

1. VPC Configuration

- Objective: To understand the fundamentals of AWS networking through the configuration of a Virtual Private Cloud (VPC).
- Approach: Students will create a new VPC, add subnets, set up an Internet Gateway, and configure route tables. The lab might also include setting up a simple EC2 instance within this VPC to demonstrate how resources are deployed in a custom network environment.
- Goal: By the end of this lab, students should be able to create and configure a VPC, understand subnetting, and the role of route tables and internet gateways in AWS.

1.1. Create VPC

Navigate to VPC dashboard and click “Create VPC” button to create a new VPC

The screenshot shows the 'Create VPC' wizard in the AWS VPC dashboard. The 'VPC settings' step is active. Key configuration options shown include:

- Resources to create:** A radio button group where 'VPC only' is selected.
- Name tag - optional:** A text input field containing 'MyVPC'.
- IPv4 CIDR block:** A text input field containing '10.0.0.0/16'. A note below states: "CIDR block size must be between /16 and /28."
- IPv6 CIDR block:** A radio button group where 'No IPv6 CIDR block' is selected.
- Tenancy:** A dropdown menu set to 'Default'.
- Tags:** A section for adding tags. One tag is present: 'Name' with value 'MyVPC'. Buttons for 'Add tag' and 'Remove tag' are available.

At the bottom right of the wizard, there are 'Cancel' and 'Create VPC' buttons.

1.2. Create Subnets

Creating two subnetworks and providing the required configurations

VPC > Subnets > Create subnet

Create subnet Info

VPC

VPC ID
Create subnets in this VPC.

vpc-0c81533c6ab6c8b1e (MyVPC) ▲

| (default)

vpc-0d8fab51a5f972f19
172.31.0.0/16

vpc-0c81533c6ab6c8b1e (MyVPC)
10.0.0.0/16 ✓

vpc-06e62a52af76c05d5 (basic-vpc)
10.0.0.0/25

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

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

Subnet 1 of 2

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

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.
 US East (N. Virginia) / us-east-1a ▼

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.
 10.0.0.0/16 ▼

IPv4 subnet CIDR block
 10.0.0.0/24 256 IPs

< > ^ ^

Tags - optional

Key	Value - optional	Remove
<input type="text"/> Name X	<input type="text"/> MyPublicSubnet X	Remove
Add new tag		
You can add 49 more tags.		
Remove		

Subnet 2 of 2

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

< > ^ v

Tags - optional

Key	Value - optional	Remove
<input type="text" value="Name"/>	<input type="text" value="MyPrivateSubnet"/>	<input type="button" value="Remove"/>

You can add 49 more tags.

The list of created subnets can be viewed, and their state is Available.

⌚ You have successfully created 2 subnets: subnet-00f735d4cd35482bd, subnet-0992acbaf36b3e7b

Subnets (2) [Info](#)

<input type="checkbox"/>	Name	Subnet ID	State	VPC	IPv4 CIDR	IPv6 CIDR	Available IPv4 addr
<input type="checkbox"/>	MyPrivateSubnet	subnet-0992acbaf36b3e7b	Available	vpc-0c81533c6ab6c8b1e MyVPC	10.0.1.0/24	-	251
<input type="checkbox"/>	MyPublicSubnet	subnet-00f735d4cd35482bd	Available	vpc-0c81533c6ab6c8b1e MyVPC	10.0.0.0/24	-	251

1.3. Create Internet Gateway

Internet gateway is created and attached to the newly created VPC.

VPC > Internet gateways > Create internet gateway

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="my_internet_gateway"/>

Add new tag
You can add 49 more tags.

Cancel **Create internet gateway**

VPC > Internet gateways > igw-06bc383a1d6bbbaa1 / my_internet_gateway

igw-06bc383a1d6bbbaa1 / my_internet_gateway

Details Info

Internet gateway ID <input type="text" value="igw-06bc383a1d6bbbaa1"/>	State <input type="text" value="Detached"/>	VPC ID <input type="text" value="-"/>	Owner <input type="text" value="612362567483"/>
---	--	--	--

Actions

- Attach to VPC
- Detach from VPC
- Manage tags
- Delete

Tags

Key	Value
Name	my_internet_gateway

Manage tags

VPC > Internet gateways > Attach to VPC (igw-06bc383a1d6bbbaa1) Info

Attach to VPC (igw-06bc383a1d6bbbaa1)

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.

AWS Command Line Interface command

Cancel **Attach internet gateway**

1.4. Create Route Table

Then the route tables are created. Here, created VPC is selected in configuration.

VPC > Route tables > Create route table

Create route table Info

A route table specifies how packets are forwarded between the subnets within your VPC, the internet, and your VPN connection.

Route table settings

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

VPC
The VPC to use for this route table.

Tags
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="myvpc_table1"/>

Add new tag
You can add 49 more tags.

Cancel **Create route table**

VPC > Route tables > rtb-0e280208bb9e9a669
rtb-0e280208bb9e9a669 / myvpc_table1 Actions ▾

Details Info

Route table ID <input type="text" value="rtb-0e280208bb9e9a669"/>	Main <input checked="checked" type="checkbox"/>	Explicit subnet associations -	Edge associations -
VPC <input type="text" value="vpc-0c81533c6ab6c8b1e MyVPC"/>	No	Owner ID <input type="text" value="612362567483"/>	

Routes Subnet associations Edge associations Route propagation Tags

Routes (1)		Edit routes	
<input type="text" value="Filter routes"/>		Both	< 1 > <small>○</small>
Destination <input type="text" value="10.0.0.0/16"/>	Target <input type="text" value="local"/>	Status <small>Active</small>	Propagated <input type="checkbox"/>

1.5. Subnet Associations

Subnet association of route table is done so that it can be exposed to internet.

VPC > Route tables > rtb-0e280208bb9e9a669 > Edit subnet associations

Edit subnet associations

Change which subnets are associated with this route table.

Available subnets (1/2)					
<input type="text"/> Filter subnet associations					
Name	Subnet ID	IPv4 CIDR	IPv6 CIDR	Route table ID	
<input type="checkbox"/> MyPrivateSubnet	subnet-0992acbfa36b3e7b	10.0.1.0/24	-	Main (rtb-07c973a324cc5e285)	
<input checked="" type="checkbox"/> MyPublicSubnet	subnet-00f735d4cd35482bd	10.0.0.0/24	-	Main (rtb-07c973a324cc5e285)	

Selected subnets

subnet-00f735d4cd35482bd / MyPublicSubnet

Cancel

VPC > Route tables > rtb-0e280208bb9e9a669 / myvpc_table1

rtb-0e280208bb9e9a669 / myvpc_table1

Actions ▾

Details	Info
Route table ID rtb-0e280208bb9e9a669	Main No
VPC vpc-0c81533c6ab6c8b1e MyVPC	Owner ID 612362567483
<input type="button"/> Routes	<input type="button" value="Subnet associations"/> <input type="button"/> Edge associations <input type="button"/> Route propagation <input type="button"/> Tags

Subnet associations (1)

Explicit subnet associations (1)					
<input type="text"/> Find subnet association <input type="button"/> Edit subnet associations					
Name	Subnet ID	IPv4 CIDR	IPv6 CIDR	Route table ID	
<input type="checkbox"/> MyPublicSubnet	subnet-00f735d4cd35482bd	10.0.0.0/24	-	Main (rtb-07c973a324cc5e285)	

Subnets without explicit associations (1)

The following subnets have not been explicitly associated with any route tables and are therefore associated with the main route table:

Subnets without explicit associations (1)					
<input type="text"/> Find subnet association <input type="button"/> Edit subnet associations					
Name	Subnet ID	IPv4 CIDR	IPv6 CIDR	Route table ID	
<input type="checkbox"/> MyPrivateSubnet	subnet-0992acbfa36b3e7b	10.0.1.0/24	-	Main (rtb-07c973a324cc5e285)	

1.6. Edit routes of route table

The created route table is edited to associate with internet gateway.

VPC > Route tables > rtb-0e280208bb9e9a669 > Edit routes

Edit routes

Destination	Target	Status	Propagated
10.0.0.0/16	local	Active	No
Q 0.0.0.0/0	Internet Gateway	-	No
	Q igw-06bc383a1d6bbbba1	X	

Add route Remove Cancel Preview Save changes

VPC > Route tables > rtb-0e280208bb9e9a669

rtb-0e280208bb9e9a669 / myvpc_table1

Actions ▾

Details		Info	
Route table ID	rtb-0e280208bb9e9a669	Main	No
VPC	vpc-0c81533c6ab6c8b1e MyVPC	Owner ID	612362567483
		Explicit subnet associations	subnet-00f735d4cd35482bd / MyPublicSubnet
		Edge associations	-

Routes (2) Both ▾ Edit routes < 1 > ⌂

Destination	Target	Status	Propagated
0.0.0.0/0	igw-06bc383a1d6bbbba1	Active	No
10.0.0.0/16	local	Active	No

1.7. Testing with Linux EC2 instance

The new EC2 instance is launched to host a static website and configured.

The screenshot shows the AWS EC2 'Launch an instance' wizard. It consists of two main panels: 'Launch an instance' on the left and 'Summary' on the right.

Launch an instance (Info)

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 (Info)

Name: myNewWebServer [Add additional tags](#)

Application and OS Images (Amazon Machine Image) (Info)

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 our full catalog including 1000s of application and OS images

Recent AMIs

- Amazon Linux (aws logo)
- macOS (Mac logo)
- Ubuntu (Ubuntu logo)
- Windows (Microsoft logo)
- Red Hat (Red Hat logo)
- SUSE Linux (SUSE logo)

Browse more AMIs

Including AMIs from AWS, Marketplace and the Community

Amazon Machine Image (AMI)

Amazon Linux 2023 AMI

ami-0440d3b780d96b29d (64-bit (x86), uefi-preferred) / ami-0f93c02efd1974b8b (64-bit (Arm), uefi)
Virtualization: hvm ENA enabled: true Root device type: ebs

Description

Amazon Linux 2023 AMI 2023.3.20240219.0 x86_64 HVM kernel-6.1

Summary

Number of instances: [Info](#)
1

Software Image (AMI): Amazon Linux 2023 AMI 2023.3.2... [read more](#)
ami-0440d3b780d96b29d

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 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, 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](#)

In network setting, it is ensured that the above created VPC and public subnet is chosen. Auto assign public IP is enabled and new security group with http and ssh inbound rule is created.

▼ Network settings [Info](#)

VPC - required [Info](#)
vpc-0c81533c6ab6c8b1e (MyVPC)
10.0.0.0/16

Subnet [Info](#)
subnet-00f735d4cd35482bd MyPublicSubnet
VPC: vpc-0c81533c6ab6c8b1e Owner: 612362567483 Availability Zone: us-east-1a
IP addresses available: 251 CIDR: 10.0.0.0/24

Create new subnet [Create new subnet](#)

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

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

Security group name - required
my_vpc_ssh

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 [Info](#)
launch-wizard-5 created 2024-02-26T06:06:27.709Z

Inbound Security Group Rules

▼ Security group rule 1 (TCP, 22, 0.0.0.0/0)

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

Source type [Info](#) Source [Info](#) Description - optional [Info](#)
Anywhere e.g. SSH for admin desktop
0.0.0.0/0 [X](#)

Remove

▼ Security group rule 2 (TCP, 80, 0.0.0.0/0)

Type [Info](#) Protocol [Info](#) Port range [Info](#)
HTTP TCP 80

Source type [Info](#) Source [Info](#) Description - optional [Info](#)
Anywhere e.g. SSH for admin desktop
0.0.0.0/0 [X](#)

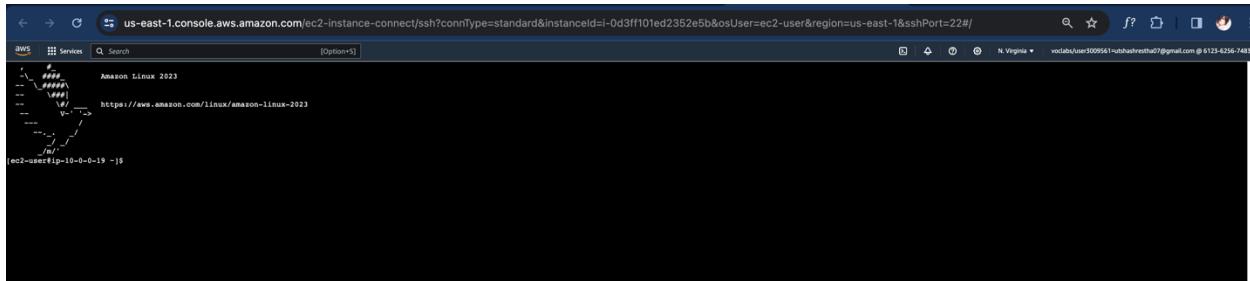
Remove

⚠ Rules with source of 0.0.0.0/0 allow all IP addresses to access your instance. We recommend setting security group rules to allow access from known IP addresses only. [X](#)

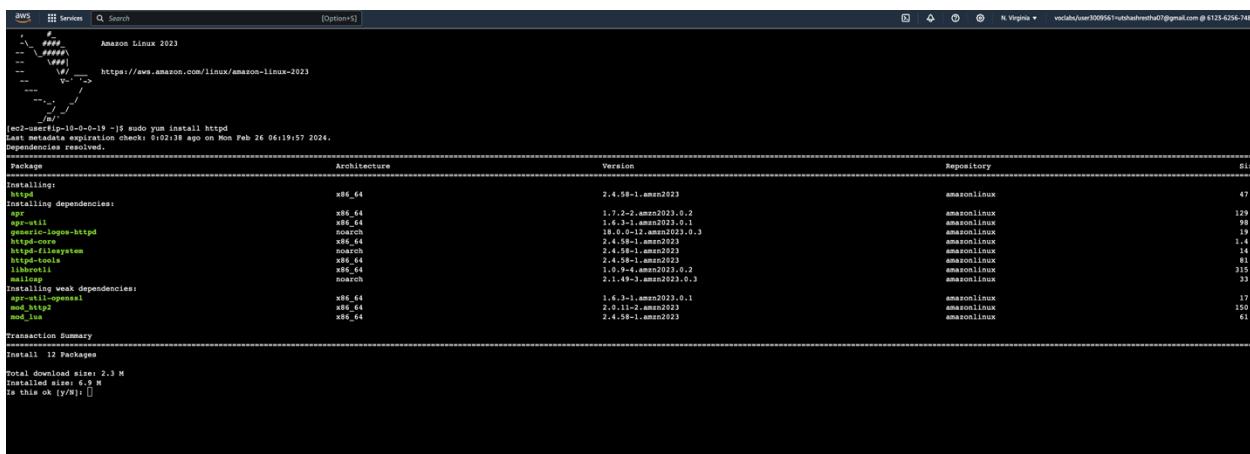
Add security group rule

► Advanced network configuration

After launch of EC2 instance, it is connected from browser. The server of EC2 instance is started and tested in the browser with public IP of the EC2 instance. The followed steps can be viewed in below screenshots.



```
[ec2-user@ip-10-0-0-19 ~]$
```

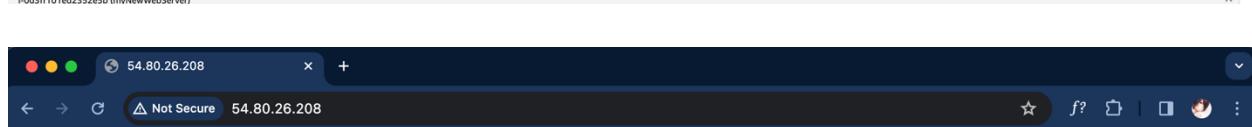


```
[ec2-user@ip-10-0-0-19 ~]$ sudo yum install httpd
Last metadata expiration check: 5:02:38 ago on Mon Feb 26 06:19:57 2024.
Dependencies resolved.
=====
Transaction Summary
=====
Installing:
httpd.x86_64 2.4.58-1.amzn2023
Installing dependencies:
apr.x86_64 1.7.2-2.amzn2023
apr-util.x86_64 1.6.3-1.amzn2023
openssl.x86_64 2.4.58-1.amzn2023
openssl-libs.x86_64 2.4.58-1.amzn2023
httpd-filesystem.x86_64 2.4.58-1.amzn2023
httpd-tools.x86_64 1.1.1-1.amzn2023
httpd-tools.x86_64 1.1.1-1.amzn2023
mailcap.x86_64 1.1.4-4.amzn2023
mailcap.x86_64 2.1.49-3.amzn2023
Installing weak dependencies:
apr-devel.x86_64 1.6.3-1.amzn2023
mod_http2.x86_64 2.0.11-2.amzn2023
mod_lua.x86_64 2.4.58-1.amzn2023
=====
Transaction Summary
=====
Total download size: 2.3 M
Installed size: 4.5 M
Is this ok [y/N]:
```

```

transaction Summary
=====
Install 12 Packages
=====
Total download size: 2.3 M
Installed sizes 6.9 M
Is this ok [y/N]: y
Downloaded packages:
(1/12) mod_lua-2.0.1-amzn2023.x86_64.rpm
(2/12) apr-util-1.6.3-1.amzn2023.0.1.x86_64.rpm
(3/12) apr-util-openssl-1.6.3-1.amzn2023.0.1.x86_64.rpm
(4/12) httpd-tools-2.4.58-1.amzn2023.x86_64.rpm
(5/12) httpd-core-2.4.58-1.amzn2023.x86_64.rpm
(6/12) libaprutil-1.6.3-1.amzn2023.x86_64.rpm
(7/12) mod_http2-2.0.1-2.amzn2023.x86_64.rpm
(8/12) httpd-2.4.58-1.amzn2023.x86_64.rpm
(9/12) generic-logos-httpsd-18.0.0-12.amzn2023.0.3.noarch.rpm
(10/12) mailcap-2.1.49-3.amzn2023.0.3.noarch.rpm
(11/12) httpd-filesystem-2.4.58-1.amzn2023.noarch.rpm
(12/12) mod_lua-2.0.1-amzn2023.x86_64.rpm
=====
Total
Running transaction check
Transaction check succeeded.
Running transaction test
transaction test succeeded.
Running transaction
Preparing : apr-1.7.2-2.amzn2023.0.2.x86_64
Installing : apr-util-openssl-1.6.3-1.amzn2023.0.1.x86_64
Installing : apr-util-1.6.3-1.amzn2023.0.1.x86_64
Installing : apr-util-openssl-1.6.3-1.amzn2023.0.1.x86_64
Installing : httpd-tools-2.4.58-1.amzn2023.x86_64
Running scriptlet: httpd-filesystem-2.4.58-1.amzn2023.noarch
Installing : httpd-2.4.58-1.amzn2023.x86_64
Installing : httpd-core-2.4.58-1.amzn2023.x86_64
Installing : mod_http2-2.0.1-2.amzn2023.x86_64
Installing : libaprutil-1.6.3-1.amzn2023.x86_64
Installing : generic-logos-httpsd-18.0.0-12.amzn2023.0.3.noarch
Installing : httpd-filesystem-2.4.58-1.amzn2023.x86_64
Installing : mod_lua-2.0.1-2.amzn2023.x86_64
Running scriptlet: httpd-2.4.58-1.amzn2023.x86_64
Verifying : httpd-2.4.58-1.amzn2023.x86_64
Verifying : generic-logos-httpsd-18.0.0-12.amzn2023.0.3.noarch
Verifying : httpd-filesystem-2.4.58-1.amzn2023.x86_64
Verifying : mod_lua-2.0.1-2.amzn2023.x86_64
Verifying : libaprutil-1.6.3-1.amzn2023.x86_64
Verifying : httpd-tools-2.4.58-1.amzn2023.x86_64
Verifying : mod_http2-2.0.1-2.amzn2023.x86_64
Verifying : apr-1.7.2-2.amzn2023.0.2.x86_64
Verifying : apr-util-openssl-1.6.3-1.amzn2023.0.1.x86_64
Verifying : generic-logos-httpsd-18.0.0-12.amzn2023.0.3.noarch
Verifying : mailcap-2.1.49-3.amzn2023.0.3.noarch
Verifying : httpd-filesystem-2.4.58-1.amzn2023.noarch
=====
Installed:
  apr-1.7.2-2.amzn2023.0.2.x86_64           apr-util-1.6.3-1.amzn2023.0.1.x86_64   apr-util-openssl-1.6.3-1.amzn2023.0.1.x86_64   generic-logos-httpsd-18.0.0-12.amzn2023.0.3.noarch   httpsd-2.4.58-1.amzn2023.x86_64   httpd-core-2.4.58-1.amzn2023.x86_64
  httpd-filesystem-2.4.58-1.amzn2023.noarch   httpd-tools-2.4.58-1.amzn2023.x86_64    libaprutil-1.6.3-1.amzn2023.x86_64   mailcap-2.1.49-3.amzn2023.0.3.noarch   mod_http2-2.0.1-2.amzn2023.x86_64   mod_lua-2.0.1-amzn2023.x86_64
=====
Complete!
[ec2-user@ip-10-0-0-19 ~] 

```



It works!

1.8. Testing with windows EC2 instance

In this, same setup is made as mentioned in Linux EC2 instance. In network setting rdp and http inbound rules are given.

The screenshot shows the 'Launch an instance' wizard on the AWS Management Console. The process is at step 2, 'Configure instance details'. The configuration includes:

- Name and tags:** Name is set to 'window-server'. There is an option to 'Add additional tags'.
- Application and OS Images (Amazon Machine Image):** The search bar shows 'Search our full catalog including 1000s of application and OS images'. Under 'Quick Start', there are icons for Amazon Linux, macOS, Ubuntu, Windows, Red Hat, and SUSE Linux. A 'Browse more AMIs' button is available.
- Amazon Machine Image (AMI):** The selected AMI is 'Microsoft Windows Server 2022 Base' (ami-0f9c44e98edf38a2b). It is marked as 'Free tier eligible'. The description states it is a 'Microsoft Windows Server 2022 Full Locale English AMI provided by Amazon'. The architecture is listed as '64-bit (x86)' and the AMI ID is 'ami-0f9c44e98edf38a2b'. A 'Verified provider' badge is present.
- Summary:** Shows the configuration summary with 1 instance, Microsoft Windows Server 2022 AMI, t2.micro instance type, New security group, and 1 volume(s) - 30 GiB storage.
- Free tier information:** A callout box details the free tier benefits: 750 hours of t2.micro (or t3.micro in regions where t2.micro is unavailable) instance usage on free tier AMIs per month, 30 GiB of EBS storage, 2 million IOPS, 1 GB of snapshots, and 100 GB of bandwidth to the internet.
- Buttons:** 'Cancel', 'Launch instance' (highlighted in orange), and 'Review commands'.

▼ Network settings [Info](#)

VPC - required [Info](#)

vpc-0c81533c6ab6c8b1e (MyVPC)
10.0.0.0/16

Subnet [Info](#)

subnet-00f735d4cd35482bd MyPublicSubnet
VPC: vpc-0c81533c6ab6c8b1e Owner: 612362567483 Availability Zone: us-east-1a
IP addresses available: 249 CIDR: 10.0.0.0/24

Create new subnet [Create new subnet](#)

Auto-assign public IP [Info](#)

Enable

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

Security group name - required

launch-wizard-5

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 [Info](#)

launch-wizard-5 created 2024-02-26T06:50:36.224Z

Inbound Security Group Rules

▼ Security group rule 1 (TCP, 3389, 0.0.0.0/0)

Type [Info](#) Protocol [Info](#) Port range [Info](#)
rdp TCP 3389

Source type [Info](#) Source [Info](#) Description - optional [Info](#)
Anywhere e.g. SSH for admin desktop
0.0.0.0/0 [X](#)

▼ Security group rule 2 (TCP, 80, 0.0.0.0/0)

Type [Info](#) Protocol [Info](#) Port range [Info](#)
HTTP TCP 80

Source type [Info](#) Source [Info](#) Description - optional [Info](#)
Anywhere e.g. SSH for admin desktop
0.0.0.0/0 [X](#)

⚠️ Rules with source of 0.0.0.0/0 allow all IP addresses to access your instance. We recommend setting security group rules to allow access from known IP addresses only. [X](#)

[Add security group rule](#)

Number of instances [Info](#)
1

Software Image (AMI) Microsoft Windows Server 2024-02-26 ami-0f9c44e98edf38a2b

Virtual server type (instance type) t2.micro

Firewall (security group) New security group

Storage (volumes) 1 volume(s) - 30 GiB

ⓘ **Free tier:** In your first 750 hours of t2.micro usage in the Regions included in the Free Tier, you get 100 hours of compute time, 100 hours of storage, 2 million database transactions, 20 snapshots, and 100 gigabytes of data transfer to and from the internet.

[Cancel](#)

After launching the windows EC2 instance, the EC2 instance is connected from RDP client. The rdp file is downloaded and opened locally using the generated password.

Connect to instance Info

Connect to your instance i-04466cf2ca2d2e938 (window-server) using any of these options

Session Manager RDP client EC2 serial console

Instance ID
 i-04466cf2ca2d2e938 (window-server)

Connection Type

Connect using RDP client
Download a file to use with your RDP client and retrieve your password.

Connect using Fleet Manager
To connect to the instance using Fleet Manager Remote Desktop, the SSM Agent must be installed and running on the instance. For more information, see [Working with SSM Agent](#)

You can connect to your Windows instance using a remote desktop client of your choice, and by downloading and running the RDP shortcut file below:

[Download remote desktop file](#)

When prompted, connect to your instance using the following username and password:

Public IP
 50.16.66.248

Username Info
 Administrator

Password [Get password](#)

(i) If you've joined your instance to a directory, you can use your directory credentials to connect to your instance.

[Cancel](#)

Get Windows password Info

Use your private key to retrieve and decrypt the initial Windows administrator password for this instance.

Instance ID

i-04466cf2ca2d2e938 (window-server)

Key pair associated with this instance

key-pair

Private key

Either upload your private key file or copy and paste its contents into the field below.

[Upload private key file](#)

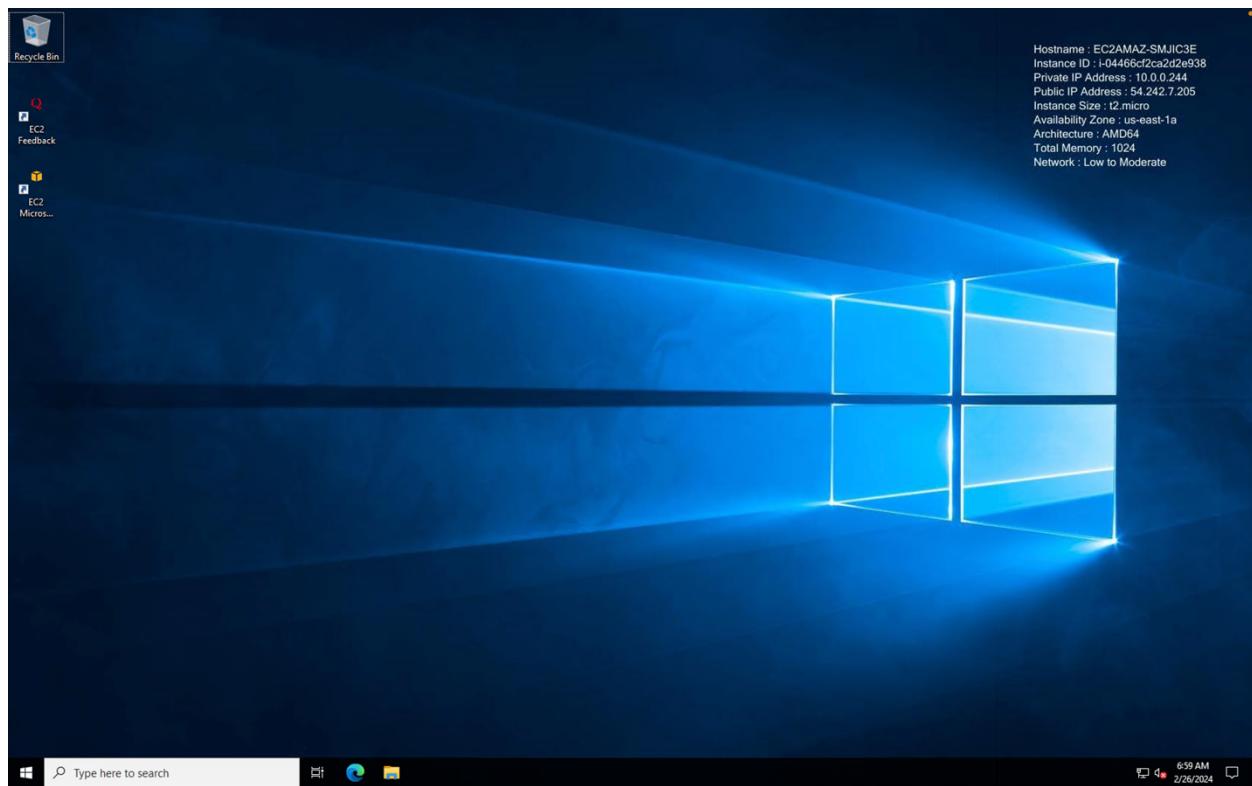
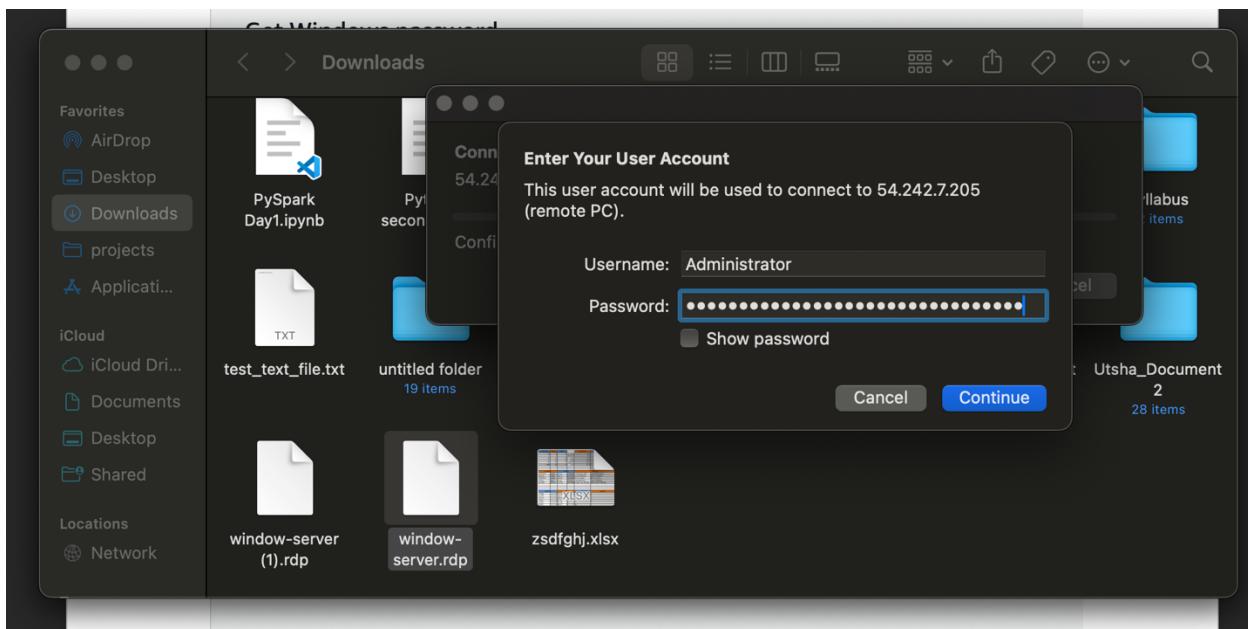
key-pair.pem
1.674KB

Private key contents - optional

```
-----BEGIN RSA PRIVATE KEY-----  
MIIEowIBAAKCAQEAK/PUwsCv/PyRlpctO9mJkOuQd46UB/sn7ObfHk+lf1AsqXq  
tRLywktrIlsqjPBZYePqXwYRGG6b6uEckxYXmdoUQsj9/X4+47kk2wDolpbDwlqu  
NyCTKTmw3upG/wDGu2tXEmR+xJb&JcBmlhgluZFA3jOfzPObkLqXoMDgquAbe9N  
bBvB9vQr10ugAOx2BzhwdHTtfVgf57vM/9MCulcozITafNP2ASZsZzRw+3/Li  
99qLy5cH9lmgfLoayuno0M1J5TwMESAbS4sB1xJvt8K2soZApC6CsriUnjlWG8Z8  
S2156MIQ6LNpr9g/HV6w74efVlwAa5oOacf/QIDAQABAoIBAFJ49Pyq1G89itmP  
bWYenN5vk897xR2s7gYHSEv9Mc8oAoyTGEZfjzro14k+iblawThkB914UF/JGHJR
```

[Cancel](#)

[Decrypt password](#)



1.9. Hosting static website in Linux EC2 instance

Uploading the static file in linux .

```
utsha_mac@Utshas-MacBook-Pro Downloads % ssh -i "key-pair.pem" ec2-user@107.23.233.69
The authenticity of host '107.23.233.69 (107.23.233.69)' can't be established.
ED25519 key fingerprint is SHA256:f+xq7EU6Lg3dNa3IpYlx02b9PNHVAK4DTZ9+sDWGmxg.
This key is not known by any other names.
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added '107.23.233.69' (ED25519) to the list of known hosts.

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

Last login: Fri Mar  1 06:27:42 2024 from 18.206.107.29
[ec2-user@ip-10-0-0-19 ~]$ mkdir temp
[ec2-user@ip-10-0-0-19 ~]$ ls
temp
```

Uploading the website files in ec2-instance temp directory

```
utsha_mac@Utshas-MacBook-Pro downloads % scp -i "key-pair.pem" /Users/utsha_mac/projects/techkraft/Bootcamp-Tasks/aws/vpc_revisio
n_lab/static_website/* ec2-user@107.23.233.69:temp/
portfolio.html                                                 100% 1234      4.1KB/s  00:00
utsha-img.jpg                                                 100% 236KB 128.0KB/s  00:01
```

```
[ec2-user@ip-10-0-0-19 temp]$ ls
portfolio.html  utsha-img.jpg
```

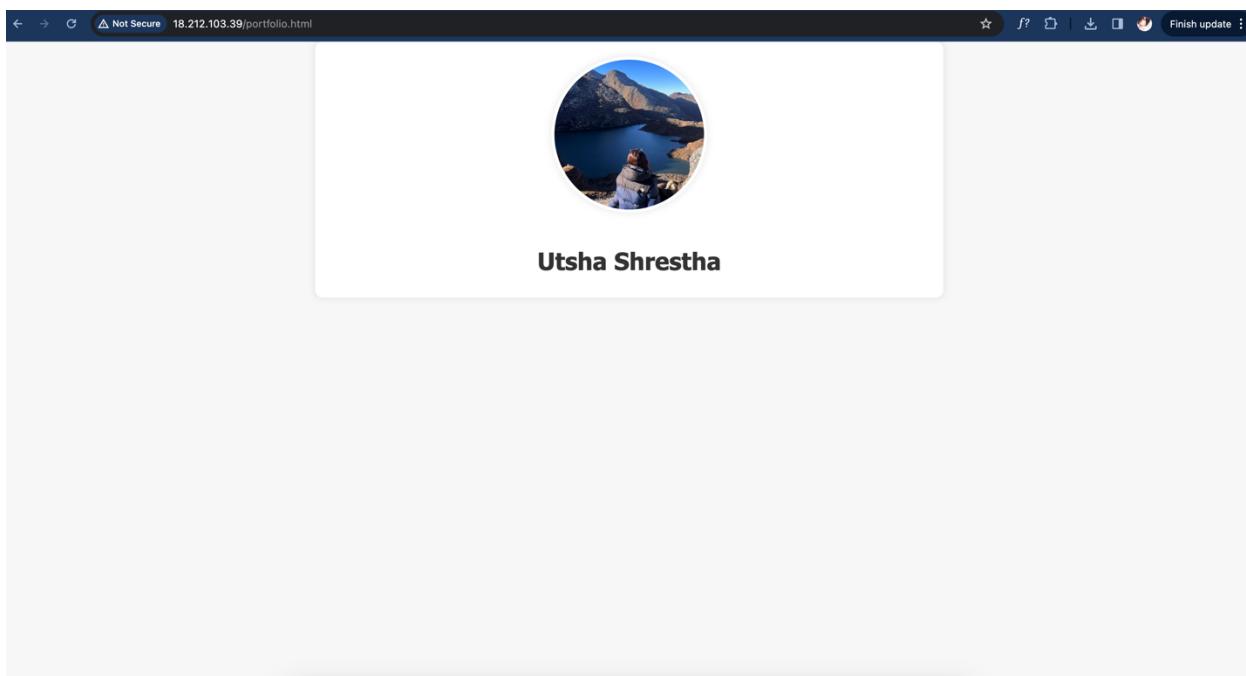
Moving the website html content to html directory and starting the server.

```
utsha_mac@Utshas-MacBook-Pro downloads % ssh -i "key-pair.pem" ec2-user@107.23.233.69
      _#
 ~\_ #####_      Amazon Linux 2023
 ~~ \#####\
 ~~ \###|
 ~~ \#/ ___ https://aws.amazon.com/linux/amazon-linux-2023
 ~~ \~' '->
 ~~ /
 ~~_. _/
 /_/
 _/m'

Last login: Fri Mar  1 06:31:16 2024 from 110.44.121.23
[ec2-user@ip-10-0-0-19 ~]$ cd temp/
[ec2-user@ip-10-0-0-19 temp]$ ls
portfolio.html  utsha-img.jpg
[ec2-user@ip-10-0-0-19 temp]$ sudo mv * /var/www/html/
[ec2-user@ip-10-0-0-19 temp]$ cd /var/www/html/
[ec2-user@ip-10-0-0-19 html]$ ls
portfolio.html  utsha-img.jpg
[ec2-user@ip-10-0-0-19 html]$
```

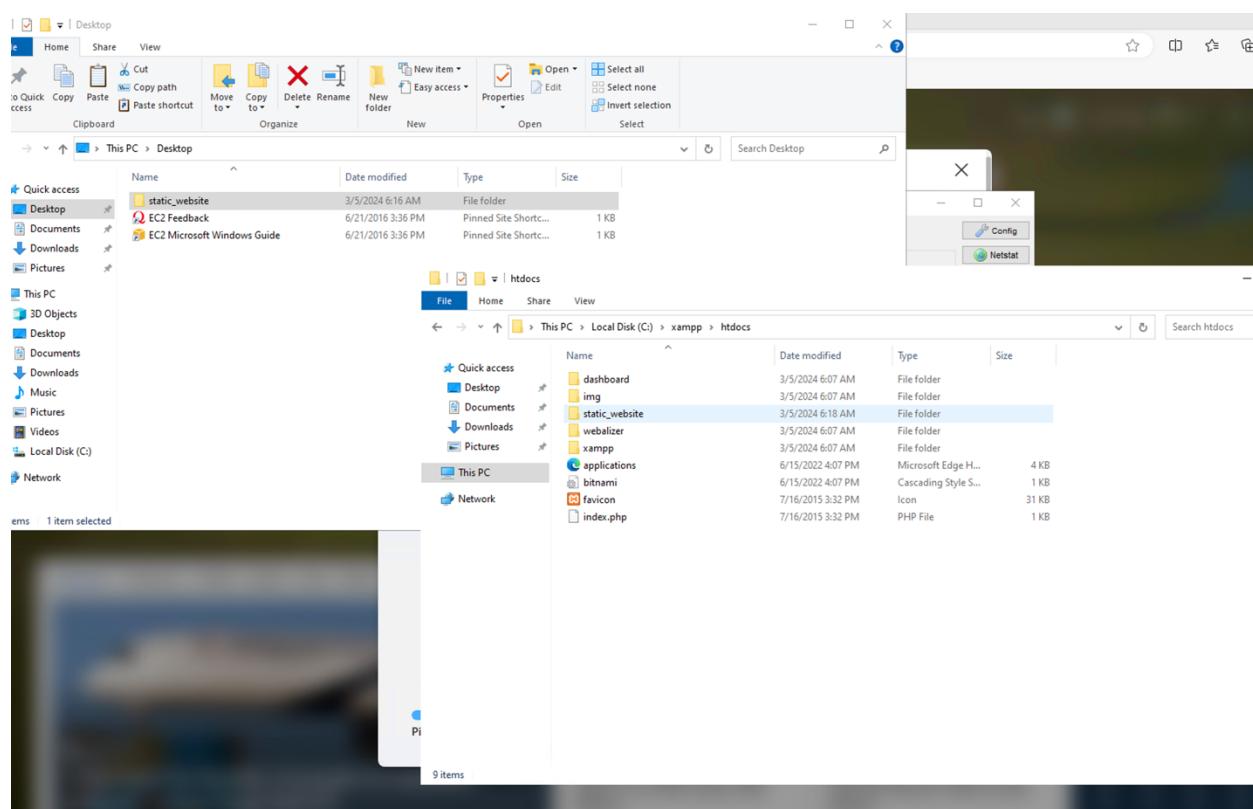
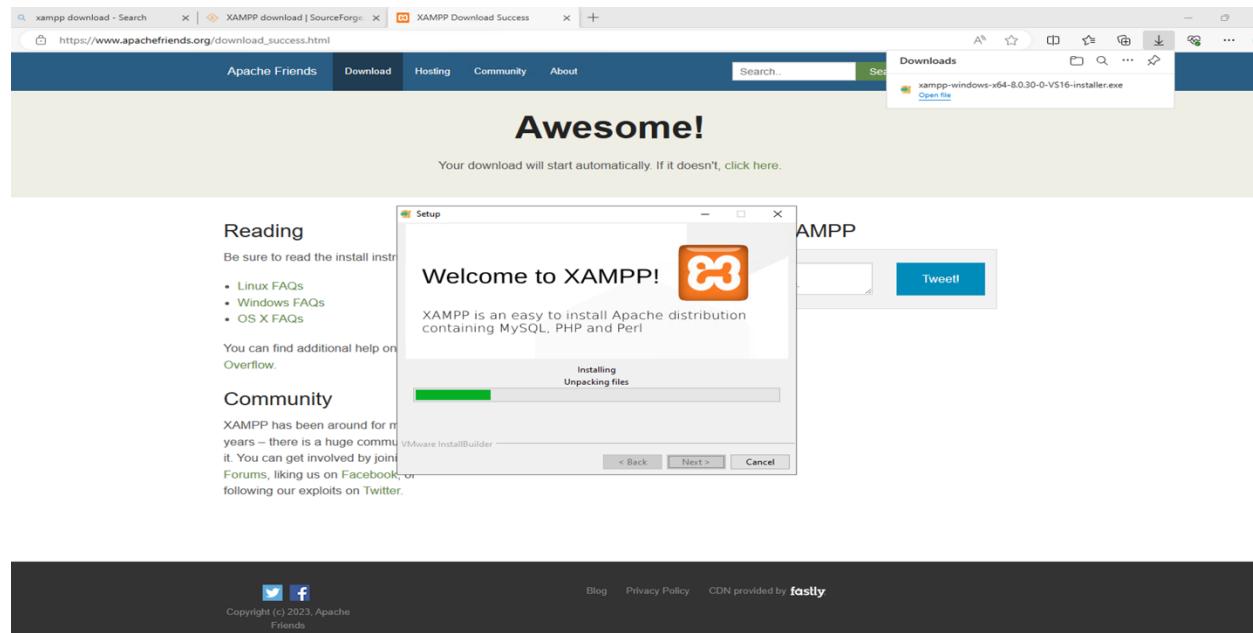
```
[ec2-user@ip-10-0-0-19 html]$ sudo service httpd start
Redirecting to /bin/systemctl start httpd.service
```

Then, the static website is hosted successfully, and it can be tested using public IP address.

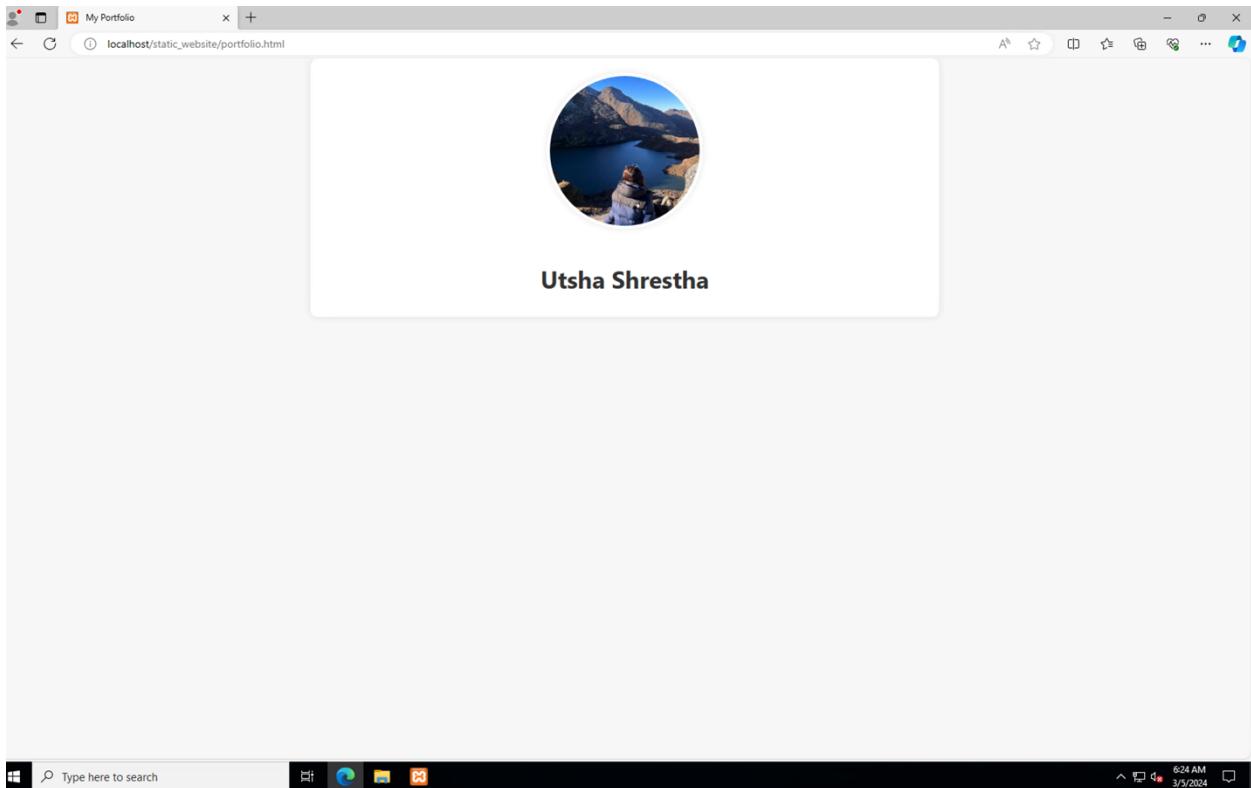


1.10. Hosting static website in Windows EC2 instance

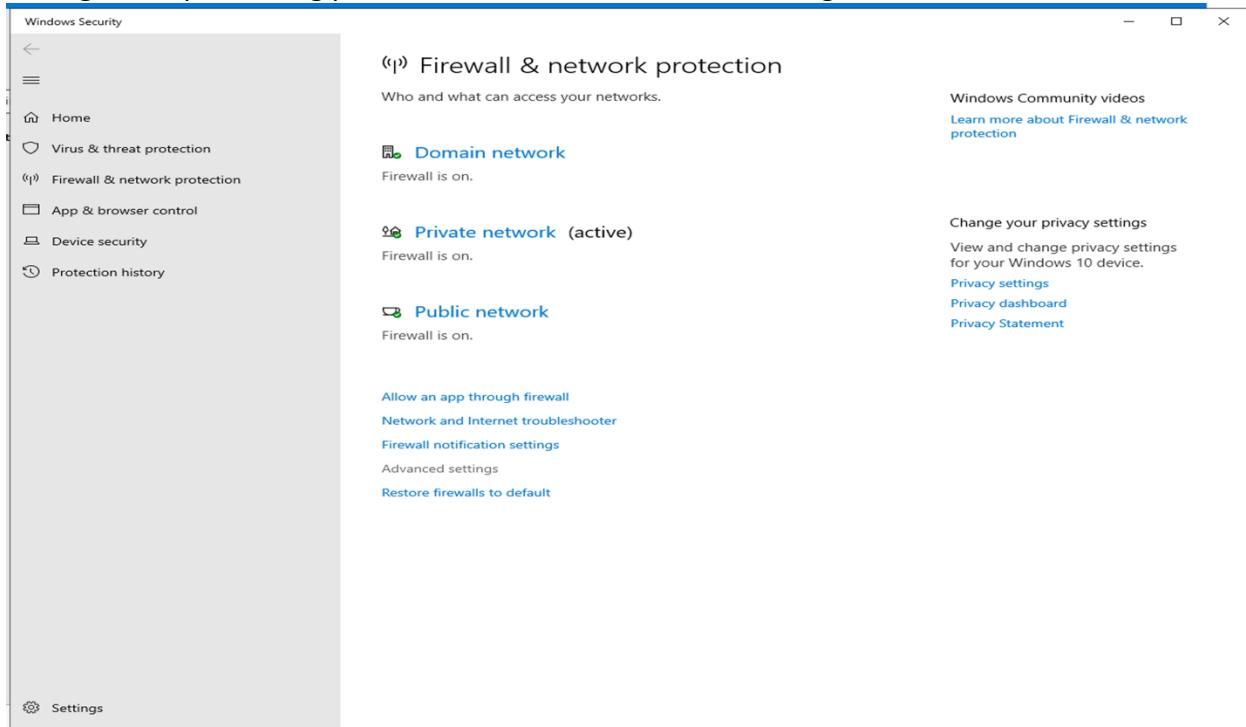
The downloaded rdp file of Windows EC2 instance is opened. There XAMPP is installed and the static website folder is moved to the htdocs directory of XAMPP.

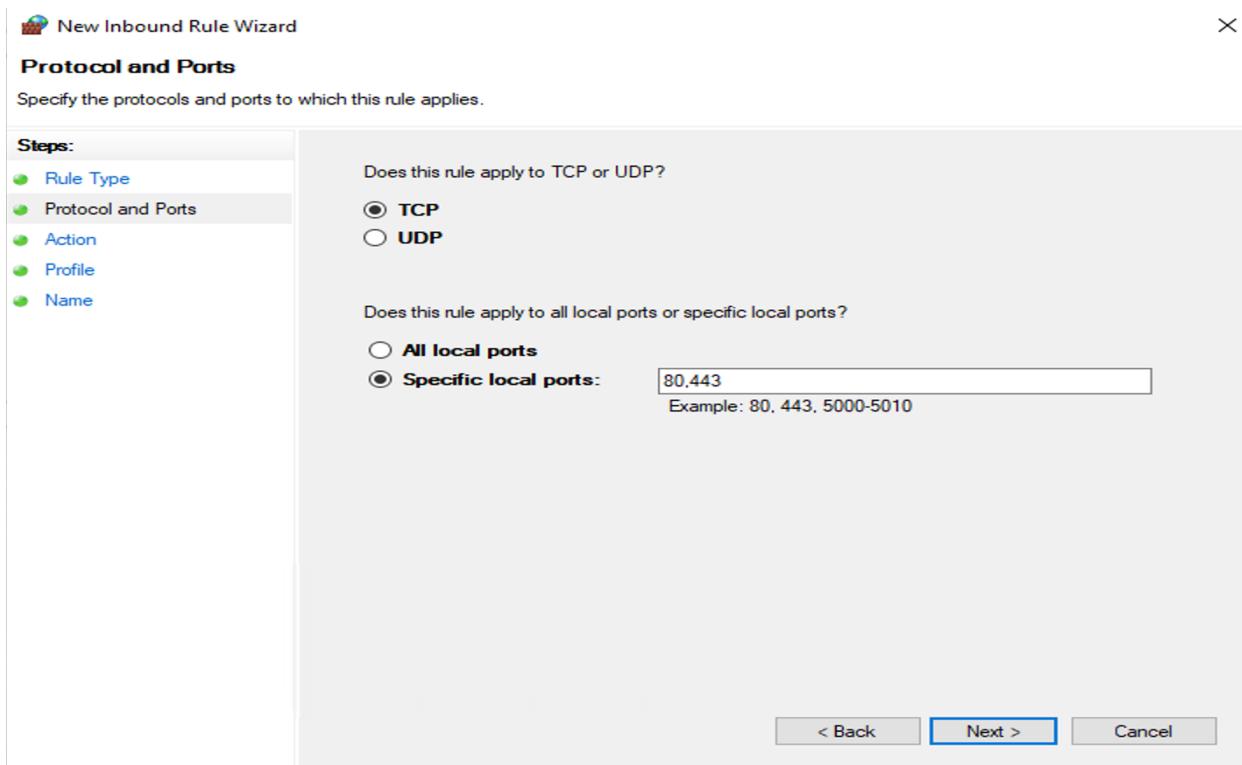
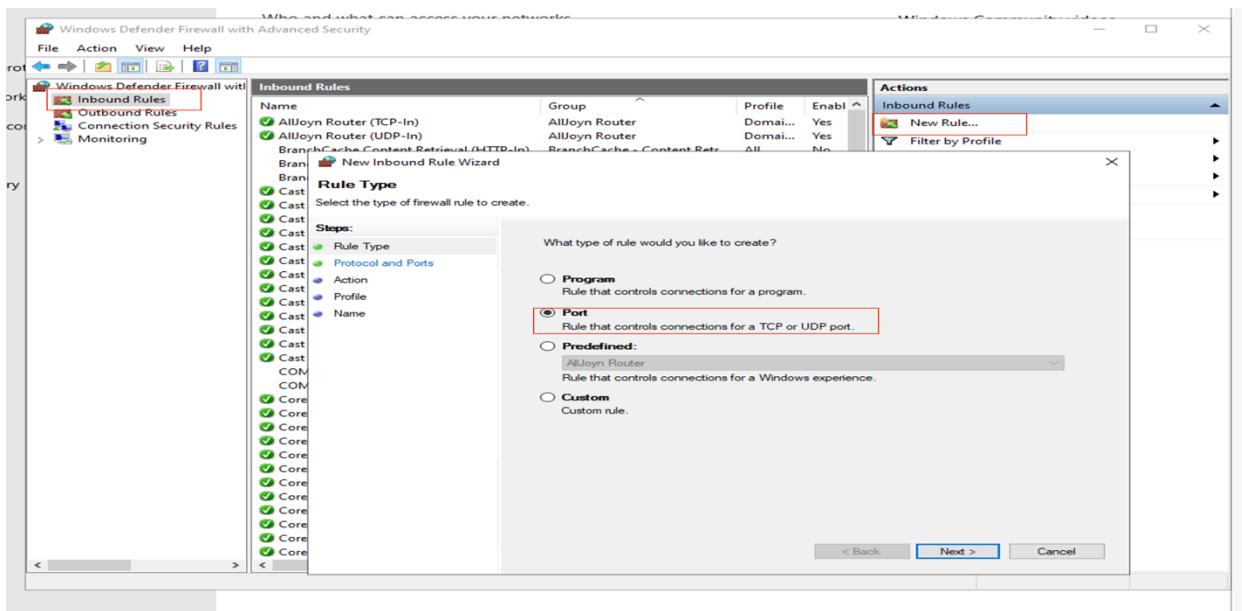


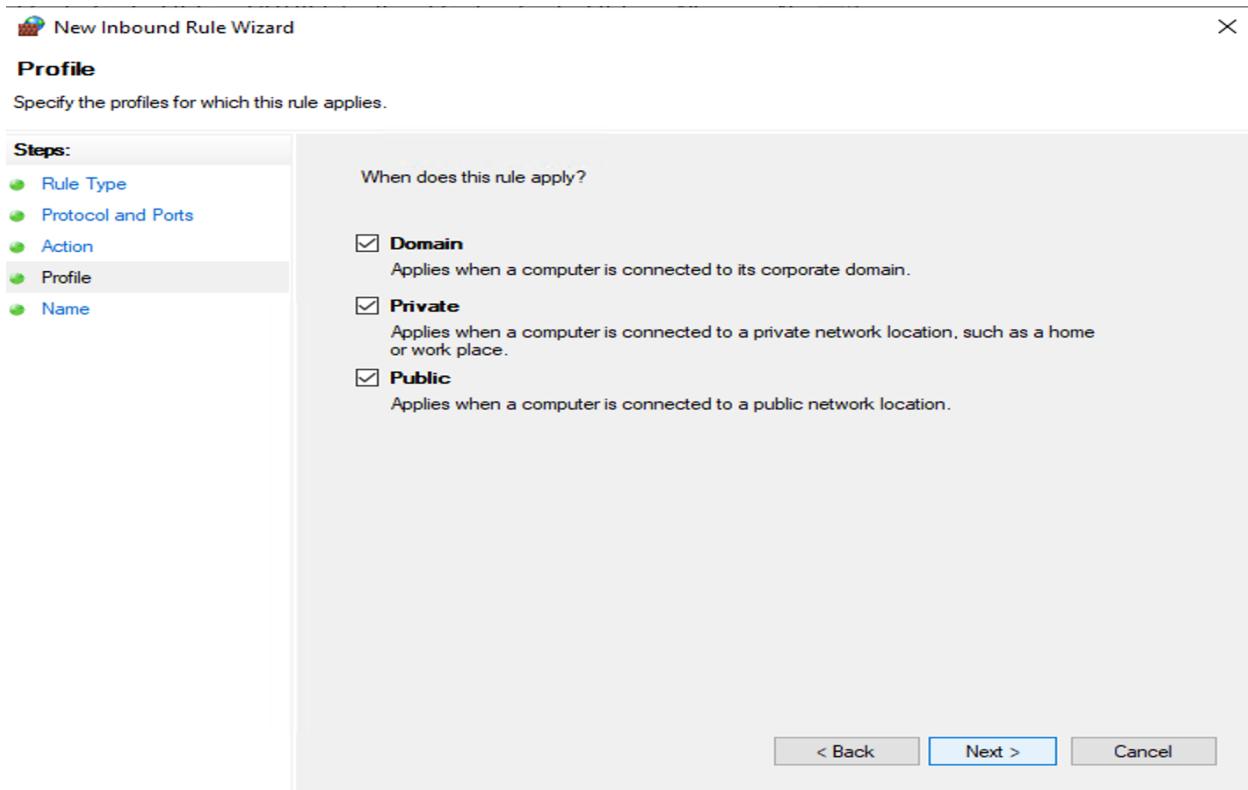
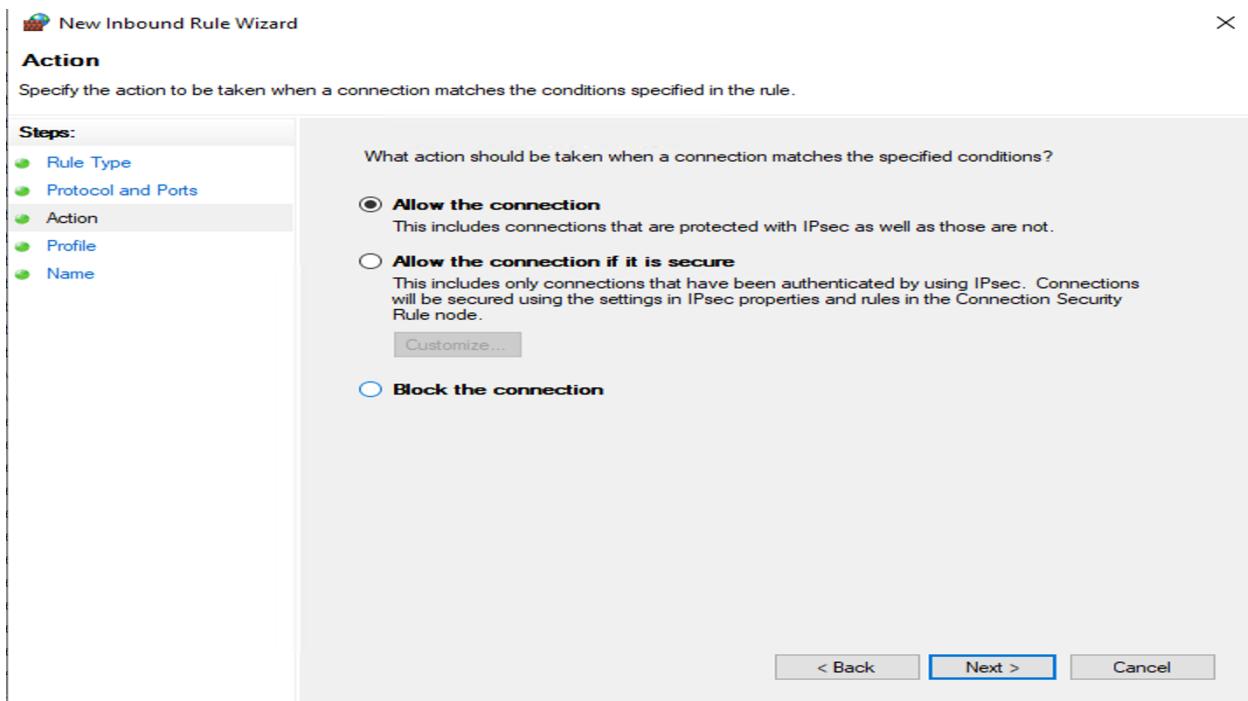
After that, the moved html content, working status is checked in windows remote desktop

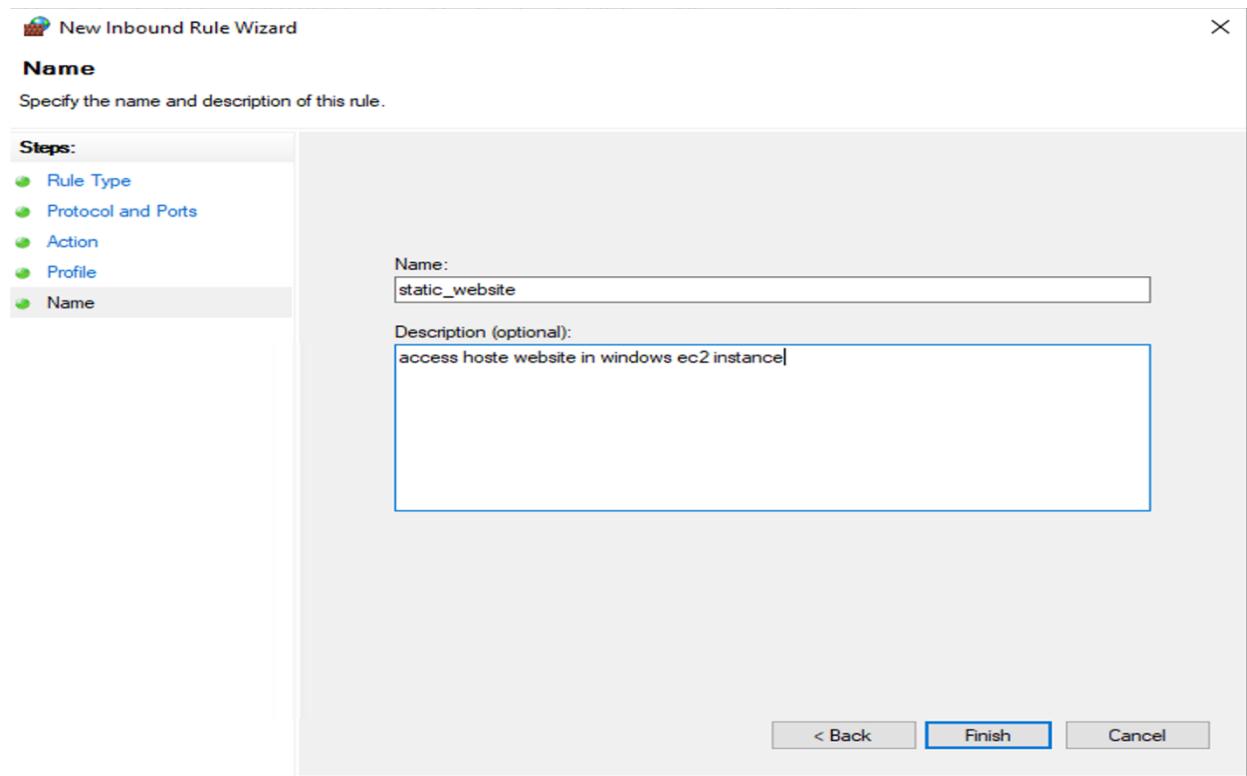


Now for checking, whether the static website is hosted in Windows EC2 instance, the firewall setting is setup, allowing port 80 and 443 from advanced setting.









After configuration of firewall setting in remote desktop. The static website is checked is checked locally using the public IP of windows EC2 instance.

