**SAI KIRAN GONTYALA**

Mobile: 909-295-2823

Email: [sgontyala15425@gmail.com](mailto:saikiran.15425@gmail.com)

LinkedIn: <https://www.linkedin.com/in/sai-gontyala-a333a01b4/>

**Professional Summary:**

* **DevOps** **Engineer** with around 7 years of experience in managing and provisioning cloud services for application development using **Amazon Web services (AWS)**, **Microsoft Azure** and **GCP**. Expertise on implementing and automating end-to-end Continuous Integration, Delivery and Deployment processes and pipelines. In depth understanding of the Software development Life Cycles(SDLC) like **Waterfall, Agile, Scrum methodologies** and server-side methodologies in the applications.
* Experience in Infrastructure Development and Operations, involved in designing and deploying utilizing **AWS** stack that include **IAM, EC2, VPC, RDS, ELB, Auto scaling, CloudFront, Cloud Formation, ElacticCache, Elastic-IP’s, NAT Gateway, Cloud Watch, SNS, Route 53**.
* Implemented the serverless architecture **(SAM)** using **API Gateway, Lambda**, **Dynamo DB** and triggered the **AWS Lambda** code from Amazon **S3 buckets**.
* Configuring and networking of **Virtual Private Cloud** (VPC), Written **CloudFormation** templates and deployed **AWS** resources using it.
* Worked on templating **Amazon Machine Images (AMI),** creating **Elastic Block Storage (EBS)** snapshots to ensure backups and promote reusability. Assigned **Storage classes** to **S3** based on **Object Lifecycle** to reduce storage consumption cost.
* Created functions and assigned roles in **AWS Lambda** to run **python scripts**, and AWS Lambda using **JAVA** to perform event driven processing.
* Used **AWS Lambda** for **HTTP** requests and developed **python scripts** in response to the event in **Amazon DynamoDB** table and **S3 bucket** usage.
* Experience in Designing and distribution of Data across all the nodes and Clusters on different availability zones in **AWS Redshift** and Scaling Amazon **RDS, MySQL, MongoDB, DynamoDB**, **Cassandra DB, PostgreSQL**, Instances Vertically and Horizontally for High Availability.
* Experience on **AWS Elastic Block Storage** (EBS), different volume types and use of various types of **EBS volumes** based on requirement.
* Experience in using **Terraform** to provision infrastructure automation like deploying **AWS** instances, **S3** buckets and maintain their state among **large clusters Nodes.**
* Experienced with setup, configuration and maintain **ELK stack** (**Elasticsearch, Logstash and Kibana**) and **OpenGrok** source code (**SCM**)
* Experience working on Core **JAVA**, DevOps, J2EE, JSP, **Hibernate**, **Angular**, Web Services (SOAP & REST), MVC **Framework** (**Struts, Spring**).
* Hands-on experience in **Azure Cloud Services** (PaaS & IaaS), Storage, Web Apps, **Active Directory**, **ARM** **templates**, **Azure Container Service**, **VPN Gateway**, **Content Delivery Management**, **Traffic Manager**, **Azure Monitoring**, **OMS**, **Key** **Vault**, **Visual** **Studio** **Online** (VSO) and SQL Azure.
* Experienced in providing cloud Infrastructure using **Microsoft Azure**.Worked on creating Virtual Private Networks (**VPN**), Virtual Machines (**VM**) and using Azure services like **Blobs, Tables, Message queues**.
* Planned and implemented the **storage** and data management solutions in Microsoft Azure **(Azure files, Blob storage, Table storage, Queue storage, File storage)** and deployed **Azure SQL Server**.
* Good Knowledge in **GCP** platform Services like **Compute Engine, Functions, Container Security, Graphic Processing Unit, App Engine Knative, Cloud storage, Persistent Disk, Cloud Datastore**.
* Experience working with **Docker**. Written **Docker files** and built **Docker images** from it. Used Docker-compose for running **multi-container Docker applications** using **YAML**.
* Experienced on **orchestration** tool **Kubernetes** to provide a platform for **automating deployment, scaling** and operations of **application containers** across clusters of hosts.
* Good at writing **Ansible playbooks,** created custom playbooks written in **YAML**, encrypted the data using **Ansible Vault** and maintained role-based access control by using **Ansible Tower** to manage web applications, Environments configuration files, Users, Mountpoints and Packages.
* Experience in Installing and using **Chef** Server Enterprise, workstation and bootstrapped the nodes using **Knife** and automated by writing **ruby scripts** in **Chef Recipes** and **Cookbooks** with test-kitchen/chef.
* Good knowledge on **Puppet** configuration management tool which includes installing **Puppet master, agents** and writing **manifests** from scratch and pushing them to agents for **several POCs.**
* Used build tools like **Maven, Ant and Gradle** for building deployable artifacts such as **jar, war and ear** from source code and Artifactory Repository like **Sonatype Nexus** for Maven and Ant builds to upload artifacts using **Jenkins**.
* Experienced with source control management software, such as Subversion **SVN, Git**. Configured and deployed **GIT** repositories with **branching, forks, tagging, merge** requests and notifications.
* Experienced in writing **shell scripts (Bash)**, **Ruby**, **Python** and **PowerShell** for automation processes across the environments on Linux and windows platforms.
* Configured **Splunk** to monitor applications deployed on server through analyzing the server log files and application. Worked on setup of various dashboards, reports, and alerts in Splunk.
* Worked with **Nagios**, **AWS CloudWatch** to monitor network infrastructure to track and analyze real time performance, metrics and traffic of resources on cloud.
* Worked on **SQL, RedShift, MongoDB and DynamoDB** databases. Migrating the databases and writing Queries, Stored procedures, Triggers as per the requirements.
* Experienced in **JIRA** ticketing tool to track systems and configure various workflows, customizations, storyboarding, report generation.

**Technical Skills:**

|  |  |
| --- | --- |
| Cloud Services | AWS, Microsoft Azure, GCP |
| Operating Systems | Windows, Linux, Ubuntu, CentOS, Red Hat |
| Programming Languages | Java, JavaScript, Java Beans, JSP, SQL, HTML, XML, C++, Angular, Spring, Hibernate |
| AWS Cloud Services | IAM, VPC, EC2, S3, EBS, ELB, Route 53, SNS, CloudFront, SNS, ECS, EKS, Auto Scaling (ASG), Cloud Watch, Cloud Formation, Elastic Beanstalk |
| SCM / Version Control Tools | GIT, GitHub, SVN, Bitbucket |
| Build Tools | Maven, ANT |
| CI/CD Tools | Jenkins, Bamboo |
| Configuration Management Tools | Ansible, Chef, Puppet, Salt |
| Containerization Tools | Docker, Kubernetes |
| Monitoring Tools | Nagios, Splunk, CloudWatch |
| Bug Tracking Tools | JIRA, Scrum Manager |
| Scripts/Languages | UNIX, Shell Scripting(bash), Ruby, JSON/YAML, Python |
| Databases | Oracle, SQL Server, PostgreSQL, MS SQL, NoSQL, LambdaDB, DynamoDB, MongoDB, Aurora DB |

**Work Experience:**

**Client:** Hussmann Corporation March 2020 - Present

**Role:** DevOps Engineer

**Description:** Hussmann is a product-based company that manufactures display merchandisers, refrigeration systems, installation and services to food retailers around the globe. As a DevOps engineer in the product catalog team I am responsible for automating the deployment of updates on the products page by maintaining the cloud infrastructure by using CloudFormation and Ansible.

**Responsibilities:**

* Launching **Amazon EC2 Cloud Instances** using Amazon Web Services (Linux/ Ubuntu/RHEL) and Configuring launched instances with respect to specific applications.
* Worked on development of **AWS** Virtual Private Cloud (**VPC**) with advanced security features including **security groups** and network access control lists to enable inbound and outbound filtering.
* Extensively used **Elastic Load Balancing** mechanism with **Auto Scaling** feature to scale the capacity of **EC2 Instances** across multiple availability zones in a region to distribute incoming high traffic for the application.
* Managed several **IAM** **entities** in **AWS** for users with specific policies attached to each of them, implemented **Multi Factor Authentication** to meet security compliance.
* Used **Amazon RDS** to perform basic database administration and Managed networks using Security Groups for instance and subnet level and worked on AWS lifecycle management of data stored and later pushing it to Glacier for archiving data.
* Creating **S3 buckets** and managing policies for S3 buckets for storing static websites, logs, build artifacts, snapshots and creating a **snapshot** of an **EBS** volume and storing it in Amazon S3.
* Implemented a 'server less' architecture using **API Gateway, Lambda, and Dynamo DB** and deployed **AWS Lambda** code from **Amazon S3 buckets**. Created a **Lambda Deployment** function and configured it to receive events from your **S3 bucket.**
* Worked on functions in **Lambda** that aggregates the data from incoming events, and then stored result data in **Amazon DynamoDB**. Wrote **Terraform** templates for **AWS Infrastructure as a code** to build staging, production environments & set up build & automations for **Jenkins**.
* Written **Terraform** modules to create VPC subnets and spin-up multiple instances with defined configuration within the VPC created utilizing public and private subnets based on requirement.
* Worked 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
* Worked with **Docker**, pulling images from **AWS ECS**, running containers based on an image, creating **Docker file** to manage customized containers, exposing a container with port redirect, container volume management, docker inspect, docker commit to capture the file changes/settings into a new image and pushing the image to ECS.
* Created, managed and performed container-based deployments using **Docker** **images** containing Middleware and Applications together and Evaluated **Kubernetes** for Docker container orchestration.
* Created **Kubernetes Pods**, clusters, replication controllers, services, labels, health checks and ingress by writing **YAML** files and deployed micro services on Docker containers.
* Developed **YAML** scripts to manage and deploy environments using **Ansible** in a time efficient manner. Created **Ansible roles** by splitting the tasks into multiple YAML files promoting code reusability.
* Configured **Apache webserver** and **Tomcat** in the Linux AWS Cloud environment and automatic deployment of applications using Ansible automation.
* Built end to end **CI/CD Pipelines** in **Jenkins** to retrieve code, compile applications, perform tests and push build artifacts to **Nexus**.
* Responsible for designing and deploying best SCM processes and procedures with Subversion(SVN), **Bitbucket**, **Github**, **Git** & **eclipse**.
* Integrated **Jenkins CI/CD** with **GIT** and implemented continuous build based on check-in for code and created Webhooks to setup triggers for commit, push, merge and pull request.
* Wrote **custom Ansible Modules** in **Python** to make Continuous Integration, faster and flawless.
* Extensively worked on **Jenkins** by installing and provisioned on **EC2**, configuring and maintaining build servers using EC2 plugin in Jenkins as a **POC.**
* Used **Apache Kafka** to decouple data streams and Systems while integrating it with the **Hadoop Bigdata** technology for **fault tolerance** and **active tracking**.
* Monitored servers, switches and ports using **Splunk** Monitoring tool and assisted internal users in designing and maintaining production quality dashboards.
* Used **JIRA** as a defect tracking system for configuring various workflows, verifying the integrity & availability of server resources, customizations and plugins for the JIRA bug and issue tracker.

**Environment**: Git, Maven, Jenkins, Nexus, Docker, Kubernetes, Ansible, ELK, Lambda, Terraform, GitHub, Splunk, Jira, Apache, Oracle, Tomcat, Haproxy, SSL, WebLogic, Amazon Web Services, EC2, ELB.

**Client:** Emerson Technologies Oct 2019 - Feb 2020

**Role:** Cloud Engineer

**Description:** Emerson Technologies is an Automation Solutions company which provides technical assistance to vendors all over the world. Being a Cloud Engineer my responsibilities were to migrate all the on-premise data to the cloud platform. In the migration team my role was to create, monitor and automate infrastructure for environments, building AWS cloud Formation and migrated on-premise servers to the AWS Cloud.

**Responsibilities:**

* Worked with **Amazon Web Services** to set up infrastructure using AWS resources like **VPC, IGW, ROUTE TABLES, EC2, Glacier, ELB (Load Balancers), SNS, SWF and EBS etc**.  Define Sprint stories for the team and run a standup call for the team to track status.
* Managed multiple AWS accounts with **multiple VPC's** for both production and non-production where primary objectives included automation, build out, integration and **cost control**.
* Managed several **IAM** accounts in **AWS** for users with specific policies attached to each of them,

implemented **Multi-Factor Authentication** to meet security compliance.

* Deployed infrastructure as code using **CloudFormation** and developed **Lambda** functions to trigger AWS services depending on the network activity.
* Managed and build policies for **S3 buckets** and used S3 bucket, **Glacier** for storage and backup on **AWS** and managed an encrypted S3 bucket for all **CloudTrail** logs Encrypted bucket, and policies to allow **MSSP** to access the logs.
* Monitoring the **EC2** server's performance like CPU utilization, disk usage etc. using **CloudWatch** alarms that would notify the users and used metrics to monitor system performance **cost of resources** in **AWS**.
* Architected and configured a virtual data center in the **AWS cloud** to support Enterprise **Data Warehouse** hosting including **Virtual Private Cloud (VPC**), Public and Private **Subnets**, **Security Groups** and **Route Tables**.
* Monitored servers using **Nagios**, **Cloud watch** and using **ELK Stack Elasticsearch Fluentd Kibana**.
* Migrate Data from **Elasticsearch-1.4.3 Cluster** to **Elasticsearch-5.6.4** using **Logstash**, **Kafka** for all environments.
* Experienced with **security groups, network ACLs, Internet Gateways**, and **Elastic IP's** to ensure a safe area for organization in **AWS public** **and private cloud.**
* Developed a **Spark** **job** in **JAVA** which indexes data into **Elasticsearch** from external **Hive tab** Administer **CI/CD(Jenkins**) tools **stack administration** and **maintenance**
* Performed code line **GIT** pushes (releases) to production and customer servers. Developed and improved GIT push scripts to create **release pipelines**.
* Built and defined dependencies to build artifacts using the build Automation tool **Maven**.
* Strong experience in Linux and **UNIX** OS administration, maintenance and support including installing, maintaining, troubleshooting, and using Linux /**UNIX** standard systems and software applications, creating **UNIX** Make files, **ANT** build scripts and maintaining up to date dependency lists .
* Installed **Multiple Plugins** in **Jenkins** UI for building pipelines, build releases and created a master, slave configuration to implement multiple parallel builds.
* Created **Jenkins jobs** using various Jenkins files using **groovy script** via DSL (domain specific language) and used them as various workflow scripts.
* Managed the configurations of nodes using **Chef**, configured Jenkins builds for **Continuous Integration** and Delivery for **automated deployments**.
* Developed **ruby scripts** to author **recipes** and **cookbooks** by incorporating chef resources like package, cron, file, batch resources to set up compute instances on cloud infrastructure. Used **knife** to execute cookbooks and recipes to maintain desired configurations on chef clients.
* Involved in migrating applications from **EC2 servers** on to the **Docker Containers** for the test and dev-environment **needs, configuration** and **automation**. Leveraged Docker Engine to run Multiple **Tomcat** Instances as Containerized AppServers.
* Created **Docker Swarm** using Docker CLI to Orchestrate, Schedule and deploy the services to Swarm and managed the Swarm behavior and created **Docker files** for each Microservice based applications to the Docker container.
* Used **Docker swarm** for automating the docker container deployment and setup **Docker swarm** clusters to build the cron jobs to avoid the downtime/failover.
* Using **Nagios** to monitor the Application performance and Infrastructure health by analyzing the user dashboard and logs management.
* Setup **JIRA** bug and issue tracker for verifying the integrity, availability of server resources and configuring various workflows and plugins.

**Environment & Tools**: AWS, UNIX, GIT, MAVEN, ANT, Nexus, Jenkins, Chef, Docker, Kubernetes, Docker swarm, Nagios, Python, Shell, Bash, groovy script, MySQL, Nagios and JIRA.

**Client:** ESN Technologies Oct 2016 - Dec 2018

**Role:** Build & Release Engineer

**Description:** ESN Technologies is an IT company that provides technical support and cloud solutions to clients across the globe. As a Build and Release engineer, I worked on automating the build, release and maintaining, monitoring the application.

**Responsibilities:**

* Planned build and release cycles and organized project metrics by coordinating with development teams as per the requirements.
* Designed and deployed **AWS solutions** using **EC2, EBS**, Elastic Load Balancer (**ELB**), **Auto-scaling** groups and **Opsworks**. Worked on AWS Auto Scaling for providing high availability of applications on **EC2 instances** based on the load of applications using **AWS Cloud Watch**.
* Migrated on-**premise Database** Servers onto **EC2** instances on Cloud. Worked with **SQL** and **NoSQL** query languages on AWS using Relational Database Service (**RDS**) and DocumentDB.
* Hands-on experience with AWS IAM to set up user roles with corresponding user and group policies using **JSON**. Created roles, groups for users and resources using **AWS IAM**.
* Designed AWS CloudFormation templates to create custom sized **VPC, subnets, NAT** to ensure successful deployment of Web applications and database templates.
* Worked in version control tools like **SVN** and **GIT** and used branching, forking, merging for maintaining project repositories and build using **Ant** and resolute the source code merging conflict in the version control system GIT.
* Developed an effective build automation, continuous integration and test environment that reduced integration issues and improved code quality.
* Responsible for performing Build operations with **ANT Scripts**, while implementing the required modifications in the ANT Script to perform build operations, in accordance with requirements.
* Used **Chef configuration management** tool for creating and managing Chef **Cookbooks** and **Recipes** to automate system operations.
* Implemented, managed and configured **Chef Nodes** environments and roles for better environment management in Dev and Prod servers.
* Responsible for working with the external Release Management team to ensure deployment and activation notices were properly performed.
* Working with the development team on enforcing source control strategies, building, packaging and deployment of products.
* Setup and configured **Splunk** to monitor application and server logs. Implemented forwarder configurations, search heads and indexing on Splunk to classify logs and generate alerts to notify the users.
* Created epics, stories, spikes and other issues on **JIRA** implementing agile methodology. Also tracked and monitored project progress creating burndown charts for sprints on JIRA.

**Environment**: Amazon Web Services, Relational Database Service (RDS), IAM, VPC, SVN, GIT, ANT, Chef, Splunk, JIRA.

**Client:** Pranitha Software Services July 2013 - Sept 2016

**Role:** Linux Administrator

**Description:** Pranitha Software Services is an IT firm that provides services and solutions in Automation, ERP, Data Analytics and more. I worked as a Linux Administrator at Pranitha Software Services, I am responsible for providing technical resources to users and developers in meeting their requirements. I administered and monitored allotment of Linux systems, setting permissions, performing backups and keeping the systems up to date ensuring system security.

**Responsibilities:**

* Handling the day-to-day Operations, installing software, applying patches, managing file systems, monitoring performance and troubleshoot alerts.
* As a member of **Linux Admin** supporting the Technology Environment group responsible for Installation, maintenance, Administration and troubleshooting of **Red Hat** Enterprise Linux 4/5 systems.
* Provided 24x7 System Administration support for RedHat **Linux 3.x, 4.x** servers and resolved trouble tickets on shift rotation basis Provided the support of building the server, patching, user administration tasks, deployment, software installation, performance tuning and troubleshooting.
* Installed, implemented, maintained and support for the corporate Linux servers **RHEL 3, 4, 5, CENTOS 5, Ubuntu** and involved in System Administration, System Builds, Server builds, Installs, Upgrades, Patches, Migration, TroubleShooting, Security, Backup, Disaster Recovery and Performance Monitoring on UNIX (Red Hat Linux) Systems.
* Administered Linux servers by applying frequent patches and upgrades. Also kept servers up to date by installing required functionalities like **Apache/Tomcat Servers**, mail servers, database servers and other dependencies.
* Ensured network security of the systems using **firewall** by specifying white and blacklists. Also configured systems to add them to specific subnets and set **rules for traffic routing** over network.
* Managed Network troubleshooting applications **TCP/IP** including Ethernet, Configured **LAN, WAN** and IP addressing & **Subnetting, routing**.
* Used **Nagios** to monitor the day to day performance and maintenance operations throughout the client’s network.

**Environment & Tools:** Red Hat Linux, UNIX, CentOS, Ubuntu, Tomcat, TCP/IP, LAN, WAN, Nagios.