

Project: Deploying VPC and EC2 Instance

***Project: AWS Virtual Private Cloud***

Aryan Vij

Date Started: 4/21/2024

Date Completed: 4/21/2024

## Table of Contents

<b>EXECUTIVE SUMMARY</b>	<b>3</b>
<b>PROCEDURES</b>	<b>4</b>
Sign in to the AWS Management Console	4
Navigate to VPC Dashboard	4
Create Subnets	4
Create Internet Gateway (IGW)	4
Modify Route Tables	5
Create NAT Gateway	5
Modify Route Tables for Private Subnets	5
Security Groups and Network ACLs	5
Review and Test	5
Navigate to EC2 Dashboard	5
Launch EC2 Instance	6
Configure Instance Details	6
Add Storage	6
Add Tags (Optional)	6
Configure Security Group	6
Review and Launch	7
Select Key Pair	7
SSH into the Instance	7
<b>RESULTS</b>	<b>8</b>
<b>BIBLIOGRAPHY</b>	<b>9</b>
<b>APPENDICES</b>	<b>11</b>
Appendix A: Visual Documentation	11

## EXECUTIVE SUMMARY

The document outlines a series of procedures for setting up a Virtual Private Cloud (VPC) in Amazon Web Services (AWS) and deploying an EC2 instance within it. Beginning with the creation of the VPC and definition of its IPv4 CIDR block, the process includes configuring subnets across multiple availability zones and establishing an internet gateway for external connectivity. Additionally, it involves setting up a NAT gateway in a public subnet to enable internet access for instances in private subnets, achieved through modifications to routing tables. Following this, the document details the launch of an EC2 instance within the configured VPC, accompanied by the configuration of security groups to regulate SSH access. Secure SSH connectivity to the EC2 instance is facilitated through the use of key pairs.

The outlined procedures culminate in a successful demonstration of the setup, with emphasis placed on network isolation, security, and connectivity. Results highlight the effective deployment of a VPC in AWS, complete with proper routing and secure access controls. The project's success is underscored by the operational efficiency achieved through the deployment of an EC2 instance within the VPC environment. Testing confirms the functionality of internet access from the EC2 instance, facilitated by the configured NAT gateway. Overall, the documented procedures provide a comprehensive guide for establishing and operating a secure and well-connected VPC environment in AWS, ensuring the reliability and functionality of cloud-based infrastructure.

## PROCEDURES

### **Sign in to the AWS Management Console**

1. Go to the AWS Management Console and sign in with your credentials.

### **Navigate to VPC Dashboard**

1. Once logged in, navigate to the VPC Dashboard by selecting *"Services"* from the top menu
2. Selecting *"VPC"* under the *"Networking & Content Delivery"* section.

### **Create VPC**

1. Click on the *"Create VPC"* button.
2. Specify a name for your VPC.
3. Define the IPv4 CIDR block for your VPC. This determines the range of IP addresses that can be used within your VPC. For example, 10.0.0.0/16.

### **Create Subnets**

1. In the VPC Dashboard, select *"Subnets"* from the left-hand menu.
2. Click on the *"Create subnet"* button.
3. Specify a name for the subnet.
4. Choose the VPC you created in the previous step.
5. Define the IPv4 CIDR block for the subnet. Ensure it falls within the CIDR block of your VPC.
6. Repeat this step to create additional subnets for different availability zones if needed.

### **Create Internet Gateway (IGW)**

1. In the VPC Dashboard, select *"Internet Gateways"* from the left-hand menu.
2. Click on the *"Create internet gateway"* button.
3. Give the internet gateway a name.
4. Select the newly created internet gateway and attach it to your VPC.

### **Modify Route Tables**

1. In the VPC Dashboard, select "*Route Tables*" from the left-hand menu.
2. Identify the route table associated with your VPC (usually named main).
3. Edit the route table and add a route to the internet gateway (0.0.0.0/0) to enable internet access from your subnets.

### **Create NAT Gateway**

1. In the VPC Dashboard, select "*NAT Gateways*" from the left-hand menu.
2. Click on the "*Create NAT Gateway*" button.
3. Choose the subnet where you want to place the NAT Gateway. This subnet should be a public subnet.
4. Select an Elastic IP address for the NAT Gateway.
5. Create the NAT Gateway.

### **Modify Route Tables for Private Subnets**

1. For each private subnet, modify the associated route table.
2. Add a route to the NAT Gateway (0.0.0.0/0) to enable internet access for instances in the private subnet.

### **Security Groups and Network ACLs**

1. Optionally, configure security groups and network ACLs to control inbound and outbound traffic to your instances.

### **Review and Test**

1. Review the configuration to ensure everything is set up correctly.
2. Test connectivity from instances in your subnets to the internet and other resources within and outside of your VPC.

### **Navigate to EC2 Dashboard**

1. Once logged in, navigate to the EC2 Dashboard by selecting *"Services"* from the top menu
2. Then select *"EC2"* under the *"Compute"* section

### **Launch EC2 Instance**

1. Click on the *"Launch Instance"* button.
2. Choose an Amazon Machine Image (AMI) based on your requirements (e.g., Amazon Linux 2, Ubuntu, etc.).
3. Select an instance type based on your workload needs. Click *"Next: Configure Instance Details"*.

### **Configure Instance Details**

1. Choose the VPC you created earlier from the *"Network"* dropdown menu.
2. Choose the subnet within the VPC where you want to launch the instance.
3. Optionally, configure additional settings such as IAM role, shutdown behavior, etc.
4. Click *"Next: Add Storage"*.

### **Add Storage**

1. Configure the size and type of the root volume (usually EBS).
2. Optionally, add additional volumes if needed.
3. Click *"Next: Add Tags"*.

### **Add Tags (Optional)**

1. Add any tags that are helpful for identifying your instance.
2. Click *"Next: Configure Security Group"*.

### **Configure Security Group**

1. Create a new security group or select an existing one.
2. Configure the inbound rules to allow SSH access (port 22) from your IP address or IP range.
3. Optionally, configure additional rules for other protocols and ports as needed.

## Project: Deploying VPC and EC2 Instance

4. Click *"Review and Launch"*.

### **Review and Launch**

1. Review the configuration to ensure everything is set up correctly.
2. Click *"Launch"*.

### **Select Key Pair**

1. Choose an existing key pair or create a new one.
2. Download the private key file (.pem) to your local machine.
3. Click *"Launch Instances"*.

### **SSH into the Instance**

1. Once the instance is launched and running, note its public IP address or DNS name from the EC2 Dashboard.
2. Open a terminal or SSH client on your local machine.
3. Change the permissions of the private key file with the command: `chmod 400 /path/to/your-key.pem`.
4. SSH into the instance using the public IP address or DNS name and the private key:
  - a. `ssh -i /path/to/your-key.pem ec2-user@<public_ip_address_or_dns_name>`
5. If you're using a different AMI, replace `ec2-user` with the appropriate username (e.g., `ubuntu` for Ubuntu AMIs, `centos` for CentOS AMIs, etc.).

## RESULTS

In this project, we successfully created a VPC in AWS and connected it to an EC2 instance. The key steps and outcomes include:

**VPC Creation:** Defined a VPC with a specified IPv4 CIDR block. Created subnets within the VPC across multiple availability zones. Established an internet gateway and attached it to the VPC for internet access. Configured routing tables to direct traffic within the VPC and to the internet gateway.

**NAT Gateway Setup:** Deployed a NAT gateway in a public subnet to allow instances in private subnets to access the internet while remaining secure. Modified routing tables for private subnets to route traffic through the NAT gateway.

**EC2 Instance Launch:** Launched an EC2 instance within the VPC. Configured security groups to allow SSH access (port 22) from specific IP addresses. Established a secure SSH connection to the EC2 instance using a key pair.

**Connectivity Testing:** Verified connectivity by SSHing into the EC2 instance remotely. Tested internet access from the EC2 instance to ensure proper NAT gateway functionality. Ensured that the EC2 instance was securely accessible and operational within the VPC environment.

Overall, the project successfully demonstrated the setup of a VPC in AWS and the deployment of an EC2 instance within that VPC, showcasing effective network isolation, security, and connectivity for cloud-based infrastructure.



## BIBLIOGRAPHY

App Store. (2016, December 5). *Termius - SSH & SFTP client*. Mac App Store.

<https://apps.apple.com/us/app/termius-ssh-sftp-client/id1176074088?mt=12>

Carl Oliver. (2013, September 24). *Subnetting made simple* [Video]. YouTube.

<https://www.youtube.com/watch?v=nFYilGQ-p-8>

CJ Saathoff. (2024, February). *Troubleshooting SSH Authentication: Understanding 'Permission Denied (publickey,gssapi-keyex,gssapi-with-mic).'* NameHero. Retrieved April 21, 2024, from

<https://www.namehero.com/blog/troubleshooting-ssh-authentication-understanding-permission-denied-publickeygssapi-keyexgssapi-with-mic/>

*Connect to your Linux instance from Linux or macOS using SSH - Amazon Elastic Compute Cloud.* (n.d.-a).

<https://docs.aws.amazon.com/AWSEC2/latest/UserGuide/connect-linux-inst-ssh.html#ssh-prereqs-linux-from-linux-macos>

*Connect to your Linux instance from Linux or macOS using SSH - Amazon Elastic Compute Cloud.* (n.d.-b).

<https://docs.aws.amazon.com/AWSEC2/latest/UserGuide/connect-linux-inst-ssh.html#connect-linux-inst-sshClient>

*How to Enable SSH on a Mac from the Command Line.* (2016, August 16). OS X Daily.

<https://osxdaily.com/2016/08/16/enable-ssh-mac-command-line/>

*How to fix SSH not working on MacOS Ventura / Sonoma.* (2023, November 13). OS X Daily.

<https://osxdaily.com/2022/12/22/fix-ssh-not-working-macos-rsa-issue/>

## Project: Deploying VPC and EC2 Instance

*Restart SSH on Mac Terminal (High Sierra)*. (n.d.). Gist.

<https://gist.github.com/influx6/46c39709a67f09908cc7542ca444fca2>

Stephane Maarek. (2019, June 11). *Amazon EC2 Basics & Instances Tutorial* [Video]. YouTube.

<https://www.youtube.com/watch?v=iHX-jtKIVNA>

Stephane Maarek. (2021, February 25). *SSH to EC2 Instances using Linux or Mac Tutorial*

[Video]. YouTube. [https://www.youtube.com/watch?v=8UqtMcX\\_kg0](https://www.youtube.com/watch?v=8UqtMcX_kg0)

*Terminate Amazon EC2 instances - Amazon Elastic Compute Cloud*. (n.d.).

<https://docs.aws.amazon.com/AWSEC2/latest/UserGuide/terminating-instances.html>

Tony Teaches Tech. (2020, September 6). *How to Use SSH on Your Mac with Terminal* [Video].

YouTube. <https://www.youtube.com/watch?v=SfTSBbaFN8Y>

*Tutorial: Get started with Amazon EC2 Linux instances - Amazon Elastic Compute Cloud*. (n.d.).

[https://docs.aws.amazon.com/AWSEC2/latest/UserGuide/EC2\\_GetStarted.html](https://docs.aws.amazon.com/AWSEC2/latest/UserGuide/EC2_GetStarted.html)

## APPENDICES

### Appendix A: Visual Documentation

These visuals offer a detailed representation of the tools used, commands executed, and findings discovered during the assessment. By including screenshots, readers can gain a clear understanding of the methodologies employed and the outcomes achieved during the testing phase.

The screenshot displays the AWS Management Console interface for the 'Subnets' page. The left-hand navigation pane includes sections for 'Virtual private cloud' (with options like VPC dashboard, EC2 Global View, and various VPC resources) and 'Security'. The main panel shows a list of subnets under the heading 'Subnets (1/6) Info'. A search bar is present above the table. The table lists six subnets, all in an 'Available' state. The first subnet, 'vpc-private-us-east-2a', is selected. Below the table, the 'Details' tab for the selected subnet is active, showing its Subnet ID and Subnet ARN.

	Name	Subnet ID	State
<input checked="" type="checkbox"/>	vpc-private-us-east-2a	<a href="#">subnet-01e2b93cc6edeaf21</a>	Available
<input type="checkbox"/>	vpc-private-us-east-2b	<a href="#">subnet-0e82fd97526b889e1</a>	Available
<input type="checkbox"/>	vpc-private-us-east-2c	<a href="#">subnet-0c5cdd79c1000dcc7</a>	Available
<input type="checkbox"/>	vpc-public-us-east-2a	<a href="#">subnet-0f2b0e177fb8b2fba</a>	Available
<input type="checkbox"/>	vpc-public-us-east-2b	<a href="#">subnet-0fbeca3d7f4e1b9d7</a>	Available
<input type="checkbox"/>	vpc-public-us-east-2c	<a href="#">subnet-0086735d2af713737</a>	Available

subnets-01e2b93cc6edeaf21 / vpc-private-us-east-2a

Details | Flow logs | Route table | Network ACL | CIDR res

Details

Subnet ID: subnet-01e2b93cc6edeaf21 | Subnet ARN: subnet-01e2b93cc6edeaf21:aws:ec2:us-east-2:123456789012

Figure 1: Configuring public and private subnets

## Project: Deploying VPC and EC2 Instance

The screenshot displays the AWS Management Console interface. The top navigation bar includes the AWS logo, 'Services', a search bar, and user information for 'aryanvij @ 9054-1802-5791' in the 'Ohio' region. The left-hand navigation pane lists various VPC services, with 'Internet gateways' highlighted. The main content area shows the details for the Internet Gateway 'igw-062e9b289485706f3 / VPC Internet Gateway'. The 'Details' section indicates the gateway is 'Attached' and shows its VPC ID as 'vpc-007ca4fccbf9a3a0 | VPC Personal Project'. The 'Tags' section shows a single tag with the key 'Name' and value 'VPC Internet Gateway'.

**igw-062e9b289485706f3 / VPC Internet Gateway**

**Details** Info

Internet gateway ID igw-062e9b289485706f3	State Attached
VPC ID <a href="#">vpc-007ca4fccbf9a3a0   VPC Personal Project</a>	Owner 905418025791

**Tags** Manage tags

Search tags

< 1 > ⚙

Key	Value
Name	VPC Internet Gateway

Figure 2: Setting up internet gateway

## Project: Deploying VPC and EC2 Instance

The screenshot displays the AWS Management Console interface. On the left, the navigation pane shows the 'Virtual private cloud' section expanded, with 'Elastic IPs' highlighted. The main content area is titled 'Elastic IP addresses (1/3)'. It includes a search bar with the placeholder 'Find resources by attribute or tag', a refresh button, an 'Actions' dropdown, and an 'Allocate Elastic IP address' button. Below this is a table listing three Elastic IP addresses:

	Name	Allocated IPv4 addr...	Type
<input checked="" type="checkbox"/>	us-east-2a	<a href="#">3.16.175.166</a>	Public
<input type="checkbox"/>	us-east-2b	<a href="#">3.20.33.252</a>	Public
<input type="checkbox"/>	us-east-2c	<a href="#">52.14.224.167</a>	Public

Below the table, the selected IP address '3.16.175.166' is shown. There are tabs for 'Summary' and 'Tags', with 'Summary' currently selected. The 'Summary' tab content is partially visible.

Figure 3: Configuring elastic IPs

## Project: Deploying VPC and EC2 Instance

The screenshot displays the AWS Management Console interface for a NAT gateway. On the left, a sidebar menu lists various VPC services, with 'NAT gateways' highlighted. The main content area shows the details for the NAT gateway 'nat-0cfa6f47ab42ea3a4' associated with the VPC 'public-us-east-2a'. The details are organized into two columns. The left column includes the NAT gateway ID, ARN, VPC, state (Pending), primary private IPv4 address, and creation time. The right column includes the connectivity type (Public), primary public IPv4 address, subnet, state message, primary network interface ID, and deletion status.

Details	
NAT gateway ID nat-0cfa6f47ab42ea3a4	Connectivity type Public
NAT gateway ARN arn:aws:ec2:us-east-2:905418025791:natgateway/nat-0cfa6f47ab42ea3a4	Primary public IPv4 address -
VPC vpc-007ca4fccbf9a3a0 / VPC Personal Project	Subnet subnet-0f2b0e177fb8b2fba / vpc-public-us-east-2a
State Pending	State message Info
Primary private IPv4 address -	Primary network interface ID -
Created Sunday, 21 April 2024 at 14:55:55	Deleted -

Figure 4: Setting up NAT gateways

### NAT gateway settings

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

public-us-east-2a

The name can be up to 256 characters long.

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

subnet-0f2b0e177fb8b2fba (vpc-public-us-east-2a) ▼

**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.

eipalloc-0e44935f5350f52a6 (us-east-2a) ▼

Allocate Elastic IP

► **Additional settings** [Info](#)

Figure 5: NAT gateway configuration

## Project: Deploying VPC and EC2 Instance

The screenshot shows the AWS Management Console interface for NAT gateways. On the left, the navigation pane is visible with options like VPC dashboard, EC2 Global View, and various VPC resources. The main content area is titled 'NAT gateways (3)' and includes a search bar and a table of existing gateways. Below the table, there is a 'Select a NAT gateway' section with three icons.

	Name	NAT gateway ID	Connectivity...	State
<input type="radio"/>	public-us-east-2a	<a href="#">nat-0cfa6f47ab42ea3a4</a>	Public	Available
<input type="radio"/>	public-us-east-2b	<a href="#">nat-054a1d3412de1cbe4</a>	Public	Pending
<input type="radio"/>	public-us-east-2c	<a href="#">nat-03e4ea043173d3332</a>	Public	Pending

Figure 6: Configured NAT

NAT gateways (3) Info								
Find resources by attribute or tag								
	Name	NAT gateway ID	Connectivity...	State	State message	Primary public I...	Primary private I..	
<input type="radio"/>	public-us-east-2a	<a href="#">nat-0cfa6f47ab42ea3a4</a>	Public	Available	-	<a href="#">3.16.175.166</a>	10.0.1.129	
<input type="radio"/>	public-us-east-2b	<a href="#">nat-054a1d3412de1cbe4</a>	Public	Available	-	<a href="#">3.20.33.252</a>	10.0.2.238	
<input type="radio"/>	public-us-east-2c	<a href="#">nat-03e4ea043173d3332</a>	Public	Pending	-	-	10.0.3.75	

Figure 7: Public and Private IP addresses



## Project: Deploying VPC and EC2 Instance

The screenshot displays the AWS Management Console interface. At the top, a green notification banner states: "Route table rtb-0ffbb8a0c0932995f | vpc-personal-project-public-us-east-1 was created successfully." The left-hand navigation pane is open, showing the "Virtual private cloud" section with "Route tables" selected. The main content area shows the details for the route table "rtb-0ffbb8a0c0932995f / vpc-personal-project-public-us-east-1".

**Details**

Route table ID	rtb-0ffbb8a0c0932995f	Main	No
VPC	vpc-007ca4fccbf9a3a0   VPC Personal Project	Owner ID	905418025791
Explicit subnet associations	—	Edge associations	—

**Routes**

Subnet associations | Edge associations | Route propagation | Tags

Routes (1) Both Edit routes

Filter routes

Destination	Target	Status	Propagated
10.0.0/16	local	Active	No

Figure 8: Configured route tables

## Project: Deploying VPC and EC2 Instance

The screenshot shows the AWS Management Console interface for editing a route table association. The breadcrumb navigation indicates the path: VPC > Subnets > subnet-0086735d2af713737 > Edit route table association. The main heading is 'Edit route table association' with an 'Info' link. Below this, there are two main sections: 'Subnet route table settings' and 'Routes (2)'. The 'Subnet route table settings' section contains a 'Subnet ID' field with the value 'subnet-0086735d2af713737' and a 'Route table ID' dropdown menu currently showing 'rtb-0ffbb8a0c0932995f (vpc-personal-project-public-us-east-1)'. The 'Routes (2)' section features a search bar with the placeholder 'Filter routes', a pagination control showing '1', and a table of routes. The table has two columns: 'Destination' and 'Target'. It lists two routes: one for '10.0.0.0/16' pointing to 'local', and another for '0.0.0.0/0' pointing to an Internet Gateway 'igw-062e9b289485706f3'. At the bottom right, there are 'Cancel' and 'Save' buttons.

**Subnet route table settings**

Subnet ID  
subnet-0086735d2af713737

Route table ID  
rtb-0ffbb8a0c0932995f (vpc-personal-project-public-us-east-1)

**Routes (2)**

Filter routes

Destination	Target
10.0.0.0/16	local
0.0.0.0/0	<a href="#">igw-062e9b289485706f3</a>

Cancel Save

Figure 9: route table association setup

## Project: Deploying VPC and EC2 Instance

Subnet (subnet-0fbeca3d7f4e1b9d7) has been successfully associated with route table (rtb-0ffbb8a0c0932995f).

**Subnets (1/6)** Info

Find resources by attribute or tag

	Name	Subnet ID	State	VPC
<input type="checkbox"/>	vpc-public-us-east-2c	<a href="#">subnet-0086735d2af713737</a>	Available	<a href="#">vpc-007ca4fccbf9a3a0</a>   <a href="#">VPC Personal Project</a>
<input checked="" type="checkbox"/>	vpc-public-us-east-2b	<a href="#">subnet-0fbeca3d7f4e1b9d7</a>	Available	<a href="#">vpc-007ca4fccbf9a3a0</a>   <a href="#">VPC Personal Project</a>
<input type="checkbox"/>	vpc-public-us-east-2a	<a href="#">subnet-0f2b0e177fb8b2fba</a>	Available	<a href="#">vpc-007ca4fccbf9a3a0</a>   <a href="#">VPC Personal Project</a>
<input type="checkbox"/>	vpc-private-us-east-2c	<a href="#">subnet-0c5cdd79c1000dcc7</a>	Available	<a href="#">vpc-007ca4fccbf9a3a0</a>   <a href="#">VPC Personal Project</a>

**subnet-0fbeca3d7f4e1b9d7 / vpc-public-us-east-2b**

Details | Flow logs | Route table | Network ACL | CIDR reservations | Sharing | Tags

**Details**

Subnet ID <a href="#">subnet-0fbeca3d7f4e1b9d7</a>	Subnet ARN <a href="#">arn:aws:ec2:us-east-2:905418025791:subnet/subnet-0fbeca3d7f4e1b9d7</a>	State Available	IPv4 CIDR <a href="#">10.0.2.0/24</a>
Available IPv4 addresses 250	IPv6 CIDR -	Availability Zone us-east-2b	Availability Zone ID use2-az2
VPC <a href="#">vpc-007ca4fccbf9a3a0</a>   <a href="#">VPC Personal Project</a>	Network ACL <a href="#">acl-01ba876fa60eebd1f</a>	Default subnet No	Customer-owned IPv4 pool -
Auto-assign public IPv4 address Yes	Route table <a href="#">rtb-0ffbb8a0c0932995f</a>   <a href="#">vpc-personal-project-public-us-east-1</a>	Auto-assign customer-owned IPv4 address No	IPv6-only No
Outpost ID	Auto-assign IPv6 address No	IPv6 CIDR reservations -	

Figure 10: Assigning subnets to route tables

## Project: Deploying VPC and EC2 Instance

The screenshot displays the AWS Management Console interface. On the left, the navigation pane shows the 'Virtual private cloud' section expanded, with 'Subnets' selected. The main content area is titled 'Subnets (1/6) Info'. It contains a table listing subnets. The subnet 'vpc-public-us-east-2a' with ID 'subnet-0f2b0e177fb8b2fba' is selected. Below the table, the 'Route table' tab is active for the selected subnet. It shows the route table 'rtb-0ffbb8a0c0932995f' associated with 'vpc-personal-project-public-us-east-1'. The 'Routes (2)' section lists two routes: a local route for '10.0.0.0/16' and an internet gateway route for '0.0.0.0/0' pointing to 'igw-062e9b289485706f3'.

Name	Subnet ID	State	VPC
vpc-public-us-east-2c	<a href="#">subnet-0086735d2af713737</a>	Available	<a href="#">vpc-007ca4fccbf9a3a0</a>   VPC
vpc-public-us-east-2b	<a href="#">subnet-0fbeca3d7f4e1b9d7</a>	Available	<a href="#">vpc-007ca4fccbf9a3a0</a>   VPC
<b>vpc-public-us-east-2a</b>	<b><a href="#">subnet-0f2b0e177fb8b2fba</a></b>	Available	<a href="#">vpc-007ca4fccbf9a3a0</a>   VPC
vpc-private-us-east-2c	<a href="#">subnet-0c5cdd79c1000dcc7</a>	Available	<a href="#">vpc-007ca4fccbf9a3a0</a>   VPC
vpc-private-us-east-2b	<a href="#">subnet-0e82fd97526b889e1</a>	Available	<a href="#">vpc-007ca4fccbf9a3a0</a>   VPC
vpc-private-us-east-2a	<a href="#">subnet-01e2b93cc6edeaf21</a>	Available	<a href="#">vpc-007ca4fccbf9a3a0</a>   VPC

  

Destination	Target
10.0.0.0/16	local
0.0.0.0/0	<a href="#">igw-062e9b289485706f3</a>

Figure 11: Route tables for subnets

## Project: Deploying VPC and EC2 Instance

The screenshot displays the AWS Management Console interface. On the left, a navigation sidebar lists various services under categories like 'Virtual private cloud', 'Security', and 'DNS firewall'. The main content area shows the details for a specific route table, 'rtb-055d3587845771074', which is associated with the VPC 'vpc-007ca4fccbf9a3a0'. Below the details, there are tabs for 'Routes', 'Subnet associations', 'Edge associations', 'Route propagation', and 'Tags'. The 'Routes' tab is active, showing a table with two routes. The first route has a destination of '0.0.0.0/0' and a target of 'nat-0cfa6f47ab42ea3a4'. The second route has a destination of '10.0.0.0/16' and a target of 'local'. Both routes are in an 'Active' status.

VPC dashboard ✕  
EC2 Global View   
Filter by VPC:

▼ Virtual private cloud  
Your VPCs  
Subnets  
**Route tables**  
Internet gateways  
Egress-only Internet gateways  
DHCP option sets  
Elastic IPs  
Managed prefix lists  
Endpoints  
Endpoint services  
NAT gateways  
Peering connections

▼ Security  
Network ACLs  
Security groups

▼ DNS firewall  
Rule groups  
Domain lists

▼ Network Firewall

VPC > Route tables > rtb-055d3587845771074

### rtb-055d3587845771074 / vpc-personal-project-private-us-east-2a

Actions ▼

**Details** [Info](#)

Route table ID rtb-055d3587845771074	Main No	Explicit subnet associations -	Edge associations -
VPC <a href="#">vpc-007ca4fccbf9a3a0</a>   VPC Personal Project	Owner ID 905418025791		

**Routes** | Subnet associations | Edge associations | Route propagation | Tags

**Routes (2)** Both ▼ Edit routes

Destination ▼	Target ▼	Status ▼	Propagated ▼
0.0.0.0/0	<a href="#">nat-0cfa6f47ab42ea3a4</a>	✔ Active	No
10.0.0.0/16	local	✔ Active	No

Figure 12: Assigning route tables to subnets

## Project: Deploying VPC and EC2 Instance

The screenshot displays the AWS Management Console interface for configuring a VPC Security Group. The left-hand navigation pane is open, showing the 'Virtual private cloud' section with options like 'Your VPCs', 'Subnets', and 'Route tables'. The 'Security' section is also visible, including 'Network ACLs', 'Security groups', 'DNS firewall', and 'Network Firewall'. The main content area is titled 'sg-0aa607dc91883e929 - VPC Personal Project Security group'. Below the title, there's a 'Details' section with a grid of information: Security group name (VPC Personal Project Security group), Security group ID (sg-0aa607dc91883e929), Description (controls the traffic that is allowed to reach and leave the resources that it is associated with), VPC ID (vpc-007ca4fccbf9a3a0), Owner (905418025791), Inbound rules count (4 Permission entries), and Outbound rules count (1 Permission entry). Below the details, there are tabs for 'Inbound rules', 'Outbound rules', and 'Tags'. The 'Inbound rules' tab is active, showing a table with 4 rules. The table has columns for Name, Security group rule..., IP version, Type, and Protocol. The rules are: 1. Name: -, Security group rule: sgr-00ca303549758bd..., IP version: IPv4, Type: SSH, Protocol: TCP. 2. Name: -, Security group rule: sgr-02d1f838698dea1fc, IP version: IPv4, Type: HTTP, Protocol: TCP. 3. Name: -, Security group rule: sgr-0ed66e34b125bc0ea, IP version: IPv4, Type: All ICMP - IPv4, Protocol: ICMP. 4. Name: -, Security group rule: sgr-08cf86864dc18c131, IP version: IPv4, Type: HTTPS, Protocol: TCP.

**Details**

Security group name VPC Personal Project Security group	Security group ID sg-0aa607dc91883e929	Description controls the traffic that is allowed to reach and leave the resources that it is associated with	VPC ID vpc-007ca4fccbf9a3a0
Owner 905418025791	Inbound rules count 4 Permission entries	Outbound rules count 1 Permission entry	

**Inbound rules (4)**

Name	Security group rule...	IP version	Type	Protocol
-	sgr-00ca303549758bd...	IPv4	SSH	TCP
-	sgr-02d1f838698dea1fc	IPv4	HTTP	TCP
-	sgr-0ed66e34b125bc0ea	IPv4	All ICMP - IPv4	ICMP
-	sgr-08cf86864dc18c131	IPv4	HTTPS	TCP

Figure 13: Security group setup

## Project: Deploying VPC and EC2 Instance

**Key pair (login)** [Info](#)

You can use a key pair to securely connect to your instance. Ensure that you have access to the selected key pair before you launch the instance.

Key pair name - required

Select [Create new key pair](#)

**Network settings** [Info](#)

**VPC - required** [Info](#)

vpc-007ca4fccbf9a3a0 (VPC Personal Project)  
10.0.0.0/16 [Create new VPC](#)

**Subnet** [Info](#)

subnet-0f2b0e177fb8b2fba vpc-public-us-east-2a  
VPC: vpc-007ca4fccbf9a3a0 Owner: 905418025791 Availability Zone: us-east-2a  
IP addresses available: 250 CIDR: 10.0.1.0/24 [Create new subnet](#)

**Auto-assign public IP** [Info](#)

Enable [Additional charges apply](#) when outside of [free tier allowance](#)

**Firewall (security groups)** [Info](#)

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

**Common security groups** [Info](#)

Select security groups

VPC Personal Project Security group sg-0aa607dc91883e929 [Compare security group rules](#)  
VPC: vpc-007ca4fccbf9a3a0

Security groups that you add or remove here will be added to or removed from all your network interfaces.

[Advanced network configuration](#)

**Summary**

Number of instances [Info](#)

1

**Software Image (AMI)**

Amazon Linux 2023 AMI 2023.4.2...[read more](#)  
ami-09b90e09742640522

**Virtual server type (instance type)**

t2.micro

**Firewall (security group)**

VPC Personal Project Security group

**Storage (volumes)**

1 volume(s) - 8 GiB

**Free tier:** In your first year includes 750 hours of t2.micro (or t3.micro in the Regions in which t2.micro is unavailable) instance usage on free tier AMIs per month, 750 hours of public IPv4 address usage per month, 30 GiB of EBS storage, 2 million I/Os, 1 GB of snapshots, and 100 GB of bandwidth to the internet.

[Cancel](#) [Launch instance](#) [Review commands](#)

Figure 14: Launching EC2 instance network configuration

## Project: Deploying VPC and EC2 Instance

```
aws
[Option+S]
Ohio
aryanjv @ 9054-1802-5791
EC2
Amazon Linux 2023
https://aws.amazon.com/linux/amazon-linux-2023
[ec2-user@ip-10-0-1-245 ~]$ ifconfig
enX0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 9001
    inet 10.0.1.245 netmask 255.255.255.0 broadcast 10.0.1.255
    inet6 fe80::a9:d8ff:feef:f7 prefixlen 64 scopeid 0x20<link>
    ether 02:a9:d8:ef:00:f7 txqueuelen 1000 (Ethernet)
    RX packets 4039 bytes 24948613 (23.7 MiB)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 1490 bytes 155279 (151.6 KiB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

lo: flags=73<UP,LOOPBACK,RUNNING> mtu 65536
    inet 127.0.0.1 netmask 255.0.0.0
    inet6 ::1 prefixlen 128 scopeid 0x10<host>
    loop txqueuelen 1000 (Local Loopback)
    RX packets 12 bytes 1020 (1020.0 B)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 12 bytes 1020 (1020.0 B)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

[ec2-user@ip-10-0-1-245 ~]$
```

Figure 15: Launching EC2 instance locally



## Project: Deploying VPC and EC2 Instance

```

Downloads — ec2-user@ip-10-0-1-245:~ — ssh -i key-pair-public-2a.pem ec2-user@ec2-...
...ssh -i key-pair-public-2a.pem ec2-user@ec2-3-23-101-249.us-east-2.compute.amazonaws.com +
Last login: Sun Apr 21 15:57:00 on ttys000
[aryanvij@MacBook-Pro-4 ~ % cd Downloads
[aryanvij@MacBook-Pro-4 Downloads % chmod 400 "key-pair-public-2a.pem"
[aryanvij@MacBook-Pro-4 Downloads % ssh -i "key-pair-public-2a.pem" ec2-user@ec2-3-23-101-249.us-east-2.compute.amazonaws.com

zsh: command not found: ec2-3-23-101-249.us-east-2.compute.amazonaws.com
[aryanvij@MacBook-Pro-4 Downloads % ssh -i "key-pair-public-2a.pem" ec2-user@ec2-3-23-101-249.us-east-2.compute.amazonaws.com

#_
~\_##### Amazon Linux 2023
~~\_#####\
~~\_###|
~~\_#/ https://aws.amazon.com/linux/amazon-linux-2023
~~V~'-'>
~~~~
~~~.-.-
~/m/'

Last login: Sun Apr 21 19:59:22 2024 from 65.175.28.225
[ec2-user@ip-10-0-1-245 ~]$
```

Figure 16: Remote SSH into EC2 instance