1   Flask deployment

Deployment on LINUX (EC2 instances)
On each instance, the flask app is deployed when the instance is in a running state. The python
application is created in the directory home/ubuntu/flask-server after installing a python virtual
environment and flask. The application runs on 0.0.0.0 at port 80. The bash script to be executed
in each instance is contained in a python file (app_cluster1.py and app_cluster2.py). For machines
in the cluster 1, the app will route the requests to the URL /cluster1, and for machines in cluster 2,
the app will route the requests to the URL /cluster2. A health check can be performed by sending
a request without a path, and the response returned will be

Healthy!

If the URL path contains either /cluster1 or /cluster2, the resonse returned will be

Instance {AWS_INSTANCE_ID} is responding now!

Where AWS_INSTANCE_ID is the ID of the instance that responded to the request.

2   Infrastructure setup

Instances
The instances created will be five M4.large instances (two 2.4 GHz Intel Xeon E5-2676 vCPUs, 8
GiB memory, moderate network performance and 450 Mbps) (Amazon Web Services, n.d.) and five
T2.large instances (two Intel vCPUs, 8 GiB of memory and low to moderate network performance)
(Amazon Web Services, n.d.). However, due to account restrictions, we can only deploy a total
of nine instances, so the second group will only have four. Each instance will allow SSH, HTTP
and HTTPS traffic from 0.0.0.0/0, meaning the whole internet. The operating system is ubuntu
and the image used is ami-08c40ec9ead489470. The availability zone will be us-east-1. Flask will
be deployed on each instance at IP address 0.0.0.0 port 80. The storage is EBS-Only for both
instances. (Amazon Web Services, n.d.)

Table 1: Specs of M4.large and T2.large

Type	vCPU	Memory (GiB)	EBS Bandwidth (Mbps)	Network performance
M4.large	2	8	450	Moderate
T2.large	2	8	-	Low to Moderate

Target groups:
The first target group will include all five M4.large instances, while the second one will be four T2.large instances. The targets will receive traffic on port 80. IPv4 will be used with the HTTP protocol.

Elastic load balancer:
The elastic load balancer will be an internet facing application load balancer (requests can be routed based on their contents such as the URL path). IPv4 is used. Its security group allows it to send and receive traffic from anywhere on the internet (TCP to and from 0.0.0.0/0). Two subnet IDs are specified (us-east-1a and us-east-1b).

Listener
The listener listens on port 80 and will send requests to one of the two target groups that are attached to it. The protocol is HTTP, and both target groups are given equal weight. The URL path for the first target group is /cluster1 and the URL path for the second target group is /cluster2. The paths are specified in the value

```
Conditions=[
    {
        'Field': 'path-pattern',
        'PathPatternConfig': {
            'Values': [
                path
            ]
        },
    },
],
```
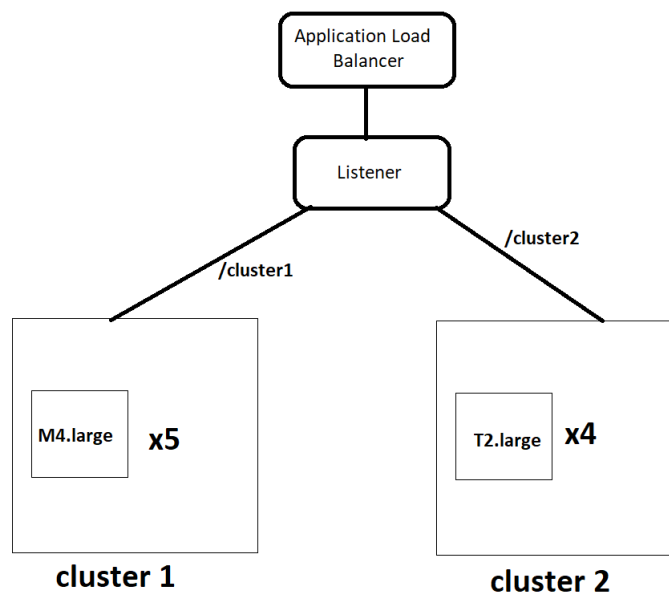
The diagram of the setup is as follows:

Figure 1. Application load balancer with 2 clusters and a forward listener

3   References

Amazon Web Services. (n.d.). *Amazon EC2 Instance Types*. Amazon Web Services. Retrieved October 17, 2022, from https://aws.amazon.com/ec2/instance-types/