

# 3 – TIER ARCHITECTURE

## ❖ What Is Meant By 3-Tier Architecture?

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

- 1.Presentation tier
- 2.Application tier
- 3.Data tier

- **Presentation tier:** Also known as the user interface, web tier or frontend this is where the end –user interacts with the system.
- **Application tier:** Also known as the middle tier or logic tier this is the core of the application where information is processed using business logic.
- **Data tier:** Also known as the databases tier, back-end, or data access tier, this is where the application's data is stored, managed, retrieved and manipulated.

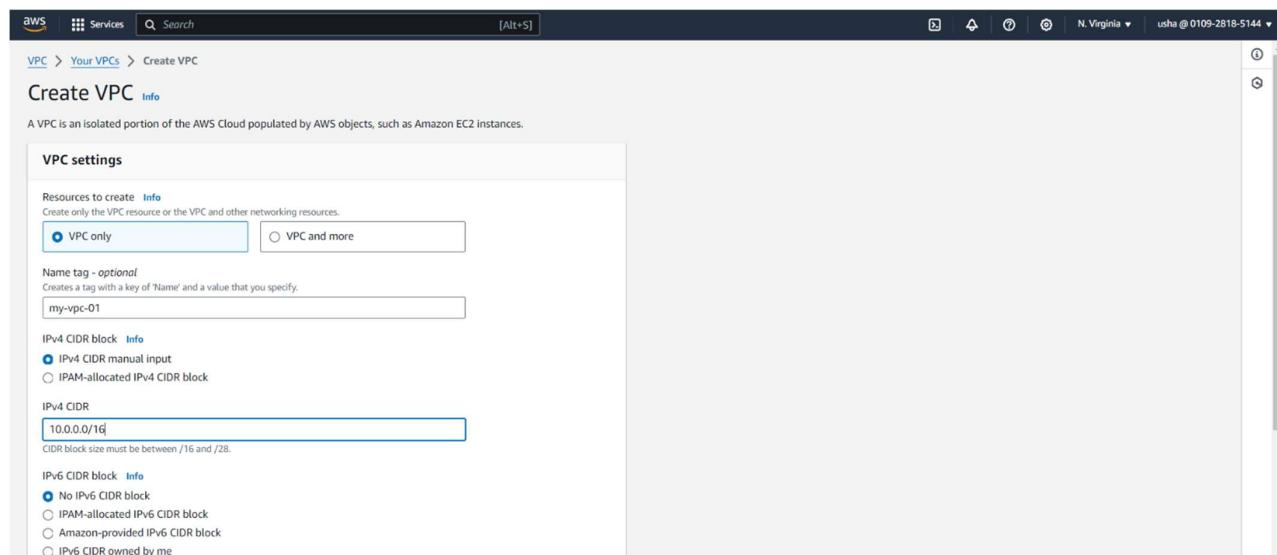


## ❖ Benefits of 3-TIER ARCHITECTURE:

1. **Scalability:** Each tier can scale independently, allowing organizations to optimize their resources minimize costs.
2. **Reliability:** Each tier can be replicated across multiple servers, improving application availability and reliability.
3. **Performance:** By dividing the application into separate layers, 3-tier architecture reduces network traffic and enhances application performance.
4. **Security:** Each tier can have its own security group, allowing different organizations to implement customized security measures for each layer.

## ❖ Create VPC In US – EAST Region:

1. login to the AWS account & select VIRGINIA region & do search for VPC in the search box.
2. click on create VPC
3. select VPC only & give name & give IPv4 CIDR and then
4. click on create VPC.



## ❖ Create SUBNETS (2- Public & 4 - Private):

1. Click on SUBNETS & Click on create subnet& Choose VPC ID (Own not Default)
2. give subnet name & select availability zone (2a or 2b) & give IPv4 subnet CIDR BLOCK
3. click on create subnet.
4. like that create 6 subnets – 2 public subnets in 2a & 2b zone and 4 private subnets – take 2 private subnets in 2a & remaining 2 private subnets in 2b zone.

## 5. some snapshots of subnets are added below.

The screenshot shows the 'Create subnet' page in the AWS VPC service. At the top, there's a navigation bar with the AWS logo, 'Services' button, and a search bar. Below the navigation is a breadcrumb trail: 'VPC > Subnets > Create subnet'. The main area is titled 'Create subnet' with an 'Info' link. A large section is labeled 'VPC' with a 'VPC ID' field containing 'vpc-0d06022cba87624ef (my-vpc-01)'. Below this is a 'Associated VPC CIDRs' section with an 'IPv4 CIDRs' field containing '10.0.0.0/16'.

This screenshot shows the 'Create subnet' page with more detailed configuration options. It includes fields for 'Subnet name' ('public-subnet-01'), 'Availability Zone' ('US East (N. Virginia) / us-east-1a'), 'IPv4 VPC CIDR block' ('10.0.0.0/16'), and 'IPv4 subnet CIDR block' ('10.0.1.0/24'). There is also a 'Tags - optional' section with a single tag ('Name: public-subnet-01') and a note about adding more tags.

This screenshot shows a second instance of the 'Create subnet' page with identical configuration to the previous one. It has a 'Subnet name' of 'public-subnet-01', 'Availability Zone' of 'US East (N. Virginia) / us-east-1a', and 'IPv4 subnet CIDR block' of '10.0.1.0/24'. It also includes a 'Tags - optional' section with a single tag ('Name: public-subnet-01').

This screenshot shows a third instance of the 'Create subnet' page with configuration matching the others. It has a 'Subnet name' of 'public-subnet-01', 'Availability Zone' of 'US East (N. Virginia) / us-east-1a', and 'IPv4 subnet CIDR block' of '10.0.1.0/24'. It includes a 'Tags - optional' section with a single tag ('Name: public-subnet-01').

**Subnet 1 of 1**

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

**Availability Zone** [Info](#)  
Choose the zone in which your subnet will reside, or let Amazon choose one for you.

**IPv4 VPC CIDR block** [Info](#)  
Choose the VPC's IPv4 CIDR block for the subnet. The subnet's IPv4 CIDR must lie within this block.

**IPv4 subnet CIDR block**  
 256 IPs  
< > ^ v

**Tags - optional**

Key	Value - optional
<input type="text" value="Name"/> X	<input type="text" value="public-subnet-02"/> X
<a href="#">Add new tag</a>	

You can add 49 more tags.

**Subnet 1 of 1**

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

**Availability Zone** [Info](#)  
Choose the zone in which your subnet will reside, or let Amazon choose one for you.

**IPv4 VPC CIDR block** [Info](#)  
Choose the VPC's IPv4 CIDR block for the subnet. The subnet's IPv4 CIDR must lie within this block.

**IPv4 subnet CIDR block**  
 256 IPs  
< > ^ v

**Tags - optional**

Key	Value - optional
<input type="text" value="Name"/> X	<input type="text" value="private-subnet-01"/> X
<a href="#">Add new tag</a>	

You can add 49 more tags.

**VPC**

**VPC ID**  
Create subnets in this VPC.

**Associated VPC CIDRs**

IPv4 CIDRs

**Subnet 1 of 1**

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

**Availability Zone** [Info](#)  
Choose the zone in which your subnet will reside, or let Amazon choose one for you.

**IPv4 VPC CIDR block** [Info](#)  
Choose the VPC's IPv4 CIDR block for the subnet. The subnet's IPv4 CIDR must lie within this block.

**IPv4 subnet CIDR block**  
 256 IPs  
< > ^ v

**Subnet 1 of 1**

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

**Availability Zone** [Info](#)  
Choose the zone in which your subnet will reside, or let Amazon choose one for you.

**IPv4 VPC CIDR block** [Info](#)  
Choose the VPC's IPv4 CIDR block for the subnet. The subnet's IPv4 CIDR must lie within this block.

**IPv4 subnet CIDR block**  
 256 IPs

**Tags - optional**

**Subnet 1 of 1**

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

**Availability Zone** [Info](#)  
Choose the zone in which your subnet will reside, or let Amazon choose one for you.

**IPv4 VPC CIDR block** [Info](#)  
Choose the VPC's IPv4 CIDR block for the subnet. The subnet's IPv4 CIDR must lie within this block.

**IPv4 subnet CIDR block**  
 256 IPs

**VPC dashboard** [X](#)

**Subnets (8) Info**

Name	Subnet ID	State	VPC	IPv4 CIDR	IPv6 CIDR
-	subnet-09b2d6ca37867b94	Available	vpc-09cccd650406d26621	172.31.80.0/20	-
public-subnet-02	subnet-0e9d00f8477fd5282	Available	vpc-0d06022cba87624ef   my-v...	10.0.2.0/24	-
public-subnet-01	subnet-0fb30c664548ed663	Available	vpc-0d06022cba87624ef   my-v...	10.0.1.0/24	-
-	subnet-00b9791dd87d3448	Available	vpc-09cccd650406d26621	172.31.32.0/20	-
private-subnet-01	subnet-0507cc4548af0cf24	Available	vpc-0d06022cba87624ef   my-v...	10.0.3.0/24	-
private-subnet-02	subnet-0021fbe703d99afcd	Available	vpc-0d06022cba87624ef   my-v...	10.0.4.0/24	-
private-subnet-rds-01	subnet-0cf588fbfc0b6921b	Available	vpc-0d06022cba87624ef   my-v...	10.0.5.0/24	-
private-subnet-rds-02	subnet-0dab6a45712cb2338	Available	vpc-0d06022cba87624ef   my-v...	10.0.6.0/24	-

## ❖ Create INTERNET GATEWAY:

- 1.click on internet gateway &create internet gateway.
2. after the creation of internet gateway, click on internet gateway, click on actions & attach it to VPC.
- 3.Snapshots of internet gateway are attached below.

aws Services Search [Alt+S]

VPC > Internet gateways > Create internet gateway

### Create internet gateway Info

An internet gateway is a virtual router that connects a VPC to the internet. To create a new internet gateway specify the name for the gateway below.

**Internet gateway settings**

Name tag  
Creates a tag with a key of 'Name' and a value that you specify.

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

Key	Value - optional
<input type="text" value="Name"/>	<input type="text" value="my-igw-01"/>
<a href="#">Remove</a>	
<a href="#">Add new tag</a>	

You can add 49 more tags.

[Cancel](#) [Create internet gateway](#)

aws Services Search [Alt+S]

VPC dashboard > VPC > Internet gateways > igw-0a0252ec9af543a68

### igw-0a0252ec9af543a68 / my-igw-01

**Details** Info

Internet gateway ID <a href="#">igw-0a0252ec9af543a68</a>	State <a href="#">Detached</a>	VPC ID -	Owner 010928
--	-----------------------------------	-------------	-----------------

**Actions** ▾

- [Attach to VPC](#) (highlighted)
- [Detach from VPC](#)
- [Manage tags](#)
- [Delete](#)

**Tags**

<a href="#">Search tags</a>	
Key	Value
Name	my-igw-01

aws Services Search [Alt+S]

VPC dashboard > VPC > Internet gateways > igw-0a0252ec9af543a68 successfully attached to [vpc-0d06022cba87624ef](#) | my-vpc-01

### igw-0a0252ec9af543a68 / my-igw-01

**Details** Info

Internet gateway ID <a href="#">igw-0a0252ec9af543a68</a>	State <a href="#">Attached</a>	VPC ID <a href="#">vpc-0d06022cba87624ef</a>   my-vpc-01	Owner 010928185144
--	-----------------------------------	---	-----------------------

**Actions** ▾

**Tags**

<a href="#">Search tags</a>	
Key	Value
Name	my-igw-01

## ❖ Create NATGATEWAY:

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

Elastic IP address 34.235.245.166 (eipalloc-089efa47f485fea3a) allocated.

### 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.

The name can be up to 256 characters long.

Subnet  
Select a subnet in which to create the NAT gateway.

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.

NAT gateway nat-057c6da972383543b | my-ngw-01 was created successfully.

VPC dashboard X

VPC > NAT gateways > nat-057c6da972383543b

nat-057c6da972383543b / my-ngw-01 Actions ▾

Details			
NAT gateway ID <a href="#">nat-057c6da972383543b</a>	Connectivity type Public	State <a href="#">Pending</a>	State message <small>Info</small> -
NAT gateway ARN <a href="#">arn:aws:ec2:us-east-1:010928185144:natgateway/nat-057c6da972383543b</a>	Primary public IPv4 address -	Primary private IPv4 address -	Primary network interface ID -
VPC <a href="#">vpc-0d06022cba87624ef / my-vpc-01</a>	Subnet <a href="#">subnet-0507ce4548af0cf24 / private-subnet-01</a>	Created <a href="#">Saturday 24 August 2024 at 15:29:00 GMT+5:30</a>	Deleted -

## ❖ Create ROUTE TABLES:

1.we have to create 2 route tables – one is public & another is private.

2.goto route table – click on create route- select VPC & create route table.

VPC > Route tables > Create route table

### Create route table Info

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

#### Route table settings

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

VPC  
The VPC to use for this route table.

3. click on route- actions-edit subnet associations-select PUBLIC SUBNETS – save associations.

AWS Services Search [Alt+S] N. Virginia usha @ 0109-2818-5144

VPC dashboard EC2 Global View Filter by VPC Virtual private cloud Your VPCs Subnets Route tables Internet gateways Egress-only Internet gateways Carrier gateways DHCP option sets Elastic IPs Managed prefix lists Endpoints Endpoint services

Route table rtb-0da51a73654360b02 | public-route-table was created successfully.

VPC > Route tables > rtb-0da51a73654360b02 / public-route-table

**Details** Info

Route table ID	Main	Explicit subnet associations	Edge associations
rtb-0da51a73654360b02	No	-	-
VPC	Owner ID	Actions	
vpc-0d06022cba87624ef   my-vpc-01	010928185144	Set main route table	
Edit subnet associations			
Edit edge associations			
Edit route propagation			
Edit routes			
Manage tags			
Delete			

Routes Subnet associations Edge associations Route propagation Tags

**Routes (1)**

Destination	Target	Status	Propagated
10.0.0.0/16	local	Active	No

Both Edit routes < 1 >

aws Services Search [Alt+S] N. Virginia usha @ 0109-2818-5144

VPC > Route tables > rtb-0da51a73654360b02 > Edit subnet associations

### Edit subnet associations

Change which subnets are associated with this route table.

**Available subnets (2/6)**

Name	Subnet ID	IPv4 CIDR	IPv6 CIDR	Route table ID
public-subnet-02	subnet-0e9d00f8477fd5282	10.0.2.0/24	-	Main (rtb-05fd860fc1e8baae)
public-subnet-01	subnet-0fb30c664548ed663	10.0.1.0/24	-	Main (rtb-05fd860fc1e8baae)
private-subnet-01	subnet-0507ce4548af0cf24	10.0.3.0/24	-	Main (rtb-05fd860fc1e8baae)
private-subnet-02	subnet-0021fbe703d99afcd	10.0.4.0/24	-	Main (rtb-05fd860fc1e8baae)
private-subnet-rds-01	subnet-0cf58ff8fc0b6921b	10.0.5.0/24	-	Main (rtb-05fd860fc1e8baae)
private-subnet-rds-02	subnet-0dab6a45712cb2338	10.0.6.0/24	-	Main (rtb-05fd860fc1e8baae)

**Selected subnets**

- subnet-0fb30c664548ed663 / public-subnet-01
- subnet-0e9d00f8477fd5282 / public-subnet-02

Cancel Save associations

4. create another route table as PRIVATE.

5. Select VPC – do edit subnet associations – select 4 private subnets – save associations.

6. snapshots are attached below.

aws Services Search [Alt+S] N. Virginia usha @ 0109-2818-5144

VPC > Route tables > Create route table

### Create route table

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

**Route table settings**

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

VPC  
The VPC to use for this route table.  
vpc-0d06022cba87624ef (my-vpc-01)

Screenshot of the AWS VPC dashboard showing the Route tables section. A route table named 'rtb-06b464d48b1dc1188' is selected. The 'Actions' menu is open, showing options like 'Set main route table', 'Edit subnet associations', and 'Delete'. The 'Details' tab shows the route table's ID, VPC, and owner.

Screenshot of the 'Edit subnet associations' page for the selected route table. It lists available subnets and selected subnets. The 'Selected subnets' section contains three subnets: 'subnet-0507ce4548af0cf24 / private-subnet-01', 'subnet-0021fbe703d99afcd / private-subnet-02', and 'subnet-0cf588f8fc0b6921b / private-subnet-rds-01'. A 'Save associations' button is at the bottom right.

Screenshot of the 'Route tables' page showing a list of four route tables. The 'private-route-table' is selected, indicated by a blue border. The 'Actions' menu for this table is open, showing options like 'Edit routes', 'Attach internet gateway', and 'Attach NAT gateway'.

7.click on public route table – edit routes – add rules- attach internet gateway – save changes.

8. for private route table – attach internet gateway & Nat gateway – save changes.

The screenshot shows the AWS VPC Route Tables interface. It displays two route entries:

- Destination: 10.0.0.0/16, Target: local, Status: Active, Propagated: No.
- Destination: 0.0.0.0/0, Target: Internet Gateway (igw-0a0252ec9af543a68), Status: Active, Propagated: No.

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

The screenshot shows the AWS VPC Route Tables interface. It displays three route entries:

- Destination: 10.0.0.0/16, Target: local, Status: Active, Propagated: No.
- Destination: 0.0.0.0/8, Target: Internet Gateway (igw-0a0252ec9af543a68), Status: Active, Propagated: No.
- Destination: 0.0.0.0/0, Target: NAT Gateway (nat-057c6da972383543b), Status: Active, Propagated: No.

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

9. Now go to the subnets - click on public subnet -01, click on actions – edit subnet settings – ENABLE Auto assign public IPv4 address.

10. Do the same for remaining subnets also.

The screenshot shows the AWS Subnets settings interface for the public-subnet-01. The "Actions" menu is open, with "Edit subnet settings" selected. The "Edit subnet settings" page shows the following configuration:

- Subnet**: Subnet ID: subnet-0fb30c664548ed663, Name: public-subnet-01.
- Auto-assign IP settings**: Info: Enable AWS to automatically assign a public IPv4 or IPv6 address to a new primary network interface for an instance in this subnet.
  - Enable auto-assign public IPv4 address
  - Enable auto-assign customer-owned IPv4 address

## ❖ Create SECURITY GROUPS:

1.we have to create TWO security groups.

2. Goto security groups - click on create security groups- select VPC – add INBOUND RULE (SSH & HTTP) & OUTBOUND RULE (All traffic) – click on create security group.

3. snapshots are attached below.

The screenshot shows the 'Create security group' page in the AWS VPC console. In the 'Basic details' section, the 'Security group name' field contains 'my-sg-01'. The 'Description' field contains 'allow'. The 'VPC Info' dropdown is set to 'vpc-0d06022cba87624ef (my-vpc-01)'.

The screenshot shows the 'Inbound rules' section of the 'Create security group' page. It lists two rules: one for SSH (TCP port 22) and one for HTTP (TCP port 80). Both rules have 'Anywhere...' as the source and '0.0.0.0/0' as the destination. There is a 'Delete' button next to each rule entry.

The screenshot shows the 'Outbound rules' section of the 'Create security group' page. It lists a single rule for 'All traffic' with 'All' as both the protocol and port range, and 'Custom' as the destination. There is a 'Delete' button next to the rule entry.

The screenshot shows the AWS VPC Info page for a VPC named "vpc-0d06022cba87624ef (my-vpc-01)". Under the "Inbound rules" section, there are two entries:

- SSH**: Protocol TCP, Port range 22, Source Anywhere..., Destination 0.0.0.0/0, Description optional.
- HTTP**: Protocol TCP, Port range 80, Source Anywhere..., Destination 0.0.0.0/0, Description optional.

At the bottom left, there is a "Add rule" button.

The screenshot shows the AWS VPC Info page under the "Outbound rules" section. It displays a single entry for "All traffic" with the following details:

- Type: All traffic
- Protocol: All
- Port range: All
- Destination: Custom, 0.0.0.0/0
- Description optional: None

At the bottom left, there is a "Add rule" button.

The screenshot shows the AWS VPC dashboard under the "Security Groups" section. It lists four security groups:

Name	Security group ID	Security group name	VPC ID	Description
-	sg-0b2619763f0cd5f7	default	vpc-09ccdf650406d26621	default VPC security group
-	sg-055bd315193a365f1	my-sg-01	vpc-0d06022cba87624ef	allow
-	sg-0d5b9a0b92bfc98ee	default	vpc-0d06022cba87624ef	default VPC security group
-	sg-09c632cb1e1be7b2b	my-sg-02	vpc-0d06022cba87624ef	allow

## ❖ NOW LAUNCH TWO TEMPLATES (Public & Private):

### ➤ PUBLIC TEMPLATE:

1. Search EC2 – Click on LAUNCH TEMPLATES – Click on CREATE LAUNCH TEMPLATES.
2. Select AMI – UBUNTU & instance type – t2.micro(1GB- Free Tier).
3. Select KEY PAIR – a new or existing.
4. In Network Settings, I am not going to specify subnets, but security group (SG) that I am created for the security-group-1(my-sg-01) is selected. Make sure the proper VPC is selected.
5. Snapshots of Public Template are attached below.

**Create launch template**

Creating a launch template allows you to create a saved instance configuration that can be reused, shared and launched at a later time. Templates can have multiple versions.

**Launch template name and description**

Launch template name - required  
my-public-template  
Must be unique to this account. Max 128 chars. No spaces or special characters like '&', '<>', '@'.

Template version description  
allow  
Max 255 chars

Auto Scaling guidance - Info  
Select this if you intend to use this template with EC2 Auto Scaling  
 Provide guidance to help me set up a template that I can use with EC2 Auto Scaling

► Template tags  
► Source template

**Summary**

Software Image (AMI)  
Canonical, Ubuntu, 24.04, amd6...read more  
ami-0e86e20daea9224db8

Virtual server type (instance type)  
t2.micro

Firewall (security group)  
my-sg-01

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

**Create launch template**

**Launch template contents**

Specify the details of your launch template below. Leaving a field blank will result in the field not being included in the launch template.

**Application and OS Images (Amazon Machine Image)**

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.

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

Recents Quick Start

Don't include in launch template Amazon Linux macOS Ubuntu Windows Red Hat AWS Mac

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

**Description**

Ubuntu Server 24.04 LTS (HVM), SSD Volume Type

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

Architecture AMI ID  
64-bit (x86) ami-0e86e20daea9224db8 Verified provider

**Summary**

Software Image (AMI)  
Canonical, Ubuntu, 24.04, amd6...read more  
ami-0e86e20daea9224db8

Virtual server type (instance type)  
t2.micro

Firewall (security group)  
my-sg-01

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

**Create launch template**

**Instance type**

Instance type  
t2.micro  
CPU: 1 vCPU | 1 GiB Memory | Current generation: true  
On-Demand Windows base pricing: \$0.162 USD per Hour  
On-Demand SUSE base pricing: \$0.116 USD per Hour  
On-Demand RHEL base pricing: \$0.226 USD per Hour  
On-Demand Linux base pricing: \$0.116 USD per Hour

Additional costs apply for AMIs with pre-installed software

**Key pair (login)**

You can use a key pair to securely connect to your instance. Ensure that you have access to the selected key pair before you launch the instance.

Key pair name  
asg-1 Create new key pair

**Summary**

Software Image (AMI)  
Canonical, Ubuntu, 24.04, amd6...read more  
ami-0e86e20daea9224db8

Virtual server type (instance type)  
t2.micro

Firewall (security group)  
my-sg-01

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

**Create launch template**

**Network settings**

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

my-sg-01 sg-055bd315193a365f1 X  
VPC: vpc-0d060222ca87624ef

Advanced network configuration

**Summary**

Software Image (AMI)  
Canonical, Ubuntu, 24.04, amd6...read more  
ami-0e86e20daea9224db8

Virtual server type (instance type)  
t2.micro

Firewall (security group)  
my-sg-01

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

**Create launch template**

## ➤ Create PRIVATE TEMPLATE:

- 1.Create same as previous template, but at SECURITY GROUP select SECURITY GROUP -2 (my-sg-02).
- 2.Snapshots of PRIVATE TEMPLATE as attached below.

**Create launch template**

Creating a launch template allows you to create a saved instance configuration that can be reused, shared and launched at a later time. Templates can have multiple versions.

**Launch template name and description**

Launch template name - required  
my-private-template  
Must be unique to this account. Max 128 chars. No spaces or special characters like %, ^, @.

Template version description  
allow  
Max 255 chars

Auto Scaling guidance [Info](#)  
Select this if you intend to use this template with EC2 Auto Scaling  
 Provide guidance to help me set up a template that I can use with EC2 Auto Scaling

**Launch template contents**

Specify the details of your launch template below. Leaving a field blank will result in the field not being included in the launch template.

**Application and OS Images (Amazon Machine Image) [Info](#)**

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

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

Recent [Quick Start](#)

Don't include in launch template [Amazon Linux](#) [macOS](#) [Ubuntu](#) [Windows](#) [Red Hat](#) [Browse more AMIs](#) Including AMIs from AWS, Marketplace and the Community

Amazon Machine Image (AMI)

Ubuntu Server 24.04 LTS (HVM), SSD Volume Type  
ami-0e86e20da9e9224db8  
Free tier eligible

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-0e86e20da9e9224db8 [Verified provider](#)

**Instance type [Info](#) | [Get advice](#)**

Advanced

Instance type t2.micro  
Family: t2 - 1 vCPU - 1 GiB Memory. Current generation: true  
On-Demand Windows base pricing: 0.0162 USD per Hour  
On-Demand SUSE base pricing: 0.0116 USD per Hour  
On-Demand RHEL base pricing: 0.025 USD per Hour  
On-Demand Linux base pricing: 0.0116 USD per Hour  
Additional costs apply for AMIs with pre-installed software

**Key pair (login) [Info](#)**

You can use a key pair to securely connect to your instance. Ensure that you have access to the selected key pair before you launch the instance.

Key pair name asg-02 [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 group my-sg-02 [Compare security group rules](#)

**Summary**

Software Image (AMI)  
Canonical, Ubuntu, 24.04, amd6...read more  
ami-0e86e20da9e9224db8

Virtual server type (instance type)  
t2.micro

Firewall (security group)  
my-sg-02

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

[Cancel](#) [Create launch template](#)

### 3.TWO TEMPLATES LAUNCHED SUCCESSFULLY.

Launch Templates (2) <small>Info</small>	
<input type="text"/> Search	
Launch Template ID	Launch Template Name
Default Version	Latest Version
Create Time	Created By
lt-0cb1bf60a4f6c736f	my-private-template
1	1
2024-08-27T05:29:49.000Z	arn:aws:iam::010928185144:user/usha
lt-046e07efb806113a7	my-public-template
1	1
2024-08-27T05:28:11.000Z	arn:aws:iam::010928185144:user/usha

## ❖ CREATE TWO AUTOSCALING GROUPS (Public & Private):

### ➤ PUBLIC ASG:

- 1.In EC2, go to autoscaling group – click on create autoscaling group.
- 2.give name to ASG – Select PUBLIC TEMPLATE (which is already created)
- 3.In network settings- choose VPC – choose 2 public subnets.
4. After that click on NEXT.

Choose launch template Info  
Specify a launch template that contains settings common to all EC2 instances that are launched by this Auto Scaling group.

Name  
Auto Scaling group name  
Enter a name to identify the group.  
  
Must be unique to this account in the current Region and no more than 255 characters.

Launch template Info  
For accounts created after May 31, 2023, the EC2 console only supports creating Auto Scaling groups with launch templates. Creating Auto Scaling groups with launch configurations is not recommended but still available via the CLI and API until December 31, 2023.

Launch template  
Choose a launch template that contains the instance-level settings, such as the Amazon Machine Image (AMI), instance type, key pair, and security groups.

Version

Description  
allow  
Launch template  
my-public-template    
lt-046e07efb806113a7

Instance type  
t2.micro

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.

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

5. We have to attach LOAD BALANCER to ASG.

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

## 8. Give HEALTH CHECK GRACE PERIOD as your wish.

Configure advanced options - *optional* [Info](#)  
Integrate your Auto Scaling group with other services to distribute network traffic across multiple servers using a load balancer or to establish service-to-service communications using VPC Lattice. You can also set options that give you more control over health check replacements and monitoring.

**Load balancing** [Info](#)  
Use the options below to attach your Auto Scaling group to an existing load balancer, or to a new load balancer that you define.

No load balancer  
Traffic to your Auto Scaling group will not be fronted by a load balancer.

Attach to an existing load balancer  
Choose from your existing load balancers.

Attach to a new load balancer  
Quickly create a basic load balancer to attach to your Auto Scaling group.

**Attach to a new load balancer**  
Define a new load balancer to create for attachment to this Auto Scaling group.

**Load balancer type**  
Choose from the load balancer types offered below. Type selection cannot be changed after the load balancer is created. If you need a different type of load balancer than those offered here, visit the [Load Balancing console](#).

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

Add tags

Step 7 [Review](#)

**Attach to a new load balancer**  
Define a new load balancer to create for attachment to this Auto Scaling group.

**Load balancer type**  
Choose from the load balancer types offered below. Type selection cannot be changed after the load balancer is created. If you need a different type of load balancer than those offered here, visit the [Load Balancing console](#).

Application Load Balancer  
HTTP, HTTPS

Network Load Balancer  
TCP, UDP, TLS

**Load balancer name**  
Name cannot be changed after the load balancer is created.

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

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**  
 my-vpc-01

**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

us-east-1b

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

Protocol	Port	Default routing (forward to)
HTTP	80	auto-scaling-public-1-tg   HTTP

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

**Health checks**  
Health checks increase availability by replacing unhealthy instances. When you use multiple health checks, all are evaluated, and if at least one fails, instance replacement occurs.

**EC2 health checks**  
 Always enabled

**Additional health check types - optional** [Info](#)

Turn on Elastic Load Balancing health checks [Info](#) [Recommended](#)  
Elastic Load Balancing monitors whether instances are available to handle requests. When it reports an unhealthy instance, EC2 Auto Scaling can replace it on its next periodic check.

Turn on VPC Lattice health checks  
VPC Lattice can monitor whether instances are available to handle requests. If it considers a target as failed a health check, EC2 Auto Scaling replaces it after its next periodic check.

Turn on Amazon EBS health checks  
EBS monitors whether an instance's root volume or attached volume stalls. When it reports an unhealthy volume, EC2 Auto Scaling can replace the instance on its next periodic health check.

**Health check grace period** [Info](#)  
This time period delays the first health check until your instances finish initializing. It doesn't prevent an instance from terminating when placed into a non-running state.

seconds

## 9. Select Group size

We want to set a minimum and maximum number of instances the ASG can provision:

- Desired capacity: 2

- **Minimum capacity: 2**
- **Maximum capacity: 5**

10. After that, click on next – next – create auto scaling group.

The image consists of three vertically stacked screenshots of the AWS Auto Scaling 'Create Auto Scaling group' wizard, showing the progression from Step 3 to Step 6.

**Screenshot 1: Step 3 - optional Configure advanced options**

- Group size info:** Set the initial size of the Auto Scaling group. Desired capacity is set to 2.
- Scaling limits:** Min desired capacity is set to 2, and Max desired capacity is set to 5.
- Automatic scaling - optional:**
  - No scaling policies:** Your Auto Scaling group will remain at its initial size and will not dynamically resize to meet demand.
  - 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.

**Screenshot 2: Step 5 - optional Add notifications**

- Add notifications - optional:** Send notifications to SNS topics whenever Amazon EC2 Auto Scaling launches or terminates the EC2 instances in your Auto Scaling group.
- Add notification:** A button to add a new notification.

**Screenshot 3: Step 6 - optional Add tags**

- Add tags - optional:** Add tags to help search, filter, and track your Auto Scaling group across AWS. A note says tags from the launch template will be overridden if there are any duplicate keys.
- Tags (0):** A table to add tags, with a single row for 'Add tag' and '50 remaining'.

Each screenshot shows the 'Next' button at the bottom right, indicating the progression through the wizard steps.

The screenshot shows the AWS Auto Scaling group creation wizard at Step 5: Add notifications. It includes sections for Target tracking policy, Instance maintenance policy, and Instance scale-in protection. Step 5: Add notifications is currently active, showing a 'Notifications' section with 'No notifications'. Step 6: Add tags is shown below, with a 'Tags (0)' section and a table for adding tags. At the bottom are 'Cancel', 'Previous', and 'Create Auto Scaling group' buttons.

## ➤ CREATE ANOTHER ASG AS PRIVATE:

- 1.follow all the steps as above.
2. But at network settings, choose 4 private subnets.
- 3.snapshots of private ASG are attached below.

The screenshot shows the AWS Auto Scaling group creation wizard at Step 6: Network. The 'Network' section is expanded, showing the 'Launch template' field set to 'my-private-template'. The 'VPC' field shows 'vpc-006022cba87624ef (my-vpc-01)'. The 'Availability Zones and subnets' section lists four private subnets across three availability zones: us-east-1a, us-east-1b, and us-east-1c. The 'Next' button is visible at the bottom.

**AWS Services Search [Alt+S]**

**Review**

**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-0d6022cba87624ef

my-vpc-01

**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

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

**Health checks**  
Health checks increase availability by replacing unhealthy instances. When you use multiple health checks, all are evaluated, and if at least one fails, instance replacement occurs.

**EC2 health checks**  
 Always enabled

**Additional health check types - optional** [Info](#)

- Turn on Elastic Load Balancing health checks [Recommended](#)  
Elastic Load Balancing monitors whether instances are available to handle requests. When it reports an unhealthy instance, EC2 Auto Scaling can replace it on its next periodic check.
- Turn on VPC Lattice health checks  
VPC Lattice can monitor whether instances are available to handle requests. If it considers a target as failed a health check, EC2 Auto Scaling can replace it on its next periodic check.
- Turn on Amazon EBS health checks  
EBS monitors whether an instance's root volume or attached volume stalls. When it reports an unhealthy volume, EC2 Auto Scaling can replace the instance on its next periodic health check.

**Health check grace period** [Info](#)  
This time period delays the first health check until your instances finish initializing. It doesn't prevent an instance from terminating when placed into a non-running state.  
 seconds

**Choose instance launch options**

**Step 3 - optional** [Configure advanced options](#)

**Step 4 - optional** [Configure group size and scaling](#)

**Step 5 - optional** [Add notifications](#)

**Step 6 - optional** [Add tags](#)

**Step 7** [Review](#)

#### 4. Successfully created TWO AUTO SCALING GROUPS.

**AWS Services Search [Alt+S]**

**auto-scaling-private, 1 Load balancer, 1 Target group, 1 Listener created successfully. 1 new target group has been attached to ASG.**

**EC2 > Auto Scaling groups**

**Auto Scaling groups (2) [Info](#)**

**Search your Auto Scaling groups**

Name	Launch template/configuration	Instances	Status	Desired capacity	Min	Max	Availability Zones
auto-scaling-private	my-private-template   Version Default	0	Updating capacity...	2	2	5	us-east-1a, us-east-1b
auto-scaling-public	my-public-template   Version Default	2	-	2	2	5	us-east-1a, us-east-1b

**Actions** [Create Auto Scaling group](#)

5. Now go to EC2 dashboard- click on instances.

6. Here we can see the 4 FOUR NEW RUNNING INSTANCES.

7. Give the names to those instances as – public-1, public-2, private-1&private-2.

The screenshot shows the AWS CloudWatch Metrics Insights interface. At the top, there's a search bar and a 'Launch instances' button. The main area displays a table of instance details, including Name, Instance ID, Instance state, Instance type, Status check, Alarm status, Availability Zone, Public IPv4 DNS, Public IPv4 IP, and Elastic IP. The table lists four instances: private-1, public-1, public-2, and private-2, all in a 'Running' state.

Name	Instance ID	Instance state	Instance type	Status check	Alarm status	Availability Zone	Public IPv4 DNS	Public IPv4 ...	Elastic IP
private-1	i-0cd688c06212d0497	Running	t2.micro	2/2 checks passed	View alarms +	us-west-1b	-	15.52.182.153	-
public-1	i-0acd83bf3f511fc2	Running	t2.micro	2/2 checks passed	View alarms +	us-west-1b	-	18.144.27.219	-
public-2	i-09c3ff556bb448976	Running	t2.micro	2/2 checks passed	View alarms +	us-west-1c	-	54.153.85.221	-
private-2	i-0985595345624623b	Running	t2.micro	2/2 checks passed	View alarms +	us-west-1c	-	54.219.251.72	-

8. Click on public -1 instance – click on connect- connect to the EC2 server.

9. After connecting to the UBUNTU, give the following commands:

- Sudo -l --> to become a root user
  - apt update -y --> to update packages
  - apt install apache2 --> to install apache2
  - cd /var/www/html --> path
  - ls --> list
  - rm index.html --> to remove index.html
  - vi index.html --> enter --> press I --> insert date -->:x (to save) – enter
  - systemctl status apache2 --> to check the server status.
  - Ping google.com --> to check whether the server is ping or not.

10.screenshots attached below.

```
aws Services Q Search [Alt+5] N. Virginia usha @ 0109-2815-14

Welcome to Ubuntu 24.04 LTS (GNU/Linux 6.8.0-1012-aws x86_64)

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

System information as of Tue Aug 27 06:22:29 UTC 2024

System load: 0.08      Processes:          102
Usage of /: 22.7% of 6.71GB   Users logged in:    0
Memory usage: 20%        IPv4 address for enx0: 10.0.1.84
Swap usage: 0%           IPv6 address for enx0: fe80::410:1ff:fe0:1084%1

Expanded Security Maintenance for Applications is not enabled.

0 updates can be applied immediately.

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

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

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

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

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

ubuntutip-10-0-1-84:~$ sudo -i
root@ubuntutip-10-0-1-84:~# apt update
Get:2 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble InRelease [126 kB]
Get:3 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble-backports InRelease [126 kB]
Get:4 http://security.ubuntu.com/ubuntu noble-security InRelease [126 kB]
Get:5 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble/universe amd64 Packages [15.0 MB]
Get:6 http://security.ubuntu.com/ubuntu noble-security/main amd64 Packages [323 kB]
No user sessions are running outdated binaries.

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

root@ubuntutip-10-0-1-84:~# cd /var/www/html
root@ubuntutip-10-0-1-84:/var/www/html# ls
index.html
root@ubuntutip-10-0-1-84:/var/www/html# rm index.html
root@ubuntutip-10-0-1-84:/var/www/html# index.html
root@ubuntutip-10-0-1-84:/var/www/html# systemctl status apache2
● apache2.service - The Apache HTTP Server
   Loaded: loaded (/usr/lib/systemd/system/apache2.service; enabled; preset: enabled)
   Active: active (running) since Tue 2024-08-27 06:23:43 UTC; 2min 17s ago
     Docs: https://httpd.apache.org/docs/2.4/
Main PID: 2535 (apache2)
   Tasks: 1 (limit: 1100)
   Memory: 6.9M (peak: 7.0M)
      CPU: 55ms
 CGroup: /system.slice/apache2.service
         ├─2535 /usr/sbin/apache2 -k start
         ├─2538 /usr/sbin/apache2 -k start
         └─2539 /usr/sbin/apache2 -k start

Aug 27 06:23:43 ip-10-0-1-84 systemd[1]: Starting apache2.service - The Apache HTTP Server...
Aug 27 06:23:43 ip-10-0-1-84 apache2[2534]: AH00558: apache2: Could not reliably determine the server's fully qualified domain name, using 10.0.1.84. Set the 'ServerName' directive globally to suppress this message
Aug 27 06:23:43 ip-10-0-1-84 apache2[1]: Started apache2.service - The Apache HTTP Server.
root@ubuntutip-10-0-1-84:~# curl -I https://www.google.com
HTTP/2 200 OK
Date: Tue, 27 Aug 2024 06:23:43 GMT
Content-Type: text/html; charset=UTF-8
Content-Length: 1452
Connection: keep-alive
Set-Cookie: __utma=100000000.14525179102; __utmb=100000000.14525179102; __utmc=100000000; __utmz=100000000.14525179102.1.1.utmcsr=(not set); __utmt=1
-- google.com ping statistics --
4 packets transmitted, 4 received, 0% packet loss, time 3006ms
rtt min/avg/max/mdev = 2.768/2.810/2.848/0.028 ms
root@ubuntutip-10-0-1-84:/var/www/html#
```

11.Copy the PUBLIC IP address & paste it in google chrome.

12.It will show the data which we inserted.



this is ec1 server.

## ➤ NOW CONNECT TO THE PRIVATE INSTANCE THROUGH THE PUBLIC INSTANCE:

1.Follow the steps as mentioned in the snapshots.

```
Welcome to Ubuntu 24.04 LTS (GNU/Linux 6.8.0-1012-aws x86_64)

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

System information as of Tue Aug 27 06:22:29 UTC 2024

System load: 0.08      Processes:          102
Usage of /: 22.7% of 6.71GB  Users logged in: 0
Memory usage: 20%          IPv4 address for enX0: 10.0.1.84
Swap usage: 0%          

Expanded Security Maintenance for Applications is not enabled.
0 updates can be applied immediately.

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

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

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

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

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

root@ip-10-0-1-84:~# sudo -i
root@ip-10-0-1-84:~# apt update -y
Hit:1 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble InRelease [126 kB]
Get:2 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble-backports InRelease [126 kB]
Get:3 http://security.ubuntu.com/ubuntu noble-security InRelease [126 kB]
Get:4 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble/universe amd64 Packages [15.0 MB]
Get:5 http://security.ubuntu.com/ubuntu noble-security/main amd64 Packages [323 kB]
```

```
-- google.com ping statistics --
3 packets transmitted, 3 received, 0% packet loss, time 2003ms
rtt min/avg/max/mdev = 1.900/1.920/1.959/0.027 ms
root@ip-10-0-1-84:~# cd /var/www/html
root@ip-10-0-1-84:/var/www/html# ls
asg-1.pem index.html
root@ip-10-0-1-84:/var/www/html# rm index.html
root@ip-10-0-1-84:/var/www/html# vi asg-1.pem
root@ip-10-0-1-84:/var/www/html# systemctl restart apache2
root@ip-10-0-1-84:/var/www/html# chmod 400 "asg-1.pem"
root@ip-10-0-1-84:/var/www/html# ssh -i "asg-1.pem" ubuntu@10.0.5.175
The authenticity of host '10.0.5.175 (10.0.5.175)' can't be established.
ED25519 key fingerprint is SHA256:F/qYKUX+ZJxKKUDldTicaR/A8tj/juKeVQgW4dfizV6.
This key is not known by any other names.
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added '10.0.5.175' (ED25519) to the list of known hosts.
Welcome to Ubuntu 24.04 LTS (GNU/Linux 6.8.0-1012-aws x86_64)

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

System information as of Tue Aug 27 09:28:54 UTC 2024

System load: 0.0      Processes:          105
Usage of /: 22.7% of 6.71GB  Users logged in: 0
Memory usage: 20%          IPv4 address for enX0: 10.0.5.175
Swap usage: 0%
```

## 2.Successfully connected to the PRIVATE INSTANCE.

```
aws Services Search [Alt+S]
gp25519 key fingerprint is SHA256:F/OyRQX+ZJxkRUdLdTcaRA8tj/juKeVQwMdfi2VE.
This key is not known by any other names.
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added '10.0.5.175' (ED25519) to the list of known hosts.
Welcome to Ubuntu 24.04 LTS (GNU/Linux 6.8.0-1012-aws x86_64)

System information as of Tue Aug 27 09:28:54 UTC 2024

System load: 0.0 Processes: 105
Usage of /: 22.7% of 6.71GB Users logged in: 0
Memory usage: 20% IPv4 address for enx0: 10.0.5.175
Swap usage: 0%

Expanded Security Maintenance for Applications is not enabled.

0 updates can be applied immediately.

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

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

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

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

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

ubuntu@ip-10-0-5-175:~$
```

### ➤ We have to add MYSQL/Aurora to the SECURITY GROUPS:

- For that, go to EC2 instance.
- Click on security groups – click on actions – edit inbound rules
- Add rule – select MYSQL/Aurora – click on save changes.

Inbound rules info										
Security group rule ID	Type	Info	Protocol	Info	Port range	Info	Source	Info	Description - optional	Info
sgr-07894d36a962258ea	SSH		TCP		22		Custom		Q	
sgr-0807332be9345b7b5	HTTP		TCP		80		Custom		Q	
-	MySQL/Aurora		TCP		3306		Anywhere-IP		Q, 0.0.0.0/0	

Add rule Cancel Preview changes Save rules

### ❖ RDS:

- First create DB-SUBNET group in RDS.

### ➤ Create DB-SUBNET group:

- 1.search RDS – click on subnet groups – click on create DB-SUBNET group.
- 2.give name as DB-SUBNET – choose VPC (own VPC) – choose availability zones (1a&1b)

### 3. Select TWO PRIVATE SUBNETS and then click on create.

Subnet group details

Name: DB-subnet

Description: allow

VPC: my-vpc-01 (vpc-0d06022cba87624ef)

Add subnets

Availability Zones: us-east-1a, us-east-1b

Subnets: subnet-0dab6a45712cb2338 (10.0.6.0/24), subnet-0cf58ff8fc0b6921b (10.0.5.0/24)

Subnets selected (2)

Availability zone	Subnet ID	CIDR block
us-east-1b	subnet-0dab6a45712cb2338	10.0.6.0/24
us-east-1a	subnet-0cf58ff8fc0b6921b	10.0.5.0/24

Create

### 4. Successfully created DB-subnet groups.

Successfully created DB-subnet. View subnet group

Name	Description	Status	VPC
db-subnet	allow	Complete	vpc-0d06022cba87624ef
default-vpc-0d06022cba87624ef	Created from the RDS Management Console	Complete	vpc-0d06022cba87624ef

## ❖ CREATE DATABASE:

1. Click on DATABASES – click on create database.
2. Choose STANDARD in creation method and MYSQL in engine options.
3. Select FREE TIER in templates.
4. Choose SELF MANAGED in settings and give PASSWORD.
5. Go to connectivity and choose CONNECT TO EC2 INSTANCE and select public instance.
6. We don't have to give VPC and DB-SUBNET group it will take by default.
7. choose existing security groups and select both public (public-sg) and private (private-sg) security groups.
8. Click on create database.

## 9. Snapshots of database are attached below.

**Create database**

**Choose a database creation method**

- Standard create
 

You set all of the configuration options, including ones for availability, security, backups, and maintenance.
- Easy create
 

Use recommended best-practice configurations. Some configuration options can be changed after the database is created.

**Engine options**

**Engine type**

- Aurora (MySQL Compatible)
- Aurora (PostgreSQL Compatible)
- MySQL
- MariaDB
- PostgreSQL
- Oracle

**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.

**Templates**

Choose a sample template to meet your use case.

- Production
 

Use defaults for high availability and fast, consistent performance.
- Dev/Test
 

This instance is intended for development use outside of a production environment.
- Free tier
 

Use RDS Free Tier to develop new applications, test existing applications, or gain hands-on experience with Amazon RDS.

**Availability and durability**

**Deployment options**

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

- Multi-AZ DB Cluster
 

Creates a DB cluster with a primary DB instance and two read-only standby DB instances, with each DB instance in a different Availability Zone (AZ). Provides high availability, data redundancy and increases capacity to serve read workloads.
- Multi-AZ DB instance [not supported for Multi-AZ DB cluster snapshot]
 

Creates a primary DB instance and a standby DB instance in a different AZ. Provides high availability and data redundancy, but the standby DB instance doesn't support connections for read workloads.
- Single DB instance [not supported for Multi-AZ DB cluster snapshot]
 

Creates a single DB instance with no standby DB instances.

**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.

**Settings**

**DB instance identifier**

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

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

**Credentials Settings**

**Master username**

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

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**

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

**Password strength** Weak

**Confirm master password**

.....

**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.

**Connectivity** [Info](#)

**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.

**EC2 instance** [Info](#)  
Choose the EC2 instance to add as the compute resource for this database. A VPC security group is added to this EC2 instance. A VPC security group is also added to the database with an inbound rule that allows the EC2 instance to access the database.

i-0164fa7f561acff6  
ec1-public

**Some VPC settings can't be changed when a compute resource is added**  
Adding an EC2 compute resource automatically selects the VPC, DB subnet group, and public access settings for this database. To allow the EC2 instance to access the database, a VPC security group rds-ec2-X is added to the database and another called ec2-rds-X to the EC2 instance. You can remove the new security group for the database only by removing the compute resource.

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

my-vpc-01 (vpc-0006022cba87624ef)  
6 Subnets, 2 Availability Zones

Only VPCs with a corresponding DB subnet group are listed.

**After a database is created, you can't change its VPC.**

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

**Choose existing**  
Choose existing DB subnet group

**Automatic setup**  
RDS creates a new subnet group for you or reuses an existing subnet group

**Existing DB subnet groups**  
default-vpc-0d06022cba87624ef  
6 Subnets, 2 Availability Zones

**Public access** [Info](#)  
Choose Yes or No. RDS assigns a public IP address to the database. Amazon EC2 instances and other resources outside of the VPC can connect to your database. Resources inside the VPC can also connect to the database. Choose one or more VPC security groups that specify which resources can connect to the database.

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

**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

**Additional VPC security group**  
Choose one or more options  
my-sg-02 X my-sg-01 X

**Amazon RDS will add a new VPC security group rds-ec2-4 to allow connectivity with your compute resource.**

**Availability Zone** [Info](#)  
us-east-1a

**Additional configuration**  
Enhanced Monitoring

**Estimated Monthly costs**

DB instance	427.05 USD
Storage	150.00 USD
Provisioned IOPS	900.00 USD
<b>Total</b>	<b>1477.05 USD</b>

This billing estimate is based on on-demand usage as described in [Amazon RDS Pricing](#). Estimate does not include costs for backup storage, IOs (if applicable), or data transfer.

Estimate your monthly costs for the DB Instance using the [AWS Simple Monthly Calculator](#).

**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.

**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.

**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.

## 10.Successfully database is created.

The screenshot shows the AWS RDS Dashboard. On the left, there's a sidebar with options like Dashboard, Databases, Query Editor, Performance insights, Snapshots, Exports in Amazon S3, Automated backups, Reserved instances, and Proxies. The main area is titled 'Databases (1)' and shows a table with one row for 'database-1'. The table includes columns for DB identifier, Status, Role, Engine, Region & AZ, Size, Recommendations, CPU, Current activity, Maintenance, and VPC. The status is 'Available' and the engine is 'MySQL Community'.

## ❖ CREATE SNAPSHOT:

- 1.Go to RDS dashboard - select snapshot –click on take snapshot.
- 2.select DB-instance and give a new name to snapshot.
- 3.Click on take snapshot.

The screenshot shows the 'Take DB Snapshot' dialog. It has a 'Preferences' section with instructions to choose a database and name the snapshot. Under 'Snapshot type', 'DB instance' is selected. In the 'DB instance' field, 'database-1' is chosen. The 'Snapshot name' field contains 'DB-Snapshot'. At the bottom are 'Cancel' and 'Take snapshot' buttons.

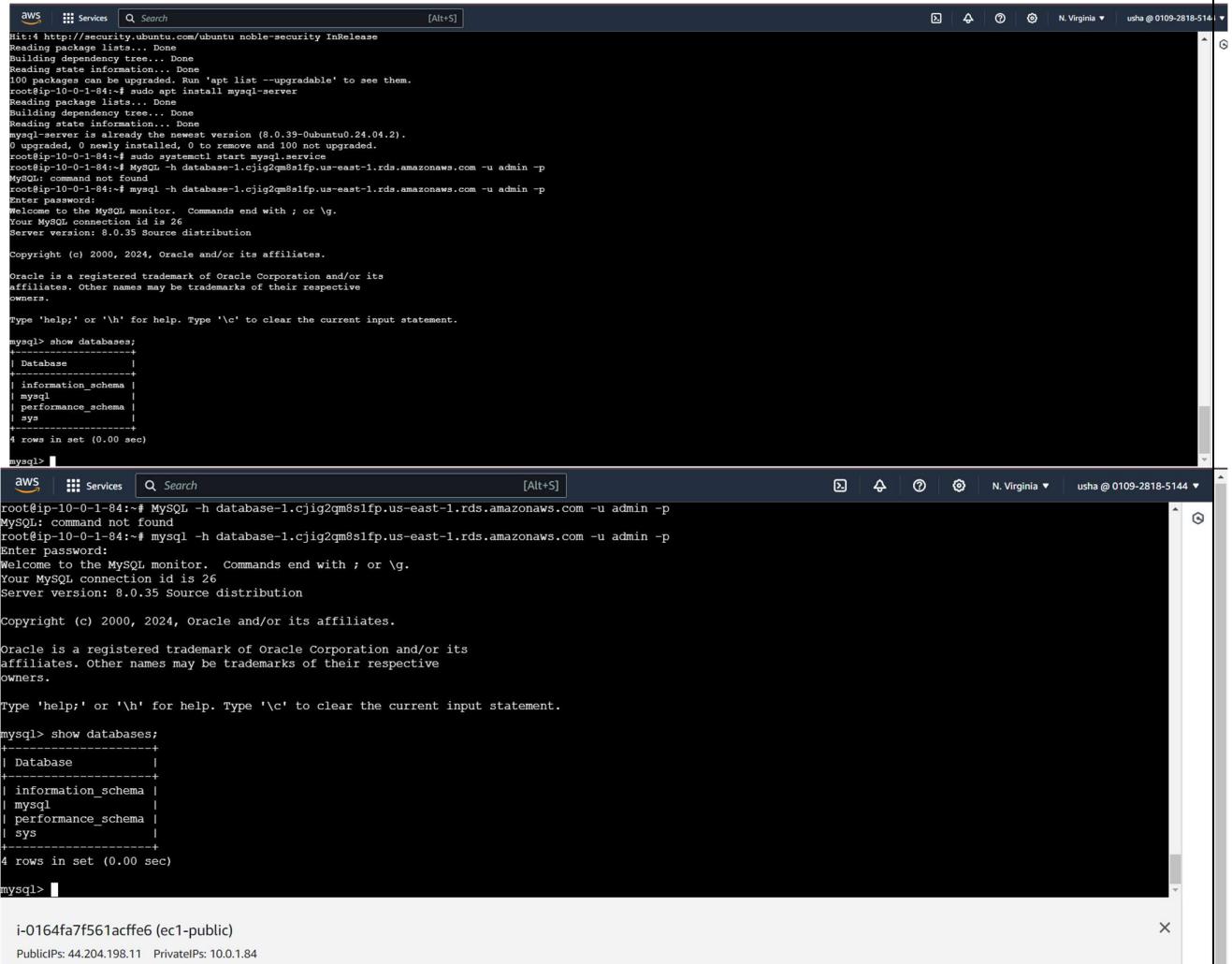
- 4.Successfully SNAPSHOT is created.

The screenshot shows the 'Schemas' section of the RDS dashboard. A green banner at the top says 'Successfully created snapshot DB-Snapshot. View details'. Below it, there's a table for 'Manual snapshots (1)'. The table has columns for Snapshot name, DB instance or cluster, Snapshot creation time, DB Instance created time, and Status. One entry is shown: 'db-snapshot' under 'Snapshot name', 'database-1' under 'DB instance or cluster', 'August 26, 2024, 11:41 (UTC+05:30)' under 'Snapshot creation time', 'August 26, 2024, 11:28 (UTC+05:30)' under 'DB Instance created time', and 'Available' under 'Status'.

## ❖ MYSQL INSTALLATION:

- 1.Go to public instance web and give the following commands.
  - Sudo apt update -y
  - Sudo apt install mysql – server -y
  - mysql -h database end point -u admin -p
  - END POINT - "database-1.cjig2qm8s1fp.us-east-1.rds.amazonaws.com"

2.once you complete giving the commands it displays as following in below snapshots.



The image contains two side-by-side screenshots of a terminal window from the AWS CloudWatch interface. Both screenshots show the same sequence of MySQL commands being run on a MySQL database instance.

```
N:4 http://security.ubuntu.com/ubuntu nobile-security InRelease  
Reading package lists... Done  
Building dependency tree... Done  
Reading state information... Done  
100 packages can be upgraded. Run 'apt list --upgradable' to see them.  
root@ip-10-0-1-84:~$ sudo apt install mysql-server  
Reading package lists... Done  
Building dependency tree... Done  
Reading state information... Done  
mysql-server is already the newest version (8.0.39-0ubuntu0.24.04.2).  
0 upgraded, 0 newly installed, 0 to remove and 100 not upgraded.  
root@ip-10-0-1-84:~$ sudo systemctl start mysql.service  
root@ip-10-0-1-84:~$ MySQL -h database-1.cjig2qm8slfp.us-east-1.rds.amazonaws.com -u admin -p  
MySQL: command not found  
root@ip-10-0-1-84:~$ mysql -h database-1.cjig2qm8slfp.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 26  
Server version: 8.0.35 Source distribution  
Copyright (c) 2000, 2024, Oracle and/or its affiliates.  
Oracle is a registered trademark of Oracle Corporation and/or its  
affiliates. Other names may be trademarks of their respective  
owners.  
Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.  
  
mysql> show databases;  
+-----+  
| Database |  
+-----+  
| information_schema |  
| mysql |  
| performance_schema |  
| sys |  
+-----+  
4 rows in set (0.00 sec)  
  
mysql>
```

```
root@ip-10-0-1-84:~# MySQL -h database-1.cjig2qm8slfp.us-east-1.rds.amazonaws.com -u admin -p  
MySQL: command not found  
root@ip-10-0-1-84:~# mysql -h database-1.cjig2qm8slfp.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 26  
Server version: 8.0.35 Source distribution  
Copyright (c) 2000, 2024, Oracle and/or its affiliates.  
Oracle is a registered trademark of Oracle Corporation and/or its  
affiliates. Other names may be trademarks of their respective  
owners.  
Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.  
  
mysql> show databases;  
+-----+  
| Database |  
+-----+  
| information_schema |  
| mysql |  
| performance_schema |  
| sys |  
+-----+  
4 rows in set (0.00 sec)  
  
mysql>
```

i-0164fa7f561acffe6 (ec1-public)  
PublicIPs: 44.204.198.11 PrivateIPs: 10.0.1.84

----THE END----