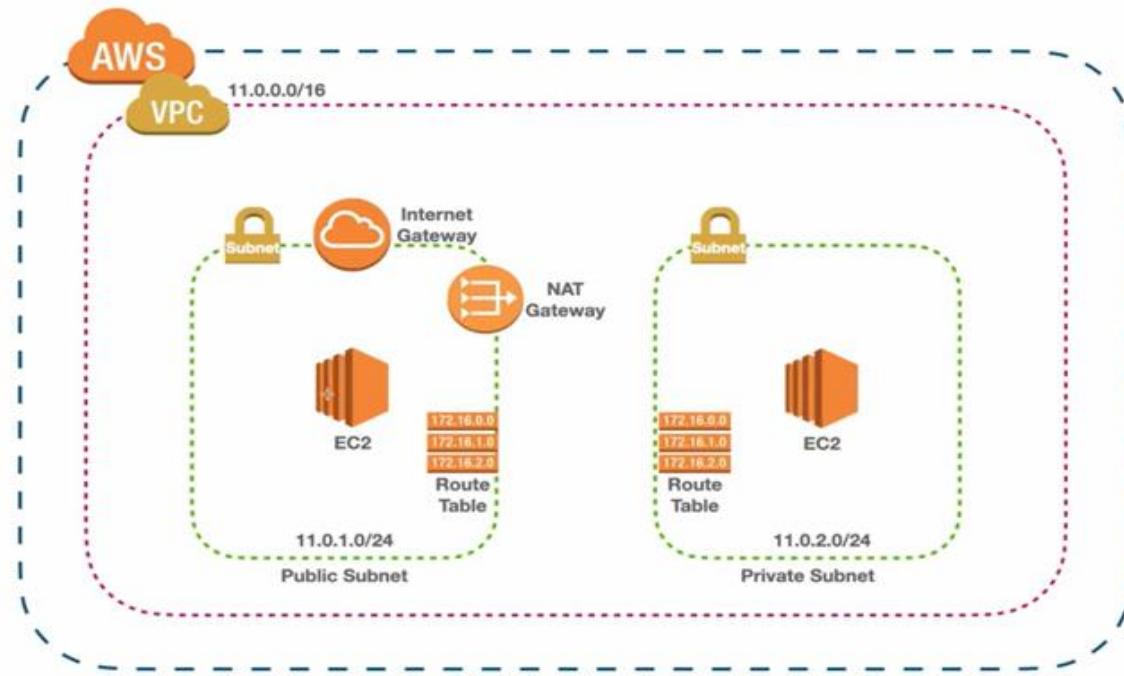


HOW TO SETUP EC2 INSTANCE IN VPC

AWS VPC (Amazon Virtual Private Cloud): It is a service that allows you to create a private network in the cloud. It gives you full access over your networking environment allowing you to define your IP address range also to create subnets and manage routing tables. With VPC you can securely connect your resources to the internet or keep them isolated from external traffic depending on your needs. It's like building your own private data center within AWS.



Internet Gateway: An Internet Gateway works by establishing a connection between a VPC and the internet. The VPC must have a public subnet, and the instances within that subnet must have a public IP address to communicate with the internet. An Internet Gateway acts as a bridge between the VPC and the internet. An Internet Gateway is commonly used when you want resources within a VPC to be accessible from the internet.

NAT Gateway: A NAT Gateway enables instances in a private subnet to connect to the internet or other AWS services but prevents the internet or other AWS services from initiating a connection with those instances. A NAT Gateway is commonly used when you have resources within a private subnet that require outbound internet access but should not be directly accessible from the internet.

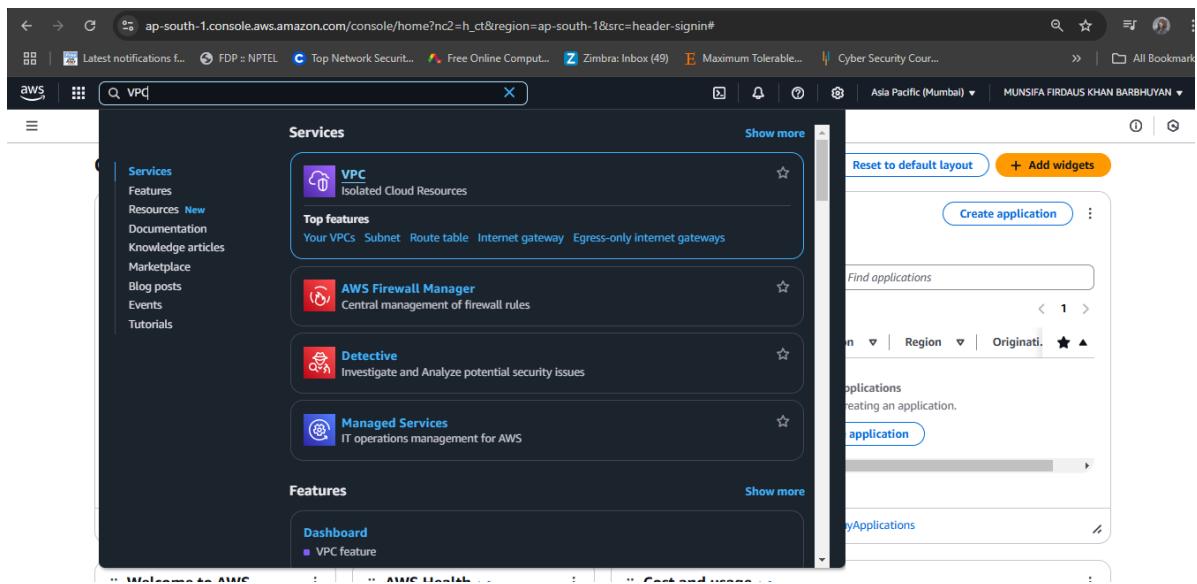
Subnet: A subnet is like a smaller group within a large network. It is a way to split a large network into smaller networks so that devices present in one network can transmit data more easily.

- The **public subnet** has instances with public IP addresses allowing them to communicate directly with the internet. These instances can be web servers load balancers or other publicly accessible services. An Internet Gateway connects the public subnet to the internet providing inbound and outbound communication.

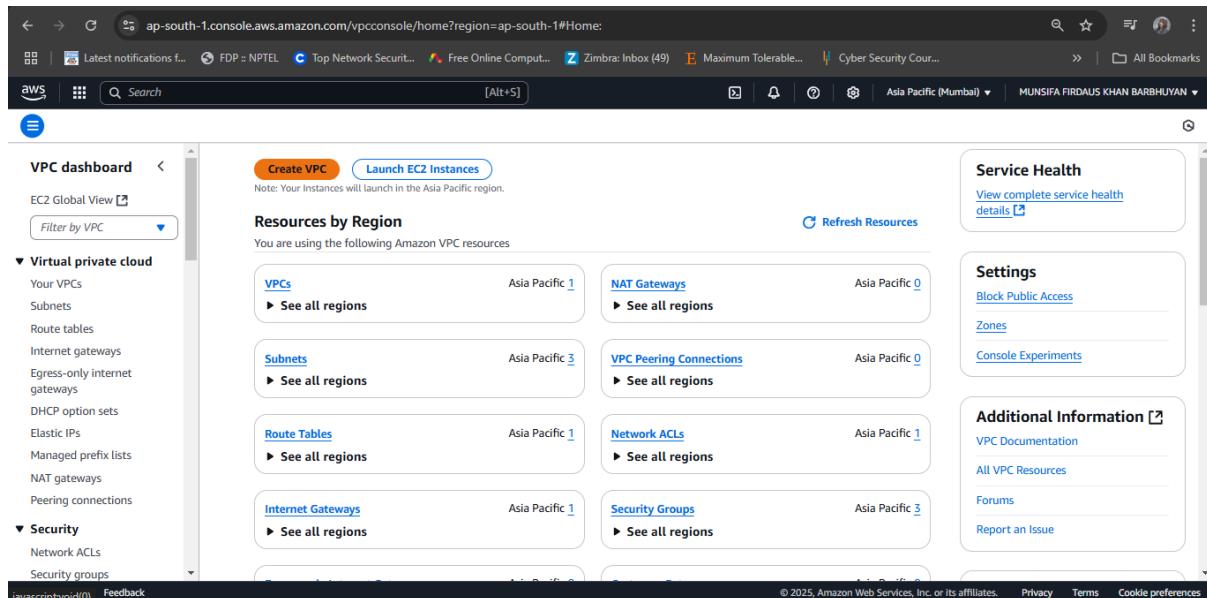
- The **private subnet** has instances with private IP addresses. These instances cannot directly access the internet but can connect to resources within the same VPC or other on-premises networks via a VPN or Direct Connect. Typically the instances in the private subnet access the internet via a NAT gateway or NAT instance which is usually placed in the public subnet.
- Public subnets have a route to the internet via an Internet Gateway. Resources in public subnets can be accessed from the internet such as web servers whereas Private subnets do not have a direct route to the internet it uses NAT device to get connected with Internet. Resources within private subnets cannot be accessed from outside the VPC unless specific configurations are made.

Route table: A route table contains a set of rules, called routes, that determine where network traffic from your subnet or gateway is directed. It is used to direct network traffic on a particular destination IP address. It allows customization of networks enabling them to connect to other subnets/gateways both within and outside of the VPC.

STEP1: LOGIN AS A ROOT USER AND SEARCH FOR VPC IN THE CONSOLE



STEP2: CLICK ON CREATE VPC



STEP3: SELECT VPC ONLY> TYPE THE VPC NAME> IPv4CIDR-11.0.0.0/16

NOTE: In AWS a VPC spans a specific IP address range using CIDR (Classless Inter-Domain Routing) blocks. The CIDR block defines the range of IP addresses that can be assigned to resources within the VPC

A VPC is an isolated portion of the AWS Cloud populated by AWS objects, such as Amazon EC2 instances.

VPC settings

Resources to create: [Info](#)
Create only the VPC resource or the VPC and other networking resources.

VPC only VPC and more

Name tag - optional:
Creates a tag with a key of 'Name' and a value that you specify.
mansifa-vpc

IPv4 CIDR block: [Info](#)
 IPv4 CIDR manual input IPAM-allocated IPv4 CIDR block
11.0.0.0/16
CIDR block size must be between /16 and /28.

IPv6 CIDR block: [Info](#)
 No IPv6 CIDR block IPAM-allocated IPv6 CIDR block
No IPv6 CIDR block available.

For better understanding of the CIDR range, Search CIDR Range Calculator in google and enter the range. Here its showing 65536 IP range, i.e. these are the maximum number of IP addresses which is accessible by the VPC. If we increase from /16 to /32, it will decrease the number of IP address and if we decrease it to /8, it will increase the number of IP addresses. (/16 means first 16bits are locked)

Input 11.0.0.0/16	Input IP 11.0.0.0	Input Long 184549376	Input Hex 0B.00.00.00
CIDR 11.0.0.0/16	CIDR IP Range 11.0.0.0 - 11.0.255.255	CIDR Long Range 184549376 - 184614911	CIDR Hex Range 0B.00.00.00 - 0B.00.FF.FF
IPs In Range 65,536		Mask Bits 16	Subnet Mask 255.255.0.0
Hex Subnet Mask FF.FF.00.00			

ABOUT SUBNET CALCULATOR

STEP4: CLICK ON CREATE VPC

The screenshot shows the 'Create VPC' wizard on the AWS VPC console. The 'IPv4 CIDR' field is set to '11.0.0.0/16'. Under 'IPv6 CIDR block', the 'No IPv6 CIDR block' option is selected. In the 'Tenancy' section, 'Default' is chosen. The 'Tags' section contains a single tag 'Name: munsifa-vpc'. At the bottom right, there are 'Cancel', 'Preview code', and a prominent orange 'Create VPC' button.

STEP5: VPC CREATED

The screenshot shows the 'VPC dashboard' page. On the left, a sidebar lists 'Virtual private cloud' and 'Security' sections. The main area displays the details of a newly created VPC: 'vpc-005d0123a256f64a1 / munsifa-vpc'. The 'Details' tab is active, showing the following configuration:

Setting	Value
VPC ID	vpc-005d0123a256f64a1
State	Available
DNS resolution	Enabled
Main network ACL	acl-062a879ad57a28e5
IPv4 CIDR (Network border group)	11.0.0.0/16
State	Available
Tenancy	default
Default VPC	No
Network Address Usage metrics	Disabled
Block Public Access	Off
DHCP option set	dopt-0a305ad2584effbbe
IPv4 CIDR	11.0.0.0/16
Route 53 Resolver DNS Firewall rule groups	-
DNS hostnames	Disabled
Main route table	rtb-040643d6b1efaf99f
IPv6 pool	-
Owner ID	767828733348

Below the details, there are tabs for 'Resource map', 'CIDRs', 'Flow logs', 'Tags', and 'Integrations'. A green success message at the top indicates 'You successfully created vpc-005d0123a256f64a1 / munsifa-vpc'.

STEP6: CLICK ON SUBNET>CREATE SUBNET

The screenshot shows the AWS VPC Subnets page. On the left, there's a navigation sidebar with sections like 'Virtual private cloud' (Your VPCs, Subnets, Route tables, Internet gateways, Egress-only internet gateways, DHCP option sets, Elastic IPs, Managed prefix lists, NAT gateways, Peering connections), 'Security' (Network ACLs, Security groups), and 'PrivateLink and Lattice'. The main area is titled 'Subnets (3) Info' and contains a table with three rows of subnet information:

Name	Subnet ID	State	VPC	Block Public...	IPv4 CIDR
-	subnet-0664e01666df6dbd0	Available	vpc-03771c780b4597cbe	Off	172.31.16.0/
-	subnet-0b3cc80b706b52a08	Available	vpc-03771c780b4597cbe	Off	172.31.0.0/2
-	subnet-0c3ad0209caf5783c	Available	vpc-03771c780b4597cbe	Off	172.31.32.0/

Below the table, there's a section titled 'Select a subnet' with three small icons.

STEP7: SELECT THE VPC THAT YOU HAVE CREATED EARLIER

The screenshot shows the 'Create subnet' wizard. The first step, 'Select a VPC', is displayed. It has a dropdown menu labeled 'Select a VPC' with two options listed:

- vpc-03771c780b4597cbe
172.31.0.0/16
- vpc-005d0123a256f64a1 (munsifa-vpc)
11.0.0.0/16

Below the dropdown, there's a note: 'Select a VPC first to create new subnets.' At the bottom right are 'Cancel' and 'Create subnet' buttons.

STEP8: SELECT ap-south-1a in the Availability Zone

Subnet settings

Specify the CIDR blocks and Availability Zone for the subnet.

Subnet

- No preference
- ✓ Asia Pacific (Mumbai) / ap-south-1a
ID: aps1-a21 Network border group: ap-south-1 ap-south-1-zg-1
- Asia Pacific (Mumbai) / ap-south-1b
ID: aps1-a25 Network border group: ap-south-1 ap-south-1-zg-1
- Asia Pacific (Mumbai) / ap-south-1c
ID: aps1-a22 Network border group: ap-south-1 ap-south-1-zg-1

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.

11.0.0.0/16

IPv4 subnet CIDR block

11.0.0.0/20

▼ Tags - optional

STEP9: ENTER IPv4 SUBNET CIDR BLOCK-11.0.1.0/24> CLICK ON ADD NEW SUBNET

Subnet name

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

munsifa-public-subnet

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.

Asia Pacific (Mumbai) / ap-south-1a

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.

11.0.0.0/16

IPv4 subnet CIDR block

11.0.1.0/24 256 IPs

▼ Tags - optional

Key Value - optional

Name munsifa-public-subnet [Remove](#)

Add new tag

You can add 49 more tags.

[Remove](#)

STEP9: ENTER THE NAME OF THE PRIVATE SUBNET> AVAILABILITY ZONE:ap-south-1b> IPv4 subnet CIDR Block-11.0.2.0/24> CLICK ON CREATE SUBNET

The screenshot shows the 'Create subnet' wizard step 2 of 2. The 'Subnet name' field contains 'munsifa-private-subnet'. The 'Availability Zone' dropdown is set to 'Asia Pacific (Mumbai) / ap-south-1b'. The 'IPv4 VPC CIDR block' dropdown is set to '11.0.0.0/16'. The 'IPv4 subnet CIDR block' dropdown is set to '11.0.2.0/24'. A single tag 'Name' is added with the value 'munsifa-private-subnet'. The page includes standard AWS navigation and footer links.

STEP10: BOTH THE PUBLIC AND PRIVATE SUBNET IS CREATED

The screenshot shows the VPC dashboard with a success message: 'You have successfully created 2 subnets: subnet-038d4320df6ba08a0, subnet-07f0b6e626eec4e1c'. The 'Subnets' table lists two entries:

Name	Subnet ID	State	VPC	Block Public...	IPv4 CIDR
munsifa-public-subnet	subnet-038d4320df6ba08a0	Available	vpc-005d0123a256f64a1 mun...	Off	11.0.1.0/24
munsifa-private-subnet	subnet-07f0b6e626eec4e1c	Available	vpc-005d0123a256f64a1 mun...	Off	11.0.2.0/24

STEP11: NOW CREATE AN INTERNET GATEWAY FOR PUBLIC SUBNET. CLICK ON INTERNET GATEWAY FROM THE DASHBOARD> CLICK ON CREATE INTERNET GATEWAY

The screenshot shows the AWS VPC dashboard with the 'Internet gateways' section selected. A table lists one internet gateway:

Name	Internet gateway ID	State	VPC ID	Owner
-	igw-07cf8e773e30f3337	Attached	vpc-03771c780b4597cbe	767828733548

Below the table, a message says "Select an internet gateway above".

STEP12: TYPE THE NAME OF THE GATEWAY

The screenshot shows the 'Create internet gateway' wizard. The first step, 'Internet gateway settings', is displayed. It includes a 'Name tag' field where the value 'munsifa-internet-gateway' is entered.

The second step, 'Tags - optional', shows a single tag named 'munsifa-internet-gateway' with a value of 'munsifa-internet-gateway'. There is also an 'Add new tag' button.

At the bottom right, there are 'Cancel' and 'Create internet gateway' buttons.

STEP13: INTERNET GATEWAY IS CREATED

The screenshot shows the AWS VPC console interface. On the left, there's a navigation sidebar with options like 'Virtual private cloud' (selected), 'Internet gateways' (selected), and 'Security'. The main content area displays a success message: 'The following internet gateway was created: igw-07d0609f48740f928 - munsifa-internet-gateway. You can now attach to a VPC to enable the VPC to communicate with the internet.' Below this, the 'igw-07d0609f48740f928 / munsifa-internet-gateway' details are shown. It includes fields for 'Internet gateway ID' (igw-07d0609f48740f928), 'State' (Detached), 'VPC ID' (empty), and 'Owner' (MUNSIFA FIRDAUS KHAN BARBHUYAN). A 'Tags' section lists a single tag 'Name: munsifa-internet-gateway'. At the bottom right, there are 'Actions' and 'Attach to a VPC' buttons.

STEP14: CLICK ON ATTACH TO VPC> SELECT THE VPC CREATED EARLIER> CLICK ON ATTACH INTERNET GATEWAY

The screenshot shows the 'Attach to VPC' dialog box. It starts with a message: 'The following internet gateway was created: igw-07d0609f48740f928 - munsifa-internet-gateway. You can now attach to a VPC to enable the VPC to communicate with the internet.' Below this, it says 'Attach to VPC (igw-07d0609f48740f928)'. The 'Available VPCs' section contains a dropdown menu with one option: 'vpc-005d0123a256f64a1 - munsifa-vpc'. At the bottom right, there are 'Cancel' and 'Attach internet gateway' buttons.

STEP15: INTERNET GATEWAY IS ATTACHED TO THE VPC

The screenshot shows the AWS VPC dashboard. On the left, there's a sidebar with navigation links like 'Virtual private cloud' (selected), 'Security', and 'CloudShell'. The main area displays an Internet gateway named 'igw-07d0609f48740f928' which has been successfully attached to a VPC. The 'Details' section shows the Internet gateway ID, state (Attached), VPC ID, and owner. Below it, the 'Tags' section lists a single tag 'Name: munsifa-internet-gateway'. At the bottom right, there's a 'Manage tags' button.

STEP16: NEXT CREATE TWO ROUTE TABLES (PRIVATE AND PUBLIC) FOR THE SUBNETS.

STEP17: TO CREATE A PUBLIC ROUTE TABLE: CLICK ON ROUTE TABLES>CREATE A ROUTE TABLE> WRITE A SUITABLE NAME FOR THE ROUTE TABLE>CHOOSE THE VPC>CLICK ON CREATE ROUTE TABLE

The screenshot shows the 'Create route table' wizard. In the 'Route table settings' step, a name 'munsifa-route-table-public' is entered. Under the 'VPC' section, 'vpc-005d0123a256f64a1 (munsifa-vpc)' is selected. A checkbox for tracking AWS costs is checked. A key-value pair 'Name: munsifa-route-table-public' is added. At the bottom right, there are 'Cancel' and 'Create route table' buttons.

STEP18: TO CREATE A PRIVATE ROUTE TABLE: CLICK ON ROUTE TABLES>CREATE A ROUTE TABLE> WRITE A SUITABLE NAME FOR THE ROUTE TABLE>CHOOSE THE VPC>CLICK ON 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.

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

Key	Value - optional
<input type="text" value="Name"/>	<input type="text" value="munsifa-route-table-private"/> X Remove

Add new tag
You can add 49 more tags.

Cancel Create route table

STEP19: BOTH THE PUBLIC AND PRIVATE ROUTE TABLES ARE CREATED AS SHOWN BELOW. NOW WE NEED TO CONNECT BOTH THE ROUTE TABLES WITH THEIR CORRESPONDING SUBNETS.

Route tables (4) Info

Route table rtb-0232c690bddc76aa4 | munsifa-route-table-private was created successfully.

Name	Route table ID	Explicit subnet associ...	Edge associations	Main	VPC
munsifa-route-table-public	rtb-06b37485e34f71f79	-	-	No	vpc-005d0123a256f64a1 [n]
-	rtb-040643d6b1efaf99f	-	-	Yes	vpc-005d0123a256f64a1 [n]
-	rtb-0ab986ed945865073	-	-	Yes	vpc-03771c780b4597cbe
munsifa-route-table-private	rtb-0232c690bddc76aa4	-	-	No	vpc-005d0123a256f64a1 [n]

Select a route table

CloudShell Feedback

STEP20: CLICK ON THE PUBLIC ROUTE TABLE>CLICK ON SUBNET ASSOCIATIONS

The screenshot shows the AWS VPC console with the URL <https://ap-south-1.console.aws.amazon.com/vpcconsole/home?region=ap-south-1#RouteTableDetails:RouteTableId=rtb-06b37485e34f71f79>. The left sidebar shows 'Virtual private cloud' and 'Route tables'. The main content area is titled 'rtb-06b37485e34f71f79 / munsifa-route-table-public'. The 'Subnet associations' tab is selected. The table has 0 rows and displays the message 'No subnet associations'.

STEP 21: CLICK ON EDIT SUBNET ASSOCIATIONS > CHOOSE PUBLIC SUBNET FROM THE OPTIONS> CLICK ON SAVE ASSOCIATIONS

The screenshot shows the 'Edit subnet associations' page with the URL <https://ap-south-1.console.aws.amazon.com/vpcconsole/home?region=ap-south-1>EditRouteTableSubnetAssociations:RouteTableId=rtb-06b37485e34f71f79>. The 'Available subnets' table lists two subnets: 'munisia-public-subnet' and 'munisia-private-subnet'. The 'Selected subnets' section contains 'subnet-038d4320df6ba08a0 / munisia-public-subnet'. The 'Save associations' button is highlighted.

STEP 21: SUBNET ASSOCIATION IS DONE. REPEAT THE SAME STEP FOR PRIVATE SUBNET ASSOCIATION.

Name	Route table ID	Explicit subnet associ...	Edge associations	Main	VPC
munsifa-route-table-public	rtb-06b37485e34f71f79	subnet-038d4320df6ba0...	-	No	vpc-005d0123a256f64a1 n
-	rtb-040643d6b1efaf9f	-	-	Yes	vpc-005d0123a256f64a1 n
-	rtb-0ab986ec9458650f3	-	-	Yes	vpc-03771c780b4597cbe
munsifa-route-table-private	rtb-0232c690bddc76aa4	subnet-07f0b6e626eecd4e...	-	No	vpc-005d0123a256f64a1 n

STEP22: NEXT WE NEED TO CREATE THE ROUTE SO THAT INTERNET CAN BE ACCESSED WITH THE HELP OF INTERNET GATEWAY THROUGH THESE ROUTE TABLES.

STEP23: GO TO PUBLIC ROUTE TABLE FROM THE AWS CONSOLE> CLICK ON ROUTES>CLICK ON EDIT ROUTES

Route table ID	Main	Explicit subnet associations	Edge associations
rtb-06b37485e34f71f79	No	subnet-038d4320df6ba08a0 / munsifa-public-subnet	-

Destination	Target	Status	Propagated
11.0.0.0/16	local	Active	No

**STEP24: CLICK ON ADD ROUTE>SELECT THE IP-0.0.0.0/0>CHOOSE INTERNET GATEWAY>
SELECT THE INTERNET GATEWAY THAT YOU HAVE CREATED> CLICK ON SAVE CHANGES
(NOTE:0.0.0.0/0 means it allows all the IP addresses to access the resources present in the public subnet)**

The screenshot shows the 'Edit routes' interface for a specific route table. A new route is being configured with the following details:

- Destination:** 0.0.0.0/0
- Target:** Internet Gateway
- Status:** Active
- Propagated:** No

At the bottom right, there are buttons for 'Cancel', 'Preview', and 'Save changes'.

The screenshot shows the 'Route table Details' page for the updated route table. A success message at the top states: "Updated routes for rtb-06b37485e34f71f79 / munsifa-route-table-public successfully".

The 'Routes' tab is selected, displaying the following routes:

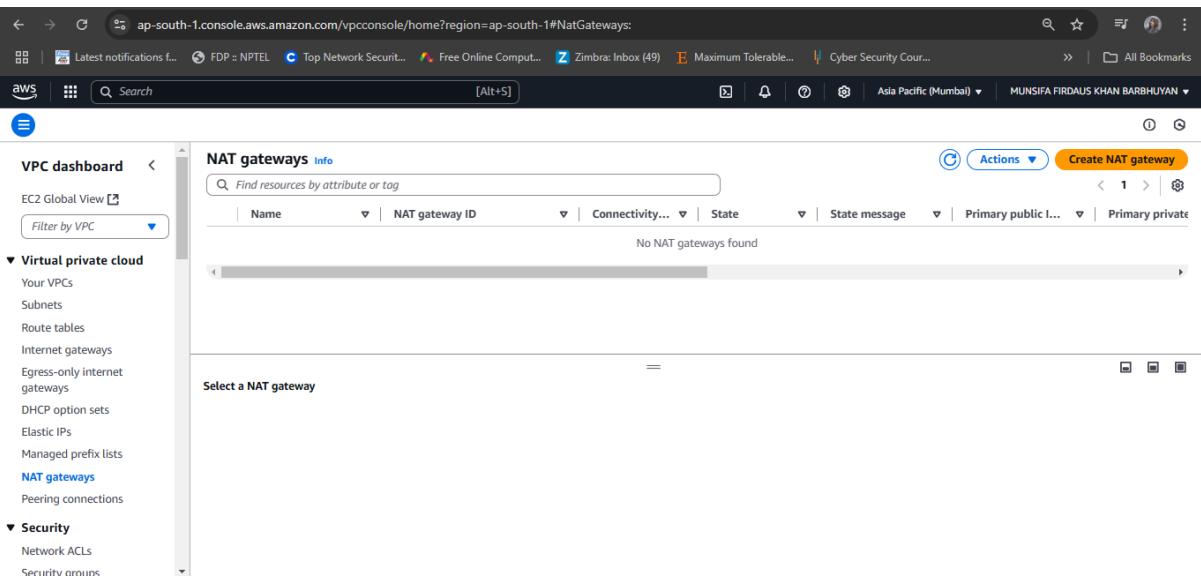
Destination	Target	Status	Propagated
0.0.0.0/0	igw-07d0609f48740f928	Active	No
11.0.0.16	local	Active	No

STEP25: WE NEED TO CREATE THE NAT GATEWAY FOR THE PRIVATE SUBNET SO THAT RESOURCES PRESENT INSIDE THE PRIVATE SUBNET CAN ACCESS THE INTERNET WITH THE HELP OF INTERNET GATEWAY.

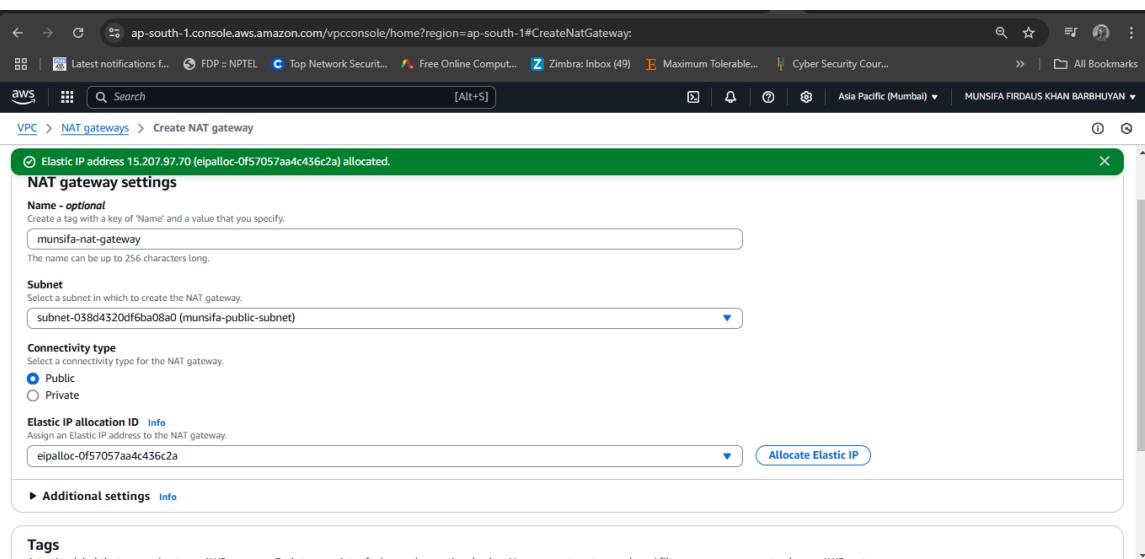
NOTE:

1. Internet Gateway is two-way process i.e., outside world can access the resources inside the public subnet as well as resources inside it can request for accessing outside.
2. NAT GATEWAY is not free and its created only in public subnet.
3. Private subnet will be able to access the NAT gateway with the help of its route table to access resources inside the public subnet.

STEP26: CLICK ON NAT GATEWAYS FROM THE DASHBOARD> CLICK ON CREATE NAT GATEWAY.



STEP27: GIVE A NAME TO THE NAT GATEWAY>CHOOSE THE PUBLIC SUBNET>ALLOCATE ELASTIC IP> CLICK ON CREATE NAT GATEWAY.



The screenshot shows the AWS VPC console interface. In the top navigation bar, the URL is `ap-south-1.console.aws.amazon.com/vpcconsole/home?region=ap-south-1#NatGatewayDetails:natGatewayId=nat-023482d369e4955cf`. The main content area displays a success message: "NAT gateway nat-023482d369e4955cf | munsifa-nat-gateway was created successfully." Below this, the title is "nat-023482d369e4955cf / munsifa-nat-gateway". The left sidebar shows the navigation path: VPC > NAT gateways > nat-023482d369e4955cf. The "Details" section provides the following information:

NAT gateway ID nat-023482d369e4955cf	Connectivity type Public	State Pending	State message Info
NAT gateway ARN arn:aws:ec2:ap-south-1:6782873348:natgateway/nat-023482d369e4955cf	Primary public IPv4 address -	Primary private IPv4 address -	Primary network interface ID -
VPC vpc-005d0123a256f64a1 / munsifa-vpc	Subnet subnet-038d4320df6ba08a0 / munsifa-public-subnet	Created Wednesday, February 12, 2025 at 21:15:59 GMT+5:30	Deleted -

The "Secondary IPv4 addresses" tab is selected, showing a table with columns: Private IPv4 address, Network interface ID, and Status. There are no entries in this table.

STEP28: NOW UPDATE THE PRIVATE ROUTE TABLE: CLICK ON THE ROUTE TABLES>CLICK ON THE PRIVATE ROUTE TABLE>CLICK ON ROUTES>CLICK ON EDIT ROUTES

This screenshot is identical to the one above, showing the creation of the NAT gateway. The "Details" section now includes the "Created" timestamp from the previous step. The "Secondary IPv4 addresses" table remains empty.

STEP29: CLICK ON ADD ROUTE>SELECT IP ADDRESS

:0.0.0.0/0>SELECT NAT GATEWAY>CHOOSE THE NAT GATEWAY THAT YOU HAVE CREATED>
CLICK ON SAVE CHANGES.

The screenshot shows the AWS VPC console with the URL ap-south-1.console.aws.amazon.com/vpcconsole/home?region=ap-south-1#EditRoutes:RouteTableId=rtb-0232c690bddc76aa4. The page title is "Edit routes".
The table shows one existing route:

Destination	Target	Status	Propagated
11.0.0.0/16	local	Active	No

A new route is being added:

Destination	Target	Status	Propagated
0.0.0.0/0	NAT Gateway	-	No

The "Target" dropdown has "nat-023482d369e4955cf" selected. The "Status" dropdown has "Active" selected. The "Propagated" dropdown has "No" selected.
At the bottom right, there are "Cancel", "Preview", and "Save changes" buttons. The "Save changes" button is highlighted in orange.

STEP30: PRIVATE SUBNET IS UPDATED WITH NAT GATEWAY

The screenshot shows the AWS VPC console with the URL ap-south-1.console.aws.amazon.com/vpcconsole/home?region=ap-south-1#RouteTableDetails:RouteTableId=rtb-0232c690bddc76aa4. The page title is "rtb-0232c690bddc76aa4 / munsifa-route-table-private".
A green success message box says: "Updated routes for rtb-0232c690bddc76aa4 / munsifa-route-table-private successfully".
The left sidebar shows the VPC dashboard and navigation links like EC2 Global View, Filter by VPC, Virtual private cloud, Route tables, Security, and Network ACLs.
The main content area shows the route table details:

Details	Info	Explicit subnet associations	Edge associations
Route table ID rtb-0232c690bddc76aa4	Main No	subnet-07f0b6e626eec4e1c / munsifa-private-subnet	-
VPC vpc-005d0123a256f64a1 munsifa-vpc	Owner ID 767828733348		

The "Routes" tab is selected, showing two routes:

Destination	Target	Status	Propagated
0.0.0.0/0	nat-023482d369e4955cf	Active	No
11.0.0.0/16	local	Active	No

At the bottom right, there are "Both", "Edit routes", and "Actions" buttons. The "Edit routes" button is highlighted in orange.

STEP31: WHOLE NETWORK SET UP IS READY, NOW WE HAVE TO CREATE EC2 INSTANCES FOR BOTH PUBLIC AND PRIVATE SUBNET. SEARCH FOR EC2>CLICK ON EC2

The screenshot shows the AWS VPC Services page. The left sidebar has sections for VPC dashboard, Virtual private clouds, Route tables, Security, and Network ACLs. The main content area is titled 'Services' and lists four items under 'Top features': EC2, EC2 Image Builder, EC2 Global View, and Recycle Bin. Each item has a description and a 'Show more' link. To the right, there's a large green box containing 'Actions' and 'Edge associations' sections, with a 'Both' dropdown and 'Edit routes' button at the bottom.

STEP 32: CLICK ON LAUNCH INSTANCE

The screenshot shows the AWS EC2 home page. The left sidebar has sections for EC2 (Dashboard, EC2 Global View, Events), Instances (Instances, Instance Types, Launch Templates, Spot Requests, Savings Plans, Reserved Instances, Dedicated Hosts, Capacity Reservations), Images (AMIs, AMI Catalog), and Elastic Block Store (Volumes). The main content area is titled 'Compute' and features a large heading 'Amazon Elastic Compute Cloud (EC2)'. Below it is the sub-headline 'Create, manage, and monitor virtual servers in the cloud.' A paragraph explains that EC2 offers a broadest and deepest compute platform with over 600 instance types. To the right, a box titled 'Launch a virtual server' contains a 'Launch instance' button and a 'View dashboard' link. Another box titled 'Get started' contains a 'Get started walkthroughs' link. At the bottom, there's a 'Benefits and features' section with a box for 'EC2 offers ultimate scalability and control'.

Compute

Amazon Elastic Compute Cloud (EC2)

Create, manage, and monitor virtual servers in the cloud.

Amazon Elastic Compute Cloud (Amazon EC2) offers the broadest and deepest compute platform, with over 600 instance types and a choice of the latest processors, storage, networking, operating systems, and purchase models to help you best match the needs of your workload.

Launch a virtual server

To get started, launch an Amazon EC2 instance, which is a virtual server in the cloud.

[Launch instance](#)

[View dashboard](#)

Get started

Take our walkthroughs to help you launch an instance, learn about EC2 best practices, and set up your account.

[Get started walkthroughs](#)

Benefits and features

EC2 offers ultimate scalability and control

Fully resizable compute capacity to support virtually any workload. This service is best if you want:

- Highest level of control of the entire technology stack, allowing full integration with all AWS services

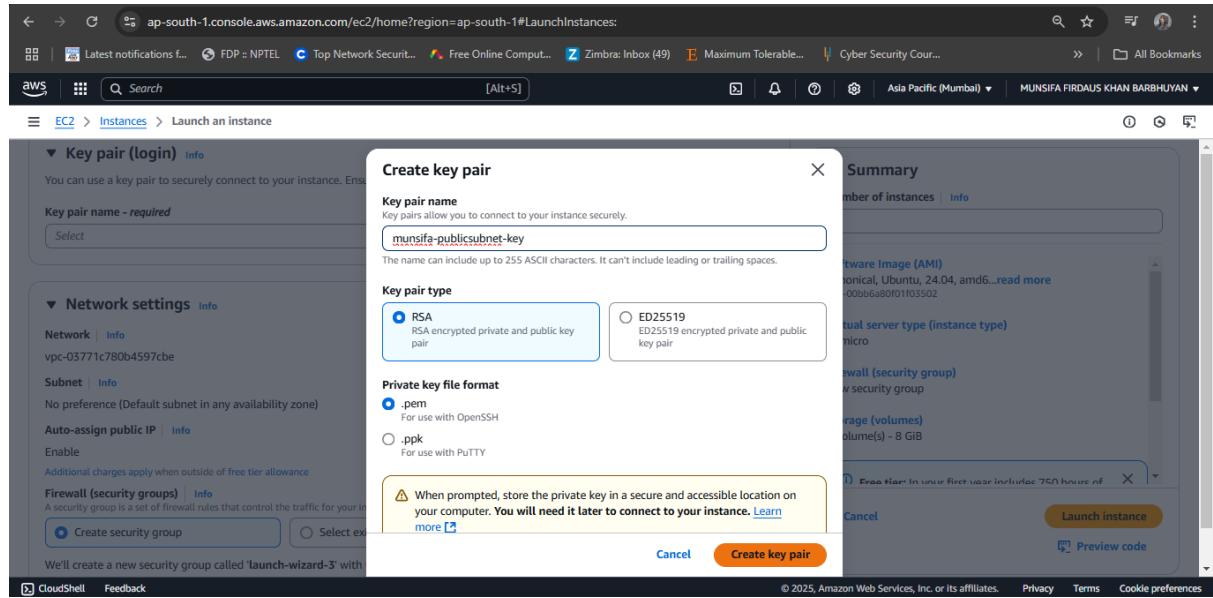
STEP 33: GIVE A SUITABLE INSTANCE NAME>CHOOSE OS:UBUNTU

The screenshot shows the AWS EC2 console interface for launching a new instance. In the 'Name and tags' section, the instance is named 'munsifa-public-ec2-instance'. The 'Application and OS Images (Amazon Machine Image)' section is expanded, showing various OS options like Amazon Linux, macOS, Ubuntu, Windows, Red Hat, SUSE Linux, and Debian. Ubuntu is selected. On the right, the 'Summary' panel shows 1 instance being launched. The 'Software Image (AMI)' section lists Canonical, Ubuntu, 24.04, amd64. The 'Virtual server type (instance type)' is set to t2.micro. The 'Storage (volumes)' section indicates 1 volume(s) - 8 GiB. A 'Free tier' message is visible. At the bottom right are 'Launch instance' and 'Preview code' buttons.

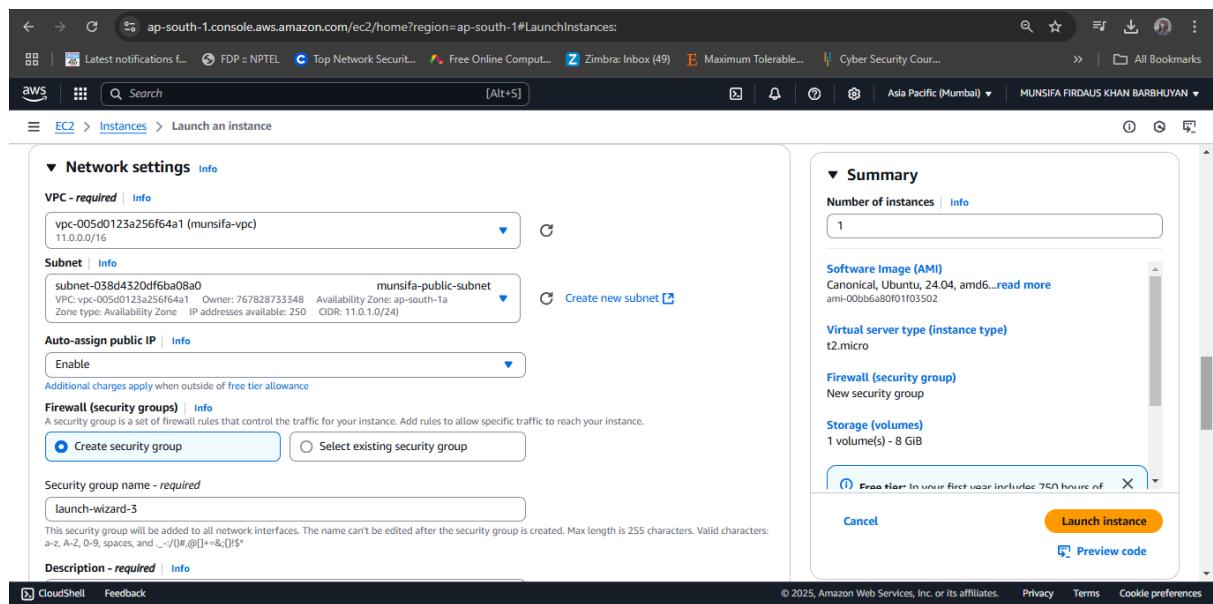
STEP 34: CHOOSE THE DEFAULT ARCHITECTURE> FREE TIER INSTANCE TYPE

This screenshot continues the instance creation process. It shows the 'Amazon Machine Image (AMI)' section with 'Ubuntu Server 24.04 LTS (HVM), SSD Volume Type' selected. The 'Architecture' dropdown is set to '64-bit (x86)'. The 'AMI ID' is 'ami-00bb6a80f01f03502'. The 'Username' is 'ubuntu' with a 'Verified provider' badge. The 'Instance type' section shows 't2.micro' selected, with details: Family: t2, 1 vCPU, 1 GiB Memory, Current generation: true. Pricing: On-Demand Linux base pricing: 0.0124 USD per Hour. The 'Free tier eligible' status is indicated. On the right, the 'Summary' panel shows 1 instance. The 'Software Image (AMI)' and 'Virtual server type (instance type)' sections are identical to the previous step. A 'Free tier' message is present. At the bottom right are 'Launch instance' and 'Preview code' buttons.

STEP 35: CREATE KEY PAIR



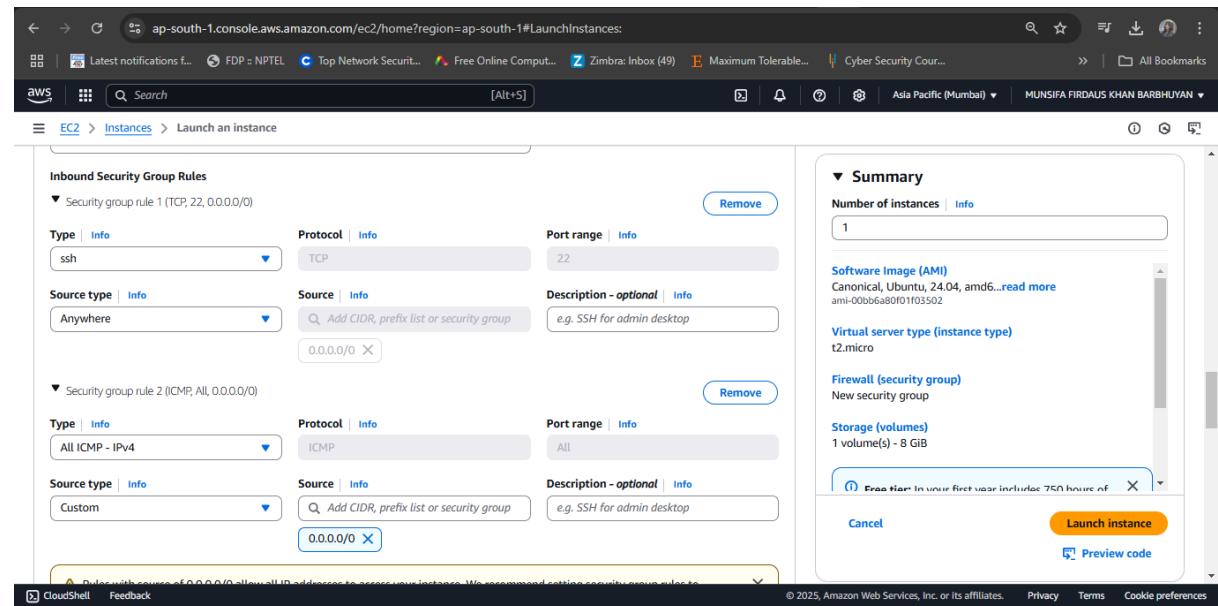
STEP 36: EDIT THE NETWORK SETTING>CHOOSE THE VPC CREATED EARLIER>SELECT THE PUBLIC SUBNET>ENABLE THE AUTO-ASSIGN PUBLIC IP>CHOOSE CREATE SECURITY GROUP



STEP 37: SECURITY GROUP RULE-1 IS DEFAULT> CLICK ON ADD SECURITY GROUP RULE

The screenshot shows the AWS EC2 console interface for launching a new instance. On the left, the 'Inbound Security Group Rules' section is open, displaying a single rule for SSH (TCP, port 22) from 'Anywhere'. A yellow warning box at the bottom left of the rules table states: '⚠ Roles with source of 0.0.0.0/0 allow all IP addresses to access your instance. We recommend setting security group rules to allow access from known IP addresses only.' Below this, there are buttons for 'Add security group rule' and 'Advanced network configuration'. On the right, the 'Summary' section shows the selected AMI (Canonical, Ubuntu, 24.04), instance type (t2.micro), and storage (1 volume(s) - 8 GiB). At the bottom right is a large orange 'Launch instance' button.

STEP 38: CHOOSE ALL ICMP-IPv4> SOURCE:0.0.0.0/0



This screenshot shows the continuation of the EC2 instance launch process. The 'Inbound Security Group Rules' section now includes two rules: one for SSH (TCP, port 22) from Anywhere and another for ICMP (All) from a 'Custom' source (0.0.0.0/0). The yellow warning message about allowing all IP addresses is still present. The 'Summary' section remains the same, and the 'Launch instance' button is visible at the bottom right.

STEP 39: CHOOSE DEFAULT CONFIGURE STORAGE: 8GB>CLICK ON LAUNCH INSTANCE

The screenshot shows the 'Configure storage' section of the AWS EC2 instance launch wizard. It specifies 1x 8 GiB gp3 volume as the root volume with 3000 IOPS (Not encrypted). A note indicates that free tier eligible customers can get up to 30 GB of EBS General Purpose (SSD) or Magnetic storage. Below this, it notes that the selected AMI contains more instance store volumes than the instance allows, and only the first 0 instance store volumes from the AMI will be accessible. There is also a note about backup information and file systems. The 'Advanced' tab is visible at the top right. On the right side of the screen, the 'Summary' section shows 1 instance being launched with a Canonical, Ubuntu, 24.04 AMI, t2.micro instance type, and 1 volume(s) - 8 GiB of storage. Buttons for 'Cancel', 'Launch instance', and 'Preview code' are present.

STEP 40: PUBLIC INSTANCE CREATED

The screenshot shows the AWS EC2 Instances page. The left sidebar is collapsed. The main area displays a table of instances. One instance is listed: 'munisifa-public' with Instance ID i-03a60b24ccb450003, running on a t2.micro instance type, and initializing. The 'Actions' dropdown menu for this instance is open, showing options like Stop, Terminate, and Reboot. The 'Launch instances' button is also visible. The bottom of the page includes standard AWS footer links.

STEP 41: CREATE EC2 INSTANCE FOR PRIVATE SUBNET WITH THE SAME STEPS

Name and tags [Info](#)

Name
munsifa-private-ec2-instance [Add additional tags](#)

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

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

[Search our full catalog including 1000s of application and OS images](#)

Recent **Quick Start**

Amazon Linux [aws](#) macOS [Mac](#) Ubuntu [ubuntu](#) Windows [Microsoft](#) Red Hat [Red Hat](#) SUSE Linux [SUSE](#) Debian [debian](#)

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

Amazon Machine Image (AMI)

CloudShell Feedback

Summary

Number of instances [Info](#)
1

Software Image (AMI)
Canonical, Ubuntu, 24.04, amd64... [read more](#)
ami-00bb6a80f01f03502

Virtual server type (instance type)
t2.micro

Firewall (security group)
New security group

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

Free tier In your first year include 750 hours of [View details](#)

[Cancel](#) [Launch instance](#) [Preview code](#)

© 2025, Amazon Web Services, Inc. or its affiliates. [Privacy](#) [Terms](#) [Cookie preferences](#)

Amazon Machine Image (AMI)

Ubuntu Server 24.04 LTS (HVM), SSD Volume Type
ami-00bb6a80f01f03502 (64-bit (x86)) / ami-09773b29dffbef1f2 (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>).

Canonical, Ubuntu, 24.04, amd64 noble image

Architecture **AMI ID** **Username** [Verified provider](#)

64-bit (x86) ami-00bb6a80f01f03502 ubuntu

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

Instance type

t2.micro [Free tier eligible](#)

Family: t2 1 vCPU 1 GiB Memory Current generation: true
On-Demand Linux base pricing: 0.0124 USD per Hour
On-Demand Windows base pricing: 0.017 USD per Hour
On-Demand Ubuntu Pro base pricing: 0.0224 USD per Hour
On-Demand Ubuntu Pro base pricing: 0.011 USD per Hour
On-Demand SUSE base pricing: 0.0174 USD per Hour

All generations [Compare instance types](#)

[Cancel](#) [Launch instance](#) [Preview code](#)

© 2025, Amazon Web Services, Inc. or its affiliates. [Privacy](#) [Terms](#) [Cookie preferences](#)

DISABLE AUTO-ASSIGN-PUBLIC IP.

Network settings [Info](#)

VPC - required [Info](#)
vpc-005d0123a256f64a1 (munsifa-vpc)
11.0.0.0/16

Subnet [Info](#)
subnet-07f0b6626e6cc4de1c
VPC: vpc-005d0123a256f64a1 Owner: 76782873348 Availability Zone: ap-south-1b
Zone type: Availability Zone IP addresses available: 251 CIDR: 11.0.2.0/24

Auto-assign public IP [Info](#)
Disable [Create new subnet](#)

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.
 Create security group Select existing security group

Security group name - required
launch-wizard-4

This security group will be added to all network interfaces. The name can't be edited after the security group is created. Max length is 255 characters. Valid characters: a-z, A-Z, 0-9, spaces, and _-/!@#[]+=&:;{}\$*

Description - required [Info](#)
launch-wizard-4 created 2025-02-12T16:45:15.365Z

Summary

Number of instances [Info](#)
1

Software Image (AMI)
Canonical, Ubuntu, 24.04, amd64... [read more](#)
ami-00bb6a80f01f03502

Virtual server type (instance type)
t2.micro

Firewall (security group)
New security group

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

Free tier In your first year include 750 hours of [View details](#)

[Cancel](#) [Launch instance](#) [Preview code](#)

© 2025, Amazon Web Services, Inc. or its affiliates. [Privacy](#) [Terms](#) [Cookie preferences](#)

The screenshot shows the AWS EC2 Instances Launch wizard. On the left, under 'Inbound Security Group Rules', a new rule is being configured:

- Type:** ssh
- Protocol:** TCP
- Port range:** 22
- Source type:** Anywhere
- Description (optional):** e.g. SSH for admin desktop

A note at the bottom of the rule list states: "Rules with source of 0.0.0.0 allow all IP addresses to access your instance. We recommend setting security group rules to allow access from known IP addresses only."

On the right, the summary panel shows:

- Number of instances:** 1
- Software Image (AMI):** Canonical, Ubuntu, 24.04, amd64... (with a 'read more' link)
- Virtual server type (instance type):** t2.micro
- Firewall (security group):** New security group
- Storage (volumes):** 1 volume(s) - 8 GB
- Free tier:** In your first year includes 750 hours of usage

At the bottom right are 'Launch instance' and 'Preview code' buttons.

STEP 42: ADD SECURITY GROUP RULE>CHOOSE ALL ICMP-IPV4> SOURCE:11.0.1.0/24(PUBLIC SUBNET IP ADDRESS, COZ ONLY PUBLIC SUBNET CAN ACCESS THE PRIVATE SUBNET)>KEEP DEFAULT STORAGE>CLICK ON LAUNCH INSTANCE

The screenshot shows the AWS EC2 Instances Launch wizard. Under 'Inbound Security Group Rules', a new rule is being added:

- Type:** All ICMP - IPv4
- Protocol:** ICMP
- Port range:** All
- Source type:** Custom
- Source:** 11.0.1.0/24
- Description (optional):** e.g. SSH for admin desktop

A note at the bottom of the rule list states: "Rules with source of 0.0.0.0 allow all IP addresses to access your instance. We recommend setting security group rules to allow access from known IP addresses only."

On the right, the summary panel shows:

- Number of instances:** 1
- Software Image (AMI):** Canonical, Ubuntu, 24.04, amd64... (with a 'read more' link)
- Virtual server type (instance type):** t2.micro
- Firewall (security group):** New security group
- Storage (volumes):** 1 volume(s) - 8 GB
- Free tier:** In your first year includes 750 hours of usage

At the bottom right are 'Launch instance' and 'Preview code' buttons.

STEP 43: PRIVATE EC2 INSTANCE IS CREATED

The screenshot shows the AWS EC2 Instances page. On the left sidebar, under the 'Instances' section, 'Instances' is selected. The main area displays a table of instances with columns: Name, Instance ID, Instance state, Instance type, Status check, Alarm status, Availability Zone, and Public IP. Three instances are listed:

Name	Instance ID	Instance state	Instance type	Status check	Alarm status	Availability Zone	Public IP
munsifa-private...	i-04a101c4c9fb1848e	Running	t2.micro	2/2 checks passed	View alarms +	ap-south-1b	-
munsifa-wind...	i-082c5eedfaf65e6f	Running	t3.micro	3/3 checks passed	View alarms +	ap-south-1b	ec2-35-...
munsifa-publi...	i-03a60b24ccb450003	Running	t2.micro	2/2 checks passed	View alarms +	ap-south-1a	-

The 'munsifa-private...' instance is selected, and its details are shown in the bottom panel. The 'Details' tab is active, displaying information such as Instance ID (i-04a101c4c9fb1848e), Public IPv4 address (11.0.2.160), and Instance state (Running).

STEP 44: NOW CONNECT THE PUBLIC EC2 INSTANCE.CLICK ON THE PUBLIC INSTANCE> CLICK ON CONNECT

The screenshot shows the AWS EC2 Instance summary page for the instance i-03a60b24ccb450003. The left sidebar shows the 'Instances' section is selected. The main area displays detailed information about the instance, including its ID, state, type, and network settings.

Instance summary for i-03a60b24ccb450003 (munsifa-public-ec2-instance)

Value	Description
Instance ID	i-03a60b24ccb450003
IPV6 address	-
Hostname type	IP name: ip-11-0-1-175.ap-south-1.compute.internal
Answer private resource DNS name	-
Auto-assigned IP address	13.235.241.171 [Public IP]
IAM Role	-
IMDSv2	Required
Public IPv4 address	13.235.241.171 open address
Instance state	Running
Private IP DNS name (IPv4 only)	ip-11-0-1-175.ap-south-1.compute.internal
Instance type	t2.micro
VPC ID	vpc-005d0123a256f64a1 (munsifa-vpc)
Subnet ID	subnet-038d4320df6ba08a0
Instance ARN	arn:aws:ec2:ap-south-1:76782873348:instance/i-03a60b24ccb450003
Private IPv4 addresses	11.0.1.175
Public IPv4 DNS	-
Elastic IP addresses	-
AWS Compute Optimizer finding	Opt-in to AWS Compute Optimizer for recommendations. Learn more
Auto Scaling Group name	-
Managed	false

The 'Connect' button is located at the top right of the instance summary page.

STEP 45: CLICK ON SSH CLIENT> COPY THE EXAMPLE AND PASTE IN TERMINAL TO CONNECT

The screenshot shows a browser window with the URL ap-south-1.console.aws.amazon.com/ec2/home?region=ap-south-1#ConnectToInstance:instanceId=i-03a60b24ccb450003. The page title is "Connect to instance". Below the title, it says "Connect to your instance i-03a60b24ccb450003 (munsifa-public-ec2-instance) using any of these options". There are four tabs: "EC2 Instance Connect", "Session Manager", "SSH client" (which is highlighted in blue), and "EC2 serial console". Under the "SSH client" tab, there is a section titled "Instance ID" with the value "i-03a60b24ccb450003 (munsifa-public-ec2-instance)". Below this, there is a numbered list of steps:

1. Open an SSH client.
2. Locate your private key file. The key used to launch this instance is munsifa-publicsubnet-key.pem
3. Run this command, if necessary, to ensure your key is not publicly viewable.
 chmod 400 "munsifa-publicsubnet-key.pem"
4. Connect to your instance using its Public IP:
 13.235.241.171

Below the steps, there is an "Example:" section with the command:
 ssh -i "munsifa-publicsubnet-key.pem" ubuntu@13.235.241.171

A note in a box says: "Note: In most cases, the guessed username is correct. However, read your AMI usage instructions to check if the AMI owner has changed the default AMI username."

STEP 46: OPEN THE TERMINAL IN YOUR LAPTOP AND TYPE THE FOLLOWING COMMAND> NOW PASTE THE COMMAND HERE FROM THE SSH CLIENT

```
Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.

Install the latest PowerShell for new features and improvements! https://aka.ms/PSWindows

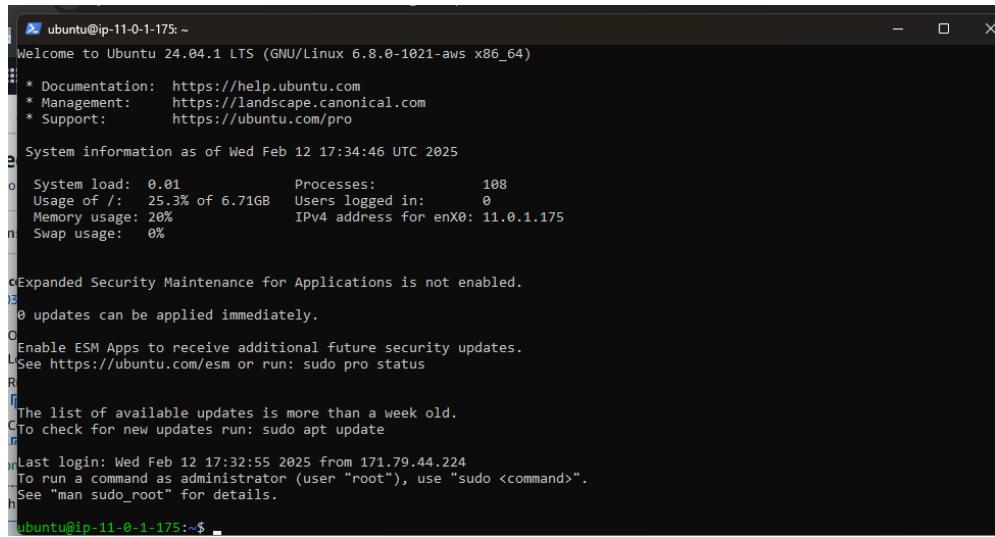
PS C:\Users\HP> cd Downloads
PS C:\Users\HP\Downloads> cd demo
PS C:\Users\HP\Downloads\demo> ls

    Directory: C:\Users\HP\Downloads\demo

Mode                LastWriteTime         Length Name
----                -----        ----- 
-a---       12-02-2025 10:17 PM          1674 munsifa-privatesubnet-key.pem
-a---       12-02-2025 09:54 PM          1678 munsifa-publicsubnet-key.pem

PS C:\Users\HP\Downloads\demo> ssh -i "munsifa-publicsubnet-key.pem" ubuntu@13.235.241.171
```

STEP 47: PUBLIC EC2 INSTANCE IS CONNECTED



```
ubuntu@ip-11-0-1-175: ~
Welcome to Ubuntu 24.04.1 LTS (GNU/Linux 6.8.0-1021-aws x86_64)

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

System information as of Wed Feb 12 17:34:46 UTC 2025

  System load: 0.01      Processes:          108
  Usage of /: 25.3% of 6.71GB  Users logged in: 0
  Memory usage: 20%           IPv4 address for enX0: 11.0.1.175
  Swap usage: 0%

  Expanded Security Maintenance for Applications is not enabled.

  0 updates can be applied immediately.

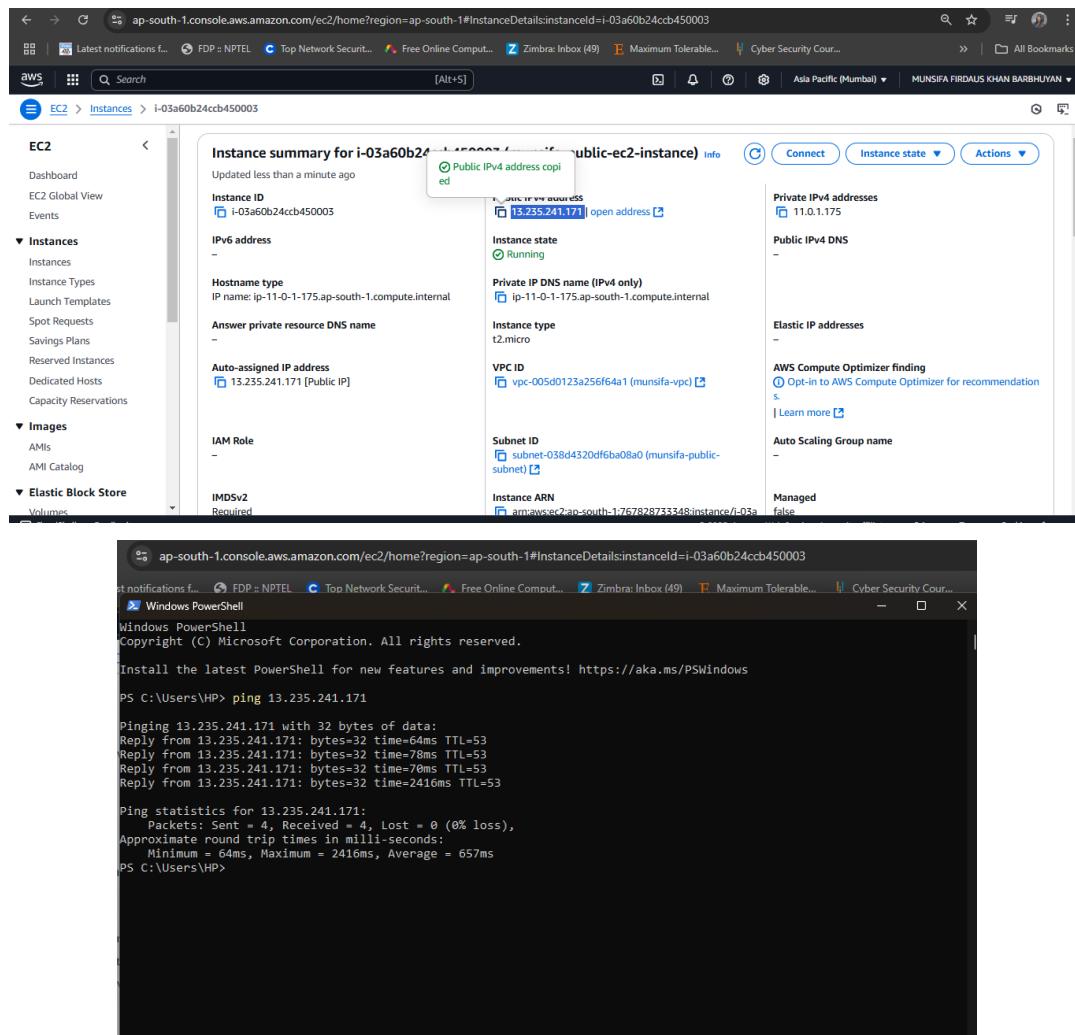
  Enable ESM Apps to receive additional future security updates.
  See https://ubuntu.com/esm or run: sudo pro status

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

  Last login: Wed Feb 12 17:32:55 2025 from 171.79.44.224
  To run a command as administrator (user "root"), use "sudo <command>".
  See "man sudo_root" for details.

ubuntu@ip-11-0-1-175:~$
```

STEP 48: NOW PING THE PUBLIC EC2 INSTANCE WITH THE HELP OF YOUR LOCAL TERMINAL. GO TO INSTANCE AND COPY ITS IP ADDRESS AND TYPE THE FOLLOWING COMMAND. (IT MEANS FROM OUTSIDE AWS, WE ARE ABLE TO CONNECT TO THE PUBLIC EC2 INSTANCE)



The screenshot shows two windows side-by-side. The left window is the AWS Management Console under the EC2 service, specifically the Instances section. It displays detailed information for an instance named 'i-03a60b24ccb450003'. Key details include:

- Instance ID: i-03a60b24ccb450003
- Public IPv4 address: 13.235.241.171
- Private IP DNS name (IPv4 only): ip-11-0-1-175.ap-south-1.compute.internal
- Instance type: t2.micro
- VPC ID: vpc-005d0123a256f64a1 (mansifa-vpc)
- Subnet ID: subnet-038d4320df6ba08a0 (mansifa-public-subnet)
- Instance ARN: arn:aws:ec2:ap-south-1:1767828733548:instance/i-03a60b24ccb450003

The right window is a Windows PowerShell session. It shows the command 'ping 13.235.241.171' being run and its output, which includes the IP address, round trip times, and statistics.

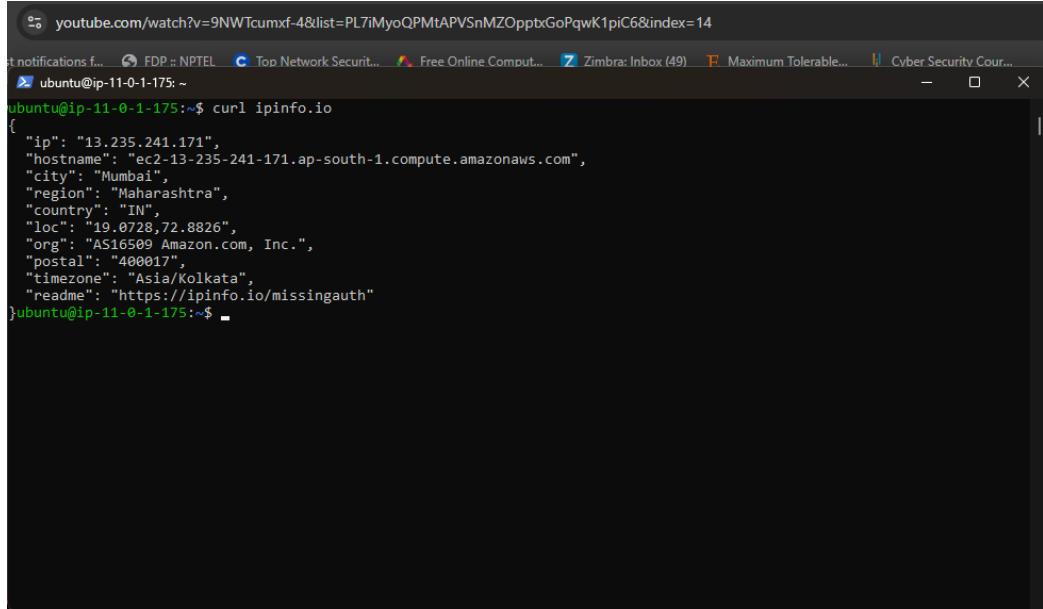
```
PS C:\Users\HP> ping 13.235.241.171

Pinging 13.235.241.171 with 32 bytes of data:
Reply from 13.235.241.171: bytes=32 time=64ms TTL=53
Reply from 13.235.241.171: bytes=32 time=78ms TTL=53
Reply from 13.235.241.171: bytes=32 time=70ms TTL=53
Reply from 13.235.241.171: bytes=32 time=2416ms TTL=53

Ping statistics for 13.235.241.171:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 64ms, Maximum = 2416ms, Average = 657ms
PS C:\Users\HP>
```

STEP 49: NOW NEXT FROM PUBLIC EC2 INSTANCE TERMINAL, TYPE THE FOLLOWING COMMAND TO ACCESS RESOURCE FROM OUTSIDE THE AWS

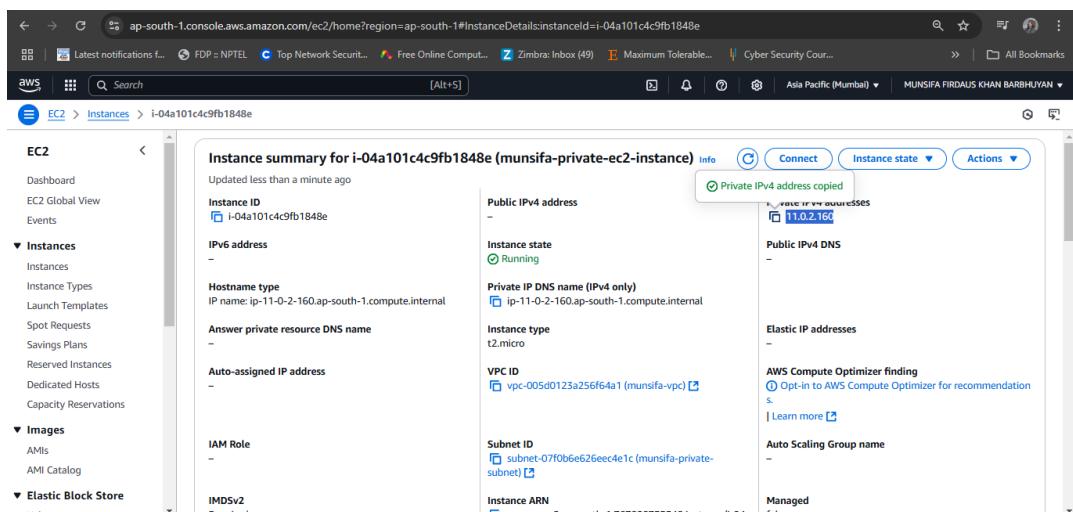
NOTE: curl ipinfo.io: It is a command used in the terminal to retrieve information about your current IP address using the "curl" tool and the "ipinfo.io" web service; essentially, it fetches details like your location, ISP, and public IP address by sending a request to the IPInfo.io API via the curl command.



```
ubuntu@ip-11-0-1-175:~$ curl ipinfo.io
{
  "ip": "13.235.241.171",
  "hostname": "ec2-13-235-241-171.ap-south-1.compute.amazonaws.com",
  "city": "Mumbai",
  "region": "Maharashtra",
  "country": "IN",
  "loc": "19.0728,72.8826",
  "org": "AS16500 Amazon.com, Inc.",
  "postal": "400017",
  "timezone": "Asia/Kolkata",
  "readme": "https://ipinfo.io/missingauth"
}ubuntu@ip-11-0-1-175:~$
```

It is seen that public EC2 instance is accessible from outside world as well as it can also access outside resources. But we know that, private instance that we have created under private subnet is not accessible from outside world but it can be accessed by the public EC2 instance only.

STEP 50: PING PRIVATE EC2 INSTANCE FROM THE LOCAL TERMINAL. COPY THE PRIVATE IP ADDRESS OF THE INSTANCE AND TYPE (PING 11.0.2.160) IN THE LOCAL TERMINAL.



```

PS C:\Users\HP> cd Downloads
PS C:\Users\HP\Downloads> cd demo
PS C:\Users\HP\Downloads\demo> ls

Directory: C:\Users\HP\Downloads\demo

Mode                LastWriteTime         Length Name
----                - - - - -             - - - - -
-a----       12-02-2025 10:17 PM          1674 munsifa-privatesubnet-key.pem
-s-a---       12-02-2025 09:54 PM          1678 munsifa-publicsubnet-key.pem

PS C:\Users\HP\Downloads\demo> ping 11.0.2.160

Pinging 11.0.2.160 with 32 bytes of data:
Request timed out.
Request timed out.
Request timed out.
Request timed out.

Ping statistics for 11.0.2.160:
    Packets: Sent = 4, Received = 0, Lost = 4 (100% loss),
PS C:\Users\HP\Downloads\demo>

```

ITS SHOWING REQUEST TIMED OUT. i.e., IT IS UNABLE TO CONNECT TO THE PRIVATE EC2 INSTANCE FROM THE LOCAL TERMINAL. NOW LET'S TRY TO CONNECT IT FROM THE PUBLIC EC2 INSTANCE TERMINAL.

STEP 50: TYPE THE FOLLOWING COMMAND TO CONNECT TO THE PRIVATE EC2 INSTANCE (PING 11.0.2.160). CONNECTION ESTABLISHED AS SHOWN BELOW.

```

ubuntu@ip-11-0-1-175: ~
ubuntu@ip-11-0-1-175: ~$ curl ipinfo.io
{
  "ip": "13.235.241.171",
  "hostname": "ec2-13-235-241-171.ap-south-1.compute.amazonaws.com",
  "city": "Mumbai",
  "region": "Maharashtra",
  "country": "IN",
  "loc": "19.0728,72.8826",
  "org": "AS16509 Amazon.com, Inc.",
  "postal": "400017",
  "timezone": "Asia/Kolkata",
  "readme": "https://ipinfo.io/missingauth"
}ubuntu@ip-11-0-1-175: ~$ ping 11.0.2.160
PING 11.0.2.160 (11.0.2.160) 56(84) bytes of data.
64 bytes from 11.0.2.160: icmp_seq=1 ttl=64 time=1.85 ms
64 bytes from 11.0.2.160: icmp_seq=2 ttl=64 time=0.955 ms
64 bytes from 11.0.2.160: icmp_seq=3 ttl=64 time=0.987 ms
64 bytes from 11.0.2.160: icmp_seq=4 ttl=64 time=1.11 ms
64 bytes from 11.0.2.160: icmp_seq=5 ttl=64 time=1.11 ms
64 bytes from 11.0.2.160: icmp_seq=6 ttl=64 time=0.840 ms
64 bytes from 11.0.2.160: icmp_seq=7 ttl=64 time=0.987 ms
64 bytes from 11.0.2.160: icmp_seq=8 ttl=64 time=1.02 ms
64 bytes from 11.0.2.160: icmp_seq=9 ttl=64 time=1.20 ms
64 bytes from 11.0.2.160: icmp_seq=10 ttl=64 time=1.09 ms
64 bytes from 11.0.2.160: icmp_seq=11 ttl=64 time=0.748 ms
64 bytes from 11.0.2.160: icmp_seq=12 ttl=64 time=1.07 ms
64 bytes from 11.0.2.160: icmp_seq=13 ttl=64 time=1.23 ms
64 bytes from 11.0.2.160: icmp_seq=14 ttl=64 time=0.984 ms
64 bytes from 11.0.2.160: icmp_seq=15 ttl=64 time=0.927 ms

```

RELEASE ALL THE RESOURCES THAT ARE USED IN THE EXECUTION OF THE LAB.

1. TERMINATE THE EC2 INSTANCES.
2. DELETE ROUTE TABLE:
 - a. ROUTE TABLE>EDIT ROUTE>REMOVE NAT GATEWAY (PRIVATE)
 - b. ROUTE TABLE>EDIT ROUTE>REMOVE INTERNET GATEWAY (PUBLIC)
 - c. ROUTE TABLE>SUBNET ASSOCIATIONS>EDIT SUBNET ASSOCIATIONS>DESELECT>SAVE CHANGES
 - d. SELECT BOTH THE ROUTE TABLES>ACTIONS>DELETE ROUTE TABLES.
3. DELETE SUBNETS:
 - a. SUBNETS> SELECT BOTH THE SUBNET ACTIONS>ACTIONS> DELETE SUBNET
4. DELETE NAT GATEWAY:
 - a. NAT GATEWAY>SELECT THE NAT GATEWAY>ACTIONS> DELETE NAT GATEWAY
5. DELETE VPC
 - a. VPC>SELECT THE VPC CREATED>ACTIONS>DELETE
6. RELEASE ELASTIC IP
 - a. ELASTIC IP>SELECT THE ELASTIC IP>ACTIONS>RELEASE THE ELASTIC IP