**Exercise20**

**Building Elasticity**

**NOTE: Only major steps are shown here. You will need to apply the knowledge and skills that you have acquired.**

1. Create an S3 Bucket

Give a name, give public access, (unzip the Ex20-Resources file) upload the two folders, *az1a* and *az1b*, and the two files *health.html* and *index.html*, give required permissions so that you can view the files in your browser Take snapshots of all the messages from your browser for each of the four files (with your IAM username to confirm this step) [2points]

1. Create a VPC

Create two public subnets in two different AZs Take snapshots of each subnet’s details tab (with your IAM username to confirm they are created in two different AZs) [1point]

1. Launch two public EC2 instance**s**

**One in each subnet**, AMI: **Linux2 AMI**, SG: Allow HTTP and SSH traffic from anywhere, IAM instance profile: create an IAM instance policy and add the AmazonS3FullAccessPolicy for theEC2 Use case, User data: copy paste the text from the file ‘*Elastic Load Balancer Script for EC2’* while making necessary changes (for the first EC2 instance, use az1 folder and for the second, use az2 folder). Copy paste the IP address of each and open it in your browser.

Take snapshots of the networking tab of each of the two EC2 machines (with your IAM username to confirm each is launched in one of the above subnets) [2points]

Take snapshots of the messages from your browser for each of the two EC2 machines (with your IAM username to confirm this step) [2points]

1. Create an Application Load Balancer (and an associated target group)

SG: the same that you created/used as above, Heath Check Path: /health.html, Healthy threshold: 3, Unhealthy threshold: 2, Timeout: 5, Interval: 10. Open the DNS name in your browser. Take a snapshot to show the toggling messages on the browser (with your IAM username to confirm this step) [4points]

1. **Terminate the two EC2 instances**
2. Deploy Auto Scaling Solution

Click auto scaling group, create launch template, SG: the same that you created/used as above, AMI: Linux2 AMI, IAM Instance Policy: the one that you created above, Copy the content from the file ‘*Script for our Auto Scaling Service’*, make necessary changes, paste it in User data, launch the template

Select the launch template, select both the subnets, choose appropriate load balancer and target group, turn on Elastic Load Balancing health checks, Desired capacity: 2, Min desired capacity: 2, Max desired capacity: 2, Automatic scaling: No scaling policies, Add tags-> Key: Name, Value: give some tag names so that you can identify the EC2 machines,

1. Verify Auto Scaling is working
   * 1. Go to instances, you will see two new instances with the tag name Take a snapshot (with your IAM username to confirm this step) [4points]
     2. Go to Auto Scaling group, go to Activity tab, you will see two successful messages Take a snapshot (with your IAM username to confirm this step) [1points]
     3. Go to load balancer, open DNS link. You will see a welcome message. Take a snapshot (with your IAM username to confirm this step) [2points]
     4. Terminate one of the two newly created instances. You will see a new one is automatically launched. Go to Auto Scaling group, go to Activity tab, you will see corresponding messages. Take a snapshot (with your IAM username to confirm this step) [2points]
2. **Cleanup:**

Terminate Auto scaling group, launch template, load balancer, target group, S3 bucket, all EC2 instances

Sources: https://iaasacademy.com/