

# Shilpa

Sr. Cloud / DevOps Engineer



+1 567.250.2450 [shilpadev213@gmail.com](mailto:shilpadev213@gmail.com) [linkedin.com/in/shilpa-d-631954207](https://www.linkedin.com/in/shilpa-d-631954207)

## Professional Summary

IT professional with **7+years** of experience which includes **Cloud/DevOps Engineer, Build/Release Management, Linux/Unix** and Network **administration**, CI/CD (Continuous Integration/Continuous Delivery), Software Configuration Management (SCM), **System administration and Cloud Management** all inside that suites the necessities of a domain under the DevOps culture.

- Experience in various Azure Services like **Compute (Web Roles, Worker Roles), Caching, SQL Azure, NoSQL, Storage and Network Services, Azure Active Directory, Internet of Things (IoT), Azure Monitoring, API Management, Scheduling, Autoscaling, Azure Identity & Access Management, Data Factory, PowerShell** Automation, and other Azure Services.
- Hands on experience on Backup and restore **Azure services** and in Design and configure **Azure Virtual Networks (VNets), subnets, Azure network settings, DHCP address blocks, DNS settings, security policies and routing. Azure cloud services, Blob storage, Active directory, Azure Service Bus, Cosmos DB.**
- Experience in **Azure App & Cloud Services, Azure Data Factory, Azure SQL Data Warehouse, Power BI, Azure Blob Storage, Web API, VM creation, ARM Templates, PowerShell** scripts, IaaS, storage, network and database.
- Worked 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** also called as **Azure DevOps Services**.
- Experience in writing **ARM (Azure Resource Manager) Templates** for **Azure IaaS** (infrastructure as a code) using **Terraform** to build staging and production environments.
- Implemented a CI/CD pipeline using **Azure DevOps (VSTS, TFS)** in both cloud and on-premises with **GIT, MS Build, Docker, Maven** along with **Jenkins** plugins.
- Good understanding of Pivotal Cloud Foundry (PCF) Architecture (Diego Architecture), PCF components and their functionalities and experienced in using Pivotal Cloud Foundry (PCF) CLI for deploying applications and other Cloud Foundry (CF) management activities.
- Involved in designing and deploying a multitude application utilizing almost all the **AWS stack** (Including **EC2, S3, EBS, VPC, ELB, AMI, CloudWatch, Route53, RDS, Dynamo DB, Autoscaling, Security Groups, RedShift, Kinesis, CloudFormation, SNS, SQS, IAM**) focusing on high-availability, fault tolerance, and auto-scaling.
- Experience in using **Lambda** to read records from the **Kinesis Data stream** and invoke functions synchronously with multiple events that contain stream records.
- Responsible for configuring and maintaining Amazon **AWS IAM Service: IAM Policies, Roles, Users, Groups, AWS Access Keys and MFA**. Also utilized **Hashicorp Vault** for secrets Management.

- Created **AWS** launch configurations based on customized **AMI** and use this launch configuration to configure auto scaling groups and Implemented AWS solutions using **EC2, S3, RDS, VPC, DynamoDB, Route53, EBS, Elastic Load Balancer, Auto Scaling groups**.
- Experience in using **Kubernetes** as an open source platform for automating deployment, scaling and operations of applications containers across clusters of hosts, providing container centric infrastructure.
- Expertise in using **Kubernetes** for **running/managing containers, images and container snapshots**, while creating a platform for **automating**, deploying, scaling of application containers across **clusters** of hosts.
- Experience in working on **cluster** management on **Kubernetes** to create Pods, Nodes, and deploying **microservices** pulled from the **Docker Registry** and managed by **Kubernetes**.
- Experience in Implementation of Docker and Various Docker Orchestration tools such as **docker-compose, docker-swarm, docker trusted registry (DTR) and universal control plane (UCP)**.
- Orchestrated Docker containers using **Docker Swarm, Docker Compose, Docker Hub, Docker Weave**.
- Worked on Docker service rolling updated, **"BLUE GREEN DEPLOYMENT"** to implement zero downtime PROD deployments.
- Strong knowledge on **Ansible Roles**, inventory, **Ansible Tower, Ansible Galaxy** concepts and used **Ansible Control Server** to deploy plays and playbooks to the machines and systems in the inventory.
- Worked on integrating **Ansible Tower** with cloud environment, provided role-based access control, setup job monitoring, email notifications, scheduled jobs, multi-playbook workflow to chain playbooks.
- Proficient in implementing **Chef** Recipes for Deployment on build on internal Data Centre Servers and reused modified same **Chef** Recipes to create a Deployment directly into Amazon **EC2** instance.
- Experience in using **Puppet** to automate repetitive tasks, quickly deploy critical applications, proactively manage, change and create custom **Puppet** module skeletons.
- Experience in Automation **framework, Build, Release and Integration** in design and Implementation of **Continuous Integration, Continuous Delivery, Continuous Deployment**.
- Experience in Implementation of **Master/Slave** architecture, Installed Plugins, created project jobs and configured jobs **in Jenkins**.
- Implemented **ELK stack (Elasticsearch, Logstash and Kibana)** for Log indexing.
- Experience in **Bitbucket** Administration in creating repositories, configuring popular plugins.
- Configured **SonarQube** code quality tool and integrated it with **Jenkins** and Implemented SonarQube to analyze code quality code quality metrics, to verify the coding standards and setup quality gates to allow/fail builds as per requirement.
- Responsible for pom, Unit test cases, Sonar dashboard for **Selenium** Test cases for all the **micro services** and worked on the ms's logs and sonar dashboard monitor for all the existing **micro-services**.
- Hands-on experience with Source Controller tools such as **Subversion (SVN), Git** and **IBM ClearCase**.
- Experience in automating the processes with Build Management Tools such as **ANT, Maven, Gradle** and **MS Build**.

- Expertise in Installing, Configuring, Managing the monitoring tools such as **Splunk, NewRelic** and **Nagios** for Resource Monitoring/Network Monitoring/ Log Trace Monitoring.
- Ability to build deployment, build scripts and automated solutions using scripting languages such as **Groovy, Shell, Python and Ruby**.
- Expert in **JIRA** Administration and Maintenance, including creating projects, workflow schemas, user authorizations, backup and Restoration.
- Worked on web servers and application servers like Web logic, Tomcat, Web sphere, JBOSS to deploy code.
- High exposure to REMEDY and JIRA tracking tools for tracking defects and changes for Change Management.
- Working experience on Automation Testing frameworks like Selenium, Katalon Studio.
- Experience in installing, configuring, supporting and troubleshooting **Unix/Linux Networking** services and protocols like **NIS, NIS+, OSPF, LDAP, DNS, NFS, DHCP, NAS, FTP, SSH and SAMBA**.
- Strong ability to troubleshoot any issues generated while building, deploying and in production support.

## Technical Skills

Operating Systems	Windows, UNIX, Linux (RHEL/Centos/Ubuntu/AIX), Mac OS, Solaris
Networking/Protocol	TCP/IP, NIS, NFS, DNS, DHCP, WAN, SMTP, LAN, Telnet, HTTP, LDAP, POP3, FTP/TFTP, TLS/SSL, DHCP.
Cloud Computing	Amazon AWS & Microsoft Azure, OpenStack
Virtualization Platforms	VMware Workstation, Oracle Virtual Box, Windows, Hyper-V, vCenter, vSphere.
Containerization Tools	Docker, Kubernetes, Apache Mesos
Build Tools	ANT, Maven, Gradle
CI/CD Tools	Jenkins, Bamboo, TeamCity, Octopus
Bug Tracking Tools	Atlassian JIRA, Remedy, Bugzilla, IBM Clear Quest
Monitoring and Reporting Tools	Cloud Watch, Nagios, Splunk, ELK (Elasticsearch, Logstash, Kibana), Zabbix
Configuration Management	Chef, Puppet, Ansible
Version Controllers	SVN, GIT, Atlassian Bitbucket, CVS,
Programming and Scripting Languages	Java, Shell, Python, Perl, Ruby, Bash
Web Servers	Tomcat, Apache HTTP, WebLogic & WebSphere
Relational and NoSQL database	MSSQL, MySQL, Dynamo DB, MongoDB

## Professional Experience

### Sr. Cloud & DevOps Engineer

Client: Marathon Petroleum Corporation, Findlay-OH

July 2018 to Present

- Involved in migrating the application from **Infrastructure as a Service (IaaS) to Platform as a Service (PaaS)** by converting existing solution to Windows **Azure Worker Role** and then Configuring **ARM, VM, Networking, Private Static IP Address** and Public IP address.
- Configured **Azure Automation DSC** configuration management to assign permissions through **RBAC**, assign nodes to proper automation accounts and **DSC configurations**, to get alerted on any changes made to nodes and their configuration.
- Implemented Enterprise Integration architecture using **Azure Service Bus, AppService, Active Directory**, Storage, hybrid connection manager, active directory authentication for **Azure SQL** server and other services offerings by **Azure**.
- Created and configured **HTTP Triggers** in the **Azure Functions** with Application insights for monitoring and performing load testing on the applications using the **VSTS** and used **Python API** for uploading all the agent logs into **Azure blob storage**. Used **Azure blob** to access required files and **Azure Storage Queues** to communicate between related processes. Configuring the **Azure load balancer** to load balance incoming traffic to virtual machines in **Azure Cloud**.
- Responsible for creating and managing **Azure AD tenants**, manage users and groups, and configure application integration with **Azure AD**. Integrate on-premises **Windows AD with Azure AD**, configure **multi-factor authentication (MFA)** and federated single sign-on (SSO).
- Migrated data from on-premise **SQL Database servers** to **Azure SQL Database servers** sent by designing the **Azure Data Factory Pipelines** using the **Azure Data Factory** copy tool and Self-Hosted Runtimes.
- Implemented a CI/CD pipeline using **Azure DevOps (VSTS, TFS)**.
- Worked 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** also called as **Azure DevOps Services**.
- Implemented greenfield project using **Docker** and **Azure Kubernetes Service (AKS)**, including configuration standards, implementing infrastructure with secure networking, and **CI/CD** pipelines. Used **Azure Kubernetes Service** to deploy a managed **Kubernetes** cluster in **Azure**.
- Created **Azure cloud** using **Kubernetes** that supports **DEV, TEST, and PROD** environments. Implemented a production ready, load balanced, highly available, fault tolerant, auto scaling **Kubernetes infrastructure** and **microservice container orchestration**.
- Developed **ARM templates** to deploy all **Azure infrastructure** with nested templates, shared templates, logic to support environment sizing, environment types, naming standardization. Migrated core networking infrastructure to **Terraform** to better align with enterprise tooling this included **Gateways, Nets, Subnets, NSGs**.
- Created **Terraform templates** for provisioning virtual networks, VM Scale sets, Load balancers, NAT rules and used **terraform graph** to visualize execution plan using the graph command. Used **Terraform** to deploy the infrastructure necessary to create development, test, and production environments for software development project
- Worked on deployment automation of all the **Microservices** to pull the image from the private **Docker registry** and deploy to **Docker swarm** cluster using **Ansible**.

- Implemented **Blue/Green Deployments** with Zero downtime where the current environment is replicated to latest version with **Kubernetes** to resolve the Bug fixes and redirect the traffic to it once the issues are resolved.
- Configured **Kubernetes** Replication controllers to allow multiple pods such as the **Jenkins master server** in multiple minions and managed **Kubernetes** charts using **Helm**.
- Extensively worked in developing **APIs** using **Kubernetes** to manage and specify the copies of the containers to run the actual servers in the cloud environment. Scheduled, deployed and managed container replicas onto a **node cluster** using **Kubernetes**.
- Configured applications that run **multi-container Docker** applications by utilizing **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**.
- Worked to set up **Jenkins** as a service inside the **Docker swarm cluster** to reduce the failover downtime to minutes and to automate the **Docker containers** deployment without using configuration management tool.
- Proficiency in writing **Docker files** with best practices along with Creating **Docker Images**, **Docker limiting**, **Docker container** management along with **Docker volumes**, container-based DB and services, **Docker Artifactory (JFrog)** configuration and setup and worked on **Docker Container Snapshots**.
- Developed **Ansible Playbooks** for automating the Infrastructure, deployment process. Managed build and deployment scripts using **YAML** in Ansible, triggered the jobs using **Jenkins** to move from one environment to across all the environments.
- Used **Ansible and Ansible Tower** as Configuration management tool, to automate repetitive tasks, quickly deploys critical applications, and proactively manages change.
- 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**.
- Created **Ansible** cloud modules for interacting with Azure services which provides the tools to easily create and orchestrate infrastructure on **Azure** and automated cloud-native applications in Azure using Azure microservices such as **Azure Functions** and **Kubernetes on Azure**.
- Integrated **Docker** container-based test infrastructure to **Jenkins** Continuous Integration test flow and set up the build environment integrating with **Git** and **Jira** to trigger builds using **Webhooks** and run jobs on Slave Machines.
- Built end to end **CI/CD Pipelines** in **Jenkins** to retrieve code, compile applications, perform tests and push build artifacts to **Nexus Repository** and Deploy to **orchestrate** changes across servers and components.
- Implemented **Nagios and Splunk** tools for monitoring and analyzing the network loads on the individual machines and automated the testing of web applications using **Selenium Web driver**.
- Created dashboards and visualizations using **Splunk, Grafana and Nagios** for performance and activity monitoring and setting up **Splunk** to capture and analyze data from various layers Load Balancers, Webservers and application servers.
- Code utility scripts in **PowerShell** to modify **XML** configuration files dynamically, which is used during the release process and Used **PowerShell scripts** to handle various SharePoint admin jobs like backup, restoration, solution install/deploy.
- Involved in developing custom scripts using **Python, Shell** to automate the deployment process and for Task scheduling, Systems backups for **RHEL**.

- Used **GIT** as a source code management tool for creating local repo, cloning the repo, adding, committing, pushing the changes in the local repo, saving changes for later (**Stash**), recovering files, **branching**, creating tags.
- Performed installation, configuration, upgrades, Package administration and support for **Linux** systems on the client side using **RedHat satellite network server** and Worked on patches installation, patch upgrades on **Red Hat Linux servers** by using **RPM & YUM**.

**Environment:** Microsoft Windows AZURE, Azure AD, Azure SQL, Azure Network, Web Applications, Kubernetes, Virtual Machines, Ansible, Jenkins, Docker, Python, Power shell, Microsoft Azure Storage, SonarQube, Groovy, Maven, Git, Gitlab, ELK, Splunk, Jira, Nexus, Tomcat, GitHub, Linux

## AWS/DevOps Engineer

Client: AmeriHealth, Philadelphia, PA

Mar 2017 to May 18

- Built servers using **AWS** includes importing necessary volumes, launching **EC2** instances, **S3** for object static webpages and created **security groups, auto scaling, load balancer, Route 53** and **SNS** as per architecture using **IAAC** tools like **Cloud formation JSON Template**.
- Using **Cloud Watch**, extensively monitored the **AWS** services like **EC2, CPU** memory, **Amazon RDS DB services, DynamoDB tables, EBS volumes** to set alarms for notification or automated actions and to monitor logs for a better understanding and operation of the system.
- 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 **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.
- Wrote automation scripts for creating resources in **OpenStack** Cloud using **Python** and **terraform** modules. Created and managed **Cinder volumes** in **OpenStack Cloud**.
- Supported **AWS** Cloud environment with **AWS** instances, configured **Elastic IP & Elastic Storage**. Used **AWS CloudFront** to deliver content from **AWS** edge locations to users, allowing for further reduction of load on front-end servers.
- Provisioned the highly available **EC2** instances using **Terraform** and **CloudFormation** templates, wrote new plugins to support new functionality in **Terraform** and involved in using **terraform** migrate legacy and monolithic systems to **Amazon Web Services**.
- Created **Docker images using Dockerfile**, worked on **Docker container snapshots**, removing images and managing **Docker Volumes**. Reduced build and deployment times by designing and implementing **Docker Workflow**.
- Worked on end to end setup of the Artifactory pro as a **Docker** container with a secure private **Docker registry** and local **Docker** repositories for storing the built **Docker images**.
- Configured applications that runs multi-container **Docker** applications by utilizing **Docker-Compose** tool which uses a file configured in **YAML** format. Managed a cluster of **Docker** nodes as a single virtual system using Clustering and Scheduling Tool, **Docker-Swarm**.
- Integrated **Jenkins** with **Docker** container using **CloudBees Docker pipeline plugin** to drive all microservices builds out to the **Docker Registry** and then deployed to **Kubernetes**. Integrated **Selenium** automation regression test suite in **Jenkins** build pipeline.
- 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**.

- Installed and configured **Chef Server, Chef-solo** along with creating **Chef Cookbooks** and implemented the latest releases of **Chef- Solo, Compliance, Habitat** and written **Chef Recipes** to install and configure **Nagios** for monitoring Infrastructure.
- Worked on Setting up **Jenkins** server and build jobs to provide **Continuous Automated** builds based on polling the **Git** source control system during the day and periodic scheduled builds overnight to support development needs using **Jenkins, Git, and Maven**.
- Included Vulnerability Detection as part of the **CI/CD** workflow using **Jenkins Pipelines, SonarQube Quality Gates**. Installed **Jenkins/Plugins** for **GIT** Repository, Setup **SCM** Polling for Immediate Build with Maven and Maven Repository.
- Built and managed a highly available monitoring infrastructure to monitor different application servers and its components using **Nagios** with **Chef** automation. Involved in editing the existing **ANT** files in case of errors or changes in the project requirements.
- Used **Nagios** as a monitoring tool to identify and resolve infrastructure problems before they affect critical processes and worked on **Nagios** Event handlers in case of an automatic restart of failed applications and servers.
- Built and maintained **ELK stack** to centrally collect logs that are used to monitor applications also Installed, monitored and configured Applications in **Nginx and Apache Tomcat Server** and establish connectivity to databases and troubleshoot issues on the fly.
- Developed **PowerShell** scripts to automate web application servers and automation of the build and release process and to automate certain processes for the smooth process of a build tool like **Maven**.
- Created **Disk Partitions, Logical Volume Manager (LVM) & RAID** Technologies on different File Systems like **Ext2, Ext3, Ext4, XFS** on various Linux Distributions & involved in creating Veritas volume manager on Solaris.

**Environment:** AWS, OpenStack, Chef, Terraform, Docker, Elasticsearch, LogStash & Kibana (ELK), Ant, Linux, Unix, Windows, Git, Maven, Jenkins, Python, PowerShell, Nagios, Kubernetes, Apache, Tomcat, Jira.

## DevOps Automation Engineer Travelport, Denver, CO

Mar 2016 to Feb 2017

- Setting up the automation environment for Application team if necessary and help them through the process of build and release automation.
- Used **MAVEN** build tools on Java projects for development of build artifacts on the source code.
- Developed Scripts for **AWS** Orchestration
- Created and maintained **Ant build.XML** and **Maven Pom.XML** for performing the builds.
- Created **branches, performed merges** in version control systems **GIT, GitHub, SVN, Stash**.
- Automated setting up server infrastructure for the DevOps services, using **Ansible**, shell and **Python scripts**.
- Working on Multiple **AWS** instances, set the security groups, Elastic Load Balancer and **AMIs**, Auto scaling to design cost effective, fault tolerant and highly available systems.
- **AWSEC2/VPC/S3/Route53/IAM/Cloud Formation/ELB** based infrastructure automation through **CHEF, Vagrant**, Bash Scripts.
- Worked on Managing the Private **Cloud Environment using CHEF**.

- Developed **CHEF Cookbooks** to install and configure **Apache, Tomcat, Jenkins, and Rundeck** and **deployment automation**.
- Automated the **JUNIT** test cases to test the applications business logic.
- Created **Continuous Integration** and **Continuous Delivery Pipelines** for the build and deployment automation in place.
- Worked with **CHEF** Enterprise Hosted as well as On-Premises, 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.
- Used **AWS** cloud services to launch Linux and windows machines, created security groups and written basic PowerShell scripts to take backups and mount network shared drives.
- Implemented build stage- to build the microservice and push the docker container image to the private **docker registry**.
- Worked on end-to-end setup of the Artifactory pro as a docker container with a secure private docker registry and local docker repositories for storing the built **docker** images.
- Worked on deployment automation of all the microservices to pull image from the private Docker registry and deploy to **Docker swarm** cluster using Ansible.
- Worked on microservices project- to build Docker containers and deploy to Dev, iTest, Scale, UAT (SIT), PROD.
- Hands on experience in migrating Business from a on-premises infrastructure environment to **AWS**.
- Excellent understanding modern cloud (compute, storage, networking) application architectures on **Java Developer** to ensure easy monitoring and high availability.
- Heavy **Shell, YAML, Python, and JSON** scripting, used ansible for configure tomcat servers.
- Working with the hardware vendors and application team for the faulty replacement and planning the downtimes accordingly.
- Working on Deployment procedures using middleware like Apache Tomcat, creating deploy scripts and setting for the Production Release.
- Have used Confluence to create, share and discuss content and projects.
- Maintained CentOS servers for any data overload and update user processors.
- Maintained the deployment properties for the **Elasticsearch, Logstash and Kibana**.
- Worked on the connectivity and Firewall issues for the installation and connectivity of the tools.

**Environment:** JAVA, GIT, Apache Maven, Docker Swarm, Apache Tomcat, Shell Script, Nexus, REST API, Ansible, Linux, Windows, JIRA, Chef, Python, Selenium, AWS, DNS, Docker

## Build and Release Engineer

Client: ICICI financial services, IND

Aug 2015 - Jan 2016

- As member of Release Engineering group, redefined processes and implemented tools for software builds, patch creation, source control, and release tracking and reporting, on UNIX platform.
- Maintained and administered **Git** source code tool.
- Support the code builds by integrating with continuous integration tool **Jenkins** and Managed build results in **Jenkins** and deployed using workflows.
- Integrated **Git** into **Jenkins** to automate the code check-out process.
- Managed to deliver specific versions of various components of an application into target environments.

- Utilized Configuration Management Tool **Chef** & created Chef **Cookbooks** using recipes to automate system operations.
- Involved in writing **pom files** to establish the code quality tools integration.
- Experienced in authoring **pom.xml** files, performing releases with the **Maven release plugin**, **Mavenization** of Java projects and managing **Maven repositories**.
- Maintain and track inventory using **Jenkins** and set alerts when the servers are full and need attention.
- Modeled the structure for multi-tiered applications orchestrate the processes to deploy each tier.
- Developed build and deployment scripts using **Ant** and **Maven** as build tools in Jenkins to move from one environment to other environments.
- Integrated delivery (CI and CD process) Using **Jenkins**, **Nexus**, **Yum** and **Puppet**.
- Created **puppet** manifests, profiles, and roles module to automate system operations.
- Performed and deployed Builds for various Environments like **QA**, **UAT** and Productions Environments
- Familiar and experienced with **Agile** Scrum development.
- Proposed and implemented branching strategy suitable for agile development in **Subversion**.
- Worked with customers, developers, testing, product management and support staff.
- In-depth understanding of large complex software systems to isolate defects, reproduce defects, assess risk and understand varied customer deployments.

**Environment:** Java/J2EE, Spring, Subversion SVN, Jenkins, JIRA, Jenkins (CICD), Maven, GIT, ANT, Maven, Python, Linux, Windows, Nexus, PowerShell, Scripting.

## Linux System Administrator

Key Point Technologies – Hyderabad, India

Dec 2014 – Jul 2015

- Installed and configured **Nagios** monitoring tool to manage infrastructure.
- Developed, customized and build packages on **RHEL** and **RPMS** on **Linux** for deployment on various servers through Software Development Life Cycle.
- Performed automated installations of Operating System using **Jumpstart** for Solaris and **Kickstart** for **Linux**.
- Troubleshooting of day-to-day operational problems, client-side issues, and hardware problems in the systems.
- Configuration and administration of LDAP, NFS, FTP, SAMBA and POSTFIX Servers in Red Hat Enterprise **Linux**.
- Maintaining Volumes manager and File systems using Veritas Storage Foundation 5, SVM on Solaris and LVM on **Linux**.
- Troubleshoot **UNIX/ Linux** networks resolve security related issues by using tools such as IP tables, firewall, TCP wrappers, NMAP
- Worked on installation, configuration, performance tuning, security, backup, recovery and upgrades of **Linux/Unix** environment.
- Directed the implementation and performance tuning of **Windows** 2003 Server environment for client's global operations.
- Assisted user with troubleshooting software and hardware on **windows** desktops and tablets.
- Experience in **Windows** Server 2008 Active directory installation and configuration, Performance tuning and management for a Linux server to resolve issues.

**Environment:** Unix/Linux, RHEL, Windows, Bash/Shell, Nagios, Kickstart.

- Used **.Net framework 4.0** throughout the project and migrated the existing code from **.Net framework 2.0** to **.Net framework 4.0** using **Visual Studio**.
- Developed a website in **ASP.NET, MVC 3** using **jQuery, JavaScript, C#, SQL Server, Visual Studio**, and **.Net 4.0**.
- Developed **ASP.Net MVC 3, Web Services** and classic **Web Forms** administrative web sites.
- Designed and developed web forms and web pages using **HTML, AJAX, ASPX, ASP.NET** and **CSS** to facilitate the users.
- Developed the **3-tier application** using **Visual Studio** and **.Net framework 4.0**.
- Developed several functions and routines in the presentation layer using **JavaScript** and **C#** to enhance customer experience.
- Developed web services to fetch look-up data by using **jQuery** and **JSON functionality**.
- Created tables, stored procedures and views in **SQL Server** database for backend development.
- Created **WCF Services** responsible for communicating and providing real time data from integrated server to Client Application.
- Utilized **LINQ to Entities** technology in Data Access Layer for data retrieving, querying, storage and manipulation.
- Used Team Foundation Server (**TFS**) for the purpose of version control and sharing within the application.

**Environment:** .NET Framework 4.0, HTML, CSS, JavaScript, jQuery, AJAX, Visual Studio 2012, ASP.NET MVC 3, C#, Entity Framework, LINQ to Entities, JSON, SQL Server 2012, Oracle, IIS, SOAP, Web Services and WCF.