Load balancer

Load balancing in the context of AWS involves distributing incoming application traffic across multiple resources to enhance availability and prevent overloading. AWS provides various types of load balancers, each serving specific use cases:

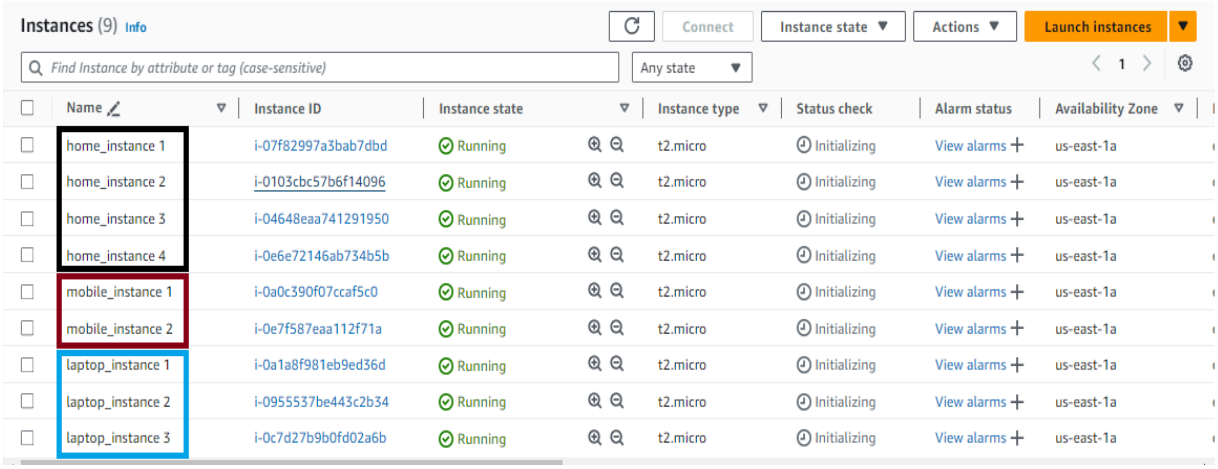
Application Load Balancer (ALB): Operates at the application layer and is designed for routing HTTP/HTTPS traffic. It is suitable for modern application architectures, providing advanced routing capabilities and supporting content-based routing.

Network Load Balancer (NLB): Works at the transport layer and is optimized for handling TCP, UDP, and TLS traffic. NLB provides high-throughput and low-latency performance, making it ideal for scenarios that require scalability and efficiency.

Classic Load Balancer (CLB): The legacy load balancer that distributes traffic across EC2 instances based on either the application or network layer. While still available, it is recommended to use ALB or NLB for more advanced features.

Gateway Load Balancer (GWLB): Provides private connectivity between virtual appliances in a service provider VPC and application resources. It facilitates secure communication and is used in scenarios where advanced networking and security services are required.

1. Create **multiple instances** for distributing the traffic.
   Home page instance = 4
   Laptop page instance = 3
   Mobile page instances = 2



Instances (9) Info

Connect | Instance state | Actions | Launch instances

Find Instance by attribute or tag (case-sensitive) | Any state | 1

Name	Instance ID	Instance state	Instance type	Status check	Alarm status	Availability Zone
home_instance 1	i-07f82997a3bab7dbd	Running	t2.micro	Initializing	View alarms	us-east-1a
home_instance 2	i-0103cbc57b6f14096	Running	t2.micro	Initializing	View alarms	us-east-1a
home_instance 3	i-04648eaa741291950	Running	t2.micro	Initializing	View alarms	us-east-1a
home_instance 4	i-0e6e72146ab734b5b	Running	t2.micro	Initializing	View alarms	us-east-1a
mobile_instance 1	i-0a0c390f07ccaf5c0	Running	t2.micro	Initializing	View alarms	us-east-1a
mobile_instance 2	i-0e7f587eaa112f71a	Running	t2.micro	Initializing	View alarms	us-east-1a
laptop_instance 1	i-0a1a8f981eb9ed36d	Running	t2.micro	Initializing	View alarms	us-east-1a
laptop_instance 2	i-0955537be443c2b34	Running	t2.micro	Initializing	View alarms	us-east-1a
laptop_instance 3	i-0c7d27b9b0fd02a6b	Running	t2.micro	Initializing	View alarms	us-east-1a

2. Add user data a per your requirement while creating the instance…..

   #For Home page

```
#!/bin/bash

yum install httpd -y
systemctl start httpd
systemctl enable httpd
echo "this is home page $HOSTNAME" > /var/www/html/index.html
```

#For laptop page

```
#!/bin/bash

yum install httpd -y
systemctl  start  httpd
systemctl  enable httpd
mkdir /var/www/html/laptop
echo "this is laptop page $HOSTNAME"  > /var/www/html/laptop/index.html
```

#For mobile page

```
#!/bin/bash

yum install httpd -y
systemctl  start  httpd
systemctl  enable httpd
mkdir /var/www/html/mobile
echo "this is mobile page $HOSTNAME"  > /var/www/html/mobile/index.html
```

3. After successfully creating multiple instances add this instances **in target group.**

   Use of target Group: - A target group in an EC2 Load Balancer is a logical grouping of instances that receive incoming traffic. It plays a crucial role in distributing traffic effectively across multiple instances to ensure optimal performance and availability.

   Here's how target groups are utilized.

   - **Routing Traffic**: The target group determines which instances should receive traffic from the Load Balancer. It defines the set of registered instances that can handle requests.

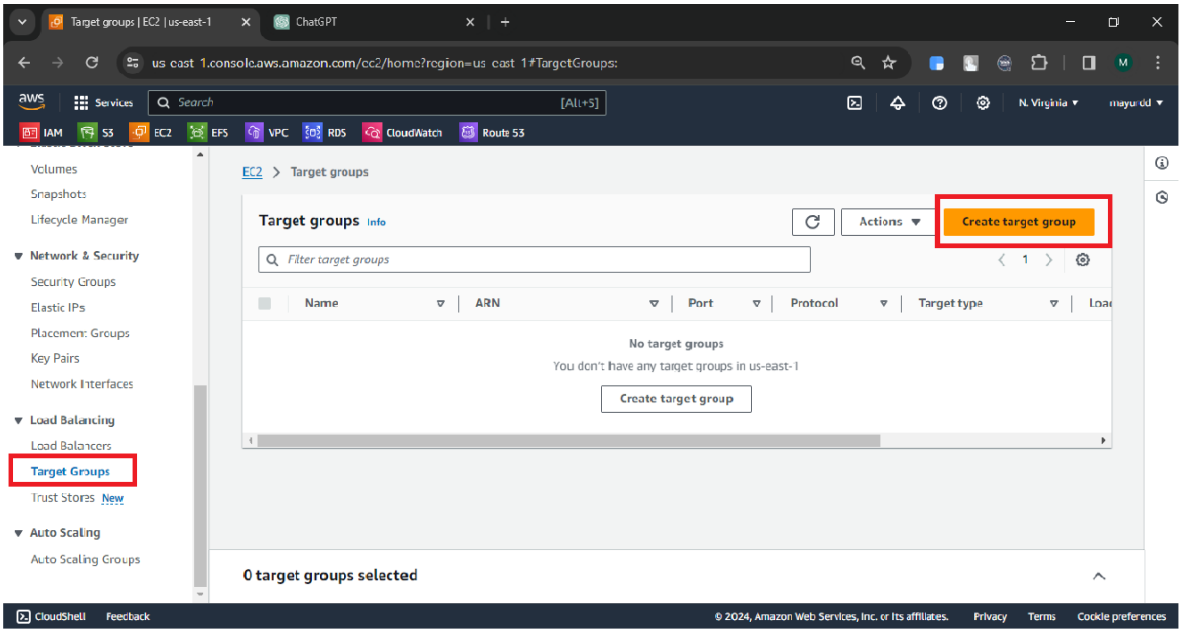
   - **Health Checks**: Target groups enable health checks on instances, monitoring their status. Instances failing health checks are automatically removed from the load-balanced pool, ensuring that traffic is directed only to healthy instances.

   - **Auto Scaling Integration**: When used in conjunction with Auto Scaling groups, target groups facilitate the dynamic addition or removal of instances based on demand. Auto Scaling responds to changes in demand by adjusting the number of instances in the target group.

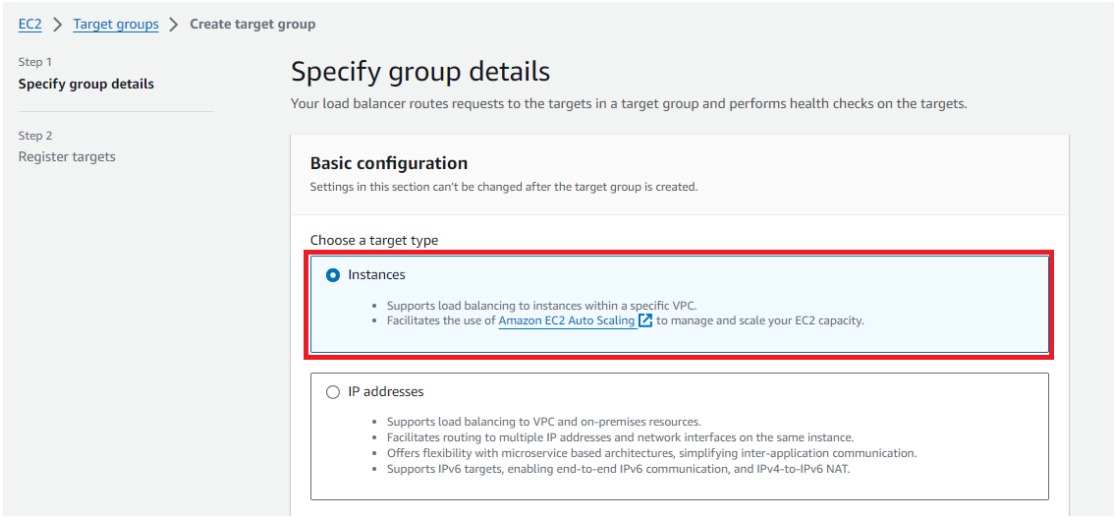
   - **Load Balancing Algorithms**: Target groups allow you to define load balancing algorithms to distribute traffic evenly, enhancing the overall performance and resource utilization.

   - **Security Groups Integration**: The Network Load Balancer manages traffic from the security groups associated with instances in the target group, ensuring a secure flow of data.

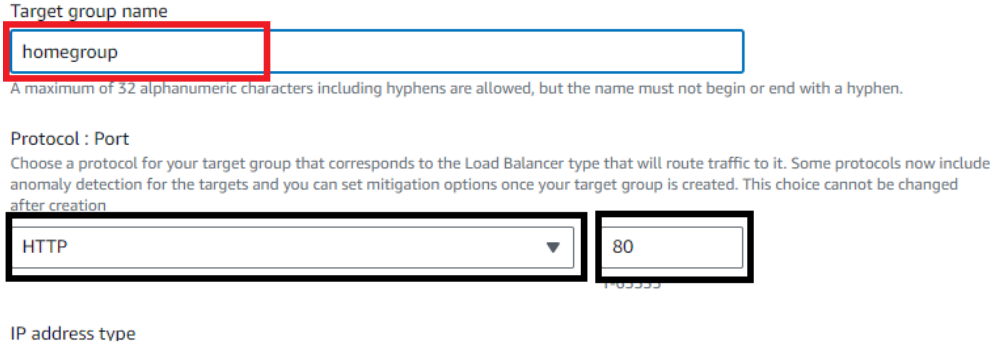
4. Click on **Target group** and click on **Create target group** option



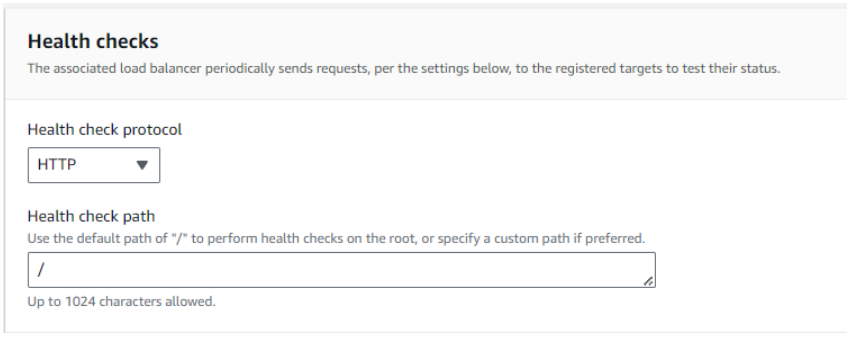
5. Select target type (in this case **instances**)



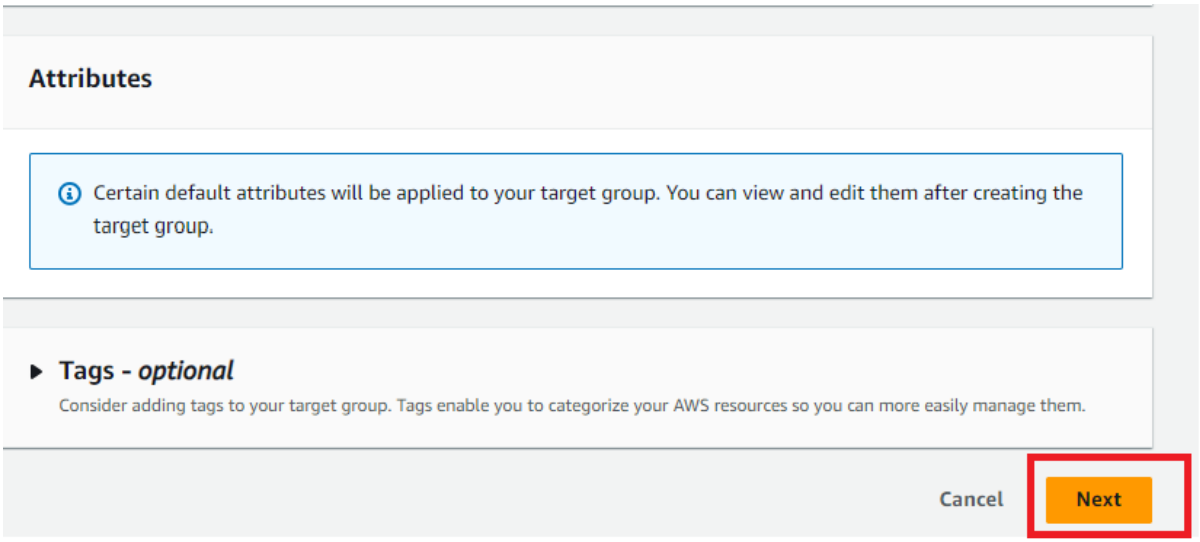
6. Specify name and the port



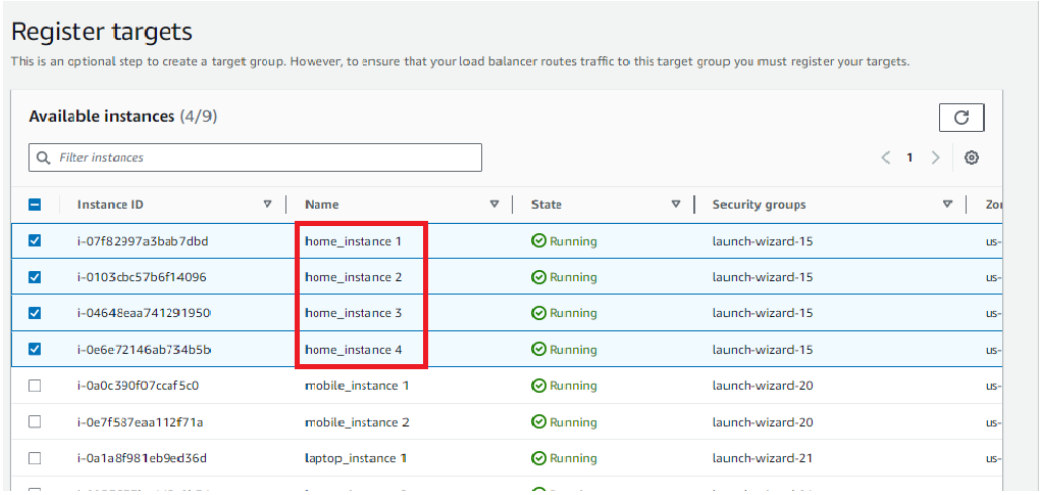
7. Select the Health check configuration….
   (health check means Aws check the instances are running or not In background, in not then it will not forward the traffic to that instance.
   Also check the Advance health option…..)



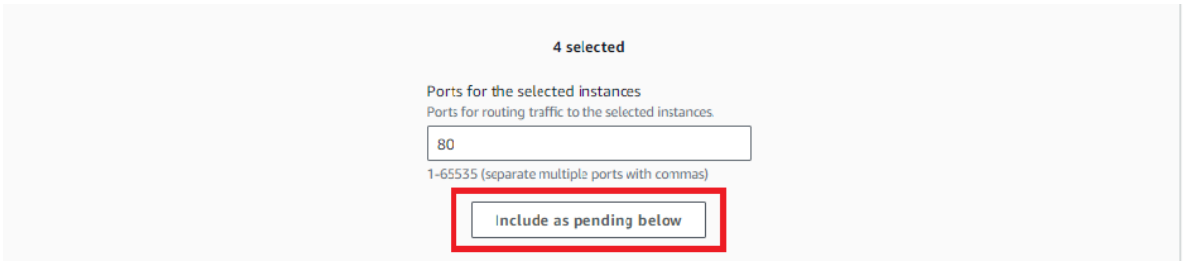
8. Click on next option



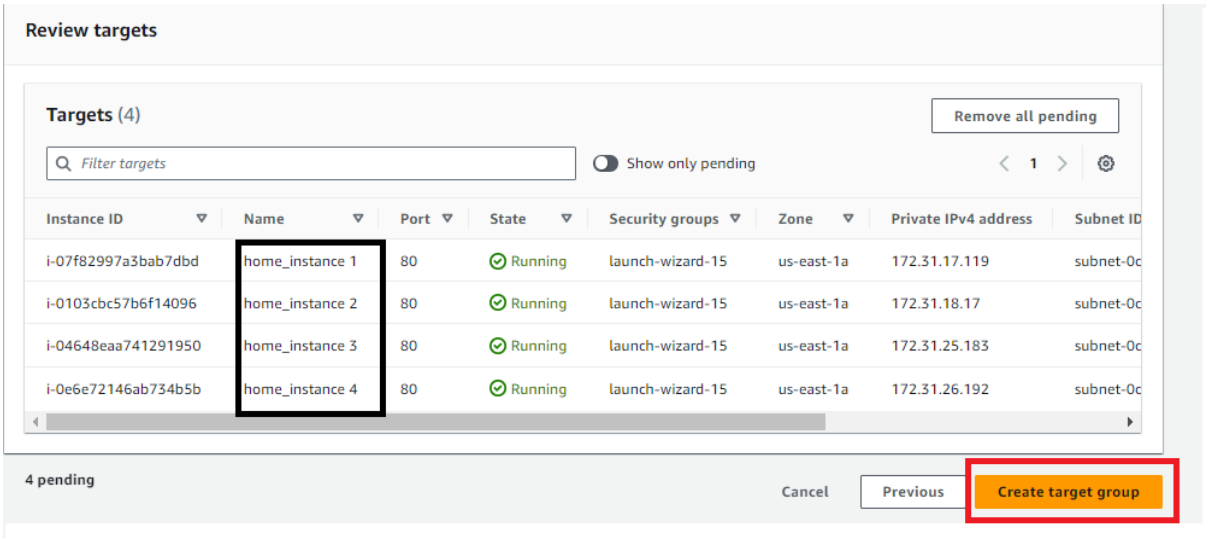
9. **<u>Select the instance</u>** which you want to add in first target group.



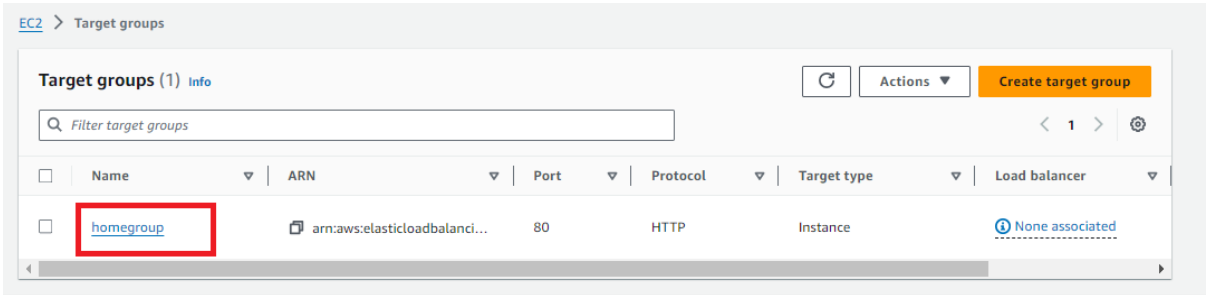
10.Click on **include as pending below** option.



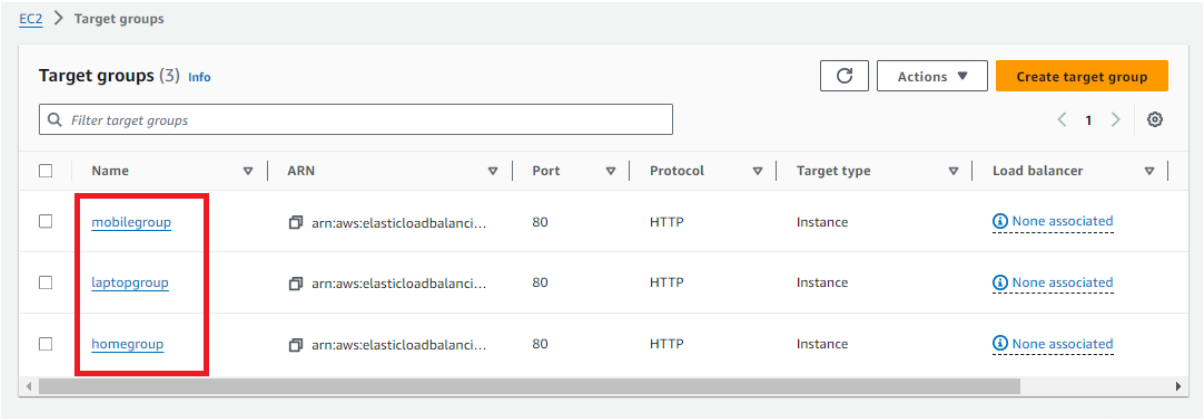
11.Click on **create group target** option



12.Target group created successfully….



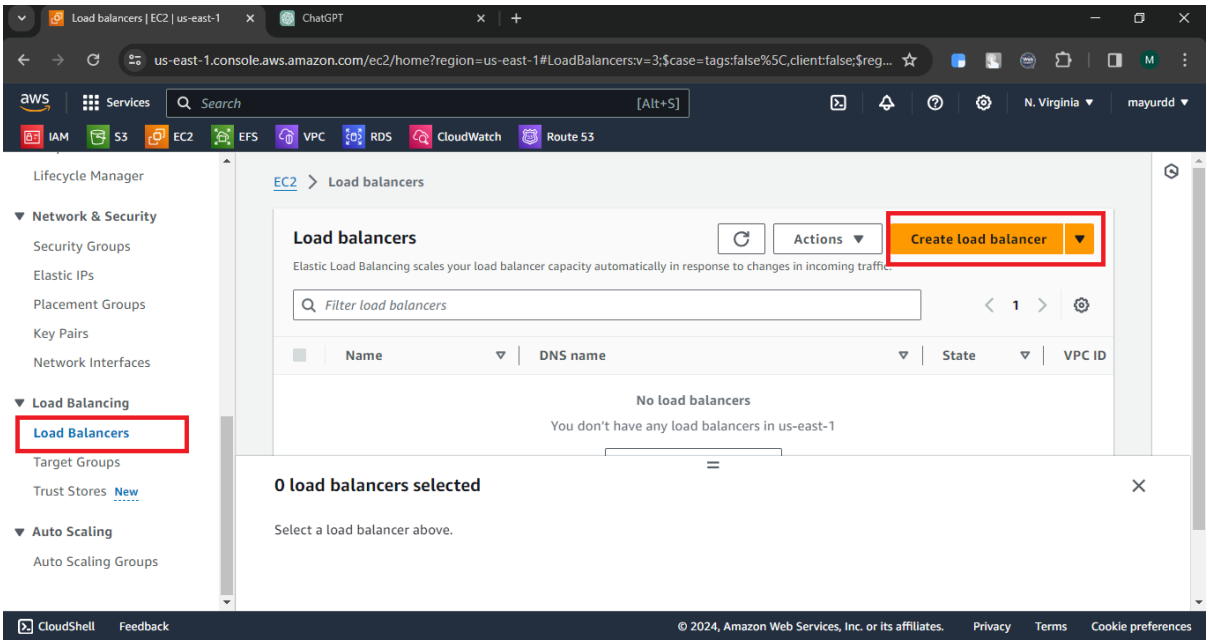
13.Perform same process and create another **three** target groups.



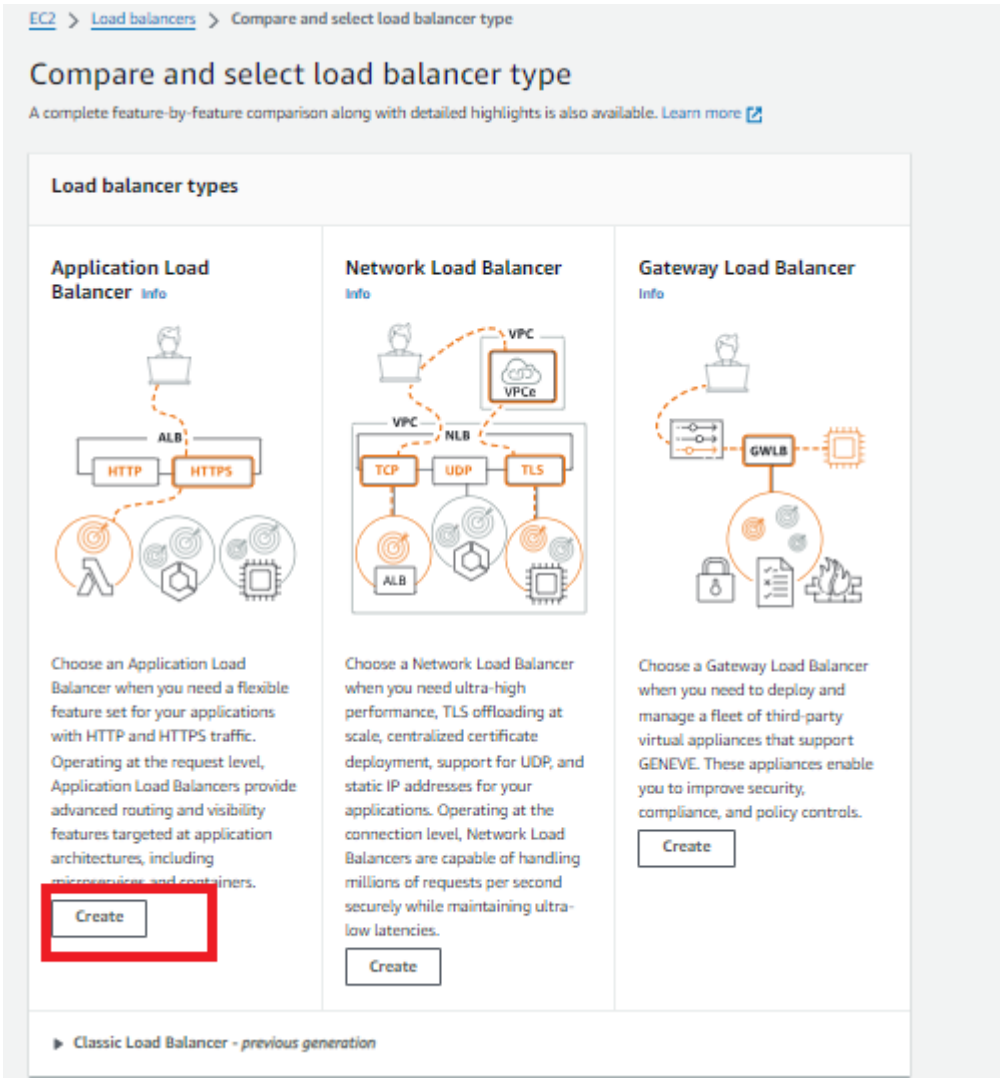
14.After creating Target group we need to create load balancer….

- Acts as the entry point for incoming traffic.
- Distributes traffic across multiple instances or targets, enhancing availability and fault tolerance.
- Manages routing decisions based on configured algorithms.
- Operates at the application, network, or transport layer, depending on the type of load balancer (Application Load Balancer, Network Load Balancer).

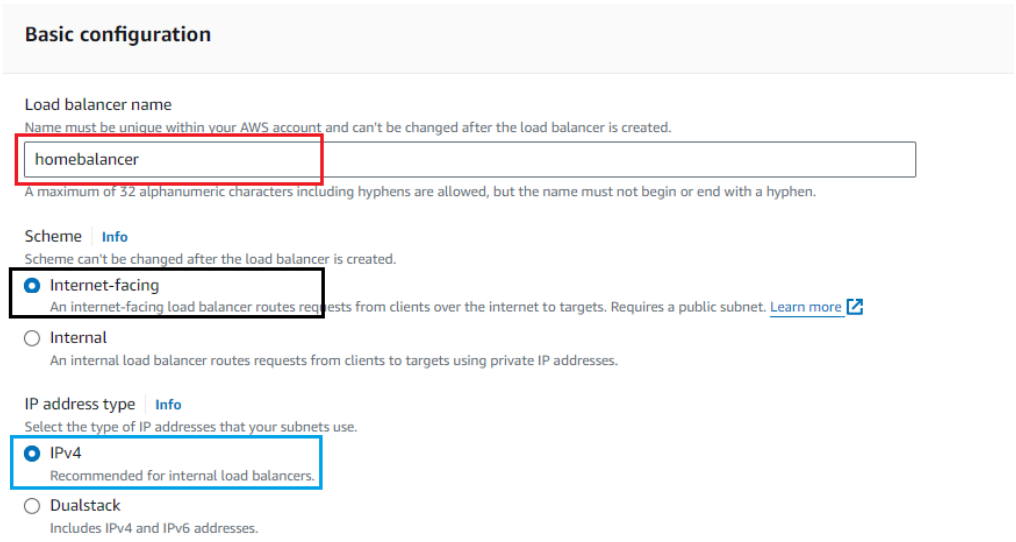
15. Click on **Load Balancers** and **Create load balancer** option



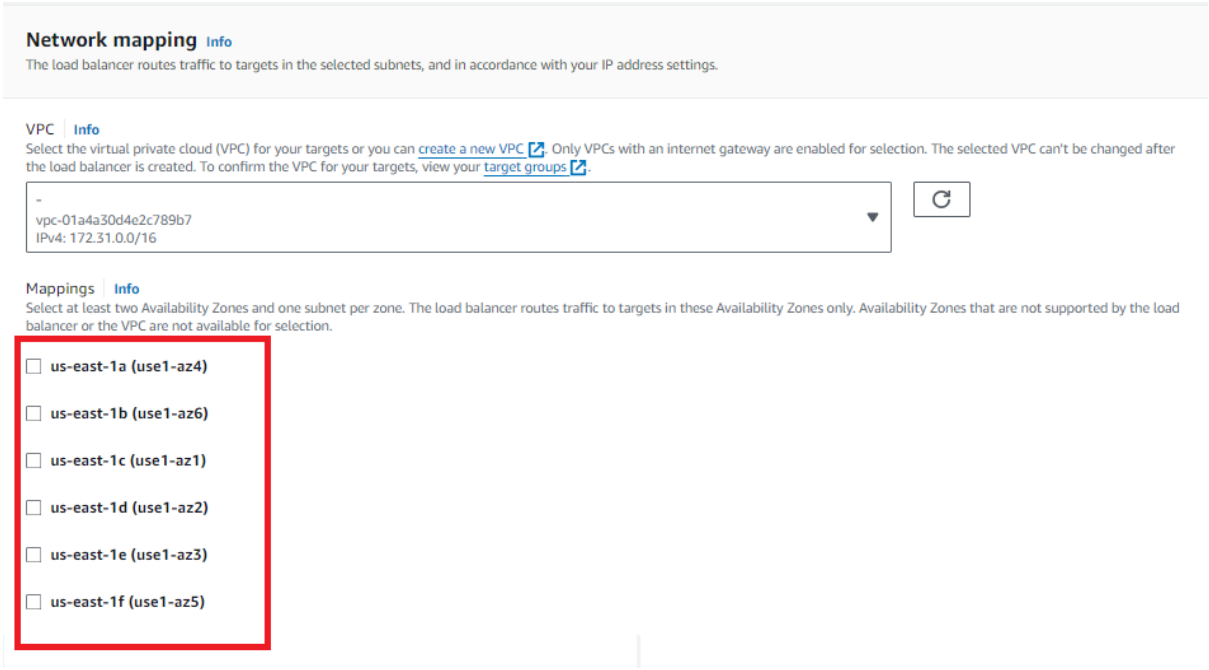
16.Click on **Application load balancer**



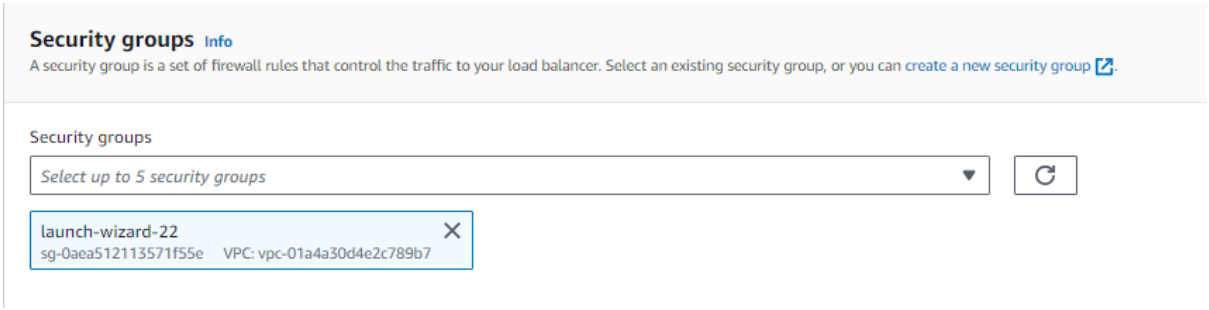
17.Basic configuration….



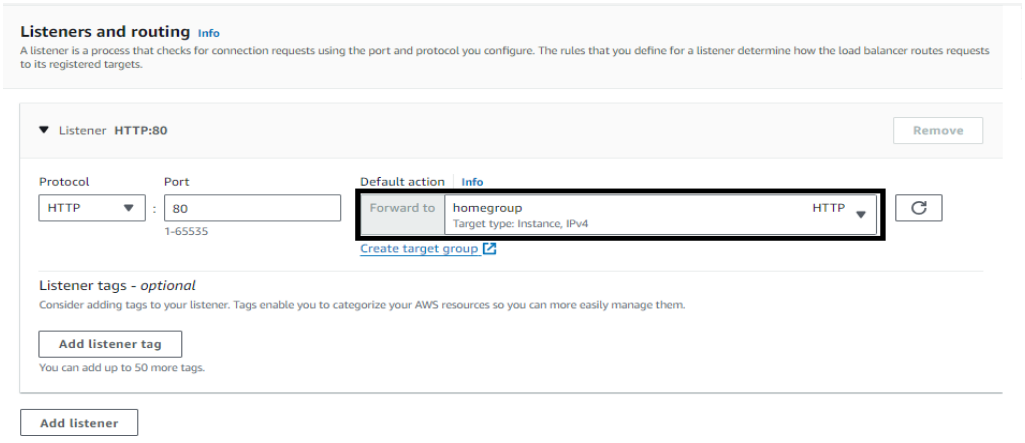
18. **Click on all zones** (because we are nor created all instances in one zone)



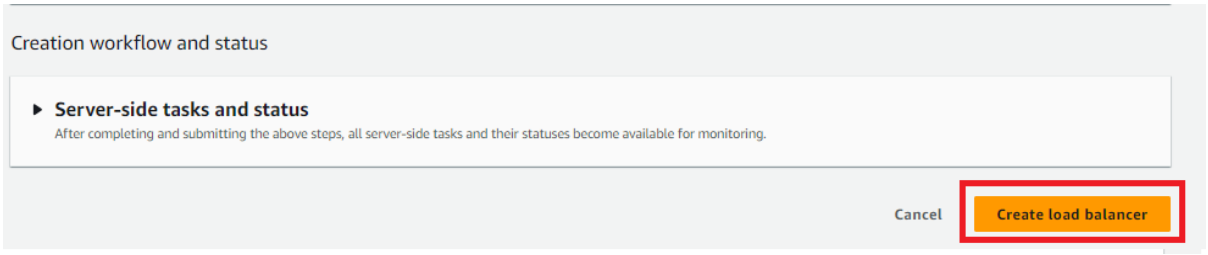
19. Select security group (note port 80 should be enabled in selected security group)



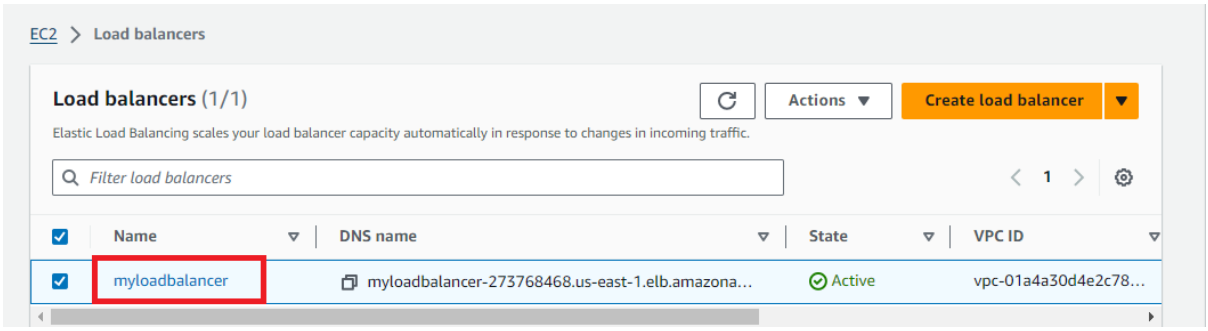
20. Select the **<u>target group, port and protocol</u>**



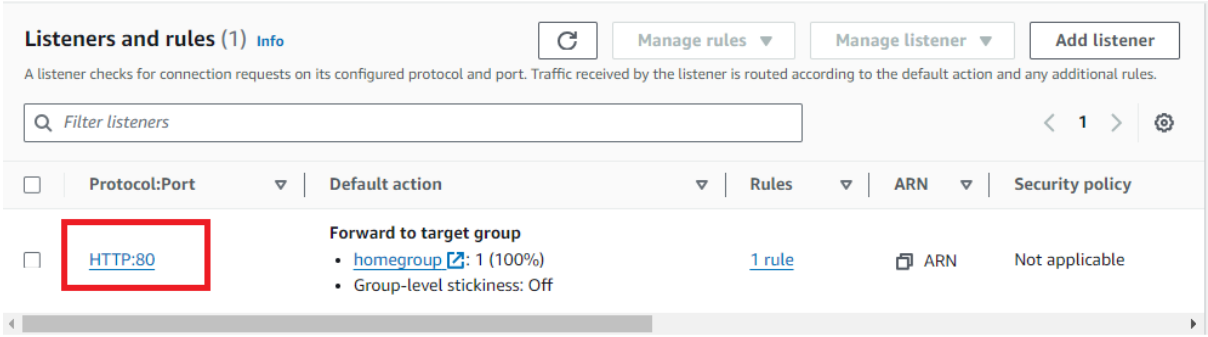
21.Click on create load balancer



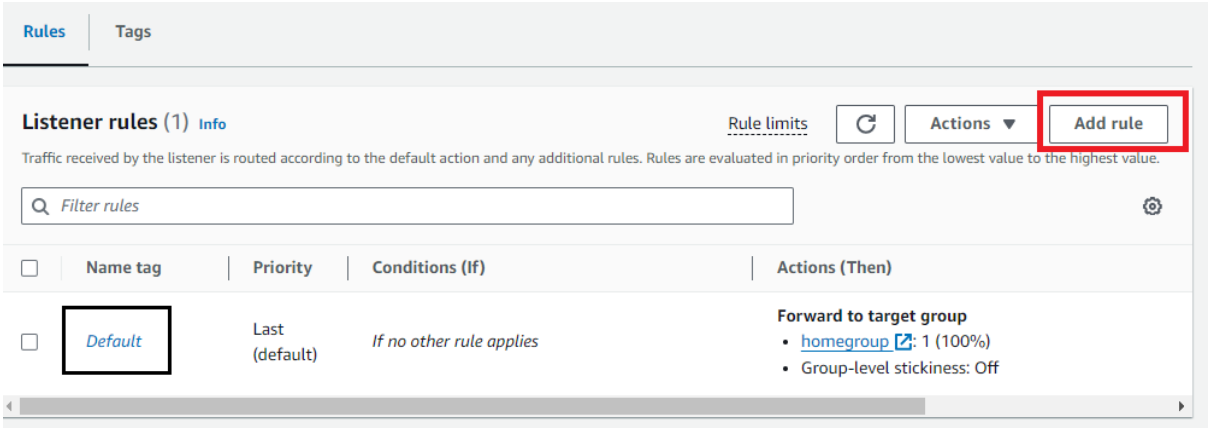
22.Load balancer Created successfully….



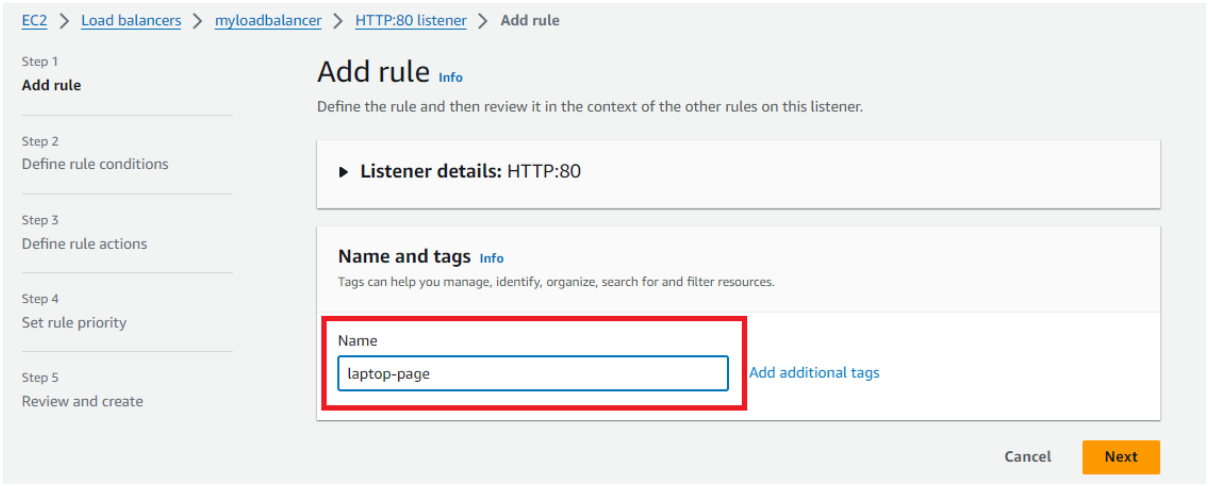
23. Select the Created load balancer and click on **(HTTP:80)** option



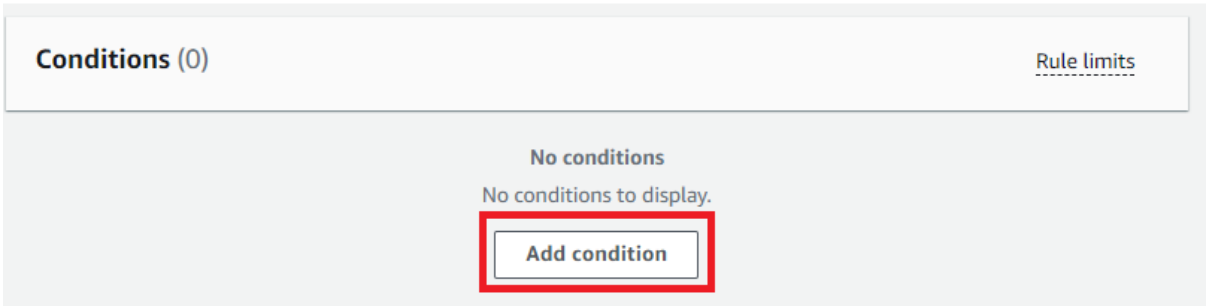
24. The Default Rule is already created. Click on **Add rule** option.



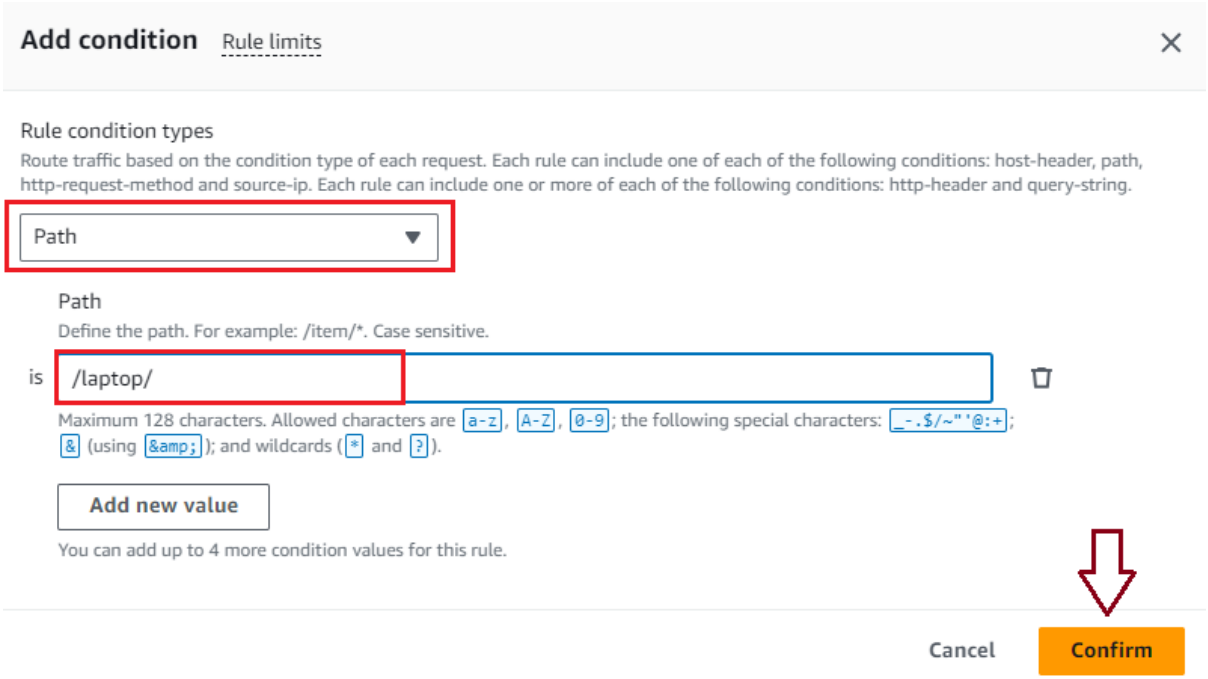
25.Give the name and click on next



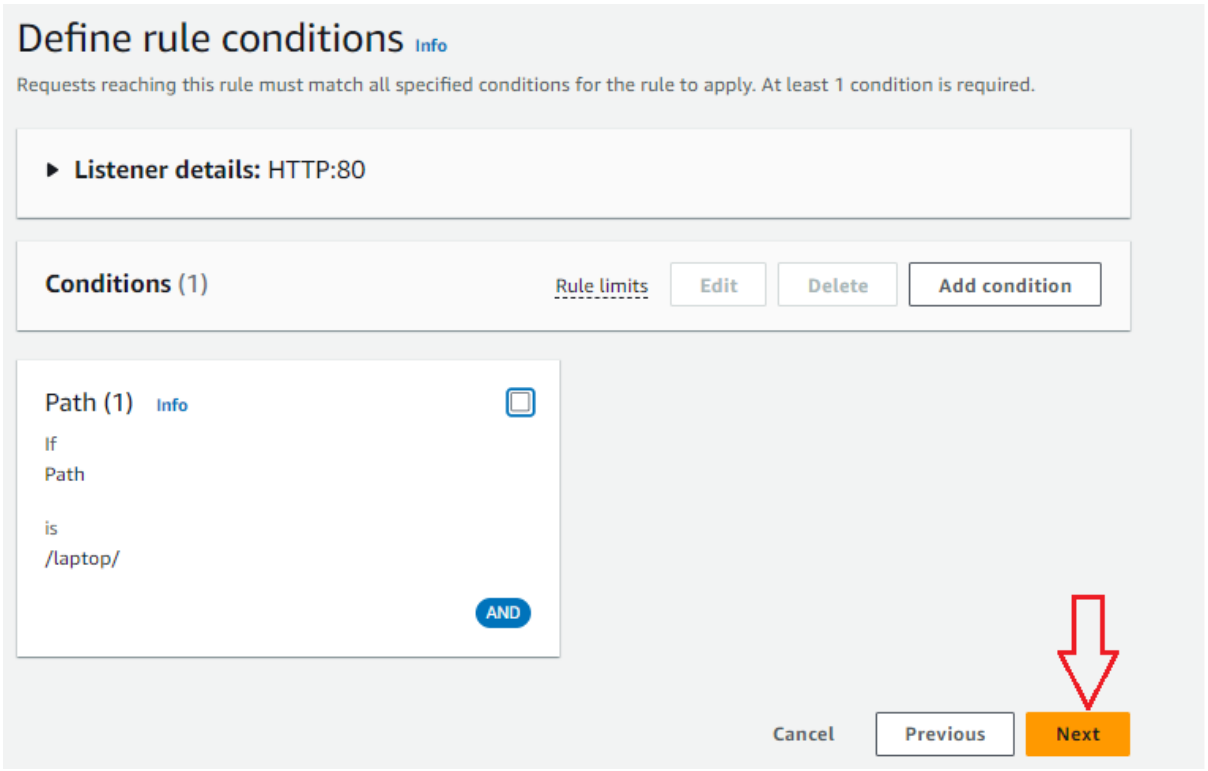
26.Click on **Add condition** option



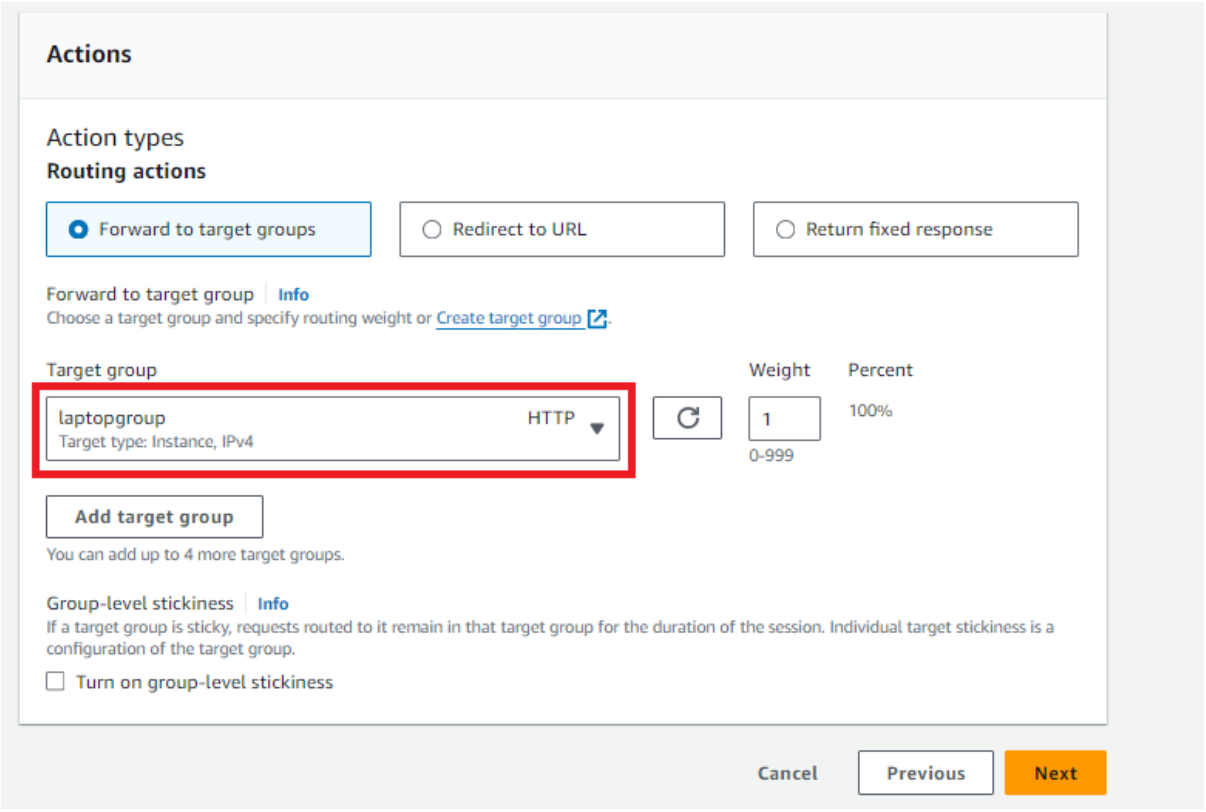
27.Select the Preferred options and click on confirm



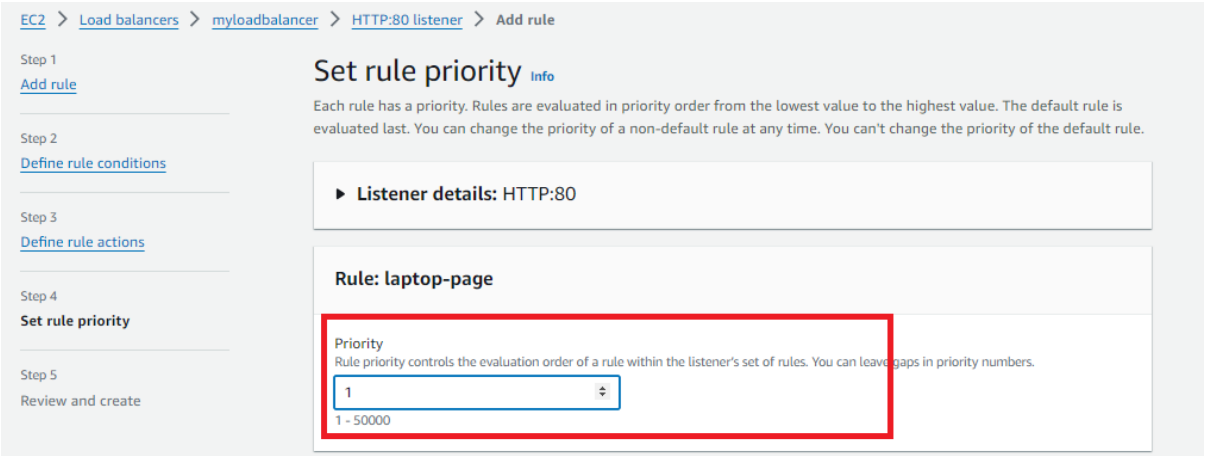
28. Click on next option



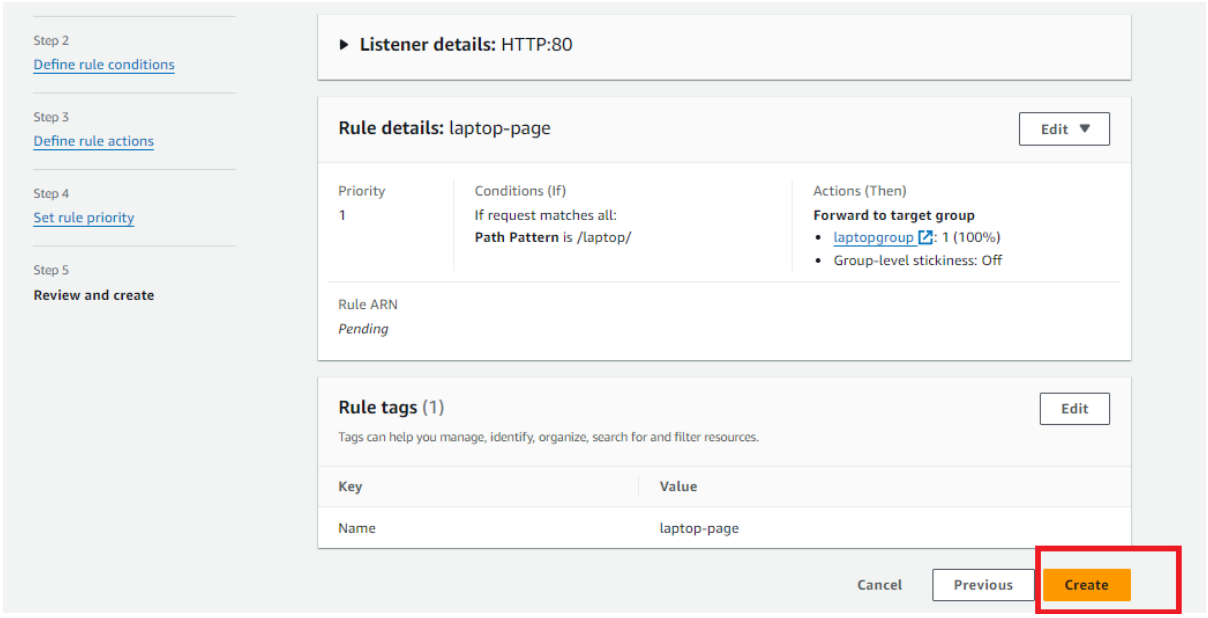
29.Select the target group and click on next



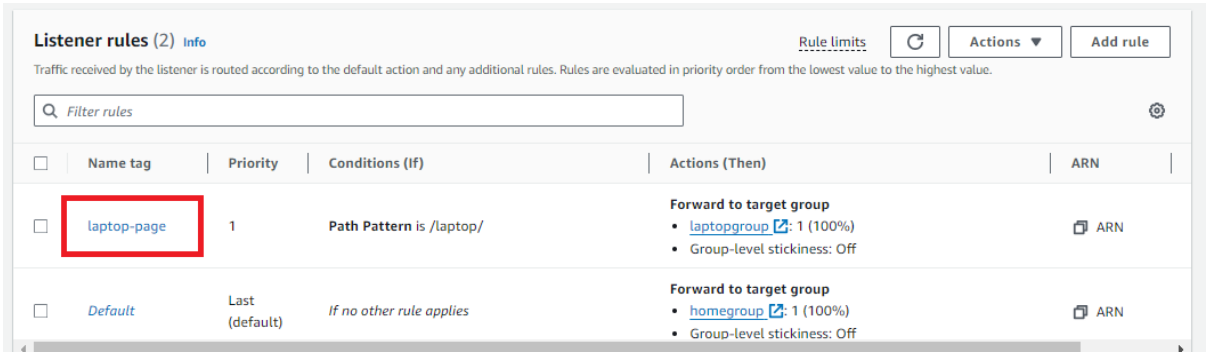
30.Set the priority is 1



31. Click on **create** option



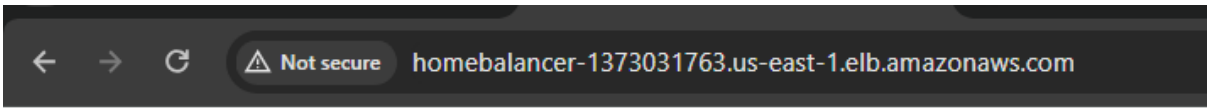
32. New rule added successfully….



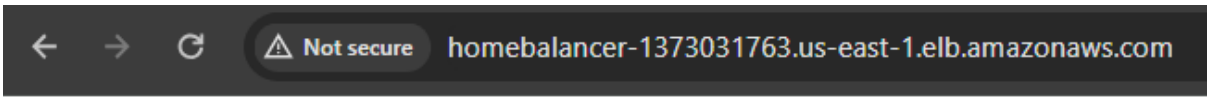
33. Follow same steps and create rule for **mobile page**.
New rule is added

34.Results

#Home page using load balancer

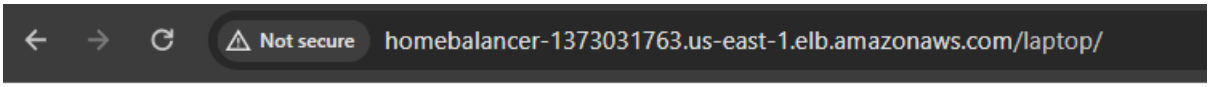


this is main homepage ip-172-31-25-183.ec2.internal

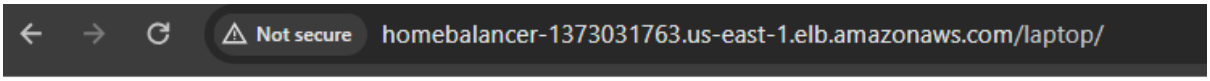


this is main homepage ip-172-31-26-192.ec2.internal

#Laptop page using load balancer

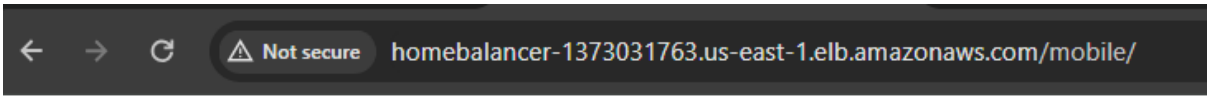


this is laptop page ip-172-31-29-60.ec2.internal



this is laptop page ip-172-31-27-97.ec2.internal

#Mobile page using load balancer



this is mobile page ip-172-31-23-126.ec2.internal

this is mobile page ip-172-31-24-42.ec2.internal