Experiment 6 Output

Aim: The experiment configures load balancing and auto-scaling using **AWS CLI, LocalStack, Docker,** and **Nginx**. It simulates cloud traffic distribution across instances for efficient workload management. The setup demonstrates scalable deployment in a local cloud environment.

Creating VPC

```
C:\Users\rawat>aws ec2 create-vpc --cidr-block 10.0.0.0/16 --endpoint-u
-------------------------------------------------------
|                      CreateVpc                      |
+-----------------------------------------------------+
||                         Vpc                       ||
|+----------------------+----------------------------+|
||  CidrBlock           |  10.0.0.0/16               ||
||  DhcpOptionsId       |  default                   ||
||  InstanceTenancy     |  default                   ||
||  OwnerId             |  000000000000              ||
||  State               |  pending                   ||
||  VpcId               |  vpc-66375b0cbe498b519     ||
|+----------------------+----------------------------+|
|||              CidrBlockAssociationSet              |||
||+----------------+-----------------------------------+||
|||  AssociationId |  vpc-cidr-assoc-0c82d2b7eaae6089a  |||
|||  CidrBlock     |  10.0.0.0/16                       |||
||+----------------+-----------------------------------+||
||||                CidrBlockState                     ||||
|||+------------------+-----------------------------+|||
||||  State           |  associated                 ||||
|||+------------------+-----------------------------+|||
```

Fig 1: VPC Setup

Pulling Nginx for Execution

```
943ea0f0c2e4: Pull complete
103f50cb3e9f: Pull complete
Digest: sha256:9d6b58feebd2dbd3c56ab5853333d627cc6e281011cfd6050fa4bcf2072c9496
Status: Downloaded newer image for nginx:latest
63d7914e78bb0791288824fffdcf46d77bd92b4f895de4a403c88012dc87b601

C:\Users\rawat\Documents\8 SEMESTER\Cloud Computing\Lab\Experiment 6\Codes>docker run -d
e565ff4cf0081c7965a7634fb1de6018e88d927dd80ca179822aa7a33994f87c

C:\Users\rawat\Documents\8 SEMESTER\Cloud Computing\Lab\Experiment 6\Codes>docker images
REPOSITORY                    TAG       IMAGE ID       CREATED        SIZE
hackvortex-backend            latest    14e63c26d40b   12 hours ago   1.05GB
postgres                      15        e45d3f5ec589   7 days ago     430MB
flask-app                     latest    269bf42596ed   3 weeks ago    126MB
nginx                         latest    b52e0b094bc0   4 weeks ago    192MB
<none>                        <none>    44b808030263   4 weeks ago    126MB
<none>                        <none>    063caad47b0a   4 weeks ago    126MB
localstack/localstack         latest    b686f3948f42   6 weeks ago    1.18GB
hello-world                   latest    74cc54e27dc4   6 weeks ago    10.1kB
public.ecr.aws/lambda/python  3.8       348b357f1c82   7 weeks ago    575MB
python                        3.9       9f98746e2033   3 months ago   999MB
```

Fig 2: Docker Pulling and Output Image

Running Flask Load balancing App

```
C:\Users\rawat\Documents\8 SEMESTER\Cloud Computing\Lab\Experiment
[+] Building 18.4s (9/9) FINISHED                                dock
 => [internal] load build definition from Dockerfile
 => => transferring dockerfile: 332B
 => [internal] load metadata for docker.io/library/python:3.9-slim
 => [internal] load .dockerignore
 => => transferring context: 2B
 => [1/4] FROM docker.io/library/python:3.9-slim
 => [internal] load build context
 => => transferring context: 1.35kB
 => CACHED [2/4] WORKDIR /app
 => [3/4] COPY . .
 => [4/4] RUN pip install --no-cache-dir flask requests
 => exporting to image
 => => exporting layers
 => => writing image sha256:6e4cbd50721df2af76ffcc7bcf3ae2550a3b500
 => => naming to docker.io/library/my-flask-app
```

Fig 3: Flask App

Output with graph showing processes

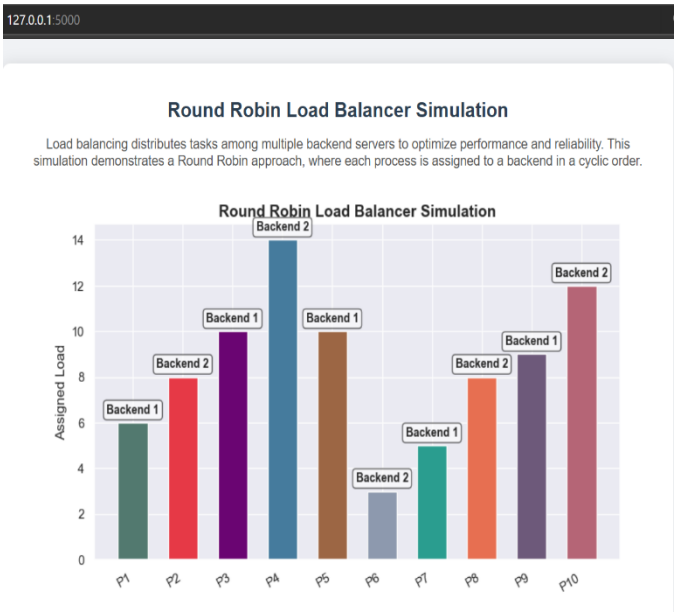

Fig 4: Flask App Output

Madhurima Rawat DS 42