**cloud computing**

**Lab 08**

**Name:** Abiha Nadeem

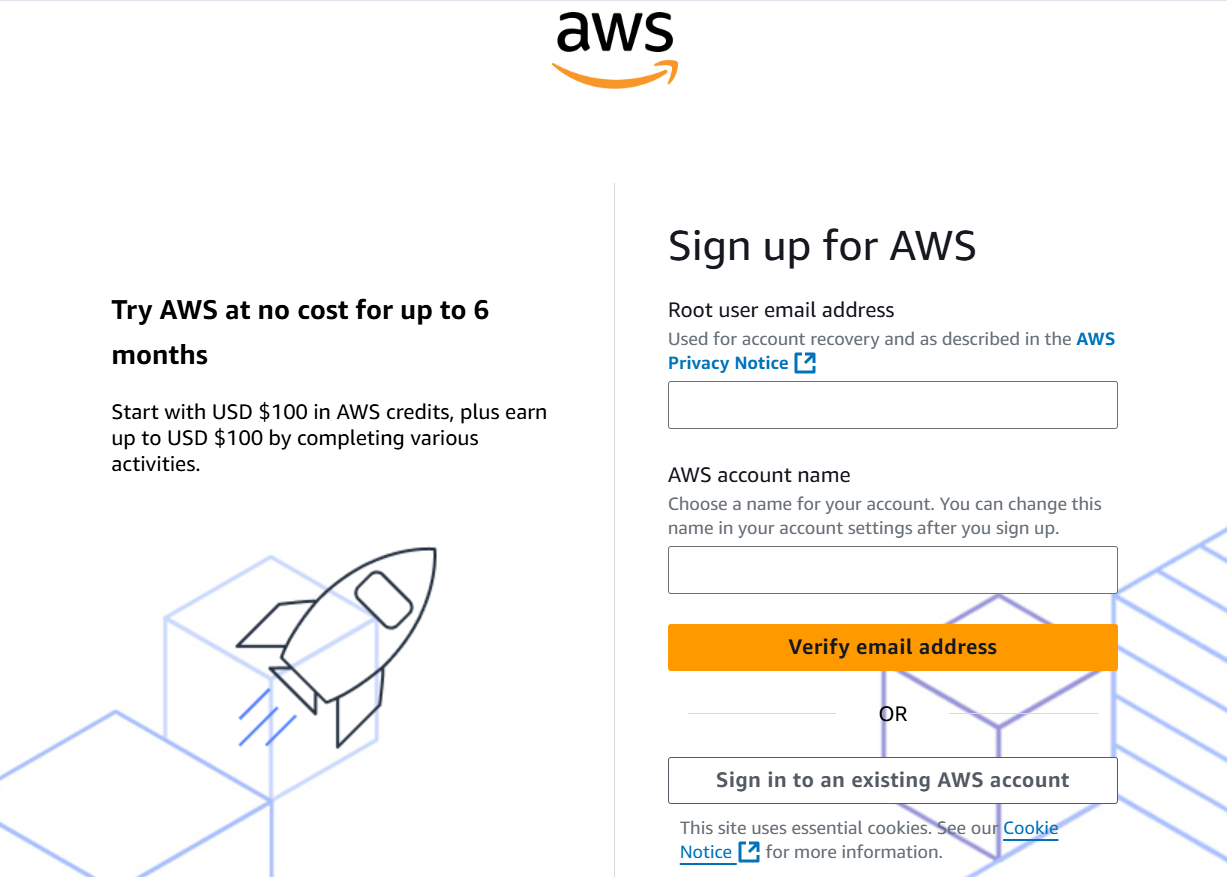
**Roll no:** 2023-BSE-001

**Submitted to:** Engr. Muhammad Shoaib

**AWS: Account Setup, IAM, VPC Inventory, EC2, Docker & Gitea**

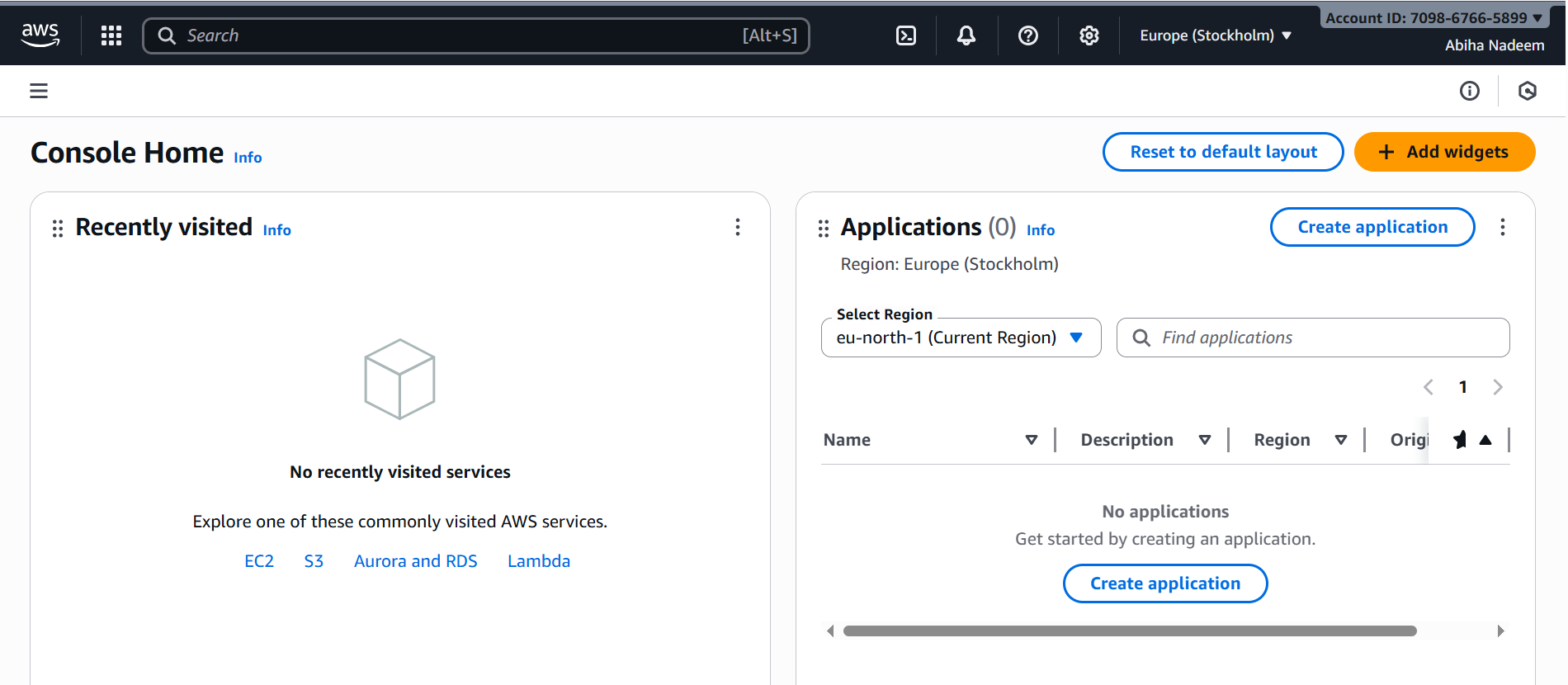
**Task 1 — Create an AWS account and enable UAE (me-central-1)**

1. **Open your browser and go to:**[**AWS Signup**](https://signin.aws.amazon.com/signup?request_type=register)

****

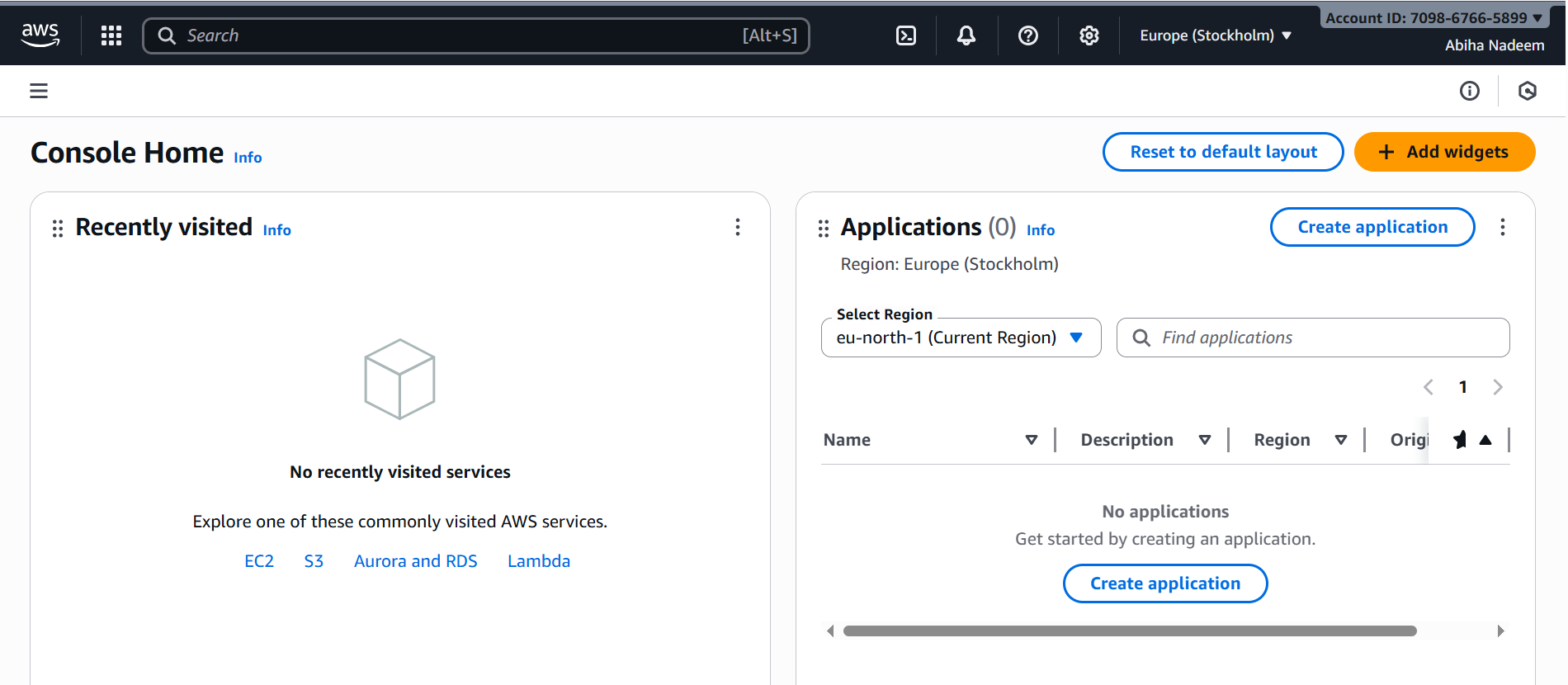
**Save screenshot as: task1\_open\_signup\_page.png — browser showing the signup page.**

1. **Complete registration (Account type: Personal, Plan: AWS Paid Plan), fill contact, billing (credit card) and phone details, complete verification. After successful registration capture:**

****

**Save screenshot as: task1\_signed\_up\_confirmation.png — registration success/confirmation page or payment confirmation (do NOT include credit card full details).**

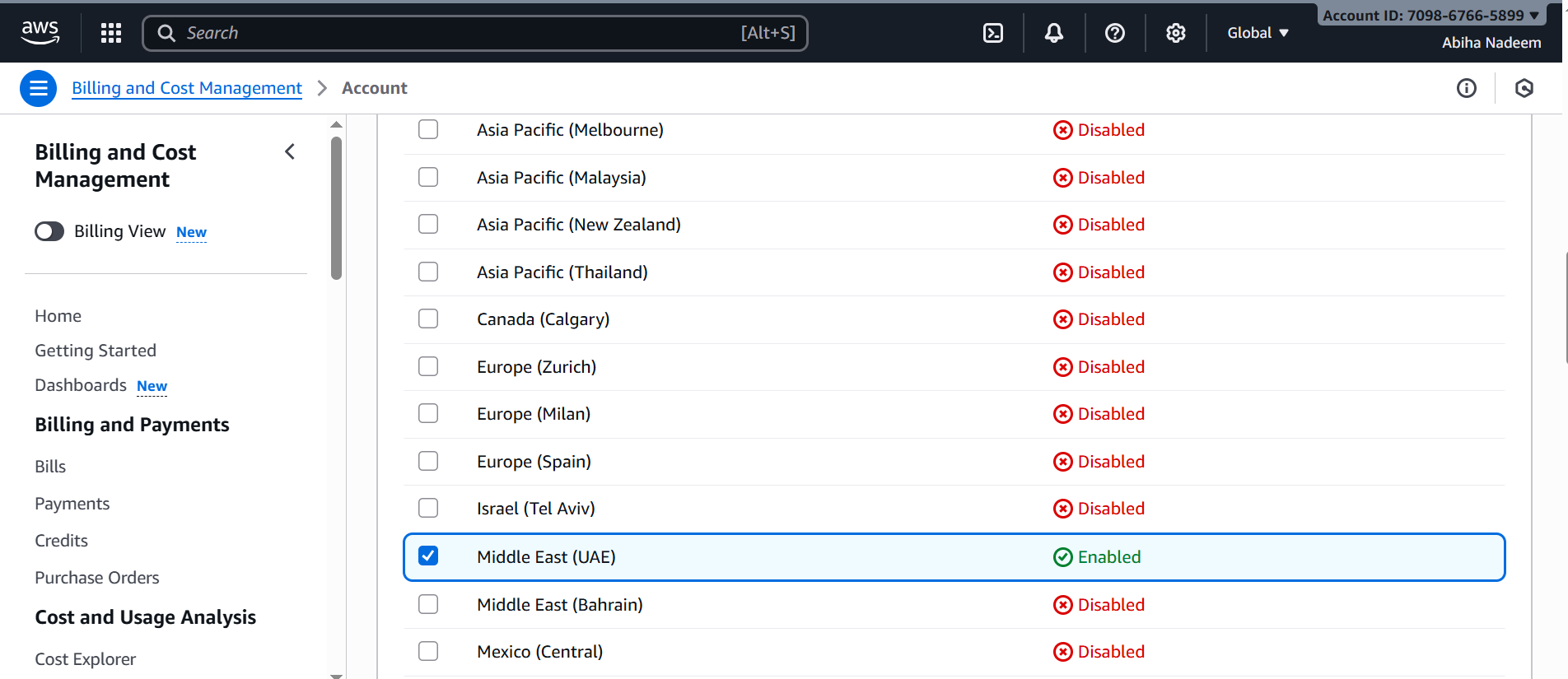
1. **Sign in as the root user (root email). Immediately capture:**

****

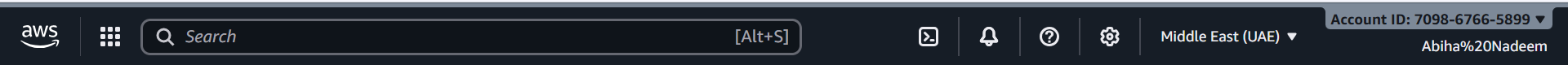
**Save screenshot as: task1\_root\_signed\_in.png — AWS Console Home after root login (top bar with root email/account alias visible).**

1. **From the Console, open the region selector and enable UAE (me-central-1), then switch to me-central-1. Capture the change:**

**Save screenshot as: task1\_enable\_region\_me-central-1.png — region selector showing me-central-1 selected.**

****

1. **Task 1 summary screenshot (combine evidence):**

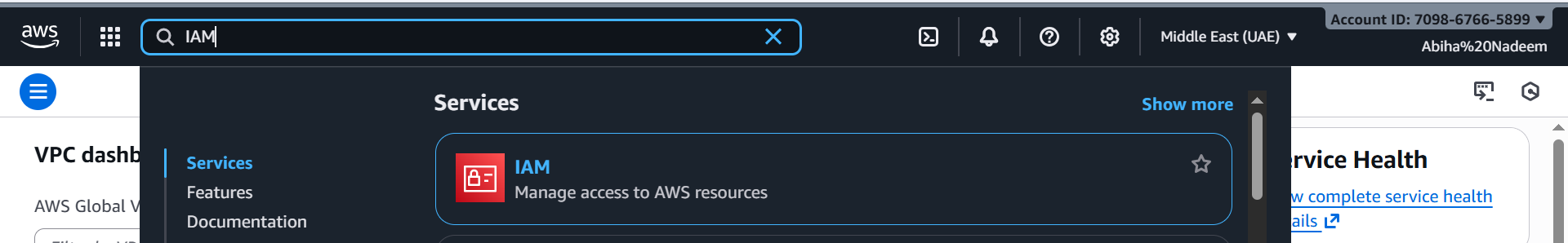
****

**Save screenshot as: task1\_summary.png — single screenshot showing root console header (root email/account alias) and region set to me-central-1.**

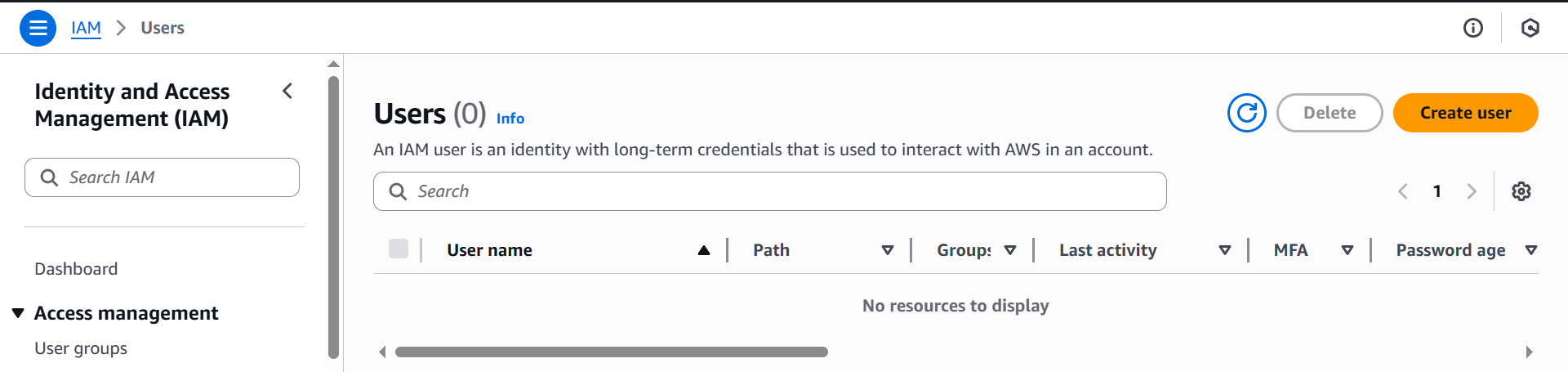
**Task 2 — Create IAM Admin and Lab8User with console access**

1. **Open IAM via Console search (Alt+S → "IAM").**

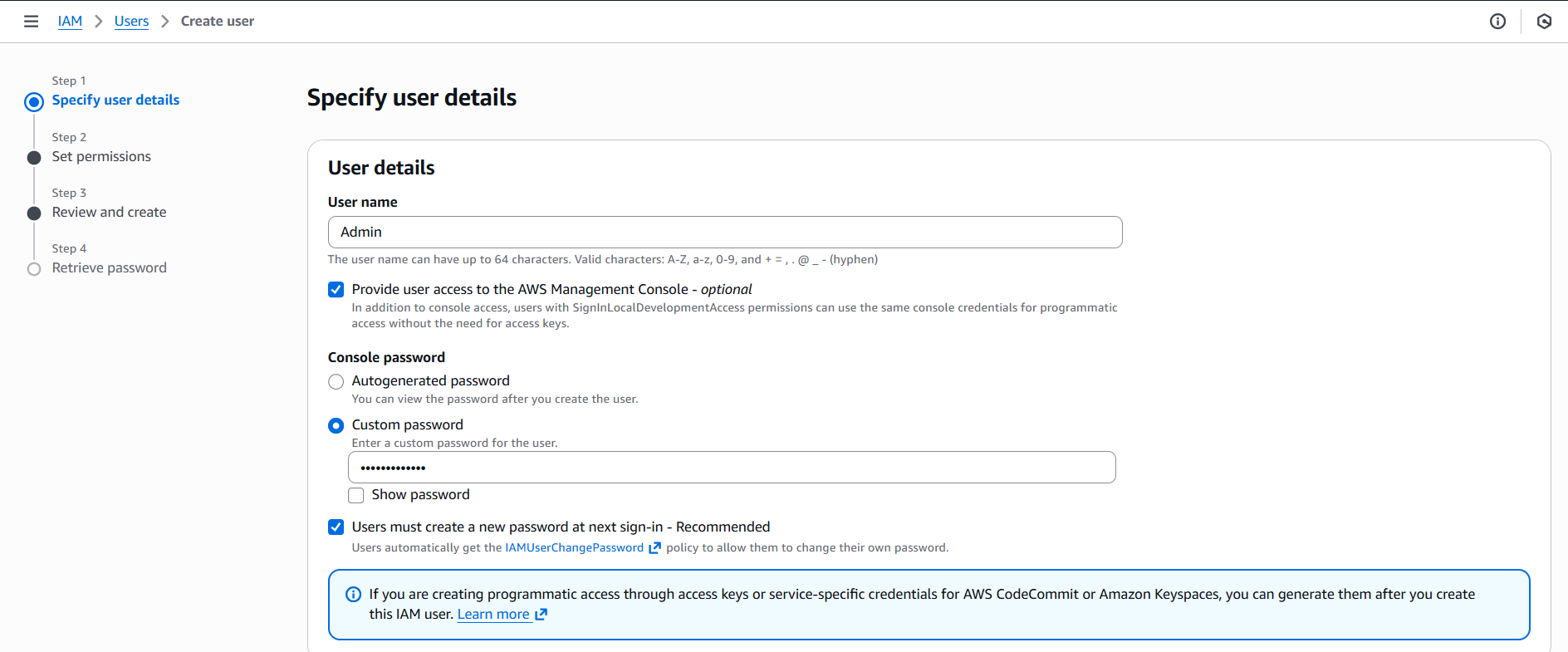
**Save screenshot as: task2\_open\_iam\_console.png — IAM console landing page (region me-central-1 visible).**

****

1. **Create the Admin user: IAM → Users → Create user. Fill:**

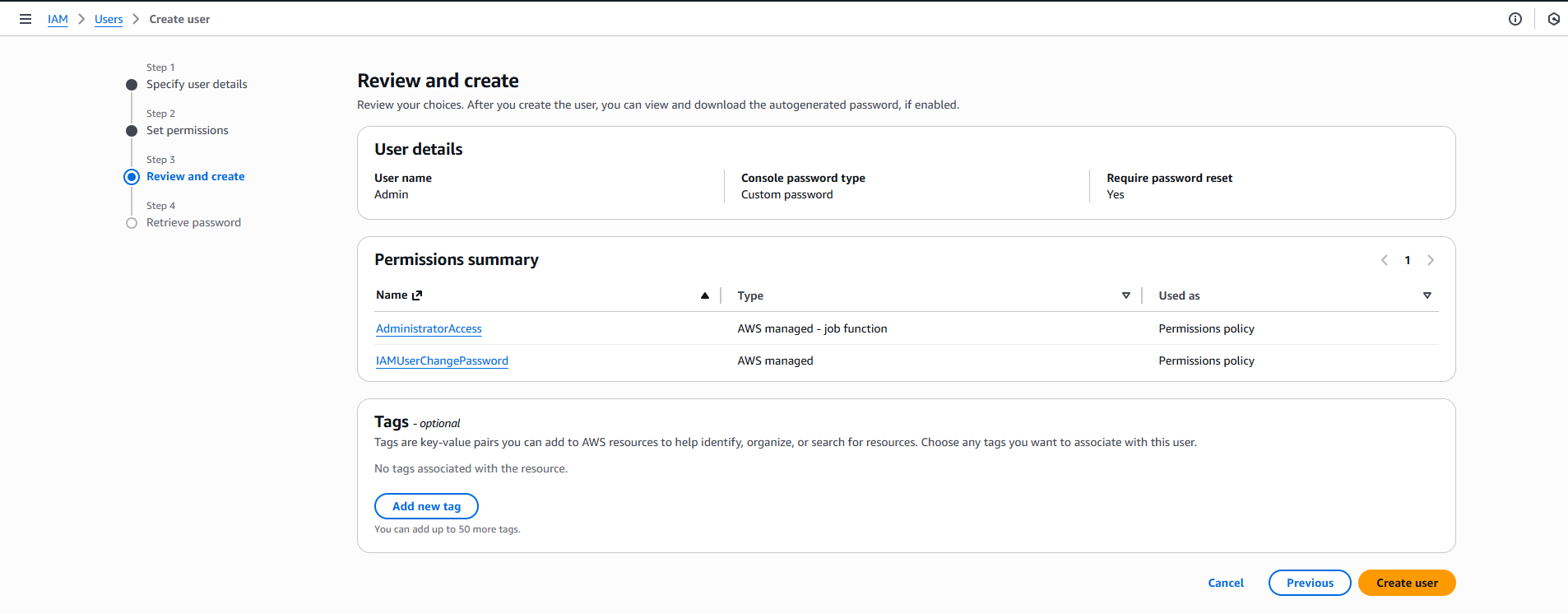
****

* Username: Admin
* Provide user access to the AWS Management Console
* Set console password (autogenerate or set)

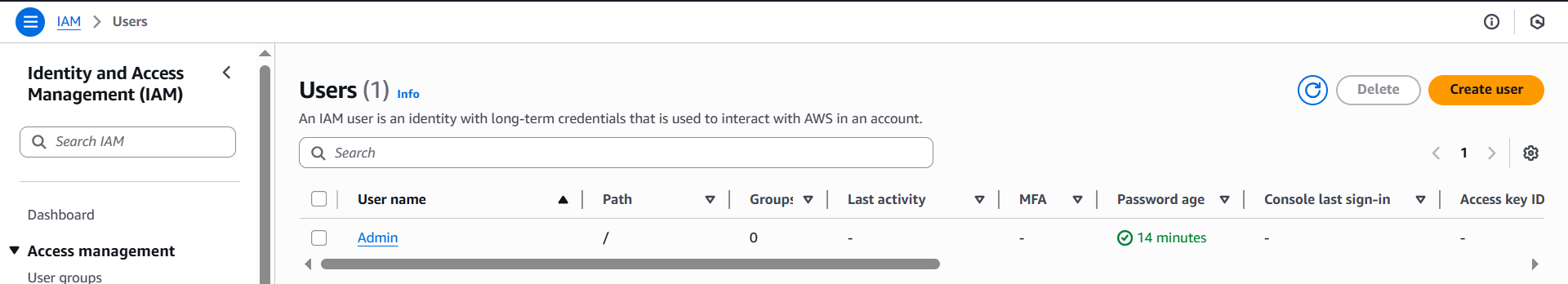


* Attach policies directly → AdministratorAccess





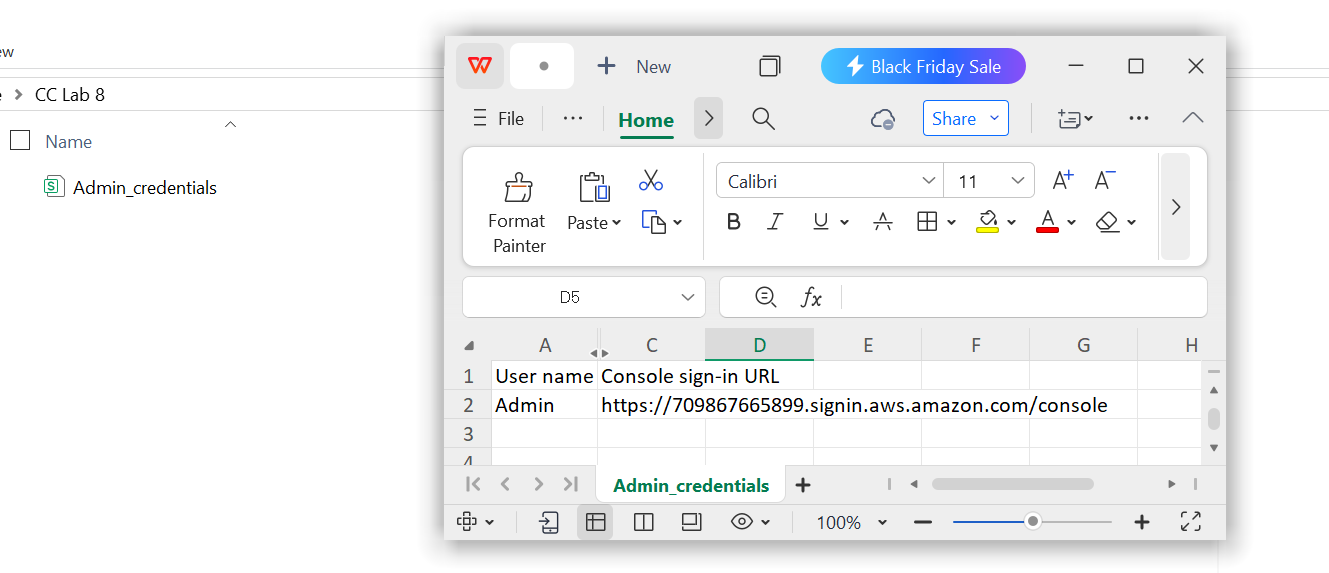
* Capture the completion screen when user is created:



**Save screenshot as: task2\_admin\_create\_confirmation.png — IAM "Create user" success screen showing Admin (do NOT include password).**

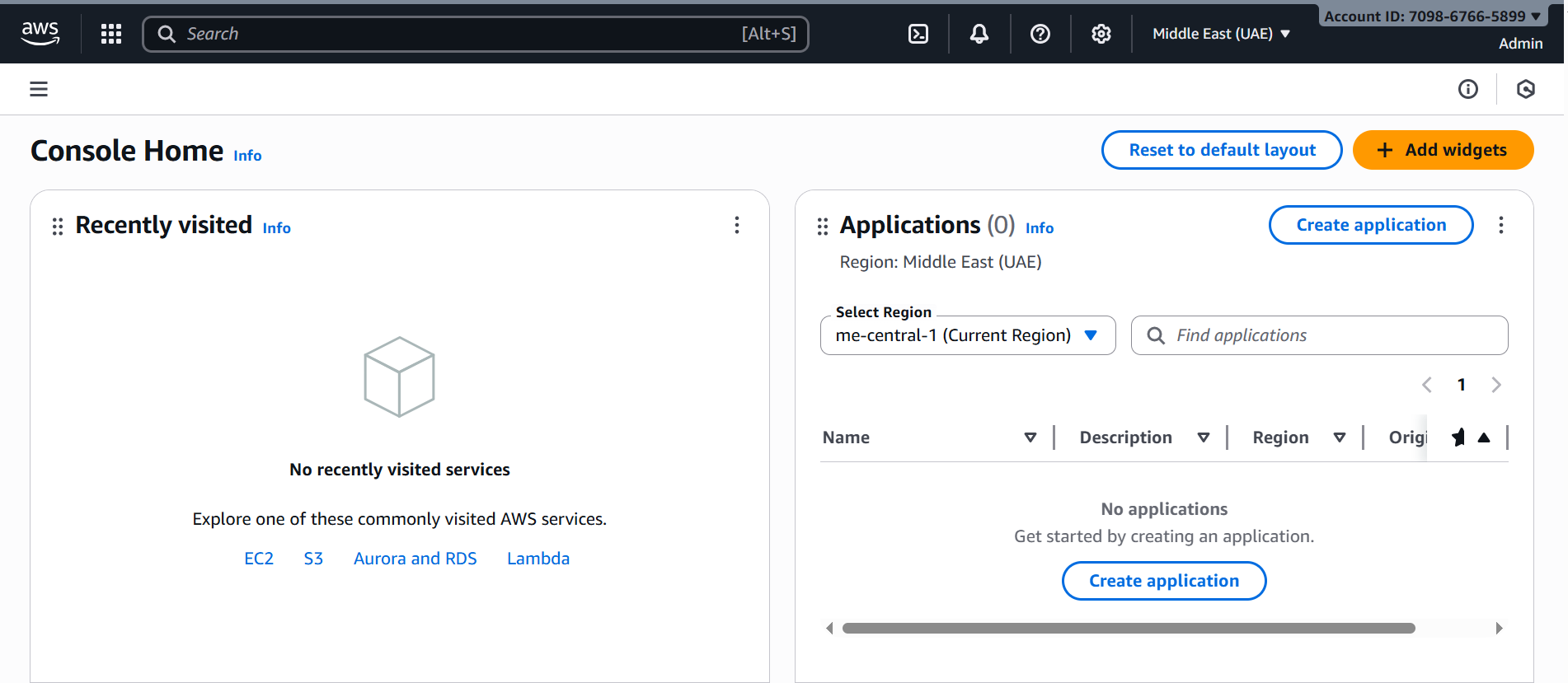
1. **Download the Admin .csv and show its presence on your Windows host (do not display the password text):**

**Save screenshot as: task2\_admin\_csv\_and\_signin\_url.png — Windows File Explorer showing the downloaded CSV filename and/or a cropped view of the CSV showing only the Sign-in URL and username (redact the password if visible).**

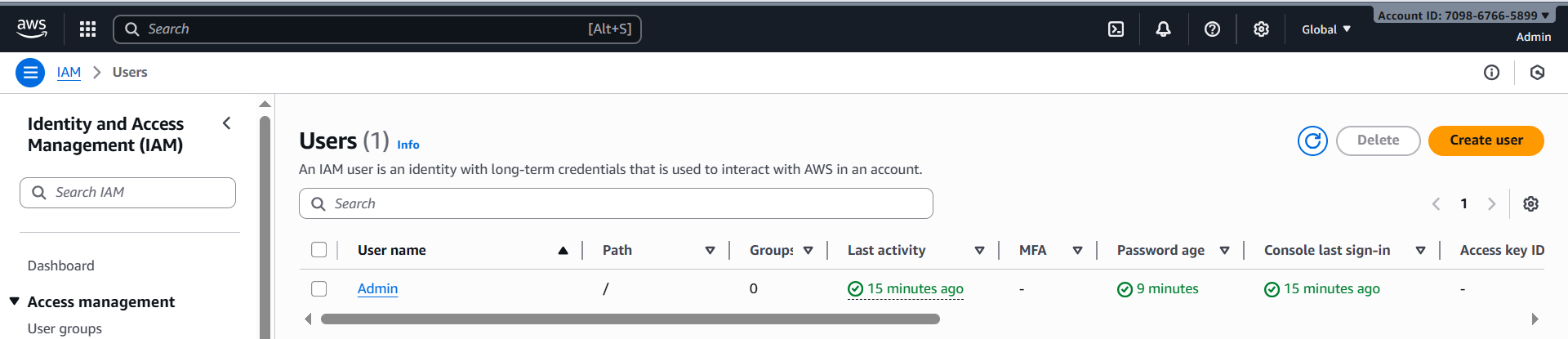


1. **Sign out of root, then sign in using the Admin account (use the signin URL from the .csv). Capture after successful Admin login:**

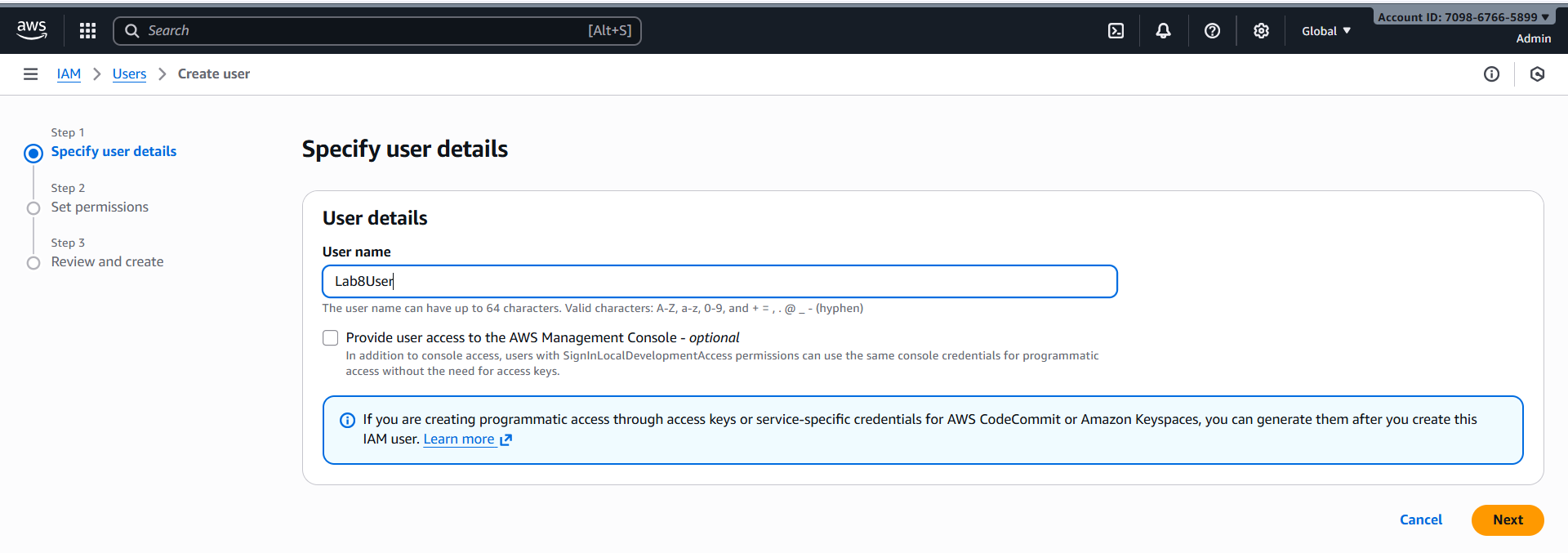
**Save screenshot as: task2\_admin\_console\_after\_login.png — Admin user console home.**

****

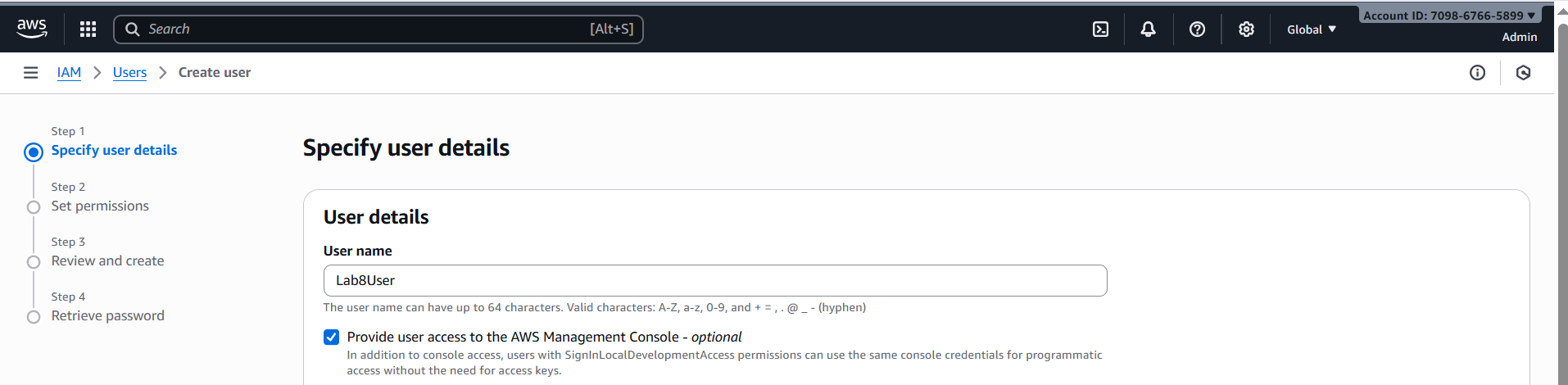
1. **While logged in as Admin, create Lab8User:**
   1. IAM → Users → Create user



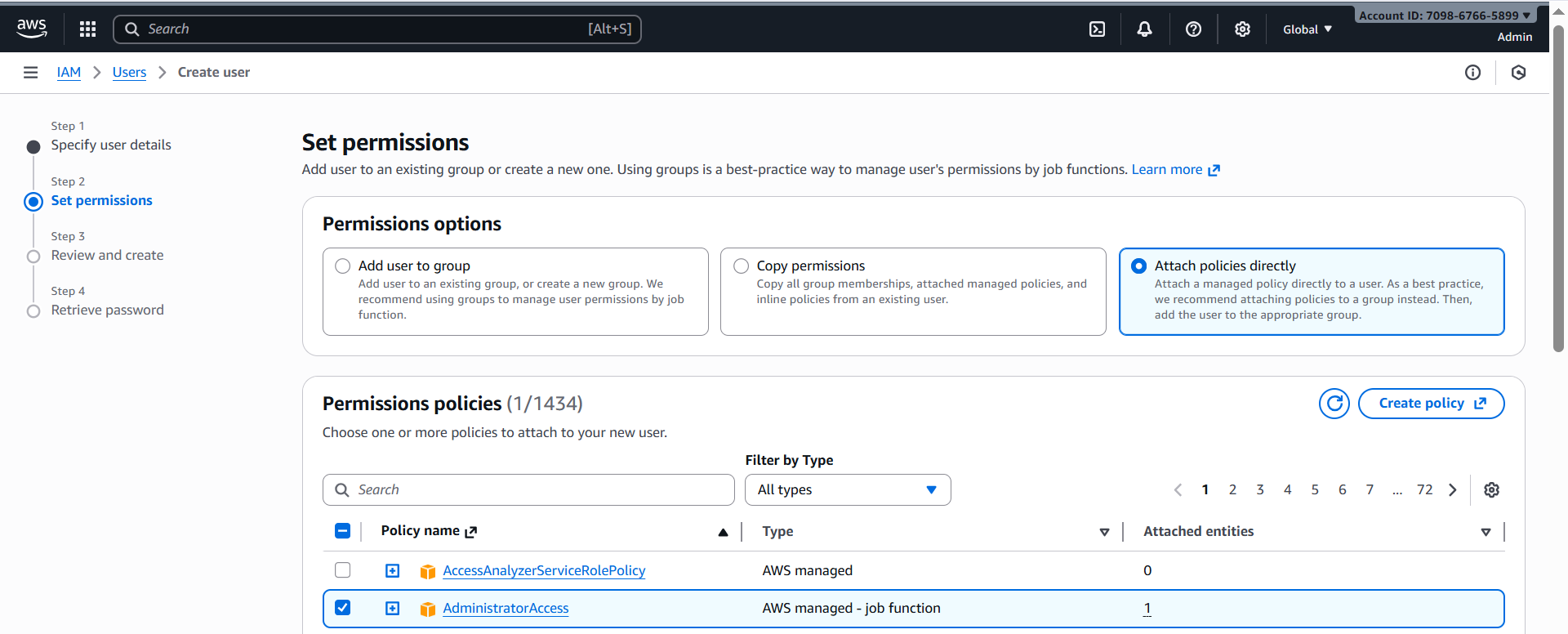
* 1. Username: Lab8User



* 1. Provide user access to the AWS Management Console

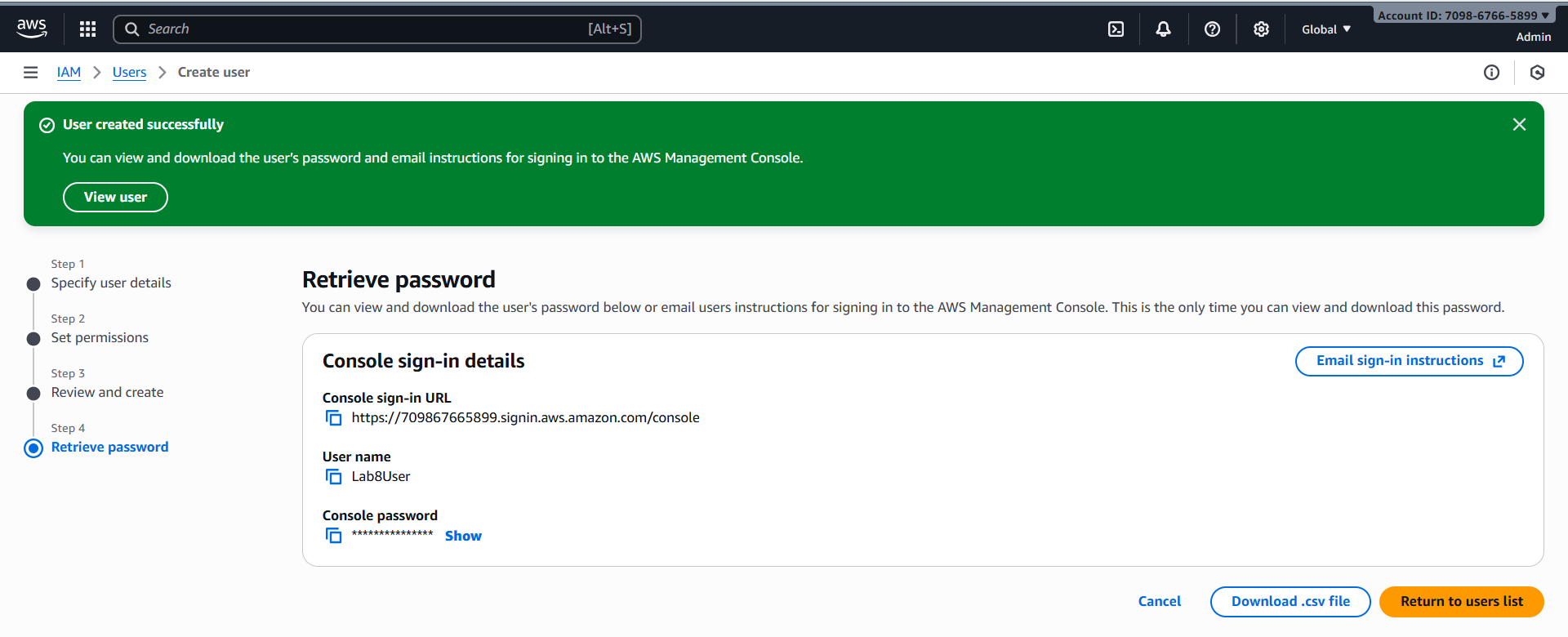


* 1. Attach AdministratorAccess policy



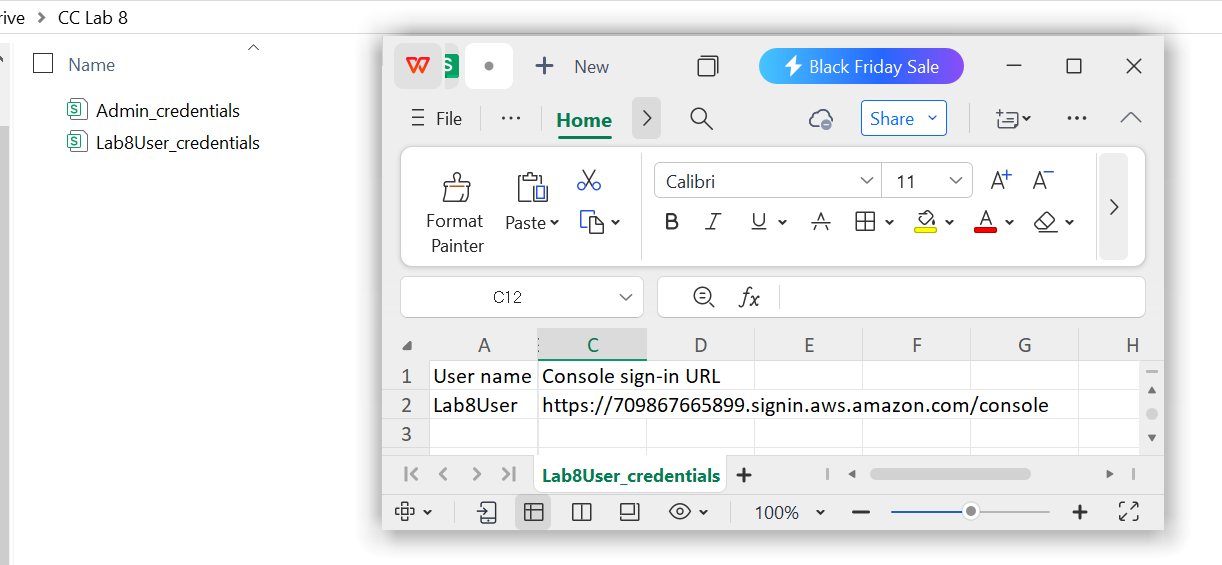
Capture the create-user success screen:

**Save screenshot as: task2\_create\_lab8user\_and\_csv.png — Lab8User create confirmation and CSV download prompt (do NOT include password).**

****

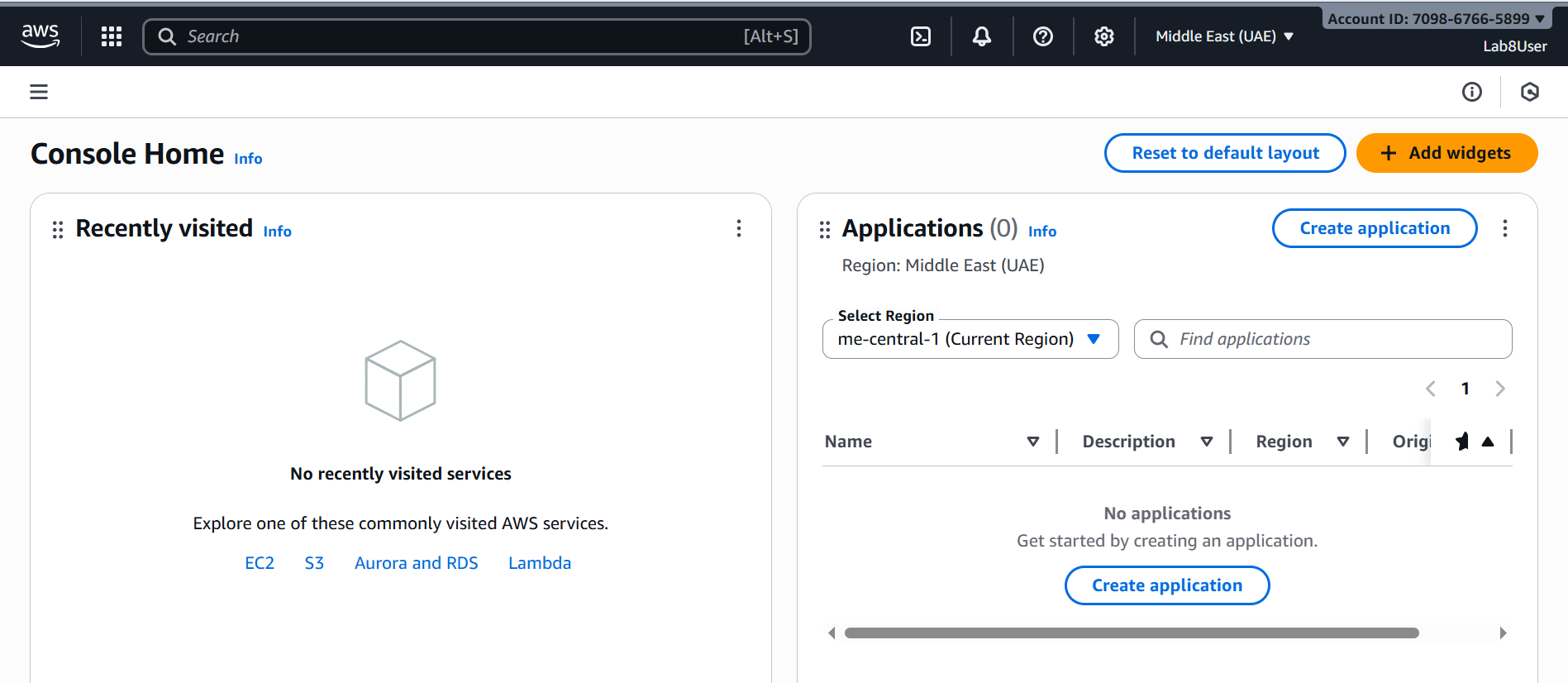
1. **Download/save the Lab8User CSV on your Windows host (do not show password).**

**Save screenshot as: task2\_lab8user\_csv\_saved.png — File Explorer showing the Lab8User CSV filename (cropped to exclude sensitive content).**

****

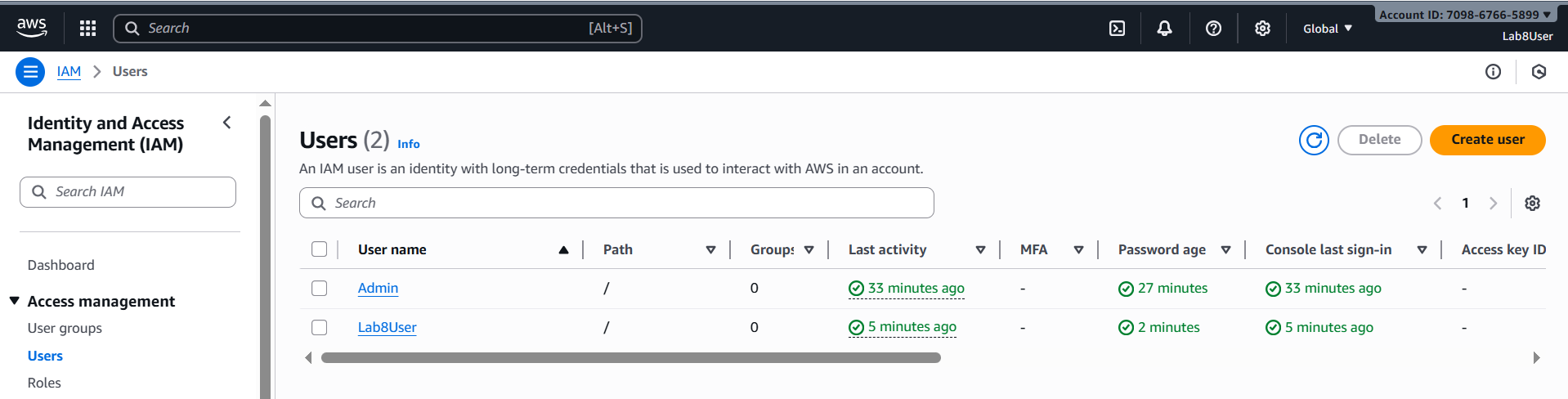
1. **Logout Admin and login as Lab8User (use the Lab8User signin URL and credentials). Capture after login:**

**Save screenshot as: task2\_lab8user\_logged\_in.png — Lab8User console home.**

****

1. **Task 2 summary (combine evidence):**

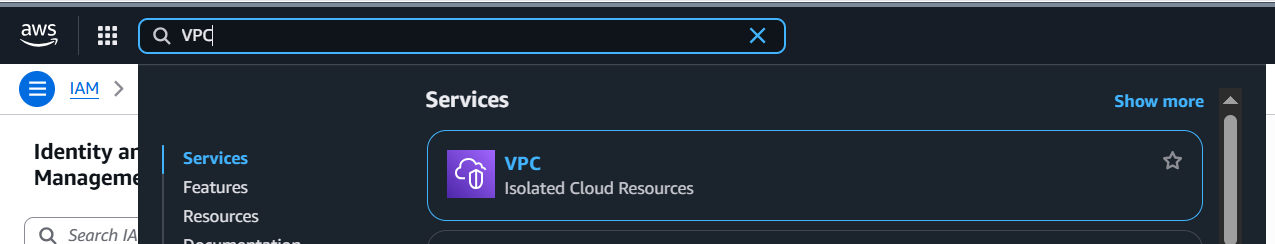
**Save screenshot as: task2\_summary.png — IAM Users list showing both Admin and Lab8User present (region me-central-1 visible).**

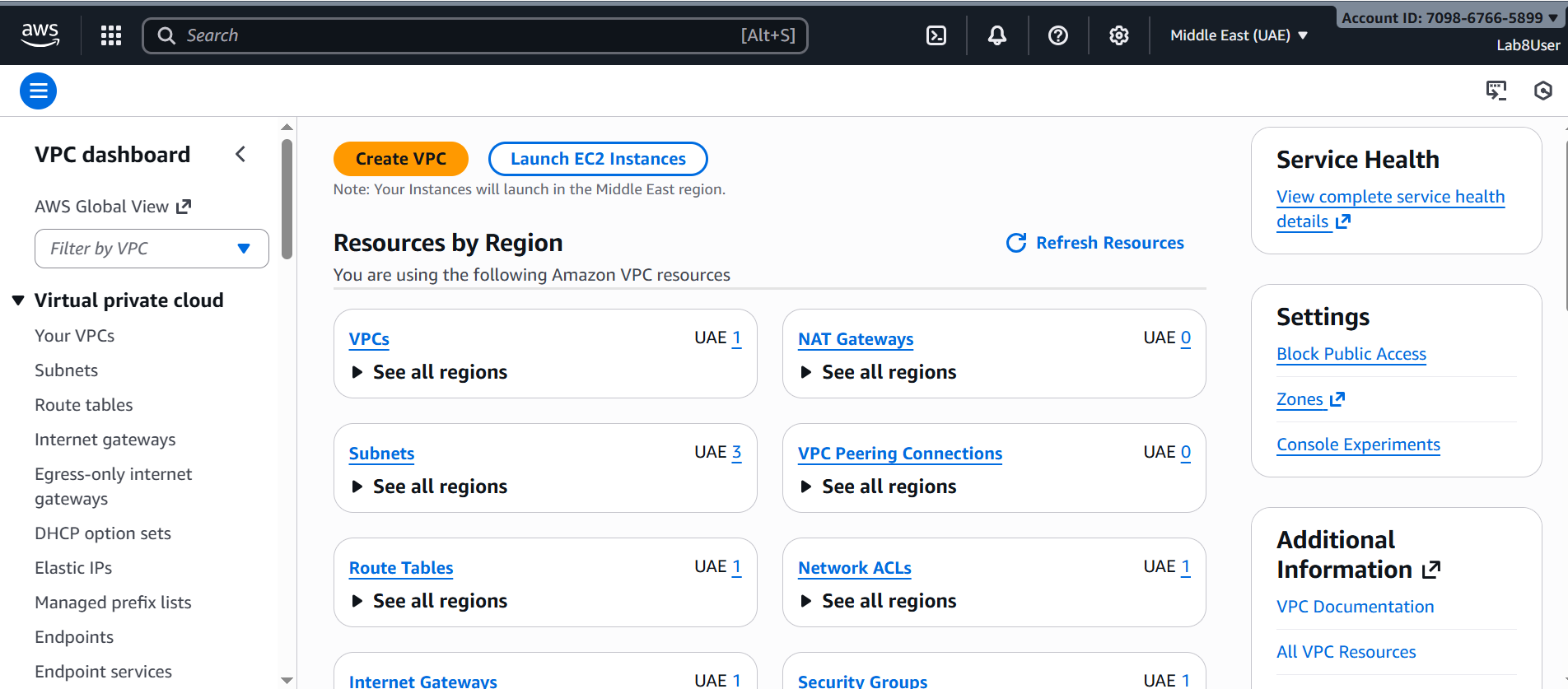
****

**Task 3 — Inspect VPC resources (in UAE me-central-1)**

1. **Open VPC console (Alt+S → "VPC") while region is me-central-1.**

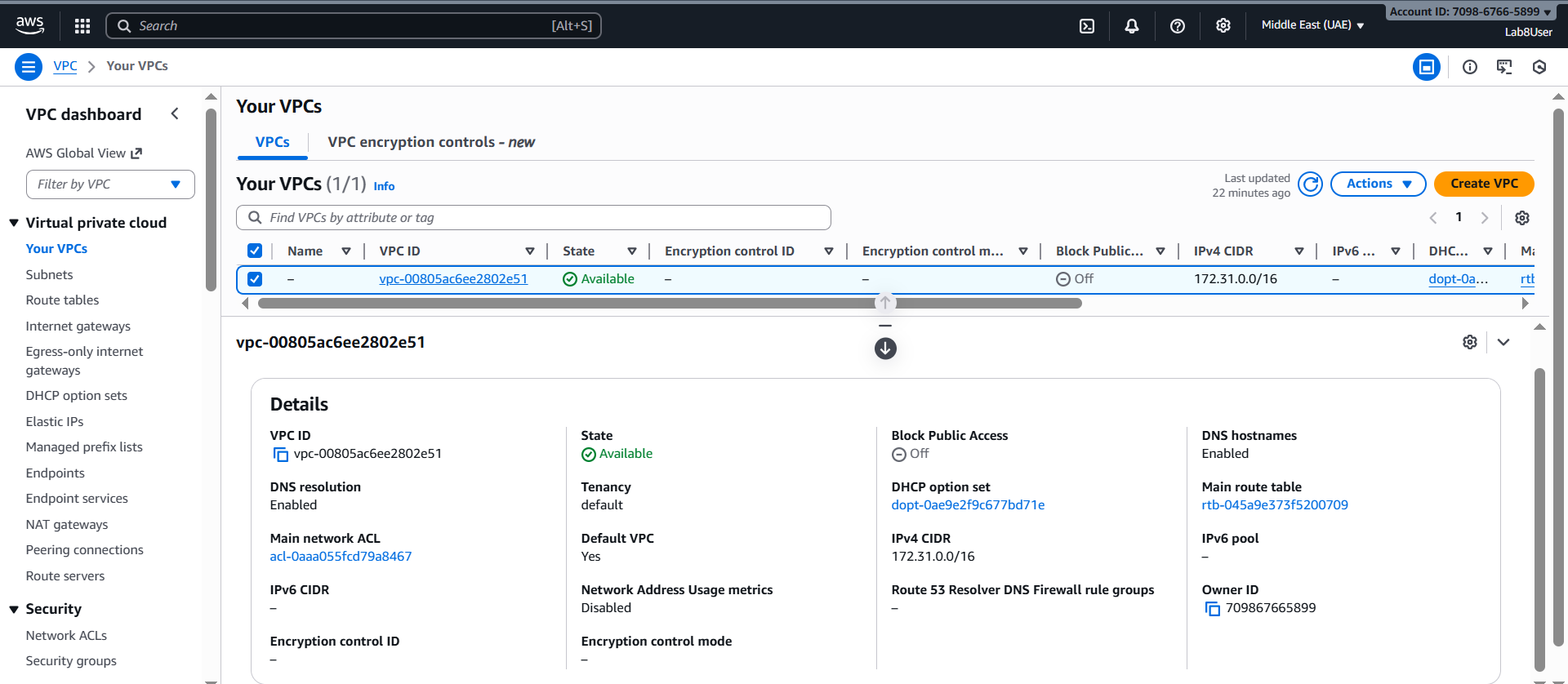
**Save screenshot as: task3\_open\_vpc\_console.png — VPC console landing page (region visible).**

****

****

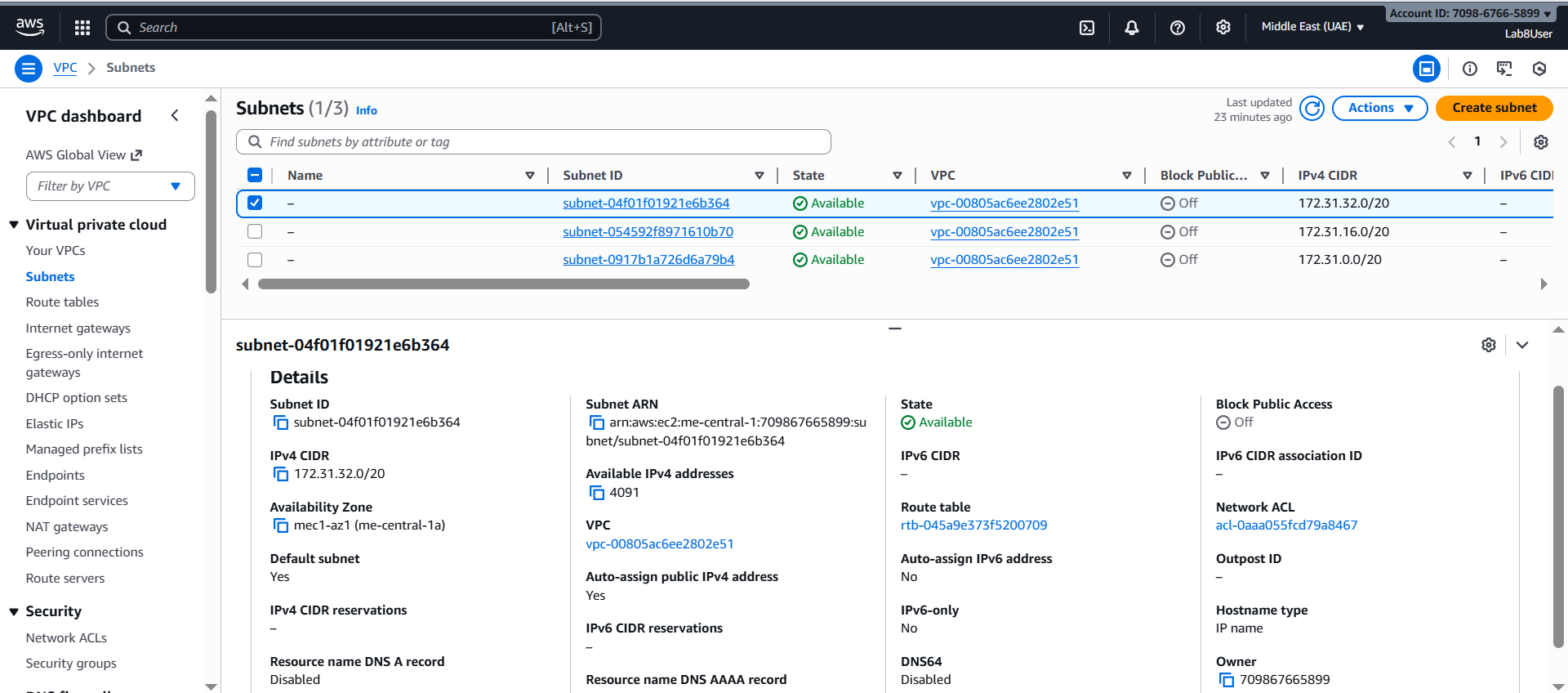
1. **View VPCs list. Capture:**

**Save screenshot as: task3\_vpcs\_list.png — VPCs list view (show default VPC if present).**

****

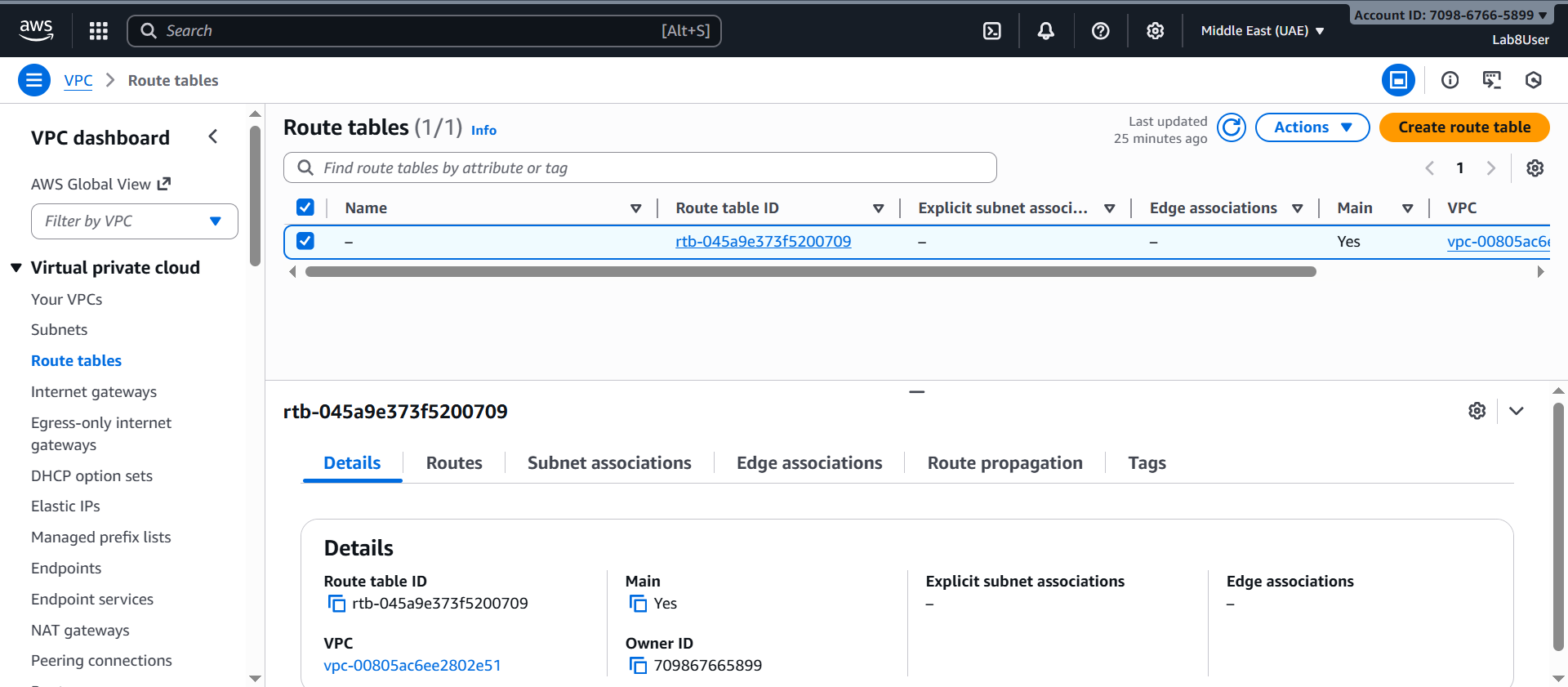
1. **View Subnets list. Capture:**

**Save screenshot as: task3\_subnets\_list.png — Subnets list view (show at least 3 default subnets if present).**

****

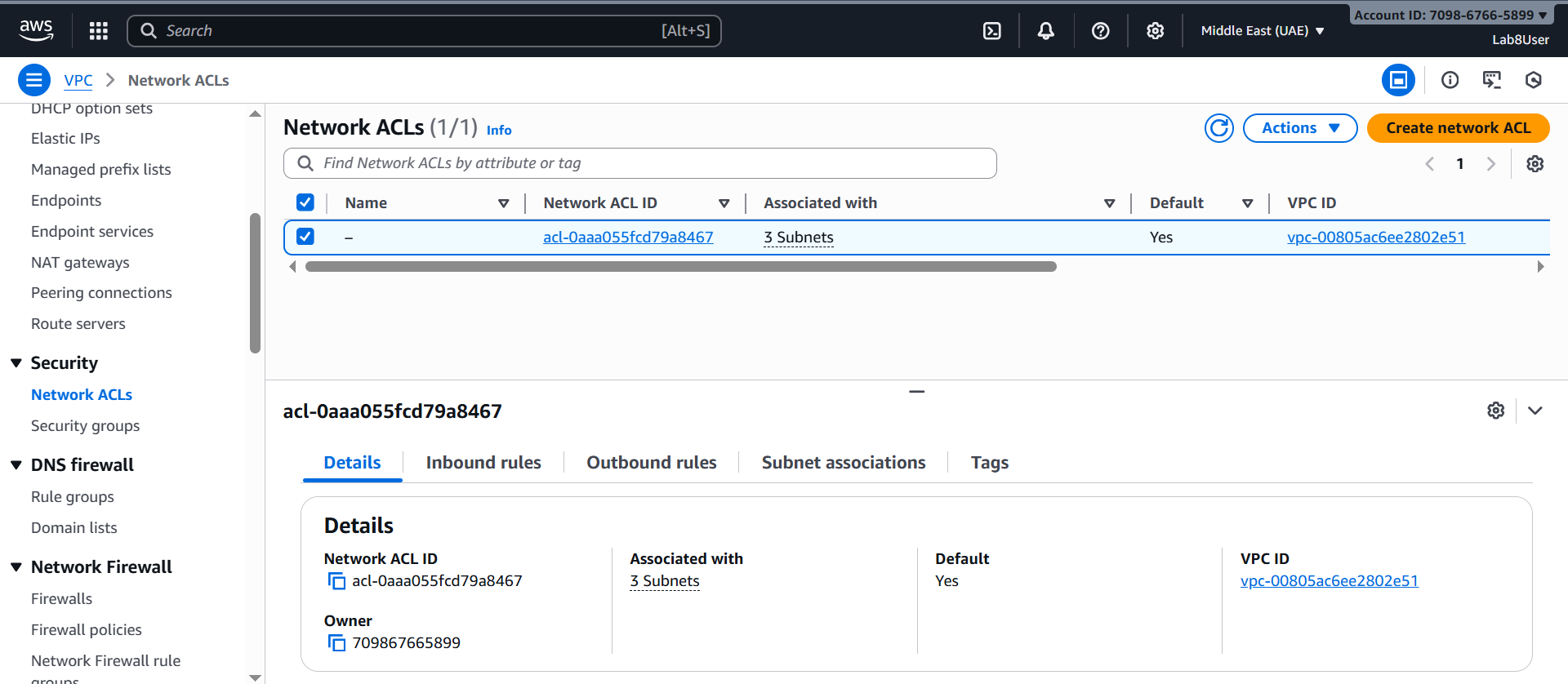
1. **View Route Tables list. Capture:**

**Save screenshot as: task3\_route\_tables\_list.png — Route Tables list view.**

****

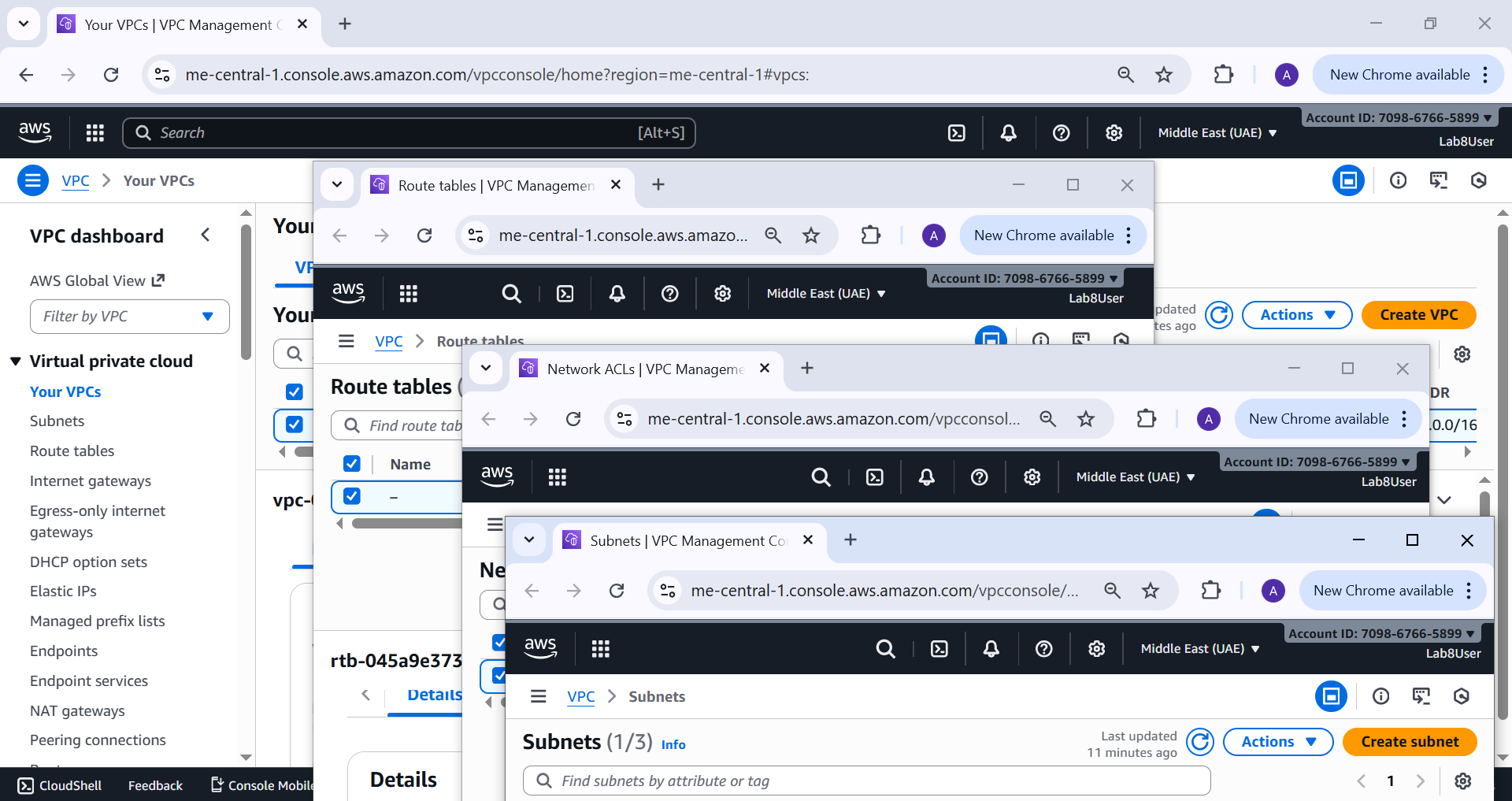
1. **View Network ACLs list. Capture:**

**Save screenshot as: task3\_network\_acls\_list.png — Network ACLs list view.**

****

1. **Task 3 summary (combine evidence):**

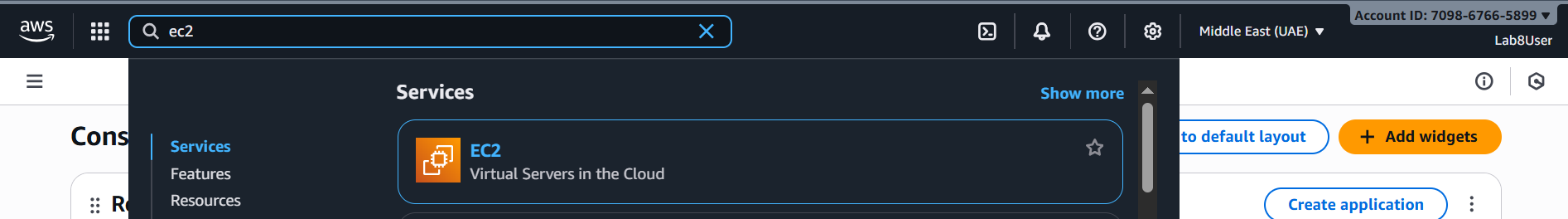
**Save screenshot as: task3\_summary.png — a single screenshot showing the VPC console left navigation and counts or multiple open tabs/windows tiled to show each resource's list (region me-central-1 visible).**

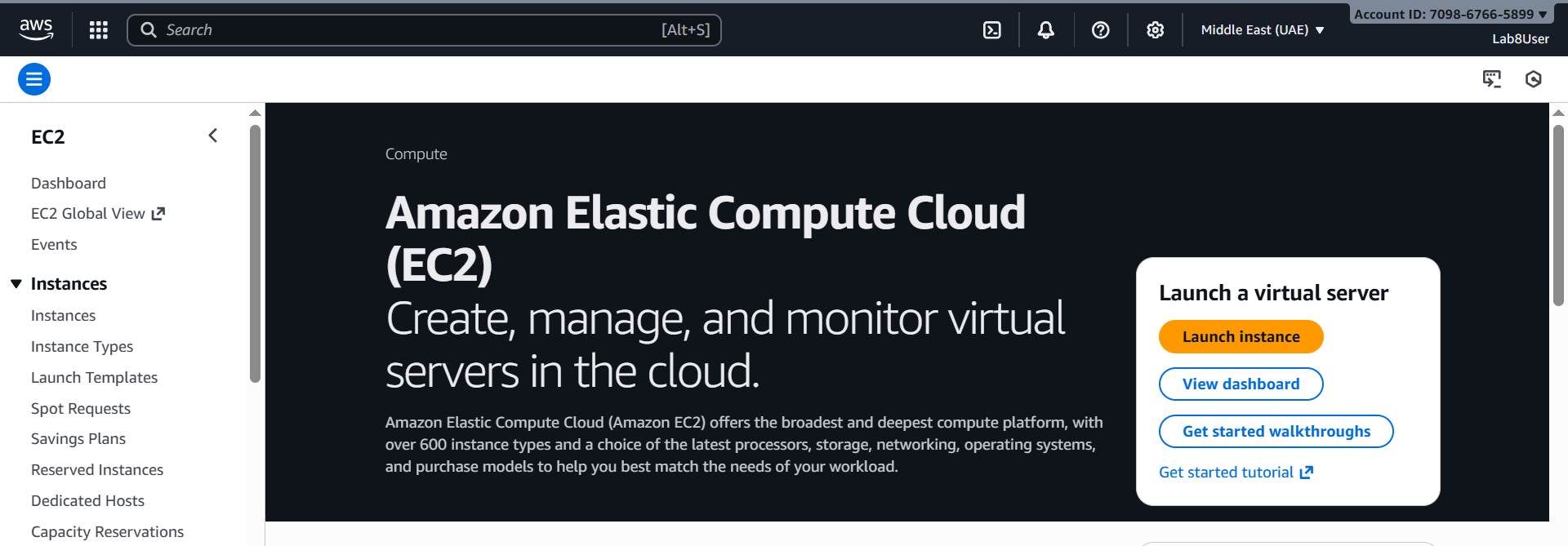
****

**Task 4 — Launch EC2, SSH, install Docker & Docker Compose, deploy Gitea**

1. **Open EC2 Console (Alt+S → "EC2") (me-central-1).**

**Save screenshot as: task4\_open\_ec2\_console.png — EC2 console landing page with region visible.**

****

****

1. **Instance Launch configuration (during review before launching). Configure:**

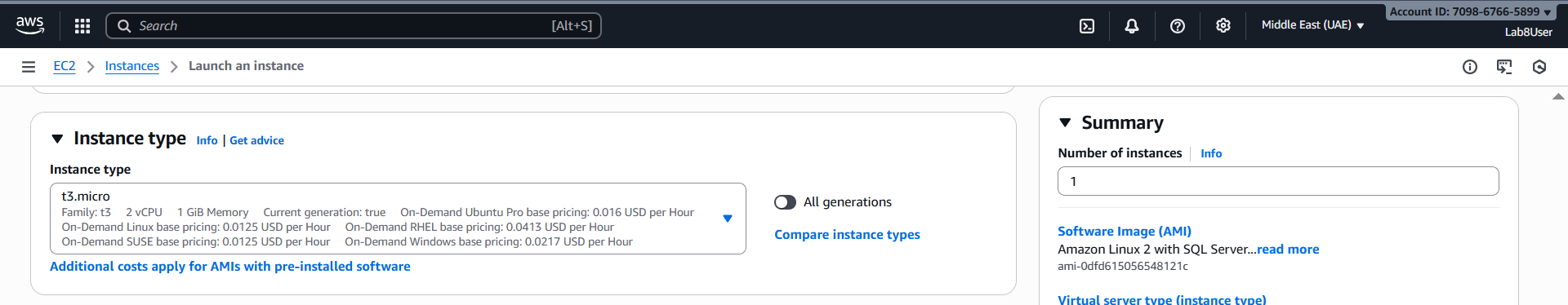
* **Name: Lab8Machine**

****

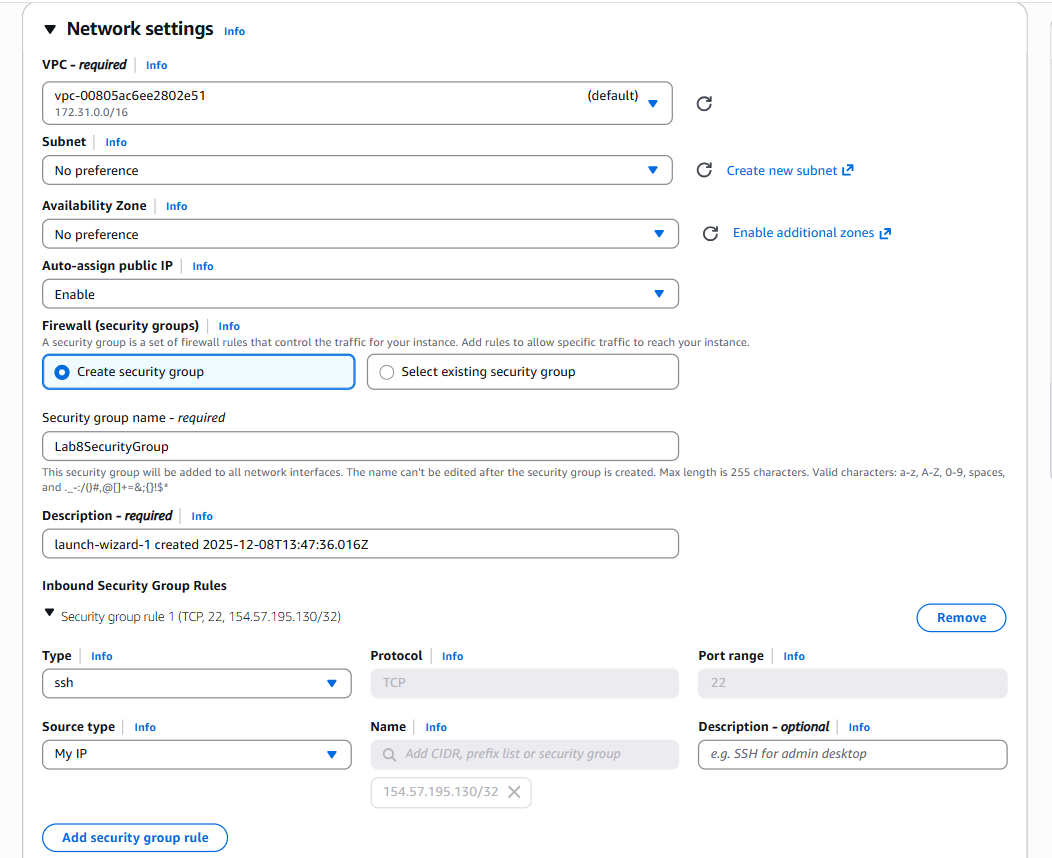
* **AMI: Amazon Linux 2**

****

* **Instance type: t2.micro**

****

* **Security group: Create Lab8SecurityGroup with SSH from My IP**

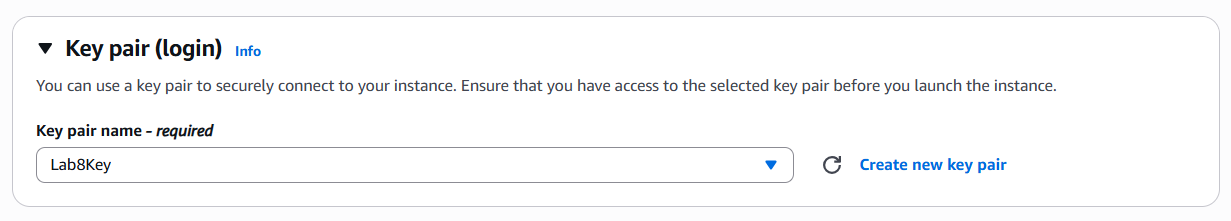
****

* **Storage: default**

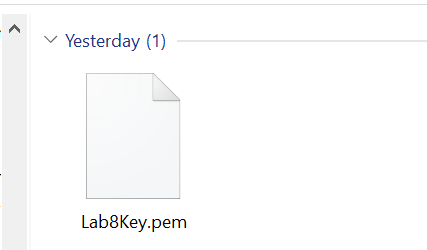
****

* **Key pair: Create Lab8Key (ED25519, .pem) and download the .pem file to your Windows host**
* **Capture the final review page and the key download prompt:**
* **Save screenshot as: task4\_launch\_instance\_config.png — final review page showing instance name, AMI, type, security group, key pair.**

****

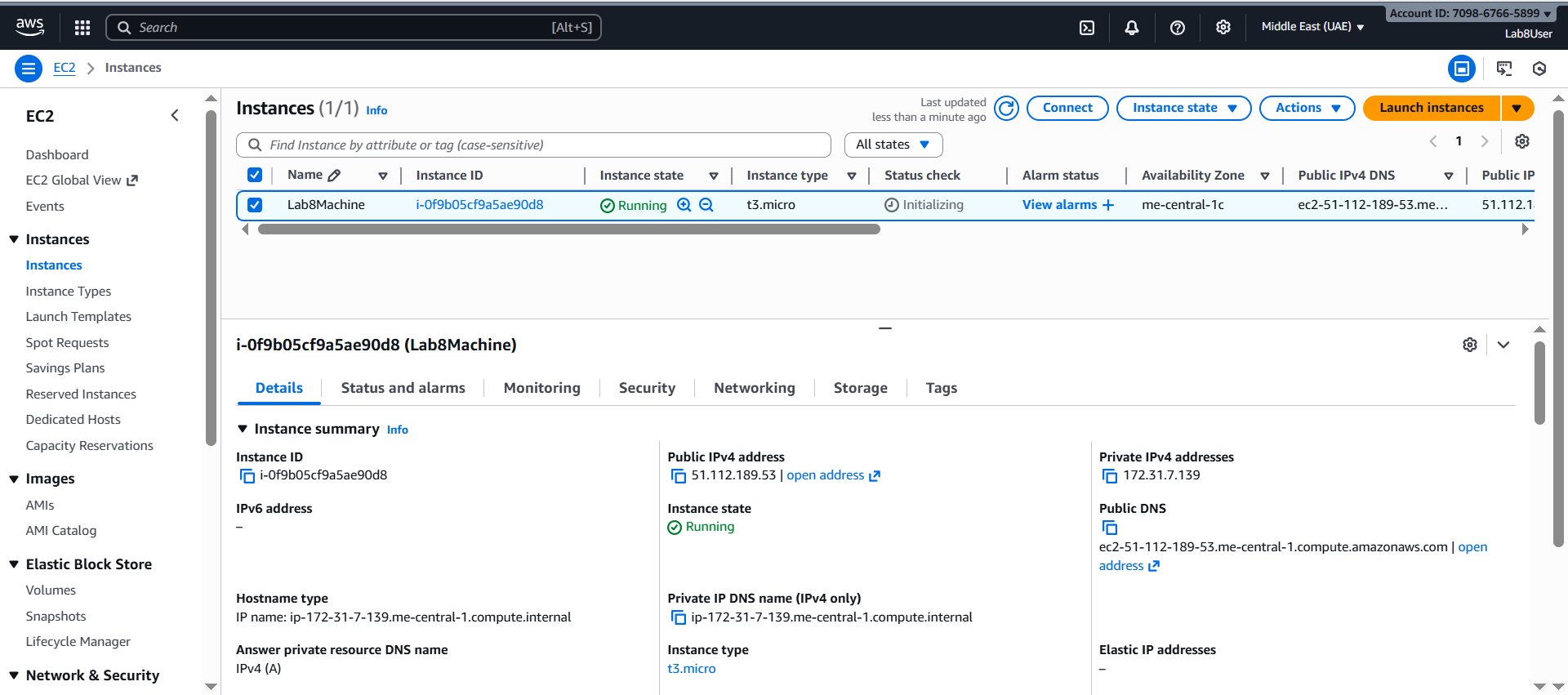
****

* **Save screenshot as: task4\_keypair\_download.png — Windows File Explorer showing Lab8Key.pem downloaded (do NOT open .pem contents).**

****

1. **After launch, EC2 Instances list showing Lab8Machine in "running" state and public IPv4 visible.**

**Save screenshot as: task4\_instance\_running\_console.png — Instances table with Lab8Machine running and Public IPv4.**

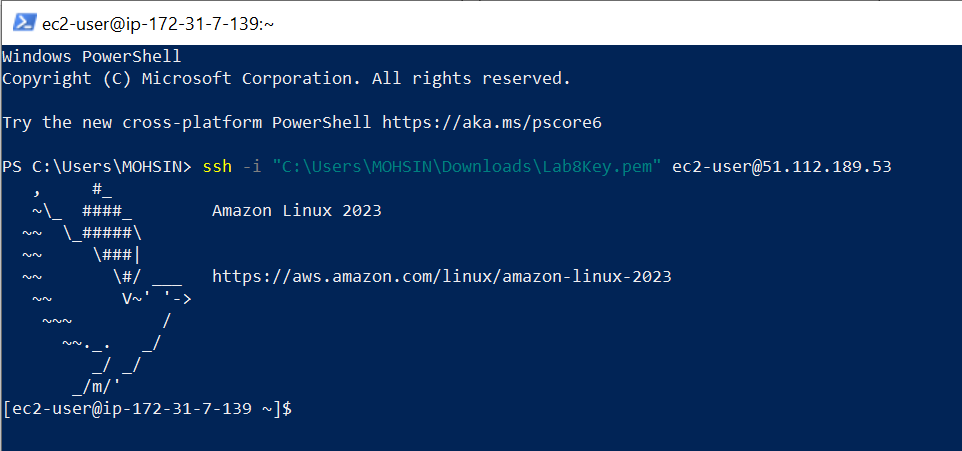
****

1. **On Windows host, run SSH using the downloaded .pem (PowerShell/Git Bash/Windows Terminal):**

ssh -i <path>/Lab8Key.pem ec2-user@<public-IP>

**Capture the SSH command and successful shell prompt on the EC2 instance:**

**Save screenshot as: task4\_ssh\_from\_windows\_to\_ec2.png — PowerShell showing ssh command and EC2 shell (do NOT show private key contents).**

****

**5. Run the install commands on the EC2 shell:**

sudo yum update -y

sudo yum install -y docker

sudo mkdir -p /usr/local/lib/docker/cli-plugins

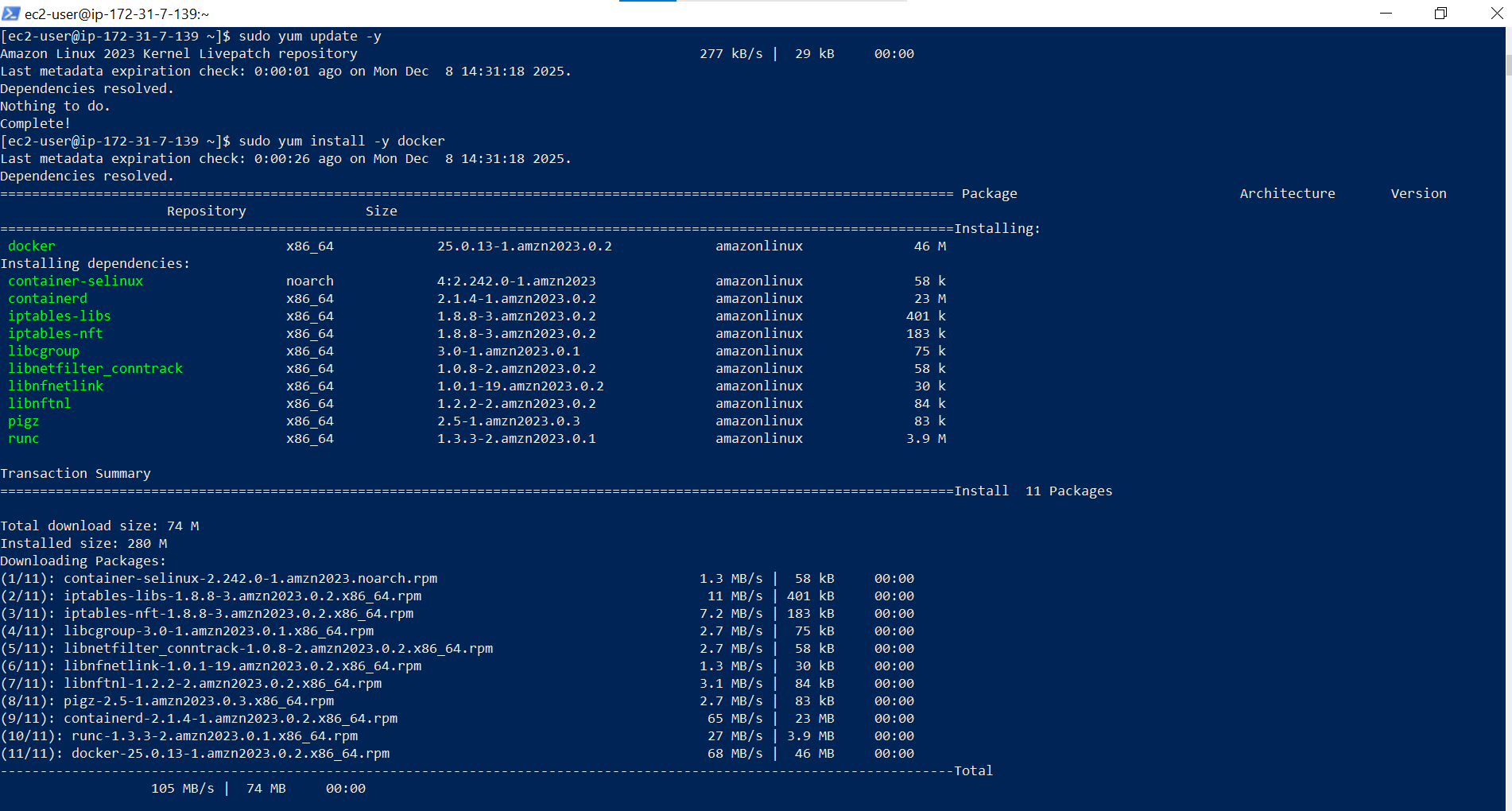
sudo curl -SL https://github.com/docker/compose/releases/latest/download/docker-compose-linux-x86\_64 -o /usr/local/lib/docker/cli-plugins/docker-compose

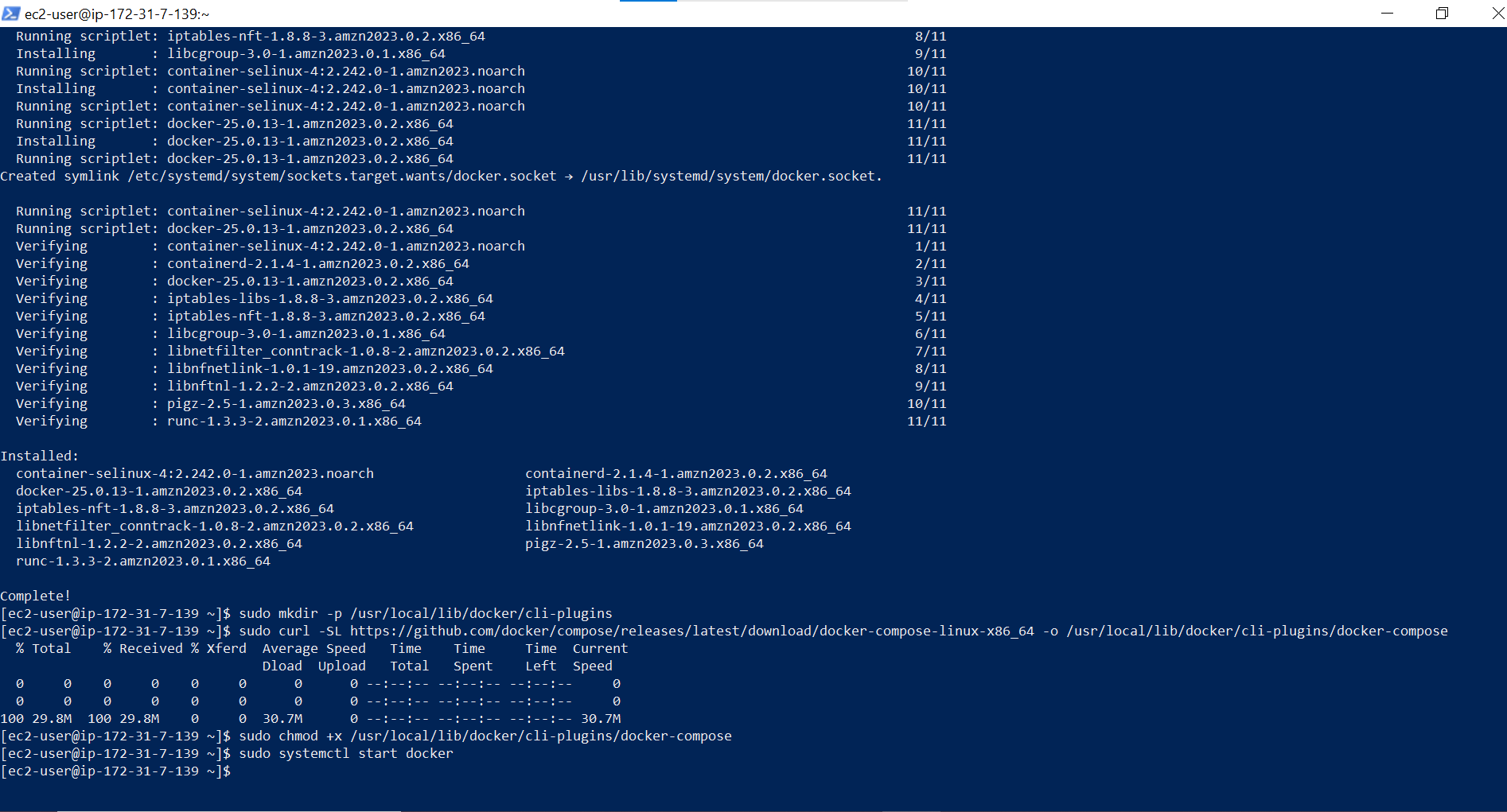
sudo chmod +x /usr/local/lib/docker/cli-plugins/docker-compose

sudo systemctl start docker

**Capture the terminal showing these commands run and successful outputs:**

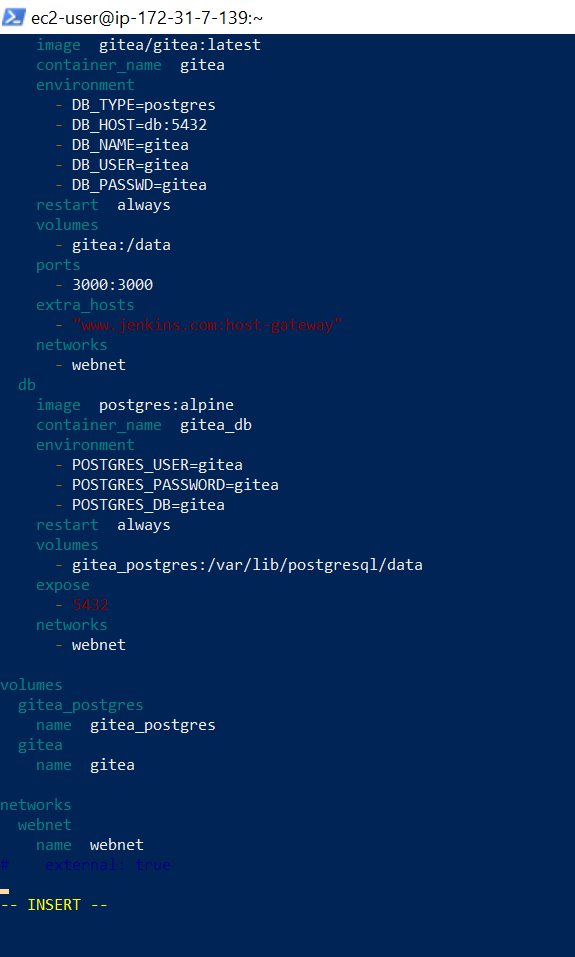
**Save screenshot as: task4\_ec2\_install\_docker\_compose\_started.png — outputs of update/install and systemctl start.**

****

****

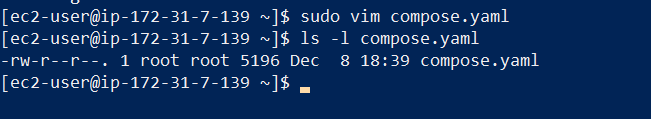
**6. Create/edit compose.yaml on the EC2 instance (sudo vim compose.yaml) and paste content from the repo: [Gitea](https://github.com/WaqasSaleem97/Gitea) . While pasting, capture the editor content:**

**Save screenshot as: task4\_vim\_compose\_yaml\_paste.png — vim editor showing compose.yaml contents while pasted.**

****

**7. Save and verify file exists:**

**Save screenshot as: task4\_compose\_yaml\_saved\_ls.png — ls -l showing compose.yaml present.**

****

**8. Add ec2-user to docker group, show groups before re-login, exit and reconnect, show groups after reconnect:**

groups # user does not docker permission

sudo usermod -aG docker $USER

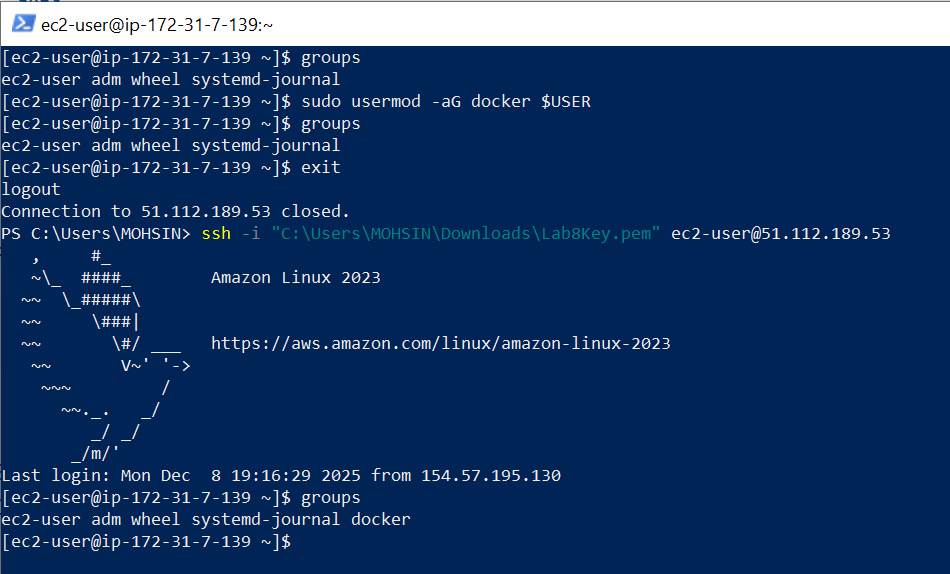
groups # before re-login

exit

# Reconnect

ssh -i <path>/Lab8Key.pem ec2-user@<public-IP>

groups # after re-login (should include docker)

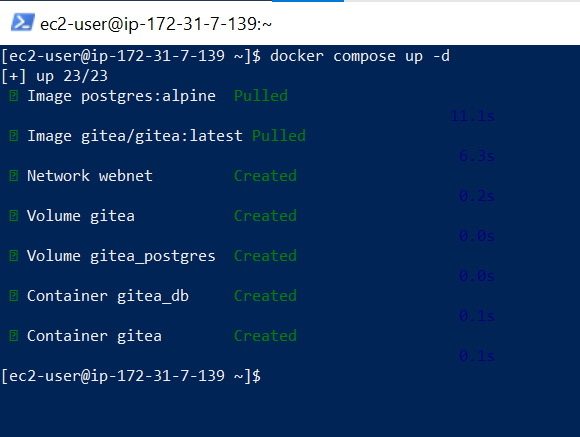


**Save screenshot as: task4\_usermod\_and\_groups\_before\_after.png — show usermod command, groups output before exit, reconnect sequence, and groups output after (docker included).**

**9. Run docker compose up -d from the directory with compose.yaml:**

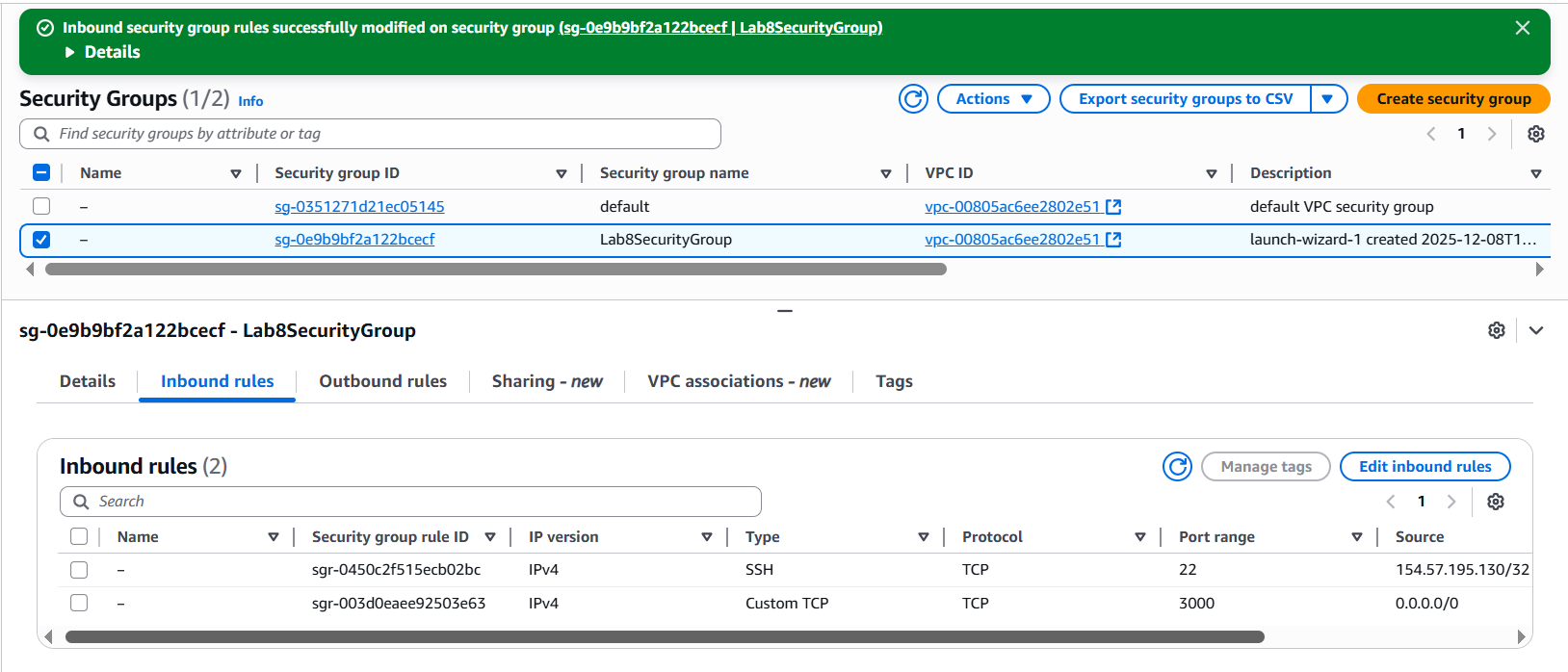
docker compose up -d

**Save screenshot as: task4\_docker\_compose\_up.png — output of docker compose up -d showing containers starting.**

****

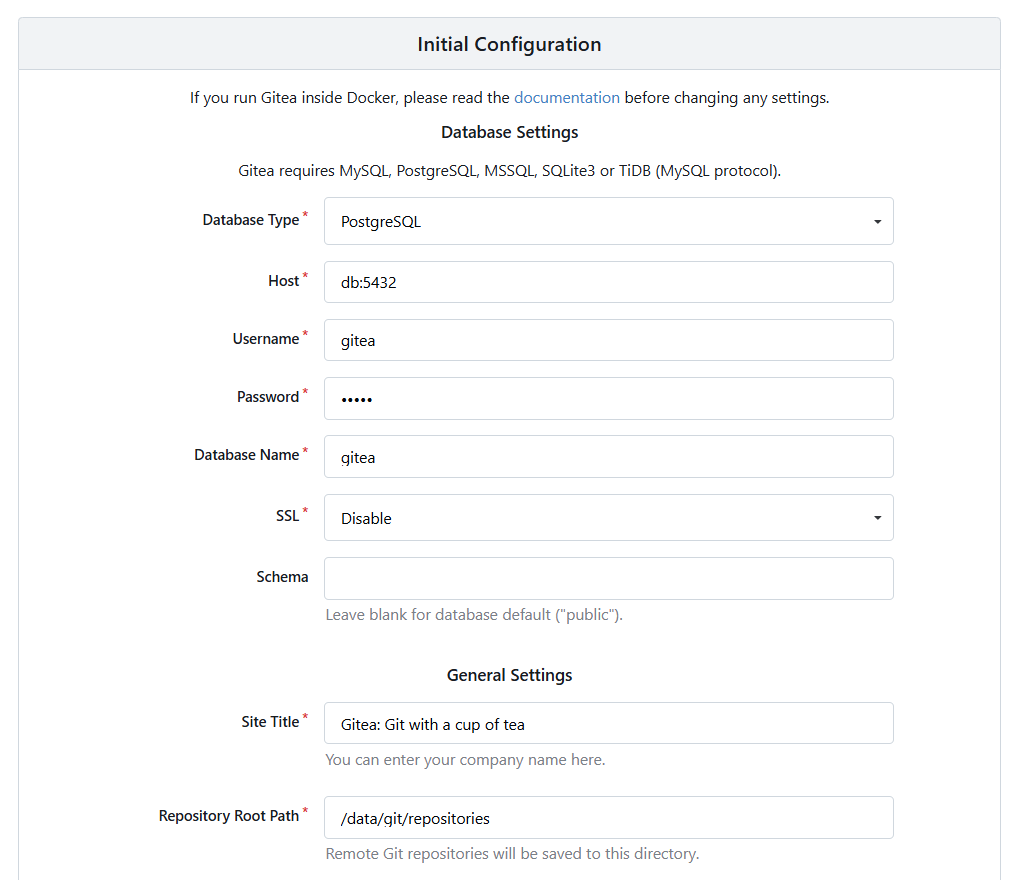
**10. Edit the security group Lab8SecurityGroup inbound rules in the EC2 console: add Custom TCP rule port 3000 source 0.0.0.0/0 and save. Capture the inbound rules after saving:**

**Save screenshot as: task4\_security\_group\_allow\_3000.png — security group inbound rules list showing SSH from My IP and Custom TCP 3000 anywhere.**

****

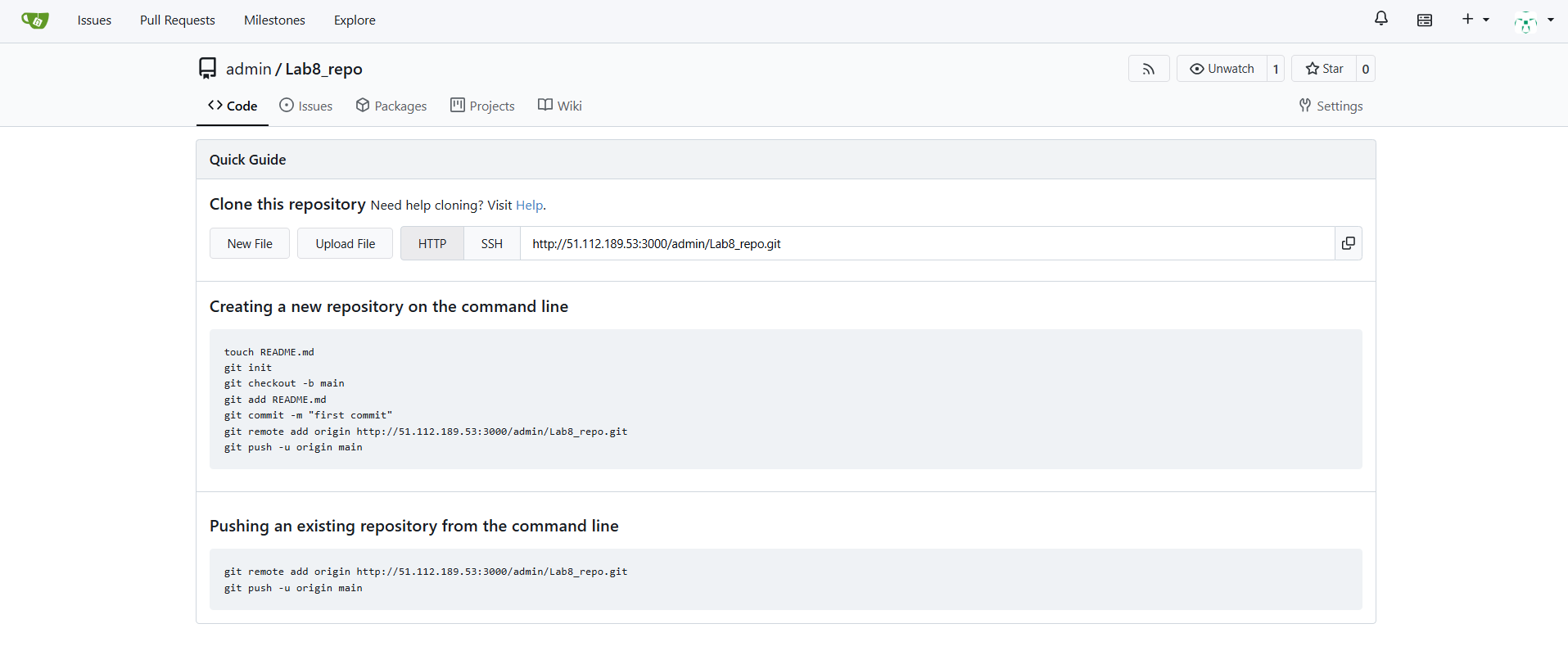
**11. From your Windows browser navigate to:**[**http://Public-IP:3000**](http://public-ip:3000/)**— capture the Gitea setup/install page:**

**Save screenshot as: task4\_gitea\_install\_page.png — Gitea installation page in browser.**

****

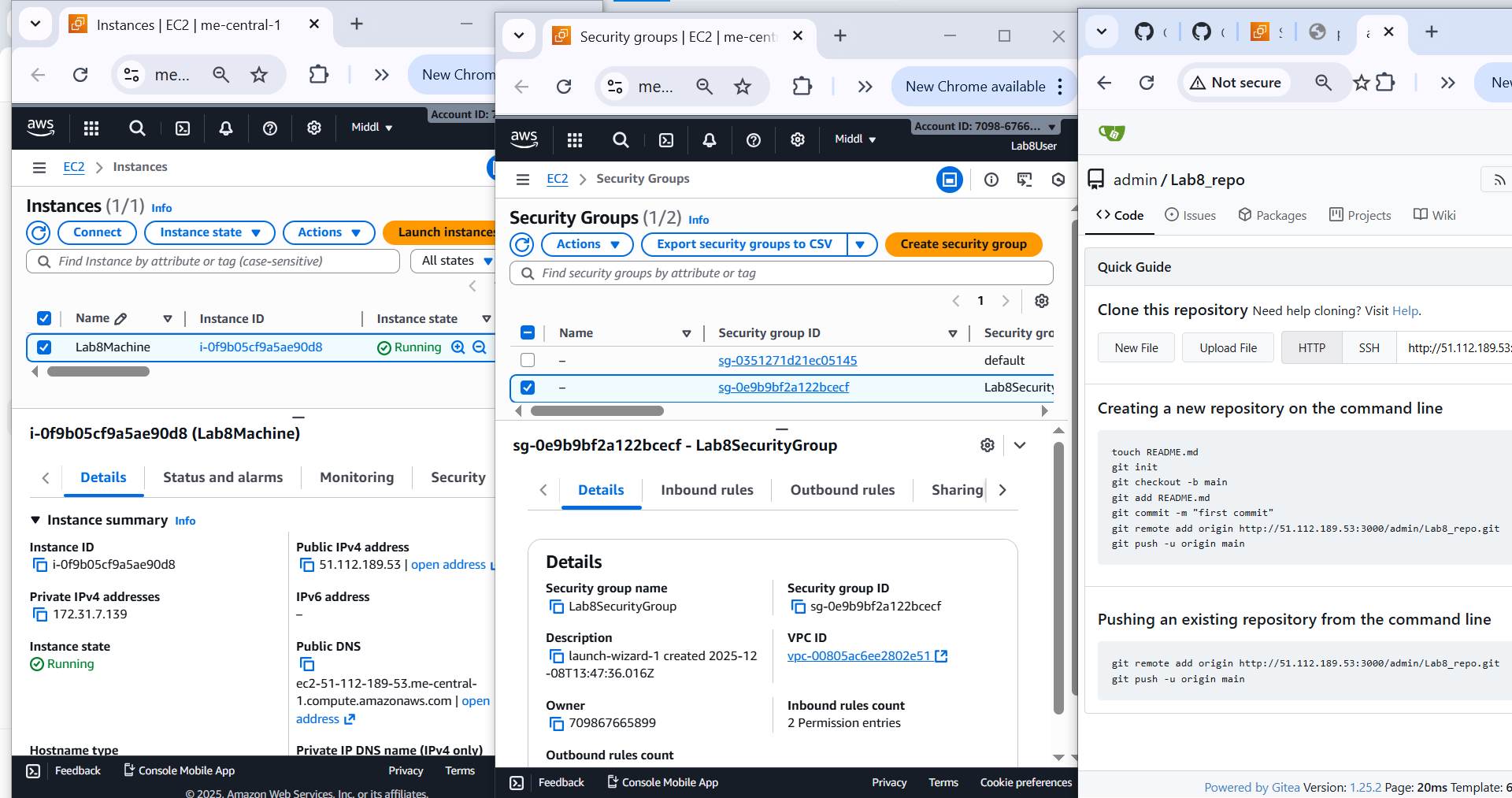
**12. Complete initial Gitea setup (create admin user, create a repo) and capture Gitea showing the created repository:**

**Save screenshot as: task4\_gitea\_create\_repo.png — Gitea UI showing the created repository.**

****

**13. Task 4 summary (combine evidence)**

**Save screenshot as: task4\_summary.png — single screenshot (or tiled screenshot) showing: EC2 Instances list with Lab8Machine running and public IP, security group inbound rules showing SSH and port 3000, and browser tab open to Gitea UI or repo list.**

****

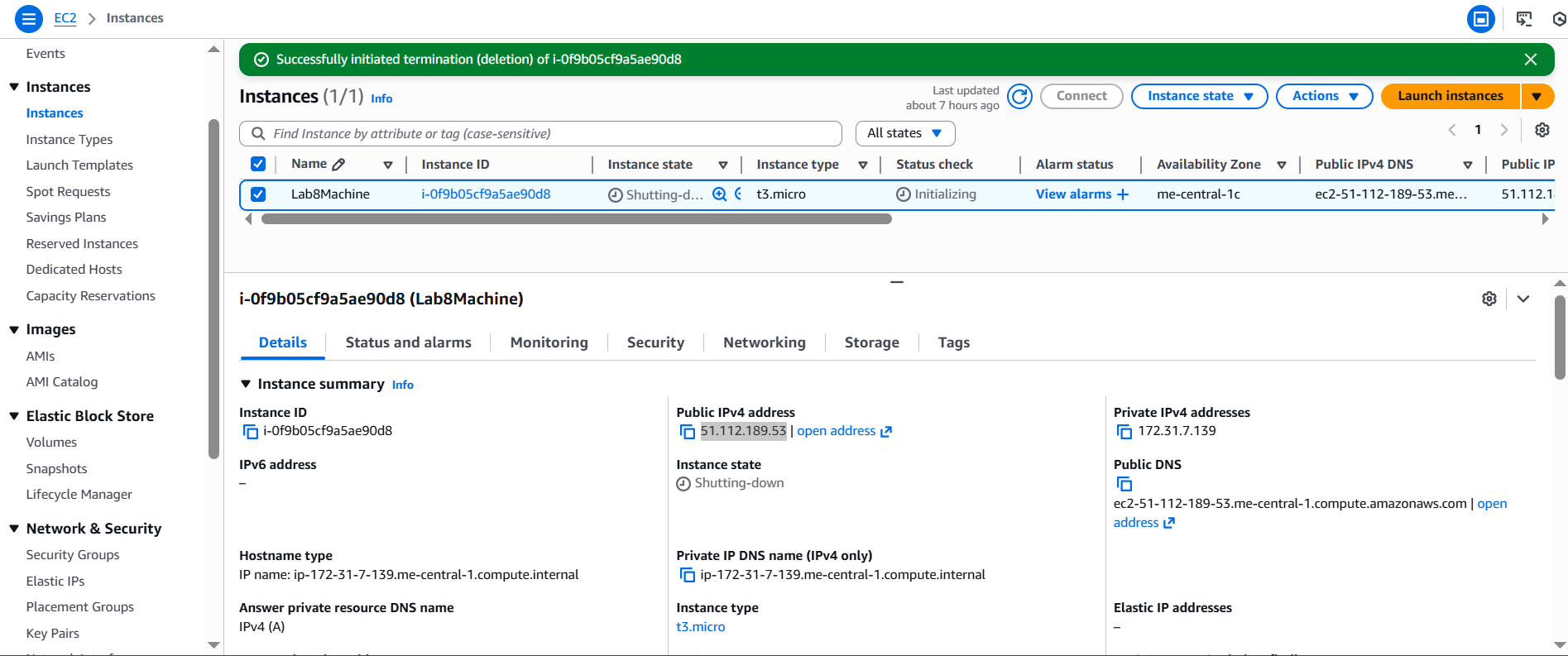
**Cleanup — Remove resources to avoid charges**

**After verification, terminate and delete everything you created. Capture screenshots immediately after each cleanup step.**

**Cleanup steps and required screenshots:**

1. **Terminate the EC2 instance Lab8Machine.**

**Save screenshot as: cleanup\_terminate\_instance.png — EC2 terminate instance confirmation.**

****

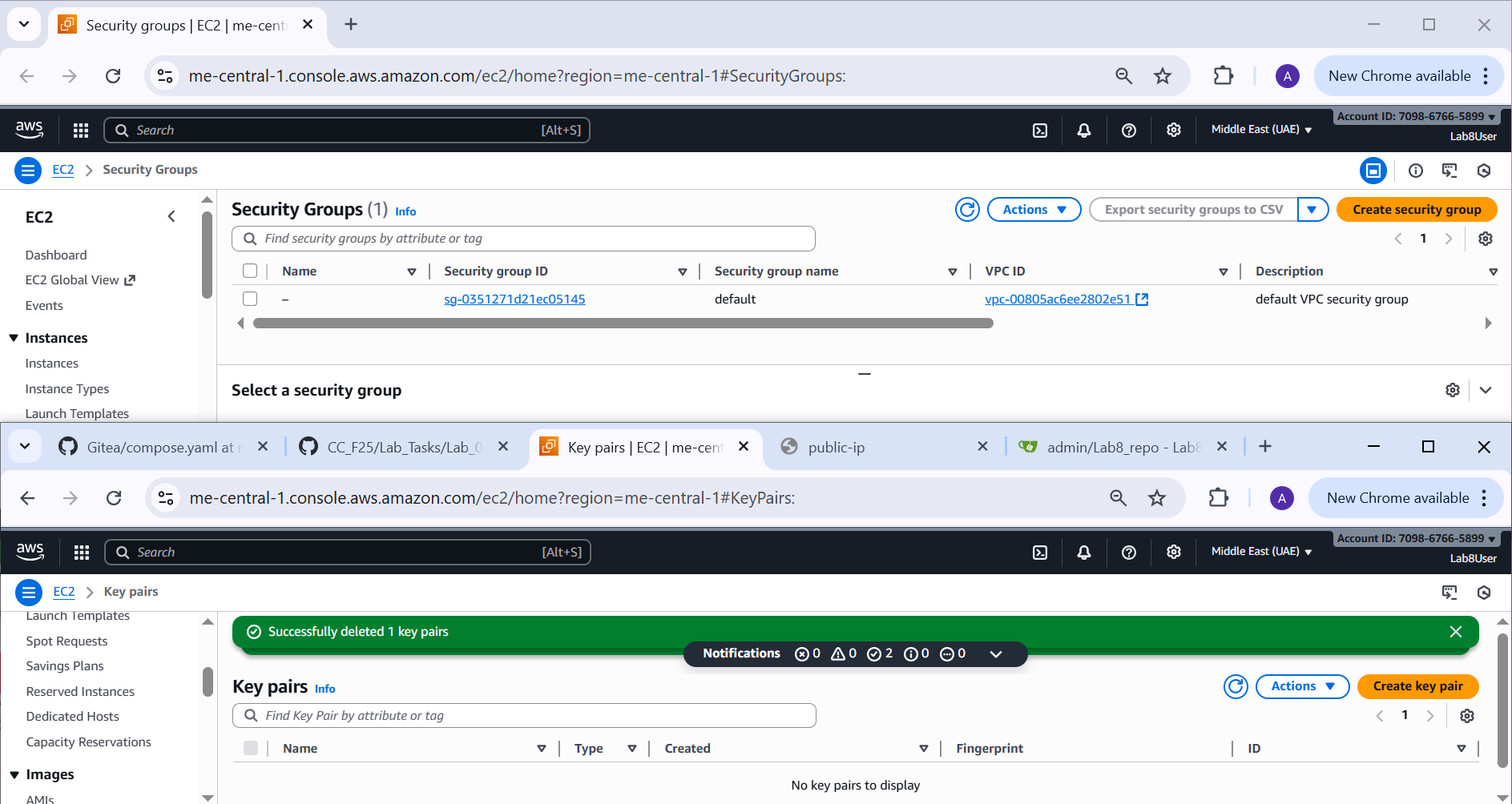
1. **Delete associated EBS volumes and snapshots (if any).**

**Save screenshot as: cleanup\_delete\_volumes\_snapshots.png — confirmation or list showing volumes/snapshots deleted.**

****

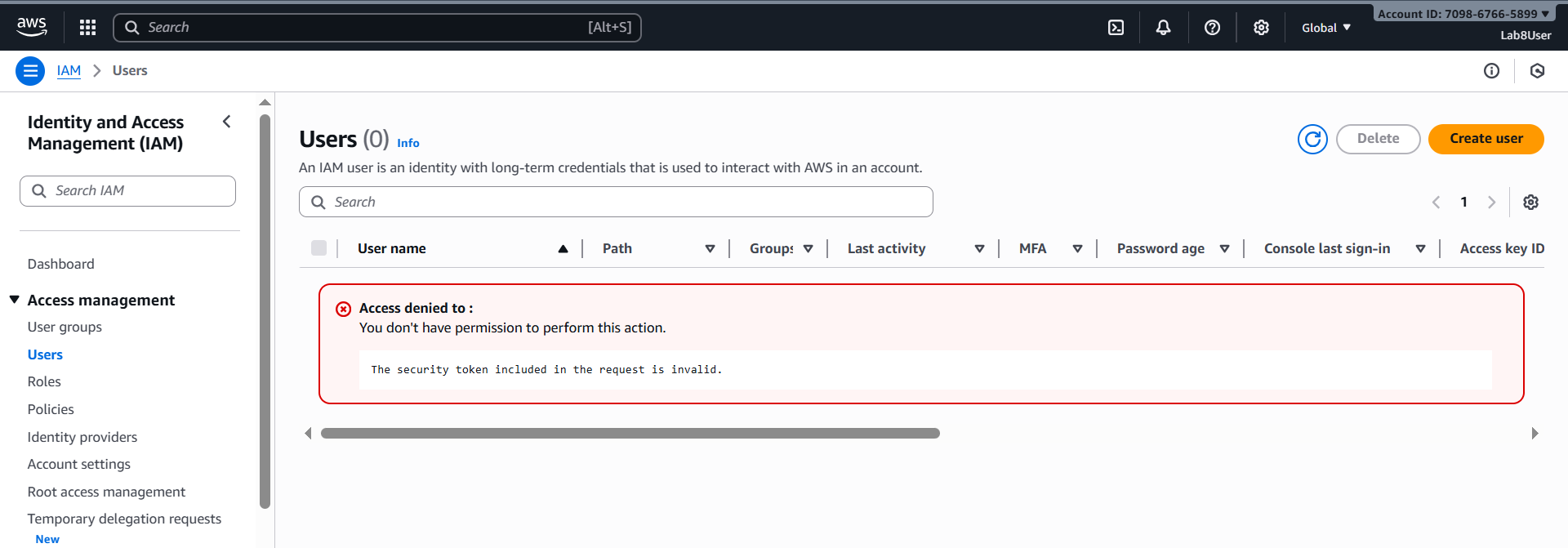
1. **Delete security group Lab8SecurityGroup and key pair Lab8Key from the EC2 console (after instances terminated).**

**Save screenshot as: cleanup\_delete\_security\_group\_and\_keypair.png — deletion confirmation(s) (show key pair list and security group list after deletion).**

****

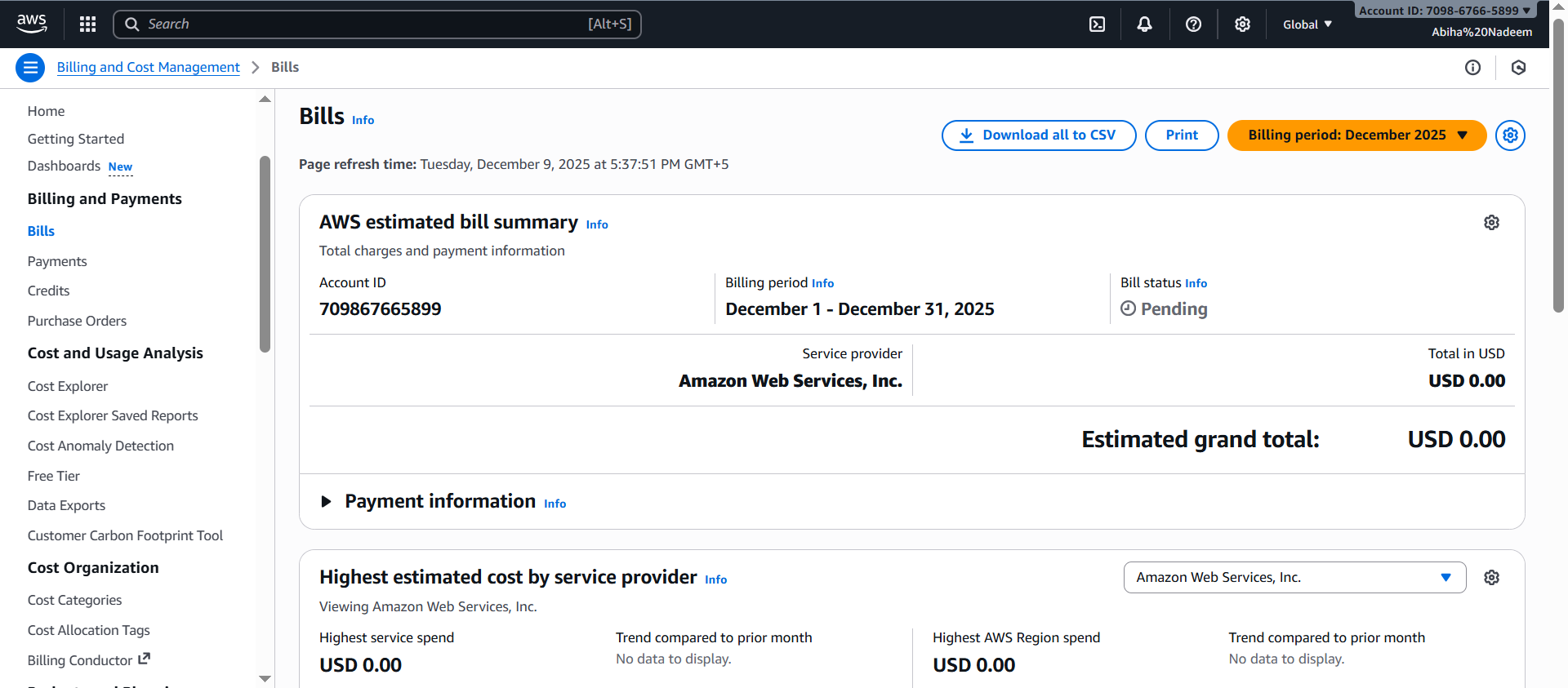
1. **Delete IAM users Lab8User and any access keys.**

**Save screenshot as: cleanup\_iam\_users\_deleted.png — IAM Users list showing Lab8User no longer present (or a deletion confirmation).**

****

1. **Final cleanup summary (show billing or resource groups with no active resources if possible).**

**Save screenshot as: cleanup\_summary.png — AWS console Billing/Resource Groups showing no active resources or no recent charges (if available).**

****

**\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\***