Sufiyan Ali

github.com/Sufiyanali07 linkedin.com/in/sufiyan-ali sufiyanali0727@gmail.com +919309908454

Education

Bachelors of Computer Science 2021-2024
(Dr babasaheb ambedkar marathwada university) *Percentage: 71.97*
MAHARASHTRA STATE BOARD(Miliya College Beed) June 2021
Percentage: 82.20

Coursework

Courses: AWS(Amazon Web Services), DevOps, Linux, Data Structures

Skills

Languages: Shell Scripting, C/C++, Python,
Services: IAM,Ec2,S3,Load balancer,Lambda,ASG,Vpc,RDS,SQS,SNS,Cloud formation & Cloud trail
Tools: Docker, Kubernetes, Jenkins, Terraform, Maven ,Ansible, Git/GitHub,Unix, Linux, VS Code

Projects

Employment Management System | Feb. 2024

- Good understanding of AWS services like EC2, S3, VPC, RDS, ALB, Cloudwatch, AutoScaling groups etc
- Create an EC2 instance and deployed applications on the same
- Create a full application with the static websiste hosted on S3 with application running on EC2 with application load balance in front of it. The application was connected to a RDS MySQL data. Able to perform CRUD operation on the database from the website
- Good understanding of VPC and created a VPC with public and private subnets with EC2, application load balancer with auto scaling group

VPC Peering | Mar. 2024

- create a VPC with 1 public and 1 private subnet using a CIDR range of 10.0.1.0/24
- create another VPC with only 1 private subnet using a CIDR range of 10.0.2.0/24
- create a VPC peering connection and accept the connection.
- create first ec2 instance in first vpc in public subnet
- create second ec2 instance in secondd vpc in private subnet.
- ssh into the public ec2 instance, copy the .pem file and try ssh into the private instance using ssh -i ec2key.pem ec2-user@¡private instance ip address¿
- now modify the route table to first vpc's public subnet and add route for the second vpc CIDR block with target as peering connection
- now modify the route table to first vpc's public subnet and add route for the second vpc CIDR block with target as peering connection

Autoscaling Group | Dec. 2023

- First create a launch template. In the LT, select amazon AMI, use the ec2 key pair, allow all traffic and enable detailed monitoring.
- Create a ASG with the launch template and once created, set the dynamic policy for CPU utilization.
- SSH to the ec2 machine and install the stress and run the stress command to increase load
- Check the CPU utilization of the machine and wait for 10 mins or so.
- Check the Auto scaling Activity tab and you will see that a new instance got created.
- Don't terminate the EC2 instances, but acutally go and delete the ASG to terminate the instances.

Genral

Language

Hindi as mother tongue, fluent English and Marathi