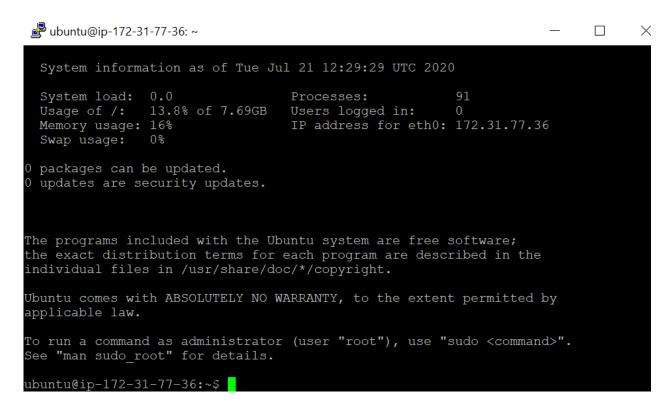


# Module 3: Hands-On: AWS ELB Weighted Routing

**Step 1**: Create 2 Ubuntu EC2 instances with the security group allowing both SSH port 22 and HTTP port 80.



**Step 2**: Login to one of your instances



Run both of these commands in your instance:

- sudo apt-get update
- sudo apt-get install apache2 -y



This is what you will see once Apache2 is installed

```
Created symlink /etc/systemd/system/multi-user.target.wants/apache-htcacheclean.service - /lib/systemd/system/apache-htcacheclean.service.

Processing triggers for libc-bin (2.27-3ubuntu1) ...

Processing triggers for systemd (237-3ubuntu10.41) ...

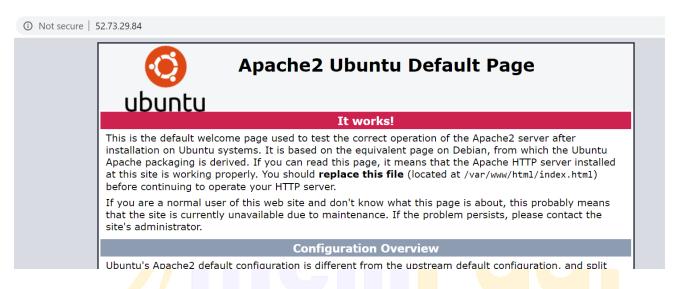
Processing triggers for man-db (2.8.3-2ubuntu0.1) ...

Processing triggers for ufw (0.36-0ubuntu0.18.04.1) ...

Processing triggers for ureadahead (0.100.0-21) ...

ubuntu@ip-172-31-77-36:~$
```

**Step 3**: Copy the Public IP address of the instance and paste in your browser to check out the Ubuntu Default Page.



**Step 4**: Now run the commands provided below in the same order to change the contents of the index.html file.

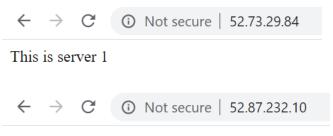
- 1. cd /var/www/html
- 2. ls
- 3. sudo rm index.html
- 4. sudo nano index.html
- 5. Type your text such as "This is server 1" then [CTRL + 0] and Enter to save. [CTRL + X] to exit the text editor

```
ubuntu@ip-172-31-77-36:/var/www/html$ ls
index.html
ubuntu@ip-172-31-77-36:/var/www/html$ sudo rm index.html
ubuntu@ip-172-31-77-36:/var/www/html$ ls
ubuntu@ip-172-31-77-36:/var/www/html$
ubuntu@ip-172-31-77-36:/var/www/html$
ubuntu@ip-172-31-77-36:/var/www/html$ sudo nano index.html
ubuntu@ip-172-31-77-36:/var/www/html$ cat index.html
This is server 1
```



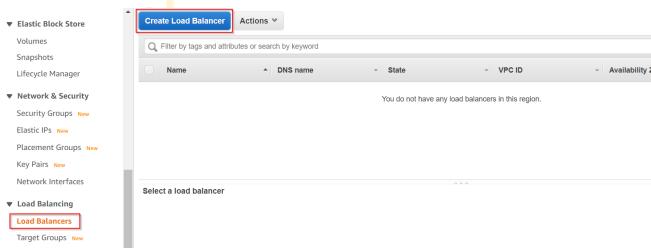
```
ubuntu@ip-172-31-68-255:~$ cd /var/www/html
ubuntu@ip-172-31-68-255:/var/www/html$ sudo rm index.html
ubuntu@ip-172-31-68-255:/var/www/html$ sudo nano index.html
ubuntu@ip-172-31-68-255:/var/www/html$ cat index.html
This is server 2
```

**Step 4**: Repeat Step 2 and 3 for your second instance. To differentiate between the servers, type different sentences in the webserver1 and webserver2 index file. For reference, check the below images:

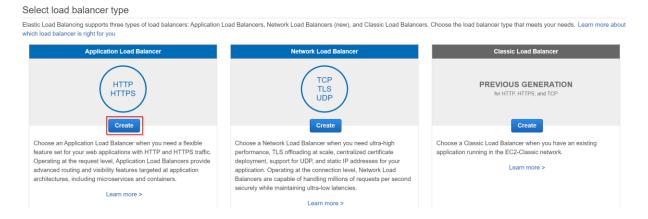


This is server 2

**Step 5**: Now, let's create an Application Load Balancer. Scroll down and click on Load balancers and click on the Create Load Balancer button.

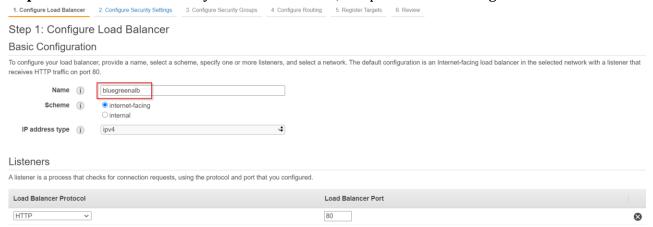


**Step 6**: Select Application Load Balancer

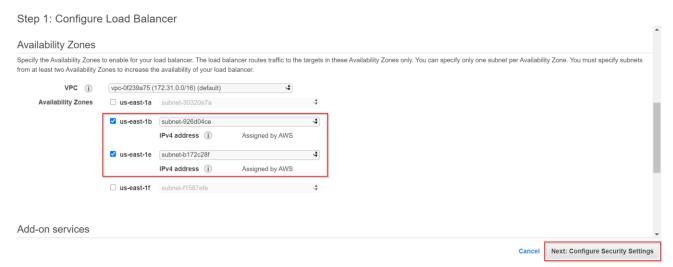




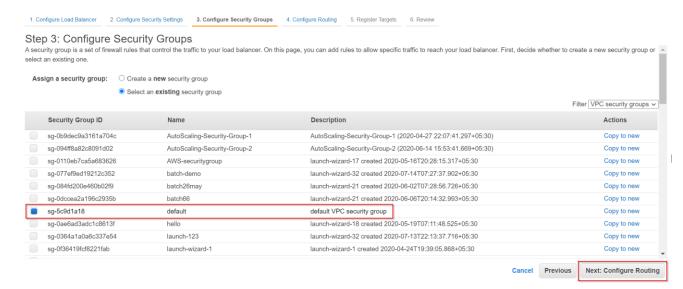
## **Step 7**: Provide the name for your Load Balancer; Keep it internet-facing and scroll down.



• Keep the VPC as default and select the Availability Zones in which your EC2 instances are and the minimum is 2 AZs.

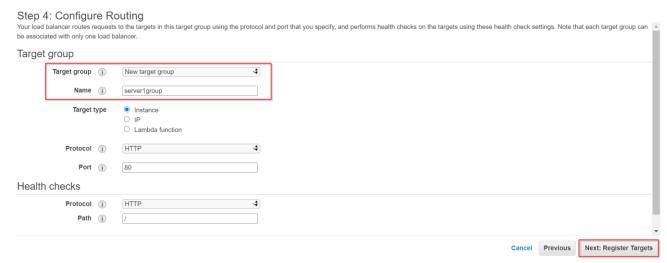


• Keep the Security Group as default too. No change is required in this step.

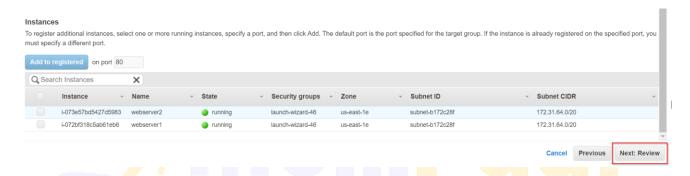




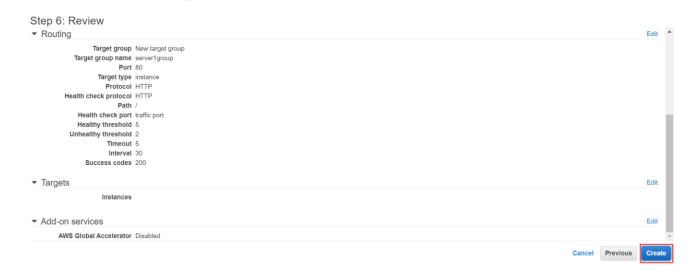
• Provide a name to a Target Group and keep every other option as default



• Now, directly click the button Next: Review

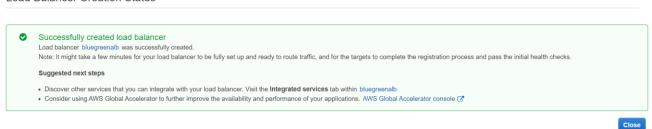


Check everything once and click Create and your load balancer is created

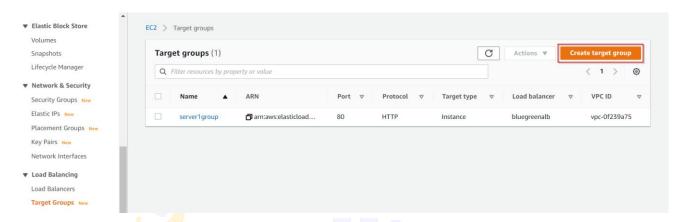




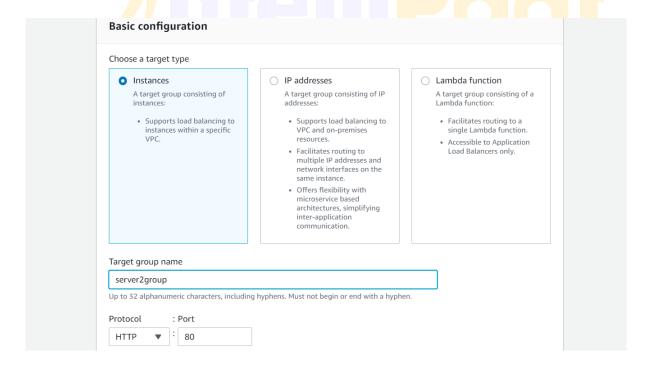
Load Balancer Creation Status



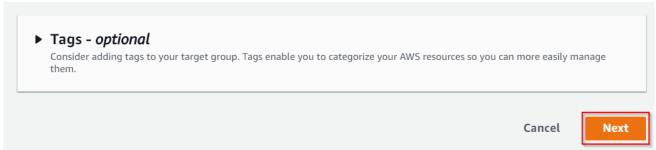
**Step 8**: We already created one target group and now to create one more click the Create target group button.



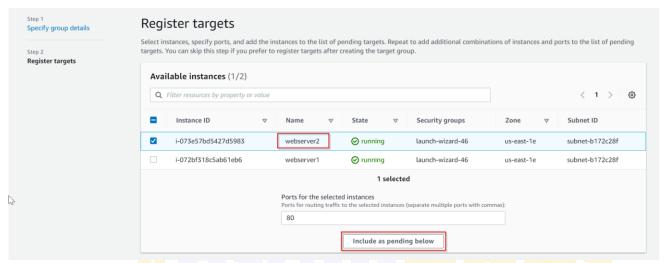
**Step 9**: Keep the target type as Instances and provide a name to your second target group. Then click **Next**.



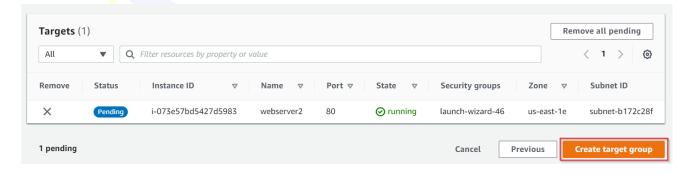




• Register your server 2 to the target group 2. Once selecting, click the below button.

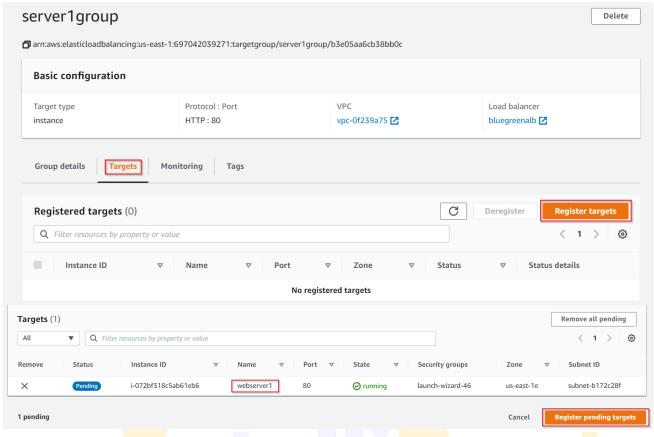


• When it shows the instance is pending, click on the Create target group button and complete creating the second target group.

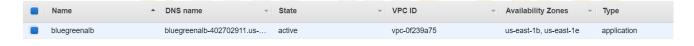


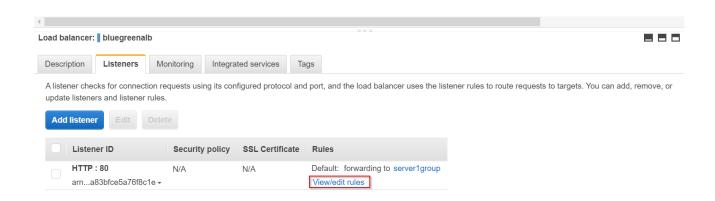
Now, register the server 1 to the target group 1.





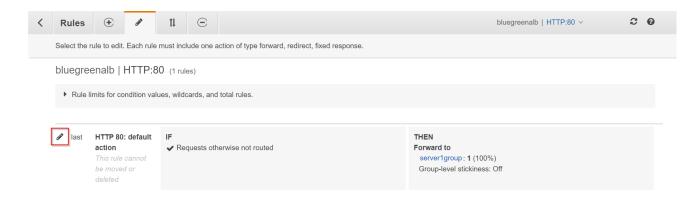
## **Step 10**: Now, open the Load balancer and go to listeners. Click on View/edit rules.



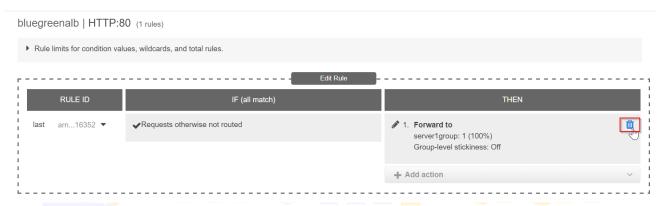




 Once opened, click on the pen icon to open the edit tab and click on the pen icon next to the rule below



• Delete the rule by clicking on the delete icon

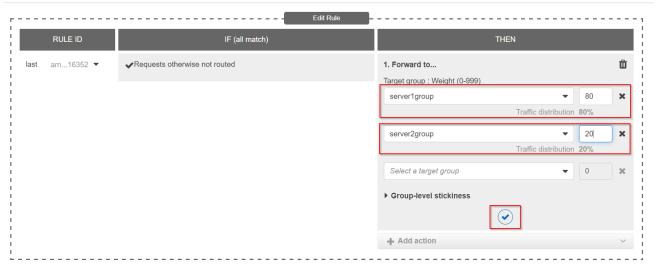


Now click on Add action and select Forward to option.

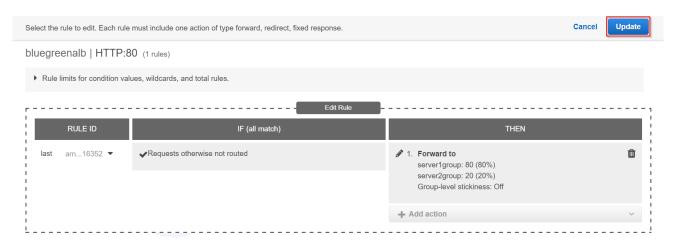


• Select both of the target groups and provide the weight next to them like the below image.

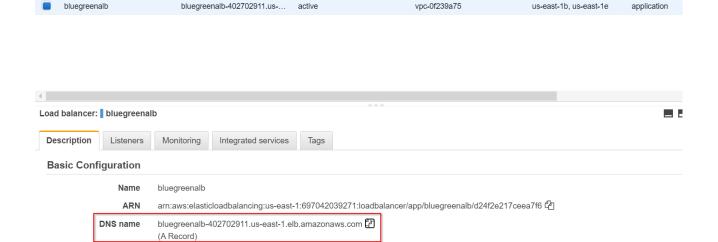




Now to finish it off, click on Update.

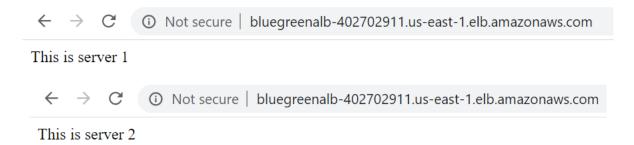


Now copy the DNS name and paste in your browsers URL space





• You will be able to see the webpage. Keep refreshing and you will get the second server eventually



**Step 11**: Now to test the traffic distribution, open one of your instances and run the below command.

• for i in {1..10}; do curl your-alb-dnsname.us-east-1.elb.amazonaws.com; done

```
ubuntu@ip-172-31-77-36:~$ for i in {1..10}; do curl bluegreenalb-402702911.us-ea
st-1.elb.amazonaws.com; done
This is server 1
This is server 2
This is server 2
This is server 1
This is server 2
This is server 2
This is server 1
This is server 2
This is server 2
Ubuntu@ip-172-31-77-36:~$ for i in {1..10}; do curl bluegreenalb-402702911.us-east-1.elb.amazonaws.com; done
This is server 1
This is server 1
This is server 1
This is server 2
```