******Sai Prasad**

Sr. Devops /Cloud engineer

Email: prasada6618@gmail.com

Contact no: +1 (309) 306-1340‬

LinkedIn: <https://www.linkedin.com/in/prasad-ashila/>

**Professional Summary:**

Over **7+ years** of IT Professional Experience includes, **Linux** and **System Administration** with a major focus on **DevOps**, **Cloud** technologies, **Continuous Integration/Deployment**, **Configuration Management**, **Source Code Management**, and **Virtualization** technologies which includes Troubleshooting Performance issues.

* Expertise in Amazon **AWS Cloud services** like **EC2, S3, EBS, VPC, ELB**, AMI, SNS, RDS, IAM, **Route 53**, **Glacier,** **Kinesis, Auto scaling, CloudFront**, **CloudWatch**, CloudTrail, Cloud Formation, Elastic Beanstalk, **OPS Work**, Amazon Lambda, Security Groups.
* **Migrated On-premise infrastructure to AWS** and involved in virtualization using (VMware, VMware ESX, Xen).
* Using **Clover ETL** migrated data to **AWS Redshift** and used **AWS Beanstalk** for fast deploying, scaling and load balancing of web applications and services developed with **Java**, **PHP**, **Node.js**, **Python**, **Ruby**, and **Docker** on familiar web servers such as **Apache**, and **IIS**.
* Experience with AWS Serverless services as **Lambda functions** , **API Gateway**, **Fargate** for container Deployment and orchestration.
* Experience in using **CloudFormation template** for reliable provisioning of initial infrastructure (**stacks**) like **IAM** **roles** & **policies**, **S3 buckets** and **bucket** backups on **AWS**.
* Extensive experience in **Azure** compute services, **Azure Web apps, Azure Data Factory, Azure Media, Content delivery, Azure Networking, Azure Hybrid Integration, Azure Identity Management, Azure SQL Database.**
* Experience in **Private Cloud** and **Hybrid cloud** configurations, patterns, and practices in **Windows Azure** and **SQL Azure** and in Azure Web and Database deployments.
* Expertise in **Azure Development** worked on Azure web application, **Azure storage, AzureSQL Database, Azure Virtual Machines, Azure AD**, Azure search, **Azure DNS, Azure VPN** Gateway and Notification hub.
* Analyzing the windows azure platform issues and reporting bugs to windows azure developers which include **Dashboard Monitoring** and **Alert Monitoring** for Azure Platform.
* Strong knowledge on Google Cloud Platform (**GCP**) **Compute** Engine, **Kubernetes** Engine, **App** Engine, **Cloud** Functions, **Datastore**, **Bigtable** and Cloud **storage** services. Hands on experience in cloning an **OpenStack** **mini** Cluster in GCP Compute service.
* Experience in installing and configuring **PCF** (Pivotal Cloud Foundry) environments and worked on Pivotal's **cloud native** platform, an **integrated stack** from infrastructure through application **development**, optimized for **rapidly scaling** applications to meet the demand.
* Experience in orchestration of **Docker Images** and **Containers** using **Kubernetes** by creating whole master and minion. Installed **artifactory-ha** on-top of **Kubernetes** for dynamic **PVC** provisioning.
* Experience in Building and deploying the application code using **CLI** of **Kubernetes** like **kubectl**, **kubelet**, **kubeadm, Kubespray** and Schedule the jobs using **kube-scheduler**. Managed **K8s** charts using **Helm** and created reproducible builds of the **Kubernetes** applications.
* Managing the helm repository created helm charts out of the configured yaml scripts.
* Creating **Grafana Dashboards** for **monitoring the Kubernetes cluster** using **Prometheus** for gauging the Availability of the API sever, scheduling rate of the Scheduler and using PromQL quires for specific metric.
* Implementing **kubernetes auth** method to authenticate with **HashiCorp Vault** using a Kubernetes Service Account **Token** then integrates Vault token into a Kubernetes **Pod**.
* Experience in exploring third-part provisions as eskctl from weaveworks to create the EKS clusters and manage them there on using **Nirmata EKS Manager** for UI.
* Experience in managing **Ansible Tower, Playbooks** with **Ansible roles**, **group variables**, **inventory files**, copy and remove files on remote systems using **file module**.
* Extensively worked on creating **Chef Recipes**, **Chef Roles**, **Templates**, **Attributes**, **Cookbooks** and testing of **Cookbooks** using **Food Critic**, **Test Kitchen** & running recipes on nodes managed by on-premise **Chef Server.**
* Expert in an Orchestration & Migration of **CI/CD** processes using **CloudFormation**, **Terraform Templates** & Containerization of Infrastructure using **Docker**, which was set up in **Vagrant**, **AWS**, and **VPC**’s.
* Expertise in working with **Terraform** key features such as Infrastructure as a code (**IaaS**), **Execution** **plans**, **Resource** **Graphs**, **Change** **Automation** and extensively used **Auto** **scaling** launch configuration **templates** for launching amazon EC2 instances while deploying **microservices**.
* Extensively worked on **Jenkins** and **Bamboo** by installing, configuring and maintaining the purpose of **Continuous Integration** (CI) and for **End-to-End** automation for all build and deployments implementing **CI/CD** for database using **Jenkins** in **Groovy**.
* Hands-on experience using **MAVEN**, **ANT**, and **Gradle** as build tools for the building of deployable artifacts (jar, ear & war) from the source and static code analysis using **SonarQube**.
* Worked in building a shared library that is been used across our organization for CI/CD. Installed and configured Jenkins using IAAC and JCACS in a Kubernetes environment.
* Worked on **Jenkins** integration with **SonarQube** checkmark back dock **Jforg** **Nexus** **Ansible**. Involved in trouble shooting and fixing build and deployment in **Jenkins**
* Experienced in branching, tagging and maintaining the version across the environments using **SCM** tools like **Git**, **GitHub**, **Subversion** (SVN) and **TFS** on Linux and Windows platforms.
* Hands-on experience using **MAVEN**, **ANT** and **Gradle** as build tools for the building of deployable artifacts (jar & war) from the source and static code analysis using **SonarQube**.
* Good knowledge on Installation and Configuration of **Nexus** repository manager for sharing the artifacts within the company and efficient in using **Nexus** and **Artifact Repository** managers for **Maven** builds.
* Experience in Monitoring server performance with tools like **Nagios**, **Splunk**, **Dynatrace**, **Datadog** and resolved network related issues with manual commands and Implemented centralized logging management using **ELK** (Elasticsearch, Logstash, and Kibana) within **EC2**/ **ELB** for **Elasticsearch**.
* Experience in working with **SQL** databases like **MySQL**, **Oracle** 10i/9i/8i, **SQL Server** and **NoSQL** database like **MongoDB**, **DynamoDB**.
* Expertise in Installation, Support, Configuration with different flavors of **LINUX RHEL**, **CentOS**, **Oracle Linux (OEL) 6.x/5.x**, **Debian**, and **Ubuntu**. Installed and configured **Red Hat LINUX Kickstart.**

**TECHNICAL SKILLS:**

|  |  |
| --- | --- |
| Cloud | AWS, Azure, OpenStack. |
| Configuration Management Tools | CHEF, PUPPET, **ANSIBLE** |
| CI/CD Tools | **Jenkins**, Hudson, Bamboo |
| Build Tools | ANT**, MAVEN, GRADLE** |
| Container Tools | Docker, Kubernetes, EKS, ECS, ECR, AKS, ACR |
| Version Control Tools | GIT, CVS, SVN, Bit Bucket, GitHub, GitLab |
| Monitoring Tools | Nagios, Splunk, ELK, Cloud watch |
| Scripting | Bash/Shell, Perl, Python, Ruby, PowerShell, **Groovy**, YAML, C, C++ |
| Databases | MySQL, Oracle 9i/10g/11g, DB2, NoSQL (MongoDB, DynamoDB) |
| Application Servers | WebLogic 9.x/10.x, WebSphere 6.x/7.x/8.x, JBOSS 4.1 |
| Web Servers | Apache HTTP, Nginx, Apache TOMCAT |
| Networking | DNS, DHCP, SMTP, HTTP, SNMP |
| IDE Tools | ECLIPSE IDE, NETBEANS |
| Operating Systems | Linux (RedHat), Ubuntu, Unix, Windows XP/Vista |
| Virtualization Tech | VMware vSphere ESXi 5.x/4.x, ESX /3.x, VMware Workstation, Oracle Virtual box |

**CERTIFICATIONS:**

1. **AWS Solution Architect Associate**

**Digital Badge:** <https://www.youracclaim.com/badges/c944afb3-8515-4c1f-98c3-9844b9ad6fa6/public_url>

1. **AWS Developer Associate**

**Digital Badge:**  https://www.youracclaim.com/badges/b96cd801-fcec-4bf8-acb5-ffb455b3df23/public\_url

1. **HashiCorp Certified Terraform Associate**

**Digital Badge**: <https://www.youracclaim.com/badges/674706da-2e08-4ffb-9976-f34a84f6c41f/public_url>

1. **Kubernetes Application Developer**

**Digital Badge**: <https://www.youracclaim.com/badges/5612e60f-c8ac-4050-b32b-1fbab0165dd1/public_url>

**Professional Experience:**

**Sr. Cloud/DevOps Engineer Aug 2019 - present**

Client: Conduent, Austin, TX

* Deployed and configured **Elasticsearch**, **Logstash**, and **Kibana** (**ELK**) for log **analytics**, and application **monitoring** in integration with AWS **Lambda** and **CloudWatch**. Then store that logs and metrics into an **S3 bucket** using Lambda **function**.
* Integrated AWS **Dynamo DB** using AWS lambda to store the values of items and backup the **Dynamo DB streams,** implemented load balanced, highly available, fault tolerant, auto-scaling **Kubernetes** **AWS** infrastructure and **microservice** container orchestration
* Deployed **AWS** Cloud services (**PaaS** role instances) into secure **VNets**, **subnets** and designed Network Security Groups (**NSG**s) to control Inbound & Outbound access to Network Interfaces (**NIC**s), **VM**s & **subnets**.
* Worked with **Terraform** Templates to automate the **AWS IaaS** virtual machines using terraform **modules** and deployed virtual machine **scale sets** in production environment.
* Configured the Kubernetes **provider** with **Terraform** which is used to interact with resources supported by Kubernetes to create several services such as **Deployments**, **services**, **ingress** **rules**, **Config Map**, **secrets** etc., in different **Namespaces**.
* Worked on Container management using **Docker** by writing **Docker files** and set up the automated build on Docker Hub and written **Docker Compose** file for **multi container** provisioning and **Makefile** file to **build**, **run**, **tag** and **publish** a docker container to **Elastic Container Registry**.
* 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**.
* Implemented **Docker-Maven plugin** 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**.
* Worked with RedHat **OpenShift** Container Platform for **Docker** and **Kubernetes**. Used **Kubernetes** to deploy **scale**, **load balance** and **manage** Docker containers with multiple **namespace** versions.
* Built docker images, Created and deployed docker images to open shift. And patch update and scale Openshift environment. Use of Docker and Openshift to manage microservice for developing and testing.
* Implementing **clusters** using **Kubernetes** and worked on creating **pods**, replication controllers, Name Spaces, deployments, Services, labels, health checks, **Ingress** resources and Controllers by writing **YAML** files. Integrated them using **weave**, **flannel**, **calico SDN networking**.
* Deployed **Kubernetes clusters** on top of Servers using **KOPS.** Managed local deployments in **Kubernetes,** creating local clusters and deploying application containers. Building/maintaining **docker container clusters** managed by **Kubernetes** and deployed Kubernetes using **helm charts**.
* Developed microservice onboarding tools leveraging **Python** and **Jenkins,** allowing for easy creation and maintenance of **build jobs**, **Kubernetes** deploy and services.
* Designed, installed and implemented the **Ansible** configuration management system and used **Ansible** to manage Webapps, used **Ansible Tower** to automate repetitive tasks, quickly deploy critical applications and proactively manages change.
* Managed Ansible **Roles** by using **tasks**, **handlers**, **vars**, **files** and **templates** in installing, configuring and deploying the webserver application.
* Wrote several **Ansible playbooks** for the automation that was defined through tasks using **YAML** format and run Ansible Scripts to provision Dev servers.
* Used **Jenkins** as Continuous Integration tools to deploy the Spring Boot **Microservices** to **AWS Cloud** and **OpenStack** using build pack.
* Worked in complete **Jenkins** plugins and administration using **Groovy Scripting**, setting up **CI** for new branches, build automation, plugin management and securing **Jenkins** and setting up **master/slave** configurations. Deployed and configured **Git** repositories with **branching**, **forks**, **tagging**, and **notifications**.
* Installed **Jenkins Plugins** for **GIT** Repository, Setup **SCM** Polling for Immediate Build with **Maven** and **Maven Repository** (**Nexus** Artifactory).
* Worked with **MAVEN** for building the application, and written maven scripts and shell scripts to automate the build process. Daily maintenance of **GIT** source repositories and builds.
* Setup **SCM** Polling for Immediate Builds with **Maven** and **Maven Repository** (**Nexus** Artifactory) by installing **Jenkins Plugins** for **GIT** Repository.
* 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.
* Used **MAVEN** as a build tool for **Java** projects and written Maven **POM** files from scratch for multi-module projects for the development of build artifacts on the source code.
* Setup Continuous Delivery pipeline using **Ansible playbooks**. This primarily consists of a **Jenkins** to run **packages** and various supporting software components such as **Maven**.
* Developed **Perl**, **PowerShell**, **Shell** scripts, used **Maven** scripts to build the source code.
* Worked on **Power Shell** scripts to automate the **AWS Cloud system** in creation of **Resource groups**, **Web Applications**, **security groups**, **firewall** rules.
* Troubleshooting of network issues using **DHCP**, **DIG**, **DNS**, **SNMP**, **SMTP**, **netstat**, **NFS**, **NIS**, **nslookup**, **RIP**, **OSPF**, **BGP**, **TCP/IP**, and **TCP dump**.
* Installed, configured and Administrated of all **UNIX/LINUX** servers on **AWS**, includes the design and selection of relevant hardware to support the installation/upgrades of **Ubuntu, CentOS**.
* Deployed **Splunk forwarders**, **indexers** and **search heads** to monitor, analyze and visualize the AWS VM's on the Splunk dashboard that helps in increasing the cluster performance.

**Environment:** Amazon Web Services, OpenStack (Kilo/Liberty), Chef, Ansible, Docker, Kubernetes, Maven, Jenkins, GIT, Cassandra, AEM, Python, Openshift, Jira, Dynatrace, Terraform.Elasticsearch, Logstash, ELK, AWS Lambda, CloudWatch, S3 bucket.

**Sr. DevOps Engineer Aug 2018 - Aug 2019**

Client: Evolent Health, Arlington, VA

* Provided highly durable and available data by using **S3** data store, **versioning**, **lifecycle policies**, and create **AMI**s for mission-critical production servers for backup.
* Designed the data models to be used in data-intensive **AWS Lambda** applications. which are aimed to do complex analysis creating analytical reports for **end-to-end traceability**, **lineage**, and definition of Key Business elements from **Aurora**.
* Migrating Data from a **PostgreSQL** DB Instance to an **Aurora** PostgreSQL DB Cluster by Using an Aurora **Read Replica** by configuring **VPC** security groups to secure network access to the DB cluster.
* Wrote **Terraform** scripts to automate **AWS** services which include Web servers, **ELB**, **Cloud Front** distribution, **Database**, **EC2**, **database security** groups and **S3** bucket.
* Managed multiple **AWS** accounts with multiple **VPC**'s for both production and non-prod where primary objectives included **automation**, **build out**, **integration** and **cost control**.
* Experienced in Provisioning of **IaaS** & **PaaS** **Virtual Machines** and Web/Worker roles on **Microsoft Azure Classic** and **Azure Resource Manager.**
* Worked on Configuration of the **Internal load balancer**, load balanced sets and **Azure Traffic manager.**
* Hands on experience on **Azure VPN-Point to Site**, **Virtual Networks**, Azure Custom security, endpoint security and firewall.
* Wrote **Chef Cookbook/Recipes** using **Chef Resources**, **Ruby DSL**, **Chef Handlers Report** and **Exception**, **Test Kitchen**, **ChefSpec** and **Vagrant**.
* Maintained automated 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.**
* Created additional **Docker** Slave Nodes for **Jenkins** using custom **Docker Images** and pulled them to Cloud, worked on all major components of Docker like **Docker Daemon**, **Hub**, **Images**, **Registry**.
* Implemented a load balanced, highly available, fault tolerant, auto-scaling **Kubernetes** **AWS** infrastructure and **microservice** container orchestration.
* Leveraging **kops** implemented a **Kubernetes** Container Orchestration solution within **AWS** allowing for easy management, creation and recovery of **AWS assets**.
* Used **MAVEN** as a build tool on **Java** projects and written maven **POM** files from scratch for multi-modules projects for the development of build artifacts on the source code.
* Provisioned the highly available **EC2** instances using **Terraform** and **CloudFormation** and wrote plugins to support new functionality in **Terraform.**
* Provided automation, **continuous integration** through **Jenkins** and Implemented **Jenkins Workflow** and Plugins for repeatable deployments of **multi-tier** applications, **artifacts** and services to **Docker.**
* Integrated AWS **Dynamo DB** using AWS lambda to store the values of items and backup the **Dynamo DB streams** and Utilized **Cloud Watch** to monitor resources such as **EC2**, **Amazon RDS DB** services, **Dynamo DB** tables, **EBS** volumes, to set alarms for notification or automated actions, and to **monitor logs** for a better understanding and operation of the system. Integrated AWS **Dynamo DB** using AWS lambda to store the values of items and backup the **Dynamo DB streams**.
* Deployed and configured **Elasticsearch**, **Logstash**, and **Kibana** (**ELK**) for log **analytics**, **full-text search**, application **monitoring** in integration with AWS **Lambda** and **CloudWatch**. Then store that logs and metrics into an **S3 bucket** using Lambda **function**.
* Designed an **ELK** (Elastic Logstash Kibana) system to monitor and search enterprise alerts and configured **ELK** stack in conjunction with **AWS** and using Logstash to output data to **AWS S3**. Terminated SSL on to **Nginx** to remove overhead on **API** servers.
* Implemented **JIRA** to track all issues pertaining to the **software development lifecycle** and Integration of **JIRA** with **Git** repository to track all code changes.

**Environment:** AWS (EC2, EBS, S3, VPC, Code Deploy, Code Commit, Elastic search, Lambda, API Gateway, Route 53 (DNS), ECS, CloudFormation, RDS, Dynamo DB, SNS, SES, SQS, IAM, RedShift, Ops Works), Maven, Chef, Docker, Kubernetes, OpenStack (kilo/ Juno), Jenkins, GIT, Python, Datadog, Jira.

**DevOps Engineer July 2016 – June 2018**

Client: General Electric, Hyderabad

* Performed the automation deployments using **AWS** by creating the **IAM**s and used the code pipeline plugin to integrate **Jenkins** with **AWS** and created the **EC2** instances to provide the virtual servers.
* Written AWS CloudFormation Templates for different services like **CloudFront** Distribution, **API** Gateway, **Route 53**, **Elastic Cache**, **VPC**, **Subnet Groups**, **Security Groups**.
* Configured **AWS IAM** and **Security Groups** in **Public** and **Private Subnets** in **VPC,** managed **IAM** accounts (with **MFA**) and **IAM** policies to meet security audit & compliance requirements.
* Involved in many different concepts of **Chef** like **Roles**, **Environments**, **Data Bags**, **Knife**, and **Chef Server** Admin/Organizations. Written **Chef Recipes** to automate build/deployment process and data bags in Chef for better environmental management.
* Wrote **Chef Cookbooks** for various **DB** configurations to modularize and optimize product configuration, converting production support scripts to Chef Recipes and **AWS** server provisioning using **Chef Recipes**.
* Tested Chef Cookbook modifications on cloud instances in **AWS** and using **Test Kitchen** and **Chef Spec** and used **Ohai** to collect attributes on a node. Worked on **Chef DK** which takes of care creating **cookbooks** and **recipes**.
* Implemented a Continuous Delivery pipeline with **Docker**, **Jenkins** and **GitHub**, whenever there is a change in **GITHUB**, our Continuous Integration server automatically attempts to build a new **Docker** container from it.
* Responsible for installation & configuration of **Jenkins** to support various **Java builds** and Jenkins plugins to automate continuous builds and publishing **Docker Images** to the **Nexus Repository**.
* 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 the **OpenShift Enterprise** which allows developers to quickly develop, host and scale applications in a self-managed cloud environment.
* Worked on **NoSQL** database **MongoDB** to replica setup and sharding. Also experienced in managing replica set. Installed, Configured, and Managed **Monitoring Tools** such as **Nagios** for Resource/Network Monitoring.
* Used Python libraries such as **Paramiko**, **Pycrypto**, **XML** parser and **logging** libraries to develop **automatic** storage, **networking** deployment tool on **scale-out** environment with Linux and **Troposphere** libraries to create AWS Cloud Formation descriptions
* Developed self-service/automation tools leveraging **Python ec2-boto**, **fabric** and **Jenkins** which increased the efficiency of **DevOps** and development tasks.
* Used **Selenium** for continuous inspection of code quality and to perform automatic reviews of code to detect bugs. Automated **Nagios** alerts and email notifications using **Python** script and executed them through **Chef**.
* Installed, configured and maintained web servers like **Apache Web Server** and **WebSphere Application Server** on **Red Hat Linux.**

**Environments:** AWS (EC2, S3, Route 53, EBS, Security Group, Auto Scaling, and RDS), GIT, Chef, Docker, Selenium, Maven, Jenkins, ANT, Python, Jira, Nagios.

**Build & Release Engineer Jun 2015 - July 2016**

 Client: Foxfire Technologies, Hyderabad

* Responsible for Creating Builds, Staging and deploying Software Content across multiple environments **(DEV, QA, Demo, Test, Staging, UAT and Production)**.
* Designed and managed Build and Release cycle activities in Agile Methodologies. Developed build and deployment processes for Pre-production environments. Managed **Sonatype** **Nexus** an enterprise-level package management server.
* Used **Puppet** to manage Web Applications, Configuration Files, Database, and Packages. Used **Puppet** to automate Configuration management & Applications.
* Implemented **Puppet** modules for server housekeeping. Built a new CI pipeline. Testing and deployment automation with Puppet.
* Enhanced Configuration Management using **PUPPET** to assist with automated, repeatable, and consist configuration and application deployments.
* Automated the Builds of **Java** and **PHP** applications using **Bamboo** CI tool. Integrated the **Bamboo** with **Maven**
* Administration of **Bamboo**. Implemented a Continuous delivery framework using Bamboo, and Oracle in Linux Environment.
* Installed and configured **Bamboo** server for CI to automate the release management for applications and created a continuous delivery pipeline.
* Installing, configuring and maintaining DB servers, LAMP server and file sharing servers like **Samba, NFS, FTP and Web Sphere** & **Web Logic** Application Servers.
* Migrating **J2EE** applications to **UNIX** and Windows environments by configuring application via Subversion, and **Jenkins** also Worked closely with the Development Team in the design phase and developed Use case diagrams using Rational Rose.
* Setting up Jobs for Plugin Management, Build/Deploy Pipeline setup and User management.
* Maintained build related scripts developed in shell for **Maven** builds. Created and modified build configuration files including **POM.xml**. Configured **Jenkins** to build Java Code using Meta Case Software and executed CI process on the Java code generated.
* Installed and configured database agents to troubleshoot slow response times and excessive loads.
* Developed and implemented Software Release Management strategies for various applications according to the **Agile process**.
* Configured Linux servers for clusters of Oracle Real Application and configured **SAN** based mount points as well.
* Integrated Unit Tests and Code Quality Analysis tools like **JUnit, PMD**, find bugs and check style.

**Environments:** Maven, Bamboo, Puppet GIT, Chef, Docker, Selenium, Maven, Jenkins, ANT, Python, Jira, Nagios.

**(Linux Administrator) Dec 2014 – June 2015**

Tryamsoft Solutions – Hyderabad

* Management of user accounts, files and directory permissions, and securing **UNIX** systems to match industry best practices. Wrote **shell scripts** for automation of daily tasks and created jobs as per performance of the client’s network statistics(/var/log/messages). Built backup **scripts** to store VM images on local machine and mirrored the images on to web-server, with rsync.
* Experience in Installation, Configuration, Upgrade, Patches, and performance tuning on **UNIX** and system software & hardware.
* Monitoring of web servers and other services using **Nagios** monitoring tool.
* Experienced with inter networking using TCP/IP and resolving network connectivity using tools like dig, **nslookup**, ping. Configured kickstart server and updating/applying patches to the servers using **Red Hat** Satellite server. Maintained **Samba** File Server for user authentication, **syslog domain** and file sharing.
* Involved in designing Firewall rules for new servers to enable communication with applications, Oracle 10g servers.

**Environments**: LINUX/UNIX, Solaris, Virtual Machines, Samba Server, NFS Server, NIS Server, TCP/IP, Oracle 10g, Kickstart, Red Hat Satellite, **VM** ware/Box.

**(Systems Engineer) Jan 2013 – Aug 2014**

Tvisha Technologies - Hyderabad

* Administered **Redhat** **Enterprise Linux 5.x/ 4.x, OEL 5.x, & Solaris 9** Servers by testing, tuning, upgrading, patching and troubleshooting both Physical and virtual server problems.
* Installed and maintained regular upgrades of Red Hat Linux Servers using **kickstart**-based network Installation and maintained a **RHEL satellite server, Apache web server** and **sftp**.
* Used Veritas File system and Veritas Volume Manager 5.0 to configure the **RAID 1** & **RAID 5** Storage System for more redundancy.
* Created the **LDAP** Scripts which monitors the **LDAP** connectivity and alerts the Admin Group if the connection is closed. Also administered **Active Directory** Servers.
* Configured Proxy Server (Squid), **DNS**, **FTP** and **DHCP** servers on Red Hat Enterprise Linux and maintained system securities using **IPTABLES**.
* Developed **PERL & Shell** scripts for automation of the build and release process. Developed automation scripting in **Python** to deploy some applications.
* Upgraded Red hat Linux OS on **WebSphere** and **Oracle** database servers from **V3**, **V4** to **V5**. Monitored servers, switches, ports etc. with **Nagios** monitoring tool.
* Responsible for setting up **Cronjobs** scripts on production servers and implementation of password less (**SSH**) authentication between servers.

**Environment:** RedHat Enterprise Linux 5.x/4.x, OEL 5.x, Solaris 9, LVM, RAID, Oracle, WebSphere, MySQL, TCP/IP, DNS, NIS, NFS, ClearQuest, Apache Tomcat.