**Deekshitha**

**Sr. Cloud / DevOps Engineer**

**Email: deekshita71@gmail.com**

**Contact no: +16145268928**

**LinkedIn:** [linkedin.com/in/deekshitha-d-5818951b4](mailto:deekshithadasari21@gmail.com)

**Professional Summary:**

Professional experience of **over 7 + years** in the field of IT that includes Development, designing with a foremost focus on DevOps, provisioning, build and release, Cloud Automation, Continuous Integration/Deployment, Configuration Management, Source Code Management, Virtualization technologies, System Administration and cloud migration.

* Strong Experience in **Amazon Web Services Resources** like **AWS EC2, VPC, AWS Security Groups, ECS, EKS, Lambda, API gateway, AWS Route 53, ELB, AMI, SNS, IAM, S3, AWS Auto scaling, AWS RDS, AWS Cloud Formation, ELK cluster management and AWS Cloud Watch.**
* Experience in maintaining **Amazon Web Services** and Hands on experience in **Architecting Legacy Data Migration projects** such as Teradata to **AWS Redshift, migration, Teradata to Snowflake Migration**, AWS Cloud from on-premises.
* Proficient in designing secure, high performance and Available applications on **AWS** across Availability Zones and Experience on **fault tolerance, Disaster Recovery** and **auto-scaling** in Amazon Web Services **Cloud Formation.**
* Deployed application which is containerized using **Docker** onto a **Kubernetes** cluster that is managed by **AWS EKS, ECS** and **Fargate**. Also Experienced Configuring Kubectl to interact with Kubernetes infrastructure and launched a cluster of worker nodes on Amazon **EC2** instances.
* Created **AWS VPC**, subnets, route tables, assigning CIDR ranges. Created a **VPC Privatelink** for a secure connection, to avoid the exposure of the application to the public internet and without the actual use of VPC peering, NAT gateways etc. Also, used VPC flow logs to capture the information of the IP traffic, and stored this data in **Amazon CloudWatch logs**.
* Experience in both **CloudFormation** and **Terraform** to **automate AWS environment** creation along with the ability to **deployment on AWS**, using **build scripts** (AWS CLI) and **automate** solutions **using Shell and Python.**
* Hands on experience in **Azure** Development, worked on **Azure web application**, **App services, Azure storage, Azure SQL Database, Virtual machines, Fabric controller, Azure AD, Azure search**, and **notification hub.** Experience in using **Azure service fabric** to package, deploy, and manage scalable and reliable **Microservices and containers**.
* Experience in Automating, Configuring and Deploying Instances on **Azure environments** and in Data centers and **migrating** on premise to **Windows Azure** using **Azure Site Recovery** and **Azure backups**.
* Experience in designing User Defined Routes with custom route tables for specific cases to force tunneling to the Internet via on-premises network and control use of virtual appliances in the customer's **Azure** Environment.
* Implemented high availability with **Azure Classic** and Azure Resource Manager deployment models. Involved in automation of Azure Virtual Networks, Network Security Groups and access control rules.
* Experience with **IAC** technologies such as **Cloud Formation, ARM** and **Terraform**. experienced in using **Terraform** for building, changing, and versioning infrastructure and manage existing and **cloud infrastructure** as well as custom in-house solutions.
* Experienced in using **GCE, GAE, GKE, GCS, GCR** and **Managing service accounts** and architecting and building the platform using terraform in **Google Cloud Platform.** Written **custom Terraform scripts** to spin-up **Admin and app clusters, Load balancers, Storage buckets, Halyard VM** to manage **Rancher** in **Google cloud Platform**.
* Experienced working with Openshift private cloud, worked on installing, configuring and managing multiple openshit clusters in both on-prem and AWS cloud. Worked with application teams to deploy/migrate applications into Openshift cluster.
* Expertise in **DevOps** in **GCP, AWS,** **Azure platform** and **On-Prem** which includes technologies and platform like **Azure DevOps, AWS CodeBuild, AWS CodeDeploy, AWS CodePipeline, Jenkins, Bamboo, Drone(CI), Spinnaker(CD), Atlassian Stack, Maven, Gradle, NPM, GitHub, Bitbucket, Terraform, Chef, Ansible, Puppet, Vagrant, CVS, Tomcat, JBoss etc.**
* Experience in Orchestration tools **like Kubernetes, Docker Swarm**. designing and deploying container-based production clusters using Docker, **Kubernetes, Docker Swarm**. Developed Docker images to support Development and Testing Teams and their pipelines; distributed Jenkins, Selenium, JMeter and ELK stack images.
* Experienced in **Kubernetes in Different environments** including **GKE, EKS, AKS, openshift** and on-prem to produce production-grade **Kubernetes clusters** that allow enterprises to reliably deploy and run containerized workloads across **private and public clouds and on-prem**.
* Experienced with orchestrating microservices **deployments into Kubernetes Clusters** using **Rancher, AWS Code Deploy, Azure DevOps pipelines**.
* Experienced configuring, implementing and managing **Istio and Helm** service mesh with **Kubernetes clusters.**
* Experience in **Ansible** setup, managing hosts file, Using **YAML** linter, authoring various playbooks and custom modules with Ansible and Ansible Playbooks to automate in **AWS** services like Launching provisioning **IAM, Configuring VPC, EBS,** Monitoring using **CloudWatch** and **CloudTrail.**
* Experience on **Ansible** and **Ansible Tower** to automate repetitive tasks, to deploy critical applications quickly, and proactively manage the changes and **wrote** **200+** custom **playbooks** to manage Web applications and infrastructure.
* Worked on Configuration Management tools like **Ansible**, Chef, Puppet to automate the environment that included developing playbooks, cookbooks, manifests in Ruby and YAML scripts. Experience on **puppet/chef configuration** management of several hundreds of servers across multiple platforms like AWS cloud, VMWare and physical servers.
* Created images and containerized the application build using **Dockerfile** and **Docker-compose** YAML files. Created and built micro services using the docker compose and created, managed docker containers on a cluster hosted on a serverless infrastructure using **ECS** by distributing the application traffic in **ELB**, Cloud Front to distribute content to edge locations, Cloud watch to set alarms and notifications.
* Hands on experience in Jenkins on creating **CI/CD pipelines** and creating freestyle projects pipeline, multi branch pipeline Jobs. Used Jenkins pipeline to provide API access to a private repo and deploy the artifact to EC2 instance. Integrated matrix Authorization Strategy plugin with **Jenkins** to specific permissions to **Jenkins jobs.**
* Developed **Python** and **shell/bash scripts** for TC Server/Tomcat web application servers and automation of the build and release process and to automate certain processes for the smooth process of build tool like Maven. Deployed **JAVA/J2EE** applications through WebLogic/WebSphere Application servers.

**TECHNICAL SKILLS:**

|  |  |
| --- | --- |
| **Cloud Environments** | Amazon Web Services (AWS), Azure, Google Cloud Platform (GCP), PCF, |
| **Configuration Management Tools** | Ansible, Ansible Tower, Chef, Chef Automate, Puppet |
| **Containerization Tools** | Docker, Kubernetes, OpenShift, Elastic Kubernetes Services (EKS), Docker swarm, Docker compose |
| **Virtualization** | Windows Hyper-V, Virtual box, Vagrant |
| **Operating Systems** | Linux (Red Hat 4/5/6/7, CENTOS), Ubuntu 12/13/14, Windows NT 2000/2003 |
| **Automation Tools** | ANT, Maven, Jenkins, Terraform |
| **Web Servers** | Apache HTTP 2.x ,3.x, Nginx, IIS, Apache Tomcat |
| **Database** | AWS RDS, MongoDB, MySQL, Postgresql, Amazon Aurora, MYSQL SERVER 2008/12, DynamoDB |
| **Scripting languages** | Python, Ruby, Bash shell, Power shell, JSON |
| **Networking/Protocol** | DNS, DHCP, WAN, TCP/IP, NIS, NFS, SMTP, LAN, FTP |
| **Version Control Tools** | GIT, Bitbucket, SVN, GITLAB, GITHUB |
| **Monitoring Tools** | Splunk, Nagios XI, Cloud Watch, ELK, New Relic |
| **Volume Manager** | LVM, VVM |
| **Application Servers** | Oracle Web Logic (8/9/10), IBM Web Sphere (4/5), JBOSS |

**Professional Experience:**

**InComm, Atlanta, GA** **Dec 2019 to Present**

**Sr. Cloud/DevOps Engineer**

* Worked on **Cloud/DevOps** team in a **TDD** environment to develop the **Microservices native Cloud Platform** from scratch to on-board domain/application teams to **migrate** the **on-prem Monolith applications** to a **Microservices** based architecture running on **AWS and OpenShift cloud**. Also worked on **on-boarding Green-Field** applications on to cloud infrastructure by designing, building and managing the cloud infrastructure.
* Created **AWS** **CloudFormation** templates to create 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** and also Worked on Simple Storage Service (S3), Simple Notification Service (SNS) and CloudWatch.
* **Migrated services** from a managed hosting environment to **AWS** including service design, network layout, data migration, automation, monitoring, deployments and cutover, documentation, overall plan, cost analysis, and timeline.
* 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**.
* Worked on Migrating applications from on-prem to ECS with Fargate and EKS.
* Worked on **AWS** Auto Scaling in **VPC**, for providing high availability of applications and **horizontal scaling** of EC2 instances and monitoring the resources such as **EC2 instances** CPU memory, applications by using AWS Cloud Watch.
* In order to perform data manipulation of incoming events created functions in **Lambda** and stored results in Amazon S3 and have setup triggers to inform via email through **SNS.** Using Amazon **Kinesis** and **Lambda** redesigned list management system to run as a scalable microservice within the Amazon cloud (AWS).
* Worked on **REST APIs**in configuring the changes and to maintain Index points.
* **CloudWatch** for monitoring CPU, memory, disk space and store logs and setup alerts for thresholds using Simple Notification Service (SNS) Setup **CloudTrail** to record API calls.
* Worked with Red Hat OpenShift Container Platform for **Docker** and **Kubernetes**. Used **Kubernetes** to manage containerized applications using its nodes, ConfigMaps, selector, Services and deployed application containers as Pods.
* Worked on creating **Terraform** scripts to **deploy OpenShift on AWS cloud** and also Worked on Building, managing and maintaining **OpenShift cluster** and orchestrated **OpenShift** deployments using **Jenkins** and **Helm**.
* Developed **Kubernetes** templates for various applications like **Cassandra**, **Grafana**, **Kafka** and setting up **Kubernetes** **Clusters** for running microservices and pushed microservices into production with **Kubernetes** backed Infrastructure.
* 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**.
* Used **Jenkins** **pipelines** to drive all micro services builds out to the Docker registry and then deployed to Kubernetes environments in **Openshift and AWS EKS**. Developed **microservice** on boarding tools leveraging **Python** and **Jenkins** allowing for easy creation and maintenance of build jobs and **Kubernetes** to deploy services.
* Configured **Kubernetes Replication** controllers to allow multiple pods such as Jenkins master server in multiple minions. Automatic deployment of **SCM/CI/CD** applications and idempotently to reduce the time costs by creating and troubleshooting the **Ansible scripts** and **Kubernetes YAML files**.
* 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**. Implemented the **Ansible** configuration management system and managed Webapps, used **Ansible Tower** to automate repetitive tasks, quickly deploy critical applications and proactively manages change.
* Deployed Jenkins servers for application version builds to dev, staging and prod across **VPC** subnets groups using VPC peering. Created **Jenkins** jobs to create **AWS** Infrastructure from **GitHub** repositories containing **Ansible** playbook.
* Extensively worked on **Ansible** to configure the servers, deploy software, and orchestrate continuous deployments or zero downtime rolling updates and was also experienced in customizing the **Ansible** **modules** for finding facts and taking actions to pause/unpause the alarms during deployments.
* Worked closely with the development and operations organizations to implement the necessary tools and process to support the automation of builds, deployments, using **Ansible** also manage Web applications, Environments configuration Files, Users, Mount points and Packages using Ansible.
* Creating an automated Build and Deployment Platform and Driving the teams and Orchestrated deployments using **Jenkins, Jfrog, Docker and other DevOps tools**.
* Worked on configuring and integrating **Jfrog, SonarQube, BlackDuck, Checkmarx, PrismaScan, AquaSec, Hashi Vault, Perfecto, Cucumber, Katalon, Ansible Tower, Splunk, Openshift and AWS with CI/CD**.
* Delivering quality code that meets requirements and specifications on time and accomplishes all aspects of the software configuration management (SCM) process, **DevOps and Build/Release** management.
* **Orchestrating CI/CD** processes by triggers, human input, and dependency chains and environment and Writing Jenkins files and Groovy scripts to create pipeline jobs based on the requirement.

**Environment & Tools:** AWS EC2, S3, VPC, Route 53, CloudFormation, Azure, Docker, Kubernetes, Ansible, Terraform, Maven, ANT, Jenkins, GIT, SVN, JIRA, Nagios, Splunk, RHEL, CentOS, Shell scripting, Python Scripting.

**DELL EMC Hopkinton, MA Dec 2018 – Dec 2019**

**Sr. Cloud/DevOps Engineer**

**Responsibilities:**

* Implemented Enterprise Integration architecture using **Azure ServiceBus, AppServices, Active Directory, Storage, hybrid connection manager, active directory authentication** for azure SQL server and other offerings by azure.
* Designed **Network Security Groups (NSGs)** to control inbound and outbound access to network **interfaces (NICs), VMs and subnets.** Provided high availability for IaaS VMs and PaaS role instances for access from other services in the VNet with **Azure Internal Load Balancer.**
* Migrating the existing applications, simplifying the deployment and management of **Microservices** based on applications using **Azure Kubernetes** Services (**AKS**). Responsible for Configuring **ARM** Templates, **ASG** Application Security Groups, Role-Based Access Control. Writing scripts with **PowerShell** for Run books and SQL **Azure** from continuous web job.
* Proficient in migrating on-premises workloads to cloud provided 3 tier managed service support for client dev, test, prod **azure** environments. Used AD connect to sync on-premises windows server to Active Directory to monitor azure resources and migrated SQL Server Database **to SQL Azure Database** using SQL Azure Migration Wizard and then Deployed application to **Azure Cloud**. Worked on CI/CD pipeline for Azure Cloud Services using **CHEF**.
* Designed a robust future state Ingestion Framework solution leveraging **Azure Data Factory, Azure SQL database** & Data warehouse. Deployed and managed **Azure storage** for SQL DB's (SaaS and PaaS applications), SQL Server on Azure, Windows and Linux instances among other azure cloud services.
* Involved in building **Azure management template** and responsible for creating virtual machines, serverless API using **azure functions**, responsibilities **Azure service fabric** to deploy, manage scalable microservices and containers, blobs for storage.
* Implemented network traffic rules and access control list for virtual network in **Microsoft azure** cloud service platform. Experience to orchestrate the jobs to and from **Azure automation.**
* Worked on various **AWS** services which includes importing necessary volumes, launching **EC2 instances, S3** for objectstatic webpages and created security groups, auto scaling, load balancer, Route 53 and SNS as per architecture using IAAC tools like Terraform.
* Configured and managed various AWS Services including **EC2, RDS, VPC, S3, Glacier**,**Cloud Watch, Cloud Front, and Route 53** etc.
* Configured various performance metrics using **AWS Cloud watch & Cloud Trial**
* Worked on configuring Cross-Account deployments using **AWS Code Pipeline**, **Code Build** and **Code Deploy** by creating Cross-Account Policies & Roles on **IAM**.
* Deployed Azure Resource Manager **JSON** Templates from PowerShell worked on Azure suite: Azure SQL Database, **Azure Data Lake**, Azure Data Factory, Azure SQL Data Warehouse, Azure Analysis Service
* Installed Workstation, Bootstrapped Nodes, Wrote Recipes and **Cookbooks** and uploaded them to **Chef Server**, Managed On-site OS Applications.
* Worked with **Terraform** Templates to automate the **Azure IaaS** virtual machines using **terraform** **modules** and deployed virtual machine **scale sets** in production environment.
* 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.
* Automation and the deployment of **cookbooks** on nodes through wrapper scripts and running the chef client on them in a Chef-Solo environment. Rolled out **Chef** to all servers, used chef node database to drive host configuration monitoring, configured Chef Server, Chef Solo including bootstrapping of Chef Client nodes for provisioning.
* Installed, tested, deployed monitoring solutions with **Splunk** services, deployment Server Topologies in Splunk.
* Used **Git** with **Jenkins** to integrate to automate the code checkout process. Coordinated developers with establishing and applying appropriate branching, labelling/naming conventions using **GIT source control**. Created the naming strategy for branches and labels and involved continuous integration system with **GIT version** control repository and Continually build as the check-in's come from the developer.
* Worked on a continuous integration tool **Jenkins** for daily automation process. Configured **CI/CD** pipeline for the automatic deployment of **artifacts**/application to the required servers or environments in Jenkins CI, and Clean for the new build process whenever code commits are done to **SCM tool Git**.
* Maintained build related scripts developed in shell for **Maven** builds. Created and modified build configuration files including **POM**.xml, converting build.xml into pom.xml in order to build the applications using **MAVEN**.
* Developed **Python, Perl** and **shell/bash scripts** for **TC Server/Tomcat** web application servers and automation of the build and release process and to automate certain processes for the smooth process of build tool like **Maven.** Developed Build related scripts in **Maven** and modified configuration files such as **Maven's Pom.xml**.

**Environment**: Azure, Azure Data Lake, AWS, Chef, Python, Linux, Maven, Jenkins, Python, Shell scripts, Splunk, GIT

**Genpact, Hyderabad, INDIA Jan 2017 to July 2018**

**Cloud/DevOps Engineer**

**Responsibilities:**

* Worked on AWS Auto Scaling in **VPC,** for providing high availability of applications and **horizontal** scaling of EC2 instances and monitoring the resources such as **EC2 instances** CPU memory through **Cloud Watch** based on a load of applications by using **AWS Cloud Watch.**
* Used **AWS CloudTrail** to track the user activity, compliance, operational auditing, and risk auditing of AWS account.
* In order to perform data manipulation of incoming events created functions in Lambda and stored results in Amazon S3 and have setup triggers to inform via email through **SNS** and have also created topic.Using Amazon **Kinesis** and **Lambda** redesigned list management system to run as a scalable microservice within the Amazon cloud (AWS).
* Stored the data in the **DynamoDB** after being pulled from the shards (Kinesis) by **EC2 instances** for the analysis.
* CloudWatch for monitoring CPU, memory, disk space and store logs and setup alerts for thresholds using **Simple Notification Service (SNS)** Setup **CloudTrail** to record **API calls**.
* Created multiple **Virtual Private Clouds (VPC)** within the domain, associated Elastic IPs with VPCs for the connectivity of our nodes and Initiated VPC Peering Connections for those VPCs authorized to talk amongst each other. Created automated pipelines in **AWS codepipeline** to deploy **docker** containers in AWS ECS using services like CloudFormation, codebuild, codedeploy. Written Lambda functions using python, used lambda functions to automate **aws** infrastructure.
* Worked on installing **Docker,** used Docker **toolbox**, creation of custom **Docker**container images, **tagging**and pushing the **images,** removing images, and managing Docker **volumes.** Used **Docker Registry API**to interact with the registry.
* Experience in implementing the **Docker** for wrapping up the final code and setting up dev and test environment using Docker **Hub** and Docker Container Network.
* Working on AWS Auto Scaling for providing high availability of applications and EC2 instances based on the load of applications by using Cloud Watch in **AWS.** Used **GCP** App Engine for deploying and scaling web applications and services developed with Java. Strong Hands-on Experience in Google container builders and **GCP** client libraries and cloud SDK's and Used **GCP** HTTP(S) load balancing with Google Cloud storage buckets.
* Worked with [**Google Container Service**](https://pivotal.io/platform/pivotal-container-service)to produce production-grade **Kubernetes** that are reliably to deploy and run containerized workloads.
* Set up **Kubernetes** (k8s) **Clusters** for running **microservices** and Pushed **microservices** into production with **Kubernetes** backed Infrastructure.
* Implemented infrastructure as a code processes (IaaC) using Terraform by creating templates, scripts to automate provisioning of services in Google Cloud Platform (GCP) and Amazon Web Services(AWS) cloud environment.
* Created **docker** container, exposing and publishing ports to the specified host/container by mapping it first to the port, built both TCP and UDP ports and also established a secure network using **docker** network connect command.
* Worked on **Kubernetes**to orchestrate deployment, scaling, clustering, and management of containerized applications as **Pods,** Docker swarm in managing Docker containers, and **Kubernetes Clusters**to support automatic high availability, load balancing, health checks, and failover.
* Used **Kubernetes** charts using **Helm** and created reproducible builds of the Kubernetes applications, managed Kubernetes manifest files, and managed releases of **Helm** packages. Deployed, managed container replicas onto a node using Kubernetes and created Kubernetes clusters work with Helm charts running on the same cluster resources.
* Performed OS installations, OS Upgrades, & Server Patching by configuring **PXE & DHCP** server using Kickstart & Jumpstart scripts on **Red Hat Linux 5.x, 6.x & 7.x**

**Environment**: Azure, Chef, Python, Linux, Docker, Kubernetes, Maven, helm, Jenkins, Python, Shell scripts, Splunk, GIT, Red Hat Linux 5.x, 6.x & 7.x., SQL Server

**KnoahSoft, Hyd, India Jan 2016- Dec 2016**

**System Administrator**

**Responsibilities:**

* Setting up, configuring and troubleshooting TCP/IP, DHCP, DNS, NFS, CIFS, NTPand Sambaservers on different multi-platforms (Windows, Linux). Local and remote administering of servers, routers and networks using Telnet and SSHlogin using SSH keygentools for password-less setup and agent forwarding on **Linux.**
* Used **ANT** build tool for scripts deployment and processes using **Jenkins** to move from one environment to other.
* Automating various activities using scripting languages like **Perl, Python, PowerShell** based on the requirements. Provided technical solutions and investigated alternative methods for solving problem to configure, imaged, deployed, and integrated new equipment into the environment.
* Performed daily system monitoring, verifying the integrity and availability of all hardware, server resources, and reviewing system and application logs.
* Migrated the entire Application from **JBOSS** to **Tomcat environment.** Experience in setting upSquid Proxy Server for internet sharing and to block the unwanted sites. Managing HP-UX, Compaq and Linux workstations and servers.

**Environment:** Linux, UNIX, Microsoft Windows, LAMP, TCP/IP, DNS, NFS, NIS, SAMBA, Perl, Python, PowerShell Java, SQL, Oracle, Agile, Bash, Shell, WebLogic, Apache, Subversion

**S4 Consultants, Hyderabad, India Aug 2014 – Dec 2015 Linux Administrator**

**Responsibilities:**

* Installation and configuration of **Linux**, UNIX, SUSE & RedHat 5.x/6using Kickstartwith PXEfor LAMP (Linux, Apache, MySQL and Perl/PHP). Worked on Linux Package installation using RPM and YUM, provisioned system with LVM.
* Experience in solving day-to-day technical issues on system integration, managing user and security administration related to Windows and Linux OS. Responsible for maintaining the various servers and Production Support of different applications in **RedHat Enterprise** **Linux** and Windowsenvironment.
* Manage users, groups and assigning file access **privileges** using **ACLs**, changing ownerships of **users** and groups using **chown**, **chgrp** commands, modifying the file permissions using **chmod**, **SUID**, **SGID**, **sticky** **bits** and **immutable** **bits**.
* Implementation of automatic **SSH login** into remote Linux servers and apply the required modifications, User and Group administration. Wrote **shell scripts** to automate the periodic tasks like removing old logs and files, applying patches and making backup and recovery.
* Created various **CRON jobs** with **shell/bash scripts** to automate the administrative tasks like taking backups, applying updates based on schedule and maintained **version control** using **GIT** **repository**.
* Developed, customized and build packages on Solaris and rpms on Linux for deployment on various servers through **Software Development Life Cycle (SDLC).**
* Installing, configuring, and implementing of **RAID 0, RAID 1, and RAID 5** levels using Logical Volume Manager (LVM), VERITAS Volume Manager (VVM)and Solaris Volume Manager.
* Creation and cloning of Linux VM’s, templates using VMware Virtual Client migrating servers between ESX hosts.
* Written **Python, Pearl and Bash** to **automate CRON** jobs and system maintenance.

**Environment:** Linux, UNIX, Microsoft Windows, LAMP, TCP/IP, DNS, NFS, NIS, SAMBA, Perl, Python, PowerShell Java, SQL, Oracle, Agile, Bash, Shell, WebLogic, Apache, Subversion