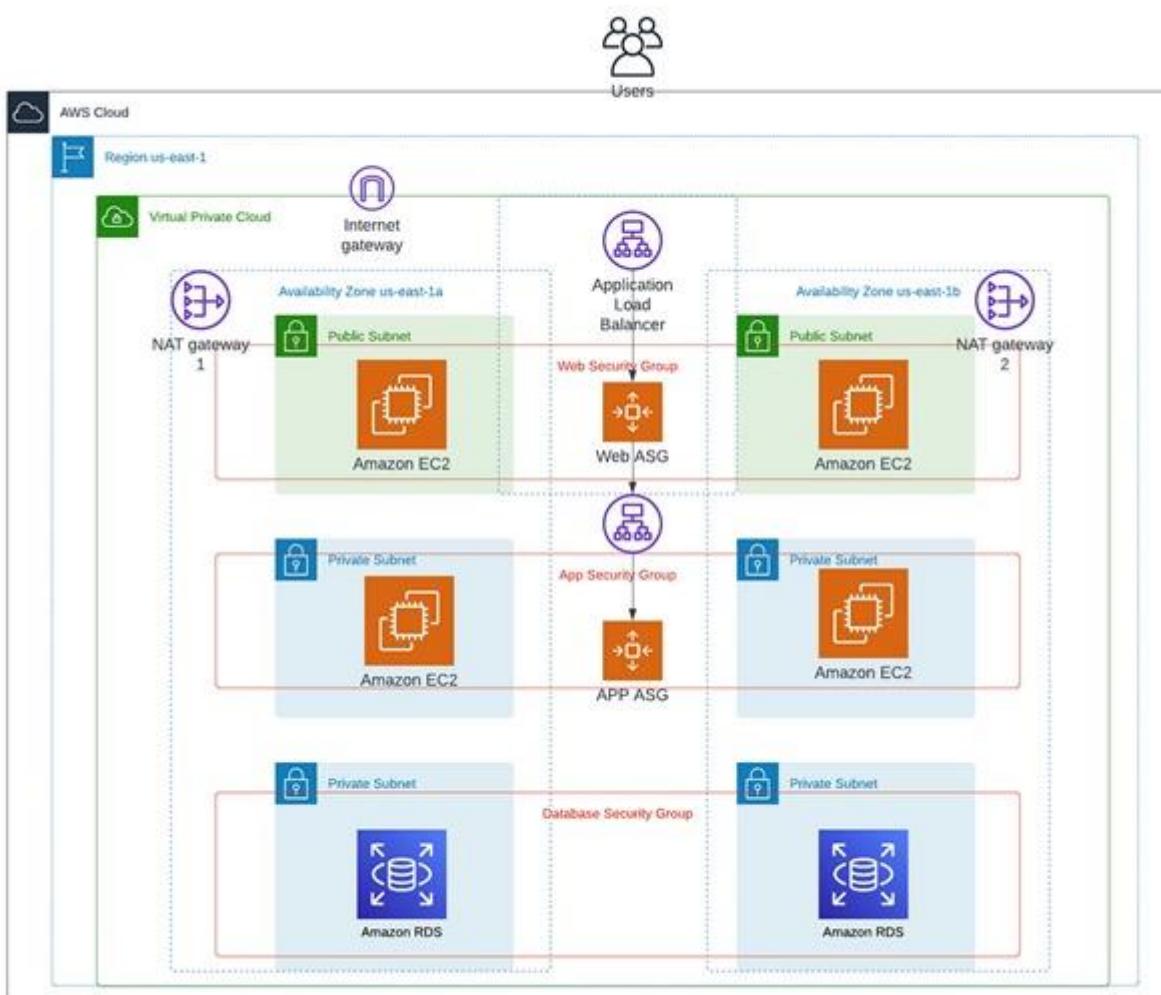


PROJECT-1

Name : Rohini Velpula

Email id : rohinivelpula10550@gmail.com



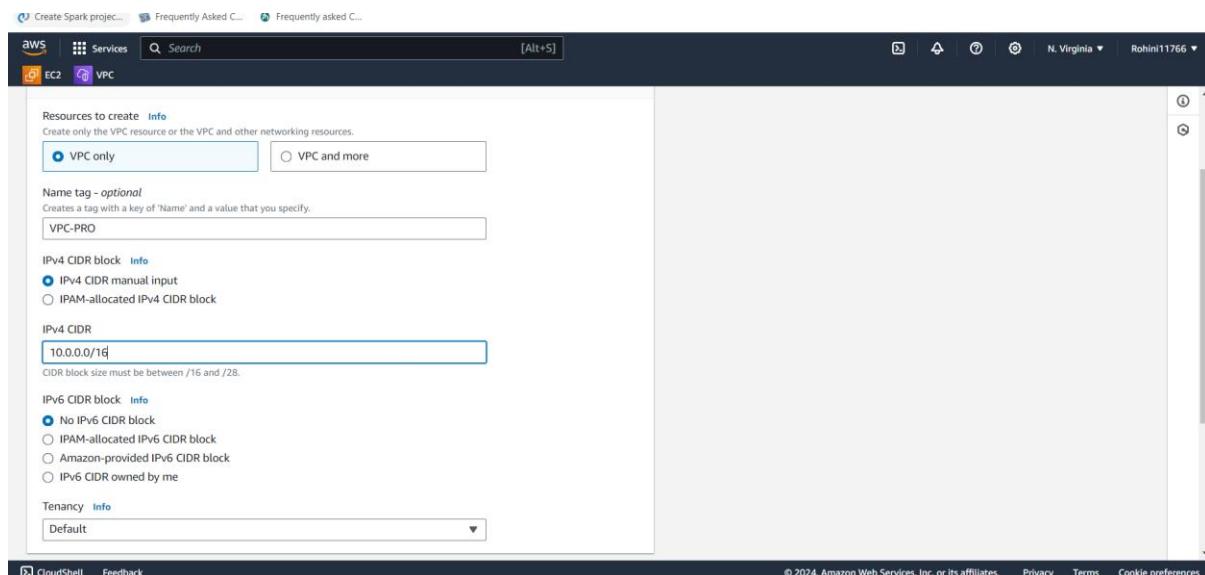
The above architecture is the “Architecting 3 Tier Architecture on AWS”.

- First tier of our architecture is a web tier. It consists of 2 public subnets in separate availability zones, and an Auto scaling group with launch template and security group.
- The second tier is an Application tier. This tier will consist of 2 Private subnets, an Autoscaling group with launch a template and same security group used in above web tier.
- The third tier is a Database tier. This tier will have an RDS(relational database service) in 2 Private subnets and an same security group used in above both the tiers.

Creating the above architecture we have to follow the following steps:

1. Create VPC, Subnets – 6, Internet gate way – 1, Route tables – 2, Nat gate way – 1.
2. Launch an EC2 instance.
3. Create an AMI (image).
4. Create Autoscaling group, Create launch template.
5. Create Subnet group.
6. Create Database (RDS).
7. Establish connection.

Step 1: Create VPC and its components:



- Go to VPC dashboard click on create VPC.
- Click on VPC only and name tag as my-project-vpc.
- Give IPV4 CIDR (classless inter domain routing) as 10.0.0.0/16.

- Click on VPC, it is created.

The screenshot shows the 'Create VPC' wizard. In the 'IPv4 CIDR' section, '10.0.0.0/16' is selected. Under 'IPv6 CIDR block', 'No IPv6 CIDR block' is chosen. The 'Tenancy' dropdown is set to 'Default'. In the 'Tags' section, a tag 'Name: VPC-PRO' is added. At the bottom right, the 'Create VPC' button is highlighted.

The screenshot shows the 'Your VPCs' list. It displays two VPCs: one with VPC ID 'vpc-0fe7be1235d6f4db2' and another with VPC ID 'vpc-0907ee23bedb098f2'. Both are listed as 'Available' with their respective IPv4 CIDRs: '172.31.0.0/16' and '10.0.0.0/16'. The 'Actions' dropdown menu for the second VPC is open, showing options like 'Edit', 'Delete', and 'Associate gateway'.

- Create 6 subnets (2-public, 4-private).
- Create first subnet.
- Click on subnet, click on create subnet, select our VPC (VPC-PRO).
- Give name tag as public-A, select availability zone as us-east-1a. Give CIDR as 10.0.13.0/24 and created it.

The screenshot shows the 'Subnets' list. A success message at the top states 'You have successfully created 1 subnet: subnet-03f0354af51d7a87c'. The list table includes columns for Name, Subnet ID, State, VPC, IPv4 CIDR, and IP range. One subnet is listed: 'public-A' with Subnet ID 'subnet-03f0354af51d7a87c', State 'Available', VPC 'vpc-0907ee23bedb098f2 | VPC...', and IPv4 CIDR '10.0.13.0/24'.

- Create Second subnet.
- Click on subnet, click on create subnet, select our VPC (VPC-PRO).
- Give name tag as public-B, select availability zone as us-east-1b. Give CIDR as 10.0.45.0/24 and created it.

The screenshot shows the AWS Subnets page with a green success message at the top: "You have successfully created 1 subnet: subnet-03e588a0a629c4cf1". Below the message, there is a table titled "Subnets (1) Info" with one item listed:

Name	Subnet ID	State	VPC	IPv4 CIDR
public-B	subnet-03e588a0a629c4cf1	Available	vpc-0907ee23bedb098f2 VPC...	10.0.45.0/24

Create Third subnet.

- Click on subnet, click on create subnet, select our VPC (VPC-PRO).
- Give name tag as private-1A, select availability zone as us-east-1a. Give CIDR as 10.0.34.0/24 and created it.

The screenshot shows the AWS Subnets page with a green success message at the top: "You have successfully created 1 subnet: subnet-0f9f37240ee41bb43". Below the message, there is a table titled "Subnets (1) Info" with one item listed:

Name	Subnet ID	State	VPC	IPv4 CIDR
Private-1A	subnet-0f9f37240ee41bb43	Available	vpc-0907ee23bedb098f2 VPC...	10.0.34.0/24

Create fourth subnet.

- Click on subnet, click on create subnet, select our VPC (VPC-PRO).
- Give name tag as public-2B, select availability zone as us-east-1b. Give CIDR as 10.0.12.0/24 and created it.

Subnets (1) Info								
<input type="text" value="Find resources by attribute or tag"/> <input type="button" value="Clear filters"/> Last updated less than a minute ago <input type="button" value="Actions"/> <input type="button" value="Create subnet"/>								
	Name	Subnet ID	State	VPC	IPv4 CIDR			
<input type="checkbox"/>	private-2B	subnet-0bf7ec612cf810ff8	Available	vpc-0907ee23bedb098f2 VPC...	10.0.12.0/24			

Create FIFTH subnet.

- Click on subnet, click on create subnet, select our VPC (VPC-PRO).
- Give name tag as public-3A, select availability zone as us-east-1b. Give CIDR as 10.0.45.0/24 and created it.

Subnets (1) Info								
<input type="text" value="Find resources by attribute or tag"/> <input type="button" value="Clear filters"/> Last updated less than a minute ago <input type="button" value="Actions"/> <input type="button" value="Create subnet"/>								
	Name	Subnet ID	State	VPC	IPv4 CIDR			
<input type="checkbox"/>	private-3A	subnet-009157f289c8c9b2c	Available	vpc-0907ee23bedb098f2 VPC...	10.0.76.0/24			

Create SIXTH subnet.

- Click on subnet, click on create subnet, select our VPC (VPC-PRO).
- Give name tag as public-4B, select availability zone as us-east-1b. Give CIDR as 10.0.83.0/24 and created it.

Subnets (1) Info								
<input type="text" value="Find resources by attribute or tag"/> <input type="button" value="Clear filters"/> Last updated less than a minute ago <input type="button" value="Actions"/> <input type="button" value="Create subnet"/>								
	Name	Subnet ID	State	VPC	IPv4 CIDR			
<input type="checkbox"/>	private-4B	subnet-0511ebc30119a6ad1	Available	vpc-0907ee23bedb098f2 VPC...	10.0.83.0/24			

These are the subnets we created.

The screenshot shows the 'Edit subnet associations' page for a specific route table. In the 'Available subnets (2/6)' section, two subnets are selected: 'publiv-A' and 'public-B'. These are listed with their respective Subnet IDs, IPv4 CIDRs, and Route table IDs. The 'Selected subnets' section at the bottom shows the two chosen subnets. A 'Save associations' button is visible at the bottom right.

Name	Subnet ID	IPv4 CIDR	IPv6 CIDR	Route table ID
publiv-A	subnet-03f0354af51d7a87c	10.0.13.0/24	-	Main (rtb-0aaa65ef018a433d0)
public-B	subnet-03e588a0a629c4cf1	10.0.45.0/24	-	Main (rtb-0aaa65ef018a433d0)

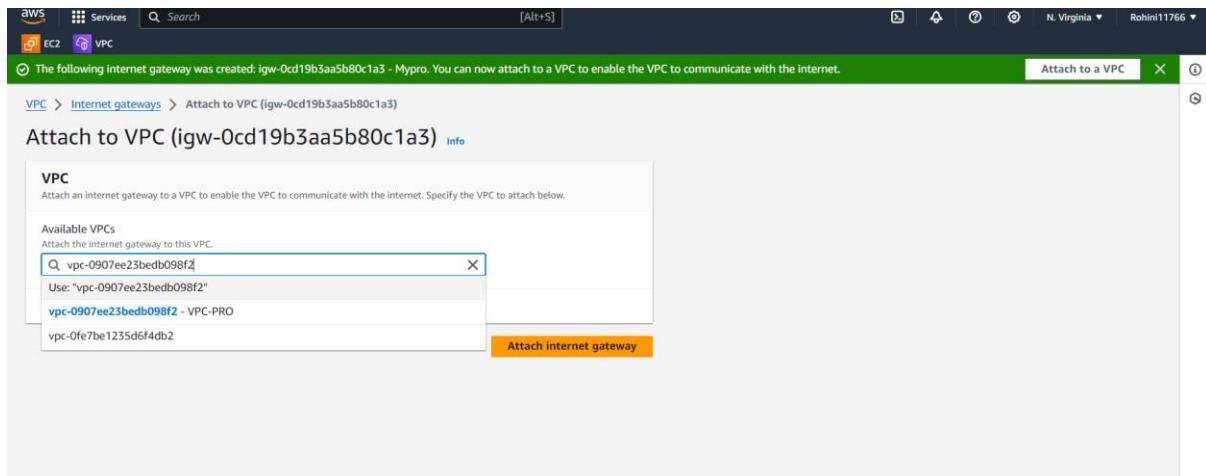
Create internet gateway, name tag as Mypro

The screenshot shows the 'Internet gateways' page. A success message indicates a new internet gateway was created: 'igw-0cd19b3aa5b80c1a3 - Mypro'. Below this, the gateway details are shown: Internet gateway ID (igw-0cd19b3aa5b80c1a3), State (Detached), VPC ID (-), and Owner (905418355949). The 'Tags' section shows a single tag named 'Name' with the value 'Mypro'. An 'Actions' dropdown menu is visible on the right.

This igw is attached to VPC.

- Go to actions in internet gate way and click on attach to VPC
- Select our VPC (VPC-PRO). Click on attach internet gateway.

Click on attach internet gateway

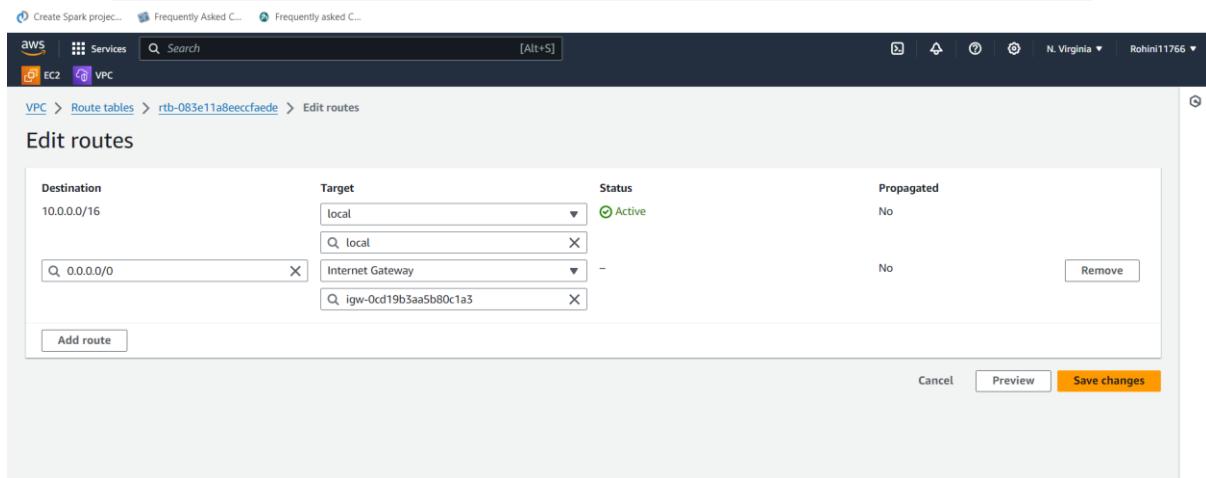


Create route table, give name as public

- Select our VPC (VPC-PRO), create it. Click on route table id, open it.
- Go down click on edit subnet association.
- Select both public subnet and click on save association.

Go to actions click on edit routes.

- Click on add routes give all traffic (0.0.0.0/0) and select our internet gateway, save changes.



Create private route table name as Private.

- Select our VPC, (VPC-PRO), create it.
- Click on route table id, open it.
- Go down click on edit subnet association.
- Select all private subnet and click on save association.

The screenshot shows the AWS Route Tables page. At the top, a green success message reads: "Route table rtb-00c6224006dc829df | Private was created successfully." Below the message, the breadcrumb navigation shows: VPC > Route tables > rtb-00c6224006dc829df. The main title is "rtb-00c6224006dc829df / Private". On the right, there is an "Actions" dropdown menu. The "Details" tab is selected, showing the following information:

Route table ID	Main	Explicit subnet associations	Edge associations
rtb-00c6224006dc829df	No	-	-
VPC	Owner ID		
vpc-0907ee23bedb098f2 VPC-PRO	905418355949		

The "Routes" tab is selected, showing one route entry:

Routes (1)		Both	Edit routes
Filter routes		< 1 >	⟳
Destination	Target	Status	Propagated
10.0.0.0/16	local	Active	No

- Create NAT gateway, give name as proNAT.
- Select public subnet(private-1A).
- Select connectivity type as IPV4.
- Click on allocate Elastic IP

aws Services Search [Alt+S] N. Virginia Rohini11766

VPC EC2 VPC

VPC > NAT gateways > Create NAT gateway

Create NAT gateway Info

A highly available, managed Network Address Translation (NAT) service that instances in private subnets can use to connect to services in other VPCs, on-premises networks, or the internet.

NAT gateway settings

Name - optional
Create a tag with a key of 'Name' and a value that you specify.
proNAT
The name can be up to 256 characters long.

Subnet
Select a subnet in which to create the NAT gateway.
subnet-0f9f37240ee41bb43 (Private-1A)

Connectivity type
Select a connectivity type for the NAT gateway.
 Public
 Private

Elastic IP allocation ID Info
Assign an Elastic IP address to the NAT gateway.
Select an Elastic IP

NAT gateway nat-0f36ee3efed0e26bd | proNAT was created successfully.

VPC > NAT gateways > nat-0f36ee3efed0e26bd

nat-0f36ee3efed0e26bd / proNAT

Actions ▾

Details			
NAT gateway ID nat-0f36ee3efed0e26bd	Connectivity type Public	State Pending	State message <small>Info</small> —
NAT gateway ARN arn:aws:ec2:us-east-1:905418355949:natgateway/nat-0f36ee3efed0e26bd	Primary public IPv4 address —	Primary private IPv4 address 10.0.34.88	Primary network interface ID eni-0417a6f62a2a41232
VPC vpc-0907ee23bedb098f2 / VPC-PRO	Subnet subnet-0f9f37240ee41bb43 / Private-1A	Created Tuesday, August 6, 2024 at 09:25:38 GMT+5:30	Deleted —

Secondary IPv4 addresses Monitoring Tags

Secondary IPv4 addresses

Search

Private IPv4 address Network interface ID Status Failure message

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

Now go to private route and click on actions.

- Click on edit routes and add route.
- Give all traffic (0.0.0.0/0) and select NAT gateway.

The screenshot shows the AWS VPC Route Tables interface. The top navigation bar includes 'aws', 'Services' (with EC2 and VPC selected), 'Search', and 'N. Virginia'. The main area shows the 'Edit routes' screen for route table 'rtb-00c6224006dc829df'. The table lists two routes:

Destination	Target	Status	Propagated
10.0.0.0/16	local	Active	No
0.0.0.0/0	NAT Gateway nat-0f36ee3efed0e26bd Use: "nat-0f36ee3efed0e26bd"	-	No

Buttons at the bottom include 'Add route', 'Cancel', 'Preview', and 'Save changes'.

The screenshot shows the 'Edit subnet associations' screen for route table 'rtb-00c6224006dc829df'. The top navigation bar includes 'aws', 'Services' (with EC2 and VPC selected), 'Search', and 'N. Virginia'. The main area shows the 'Edit subnet associations' screen for route table 'rtb-00c6224006dc829df'. The table lists available subnets:

Name	Subnet ID	IPv4 CIDR	IPv6 CIDR	Route table ID
public-A	subnet-03f0354af51d7a87c	10.0.13.0/24	-	rtb-083e11a8eecfaede / public
public-B	subnet-03e588a0a629c4cf1	10.0.45.0/24	-	rtb-083e11a8eecfaede / public
<input checked="" type="checkbox"/> Private-1A	subnet-0f9f37240ee41bb43	10.0.34.0/24	-	Main (rtb-0aaa65ef018a433d0)
<input checked="" type="checkbox"/> private-3A	subnet-009157f289c8c9b2c	10.0.76.0/24	-	Main (rtb-0aaa65ef018a433d0)
<input checked="" type="checkbox"/> private-2B	subnet-0bf7ec612cf810ff8	10.0.12.0/24	-	Main (rtb-0aaa65ef018a433d0)
<input checked="" type="checkbox"/> private-4B	subnet-0511ebc30119a6ad1	10.0.83.0/24	-	Main (rtb-0aaa65ef018a433d0)

The 'Selected subnets' section contains four subnets: 'subnet-0f9f37240ee41bb43 / Private-1A', 'subnet-009157f289c8c9b2c / private-3A', 'subnet-0bf7ec612cf810ff8 / private-2B', and 'subnet-0511ebc30119a6ad1 / private-4B'. Buttons at the bottom include 'Cancel' and 'Save associations'.

Launch an EC2 instance.

- Go to EC2 dashboard click on launch instance.
- Name as project-ec2 and select ami as ubuntu.
- Instance type as t2.micro and key pair as project.
- Click on edit network settings, select our VPC and public subnet.
- Auto assign IP enable and create a security group as project-sg.

Launching the public instance names as public-A

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

Recents | Quick Start

Amazon Linux | macOS | Ubuntu | Windows | Red Hat | SUSE Linux |

Amazon Machine Image (AMI)

Ubuntu Server 24.04 LTS (HVM), SSD Volume Type | Free tier eligible

ami-04a81a99f5ec58529 (64-bit (x86)) / ami-0c14ff530901e49ff (64-bit (Arm))
Virtualization: hvm ENA enabled: true Root device type: ebs

Description
Ubuntu Server 24.04 LTS (HVM), EBS General Purpose (SSD) Volume Type. Support available from Canonical (<http://www.ubuntu.com/cloud/services>).

Architecture: 64 bit (x86) | AMI ID: ami-04a81a99f5ec58529 |

Summary

Number of instances: 1

Software Image (AMI): Canonical, Ubuntu, 24.04 LTS, ...read more
ami-04a81a99f5ec58529

Virtual server type (instance type): t2.micro

Firewall (security group): New security group

Storage (volumes): 1 volume(s) - 8 GiB

Cancel

us-east-1.console.aws.amazon.com/ec2/home?region=us-east-1#Instances?v=3;\$case=true%5C.client:false;\$regex=false%5C.client:false

Frequently Asked Questions | Frequently asked C...

Instances | [Alt+S]

Instances (1) Info

Name	Instance ID	Instance state	Instance type	Status check	Alarm status	Availability Zone	Public IP
publicA	i-0fb03f43f8f066c0a	Running	t2.micro	Initializing	<input type="button" value="View alarms"/>	us-east-1a	ec2-18-214-111-111.us-east-1.compute.amazonaws.com

Launch an EC2 instance.

- Go to EC2 dashboard click on launch instance.
- Name as project-ec2 and select ami as ubuntu.
- Instance type as t2.micro and key pair as project.
- Click on edit network settings, select our VPC and public subnet.
- Auto assign IP enable and create a security group as project-sg.

Launching the public instance names as public-B

Connected to the public-A instance and created the file

```

aws
Services Search [Alt+S]
EC2 VPC

Processing triggers for man-db (2.12.0-4build2) ...
Processing triggers for libc-bin (2.39-0ubuntu8.2) ...
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-10-0-45-40:~# cd /var/www/html
root@ip-10-0-45-40:/var/www/html# ls
index.html
root@ip-10-0-45-40:/var/www/html# rm index.html
root@ip-10-0-45-40:/var/www/html# vi index.html
root@ip-10-0-45-40:/var/www/html# systemctl restart apache2
root@ip-10-0-45-40:/var/www/html# ls
index.html
root@ip-10-0-45-40:/var/www/html# curl
curl: try 'curl --help' or 'curl --manual' for more information
root@ip-10-0-45-40:/var/www/html# curl 10.0.45.40:80
this is from public B
root@ip-10-0-45-40:/var/www/html#

```

```

root@ip-10-0-13-46:~# apt install apache2 -y
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
apache2 is already the newest version (2.4.58-1ubuntu8.4).
0 upgraded, 0 newly installed, 0 to remove and 42 not upgraded.
root@ip-10-0-13-46:~# ls
snap
root@ip-10-0-13-46:~# cd /var/www/html
root@ip-10-0-13-46:/var/www/html# ls
index.html index.nginx-debian.html
root@ip-10-0-13-46:/var/www/html# rm index.html
root@ip-10-0-13-46:/var/www/html# ls
root@ip-10-0-13-46:/var/www/html# vi index.html
root@ip-10-0-13-46:/var/www/html# curl 10.0.13.46
this is fm public-AAAAAAAAAAAAA
root@ip-10-0-13-46:/var/www/html#

```

i-0fb03f43f8f066c0a (publicA)

LOAD BALANCER:

This is an optional step to create a target group. However, to ensure that your load balancer routes traffic to this target group you must register your targets.

Available instances (2/2)					
	Instance ID	Name	State	Security groups	
<input checked="" type="checkbox"/>	i-0c4ae148c143ec56b	publicB	Running	public	<input type="button" value="Edit"/>
<input checked="" type="checkbox"/>	i-0fb03f43f8f066c0a	publicA	Running	public	<input type="button" value="Edit"/>

2 selected

Ports for the selected instances
Ports for routing traffic to the selected instances.

80
1-65535 (separate multiple ports with commas)

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

Successfully created the target group: public. Anomaly detection is automatically applied to all registered targets. Results can be viewed in the Targets tab.

[EC2](#) > [Target groups](#) > [public](#)

public

[Actions ▾](#)

Details					
arn:aws:elasticloadbalancing:us-east-1:905418355949:targetgroup/public/f3d7c2fcae233ca7					
Target type Instance	Protocol : Port HTTP: 80	Protocol version HTTP1	VPC vpc-0907ee23bedb098f2		
IP address type IPv4	Load balancer None associated				
2 Total targets	0 Healthy	0 Unhealthy	2 Unused	0 Initial	0 Draining
0 Anomalous					
Distribution of targets by Availability Zone (AZ) Select values in this table to see corresponding filters applied to the Registered targets table below.					

WS Services Search [Alt+S]

[EC2](#) [VPC](#)

VPC [Info](#)
The load balancer will exist and scale within the selected VPC. The selected VPC is also where the load balancer targets must be hosted unless routing to Lambda or on-premises targets, or if using VPC peering. To confirm the VPC for your targets, view [target groups](#). For a new VPC, [create a VPC](#).

vpc-0907ee23bedb098f2
IPv4 VPC CIDR: 10.0.0.0/16

Mappings [Info](#)
Select at least two Availability Zones and one subnet per zone. The load balancer routes traffic to targets in these Availability Zones only. Availability Zones that are not supported by the load balancer or the VPC are not available for selection.

Availability Zones

us-east-1a (use1-az6)
Subnet
subnet-03f0354af51d7a87c
IPv4 subnet CIDR: 10.0.13.0/24

IPv4 address
Assigned by AWS

us-east-1b (use1-az1)
Subnet
subnet-03e588a0a629c4cf1
IPv4 subnet CIDR: 10.0.45.0/24

IPv4 address
Assigned by AWS

The screenshot shows the AWS VPC configuration interface. At the top, there's a search bar and a 'Security groups' dropdown menu with the option 'Select up to 5 security groups'. Below it, a list contains 'public' with a delete icon. In the main area, under 'Listeners and routing', a section for 'Listener HTTP:80' is shown. It includes a table with columns 'Protocol', 'Port', 'Default action', and 'Info'. The 'Protocol' is set to 'HTTP' and 'Port' to '80'. The 'Default action' is 'Forward to public' with 'Target type: Instance, IPv4'. A 'Create target group' button is also present. Below this, there's a section for 'Listener tags - optional'.

The screenshot shows a web browser window with the address bar displaying 'public-1753169652.us-east-1.elb.amazonaws.com'. The page content is mostly blank, with a small note at the top left: 'this is frm public B'.

AMI

Create an AMI (image)

- After running the instance, click on actions.
- Click on image and templates and click on create image.
- Give image name as my-image.
- Wait until the image is available

The screenshot shows the AWS EC2 Instances page. At the top, there's a search bar and navigation links for 'N. Virginia' and 'Rohini1176'. Below the header, a table lists two instances:

Name	Instance ID	Instance state	Instance type	Status check
publicA	i-0fb03f43f8f066c0a	Running	t2.micro	2/2 checks passed
publicB	i-0c4ae148c143ec56b	Running	t2.micro	2/2 checks passed

On the right side, there's a 'Actions' dropdown menu with options like 'Connect', 'View details', 'Manage instance state', 'Instance settings', 'Networking', 'Security', 'Image and templates', 'Monitor and troubleshoot', and 'Launch more like this'.

This screenshot shows the 'Add volume' step of the AWS Create Image wizard. It allows you to configure a new volume:

- Storage type:** EBS
- Device:** /dev/sda1
- Snapshot:** Create new snapshot
- Size:** 8
- Volume type:** EBS General Purpose (SSD)
- IOPS:** 3000
- Throughput:** 1000
- Delete on termination:** Enabled
- Encrypted:** Enabled

A note indicates that during the image creation process, Amazon EC2 creates a snapshot of each of the above volumes.

Tags - optional:

- Tag image and snapshots together: Tag the image and the snapshots with the same tag.
- Tag image and snapshots separately: Tag the image and the snapshots with different tags.

No tags associated with the resource. You can add up to 50 more tags.

Buttons at the bottom: 'Cancel' and 'Create image'.

The screenshot shows the AWS Amazon Machine Images (AMIs) page. It displays a single AMI entry:

Name	AMI ID	Source	Owner	Visibility
myimage	ami-0e37f2e704c0b715f	905418355949/myimage	905418355949	Private

Create Autoscaling group.

- For creating autoscaling group we need to create an launch template.
- After available of image. Click on create a launch template.
- Template name as my-public-template, description as nothing.
- Select AMI's as share with me, select my-image.
- Instance type as t2.micro and key pair as project.

- Select existing security group (public) which is used to launch an EC2 instance.
- Now click on create launch template

Auto scaling for the public subnets(instances public-A & public-B)

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

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

Recent | My AMIs | Quick Start

Owned by me Shared with me

Browse more AMIs
Including AMIs from AWS, Marketplace and the Community

Amazon Machine Image (AMI)
myimage
ami-0e37f2e704c0b715f
2024-08-06T09:54:06.000Z Virtualization: hvm ENA enabled: true Root device type: ebs

ntg
ami-0e37f2e704c0b715f

Virtual server type (instance type)
-

Firewall (security group)
-

Storage (volumes)
1 volume(s) - 8 GiB

Free tier: In your first year includes 750 hours of t2.micro (or t3.micro in the Regions in which t2.micro is unavailable) instance usage on free tier AMIs per month, 750 hours of public IPv4

Cancel | Create launch template

Key pair name
fresh

Create new key pair

▼ Network settings Info

Subnet Info
Don't include in launch template

Create new subnet

When you specify a subnet, a network interface is automatically added to your template.

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

Select existing security group Create security group

Security groups Info
Select security groups

Compare security group rules

public sg-01937f4503d7c7dd5 X
VPC: vpc-0907ee23bedb098f2

► Advanced network configuration

▼ Summary

Software Image (AMI)
ntg
ami-0e37f2e704c0b715f

Virtual server type (instance type)
t2.micro

Firewall (security group)
public

Storage (volumes)
1 volume(s) - 8 GiB

Free tier: In your first year includes 750 hours of t2.micro (or t3.micro in the Regions in which t2.micro is unavailable) instance usage on free tier AMIs per month, 750 hours of public IPv4

Cancel | Create launch template

Open autoscaling group.

- Click on create autoscaling group.
- Give name as public-asg.
- Choose the created launch template (publicLT) and click on next.
- Select our VPC (VPC-PRO), and both public subnets.

- Click on next and click on no load balancer.
- Give desired capacity as 2 in sizing desired capacity min – 2 and max – 3 and click on next.
- Click on next and click on create auto scaling group

st-1.console.aws.amazon.com/ec2/home?region=us-east-1#CreateAutoScalingGroup:

Frequently Asked C... [Frequently asked C...](#)

Search [Alt+S] |

N. Virginia Rohini1176

Network [Info](#)

For most applications, you can use multiple Availability Zones and let EC2 Auto Scaling balance your instances across the zones. The default VPC and default subnets are suitable for getting started quickly.

VPC
Choose the VPC that defines the virtual network for your Auto Scaling group.

vpc-0907ee23bedb098f2 (VPC-PRO)
10.0.0.0/16 [Edit](#)

[Create a VPC](#)

Availability Zones and subnets
Define which Availability Zones and subnets your Auto Scaling group can use in the chosen VPC.

Select Availability Zones and subnets [Edit](#)

us-east-1a | subnet-03f0354af51d7a87c (public-A) X
10.0.13.0/24

us-east-1b | subnet-03e588a0a629c4cf1 (public-B) X
10.0.45.0/24

[Create a subnet](#)

Cancel [Skip to review](#) [Previous](#) **Next**

[Alt+F5] N. Virginia

Desired capacity type
Choose the unit of measurement for the desired capacity value. vCPUs and Memory(GiB) are only supported for mixed instances groups configured with a set of instance attributes.

Units (number of instances) ▾

Desired capacity
Specify your group size.

2

Scaling Info
You can resize your Auto Scaling group manually or automatically to meet changes in demand.

Scaling limits
Set limits on how much your desired capacity can be increased or decreased.

Min desired capacity	Max desired capacity
2	3

Equal or less than desired capacity Equal or greater than desired capacity

Automatic scaling - optional
Choose whether to use a target tracking policy [Info](#)
You can set up other metric-based scaling policies and scheduled scaling after creating your Auto Scaling group.

No scaling policies Target tracking scaling policy

Instances (4) [Info](#)

Find Instance by attribute or tag (case-sensitive)

All states ▾

1

Instances (4)

Name	Instance ID	Instance state	Instance type	Status check	Alarm status	Availability Zone	Pub
pubas1	i-0ac759bcde30fdef7	Running	t2.micro	2/2 checks passed	View alarms +	us-east-1a	-
publicA	i-0fb03fa43f8f066c0a	Running	t2.micro	2/2 checks passed	View alarms +	us-east-1a	ec2
publicB	i-0c4ae148c143ec56b	Running	t2.micro	2/2 checks passed	View alarms +	us-east-1b	ec2
pubas2	i-0cb9db476d52f1045	Running	t2.micro	2/2 checks passed	View alarms +	us-east-1b	-

In similar way create another launch template name as my-privatemplate and create auto scaling group name as private-asg. • In it select create VPC and give two private subnets

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

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

Recent | **My AMIs** | Quick Start

Owned by me | Shared with me

Amazon Machine Image (AMI)

myimage	ami-0e37f2e704c0b715f	2024-08-06T09:54:06.000Z	Virtualization: hvm	ENI enabled: true	Root device type: ebs
---------	-----------------------	--------------------------	---------------------	-------------------	-----------------------

Description: ntg

Architecture: x86_64 AMI ID: ami-0e37f2e704c0b715f

Summary

Software Image (AMI): ami-0e37f2e704c0b715f

Virtual server type (instance type): -

Firewall (security group): -

Storage (volumes): 1 volume(s) - 8 GiB

Free tier: In your first year includes 750 hours of t2.micro (or t3.micro in the Regions in which t2.micro is unavailable) instance usage on free tier AMIs per month, 750 hours of public IPv4

Cancel | **Create launch template**

Search [Alt+S] | N. Virginia | Region

Network Info

For most applications, you can use multiple Availability Zones and let EC2 Auto Scaling balance your instances across the zones. The default VPC and default subnets are suitable for getting started quickly.

VPC
Choose the VPC that defines the virtual network for your Auto Scaling group.

vpc-0907ee23bedb098f2 (VPC-PRO) 10.0.0.0/16

Create a VPC

Availability Zones and subnets
Define which Availability Zones and subnets your Auto Scaling group can use in the chosen VPC.

Select Availability Zones and subnets

us-east-1a | subnet-0f9f37240ee41bb43 (Private-1A) 10.0.34.0/24

us-east-1b | subnet-0bf7ec612cf810ff8 (private-2B) 10.0.12.0/24

Create a subnet

Cancel | Skip to review | Previous | **Next**

[Alt+S] New N. Virginia ▾

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.
 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-PRO

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-1a

resolution.

us-east-1a

us-east-1b

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	<input type="text" value="Create a target group"/> New target group name An instance target group with default settings will be created. <input type="text" value="priv12-1"/>

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
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.

[Alt+S] N. Virginia ▾

Desired capacity
Specify your group size.

Scaling Info
You can resize your Auto Scaling group manually or automatically to meet changes in demand.

Scaling limits
Set limits on how much your desired capacity can be increased or decreased.

Min desired capacity <input type="text" value="2"/>	Max desired capacity <input style="border: 1px solid #ccc; padding: 2px; width: 20px; height: 20px; vertical-align: middle;" type="text" value="3"/> ▼
Equal or less than desired capacity	Equal or greater than desired capacity

Automatic scaling - optional
Choose whether to use a target tracking policy Info
You can set up other metric-based scaling policies and scheduled scaling after creating your Auto Scaling group.

<input checked="" type="radio"/> No scaling policies Your Auto Scaling group will remain at its initial size and will not dynamically resize to meet demand.	<input type="radio"/> Target tracking scaling policy Choose a CloudWatch metric and target value and let the scaling policy adjust the desired capacity in proportion to the metric's value.
--	--

Instance maintenance policy Info

Instances (6) Info

Find Instance by attribute or tag (case-sensitive) All states ▾

<input type="checkbox"/>	Name ▼	Instance ID	Instance state ▼	Instance type ▼	Status check	Alarm status	Availability Zone ▼	Pub
<input type="checkbox"/>	pubas1	i-0ac759bcde30fdef7	Running Q Q	t2.micro	Q 2/2 checks passed	View alarms +	us-east-1a	-
<input type="checkbox"/>	publicA	i-0fb03f43f8f066c0a	Running Q Q	t2.micro	Q 2/2 checks passed	View alarms +	us-east-1a	ec2
<input type="checkbox"/>		i-051d1e529c9ddcc4a	Pending Q Q	t2.micro	-	View alarms +	us-east-1b	-
<input type="checkbox"/>	publicB	i-0c4ae148c143ec56b	Running Q Q	t2.micro	Q 2/2 checks passed	View alarms +	us-east-1b	ec2
<input type="checkbox"/>	pubas2	i-0cb9db476d52f1045	Running Q Q	t2.micro	Q 2/2 checks passed	View alarms +	us-east-1b	-
<input type="checkbox"/>		i-07b65b99594872956	Pending Q Q	t2.micro	-	View alarms +	us-east-1a	-

Create subnet group

- Give name as my-subnet-grp and description nothing.
- Select created VPC
 - Give availability zones and select private subnets from each zone.
 - Create the DB subnet group.

The screenshot shows the AWS RDS Subnet groups page. At the top, there is a search bar and a navigation breadcrumb: RDS > Subnet groups. Below the breadcrumb, the title "Subnet groups (0)" is displayed. There is a "Create DB subnet group" button in an orange box. A message states "No db subnet groups" and "You don't have any db subnet groups." A "Create DB subnet group" button is located at the bottom of the list.

Create Database(RDS) • Click on create database, select standard create, select engine type as MySQL. • Select templates as production and select multi-AZ DB cluster. • Select on self-managed, give password and confirm the password. • Select memory optimized class. • In connectivity, click on Don't connect to the EC2 compute resource and select created vpc (my-project-vpc). • Select subnet group (my-subnet-grp) and give public access as yes. • Choose existing security group (project-sg). • Go to VPC dashboard, click on VPC, click on actions, go to edit VPC settings and click on the enable DNS hostnames. • Create the database

The screenshot shows the AWS VPC Subnet settings configuration page. The top navigation bar includes services like EC2 and VPC. The main section is titled "Subnet settings" with the sub-section "Subnet 1 of 1". It contains fields for "Subnet name" (with a value of "extra"), "Availability Zone" (set to "US East (N. Virginia) / us-east-1c"), "IPv4 VPC CIDR block" (set to "10.0.0.0/16"), and "IPv4 subnet CIDR block" (set to "10.0.0.87").

You have successfully created 1 subnet: subnet-03715c34b4f583cd

Subnets (1) Info

Last updated less than a minute ago

Actions Create subnet

Name	Subnet ID	State	VPC	IPv4 CIDR
extra	subnet-03715c34b4f583cd	Available	vpc-0907ee23bedb098f2 VPC...	10.0.0.0/24

Select a subnet

arch [Alt+S]

RDS > Subnet groups

Subnet groups (0)

Create DB subnet group

No db subnet groups
You don't have any db subnet groups.

Create DB subnet group

Create Database(RDS)

- Click on create database, select standard create, select engine type as MySQL.
- Select templates as production and select multi-AZ DB cluster.
- Select on self-managed, give password and confirm the password.
- Select memory optimized class.
- In connectivity, click on Don't connect to the EC2 compute resource and select created vpc (VPC-PRO).
- Select subnet group (extra) and give public access as yes.
- Choose existing security group (PUBLIC).
- Go to VPC dashboard, click on VPC, click on actions, go to edit VPC settings and click on the enable DNS hostnames.
- Create the database

The screenshot shows the 'Engine options' section of the AWS RDS console. On the left, there's a grid of database engine icons. MySQL is selected and highlighted with a blue border. To its right is a detailed description of MySQL, including its popularity, features like automated backup, and support for up to 15 read replicas per instance. The MySQL icon is a blue globe with orange and yellow highlights.

Engine options

Engine type: Info

- Aurora (MySQL Compatibl)
- Aurora (PostgreSQL Compatibl)
- MySQL
- MariaDB
- PostgreSQL
- Oracle
- Microsoft SQL Server
- IBM DB2

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.

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

This screenshot shows the 'Launch instance' step of the AWS RDS creation wizard. It includes sections for 'Templates', 'Availability and durability', and 'Deployment options'. The MySQL template is selected and highlighted with a blue border. The MySQL deployment option is also selected and highlighted. Both descriptions and icons are identical to the ones in the previous screenshot.

Templates

Choose a sample template to meet your use case.

- Production
- Dev/Test
- Free tier

Availability and durability

Deployment options Info

The deployment options below are limited to those supported by the engine you selected above.

- Multi-AZ DB Cluster
- Multi-AZ DB Instance
- Single DB instance

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.

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

Add subnets

Availability Zones
Choose the Availability Zones that include the subnets you want to add.

Select an availability zone ▾

us-east-1a X us-east-1b X us-east-1c X

Subnets
Choose the subnets that you want to add. The list includes the subnets in the selected Availability Zones.

Select subnets ▾

subnet-009157f289c8c9b2c (10.0.76.0/24) X
 subnet-0511ebc30119a6ad1 (10.0.83.0/24) X
 subnet-03715c34b4f583cd (10.0.0.0/24) X

Info For Multi-AZ DB clusters, you must select 3 subnets in 3 different Availability Zones.

Subnets selected (3)

WS Services Search [Alt+S]

EC2 VPC

privpro 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 ▾

public X

RDS Proxy
RDS Proxy is a fully managed, highly available database proxy that improves application scalability, resiliency, and security.

Create an RDS Proxy Info RDS automatically creates an IAM role and a Secrets Manager secret for the proxy. RDS Proxy has additional costs. For more information, see [Amazon RDS Proxy pricing](#).

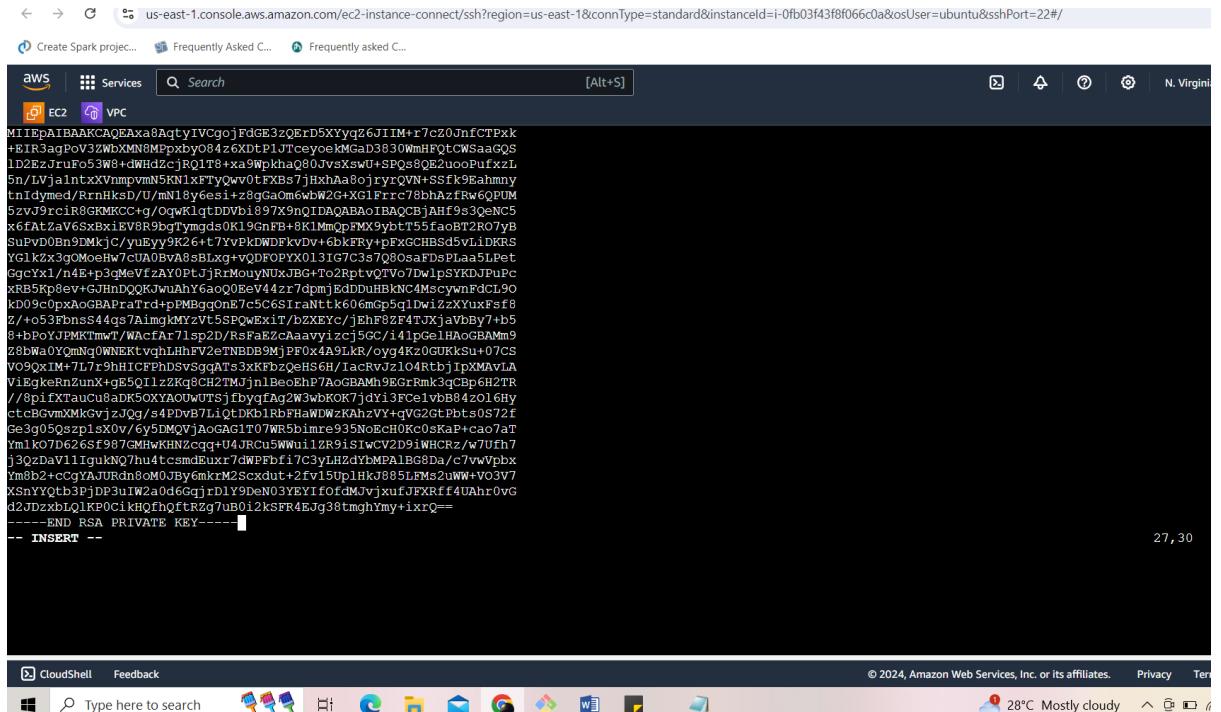
CloudShell Feedback © 2024, Amazon Web

The screenshot shows the 'Edit VPC settings' page for a VPC named 'my-project-vpc'. Key details include:

- VPC details:** VPC ID: `vpc-0d7f16b4be6ae8b55`, Name: `my-project-vpc`.
- DHCP settings:** DHCP option set: `dopt-0873a7fc1667054c`.
- DNS settings:** Enable DNS resolution: `checked`, Enable DNS hostnames: `checked`.
- Network Address Usage metrics settings:**

The screenshot shows the 'Databases' page for an RDS MySQL cluster. The table displays the following information:

DB identifier	Status	Role	Engine	Region & AZ	Size
<u>database-1</u>	Backing-up	Multi-AZ DB cluster	MySQL Community	us-east-1	3 instances
— database-1-instance-1	Available	Writer instance	MySQL Community	us-east-1a	db.m5d.large
— database-1-instance-2	Available	Reader instance	MySQL Community	us-east-1b	db.m5d.large
— database-1-instance-3	Available	Reader instance	MySQL Community	us-east-1c	db.m5d.large



```

< → ⌂ us-east-1.console.aws.amazon.com/ec2-instance-connect/ssh?region=us-east-1&connType=standard&instanceId=i-0fb03f43f8f066c0a&osUser=ubuntu&sshPort=22#/
Create Spark project... Frequently Asked C... Frequently asked C...
aws Services Search [Alt+S] N. Virginia
EC2 VPC
MIIIEpaIBAAKCAQEAxa8AqtyIVCqo]FdgE3zQErD5XYyqZ6JIIM+r7cZ0JnfCTPxk
+EIRe3agPoV3WbXNMMPpxbyo8z6DtP1JtceyoekGad330WmHFotCSaGOS
1D2RzJrufo53W8+dWbd2CjRQ1T8+x+a9wpkhaq80JvsXswD+SPQs8pE2uooPuFxzL
5n/1VjalntxXNmpvNmKN1xTyQvv0TEFBs7jhXhAa8ojryzQVN+SSfx9Eahmy
tnIdnyed/RrnHksD/U/mN18y6es1+z8gGaOn6wbW2G+XG1Frcc7bhA2fRwQfUM
5zv09rc1R8gRMKCC+g/OgwklqgtDDVb197X9nQ1DAQABAc1BAQCBfAHf9s3qeNC5
x6fatzaV6SxBiEV8R9bgfymqds0K19gnFB+8KMmQpFMX9ybT55faotB2R07yB
SuPVD0Bn9DMk/C/yuuyy9R26+tYYPkWDFkvDv+6bKEFry+pFXGCHBSd5Li1DRRS
YGlKzx3qMoeHw7CUa0BvA8sLxg+vQDFOPYX0131G7C3s7g8OsarDsPla5Lpet
GgcYx1/n4E+p3qMeV1zAYOPTjkrMouyNUxJBG+To2RptQTVo7DwlpsYKDJPuPc
xR85Kp8ev+GJhnDQQRJwuAyHgaoQ0Bev44zz7dpmjEdDuHBkNC4MscyvnFcl90
kd09c0pxAogRAPratraDrd+pPMBgqqnb7c5c6StraNttk606Gp5q1Dwi2zXXuxFsF8
Z/+053EBnss44gs7aimqkMyzVt5SPQwXxLT/bZXELYC/jEhF8Zf4TXj|avBBy7+b5
9+bPoYJPMKTMwT/WAcfa1r71sp2D/RsFaEzcaayvi_zcj5GC/i41pGe1HaoGBAMm9
28BWa0YqmNg0WNNEKtvghlHhFV2eTNEDB9MjPF0x4a9LKR/oyg4KzOGUKKsu+07CS
VO9QXIM+7L/r5hHICFPHD5vsgqATs3xRfdZqeHS6H/taCRVzlo4RtbjtpXMAVLA
ViEgkeRnZuunX+gE5Q11zZkqgCH2TMjJnlBeeEHp7AeGBAMh9EGrRmk3qBp6H2TR
//8pi1fxtaucu8abDK5OXYAOuWuTSjfbycfgz2W3wbOKR7j0Y13fce1vbB94z01GhY
ctcBGMwXMKcvjzQg/s4PDvB7LiqtDKh1rbFHAwDWzKAhzVV+yQG2GtPbtsoS72f
Ge3g0505zp1sX0v/6y5IMQVJAaoGAG1T07WR5bimre935NeECHOKECoSkap+cao7aT
ymlko7D626sf987CMwKHNIcqg+U4jRCu5WNu11ZB91sIwCV2D91wHCRz/w7Ufh7
j3Q2DaV11IguKw07hu44tcsmduxr7dWFbf17C3ylH2dzYbMPAlBG8Da/c7vwVpbx
Ym8B2+ccqgYAJRdn8oM0JBy6mkRM2Scxdut+2fv15Up1HKJ885LFWszuWW+vO3V7
XShnYQtb3PjDP3uIW2a0d6GqjrfDly9DcN03YEYffofdMjvjxuiJFXRif4UAhr0VG
d2JDz2b1Q1kP0C1khQfhQqttrZg/uBo12kSF94E0g38tmghYmy+ixrQ==

-- END RSA PRIVATE KEY -- | 27,30
-- INSERT --

```

Establish connection

- Go to EC2 instance, click on project-ec2, open it click on connect and again click on connect.

Give commands as: 1. sudo -i (convert from normal user to root user). 2. apt update -y 3. sudo apt install mysql-server (to install mysql)

EC2 VPC

```
Welcome to Ubuntu 24.04 LTS (GNU/Linux 6.8.0-1009-aws x86_64)

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

System information as of Tue Aug 6 10:47:24 UTC 2024

System load: 0.0          Processes:           108
Usage of /: 26.3% of 6.71GB  Users logged in: 0
Memory usage: 31%          IPv4 address for enX0: 10.0.13.46
Swap usage: 0%           

Expanded Security Maintenance for Applications is not enabled.

48 updates can be applied immediately.
27 of these updates are standard security updates.
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: Tue Aug 6 09:44:47 2024 from 18.206.107.29
ubuntu@ip-10-0-13-46:~$ sudo -i
root@ip-10-0-13-46:~# vi fresh.pem
root@ip-10-0-13-46:~# chmod 400 fresh.pem
root@ip-10-0-13-46:~# 
```

us-east-1.console.aws.amazon.com/ec2-instance-connect/ssh?region=us-east-1&connType=standard&instanceId=i-0fb03f43f8f066c0a&osUser=ubuntu&sshPort=22#/

Create Spark project... Frequently Asked C... Frequently asked C...

EC2 VPC

```
root@ip-10-0-13-46:~# mysql -h database-1.cluster-c7q8ae2wkwe.us-east-1.rds.amazonaws.com -u admin -p
Enter password:
ERROR 1045 (28000): Access denied for user 'admin'@'10.0.13.46' (using password: YES)
root@ip-10-0-13-46:~# mysql -h database-1.cluster-c7q8ae2wkwe.us-east-1.rds.amazonaws.com -u admin -p
Enter password:
Welcome to the MySQL monitor. Commands end with ; or \g.
Your MySQL connection id is 60
Server version: 8.0.35 Source distribution

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

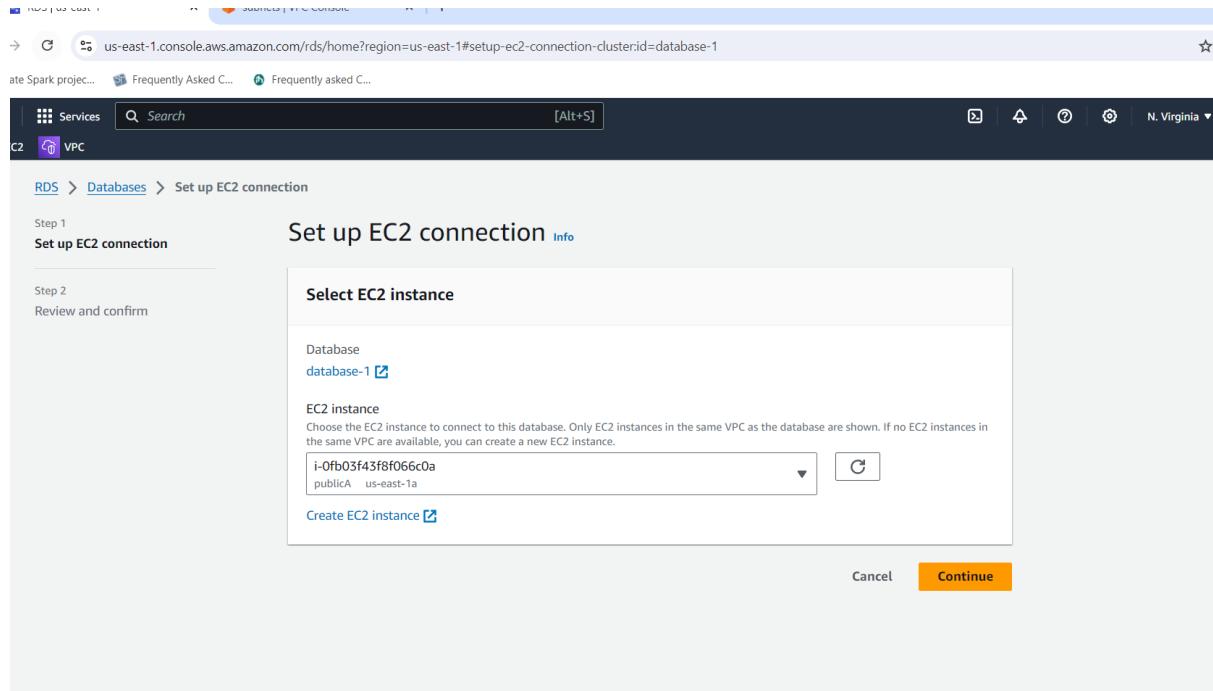
Oracle is a registered trademark of Oracle Corporation and/or its
affiliates. Other names may be trademarks of their respective
owners.

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

i-0fb03f43f8f066c0a (publicA)

Public IPs: 52.203.133.220 Private IPs: 10.0.13.46

Click on database-1, come down and click on set EC2 connection. • Choose our instance (project-ec2). • Click on continue and set up it.



Now we can see that we are connected to MySQL server.

Use commands like:

- show databases; (to show the list of databases are there)
-) • create database chinni; (to create a new database)
- use rohini; (server will take that database to use)
- create a table using this command

```
owners.

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

mysql> show database
      -> create database
      -> /c
      -> help
      -> ^C
mysql> show databases;
+-----+
| Database |
+-----+
| information_schema |
| mysql |
| performance_schema |
| sys |
+-----+
4 rows in set (0.00 sec)

mysql> i-0fb03f43f8f066c0a (publicA)
PublicIPs: 52.203.133.220  PrivateIPs: 10.0.13.46
```

```
EC2 VPC
mysql> show databases;
+-----+
| Database |
+-----+
| information_schema |
| mysql |
| performance_schema |
| sys |
+-----+
4 rows in set (0.00 sec)

mysql> create database rohini;
Query OK, 1 row affected (0.04 sec)

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

mysql> 
```

i-0fb03f43f8f066c0a (publicA)
PublicIPs: 52.203.133.220 PrivateIPs: 10.0.13.46

[CloudShell](#) [Feedback](#) © 2024, Amazon Web Services, Inc. or its affiliates. [Privacy](#) [Terms](#)

```
CREATE TABLE Persons (
    'Emp_ID' varchar(20) NOT NULL,
    'Name' varchar(255) NOT NULL,
    'Age' INT NOT NULL,
    'Phone_No' varchar(10) NOT NULL,
    'Address' varchar(100) NOT NULL
    PRIMARY KEY (Emp_ID)
);
```

Insert the fields and the values

```
INSERT INTO Employee (Emp_ID, Name, Age, Phone_No, Address ,)
VALUES ('6985', 'VELPULA', '21', '9550652991', 'Kothapeta');
```

EC2 VPC

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

mysql> CREATE TABLE Persons (
    ->     PersonID int,
    ->     LastName varchar(255),
    ->     FirstName varchar(255),
    ->     City varchar(255)
    -> );
ERROR 3750 (HY000): Unable to create or change a table without a primary key, when the system variable 'sql_require_primary_key' is set
table or unset this variable to avoid this message. Note that tables without a primary key can cause performance problems in row-based
ult your DBA before changing this setting.
mysql> CREATE TABLE `Employee` ( `Emp_ID` VARCHAR(20) NOT NULL ,`Name` VARCHAR(50) NOT NULL ,
-> `Age` INT NOT NULL ,`Phone_No` VARCHAR(10) NOT NULL ,`Address` VARCHAR(100) NOT NULL ,
-> PRIMARY KEY (`Emp_ID`));
Query OK, 0 rows affected (0.01 sec)

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

mysql> INSERT INTO Employee (Emp_ID, Name, Age, Phone_No, Address, Country)
-> VALUES ('16985', 'VELPULIA', '21', '9550652991', 'kothapeta', 'india');
ERROR 1054 (42S22): Unknown column 'Country' in 'field list'
mysql> INSERT INTO Employee (Emp_ID, Name, Age, Phone_No, Address)
-> VALUES ('16985', 'VELPULIA', '21', '9550652991', 'kothapeta');
Query OK, 1 row affected (0.01 sec)

mysql> show tables;
```

```
mysql> show tables rohini;
ERROR 1064 (42000): You have an error in your SQL syntax; check the manual that corresponds to your MySQL server version for the right syntax to use near 'rohini'
at line 1
mysql> show database rohini;
ERROR 1064 (42000): You have an error in your SQL syntax; check the manual that corresponds to your MySQL server version for the right syntax to use near 'databas
e rohini' at line 1
mysql> select from Employee_table
-> ^C
mysql> show tables;
+-----+
| Tables_in_rohini |
+-----+
| Employee         |
+-----+
1 row in set (0.00 sec)

mysql> select * from Employee;
+-----+-----+-----+-----+
| Emp_ID | Name  | Age   | Phone_No | Address  |
+-----+-----+-----+-----+
| 16985  | VELPULIA | 21   | 9550652991 | kothapeta |
+-----+-----+-----+-----+
1 row in set (0.00 sec)

mysql>
```