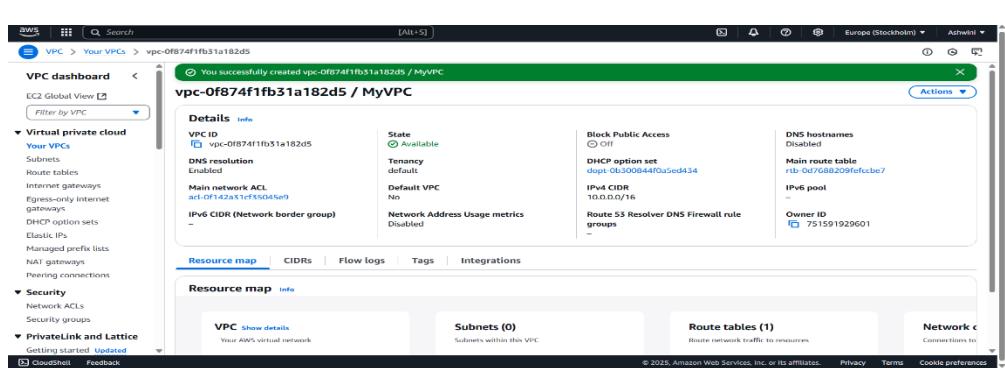


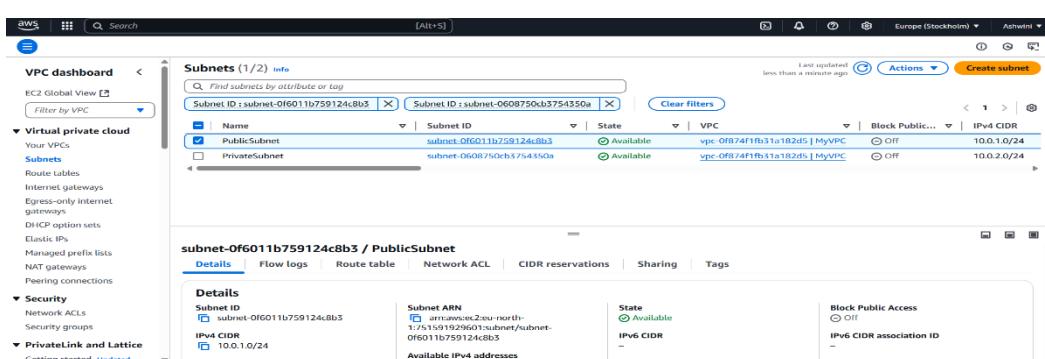
1. Create a VPC

- Go to **VPC Dashboard** → Your VPCs → Create VPC
- Choose:
 - **VPC only**
 - Name tag: MyVPC
 - IPv4 CIDR block: 10.0.0.0/16
- Click **Create VPC**



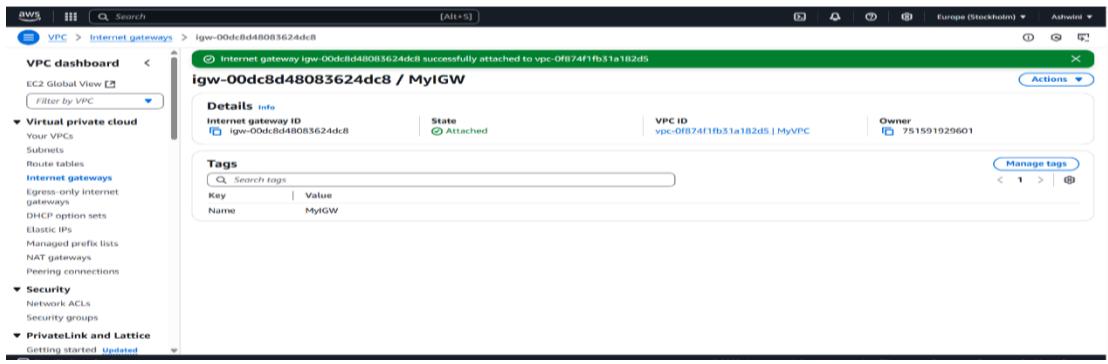
2. Create Two Subnets (Public and Private)

- Go to **Subnets** → Create subnet
- Choose:
 - **VPC**: MyVPC
 - Name: PublicSubnet
 - Availability Zone: (e.g., us-east-1a)
 - CIDR block: 10.0.1.0/24
- Click **Add new subnet**:
 - Name: PrivateSubnet
 - Availability Zone: (e.g., us-east-1a)
 - CIDR block: 10.0.2.0/24
- Click **Create subnet**



3.Create an Internet Gateway (IGW) and Attach to VPC

- Go to **Internet Gateways** → **Create internet gateway**
- Name: MyIGW
- Click **Create Internet Gateway**
- Click on it → **Actions** → **Attach to VPC** → Choose MyVPC



4. Create a Route Table and Associate It with the Public Subnet

- Go to **Route Tables** → **Create route table**
 - Name: PublicRouteTable
 - VPC: MyVPC
- Select the new route table → Go to **Routes** → **Edit routes**
 - Click **Add route**
 - Destination: 0.0.0.0/0
 - Target: Select **Internet Gateway**, choose MyIGW
 - Save changes
- Go to **Subnet associations** → **Edit subnet associations**
 - Select PublicSubnet
 - Save

🔍 Why only associate this with the public subnet?

- Only the **public subnet** needs internet access (via the IGW).
- The **private subnet** should not have direct internet access — it's usually for databases or internal services.

The screenshot shows the AWS VPC dashboard with the 'Route tables' section selected. A green success message at the top says 'Updated routes for rtb-Of326b33d23f57573 / PublicRouteTable successfully'. Below it, the 'rtb-Of326b33d23f57573 / PublicRouteTable' details are shown, including its ID, VPC association, and owner. The 'Routes' tab is selected, displaying two routes: one for 0.0.0.0/0 targeting the local gateway (igw-00dc8d48083624dc8) and another for 10.0.0.0/16 targeting the local subnet.

5. Create a Security Group for EC2

➤ Go to **Security Groups** → **Create security group**

- Name: MyEC2SG
- VPC: MyVPC

➤ In **Inbound rules**:

- Allow SSH (TCP port 22) from 0.0.0.0/0
- Optionally add HTTP (TCP port 80) from 0.0.0.0/0

➤ Click **Create security group**

The screenshot shows the AWS VPC dashboard with the 'Security groups' section selected. A green success message at the top says 'Security group (sg-0950e9d9a5931afa9 | MyEC2SG) was created successfully'. Below it, the 'sg-0950e9d9a5931afa9 - MyEC2SG' details are shown, including its name, ID, owner, and VPC association. The 'Inbound rules' tab is selected, showing a table with columns for Name, Security group rule ID, IP version, Type, Protocol, and Port range. The table is currently empty, indicating 'No security group rules found'.

6. Launch an EC2 Instance in the Public Subnet

➤ Go to **EC2 Dashboard** → **Instances** → **Launch instance**

➤ Choose:

- Name: MyEC2Instance
- AMI: Amazon Linux 2 (or any you prefer)
- Instance type: t2.micro (free tier eligible)
- Key pair: Select or create one (download it!)

➤ Under **Network settings**:

- VPC: MyVPC
- Subnet: PublicSubnet
- Auto-assign public IP: Enabled
- Security group: Choose MyEC2SG

➤ Click Launch Instance

The screenshot shows the AWS EC2 Instances page. On the left, there's a navigation sidebar with options like Dashboard, EC2 Global View, Events, Instances (selected), Instance Types, Launch Templates, Spot Requests, Savings Plans, Reserved Instances, Dedicated Hosts, Capacity Reservations, Images, AMIs, and AMI Catalog. Below that is the Elastic Block Store section with Volumes, Snapshots, and Lifecycle Manager.

The main content area displays the "Instance summary for i-05218a82fd4a8b232 (MyEC2Instance)". The summary includes:

- Instance ID:** i-05218a82fd4a8b232 (Updated less than a minute ago)
- IPV6 address:** -
- Hostname type:** IP name: ip-10-0-1-205.eu-north-1.compute.internal
- Answer private resource DNS name:** -
- Auto-assigned IP address:** 16.171.63.127 [Public IP]
- IAM Role:** -
- IMDSv2:** Required
- Operator:** -
- Public IPv4 address:** 16.171.63.127 | open address
- Instance state:** Running
- Private IP DNS name (IPv4 only):** ip-10-0-1-205.eu-north-1.compute.internal
- Instance type:** t3.micro
- VPC ID:** vpc-0f874f1fb31a182d5 (MyVPC)
- Subnet ID:** subnet-0f6011b759124c8b5 (PublicSubnet)
- Instance ARN:** arn:aws:ec2:eu-north-1:751501929601:instance/i-05218a82fd4a8b232
- Private IPv4 addresses:** 10.0.1.205
- Public IPv4 DNS:** -
- Elastic IP addresses:** -
- AWS Compute Optimizer finding:** Opt-in to AWS Compute Optimizer for recommendation
- Auto Scaling Group name:** -
- Managed:** false