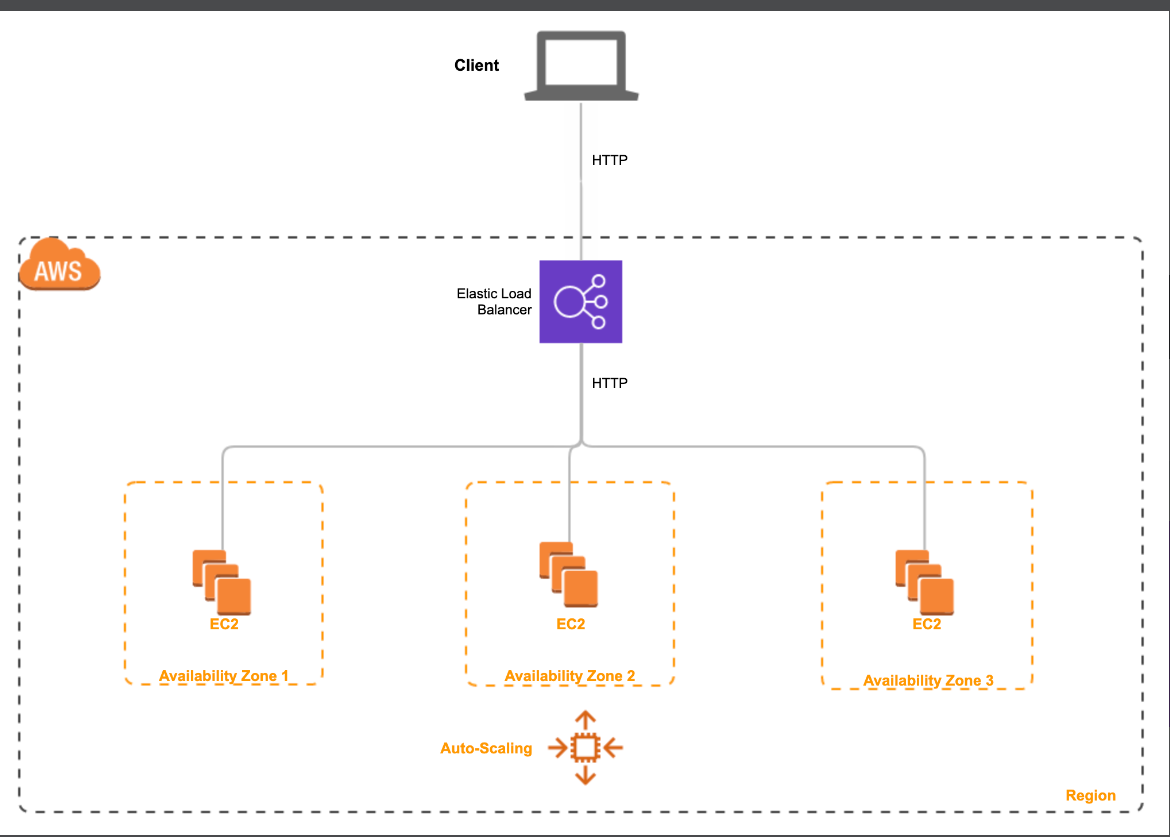
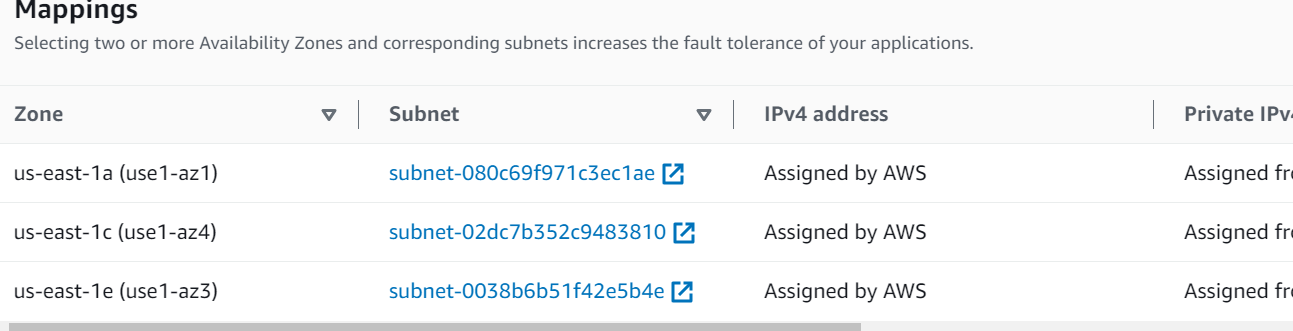
****

In this lab, you will be an engineer at a company that has recently launched a video-sharing social networking service. The application server keeps getting overloaded at peak usage times. The engineering team needs to implement an Auto Scaling group (ASG) running behind an Application Load Balancer (ALB) to horizontally scale an adequate number of servers to respond to the varying workload.

**Create an Application Load Balancer (ALB)**

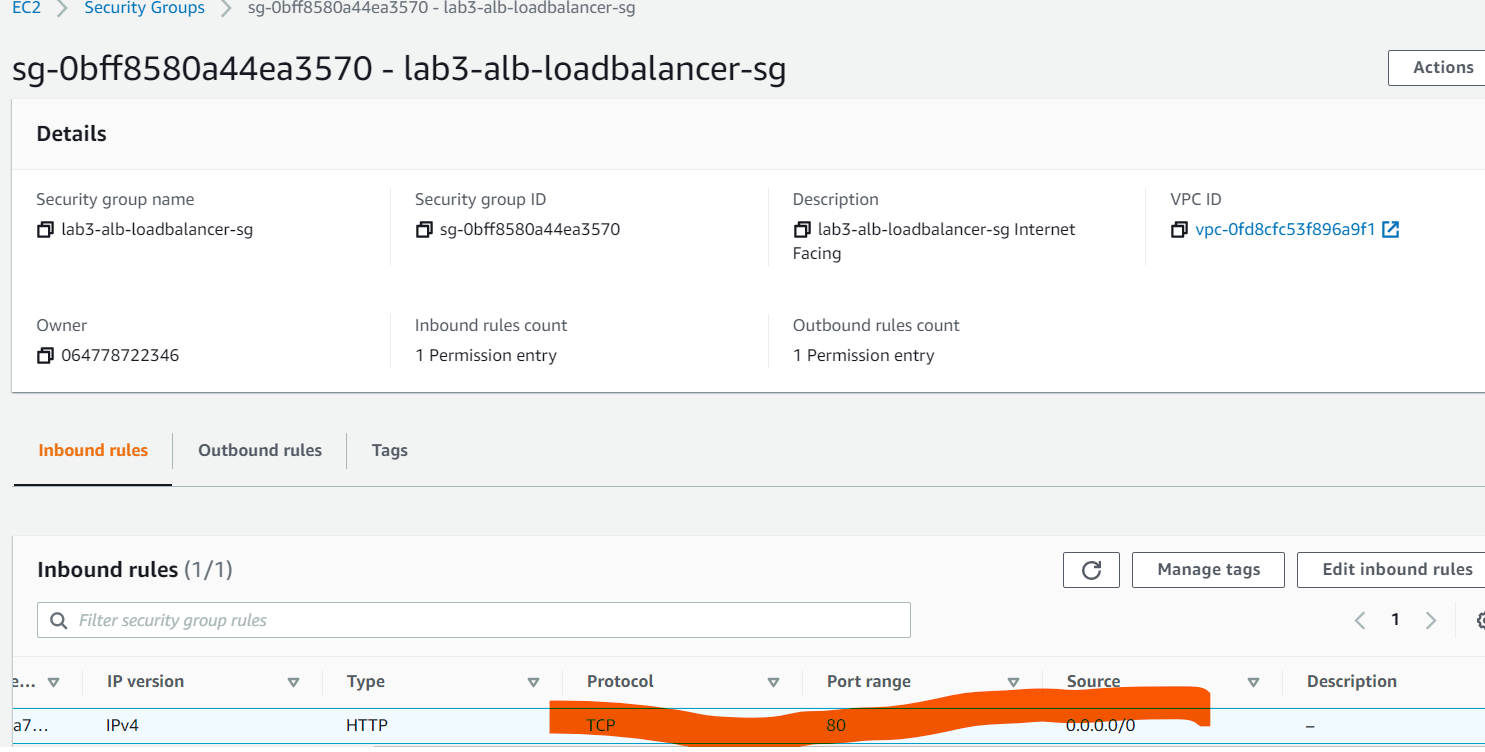
You will create an internet-facing Application Load Balancer (ALB) that handles incoming HTTP web traffic on port 80. The ALB will be configured in the default VPC with 3 Availability Zones enabled for your ALB so that it can distribute the traffic to the targets with cross-zone load balancing.

* Create ALB - lab3-alb-loadbalancer



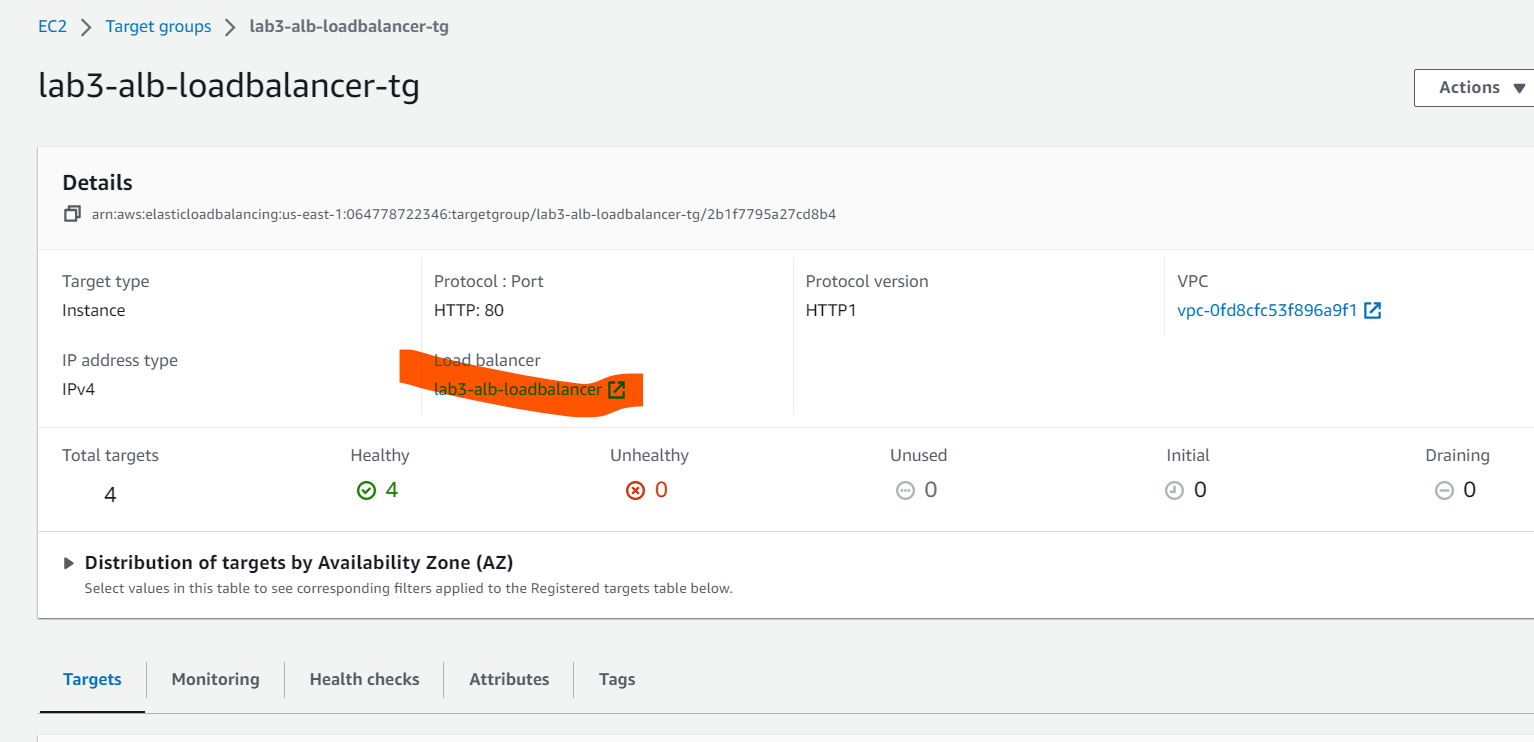
* Assign Security Group –

# **lab3-alb-loadbalancer-sg**



* Assign Target Group

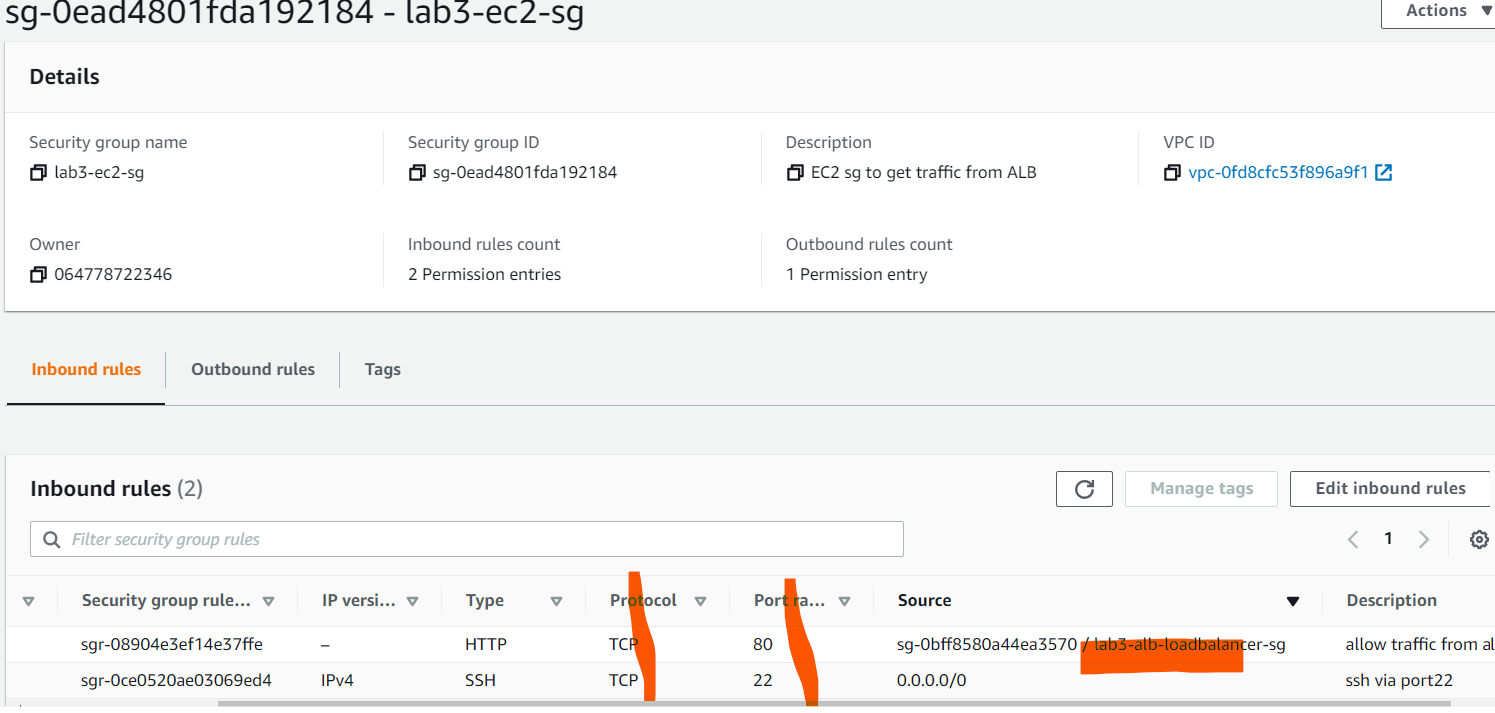
lab3-alb-loadbalancer-tg



### Create Launch Template for the EC2 instances

* Create SG for instance -> Allow inbound rule for HTTP port 80 for assign SG of ALB and SSH Port 80

# **lab3-ec2-sg**



* Launch Template assign the SG create above - lab3-ec2-alb-autoscale-template

#!/bin/bash

yum update -y

yum install httpd -y

systemctl start httpd

systemctl enable httpd

cd /var/www/html

echo "Hello World from instance $(wget -q -O- http://169.254.169.254/latest/meta-data/instance-id)" > index.html

### Configure Auto Scaling groups to leverage the Launch Templates to launch EC2 instances and use the ALB to distribute the traffic to the provisioned EC2 instances

Run the following commands to install the **stress** utility on the EC2 instance

sudo amazon-linux-extras install epel -y

### sudo yum install stress -y

Invoke the stress test on the instance using the following command:

stress --cpu 8 --timeout 300

