Project - 1: Deploying a Multi-Tier Website Using AWS EC2

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

Steps To Solve:

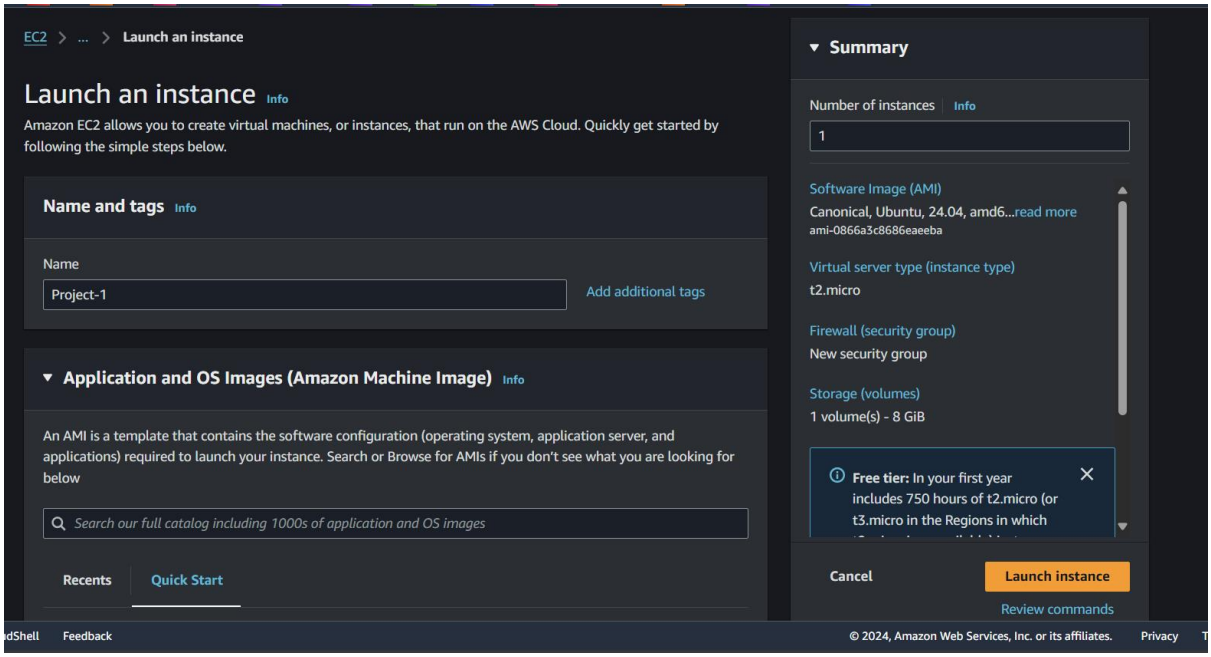
1. Launch an EC2 Instance
2. Enable Auto Scaling on these instances (minimum 2)
3. Create an RDS Instance
4. Create Database & Table in RDS instance:

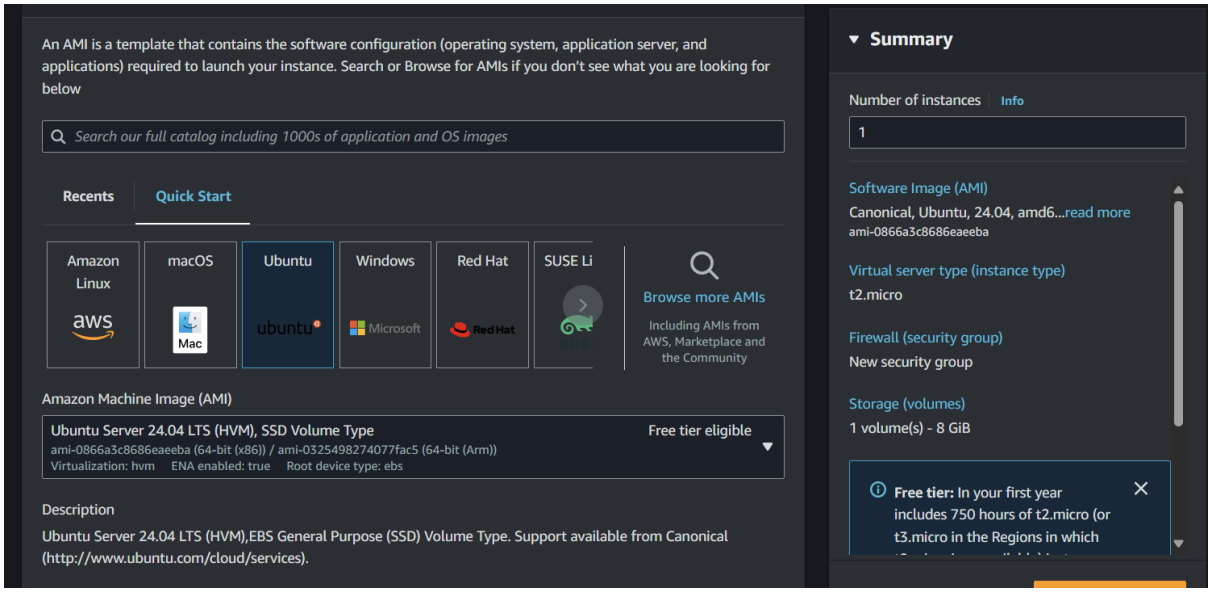
a. Database name: intel

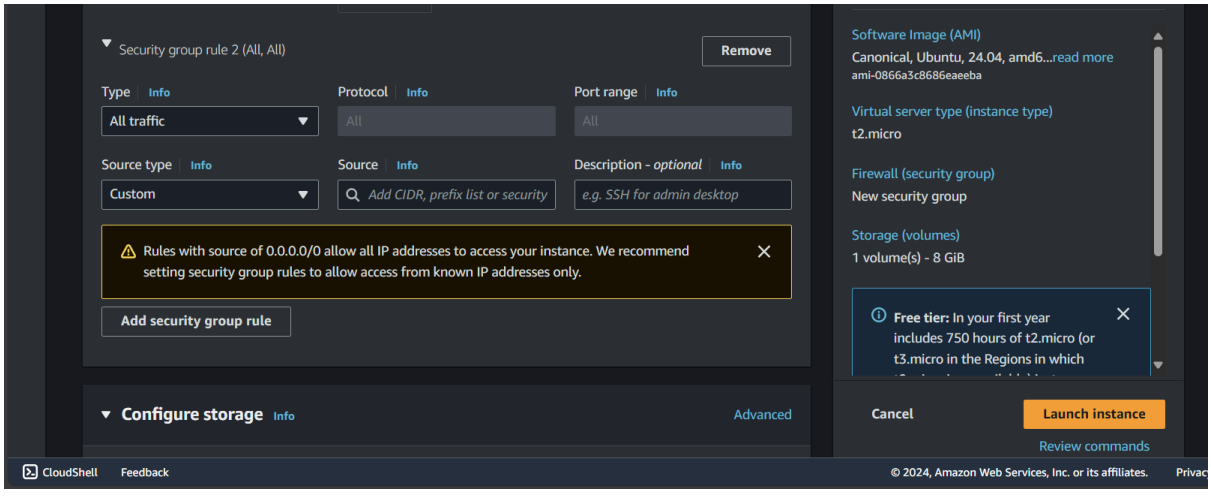
b. Table name: data

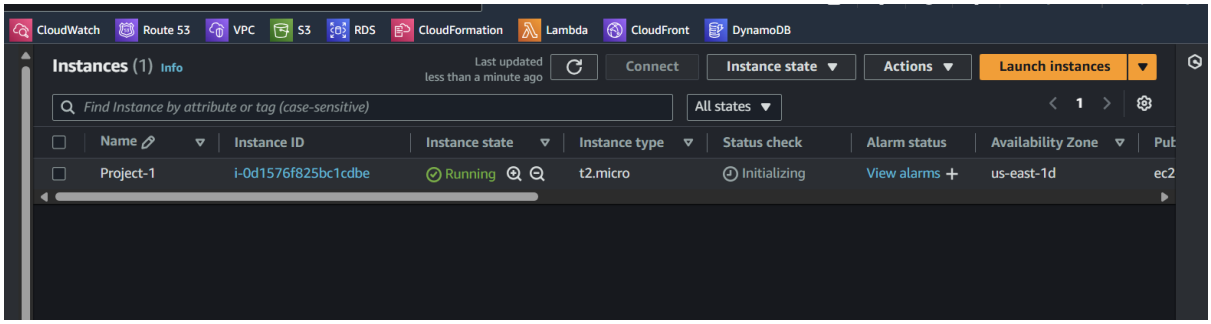
c. Database password: intel123

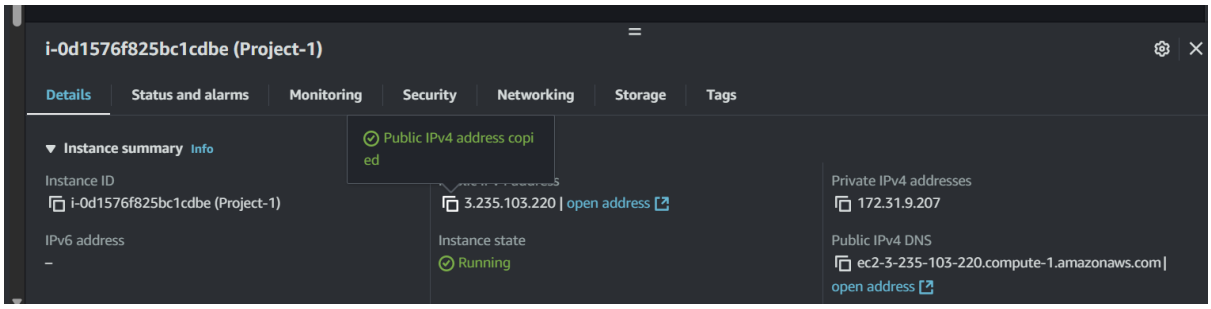
5. Change hostname in website
6. Allow traffic from EC2 to RDS instance
7. Allow all-traffic to EC2 instance

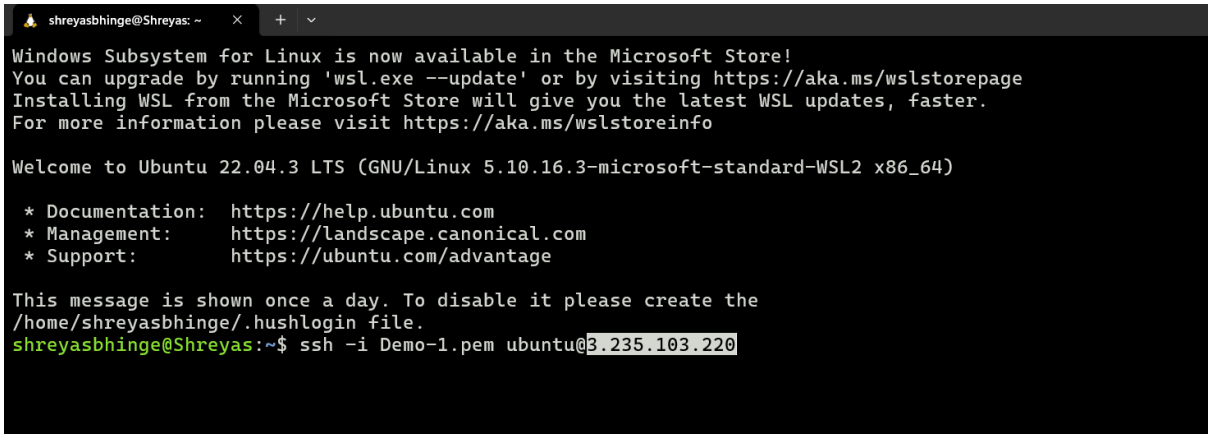

EC2 > ... > Launch an instance
Launch an instance Info
Amazon EC2 allows you to create virtual machines, or instances, that run on the AWS Cloud. Quickly get started by following the simple steps below.
Name and tags Info
Name
Project-1
Add additional tags
Application and OS Images (Amazon Machine Image) Info
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
Recents
Quick Start
Summary
Number of instances Info
1
Software Image (AMI)
Canonical, Ubuntu, 24.04, amd6...read more
ami-0866a3c8686eaeeba
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
Cancel
Launch instance
Review commands
dShell
Feedback
© 2024, Amazon Web Services, Inc. or its affiliates.
Privacy


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
Recents
Quick Start
Amazon Linux
macOS
Ubuntu
Windows
Red Hat
SUSE Li
Browse more AMIs
Including AMIs from AWS, Marketplace and the Community
Amazon Machine Image (AMI)
Ubuntu Server 24.04 LTS (HVM), SSD Volume Type
Free tier eligible
ami-0866a3c8686eaeeba (64-bit (x86)) / ami-0325498274077fac5 (64-bit (Arm))
Virtualization: hvm ENA enabled: true Root device type: ebs
Description
Ubuntu Server 24.04 LTS (HVM),EBS General Purpose (SSD) Volume Type. Support available from Canonical (http://www.ubuntu.com/cloud/services).
Summary
Number of instances Info
1
Software Image (AMI)
Canonical, Ubuntu, 24.04, amd6...read more
ami-0866a3c8686eaeeba
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


Security group rule 2 (All, All)
Remove
Type Info
All traffic
Protocol Info
All
Port range Info
All
Source type Info
Custom
Source Info
Add CIDR, prefix list or security
Description - optional Info
e.g. SSH for admin desktop
Rules with source of 0.0.0.0/0 allow all IP addresses to access your instance. We recommend setting security group rules to allow access from known IP addresses only.
Add security group rule
Configure storage Info
Advanced
Software Image (AMI)
Canonical, Ubuntu, 24.04, amd6...read more
ami-0866a3c8686eaeeba
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
Cancel
Launch instance
Review commands
CloudShell
Feedback
© 2024, Amazon Web Services, Inc. or its affiliates.







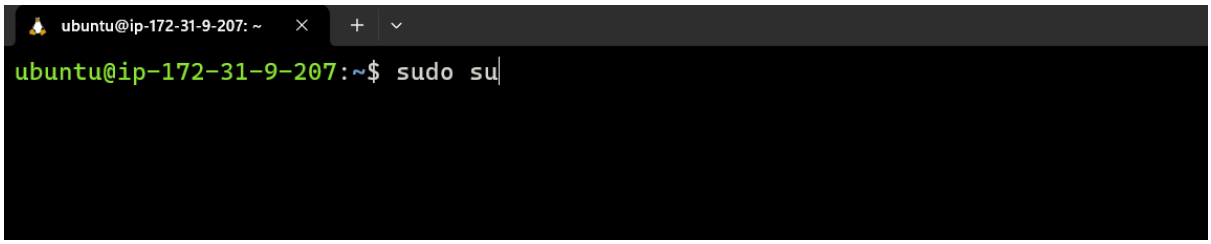
1. sudo su ---> to convert into root user
2. apt-get update ----> to get latest packages which are installed in ubuntu
3. apt install apache2 -y ---> to install web server
4. systemctl status apache2 ---> to check the satus of your web server
5. cd /var/www/html
6. rm index.html
7. nano index.php

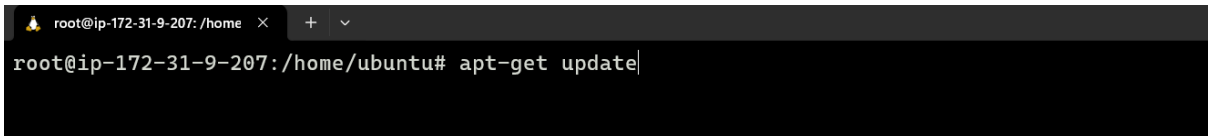
---> copy the contents of index.php into the terminal

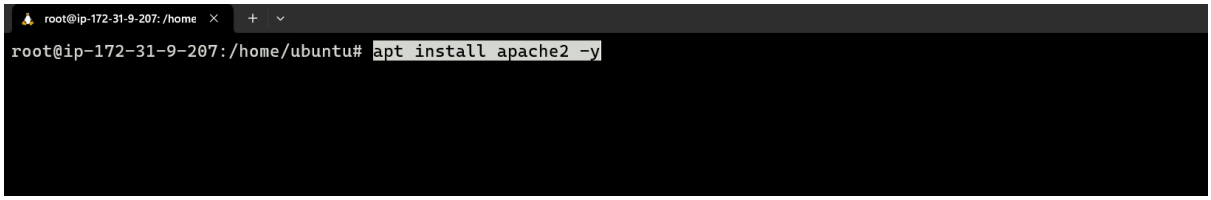
---> ctl + s -> to save the file

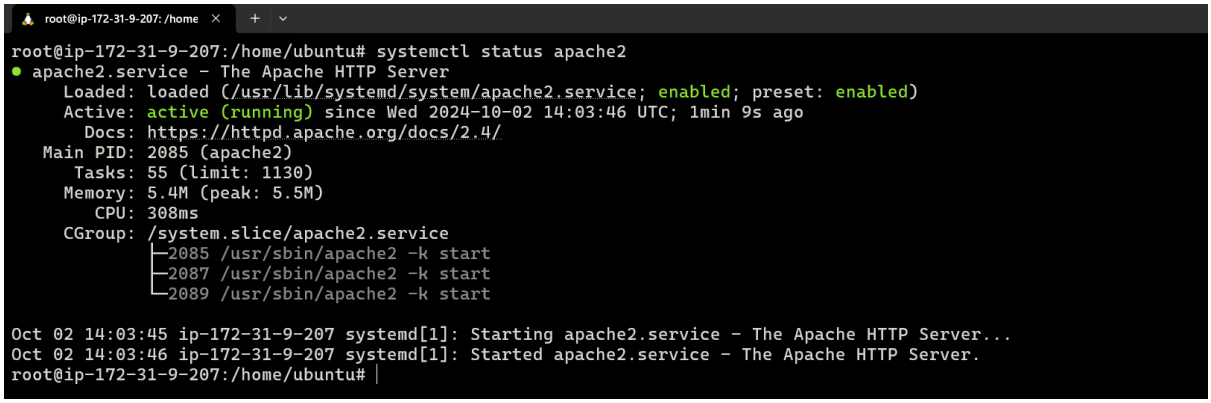
---> ctl + x -> exit the editor

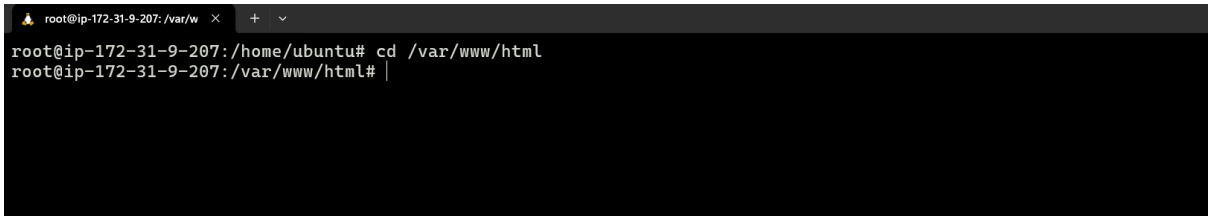
8. apt-add-repository -y PPa:ondrej/php ----> to add the repository into ubuntu instance which enables us to install the requrired php version
9. apt install php5.6 mysql-client php5.6-mysqli -y ---> installing the required packages

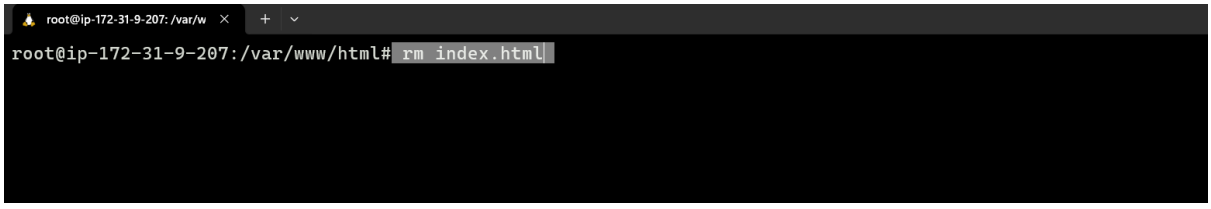

ubuntu@ip-172-31-9-207: ~
ubuntu@ip-172-31-9-207:~$ sudo su


root@ip-172-31-9-207: /home
root@ip-172-31-9-207:/home/ubuntu# apt-get update

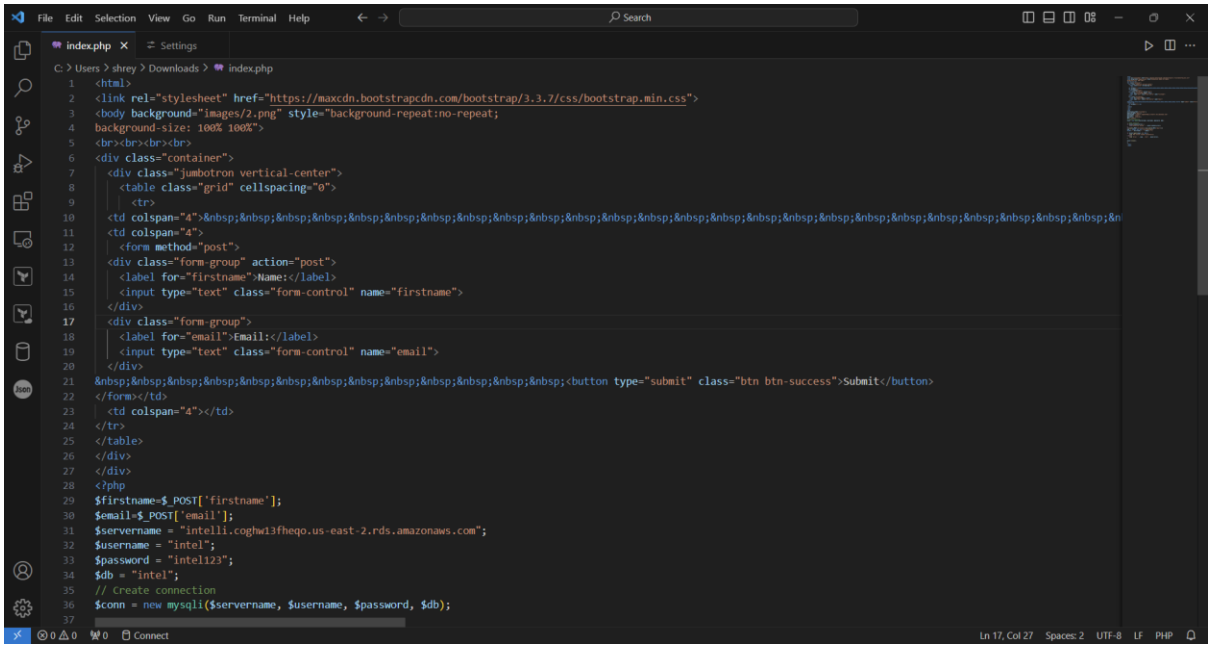

root@ip-172-31-9-207: /home
root@ip-172-31-9-207:/home/ubuntu# apt install apache2 -y


root@ip-172-31-9-207: /home
root@ip-172-31-9-207:/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 Wed 2024-10-02 14:03:46 UTC; 1min 9s ago
Docs: https://httpd.apache.org/docs/2.4/
Main PID: 2085 (apache2)
Tasks: 55 (limit: 1130)
Memory: 5.4M (peak: 5.5M)
CPU: 308ms
CGroup: /system.slice/apache2.service
├─2085 /usr/sbin/apache2 -k start
├─2087 /usr/sbin/apache2 -k start
└─2089 /usr/sbin/apache2 -k start
Oct 02 14:03:45 ip-172-31-9-207 systemd[1]: Starting apache2.service - The Apache HTTP Server...
Oct 02 14:03:46 ip-172-31-9-207 systemd[1]: Started apache2.service - The Apache HTTP Server.
root@ip-172-31-9-207:/home/ubuntu#


root@ip-172-31-9-207: /var/w
root@ip-172-31-9-207:/home/ubuntu# cd /var/www/html
root@ip-172-31-9-207:/var/www/html#


root@ip-172-31-9-207: /var/w
root@ip-172-31-9-207:/var/www/html# rm index.html

root@ip-172-31-9-207: /var/w
root@ip-172-31-9-207:/var/www/html# nano index.php


File Edit Selection View Go Run Terminal Help
Search
index.php
Settings
C: > Users > shrey > Downloads > index.php
<html>
<link rel="stylesheet" href="https://maxcdn.bootstrapcdn.com/bootstrap/3.3.7/css/bootstrap.min.css">
<body background="images/2.png" style="background-repeat:no-repeat;
background-size: 100% 100%">

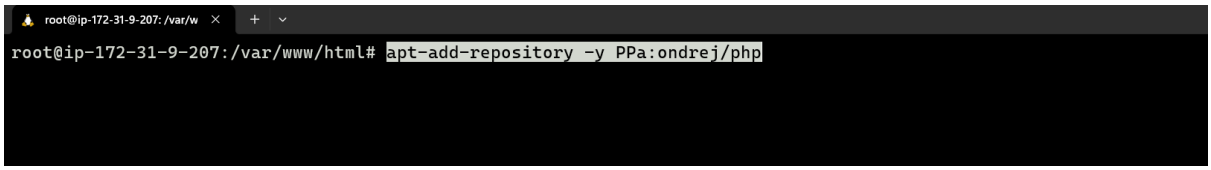
<table class="grid" cellspacing="0">
<tr>
<td colspan="4">   &n
<td colspan="4">
<form method="post">

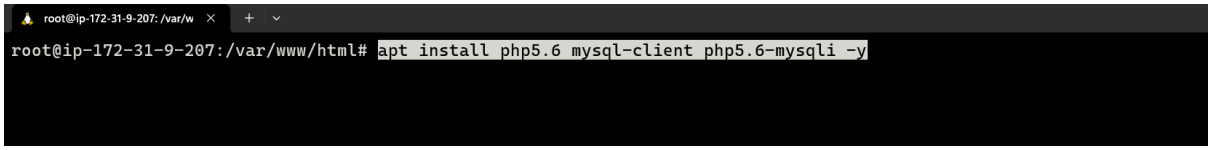
<label for="firstname">Name:</label>
<input type="text" class="form-control" name="firstname">

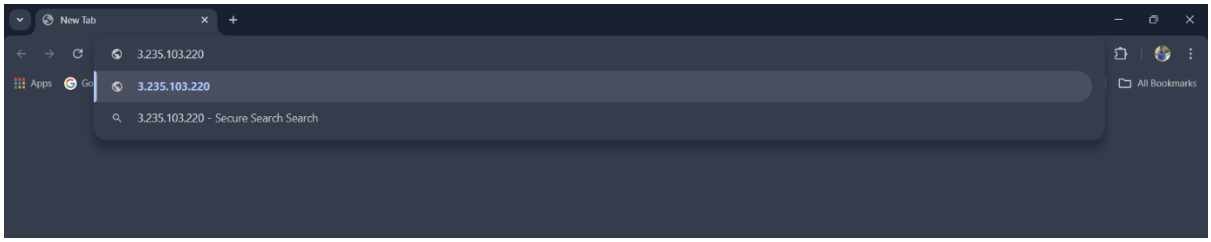
<label for="email">Email:</label>
<input type="text" class="form-control" name="email">

                    <button type="submit" class="btn btn-success">Submit</button>
</form></td>
<td colspan="4"></td>
</tr>
</table>

<?php
$firstname=$_POST['firstname'];
$email=$_POST['email'];
$servername = "intelli.coghw13fheqo.us-east-2.rds.amazonaws.com";
$username = "intel";
$password = "intel123";
$db = "intel";
// Create connection
$conn = new mysqli($servername, $username, $password, $db);
Connect
Ln 17, Col 27 Spaces: 2 UTF-8 LF PHP


root@ip-172-31-9-207: /var/w
root@ip-172-31-9-207:/var/www/html# apt-add-repository -y PPa:ondrej/php


root@ip-172-31-9-207: /var/w
root@ip-172-31-9-207:/var/www/html# apt install php5.6 mysql-client php5.6-mysqli -y

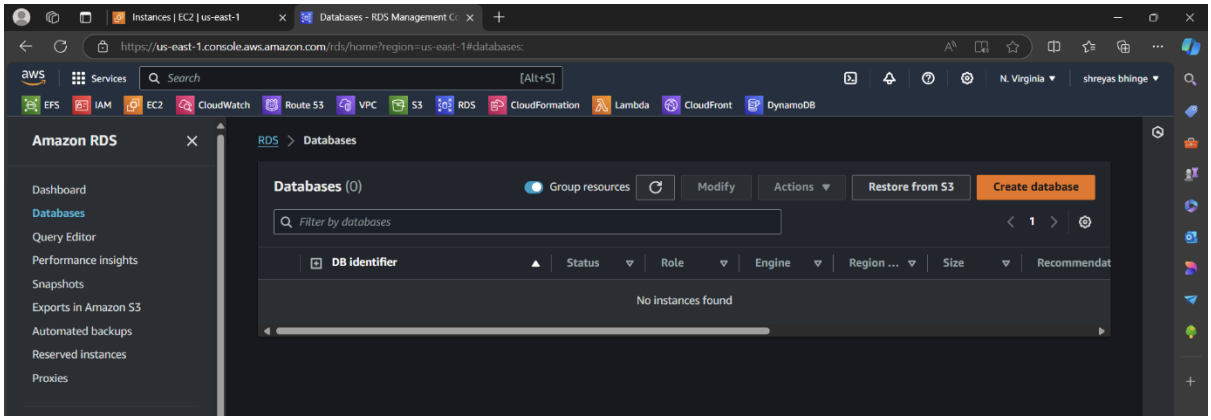

New Tab
3.235.103.220
Apps
3.235.103.220
All Bookmarks
3.235.103.220 - Secure Search Search

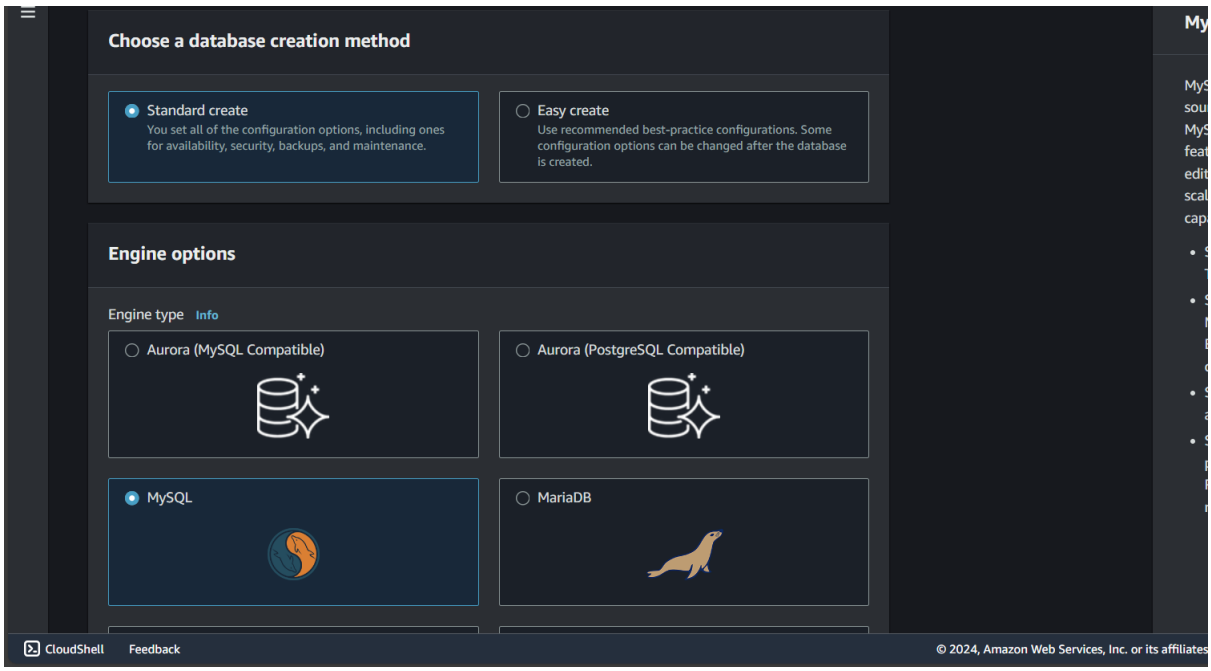
Name:

Email:

Submit

Connection failed: php_network_getaddresses: getaddrinfo failed: Name or service not known





Templates

Choose a sample template to meet your use case.

- ○ **Production**
  Use defaults for high availability and fast, consistent performance.
- ○ **Dev/Test**
  This instance is intended for development use outside of a production environment.
- ◉ **Free tier**
  Use RDS Free Tier to develop new applications, test existing applications, or gain hands-on experience with Amazon RDS.
  Info

Availability and durability

Settings

DB instance identifier Info

Type a name for your DB instance. The name must be unique across all DB instances owned by your AWS account in the current AWS Region.

project-database-1

The DB instance identifier is case-insensitive, but is stored as all lowercase (as in "mydbinstance"). Constraints: 1 to 60 alphanumeric characters or hyphens. First character must be a letter. Can't contain two consecutive hyphens. Can't end with a hyphen.

▼ **Credentials Settings**

Master username Info

Type a login ID for the master user of your DB instance.

intel

1 to 16 alphanumeric characters. The first character must be a letter.

Credentials management

You can use AWS Secrets Manager or manage your master user credentials.

- ○ **Managed in AWS Secrets Manager - *most secure***
  RDS generates a password for you and manages it throughout its lifecycle using AWS Secrets Manager.
- ◉ **Self managed**
  Create your own password or have RDS create a password that you manage.

☐ **Auto generate password**

Amazon RDS can generate a password for you, or you can specify your own password.

Master password Info

Password strength

CloudShell Feedback © 2024, Amazon We

Amazon RDS can generate a password for you, or you can specify your own password.

Master password Info

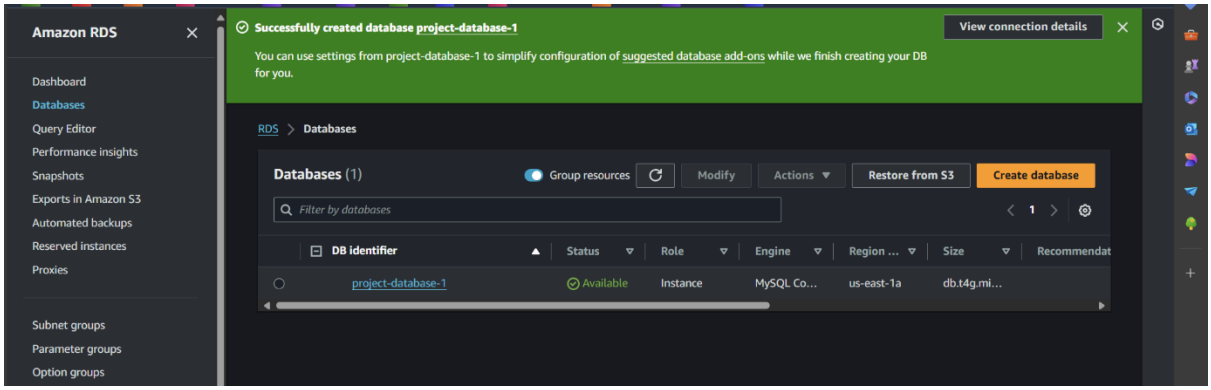
••••••••

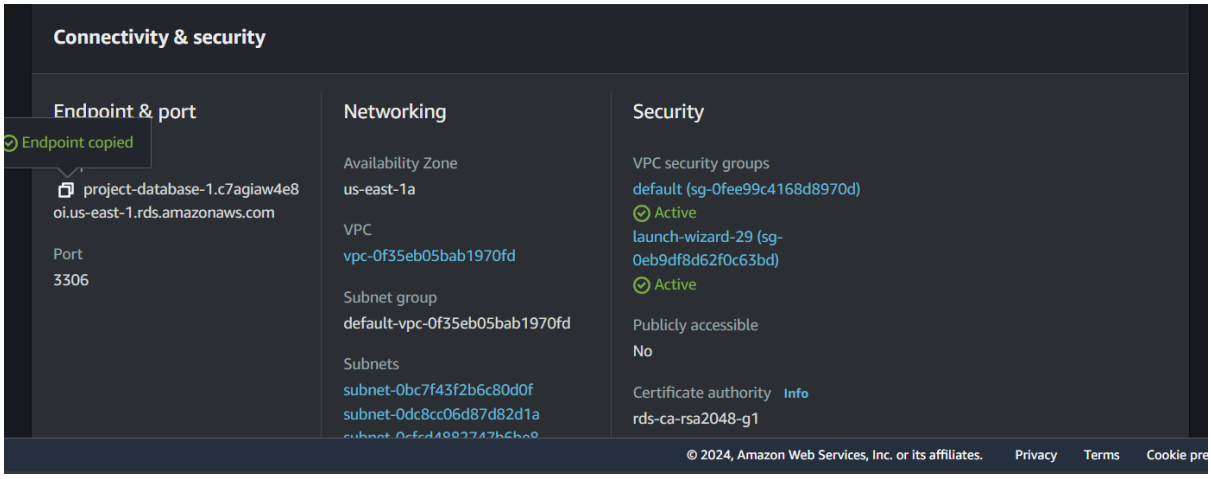
Password strength Weak

Minimum constraints: At least 8 printable ASCII characters. Can't contain any of the following symbols: / ' " @

Confirm master password Info

••••••••





Connect the instance and enter the following commands

1. mysql -h project-databaseinstance.cvwwkjmmcvgi.us-east-1.rds.amazonaws.com -u intel -p ---> conecting to mysql shell

2. show databases;

3. use intel;

4. show tables;

5. create table data(firstname varchar(50),email varchar(50));

6. insert into data(firstname, email) values ("satvik","satvikintellipaat@gmail.com"); ---> isertind temp data into data table

7. select * from data;

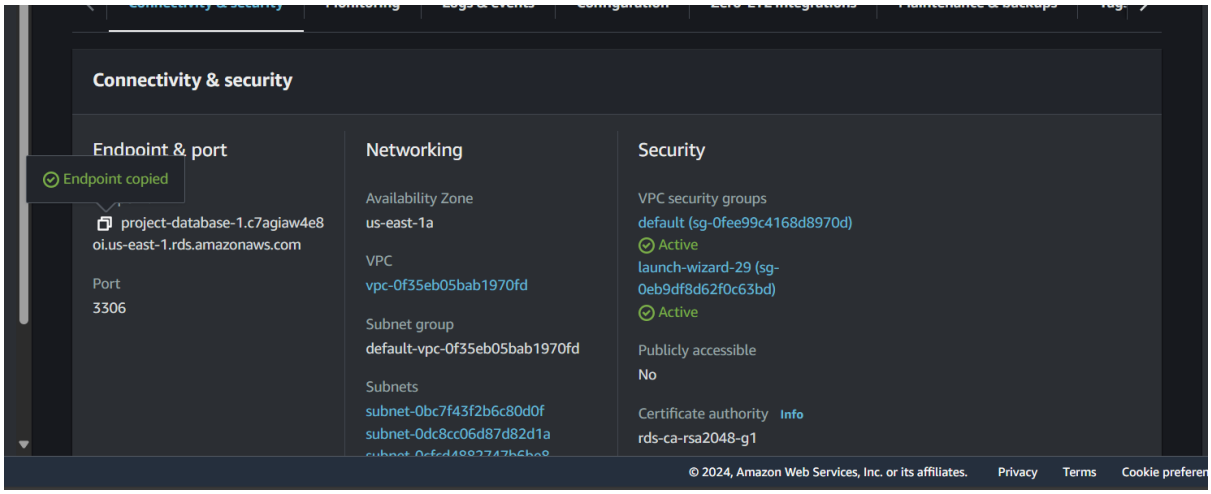
```
mysql> select * from data;
+-----------+-------------------------+
| firstname | email                   |
+-----------+-------------------------+
| shreyas   | shreyasbhinge@gmail.com |
+-----------+-------------------------+
1 row in set (0.00 sec)

mysql>
```

- Edit index.php file

```
root@ip-172-31-9-207: /var/w
root@ip-172-31-9-207:/var/www/html# nano index.php
```

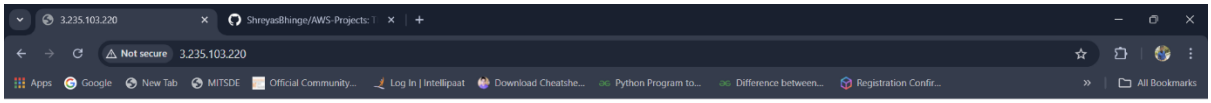
- Change service name and enter endpoint of database

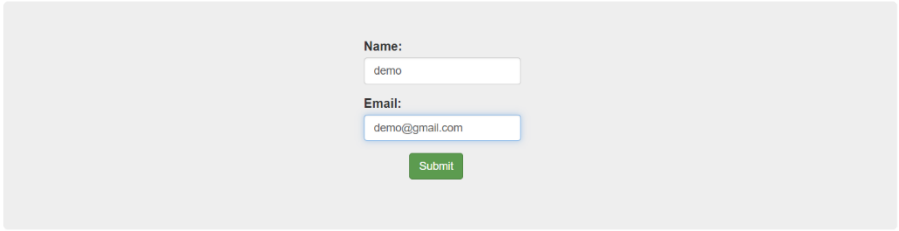
```
</table>
</div>
</div>
<?php
$firstname=$_POST['firstname'];
$email=$_POST['email'];
$servername = "intelli.coghw13fheqo.us-east-2.rds.amazonaws.com";
$username = "intel";
$password = "intel123";
$db = "intel";

^G Help        ^O Write Out   ^W Where Is    ^K Cut         ^T Execute     ^C Location    M-U Undo       M-A Set Mark
^X Exit        ^R Read File   ^\ Replace     ^U Paste       ^J Justify     ^/ Go To Line  M-E Redo       M-6 Copy
```

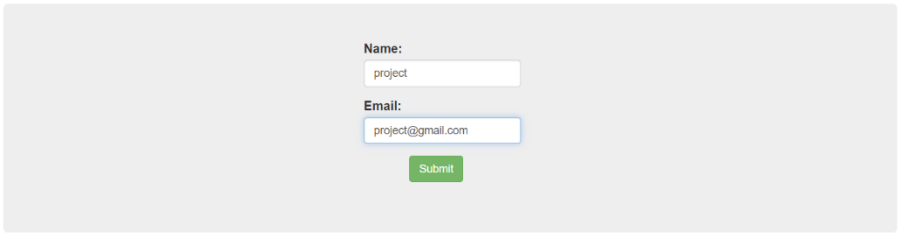


```
</table>
</div>
</div>
<?php
$firstname=$_POST['firstname'];
$email=$_POST['email'];
$servername = "project-database-1.c7agiaw4e8oi.us-east-1.rds.amazonaws.com";
$username = "intel";
$password = "intel123";
$db = "intel";

^G Help        ^O Write Out   ^W Where Is    ^K Cut         ^T Execute     ^C Location    M-U Undo       M-A Set Mark
^X Exit        ^R Read File   ^\ Replace     ^U Paste       ^J Justify     ^/ Go To Line  M-E Redo       M-6 Copy
```

Enter the email ids







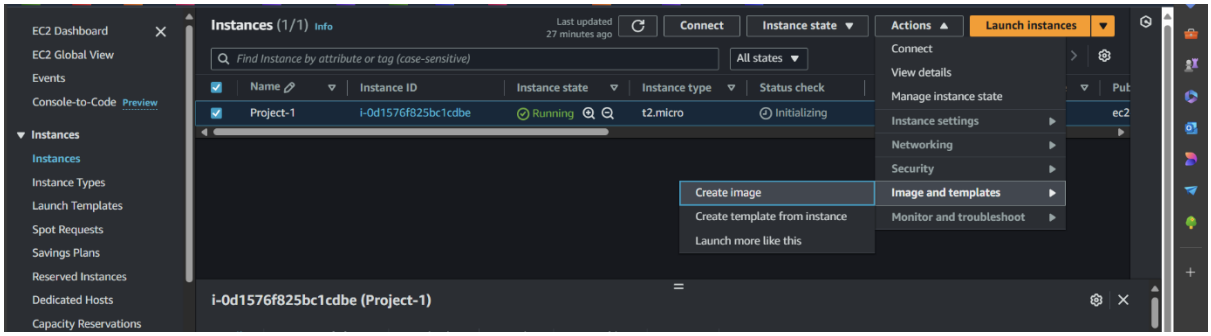
New record created successfully

Checked Data has been entered or not

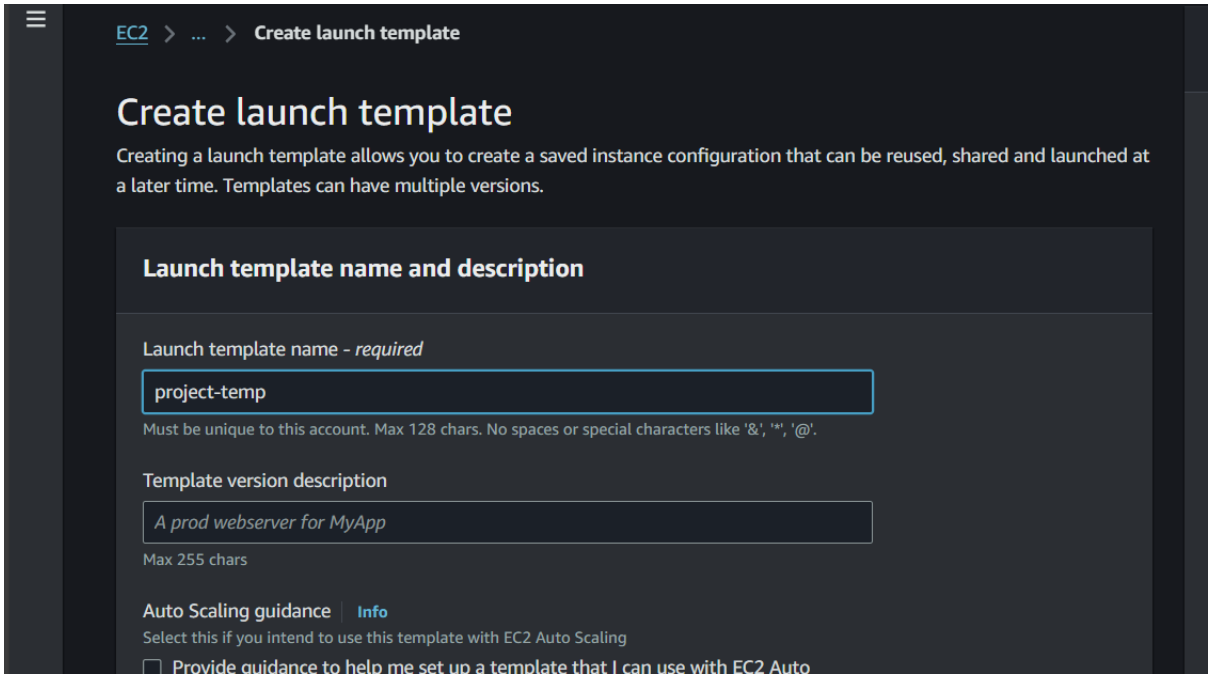
```
Database changed
mysql> select * from data;
+-----------+-------------------------+
| firstname | email                   |
+-----------+-------------------------+
| shreyas   | shreyasbhinge@gmail.com |
| demo      | demo@gmail.com          |
| project   | project@gmail.com       |
+-----------+-------------------------+
3 rows in set (0.00 sec)

mysql>
```

Successfully entered the data !!!!!!

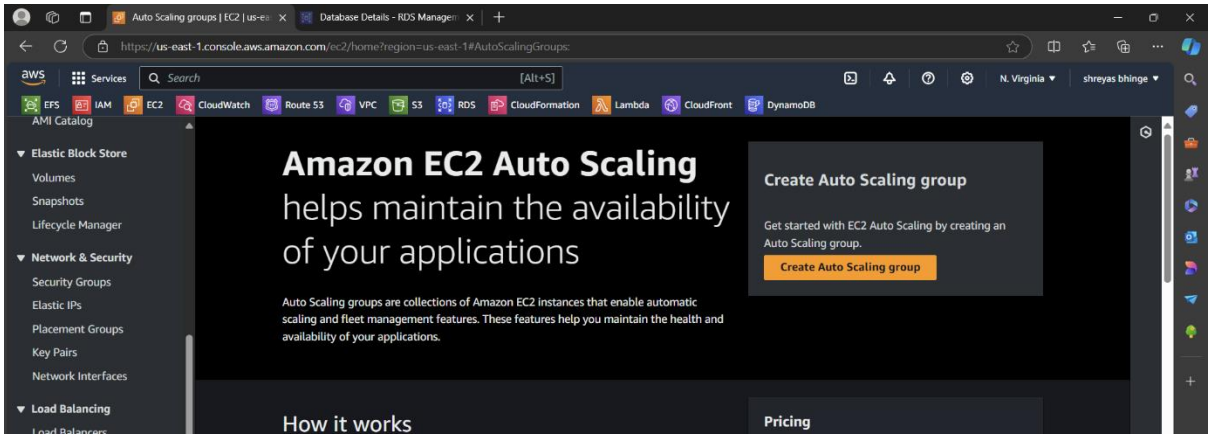
Create the image

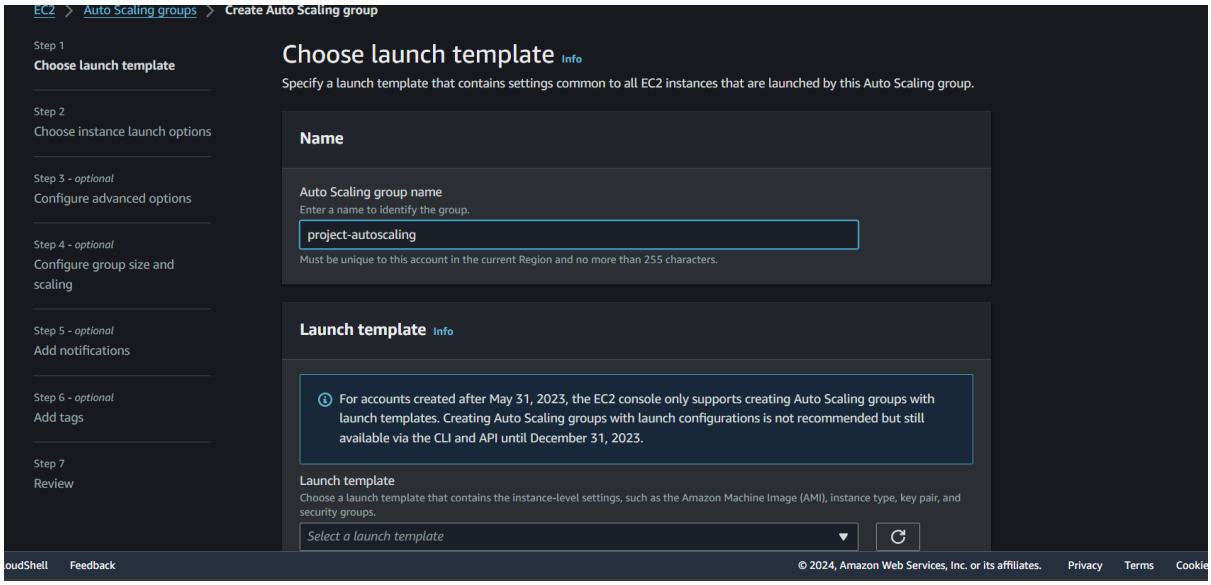


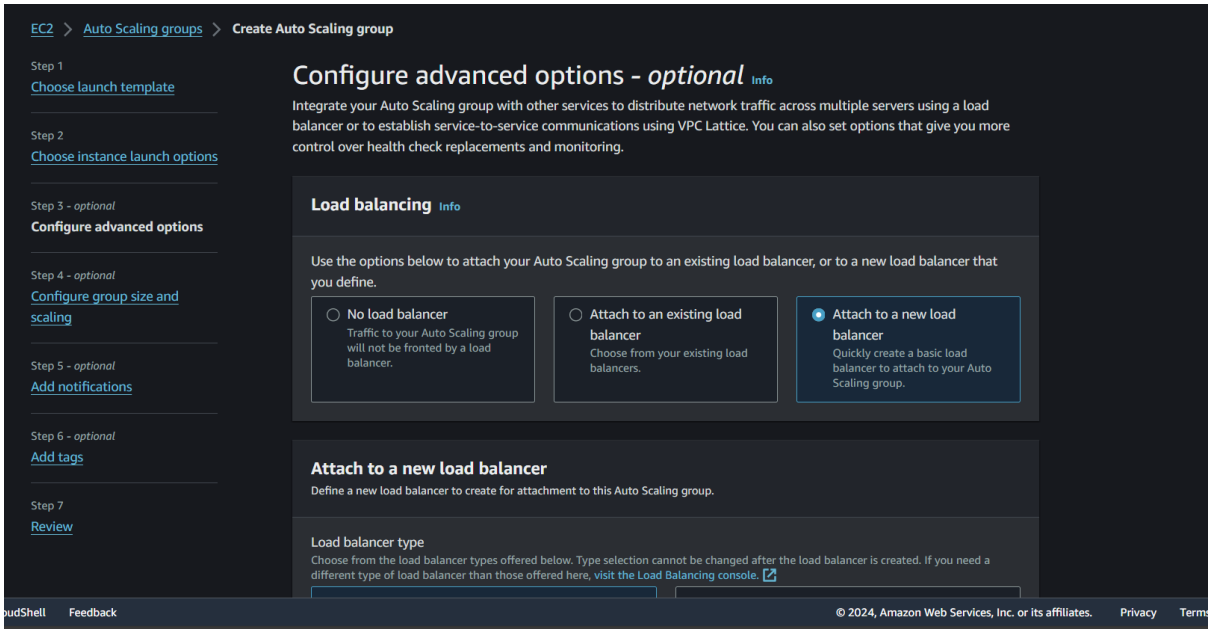
Create the launch template

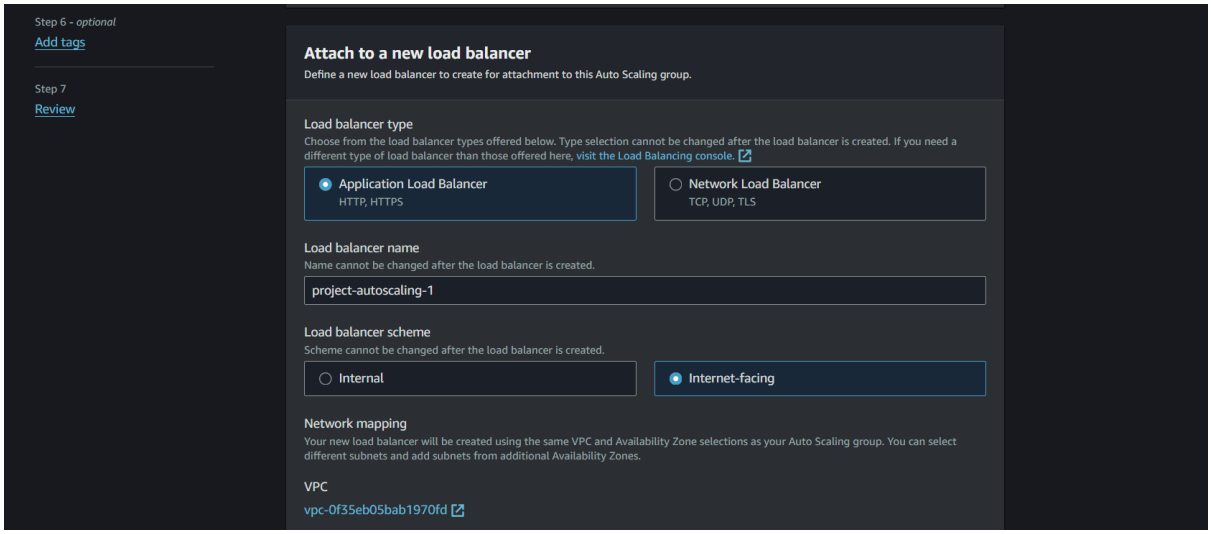


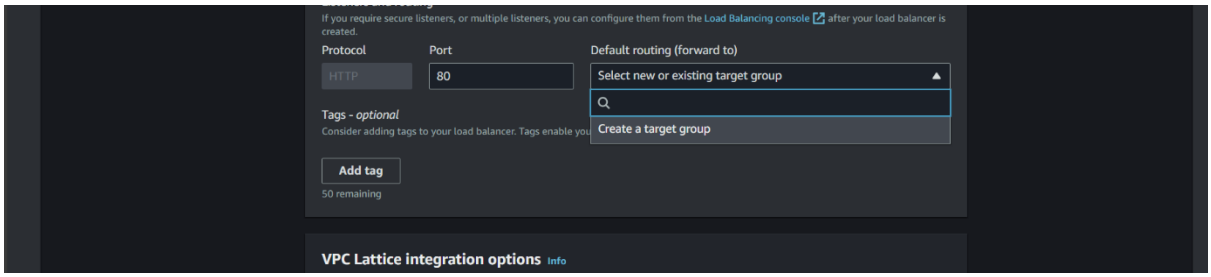
Create the Auto scaling

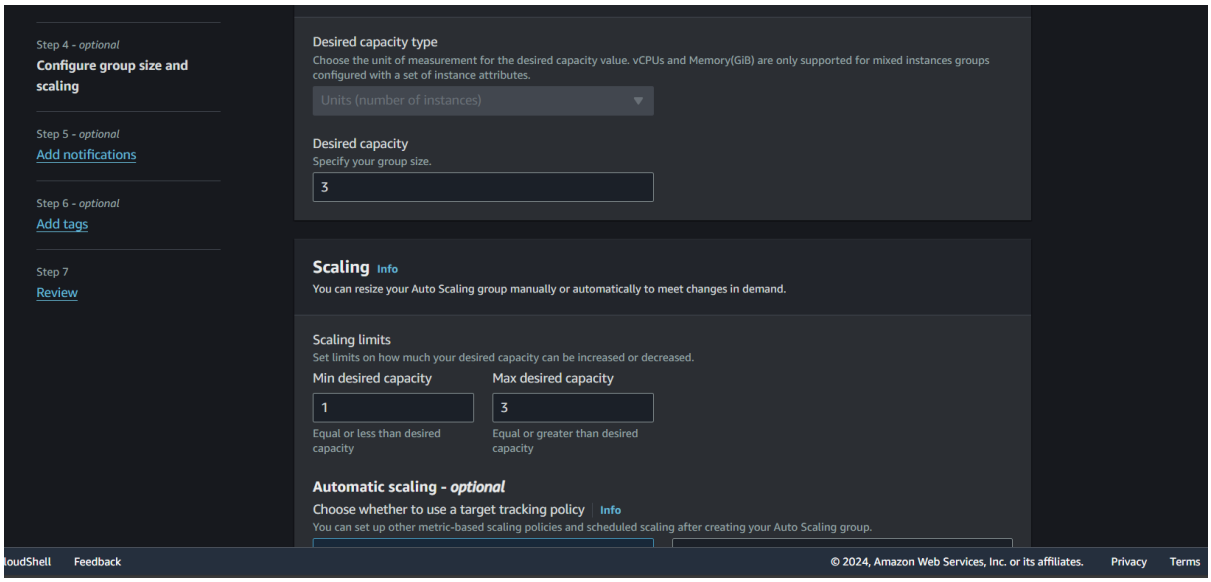

Auto Scaling groups | EC2 | us-ea
Database Details - RDS Manager
https://us-east-1.console.aws.amazon.com/ec2/home?region=us-east-1#AutoScalingGroups:
Services
Search
[Alt+S]
N. Virginia
shreyas bhinge
EFS
IAM
EC2
CloudWatch
Route 53
VPC
S3
RDS
CloudFormation
Lambda
CloudFront
DynamoDB
AMI Catalog
Elastic Block Store
Volumes
Snapshots
Lifecycle Manager
Network & Security
Security Groups
Elastic IPs
Placement Groups
Key Pairs
Network Interfaces
Load Balancing
Amazon EC2 Auto Scaling
helps maintain the availability of your applications
Auto Scaling groups are collections of Amazon EC2 instances that enable automatic scaling and fleet management features. These features help you maintain the health and availability of your applications.
Create Auto Scaling group
Get started with EC2 Auto Scaling by creating an Auto Scaling group.
Create Auto Scaling group
How it works
Pricing

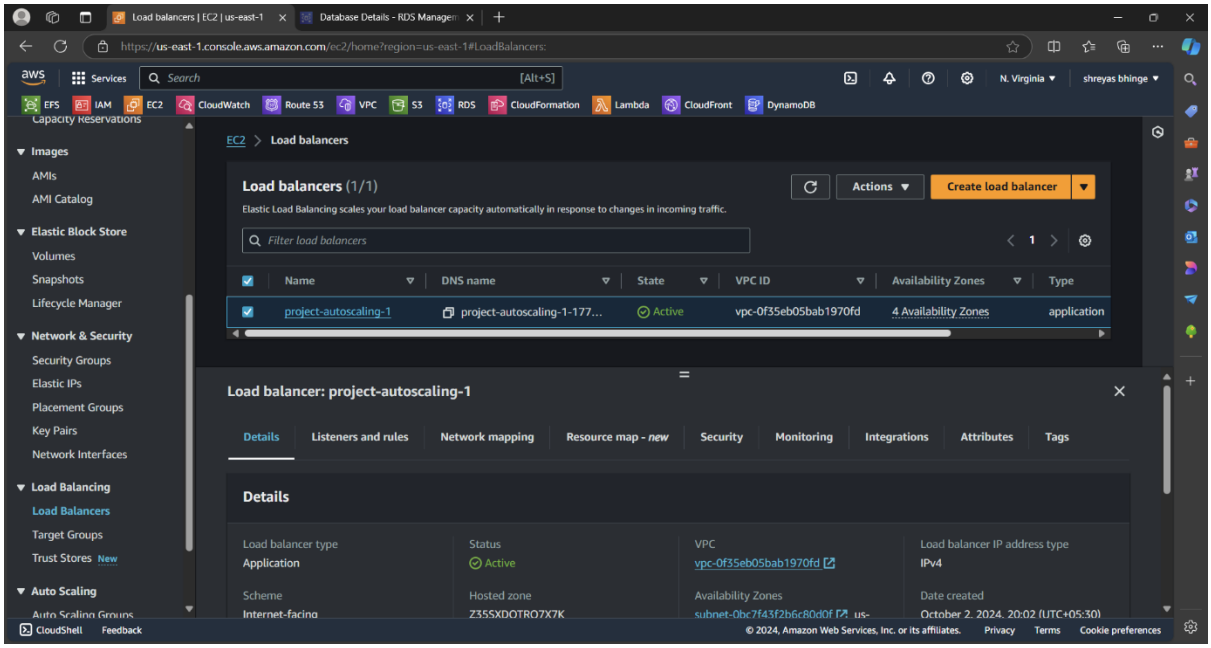

EC2 > Auto Scaling groups > Create Auto Scaling group
Step 1
Choose launch template
Step 2
Choose instance launch options
Step 3 - optional
Configure advanced options
Step 4 - optional
Configure group size and scaling
Step 5 - optional
Add notifications
Step 6 - optional
Add tags
Step 7
Review
Choose launch template Info
Specify a launch template that contains settings common to all EC2 instances that are launched by this Auto Scaling group.
Name
Auto Scaling group name
Enter a name to identify the group.
project-autoscaling
Must be unique to this account in the current Region and no more than 255 characters.
Launch template Info
For accounts created after May 31, 2023, the EC2 console only supports creating Auto Scaling groups with launch templates. Creating Auto Scaling groups with launch configurations is not recommended but still available via the CLI and API until December 31, 2023.
Launch template
Choose a launch template that contains the instance-level settings, such as the Amazon Machine Image (AMI), instance type, key pair, and security groups.
Select a launch template
CloudShell
Feedback
© 2024, Amazon Web Services, Inc. or its affiliates.
Privacy
Terms
Cookie


EC2 > Auto Scaling groups > Create Auto Scaling group
Step 1
Choose launch template
Step 2
Choose instance launch options
Step 3 - optional
Configure advanced options
Step 4 - optional
Configure group size and scaling
Step 5 - optional
Add notifications
Step 6 - optional
Add tags
Step 7
Review
Configure advanced options - optional Info
Integrate your Auto Scaling group with other services to distribute network traffic across multiple servers using a load balancer or to establish service-to-service communications using VPC Lattice. You can also set options that give you more control over health check replacements and monitoring.
Load balancing Info
Use the options below to attach your Auto Scaling group to an existing load balancer, or to a new load balancer that you define.
No load balancer
Traffic to your Auto Scaling group will not be fronted by a load balancer.
Attach to an existing load balancer
Choose from your existing load balancers.
Attach to a new load balancer
Quickly create a basic load balancer to attach to your Auto Scaling group.
Attach to a new load balancer
Define a new load balancer to create for attachment to this Auto Scaling group.
Load balancer type
Choose from the load balancer types offered below. Type selection cannot be changed after the load balancer is created. If you need a different type of load balancer than those offered here, visit the Load Balancing console.
CloudShell
Feedback
© 2024, Amazon Web Services, Inc. or its affiliates.
Privacy

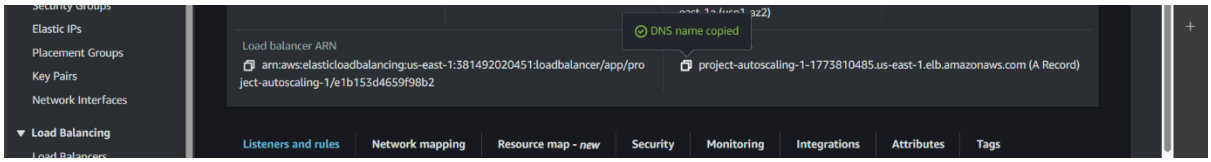

Step 6 - optional
Add tags
Step 7
Review
Attach to a new load balancer
Define a new load balancer to create for attachment to this Auto Scaling group.
Load balancer type
Choose from the load balancer types offered below. Type selection cannot be changed after the load balancer is created. If you need a different type of load balancer than those offered here, visit the Load Balancing console.
Application Load Balancer
HTTP, HTTPS
Network Load Balancer
TCP, UDP, TLS
Load balancer name
Name cannot be changed after the load balancer is created.
project-autoscaling-1
Load balancer scheme
Scheme cannot be changed after the load balancer is created.
Internal
Internet-facing
Network mapping
Your new load balancer will be created using the same VPC and Availability Zone selections as your Auto Scaling group. You can select different subnets and add subnets from additional Availability Zones.
VPC
vpc-0f35eb05bab1970fd

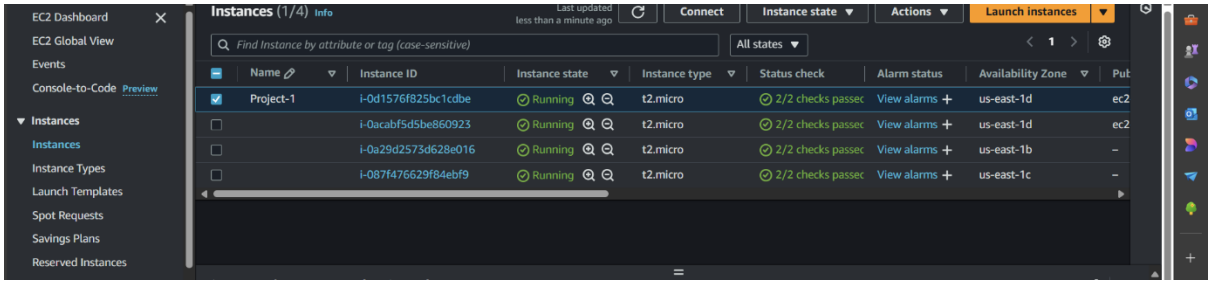

If you require secure listeners, or multiple listeners, you can configure them from the Load Balancing console after your load balancer is created.
Protocol
HTTP
Port
80
Default routing (forward to)
Select new or existing target group
Create a target group
Tags - optional
Consider adding tags to your load balancer. Tags enable you
Add tag
50 remaining
VPC Lattice integration options Info

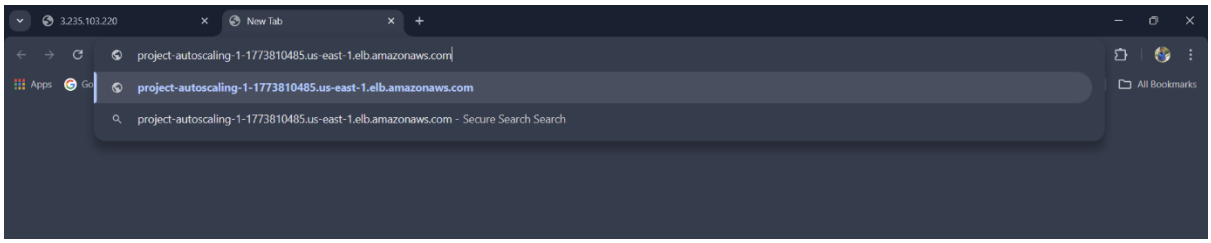

Step 4 - optional
Configure group size and scaling
Step 5 - optional
Add notifications
Step 6 - optional
Add tags
Step 7
Review
Desired capacity type
Choose the unit of measurement for the desired capacity value. vCPUs and Memory(GiB) are only supported for mixed instances groups configured with a set of instance attributes.
Units (number of instances)
Desired capacity
Specify your group size.
3
Scaling Info
You can resize your Auto Scaling group manually or automatically to meet changes in demand.
Scaling limits
Set limits on how much your desired capacity can be increased or decreased.
Min desired capacity
1
Equal or less than desired capacity
Max desired capacity
3
Equal or greater than desired capacity
Automatic scaling - optional
Choose whether to use a target tracking policy Info
You can set up other metric-based scaling policies and scheduled scaling after creating your Auto Scaling group.
loudShell
Feedback
© 2024, Amazon Web Services, Inc. or its affiliates.
Privacy
Terms

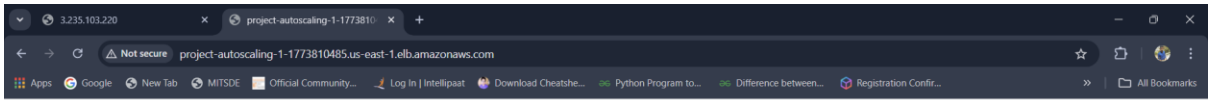


Copied the DNS and checked the autoscailing works









Name:

Email:

Submit