Description:

Amazon Elastic Compute Cloud (Amazon EC2) provides scalable computing capacity in the Amazon Web Services (AWS) cloud. Using Amazon EC2 eliminates your need to invest in hardware up front so you can develop and deploy applications faster. You can use Amazon EC2 to launch as many or as few virtual servers as you need, configure security and networking, and manage storage. Amazon EC2 enables you to scale up or down to handle changes in requirements or spikes in popularity, reducing your need to forecast traffic.

Problem Statement:

Company ABC wants to move their product to AWS. They have the following things set up right now:

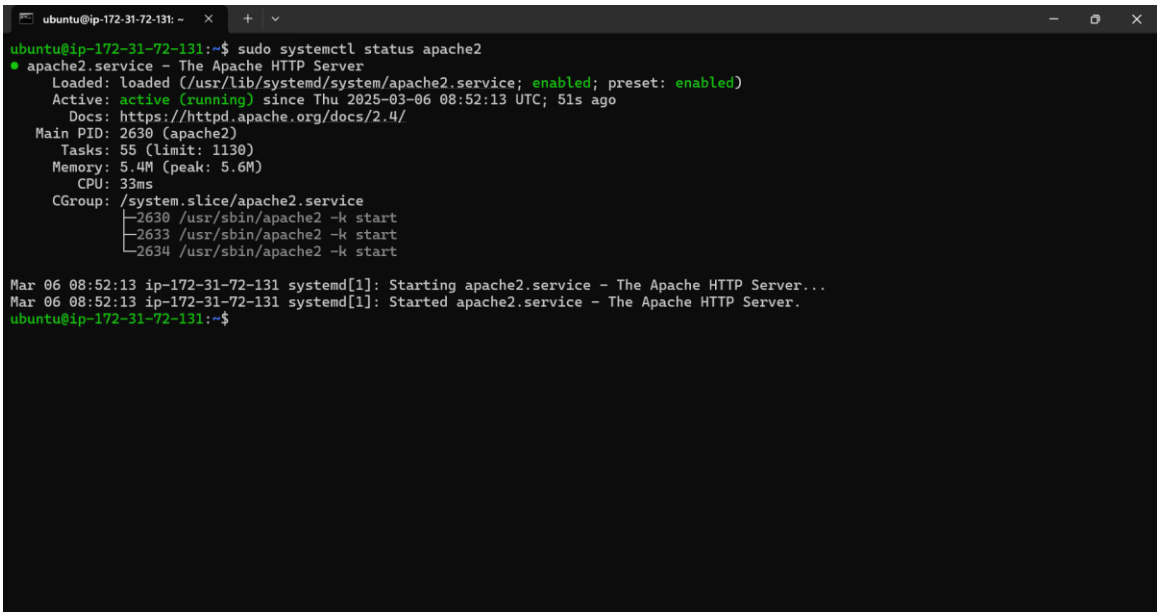
1. MySQL DB
2. Website (PHP)

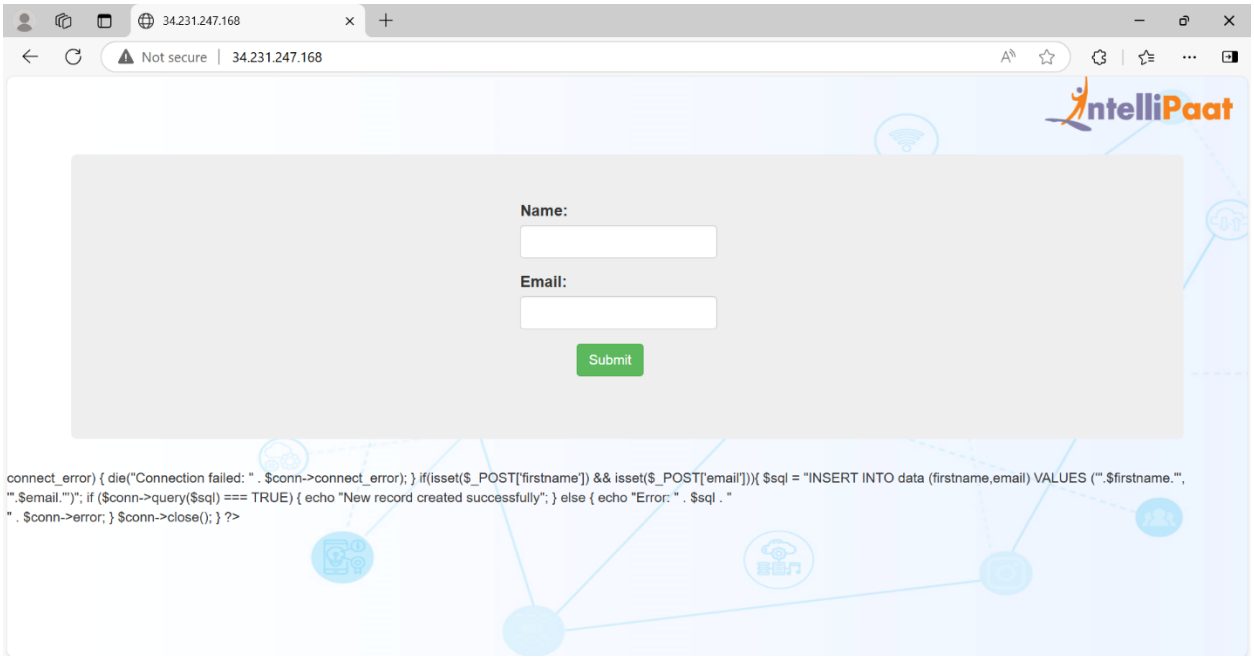
The company wants high availability on this product, therefore wants Auto Scaling to be enabled on this website

Solution:

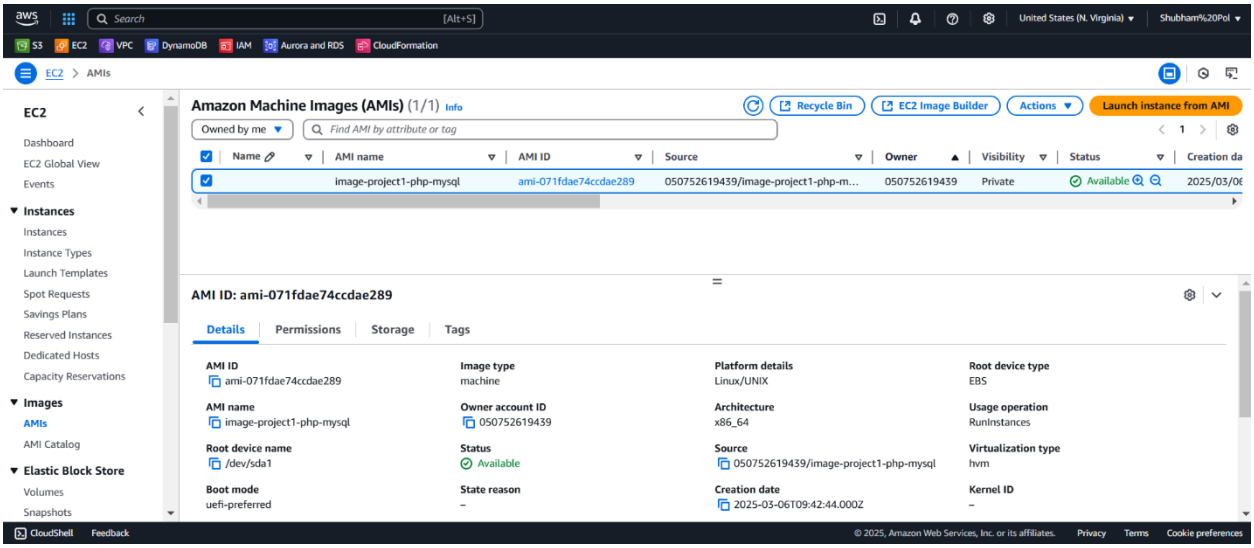
Steps To Solve:

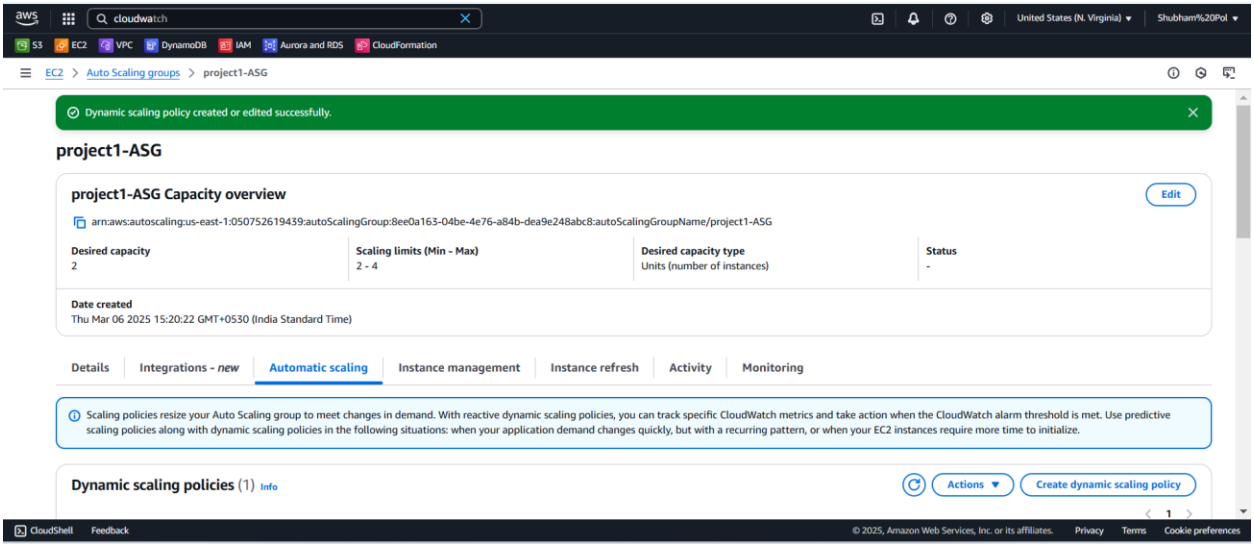
1. Launch an EC2 Instance

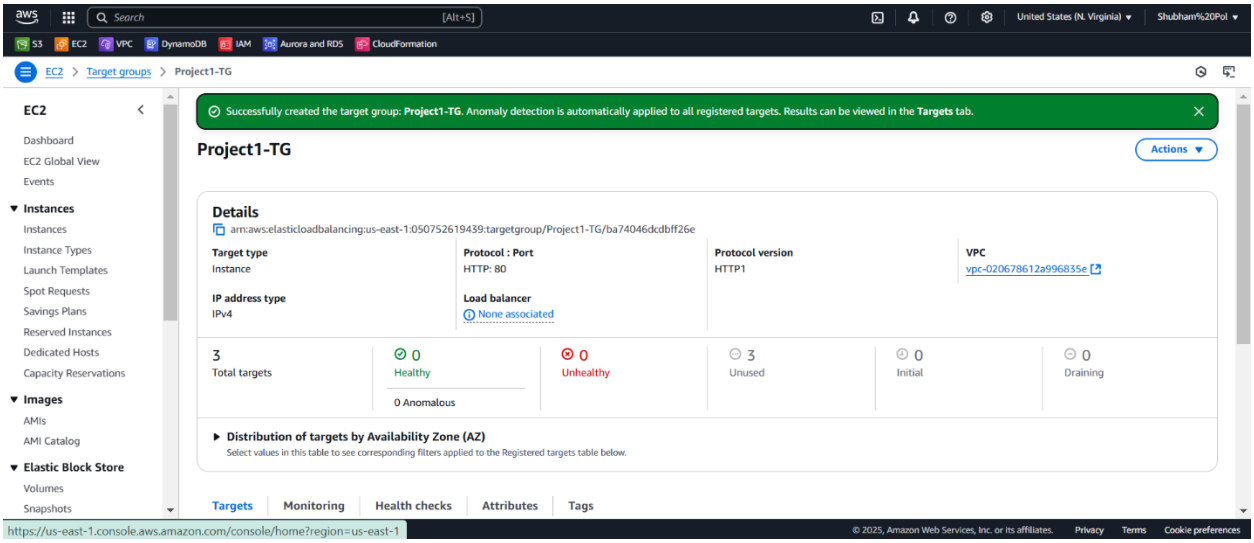


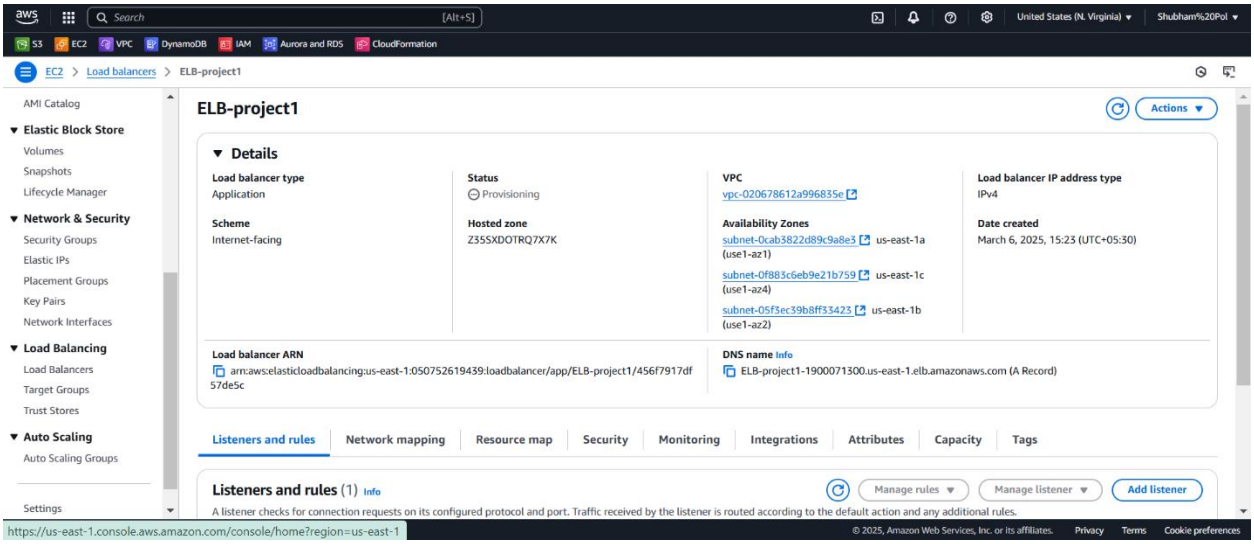


2. Enable Auto Scaling on these instances (minimum 2)

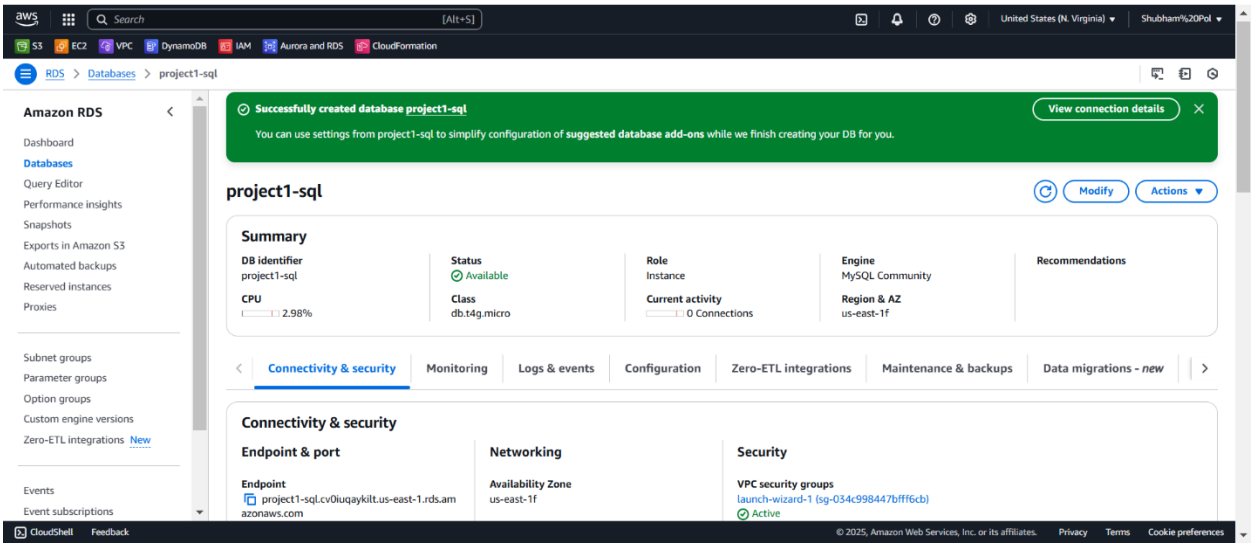



cloudwatch
EC2 > Auto Scaling groups > project1-ASG
Dynamic scaling policy created or edited successfully.
project1-ASG
project1-ASG Capacity overview
Edit
arn:aws:autoscaling:us-east-1:050752619439:autoScalingGroup:8ee0a163-04be-4e76-a84b-dea9e248abc8:autoScalingGroupName/project1-ASG
Desired capacity
2
Scaling limits (Min - Max)
2 - 4
Desired capacity type
Units (number of instances)
Status
-
Date created
Thu Mar 06 2025 15:20:22 GMT+0530 (India Standard Time)
Details
Integrations - new
Automatic scaling
Instance management
Instance refresh
Activity
Monitoring
Scaling policies resize your Auto Scaling group to meet changes in demand. With reactive dynamic scaling policies, you can track specific CloudWatch metrics and take action when the CloudWatch alarm threshold is met. Use predictive scaling policies along with dynamic scaling policies in the following situations: when your application demand changes quickly, but with a recurring pattern, or when your EC2 instances require more time to initialize.
Dynamic scaling policies (1) Info
Actions
Create dynamic scaling policy
CloudShell
Feedback
© 2025, Amazon Web Services, Inc. or its affiliates.
Privacy
Terms
Cookie preferences


Search
EC2 > Target groups > Project1-TG
Successfully created the target group: Project1-TG. Anomaly detection is automatically applied to all registered targets. Results can be viewed in the Targets tab.
Project1-TG
Actions
Details
arn:aws:elasticloadbalancing:us-east-1:050752619439:targetgroup/Project1-TG/ba74046dcdbff26e
Target type
Instance
Protocol : Port
HTTP: 80
Protocol version
HTTP1
VPC
vpc-020678612a996835e
IP address type
IPv4
Load balancer
None associated
3
Total targets
0
Healthy
0 Anomalous
0
Unhealthy
3
Unused
0
Initial
0
Draining
Distribution of targets by Availability Zone (AZ)
Select values in this table to see corresponding filters applied to the Registered targets table below.
Targets
Monitoring
Health checks
Attributes
Tags
https://us-east-1.console.aws.amazon.com/console/home?region=us-east-1
© 2025, Amazon Web Services, Inc. or its affiliates.
Privacy
Terms
Cookie preferences


Search
EC2 > Load balancers > ELB-project1
ELB-project1
Actions
Details
Load balancer type
Application
Status
Provisioning
VPC
vpc-020678612a996835e
Load balancer IP address type
IPv4
Scheme
Internet-facing
Hosted zone
Z35SXDOTRQ7X7K
Availability Zones
subnet-0cab3822d89c9a8e3 us-east-1a (use1-az1)
subnet-0f883c6eb9e21b759 us-east-1c (use1-az4)
subnet-05f3ec39b8ff33423 us-east-1b (use1-az2)
Date created
March 6, 2025, 15:23 (UTC+05:30)
Load balancer ARN
arn:aws:elasticloadbalancing:us-east-1:050752619439:loadbalancer/app/ELB-project1/456f7917df57de5c
DNS name Info
ELB-project1-1900071300.us-east-1.elb.amazonaws.com (A Record)
Listeners and rules
Network mapping
Resource map
Security
Monitoring
Integrations
Attributes
Capacity
Tags
Listeners and rules (1) Info
Manage rules
Manage listener
Add listener
A listener checks for connection requests on its configured protocol and port. Traffic received by the listener is routed according to the default action and any additional rules.
https://us-east-1.console.aws.amazon.com/console/home?region=us-east-1
© 2025, Amazon Web Services, Inc. or its affiliates.
Privacy
Terms
Cookie preferences

3. Create an RDS Instance



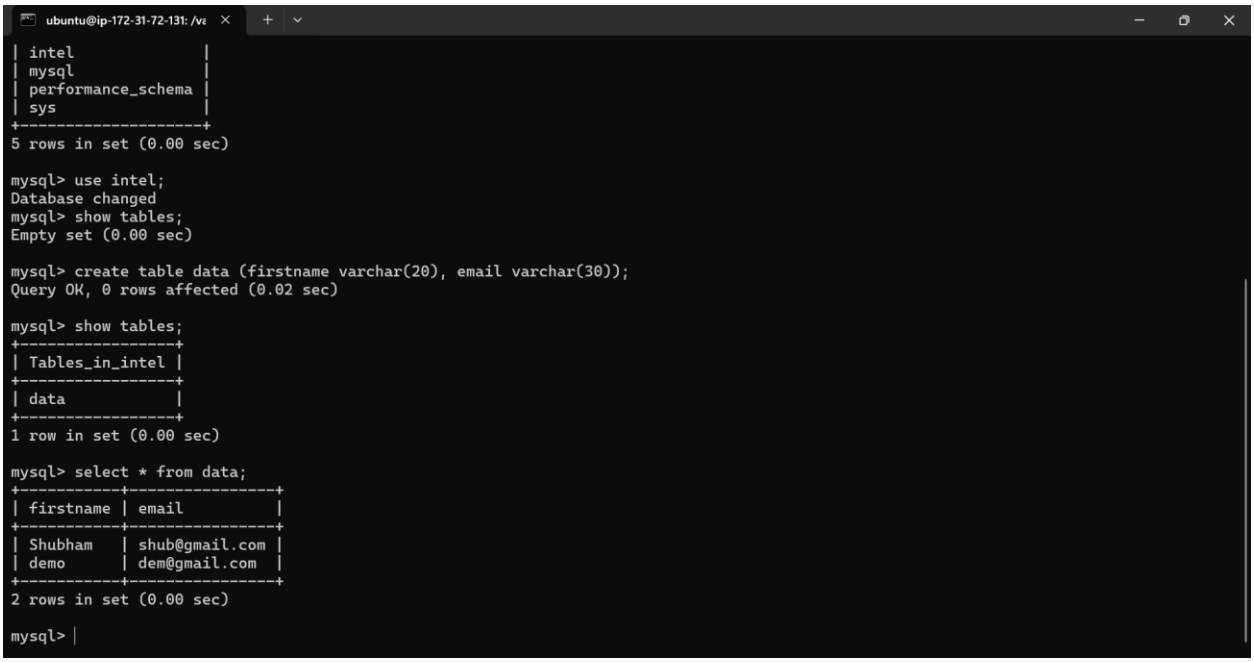
4. Create Database & Table in RDS instance:

a. Database name: intel

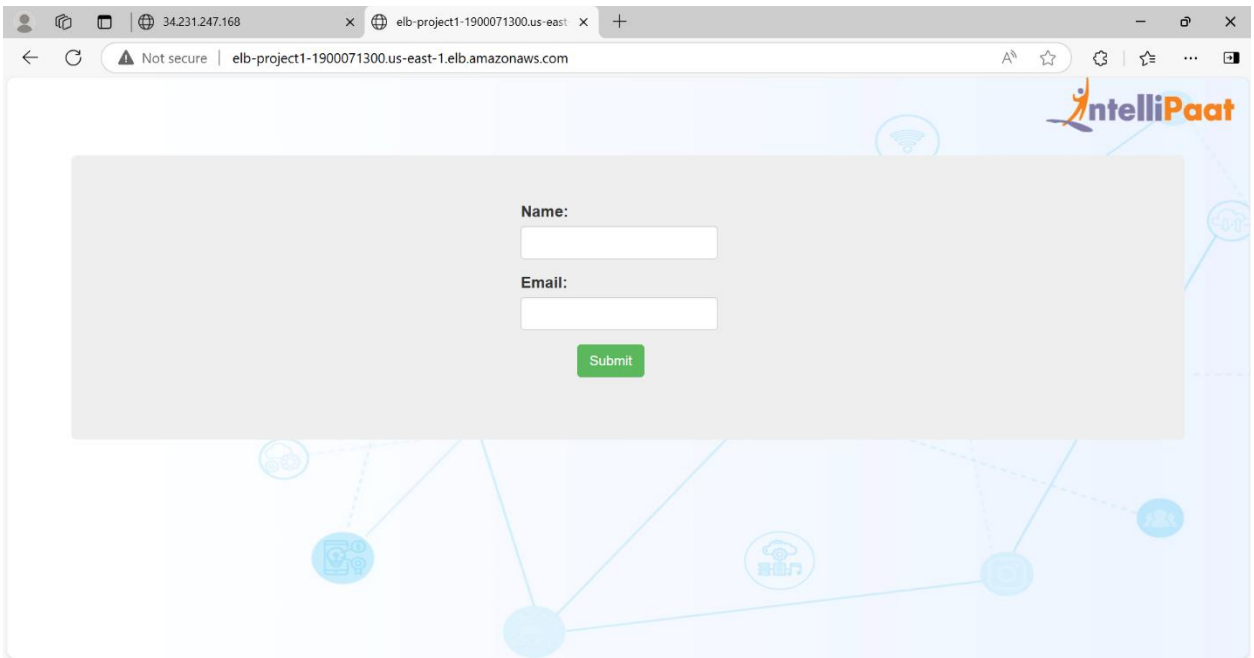
b. Table name: data

c. Database password: intel123

```
ubuntu@ip-172-31-72-131: /va

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

mysql> show databases
    -> ;
+--------------------+
| Database           |
+--------------------+
| information_schema |
| intel              |
| mysql              |
| performance_schema |
| sys                |
+--------------------+
5 rows in set (0.00 sec)

mysql> use intel;
Database changed
mysql> show tables;
Empty set (0.00 sec)

mysql> create table data (firstname varchar(20), email varchar(30));
Query OK, 0 rows affected (0.02 sec)

mysql> show tables;
+-----------------+
| Tables_in_intel |
+-----------------+
| data            |
+-----------------+
1 row in set (0.00 sec)

mysql>
```



5. Using ELB hosted the webpage



6. Modified the source code to see traffic is served by multiple server