master, Puppet Agents for managing enterprise policies & configurations that makes applications and systems easier to deploy.
* Deployed & configured **Chef** Server including **bootstrapping** of chef-client nodes for provisioning and managed Cookbooks, Recipes, attributes and templates.
* Installed and worked with **Docker** containerization to run software packages on the containers to improve **Continuous Delivery** framework and Virtualized the servers using Docker for the development, test environment and performed configuration automation using Docker containers.
* Good experience in several Docker components like Docker Engine, Hub, Compose & Docker Registry and used **Kubernetes** to automate deployments, management of applications running on multiple docker containers.
* Designed & deployed container-based production clusters using Docker, Kubernetes, Docker Swarm and used Kubernetes to deploy, load balance & manage docker containers with multiple namespace versions.
* Hands on experience in **JIRA** for creating bug-tickets, storyboarding, report generation, ticket management and managed configuring changes into production environment using the Jira tracking tool.
* Monitored servers, applications by using **shell**/**python** scripts and used **Splunk**, **Nagios** to resolve infrastructure problems and worked on Nagios Event handlers to automatically restart failed services and applications.
* Hands on experience on development of utilities using **Python**, **Ruby** scripting to automate processes and performed various day-to-day administration tasks.
* Expertise in querying RDBMS such as Oracle SQL, MYSQL, SQLSERVER and AWS Aurora by using PL/SQL for data integrity.

**TECHNICAL SKILLS:**

|  |  |
| --- | --- |
| **Cloud Platform** | AWS, Azure, Google Cloud Platform |
| **Infrastructure as a Code** | AWS CloudFormation, Terraform |
| **SCM/Version Control Tools** | GIT, GitHub, SVN, Bitbucket, GitLab, AWS Code Commit |
| **Continuous Integration Tools** | Jenkins, Bamboo, TeamCity |
| **Build Tools** | Maven, ANT, Gradle |
| **AWS Cloud Services** | EC2, S3, VPC, Route53, SNS, IAM, Cloud Front, EBS, EKS, ELB, ECS, Lambda, Cloud Watch, Auto Scaling, SQS, Elastic Beanstalk |
| **Configuration Management Tools** | CHEF, Ansible, Puppet |
| **Containerization Tools** | Docker, Kubernetes, Docker Swarm, OpenShift |
| **Virtualization Tools** | VMware, Virtual Box |
| **Scripts/ Languages** | UNIX, HTML, Shell Scripting, YAML, Python, Perl, Groovy, Power Shell |
| **Databases** | Oracle, MongoDB, SQL Server, MS SQL |
| **Automation Tools** | Jenkins, Jfrog, Nexus, Docker, Kubernetes, Ansible, Terraform, compliance, Kitchen, chef solo, Puppet |
| **Monitoring Tools** | Nagios, Splunk, CloudWatch, Prometheus, Grafana, CloudTrail, ELK |
| **SDLC** | Waterfall, Agile, Scrum |
| **Bug Tracking Tools** | JIRA, Scrum Master |
| **Operating System** | UNIX, Linux (Ubuntu, RHEL, Centos), Windows, MacOS. |
| **Other Tools** | Apache Tomcat, Siebel CRM, SMP (Service Management Platform), BRM, PyCharm, .Net, SQL Developer, Eclipse. |

**EDUCATION:**

**Master of Professional Studies in Informatics** **GPA: 3.65/4**

Northeastern University, Boston, MA, USA

**Bachelor of Engineering in Computer Science GPA: 8.75/10**

SRM University, Chennai, India

**CERTIFICATIONS:**

* **AWS Certified Solutions Architect** – **Associate** (Validation Number 62WGXXFBCNE11XWP)
* **AWS Certified Developer – Associate** (Validation Number PLQTETLJKN41QG3W)
* **Microsoft Certified: Azure Developer – Associate**

**EXPERIENCE:**

**Washington Hospital Healthcare System, Fremont, CA** **Dec 2018 - Present**

**Sr. Cloud/DevOps Engineer**

**Responsibilities:**

* Working on **Azure Site Recovery** and **Azure Backup**- Deployed Instances on Azure environments and in Data centers and migrating to Azure using **Azure Site Recovery**and collecting data from all Azure Resources using Log Analytics and analyzed the data to resolve issues.
* Configuring Azure **Multi-Factor Authentication (MFA)** as a part of **Azure AD Premium** to securely authenticate users and worked on creating custom **Azure templates** for quick deployments and advanced **PowerShell**scripting.
* Working on **Serverless services**, created and configured **HTTP** Triggers in the **Azure Functions** with application insights for **monitoring** and **performing** load testing on the applications using the **Visual Studio Team Services** (**VSTS)**alsocalled as**Azure DevOps Services.**
* Creating recommendations on how to duplicate a subset of on-premises machines to the **Azure Infrastructure as a Service (IAAS)** offering which will be used for disaster recovery.
* Creating Azure **ARM** templates, deployed an entire **Azure** infrastructure declaratively and used deployment scripts in **Azure Resource Manage** **(ARM)** templates.
* Configuring **Continuous Integration** from source control, setting up build definition within **Visual Studio Team Services (VSTS)** and configure **continuous delivery**to automate the deployment.
* Implementing a **CI/CD** pipeline with Docker, **Jenkins** (**TFS** Plugin installed), **Team Foundation Server (TFS)**, **GitHub** and **Azure Container Service**, whenever a new **TFS/GitHub branch** gets started, **Jenkins**, our **Continuous Integration (CI)** server, automatically attempts to build a new Docker container from it.
* Working with **Terraform** Templates to automate the Azure **Iaas** virtual machines using **terraform modules** and deployed virtual machine scale sets in production environment.
* Integrating **Azure Log Analytics** with **Azure VMs** for monitoring the **log files**, store them and track metrics and used **Terraform** as a tool.
* Managing **Azure Infrastructure,** **Azure Web Roles**, Worker Roles, VM Role, Azure SQL, Azure Storage, Azure AD Licenses, **Virtual Machine Backup** and Recover from a Recovery Services Vault using **Azure PowerShell** and **Azure Portal**.
* Integrating **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**.
* Working on **OpenShift** for **container** **orchestrati**on with **Kubernetes** container storage, automation to enhance **container** platform multi-tenancy also worked on with **Kubernetes** architecture.
* Deploying windows **Kubernetes (K8s)** cluster with **Azure Container Service (ACS)** from Azure CLI and utilized **Kubernetes** and **Docker** for the runtime environment of the **CI/CD** system to build, test and Deploy.
* Managed **Kubernetes** charts using Helm, Created reproducible builds of the Kubernetes applications, managed **Kubernetes** manifest files and Managed releases of Helm packages.
* Using **Ansible** created multiple **playbooks** for machine creations and **SQL server, cluster server** and my SQL installations.
* 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 **Ansible handlers** with multiple tasks to trigger multiple handlers and to decouple handlers from their names, making it easier to share handlers among Playbooks and Roles.
* Implemented **Docker -maven-plugin** in **Maven**pom.xml files to build Docker images for all microservices and later used Docker file to build the Docker images from the **Java**jar files.
* Designed, wrote, and maintained systems in **Python scripting** for administering **GIT**, by using **Jenkins** as a full cycle **continuous delivery** tool involving package creation, distribution, and deployment onto **Tomcat application** servers via shell scripts embedded into **Jenkins** jobs.
* Maintained **Artifacts** in binary **repositories** using **JFrog** Artifactory and pushed **new Artifacts** by configuring the **Jenkins** project **Jenkins Artifactory plugin**.
* Involved in **Jira** as **defect tracking system** and configure various workflows, customizations and plugins for Jira bug/issue tracker integrated **Jenkins** with Jira, GitHub.
* Worked with **Nagios** for **Azure Active Directory** & **LDAP** and Data consolidation for **LDAP users**.
* Built and managed a highly available monitoring infrastructure to monitor different application servers like **JBoss, Apache Tomcat** and its components using **Nagios**.
* Creating restful web services for Catalog and Pricing with **Django** MVT, Jersey, **MySQL**, and **MongoDB**.
* Involved in writing **shell scripts** to automate **WebSphere** admin tasks and application specific syncs / backups and other schedulers.

**Environment:** Azure, PCF, Office 365, Terraform, Maven, Jenkins, Ansible, Azure ARM, Azure AD, Azure Site Recovery, Kubernetes, Python, Ruby, XML, Shell Scripting, PowerShell, Nexus, JFrog Artifactory, Jenkins, Git, Jira, GitHub, Ansible, Docker, Windows Server, TFS, VSTS, LDAP, Nagios.

**Fidelity Investments, Boston, MA** **April 2017 – Nov 2018**

**Sr. Cloud/DevOps Engineer**

**Responsibilities:**

* Analyzed the existing standalone environment and designed the new infrastructure architecture with **AWS** services without losing the functionality of existing environment.
* Involved in maintaining the user accounts (**IAM**), **RDS**, **S3**, **Route 53**, AWS Lambda, VPC, **Dynamo DB**, SES, SQS and SNS EMR services in **AWS cloud**.
* Leveraged AWS cloud services such as **EMR**, auto-scaling and **VPC (Virtual Private Cloud)** to build secure, highly scalable and flexible systems that handled expected and unexpected load bursts and can quickly evolve during development iterations.
* Worked on Auto-scaling **Cloud Watch**(monitoring), **AWS Elastic Beanstalk** (app deployments), **AWS Lambda**, **Amazon S3**(storage) and **Amazon EBS** (persistent disk storage) and **Dynamo DB**.
* Built **S3**buckets and managed policies for S3 buckets and used **S3 bucket**and **Glacier**for storage and backup on **AWS**.
* Configuring **Kinesis Shards**for optimal throughput **in Kinesis Streams** for **Spark Streaming** Applications on **AWS**.
* Expertise in building **CI/CD** on AWS environment using **AWS Code Commit**, **Code Build**, **Code Deploy** and **Code Pipeline** and experience in using **AWS CloudFormation, API Gateway,** and **AWS Lambda** in automation and securing the infrastructure on **AWS**.
* Created an **AWS RDS** **Aurora DB** cluster and connected to the database through an Amazon RDS **Aurora DB** Instance using the **Amazon RDS** Console.
* Utilized **Jenkins**master/slave architecture to distribute builds on nodes and trigger **Jenkins**job to build the artifacts using **Maven**, deployed the **Terraform**templates to create the stack.
* Implemented a **CI**/**CD** pipeline with **Docker**, **Jenkins** and **GitHub** by virtualizing the servers using Docker for the Dev and Test environments by achieving needs through configuring automation using Containerization.
* Implemented **AWS** **Code Pipeline**and Created **Cloud formation** **JSON**templates in **Terraform**for infrastructure as code.
* Extensively involved in **infrastructure as code**, execution plans, resource graph and change automation using **Terraform.**
* Experience on infrastructure as code with **Terraform** tool by operating AWS infrastructure through Terraform and **Terraform Enterprise** (TFE) by writing Terraform scripts, modules etc.…
* Converted existing **terraform modules** that had version conflicts to utilize cloud formation during **Terraform** deployments to enable more control or missing capabilities.
* Migrated Windows Active Directories and .net applications from **AWS** to **Azure** cloud by implementing **POC’s** in the organization and Performing **POT’s** in Azure implementation.
* Used **Python** to automate provision by **Terraform**for tasks such as encrypting **EBS**volumes backing **AMIs**and scheduling Lambda functions for routine **AWS**tasks and stored result in **Dynamo DB**.
* Worked with container-based deployments using **Docker**, working with **Docker** images, **Docker** hub and **Docker** registries, installation and configuring **Kubernetes** and clustering them.
* Proficient in creating **Docker images**using **Docker File**, worked on **Docker container**snapshots, removing images, and managing **Docker volumes**and implemented **Docker** automation solution for **CI/CD** model.
* Managed **Kubernetes** charts using **Helm**, Created reproducible builds of the **Kubernetes** applications, managed Kubernetes manifest files and Managed releases of Helm packages.
* Deep hands-on experience of **Kubernetes** including setting up a **Kubernetes cluster**, managing clusters, creating and managing pods, deployments, application management, rolling updates, exposing using services & routes, persistent volumes and persistent volume claims.
* Automated configuration management and deployments using **Ansible** playbooks and **Yaml**for resource declaration. And creating roles and updating Playbooks to provision servers by using **Ansible**.
* Installing, Configured and management in **Ansible** Centralized Server and creating the playbooks to support various middleware application servers, and involved in configuring the **Ansible** tower as a configuration management tool to automate repetitive tasks.
* Setup AWS infrastructure monitoring through **Datadog** and application performance monitoring through **App Dynamics.**
* Experience with installation and configuration of **Dynatrace** monitoring tool. And created email alerts and threshold values using Dynatrace for our environment.
* Experience in Installing, configuring Cloud Foundry Ops Manager, App Manager, Etc. Configuring LDAP for authorization, configuring log regulator for logs in **PCF** (ELK/Splunk).
* **Pager-duty** integrations with **App dynamics, Splunk, Wave-front, Cloud watch** and slack through API’s and define escalation policies and schedules.
* Installed and upgraded **Splunk software**in distributed and clustered environments for numerous corporations and public entities**.**
* Extensive Programming experience in **Python, PowerShell**and **Bash**scripting. Wrote bash and Shell scripts for auto launch and web server configurations.
* Designed workflows in **Atlassian JIRA** to deal with issues and maintained all the user stories for tracking as per Agile style.

**Environment:** AWS, Bitbucket, GIT, MAVEN, Ant, Jenkins, Ansible, Docker, Kubernetes, Python script, MySQL, JavaScript, JBoss, J2EE, Shell Script and JIRA.

**Office Depot, Columbus, OH Feb 2016 – April 2017**

**Cloud/DevOps Engineer**

**Responsibilities:**

* Created multiple **EC2** nodes, configured launched instances and maintained the AWS cloud infrastructure as required in the environment.
* Used **Rehosting** Migration strategy to migrate on premise Infrastructure to AWS Cloud and created cloud formation templates and implemented using AWS **code pipeline**.
* Designed roles and groups for new users using AWS Identity and Access Management (IAM) and implemented policies to provide additional security to AWS account and its resources**.**
* Migrated the relational database files (binary and transaction logs) to Amazon **RDS** instances using the standard “MySQL import” utility.
* Created and utilized **auto scaling** launch configuration templates for amazon EC2 instances while deploying Micro services.
* Created **Snapshots** and **AMIs** of EC2 instances and moved AMIs to different regions to create cross region failover, created **S3** buckets, managed policies on S3 buckets for storage and backup on AWS.
* Experienced in writing **terraform** modules for **automation** the creation of **VPC-VPN** from data center to **Prod environment** and Cross account **VPC Peering**.
* Wrote **Terraform** scripts to create **VPC** subnets and spin-up multiple instances with defined configuration within the VPC created utilizing public and private subnets based on requirement.
* Used **GCP** cloud **CDN** (content delivery network) to deliver content from **GCP** cache locations by drastically improving user experience and latency.
* Setup, alerting and monitoring using **Stackdriver** in GCP and created custom log metrics using **Stackdriver** logging and create charts and alerts using the custom log metrics and provided the permissions and required access to all the **subtopics** to push/write the data to **Stackdriver**.
* Used **MAVEN** as a build tool for the development of build artifacts on the source code and deployed artifacts into **Nexus** (Jar and War file) repository and defined dependencies, plugins in **Maven** **pom**.**xml** for various activities and integrated Maven with GIT to manage and deploy project related tags.
* Implemented continuous delivery framework using **Jenkins** in Linux environment, created pipelines for Jenkins jobs integrating with (SCM poll) **GIT** and setup GIT on a Linux environment, configured the users and created the repositories and branching structure for all the applications.
* Configuration of nodes for **Jenkins** to customize builds for different operating systems, smooth running of builds jobs, testing and deployed in **Apache Tomcat server.**
* Built and configured **Jenkins slaves** for parallel job execution and Created **Chef** Cookbooks and Recipes to integrated with Jenkins jobs and automate the deployment process.
* Worked on bootstrapping nodes with **Chef** server for on-site application, cookbooks and wrote recipes and uploaded them to chef server to manage on-site application and services.
* Written **Chef** **Cookbooks** using **Ruby** for DB configurations to modularize and optimize product configuration, converting production support scripts to **Chef** **Recipes** and AWS server provisioning using Chef Recipes.
* Automated application deployment in the cloud using Docker and created, managed a Docker deployment pipeline for custom application images in the cloud using Jenkins.
* Worked on **Docker** **container** snapshots and deployed Docker Swarm using **Chef** and automated container deployment and created docker images for various environments.
* Managed Clusters using **Kubernetes** and worked on creating many pods, services, deployments, labels, health checks.
* Used **Kubernetes** and Docker for building and containerizing the **CI/CD** application to deploy on cloud and managed container replicas on node clusters using **Kubernetes**.
* Developed **Perl** and **shell** **scripts** for automation of the build and release process and used UNIX scripts for build activities in QA and Staging environments.
* Configured **Splunk** forwarders to detect SSL certificate expirations, analyze the system logs and index the data from various database types.
* Used **Nagios** to detect network latencies between systems and monitored system health performance by scheduling **Splunk** based reports and alerts.
* Used **Scrum Master** for generated tickets, reports on different bug issues, tracking & updating project defects & tasks.

**Environment:** AWS, Jenkins, GIT, Python, Bash Script, Chef, Docker, Kubernetes, Apache Tomcat, Maven, Nagios, Splunk and Scrum Master.

**Bell Telecommunications,** **Hyderabad, India** **April 2015 - Feb 2016**

**Build & Release Engineer**

**Responsibilities:**

* Worked for a team that involved different development teams and multiple simultaneous software releases and participated in weekly release meetings with Technology stakeholders to identify and mitigate potential risks associated with the releases.
* Developed and implemented software Release Management strategies for various applications according to the **agile process**.
* Involved in creating new Build Definitions, creating and updating Build Process Templates for new releases and new requirements.
* Created and managed various **EC2** instances, **EBS** volumes, configuring **security groups, elastic ip**’s, **AMI’s**, **S3** buckets and configured them with permissions, versioning, logging and tagging.
* Managed multiple server instances by enabling password less **SSH** communication between the servers and deployed JAR, WAR files to Apache Tomcat and Web Sphere application servers.
* Administered and Maintained **GIT** source code, pushed, pulled and performed merge requests into the **Git Repository and** configured, deployed GIT repositories with tagging, branching and notifications.
* Managed Maven project dependencies by creating parent-child relationships between projects and tweaked the existing **Maven** files in case of errors or changes in the project requirements.
* Executed Maven scripts in **Jenkins** to build several java-based projects where there is a change in the working environment.
* Carried out deployments and builds on various environments using Jenkins and developed **Jenkins** build pipeline jobs using **groovy** for **Node.js** and **Java**-based applications.
* Installed, maintained and administering **Chef** (configuration management) for existing infrastructure and used Chef to manage Config files, Web Applications and Database.
* Wrote **Chef** **cookbook** to integrate into Jenkins jobs for a continuous delivery framework and automate the deployment process.
* Wrote scripts in SHELL and Python to automate Cron jobs and system maintenance and developed the ruby scripts to automate system deployment to scale infrastructure.
* Provided support to **Production**, **Staging**, **QA**, Development environments for code deployments, changes, and general support.
* Documented software for the intranet application of **USSD, OBRM** which consists of back-office applications such as Campaign management, Journaling, Balance Transfer, Pre-activation & Post-activation.
* Integrated **JIRA** to ensure the process is followed and customized JIRA for workflow, user, project management and automated JIRA tasks using **JIRA CLI.**

**Environment:**AWS (EC2, EBS, AMI, S3), GIT, Maven, Chef, Jenkins, Web sphere,Tomcat, Maven, Shell script, JIRA, JSON, USSD, OBRM, Siebel CRM, SMP, BRM, groovy and Python script.

**Chakravuyha Software Technologies PVT. LTD.,** **Hyderabad, India Jan 2014 - Mar 2015**

**Linux Administrator**

**Responsibilities:**

* Used Operating Systems like Solaris, Managing SUN Solaris, Compaq and Linux workstations and performed system administration of **UNIX** servers.
* Installed, implemented, maintained and support for the corporate Linux servers **RHEL** 3, 4, 5, **CENTOS** 5, **Ubuntu** and involved in System Administration, System Builds, Server builds, Installs, Upgrades, Patches, Migration, Trouble Shooting, Security, Backup, Disaster Recovery and Performance Monitoring on **UNIX** (Red Hat Linux) Systems
* Installed and configured RHEL and **Windows** operating system and software packages like Apache, MySQL and other Linux and Windows environments.
* Configured the web servers like **Apache** with **Tomcat** servers and installed all the dependencies and environmental properties within the Linux and Windows environment.
* Used **SCP** servers to deploy Java web application across multiple webservers in the Tomcat webservers and **WAR** files
* Monitored system firewall rules and troubleshooted the common OS and Network issues in the servers to allow and deny traffic in the network.
* Experience in troubleshooting hardware, **Network, Storage, File system** and application installation issues in the Linux and Windows servers.
* Managed Network troubleshooting applications **TCP/IP** including Ethernet, Configured **LAN, WAN** and IP addressing & Sub netting, routing.
* Used **Nagios** monitoring for overall system performance, performed user management, system updates, file systems, network and storage management.

**Environment:** Red Hat Linux, Unix, CentOS, Apache Tomcat, Windows, Java Web Applications, MySQL, LAN, WAN and Nagios.