

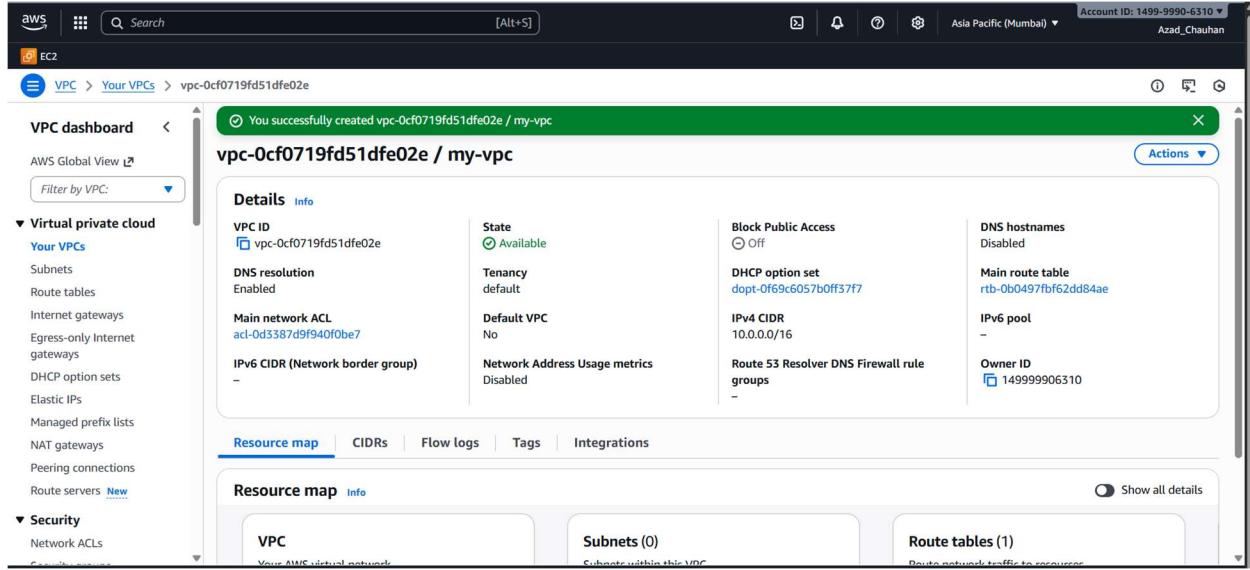
VPC with Public & Private Subnets + Bastion Host Setup

&

Private EC2 Access via Bastion Host (Jump Server)

STEP 1: Create VPC

1. AWS Console → **VPC**
2. **Create VPC**
3. Choose: **VPC Only**
4. Name: **my-vpc**
5. **IPv4 CIDR: 10.0.0.0/16**
6. **Tenancy : Default**
7. **Create**



STEP 2: Create Subnets

👉 Public Subnet

1. Subnets → Create subnet

2. **Name:** public-subnet
3. **VPC:** my-vpc
4. **CIDR:** 10.0.1.0/24
5. **Availability Zone:** ap-south-1a
6. **Enable Auto-assign Public IP:** Turn ON

👉 Private Subnet

1. Subnets → Create subnet
2. **Name:** private-subnet
3. **CIDR:** 10.0.2.0/24
4. **AZ:** ap-south-1a
5. **Auto-assign Public IP:** OFF

The screenshot shows the AWS VPC Subnets page. On the left, there's a navigation sidebar with options like VPC dashboard, 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, Route servers), and Security (Network ACLs). The main content area has a header "Subnets (2) Info" with filters for Subnet ID, Name, State, VPC, Block Public, and IPv4 CIDR. It shows two subnets: "public-subnet" (Subnet ID: subnet-0d4e2664802795c8a, State: Available, VPC: vpc-0cf0719fd51dfe02e | my-vpc, IPv4 CIDR: 10.0.1.0/24) and "private-subnet" (Subnet ID: subnet-0f878426453090756, State: Available, VPC: vpc-0cf0719fd51dfe02e | my-vpc, IPv4 CIDR: 10.0.2.0/24). Below the table is a section titled "Select a subnet".

STEP 3: Create & Attach Internet Gateway (IGW)

1. Internet Gateways → Create
2. Name: my-igw

3. Attach to VPC: my-vpc

The screenshot shows the AWS VPC Internet Gateways page. A success message at the top states: "Internet gateway igw-0cd75ebdd05c40c22 successfully attached to vpc-0cf0719fd51dfe02e". The main card displays the following details:

- Internet gateway ID:** igw-0cd75ebdd05c40c22
- State:** Attached
- VPC ID:** vpc-0cf0719fd51dfe02e | my-vpc
- Owner:** 149999906310

The "Tags (1)" section shows a single tag: Name = my-igw. There is a "Manage tags" button and a navigation bar with a single item.

STEP 4: Route Table for Public Subnet

1. Route Tables → Create
2. Name: public-rt
3. VPC: my-vpc
4. Create

The screenshot shows the AWS VPC Route Tables page. A success message at the top states: "Route table rtb-0ec97f0fef5aba7ef | public-rt was created successfully." The main card displays the following details:

- Route table ID:** rtb-0ec97f0fef5aba7ef
- Main:** No
- VPC:** vpc-0cf0719fd51dfe02e | my-vpc
- Owner ID:** 149999906310
- Explicit subnet associations:** -
- Edge associations:** -

The "Routes (1)" section shows one route: Destination = 10.0.0/16, Target = local, Status = Active, Propag... = No, and Route Origin = Create Route Table. There is a "Both" dropdown, an "Edit routes" button, and a navigation bar with a single item.

5. Edit routes:

- Destination: 0.0.0.0/0
- Target: **Internet Gateway (my-igw)**
- **Save Changes**

The screenshot shows the AWS VPC Route Tables page. A green success message at the top states: "Updated routes for rtb-0ec97f0fef5aba7ef / public-rt successfully". Below this, the route table details are shown: Route table ID (rtb-0ec97f0fef5aba7ef), Main (No), Owner ID (vpc-0cf0719fd51dfe02e | my-vpc). The "Routes" tab is selected, showing two routes: Destination 0.0.0.0/0 Target my-igw Status Active Propagated Route Origin Main. There are buttons for "Edit routes" and "Actions".

6. Subnet associations:

- Select **public-subnet**

The screenshot shows the "Edit subnet associations" page for the route table. It lists available subnets: public-subnet (selected) and private-subnet. The public-subnet is associated with the Main route table. The "Selected subnets" section shows the public-subnet selected. At the bottom are "Cancel" and "Save associations" buttons.

Available subnets (1/2)				
Name	Subnet ID	IPv4 CIDR	IPv6 CIDR	Route table ID
<input checked="" type="checkbox"/> public-subnet	subnet-0d4e2664802795c8a	10.0.1.0/24	-	Main (rtb-0b0497fbf62dd84ae)
<input type="checkbox"/> private-subnet	subnet-0f878426453090756	10.0.2.0/24	-	Main (rtb-0b0497fbf62dd84ae)

Selected subnets

subnet-0d4e2664802795c8a / public-subnet X

[Cancel](#) Save associations

★ Public subnet now has internet access.

STEP 5: NAT Gateway for Private Subnet

Private subnet ko internet dene ke liye NAT Gateway use hota hai.

1. NAT Gateway → Create
2. Subnet: **public-subnet**
3. Elastic IP: Allocate + attach
4. Create

The screenshot shows the AWS VPC dashboard with the 'NAT gateways' section selected. A success message at the top states: "NAT gateway nat-Off019b2ad11590ae | my-nat-gateway was created successfully." The main details pane shows the following information:

NAT gateway ID nat-Off019b2ad11590ae	Connectivity type Public	State Pending	State message Info
NAT gateway ARN arn:aws:ec2:ap-south-1:14999990631 0:natgateway/nat-Off019b2ad11590ae	Primary public IPv4 address -	Primary private IPv4 address -	Primary network interface ID -
VPC vpc-0cf0719fd51dfe02e / my-vpc	Subnet subnet-0d4e2664802795c8a / public-subnet	Created Friday 14 November 2025 at 19:58:51 GMT+5:30	Deleted -

Below the details, there are tabs for "Secondary IPv4 addresses", "Monitoring", and "Tags". The "Secondary IPv4 addresses" tab is active, showing a message: "Secondary IPv4 addresses are not available for this nat gateway." There is also a "Edit secondary IPv4 address associations" button.

STEP 6: Route Table for Private Subnet

1. Route Tables → Create
2. Name: **private-rt**
3. VPC: **my-vpc**
4. Create

Azad Chauhan

The screenshot shows the AWS VPC Route Tables page. A success message at the top states "Route table rtb-0a4d8e32c87276821 | private-rt was created successfully." The main section displays the details of the route table "rtb-0a4d8e32c87276821 / private-rt". The "Details" tab is selected, showing the route table ID, VPC, and owner ID. The "Routes" tab is active, showing one route entry: Destination 10.0.0.0/16, Target local, Status Active, Propagated No, and Route Origin CreateRouteTable. The "Subnet associations" tab is also present.

5. Edit routes:

- Destination: 0.0.0.0/0
- Target: **NAT Gateway**

The screenshot shows the "Edit routes" dialog box for the route table "rtb-0a4d8e32c87276821". It lists a route entry with Destination 10.0.0.0/16, Target local, Status Active, Propagated No, and Route Origin CreateRouteTable. Below it, another route entry is being added: Destination 0.0.0.0/0, Target NAT Gateway, Status - (pending), Propagated No, and Route Origin CreateRoute. The "Add route" button is visible at the bottom left, and "Cancel", "Preview", and "Save changes" buttons are at the bottom right.

6. Subnet Association:

- Select **private-subnet**

Azad Chauhan

The screenshot shows the AWS VPC Route Tables interface. A green success message at the top states: "You have successfully updated subnet associations for rtb-0a4d8e32c87276821 / private-rt." The main page title is "rtb-0a4d8e32c87276821 / private-rt". On the left sidebar under "Virtual private cloud", "Route tables" is selected. The "Details" tab is active, showing the route table ID (rtb-0a4d8e32c87276821), Main status (No), Owner ID (149999906310), and explicit subnet associations (subnet-0f878426453090756 / private-subnet). The "Routes" tab is selected, displaying two routes:

Destination	Target	Status	Propagated	Route Origin
0.0.0.0/0	nat-0ff019b2ad11590ae	Active	No	Create Route
10.0.0.0/16	local	Active	No	Create Route Table

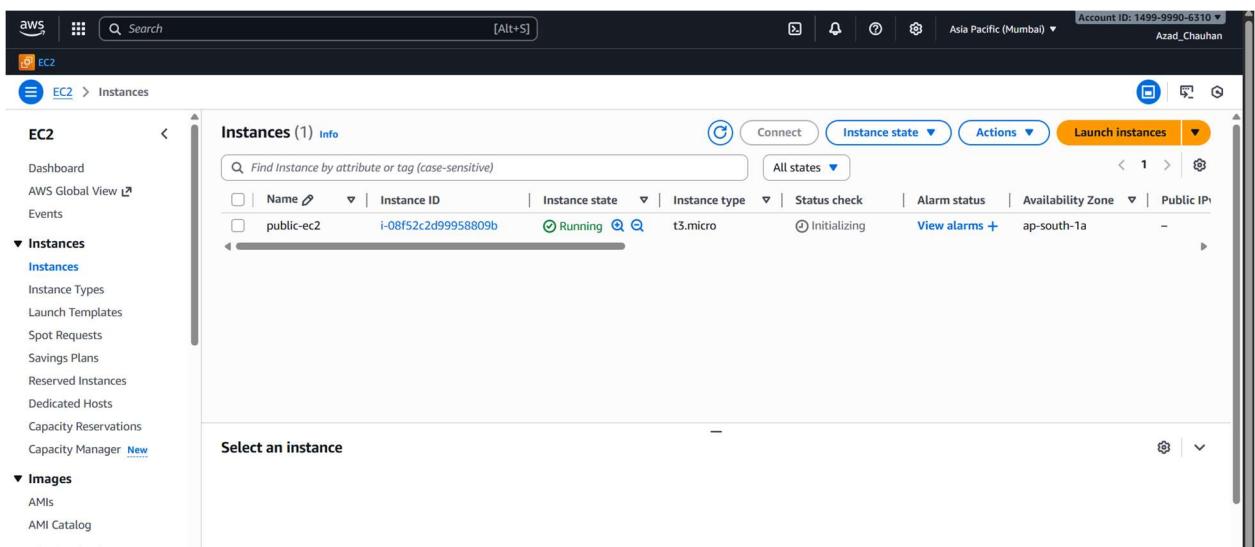
★ Ab private subnet ke resources **internet access le sakte hain**, but inbound internet se hidden rahenge.

sMETHOD: Connect Private EC2 Using Public EC2 (Bastion Host)

(Requires 2 EC2 instances: 1 Public + 1 Private)

✓ STEP 1 — Launch Public EC2

1. Go to EC2 → Launch instance
2. Name: public-ec2
3. Choose AMI: Ubuntu
4. Network settings:
 - VPC: *your VPC*
 - Subnet: public-subnet
 - Auto-assign Public IP: Enabled
5. Security Group:
 - Allow SSH (22) from My IP
6. Select / Create Key Pair
7. Launch instance



STEP 2 — Launch Private EC2

1. EC2 → Launch instance

2. Name: private-ec2

3. AMI: Ubuntu

4. Network settings:

- **VPC: same**
- **Subnet: private-subnet**
- **Auto-assign Public IP: Disabled**

5. Security Group:

- **Allow SSH (22)**
- **Source: Public EC2's security group (important)**

6. Launch instance

The screenshot shows the AWS EC2 Instances page. The left sidebar has 'Instances' selected. The main table lists two instances:

Name	Instance ID	Instance state	Instance type	Status check	Alarm status	Availability Zone	Public IP
public-ec2	i-08f52c2d99958809b	Running	t3.micro	3/3 checks passed	View alarms +	ap-south-1a	-
private-ec2	i-0e992f9b526f5c2af	Running	t3.micro	Initializing	View alarms +	ap-south-1a	-

STEP 3 — Connect to Public EC2

Open your terminal and run:

```
scp -i "john.pem" john.pem ubuntu@<PUBLIC_EC2_IP>/home/ubuntu/
```

(This command securely copies the john.pem key file from your local machine to the /home/ubuntu/ directory on the public EC2 instance using the provided SSH key.)

Azad Chauhan

```
C:\Users\chauh>cd Downloads  
C:\Users\chauh\Downloads>scp -i "john.pem" john.pem ubuntu@15.206.122.18:/home/ubuntu/  
john.pem 100% 1678 1.8KB/s 00:00
```

After this command you can connect your EC2 via ssh

```
ssh -i "john.pem" ubuntu@15.206.122.18
```

```
ubuntu@ip-10-0-2-40:~ x + v  
C:\Users\chauh>scp -i "john.pem" john.pem ubuntu@15.206.122.18:/home/ubuntu/  
scp: stat local "john.pem": No such file or directory  
C:\Users\chauh>cd Downloads  
C:\Users\chauh\Downloads>scp -i "john.pem" john.pem ubuntu@15.206.122.18:/home/ubuntu/  
john.pem 100% 1678 1.8KB/s 00:00  
C:\Users\chauh\Downloads>ssh -i "john.pem" ubuntu@15.206.122.18  
Welcome to Ubuntu 24.04.3 LTS (GNU/Linux 6.14.0-1015-aws x86_64)  
  
* Documentation: https://help.ubuntu.com  
* Management: https://landscape.canonical.com  
* Support: https://ubuntu.com/pro  
  
System information as of Fri Nov 14 15:24:16 UTC 2025  
  
System load: 0.0 Temperature: -273.1 C  
Usage of /: 26.2% of 6.71GB Processes: 117  
Memory usage: 24% Users logged in: 1  
Swap usage: 0% IPv4 address for ens5: 10.0.1.203  
  
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: Fri Nov 14 15:16:01 2025 from 47.11.23.2  
To run a command as administrator (user "root"), use "sudo <command>".  
See "man sudo_root" for details.  
  
ubuntu@ip-10-0-1-203:~$ chmod 400 john.pem  
ubuntu@ip-10-0-1-203:~$ ls -l john.pem  
-r----- 1 ubuntu ubuntu 1678 Nov 14 15:20 john.pem
```

chmod 400 john.pem

(This command changes the file permissions of john.pem so that only the owner can read it.
SSH requires private key files to have strict permissions for security.)

ls -l john.pem

(This command lists the file details and shows the new permissions.
You can see -r----- which means read-only for the owner.)

STEP 4 — Find Private EC2 Private IP

EC2 → Instances → private-ec2 → Private IPv4 address

Example:

10.0.2.40

Run SSH command from inside Public EC2 terminal

Inside your public EC2, run:

```
ubuntu@ip-10-0-2-40:~$ chmod 400 john.pem
ubuntu@ip-10-0-1-203:~$ ls -l john.pem
-r----- 1 ubuntu ubuntu 1678 Nov 14 15:20 john.pem
ubuntu@ip-10-0-1-203:~$ ssh -i "john.pem" ubuntu@10.0.2.40
Welcome to Ubuntu 24.04.3 LTS (GNU/Linux 6.14.0-1015-aws x86_64)

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

System information as of Fri Nov 14 15:27:41 UTC 2025

 System load:  0.08           Temperature:      -273.1 C
 Usage of /:   25.8% of 6.71GB  Processes:        109
 Memory usage: 24%            Users logged in:   0
 Swap usage:   0%             IPv4 address for ens5: 10.0.2.40

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-10-0-2-40:~$ |
```