

# SAHITHI P

716-934-8438 | sahithip82@gmail.com

Linked In : [profile url](#)



## PROFESSIONAL SUMMARY

*AWS and Microsoft certified professional with 8+ years of experience in implementing **DevOps** practices across organisation using **source code management tools** such as **GitLab, Bitbucket, git**, and implemented **CI/CD lifecycle** adaptations by using tools such as **Jenkins, Bamboo, Hudson, Azure Devops** and configuration management tools such as **Ansible, Chef, Puppet**, and **Kubernetes** as Orchestration tool for managing containerized **applications**, deployed applications in **AWS, Azure** cloud platforms and implemented **IaaS** using **Terraform, Cloud Formation** and on-premise virtualization using **VMware, Cloud Foundry**.*

- Proficient in working with **AWS services** like **Lambda, Dynamo DB, EC2, S3, VPC, ELB, AutoScaling Groups, Route53, IAM, Cloud Trail, OpsWorks, Cloud Watch, KMS, Kinesis, EMR, Redshift, Cloud Formation, SNS, RDS**, and also designed **AWS Cloud Formation** templates to create custom-sized **VPC, subnets, NAT** to ensure successful deployment of **Web applications** and **database templates**.
- Proficient in architecting secure **VPC Solutions** in **AWS** with the help of **Network ACLs, Security groups, NAT implementations**, public and private network configurations and also maintained **IAM policies** for organizations in **AWS** to define **groups, create users, assign roles**, and define **rules** for role-based access to **AWS** resources
- Implemented various services like **AWS Device Farm, Fargate, SES, Time Stream, Aurora, RDS**, etc and integrated with the existing tools and authentication methods.
- Implemented **lambda** function using **python**, configured **SNS (Simple Notification Service)** to trigger a **lambda function** when there is any new bucket or objects created in the **S3 bucket**
- Expertise in **Microsoft Azure IAAS, SAAS, and PAAS** such as **Azure storage, Azure active directory, Azure resource management, Azure VM, Blob Storage, SQL Database, Azure Data Factory, Resource Groups, Express Route, Traffic Manager, VPN, Load Balancing, Application Gateways, Auto-Scaling, Azure Web apps, Azure Application Insights**.
- Expertise in writing **Terraform** modules for **azure** services and deployed using **Azure DevOps** pipelines. Deployed services like **App Service Plan, App Service, and App Insights**. Expertise in the creation of **CI/CD** pipelines using **Azure DevOps** and in integrating **SonarQube** code analysis, code coverage in **CI/CD** pipelines
- Used **Azure Kubernetes Service** to deploy a **Azure managed Kubernetes cluster** and also used template driven deployment options such as **Resource Manager templates and Terraform**.
- Expertise in writing **Docker files** and scanned those images for the vulnerability, stored those images in **ACR**. To orchestrate those containers, I used **Kubernetes** where I am responsible to deploy **Kubernetes Cluster** on **cloud/on-premises** environments with **master/minion** architecture and used **YAML** files to create many services like **Pods, deployments, services, auto-scaling, load balancers, labels, health checks, Namespaces, Config Map**.
- Experienced with container-based deployments using **Docker**, worked extensively with **Docker files, Docker images, DockerHub, and Docker registries**, and also proficient in with **Nagios and Splunk, Elastic Search, Log stash, Kibana, Data dog**
- Implemented fully **automated build (CI)** and **deployment (CD)** infrastructure and processes for multiple projects using **Docker, Jenkins, sonar cube, VMware, RHEL, Kubernetes, Artifactory, Ansible & GitLab**.
- Strong experience on **DevOps** essential tools like **Chef, Puppet, Ansible, Docker, Kubernetes, GIT, Hudson, Jenkins, Ant, Maven**.
- Used **Terraform** to set up infrastructure in **Azure** Environments. Converted existing **Terraform** modules that had version conflicts to utilize **arm** templates during **Terraform** deployments to enable more control or missing capabilities.

- Created clusters using **terraform** with enabling autoscaling of **Kubernetes** cluster and also created Helm charts to deploy multiple **micro services** in the environment.
- Expert in using **Ansible** and **Ansible Tower** as a Configuration Management tool, to automate repetitive tasks, quickly deploys critical applications, and proactively managed change by writing **Python** code by using **Ansible Python API** to automate **Cloud Deployment** Process.
- Experienced in **DevSecOps** security auditing technique, continuous monitoring of the entire system, facilitating improved enforcement, and removing security breaches and vulnerabilities.
- Experienced in working with Source code management client tools like **GitBash, Github, Git GUI, GitLab, Visual GIT, BIT BUCKET** etc.
- Installed and configured an automated tool Puppet that included the installation and configuration of the **Puppet master, agent nodes,** and an **admin control workstation.**
- Installed **Chef-Server** Enterprise On-Premise/WorkStation/ Bootstrapped the Nodes, used Knife for chef-server connectivity. Managed nodes, **Run-Lists, roles, environments, Data Bags, cookbooks, recipes** in **Chef.** Worked with **CHEF CD** pipeline, workflow, and delivery truck.
- Monitored resources and Applications using **AWS Cloud Watch**, including creating alarms to monitor metrics such as **EBS, EC2, ELB, RDS, S3, SNS,** and configured notifications for the alarms generated based on events defined. Created and managed Kinesis streams, configured logs in Splunk from AWS cloud watch

### TECHNICAL SKILLS SUMMARY:

<b>Operating system</b>	Linux (Red Hat 4/5/6/7), Ubuntu 14/16/18, Windows NT 2000/2003, CentOS 6, 7
<b>Application Servers</b>	Oracle Web Logic (8/9/10), IBM Web Sphere (5/6/7), JBOSS
<b>Automation Tools</b>	Jenkins, Jfrog, Nexus, Docker, Kubernetes, Ansible, Chef, habitat, compliance, Kitchen, Puppet, Vagrant, Terraform, Cloud Foundry
<b>Version control tools</b>	GIT, GITHUB, SVN, BitBucket, GitLab, Gerrit, IBM Rational Clearcase, CodeCommit
<b>Web servers</b>	Apache HTTP 2.x ,3.x, Nginx, IIS, Apache Tomcat
<b>Networking/protocol</b>	DNS, DHCP, CISCO ASA Firewalls Routers/Switches, WAN, TCP/IP, NIS, NFS, SMTP, LAN, FTP
<b>Scripting</b>	Perl, Python, Ruby, Bash shell, Power shell scripting, Groovy, PHP, Hashicorp, JSON
<b>Virtualization Technologies</b>	VMWare ESX/ESXi, Windows Hyper-V, Power VM, Virtual box, Citrix Xen
<b>Cloud Environments</b>	Open stack, AWS, Azure, OpsWorks
<b>Volume manager</b>	VERITAS volume manager, LVM with Linux, Solaris Volume Manager
<b>Database</b>	AWS RDS, Aurora, DynamoDB, MongoDB, MySQL, MS SQL SERVER 2008/12
<b>Monitoring Tools</b>	Nagios XI, Wavefront, AppDynamics, Splunk, SolarWinds, Cloud Watch, ELK, NewRelic

### CERTIFICATIONS:

- **AWS Solution Architect Associate**
- **MICROSOFT CERTIFIED: Azure Administrator Associate**
- **MICROSOFT CERTIFIED: Designing And Implementing Microsoft Devops Solutions**
- **MICROSOFT CERTIFIED: DevOps Engineer Expert**
- **CKA: Certified Kubernetes Administrator**

## WORK EXPERIENCE:

**KAISER PERMANENTE, OAKLAND, CA**  
**SR. AWS CLOUD INFRASTRUCTURE ENGINEER**

**AUGUST 2019 TO PRESENT**

## RESPONSIBILITIES:

- Expertise in core **AWS services (S3, EC2, ELB, EBS, Route53, VPC, auto-scaling)** and deployment services (**OpsWorks and Beanstalks, Lambda**) and security practices (**IAM, KMS, SSM, Cloud watch and Cloud trail**) and services **Kinesis, SNS, SQS, EMR, Redshift, ECS, Elastic Bean Stalk, X-ray**.
- Implemented **Cloud Formation** for AWS orchestration and automated the workflow of AWS Cloud at every layer (such as **application, network, storage, and monitoring**, etc.) of AWS by writing cloud formation templates to create custom-sized **VPC, subnets**, NAT to ensure successful deployment of Web applications and database templates.
- Used **IAM** for creating **roles, users, groups** and implemented **MFA** (Multi-Factor Authentication) to provide additional security to AWS account and its resources. Design and Develop ETL Processes in **AWS Glue** to migrate Campaign data from external sources like S3, ORC/Parquet/Text Files into **AWS Redshift**.
- Managed cluster platform that simplifies running big data frameworks, such as **Apache Hadoop** and **Apache Spark**, on **AWS** to process and analyze vast amounts of data using **Elastic Map Reduce(EMR)**
- Hands-on experience in creating securing **VPC(Virtual Private Cloud), IP CIDR Blocks, Public Subnets, Availability Zones, Network ACL, NAT Gateway**, launched **EC2 instances** in the **public subnet**, and assigned **Elastic IP** to the instances. Built a **VPC**, established the site-to-site VPN connection between **Data Center** and **AWS**.
- 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.
- Experience in different migration services like **AWS Server Migration Service (SMS)** to migrate **on-premises** workloads to AWS in an easier and faster way using Rehost "lift and shift" methodology and **AWS Database Migration Service (DMS)**, **AWS Snowball** to transfer large amounts of data and **Amazon S3 Transfer Acceleration**.
- Created automated pipelines in **AWS Code Pipeline** to deploy **Docker** containers in **AWS ECS** using services like **CloudFormation, CodeBuild, Code Commit, CodeDeploy, S3, and puppet**.
- Built **S3** buckets and managed policies for S3 buckets and used **S3 bucket** and **Glacier** for backup on **AWS**, built migrations and integrations of **Kubernetes, Jenkins, Spinnaker, Istio, Prometheus, GitLab CI, and Nexus**
- Created **Kubernetes cluster** and worked on creating many **pods, replicationcontrollers, replicaset, services, deployments, labels, health checks**, and ingress by writing **Yaml** files
- Experienced in **code deployment, orchestration, and scheduling**. Used tools such as **Kubernetes, Docker Swarm, CoreOS Fleet**, and automation validation using **Test Kitchen, Vagrant, Ansible, and Terraform**.
- Wrote **Jenkins shared libraries** for CI and CD life cycles and created generic **pipelines** across the cloud applications to automate **application** deployment as well infrastructure **deployments** and introduced various security compliance rules to validate the organization standards over the release workflows.
- Extensively worked with tools like **Jenkins, Team City, and Anthill Pro** to implement build automation. Skilled in setting up **Jenkins Master** and multiple **slaves** for the entire team as a CI tool as part of the Continuous development and deployment process.
- Expertise in integrating **Terraform** with **Ansible, Packer** to create and version of the AWS Infrastructure, designing, automating, implementing, and sustainment of **Amazon machine images (AMI)** across the AWS Cloud environment.
- Automated delivery pipeline using integration and infrastructure tools **GIT, Jenkins, and Ansible** and implemented **Ansible** to manage all existing servers and **automate** the build/configuration of new servers and also developed an **Ansible role** for **Zabbix-agent** integrated into the **CICD** pipeline.

- Installed **Ansible Registry** for local upload and download of **Docker** images and even from **Docker Hub**. Used **Ansible to set up/teardown of ELK stack (Elastic Search, Log stash, Kibana)** and troubleshoot the build issues with **ELK** and worked towards the solution.
- Converted **DevOps** methodologies into **DevSecOps** model by introducing security/vulnerability scans different phases of DevOps like clear text password scan during the build and packaging phase of the application and on the another hand responsible for implementing pre-provisioning scans and post provisioning scans for **IaC** by adopting tools like **Terraform** enterprise (**sentinel**) and inspect framework for **AWS** cloud resource validations.
- Monitored the **containerized** applications in **AWS** using **Prometheus** where we just setup **Prometheus** docker on one machine and just deploy agents on the nodes and you are set for monitoring everything that happens on that node and used **Grafana** tool to collect metrics, queries, and logs.

**ERIE INSURANCE, ERIE, PA**  
**SR. AZURE CLOUD ENGINEER**  
**RESPONSIBILITIES:**

**JANUARY 2018 TO JULY 2019**

- Experienced in **designing, configuring, and deploying solutions on Microsoft Azure using (Azure Resource Manager) ARM Templates, AZURE Power Shell Module** and, **Azure CLI** focusing on **high-availability** and **auto-scaling**. Designed & implemented migration strategies for traditional systems on **Azure (Lift and shift/Azure Migrate, other third-party tools**
- Performed **POT's** in Azure implementation and **POC'S** to implement in the organisation and involved in migration from **AWS** to **Azure** cloud. Developed strategy to migrate various **applications** from an enterprise VMware infrastructure to **IAAS Amazon Web Services(AWS)**.
- Migrated the workloads from **Amazon Web Services (AWS) EC2** Cloud to Microsoft Azure Cloud, deployed the **Plate Spin Migrate** server in the target Azure environment. Migrated **Agent** to register workloads with the cloud-based Migrate server using secure communications over the public Internet.
- Post-migration enabled the **Azure Site Recovery(ASR)** for the **disaster recovery** purpose and to backup the virtual machines we used **Azure Backups** and created replicas in the Azure to prevent the downtime of applications if any region or availability zone goes down.
- Hands-on experience in Azure Development worked on **Azure web application, App services, Azure storage, Azure SQL Database, Virtual machines, Fabric controller, Azure AD, Azure search, and notification hub**.
- Worked on **Azure Data bricks** cloud to organizing the data into **notebooks** and making it easy to visualize data through the use of **dashboards**. Deployed an **Azure Data bricks** workspace to an existing virtual network that has **public** and **private** subnets and properly configured **network security groups**.
- Created **Azure Event Grid** to **publish** and **consume** data, used **Azure Logic apps** to implement workflows, and **Azure Key Vault** to maintain secrets in Azure environment. Implemented **Code reviews** and **queries** performance tuning.
- Provided quick solution to Azure teams for self-servicing data maintenance operations, which uses SQL Master Data Services. Configured and setup **SQL Server Master Data Services (MDS)** for self-serving data maintenance across different teams.
- Created **Azure Automation Assets, Graphical runbook, Power Shell** runbook that will automate specific tasks, deployed **Azure AD Connect**, configuring **Active Directory Federation Service (ADFS)** authentication flow, ADFS installation using Azure AD Connect, and involved in administrative tasks that include **Build, Design, Deploy of Azure environment**.
- Used **Azure DevOps** to manage **test plans, suites, and cases**. Installed the **Test and Feedback Extension** in the Azure DevOps Marketplace to the author, run and analyze the manual tests.
- The **data lake** is built to ingest historical data (PageView, TagDetected & EmailsOpens) and HealthCare Professional profile data using Sqoop. Future usecases being designed to ingest transactional data from AIM application via **Kinesis Firehose**.

- Skilled in **Kubernetes** to deploy **scale, load balance**, and manage **Docker** containers with multiple namespaced versions and a good understanding of **Open Shift Platform** in managing **Docker Containers** and **Kubernetes Clusters**. Worked on **developing APIs** using **Kubernetes** to manage and specify the copies of the containers to run the actual servers in the cloud environment.
- Used **Jenkins** pipelines to drive all micro services build out to the Docker registry and then deployed to **Kubernetes**. Created pods and managed them by using **Kubernetes**.
- Integrated **Terraform** with **Ansible** and automated **Terraform** life cycles with **Ansible playbooks**. Conducted POC to identify the suitable telemetry tool for **containers, service mesh** tools like **Linkerd**.

**CAPITAL ONE, RICHMOND, VA**

**AWS CLOUD ADMINISTRATOR**

**JUNE 2016 TO DEC 2017**

**RESPONSIBILITIES:**

- Automated AWS infrastructure by writing various **Terraform modules**, null resource scripts to create **AWS IAM users, groups, roles, policies, custom policies, AWS Glue, Crawlers, Redshift clusters, snapshots of clusters, EC2, S3 buckets, application deployments**.
- Transformed and moved large amounts of data into and out of other **AWS data stores (RDS)** and **databases**, such as **Simple Storage Service (Amazon S3)** and **Amazon Dynamo DB** using **Elastic Map Reduce**. Implemented **AWS Lambda functions** to run scripts in response to the event in **Amazon Dynamo DB table, S3 buckets, and HTTP requests** using **Amazon API Gateway**.
- Worked on **AWS Cloud Watch, Cloud Formation, Cloud Trail, and Cloud Front** to set up and manage cached content delivery and automated the **spin-up APIs** and database servers in each layer of the **OpsWorks Stack** in AWS cloud using **boto3** python scripting.
- Built servers using **AWS**, importing volumes, launching **EC2, RDS**, creating **security groups, auto-scaling, load balancers (ELBs)** in the defined **virtual private connection (VPC)** by providing high availability of applications of EC2 instances by taking the advantages of AWS High availability options and configured **S3 versioning** and lifecycle policies in AWS, to backup files and archive files in **Glacier**
- Implemented **AWS** web application firewalls to monitor the web requests that are forwarded to **API Gateways** and Application Load balancers to enable the security model.
- Authored **Terraform modules** for infrastructure management and published a module to the **Terraform registry** for enterprise customers to deploy our product inside their **AWS environment** and set up build and deployment automation for **Terraform scripts** using **Jenkins**.
- Implemented a **production-ready, load-balanced, highly available, fault-tolerant Kubernetes infrastructure**. Used Jenkins pipelines to drive all **microservices** builds out to the **Docker** registry and then deployed to **Kubernetes**, created **Pods**, and managed using **Kubernetes**.
- Experienced in implementing **Docker** for wrapping up the final code and setting up development and testing environment using **Docker Hub**, and **Docker Container Network**.
- Worked on creation of **Ansible manifest** files to install **tomcat instances** and to manage configuration files for multiple applications and responsible for automated identification of **application server** and **database server** using **Ansible Scripts**.
- Monitored **Linux servers** round the clock and identified, found the system issues, and gave the optimal solution. Updated kernel & security patches in **Amazon Linux environment**, handling out-of-memory issues in **Linux kernels** during rebalance in **Kafka** cluster.
- Experienced in **System Administration** on various platforms which include **Red hat Linux, Solaris, SUSE Linux, CentOS, Windows Environment, VMware ESX/ESXi, and VMware vSphere**.
- Wrote **AWS Lambda functions** to trigger the **events** related to the migration of data from **AWS S3** to **AWS Redshift**.

- Worked with **AWS IAM** to securely control users' access to AWS services and resources by enforcing granular permissions, controlling share access to **AWS accounts**, applying **multi-factor authentication**, and enabling **identity federation**.
- Deployed and configured **Elastic search ELK, Logstash, and Kibana (ELK)** for **log analytics, full-text search, application monitoring in integration with AWS Lambda and Cloud Watch**.

**CONGRUENT SOFTWARE, SEATTLE, WA.**

**APRIL 2015 TO MAY 2016**

**DEVOPS ENGINEER**

**RESPONSIBILITIES:**

- Implemented **Cloud Formation** for AWS orchestration and automated the workflow of **AWS Cloud** at every layer (such as application, network, storage, and monitoring, etc.) of **AWS** by writing cloud formation templates.
- Successfully deployed applications onto EC2 Servers with **zero downtime** by utilizing features of AWS auto-scaling **group** with **launch templates** and made the application release life cycles **immutable** with AMI orchestration.
- Created load test framework using **Vegeta** and configured API's using **Docker** containers and **Cassandra** instances at **AWS ops works** layers' level
- Developed Docker images to support Development and Testing Teams and their pipelines, distributed **Jenkins, Elastic Search, Kibana and Log stash (ELK & EFK)**, etc, and also monitored APIs and **Cassandra's** performance using New Relic, Cloud Watch, and Log stash.
- Wrote **Docker** files to build Microservice applications and created **docker images** to reuse them to create the servers and created the lightweight deployment workflow by adopting container strategies.
- Worked on **Jenkins, Git Hub, Puppet, Artifactory**, and all internal build systems for the core development team on an enterprise-level Java-based cloud **orchestration/automation** tool.
- Provided backup support for **Jenkins** build environment and involved in **upgrade of Bamboo & Artifactory** Server and adopted **Jenkins** for CI practice and revised existing build procedures. Created multiple **Maven**, Shell scripts for build automation and deployment automation
- Developed build and deployment scripts using **ANT** and **MAVEN** as build tools in **Jenkins** to move from one environment to other environments for Continuous Integration.
- Developed **Chef Cookbooks** to install and configure **Apache, Tomcat, Splunk, Jenkins, WebLogic**, and deployment automation and Integrated **Jenkins, Rundeck, and Chef**.
- Configured and maintained **Puppet Master** server, also responsible for creating, and updating **Puppet** modules using manifests and pulling them from **Puppet Agents**.
- Installed and deployed **Red Hat Enterprise Linux 6.x/7.x, Centos** and installation of packages and patches for **Red Hat Linux Servers**. Built and managed a large deployment of **Ubuntu Linux instances** systems with **Chef Automation** and written recipes, tools, shell scripts, and monitoring checks

**BARCLAYS, WHIPPANY, NJ**

**FEB 2014 TO MARCH 2015**

**SITE RELIABILITY ENGINEER**

**RESPONSIBILITIES:**

- Implemented & maintained the branching and build/release strategies using **SCM** tools Subversion, **GIT**, and maintain build packages using **Team Foundation Server (TFS)**.
- Integrated **GIT** with Jenkins using the **GitHub** plugin to automate the process of source code check-out by providing the URL and credentials of the GIT repository
- Worked closely with Web Administrators to understand, participate, and experience the technical architect for websites such as **Apache, JBoss, WebSphere, WebLogic** and **deploying J2EE** Applications to these environments.
- Installing, upgrading, and configuring Linux Servers using **Kickstart** as well as manual installations and the recovery of the root password..

- Installed **Jenkins** on a **Linux** machine and created a **Master** and **Slave** configuration to implement multiple parallel builds. Integrated **GIT** into **Jenkins** to automate the code check-out process. Defining Release Process & Policy for projects early in **SDLC**.
- Researched and implemented an **Agile** workflow for continuous integration and testing of applications using **Jenkins**. Administered and Engineered **Jenkins** to manage weekly **Build**.
- Server automation with **Puppet** and used tools like **Maven** for deployment and build management.
- Involved in design, configuration, installation, implementation, management, maintenance, and support for the Corporate **Linux servers RHEL 4x, 5.x, CENTOS 5.x**.

**ACCENTURE, HYDERABAD, INDIA**

**JAN 2013 TO JAN 2014**

**LINUX SYSTEM ADMINISTRATOR**

**RESPONSIBILITIES:**

- Built **Yum** and **RPM** repository server for the Development team to upload packages. Maintained appropriate file system security and monitoring the status of the process to increase system efficiency. Performed backup/restore operations using tar, and gzip. Wrote UNIX Shell scripts for monitoring, and clean up old files.
- Implemented advanced disk configuration such as **RAID 5** using Linux Software **RAID** programs and also configured Red hat Linux servers for production, staging, and test environments running Database and Middleware applications.
- Involved in design, configuration, installation, implementation, management, maintenance, and support for the MMPLATES. Corporate **Linux servers RHEL 4.x, 5.x, SLES 9, CENTOS 5.x**, Solaris for business-critical with Disaster Recovery.
- Managed systems routine backup, **CRON** jobs, enabled system logging, network logging of servers for maintenance, performance tuning, testing in Linux environments.
- Installed, configured, and maintained Linux / UNIX operating systems and components. Diagnosed and resolved problems associated with **FTP, DNS, DHCP, VPN, NFS, and Apache**

**VALUE LABS INC. HYDERABAD, INDIA**

**JAN 2012 TO DEC 2012**

**PYTHON DEVELOPER**

**RESPONSIBILITIES:**

- Managed datasets using data frames and MySQL, queried MYSQL database queries from python using Python-MYSQL connector MySQL DB package to retrieve information. Development of real-time multi-tasking systems using Python. Created data access modules in python.
- Designed and developed components using Python with Django framework. Implemented code in python to retrieve and manipulate data. sed Python & Django creating graphics, XML processing of documents, data exchange & business logic implementation between servers.
- Utilize PyUnit, the Python Unit test framework, for all Python applications. Performed troubleshooting, fixed and deployed many Python bug fixes of the two main applications that were the main source of data for both customers and the internal customer service team
- Implemented and worked on the python code using shell scripting. Performed troubleshooting, fixed, and deployed many Python bug fixes. Developed web-based applications using Python, Django, XML, CSS, HTML, JavaScript, AngularJS, and jQuery and also worked on several python packages.