A blue hexagon with white text

Description automatically generated **Ravi Lurdhu**

**[lurdhuravi123@gmail.com](mailto:lurdhuravi123@gmail.com)**

**Phone: +1(314)-514-5305**

**Sr. Cloud DevOps Engineer**

**PROFESSIONAL SUMMARY**

Around 9 years of extensive experience in Automating, configuring, and deploying instances on cloud environments and Data centers. Experience in the areas of DevOps, CI/CD Pipeline, Build and release management, **AWS/Azure**, and Linux/Windows Administration.

* Experienced in designing and deploying applications utilizing almost all the **AWS** stack (Including **EC2, S3, Route53, ELB, EBS, VPC, RDS, DynamoDB, SNS, SQS, IAM, KMS, Lambda, Kinesis**) and focusing on **high- availability, fault tolerance** and **auto-scaling** in **AWS Cloud Formation,** deployment services (**Ops-Works and Cloud Formation**) and security practices (**IAM, CloudWatch, CloudTrail**).
* Experienced in cloud automation using**, AWS cloud Formation Templates**, **Chef, Puppet**.
* Implemented multiple **CI/CD** pipelines as part of DevOps role for on-premises and cloud-based software using **Jenkins, Chef and AWS/Docker.**
* Worked with **AWS** and **OpenStack** to provision new machines for clients.
* Configured and managed Elastic Load Balancing **(ELB)** to implement fault tolerance and avoid single point of failure of applications, thus providing high reliability and efficient usage.
* Involved heavily in setting up the **CI/CD** pipeline using **Jenkins, Maven, Nexus, GitHub, Puppet, Terraform, AWS** and **Azure.**
* Experienced in **Azure** IaaS, Provisioning VM's, Virtual Hard disks, Virtual Networks, Deploying Web Apps and Creating Web-Jobs, **Azure Windows server** 2008/2012/R2, Microsoft SQL Server, Microsoft Visual Studio, Windows PowerShell, Cloud infrastructure.
* Worked on **Kubernetes** to orchestrate the deployment, scaling, and management of **Docker** Containers and for the runtime environment of the **CI**/**CD** system to build, test and deploy.
* Used **Jenkins pipelines** to drive all microservices builds out to the **Docker registry** and then deployed to **Kubernetes**, Created **Pods** and managed using **Kubernetes.**
* Built and maintained **Docker** container clusters managed by **Kubernetes Linux**, **Bash**, **GIT**, **Docker**, on **GCP (Google Cloud Platform)**.
* Experienced in large production environments using configuration management tools like **Chef**, **Ansible** and **Puppet**, supporting **Chef** Environment with 500+ servers and involved in developing **manifests**. Developed Chef **Cookbooks** to manage systems configuration.
* Worked on writing **Chef Cookbooks** and recipes to automate installation of Infrastructure like **Apache Tomcat**, **JDK** and configuration tasks for new environments etc.
* Worked on developing **puppet** modules (blueprints) for installation, configuration, and continuous integration (**CI**) using combination of puppet master, **git**, **Open stack** (Horizon), **Vagrant** and **Jenkins**.
* Worked on functional usage and deployment of applications in **WebLogic, WebSphere,** and **Apache Tomcat Server**.
* Hands-on in using **OpenShift** for container orchestration with **Kubernetes**, container storage, automation, to enhance container platform multi-tenancy. Experience with **PCF, OpenShift, Kubernetes** architecture and design, troubleshooting issues and multi-regional deployment models and patterns for large-scale applications.
* Extensively worked on **Jenkins** and **Bamboo** by installing, configuring, and maintaining the purpose of **CI** and **End-to-End** automation for all build and deployments implementing **CI/CD** for the database using **Jenkins**. Used **Jenkins API** to query the **Jenkins** server state and change configuration and automate jobs on nodes.
* Experienced in branching, tagging, and maintaining the version across the environments using **SCM tools** like **GitHub, Subversion (SVN)** and **TFS** on **Linux** and **Windows platforms.** Maintained and administered **GIT source code** tool. Imported and managed multiple corporate applications into **GitHub code** management repo.

* In-depth knowledge of computer applications and scripting **like Shell, Ruby, Groovy, Python, Perl, and XML**. Utilized **MySQL, MongoDB**, **DynamoDB** and **Elastic cache** to perform essential database administration.
* Experience with **Python scripts** for pulling the data from **AWS** console for several resources like **EBS, EC2, Subnet**, **VPC, S3,** etc. to create the **Cloud Inventory**.
* Expertise in **Agile** Testing Methodologies & Software Test Life Cycle (**STLC**).
* Experienced with principles and best practices of Software Configuration Management **(SCM)** in **Agile, scrum, and Waterfall methodologies.**

**TECHNICAL SKILLS**

|  |  |
| --- | --- |
| **Operating System** | Linux (RedHat 4/5/6/7, CENTOS), Unix, Solaris 11/10/9/8, Ubuntu 12/13/14, Windows NT, Windows 8, 12, 12R2, MacOS. |
| **Cloud Platforms** | Azure, AWS |
| **Web Technologies** | HTML, CSS, XML, AJAX |
| **Automation/Configuration**  **Management Tools** | Jenkins, Chef, Puppet, Ansible, Kickstart, Jumpstart |
| **Virtualization Technologies** | Windows Hyper-V, VMware ESXi, Virtual Box, vCenter, vSphere, Power VM |
| **Containerization Tools** | Kubernetes, Docker, OpenShift, Spinnaker |
| **PaaS** | PCF, Open stack |
| **Application/Web Server** | Tomcat, Apache, Application Server, IIS, Oracle WebLogic Server 11g |
| **CI and Build Tools** | ANT, Maven, Hudson, Bamboo |
| **Testing and ticketing tools** | SonarQube, Selenium, JIRA, ServiceNow, Remedy, ConnectWise, |
| **Monitoring and Logging Tools** | Nagios, Splunk, ELK, Datadog, Dynatrace |
| **Version Control Tools** | GIT, GitHub, Subversion, Bit-Bucket, CVS, Clear Case |
| **Scripting** | Python, Shell Scripting, Ruby Scripting, Power Shell, YAML Perl |
| **Database Systems** | Oracle 10g/9i, SQL Server, MySQL, SQL Server, NoSQL, Mongo DB, Cassandra |
| **Networking/Protocol** | DNS, DHCP, Cisco Routers/Switches, NIS, NFS, WAN, SMTP, LAN, FTP/TFTP, TCP/IP |

**PROFESSIONAL EXPERIENCE**

**State Farm (Tempe, AZ) March 2019 – Present**

**Sr. AWS/ DevOps Engineer**

**Responsibilities:**

* Designed Azure Resource Manager (**ARM**) templates, deployed to Azure and customized build process using PowerShell. Created **Azure** **Backup** **vault** and protected 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.

A blue hexagon with white text

Description automatically generated

* 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, **DN**S settings, security policies and routing.
* Configured 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.
* In **AWS**, creating snapshots and amazon machine images (**AMIs**) of the instances for backup and creating clone instances.
* Installed and used **Chef** Server enterprise on-premises, workstation and bootstrapped the nodes using **Knife** command. Created various **Roles** & **Environments** & used **Chef Handlers** for auto kick-off requirement jobs.
* Deployed & configured **Chef server** & **Chef solo** including bootstrapping of client nodes for provisioning; Managed & configured hundreds of servers using **Chef**. Written several **Chef Cookbooks** & **Recipes** with **Ruby** scripting to automate the installation of WebLogic domain & **JDK** & customized **Recipes** from **Chef Supermarket** to align.
* Responsible for applying the Corporation's Software Configuration Management processes to projects, setting up and maintaining **TFS/GIT/GitHub** infrastructure and supporting a continuous delivery model by automating software build and package migration processes.
* Build and Deployment of **CI/CD** Pipelines.
* Used **IAM** to create new accounts, roles and groups and engaged in enabling **lambda** functions for dynamic creation of Roles.
* Configured AWS Identity and Access Management (**IAM**) users and groups for improved login authentication.
* Built scripts using **ANT** and **MAVEN** build tools in **Jenkins** to move from one environment to environments.
* Used **Nagios** log server for network device log collection, network device log visualization, **Nagios** Network analyzer for net flow collection and analysis of IP transactions.
* Created a **Python** Script to stop all the instances with a specific tag in AWS Instance using Lambda functions and made it into Cloud Watch Scheduler to schedule it every night.
* Integrated Team city with **TFS instance for CI and Octopus Deploy for CD**.
* Worked on **TFS** rally connector to synchronize user stories and tasks to directional.
* Worked with Dev teams on TFS performance monitoring in the new TFS 2013 environment.
* Migration of share point from WSS 3.0 to TFS 2013 and integrating it with **TFS** Involved in writing **Ansible** playbooks for setup and configuration of tools including **Jenkins** and antifactory on remote servers using **REST API protocol.**
* Automated build and the deployment process. Worked with the testing teams to automate test cases as part of the post deployment action and did Cucumber setup for testing automation.
* Installed and configured **KUBERNETES**, Chef Server/workstation, and nodes via CLI tools and wrote **Docker** files to create new images based on working environments for testing purposes before deployment.
* Created **Chef** Cookbooks to provision **Apache Web servers**, **Tomcat servers**, **Nginx**, **Apache Spark** and used **GZIP** to forward compressed files to destination node/instances.
* Installed and configured Splunk to monitor **EC2** Linux instances with **puppet** automation. Created indexes and generated reports.
* Installed and configured the latest version of **Docker** Datacenter including universal control plane (UCP) and **Docker** trusted registry (DTR) on a cluster of hosts.
* Created the branches in **GIT** to implement the parallel development process.
* Coordinated Release effort amongst various teams (Dev, QA, DBA and Business Analysis) in geographically separated environment.
* Performed tasks **on F5 load balancer** like ordering new certs, installing, and renewing SSL certificates, bouncing servers with performance issues after taking them out of the server pools, ID maintenance, User, Group management for all the **UNIX** servers.
* Associated in implementing **SonarQube** code quality analysis tool for the code.
* Created **MySQL Database** Objects like Schemas, Tables, Views, Stored Procedures, Indexes (cluster/Non cluster), and Constraints, Triggers, Buffer Pools, User Roles, and functions.

A blue hexagon with white text

Description automatically generated

* Have successfully managed high availability of production systems for corporate sectors and Individual Organizations as **MySQL DBA.**
* Implemented a real-time processing platform using **Kafka** and **Spark** streaming to persist data in **Cassandra**.

**Straviso (Pune, India) May 2017 – Jan 2019**

**Sr. Azure/DevOps Engineer**

**Responsibilities:**

* Configured **Azure Virtual Networks**, **subnets, DHCP address blocks, Azure network settings, DNS settings, security policies and routing.** Also, deployed **Azure IaaS** virtual machines and Cloud services (PaaS role instances) into secure **Virtual Networks and subnets.**
* Configured Azure Multi-Factor Authentication (MFA) as a part of Azure AD Premium to securely authenticate users and worked on creating custom Azure templates for quick deployments and advanced **PowerShell**scripting. Deployed Azure SQL DB with GEO Replication, Azure SQL DB Sync to standby database in another region & fail over configuration.
* Managed Azure Infrastructure **Azure Web Roles, Worker Roles, SQL Azure, Azure Storage, Azure AD Licenses, Office365**. Virtual Machine Backup and Recover from a **Recovery Services Vault** using Azure PowerShell and Portal.
* Used **Azure Terraform** to deploy the **Azure** infrastructure necessary to create development, test, and production environments for a software development project.
* Used Azure Kubernetes service to deploy a managed **Kubernetes** cluster in **Azure** and create an AKS cluster in the Azure portal, with the Azure CLI.
* Expertise in **Azure IaaS** virtual machines (VMs) and Cloud services (**PaaS** role instances) into secure **VNets and subnets**, designed **VNets** and subscriptions to conform to **Azure Network Limits.**
* Responsible for configuring **continuous integration (CI)** from source control, setting up build definition within **Visual Studio Team Services (VSTS)** and configure **continuous delivery** **(CD)** to automate the deployment of **ASP.NET MVC** applications to Azure web apps.
* 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)**alsocalled as**Azure DevOps Services**.
* Used Ansible and **Ansible** Tower as configuration management tool, to automate repetitive tasks, deploy applications and manage changes and automate software update and verify functionality.
* Configured Automation with Centralized Management using **Ansible** and **Cobbler**. Implemented **Ansible** to manage existing servers to automate the build/configuration of new servers.
* Wrote **Ansible** playbooks, inventories, created custom playbooks written in **YAML** language, encrypted the data using **Ansible** **Vault** and maintained role-based access control by using **Ansible** **Tower**.
* Implemented **Chef Recipes** for Deployment on build on internal **Data Centre Servers**. Also re-used and modified same **Chef Recipes** to create a Deployment directly into **Azure VM's.**
* Designed strategies for optimizing all aspect of the continuous integration, release and deployment processes using container and virtualization techniques like **Docker** and **Kubernetes**. Built **docker** containers using **microservices** project and deploy to **Dev.**
* Performed **IBM API** Connect infrastructure design, software installation, configuration and Advanced Developer Portal Installation and Configuration.
* **API** enabled CLEAR app functions using **IBM API** Connect, **APIs** are designed, developed, and tested locally using Developer Toolkit and published to **API** Manager catalog by using the Designer console.
* Worked with **Spinnaker** for **Continuous Deployment** with **Kubernetes**. And used **OpenShift** for container orchestration with **Kubernetes**, **Container Storage**, automation, to enhance container platform multi-tenancy.
* Managed **Kubernetes** charts using **Helm**. Created reproducible builds of the Kubernetes applications, managed **Kubernetes manifest files** and managed releases of **Helm packages.** Established a local dev workflow that centered around to validate deployments in **Kubernetes**.

A blue hexagon with white text

Description automatically generated

* Configured applications that run multi-container **Docker** applications by utilizing the Docker-Compose tool which uses a file configured in **YAML** format. Used **Kubernetes** to manage containerized applications using its nodes, Config-Maps, selector, Services, and deployed application containers as **Pods**.
* Setting up **Kubernetes** (k8s) Clusters for running microservices and pushed microservices into production with Kubernetes backed Infrastructure. Development of automation of **Kubernetes clusters** via playbooks in **aws.**
* Use **Ansible, Jenkins, Git**, for implementing Continuous Integration from **scratch** and optimize the Continuous Integration using **Jenkins** and troubleshoot the deployment build issues using the triggered logs.
* Defined and managed release planning and branching strategy of **GIT** servers. Also, helped educate engineering community on the appropriate use of **GIT**.  Implemented auto environment creation using **ansible** and **Azure**.
* Work with **MAVEN** for building the application has written **maven scripts** and **shell scripts** to automate the build process. Maintained the **GIT** source repositories and builds. Build tool, **Maven** used it on **Java** projects for the development of build artifacts on the source code.
* Planned and implemented **Disaster** **Recovery** solutions, data archiving, backup/recovery strategies, Data/Table partitioning, online Indexing, rebuild/reorg indexes, Performance Analysis, and optimization of **SQL server**.
* Developed **PowerShell** **scripts** to automate the project creation, setting permissions for users, groups in **TFS**. Developed automation scripting in **Python** using **Chef** to deploy & manage **Java .Net** applications across **Linux/Windows** servers.
* Installation, Configuring, Deploying, Troubleshooting, Administration, and maintenance of **IIS.** Analyzing the **IIS logs** to collect information about server events, server performance, and user activity.
* Work on **Nagios and ELK** for server monitoring, also used Net Scalar and monitoring tools like **Nagios** for Load Balancing and Controlling Flow of Traffic.
* Proficient with **Shell, Bash, Python, PowerShell, YAML, Ruby and Perl scripting** languages for automation, monitoring. Created **Azure PowerShell** script for APIM backup and restore operations. Created several **SQL** scripts as part of bug fixes to provide immediate solutions for some of the common issues on **Windows** machines.
* Created **Shell** and **Python** Scripts to automate creation of **AMI’s** through preboot and **bootstrapping** techniques. Developed automated processes that run daily to check disk usage and perform cleanup of file systems on **LINUX** environments using **shell scripting**.
* 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.
* Build **Data Sync** job on Windows Azure to synchronize data from **SQL 2012 databases to SQL Azure** as well as in installation, configuration & administration of Microsoft **Windows 2012 R2** Servers and **Microsoft Visual Studio,** used **GIT HUB** or source control, data collection, reporting, and project tracking.
* Analyze, troubleshooting, and resolve **RedHat Linux and Server HW**, **Network problems** in a Production Server environment.

**Ellucian (Banglore,India) Sep 2015-Feb 2017**

**Cloud/DevOps Engineer**

**Responsibilities:**

* Created highly available and scalable infrastructure in **AWS cloud** by using various **AWS** services like **EC2, VPC, Auto scaling, ELB, RDS, Route53.**
* Deploying the **OpenStack** components on multimode with High availability environment using **Red Hat** directory.
* Configuration subnet pools, networks, subnets, routers, floating-up to provide connectivity for OpenStack tenants.
* Experience with OpenStack Services such as Compute, Network, Storage, Dashboard, Image, Identity, Monitoring
* Configured **AWS EC2 Instances** using **AMIs** and launched instances with requirements of specific applications.
* Experience working with **IAM** to create new accounts, roles, and groups.
* Cloud watch monitoring of EC2 instances on CPU utilization, disk space and custom metrics and provided alerts to developers and Nagios Monitoring for on-site hosts and servers.
* Integrated **AWS** **Dynamo** **DB** using **AWS** **lambda** to store the values of items and backup the **Dynamo DB** streams.

A blue hexagon with white text

Description automatically generated

* Experienced in **AWS Elastic Beanstalk** for app deployments and worked on **AWS lambda** with **Amazon kinesis**.
* Automate operations processes using lambda, Cloud Watch Events, Schedules.
* Providing High availability with the help of RDS and multi-AZ failovers.
* Configuring bastion hosts on public subnets and gateway instances on public instances.
* Managing security groups on AWS, focusing on high-availability, fault tolerance, and auto-scaling using **Terraform** templates. Along with Continuous Integration and Deployment with **AWS Lambda** and **AWS Code Pipeline**.
* Creating **AMI Image**s and Cloud formation scripts to deploy instances and snapshots of **EC2** instance, **S3** backups are done on a regular basis. S3 backups are backed up as Glaciers when the data is not in usage.
* Performed **S3 buckets** creation, policies and on the **IAM** role based polices and customizing the **JSON template** and **Glacier** for storage and backup on **AWS.**
* Implemented and Deployed **Urban Code/Deploy** application to dynamically deploy Company Website Builds and created end to end automation of Continuous Deployment and Configuration Management in **Deploy.**
* Streamlined installation of **OpenShift** on partner cloud infrastructure such as AWS.
* Created private cloud using **Kubernetes** that supports **DEV, TEST**, and **PROD** environments.
* Wrote scripts to create **Kubernetes** cluster with RHEL and **Vagran**t. Defined **Kubernetes** services to discover and proxy requests to appropriate Minion.
* Configured **Kubernetes** **Replication** **controllers** to allow multiple pods such as Jenkins master server in multiple minions.
* Wrote an **Ansible** role that combined with **Terraform** will deploy an Elastic Beanstalk application and environment. It can auto discover SSL certificates and subnets for deployment. It allows us to enforce security best practice when deploying a new application.
* Implemented a Continuous Delivery pipeline with **Docker, Jenkins,** and **GitHub**. Whenever a new **GitHub** branch gets started, **Jenkins** automatically builds a new **Docker** container from it.
* Setting up **Jenkins** master, adding the necessary plugins and adding more slaves to support scalability.
* Installed, configuring, and administering **Jenkins** as **Continuous Integration** Server on Linux machines.
* Configured **Administration** **Configuration** for storing **Atlassian** **Bamboo's** system-wide properties such as the default URL, instance name, global variables, and System Properties.
* Maintained high availability clustered and standalone server environments and refined automation components with scripting and configuration management (Ansible) and experienced in writing **Ansible scripts.**
* Worked on installations and upgrades of Atlassian applications like **JIRA**, Confluence, **Stash/Bitbucket** and **Bamboo** using **PostgreSQ**L and **MySQ**L as databases.
* Managed data warehouse using **Amazon RedShift** to analyze all data using **SQL** & existing business intelligence tools. Converted existing **AWS** infrastructure to serverless architecture (**AWS** **Lambda, Kinesis**) deployed via **Terraform** & **AWS** **Cloud formation.**
* Integrated AWS **Dynamo DB** using AWS lambda to store the values the items and backup the **Dynamo DB** streams and implemented **Terraform** modules for deployment of applications across multiple cloud providers.
* Configured **AWS** Identity and Access Management (**IAM**) Groups and Users for improved login authentication. Provided policies to groups using policy generator and set different permissions based on the requirement along with providing **Amazon Resource Name** (**ARN**).
* Led implementation of **Azure Active Directory** for single sign-on access to thousands of cloud SaaS applications like **Office 365**, **Dropbox**. Also configured **Azure Role-based Access Control** (**RBAC**) to segregate duties within our team and grant only the amount of access to users that they need to perform their jobs.
* Configure **ELK** stack in conjunction with **AWS** and using Logstash to output data to **AWS S3.**
* Using **Chef** deployed and configured Elastic Search Log stash and Kibana (**ELK**) for log analytics, full text search, application monitoring in integration with **AWS** Lambda and Cloud Watch.
* Expertise in configuring the monitoring and alerting tools according to the requirement like AWS **CloudWatch, AWS CloudTrail, Dynatrace, Nagios, Splunk Enterprise** for the **VPN** connections.

A blue hexagon with white text

Description automatically generated

* Created AWS RDS MySQL DB cluster and connected to the database through an **Amazon RDS, MySQL DB** Instance using the **Amazon RDS Console**.
* Implemented Terraform modules for deployment of various applications across multiple cloud providers and managing infrastructure.
* Worked on **Nginx** webserver and maintained access logs, error logs and maintained different vhosts
* Written scripts in **Python** to automate log rotation of multiple logs from web servers.
* Implemented **Agile/Scr**um infrastructure to improve visibility and productivity using **JIRA.**
* Managed all the bugs and changes into a production environment using the **JIRA** tracking tool.
* Implementation of **RAID** 0/1/5 for creation of logical volumes using **VERITAS** Volume Manager and Red hat cluster servers in SAN Storage Area Environment.
* Maintaining on-site infrastructure with frequent backups and writing different scripts and cron files for backing data.

**Certifications:**

* AWS Certified Solutions Architect - Associate (SAA)
* <https://www.certmetrics.com/amazon/candidate/cert_summary.aspx>
* Pursuing AWS DevOps Engineer Certification Training
* Enrolled in Microsoft Certified: Azure DevOps Engineer Expert Training Program