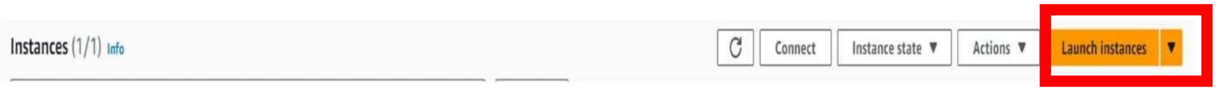
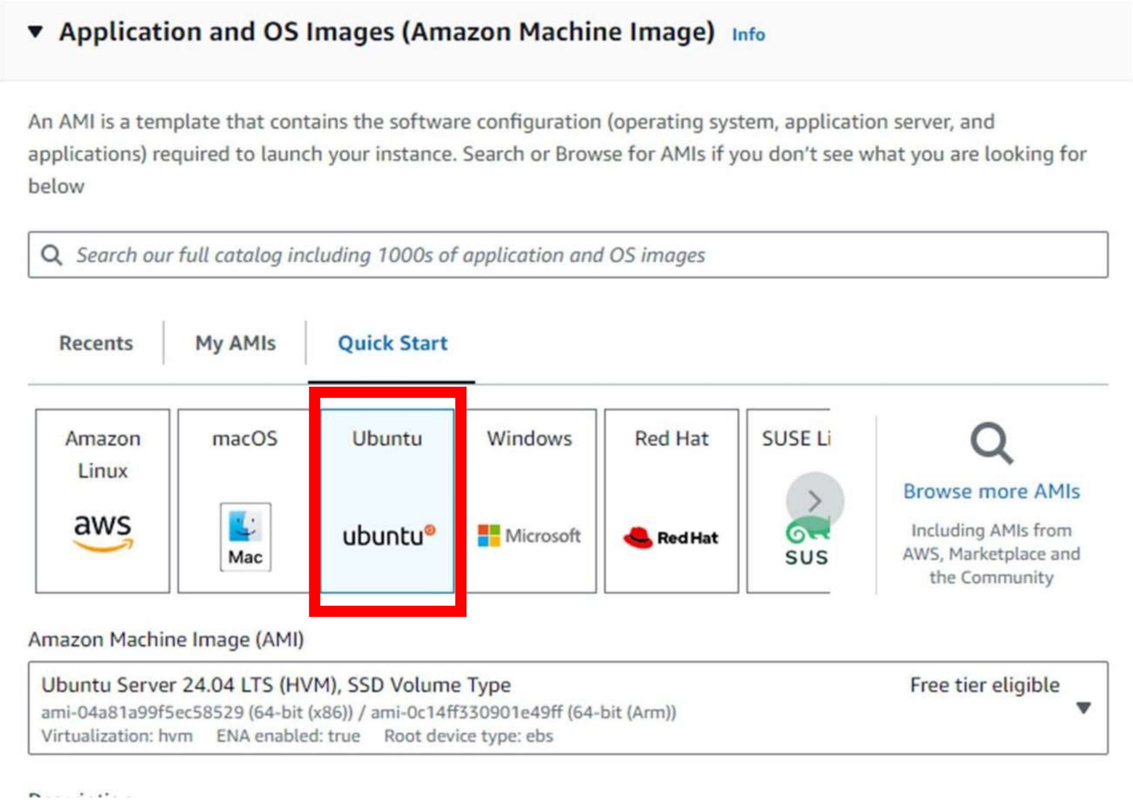
Step 1: Launch an EC2 Instance

1. **Log in to AWS Management Console**:
   - Go to the AWS Management Console at https://aws.amazon.com/console/
   - Sign in with your AWS credentials.
2. **Navigate to EC2 Dashboard**:
   - In the AWS Management Console, type "EC2" in the search bar and select EC2 to navigate to the EC2 Dashboard.
3. **Launch an Instance**:
   - Click on the "Launch Instance" button.



   - Choose an Amazon Machine Image (AMI): Select "Ubuntu Server 20.04 LTS (HVM), SSD Volume Type".



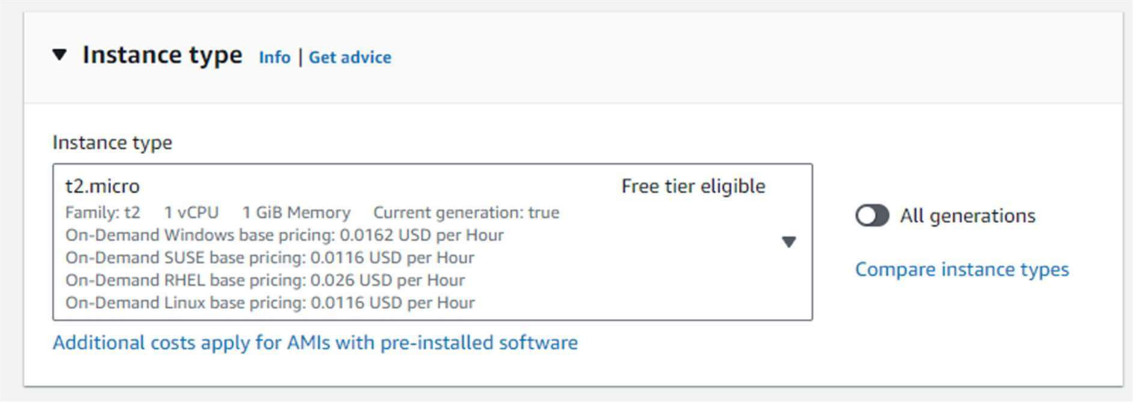
An AMI is a template that contains the software configuration (operating system, application server, and applications) required to launch your instance. Search or Browse for AMIs if you don't see what you are looking for below

Search our full catalog including 1000s of application and OS images

Amazon Machine Image (AMI)

Ubuntu Server 24.04 LTS (HVM), SSD Volume Type — Free tier eligible
ami-04a81a99f5ec58529 (64-bit (x86)) / ami-0c14ff330901e49ff (64-bit (Arm))
Virtualization: hvm ENA enabled: true Root device type: ebs

- Choose an Instance Type: Select `t2.micro` (eligible for the free tier).



▼ Instance type Info | Get advice

Instance type

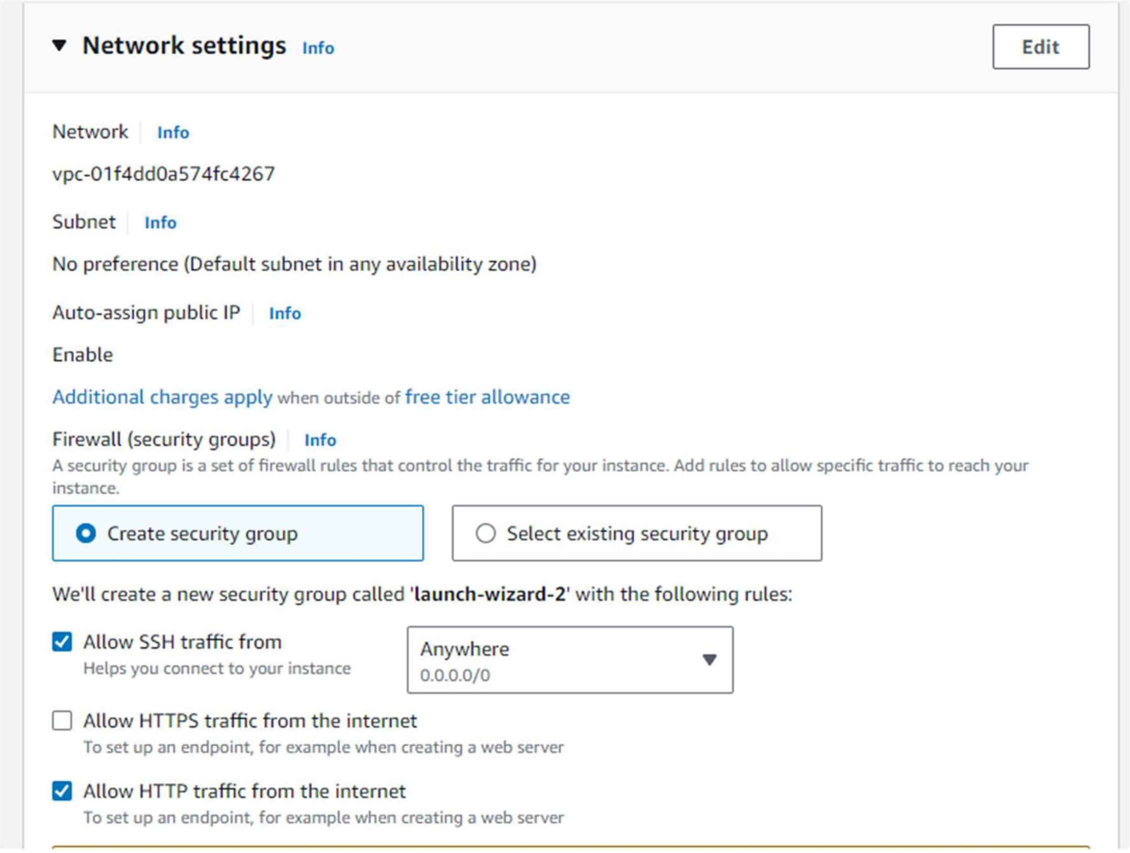
t2.micro — Free tier eligible
Family: t2 1 vCPU 1 GiB Memory Current generation: true
On-Demand Windows base pricing: 0.0162 USD per Hour
On-Demand SUSE base pricing: 0.0116 USD per Hour
On-Demand RHEL base pricing: 0.026 USD per Hour
On-Demand Linux base pricing: 0.0116 USD per Hour

All generations

Compare instance types

Additional costs apply for AMIs with pre-installed software

- Configure Instance:
  - Select an existing key pair or create a new one.
  - Network: Choose the default VPC.
  - Subnet: Choose a subnet in the US-East-1 (N. Virginia) region.
  - Enable Auto-assign Public IP.



▼ Network settings Info — Edit

Network Info
vpc-01f4dd0a574fc4267

Subnet Info
No preference (Default subnet in any availability zone)

Auto-assign public IP Info
Enable

Additional charges apply when outside of free tier allowance

Firewall (security groups) Info
A security group is a set of firewall rules that control the traffic for your instance. Add rules to allow specific traffic to reach your instance.

● Create security group ○ Select existing security group

We'll create a new security group called '**launch-wizard-2**' with the following rules:

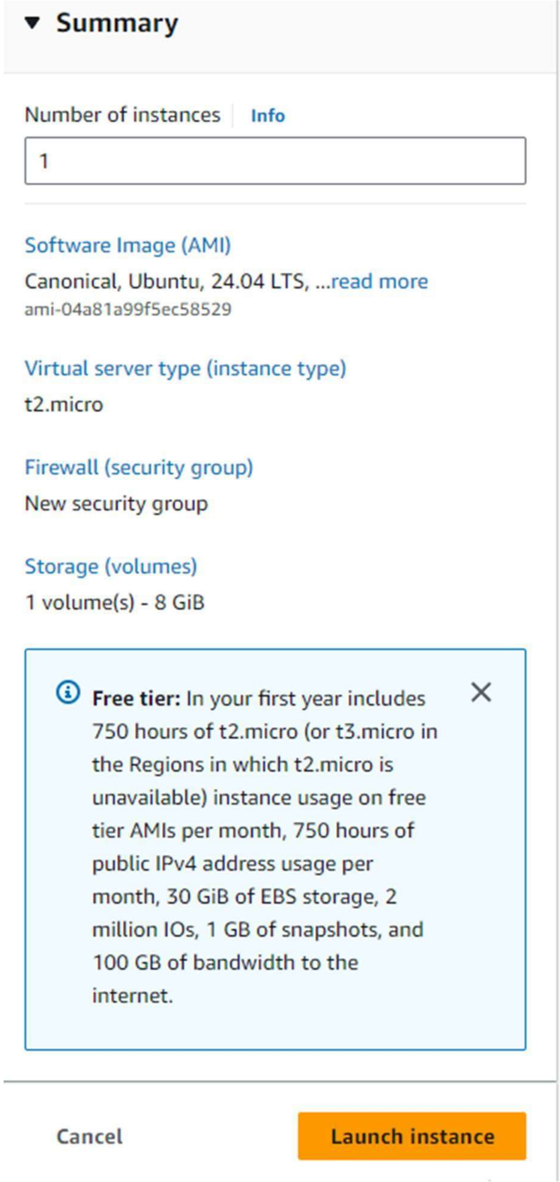
☑ Allow SSH traffic from — Anywhere 0.0.0.0/0
Helps you connect to your instance

□ Allow HTTPS traffic from the internet
To set up an endpoint, for example when creating a web server

☑ Allow HTTP traffic from the internet
To set up an endpoint, for example when creating a web server

- Add Storage: Keep the default settings.
- Add Tags: Add a tag to identify your instance (e.g., Key: Name, Value: Nginx).

4. **Review and Launch**:
   - Review your instance settings and click "Launch".



Number of instances | Info

1

Software Image (AMI)

Canonical, Ubuntu, 24.04 LTS, ...read more
ami-04a81a99f5ec58529

Virtual server type (instance type)

t2.micro

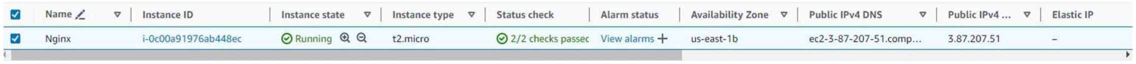
Firewall (security group)

New security group

Storage (volumes)

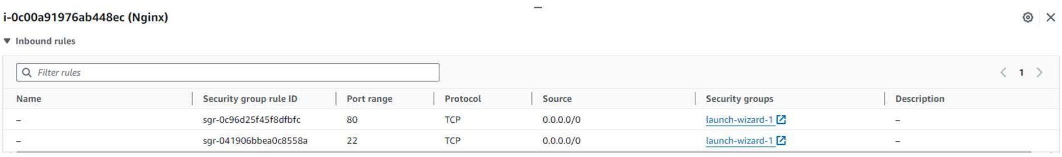
1 volume(s) - 8 GiB

Free tier: In your first year includes 750 hours of t2.micro (or t3.micro in the Regions in which t2.micro is unavailable) instance usage on free tier AMIs per month, 750 hours of public IPv4 address usage per month, 30 GiB of EBS storage, 2 million IOs, 1 GB of snapshots, and 100 GB of bandwidth to the internet.



☑	Name	Instance ID	Instance state	Instance type	Status check	Alarm status	Availability Zone	Public IPv4 DNS	Public IPv4 ...	Elastic IP
☑	Nginx	i-0c00a91976ab448ec	✓ Running	t2.micro	✓ 2/2 checks passec	View alarms +	us-east-1b	ec2-3-87-207-51.comp...	3.87.207.51	–

5. **Configure Security Group:**
    - Add a new security group with the following rules:
        - Type: HTTP, Protocol: TCP, Port Range: 80, Source: 0.0.0.0/0
        - Type: SSH, Protocol: TCP, Port Range: 22, Source: 0.0.0.0/0



i-0c00a91976ab448ec (Nginx)

▼ Inbound rules

Name	Security group rule ID	Port range	Protocol	Source	Security groups	Description
–	sgr-0c96d25f45f8dfbfc	80	TCP	0.0.0.0/0	launch-wizard-1	–
–	sgr-041906bbea0c8558a	22	TCP	0.0.0.0/0	launch-wizard-1	–

Step 2: Connect to Your Instance

1. **Connect to the EC2 Instance**:
    - In the EC2 Dashboard, select your instance.
    - Click on "Connect" and follow the instructions to connect to your instanceusing SSH.

Step 3: Install Apache and PHP

1. **Update the package index**:

    sudo apt update -y

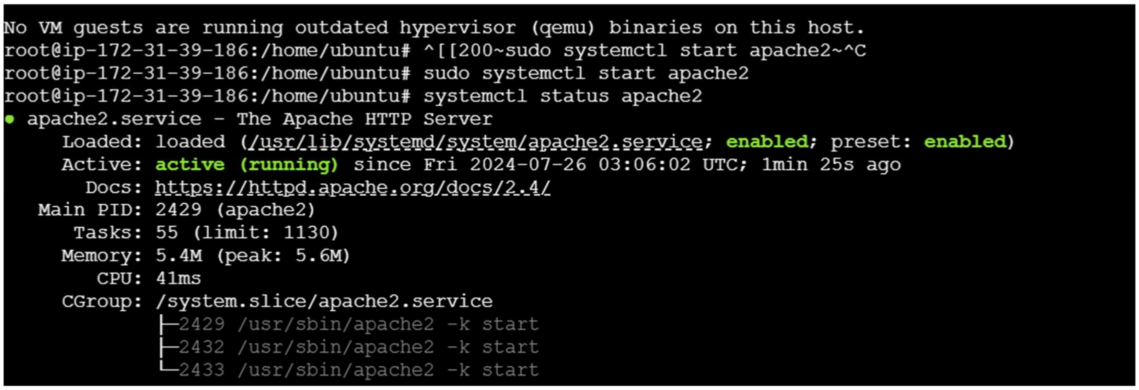
2. **Install Apache**:

    sudo apt install apache 2 -y

3. **Start Apache**:

    sudo systemctl start apache2

    sudo systemctl enable apache2



```
No VM guests are running outdated hypervisor (qemu) binaries on this host.
root@ip-172-31-39-186:/home/ubuntu# ^[[200~sudo systemctl start apache2~^C
root@ip-172-31-39-186:/home/ubuntu# sudo systemctl start apache2
root@ip-172-31-39-186:/home/ubuntu# systemctl status apache2
● apache2.service - The Apache HTTP Server
     Loaded: loaded (/usr/lib/systemd/system/apache2.service; enabled; preset: enabled)
     Active: active (running) since Fri 2024-07-26 03:06:02 UTC; 1min 25s ago
       Docs: https://httpd.apache.org/docs/2.4/
   Main PID: 2429 (apache2)
      Tasks: 55 (limit: 1130)
     Memory: 5.4M (peak: 5.6M)
        CPU: 41ms
     CGroup: /system.slice/apache2.service
             ├─2429 /usr/sbin/apache2 -k start
             ├─2432 /usr/sbin/apache2 -k start
             └─2433 /usr/sbin/apache2 -k start
```

4. **Install PHP**:

    sudo apt install php php-mysql -y

5. **Restart Apache** to apply PHP installation:

    sudo systemctl restart apache2

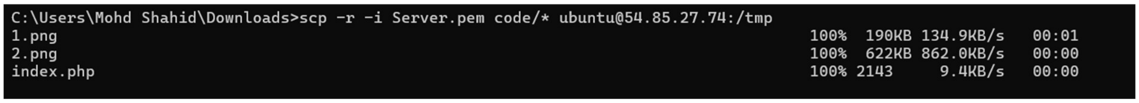
Step 4: Create an RDS Instance

1. **Navigate to RDS Dashboard**:
   - Click on **Create Database**.
   - Choose **Standard Create**.
   - Select **MySQL**.
   - Choose a DB instance class (e.g., db.t3.micro).
   - Set storage and other configurations.
   - In the **Settings** section:
     - DB instance identifier: my-rds-instance.
     - Master username: intel.
     - Master password: intel123.
   - Configure additional settings (VPC, subnet, security groups).
2. **Create the RDS instance**.

Step 5: Upload Website Files

1. **Upload your PHP website files** to the Apache document root:
   - Delete the default index file.
   - The default document root is /var/www/html/.
   - You can use SCP or any other method to transfer files. For example, using SCP:

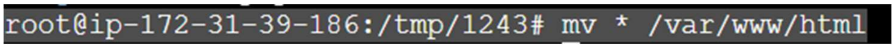
   scp -r -i your-key.pem path-to-your-local-files/* ec2-user@your-ec2-public-ip:/tmp



```
C:\Users\Mohd Shahid\Downloads>scp -r -i Server.pem code/* ubuntu@54.85.27.74:/tmp
1.png                                                                       100%  190KB 134.9KB/s   00:01
2.png                                                                       100%  622KB 862.0KB/s   00:00
index.php                                                                   100% 2143     9.4KB/s   00:00
```

   - Then move all the file into var/www/html

     mv * /var/www/html



```
root@ip-172-31-39-186:/tmp/1243# mv * /var/www/html
```

Step 6: Create Database & Table in RDS instance

1. **Connect to the RDS Instance**:
   - Obtain the endpoint from the RDS dashboard.
   - Connect MySQL:

     mysql -h <RDS_ENDPOINT> -u admin -p

2. **Create the Database and Table**:

   CREATE DATABASE intel;

   USE intel;

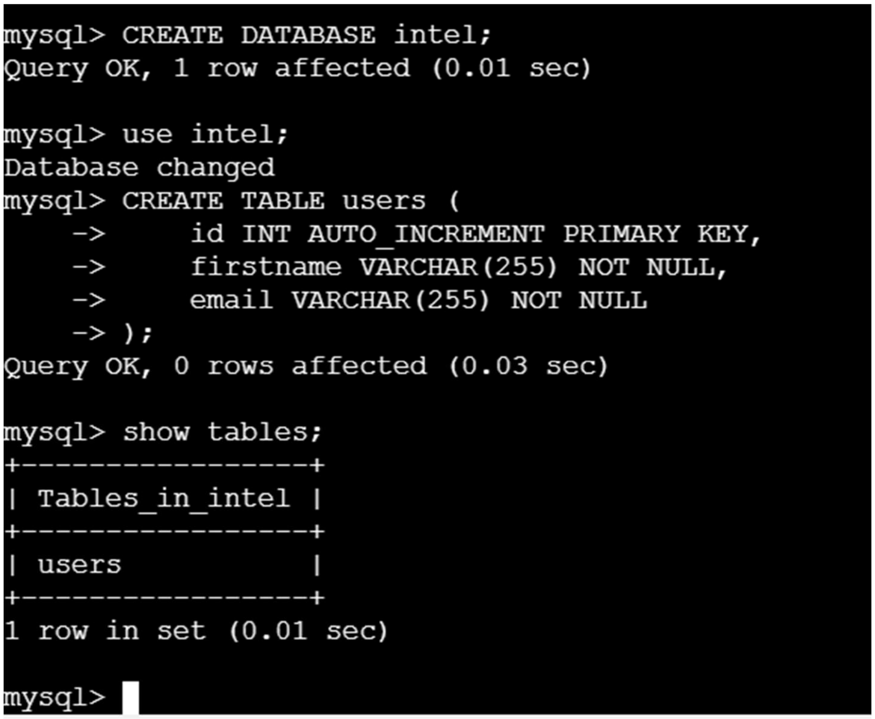
   CREATE TABLE users (

      id INT AUTO_INCREMENT PRIMARY KEY,

      firstname VARCHAR(255) NOT NULL,

      email VARCHAR(255) NOT NULL

   );

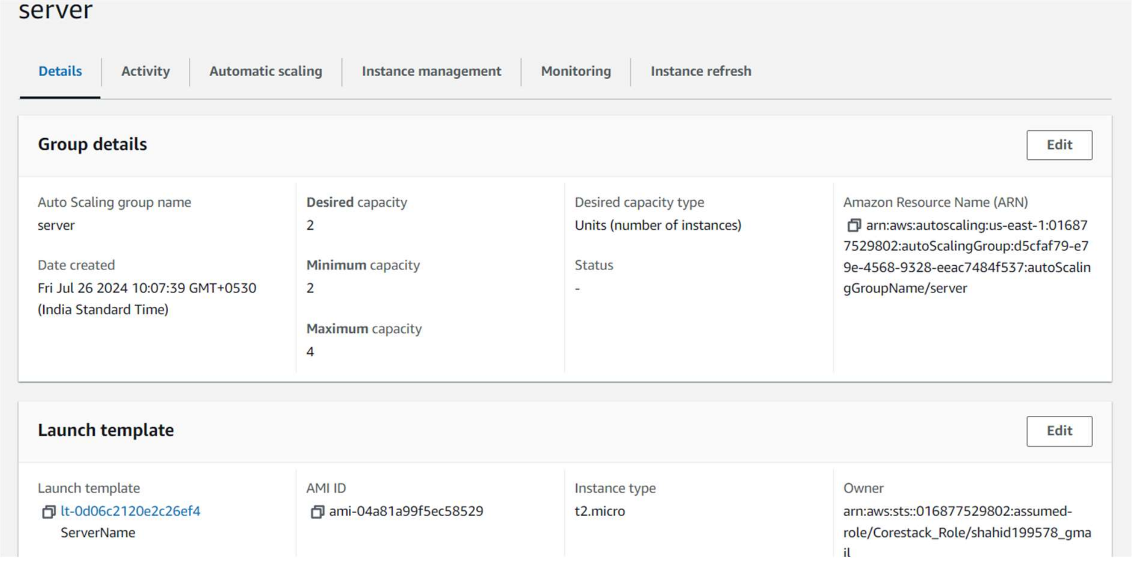


```
mysql> CREATE DATABASE intel;
Query OK, 1 row affected (0.01 sec)

mysql> use intel;
Database changed
mysql> CREATE TABLE users (
    ->     id INT AUTO_INCREMENT PRIMARY KEY,
    ->     firstname VARCHAR(255) NOT NULL,
    ->     email VARCHAR(255) NOT NULL
    -> );
Query OK, 0 rows affected (0.03 sec)

mysql> show tables;
+-----------------+
| Tables_in_intel |
+-----------------+
| users           |
+-----------------+
1 row in set (0.01 sec)

mysql> 
```

Step 7: Enable Auto Scaling on These Instances (Minimum 2)

1. **Create a Launch Template**:
   - Navigate to **Launch Templates** in the EC2 dashboard.
   - Click on **Create launch template**.
   - Fill in template details and instance configuration.
   - Ensure to use the same AMI, instance type, and security group as your manually launched instance.
2. **Create an Auto Scaling Group**:
   - Navigate to **Auto Scaling Groups**.
   - Click on **Create Auto Scaling group**.
   - Choose your launch template.

- Set the desired capacity to 2, minimum capacity to 2, and maximum capacity to 4.
- Configure network and subnets.
- Set up scaling policies (optional).



Group details

Auto Scaling group name
server

Date created
Fri Jul 26 2024 10:07:39 GMT+0530 (India Standard Time)

Desired capacity
2

Minimum capacity
2

Maximum capacity
4

Desired capacity type
Units (number of instances)

Status
-

Amazon Resource Name (ARN)
arn:aws:autoscaling:us-east-1:016877529802:autoScalingGroup:d5cfaf79-e79e-4568-9328-eeac7484f537:autoScalingGroupName/server

Launch template

Launch template
lt-0d06c2120e2c26ef4
ServerName

AMI ID
ami-04a81a99f5ec58529

Instance type
t2.micro

Owner
arn:aws:sts::016877529802:assumed-role/Corestack_Role/shahid199578_gmail

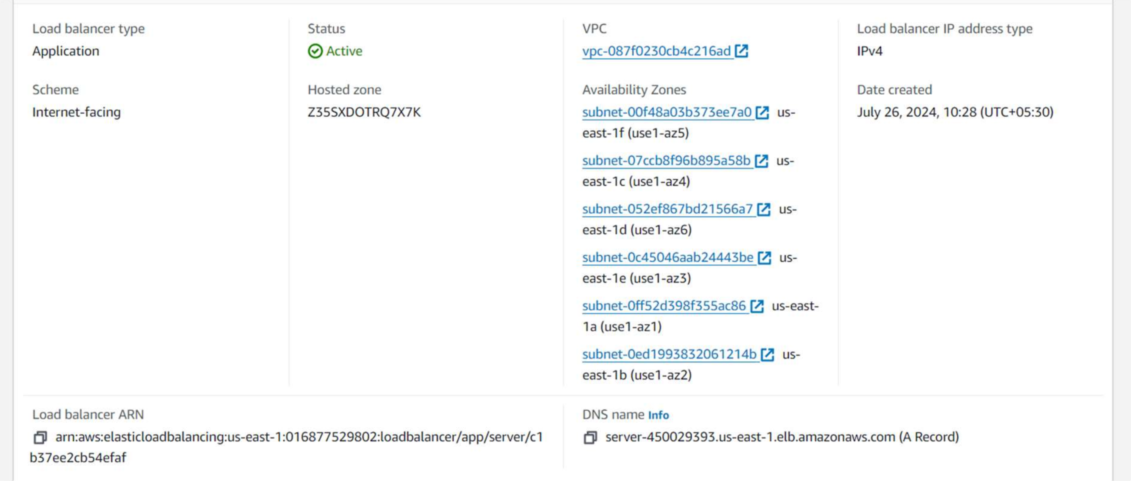
Step 8: Create a Load Balancer

1. **Navigate to the EC2 Dashboard**:
   - Click on **Load Balancers** under the Load Balancing section.
   - Click on **Create Load Balancer**.
   - Choose **Application Load Balancer**.
   - Configure the load balancer:
     - Name: my-load-balancer.
     - Scheme: Internet-facing.
     - Listeners: HTTP (port 80).
     - Availability Zones: Select the VPC and subnets.
2. **Configure Security Groups** for the load balancer:
   - Ensure it allows HTTP traffic.

3. **Configure Routing**:
   - Create a target group:
     - Name: my-target-group.
     - Target type: Instances.
     - Protocol: HTTP.

    - Port: 80.
    - Health checks: HTTP.
  - Register your instances in the target group.

4. **Review and Create** the load balancer.



Load balancer type
Application

Status
Active

VPC
vpc-087f0230cb4c216ad

Load balancer IP address type
IPv4

Scheme
Internet-facing

Hosted zone
Z35SXDOTRQ7X7K

Availability Zones
subnet-00f48a03b373ee7a0 us-east-1f (use1-az5)
subnet-07ccb8f96b895a58b us-east-1c (use1-az4)
subnet-052ef867bd21566a7 us-east-1d (use1-az6)
subnet-0c45046aab24443be us-east-1e (use1-az3)
subnet-0ff52d398f355ac86 us-east-1a (use1-az1)
subnet-0ed1993832061214b us-east-1b (use1-az2)

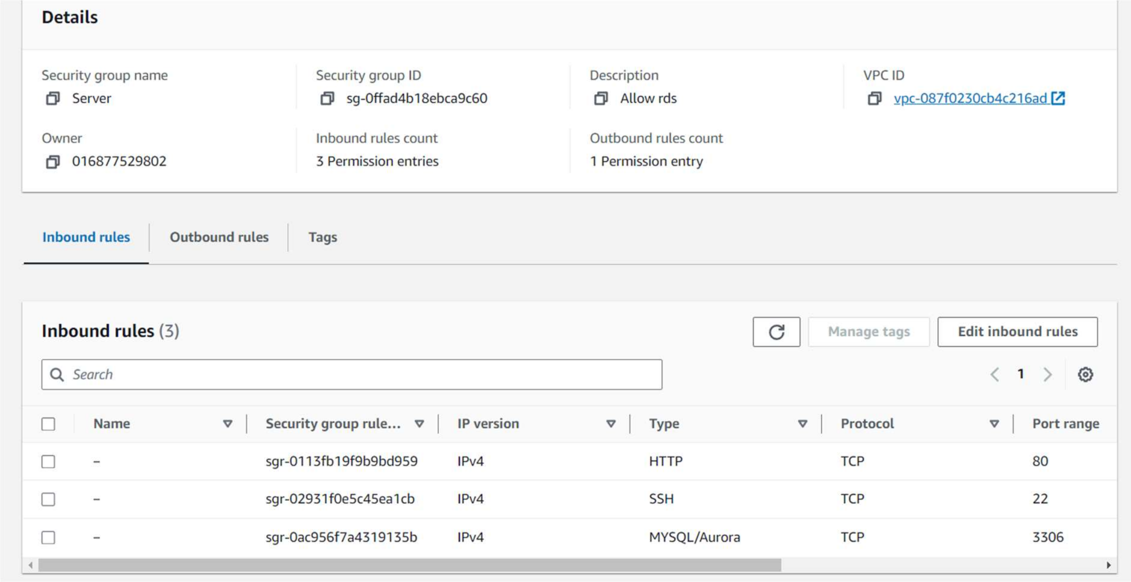
Date created
July 26, 2024, 10:28 (UTC+05:30)

Load balancer ARN
arn:aws:elasticloadbalancing:us-east-1:016877529802:loadbalancer/app/server/c1b37ee2cb54efaf

DNS name Info
server-450029393.us-east-1.elb.amazonaws.com (A Record)

Step 9: Allow Traffic from EC2 to RDS Instance

1. **Modify RDS Security Group**:
   - Go to the **RDS dashboard**, select your instance.
   - Click on **Modify** > **Security Groups**.
   - Add a rule to allow inbound MySQL/Aurora traffic (port 3306) from the EC2 instance's security group.



Details

Security group name
Server

Security group ID
sg-0ffad4b18ebca9c60

Description
Allow rds

VPC ID
vpc-087f0230cb4c216ad

Owner
016877529802

Inbound rules count
3 Permission entries

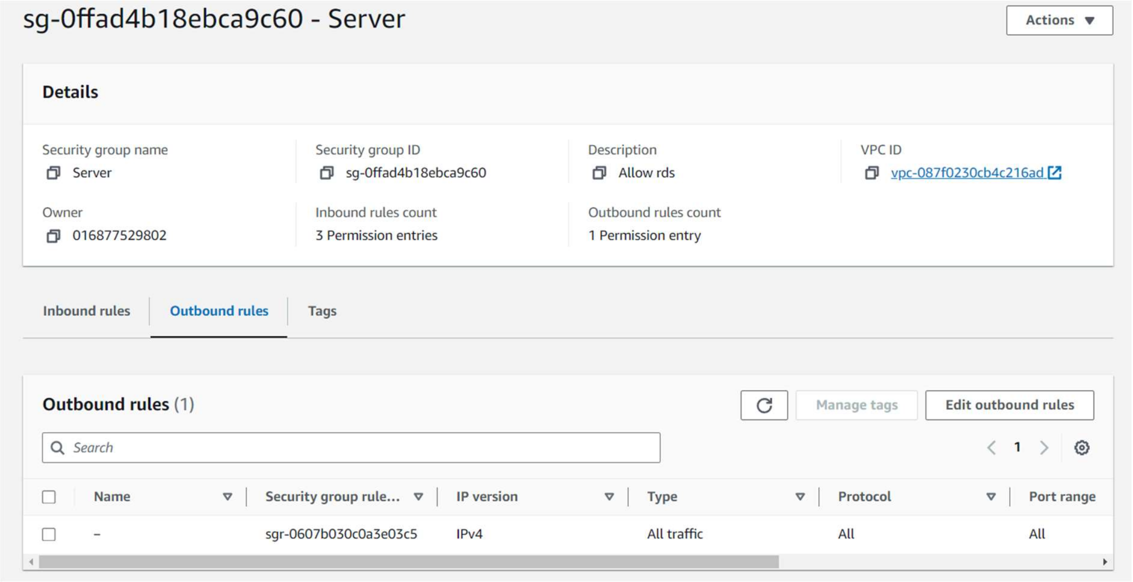
Outbound rules count
1 Permission entry

Inbound rules (3)

□	Name	Security group rule...	IP version	Type	Protocol	Port range
□	–	sgr-0113fb19f9b9bd959	IPv4	HTTP	TCP	80
□	–	sgr-02931f0e5c45ea1cb	IPv4	SSH	TCP	22
□	–	sgr-0ac956f7a4319135b	IPv4	MYSQL/Aurora	TCP	3306

2. Allow All Traffic to EC2 Instance

1. **Modify EC2 Security Group**:
   - Go to the **EC2 dashboard**, select your instance.
   - Click on **Security Groups**.
   - Edit inbound rules to allow all traffic:
     - Custom TCP Rule, Source: 0.0.0.0/0 (All traffic).



sg-0ffad4b18ebca9c60 - Server

Details

Security group name: Server

Security group ID: sg-0ffad4b18ebca9c60

Description: Allow rds

VPC ID: vpc-087f0230cb4c216ad

Owner: 016877529802

Inbound rules count: 3 Permission entries

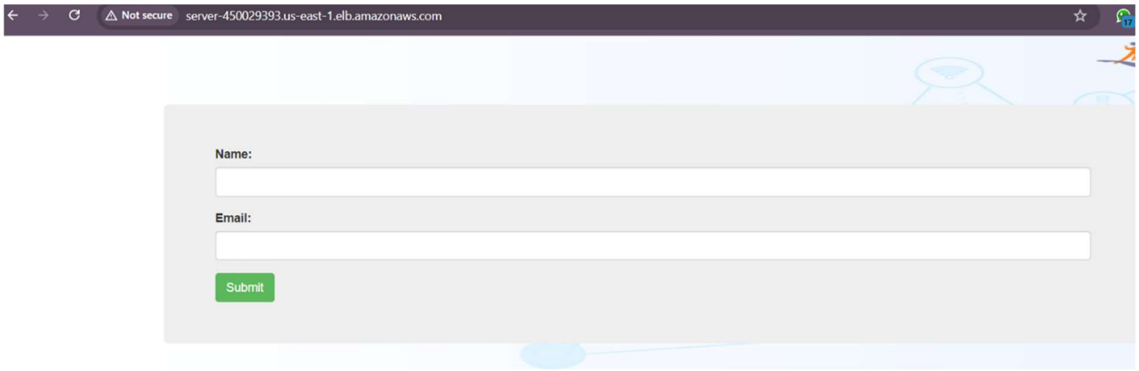
Outbound rules count: 1 Permission entry

Outbound rules (1)

□	Name	Security group rule...	IP version	Type	Protocol	Port range
□	–	sgr-0607b030c0a3e03c5	IPv4	All traffic	All	All

Step 11: Final Steps

1. **Test the Configuration**:
   - Ensure the website is accessible via the domain name.



Not secure server-450029393.us-east-1.elb.amazonaws.com

Name:

Email: