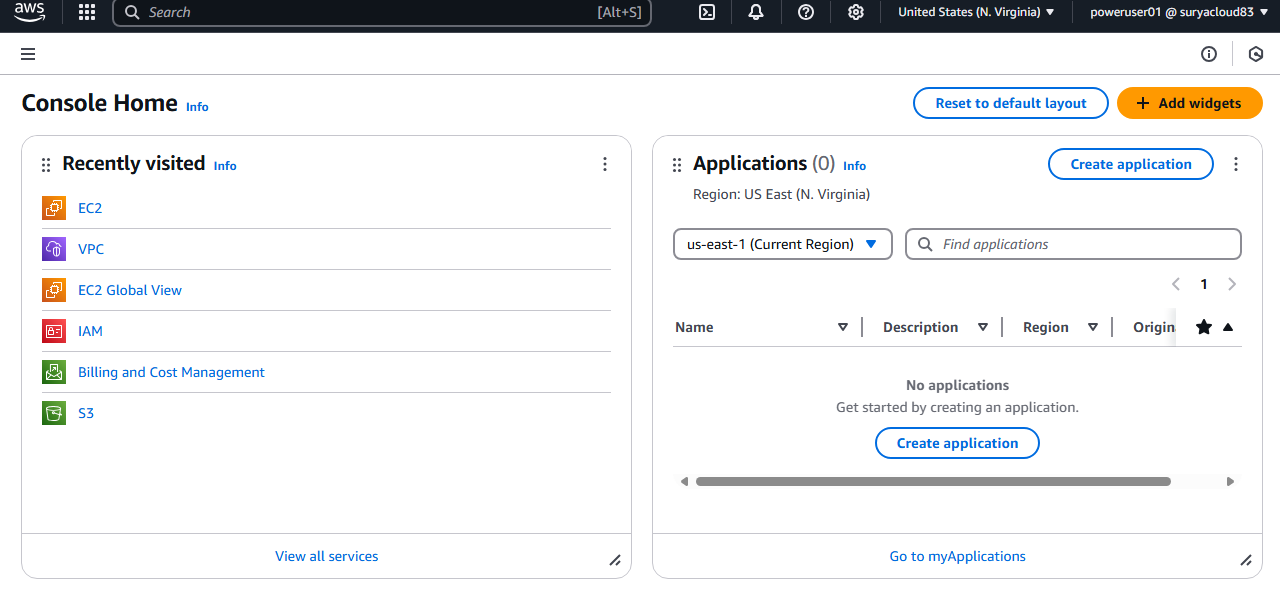
**Task Description:**

Set up a VPC with an Internet gateway, create a public subnet with 256 IP addresses, a private subnet with 256 IP addresses, make a route table connecting the Internet gateway and the subnets, and launch a Linux EC2 instance by using the above VPC and public subnet.

**Solution:**

**Step 1: Create a VPC**

1. Log in to the AWS Management Console.



1. Go to the VPC Dashboard. A screenshot of a computer

   AI-generated content may be incorrect.
2. Click on Create VPC.
   * Name tag: project-vpc
   * IPv4 CIDR block: 192.168.0.0/16 (This gives you 65,536 IP addresses; enough for multiple subnets).
   * IPv6 CIDR block: Optional (you can leave it empty if not using IPv6).
   * Tenancy: Default.

Click Create VPC.

A screenshot of a computer project

AI-generated content may be incorrect.

**Step 2: Create an Internet Gateway**

1. In the VPC Dashboard, go to Internet Gateways.
2. Click Create internet gateway.
   * Name tag: Outside World
3. After the gateway is created, select the new Internet Gateway and click Actions -> Attach to VPC.
4. Choose your newly created VPC (Project-vpc) and click Attach. A screenshot of a computer

   AI-generated content may be incorrect.

**Step 3: Create Subnets**

Create Public Subnet:

1. In the VPC Dashboard, go to Subnets.
2. Click Create subnet.
3. Select your VPC (Project-vpc).
4. Name tag: Public Net
5. Availability Zone: Choose any available zone (e.g., ap-south-2a).
6. IPv4 CIDR block: 192.168.1.0/24 (This provides 256 IP addresses).

Click Create subnet. A screenshot of a computer

AI-generated content may be incorrect.

Create Private Subnet:

1. Click Create subnet again.
2. Select your VPC (Project-vpc).
3. Name tag: Private Net
4. Availability Zone: Choose the same or another availability zone.
5. IPv4 CIDR block: 192.168.2.0/24 (This also provides 256 IP addresses).

Click Create subnet. A screenshot of a computer

AI-generated content may be incorrect.

**Step 4: Create Route Tables**

**Create Route Table for Public Subnet:**

1. Go to Route Tables in the VPC Dashboard and click Create route table.
   * Name tag: Internet Access Route Table
   * VPC: Choose Project-vpc.
   * Click Create

A screenshot of a computer

AI-generated content may be incorrect.

1. After the route table is created, select it, go to the Routes tab, and click Edit routes.
2. Add a route to allow internet access:
   * Destination: 0.0.0.0/0
   * Target: Choose the Internet Gateway (MyInternetGateway).
3. Click Save routes. A screenshot of a computer

   AI-generated content may be incorrect.
4. Now, associate this route table with the Public Net:
   * In the Subnet Associations tab, click Edit subnet associations.
   * Select Public Net and click Save.

**Step 5: Launch EC2 Instance in the Public Subnet**

1. Go to the EC2 Dashboard and click Launch Instance.
2. Choose an AMI: Select Amazon Linux 2 or any other Linux-based AMI.
3. Choose Instance Type: Select an instance type (e.g., t3.micro for testing).
4. Configure Instance:
   * Network: Choose Project-vpc.
   * Subnet: Choose Public Net.
   * Auto-assign Public IP: Ensure this is Enable to assign a public IP to the instance.
   * IAM Role: If needed, select a role for the instance (for this example, you can leave it blank).
5. Add Storage: The default is usually fine.
6. Add Tags: Optionally add a name tag (e.g., MyInstance).
7. Configure Security Group:
   * Create a new security group or use an existing one.
   * Add rules to allow:
     + SSH (port 22) from your IP address for SSH access.
     + Optionally, HTTP (port 80) if you want web access.
8. Review and Launch: Review your settings and click Launch.
9. Select a Key Pair for SSH access and click Launch Instances.

**A screenshot of a computer

AI-generated content may be incorrect.**

**Step 6: Access the EC2 Instance**

* Once the EC2 instance is running, you can SSH into it:
  1. Find the Public IP of the instance on the EC2 Dashboard.
  2. Use the following SSH command:

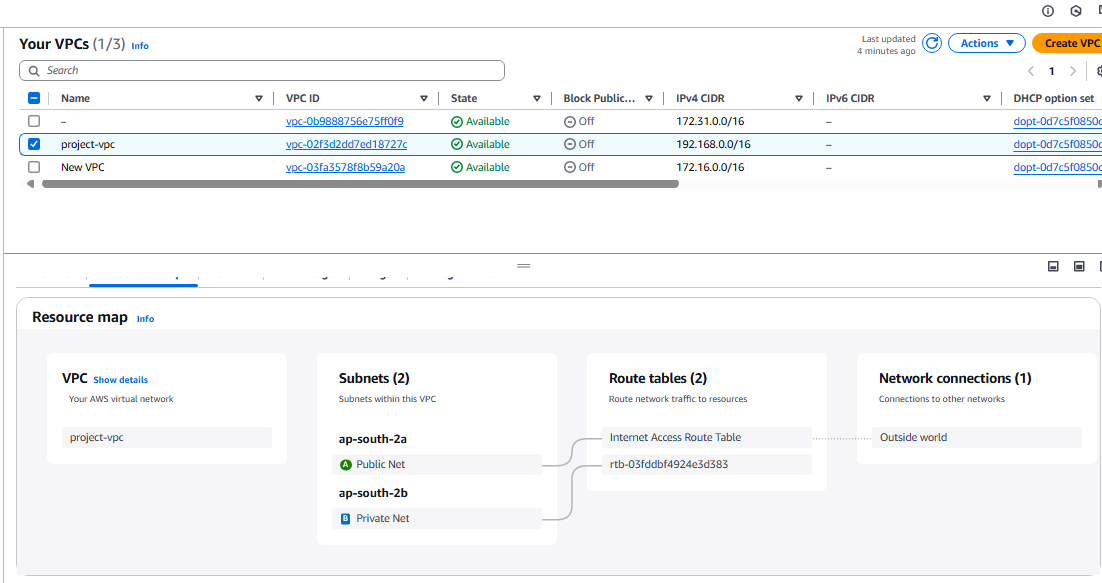
bash

Copy

ssh -i /path/to/your-key.pem ec2-user@<Public\_IP\_of\_instance>

**A screenshot of a computer

AI-generated content may be incorrect.**

**OVERVIEW MAP of VPC:  
**