

## 3 – TIER ARCHITECTURE

**Def:** Three-tier architecture is a software development model that organizes applications into three logical and physical computing tiers. They are -

### 1.Presentation Tier

### 2.Application Tier

### 3.Database Tier

#### 1.Presentation Tier:

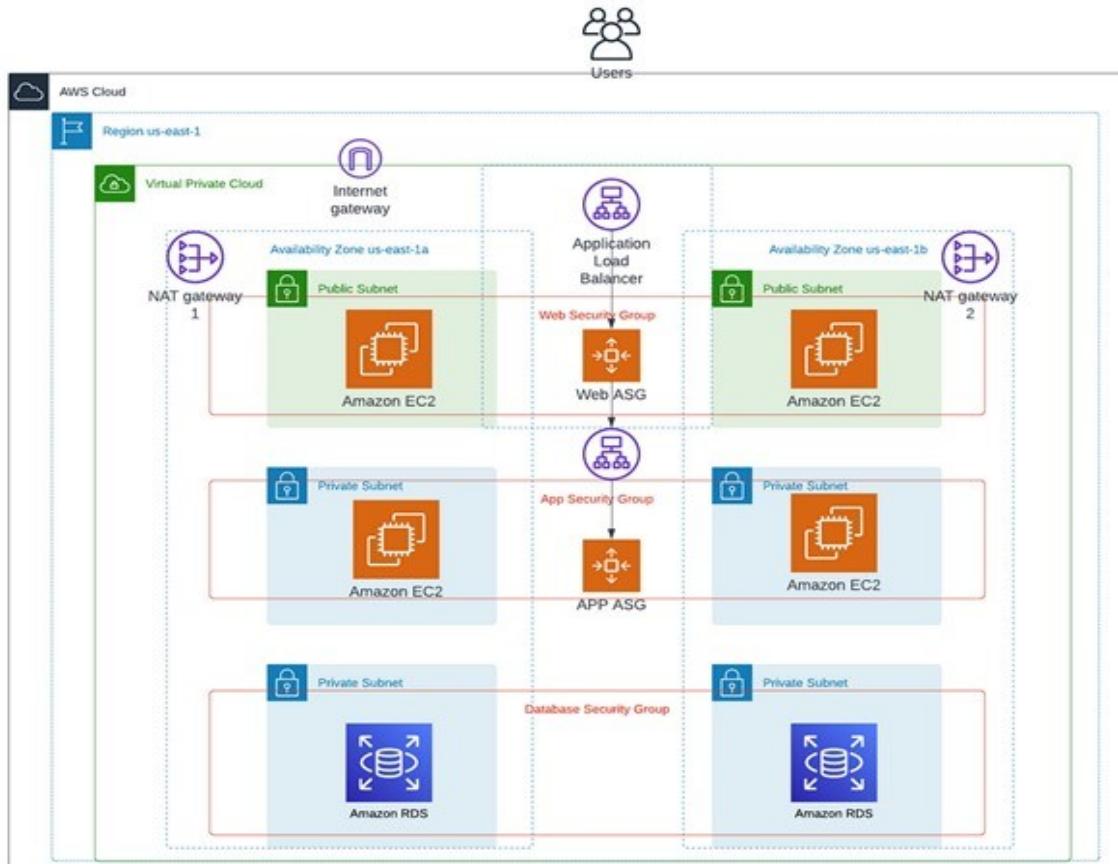
It is also known as the user interface, web tier or frontend this is where the end –user interacts with the system.

#### 2.Application Tier:

It is also known as the middle tier or logic tier this is the core of the application where the information is processed using business logic.

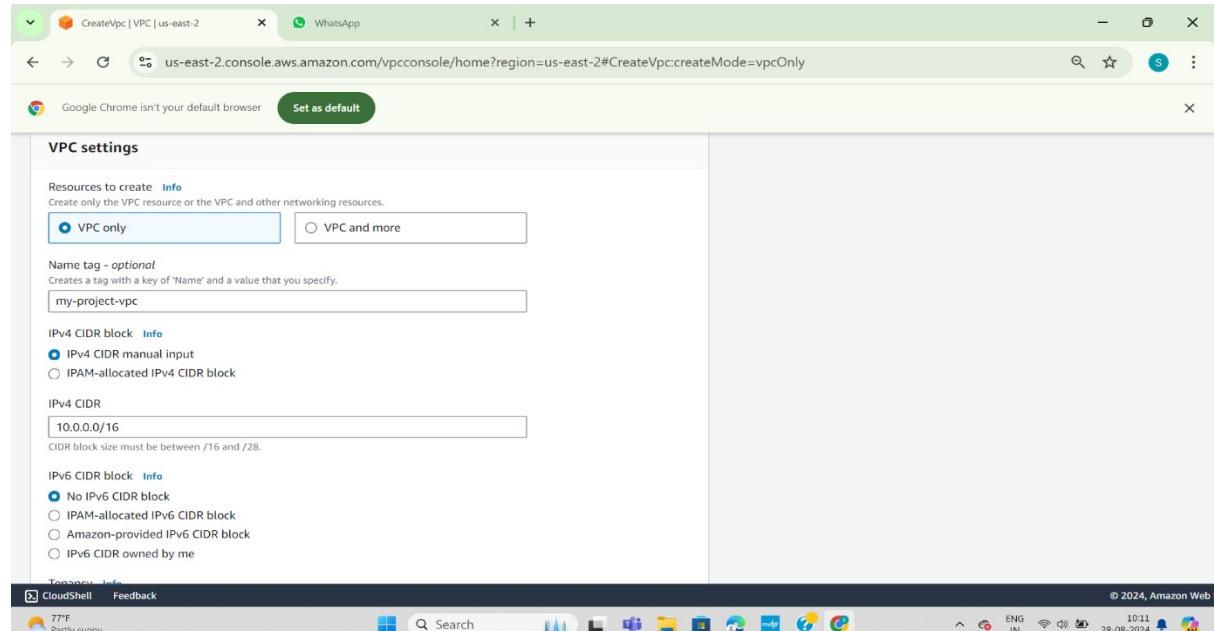
#### 3.Data Tier:

It is also known as the databases tier, back-end, or data access tier, this is where application's data is stored , managed ,retrieved and manipulated.

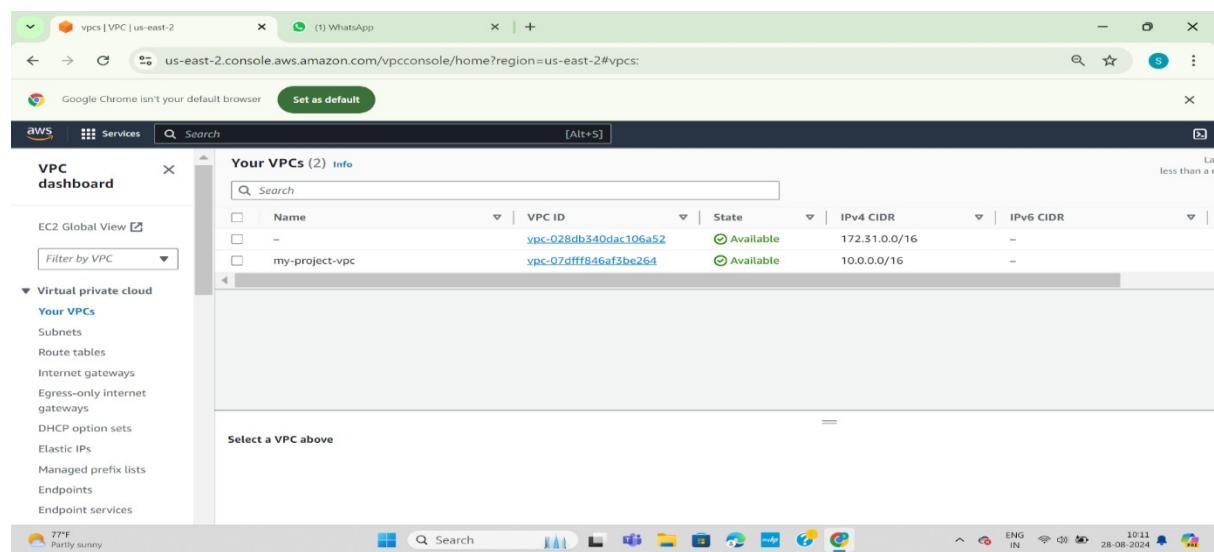


## Region : Ohio

### 1.Create VPC



The screenshot shows the 'CreateVpc | VPC | us-east-2' wizard. Under 'Resources to create', 'VPC only' is selected. A 'Name tag - optional' field contains 'my-project-vpc'. Under 'IPv4 CIDR block', 'IPv4 CIDR manual input' is selected, and the CIDR block '10.0.0.0/16' is entered. The status bar at the bottom right shows 'CloudShell' and 'Feedback'.

The screenshot shows the 'vpcs | VPC | us-east-2' dashboard. Under 'Your VPCs', there are two entries: one unnamed entry with VPC ID 'vpc-028db340dac106a52' and another named 'my-project-vpc' with VPC ID 'vpc-07dfff846af3be264'. The status bar at the bottom right shows 'CloudShell' and 'Feedback'.

### 2. Create 6 SUBNETS

- **Availability Zone 2a - 1 public & 2 private**
- **Availability Zone 2b – 1 public & 2 private**
- ✓ Click on SUBNETS & Click on create subnet & choose VPC ID (Own VPC not Default).
- ✓ Give subnet name & select availability zone (2a & 2b) & give IPv4 subnet CIDR BLOCK.
- ✓ click on create subnet.
- ✓ Create 6 subnets – 2 public subnets in 2a & 2b zones and 4 private subnets – take 2 private subnets in 2a & remaining 2 private subnets in 2b zone.

## Subnet - 1: my-Public-2a

The screenshot shows the AWS VPC console interface for creating a new subnet. The 'VPC ID' dropdown is set to 'vpc-07dff846af3be264 (my-project-vpc)'. Under 'Associated VPC CIDRs', the value '10.0.0.0/16' is listed. In the 'Subnet settings' section, the 'Subnet name' is 'my-public-2a' and the 'Availability Zone' is 'US East (Ohio) / us-east-2a'. The browser status bar at the bottom indicates 'Finance headline US Crude Oil Inv...'.

## Subnet – 2: my-private1-2a

The screenshot shows the AWS VPC console interface for creating a new subnet. The 'VPC ID' dropdown is set to 'vpc-07dff846af3be264 (my-project-vpc)'. Under 'Associated VPC CIDRs', the value '10.0.0.0/16' is listed. In the 'Subnet settings' section, the 'Subnet name' is 'my-private1-2a' and the 'Availability Zone' is 'US East (Ohio) / us-east-2a'. The browser status bar at the bottom indicates 'CloudShell Feedback' and 'Home Now'.

## Subnet – 3: my-private2-2a

The screenshot shows the AWS VPC console interface for creating a new subnet. The 'VPC ID' dropdown is set to 'vpc-07dff846af3be264 (my-project-vpc)'. Under 'Associated VPC CIDRs', the value '10.0.0.0/16' is listed. In the 'Subnet settings' section, the 'Subnet name' is 'my-private2-2a' and the 'Availability Zone' is 'US East (Ohio) / us-east-2a'. The browser status bar at the bottom indicates 'CloudShell Feedback' and 'Partly sunny'.

- Subnets in availability zone 2a are successfully created.

The screenshot shows the AWS VPC Subnets list page. The left sidebar includes sections for EC2 Global View, Virtual private cloud (Your VPCs), Subnets, Route tables, Internet gateways, Egress-only Internet gateways, DHCP option sets, Elastic IPs, Managed prefix lists, Endpoints, Endpoint services, NAT gateways, Peering connections, Security, Network ACLs, and CloudShell. The main content area displays a table of subnets:

Name	Subnet ID	State	VPC	IPv4 CIDR	IPv6 CIDR
-	subnet-07056da7f1512f1ff	Available	vpc-028db340dac106a52	172.31.16.0/20	-
-	subnet-0b52ca8f26a063d02	Available	vpc-028db340dac106a52	172.31.0.0/20	-
-	subnet-07abb95d8eb130d92	Available	vpc-028db340dac106a52	172.31.32.0/20	-
my-public-2a	subnet-0997bdb613658f9b	Available	vpc-07dff846af3be264   my-pr...	10.0.1.0/24	-
my-private1-2a	subnet-02cea2cf84ca1c1	Available	vpc-07dff846af3be264   my-pr...	10.0.2.0/24	-
my-private2-2a	subnet-0283512793f029b8c	Available	vpc-07dff846af3be264   my-pr...	10.0.3.0/24	-

A message at the top right indicates "less than a minute ago". The bottom of the screen shows the Windows taskbar with various pinned icons.

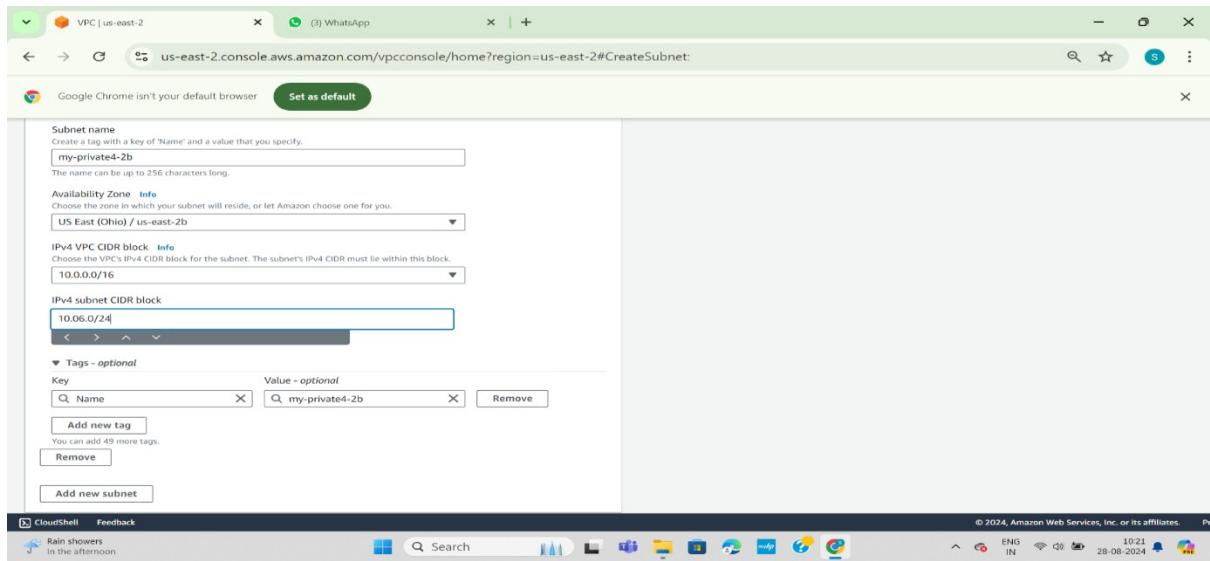
## Subnet - 4: my-public-2b

The screenshot shows the AWS Create Subnet wizard. The left sidebar has a single item: Subnet settings. The main content area is titled "Subnet 1 of 1". It contains fields for Subnet name (my-public-2b), Availability Zone (US East (Ohio) / us-east-2B), IPv4 VPC CIDR block (10.0.0.0/16), and an optional Tag (Key: Name, Value: my-public-2b). The bottom of the screen shows the Windows taskbar.

## Subnet – 5: my-private3-2b

The screenshot shows the AWS Create Subnet wizard. The left sidebar has a single item: Subnet settings. The main content area is titled "Subnet 1 of 1". It contains fields for Subnet name (my-private3-2b), Availability Zone (US East (Ohio) / us-east-2B), IPv4 VPC CIDR block (10.0.0.0/16), and an optional Tag (Key: Name, Value: my-private3-2b). The bottom of the screen shows the Windows taskbar.

## Subnet – 6: my-private4-2b



- The subnets in availability zone 2b are successfully created.

Name	Subnet ID	State	VPC	IPv4 CIDR	IPv6 CIDR
-	subnet-0000000000000000	Available	vpc-0000000000000000	172.31.0.0/20	-
my-public-2a	subnet-07abb95d8eb130d92	Available	vpc-028db540dac106a52	172.31.32.0/20	-
my-private-1-2a	subnet-0997bdb61365bf9fb	Available	vpc-07dff846af3be264   my-pr...	10.0.1.0/24	-
my-private-2-2a	subnet-02e0a2cf84cc141	Available	vpc-07dff846af3be264   my-pr...	10.0.2.0/24	-
my-public-2b	subnet-028512793f029b0c	Available	vpc-07dff846af3be264   my-pr...	10.0.3.0/24	-
my-private3-2b	subnet-04825c20ae673d483	Available	vpc-07dff846af3be264   my-pr...	10.0.4.0/24	-
my-private4-2b	subnet-007c256b5c387120c	Available	vpc-07dff846af3be264   my-pr...	10.0.5.0/24	-
	subnet-09109e7faaa2d0ca	Available	vpc-07dff846af3be264   my-pr...	10.0.6.0/24	-

## 3. Create INTERNET GATEWAY

- Click on internet gateway & create internet gateway.
- After the creation of internet gateway, click on internet gateway , click on actions & attach it to VPC.

The screenshot shows the AWS VPC console with the 'Internet gateways' section selected. There are two internet gateways listed:

Name	Internet gateway ID	State	VPC ID	Owner
—	igw-045d1198a221e7062	Attached	vpc-028db1540daa106a52	010928185144
my-lgw-project	igw-0bad0ec3bf3a6159	Detached	—	010928185144

Below the table, there is a message: "Select an internet gateway above".

## 4. Create NAT GATEWAY

- Click on Nat gateway & click on create.
- Select PRIVATE SUBNET & choose connectivity type as PUBLIIC & Allocate ELASTIC IP.
- Click on create NAT GATEWAY.

The screenshot shows the 'Create NAT gateway' wizard. Step 1: NAT gateway settings. The form includes:

- Name - optional:** my-ngw-public-project
- Subnet:** subnet-0997bdb613658f9fb (my-public-2a)
- Connectivity type:** Public (radio button selected)
- Elastic IP allocation ID:** eipalloc-04d8f78d386f510d1

## 5. Create ROUTE TABLES

- We have to create 2 route tables – one is PUBLIC & other is PRIVATE.
- Goto route table – click on create route- select VPC & create route table.
- Click on route- actions-edit **subnet associations**-select PUBLIC SUBNETS – save associations.
- create another route table as PRIVATE.
- Select VPC – do edit subnet associations – select 4 private subnets – save associations

VPC | us-east-2

(5) WhatsApp

us-east-2.console.aws.amazon.com/vpcconsole/home?region=us-east-2#CreateRouteTable:

Google Chrome isn't your default browser Set as default

aws Services Search [Alt+S]

Create route table Info

A route table specifies how packets are forwarded between the subnets within your VPC, the internet, and your VPN connection.

**Route table settings**

Name - optional Create a tag with a key of 'Name' and a value that you specify.

my-route-private

VPC The VPC to use for this route table.

vpc-07dfff846af3be264 (my-project-vpc)

**Tags**

A tag is a label that you assign to an AWS resource. Each tag consists of a key and an optional value. You can use tags to search and filter your resources or track your AWS costs.

Key Value - optional

Name my-route-private Remove

Add new tag You can add up to 49 more tags.

CloudShell Feedback © 2024, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

78°F Partly sunny

VPC | us-east-2

(5) WhatsApp

us-east-2.console.aws.amazon.com/vpcconsole/home?region=us-east-2#EditRouteTableSubnetAssociations:RouteTableId=rtb-06e9c99f67434cd4b

Google Chrome isn't your default browser Set as default

aws Services Search [Alt+S]

Edit subnet associations

Change which subnets are associated with this route table.

**Available subnets (4/6)**

Name	Subnet ID	IPv4 CIDR	IPv6 CIDR	Route table ID
my-public-2a	subnet-0997bdb6136589fb	10.0.1.0/24	-	rtb-0fb01b7a76cdc04c / my-route-p
<input checked="" type="checkbox"/> my-private1-2a	subnet-02eea2caf84cac141	10.0.2.0/24	-	Main (rtb-0321b31c5c601104f)
<input checked="" type="checkbox"/> my-private2-2a	subnet-0283512793f029b8c	10.0.3.0/24	-	Main (rtb-0321b31c5c601104f)
my-public-2b	subnet-04825c20a6c73d483	10.0.4.0/24	-	rtb-0fb01b7a76cdc04c / my-route-p
<input checked="" type="checkbox"/> my-private3-2b	subnet-007c256b5c387120c	10.0.5.0/24	-	Main (rtb-0321b31c5c601104f)
<input checked="" type="checkbox"/> my-private4-2b	subnet-09c109e7afaa2deca	10.0.6.0/24	-	Main (rtb-0321b31c5c601104f)

**Selected subnets**

subnet-02eea2caf84cac141 / my-private1-2a X subnet-007c256b5c387120c / my-private3-2b X subnet-09c109e7afaa2deca / my-private4-2b X subnet-0283512793f029b8c / my-private2-2a X

CloudShell Feedback © 2024, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

78°F Partly sunny

- click on **public** route table – edit routes – add rules- attach **internet gateway** – save changes.
- for **private** route table – attach **Nat gateway** – save changes.

VPC | us-east-2

(5) WhatsApp

us-east-2.console.aws.amazon.com/vpcconsole/home?region=us-east-2#EditRoutes:RouteTableId=rtb-0fb01b7a76cdc04c

Google Chrome isn't your default browser Set as default

aws Services Search [Alt+S]

VPC > Route tables > rtb-0fb01b7a76cdc04c > Edit routes

**Edit routes**

Destination	Target	Status	Propagated
10.0.0.0/16	local	Active	No
0.0.0.0/0	Internet Gateway	-	No
	igw-0bade4ec3bf3a6159		

Add route Cancel Preview Save changes

CloudShell Feedback © 2024, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

78°F Partly sunny

- Now go to the subnets - click on public subnet -01, click on actions – edit subnet settings – ENABLE Auto assign public IPv4 address.
- Do the same for remaining subnets also.

## 6. Create Security Groups

- we have to create TWO security groups.
- Go to security groups - click on create security groups- select VPC – add INBOUND (SSH & HTTP) & OUTBOUND RULES (All traffic) – click on create security group.

### Public:

**Inbound rules**

Type	Protocol	Port range	Source	Description - optional
Custom TCP	TCP	80	Anywhere	0.0.0.0/0
SSH	TCP	22	Anywhere	0.0.0.0/0
All traffic	All	All	Anywhere	0.0.0.0/0

**Rules with source of 0.0.0.0/0 or ::/0 allow all IP addresses to access your instance. We recommend setting security group rules to allow access from known IP addresses only.**

## Private:

**Create security group**

A security group acts as a virtual firewall for your instance to control inbound and outbound traffic. To create a new security group, complete the fields below.

**Basic details**

Security group name **Info**  
my-sg-private  
Name cannot be edited after creation.

Description **Info**  
allow

VPC **Info**  
vpc-07dff846af3be264 (my-project-vpc)

**Inbound rules**

Type	Protocol	Port range	Source	Description - optional
Custom TCP	TCP	80	Anywhere	0.0.0.0/0
SSH	TCP	22	Anywhere	0.0.0.0/0
All traffic	All	All	Anywhere	0.0.0.0/0

**Inbound rules**

Type	Protocol	Port range	Source	Description - optional
Custom TCP	TCP	80	Anywhere	0.0.0.0/0
SSH	TCP	22	Anywhere	0.0.0.0/0
All traffic	All	All	Anywhere	0.0.0.0/0

**Rules with source of 0.0.0.0/0 or ::/0 allow all IP addresses to access your instance. We recommend setting security group rules to allow access from known IP addresses only.**

## 7. Launch TWO Templates (public & private)

### Public Template:

- Search EC2 – Click on LAUNCH TEMPLATES – Click on CREATE LAUNCH TEMPLATES.
- Select AMI – UBUNTU & instance type - t2. Micro (1GB- Free Tier).
- Select KEY PAIR – a new or existing.
- In Network Settings I am not going to specify subnets, but security group (public -sg) that I am created is selected. Make sure the proper VPC is selected.

The screenshot shows the AWS CloudShell interface with a browser window titled "Create launch template | EC2 | us-east-2". The main form is titled "Create launch template name and description". It contains fields for "Launch template name - required" (set to "my-public"), "Template version description" (set to "nothing"), and "Auto Scaling guidance" (checkbox checked). Below these are sections for "Template tags" and "Source template". On the right side, a "Summary" panel shows the configuration: Software Image (AMI) is Canonical, Ubuntu, 24.04, amd64 (ami-085f9c64a9b75eed5); Virtual server type (instance type) is t2.micro; Firewall (security group) is my-sg-public; and Storage (volumes) is 1 volume(s) - 8 GiB. A tooltip for the "Free tier" is displayed, stating: "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". At the bottom right of the summary panel is a yellow "Create launch template" button.

The screenshot shows the AWS CloudShell interface with a browser window titled "Create launch template | EC2 | us-east-2". The main form is titled "Application and OS Images (Amazon Machine Image) Info". It features a search bar and a "Quick Start" section with icons for "Don't include in launch template", "Amazon Linux", "macOS", "Ubuntu", "Windows", and "Red Hat". A "Browse more AMIs" link is also present. Below this is a detailed view of the "Ubuntu Server 24.04 LTS (HVM), SSD Volume Type" AMI, including its ID (ami-085f9c64a9b75eed5), type (64-bit (x86)), and other details like ENA enabled and Root device type. A "Free tier eligible" badge is shown. On the right side, a "Summary" panel shows the configuration: Software Image (AMI) is Canonical, Ubuntu, 24.04, amd64 (ami-085f9c64a9b75eed5); Virtual server type (instance type) is t2.micro; Firewall (security group) is my-sg-public; and Storage (volumes) is 1 volume(s) - 8 GiB. A tooltip for the "Free tier" is displayed, stating: "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". At the bottom right of the summary panel is a yellow "Create launch template" button.

The screenshot shows the 'Create launch template' wizard on the AWS EC2 console. In the 'Instance type' section, a t2.micro instance is selected. The summary panel indicates a free tier eligible for 750 hours. In the 'Network settings' section, a subnet named 'my-public-2b' is chosen. A tooltip for the subnet shows it's in the 'us-east-2' region with a CIDR of 10.0.4.0/24. The summary panel also lists a security group 'my-sg-public' and a volume of 8 GiB.

## Private Template:

- Create same as previous template, but at SECURITY GROUP select security group 2(private -sg).

The screenshot shows the 'Create launch template' wizard. In the 'Network settings' section, the subnet 'my-public-2b' is selected. In the 'Security group' dropdown, 'my-sg-public' is chosen. A tooltip for the security group indicates it's a private group. The summary panel shows the same details as the previous screenshot, including the subnet, volume, and security group.

Screenshot of the AWS CloudShell interface showing the creation of a launch template for an Ubuntu Server 24.04 LTS (HVM) instance. The instance is configured with 1 vCPU, 1 GiB Memory, and a t2.micro instance type. It includes a my-sg-private security group and 8 GiB of storage. A key pair named 'project1' is selected. The 'Create launch template' button is highlighted.

Screenshot of the AWS CloudShell interface showing the continuation of the launch template creation process. The instance type is now set to t2.micro. The 'Create launch template' button is highlighted.

Screenshot of the AWS CloudShell interface showing the final steps of creating the launch template. The subnet 'my-private1-2a' is selected, and the 'Create launch template' button is highlighted.

- Two Templates launched Successfully.

The screenshot shows the AWS EC2 Launch Templates page. On the left, there's a sidebar with navigation links like EC2 Dashboard, EC2 Global View, Instances, Images, and Elastic Block Store. The main area displays a table titled "Launch Templates (2) Info". The table has columns for Launch Template ID, Launch Template Name, Default Version, Latest Version, Create Time, and Created By. Two entries are listed:

Launch Template ID	Launch Template Name	Default Version	Latest Version	Create Time	Created By
lt-02c667985771aa9d	my-private	1	1	2024-08-28T05:10:53.000Z	arn:aws:iam::...
lt-043b1ff9e02b5f2bf	my-public	1	1	2024-08-28T05:09:06.000Z	arn:aws:iam::...

A modal window titled "Select a launch template" is open at the bottom, showing the same two options: "my-private" and "my-public".

## 8. Create two Autoscaling Groups (Public & Private):

### Public ASG:

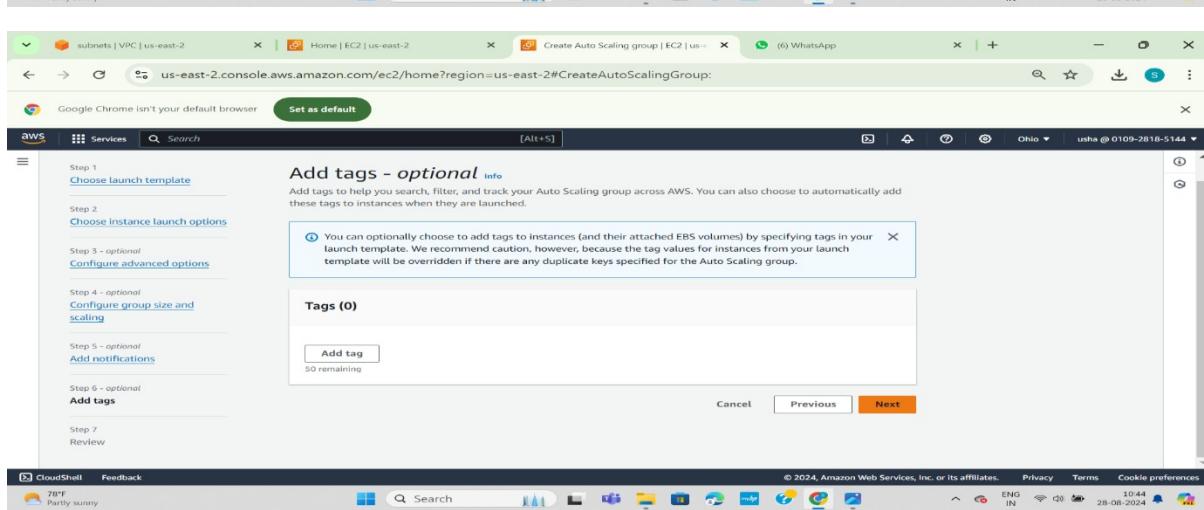
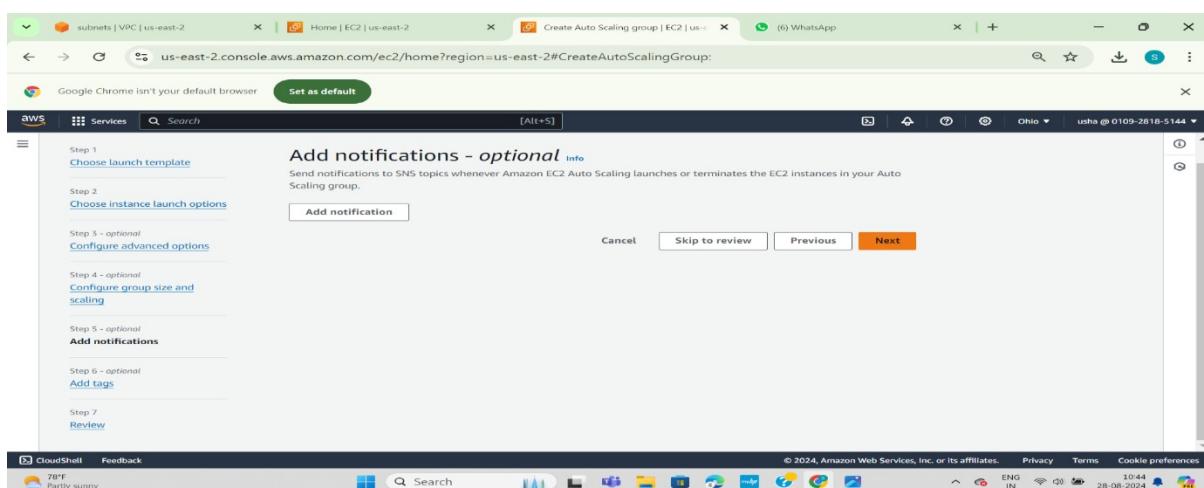
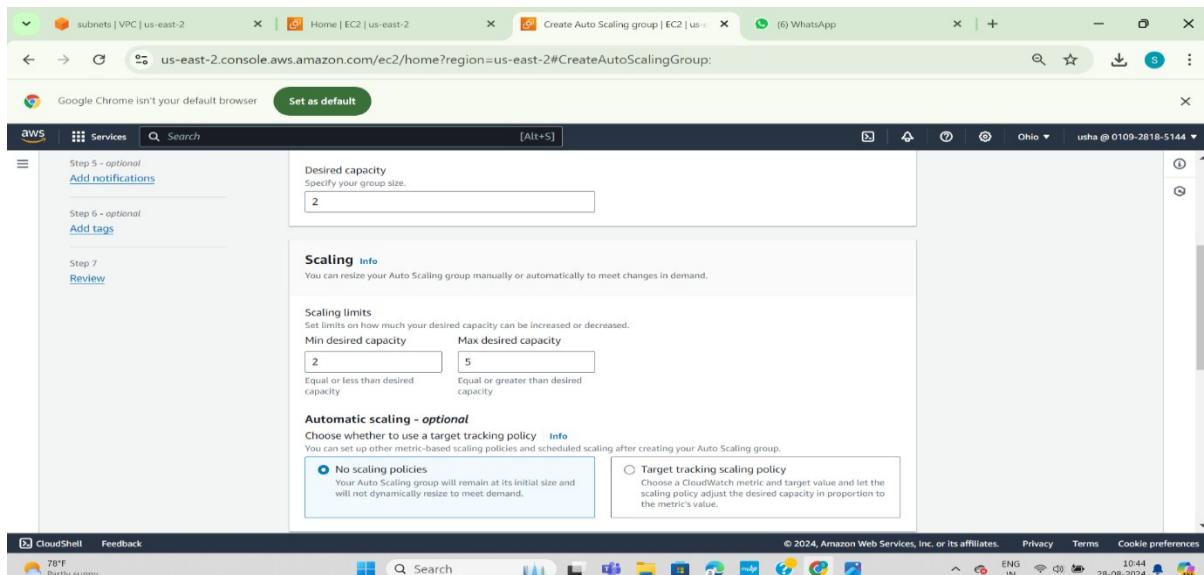
- In EC2, go to autoscaling group – click on create autoscaling group.
- give name to ASG – Select PUBLIC TEMPLATE (which is already created).
- In network settings - choose VPC – choose 2 public subnets.
- After that click on NEXT.
- We have to attach new LOAD BALANCER to ASG.

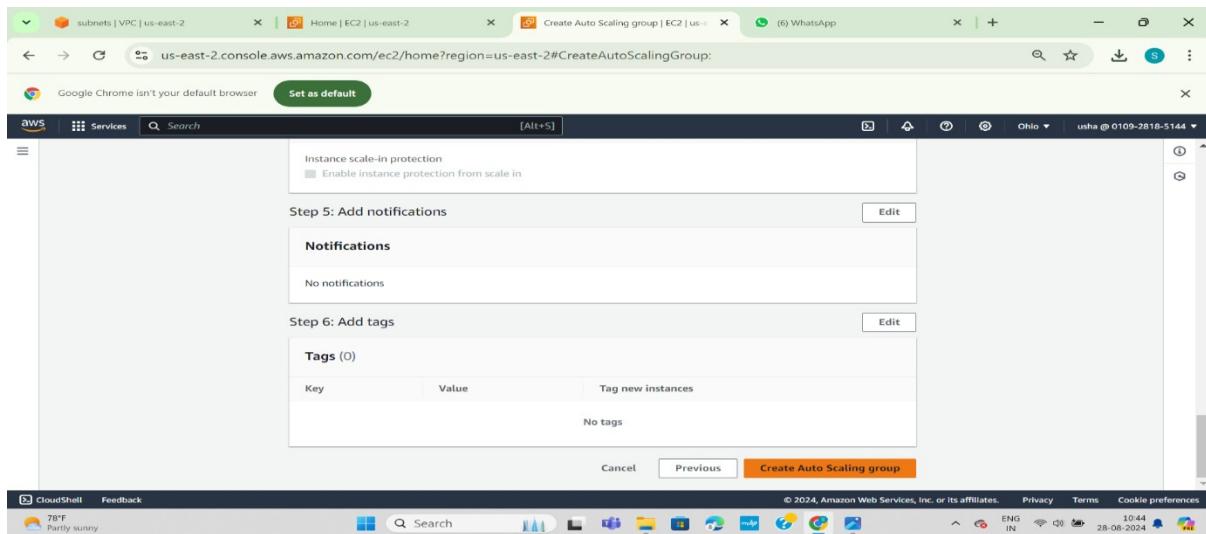
The screenshot shows the "Create Auto Scaling group" wizard. The current step is "Step 4 - optional". The "Auto Scaling group name" field is populated with "my-asg-public". Below it, a note says: "For accounts created after May 31, 2023, the EC2 console only supports creating Auto Scaling groups with launch templates. Creating Auto Scaling groups with launch configurations is not recommended but still available via the CLI and API until December 31, 2023." The "Launch template" dropdown is set to "my-public". Other fields include "Version" (Default (1)), "Create a launch template version" (unchecked), and "Description" (empty). The "Instance type" dropdown is also visible.

The screenshot shows the "Step 7 - Review" step. The "VPC" section shows "vpc-07df184af8be264 (my-project-vpc)" selected. The "Availability Zones and subnets" section shows two subnets: "us-east-2a | subnet-0997bdb613658f9fb (my-public-2a)" and "us-east-2b | subnet-04825c20a6c73d483 (my-public-2b)". The "Create a subnet" button is visible below the list.

- Attach load balancer- choose application load balancer- LB name should be same as ASG , if you want to edit it you can edit the name.
- select subnets – give PORT NO: 80 for HTTP – Select TARGET GROUP (new or existing).
- Give HEALTH CHECK GRACE PERIOD as your wish.

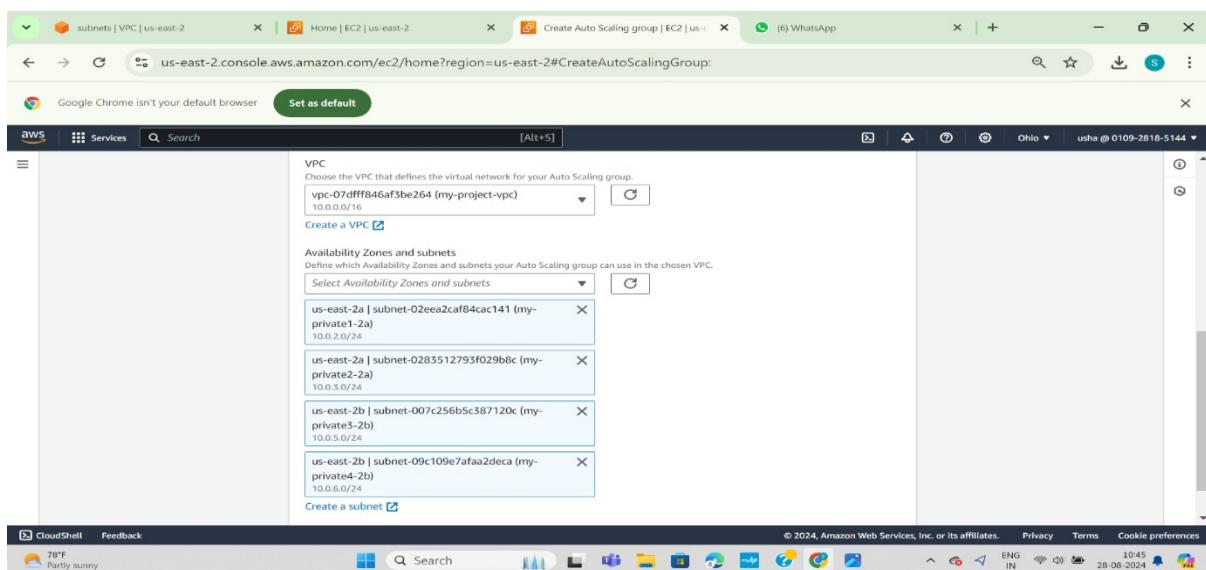
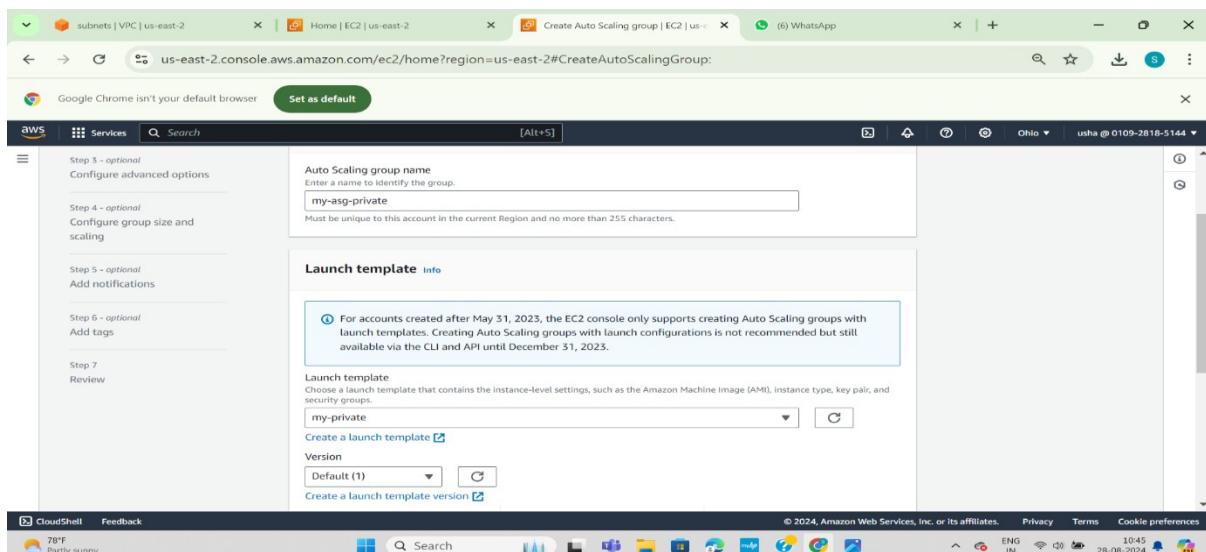
- Select Group size We want to set a minimum and maximum number of instances the ASG can provision:
- Desired capacity: 2
- Minimum capacity: 2
- Maximum capacity: 5
- After that, click on next – next – create auto scaling group.





## Private ASG:

- follow all the steps as above.
- But at network settings, choose 4 private subnets.



**Step 4 - optional**  
**Configure group size and scaling**

**Step 5 - optional**  
**Add notifications**

**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.

**Load balancer scheme**  
 Scheme cannot be changed after the load balancer is created.

**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-07dfff846af3be264

**Availability Zones and subnets**  
 You must select a single subnet for each Availability Zone enabled. Only public subnets are available for selection to support DNS resolution.

us-east-2b  
 us-east-2a  
 subnet-007c256b5c387120c  
 subnet-02eea2caf84cac141

**Listeners and routing**  
 If you require secure listeners, or multiple listeners, you can configure them from the Load Balancing console after your load balancer is created.

Protocol	Port	Default routing (forward to)
HTTP	80	Create a target group <input type="button" value="New target group name"/> <small>An instance target group with default settings will be created.</small>

**Tags - optional**  
 Consider adding tags to your load balancer. Tags enable you to categorize your AWS resources so you can more easily manage them.

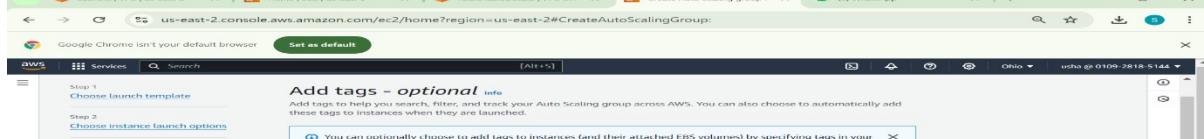
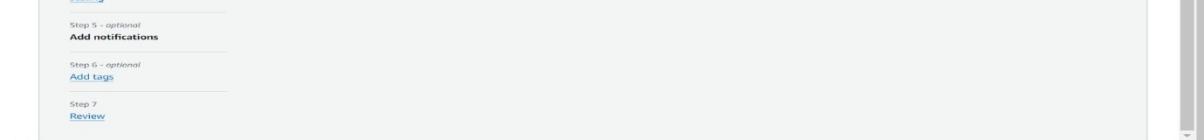
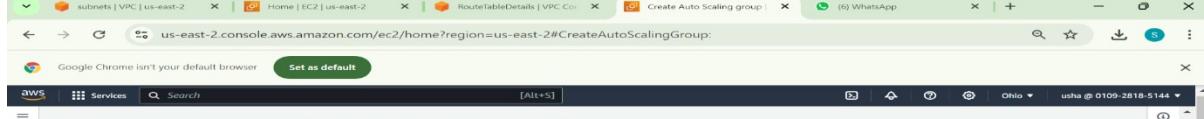
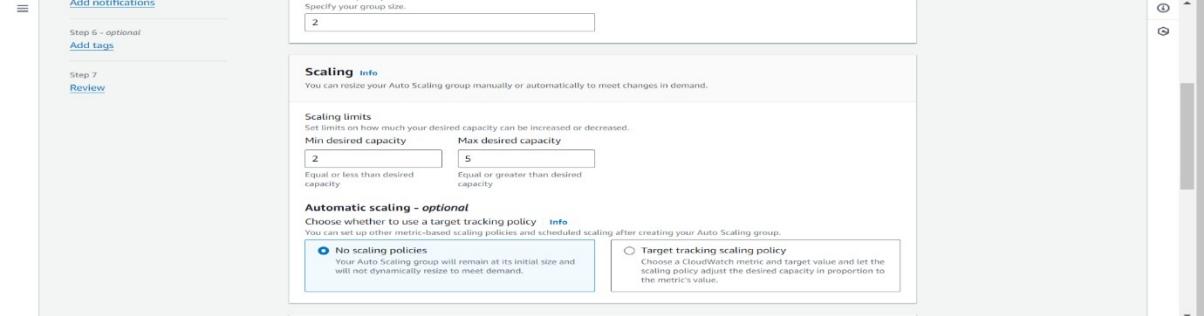
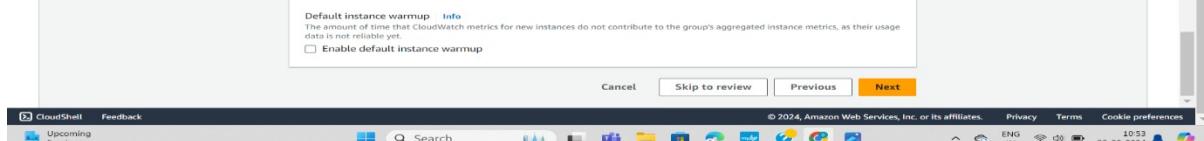
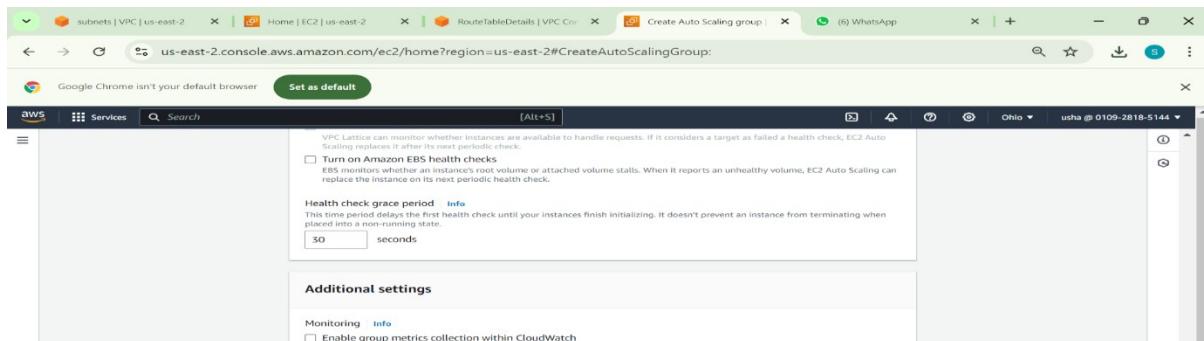
50 remaining

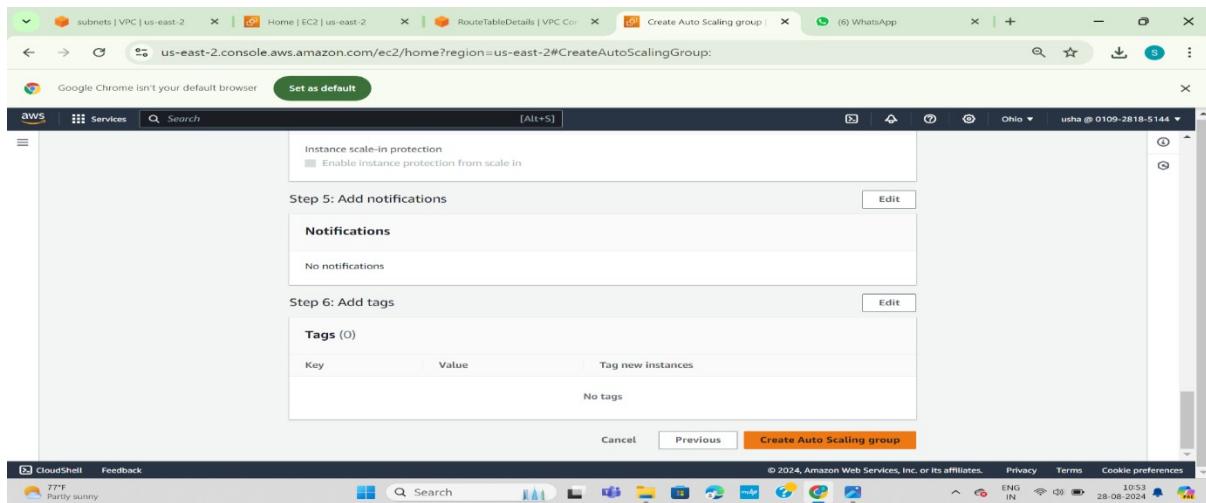
**VPC Lattice integration options** Info  
 To improve networking capabilities and scalability, integrate your Auto Scaling group with VPC Lattice. VPC Lattice facilitates communications between AWS services and helps you connect and manage your applications across compute services in AWS.

**Select VPC Lattice service to attach**

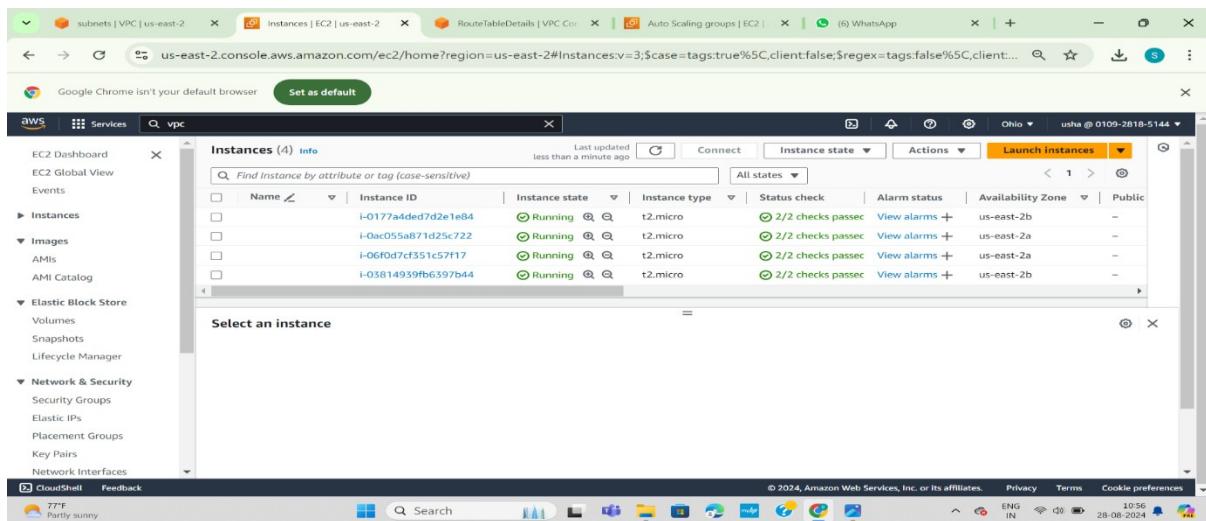
No VPC Lattice service  
 VPC Lattice will not manage your Auto Scaling group's network access and connectivity with other services.

Attach to VPC Lattice service  
 Incoming requests associated with specified VPC Lattice target groups will be routed to your Auto Scaling group.

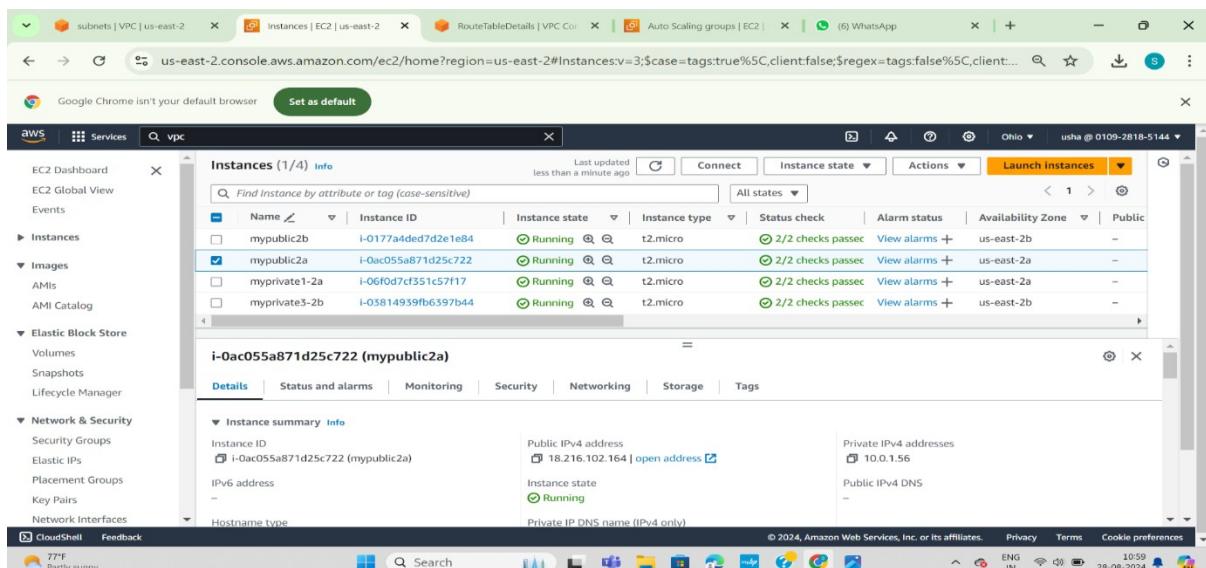




- Now go to EC2 dashboard- click on instances.
- Here we can see the 4 FOUR NEW RUNNING INSTANCES.



- Give the names to those instances as –mypublic-2b,mypublic-2a,myprivate1-2a & myprivate3-2b.



- Click on public -1 instance – click on connect- connect to the EC2 server.
- After connecting to the UBUNTU, give the following commands:
- sudo -i -to become a root user.
- apt update -y – to update packages.
- apt install apache2 – to install apache2 .
- cd /var/www/html – path.
- ls – list.
- rm index.html – to remove index.html.
- vi index.html – enter – press I –insert date --:x (to save) – enter - systemctl status apache2 – to check the server status.
- Ping google.com – to check whether the server is ping or not.
- Copy the PUBLIC IP address & paste it in google chrome.
- It will show the data which we inserted.

The screenshots illustrate the process of connecting to an EC2 instance via EC2 Instance Connect.

**Screenshot 1: AWS EC2 Instances Page**

This screenshot shows the AWS EC2 Instances page for an instance named "i-0177a4ded7d2e1e84 (mypublic2b)". The instance is running in a VPC with a public IPv4 address of 18.191.190.153 and a private IPv4 address of 10.0.4.244. The instance type is t2.micro. The VPC ID is vpc-07dff846af5be264. The instance is associated with a subnet ID of subnet-04825c20a6c73d483 and an Auto Scaling Group named "my-asg-public".

**Screenshot 2: Connect to instance Dialog**

This screenshot shows the "Connect to instance" dialog for the same instance. It displays the public IP address 18.191.190.153 and the connection type options. The "Connect using EC2 Instance Connect" option is selected, with a note explaining it uses a browser-based client. The "Username" field is set to "ubuntu". A note at the bottom states: "Note: In most cases, the default username, ubuntu, is correct. However, read your AMI usage instructions to check if the AMI owner has changed the default AMI username."

**Screenshot 3: AWS EC2 Instance Connect Page**

This screenshot shows the final "EC2 Instance Connect" page where the connection is being established. The progress bar indicates the connection is in progress.

```
aws Services Search [Alt+S] To check for new updates run: sudo apt update

The programs included with the Ubuntu system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*copyright.

Ubuntu comes with ABSOLUTELY NO WARRANTY, to the extent permitted by
applicable law.

To run a command as administrator (user "root"), use "sudo <command>".
See "man sudo_root" for details.

ubuntu@ip-10-0-4-244:~$ sudo -i
root@ip-10-0-4-244:~# apt update -y
Get:1 http://us-east-2.ec2.archive.ubuntu.com/ubuntu noble InRelease
Get:2 http://us-east-2.ec2.archive.ubuntu.com/ubuntu noble-updates InRelease [126 kB]
Get:3 http://us-east-2.ec2.archive.ubuntu.com/ubuntu noble-backports InRelease [126 kB]
Get:4 http://us-east-2.ec2.archive.ubuntu.com/ubuntu noble/universe amd64 Packages [15.0 MB]
Get:5 http://security.ubuntu.com/ubuntu noble-security InRelease [126 kB]
Get:6 http://us-east-2.ec2.archive.ubuntu.com/ubuntu noble/universe Translation-en [5982 kB]
Get:7 http://us-east-2.ec2.archive.ubuntu.com/ubuntu/noble/universe amd64 Components [3871 kB]
Get:8 http://security.ubuntu.com/ubuntu/noble-security/main amd64 Packages [323 kB]
Get:9 http://us-east-2.ec2.archive.ubuntu.com/ubuntu/noble/universe amd64 c-n-f Metadata [301 kB]

i-0177a4ded7d2e1e84 (mypublicic2b)
PublicIP: 18.191.190.153 PrivateIP: 10.0.4.244
```

```
subnets | VPC | us-east-2 | Instance details | EC2 | EC2 Instance Connect | RouteTableDetails | Auto Scaling groups | (6) WhatsApp | + | - | _ | 8 | Google Chrome isn't your default browser | Set as default | AWS Services Search [Alt+S] | us-east-2.console.aws.amazon.com/ec2-instance-connect/ssh?connType=standard&instanceId=i-0177a4ded7d2e1e84&osUser=ubuntu&re... | Search | Ohio | usha @ 0109-2018-5144 | Get:48 http://security.ubuntu.com/ubuntu noble-security/multiverse amd64 c-n-f Metadata [344 B] | Fetched 28.7 MB in 4s (6952 kB/s) | Reading package lists... done | Building dependency tree... done | Reading state information... done | 102 packages can be upgraded. Run 'apt list --upgradable' to see them. | root@ip-10-0-4-244:~# apt install apache2 -y | Reading package lists... done | Building dependency tree... done | Reading state information... done | The following additional packages will be installed: | apache2-bin apache2-data apache2-utils libaprutil1 libaprutil1-dbd-sqlite3 libaprutil1-ldap libaprutil1t64 liblbu5.4-0 ssl-cert | Suggested packages: | apache2-doc apache2-suexec-pristine | apache2-suexec-custom www-browser | The following NEW packages will be installed: | apache2-bin apache2-data apache2-utils libaprutil1 libaprutil1-dbd-sqlite3 libaprutil1-ldap libaprutil1t64 liblbu5.4-0 ssl-cert | 0 upgraded, 10 newly installed, 0 to remove and 102 not upgraded. | Need to get 2083 kB of archives. | After this operation, 8094 kB of additional disk space will be used. | Get:1 https://us-east-2.ec2.archive.ubuntu.com/ubuntu noble/main amd64 libaprutil1t64 amd64 1.7.2-3.1build2 [107 kB] | Get:2 https://us-east-2.ec2.archive.ubuntu.com/ubuntu noble/main amd64 libaprutil1t64 amd64 1.6.3-1.1ubuntu7 [9.9 kB] | Get:3 https://us-east-2.ec2.archive.ubuntu.com/ubuntu noble/main amd64 libaprutil1-dbd-sqlite3 amd64 1.6.3-1.1ubuntu7 [11.2 kB] | Get:4 https://us-east-2.ec2.archive.ubuntu.com/ubuntu noble/main amd64 libaprutil1-ldap amd64 1.6.3-1.1ubuntu7 [9116 B] | Get:5 https://us-east-2.ec2.archive.ubuntu.com/ubuntu noble/main amd64 liblbu5.4-0 amd64 5.4.6-3build2 [166 kB] | i-0177a4ded7d2e1e84 (mypublicI2b) | PublicIPs: 18.191.190.153 PrivateIPs: 10.0.4.244 | CloudShell | Feedback | © 2024, Amazon Web Services, Inc. or its affiliates. | Privacy | Terms | Cookie preferences | ENG | 11:02
```

```
curl -s http://169.254.169.254/latest/meta-data/instance-id
i-0177a4ded7d2e1e84
```

No VM guests are running outdated hypervisor (gemu) binaries on this host.

```
root@ip-10-0-4-2441:~# cd /var/www/html
root@ip-10-0-4-2441:/var/www/html# rm index.html
root@ip-10-0-4-2441:/var/www/html# vi index.html
root@ip-10-0-4-2441:/var/www/html# systemctl restart apache2
root@ip-10-0-4-2441:/var/www/html# cd
root@ip-10-0-4-2441:# systemctl restart apache2
root@ip-10-0-4-2441:# systemctl status apache2
Unknown command verb 'status', did you mean 'status'?
root@ip-10-0-4-2441:# 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 2024-05-20 05:34:01 UTC; 23s ago
     Docs: https://httpd.apache.org/docs/2.4/
Process: 2829 execStart=/usr/sbin/apachectl start (code=exited, status=0/SUCCESS)
Main PID: 2829 (apachectl)
  Tasks: 55 (limit: 1130)
    Memory: 5.4M (peak: 5.5M)
      CPU: 16ms
CGroup: /system.slice/apache2.service
└─2831 /usr/sbin/apache2 -k start
  ├─2833 /usr/sbin/apache2 -k start
  └─2834 /usr/sbin/apache2 -k start
```

i-0177a4ded7d2e1e84 (mypublic2b)

Public IPs: 18.191.190.153 Private IPs: 10.0.4.244

```
Last login: Wed Aug 28 05:30:01 2024 from 3.16.146.4
ubuntu@ip-10-0-4-244:~$ sudo -i
root@ip-10-0-4-244:~# apt install apache2-y
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
apache2 is already the newest version (2.4.58-lubuntu0.4).
0 upgraded, 0 newly installed, 0 to remove and 102 not upgraded.
root@ip-10-0-4-244:~# cd /var/www/html
root@ip-10-0-4-244:/var/www/html$ ping google.com
PING google.com [142.250.191.174] 56(84) bytes of data.
64 bytes from ord36830-in-f14.1e100.net (142.250.191.174): icmp_seq=1 ttl=117 time=0.63 ms
64 bytes from ord36830-in-f14.1e100.net (142.250.191.174): icmp_seq=2 ttl=117 time=0.67 ms
64 bytes from ord36830-in-f14.1e100.net (142.250.191.174): icmp_seq=3 ttl=117 time=0.38 ms
64 bytes from ord36830-in-f14.1e100.net (142.250.191.174): icmp_seq=4 ttl=117 time=0.69 ms
64 bytes from ord36830-in-f14.1e100.net (142.250.191.174): icmp_seq=5 ttl=117 time=0.56 ms
64 bytes from ord36830-in-f14.1e100.net (142.250.191.174): icmp_seq=6 ttl=117 time=0.79 ms
PC
-- google.com ping statistics --
6 packets transmitted, 6 received, 0% packet loss, time 5010ms
rtt min/avg/max/mdev = 0.377/0.618/0.705/0.126 ms
root@ip-10-0-4-244:/var/www/html$
```

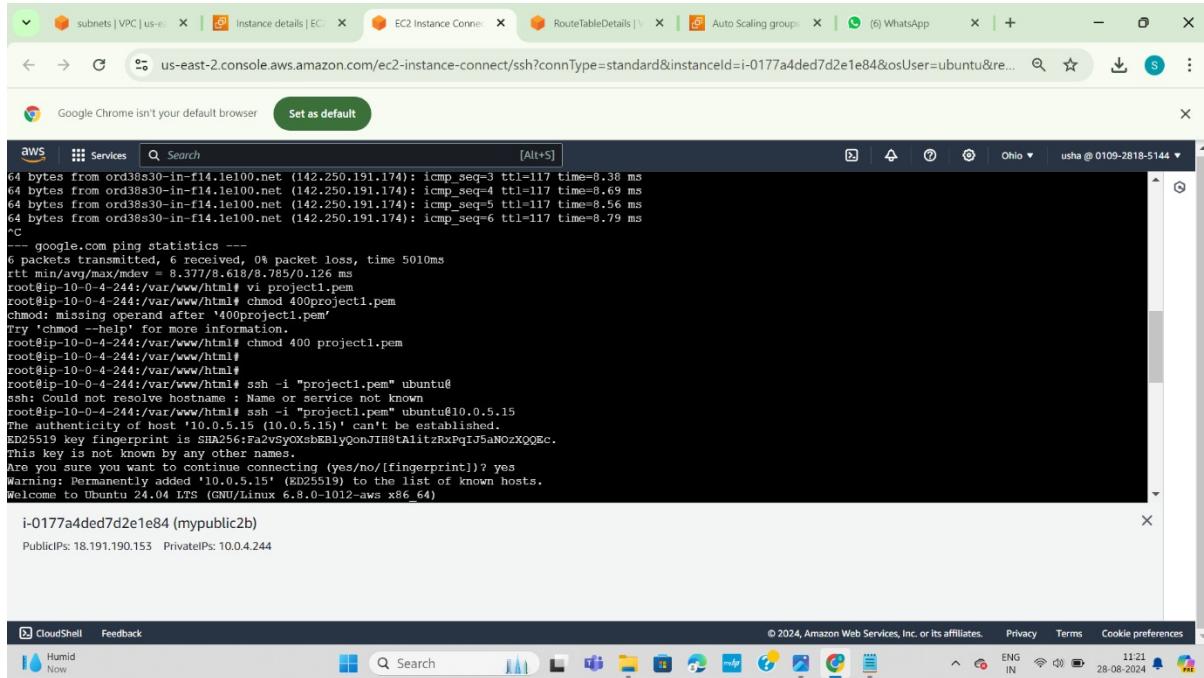
A screenshot of a Windows 10 desktop. The taskbar at the bottom shows several pinned icons: CloudShell, Feedback, Search, File Explorer, Task View, Task Manager, File History, Photos, Mail, Edge, File Explorer, and File History. The Start button is visible in the bottom-left corner. The main screen displays a Microsoft Edge browser window with the URL "us-east-2.console.aws.amazon.com/ec2-instance-connect/ssh?connType=standard&instanceId=i-0177a4ded7d2e1e84&osUser=ubuntu&re...". The page content shows EC2 instance details, including subnet information, instance connection status, route table details, and auto scaling group. Below the browser is a terminal window with a green title bar containing the AWS logo and the text "aws services". The terminal output is a long string of characters representing an RSA private key, starting with "-----BEGIN RSA PRIVATE KEY-----" and ending with "-----END RSA PRIVATE KEY-----". The terminal window has a dark theme with white text on a black background.

CloudShell Feedback © 2024, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences  
80°F Mostly cloudy Search ENG 11:17 28-08-2024

 CloudShell Feedback  Finance Feature  CloudWatch Metrics  Search                                              ENG   28-08-2024          <img alt="Feedback icon" data-bbox="8575 970 8590 985

## Now connect to the PRIVATE INSTANCE through the PUBLIC INSTANCE:

- Follow the steps as mentioned in the snapshots.
- Successfully connected to the PRIVATE INSTANCE.

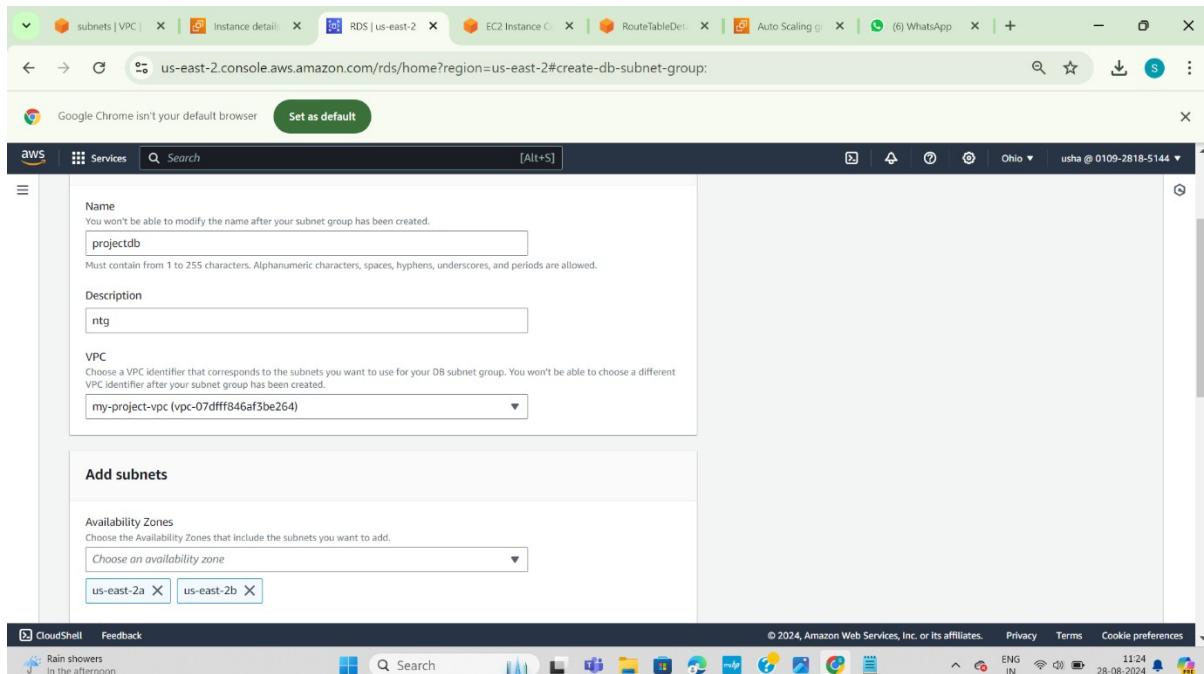


```
64 bytes from ord3s8s30-in-f14.le100.net (142.250.191.174): icmp_seq=3 ttl=117 time=8.38 ms
64 bytes from ord3s8s30-in-f14.le100.net (142.250.191.174): icmp_seq=4 ttl=117 time=8.69 ms
64 bytes from ord3s8s30-in-f14.le100.net (142.250.191.174): icmp_seq=5 ttl=117 time=8.56 ms
64 bytes from ord3s8s30-in-f14.le100.net (142.250.191.174): icmp_seq=6 ttl=117 time=8.79 ms
^C
--- google.com ping statistics ---
6 packets transmitted, 6 received, 0% packet loss, time 5010ms
rtt min/avg/max/mdev = 8.377/8.618/8.785/0.126 ms
root@ip-10-0-4-244:/var/www/html# vi project1.pem
root@ip-10-0-4-244:/var/www/html# chmod 400 project1.pem
chmod: missing operand after '400project1.pem'
Try 'chmod --help' for more information.
root@ip-10-0-4-244:/var/www/html# chmod 400 project1.pem
root@ip-10-0-4-244:/var/www/html#
root@ip-10-0-4-244:/var/www/html#
root@ip-10-0-4-244:/var/www/html# ssh -i "project1.pem" ubuntu@
ssh: Could not resolve hostname: Name or service not known
root@ip-10-0-4-244:/var/www/html# ssh -i "project1.pem" ubuntu@10.0.5.15
The authenticity of host '10.0.5.15 (10.0.5.15)' can't be established.
ED25519 key fingerprint is SHA256:Fa2vSyOxshEbyQondIIR8A1litzRxhqiJ5aNoZXQQEc.
this key is not known by any other names.
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added '10.0.5.15' (ED25519) to the list of known hosts.
Welcome to Ubuntu 24.04 LTS (GNU/Linux 6.8.0-1012-aws x86_64)

i-0177a4ded7d2e1e84 (mypublic2b)
PublicIPs: 18.191.190.153 PrivateIPs: 10.0.4.244
```

## 9. Create Subnet Group

- select create subnet group on rds dashboard.
- give a name to new subnet group-select vpc.
- add availability zones (us-west-2a & us-west-2b) and 6 subnets.



Name: projectdb  
Description: ntg  
VPC: my-project-vpc (vpc-07dff846af3be264)  
Add subnets:  
Availability Zones:  
Choose an availability zone  
us-east-2a X us-east-2b X

Screenshots showing the creation of an RDS DB subnet group named "projectdb" across two Availability Zones (us-east-2a and us-east-2b) using AWS CloudShell.

**Step 1: Selecting Subnets**

The screenshot shows the AWS CloudShell interface. In the top navigation bar, there are tabs for "CloudShell" and "Feedback". Below the navigation bar, a message says "Rain showers In the afternoon". The main content area displays a list of subnets selected from two Availability Zones:

Subnet	Description
subnet-02eea2caf84cac141	(10.0.2.0/24)
subnet-0283512793f029b8c	(10.0.3.0/24)
subnet-0997bdb613658f9fb	(10.0.1.0/24)
subnet-04825c20a6c73d483	(10.0.4.0/24)
subnet-007c256b5c387120c	(10.0.5.0/24)
subnet-09c109e7afaa2deca	(10.0.6.0/24)

A note at the bottom states: "For Multi-AZ DB clusters, you must select 3 subnets in 3 different Availability Zones."

**Step 2: Subnets Selected**

The screenshot shows the AWS CloudShell interface. The "Subnets selected" section now lists all six subnets from both Availability Zones:

Availability zone	Subnet ID	CIDR block
us-east-2a	subnet-02eea2caf84cac141	10.0.2.0/24
us-east-2a	subnet-0283512793f029b8c	10.0.3.0/24
us-east-2a	subnet-0997bdb613658f9fb	10.0.1.0/24
us-east-2b	subnet-04825c20a6c73d483	10.0.4.0/24
us-east-2b	subnet-007c256b5c387120c	10.0.5.0/24
us-east-2b	subnet-09c109e7afaa2deca	10.0.6.0/24

A note at the bottom states: "For Multi-AZ DB clusters, you must select 3 subnets in 3 different Availability Zones."

**Step 3: Success Message**

The screenshot shows the AWS CloudShell interface. A green success message at the top states: "Successfully created projectdb. View subnet group". The main content area shows the "Subnet groups" table with one entry:

Name	Description	Status	VPC
projectdb	ntg	Complete	vpc-07dff846af5be264

## 10. Create Database

- select database on rds dashboard and click create database.
- choose STANDARD in creation method and MYSQL in engine methods.
- select FREE TIER in template.
- choose SELF MANAGED in settings and give PASSWORD.
- go to connectivity and choose CONNECT TO EC2 INSTANCE and select public instance.
- we don't have to give vpc and db subnet group it will take by default.
- choose existing security groups and select both public (public-sg) and private (private-sg) security groups.

The screenshot shows the 'Create database' wizard in the AWS RDS console. The first step, 'Choose a database creation method', has 'Standard create' selected. The 'Easy create' option is also available but not selected. Below this, the 'Engine options' section shows 'Aurora (MySQL Compatible)' and 'Aurora (PostgreSQL Compatible)' as options. The 'Aurora (MySQL Compatible)' option is selected. The interface includes standard AWS navigation elements like CloudShell and Feedback, and a status bar at the bottom indicating 'Heavy rain In about 1.5 hours'.

The screenshot shows the 'Engine options' page for MySQL in the AWS RDS console. It lists several MySQL-compatible engines: Aurora (MySQL Compatible), Aurora (PostgreSQL Compatible), MySQL, MariaDB, PostgreSQL, and Oracle. 'MySQL' is selected. To the right of the MySQL section, there is a detailed description of MySQL as the most popular open source database and a list of its features, including support for database sizes up to 64 TiB, various instance classes, automated backup, and up to 15 read replicas. The interface includes standard AWS navigation elements and a status bar at the bottom.

The screenshot shows the AWS RDS MySQL configuration page. In the 'Templates' section, the 'Free tier' option is selected. Under 'Availability and durability', the 'Multi-AZ DB Cluster' deployment option is chosen. The page also includes sections for 'Deployment options' and 'CloudShell Feedback'.

The screenshot shows the AWS RDS MySQL connectivity configuration page. Under 'Compute resource', the 'Connect to an EC2 compute resource' option is selected. A note states that some VPC settings can't be changed when a compute resource is added. Under 'Network type', the 'IPv4' option is selected. The page also includes sections for 'EC2 instance' and 'CloudShell Feedback'.

The screenshot shows the AWS RDS console with the URL <https://us-east-2.console.aws.amazon.com/rds/home?region=us-east-2#launch-dbinstance>. The page is titled "DB subnet group" and displays configuration options for a new DB instance. It includes sections for "Choose existing" or "Automatic setup" of a subnet group, "Existing DB subnet groups" (listing "projectdb" with 6 Subnets, 2 Availability Zones), "Public access" (set to "Yes"), "VPC security group (firewall)" (set to "Choose existing"), and "CloudShell" and "Feedback" buttons.

The screenshot shows the AWS RDS console with the URL <https://us-east-2.console.aws.amazon.com/rds/home?region=us-east-2#launch-dbinstance>. The page is titled "VPC security group (firewall)" and displays configuration options for a new DB instance. It includes sections for "Choose existing" or "Create new" VPC security groups, "Additional VPC security group" (listing "my-sg-public" and "my-sg-private"), and "Amazon RDS will add a new VPC security group rds-ec2-1 to allow connectivity with your compute resource". Other sections include "Availability Zone" (set to "us-east-2a") and "Certificate authority - optional" (set to "rds-ca-rsa2048-g1 (default) Expiry: May 22, 2061").

The screenshot shows the AWS RDS console with the URL <https://us-east-2.console.aws.amazon.com/rds/home?region=us-east-2#databases>. The page is titled "Amazon RDS" and displays the "Databases" section. A green notification bar at the top says "Successfully set up a connection between database-1 and EC2 Instance i-0ac055ab71d25c722". Below it, a blue banner introduces "Aurora's I/O-Optimized" storage configuration. The main table lists one database entry: "database-1" (Status: Available, Instance: MySQL Community, Region: us-east-2a, Engine: db.t3.micro). The table has columns for DB identifier, Status, Role, Engine, Region & ..., Size, Recommendations, and CPU.

## 11. Take DB Snapshot

- Go to rds dashboard - select snapshot –click on take snapshot.
- select db instance and give a new name to snapshot.
- click on take snapshot

The screenshot shows the 'Take DB Snapshot' dialog box on the AWS RDS console. The 'Preferences' section is visible, showing the snapshot type as 'DB instance' and the DB instance identifier as 'database-1'. The 'Snapshot name' field contains the value 'sai'. The 'Take snapshot' button is highlighted in orange at the bottom right of the dialog.

CloudShell Feedback 80°F Mostly cloudy 80°F Mostly cloudy 11:36 28-08-2024 ENG IN

subnets | VPC | Instance details | RDS | us-east-2 | EC2 Instance C | RouteTableDet | Auto Scaling | (6) WhatsApp | +

Google Chrome isn't your default browser Set as default

aws Services Search [Alt+S]

RDS > Snapshots > Take snapshot

Take DB Snapshot

Preferences To take a DB Snapshot, choose a database and name your DB Snapshot.

Snapshot type  DB instance  DB cluster

DB instance DB Instance identifier. This is the unique key that identifies a DB Instance. database-1

Snapshot name Identifier for the DB Snapshot. sai

Snapshot identifier is case insensitive, but stored as all lower-case, as in "mysnapshot". Cannot be null, empty, or blank. Must contain from 1 to 255 alphanumeric characters or hyphens. First character must be a letter. Cannot end with a hyphen or contain two consecutive hyphens.

Cancel Take snapshot © 2024, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

CloudShell Feedback 80°F Mostly cloudy 80°F Mostly cloudy 11:36 28-08-2024 ENG IN

subnets | VPC | Instance details | RDS | us-east-2 | EC2 Instance C | RouteTableDet | Auto Scaling | (6) WhatsApp | +

Google Chrome isn't your default browser Set as default

aws Services Search [Alt+S]

RDS > Snapshots > Take snapshot

Take DB Snapshot

Preferences To take a DB Snapshot, choose a database and name your DB Snapshot.

Snapshot type  DB instance  DB cluster

DB instance DB Instance identifier. This is the unique key that identifies a DB Instance. database-1

Snapshot name Identifier for the DB Snapshot. sai

Snapshot identifier is case insensitive, but stored as all lower-case, as in "mysnapshot". Cannot be null, empty, or blank. Must contain from 1 to 255 alphanumeric characters or hyphens. First character must be a letter. Cannot end with a hyphen or contain two consecutive hyphens.

Cancel Take snapshot © 2024, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

CloudShell Feedback 80°F Mostly cloudy 80°F Mostly cloudy 11:36 28-08-2024 ENG IN

subnets | VPC | Instance details | Snapshots | RDS | EC2 Instance C | RouteTableDet | Auto Scaling | (6) WhatsApp | +

Google Chrome isn't your default browser Set as default

aws Services Search [Alt+S]

RDS > Snapshots

Creating snapshot sai.

Snapshots

Manual System Shared with me Public Backup service Exports in Amazon S3

Manual snapshots (1)

Filter by manual snapshots

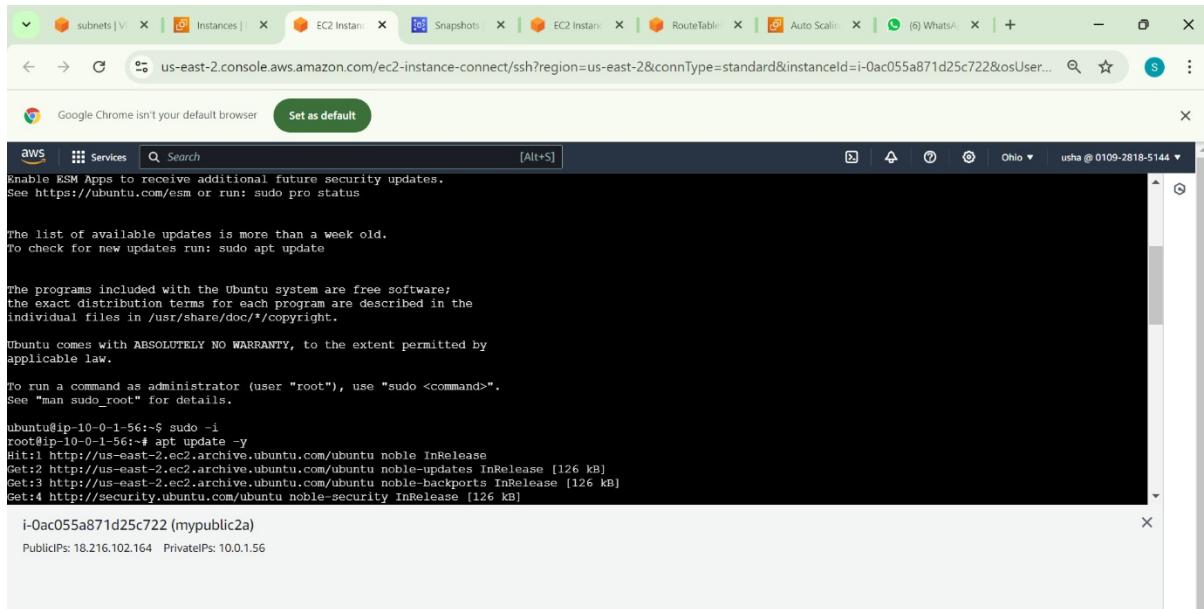
Actions Take snapshot

Snapshot name	DB Instance or cluster	Snapshot creation time	DB Ins
sai	database-1	August	

CloudShell Feedback 80°F Mostly cloudy 80°F Mostly cloudy 11:36 28-08-2024 ENG IN

## 12. MySQL installation

- Go to public instance web and give the following commands.
- Sudo apt update -y.
- Sudo apt install mysql –sever -y
- mysql -h database-1.cv2muissww5k.us-west-1.rds.amazonaws.com -u admin -p  
(database-1.cv2muissww5k.us-west-1.rds.amazonaws.com – end point of database)
- Once you complete giving the commands it displays as following in below snapshots.



```
aws Services Search [Alt+S] [Alt+S]
Enable ESM Apps to receive additional future security updates.
See https://ubuntu.com/esm or run: sudo pro status

The list of available updates is more than a week old.
To check for new updates run: sudo apt update

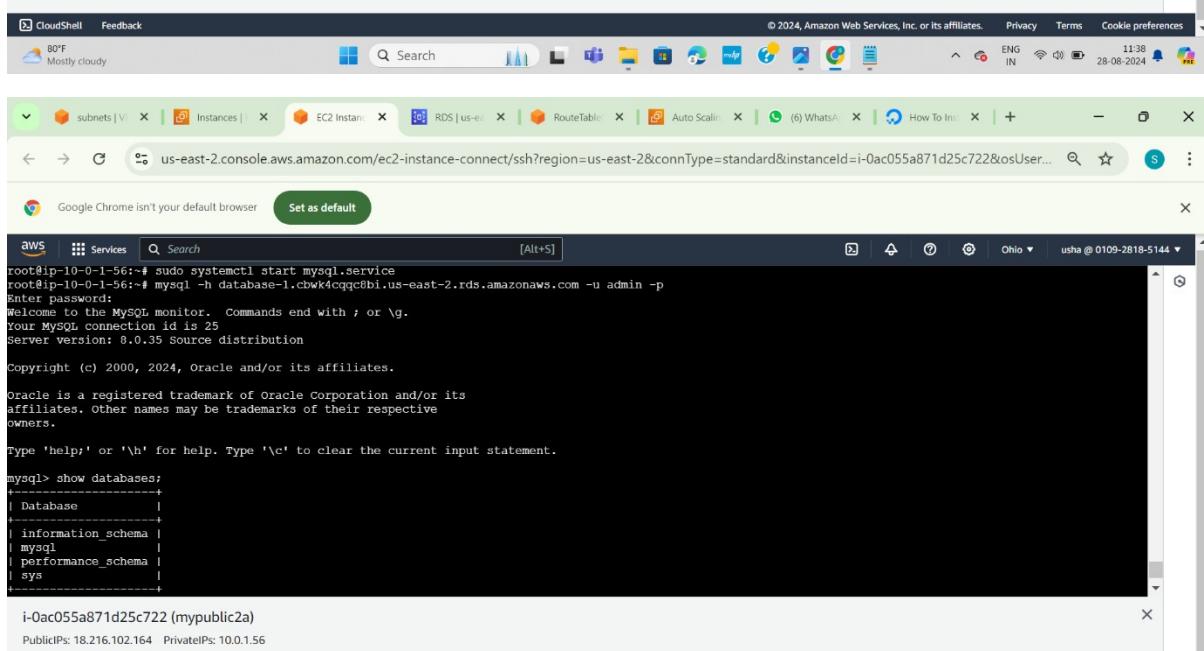
The programs included with the Ubuntu system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*copyright.

Ubuntu comes with ABSOLUTELY NO WARRANTY, to the extent permitted by
applicable law.

To run a command as administrator (user "root"), use "sudo <command>".
See "man sudo_root" for details.

ubuntu@ip-10-0-1-56:~$ sudo -i
root@ip-10-0-1-56:~# apt update -y
Hit:1 http://us-east-2.ec2.archive.ubuntu.com/ubuntu noble InRelease [126 kB]
Get:2 http://us-east-2.ec2.archive.ubuntu.com/ubuntu noble-updates InRelease [126 kB]
Get:3 http://us-east-2.ec2.archive.ubuntu.com/ubuntu noble-backports InRelease [126 kB]
Get:4 http://security.ubuntu.com/ubuntu noble-security InRelease [126 kB]

i-Oac055a871d25c722 (mypublic2a)
PublicIPs: 18.216.102.164 PrivateIPs: 10.0.1.56
```



```
aws Services Search [Alt+S] [Alt+S]
root@ip-10-0-1-56:~# sudo systemctl start mysql.service
root@ip-10-0-1-56:~# mysql -h database-1.cbwk4cqqc8bi.us-east-2.rds.amazonaws.com -u admin -p
Enter password:
Welcome to the MySQL monitor. Commands end with ; or \g.
Your MySQL connection id is 25
Server version: 8.0.35 Source distribution

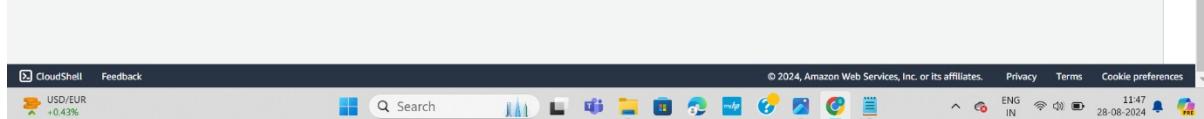
Copyright (c) 2000, 2024, Oracle and/or its affiliates.

Oracle is a registered trademark of Oracle Corporation and/or its
affiliates. Other names may be trademarks of their respective
owners.

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

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

i-Oac055a871d25c722 (mypublic2a)
PublicIPs: 18.216.102.164 PrivateIPs: 10.0.1.56
```



```
aws Services Search [Alt+S] [Alt+S]
CloudShell Feedback © 2024, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences
CloudShell Feedback © 2024, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences
CloudShell Feedback © 2024, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences
```

THE END