

PROJECT-1

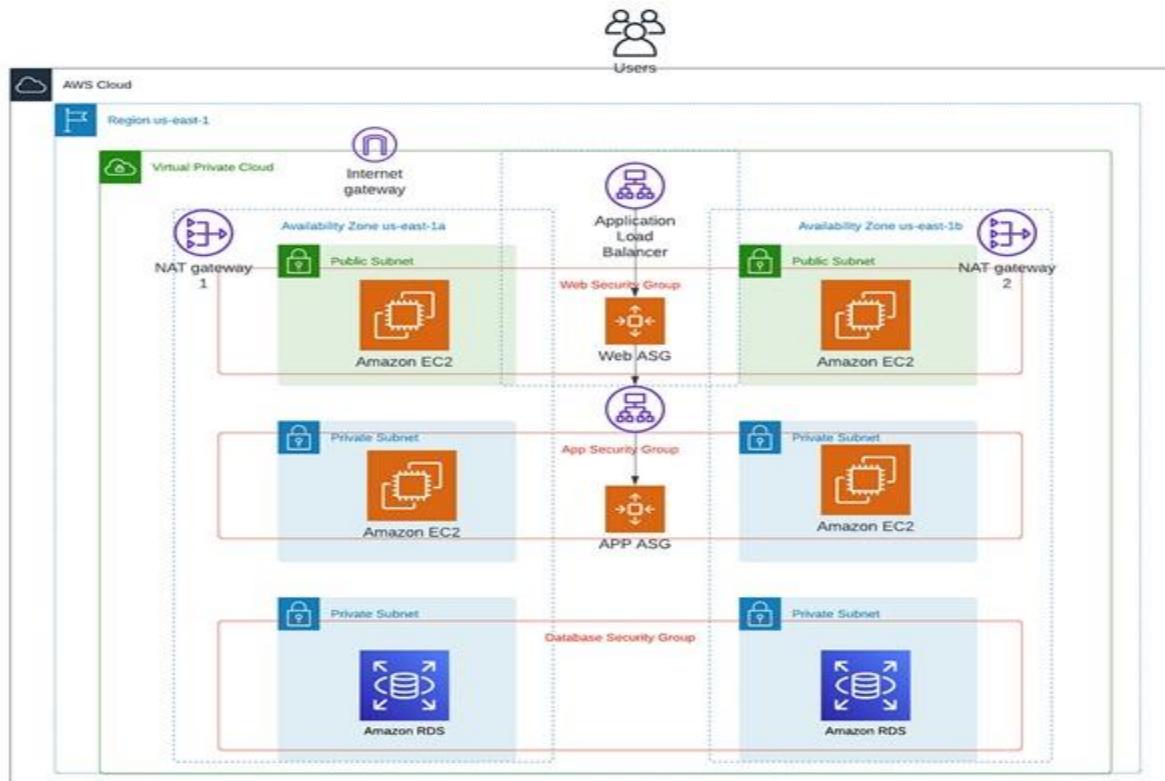
Name: T. Vishnuvardhan

Batch no: 129

Phn no: 8520081327

Gmail: thappetlavishnuvardhan9@gmail.com

Topic: 3-Tier Architecture



Step: 1

Create a VPC with name --> Project-VPC

After creating VPC.

Go to Actions, click on Edit VPC and enable DNS hostnames.

The screenshot shows the AWS VPC dashboard. In the left sidebar, under 'Virtual private cloud', 'Your VPCs' is selected. The main area displays a table titled 'Your VPCs (1/2) Info'. The table has columns: Name, VPC ID, State, IPv4 CIDR, IPv6 CIDR, and DHCP opt. There are two entries: 'Project-VPC' (VPC ID: vpc-0eacc52b630cb26d1, State: Available, IPv4 CIDR: 10.0.0.0/16, IPv6 CIDR: -, DHCP opt: dopt-0135) and another entry (VPC ID: vpc-098575d697ab2e58b, State: Available, IPv4 CIDR: 172.31.0.0/16, IPv6 CIDR: -, DHCP opt: dopt-0135). Below the table, a specific VPC ('vpc-0eacc52b630cb26d1 / Project-VPC') is selected, and its details are shown in the bottom panel. The 'Details' tab is active, showing the following information:

VPC ID vpc-0eacc52b630cb26d1	State Available	DNS hostnames Enabled	DNS resolution Enabled
Tenancy DHCP option set		Main route table	Main network ACL

Step: 2

Create 7 subnets

2 subnets for public, 5 subnets for private --> 2 for APP --> 2 for DB

1 in AZ-c >> IPV4>>10.0.9.0/24

Name: Web-pub1-sub-AZ-a

Name: Web-pub2-sub-AZ-b

IPV4 CIDR block >>10.0.1.0/24

IPV4 CIDR block>>10.0.2.0/24

APP-Sub-AZ-a,b

DB-sub-AZ-a,b

IPV4>>10.0.3.0/24

IPV4>>10.0.7.0/24

IPV4>>10.0.4.0/24

IPV4>>10.0.8.0/24

The screenshot shows the AWS VPC dashboard with the 'Subnets' section selected. The table lists 11 subnets, each with its Name, Subnet ID, State, VPC, and IPv4 CIDR. The subnets are categorized under different VPCs.

Name	Subnet ID	State	VPC	IPv4 CIDR
App-pvt1-sub-Az-a	subnet-0d2399c597195f269	Available	vpc-0eacc52b630cb26d1 Proj...	10.0.3.0/24
Web-pub1-sub-AZ-a	subnet-0c2baa1999ef59e36	Available	vpc-0eacc52b630cb26d1 Proj...	10.0.1.0/24
-	subnet-082c97eea96749db8	Available	vpc-098575d697ab2e58b	172.31.0.0/20
pvt-Sub-c	subnet-03cede8a6ce771dfc	Available	vpc-0eacc52b630cb26d1 Proj...	10.0.9.0/24
-	subnet-d47c6b2dd96046a	Available	vpc-098575d697ab2e58b	172.31.32.0/20
App-pvt2-sub-AZ-b	subnet-0fabef11bfcbb43931e	Available	vpc-0eacc52b630cb26d1 Proj...	10.0.4.0/24
DB1-pvt-sub-a	subnet-02db8d5e186def5d6	Available	vpc-0eacc52b630cb26d1 Proj...	10.0.7.0/24
DB2-pvt-sub-b	subnet-079a88f4360106178	Available	vpc-0eacc52b630cb26d1 Proj...	10.0.8.0/24
-	subnet-0982d49428f6ba300	Available	vpc-098575d697ab2e58b	172.31.48.0/20
-	subnet-07f5fe7e7c4a5b7f2	Available	vpc-098575d697ab2e58b	172.31.16.0/20
Web-pub2-sub-AZ-b	subnet-061c5be652c6f21a0	Available	vpc-0eacc52b630cb26d1 Proj...	10.0.2.0/24

Step: 3

Create an INTERNET GATEWAY (IGW-Public)

Attach to VPC

The screenshot shows the AWS VPC dashboard with the 'Internet gateways' section selected. The table lists two internet gateways, each with its Name, Internet gateway ID, State, VPC ID, and Owner.

Name	Internet gateway ID	State	VPC ID	Owner
IGW-public	igw-0511e6afc3e9e57d1	Attached	vpc-0eacc52b630cb26d1 Project-VPC	851725193783
-	igw-07569eb601df9f746	Attached	vpc-098575d697ab2e58b	851725193783

Step: 4

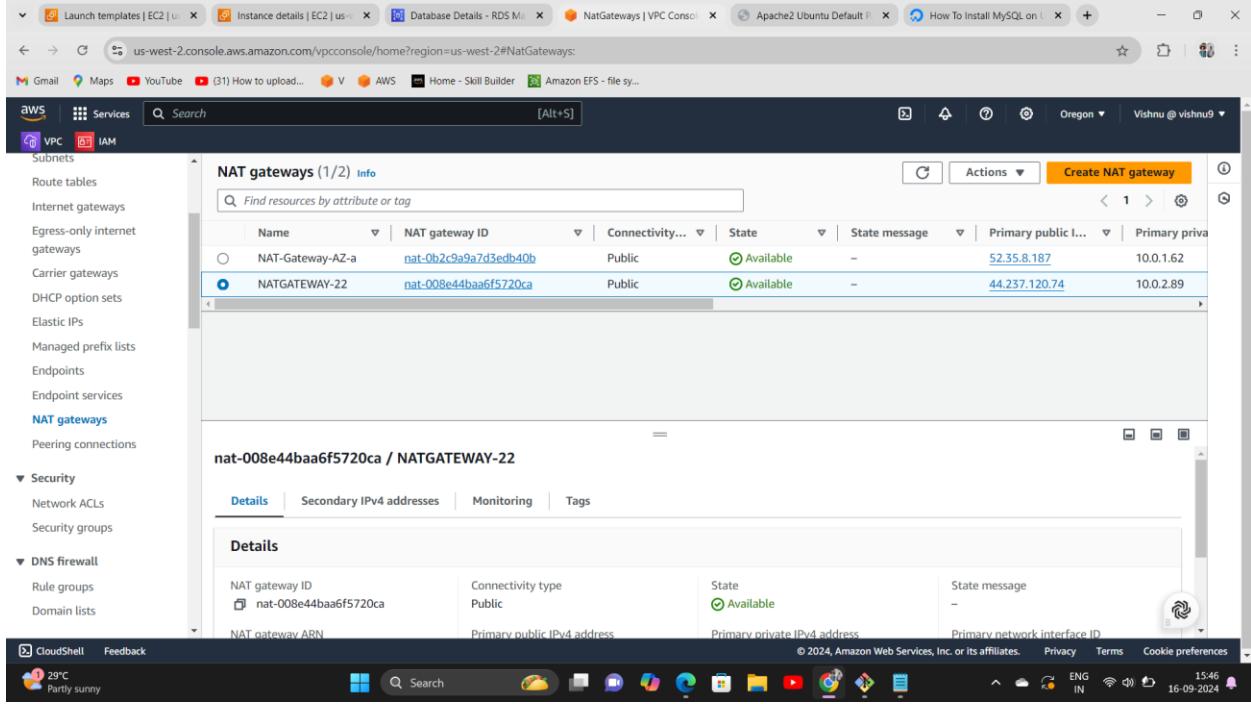
Create a NAT-Gateway

Name: NAT-Gateway-AZ-a

Subnet: Public-AZ-a, Connectivity: Public, Click on Allocate Elastic IP

Name: NATGATEWAY-22

Subnet: Public-AZ-b, Connectivity: Public, Click on Allocate Elastic IP



Name	NAT gateway ID	Connectivity...	State	Primary public I...	Primary priva...
NAT-Gateway-AZ-a	nat-0b2c9a97d3edb40b	Public	Available	52.35.8.187	10.0.1.62
NATGATEWAY-22	nat-008e44baa6f5720ca	Public	Available	44.237.120.74	10.0.2.89

Step: 5

Create 3 route tables

ROUTE-TABLE-1

Name: Web-public-RT, Click on Our VPC and Create

Go to Subnet Associations, Click on Edit and Select public subnets a and b , Click on save

The screenshot shows the AWS VPC Route Tables interface. The user is editing subnet associations for route table `rtb-0a6214ece768f23a1`. In the 'Available subnets (2/7)' section, several subnets are listed with their names, IDs, IPv4 CIDRs, and route table IDs. Subnets `Web-pub1-sub-AZ-a` and `Web-pub2-sub-AZ-b` are selected and highlighted in blue. In the 'Selected subnets' section, the same two subnets are shown. The bottom status bar indicates it's 29°C and partly sunny.

❖ Goto Actions, edit routs and Add route. Now add our Internet gateway

The screenshot shows the AWS VPC Route Tables interface. The user is editing routes for route table `rtb-0a6214ece768f23a1`. The 'Edit routes' section displays two routes. The first route has a destination of `10.0.0.0/16`, a target of `local`, and a status of `Active`. The second route has a destination of `0.0.0.0/0`, a target of `Internet Gateway`, and a status of `Active`. A 'Remove' button is visible next to the second route. At the bottom, there are 'Add route', 'Cancel', 'Preview', and 'Save changes' buttons. The bottom status bar indicates it's 30°C and partly sunny.

ROUTE-TABLE-2

Name: -RT-Pvt-AZ-a, Click on Our VPC and Create

Go to Subnet Associations, Click on Edit and Select APP private subnets a and DB private subnet a, click on save

The screenshot shows the 'Edit subnet associations' page in the AWS VPC console. Under 'Selected subnets', two subnets are selected: 'subnet-02db8d5e186def5d6 / DB1-pvt-sub-a' and 'subnet-0d2399c597195f269 / App-pvt1-sub-Az-a'. A 'Save associations' button is visible at the bottom right.

Name	Subnet ID	IPv4 CIDR	IPv6 CIDR	Route table ID
<input checked="" type="checkbox"/> App-pvt1-sub-Az-a	subnet-0d2399c597195f269	10.0.3.0/24	-	rtb-0d9a03bd3453a9261 / Private-RT
<input type="checkbox"/> Web-pub1-sub-AZ-a	subnet-0c2baa1999ef5e36	10.0.1.0/24	-	rtb-0a6214ece768f23a1 / Web-Public
<input type="checkbox"/> pvt-Sub-c	subnet-03cedeb86ce771dfc	10.0.9.0/24	-	rtb-020200b322e69b59c / Private-RT
<input type="checkbox"/> App-pvt2-sub-AZ-b	subnet-0fabe11bfcb43931e	10.0.4.0/24	-	rtb-020200b322e69b59c / Private-RT
<input checked="" type="checkbox"/> DB1-pvt-sub-a	subnet-02db8d5e186def5d6	10.0.7.0/24	-	rtb-0d9a03bd3453a9261 / Private-RT
<input type="checkbox"/> DB2-pvt-sub-b	subnet-079a8f4360106178	10.0.8.0/24	-	rtb-020200b322e69b59c / Private-RT
<input type="checkbox"/> Web-pub2-sub-AZ-b	subnet-061c5be652c6f21a0	10.0.2.0/24	-	rtb-0a6214ece768f23a1 / Web-Public

The screenshot shows the 'Edit routes' page in the AWS VPC console. A new route entry is being added for destination '10.0.0.0/16' with target 'NAT Gateway' and status 'Active'. The 'Save changes' button is visible at the bottom right.

Destination	Target	Status	Propagated
10.0.0.0/16	local	Active	No
0.0.0.0/0	NAT Gateway	Active	No
	nat-0b2c9a9a7d3edb40b		
	nat-0b2c9a9a7d3edb40b (NAT-Gateway-AZ-a)		
	nat-008e44baa6f5720ca (NAT-Gateway-AZ-b)		

ROUTE-TABLE-3

Name: -RT-Pvt-AZ-b, Click on Our VPC and Create

Go to Subnet Associations, Click on Edit and Select APP private subnets b , DB private subnet b and PVT-c subnet, click on save

Screenshot of the AWS VPC console showing the 'Edit subnet associations' page for route table `rtb-020200b322e69b59c`. The page lists available subnets and selected subnets.

Available subnets (3/7)

Name	Subnet ID	IPv4 CIDR	IPv6 CIDR	Route table ID
App-pvt1-sub-Az-a	subnet-0d2399c597195f269	10.0.3.0/24	-	rtb-0d9a03bd3453a9261 / Private-RT
Web-pub1-sub-AZ-a	subnet-0c2baa1999ef59e36	10.0.1.0/24	-	rtb-0a6214ecc768f23a1 / Web-Public
<input checked="" type="checkbox"/> pvt-Sub-c	subnet-03cede8a6ce771dfc	10.0.9.0/24	-	rtb-020200b322e69b59c / Private-RT
<input checked="" type="checkbox"/> App-pvt2-sub-AZ-b	subnet-fabae11fcfb43931e	10.0.4.0/24	-	rtb-020200b322e69b59c / Private-RT
<input type="checkbox"/> DB1-pvt-sub-a	subnet-02db8d5e186def5d6	10.0.7.0/24	-	rtb-0d9a03bd3453a9261 / Private-RT
<input checked="" type="checkbox"/> DB2-pvt-sub-b	subnet-079a88f4360106178	10.0.8.0/24	-	rtb-020200b322e69b59c / Private-RT
<input type="checkbox"/> Web-pub2-sub-AZ-b	subnet-061c5be652c6f21a0	10.0.2.0/24	-	rtb-0a6214ecc768f23a1 / Web-Public

Selected subnets

- subnet-079a88f4360106178 / DB2-pvt-sub-b
- subnet-0fabe11bfcb43931e / App-pvt2-sub-AZ-b
- subnet-03cede8a6ce771dfc / pvt-Sub-c

CloudShell Feedback © 2024, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences 30°C Partly sunny ENG IN 16:05 16-09-2024

Screenshot of the AWS VPC console showing the 'Edit routes' page for route table `rtb-020200b322e69b59c`. The page displays existing routes and allows adding new ones.

Edit routes

Destination	Target	Status	Propagated
10.0.0.0/16	local	Active	No
0.0.0.0/0	NAT Gateway	Active	No
	nat-008e44baa6f5720ca		

Add route Cancel Preview Save changes

CloudShell Feedback © 2024, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences 30°C Partly sunny ENG IN 16:06 16-09-2024

The screenshot shows the AWS VPC Dashboard with the 'Route tables' section selected. The table lists five route tables:

Name	Route table ID	Explicit subnet associations	Main	VPC
Web-Public-RT	rtb-0a6214ecc768f23a1	2 subnets	-	vpc-0eacc52b630cb26d1
Private-RT-a	rtb-0d9a03bd3453a9261	2 subnets	-	vpc-0eacc52b630cb26d1
-	rtb-0674f859cbf707235	-	-	vpc-0eacc52b630cb26d1
-	rtb-094d2faec652d513b	-	-	vpc-098575d697ab2e58b
Private-RT-b	rtb-0202000b322e69b59c	3 subnets	-	vpc-0eacc52b630cb26d1

Below the table, there is a section titled 'Select a route table'.

Step: 6

Create “3” Security Groups

Web-SG

Inbound rules 1)SHH --> 0.0.0.0/0

2) HTTP -->0.0.0.0/0

3) HTTPS -->0.0.0.0/0

Outbound rules 1) All traffic -->0.0.0.0/0

App-SG

Outbound rules 1) All traffic -->0.0.0.0/0

Inbound rules has mentioned

The screenshot shows the AWS VPC Security Groups console. A security group named "APP-SG" is selected. The details pane shows the security group ID as "sg-0d921befa7ce0150d", owned by "851725193783". It has 4 inbound rules and 2 outbound rules. The "Inbound rules" tab is selected, displaying four entries:

Name	Security group rule...	IP version	Type	Protocol	Port range
-	sgr-03e42979f8019acf	IPv4	SSH	TCP	22
-	sgr-0eaea16d19bb80ac8	IPv4	HTTP	TCP	80
-	sgr-031ef2794c3e79f62	IPv4	MySQL/Aurora	TCP	3306
-	sgr-070333bf4cb18df7	IPv4	Custom TCP	TCP	8080

DB-SG

Outbound rules 1) All traffic -->0.0.0.0/0

Inbound rules have mentioned.

The screenshot shows the AWS VPC Security Groups console. A security group named "DB-SG" is selected. The details pane shows the security group ID as "sg-0665f4d9d156b49b5", owned by "851725193783". It has 4 inbound rules and 1 outbound rule. The "Outbound rules" tab is selected, displaying one entry:

Name	Security group rule...	IP version	Type	Protocol	Port range
-	sar-08fda2a5895c8532f	IPv4	SSH	TCP	22

The screenshot shows the AWS VPC Security Groups console. The left sidebar includes sections for Capacity, Images (AMIs, AMI Catalog), Elastic Block Store (Volumes, Snapshots, Lifecycle Manager), Network & Security (Security Groups, Elastic IPs, Placement Groups, Key Pairs, Network Interfaces), and Load Balancing (Load Balancers, Target Groups). The main area displays a table titled "Security Groups (3/5) Info" with the following data:

Name	Security group ID	Security group name	VPC ID	Description
<input checked="" type="checkbox"/> WEB-SG	sg-0441859b544f55435	Web-SG	vpc-0eacc52b630cb26d1	allow
<input type="checkbox"/> -	sg-05f9f18c5d06fffc96	default	vpc-0eacc52b630cb26d1	default VPC s
<input type="checkbox"/> -	sg-078a05aaa7f743c7f	default	vpc-098575d697ab2e58b	default VPC s
<input checked="" type="checkbox"/> APP-SG	sg-0d921befa7ce0150d	APP-SG	vpc-0eacc52b630cb26d1	allow
<input checked="" type="checkbox"/> DB-SG	sg-0665f4d9d156b49b5	DB-SG	vpc-0eacc52b630cb26d1	allow

At the bottom, there is a note: "Security Groups: sg-0441859b544f55435, sg-0d921befa7ce0150d, sg-0665f4d9d156b49b5". The status bar at the bottom right shows "16:19 16-09-2024".

Step: 7 Creating Web Tier

Create 4 EC2 instances >> 2 in public >> 2 in private

Web-pub1-sub-a -> Ubuntu -> t2. micro(free) -> public-key -> our VPC -> pub-sub-AZ-a -> Auto-assign public Enable -> Web-SG security group -> user data

```
#!/bin/bash
```

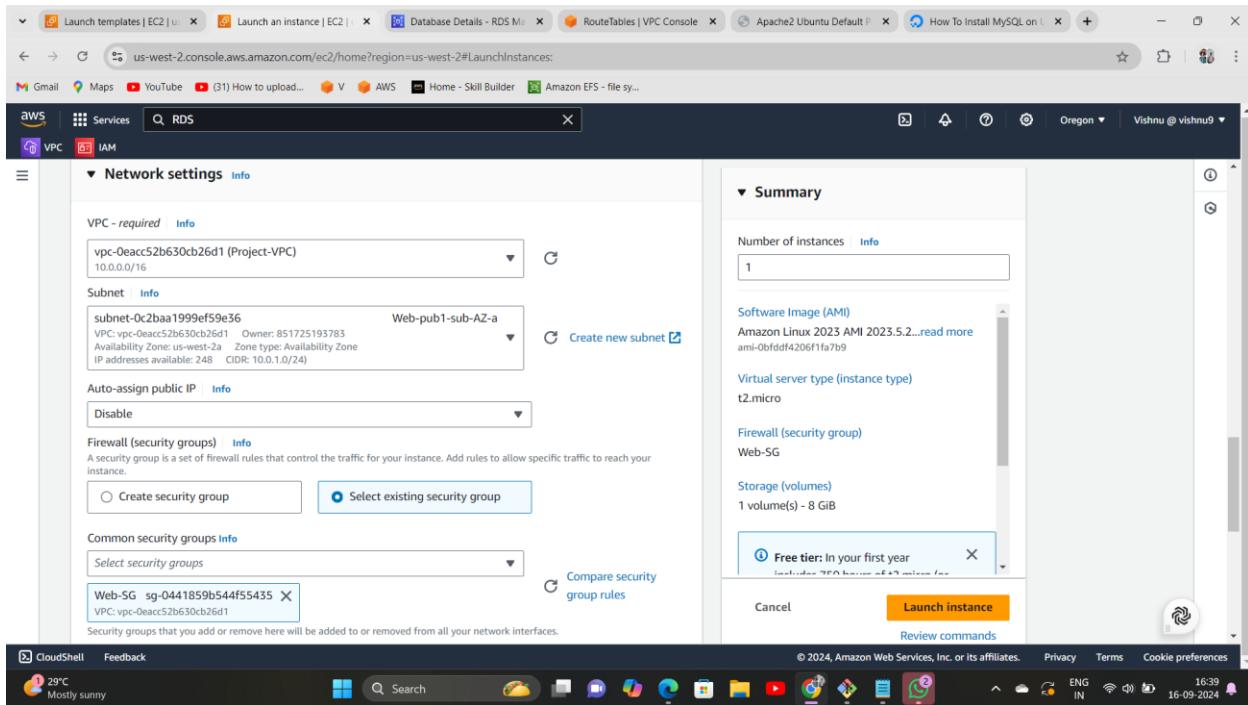
Sudo -i

apt update -y

apt install apache2 -y

Click on Launch instances

Repeat the same process to Another public-2 instance but create in AZ -b



Launching App Tier Instances

App-Pvt1-sub-a -> Ubuntu -> t2. micro(free) -> private-key -> our VPC -> pvt-sub-AZ-a -> Auto-assign public Enable -> App-SG security group -> user data

```
#!/bin/bash
```

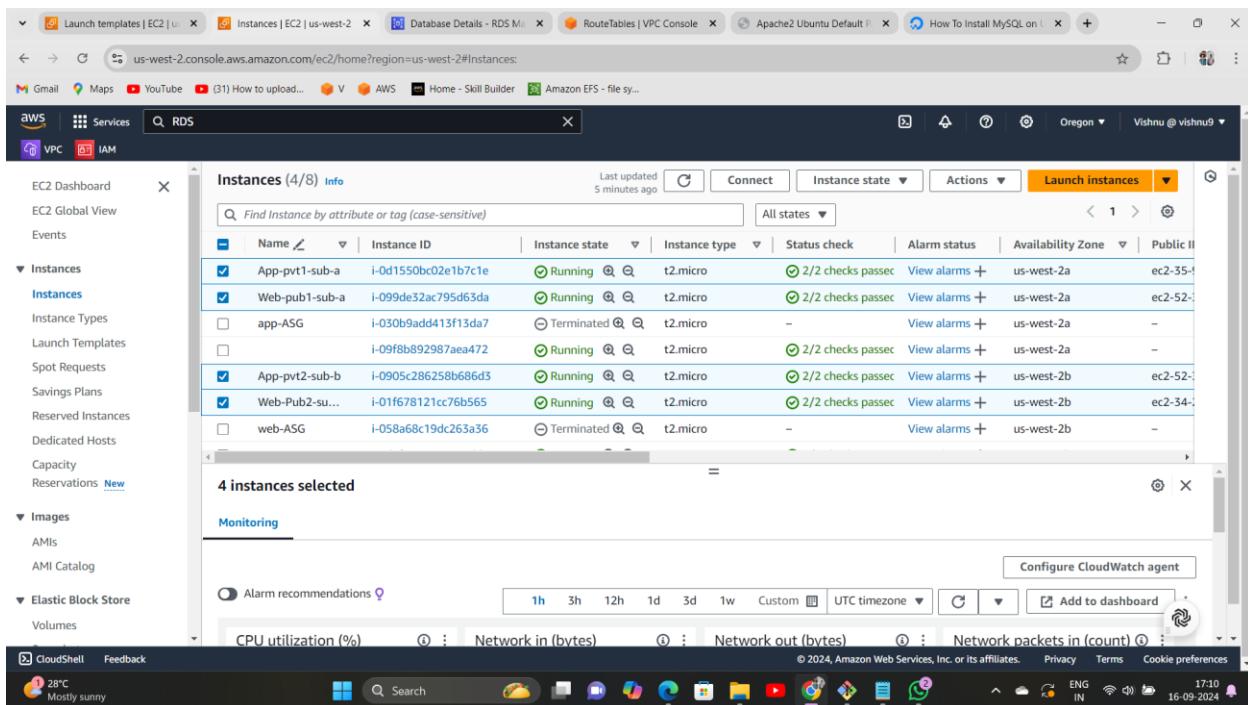
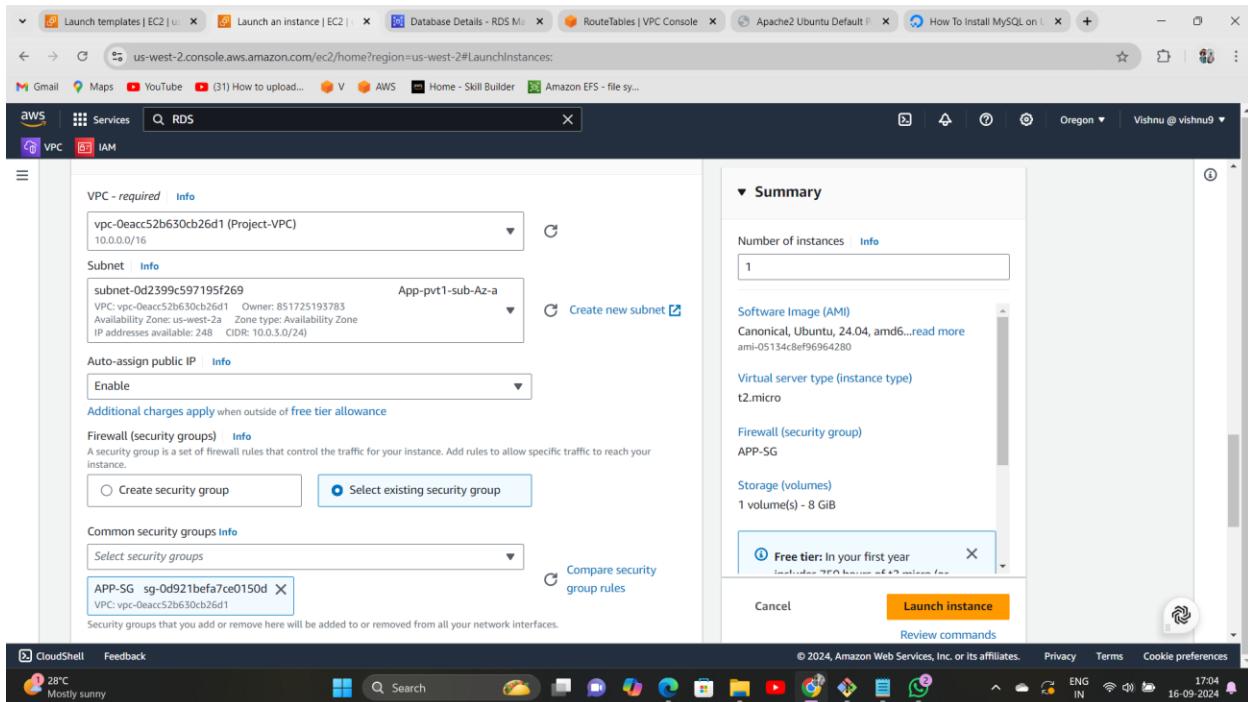
Sudo -i

```
apt update -y
```

```
apt install apache2 -y
```

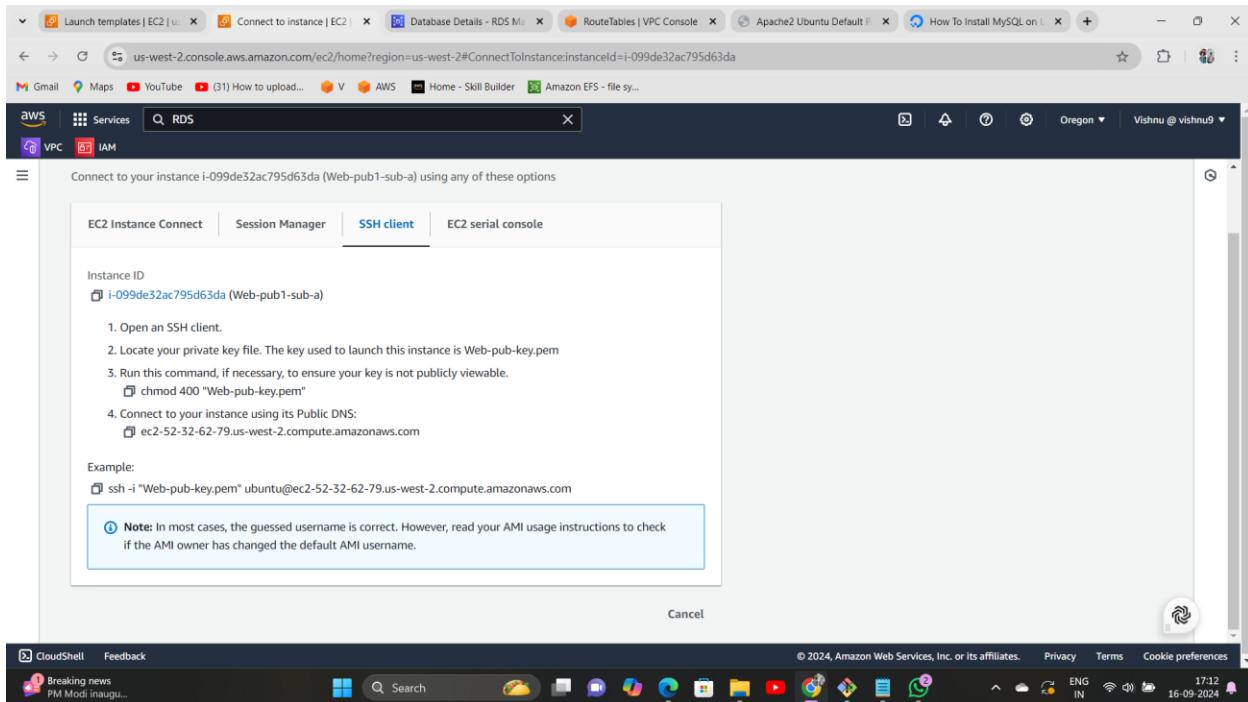
Click on Launch instances

Repeat the same process to Another private-2 instance but create in AZ -b



Step: 8

Connect to Public-1 EC2 instances by SSH Client link



Copy SSH link and paste in git bash

```

root@ip-10-0-3-161:~#
Discord@DELLINSPIRON MINGW64 ~/OneDrive/Desktop (master)
$ ssh -i "Web-pub-key.pem" ubuntu@ec2-52-32-62-79.us-west-2.compute.amazonaws.co
m
The authenticity of host 'ec2-52-32-62-79.us-west-2.compute.amazonaws.com (52.32
.62.79)' can't be established.
ED25519 key fingerprint is SHA256:xxXxGnBuYDSNoZml2bPucuSc5F1GANumx/xKQoWaN1.
This host key is known by the following other names/addresses:
~/.ssh/known_hosts:13: ec2-34-217-84-77.us-west-2.compute.amazonaws.com
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
warning: Permanently added 'ec2-52-32-62-79.us-west-2.compute.amazonaws.com' (ED
25519) to the list of known hosts.
Welcome to Ubuntu 24.04 LTS (GNU/Linux 6.8.0-1015-aws x86_64)

 * Documentation: https://help.ubuntu.com
 * Management: https://landscape.canonical.com
 * Support: https://ubuntu.com/pro

System information as of Mon Sep 16 09:15:01 UTC 2024
System load: 0.24      Processes:          109
Usage of /: 29.4% of 6.71GB  Users logged in:      0
Memory usage: 21%        IPv4 address for enx0: 10.0.1.221
Swap usage:  0%
* Ubuntu Pro delivers the most comprehensive open source security and
  compliance features.
  https://ubuntu.com/aws/pro

Expanded Security Maintenance for Applications is not enabled.

97 updates can be applied immediately.
To see these additional updates run: apt list --upgradable

Enable ESM Apps to receive additional future security updates.
See https://ubuntu.com/esm or run: sudo pro status

Last login: Fri Sep 13 11:30:01 2024 from 103.160.27.99
ubuntu@ip-10-0-1-221:~$ sudo -i

```

Automatically apache2 server will install by user data.

Use these commands to modify index.html file

Sudo –i

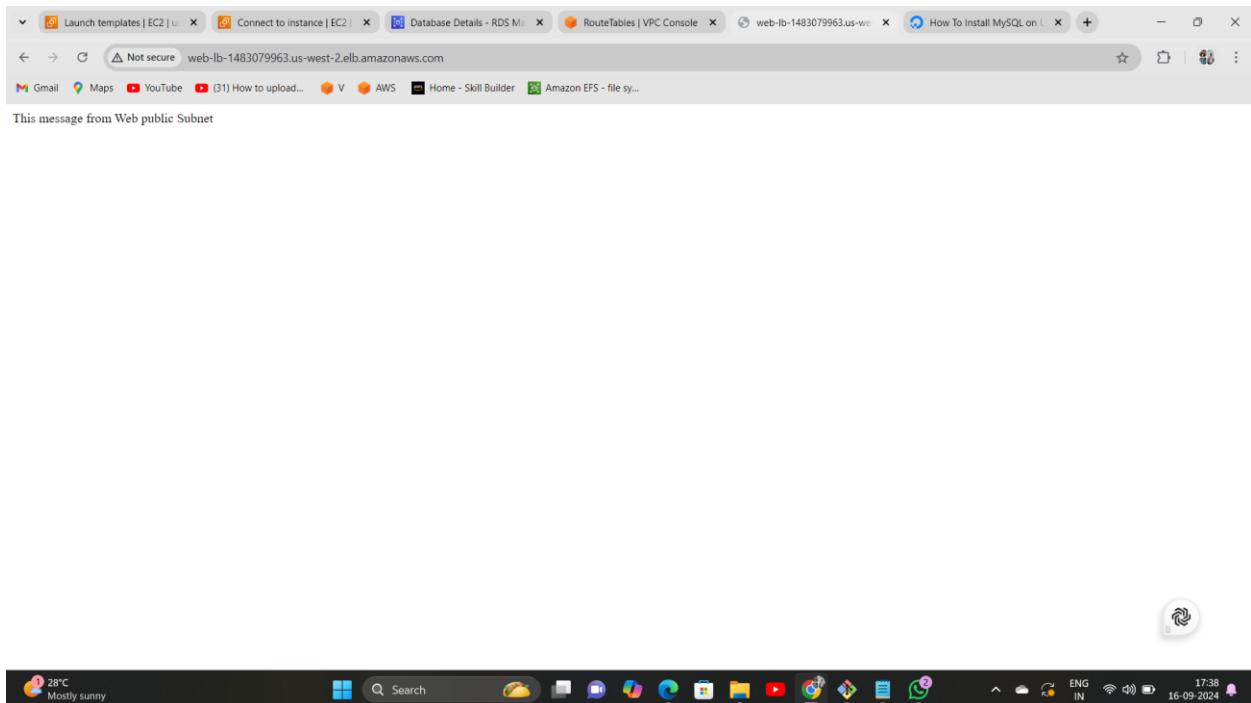
Cd /var/www/html

```
rm index.html
```

Vi index.html --> type a message -> esc+ shift+ semi colon

systemctl restart apache2

Go to URL and paste the public1 PUBLIC IPV4



Repeat same process to public-2

Screenshot of a web browser showing the AWS RDS console. The user is connecting to an instance named i-099de32ac795d63da. The SSH client tab is selected, providing instructions and a command example:

```

Connect to your instance i-099de32ac795d63da (Web-pub1-sub-a) using any of these options

EC2 Instance Connect | Session Manager | SSH client | EC2 serial console

Instance ID
i-099de32ac795d63da (Web-pub1-sub-a)

1. Open an SSH client.
2. Locate your private key file. The key used to launch this instance is Web-pub-key.pem
3. Run this command, if necessary, to ensure your key is not publicly viewable.
   chmod 400 "Web-pub-key.pem"
4. Connect to your instance using its Public DNS:
   ec2-52-32-62-79.us-west-2.compute.amazonaws.com

Example:
ssh -i "Web-pub-key.pem" ubuntu@ec2-52-32-62-79.us-west-2.compute.amazonaws.com

Note: In most cases, the guessed username is correct. However, read your AMI usage instructions to check if the AMI owner has changed the default AMI username.

```

The browser window also shows other tabs like Launch templates, Connect to instance, Database Details, RouteTables, Apache2 Ubuntu Default, and How To Install MySQL on.

Below the browser is a screenshot of a Windows desktop environment showing a terminal window with root privileges. The terminal output includes:

```

root@ip-10-0-1-161: ~
[1] DSCD@DELLINSPIRON MINGW64 ~/OneDrive/Desktop (master)
$ ssh -i "web-pub-key.pem" ubuntu@ec2-52-32-62-79.us-west-2.compute.amazonaws.co
m
The authenticity of host 'ec2-52-32-62-79.us-west-2.compute.amazonaws.com (52.32
.62.79)' can't be established.
ED25519 key fingerprint is SHA256:xVxxdGnbUYDSNOzml2bPUcUSc5F1GANumx/xkQOWaNI.
This host key is known by the following other names/addresses:
  ./ssh/known_hosts:13: ec2-34-217-84-77.us-west-2.compute.amazonaws.com
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added 'ec2-52-32-62-79.us-west-2.compute.amazonaws.com' (ED
25519) to the list of known hosts.
Welcome to Ubuntu 24.04 LTS (GNU/Linux 6.8.0-1015-aws x86_64)

 * Documentation: https://help.ubuntu.com
 * Management: https://landscape.canonical.com
 * Support: https://ubuntu.com/pro

System information as of Mon Sep 16 09:15:01 UTC 2024
System load: 0.24 Processes: 109
Usage of /: 29.4% of 6.71GB Users logged in: 0
Memory usage: 21% IPV4 address for enx0: 10.0.1.221
Swap usage: 0%

* Ubuntu Pro delivers the most comprehensive open source security and
  compliance features.
  https://ubuntu.com/aws/pro

Expanded Security Maintenance for Applications is not enabled.

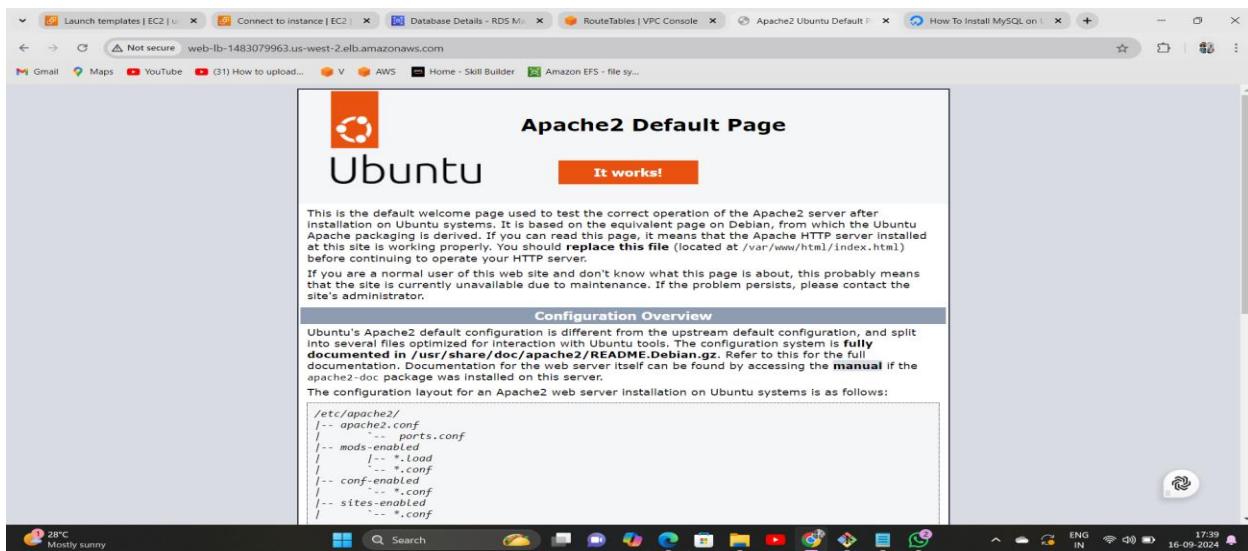
97 updates can be applied immediately.
To see these additional updates run: apt list --upgradable

Enable ESM Apps to receive additional future security updates.
See https://ubuntu.com/esm or run: sudo pro status

Last login: Fri Sep 13 11:30:01 2024 from 103.160.27.99
ubuntu@ip-10-0-1-221:~$ sudo -i

```

The desktop taskbar shows various icons for applications like File Explorer, Edge, and other system tools. The system tray indicates it's 17:12, ENG IN, and the date is 16-09-2024.



Step: 9

Create a Target Group for both Public and Private .

Instance -> Name(MyTG) -> our VPC -> next -> select web-public instance (a,b)
-> include as pending -> create.

The screenshot shows the AWS EC2 Target Groups console. On the left, there is a navigation sidebar with various EC2-related options like Instances, Images, and Elastic Block Store. The main area is titled 'Target groups (1/2) Info'. It shows a table with one entry: 'My-tG' (arn:aws:elasticloadbalancing:us-west-2:851725193783:targetgroup/My-tG/2923784647b7b57b). The table columns include Name, ARN, Port, Protocol, Target type, and Load balancer. Below the table, there is a detailed view of the target group 'My-tG' with tabs for Details, Targets, Monitoring, Health checks, Attributes, and Tags. The 'Details' tab shows the ARN and protocol information. The bottom right corner of the screenshot shows the AWS footer with copyright information and links to Privacy, Terms, and Cookie preferences.

Instance -> Name(TG-2) -> our VPC -> next -> select App-private instance (a,b) -> include as pending -> create.

The screenshot shows the AWS Cloud Console with the 'Target groups' page open. The left sidebar shows navigation options like EC2 Dashboard, Instances, Images, and Elastic Block Store. The main area shows a table of target groups:

Name	ARN	Port	Protocol	Target type	Load balancer
My-tG	arn:aws:elasticloadbalancing:us-west-2:851725193783:targetgroup/TG-2/d8cf91aad53cd598	80	HTTP	Instance	Web-LB
TG-2	arn:aws:elasticloadbalancing:us-west-2:851725193783:targetgroup/TG-2/d8cf91aad53cd598	80	HTTP	Instance	App-LB

The 'TG-2' row is selected. Below the table, a detailed view for 'Target group: TG-2' is shown with tabs for Details, Targets, Monitoring, Health checks, Attributes, and Tags. The 'Details' tab is active, displaying:

Target type	Protocol : Port	Protocol version	VPC
Instance	HTTP: 80	HTTP1	vpc-0eacc52b630cb26d1
IP address type	Load balancer		

The VPC link 'vpc-0eacc52b630cb26d1' is highlighted with a blue box.

Step: 10

Create “2” Load balancers one for public(a,b) instances and another is private(a,b) instances.

- 1) Name -> our VPC -> AZ(a,b) -> SG(Web-SG) -> Target group (mytg) -> create.
- 2) Name -> our VPC -> AZ(a,b) -> SG(App-SG) -> Target group (TG-2) -> create.

The screenshot shows the AWS CloudShell interface. At the top, there are several browser tabs: 'Load balancers | EC2 | us-west-2...', 'Connect to instance | EC2 | us-west-2...', 'Database Details - RDS MySQL | us-west-2...', 'RouteTables | VPC Console | us-west-2...', 'Apache2 Ubuntu Default | us-west-2...', and 'How To Install MySQL on | us-west-2...'. Below the tabs, the AWS Management Console navigation bar is visible with links for 'Services' (selected), 'Search', and 'Create load balancer'. On the left, a sidebar menu includes 'EC2 Dashboard', 'EC2 Global View', 'Events', 'Instances' (selected), 'Images', 'Elastic Block Store', and 'CloudShell Feedback'. The main content area displays a table titled 'Load balancers (2)'. The table has columns for 'Name', 'DNS name', 'State', 'VPC ID', 'Availability Zones', and 'Type'. It lists two entries: 'Web-Lb' (DNS name: Web-Lb-1483079963.us-west-2..., State: Active, VPC ID: vpc-0eacc52b630cb26d1, Availability Zones: 2 Availability Zones, Type: application) and 'App-LB' (DNS name: App-LB-1063789029.us-west-2..., State: Active, VPC ID: vpc-0eacc52b630cb26d1, Availability Zones: 2 Availability Zones, Type: application). A message at the bottom says '0 load balancers selected' and 'Select a load balancer above.'

This screenshot is identical to the one above, showing the AWS CloudShell interface with the AWS Management Console. The navigation bar, sidebar menu, and main content area displaying the 'Load balancers' table are all present.

Click on Web-LB and Copy the DNS and Paste in Google.

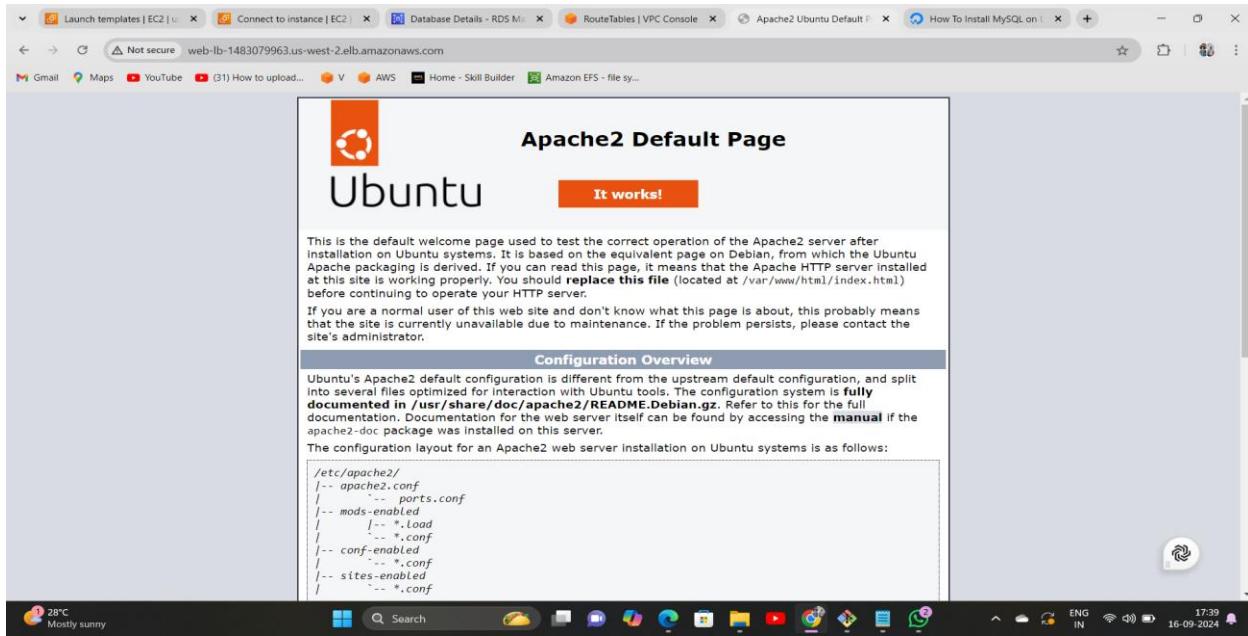
If you refresh the WEB PAGE alternatively two instances output will DISPLAY

The screenshot shows the AWS VPC console with the URL us-west-2.console.aws.amazon.com/ec2/home?region=us-west-2#LoadBalancers:loadBalancerArn=arn:aws:elasticloadbalancing:us-west-2:851725193783:loadbalancer/app/Web-Lb/65db3c30d805. The page displays the details of an Application Load Balancer named "Web-Lb". The "Details" section shows the following configuration:

Load balancer type	Status	VPC	Load balancer IP address type
Application	Active	vpc-0eacc52b650cb26d1	IPv4
Scheme	Internet-facing	Hosted zone Z1H1FL5HABSF5	Availability Zones subnet-0c2baa1999ef59e36 us-west-2a (usw2-az2) subnet-061c5be652c6f21a0 us-west-2b (usw2-az1)
Load balancer ARN	arn:aws:elasticloadbalancing:us-west-2:851725193783:loadbalancer/app/Web-Lb/65db3c30d24dc805		

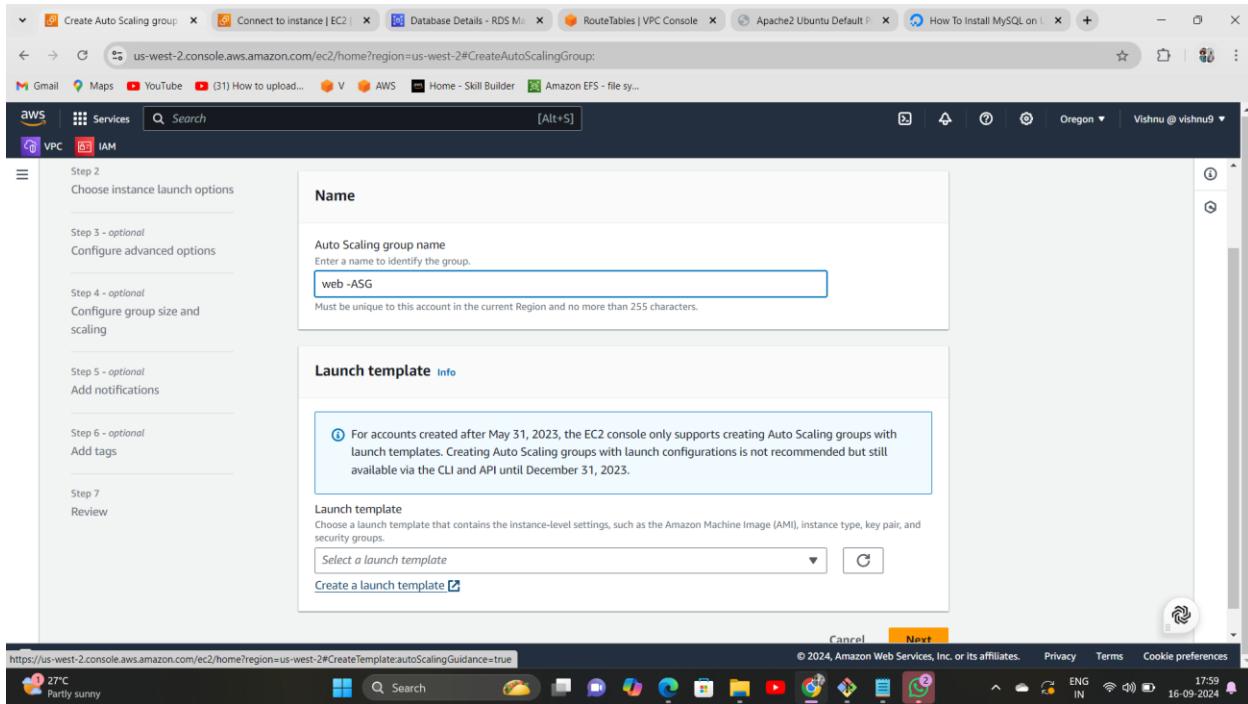
A tooltip "DNS name copied" is visible over the ARN field. Below the details, there are tabs for "Listeners and rules", "Network mapping", "Resource map - new", "Security", "Monitoring", "Integrations", "Attributes", and "Tags". The status bar at the bottom shows "CloudShell Feedback", the date "16-09-2024", and the time "17:54".

The screenshot shows a web browser window with the URL [Not secure web-lb-1483079963.us-west-2.elb.amazonaws.com](http://web-lb-1483079963.us-west-2.elb.amazonaws.com). The page content is minimal, displaying the message "This message from Web public Subnet". The browser's status bar shows "CloudShell Feedback", the date "16-09-2024", and the time "17:58".



Step: 11

Create “2” Auto Scaling for both Web-Tier (Public EC2) and App-Tier(Private EC2)



Before that Create Launch Templates one for WEB and one for APP

The screenshot shows the AWS EC2 Launch Templates page. On the left, there's a navigation sidebar with options like Instances, Images, and Elastic Block Store. The main area displays a table of launch templates:

Launch Template ID	Launch Template Name	Default Version	Latest Version	Create Time	Created ...
lt-093ca646b7df53226	Web-ASG	1	1	2024-09-16T08:56:57.000Z	arn:aws:la...
lt-04eebfdf1cfb2c811	App-LT	1	1	2024-09-16T09:02:40.000Z	arn:aws:la...

A modal window for "Web-ASG (lt-093ca646b7df53226)" is open, showing its details. The "Launch template details" section includes the launch template ID (lt-093ca646b7df53226), name (Web-ASG), default version (1), and owner (arn:aws:iam::851725193783:user/Vishnu).

Select -> our VPC → select public instances AZ a and b subnets -> next ->
Desired (1) ->Min(1) -> Max(5) -> Create Auto Scaling

The screenshot shows the AWS Auto Scaling groups page. The left sidebar has the same navigation as the previous screen. The main area shows a table of auto scaling groups:

Name	Launch template/configuration	Instances	Status	Desired capacity	Min
App-ASG	App-LT Version Default	1	-	1	1
Web-ASG	Web-ASG Version Default	1	-	1	1

A modal window for "Auto Scaling group: Web-ASG" is open, showing "Group details". The details include the Auto Scaling group name (Web-ASG), Desired capacity (1), Desired capacity type (Units (number of instances)), and Amazon Resource Name (ARN) (arn:aws:autoscaling:us-west-2:851725193783:autoScalingGroup:xc2f169ef-...).

Select -> our VPC → select Private instances AZ a and b subnets -> next ->
Desired (1) ->Min(1) -> Max(5) -> Create Auto Scaling

The screenshot shows the AWS Auto Scaling Groups page. On the left, there's a sidebar with navigation links like EC2 Dashboard, EC2 Global View, Events, Instances, Images, and Elastic Block Store. The main content area has tabs for 'Auto Scaling groups (1/2)' and 'Launch configurations'. Under 'Auto Scaling groups', there's a table with two rows:

Name	Launch template/configuration	Instances	Status	Desired capacity	Min
App-ASG	App-LT Version Default	1	-	1	1
Web-ASG	Web-ASG Version Default	1	-	1	1

Below this, there's a detailed view for the 'App-ASG' group, showing its group details and configuration.

Now Go to EC2 instance Dash board there you can see 2 new instance which we auto scaled Public and Private instances

Named as App-ASG and Web-ASG

The screenshot shows the AWS Instances page. The sidebar includes links for EC2 Dashboard, EC2 Global View, Events, Instances, Images, and Elastic Block Store. The main area displays a table of instances:

Name	Instance ID	Instance state	Instance type	Status check	Alarm status	Availability Zone	Public IP
App-pvt1-sub-a	i-0d1550bc02e1b7c1e	Running	t2.micro	2/2 checks passed	View alarms +	us-west-2a	ec2-35-4
Web-pub1-sub-a	i-099de32ac795d63da	Running	t2.micro	2/2 checks passed	View alarms +	us-west-2a	ec2-52-3
App-ASG	i-09f86892987ae472	Running	t2.micro	2/2 checks passed	View alarms +	us-west-2a	-
App-pvt2-sub-b	i-0905c28625b686d3	Running	t2.micro	2/2 checks passed	View alarms +	us-west-2b	ec2-52-2
Web-Pub2-su...	i-01f678121cc76b565	Running	t2.micro	2/2 checks passed	View alarms +	us-west-2b	ec2-34-2
Web-ASG	i-d1f64221a45e6dda	Running	t2.micro	2/2 checks passed	View alarms +	us-west-2b	-

At the bottom, there are monitoring and dashboard options for the selected instances.

Step: 12

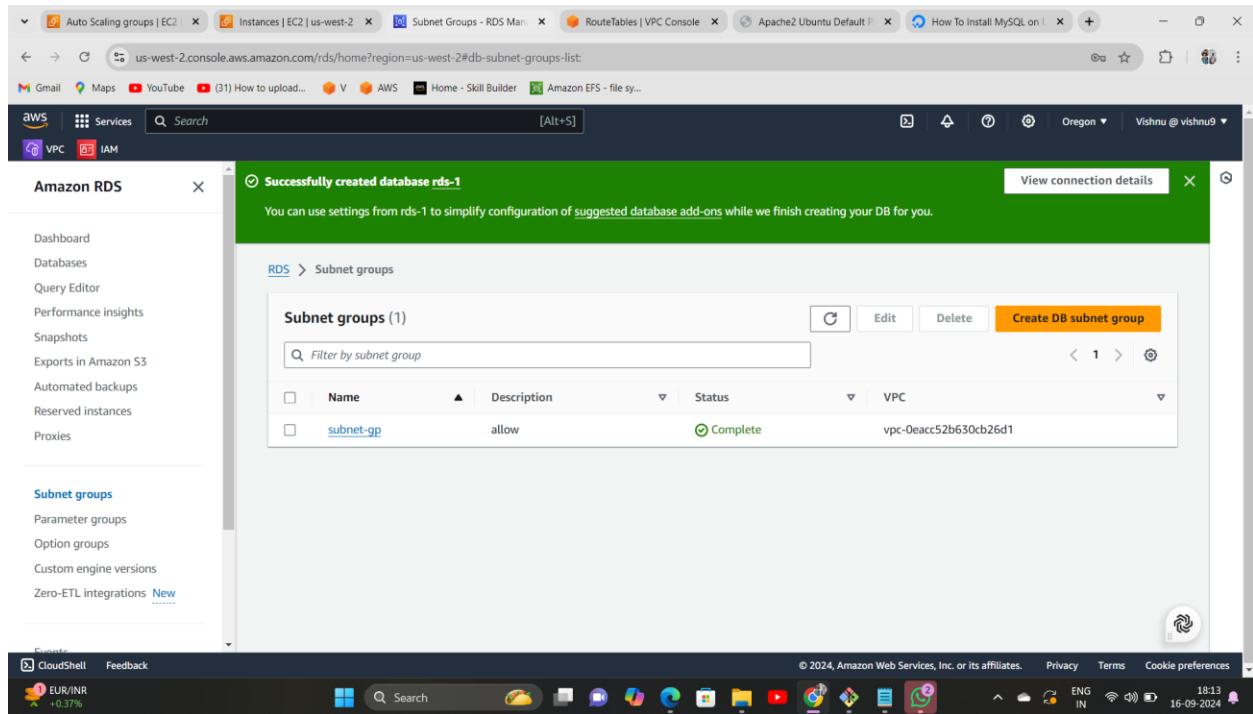
Now goto RDS Dash Board and Select Subnet Group

Create a Subnet group

The screenshot shows the 'Create DB subnet group' wizard in the AWS RDS Management Console. The 'Subnet group details' step is selected. In the 'Name' field, 'subnet-gp' is entered. In the 'Description' field, 'allow' is entered. Under 'VPC', 'Project-VPC (vpc-0eacc52b630cb26d1)' is selected. At the bottom, there is a 'Next Step' button.

Select DB Private subnets which we Created in AZ a ,b and c

The screenshot shows the 'Add subnets' step of the 'Create DB subnet group' wizard. Under 'Availability Zones', 'us-west-2a', 'us-west-2b', and 'us-west-2c' are selected. Under 'Subnets', three subnets are selected: 'subnet-02db8d5e186def5d6 (10.0.7.0/24)', 'subnet-03cede8a6ce771dfc (10.0.9.0/24)', and 'subnet-079a88f4360106178 (10.0.8.0/24)'. At the bottom, there is a 'Next Step' button.



Step: 13

Goto database and Create Database

Select Standard Create -> Engine MYSQL -> Enable Multi-AZ DB Cluster -> Name (RDS-1) -> Self-Managed -> Enter Password -> Create DB

Auto Scaling groups | EC2 Instances | EC2 us-west-2 Create database - RDS Ma RouteTables | VPC Console Apache2 Ubuntu Default How To Install MySQL on ...

Gmail Maps YouTube (31) How to upload... AWS Home - Skill Builder Amazon EFS - file sy...

aws Services Search [Alt+S]

VPC IAM

DB cluster identifier **Info**
Enter a name for your DB cluster. The name must be unique across all DB clusters owned by your AWS account in the current AWS Region.
RDS-1

The DB cluster identifier is case-insensitive, but is stored as all lowercase (as in "mydbcluster"). 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 cluster.
admin

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 **Strong**

MySQL

MySQL is the most popular open source database in the world. MySQL on RDS offers the rich features of the MySQL community edition with the flexibility to easily scale compute resources or storage capacity for your database.

- Supports database size up to 64 TiB.
- Supports General Purpose, Memory Optimized, and Burstable Performance instance classes.
- Supports automated backup and point-in-time recovery.
- Supports up to 15 Read Replicas per instance, within a single Region or 5 read replicas cross-region.

CloudShell Feedback 27°C Partly sunny Search Privacy Terms Cookie preferences © 2024, Amazon Web Services, Inc. or its affiliates. ENG IN 18:16 16-09-2024

Auto Scaling groups | EC2 Instances | EC2 us-west-2 Create database - RDS Ma RouteTables | VPC Console Apache2 Ubuntu Default How To Install MySQL on ...

Gmail Maps YouTube (31) How to upload... AWS Home - Skill Builder Amazon EFS - file sy...

aws Services Search [Alt+S]

VPC IAM

Storage

Storage type **Info**
Provisioned IOPS SSD (io2) storage volumes are now available.
General Purpose SSD (gp3)
Performance scales independently from storage

Allocated storage **Info**
200 GiB
Minimum: 20 GiB. Maximum: 65,536 GiB

After you modify the storage for a DB instance, the status of the DB instance will be in storage-optimization. Your instance will remain available as the storage-optimization operation completes. [Learn more](#)

Advanced settings
Baseline IOPS of 5,000 IOPS and storage throughput of 125 MiBps are included for allocated storage less than 400 GiB.

Storage autoscaling

Connectivity **Info**

MySQL

MySQL is the most popular open source database in the world. MySQL on RDS offers the rich features of the MySQL community edition with the flexibility to easily scale compute resources or storage capacity for your database.

- Supports database size up to 64 TiB.
- Supports General Purpose, Memory Optimized, and Burstable Performance instance classes.
- Supports automated backup and point-in-time recovery.
- Supports up to 15 Read Replicas per instance, within a single Region or 5 read replicas cross-region.

CloudShell Feedback 27°C Partly sunny Search Privacy Terms Cookie preferences © 2024, Amazon Web Services, Inc. or its affiliates. ENG IN 18:18 16-09-2024

Compute resource
Choose whether to set up a connection to a compute resource for this database. Setting up a connection will automatically change connectivity settings so that the compute resource can connect to this database.

Don't connect to an EC2 compute resource
Don't set up a connection to a compute resource for this database. You can manually set up a connection to a compute resource later.

Connect to an EC2 compute resource
Set up a connection to an EC2 compute resource for this database.

Virtual private cloud (VPC) [Info](#)
Choose the VPC. The VPC defines the virtual networking environment for this DB cluster.

Project-VPC (vpc-0eac52b630cb26d1)
7 Subnets, 3 Availability Zones

Only VPCs with a corresponding DB subnet group are listed.

DB subnet group [Info](#)
Choose the DB subnet group. The DB subnet group defines which subnets and IP ranges the DB cluster can use in the VPC that you selected.

subnet-gp
3 Subnets, 3 Availability Zones

MySQL

MySQL is the most popular open source database in the world. MySQL on RDS offers the rich features of the MySQL community edition with the flexibility to easily scale compute resources or storage capacity for your database.

- Supports database size up to 64 TiB.
- Supports General Purpose, Memory Optimized, and Burstable Performance instance classes.
- Supports automated backup and point-in-time recovery.
- Supports up to 15 Read Replicas per instance, within a single Region or 5 read replicas cross-region.

DB subnet group [Info](#)
Choose the DB subnet group. The DB subnet group defines which subnets and IP ranges the DB cluster can use in the VPC that you selected.

subnet-gp
3 Subnets, 3 Availability Zones

Public access [Info](#)
 Yes
RDS assigns a public IP address to the cluster. Amazon EC2 instances and other resources outside of the VPC can connect to your cluster. Resources inside the VPC can also connect to the cluster. Choose one or more VPC security groups that specify which resources can connect to the cluster.

No
RDS doesn't assign a public IP address to the cluster. Only Amazon EC2 instances and other resources inside the VPC can connect to your cluster. Choose one or more VPC security groups that specify which resources can connect to the cluster.

VPC security group (firewall) [Info](#)
Choose one or more VPC security groups to allow access to your database. Make sure that the security group rules allow the appropriate incoming traffic.

Choose existing
Choose existing VPC security groups

Create new
Create new VPC security group

Existing VPC security groups
Choose one or more options

DB-SG X

RDS Proxy

MySQL

MySQL is the most popular open source database in the world. MySQL on RDS offers the rich features of the MySQL community edition with the flexibility to easily scale compute resources or storage capacity for your database.

- Supports database size up to 64 TiB.
- Supports General Purpose, Memory Optimized, and Burstable Performance instance classes.
- Supports automated backup and point-in-time recovery.
- Supports up to 15 Read Replicas per instance, within a single Region or 5 read replicas cross-region.

The screenshot shows the 'Create database' wizard for MySQL. In the 'Additional configuration' step, it lists options like encryption turned on, backup turned on, backtrack turned off, maintenance, CloudWatch Logs, and delete protection turned on. Below this is the 'Estimated Monthly costs' section, which shows:

DB instance	762.12 USD
Storage	69.00 USD
Total	831.12 USD

A note states: "This billing estimate is based on on-demand usage as described in [Amazon RDS Pricing](#). Estimate does not include costs for backup storage, I/Os (if applicable), or data transfer." A link to "Estimate your monthly costs for the DB Instance using the [AWS Simple Monthly Calculator](#)" is provided.

In the bottom right corner of the main window, there is a modal titled "MySQL" with the following text and bullet points:

MySQL is the most popular open source database in the world. MySQL on RDS offers the rich features of the MySQL community edition with the flexibility to easily scale compute resources or storage capacity for your database.

- Supports database size up to 64 TiB.
- Supports General Purpose, Memory Optimized, and Burstable Performance instance classes.
- Supports automated backup and point-in-time recovery.
- Supports up to 15 Read Replicas per instance, within a single Region or 5 read replicas cross-region.

At the bottom of the main window, there are "Cancel" and "Create database" buttons. The status bar at the bottom indicates "CloudShell Feedback", "27°C Partly sunny", and system icons.

The screenshot shows the "Databases" page in the AWS RDS console. On the left, the navigation menu includes "Dashboard", "Databases", "Query Editor", "Performance insights", "Snapshots", "Exports in Amazon S3", "Automated backups", "Reserved instances", "Proxies", "Subnet groups", "Parameter groups", "Option groups", "Custom engine versions", and "Zero-ETL integrations".

The main content area displays a green success message: "Successfully created database rds-1. You can use settings from rds-1 to simplify configuration of suggested database add-ons while we finish creating your DB for you." Below this, the "View connection details" button is visible.

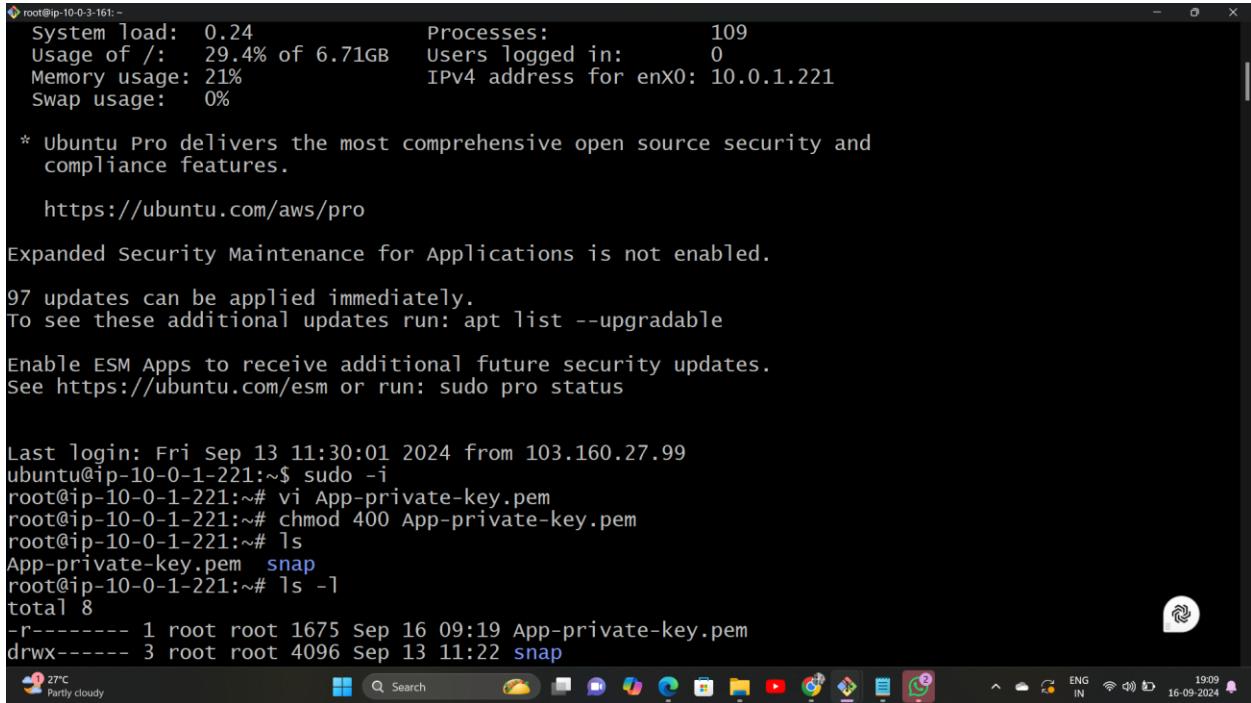
The "Databases (4)" table lists the following instances:

DB Identifier	Status	Role	Engine	Region & AZ	Size
rds-1	Available	Multi-AZ DB cluster	MySQL Community	us-west-2	3 instances
rds-1-instance-1	Available	Writer instance	MySQL Community	us-west-2b	db.m6idn.8xlarge
rds-1-instance-2	Available	Reader instance	MySQL Community	us-west-2a	db.m6idn.8xlarge
rds-1-instance-3	Available	Reader instance	MySQL Community	us-west-2c	db.m6idn.8xlarge

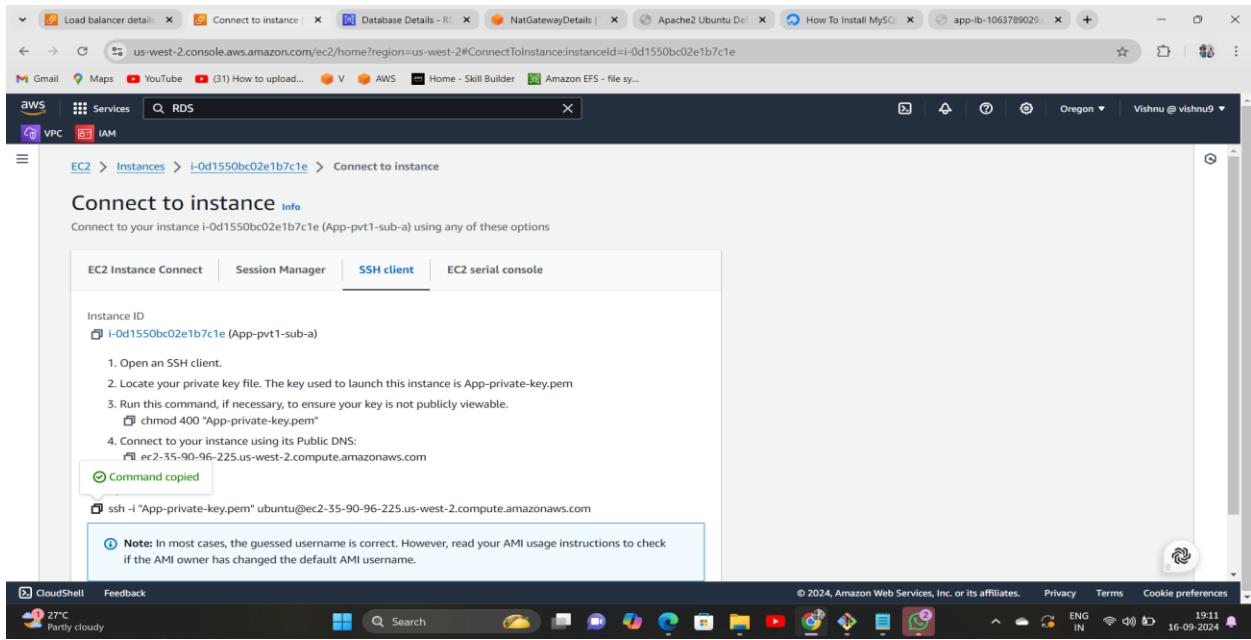
At the bottom of the main window, there is a "CloudShell Feedback" button, a status bar indicating "CloudShell Feedback", "27°C Partly sunny", and system icons.

Step: 13

Go back to EC2 instance Dash board and Connect private-AZ-a instance through public-AZ-a by using SSH Client link and Install MySQL



```
root@ip-10-0-3-161:~  
System load: 0.24 Processes: 109  
Usage of /: 29.4% of 6.71GB Users logged in: 0  
Memory usage: 21% IPv4 address for enx0: 10.0.1.221  
Swap usage: 0%  
  
* Ubuntu Pro delivers the most comprehensive open source security and  
compliance features.  
  
https://ubuntu.com/aws/pro  
  
Expanded security Maintenance for Applications is not enabled.  
  
97 updates can be applied immediately.  
To see these additional updates run: apt list --upgradable  
  
Enable ESM Apps to receive additional future security updates.  
See https://ubuntu.com/esm or run: sudo pro status  
  
Last login: Fri Sep 13 11:30:01 2024 from 103.160.27.99  
ubuntu@ip-10-0-1-221:~$ sudo -i  
root@ip-10-0-1-221:~# vi App-private-key.pem  
root@ip-10-0-1-221:~# chmod 400 App-private-key.pem  
root@ip-10-0-1-221:~# ls  
App-private-key.pem snap  
root@ip-10-0-1-221:~# ls -l  
total 8  
-r----- 1 root root 1675 Sep 16 09:19 App-private-key.pem  
drwx---- 3 root root 4096 Sep 13 11:22 snap
```



Load balancer details | Connect to instance | Database Details - RDS | NatGatewayDetails | Apache2 Ubuntu De... | How To Install MySQL | app-lb-1063789029...

Gmail Maps YouTube (31) How to upload... V AWS Home - Skill Builder Amazon EFS - file sy...

aws Services RDS

EC2 Instances i-0d1550bc02e1b7c1e Connect to instance

Connect to instance Info

Connect to your instance i-0d1550bc02e1b7c1e (App-pvt1-sub-a) using any of these options

EC2 Instance Connect Session Manager **SSH client** EC2 serial console

Instance ID
i-0d1550bc02e1b7c1e (App-pvt1-sub-a)

1. Open an SSH client.
2. Locate your private key file. The key used to launch this instance is App-private-key.pem
3. Run this command, if necessary, to ensure your key is not publicly viewable.
 `chmod 400 "App-private-key.pem"
4. Connect to your instance using its Public DNS:
 `ssh -i "App-private-key.pem" ubuntu@ec2-35-90-96-225.us-west-2.compute.amazonaws.com`

Command copied

ssh -i "App-private-key.pem" ubuntu@ec2-35-90-96-225.us-west-2.compute.amazonaws.com

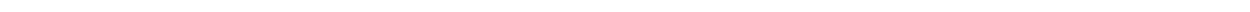
Note: In most cases, the guessed username is correct. However, read your AMI usage instructions to check if the AMI owner has changed the default AMI username.

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

27°C Partly cloudy 19:11 16-09-2024

```
root@ip-10-0-3-161:~  
-r----- 1 root root 1675 Sep 16 09:19 App-private-key.pem  
drwx---- 3 root root 4096 Sep 13 11:22 snap  
root@ip-10-0-1-221:~# ssh -i "App-private-key.pem" ubuntu@ec2-10-0-3-161.us-west-2.compute.amazonaws.com  
The authenticity of host 'ec2-10-0-3-161.us-west-2.compute.amazonaws.com (10.0.3.161)' can't be established.  
ED25519 key fingerprint is SHA256:h4bm1lh6sqfydAyKjraVlBADevDYM1ep/UNu3euVaY.  
This key is not known by any other names.  
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes  
Warning: Permanently added 'ec2-10-0-3-161.us-west-2.compute.amazonaws.com' (ED25519) to the list of known hosts.  
Welcome to Ubuntu 24.04 LTS (GNU/Linux 6.8.0-1015-aws x86_64)  
  
* Documentation: https://help.ubuntu.com  
* Management: https://landscape.canonical.com  
* Support: https://ubuntu.com/pro  
  
System information as of Mon Sep 16 09:22:20 UTC 2024  
  
System load: 0.09 Processes: 107  
Usage of /: 29.3% of 6.71GB Users logged in: 0  
Memory usage: 20% IPv4 address for enx0: 10.0.3.161  
Swap usage: 0%  
  
* Ubuntu Pro delivers the most comprehensive open source security and compliance features.  
  
https://ubuntu.com/aws/pro  

```

```
root@ip-10-0-3-161:~  
ubuntu@ip-10-0-3-161:~$ sudo -i  
root@ip-10-0-3-161:~# sudo apt update  
Hit:1 http://us-west-2.ec2.archive.ubuntu.com/ubuntu noble InRelease  
Get:2 http://us-west-2.ec2.archive.ubuntu.com/ubuntu noble-updates InRelease [126 kB]  
Get:3 http://us-west-2.ec2.archive.ubuntu.com/ubuntu noble-backports InRelease [126 kB]  
Hit:4 http://security.ubuntu.com/ubuntu noble-security InRelease  
Get:5 http://us-west-2.ec2.archive.ubuntu.com/ubuntu noble-updates/main amd64 Packages [502 kB]  
Get:6 http://us-west-2.ec2.archive.ubuntu.com/ubuntu noble-updates/universe amd64 4 Packages [365 kB]  
Get:7 http://us-west-2.ec2.archive.ubuntu.com/ubuntu noble-backports/universe amd64 Packages [10.6 kB]  
Get:8 http://us-west-2.ec2.archive.ubuntu.com/ubuntu noble-backports/universe Translation-en [10.8 kB]  
Get:9 http://us-west-2.ec2.archive.ubuntu.com/ubuntu noble-backports/universe amd64 c-n-f Metadata [1104 B]  
Fetched 1142 kB in 1s (1394 kB/s)  
Reading package lists... Done  
Building dependency tree... Done  
Reading state information... Done  
101 packages can be upgraded. Run 'apt list --upgradable' to see them.  
root@ip-10-0-3-161:~# sudo apt install mysql-server  
Reading package lists... Done  
Building dependency tree... Done  
Reading state information... Done  
The following additional packages will be installed:  

```

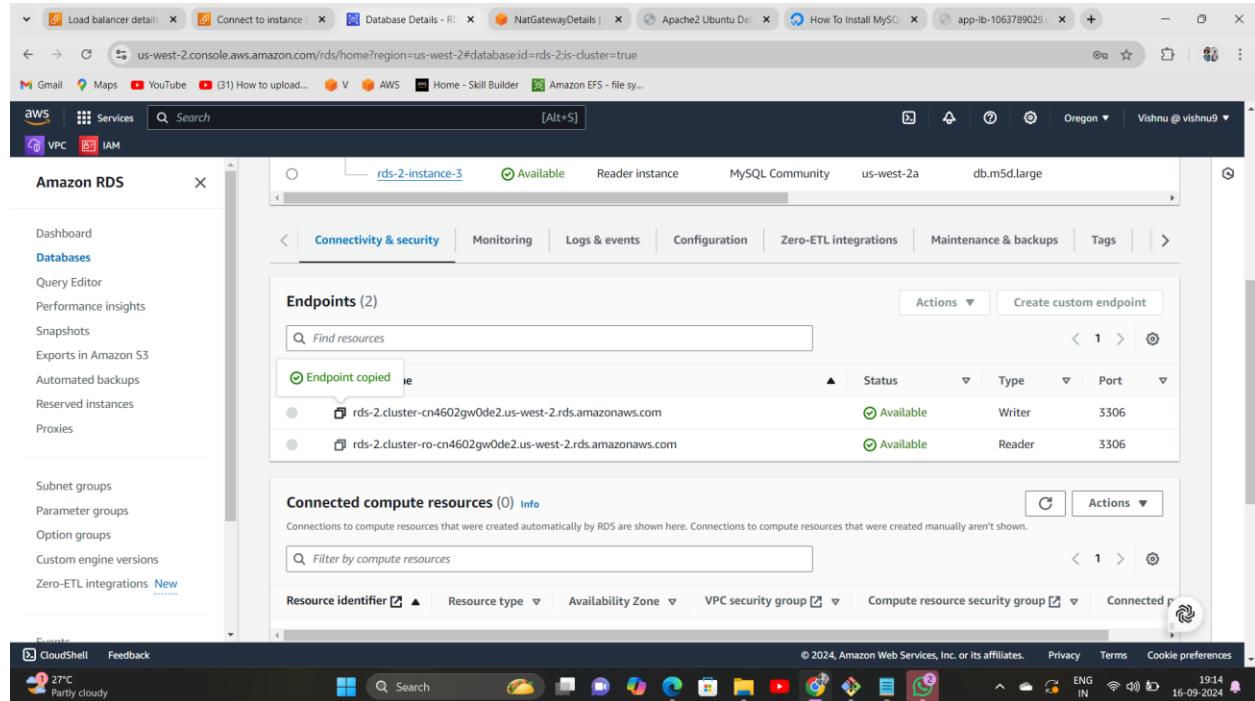
Step: 14

Goto Our database and Copy the Writer Endpoint and paste it in

private AZ-a by using these command

-->mysql -h [link] -u admin -p

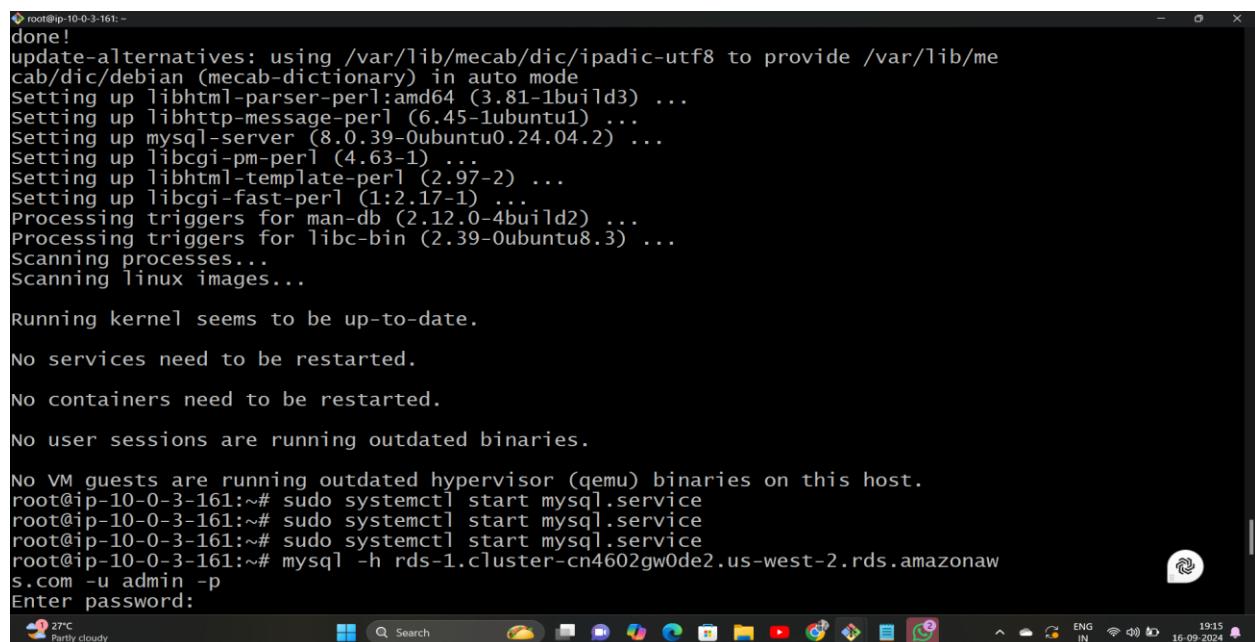
--> enter password *****



The screenshot shows the AWS RDS console for an RDS instance named 'rds-2-instance-3'. The 'Connectivity & security' tab is selected. Under 'Endpoints (2)', there are two entries:

Endpoint	Status	Type	Port
rds-2.cluster-cn4602gw0de2.us-west-2.rds.amazonaws.com	Available	Writer	3306
rds-2.cluster-ro-cn4602gw0de2.us-west-2.rds.amazonaws.com	Available	Reader	3306

Below the endpoints, the 'Connected compute resources (0)' section is shown, indicating no manual connections have been created.



```
root@ip-10-0-3-161:~# update-alternatives: using /var/lib/mecab/dic/ipadic-utf8 to provide /var/lib/mecab/dic.debian (mecab-dictionary) in auto mode
Setting up libhtml-parser-perl:amd64 (3.81-1build3) ...
Setting up libhttp-message-perl (6.45-1ubuntu1) ...
Setting up mysql-server (8.0.39-0ubuntu0.24.04.2) ...
Setting up libcgi-pm-perl (4.63-1) ...
Setting up libhtml-template-perl (2.97-2) ...
Setting up libcgi-fast-perl (1:2.17-1) ...
Processing triggers for man-db (2.12.0-4build2) ...
Processing triggers for libc-bin (2.39-0ubuntu8.3) ...
Scanning processes...
Scanning linux images...

Running kernel seems to be up-to-date.

No services need to be restarted.

No containers need to be restarted.

No user sessions are running outdated binaries.

No VM guests are running outdated hypervisor (qemu) binaries on this host.
root@ip-10-0-3-161:~# sudo systemctl start mysql.service
root@ip-10-0-3-161:~# sudo systemctl start mysql.service
root@ip-10-0-3-161:~# sudo systemctl start mysql.service
root@ip-10-0-3-161:~# mysql -h rds-1.cluster-cn4602gw0de2.us-west-2.rds.amazonaws.com -u admin -p
Enter password:
```

Step:15

After connection to MySQL

Now Create a Database

```
root@ip-10-0-3-161:~  
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)  
  
mysql> create database vishnu;  
Query OK, 1 row affected (0.03 sec)  
  
mysql> show databases;  
+-----+  
| Database |  
+-----+  
| information_schema |  
| mysql |  
| performance_schema |  
| sys |  
| vishnu |  
+-----+  
5 rows in set (0.01 sec)  
  
mysql>
```

THANK YOU