**AKSHAY D.** Icon

Description automatically generated with low confidence[akshaydr948@gmail.com](mailto:akshaydr948@gmail.com)

A close up of a sign

Description generated with very high confidence

**+1(919)228-9429**

**Senior Cloud DevOps engineer with over 8 years’ experience in Cloud (Azure, AWS), DevOps, Configuration management, Infrastructure automation, Continuous Integration and Delivery (CI/CD). I can implement effective strategies for N - Tier application development in both Cloud and On-premises environments. Experience in dealing with Unix/Linux and Windows server administration.**

**PROFESSIONAL SUMMARY:**

* I am an **AWS Certified solution architect** with experience in solution architecting applications and running them in the cloud and experience in network and security design for applications in the cloud. Also done extensive automation using **cloud APIs** and **SDK’s** to create infrastructure programmatically in the cloud.
* Experience in configuring **Azure** web apps, **Azure** App services, **Azure** Application insights, **Azure** Application gateway, **Azure** DNS, **Azure** Traffic manager, App services, Analyzing **Azure** Networks with **Azure** Network Watcher, Implementing **Azure** Site Recovery, **Azure** stack, **Azure** Backup and Azure Automation.
* Handled escalated Support tickets till closure for MS **Azure** PaaS platform. Configured AD connect to configure federation with on-premises ADFS and **Azure** AD. Experience is using Microsoft **Azure**, Tomcat, **JBOSS**, Web logic and WebSphere Application servers for deployment.
* Hands-on experience Designing, planning and implementation for existing On-Prem applications to **Azure** Cloud (ARM), Configured and deployed **Azure** Automation Scripts utilizing **Azure** stack (Compute, Web and Mobile, Blobs, ADF, Resource Groups, **Azure** Data Lake, HDInsight Clusters, **Azure** Data Factory, **Azure** SQL, Cloud Services and **ARM**), Services and Utilities focusing on Automation.
* Involved in designing and deploying applications utilizing almost all **AWS** stack like **EC2, Route53,** **S3, ELB, EBS, VPC, RDS, DynamoDB, SNS, SQS, IAM** and focusing on high availability, fault tolerance and auto-scaling in **AWS** Cloud formation, deployment services (OpsWorks and Cloud Formation) and security practices (**IAM**, **CloudWatch** and CloudTrail)
* Worked in an environment of Amazon Web Services (**AWS**) products gained good knowledge in Elastic Cloud Computing, Simple Storage Services, **Cloud** **Formation**, **Glacier**, **Block** **Storage**, **Elastic** **Beanstalk**, **Amazon** **Lambda**, **Virtual** **Private** **cloud**, **Load** **balancing**, **Relational** **Database** **Service**, and **Cloud** **Watch**.
* Expertise in implementing a production ready, highly available, fault tolerant **Kubernetes** infrastructure. Worked on Scheduling, deploying and managing container replicas on a node cluster using **Kubernetes.**
* Experience on **Kubernetes** for creating Projects, Services for load balancing and adding them to Routes to be accessible from outside, Creation of Pods through new application and control the scaling of pods, troubleshooting pods through ssh and logs.
* Hands on experience in deploying **Kubernetes clusters** on top of amazon EC2 instances using **KOPS** and managed local deployments in **Kubernetes** creating local clusters and deploying application containers.
* Building/Maintaining **Docker** container clusters managed by **Kubernetes** Linux, Bash, **GIT**, **Docker**, on **GCP** (Google Cloud Platform). Utilized **Kubernetes** and **Docker** for the runtime environment of the CI/CD system to build, test deploy.
* Virtualized **Docker Images** on Amazon **EC2,** created Docker files, used **Nexus** Repository as a Private **Docker Registry** for **Docker Images** and Clustered the Containers by using **Docker Swarm** and **Kubernetes.**
* Extensively worked with Scheduling, deploying, managing container replicas onto a node using Kubernetes and experienced in creating **Kubernetes** clusters work with Helm charts running on the same cluster resources.
* Orchestrated and migrated continuous integration & continuous deployment (**CI/CD**) processes using **Cloud** **Formation** and **Terraform**, packer Templates and Containerized the infrastructure using **Docker**, which was setup in Open Shift, **AWS** and VPCs.
* Experience with **Docker** container configuration, Docker Engine, **HUB**, Machine, **Compose** and **Docker** **registry** and activation, handling live webhosting & troubleshooting.
* Configured **Ansible** control machine **Ansible Playbooks, Roles and Modules**. Used **Ansible Vault** and **Ansible Tower** to secure sensitive data and control the access of playbooks.
* Experience in working with **Ansible** **playbooks** for virtual and physical instance provisioning, configuration management, patching and software deployment on cloud environments.
* Worked with **Jenkins** and **Hudson** to provide continuous automated builds for deployment and build management system. And Application management can use **Service Now** (CI) data to bring computers into a desired state by Managing Files, Services, or Packages installed on **Physical or Virtual Machines.**
* Integrated **Jenkins** with **Docker** container using **Cloud bees Docker pipeline** **plugin** and provisioned the EC2 instance using **Amazon EC2 plugin**.
* Implemented a centralized logging system using log stash configured as an **ELK** stack (**Elastic search, Log stash, and Kibana**) to monitor **system logs, VPC Flow logs, Cloud Trail Events**, changes in S3 etc.
* Experience in working on source control management tools like **GIT** (**GitHub,GitLab**), Subversion (SVN) and creating **GIT** repositories with specified branching strategies.
* Strong experience in installation, configuration, maintenance and troubleshooting of **RedHat**Linux, **SUSE Linux**, **Ubuntu, Fedora, CentOS**, **Sun Solaris**, **Windows** Servers and customizing the **Kickstart and Jumpstart** scripts to automate the installation of various servers used for various internal projects.
* **TECHNICAL SUMMARY:**

|  |  |
| --- | --- |
| Operating systems | Unix, SUSE Linux, Ubuntu, Centos, RHEL, Windows |
| Application/Web servers | WebSphere, WebLogic Server, Red Hat JBoss, Apache Tomcat |
| Databases | MySQL Server, Oracle, DynamoDB |
| Build/Project tools | Apache Ant, Apache Maven |
| Continuous Integration  tools | Jenkins, Atlassian Bamboo, Hudson |
| Configuration tools | Chef, Puppet, Ansible |
| Monitoring/logging tools | Nagios, Splunk, Elastic Search, Logstash, Kibana |
| Containerization tools | Docker, Kubernetes, Mesos, OpenShift |
| Virtualization technologies | VMware Workstation**,** Oracle VirtualBox |
| Collaboration tools | Jira, Slack, ServiceNow |
| Scripting | Perl, Shell, Bash, Python, PowerShell, Ruby, YAML |
| Version control tools | GIT, GitHub, GitLab |
| Cloud environments | Amazon Web Services, Microsoft Azure,OpenStack, PCF,GCP |

**WORK EXPERIENCE:**

**Client: Donuts.inc, Bellevue, Seattle Nov 2019-present Role: Sr. DevOps/cloud engineer**

**Responsibilities:**

* Involved in designing **Azure Resource Manager** (ARM) templates, deployed to Azure and customized build process using **PowerShell**. Creating **Azure Backup vault** and protecting required VMs to take the VM level backups.
* Configured VMs availability sets using **Azure** **portal** to provide resiliency for **IaaS** based solution and scale sets using **Azure** **Resource** **Manager** to manage network traffic. Created **Azure** Automation Assets, Graphical runbooks, **PowerShell** runbooks that will automate specific tasks.
* Developed and deployed resources using **Azure Resource Manager** (ARM)Templates and maintain, support infrastructure for teams using VSTS. Build out **VSTS** Infrastructure of the application using Azure Automation Runbooks.
* 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 (VSTS) also called as **Azure DevOps** Services.
* Designed and configured **Azure Virtual Networks (VNets), subnets, Azure network settings, DHCP address blocks, DNS settings, security policies and routing.**
* Configure diagnostics, monitoring and analytics on **Azure** using **API Management,** Logic Apps and set up documentation using Swagger API, while focusing on scale and resilience for **Azure Web Apps**.
* Worked on deploying AWS environment and Azure DevOps environment using Terraform to create infrastructure.
* Deployed the initial **Azure** components like **Azure** Virtual Networks, **Azure** Application Gateway, **Azure** Storage and Affinity groups.
* Configuring **Active Directory** Federation Services **(ADFS)** authentication flow, **Azure VPN** for Virtual Network. Integrated **office 365** with **Azure Active directory** to synchronize with SAAS applications.
* Created, Monitored and Restored **Azure SQL** **database**. Migrated Microsoft **SQL server** to Windows **Azure SQL database** and updating the connection strings**.**
* Worked with **Terraform** Templates to automate the Azure Iaas virtual machines using terraform modules and deployed virtual machine scale sets in production 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.
* Experienced in Using **Jenkins** pipelines to drive all Microservices builds out to the **Docker** registry and then deployed to **Kubernetes**, Created Pods and managed using **Kubernetes.**
* Implemented a load balanced, highly available, fault tolerant, auto scaling **Kubernetes AWS** microservices container, developed networking policies for **Docker** Containers, created Pods and deployed to **Kubernetes**.
* Built Jenkins pipeline to drive all microservices builds to Docker registry and deploy to Kubernetes. Created Jenkins jobs to create **AWS** Infrastructure from **GitHub** repositories containing **Ansible** playbooks.
* Integrated **AWS** Dynamo DB using **AWS** lambda to store the values, items and backup the **Dynamo** **DB** streams and Used **Kubernetes** cluster level logging to separate master and data elastic search nodes into their own pods.
* Developed microservice on boarding tools leveraging Python and Jenkins allowing for easy creation and maintenance of build jobs and **Kubernetes** deploy and services.
* Container management using **Docker** by writing Docker files and set up the automated build on Docker HUB and installed, configured **Kubernetes**. Deploying cluster on **AWS** with **Jenkins**, **Docker** pipeline implementation.
* Written Templates for **AWS** infrastructure as a code using **Terraform** to build staging and production environments. Defined Terraform modules such as Compute, Network, Operations, and Users to reuse in different environments.
* Worked with **AWS** Cloud Formation **Templates**, terraform along with Ansible to render templates and Murano with Heat Orchestration templates in OpenStack Environment.
* Provisioned the highly available **EC2** Instances using Terraform and wrote new plugins to support new functionality in **Terraform**. Converted existing terraform modules, had version conflicts to utilize cloud formation in **terraform** deployments. Converted existing **terraform** modules, had version conflicts to utilize cloud formation in **terraform** deployments.
* Worked on Docker to create images and containerize the application build, by writing Dockerfile and Docker-compose **YAML** files.
* Created Ansible roles in **YAML** and defined tasks, variables, files, handlers and templates. Created inventory and configured the Ansible files for parallel deployment in Ansible for automating the Continuous delivery process.
* Used Ansible to Setup/teardown of **ELK** stack (Elasticsearch, Log stash, Kibana) and troubleshoot the build issues with ELK and work towards the solution. Used Ansible playbooks for provisioning instances on Open stack.
* Wrote **Ansible Playbooks** with **Python** SSH as the Wrapper to Manage Configurations of AWS Nodes and Test Playbooks on **AWS** instances using Python. Run **Ansible** Scripts to provision Development servers.
* Set 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**, **Gitlab**.
* Developed **Ansible** **roles** to maintain the large playbooks easily. Used the Ansible **Galaxy**, a shared repository for the roles to download, share and manage the roles
* Created and wrote shell scripts (ksh, Bash), Ruby, Python and PowerShell for setting up baselines, branching, merging, and automation processes across the environments using SCM tools like **GIT**, Subversion (SVN), Stash and **TFS** on Linux and windows platforms.
* Created **Linux** Virtual Machines using **VMware Virtual Center**, creating **VM** Templates, and **troubleshooting** all **Virtualization** related issues.
* Experience in administration, deployment, automation and management with various Version Control Systems like Apache Subversion (**SVN**) and **GIT, GitHub.**
* Implemented a Continuous Delivery pipeline with **Docker, Jenkins and GitHub**. Whenever there is a change in **GitHub, Jenkins** server automatically build a new **Docker** container from it.

**Client: *TSYS, Columbus, GEORGIA.*  Aug 2018-Oct 2019**

**ROLE: Sr. DevOps /cloud engineer**

**Responsibilities:**

* Worked with **AWS** CodePipeline and creating CloudFormation **JSON** templates to create custom VPC. Migrated a production infrastructure into **AWS** utilizing CodeDeploy, CodeCommit, OpsWorks. Knowledge of Amazon **ECS** Clusters and able to manage containers in it.
* Utilized **AWS CLI** to automate backups of ephemeral data-stores to **S3** buckets, **EBS** and create nightly AMIs for mission-critical production servers as backups. Deployed Elasticsearch with 10 nodes cluster in **AWS** environment for client applications.
* Configure and Manage **EC2**, Elastic Beanstalk, RDS, S3 Buckets, VPC, Security Groups, ELB, Auto scaling, Elastic Cache, Glacier, **SNS, SQS, SES,** CloudWatch, **ECS**, CloudTrail, **CloudFormation**.
* Highly expertise with AWS solutions using **CloudFormation** Templates and exhibited leading launch configurations to automate continuous provisioning of **AWS** resources for applications.
* Strong experience in setting up the enterprise infrastructure on Amazon Web Services (AWS) like Auto Scaling, CloudFormation, CloudTrail, CloudWatch, DynamoDB, **EBS, EC2**, **ELB, IAM**, Redshift**, RDS, S3, VPC**, Route 53, CloudFront, security group and **AWS** lambda.
* Designed **Terraform** templates to create custom sized VPC, subnets, NAT to ensure successful deployment of Web applications and database templates and migration from traditional to cloud environment.
* Developed CI/CD system with **Jenkins** on Kubernetes container environment, utilizing **Kubernetes** and **Docker** for the runtime environment for the CI/CD system to build, test and deploy. Worked on **Kubernetes** to manage containerized applications using its nodes, ConfigMaps, selector services and deployed application containers as Pods.
* Configured **Kubernetes** Replication controllers to allow multiple pods such as **Jenkins** master server in multiple minions Managed Kubernetes charts using **Helm**, managed Kubernetes manifest files and created reproducible builds of the **Kubernetes** application.
* Integrated **Docker** container orchestration framework using **Kubernetes** by creating pods, Config Maps and deployments and Created **Docker** **images** using a **Docker** file, worked on **Docker** container snapshots, removing images and managing **Docker** volumes.
* Collaborated with development support teams to setup a continuous delivery environment with the use of **Docker** and Evaluated **Kubernetes** for **Docker** Container Orchestration and Used ServiceNow and **Jira** for ticketing/bug tracking.
* Deployed multiple databases and applications using **Kubernetes** cluster management, some of the services are Redis, Nginx and maintained Kubernetes to manage Containerized applications.
* Created **Docker** **images** using a **Docker** file, worked on **Docker** container snapshots, removing images and managing **Docker** volumes.
* Worked on Configuration Management tools like **Ansible** for transferring and running the modules with the help of **SSH** based systems.
* Created **Ansible** **Playbooks** and **Puppet** **Manifests** to provision **Apache** **Web** **servers**, **Tomcat** servers, **Nginx**, Apache **Spark** and other applications.
* Worked with **playbooks**, tasks, roles, facts and templates for **VAR** files and also configured files by conditions by keeping dynamic values and triggering **YML** files in **Ansible**.
* Automated various like **Continuous Deployment**, Application Server setup, **Stack Monitoring** using **Ansible Playbooks** and integrated **Ansible** with **Run Deck** and **Jenkins**.
* Deployed and configured Elastic Search, Log stash and Kibana (for log analytics, full text search, application monitoring in integration with Azure.

**CLIENT: *NIKE, BEVERTON, OREGON.* Jan 2018 to July 2018**

**ROLE: Cloud Engineer**

**Responsibilities:**

* Created **AWS** **CloudFormation** templates to create custom-sized **VPC**, subnets, EC2 instances, **ELB**, security groups. Worked on tagging standard for proper identification and ownership of **EC2** instances and other AWS Services like **Cloud Front, cloud watch, RDS, S3, Route53, SNS, SQS, Cloud Trail**.
* Implemented and maintained the monitoring and alerting of production and corporate servers using **AWS** Cloud watch. Used **AWS** CLI command line client and management console to interact with **AWS** resources and **APIs**.
* Designed and implemented scalable, source cloud architecture-based Amazon Web Services, **AWS** Cloud services such as **EC2**, auto scaling and Virtual Private Cloud (**VPC**) to build secure. Highly scalable and flexible systems that handled expected and unexpected load bursts and are able to quickly evolve during development iterations.
* Experienced in working with **Docker** container snapshots, attaching it to a running container, managing containers, directory structures and removing **Docker** **images**.
* Extensively used **Docker** for virtualization, Ship, Run and Deploy the application securely for fasten the Build/Release Engineering.
* Configured **Docker container** for branching purposes and Implemented Docker automation solution for Continuous Integration / Continuous Delivery model.
* Provided continuous improvement to agile software development teams by working with Jenkins under the CI/CD pipeline. Integrated Ant, Nexus, Jenkins, SVN, Confluence and JIRA.
* Used **Ansible** Workstation to maintain and manage **Ansible** Playbooks responsible for deploying Framework and other cloud tools.
* Migrating servers with the required configurations changes and **testing** and **deploying** the machines using **Ansible commands.**
* Followed **Ansible** best practices such as version control, variables, and vaults, building Ansible **module** with Ansible **playbook.**

**Client: *LABCORP, RALIEGH, NC* Feb 2017 to Dec 2017**

**ROLE: DevOps 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**).
* Planed, deployed, monitored, and maintained Amazon **AWS cloud infrastructure** consisting of multiple **EC2** nodes and VMWare virtual machines as required in the environment.
* Utilized Cloud watch to monitor resources such as EC2, CPU memory, Amazon to design high availability applications on **AWS** across availability zones.
* Experienced working with Ansible tower. Integrating Ansible tower with Jenkins to deploy code to different servers. Played a key role in converting 200+ existing Jenkins jobs into the new pipeline process. Prepared document on the process of migrating Jenkins jobs from development environment to production server.
* Collaborated with Development and Support teams to setup a **CD (Continuous Delivery) environment** with the use of **Docker** and **Wrecker**, automate continuous build and delivery tools.
* Virtualized the servers using the **Docker** for the test environments and development environment and performed configuration automation using **Docker** containers.
* Responsible for managing the **Chef** Client nodes and upload the cookbooks to chef-server from Workstation. Used **Chef** Configuration management system to develop automation and monitoring web application and database server.
* Installed Workstation, wrote **Recipes** and **Cookbooks** and uploaded them to **Chef** Server. Bootstrapped various nodes and the **Chef** server.
* Developed **Chef Recipes** in **Ruby** to configure, deploy and maintain software components of existing and cloud infrastructure and bootstrapped **chef client nodes.**
* Extensively worked in using **MAVEN** as build tools for building of Deployable Artifacts (jar, war & ear) from source code.
* Worked on **Jenkins**, **GitHub**, **Artifactory** and all internal build systems for the core development team on an enterprise-level Java-based cloud orchestration/automation tool.

**Client: *GENPACT, Hyderabad, INDIA* Mar 2014-Nov 2016**

**ROLE: Build and Release Engineer**

**Responsibilities:**

* Involved in **DevOps** migration/automation processes for build and deploy systems. Used Jenkins/Hudson and implemented CI/CD pipeline using plugins like conditional build step, deploy to Git.
* Created Jenkins jobs and distributing load on Jenkins server by configuring Jenkins nodes, which will enable parallel builds. Performed automation of **CI/CD** pipeline in private cloud using Jenkins shared libraries and multibranch pipeline and automated the static code analysis through SonarQube in Jenkins pipeline for checking code quality.
* Experience in creating various modules and automation of various facts in Puppet, adding nodes to enterprise **Puppet** master and managing Puppet agents. Creating Puppet manifests files and implementing Puppet to convert Infrastructure as code.
* Experience in writing **Puppet** manifests and modules to automate the deployment process and to integrating Puppet manifests into Jenkins/Hudson jobs for a continuous delivery(CD)framework.
* Worked on Python to pull a report from access report from Git through Rest API and configured through Jenkins.
* Implemented **Maven** as build tool on Java projects for the development of build artifacts on the source code.
* Integrated ant with **Shell scripts** created in Bash to automate the deployments for the Java-based applications.

**Client: *OMNISCIENT SOFTWARE PVT LIMITED, PUNE, INDIA* Jul 2013 TO Feb 2014**

**Role: Linux Administrator**

**Responsibilities:**

* **Linux Administrator** in a large team responsible for maintaining Linux operating systems such as RHEL, CentOS, Ubuntu and SUSE 10/11.
* Administered RedHat Linux servers for several functions including managing Apache/Tomcat server, Mail server, MySQL database and firewalls in both development and production.
* In charge of reviewing existing manual Software Builds, developing scripts to automate repeated tasks that are more susceptible to errors and risks using Shell scripting.
* Responsible for configuring and maintaining Squid server in Linux. Deployed Java applications into Apache Tomcat Application Servers. Used Test driven approach for developing the application and Implemented the unit tests using Python Unit test framework.
* Provided 24/7 on-call support on Linux Production Servers. Responsible for maintaining security on RHEL.