**Pranay chowdary**

**Email:** [**pranaycdevops@gmail.com**](mailto:pranaycdevops@gmail.com)

**PH: 650-550-0798**

**AWS DevOps Engineer**

**PROFESSIONAL SUMMARY**

* Over 10**+ years** of experience in IT comprising of Designing, Development, Integration of **DevOps** tools, Build Automation, Configuration Management, Build/Release and Delivery Management, CI/CD, involving cloud infrastructure services like (**Amazon Web Services**) as well as Cloud Migration services using Cloud formation and Terraform templates.
* Extensive experience on **AWS**services like **ELB**, **EC2**, **S3**, **VPC**, **NACL's**, Cloud Formation, Route 53, Network Connectivity troubleshooting, Hybrid environments and **VPC peering.**
* Experienced in both framework and **CloudFormation**to automate **AWS**environment creation along with the ability to deployment on **AWS**, using build scripts (**Boto3**and **AWSCLI**) and automate solutions using **Shell**and **Python**.
* Experience on **AWS**, focusing on high - availability, fault tolerance, and auto-scaling using **Terraform**templates. Along with (CI/CD) with **AWS**Lambda and **AWS**Code Pipeline
* Experience in Infrastructure on **AWS**using **IAM**, **API**Gateway, CloudTrail, Cloud Watch, Amazon Simple Queue Service (Amazon SQS), **AWS**Kinesis, Lambda, NACL, Elastic Beanstalk, Redshift, and CloudFormation
* Hands on experience with **Terraform API** modules to manage infrastructure and publishing module to the **Terraform** registry and contributed for end-to-end configuration management to deploy our products inside **AWS** environment.
* Hands-On experience with Splunk Search queries and created jobs/alerts and dashboards for monitoring.
* Expertise in using **Ansible**to setup Continuous Delivery Pipeline. Deployed micro services, including provisioning **AWS**environments using **Ansible**Playbooks. Expertise in **Ansible**Playbook, **YAML**for Maintaining Roles Inventory Files and Groups Variables and Extensively worked on **Jenkins.**
* Design and implement scalable enterprise monitoring systems by applying continuous integration/delivery concepts • Perform maintenance and troubleshooting of our enterprise RedHat **OpenShift** systems • Work to continuously improve speed, efficiency and scalability of **OpenShift** systems
* Designed, developed, and maintained **Groovy** scripts for **integration** with existing systems and applications.
* Created custom plugins using **Groovy** for enhancing the functionality of the company's internal tools.
* Integrated **Spring Boot applications** with relational databases
* Extensively worked with configuring and management tools like **Chef,** **Puppet**, **Ansible**and expertise in developing Recipes, Manifests and Playbooks in Ruby and YAML scripts to automate the environment.
* Experience in implementation of **Puppet**agent based and agentless configurations, using **Puppet**dashboards, **Collective** with active **MQ** server, plugin-sync, stored configs, and wrote **Puppet** manifests.
* Leveraged **Spring Boot's** Dependency Injection mechanism for managing component **dependencies**
* Worked on migration of **Splunk**to AWS (cloud) instances. Involved in standardizing **Splunk**forwarder deployment, configuration and maintenance across UNIX and Windows platforms
* Kept up-to-date with the latest **Spring Boot** and **Java** advancements
* Expertise in App Containerization technology **Docker,** creating **Docker**images, Containers, **Docker**Registry to store images, cloud-based registry **Docker**Hub, **Docker**Swarm to manage containers
* Experience in deployment automation of all the **Microservices** to pull the image from the private **Docker** Registry and deployed into **Docker** Swarm to create a complete docker cluster by using **Ansible**
* Experience in managing **Kubernetes** charts using **Helm**. Created reproducible builds of the **Kubernetes** applications, managed manifest files & releases of **Helm** packages. Created Pods in **Kubernetes**and monitored the health of pods using **Helm**charts.
* Experience in code deployment, Orchestration and Scheduling using tools such as **Kubernetes**, **Docker Swarm**, , CloudFormation, and automation validation using Test Kitchen, Vagrant, Ansible and **Terraform**.
* Experience in **Kubernetes** to deploy scale, load balance and manage **Docker** containers with multiple name spaced versions and good understanding of **Open Shift Platform** in managing Docker Containers and **Kubernetes** Clusters.
* Experience in integrating **Jenkins** with **Docker** container using Cloud bees **Docker, Kubernetes**pipeline plugin and provisioned the **EC2** instance using **Amazon EC2 plugin**.
* Experience in implementing CI/CD for J2EE, SOA & Micro services Architecture Environment (NodeJS, .Net Core, Zookeeper, Kafka) using **Jenkins**, Anthill Pro. Used **Jenkins** with SonarQube to analyse and get the better quality of the Source code.
* Hands-on experience in integrating **Jenkins** CI tool with build automation tools like **ANT**, **Maven**, **Gradle** for building of deployable artifacts (JAR, WAR & EAR) from source code.
* Hands-on experience in version control tools like Subversion, **TFS,** **GUI**, **BIT Bucket**, **CVS**, Perforce Version wherein experienced with tagging, branching on platforms like Linux/Windows and used Source code management client tools like, **GIT Bash**, **GitHub**, **GIT**- **GUI**and other **CLI**applications etc.
* Worked with **Splunk**professional services to make the best practices that can be followed by everyone to maintain the performance of **Splunk.**
* Proficient with **Shell,** **Bash**, **Python**, **PowerShell**, **YAML**, **Ruby Scripting**languages for automation, monitoring.
* Worked with RedHat **OpenShift** Container Platform for Docker and Kubernetes, used Kubernetes to manage containerized applications using its nodes, Config Maps, node-selector, Services and deployed application containers as Pods.
* Expertise in **LINUX**/ **UNIX**system builds, administration, Installations, Upgrades, Troubleshooting on different distributions such as **Ubuntu,** **CentOS**, **RHEL**5.x/6.X/7. X. Expertise in creating Exception Group policies for Windows Server **2003**/ **2008**/ **2012R2**in AD domain.

**TECHNICAL SKILLS:**

* **Operating Systems:** Linux (RedHat 4/5/6/7, CENTOS & SUSE), RedHat Linux 4/5/6/7, Windows servers 2003, 2008, 2008 R2, 2012, 2012R2, Windows 2000, XP, Windows 7, Ubuntu 12/13/14, Solaris 11/10/9/8.
* **Cloud platforms:**Amazon Web Services, Azure.
* **Container Orchestration Tools:**Docker Swarm, Kubernetes, EC2 Container Services, OpenShift
* **Virtualization Platforms:**Oracle VM Virtual Box, VMware, Vagrant, Windows Hyper-V
* **Application/Web Servers:**Amazon AWS, Apache, Tomcat, JBoss, WebSphere, VMware, Web Logic, Nginx, Send Mail
* **Scripting languages:**Perl, Python, Ruby, Bash/Shell Scripting, PowerShell, YAML, PHP, JSON, Terraform.
* **Languages:**C, C++, Core Java, J2EE
* **Build Tools:**ANT, Maven, Gradle.
* **Configuration Management tool:**Ansible, chef, puppet, salt stack,
* **Version Control Tools:**GIT, GITHUB, SVN, Perforce, Bitbucket
* **Continuous Integration tools:**Bamboo, Hudson, Jenkins
* **Logging & Monitoring Tools:**Nagios, Splunk, CloudWatch, Dynatrace, ELK, Datadog, Grafana, New Relic
* **Databases:**MS SQL Server, Oracle 8/10.X, Mongo DB, DynamoDB, PostgreSQL, Cassandra
* **Repository Management:**Nexus, Jfrog, Artifactory
* **Networking:**TCP/IP, DHCP, DNS, SNMP, SMTP, Ethernet, NFS, LDAP
* **JIRA, Remedy, Clear Quest:** Track

**PROFESSIONAL EXPERIENCE**

**Client: CapitalOne, Dallas, TX. Nov’21 – Till Date**

**Role: Lead AWS DevOps Engineer**

**Responsibilities:**

* Worked in container-based technologies like **Docker**, **Kubernetes** and **OpenShift**. Point team player on **OpenShift** for creating new Projects, Services for load balancing and adding them to Routes to be accessible from outside, troubleshooting pods through ssh and logs, modification of Build configs, templates, Image streams, etc.
* Managing the **OpenShift** cluster that includes scaling up and down the AWS app nodes.
* Had very strong exposure using ansible automation in replacing the different components of **OpenShift** like **ECTD, MASTER, APP, INFRA, Gluster.**
* Used **Splunk** for configuration and installation process.
* Working on implementing new OCR solution; Spring Boot, **OpenShift**, microservices. Member of group developing containerized applications; Docker, Spring Boot, Kubernetes, **OpenShift**. Deployed Microservices to IBM Bluemix Cloud Foundry and later migrated to **OpenShift**.
* The deployment model uses Atlassian development repository tools, **Jenkins** as the build engine, while execution deployments to container orchestration tools ranged over time from **OpenShift** on EC2, **AWS**.
* Enterprise Container Services, and today using **AWS** Faregate. Implemented **Micro Services** framework with **Spring Boot, NODE.JS** and **OpenShift** containerization platform (OCP).
* Performed configuration, deployment and support of cloud services including Amazon Web Services (**AWS**) Creating Amazon **EC2** Cloud Instances using Amazon Web Services and configuration of launched instances with respect to specific applications.
* Implemented robust authentication and authorization mechanisms using **Spring Security**, safeguarding applications from unauthorized access.
* Employed **Spring Boot Actuator** for real-time application monitoring and management, enabling proactive issue detection and resolution.
* Defined **AWS** Security Groups which acted as virtual firewalls that controlled the traffic allowed to reach one or more **AWS** EC2 instances.
* Worked in installing and upgrading Splunk apps and configured them. Guided all the SME's in using Splunk to create dashboards, reports, Alerts etc.
* Developed an API for managing monitoring and alerting hardware resources in Linux OS environment using **C#, C++, mono framework** and integration with various Linux modules and protocols.
* Executed the automation from commit to deployment directly into **Azure**VM by implementing a CI/CD pipeline with the help of **Jenkins**and **Chef.** Setting up **Chef**Infra, bootstrapping nodes, creating and uploading recipes, node convergence in **Chef**SCM.
* Responsible for Designing, Building, installation, configuration, tuning, security, backup, recovery and upgrades of **RedHat & IBM AIX.**
* Focused on containerization and immutable infrastructure. **Docker**has been a core to this experience, along with **Marathon**and **Kubernetes**from the logs and database in the required format.
* Scheduled, deployed and managed container replicas onto a node cluster using **Kubernetes**and converted VM based application to microservices and deployed as a container managed by **Kubernetes**
* Managed Splunk configuration files like **inputs, props, transforms,**and lookups. Deploy, configure and maintained Splunk forwarder in different platforms.
* Created additional **Docker**Slave Nodes for **Jenkins**using custom **Docker**Images and pulled them to **ECR.** Worked on all major components of **Docker**like, **Docker**Daemon, Hub, Images, Registry, Swarm.
* Utilized **Jenkins**master/slave architecture to distribute builds on nodes and trigger **Jenkins**job to build the artifacts using **Maven** deployed the **Terraform**templates to create the stack.
* Worked on developing **APIs** using **Kubernetes** to manage and specify the copies of the containers to run the actual servers in the cloud environment.
* Managed **Kubernetes**charts using Helm. Created reproducible builds of the **Kubernetes**applications, templatize Kubernetes manifests, provide a set of configuration parameters to customize the deployment and Managed releases of **Helm packages**
* Implemented a production ready, load balanced, highly available, fault tolerant, auto scaling **Kubernetes** cloud infrastructure and microservice container **orchestration**.
* Created Clusters using **Kubernetes** and worked on creating many **pods**, **replication** controllers, replica sets, services, deployments, labels, health checks and ingress by writing **YAML** files.
* **Spinnaker** is introduced for continuous delivery which bakes the image and performs prod deployment. Setting the image which is stable which performs release management.
* Working knowledge of **Docker Hub**, **Docker Container network**, creating Image files primarily for middleware installations & domain configurations. Evaluated **Kubernetes** for Docker Container **Orchestration**.
* Involved in creation of Architecture and built Minimum Viable product (MVP) for containerization platform using Docker and Kubernetes.
* Involved in development of test environment on Docker containers and configuring the Docker containers using Kubernetes.
* Scheduled, deployed and managed container replicas onto a node cluster using Kubernetes.
* Worked on **Docked**-**Compose**, **Docker**-Machine to create **Docker containers**for Testing applications in the QA environment and automated the deployments, **scaling**and management of containerized applications across clusters of hosts.
* Configured and integrated **GIT**into the **continuous integration**(CI) environment along with **Jenkins**and written scripts to containerize using **Ansible**with **Docker**and orchestrate it using **Kubernetes.**
* Experience in managing **Ansible** Playbooks with **Ansible** roles and created inventory in **Ansible** for automating the continuous deployment.
* Maintained Artifacts in binary repositories using **JFrog**Artifactory and pushed new Artifacts by configuring the **Jenkins**project using **Jenkins**Artifactory Plugin.
* Created restful APIs utilizing jersey spring, groovy, hibernate, and spoke unit testing framework to expose data contained in a repository. Developed agile processes using **Groovy**, **JUnit and Cobertura** to use continuous integration tools.
* Developed **Groovy** script to integrate with **SCM** and **Chef**. Read the Docs, Docker, Python, Groovy, Power shell, Unreal Engine Build System). Automated release management Wrote custom groovy scripts to automate CI/CD pipelines in Jenkins.
* Performed OS installations, OS Upgrades, & Server Patching by configuring **PXE**& **DHCP**server using **Kickstart**& **Jumpstart**scripts on **RedHat Linux**5.x, 6.x & 7.x.

**Environment**: Amazon Web Services, Chef 13.x/12.x, Ansible 2.4, Docker 1.13.1/1.12.6, Kubernetes 1.8/1.7,Openshift, Maven 3.5.x, Jenkins 2.10x, GIT-2.1x/2.x, Cassandra 2.x/3.0, AEM 6.2/6.0, Python 3.6, Ruby, Shell, Jira 6.4, Dynatrace, TFS, Splunk, Oracle10.x, WebLogic Server 11g VMware, Nexus, Java, .Net, LDAP, JDK1.7, GitHub, Linux.

**Client: USCellular, Chicago, IL. April’19 – Oct’21**

**Role: AWS DevOps Engineer**

**Responsibilities:**

* Provision **AWS** resources using management console as well as Command Line Interface. (**CLI**)
* Plan, build and configure network infrastructure within the **VPC** with public and private subnets and configure routing tables and internet gateway.
* Responsible for launching **EC2** instances w/ Linux AMI and bootstrapping with Apache using Bash scripting, auto-scaling and Load balancers (**ELB**).
* Define Security Groups depending on access parameters provided.
* Create **IAM** user accounts and role-based policies for access to **AWS** services.
* Perform automated snapshots using **CloudWatch** rules and create **EBS** volumes from snapshots.
* Implement and maintain monitors, alarms, and notifications for **EC2** instances using **CloudWatch** and **SNS**.
* Created and managed **AWS** Storage services like **S3**, **EBS** and **Amazon** **CloudFront**.
* Created an Amazon **RDS** and Handled software installation, backups, and patches as well as routine administrative tasks.
* Implemented popular networking services like **Amazon Virtual private Cloud (VPC)** for the private cloud in the public cloud.
* Worked with **Amazon Web Services** (EC2, Elastic search, Route53, Elastic Beanstalk, VPC, Iaas).
* Configured and implemented the Amazon **EC2** instances for our application teams.
* Troubleshooting issues with the application developers to the **EC2** instances.
* Configured the SSL certificates for **EC2**, database and Amazon cli servers.
* Created User accounts for the Development group, DB-Group, Testing group by using **AWS IAM service**.
* Monitoring the AWS Infrastructure Response Time, Application/Service availability, Backend Transaction time, Throughput using **CloudWatch** and Site24x7.
* Configured Federation services for the Applications Single-SignOn in AWS cloud using AWS IAM service
* Worked in an agile development team to deliver an end to end continuous integration/continuous delivery product in an open source environment using tools like **Chef& Jenkins**.
* Developed and implemented Software Configuration Management strategies for various applications according to the agile process using Git/**GitHub**.

**Environment:** VPC, subnets, EC2, AMI, Internet gateways, CloudWatch, SNS, S3, CloudFront, CloudFormation, Apache Web Server, Linux, Ansible, Jenkins, Docker, Nexus, Dynamo DB, Elastic cache and Lambda

**Client: WELLSFARGO, Charlotte, NC. March’17–March’19**

**Role: DevOps/Cloud Automation Engineer**

**Responsibilities:**

* Build servers using **AWS,** importing volumes, launching **EC2**, **RDS**, creating security groups, auto-scaling, Elastic load balancers ( **ELBs**) in the defined virtual private cloud ( **VPC**).
* Created **Amazon Virtual Private Cloud (VPC)** in the scalable environment which provides advanced security features such as security groups and network access control lists to enable inbound and outbound filtering at the instance level and subnet level.
* Created **S3**buckets and managed policies for **S3**buckets and Utilized **S3**bucket and Glacier for Archival storage and backup on **AWS**
* Created **AWS**Multi-Factor Authentication (MFA) for instance **RDP**/ **SSH**logon, worked with teams to lock down security groups
* Design, installation, configuration and administration of Linux 5,6 and 7 servers and support of OpenShift Enterprise and non-OpenShift support.
* Support Design and implementation of the continuous delivery pipeline, using continuous integration, continuous delivery and automated environment provisioning
* Experience installing and configuring SSH (Secure Shell) encryption in order to access securely on Ubuntu and RedHat Linux
* Responsible for development and version control of **Chef**Cookbooks, testing of Cookbooks using Test Kitchen and running recipes on nodes managed by on premise **Chef**Server.
* Maintained multiple cookbooks in **Chef**by converting production support scripts to **Chef**recipes to automate the configurations in servers.
* Managed servers on **AWS**cloud platform using **Puppet,** **Ansible**as configuration management tools. Writing **JSON**templates for cloud formation and ruby scripts for **puppet**automation and contributing to our repository in **Bitbucket.**
* Improved the virtualization for the servers using the **Docker**for the test environments and Worked in **DevOps**group running **Jenkins**in a **Docker**container with **EC2**slaves in Amazon **AWS**cloud configuration.
* Configured the **Weave Net**such that it creates the Virtual networks which connects **Docker**Containers to multiple hosts. Created a **Swarm**using **Docker**CLI to deploy the services to a **Swarm**and managed the **Swarm**behaviour
* Designed and developed **Groovy** scripts for automating **build** and **deployment** processes, resulting in a 40% reduction in release cycle time.
* Implemented custom **Groovy** scripts to optimize data processing pipelines, improving data accuracy and reducing processing time by 30%.
* Implemented the setup for master slave infrastructure to enhance the performance of **Jenkins**where the projects are the part of release process and were written in shell scripts. Monitored the deployment solutions, tools and technologies including **Jenkins**Nodes/Agent and configured various plugins for **Jenkins**to automate the workflow.
* Created **Maven**POMs to automate the build process for the new projects and integrated them with third party tools like, **SonarQube,** **Nexus**and created artifacts through source code and internal deployment in **Nexus**repository.
* Implemented multi-tier application provisioning in OpenStack cloud, integrating it with Chef/Puppet.
* Worked on **NoSQL**database **MongoDB**to **Replica**setup and sharing. Also experienced in managing **Replica**set
* Implemented **Zabbix,** **Nagios**for network monitoring and application monitoring of multiple metrics of various network services, servers in all the environments.
* Managed central repositories. Implemented **Atlassian**Stash along with **GIT**to host **GIT**central repositories for source code across products & login audits for Security Compliance. Performed code line **GIT**pushes (releases) to production and customer servers and developed and improved **GIT**push scripts.
* Worked with **JIRA**for creating Projects, assigning permissions to users and groups for the projects and created Mail handlers and notification Schemes for **JIRA.**Automated setting up server infrastructure for the **DevOps**services, using **Ansible,** by writing shell and **python**scripts.
* Install and configure **DHCP,** **DNS**( **BIND**, **MS**), web ( **Apache**, **IIS**), mail ( **SMTP**, **IMAP**, **POP3**), and file servers on Linux servers.
* Configured and integrated the servers for different environments to automatically provision and configuration management of **Linux**instances using **CM**/ Provisioning tool **Puppet**

**Environment:** AWS, Linux, Perl, Python, Chef, Puppet, Jenkins, PHP, Ruby, Maven, JIRA, GIT, Grafana, SVN, Nexus, Nagios, JIRA, Java, ANT, RHEL 6.7/6.8/7.1/7.2, Bash, C#, PowerShell, NT Command Shell

**Client: Princess Cruises, Santa Clarita, CA. April’16 – Feb’17**

**Role: DevOps Engineer**

**Responsibilities:**

* Developed and implemented Software Release Management strategies for various applications through the agile process Using DevOps tools.
* Worked with Development and QA teams to continuously integrate software development using Maven, GIT, and Jenkins.
* Developed build and deployment scripts using Maven as build tool in Jenkins to move from one environment to other environments and Involved in editing the existing Maven files in case of errors or changes in the project requirements.
* Used **Aws Lambda** for Server less code compiling and Kinesis for the product development.
* Implemented continuous delivery pipeline involving Jenkins, Ansible to complete the automation from commit to deployment.
* Installed Pivotal Cloud Foundry on EC2 to manage the containers created by **PCF.**
* Used Docker to virtualize deployment containers and push the code to **EC2 Cloud** using **PCF.**
* Debug and resolve Pre-Post OpenStack deployment failures.
* Communicated with groups to create **REST API** services to expose team data in a secure way with token-based authentication.
* Worked with the groovy scripts in Jenkins to execute jobs for a continuous integration pipeline where 'Groovy Jenkins Plugin' and 'Groovy Post-Build Action Plugin' is used as a build step and post build actions environments needs. And, configuration automation using Docker containers.
* Implemented **RESTful** APIs using **Groovy** and **Spring** Boot, facilitating seamless communication between different components of the application.
* Utilized **Groovy** scripting to develop and maintain backend services, contributing to the enhancement of user experiences.
* Collaborated with front-end developers to integrate scripts and ensure seamless communication between front-end and back-end systems.
* Wrote AWS Lambda functions in python for AWS's Lambda which invokes python scripts to perform various transformations and analytics on large data sets in EMR clusters.
* Implemented AWS Lambda functions to run scripts in response to event in Amazon Dynamo DB table or S3 bucket or to HTTP requests using Amazon API Gateway.
* Achieved Continuous Delivery on high scalable environment, used **Docker** coupled with **Nginx.**
* Deploy and monitor scalable infrastructure on **Amazon web services (AWS)** & configuration management using **Ansible.**
* Installed Ansible Tower and wrote playbooks, created roles to rollout configurations to different environments.
* Configured **Chef** Infrastructure. Setup chef infrastructure for various environments. Used opensource chef, chef-solo and chef-zero. Also, written various cookbooks for infrastructure automation.
* Built a configuration driven API which can work with multiple messaging systems like **RabbitMQ and WebSphere MQ.**
* Worked with **Nexus** repository manager to proxy, collect, and manage the dependencies on the collection of JARs.
* Used Kubernetes to deploy scale, load balance, scale and manage Docker containers with multiple namespace versions.
* Configured and managed environments DEV, QA, UAT and PROD on OpenStack for various releases and designed instance strategies.
* Worked with **Terraform** for automating VPC's, ELB's, security group's, SQS queues, S3 buckets, and continuing to replace the rest of our infrastructure.
* Developed Dev/Test/Prod environments of different applications on AWS by provisioning Kubernetes clusters on EC2 instances using Docker, Bash, Chef, and Terraform.
* Used **JIRA** tool to track all the defects and changes related to build and Release Team.

**Environment:** AWS, Jenkins, Maven, Docker, Ansible, OpenStack, Terraform, Kubernetes, Powershell, WebSphere, Chef, TFS, JIRA, Windows.

**Client: WELLSFARGO, Charlotte, NC. July’14 – March’16**

**Role: DevOps/Build & Release Engineer**

**Responsibilities:**

* Plan, deploy, monitor, and maintain Amazon AWS cloud infrastructure consisting of multiple EC2 nodes required in the environment and worked on application deployment and data migration on AWS (Cloud formation, Cloud Front).
* Involved in designing and deploying a multitude application utilizing almost all the AWS stack (Including EC2, S3, RDS, EBS, ELB, IAM) focusing on high-availability, fault tolerance and auto-scaling in AWS Cloud formation.
* Involved in improvement of CI/CD (Continuous Integration /Continuous Delivery) Software Release Management strategies for various applications.
* Created Python scripts to, totally automate AWS services which include web servers, ELB, Cloud front distribution, Database, EC2 and database security groups, S3 bucket and application configuration, this script creates stacks, single servers, or joins web servers to stacks.
* Maintained the FTP server in which the builds were copied.
* Designed the Infrastructure, and implemented on AWS cloud.
* Setup AWS VPC's for dev, staging and Prod environments.
* Vulnerability Assessment & Penetration Testing on the infrastructure on AWS for security.
* Responsible for using AWS console for deploying and operating AWS services specifically VPC, EC2, S3, EBS, IAM, ELB and Cloud Formation.
* Involved in designing and deploying multitude applications utilizing almost all the AWS stack (Including EC2, Route53, S3, RDS, Dynamo DB, SNS, SQS, IAM) focusing on high-availability, fault tolerance, and auto-scaling in AWS Cloud Formation.
* Knowledge in docker swarm orchestration, networking, security, storage and volumes.
* Continuous Integration: implemented and promoted use of Jenkins within the developer community. Validated Jenkins along with Bamboo.
* Regularly monitoring application server instances using Nagios, Dynatrace and Blue stripe.

**Environment:** Java, AWS, Eclipse, Tomcat, Apache, Chef, Oracle 11g, Jenkins, Python, Ruby, Bitbucket, JIRA, Maven, Git, Windows, Ruby, Chef, Windows.

**Client: Symbiosys Technologies, Visakhapatnam, India. Feb’12 – Aug’13**

**Role: DevOps/Build & Release Engineer**

**Responsibilities:**

* Supporting build, deployment, testing and release process in waterfall setup.
* Build tools and processes to support and enhance build and release infrastructure.
* Developed and maintained UNIX/Perl/Ant scripts for build and release tasks.
* Performed all necessary day-to-day GIT support for different projects.
* Created Clear Case configuration record for builds using derived objects generated during build audit process.
* Managed Creating and deleting **dynamic views** for developers as requested by user.
* Responsible for building and releasing packages for testing.
* Performing clean builds according to the schedule published for a given releases.
* Carried deployments and builds on various environments using continuous integration tool
* Installed **Build Forge** and performed troubleshooting during installation.
* Providing daily technical support, re-engineering and administration of an enterprise configuration Management system that supports multiple development applications.
* Responsible for creating Libraries, projects and jobs in Build forge for automating the build and deploy processes.
* Responsible for User Management and Group Management in **Build Forge**.
* Verifying that the methods used to create and recreate software builds are reliable and repeatable.
* Research the project application architecture to support/resolve build, compile, and test issues/problems.
* Releasing code to testing regions or staging areas according to the schedule published.

**Environment:** Version Control Tools, SCM, Clear Case, ANT, Shell Scripts, XML, UNIX, Build Forge.