

VARUN GUDUGUNTLA

guduguntlavarun@gmail.com

Cloud Developer

770-765-5386

PROFESSIONAL SUMMARY

- Overall 7+ years, proficient in Build and Release Management, DevOps Engineering, Python scripting, Linux, and UNIX system Administrator with application configurations, code compilation, packaging, building, automating, managing, and Cloud Computing Platforms like **Microsoft Azure, AWS, Google Cloud Platform**, and **Containerization**
- Experienced in AWS Cloud Services such as **IAM, EC2, S3, AMI, VPC**, Auto-Scaling, Security Groups, Route53, **ELB, AWS Lambda, EBS, EMR, RDS, SNS, SQS, CloudWatch, CloudFormation, CloudFront, and Glacier**
- Worked on optimizing volumes and EC2 instances and created multiple VPC instances. Deployed applications on AWS using **Elastic Beanstalk**, Implemented and set up **Route53** for AWS Web Instances
- Experienced in creating Multiple AWS instances, creating Elastic Load Balancer and Auto scaling to design cost effective, fault-tolerant, and highly available systems
- Skillful in dealing with Microsoft Azure Services like PaaS (Azure Web Apps, Web Roles, Worker Roles, SQL), IaaS (Virtual Networks, VPN, Resource Groups, Load Balancing, Application Gateways, Autoscaling), and worked on storages. Well experienced in deployment, configuration management, and Virtualization
- Experienced in writing **Terraform** scripts for creating fault-tolerant and highly scalable cloud environments (in AWS/AZURE) and also developed, and maintained Cloud Formation JSON Templates for automating the provisioning of AWS resources (**IAM, EC2, S3, SNS, RDS, ELB**, and **Auto Scaling**), and installed workstations
- Experienced in installing **Kubernetes** and configure multiple nodes on AWS provisioning **EC2** instances for cluster formation and deployed a web application through nodes
- Experienced in Docker container orchestration framework using **Kubernetes** by creating Pods, Configuring Maps and deployments of **Microservices** based applications
- Worked on **Ansible** Configuration management and wrote **Ansible** playbooks to launch and Automate AWS instances on Ubuntu, Amazon Linux, and RHEL.
- Experience in the implementation, and maintenance of **VMware, DNS, DHCP, SMTP**, and Amazon Linux instances
- Good knowledge and experience in using Elastic search, kibana and fluent, **Cloud Watch, Nagios, Splunk, Prometheus** and **Grafana** for logging and monitoring
- Configured Networking Concepts **DNS, NIS, NFS**, and **DHCP**, troubleshooting network problems such as **TCP/IP**, providing support for users in solving their problems. Deployed code through ApacheTomcat server
- Wrote different **Ansible** playbooks for installing, configuration, and upgrading different Applications on the Servers
- Deployed code through web servers like **Apache** and **Nginx**. Strong knowledge of source controller concepts like Branches, Merges, and Tags
- Experienced in **MAVEN & ANT** as Build Tool for the building of deployable artifacts (war) from source code. Installation and deployment management repository generated by Maven and Ant artifacts, and store the build process using dependency jar
- Deployed the artifacts in Nexus central repository for internal deployments and Managed Nexus repository for the current project created a new repository and provisioned it. Have hands-on experience with bitbucket servers
- Experienced in setting up a CD environment with a focus on **UDeploy** and setup **UDeploy** server along with automated scripts to backup the server and restore from backup for disaster recovery
- Automated processes with custom-built Python & Shell scripts
- Deployed Java Applications and Web services using **CI/CD** tools like Jenkins, puppet in standalone and clustered environment
- Responsible for creating puppet modules and manifest files from scratch and experience in editing existing puppet manifests and automated manual installation process using puppet
- Expertise in using **Chef** for Infrastructure as Code(IaC), configuring chef-server which includes bootstrapping of chef-client nodes for provisioning and creating custom cookbooks, recipes, and uploaded them to the chef server

TECHNICAL SKILLS	
Cloud Computing	AWS, Microsoft Azure, Google Cloud, Open Stack
Configuration Management Tools	Ansible, Chef, Puppet, SaltStack
Scripting Languages	Python, Perl, Shell, Bash, Ruby
OS Administration	RedHat, Linux
Web/AppServers	Apache, Nginx, Web Logic, Apache Tomcat, LAMP, Airflow
Networking	DNS, DHCP, TCP/IP, SMTP, LDAP
Build Tools	ANT, Maven, Jenkins, TeamCity, UDeploy
Project Tracking/Bug Tracking Tools	Service NOW, JIRA, Remedy, Rally, IBM Clear Quest, Crucible, Bugzilla
Operating Systems	RedHat, CentOS, Ubuntu, Windows
Database Servers	MYSQL Server, Oracle, MONGO DB, Post greSQL, Cassandra
Monitoring Tools	Nagios, Splunk, ELK Stack
Version control tools	Git, GitHub, SVN
Virtualization/Container	Docker, Kubernetes, Vagrant, VMware vSphere 5.5/6.0

EXPERIENCE

Client : Cigna, Hartford, Connecticut

Aug'20 to Present Role: Cloud Developer

- Worked on developing the **AWS lambda** code using **Nodejs** modules and built code to call API for Brainshark Analytics Platform.
- Deployed and tested the **AWS Lambda** which runs code with the response of events
- Created functions and assigned roles in AWS Lambda to run **python** scripts
- Automating the deployment process using Shell scripts, where it copies the Jar and Ear files and automated artifacts and configuration files into WebLogic using Shell script and Jenkins.
- Configured **Cloud Watch** Scheduler to trigger Lambda function daily and build Lambda code such that it would send an **SNS** notification to the **S3** bucket
- Monitored **CloudWatch** logs to debug the code and check whether the response of events are running with AWS Lambda code
- Created python scripts to totally automate AWS services which includes web servers, **ELB, EC2, S3 bucket**
- Worked on **Terraform** scripts for **Tableau** automated install in **EC2 instances, primary nodes and worker nodes**. Expertise in writing new plugins to support new functionality in **Terraform**
- Worked on building resources through **CI/CD** Jenkins pipeline which runs when there are code changes in gitlab.
- Provisioned **EC2 instances, Primary Nodes, Worker Nodes, VPC, ELB, ALB , Cigna Golden AMI, Security groups, Route 53** in AWS through terraform scripts for infrastructure set up in tableau installation.
- Configured and installed **tableau** in AWS **primary nodes** and **worker nodes** through **shell scripts, python**.
- Built **Devops** for Configuration of Cloud Watch Scheduler for triggering Lambda function
- Deployed **Lambda** function for sending an output file for analytics platform to S3 bucket, sending SNS notification for copying the file from source bucket to vendor bucket.
- Configured **Docker** in Windows through Cigna Private Cloud(CPC) and deployed the Lambda function locally through **Docker** using plz commands
- Worked on troubleshooting AWS Lambda logic for the Community Cloud Extract by debugging the code and deploying the Lambda function locally using Docker
- Created **S3 buckets** and managed policies for **S3 buckets** and utilized **S3 buckets** and **Glacier** for storage and backup on AWS
- Worked in creating User/Group Accounts and attaching policies to User/Group Account using **AWS IAM**.
- Used **Tableau** to create data charts and visualizations to understand resource usage and for cost optimization.

Environments: Jenkins, AWS Cloud, Shell, Python, Rally, Service now, Active directory, Web logic server, Apache tomcat, Load balancer, Route 53, VPC, Elastic Beanstalk, Docker, Docker Containers, Kubernetes, Apache, Nginx, MYSQL and MONGO DB Databases

Client : Stanley Black & Decker, Atlanta, GA Jan'20 to July'20 Role: Sr. DevOps Automation Engineer

- Maintained Jenkins servers to deploy into production by and non-production environments
- Implemented a continuous Delivery framework using **Jenkins** and Maven in Linux environment.
- Integrated GIT into Jenkins to automate the code check-out process.
- Implemented multiple **CI/CD** pipelines as part of Devops role for on-premises using **Jenkins, Chef and AWS/Docker**
- Worked in implementing Datawarehouse solutions in **AWS Redshift**, worked on various projects to migrate data from one database to **AWS Redshift, RDS, ELB, EMR, Dynamo DB**
- Worked with Terraform to write CloudFormation scripts automating the provisioning of AWS resources like

IAM, EC2, S3, Route 53, SNS, RDS, ELB, AutoScaling.

- Defined AWS Security Groups which acted as firewalls that controlled the traffic allowed to reach one or more AWS EC2 instances.
- Worked on **AWS/EC2/VPC/S3/SQS/SNS** based on automation **Terraform, Ansible, Python, Bash** Scripts.
- Setting up the build and deployment automation for Terraform scripts using **Jenkins**.
- Worked with AWS cloud resources which include services like **EC2, S3, IAM, Security Groups, RDS, VPC, Route 53, Autoscaling, Cloud watch, Key Pairs** and **Elastic Beanstalk**.
- Worked on creation and managing IT infrastructure and application stack using **AWS Cloud Formation** and writing the template file using **JSON**. Created, monitors, alarms and notifications for **EC2** hosts using **Cloud Watch**
- Designing and Implementing for fully automated server build management, monitoring and deployment by using DevOps technologies like **Ansible** and **Chef**. Integration of **Configuration Management** Systems like Ansible and AWS for code deployment.
- Wrote Python modules for Ansible customizations
- Launched different **EC2** Virtual Servers in public cloud using **AWS** auto-scaling and pre-baked **Amazon Machine Image(AMI)** through Ansible Playbooks to make configuration changes.
- Used Ansible python script to generate inventory and push the deployment to AWS instances.
- Deployed Docker Images to **AWS ECS** by writing Task Definitions.
- Build **Docker** Images and run containers in the testing environment. Experienced with **Docker Swarm** for orchestration of Docker containers.
- Worked with **Docker** and **Kubernetes** for maintaining and administration of containers.
- Used **Kubernetes** to provide a platform for automating deployment, scaling, and operations of application containers across clusters of hosts
- Setup and maintained Logging and Monitoring subsystems using **Graphite-Grafana** and **Elastic Search Logstash-Kibana**. Established infrastructure and service monitoring using **Prometheus** and **Grafana**.
- Used **Zabbix** as monitoring tool and used zabbix plugin for **Grafana** for analysis & visualization.
- Created consolidated dashboards for code coverage and security vulnerabilities by data visualization tool Tableau.
- Automated **Sonar** and **Fortify** code analysis results to Tableau dashboard using API's for real time metrics
- Installed, deployed and managed Linux RedHat Enterprise, Ubuntu, and installed its patches and packages.
- Implemented Shell, Perl and Python scripts for release and build automation in respect to the requirements.
- Worked in scripting languages including **Shell scripting, Batch scripting, PERL scripting**.
- Developed automation scripting in **python** (core) to deploy and manage applications across Linux servers.

Environments: Jenkins, AWS Cloud, Chef Workstation, Ansible, Shell, Python, JIRA, Service now, Active directory, Web logic server, Apache tomcat, Load balancer, Route 53, VPC, Elastic Beanstalk, Codebuild, Docker, Docker Containers, Kubernetes, Apache, Nginx, MYSQL and MONGO DB Databases

Client : Amex GBT, Phoenix, AZ Aug'18 to Nov'19

Role: Sr. DevOps Engineer

- Worked on **AWS** and **EC2, VPC, Cloud watch, Key Pairs, Route 53, IAM** and **Elastic Beanstalk** cloud for provisioning , and managing infrastructure through automation.
- Deployed and configured Git repositories with branching, forks, tagging, and notifications. Experienced and proficient deploying and administering GitHub.
- Used Terraform to write CloudFormation scripts to automate the provisioning of AWS resources like **IAM, EC2, S3, Route 53, SNS, RDS, ELB** , and **Auto Scaling** and make calls to Puppet scripts that provision / configuration of servers on the instances.
- Involved in Devops role converting existing **AWS** infrastructure to Serverless Architecture(**AWS Lambda, Kinesis**) deployed via CloudFormation.
- Integrated **Tableau** with AWS to monitor data sources like **Amazon Redshift, Amazon Aurora**, and **Amazon EMR(Elastic Map Reduce)**.
- Created monitors, alarms and notifications for **EC2** hosts using **Cloud Watch**.
- End to end deployment for projects on **AWS** using python scripts for automation, scalability.

- Expertise in writing new plugins to support new functionality in **Terraform**.
- Setting up the build and deployment automation for Terraform scripts using **Jenkins**.
- Created and managed User/Group Accounts attached policies to User/Group Accounts using **AWS IAM** service.
- Worked on Google Cloud Platform(**GCP**) services like compute engine, cloud load balancing, cloud storage, and cloud deployment.
- Setup **GCP** Firewall rules to allow or deny traffic to and from the VM's instances based on specific configuration and used **GCP** cloud CDN(Content Delivery Network) to deliver content from **GCP** cache locations.
- Setup alerting and monitoring using Stackdriver in GCP.
- Hands on experience in **GCP** services like **EC2, S3, ELB, RDS, SQS, EBS, VPC, AMI, SNS, CloudWatch, CloudTrail, Cloud Formation GCP Config, Autoscaling, CloudFront, IAM, R53**.
- Setup databases in **GCP** using **RDS**, storage using **S3** bucket and configuring instance backups to S3 buckets.
- Configured virtualization environment to spin up multiple computer resources on the same hardware with high performance computing.
- Implemented a Continuous Delivery framework using **Jenkins** and **Maven** in a Linux environment.
- Worked to setup **Jenkins** as a service inside the **Docker swarm** cluster to reduce the failover downtime to minutes, and to automate the Docker containers deployment without using the configuration management tool
- Created **Docker** images using a **Docker** file, worked on **Docker** container snapshots, removing images and managing **Docker** volumes also deployed Docker Swarm using **Ansible**
- Designing and implementing for fully automated server build management, monitoring and deployment by using DevOps Technologies like **Ansible** and **chef**.
- Worked on **Ansible** to spin up a Jenkins server with all the dependencies.
- Wrote **Ansible** playbooks with **Python SSH** as the wrapper to manage configurations on Openstack Nodes and test playbooks on AWS instances using Python.
- Integration Configuration Management Systems like Ansible with **AWS** for code deployment.
- Designed a rapid deployment method using **Ansible** to auto deploy servers as needed. Experienced on ansible server build from scratch and writing **Ansible** playbooks to make configuration changes.
- Created and maintained built wrapper scripts using **PERL** and **Unix Shell** script
- Proficient in python, Shell scripting, SQL, build utilities like ANT and Cruise Control.
- Experience working with chef data bags, attributes, cookbooks, recipes and templates in chef. And created cron jobs for chef clients to interact with chef servers on a time period basis.
- Used **CloudWatch** and **LogicMonitor** to monitor resources such as **EC2, EBS, ELB, RDS and S3**.
- Implemented a CI/CD framework using Jenkins, Maven & Artifactory in Linux environment and Integration of **Maven/Nexus, Jenkins, Git and JIRA**.

Environments: Maven, Ant, Jenkins, AWS Cloud, Chef, Ansible, Shell, Python, JIRA, Service now, Web logic server, VMware, Apache tomcat, Load balancer, Route 53, VPC, Elastic Beanstalk, Docker, Kubernetes, Apache, Nginx, MYSQL and MONGO DB Databases.

Client: Eversana, Cincinnati, OH August'17 to July'18

Role: AWS/Azure Cloud Engineer

- Created recommendations on how to duplicate a subset of on-premise machines to the Azure Infrastructure asService(IaaS) offering which will be used for disaster recovery
- Implemented **CI/CD** pipeline with **Docker, Jenkins, GitHub** and **Azure Container Service**, whenever a new **TFS/GitHub** branch gets started, Jenkins, our Continuous Integration(CI) server, automatically attempts to build a new container from it.
- Worked on Amazon Web Services(AWS) for a multitude of applications utilizing the **AWS stack(including EC2, VPC, Glacier, Route53, S3, RDS, CloudWatch, Cloud Trial**.
- Used the **MySQL** database on EC2 and get the data Backup using Linux scripting language on EC2.
- Wrote python scripts using the Boto3 library to automatically sign up the instances in AWS EC2.
- Used **AWS Bean Stalk** for deploying web applications and services with python on Apache server
- Wrote python scripts to deploy java applications from Jenkins to remote server.
- Worked with **Terraform** templates to automate the Azure IaaS virtual machines using terraform modules and deployed virtual machine scale sets in production environments.

- Written **Templates** for **Azure Infrastructure** as code using Terraform to build staging and production environments.
- Worked on OpenShift for container orchestration with **Kubernetes** container storage, automation to enhance container platform multi-tenancy.
- Worked with **Kubernetes** architecture and designed troubleshooting issues and multi-regional deployment models and patterns for large-scale applications.
- Deployed windows **kubernetes(K8s)** cluster with **Azure Container Services(ACS)** from **Azure CLI** and utilized Kubernetes, Docker for the runtime environment of the CI/CD system to build, test and deploy.
- Used Ansible to setup/teardown of **ELK stack(Elastic Search, Log stash, Kibana)** and troubleshoot the build issues with **ELK** and work towards the solution.
- Managed Kubernetes charts using Helm, created reproducible builds of the **Kubernetes** applications, managed Kubernetes manifest files and managed releases of **Helm** packages.
- Designed, wrote and maintained systems in **python** scripting for administering **GIT**, by using Jenkins as a full cycle continuous delivery tool involving package creation, distribution and deployment into Tomcat application servers via shell scripts embedded into Jenkins jobs.
- Administered several application servers running JVM microservices.
- Involved in **Jira** as defect tracking system and configure various workflows.
- Worked with **Nagios** for Azure Active Directory and data consolidation for LDAP users. Monitored system performance using **Nagios**, maintained **Nagios** servers.
- Leveraged Azure Automation and **PowerShell** and **Chef** to automate processes in the Azure Cloud.
- Moderate and contribute to the support forms (specific to **Azure Networking**, **Azure Virtual Machines**, **Azure Active Directory**, **Azure Storage**) for Microsoft Developers Network using Partners and MVPs.
- Designed Network Security Groups (NSGs) to control inbound and outbound access to **VMs and subnets**
- Experienced in deploying infrastructure as code applications using **ARM** templates(JSON)
- Settled up **elastic search** clusters on **Microsoft Azure** using **Terraform** scripts primarily.
- Developed and maintained **Perl/Shell** scripts for build and release tasks.

Environments: BitBucket, Jenkins, Python, Ruby, Pearl, Powershell, Chef, Docker, Terraform, Nagios, Azure Site recovery, Kubernetes, XML

Client: Accenture, India Jun'15 to July'17

Role: Python Developer

- Built database model, Views and API's using **Python** for interactive web based solutions.
- Placed data into **JSON** files using **Python** to test Django websites. Python scripts performed ETL.
- Developed the entire frontend and backend using Python Django and used **Mongo DB** for databases.
- Developed modules using **Python** on **Django** and **Flask Framework** by implementing **MVC** architecture.
- Developed web based application using **Django** framework with python concepts.
- Used **Python** and **Django** creating graphics.
- Implemented the application using Python Spring IOC(Inversion of Control), **Django** Framework.
- Implemented admin sites to manage database information with FlaskAdmin, MySQL.
- Used **REST** API's, for developing web-based application for insurance domain, using Django's framework.
- Correspondingly involved in writing **REST APIs** using **Django** MVC Framework for data exchange.
- Involved in developing RESTful service using Python Flask Framework.
- Implemented **SOAP/RESTful** web services in **JSON** format.
- Experienced in consuming web services (REST) with python programming language.
- Recorded the online user's data using **Python** Django forms and implemented a test case using Pytest.
- Good experience on various **Python** integrated IDE's such as Sublime, Pycharm, Eclipse and Netbeans.
- Written scripts in **python** to automate log rotation of multiple logs from web servers.
- Written **python** scripts to create test cases during two week sprints using **Agile** methodology.
- Created Unit test/regression test framework for working/new code.
- Installed and maintained web servers **Tomcat** and **Apache** HTTP in Unix.

- Maintained site reliability on over multiple servers. Converted existing PHP scripts to **python**.
- Written python scripts to create test cases during two week sprints using **Agile** methodology.
- Developed and designed automation framework using **Python** and **Shell scripting**.
- Heavily leveraged Python's graphics APIs for encoding data in **XML/JSON** formats.
- Developed and designed automation framework using **Python** and **Shell scripting**.
- Designed and implemented custom Bash scripts for controlling the data flow from the **PostgreSQL DB**.
- Responsible for debugging the project monitored on **JIRA** using Agile methodology.
- Executed code reviewing using **GitHub** pull requests, improved code quality, and conducted meetings.

Environments: Python, Django, Git, GitHub, AWS, Linux, Shell Scripting, Apache HTTP, Tomcat, JIRA

Client: VSoft Technologies Pvt Ltd., Aug'13 to May'15

Role: Build and Release Engineer(Linux Administrator)

- Responsible for installation, configuration, and administration of **Red Hat Linux**.
- Setup and maintained **TCP/IP** network, configured the systems for **LAN**, setting up **SSH** and **SCP**.
- Integrated the Linux environment with Active Directory providing a Single Sign On (SSO) solution.
- Deployed **Redhat Satellite Server** to provide a stable management solution for the Linux environment.
- Installed, maintained and tuned the **Apache-Tomcat** server and **Websphere Application** Server in Linux.
- Created **Bash/Korn Shell** scripts to monitor system resources and system maintenance.
- Expertise in writing **Shell scripting** using **Ksh/Csh/bash** and **Perl** to ease administration tasks.
- Experienced in writing **Shell scripts** using **bash, python** for group/user administration, startup/shutdown.
- Experienced in writing **shell scripts** to automate the administrative tasks and management using cron.
- Implemented admin sites to manage database information with **Flask Admin, MySQL**.
- Developed **python** based API(RESTful Web Service) to track the events and perform analysis using Flask.
- Wrote **Perl** and **Python** scripts to generate and monitor processes.
- Used **bash** and **Python** to manage the environments.
- Involved in running **LAMP**(Linux, Apache, MySQL and Python) systems in cloud environment.
- Provided server administration, **shell scripts, Linux OS** installation and configurations, utilizing **LAMP**.
- Deployed applications (EAR/WAR) files on multiple servers/clusters and maintained Load balancing.
- Experienced in different tools for monitoring the health checkups using **Nagios**.
- Responsible for setting cronjob on the servers.
- Developed, maintained, updated various **Unix** shell scripts for services (start, stop, restart, cron jobs).

Environment: Redhat Linux, Oracle, Apache Web Server, Oracle , DNS, Shell Script, VMWare servers

EDUCATION

University of Cincinnati, Ohio - Master of Science (M.S), Dec 2019