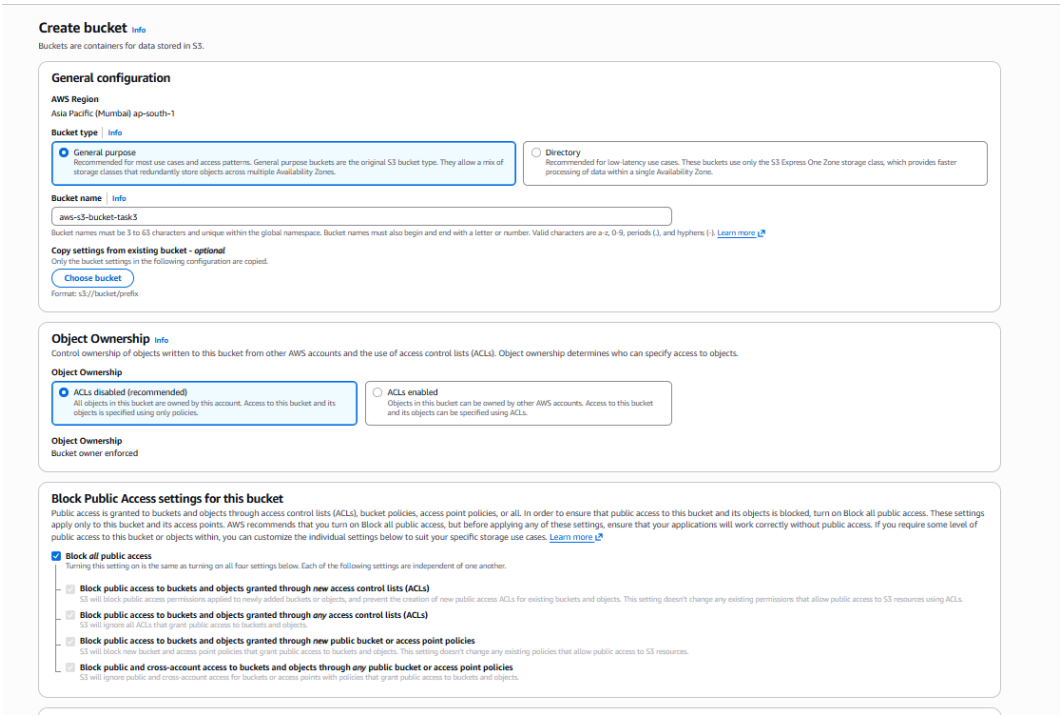
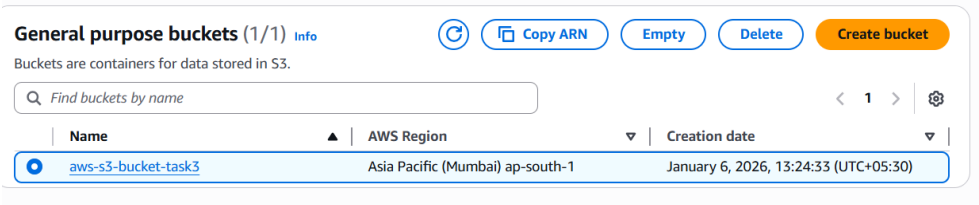
AWS Task-3

Task-1: Create a S3 bucket, with no public access and upload files to the bucket & view the logs using cloudwatch for the uploaded files.

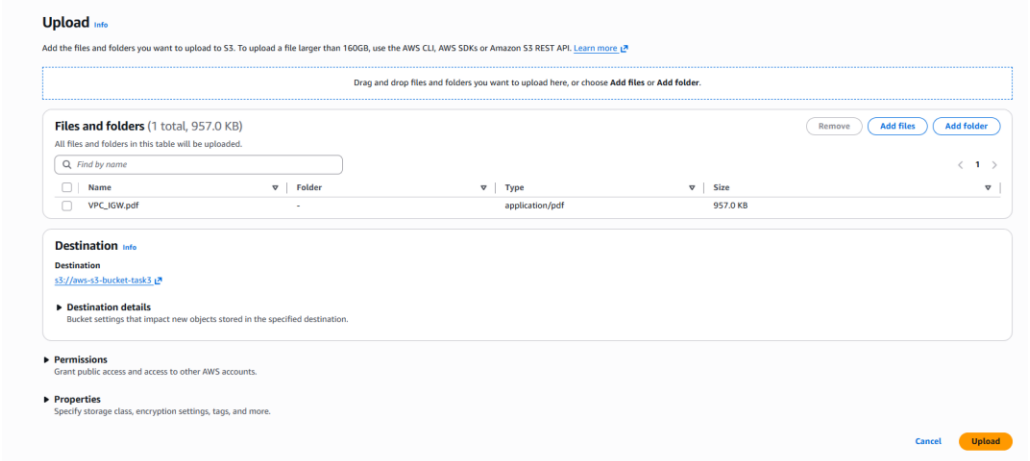
1. Creating S3 bucket.



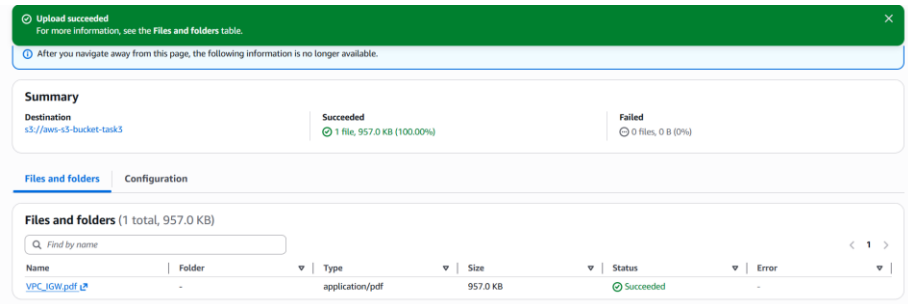
2. s3 bucket created.



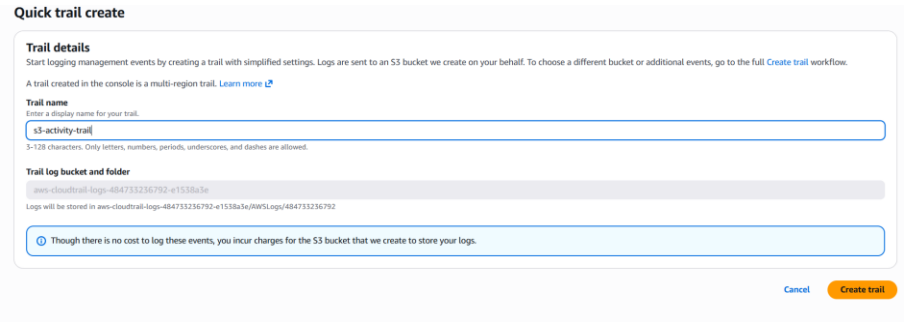
3. uploading files to s3 bucket.



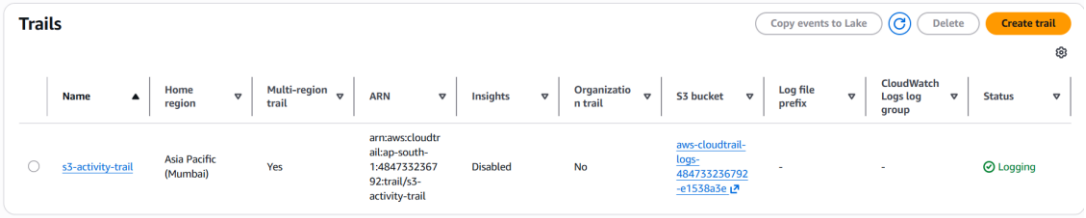
4. files uploaded.



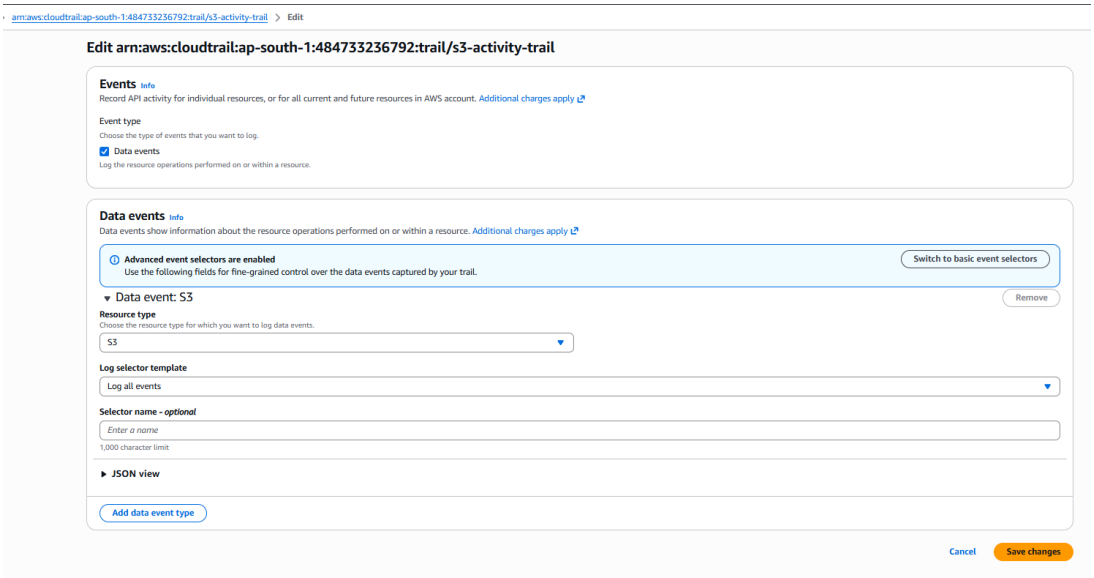
5. creating cloud trail.



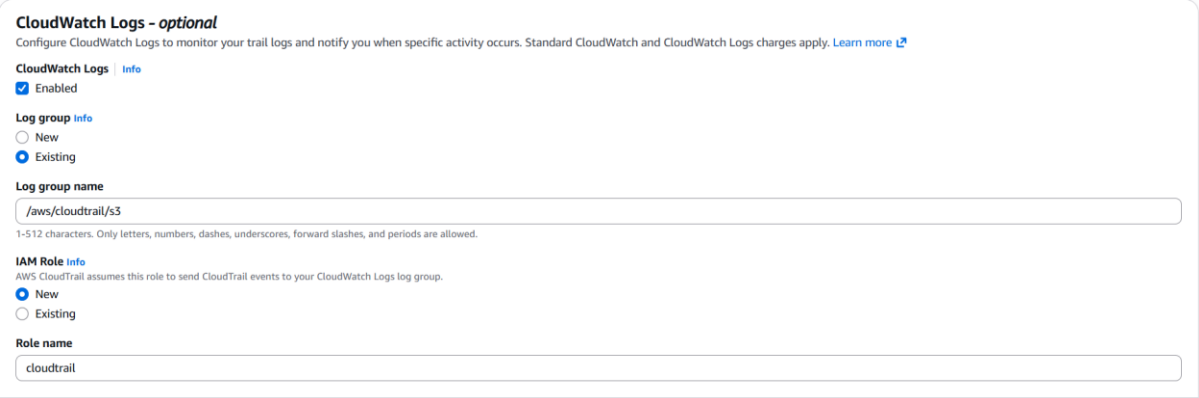
6. could trail created.



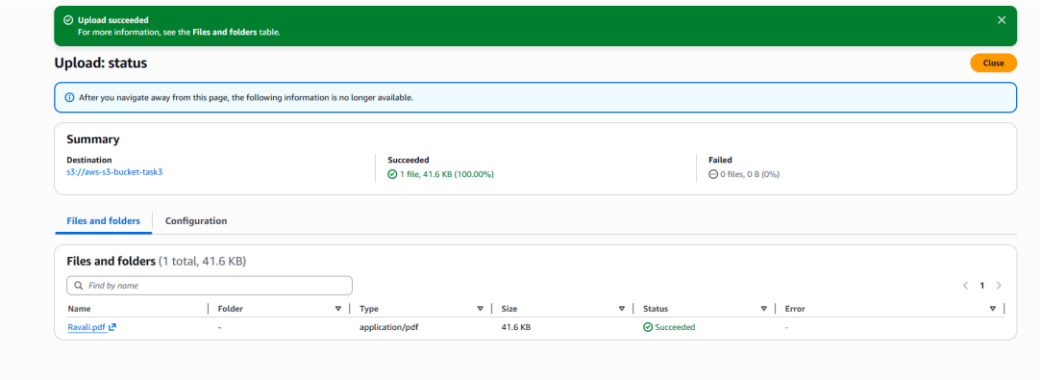
7. adding s3 bucket details in cloud trail under data events.



8. send cloud trail logs to cloud watch. Enable cloud watch logs and create log group and IAM role.



9. uploading file again in s3 bucket.



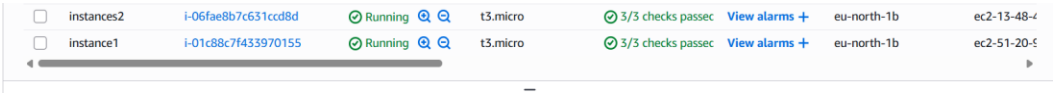
10. checking logs in cloud watch under log groups.

```
{
    "eventVersion": "1.11",
    "userIdentity": {
        "type": "AWSService",
        "invokedBy": "cloudtrail.amazonaws.com"
    },
    "eventTime": "2026-01-06T08:51:19Z",
    "eventSource": "s3.amazonaws.com",
    "eventName": "PutObject",
    "awsRegion": "ap-south-1",
    "sourceIPAddress": "cloudtrail.amazonaws.com",
    "userAgent": "cloudtrail.amazonaws.com",
    "requestParameters": {
        "bucketName": "aws-cloudtrail-logs-484733236792-e1538a3e",
        "Host": "aws-cloudtrail-logs-484733236792-e1538a3e.s3.ap-south-1.amazonaws.com",
        "x-amz-acl": "bucket-owner-full-control",
        "x-amz-server-side-encryption": "AES256",
        "key": "AWSLogs/484733236792/CloudTrail/us-east-1/2026/01/06/484733236792_CloudTrail_us-east-1_20260106T0850Z_mK9JY1sKjdtWczER.json.gz"
    },
    "responseElements": {
```

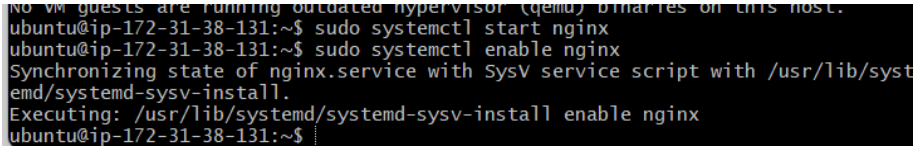

Back to top

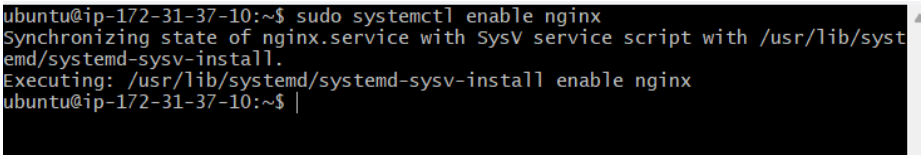
Task-2: Launch two ec2-instances and connect it to a application load balancer, where the output traffic from the server must be an load balancer IP address.

1. Created two ec2 instances.

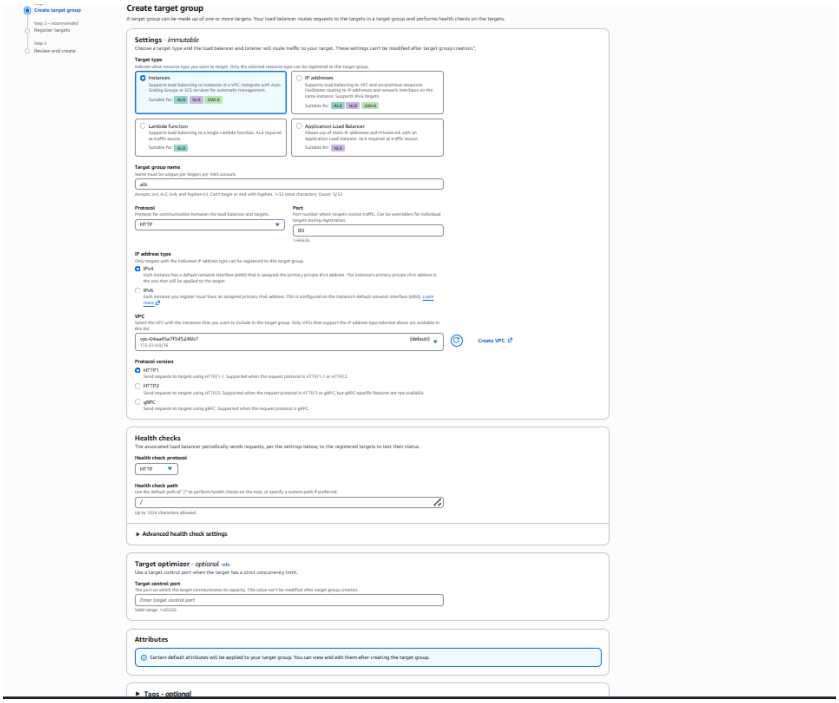


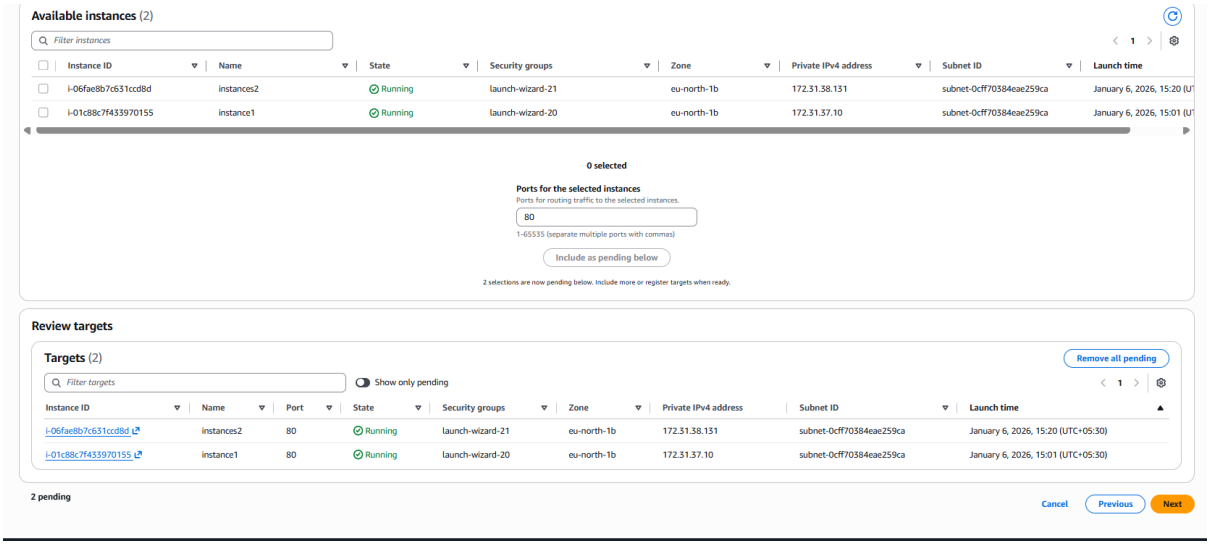
2. Install nginx in both instances.

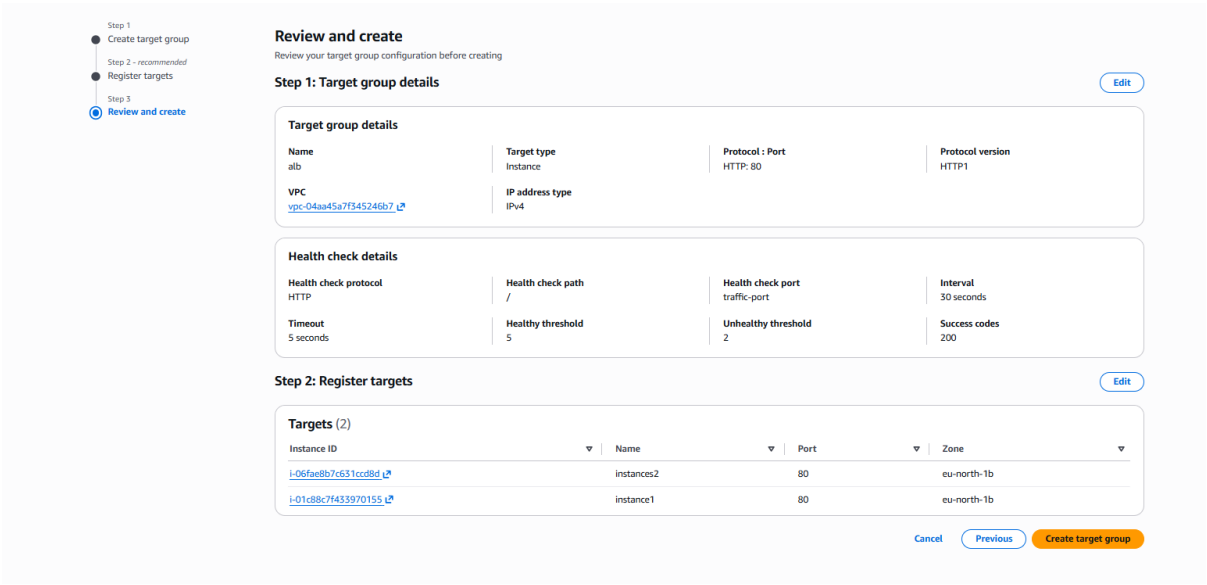




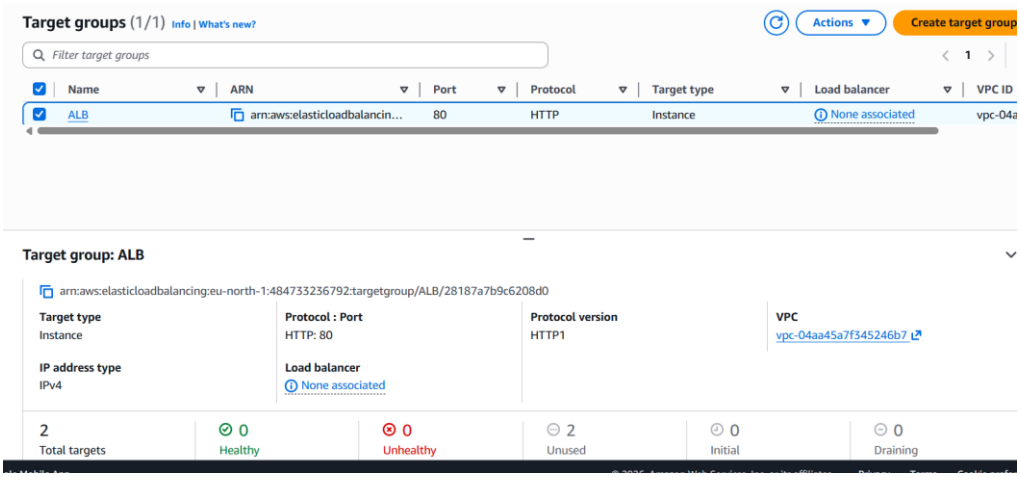
3. creating target groups.



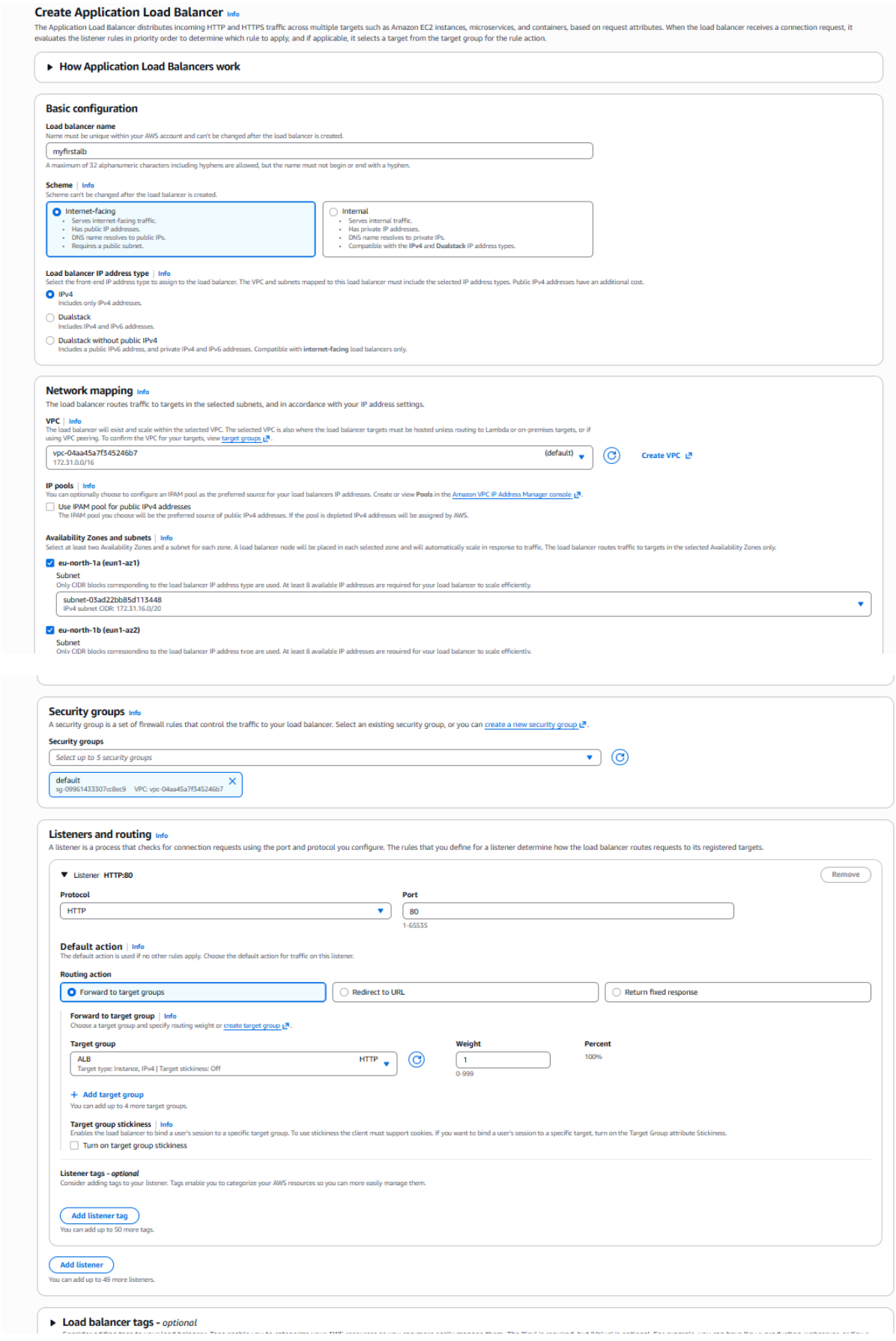




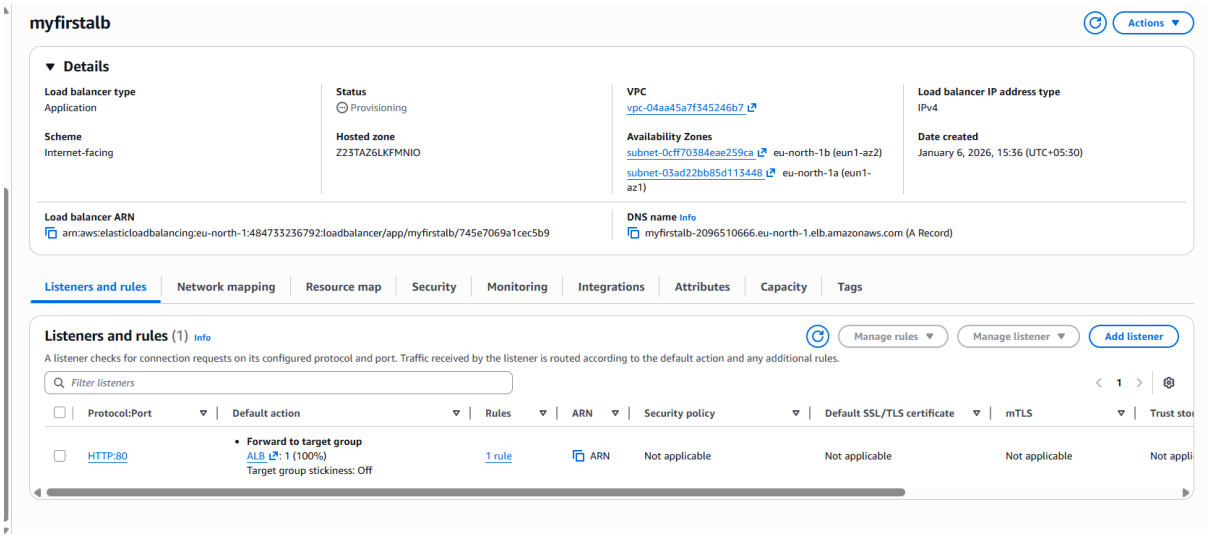
4. target group is created.



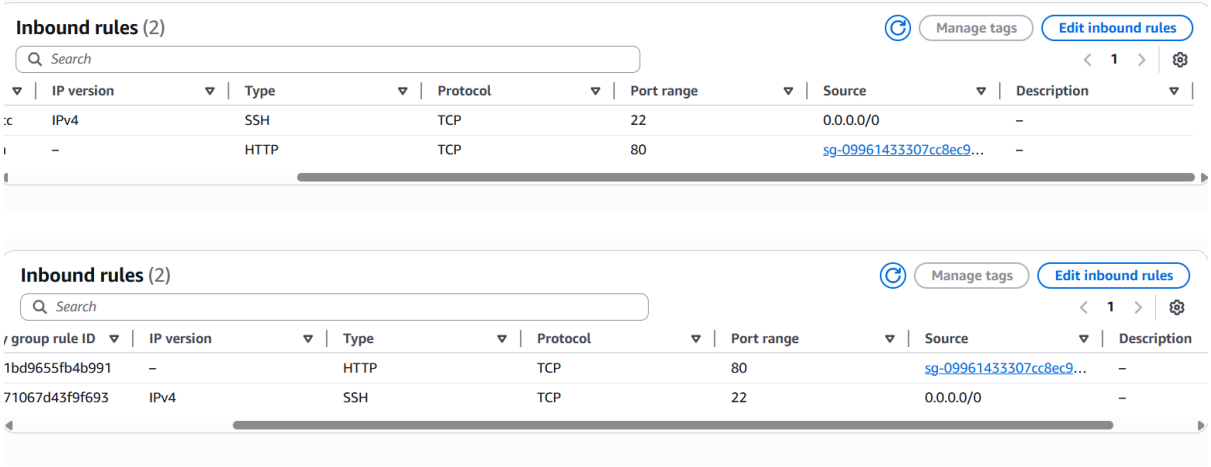
5. creating application load balancer.



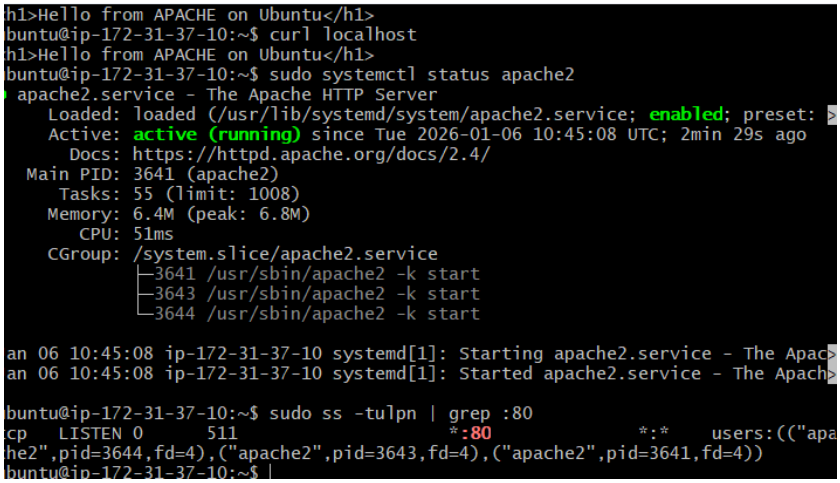
6. application load balancer created.



7. Updating ec2 security groups. (adding application load balancer security groups)



8. Installed nginx on one server and installed Apache on anther ec2.



```
OF
h1>Hello from NGINX on Ubuntu</h1>
buntu@ip-172-31-38-131:~$ curl localhost
!DOCTYPE html>
html>
head>
title>Welcome to nginx!</title>
style>
tml { color-scheme: light dark; }
ody { width: 35em; margin: 0 auto;
ont-family: Tahoma, Verdana, Arial, sans-serif; }
/style>
/head>
body>
h1>Welcome to nginx!</h1>
p>If you see this page, the nginx web server is successfully installed and
orking. Further configuration is required.</p>

p>For online documentation and support please refer to
a href="http://nginx.org/">nginx.org</a>.<br/>
ommercial support is available at
a href="http://nginx.com/">nginx.com</a>.</p>

p><em>Thank you for using nginx.</em></p>
```

9. Checking load balancer working fine or not. (copy and open ALB dns name in browser).

Hello from APACHE on Ubuntu

Welcome to nginx!

If you see this page, the nginx web server is successfully installed and working. Further configuration is required.

For online documentation and support please refer to nginx.org.
Commercial support is available at nginx.com.

Thank you for using nginx.

10. By accessing the ALB DNS and observing alternating responses, I confirmed proper load balancing.