

TASK 5.2C

CLOUD COMPUTING

Creating a VPC Networking Environment for the Café

Challenge 1: Set up Secure Admin Access & Internet Access for Private Instances

Task 1: Creating a Public Subnet

- **Purpose:** Create a subnet that is connected to the internet.
- **Steps:**
 - Create a subnet (CIDR: 10.0.0.0/24) in **Availability Zone A**.
 - Create and attach an **Internet Gateway** to the Lab VPC.
 - Update the **Route Table** associated with this subnet to direct internet traffic (0.0.0.0/0) through the internet gateway.

Subnet settings
Specify the CIDR blocks and Availability Zone for the subnet.

Subnet 1 of 1

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

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.

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.

IPv4 subnet CIDR block
 256 IPs

Tags - optional
Key Value - optional [Remove](#)

VPC dashboard

You have successfully created 1 subnet: subnet-063ddf9bbc816df8e

Subnets (1) [Info](#)

Subnet ID	Name	Subnet ID	State	VPC	Block Public...	IPv4 CIDR
subnet-063ddf9bbc816df8e	Public Subnet	subnet-063ddf9bbc816df8e	Available	vpc-03f83f212156a6190 Lab...	Off	10.0.0.0/24

Select a subnet

Create internet gateway Info

An internet gateway is a virtual router that connects a VPC to the internet. To create a new internet gateway specify the name for the gateway below.

Internet gateway settings

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

Tags - optional
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="lab-igw"/> X

[Add new tag](#)
You can add 49 more tags.

[Cancel](#) [Create internet gateway](#)

VPC
Attach an internet gateway to a VPC to enable the VPC to communicate with the internet. Specify the VPC to attach below.

Available VPCs
Attach the internet gateway to this VPC.
 X

[▶ AWS Command Line Interface command](#)

[Cancel](#) [Attach internet gateway](#)



The screenshot shows the AWS VPC Route Tables interface. The URL in the browser is us-east-1.console.aws.amazon.com/vpcconsole/home?region=us-east-1#EditRoutes:RouteTableId=rtb-058d07890c54f9b1c. The page title is "Edit routes". The main content area shows a table with columns: Destination, Target, Status, and Propagated. There are two rows: one for destination 10.0.0.0/16 targeting 'local' (status Active) and another for 0.0.0.0/0 targeting 'Internet Gateway' (status Inactive). A third row is shown as a placeholder. Buttons at the bottom include "Add route", "Cancel", "Preview", and "Save changes".

Task 2: Creating a Bastion Host

- **Purpose:** Set up a secure admin jump box for remote access to private instances.
- **Steps:**
 - Launch a **Linux EC2 instance** (t2.micro) in the public subnet.
 - Enable **public IP assignment**.
 - Create and assign a **Security Group (Bastion Host SG)** that allows **SSH (port 22)** access only from **your IP address**.
 - Use **Amazon Linux 2023 AMI** and key pair vockey.

Challenge (Cafe) lab: Creating a new EC2 instance

Launch an instance

Amazon EC2 allows you to create virtual machines, or instances, that run on the AWS Cloud. Quickly get started by following the simple steps below.

Name and tags

Name: e.g. My Web Server

Application and OS Images (Amazon Machine Image)

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: Search our full catalog including 1000s of application and OS images

Quick Start

Amazon Linux, macOS, Ubuntu, Windows, Red Hat, SUSE Linux, Debian

Browse more AMIs

Including AMIs from AWS, Marketplace and the Community

Amazon Machine Image (AMI)

CloudShell Feedback

14°C Cloudy

Search

© 2025, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences ENG IN 10-04-2025

Launch an instance

Enable Additional charges apply when outside of free tier allowance

Firewall (security group)

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

Bastion Host SG

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

launch-wizard-1 created 2025-04-09T14:37:05.841Z

Inbound Security Group Rules

Security group rule 1 (TCP 22, 49.184.231.224/32)

Type: ssh Protocol: TCP Port range: 22

Source type: My IP Name: Add CIDR, prefix list or security group Description - optional: e.g. SSH for admin desktop

49.184.231.224/32

Add security group rule

Summary

Number of instances: 1

Software Image (AMI): Amazon Linux 2023 AMI 2023.7.2...read more

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 of opening an AWS account, you get 750 hours per month of t2.micro instance usage (or t3.micro where t2.micro isn't available) when used with free tier

Cancel Launch instance Preview code

CloudShell Feedback

14°C Cloudy

Search

© 2025, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences ENG IN 10-04-2025

The screenshot shows the AWS Cloud9 IDE interface. The top navigation bar includes tabs for 'Challenge (Cafe) lab: Creating', 'Launch an instance | EC2 | us-east-1', 'OnTrack', 'Submission History - SEBE Studio', and 'Feedback Studio'. The main content area displays the 'Launch an instance' wizard. The 'Key pair (login)' section shows a dropdown for 'Key pair name - required' with 'vockey' selected. The 'Network settings' section shows a VPC dropdown with 'vpc-03f83f212156a6190 (Lab VPC)' and a Subnet dropdown with 'subnet-063df9bbc816df0e'. The 'Summary' section shows 'Number of instances' set to 1, using 'Amazon Linux 2023 AMI 2023.7.2...' and 't2.micro' as the virtual server type. A tooltip for the 'Free tier' is visible, stating: 'Free tier: In your first year of opening an AWS account, you get 750 hours per month of t2.micro instance usage (or t3.micro where t3.micro isn't available) when used with Free Tier'. The bottom of the screen shows the AWS CloudShell and a taskbar with various icons.

Task 3: Testing the Connection to the Bastion Host

- **Purpose:** Ensure you can remotely connect to the bastion host using SSH.
- **Steps:**
 - Download the appropriate SSH key (.pem or .ppk).
 - Use an SSH client (e.g., PuTTY or terminal) to connect to the Bastion Host.

Task 4: Creating a Private Subnet

- **Purpose:** Isolate internal resources from direct internet access.
- **Steps:**
 - Create a **new subnet (CIDR: 10.0.1.0/24)** in the same AZ as the public subnet.
 - Do **not** attach an internet gateway or enable public IP assignment.

The screenshot shows two stacked screenshots of the AWS Cloud Console interface. The top screenshot displays the 'Elastic IP addresses' page under the 'Network & Security' section. The sidebar includes options like AMI Catalog, Elastic Block Store, Network & Security (selected), Load Balancing, Auto Scaling, and Settings. The main area shows a search bar and a table with columns for Name, Allocated IPv4 addr..., Type, Allocation ID, and Reverse DNS record. A message at the bottom states 'No Elastic IP addresses found in this Region'. The bottom screenshot shows the 'Allocate Elastic IP address' step. It has two radio button options: 'Customer-owned pool of IPv4 addresses created from your on-premises network for use with an Outpost.' (disabled) and 'Allocate using an IPv4 IPAM pool' (disabled). Below this is a 'Network border group' dropdown set to 'us-east-1'. A 'Create accelerator' button is available. A 'Tags - optional' section allows adding up to 50 tags, with a note about AWS Global Accelerator. At the bottom right are 'Cancel' and 'Allocate' buttons.

Screenshot of the AWS CloudShell interface showing the association of an Elastic IP address to an instance.

Associate Elastic IP address

Elastic IP address: 34.237.171.44

Resource type
Choose the type of resource with which to associate the Elastic IP address.

Instance
 Network interface

Instance
I-049655c9db883e5e4

Private IP address
The private IP address with which to associate the Elastic IP address.
Choose a private IP address

Reassociation
Specify whether the Elastic IP address can be reassigned to a different resource if it is already associated with a resource.
 Allow this Elastic IP address to be reassigned

Associate

Elastic IP address associated successfully.
Elastic IP address 34.237.171.44 has been associated with instance I-049655c9db883e5e4

Elastic IP addresses (1)

Name	Allocated IPv4 address	Type	Allocation ID	Reverse DNS record
34.237.171.44	34.237.171.44	Public IP	eipalloc-096dad93ce1dc3246	-

Actions | Allocate Elastic IP address

EC2

- Dashboard
- EC2 Global View
- Events
- Instances
 - Instances
 - Instance Types
 - Launch Templates
 - Spot Requests
 - Savings Plans
 - Reserved Instances
 - Dedicated Hosts
 - Capacity Reservations
- Images
- Elastic Block Store
 - Volumes
 - Snapshots
 - Lifecycle Manager

CloudShell Feedback © 2025, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences ENG IN 10-04-2025

The screenshot shows the 'Create subnet' page in the AWS VPC console. The 'VPC ID' dropdown is set to 'vpc-05fb3f212156a6190 (Lab VPC)'. Under 'Associated VPC CIDRs', the 'IPv4 CIDRs' field is set to '10.0.0.0/16'. In the 'Subnet settings' section, 'Subnet 1 of 1' is being configured with a 'Subnet name' of 'my-subnet-01'. The 'Availability Zone' dropdown is set to 'Choose the zone in which your subnet will reside, or let Amazon choose one for you.' The top navigation bar includes tabs like 'Challenge (Cafe) labz: Create', 'View status checks for Am...', 'VPC | us-east-1', 'OnTrack', 'Submission History - SEBE', 'Feedback Studio', and account information 'United States (N. Virginia)'. The bottom of the screen shows a Windows taskbar with various icons and system status.

The screenshot shows two consecutive screenshots of the AWS VPC console.

Screenshot 1: Create Subnet

- The URL is [https://us-east-1.console.aws.amazon.com/vpcconsole/home?region=us-east-1/CreateSubnet](https://us-east-1.console.aws.amazon.com/vpcconsole/home?region=us-east-1>CreateSubnet).
- The "Availability Zone" dropdown is set to "United States (N. Virginia) / us-east-1a".
- The "IPv4 VPC CIDR block" dropdown is set to "10.0.0.0/16".
- The "IPv4 subnet CIDR block" dropdown is set to "10.0.1.0/24".
- A "Tags - optional" section contains a tag named "Name" with the value "Private Subnet".
- At the bottom right are "Cancel" and "Create subnet" buttons.

Screenshot 2: Subnets Dashboard

- The URL is <https://us-east-1.console.aws.amazon.com/vpcconsole/home?region=us-east-1#subnets:subnetId=subnet-098a3fbac037e091f>.
- A green success message at the top says "You have successfully created 1 subnet: subnet-098a3fbac037e091f".
- The "Subnets (1) Info" table shows one entry:

Name	Subnet ID	State	VPC	Block Public...	IPv4 CIDR
Private Subnet	subnet-098a3fbac037e091f	Available	vpc-03f83f212156a6190 Lab...	Off	10.0.1.0/24

- At the bottom right are "Actions" and "Create subnet" buttons.

Task 5: Creating a NAT Gateway

- Purpose:** Allow **outbound internet access** from private subnet instances (e.g., for updates).
- Steps:**
 - Create a **NAT Gateway** in the **Public Subnet**, and allocate an **Elastic IP**.
 - Create a **new route table**, route all traffic (0.0.0.0/0) to the NAT Gateway.
 - Associate this route table with the **Private Subnet**.

Create NAT gateway Info

A highly available, managed Network Address Translation (NAT) service that instances in private subnets can use to connect to services in other VPCs, on-premises networks, or the internet.

NAT gateway settings

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

Subnet
Select a subnet in which to create the NAT gateway.

Connectivity type
Select a connectivity type for the NAT gateway.
 Public
 Private

Elastic IP allocation ID Info
Assign an Elastic IP address to the NAT gateway.

Additional settings Info

Tags

VPC dashboard

nat-093ff1c66fc00aa17 / Lab NAT Gateway

Details

NAT gateway ID <input type="text" value="nat-093ff1c66fc00aa17"/>	Connectivity type Public	State <input type="text" value="Pending"/>	State message <small>Info</small> -
NAT gateway ARN <input type="text" value="arn:aws:ec2:us-east-1:471112865201:natgateway/nat-093ff1c66fc00aa17"/>	Primary public IPv4 address -	Primary private IPv4 address -	Primary network interface ID -
VPC <input type="text" value="vpc-03f83f212156a6190 / Lab VPC"/>	Subnet <input type="text" value="subnet-063ddf9bbc816df8e / Public Subnet"/>	Created <input type="text" value="Thursday, April 10, 2025 at 00:53:21 GMT+10"/>	Deleted -

Secondary IPv4 addresses

Secondary IPv4 addresses

Secondary IPv4 addresses are not available for this nat gateway.

The screenshot shows the 'Create route table' wizard in the AWS VPC console. The first step, 'Route table settings', is displayed. It includes fields for 'Name - optional' (set to 'Private Route Table') and 'VPC' (set to 'vpc-03f83f212156a6190 (Lab VPC)'). Below this, the 'Tags' section allows adding key-value pairs; one tag ('Name' key, 'Private Route Table' value) is already present. A note indicates that up to 49 more tags can be added. At the bottom right are 'Cancel' and 'Create route table' buttons.

Screenshot of the AWS VPC console showing the creation and configuration of a Private Route Table.

Route table rtb-0e0cbbf788eaaf7a7 | Private Route Table was created successfully.

rtb-0e0cbbf788eaaf7a7 / Private Route Table

Details Info

Route table ID rtb-0e0cbbf788eaaf7a7	Main No	Explicit subnet associations -	Edge associations -
VPC vpc-05f83f212156a6190 Lab VPC	Owner ID 471112865201		

Routes Subnet associations Edge associations Route propagation Tags

Routes (1)

Destination	Target	Status	Propagated
10.0.0.0/16	local	Active	No

Edit routes

Destination	Target	Status	Propagated
10.0.0.0/16	local	Active	No
0.0.0.0/0	NAT Gateway	-	No
nat-093ff1c66fc00aa17			

Add route **Remove** **Cancel** **Preview** **Save changes**

Screenshot of the AWS VPC console showing the 'Edit routes' page for a route table. The table has two entries:

Destination	Target	Status	Propagated
10.0.0.0/16	local	Active	No
0.0.0.0/0	NAT Gateway	-	No

Buttons at the bottom include 'Add route', 'Cancel', 'Preview', and 'Save changes'.

Screenshot of the AWS VPC console showing the 'rtb-0e0cbbf788eaaf7a7 / Private Route Table' details page. A success message says 'Updated routes for rtb-0e0cbbf788eaaf7a7 / Private Route Table successfully'.

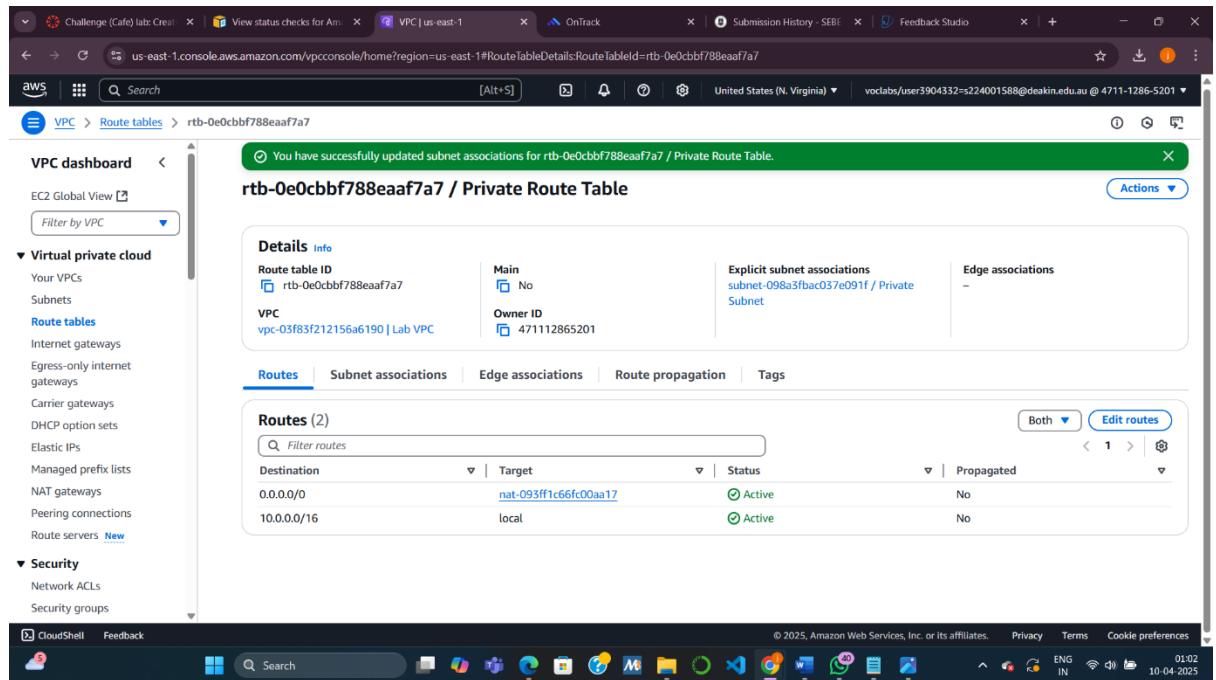
Details

- Route table ID: rtb-0e0cbbf788eaaf7a7
- Main: No
- VPC: vpc-03f83f212156a6190 | Lab VPC
- Owner ID: 471112865201
- Explicit subnet associations: -
- Edge associations: -

Routes (2)

Destination	Target	Status	Propagated
0.0.0.0/0	nat-093ff1c66fc00aa17	Active	No
10.0.0.0/16	local	Active	No

Buttons at the top right include 'Actions' and 'Edit routes'.



Task 6: Creating an EC2 Instance in the Private Subnet

- **Purpose:** Deploy a backend EC2 instance with no direct internet access.
- **Steps:**
 - Create a **new key pair** (vokey2).
 - Launch an **EC2 instance** (t2.micro, Amazon Linux 2023) in the **Private Subnet** using vokey2.
 - Create a **Security Group (Private Instance SG)** that allows **SSH (port 22)** only from the **Bastion Host SG**.

JASVEENA - 224001588

The screenshot shows the AWS EC2 'Launch an instance' wizard at the 'Create key pair' step. The 'Key pair name' field contains 'vockey2'. The 'Key pair type' section shows two options: 'RSA' (selected) and 'ED25519'. The 'Private key file format' section shows two options: '.pem' (selected) and '.ppk'. A note below states: 'When prompted, store the private key in a secure and accessible location on your computer. You will need it later to connect to your instance.' A 'Create key pair' button is at the bottom right.

Create key pair

Key pair name
Key pairs allow you to connect to your instance securely.
vockey2
The name can include up to 255 ASCII characters. It can't include leading or trailing spaces.

Key pair type

RSA RSA encrypted private and public key pair
 ED25519 ED25519 encrypted private and public key pair

Private key file format

.pem For use with OpenSSH
 .ppk For use with PuTTY

When prompted, store the private key in a secure and accessible location on your computer. You will need it later to connect to your instance. [Learn more](#)

Create key pair

Summary

Number of instances [Info](#)

Software Image (AMI)
Amazon Linux 2023 AMI 2023.7.2...[read more](#)

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 of opening an AWS account, you get 750 hours per month of t2.micro instance usage (or t3.micro where t2.micro isn't available) when used with certain AWS services.

Launch instance

Preview code

Description - required [Info](#)
launch-wizard-1 created 2025-04-09T15:06:11.785Z

Inbound Security Group Rules

Security group rule 1 (TCP, 22, sg-048ea7c65bf40b82f)

Type [Info](#) **Protocol** [Info](#) **Port range** [Info](#)
ssh TCP 22

Source type [Info](#) **Source** [Info](#) **Description - optional** [Info](#)
Custom Add CIDR, prefix list or security group e.g. SSH for admin desktop
sg-048ea7c65bf40b82f

Add security group rule

Advanced network configuration

Network interface 1

Device index [Info](#) **Network interface** [Info](#) **Description** [Info](#)
0 New interface

Subnet [Info](#) **Security groups** [Info](#) **Auto-assign public IP** [Info](#)
subnet-098a3fbac037e091f New security group Disable

CloudShell **Feedback**

The screenshot shows the AWS EC2 Instances Launch an instance page. At the top, there is a green success message: "Successfully initiated launch of instance (i-0b13934c25586481e)". Below this, there is a "Launch log" button. Under "Next Steps", there are several options: "Create billing and free tier usage alerts", "Connect to your instance", "Connect an RDS database", "Create EBS snapshot policy", "Manage detailed monitoring", "Create Load Balancer", "Create AWS budget", and "Manage CloudWatch alarms". The browser status bar at the bottom indicates the URL is us-east-1.console.aws.amazon.com/ec2/home?region=us-east-1#LaunchInstances.

Task 7: Configuring SSH Passthrough (Agent Forwarding)

- **Purpose:** Access the private instance **through the Bastion Host** without copying key files.
- **Steps:**
 - **macOS/Linux:** Use ssh-add to load both vockey and vockey2 into the ssh-agent.
 - **Windows:** Use **Pageant** to load both .ppk files.
 - Enable **agent forwarding** so you can jump from Bastion Host to the Private Instance using the correct key.

This page contains download links for the latest released version of PuTTY. Currently this is 0.83, released on 2025-02-08.

When new releases come out, this page will update to contain

Release versions of PuTTY are versions we think are reasonable to put in our repository. If you have a problem with this release, then it might be worth trying out the [development snapshots](#), to see if the problem has already been fixed.

Package files

You probably want one of these. They include versions:

(Not sure whether you want the 32-bit or the 64-bit version?)

We also publish the latest PuTTY installers for all Windows platforms:

MSI (Windows Installer)

64-bit x86:	putty-64bit-0.83-installer.msi
64-bit Arm:	putty-arm64-0.83-installer.msi
32-bit x86:	putty-0.83-installer.msi

Unix source archive

.tar.gz:	putty-0.83.tar.gz
----------	-----------------------------------

[\(signature\)](#)

Alternative binary files

here is a [permanent link to the 0.83 release](#).

Windows pterm).

g; they usually take a few days to appear there after we release them.

This page contains download links for the latest released version of PuTTY. Currently this is 0.83, released on 2025-02-08.

When new releases come out, this page will update to contain

Release versions of PuTTY are versions we think are reasonable to put in our repository. If you have a problem with this release, then it might be worth trying out the [development snapshots](#), to see if the problem has already been fixed.

Package files

You probably want one of these. They include versions:

(Not sure whether you want the 32-bit or the 64-bit version?)

We also publish the latest PuTTY installers for all Windows platforms:

MSI (Windows Installer)

64-bit x86:	putty-64bit-0.83-installer.msi
64-bit Arm:	putty-arm64-0.83-installer.msi
32-bit x86:	putty-0.83-installer.msi

Unix source archive

.tar.gz:	putty-0.83.tar.gz
----------	-----------------------------------

[\(signature\)](#)

Alternative binary files

here is a [permanent link to the 0.83 release](#).

Windows pterm).

g; they usually take a few days to appear there after we release them.

You have successfully updated inbound rules for acl-060c97985175c6b87 / Lab Network ACL.

Name	Network ACL ID	Associated with	Default	VPC ID	Inbound Rules
acl-05669d05258d57cba	6 Subnets	Yes	vpc-02732b9740d4330d4	2 Int	
acl-00e385484b5c46011	subnet-063dd9bbc816df8e / Public Subnet	Yes	vpc-03f83f212156a6190 / Lab VPC	2 Int	
Lab Network ACL	acl-060c97985175c6b87	subnet-098a3fbac037e091f / Private Subnet	No	vpc-03f83f212156a6190 / Lab VPC	3 Int

acl-060c97985175c6b87 / Lab Network ACL

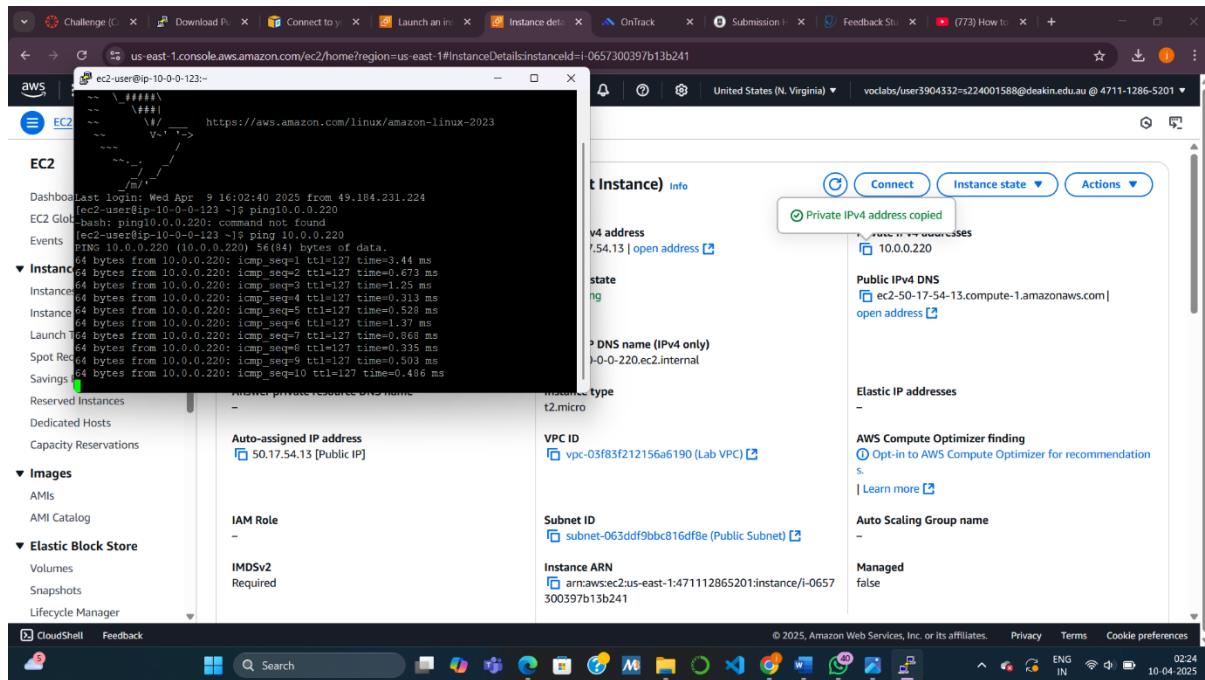
Details | Inbound rules | Outbound rules | Subnet associations | Tags

Details

Network ACL ID acl-060c97985175c6b87	Associated with subnet-098a3fbac037e091f / Private Subnet	Default No	VPC ID vpc-03f83f212156a6190 / Lab VPC
Owner 471112865201			

Task 8: Testing the SSH Connection from Bastion to Private Instance

- **Purpose:** Confirm successful passthrough SSH.
- **Steps:**
 - SSH to Bastion Host with agent forwarding enabled.
 - From there, SSH into the **Private EC2 instance** using its **private IP address**.
 - Test internet access from the private instance using ping 8.8.8.8.



Challenge 2: Implement Additional Security with Network ACLs

Task 9: Creating a Network ACL

- Purpose:** Add another security layer for the private subnet.
- Steps:**
 - Create a **custom network ACL** (Lab Network ACL).
 - By default, all traffic is denied.
 - Add **inbound and outbound rules** to allow **all traffic** initially.
 - Associate this ACL with the **Private Subnet**.

The screenshot shows the AWS VPC dashboard with the Network ACLs section. A green success message at the top states: "You have successfully updated subnet associations for acl-060c97985175c6b87 / Lab Network ACL." Below this, a table lists three Network ACLs:

Name	Network ACL ID	Associated with	Default	VPC ID	Inbo
acl-05669d05258d57cba	acl-05669d05258d57cba	6 Subnets	Yes	vpc-02732b9740d4330d4	2 Int
acl-00e385484b5c46011	acl-00e385484b5c46011	subnet-053ddf9bbc816df8e / Public Subnet	Yes	vpc-03f83f212156a6190 / Lab VPC	2 Int
Lab Network ACL	acl-060c97985175c6b87	subnet-098a3fbac037e091f / Private Subnet	No	vpc-03f83f212156a6190 / Lab VPC	2 Int

Below the table, the details for "acl-060c97985175c6b87 / Lab Network ACL" are shown, including its associated subnet and VPC ID.

The screenshot shows the AWS EC2 Instances page with the "Launch an instance" wizard. The "VPC - required" step is selected. The "Subnet" dropdown is set to "subnet-063df9bbc816df8e (Public Subnet)". The "Auto-assign public IP" dropdown is set to "Enable". The "Firewall (security groups)" dropdown has "Create security group" selected. The "Description - required" dropdown contains the text "launch-wizard-1 created 2025-04-09T16:17:43.227Z". On the right, the "Summary" section shows "Number of instances": 1, "Software Image (AMI)": Amazon Linux 2023 AMI 2023.7.2...read more, "Virtual server type (instance type)": t2.micro, and a note about the "Free tier".

Task 10: Testing Your Custom Network ACL

- **Purpose:** Demonstrate control over subnet traffic with the custom ACL.
- **Steps:**
 - Launch a **Test EC2 instance** in the **Public Subnet** with ICMP (ping) enabled via its **Security Group**.
 - From the **Private EC2 instance**, ping the **Test Instance's private IP** — it should respond.
 - Now **modify the ACL** to deny ICMP IPv4 to the Test Instance's private IP (/32).

- o Verify that ping **fails**, showing that traffic is blocked.

The screenshot shows the 'Edit inbound rules' section of the AWS VPC Network ACL configuration. It displays three rules:

Rule number	Type	Protocol	Port range	Source	Action
1	All traffic	All	All	0.0.0.0/0	Allow
2	All ICMP - IPv4	ICMP (1)	All	0.0.0.0/0	Allow
*	All traffic	All	All	0.0.0.0/0	Deny

Buttons at the bottom include 'Add new rule', 'Sort by rule number', 'Cancel', 'Preview changes', and 'Save changes'.

The screenshot shows a web browser window with three stacked question boxes. Each box contains a question, a list of options with one selected, and a 'Submit' button.

Question 4: Why do you use two different key pairs to access the private instance and the bastion host?

- Each instance needs a different key pair
- It provided practice with creating key pairs
- Separate key pairs could help reduce the impact of a compromised bastion host
- Key pairs can't be reused

Question 5: Can the bastion host use ping and get a reply from the instance in the private subnet?

- Yes
- No

Question 6: Which security group rules allow the private EC2 instance to receive the return traffic when it pings the test instance?

- Outbound on private and outbound on test
- Outbound on private and inbound on test
- Inbound on private and outbound on test
- Inbound on private and inbound on test

The screenshot shows a web browser window with three stacked question boxes, identical to the ones in the previous screenshot.

Question 1: What is the purpose of the internet gateway in the public subnet?

- Allows instances in the private subnet to obtain a public IP address
- Allows instances in the public subnet to obtain a public IP address
- Allows instances in the public subnet with a public IP address to communicate with the internet
- Allows instances in the private subnet with a public IP address to communicate with the internet

Question 2: What allows the instance in the private subnet to connect to the internet so that it can download updates?

- The internet gateway in the public subnet
- The NAT gateway
- The Elastic IP address
- The default network ACL

Question 3: Can the instance in the private subnet be accessed directly from the internet?

- Yes
- No

The screenshot shows the AWS Academy challenge interface for the 'Challenge (Café) lab: Creating a VPC Networking Environment for the Café'. The left sidebar includes links for Account, Dashboard, Courses, Calendar, Inbox, History, and Help. The main content area displays the challenge title and instructions. A central window titled 'AWS' contains a dropdown menu set to 'EN_US' and a question about the bastion host. Below the question are two numbered questions: 5 and 6. To the right of the question window is a 'Submission Report' panel showing a total score of 56/56. The report lists several tasks: [Task 1A] Public Subnet, [Task 1B] Lab VPC has IGW, [Task 1C] Public Subnet with Internet G, [Task 2] Bastion Host exists, and [Task 3] Bastion Host has public ip. At the bottom of the challenge page, there are instructions for submitting work, including steps 48 and 49. The task list at the bottom of the challenge page includes: 48. At the top of these instructions, choose **Submit** to record your progress, and when prompted, choose **Yes**. 49. If the results don't display after a couple of minutes, return to the top of these instructions and choose **Grades**.

What is the purpose of a Bastion Host?

To securely access private instances in a VPC by acting as a jump point from a public subnet.

Why do we need an Elastic IP address?

To provide a **static public IP** for the NAT Gateway so that private instances can access the internet reliably.

Why do we need to use SSH pass through in this lab?

To connect from your local machine → bastion host → private instance **without copying private keys** to the bastion host (enhances security using SSH agent forwarding).