**[A picture containing text, sign, blue

Description automatically generated](https://www.credly.com/badges/c29701b4-32dc-457b-9faa-a096171b277c/public_url)Rahul Sainath Nilagiri**

Sr. Cloud/ DevOps Engineer

Email:rahulnilagiri20@gmail.com

|  |
| --- |
| **PROFESSIONAL SUMMARY** |

Contact : +1 510-402-3577   
LinkedIn: <http://www.linkedin.com/in/rahul-sainath-nilagiri>

Over **8 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 on various **AZURE** services like **Compute (Web Roles, Worker Roles)**, **Caching**, **SQL AZURE**, **NoSQL**, **USQL**, **Network** services, **AZURE Active Directory**, **AZURE Websites**, **Application Insights**, **Power BI**, Internet of Things (**IoT**), Azure **Monitoring**, **Key Vault**, Visual Studio Online (**VSO**), **Storage**, **Data Factory, API Management**, **Scheduling**, Auto **Scaling**, **PowerShell** Automation and other Azure Services.
* Experience in provisioning of **IaaS**, **PaaS** Virtual Machines and Web, Worker roles on Microsoft AZURE **Classic** and Azure Resource Manager (**ARM**), Deployed Web applicatios on Azure using **PowerShell** Workflow.
* Experience in using Azure **AD** for integrating **SaaS** applications which provides identity and access management and enables users to **single sign-on** to applications directly or discover and launch them from a portal such as **Office 365** or the Azure AD **access panel**.
* Experience in deploying a **Dynamics NAV** database that is exported as a data-tier application (DAC) **.bacpac** file into Azure SQL using **SQL Server Manager**.
* Experience in **Architecting** and securing the **Infrastructure** on **AWS** using **EC2, S3, RDS, EBS, VPC, ELB, IAM,** **KMS**, **EMR**, **Cognito**, **API Gateway**, **CloudTrail**, **Cloud Watch**, **Amazon SQS**, **SNS, SES,**  **Config**, **AWS** **Kinesis**, **Lambda**, **Network ACLs**, **Security** **Groups**, **Route** **Tables**, **Route53**, **Elastic Beanstalk**, **Redshift** and deploying applications utilizing all these services majorly focusing on **high-availability**, **fault tolerance**, and **Autoscaling** in **AWS CloudFormation**.
* 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 in using **CloudFormation template** for reliable provisioning of initial infrastructure (**stacks**) like **IAM** **roles** & **policies**, **S3 buckets** and **bucket** backups on **AWS**.
* Experience in managing **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 in using **OpenShift** for container orchestration with **Kubernetes** , container storage, automation, to enhance container platform multi-tenancy. Experience with **OpenShift, Kubernetes** architecture and design, troubleshooting issues and multi-regional deployment models and patterns for large-scale applications.
* Proficient with container systems like **Docker** and container orchestration like **EC2 Container Service**, **Kubernetes**, worked with **Terraform.** Used **Kubernetes** to orchestrate the deployment, scaling and management of Docker Containers.
* Written Templates for AWS infrastructure as a code using **Terraform** to build staging and production environments.
* As part of Cloud team, expertise in installing, configuring and troubleshooting **OpenStack** environment (**Nova**, **Swift**, **Cinder**, **Neutron**, **Horizon**, **Glance**, **Keystone** and **Ceilometer**) on RedHat and Ubuntu distribution in multi node cluster with **Ceph** storage.
* 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
* Good working experience on several prototype projects involving clustered container orchestration and management. Contributed a **MySQL** cluster application to the **Kubernetes** project.
* Experience in Building and service. 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.
* 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.
* 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 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**.
* 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.
* 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 **JIRA** for ticketing and as well as defect tracking system and configure various workflows, customizations and plug-ins for **JIRA** bug/issue tracker, integrated **Jenkins** with **JIRA**, **GitHub**, **Maven.**
* Proficient in **Ruby**, **Perl**, **Power Shell**, **JSON**, **YAML**, **Groovy**, **Bash/Shell**, **Python**. Developed python, and shell scripts for automation of the build and release process.
* Experience in working with **SQL** databases like **MySQL**, **PostgreSQL, Oracle** 10i/9i/8i, **SQL Server** and **NoSQL** database like **MongoDB**, **DynamoDB**, and **Cassandra**.
* Expertise in Installation, Support, Configuration with different flavors of **LINUX RHEL 7/6/5/4**, **CentOS 7/6/5**, **Oracle Linux (OEL) 6.x/5.x**, **Debian**, and **Ubuntu 16/14/13**. Installed and configured **Red Hat LINUX Kickstart** and **Solaris jumpstart Servers**.

**TECHNICAL SKILLS**

|  |  |
| --- | --- |
| **Database** | MySQL, MS SQL, PostgreSQL, Oracle, Dynamo DB |
| **Scripting/Languages** | Bash/Shell, Ruby, Python, Perl, JavaScript, PowerShell, JSON, Groovy, YAML, C, C++ |
| **CI & Build Tools** | Bamboo, Hudson, Jenkins, Maven, Ant, Gradle. |
| **Configuration Management & Deployment Tools** | Puppet, Chef, Ansible, Terraform |
| **Version Control Tools** | GIT, GITHUB, SVN, Perforce, Bitbucket. |
| **Virtualization** | VMware ESX/ESXi, Windows Hyper-V, Power VM, Vagrant, Virtual box. |
| **Container/Orchestration Tools** | Kubernetes, Docker, OpenShift |
| **Cloud** | Amazon Web Services, Microsoft Azure (IaaS, PaaS & SaaS), PCF, OpenStack. |
| **Monitoring & Ticketing Tools** | Nagios, Splunk, Dynatrace, ELK, Datadog, Selenium, SonarQube, Jira, ServiceNow. |
| **Web/Application Servers** | JBOSS 4.x/5.x, Apache Tomcat 5.x/7.x, Web Sphere (4.0, 5.0), Web Logic (8/9/10). |
| **Operating System** | Red Hat Linux 7/6/5/4, Ubuntu 16/14/13/12, Debian, Fedora, CentOS, Windows 10/8/7/Vista/XP, Solaris 11/10/9/8, Mac OS. |
| **Networking/protocol** | TCP/IP, DNS, NFS, ICMP, SMTP, DHCP, OSPF, BGP, UDP and RIP. |

**TECHNICAL CERTIFICATIONS**

* Certified AZ104(**Microsoft Azure Administrator)**

**PROFESSIONAL EXPERIENCE**

|  |
| --- |
| **Sr. Cloud/DevOps Engineer Sep 2019 –Present**  **Cipla Pharmaceuticals, NY** |

* Expertise in dealing with windows (**IaaS**) Virtual Networks, **Virtual Machines**, **Cloud Services**, **Resource Groups**, **Express Route**, **Traffic Manager**, **VPN**, **Load Balancing**, **Application Gateways**, **Auto-Scaling**.
* Designed AZURE Resource Manager (**ARM**) templates and custom build steps using **PowerShell**. Identified and logged defects in **VSTS** and interacted with developers to prioritize the issues.
* Created **Azure SQL** database, performed **monitoring** and **restoring** of Azure SQL database. Performed migration of **On-premises** Microsoft SQL server to Azure SQL database.
* Involved in managing **Private Cloud** and **Hybrid cloud** configurations and practices in Windows **Azure**, **SQL Azure**, **Azure Web** and **Database** deployments. **Upgraded** and **Migrated** web applications to latest **.Net framework** versions and **Azure platforms**.
* Deployed **Azure** 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**.
* Handled migration of on-premises applications to the cloud and created resources inside cloud and expert in deploying of **OpenStack** hosted applications on Azure platform and issues related to **RDP** of **Azure Infrastructure**.
* Involved in **OpenStack** environments to create and manage users using **Keystone** and **VM management** using **OpenStack**, volume management using **Cinder**, network and port management using **Neutron**. Used **Helm** packet manager for installing Jenkins, artifactory tools etc., by running **Helm charts.**
* Worked with **Terraform** Templates to automate the **Azure 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**.
* Designed infrastructure for **AWS** application and workflow using **Terraform** and had done implementation and continuous delivery of AWS infrastructure using **Terraform**.
* Developed strategy to migrate Dev/Test/Production from an enterprise VMware infrastructure to the IaaS Amazon Web Services (**AWS**) Cloud environment.
* Worked on **IAM** role for conveying the measurements to **CloudWatch** dashboard and Created notifications and alarms for **EC2** instances using **CloudWatch** .
* Automated the provisioning the AWS instance creation using Ansible playbooks and provided them with the security groups, type, Subnet, VPC and created the public DNS.
* 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 **Azure Container Registry**.
* 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.
* 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**.
* 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.
* Worked on Upgrading **Cassandra** to a new version and Implemented **Ansible playbook** for automated Orchestration of Cassandra **clusters**.
* Wrote several **Ansible playbooks** for the automation that was defined through tasks using **YAML** format and run Ansible Scripts to provision Dev servers. Customized Ansible role for automated deployments of the **Dynatrace** Java Agent for **Glassfish**.
* Used **Jenkins** as Continuous Integration tools to deploy the Spring Boot **Microservices** to **Azure 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**.
* 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**.
* Developed microservice onboarding tools leveraging **Python** and **Jenkins,** allowing for easy creation and maintenance of **build jobs**, **Kubernetes** deploy and services.
* Installed **Jenkins Plugins** for **GIT** Repository, Setup **SCM** Polling for Immediate Build with **Maven** and **Maven Repository** (**Nexus** Artifactory).
* Pulled code from development line using cherry-pick, creating a tag and pushing it to main line using GIT version control after integration and build.
* Integrated **Jira** with **Dynatrace** to generate Incidents from **Dynatrace** and deployed new Dynatrace architecture at disaster recovery site & automated provisioning repetitive tasks using **Terraform** and **Python.**
* Involved in **Jira** as defect tracking system and configure various workflows, customizations and plug-ins for **Jira** bug/issue tracker.
* Deployed Splunk **forwarders**, **indexers** and **search heads** to monitor, analyze and visualize the azure VM's on the Splunk dashboard that helps in increasing the cluster performance.
* Worked on **Power Shell** scripts to automate the **Azure Cloud system** in creation of **Resource groups**, **Web Applications**, **Azure Storage Blobs** & **Tables**, **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 **Azure**, includes the design and selection of relevant hardware to support the installation/upgrades of **Ubuntu, CentOS**.

***Environment:*** *Azure (Web Roles, Worker Roles, SQL Azure, Azure Storage, Azure AD, Resource Groups, Office365, RBAC), OpenStack (Kilo/Liberty), Chef, Ansible, Docker, Kubernetes, Maven, Jenkins, GIT, Cassandra, AEM, Python, Jira, Dynatrace, Terraform, (AWS) Cloud environment,**CloudWatch*

|  |
| --- |
| **Sr.Cloud/ DevOps Engineer April 2018 – Sep 2019**  **Anthem, Virginia Beach** |

* 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**.
* 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.
* 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**.
* Integrated AWS **Dynamo DB** using AWS lambda to store the values of items and backup the **Dynamo DB streams**.
* 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.
* 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.
* 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**.
* Involved in environments to create and manage users using **Keystone** and **VM management** using **OpenStack**, volume management using **Cinder**, network and port management using **Neutron**.
* Worked on writing **Ansible** playbooks to automate the given process and **Ansible Tower** as Configuration management tool, to automate repetitive tasks, quickly deploys critical applications.
* Developed **Ansible** scripts in YAML for an automated server provisioning and **Docker** images, reducing the time between provisioning and deployment.
* Created **playbooks** for **OpenStack** deployments and bug fixes with **Ansible**. Deployed and tested playbooks in **Docker** containers before pushing changes in **Git**.
* Integrated **Ansible** with **Jenkins** to provide automation, **continuous integration** through **Jenkins** and Implemented **Jenkins Workflow** and Plugins for repeatable deployments of **multi-tier** applications, **artifacts** and services to **Docker.**
* 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**.
* Used **Kubernetes** to manage containerized applications using its nodes, ConfigMaps, selector, Services and deployed application containers as Pods.
* 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**.
* Migrated 9 microservices to **Google Cloud Platform** from skava and have one more big release planned with 4 more microservices.
* Worked on google cloud platform (**GCP**) services like **compute** **engine**, **cloud** **load** **balancing**, cloud storage, cloud SQL, stack driver monitoring and cloud deployment manager.
* Setup **GCP** Firewall rules to allow or deny traffic to and from the VM's 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.
* 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**.
* 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.
* Created **Chef Recipes** to provision **Apache Web servers**, **Tomcat servers**, **Nginx**, **Apache Spark** and other applications.
* Automated **Datadog** Dashboards with the stack through **Terraform** Scripts. Configured **CloudWatch** and **Datadog** to monitor real-time **granular metrics** of all the **AWS** Services and configured individual dashboards for each resource Agents.
* 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, Ansible, Docker, Kubernetes, OpenStack (kilo/ Juno), Jenkins, GIT, Python, Datadog, Jira,**compute engine, cloud load balancing, cloud storage, cloud SQL,*

|  |
| --- |
| **DevOps Engineer Feb 2017 –April 2018**  **Dxc Technology, Arkansas** |

* Written AWS CloudFormation Templates for different services like **CloudFront** Distribution, **API** Gateway, **Route 53**, **Elastic Cache**, **VPC**, **Subnet Groups**, **Security Groups**.
* 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.
* 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.
* Worked with **Knife** command line tool for creating **Recipes**, **Cookbooks**, **bootstrapping** nodes and worked with **chef supermarket**.
* 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**.
* Performed configuration of standard Linux and network protocols, such as **SMTP**, **DHCP**, **DNS**, **LDAP**, **NFS**, **SMTP**, **HTTP**, **SNMP**.

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

|  |
| --- |
| **Build and Release Engineer Dec 2015 – Jan 2017**  **Sacrosacinfo, Fremont** |

* Wrote scripts using **ANT** and **MAVEN** tools and automated the build and deploy process using **Jenkins** to move from one environment to other environments. Also edited the existing **ANT/MAVEN** files in case of **errors**.
* Implemented **Jenkins** **pipeline** stage which synchronize releases stored in **Artifactory** with **Puppet** yum repo.
* Responsible for automated CI, Scheduled Builds/Emergency Builds and Release using **ANT** scripts for an Enterprise application (**J2EE**) using **Jenkins**.
* Setting up continuous integration and formal builds using Bamboo with Artifactory repository.
* Wrote **Puppet** models for installing and managing **java versions** and took the sole responsibility to maintain the **CI Bamboo** server.
* Involved in creating **Bamboo** pipeline jobs for **Puppet** release process for module deployment, using the Agile methodology for **puppet** development.
* Installed and administered repository to deploy the artifacts generated by **ANT** and to store the dependent **jars**, which were used during the build.
* Manage and optimize the **Continuous Integration** using **Jenkins**, **Bamboo** and troubleshoot the deployment build issues using the **trigged logs**.
* Developed automation scripting in **Python** (core) using **Puppet** to deploy and manage Java applications across Linux servers.
* Used **ANT** as a build tool in java projects for the development of build artifacts on the source code. Involved in editing the existing **ANT** files in case of errors or changes in the project requirements.
* Managed user authentication and authorization for the users in both **Subversion** & **Perforce**. Created branches, performed merges in version control systems **SVN** and **Perforce**.
* Worked with different tools for monitoring the health checkups using **Nagios**, Searching and Reporting with **Splunk**.
* Installed, upgraded and configured Linux Servers using **Kickstart** as well as manual installations and the recovery of a **root password**.
* Integrated ant with **Shell** scripts created in **Bash** to automate the deployments for the Java-based applications.

***Environment:*** *RedHat Enterprise Linux, Bamboo, Subversion, Perforce, Nagios, ANT, Python, Puppet, CentOS, Ubuntu, Kickstart, VMware, TCP/IP, NIS, NFS, DNS, SNMP, VSFTP and DHCP.*

|  |
| --- |
| **Linux Administrator Jun 2013 – Mar 2015**  **Icici bank, Hyderabad, India** |

* Installed, Configurated and Managed **RedHat Enterprise Linux 5.x**, Windows Server **2003 Active Directory**, Servers like **WebSphere** and Web servers like Web Logic, **Apache**, **HTTP** and **Tomcat.**
* Wrote UNIX shell scripts for automated installations, to extract logs using **Bash**, **Perl**, and **Python**.
* Managed users, groups and permissions, configured **LDAP** and given **SUDO** Access in test and development servers and allowed password less logins for authorized users on servers.
* Setting up user and group login ID's, printing parameters, network configuration, password, resolving permissions and access related issues via **ACL**.
* Developed Linux, UNIX, **Perl** and **Shell Scripts** and AWT programming for manual deployment of the code to various environments.
* Worked on setting up **NIC bonding** on Linux Enterprise servers for high availability, redundancy and Increased bandwidth.
* Installed, configured, performance tuning, security, backup, recovery and upgrade/patches of Linux and UNIX servers using **Crontab**.
* Installing Red Hat Linux using **kick-start** and applying security policies for hardening the server based on the company policies.
* Administered and supported application servers, responsibilities included troubleshooting and monitoring servers using **Nagios**.
* Assisted project management team with design issues, troubleshoot technical issues, rotating on-call pager, Install **OS patches**, **NFS** mounts and **Automount**.
* Setting up network environments using **TCP/IP**, **NIS**, **NFS**, **DNS**, **SNMP**, **VSFTP** and **DHCP**.
* Worked in setting up **LDAP**, **DNS**, **DHCP** Server along with effective group and System Level policies and roaming profile features by using **Samba** and **NFS** servers.

***Environment:*** *RedHat Enterprise Linux 5.x, YUM, User and Group management, Bash, Firewall, Apache, Volume Groups, LVM, NFS, FTP, SAMBA, SSH, DHCP, DNS, VERITAS Cluster Server (VCS), Windows Server 2003 Active Directory, Kickstart server, TCP/IP, IP addressing & Subnetting, routing.*