****

**Yohann**

**DevOps Engineer**

**E: yohann.devops@gmail.com| P: 315-710-0471**

**SUMMARY:**

* **6+ years of Design/Plan, Install, Configure Linux Flavors** (CentOS, RHEL, and Ubuntu),Three years of experience with SQL and NoSQL databases, Two years on AWS in a DevOps Culture through Continuous Integration (CI) & Continuous Deployment/Delivery (CD) as an iterative process and Automation of Infrastructure as Code.
* Certified **AWS Solutions Architect** and **Professional SCRUM master**.
* Deployed **Docker Engines** in Virtualized Platforms for **containerization** of multiple apps.
* Worked with **Ansible** **playbooks** for virtual and physical instance provisioning, configuration management, patching and software deployment**.**
* Worked in infrastructure team on installation, configuration and administration of **CentOS 5.x/6.x/7**, **Red Hat Linux 8/9, RHEL 5.x/6.x/7**, **Red Hat Satellite 6, Windows Server 2008-R2/2012-R2** and **SUSE Linux** 10.x/11.
* Worked with Chef Enterprise Hosted as well as On-Premise, Installed Workstation, **BootstrappedNodes**, 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 **ChefCookbooks**.
* Experience in **Package Management** using Red Hat **RPM/YUM** and **Red Hat Satellite server**.
* Experience in **System Administration, System Builds, Server builds,Installs, Upgrades, Patches, Migration, Troubleshooting, Security, Backup, Disaster Recovery, Performance Monitoring and Fine-tuning on UNIX** Red Hat LinuxSystems.
* Experience working with **Puppet Enterprise and Puppet Open Source**. Installed, configured, upgraded and managed Puppet Master, Agents & Databases. Integration of **Puppet with Apache and Passenger**.
* Managed Amazon Web Services like **EC2, S3 bucket, RDS, EBS, ELB, Auto-Scaling, AMI, IAM** through AWS Console and **API Integration with Puppet Code**.
* Experience in working on source control tools like **Tortoise SVN, CVS, IBM Clear Case, Perforce, and GIT**.
* Strong knowledge on source controller concepts like **Branches, Merges and Tags**.
* Extensively worked on **Jenkins/Hudson by installing, configuring and maintaining for the purpose of continuous integration (CI)** and for **End to End automation** for all build and deployments.
* Created Custom as well as downloaded pre-written **Modules and Manifests**, Worked and utilized **Puppet hierarchy files, facts, and global variables.**
* Day to day job included but not limited to **handling Tickets, Monitoring, Troubleshooting and maintenance.**
* Experience using **MAVEN and ANT** as build tools for the building of deployable artifacts (jar, war & ear) from source code.
* Managed all the bugs and changes into a production environment using the Jira tracking tool.
* Involved in **setting up JIRA** as defect tracking system and configured various workflows, customizations and plugins for the JIRA bug/issue tracker.
* Managing **DNS, network load balancersandApache web servers** on **Linux** servers.
* Expertise in implementation of REST APIs and creating testing workflows using POSTMAN to test custom HTTP headers and bodies.
* Experienced with the installation, configuration of EC2 fleet c.x, m.x and g.x instances with EBS backed volumes.
* Installed, Configured, Managed Monitoring Tools such as **Splunk, AWS CloudWatch, CloudTrail** and used it with notification services like SNS and RollBar.
* Expertise in writing **Bash Scripts, Pearl Scripts (hash and arrays), Python programing** for deployment of Java applications on bare servers or Middleware tools.
* Installation, Configuration and Management of **RDBMS and NoSql** tools such as **SQL Server 2008/2012/2014, MySQL, DB2, PostgreSQL, MongoDB & Cassandra.**
* Day to day administration of the Development environment and Production and Test environment systems. 24x7 **on-call** support
* **Strong appetite for learning new technologies with positive attitude.**

**EDUCATION:**

* Master`s in Information Management, Syracuse University, School of information Studies
* Certificate of Advanced studies in Data Analytics, Syracuse University
* Bachelor of Engineering in Computer Science, St. Francis Institute of Technology, Mumbai, India

**Certifications:**

**Amazon Web Services Certified Solutions Architect – Associate** Dec2018-Dec 2020

**Profession Scrum Master (PSM1)** November 2019

**Google Analytics Individual Qualification (GAIQ)** October2019-October 2022

**TECHNICAL SKILLS:**

|  |  |
| --- | --- |
| **Build/Automation Tools** | Docker, Kubernetes, Puppet, Chef, Ansible, Jenkins |
| **Languages** | Shell, Bash, Ruby,Nodejs and Python scripting |
| **Databases** | SQL ServerPostgreSQL, MySQL, NoSQL, MongoDb, DynamoDb, Firebase, Redis |
| **Web/App Server** | Jenkins, Apache Tomcat, Nginx, AWS Lambda, EC2 |
| **Bug Tracking Tools** | JIRA, AWS CloudWatch, GitHub, GitLab, Rollbar |
| **Version Control Tools** | Subversion, GIT, Tortoise SVN, Visual SVN |
| **Web Technologies/Programing Languages** | React.js, Node.js, Sails.js, HTML, CSS, Python, Javascript, NodeJS, C++, Java, Solidity |
| **Cloud Technologies** | AWS, Azure, GCP, IBM Cloud |

**PROFESSIONAL EXPERIENCE:**

**Equitable -AXA Bank, Syracuse, NY May’19-Till Date**

**AWS/DevOps Engineer**

**Responsibilities:**

* Orchestrated Blockchain environments built on NodeJS and AWS CloudFormation stacks which were hosted using Docker Swarms.
* Managed AWS Linux environment instances on the cloud to act as master nodes in private blockchain networks. These instances primarily had python and bash script environments.
* Developed automated **CI/CD workflows** for several projects in the company throughout the years, including several **Blockchain** applications and developing **ETL workflows**.
* Used Docker Swarms to deploy and host nodes on Blockchain networks for creating test nodes.
* Implemented an **error reporting system** using Rollbar and **Nodejs** to notify developers of application crashes.
* Leveraged **AWS SNS** to notify developers via email when **API failures** are detected.
* Deployed **Linux servers** pre-configured with awscli and other packages specific to project environment for development use.
* Used **AWS Cloud9** to create VMs for development that were remotely accessible by company developers.
* Managed **IAM policies** for access to these development environments and used AWS Cloud Trails for monitoring use and access of these Virtual Machines.
* Utilized **AWS Secrets manager** to manage security of API keys used for testing.
* Enforced **company-wide coding standards by implementing event-driven checks** using Husky to revoke pushes if linting or commit formats did not meet company standards.
* Implemented **Data lake** for managing data generated through app and storing it based on schema, into respective **PostgreSQL databases and S3 buckets** on **AWS**.
* Created tests in Nodejs using **mocha** and **chai** libraries for testing this workflow.
* Developed workflows for managing docker containers based on tags to classify them into prod, dev and feat tags.
* Utilized project boards on **Github** to keep track of issues remaining and move them across boards as they progress toward being merged.
* For testing on Docker containers, used **Docker Secrets Manager** to secure API keys and implemented and API for easy fetching of keys from the Secrets Manager.
* Developed a pipeline on **AWS Code Pipeline using YAML** to create a workflow from pushing to commit to deploying to prod.
* Managed replica sets in docker containers to ensure high availability.
* **Migrated to Docker swarms to Kubernetes** services by transforming the Docker compose file to match the Kubernetes config spec.
* Deployed blockchain applications across Kubernetes cluster with each blockchain node acting as a Kubernetes pod.
* Implemented **logging services inside pods** which made requests to AWS Cloudwatch.
* Deployed the pod configs to **AWS EKS and used AWS Fargate** to get the clusters working into prod.

**Environments:***Kubernetes, Docker Swarms, NodeJs, mocha, chai, microservices, YAML, AWS EKS, S3, IAM, Fargate, Cloud9, Cloudwatch, ECS, SNS, Rollbar, Git, Github, PostgresSQL, python, shell scripting.*

**Syracuse University, Syracuse, NY Dec’18-May’19**

**AWS/DevOps Engineer**

**Responsibilities:**

* Managed several environments across AWS by separating them across CloudFormation stacks and ECS containers. Managed Apache servers and Linux environments on Docker containers and EC2 instances respectively.
* Used Jenkins server environments coupled with Code Chef pipelines to deploy CI/CD workflows that managed EC2 RedHat Linux environments.
* Wrote **Python Scripts to Monitor** Variety of Services &**Perl Scripts with Hash/Arrays** to Insert/Delete/Modify content in multiple servers
* Leveraged **Docker Engine** to run **Multiple Tomcat Instances** as Containerized AppServers.
* Installed, Deployed **Red Hat Enterprise Linux** 6.x/7.x, CentOS and installation of packages and patches for Red Hat Linux Servers.
* Administered and Engineered workflows using **Docker** containersto manage weekly **Build, Test and Deploy** chains as a **CI/CD** process, on **GIT** with **Dev/Test/Prod Branching** Model for weekly releases.
* Automated Weekly releases **Nodejs** scripting using **npm** package manager and **Github Actions** workflows.
* Working with the hardware vendors and application team for the **faulty replacement** and planning the downtimes accordingly
* Installed **Chef-Server Enterprise On-Premise/WorkStation/ Bootstrapped the Nodes** using **Knife.**
* Designed and implemented**Chef, including the** internal best practices**, cookbooks, automated cookbook CI and CD system.**
* Made use of **Jenkins for Chef CI, Thor for versioning, Berkshelf**for dependency management, and Test-Kitchen for testing and development.
* Migrated VMWARE VMs to AWS and Managed Services like **EC2, S3 Bucket, Route53, ELB, EBS Etc… with Opscode Chef Cookbooks/Recipes.**
* Applied patches every quarter regularly to meet audit requirements using **AWS Systems Manager** and monitored service utilization using **AWS CloudTrail.**
* Performance Tuning and Management for **Linux/AIX** server and working with the application/database team to resolve issues.
* Worked on **Installation/Configuration/Administrated**microservice containers using a **Docker swarm manager.**
* Creating the file systems using **EFS and EBS** volumes on AWS.
* Worked on**GitHub** to create project boards and aided the development team in managing weekly **SCRUM** sprint tickets.
* Created **Docker Swarms** for each project on Docker Enterprise, with each container running a microservice.
* Enforced **SCRUM** agile workflow and worked with development teams and business areas to plan future capacity requirements and hold regular meetings to review usage as well as create, revise, and report any new measurements required to manage distributed environments.
* Provide drill down reporting for application teams to use in monitoring their application **resource usage/performance** using **AWS CloudWatch**.
* Promote involvement of application development teams to make recommended changes that improve **program performance.**
* Insure impact of software releases to **system performance** and **resource utilization** of distributed and mainframe platforms are acceptable during load testing.
* Configured application **Load Balancers** on AWS to monitor server capacity by calculating **CPU** and **Memory**usage to provision new servers to meet current and future demand.

**Environments:***Docker, Kubernetes, Chef Cookbooks, Opscode, AWS ECS, ELB, S3, Cloudwatch, CloudTrail, Perl, Python, Github, RedHat Enterprise Linux, SCRUM Agile, AWS Systems Manager.*

**I Space, India Oct’16-Jul’18**

**DevOps Engineer**

**Responsibilities:**

* Worked across multiple environments, from on premises VMSpheres to AWS CloudFormation stacks. Jenkins server environment was used to create pipelines in both, VMSphere and across AWS EC2 instances.
* Architected cost saving solutions by migrating from VMSphere to AWS services and developed a web service to help NGOs in cloud security compliance.
* Experience in **AWS**, which included managing application in the **cloud and creating instances**.
* Manged **VMWARE vSphere HOSTS in a HA/DRS Environment with vMotion/ Site Recovery Manager, Replication Manager vRealize Automation Center and Fiber SAN.**
* Implemented Migration from**VMWARE to AWS using VMDK Import tools**.
* Installed **Jenkins/Plugins for GIT Repository,** Setup **SCM Polling** for Immediate Build with **Maven** and **Maven Repository (Nexus Artifactory) and** Deployed Apps using custom ruby modules through **Puppet** as a **CI/CD Process.**
* Managing Amazon Web Services (AWS) infrastructure with automation and configuration management tools such as **Ansible, Puppet,** or custom-built .designing cloud-hosted solutions, specific AWS product suite experience.
* **Installed/Configured/Managed Puppet Master/Agent. Wrote custom Modules and Manifests**, **downloaded pre-written modules from puppet-forge**. **Upgradation or Migration of Puppet Community and Enterprise**.
* Deploy and monitor scalable infrastructure on Amazon web services **(AWS) & configuration management using puppet.**
* Developed and supported the Red Hat Enterprise Linux based infrastructure in the cloud environment.
* Worked on installation, configuration and **maintenance Red hat, CentOS and Suse Servers at multiple Data Centers.**
* Experience in migration of consumer data from one production server to another production server over the network with the help of Bash and Perl scripting.
* Installed and configured monitoring tools **Nagios for monitoring the network bandwidth** and the hard drives status.
* Developed automation scripting in**Python (core) using Puppet to deploy and manage Java applications across Linux servers.**
* Installed and configured **DCHP server to give IP leases to production servers.**
* Management of RedHat Linux user accounts, groups, directories and file permissions.
* Applied **the Clustering Topology** that meets High Availability and Failover requirement for performance and functionality.
* Installation, Configuration and administration **of DNS, LDAP, NFS, NIS, NIS+ and Send mail on Redhat Linux/Debian Servers.**
* Configured, managed **ESX VM’s with virtual center and VI client**.
* Performance monitoring using **Sar, Iostat, VMstat and MPstaton servers** and also logged to munin monitoring tool for graphical view.
* Installed packages with yum and rpm.
* Installation and configuration **of PostgresSQL database on RedHat/Debian Servers**.
* Performed Disk management with the help of **LVM (Logical Volume Manager).**
* Configuration and Administration of **Tomcat**/**Apache Web Server and SSL.**
* Created and maintained network users, user environment, directories, and security.
* Provide the support of building the server, patching, user administration tasks, deployment, software installation, performance tuning and troubleshooting and KVM.
* Provided 24/7 on call support on Linux Production Servers. Responsible for maintaining security on Red hat Linux.

**Environments:**Apache Tomcat, AWS EC2, CloudFormation, Git, Github, PostgreSQL, Debian, CentOS, VMware, Ansible, Puppet, Python, yum, rpm, Gradle, Maven.

**Inversedime Solutions, India Jan’14-Sep’16**

**System Engineer**

**Responsibilities:**

* Worked on Red hat Linux servers to maintain, monitor and host company applications. Used shell scripting in a bash environment to create corn jobs for automating tasks.
* Administered the deployment and maintenance of a food service application that was put into production.
* Deploying the **Redhat Linux enterprise** server as like existing versions **5.x, 6.x** versions
* Creation of Kickstart images for the different version of Redhat Linux Operating system like 5.x & 6.x
* Creating the **server profile** and making the network and SAN virtual configuration using **Virtual Connect** in blade center C7000
* Creating the **filesystems** using redhat volume manager and performing the health check on regular basis for all **Linux** serves
* Scanning the newly assigned **LUNs** to the serves and assigning them to **volume group** and increasing the **filesystem** using redhat volume manager
* Mounting & unmounting the netapp storage LUNs to the **Redhat Linux** servers and troubleshooting the issues encountered
* Creating Link aggregation **(LACP)** with **VLAN** tunneling using **virtual connect (VC)**
* Share uplink set **(SUS)** using **LACP**&**VLAN** tagging
* Performance Tuning and Management for **Linux/AIX** server and working with the application/database team to resolve issues.
* Creating the **filesystem** and **RAW** logical volumes for the **Oracle RAC** based on the requirement
* Upgrading the **kernel** in all the **Redhat** Linux servers and creating initrd image to boot from the Upgraded kernel
* Migration of **Redhat** Linux servers from 4.x version and working with the application team to resolve the issues post migration.
* Installation/Configuration of **ESXi** 3.5, 4.x servers and applying **security patches** to the ESXi servers
* Storage mapping on **Red hat Linux** and create the **Volume group** , **Filesystems** using Logical Volume manager
* Installed/Configured **red hat Linux cluster 5x** version & Configuring the cluster resources
* Adding storage to the **cluster disks** and **increasing/ decreasing** the filesystem in **RHEL.**
* Creation of Fence devices in the cluster, Creation Failover domains for the cluster and **Flip over/ Failover** test in between the **nodes** in the cluster
* Working on **SAN** storage on **ESXI** servers and assigning the **LUNs** to the Virtual Machines running on the **Six** server.
* Performed **V**up 5.0, Sybase 11.5.1, Perl Scripting, NIS, NFS, LDAP server, WebLogic, Apache Tomcat, EMC Storage SAN.
* Working with **backup** team for the **Legato** backup & Restore, **Legato** client Installation in the  **Redhat Linux** server
* Installed and Configured **SAMBA** Server in Linux for accessing the shared files from the windows environment
* Configuring and Maintaining of Network File Systems like **(N.F.S)** and Auto mount in networks
* Installing and updating **Kernel**. Created and modified **swap** files and added swap space
* **NTP** server and client configuration for **synchronization** of timing of all the server with the NTP server

**Environments:**Redhat Enterprise Linux, Shell scripting, Bash, Apache, Ruby, Python, Oracle DBMS, PowerShell.