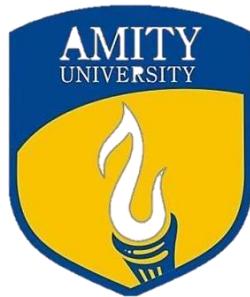


CLOUD COMPUTING ARCHITECT

Lab File

Submitted to

AMITY UNIVERSITY UTTAR PRADESH



In partial fulfilment of the requirements for the award of the degree of
Bachelor of Technology
In
Computer Science & Engineering By

Zaeem Uddin Khan
(A2305221555)

Submitted to:
Mr. Kunal Gupta

DEPARTMENT OF AMITY SCHOOL OF ENGINEERING AND TECHNOLOGY AMITY
UNIVERSITY UTTAR PRADESH
NOIDA (U.P.)

INDEX

S.No.	Name	Date	Remarks
1.			
2.			
3.			
4.			
5.			
6.			
7.			
8.			
9.			
10.			

Experiment-1

Aim:- To host a website through RDP client (windows) using AWS console.

Procedure:-

Step-1: Create a VPC

The screenshot shows the 'Create VPC' settings page. Under 'Resources to create', the 'VPC only' option is selected. A 'Name tag - optional' field contains 'abc'. Under 'IPv4 CIDR block', 'IPv4 CIDR manual input' is selected, and the CIDR block '10.0.0.0/16' is specified. A note states: 'CIDR block size must be between /16 and /28.'

Step-2: Create a Subnet

The screenshot shows the 'Create subnet' settings page. Under 'VPC ID', 'vpc-0a647f3490b516713 (abc)' is selected. Under 'Associated VPC CIDRs', '10.0.0.0/16' is listed. In the 'Subnet settings' section, 'Subnet 1 of 2' is selected. The 'Subnet name' is 'public'. Under 'Availability Zone', 'No preference' is selected. Under 'IPv4 VPC CIDR block', '10.0.0.0/16' is selected. Under 'IPv4 subnet CIDR block', '10.0.0.0/24' is selected, with '256 IPs' indicated.

Step-3: Create an Internet gateway and attach it to the VPC created above and then edit route tables

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.

abcig

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"/> X	<input type="text" value="abcig"/> X

Add new tag

You can add 49 more tags.

Cancel Create internet gateway

VPC > Internet gateways > Attach to VPC (igw-02b8f7ce8499f3abc)

Attach to VPC (igw-02b8f7ce8499f3abc) Info

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.

vpc-0a647f3490b516713 X

▶ AWS Command Line Interface command

Cancel Attach internet gateway

VPC > Route tables > rtb-01682d6b44dbf9b3f > 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
	<input type="text" value="igw-02b8f7ce8499f3abc"/> X		Remove

Add route

Cancel Save changes

Step-4: Go to EC2 and create an instance

The screenshot shows the 'Launch an instance' wizard in the AWS EC2 console. It includes sections for:

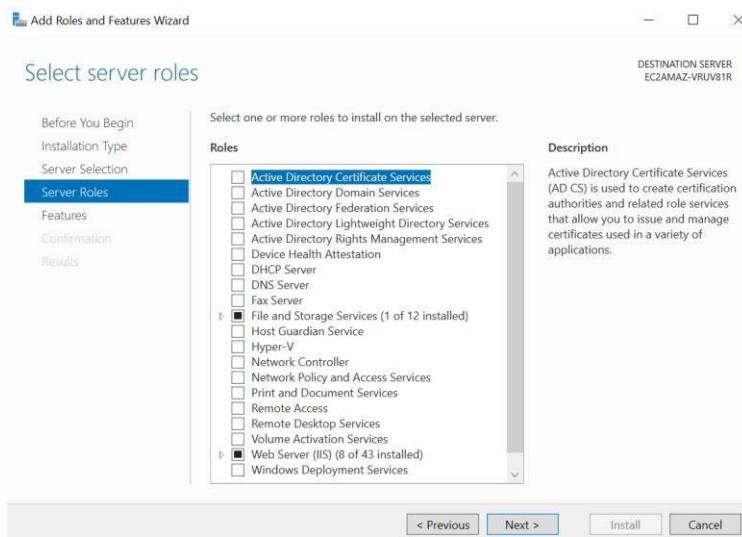
- Name and tags**: A 'Name' field containing 'web' and a link to 'Add additional tags'.
- Key pair (login)**: A 'Key pair name - required' dropdown set to 'aa', a 'Create new key pair' button, and a note about using a key pair for Windows instances.
- Network settings**: A 'VPC - required' dropdown set to 'vpc-0a647f3490b516713 (abc)', a 'Subnet' dropdown set to 'subnet-091df7b18b99f6681', and an 'Auto-assign public IP' dropdown set to 'Enable'. A note at the bottom states 'Additional charges apply when outside of free tier allowance'.

Step-5: Open PuTTY and enter the public IP in it

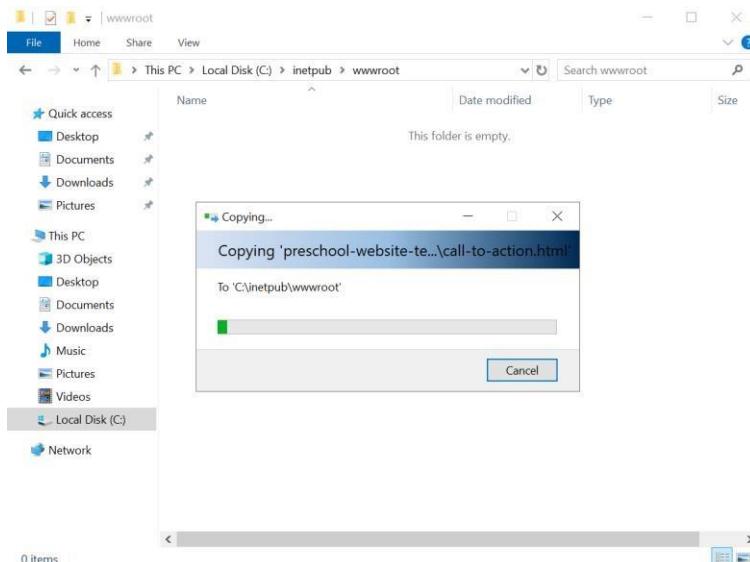
The screenshot shows the 'Connect to instance' page for instance **i-0777032c0c1596a5b (web)**. It includes:

- A 'Session Manager' tab selected, showing the instance ID **i-0777032c0c1596a5b (web)**.
- A 'Connection Type' section with two options:
 - Connect using RDP client** (selected): '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.'
- A note: '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:' followed by a 'Download remote desktop file' button.
- A note at the bottom: 'When prompted, connect to your instance using the following username and password:'.

Step-6: Add Web Server (IIS) feature in Server Manager in the RDP client



Step-7: Goto C: drive, then inetpub, then wwwroot and then delete all the files and paste a website's source code



Step-8: Open the public IP of the instance and view the website



Output:- Successfully hosted a website through RDP client (windows) using AWS console.

Experiment-2

Aim:- To host a website through RDP client (linux) using AWS console.

Procedure:-

Step-1: Create a VPC

The screenshot shows the 'Create VPC' settings page. At the top, it says 'A VPC is an isolated portion of the AWS Cloud populated by AWS objects, such as Amazon EC2 instances.' Below this is the 'VPC settings' section. It includes fields for 'Resources to create' (set to 'VPC only'), 'Name tag - optional' (containing 'abc'), 'IPv4 CIDR block' (set to '10.0.0.16'), and a note that the CIDR block size must be between /16 and /28.

Step-2: Create a Subnet

The screenshot shows the 'Create subnet' settings page. It starts with a 'VPC' section where the 'VPC ID' is selected as 'vpc-0a647f3490b516713 (abc)'. Below this is the 'Associated VPC CIDRs' section, which lists '10.0.0.0/16'. The main part of the page is the 'Subnet settings' section, which contains the 'Subnet 1 of 2' configuration. It includes fields for 'Subnet name' (set to 'public'), 'Availability Zone' (set to 'No preference'), and 'IPv4 VPC CIDR block' (set to '10.0.0.16'). The 'IPv4 subnet CIDR block' field is also visible at the bottom.

Step-3: Create an Internet gateway and attach it to the VPC created above and then edit route tables

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.

abcig

Tags - optional

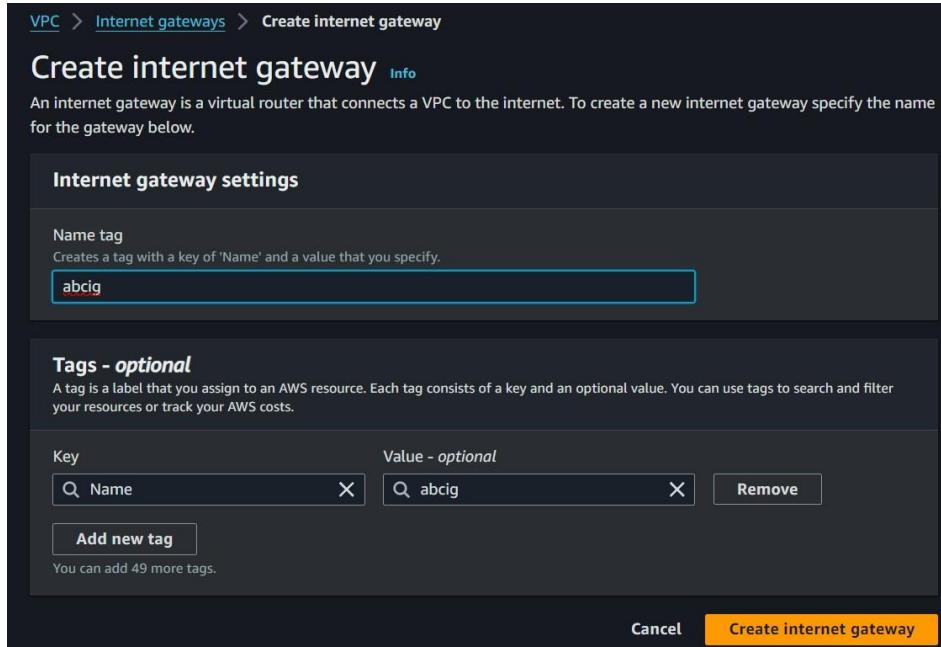
A tag is a label that you assign to an AWS resource. Each tag consists of a key and an optional value. You can use tags to search and filter your resources or track your AWS costs.

Key	Value - optional
Q Name	Q abcig

Add new tag

You can add 49 more tags.

Cancel **Create internet gateway**



VPC > Internet gateways > Attach to VPC (igw-02b8f7ce8499f3abc)

Attach to VPC (igw-02b8f7ce8499f3abc) Info

VPC

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

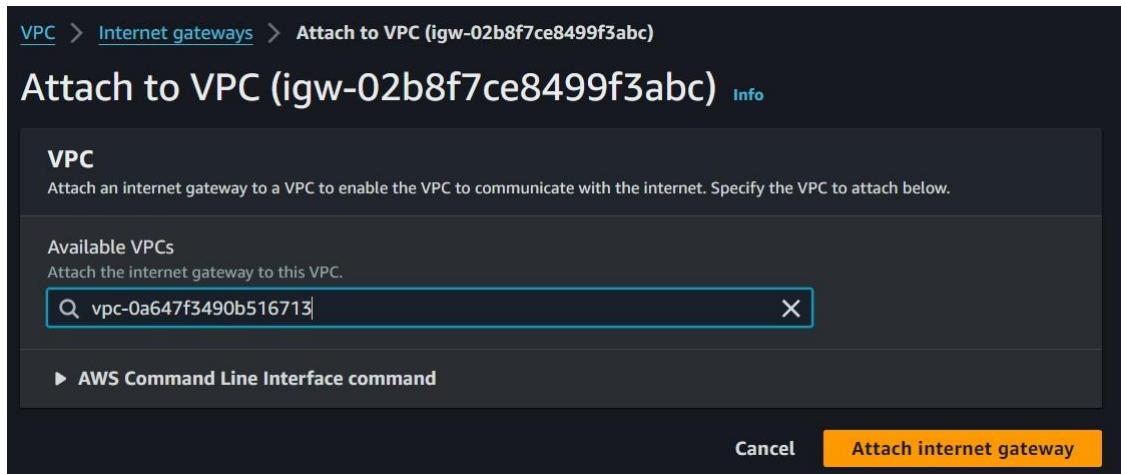
Available VPCs

Attach the internet gateway to this VPC.

Q vpc-0a647f3490b516713 X

▶ AWS Command Line Interface command

Cancel **Attach internet gateway**



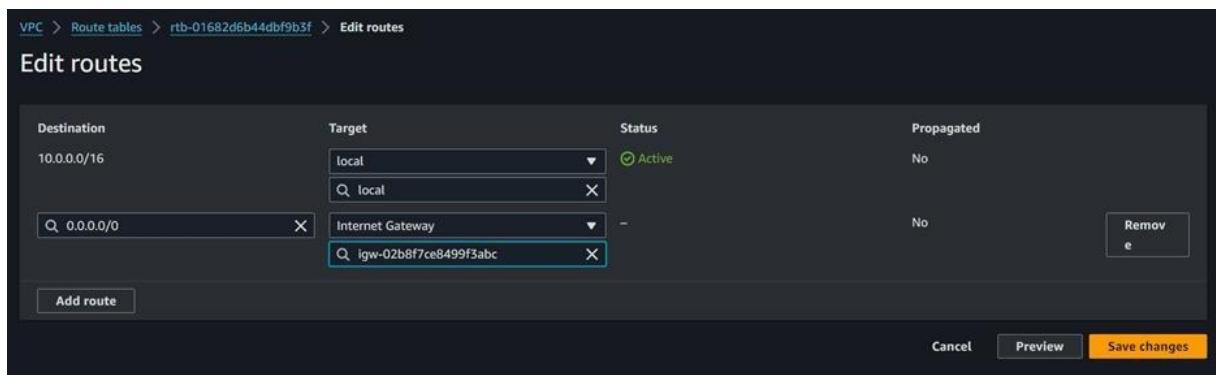
VPC > Route tables > rtb-01682d6b44dbf9b3f > Edit routes

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-02b8f7ce8499f3abc		Remove

Add route

Cancel **Preview** **Save changes**

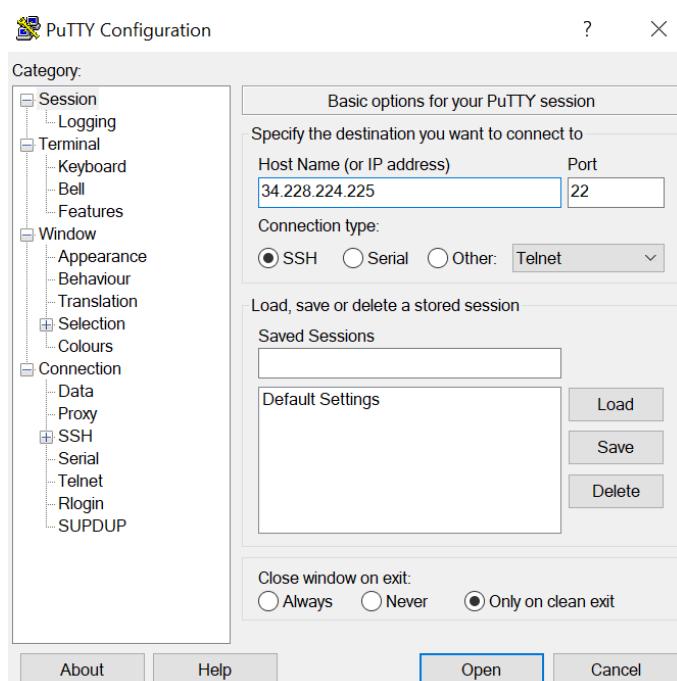


Step-4: Go to EC2 and create an instance

The screenshot shows the 'Launch an instance' wizard in the AWS Management Console. It consists of three main sections:

- Name and tags**: A 'Name' field contains 'web'. An 'Add additional tags' button is present.
- Key pair (login)**: A 'Key pair name - required' dropdown is set to 'aa'. A 'Create new key pair' button is available. A note states: "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."
- Network settings**:
 - VPC - required**: A dropdown shows 'vpc-0a647f3490b516713 (abc)' and '10.0.0.0/16'.
 - Subnet**: A dropdown shows 'subnet-091df7b18b99f6681' and details: 'VPC: vpc-0a647f3490b516713 Owner: 654654312679 Availability Zone: us-east-1a IP addresses available: 251 CIDR: 10.0.0.0/24'.
 - Auto-assign public IP**: A dropdown is set to 'Enable'.A note at the bottom states: "Additional charges apply when outside of free tier allowance".

Step-5: Open PuTTY and enter the public IP in it



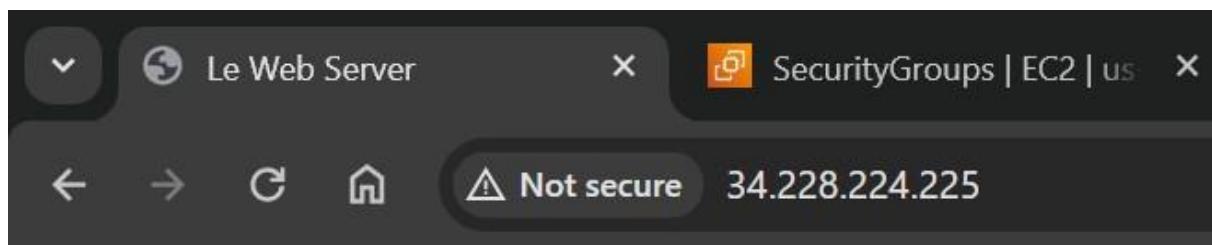
Step-6: Enter the commands in the terminal and install httpd to host the website

```
root@ip-10-0-0-169:~ Verifying : httpd-filesystem-2.4.58-1.amzn2023.noarch 12/12 ^

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-httpd-18.0.0-12.amzn2023.0.3.noarch
  httpd-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
  libbrotli-1.0.9-4.amzn2023.0.2.x86_64
  mailcap-2.1.49-3.amzn2023.0.3.noarch
  mod_http2-2.0.11-2.amzn2023.x86_64
  mod_lua-2.4.58-1.amzn2023.x86_64

Complete!
[root@ip-10-0-0-169 ~]# sudo systemctl start httpd
[root@ip-10-0-0-169 ~]# sudo systemctl enable httpd
Created symlink /etc/systemd/system/multi-user.target.wants/httpd.service → /usr/lib/systemd/system/httpd.service.
[root@ip-10-0-0-169 ~]# sudo chmod 777 /var/www/html
[root@ip-10-0-0-169 ~]# sudo vi /var/www/html/index.html
[root@ip-10-0-0-169 ~]# █
```

Step-7: Open the public IP and view the website



This is Lab Experiment-2

Output:- Successfully hosted a website through RDP client (linux) using AWS console.

Experiment-3

Aim:- To create an NAT Gateway using AWS console.

Procedure:-

Step-1: Create VPC

The screenshot shows the 'Create VPC' settings page. Under 'Resources to create', 'VPC only' is selected. In the 'Name tag - optional' section, the tag 'abc' is entered. Under 'IPv4 CIDR block', 'IPv4 CIDR manual input' is selected, and the CIDR block '10.0.0.0/16' is specified. A note at the bottom states: 'CIDR block size must be between /16 and /28.'

Step-2: Create 2 Subnets and a NAT Gateway

The screenshot shows the 'Create subnet' interface. On the left, 'Subnet 1 of 2' is being configured with a name 'public', no preference for availability zone, and CIDRs '10.0.0.0/16'. On the right, 'Subnet 2 of 2' is being configured with a name 'private', no preference for availability zone, and CIDR '10.0.0.0/16'. Both panels include sections for 'Associated VPC CIDRs' and 'IPv4 subnet CIDR block'.

VPC > [NAT gateways](#) > [Create NAT gateway](#)

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.

The name can be up to 256 characters long.

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.

[Allocate Elastic IP](#)

Step-3: Create Internet gateway and attach it to the VPC created above and then edit route tables

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="abcig"/>

[Add new tag](#)

You can add 49 more tags.

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

VPC > [Internet gateways](#) > [Attach to VPC \(igw-02b8f7ce8499f3abc\)](#)

Attach to VPC (igw-02b8f7ce8499f3abc) Info

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](#)

Edit routes				
Destination	Target	Status	Propagated	
10.0.0.0/16	local	Active	No	
<input type="text"/> 0.0.0.0/0	<input type="text"/> Internet Gateway	Active	No	<button>Remove</button>
<input type="text"/> 0.0.0.0/0	<input type="text"/> NAT Gateway	-	No	<button>Remove</button>
Add route				
				Cancel Preview Save changes

Step-4: Go to EC2 and create 2 instances

EC2 > Instances > Launch an instance

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
 public [Add additional tags](#)

EC2 > Instances > Launch an instance

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
 private [Add additional tags](#)

▼ **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*
 aa [Create new key pair](#)

For Windows instances, you use a key pair to decrypt the administrator password. You then use the decrypted password to connect to your instance.

▼ **Network settings Info**

VPC - *required* [Info](#)
 vpc-0a647f3490b516713 (abc)
10.0.0.0/16 [Create new subnet](#)

Subnet [Info](#)
 subnet-091df7b18b99f6681 sub1
VPC: vpc-0a647f3490b516713 Owner: 654654312679 Availability Zone: us-east-1a IP addresses available: 251 CIDR: 10.0.0.0/24

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

Additional charges apply when outside of free tier allowance

Step-5: Download RDP client for public instance and open it

EC2 > Instances > i-024facdfb7af5ba25 > Connect to instance

Connect to instance info

Connect to your instance i-024facdfb7af5ba25 (public) using any of these options

Session Manager | **RDP client** | EC2 serial console

Instance ID
i-024facdfb7af5ba25 (public)

Connection Type

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

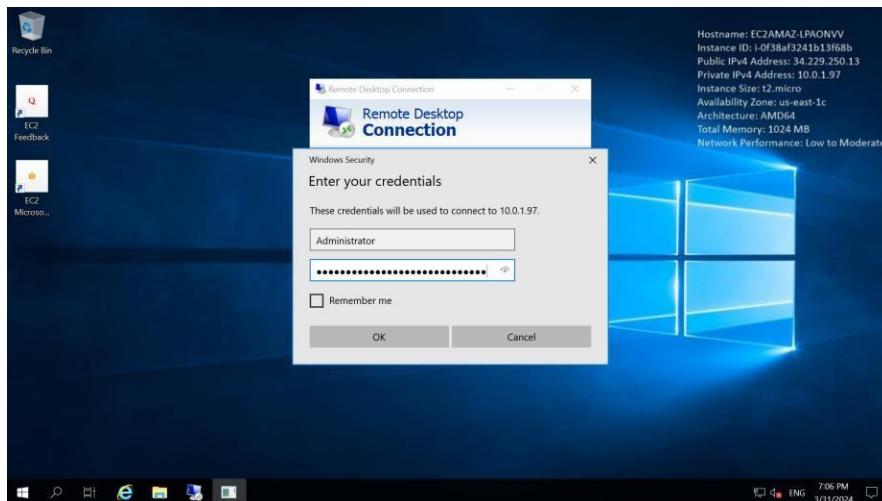
Connect using Fleet Manager
To connect to the instance using Fleet Manager Remote Desktop, the SSM Agent must be installed and running on the instance. For more information, see [Working with SSM Agent](#)

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:

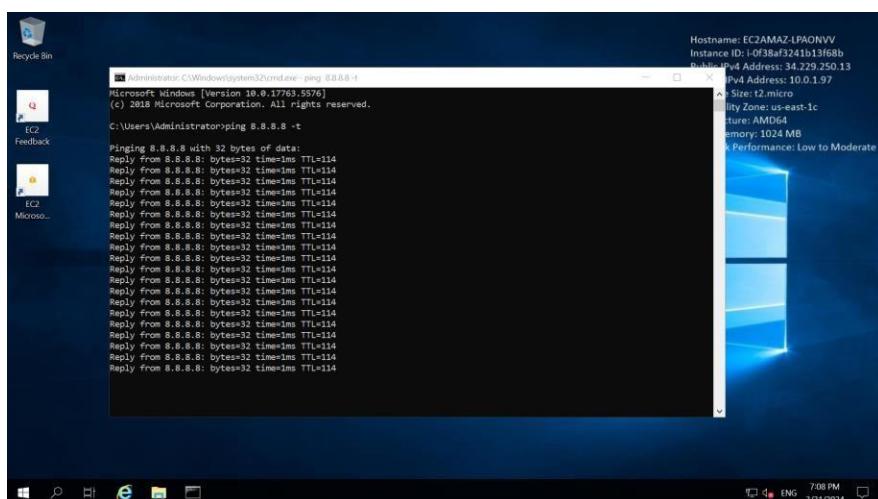
[Download remote desktop file](#)

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

Step-6: Open RDP client in the opened RDP client and enter private IP of private instance



Step-7: Open cmd and ping “8.8.8.8 – t”



Output:- Successfully created an NAT Gateway using AWS console.

Experiment-4

Aim:- To create a VPC Peering using AWS console.

Procedure:-

Step-1: Create 2 VPCs

The image shows two screenshots of the AWS VPC 'Create VPC' wizard. Both screenshots are nearly identical, showing the 'VPC settings' step. In both, the 'Resources to create' section has 'VPC only' selected. There is an optional 'Name tag - optional' field with values 'abc' and 'xyz'. Under 'IPv4 CIDR block', 'IPv4 CIDR manual input' is selected, with values '10.0.0.0/16' and '172.16.0.0/16' respectively. A note at the bottom of the right screenshot states: 'A VPC is an isolated portion of the AWS Cloud populated by AWS objects, such as Amazon EC2 instances. IPAM-allocated CIDR blocks are not supported for VPCs created via the VPC Wizard.'

Step-2: Create 2 Subnets

The image shows two screenshots of the AWS Subnet creation wizard. The left screenshot shows the 'Create subnet' step. It has a 'VPC' section with 'VPC ID' set to 'vpc-0a647f3490b516713 (abc)'. Below it is an 'Associated VPC CIDRs' section with 'IPv4 CIDRs' set to '10.0.0.0/16'. The right screenshot shows the 'Subnet 2 of 2' step. It has a 'Subnet settings' section with 'Subnet name' set to 'private'. Below it is an 'Availability Zone' section with 'No preference'. The 'IPv4 VPC CIDR block' section shows '10.0.0.0/16'.

Step-3: Create 2 Internet gateways and attach it to the VPC created above and then edit route tables

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.

abcig

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"/> X	<input type="text" value="abcig"/> X

Add new tag

You can add 49 more tags.

Cancel Create internet gateway

VPC > Internet gateways > Attach to VPC (igw-02b8f7ce8499f3abc)

Attach to VPC (igw-02b8f7ce8499f3abc) Info

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.

vpc-0a647f3490b516713 X

▶ AWS Command Line Interface command

Cancel Attach internet gateway

VPC > Route tables > rtb-01682d6b44dbf9b3f > 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
	<input type="text" value="igw-02b8f7ce8499f3abc"/> X		Remove

Add route

Cancel Save changes

Step-4: Go to EC2 and create 2 instances

The screenshot shows the 'Launch an instance' wizard in the AWS Management Console. It displays two parallel steps for creating instances:

- Instance 1 (Left):** Name is set to "public".
 - Key pair (login):** Key pair name is "aa". A note says: "For Windows instances, you use a key pair to decrypt the administrator password. You then use the decrypted password to connect to your instance."
 - Network settings:** VPC is "vpc-0a647f3490b516713 (abc)". Subnet is "subnet-091df7b18b99f6681". Auto-assign public IP is set to "Enable".
- Instance 2 (Right):** Name is set to "private".
 - Key pair (login):** Key pair name is "aa". A note says: "For Windows instances, you use a key pair to decrypt the administrator password. You then use the decrypted password to connect to your instance."
 - Network settings:** VPC is "vpc-0a647f3490b516713 (abc)". Subnet is "subnet-091df7b18b99f6681". Auto-assign public IP is set to "Enable".

Step-5: Goto Peering Connections

The screenshot shows the "Peering connection settings" page in the AWS Management Console. It is used to set up a peering connection between two VPCs:

- Name - optional:** Tag name is "vpc".
- Select a local VPC to peer with:** VPC ID (Requester) is "vpc-0a647f3490b516713 (abc)".
- VPC CIDRs for vpc-0a647f3490b516713 (abc):**

CIDR	Status	Status reason
10.0.0.0/16	Associated	-
- Select another VPC to peer with:**
 - Account:** My account is selected.
 - Region:** This Region (us-east-1) is selected.
 - VPC ID (Acceptor):** VPC ID is "vpc-0cb0a889d51515ac0 (xyz)".
 - VPC CIDRs for vpc-0cb0a889d51515ac0 (xyz):**

CIDR	Status	Status reason
172.16.0.0/16	Associated	-

Step-6: Goto route table and edit xyz routes to add abc VPC through peering connection and then edit abc routes to add xyz VPC through peering connection

Destination	Target	Status	Propagated
172.16.0.0/16	local	Active	No
Q 0.0.0.0/0	Internet Gateway	Active	No
Q 0.0.0.0/0	Peering Connection	-	No
	Q pcx-0674eec2e9bae3f95	X	

Add route Cancel Preview Save changes

Destination	Target	Status	Propagated
10.0.0.0/16	local	Active	No
Q 0.0.0.0/24	Peering Connection	Active	No
Q 0.0.0.0/0	Internet Gateway	Active	No
	Q igw-02b8f7ce8499f3abc	X	

Add route Cancel Preview Save changes

Step-7: Download RDP client for abc and xyz instances

EC2 > Instances > i-024facdfb7af5ba25 > Connect to instance

Connect to instance Info

Connect to your instance i-024facdfb7af5ba25 (public) using any of these options

Session Manager | **RDP client** | EC2 serial console

Instance ID: **i-024facdfb7af5ba25 (public)**

Connection Type:

- Connect using RDP client** (selected): 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:

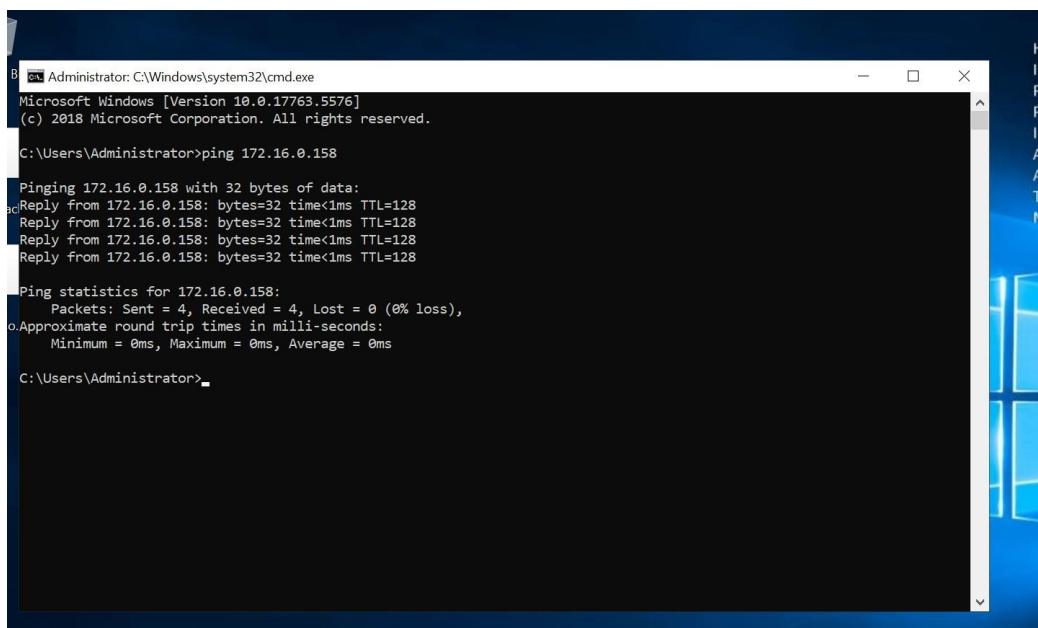
Download remote desktop file

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

Step-8: Open both RDP clients i.e. abc and xyz and remove firewall in them



Step-9: Open cmd in abc RDP client and ping public IP of xyz RDP client



Output:- Successfully created a VPC Peering using AWS console.

Experiment-5

Aim:- To create a VPC End-Point using AWS console.

Procedure:-

Step-1: Create a VPC

The screenshot shows the 'Create VPC' configuration page. Under 'VPC settings', the 'Resources to create' section has 'VPC only' selected. A 'Name tag - optional' field contains 'abc'. Under 'IPv4 CIDR', 'IPv4 CIDR manual input' is selected, and the CIDR block '10.0.0.0/16' is specified. A note at the bottom states 'CIDR block size must be between /16 and /28.'

Step-2: Create 2 Subnets

The screenshot shows the 'Create subnet' interface. On the left, 'Subnet 1 of 2' is being configured with a 'Subnet name' of 'public', 'Availability Zone' set to 'No preference', and 'IPv4 VPC CIDR block' set to '10.0.0.0/16'. On the right, 'Subnet 2 of 2' is being configured with a 'Subnet name' of 'private', 'Availability Zone' set to 'No preference', and 'IPv4 VPC CIDR block' set to '10.0.0.0/16'. Both panels include notes about tag creation and CIDR block requirements.

Step-3: Create an Internet gateway and attach it to the VPC created above and then edit route tables

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.

abcig

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"/> X	<input type="text" value="abcig"/> X

Add new tag

You can add 49 more tags.

Cancel Create internet gateway

VPC > Internet gateways > Attach to VPC (igw-02b8f7ce8499f3abc)

Attach to VPC (igw-02b8f7ce8499f3abc) Info

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.

vpc-0a647f3490b516713 X

▶ AWS Command Line Interface command

Cancel Attach internet gateway

VPC > Route tables > rtb-01682d6b44dbf9b3f > 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
	<input type="text" value="igw-02b8f7ce8499f3abc"/> X		Remove

Add route

Cancel Save changes

Step-4: Go to EC2 and create 2 instances

The screenshot shows the 'Launch an instance' wizard in the AWS EC2 console. It displays two parallel steps for creating instances:

- Instance 1 (Left):** Name is set to 'public'. Under 'Key pair (login)', the key pair name is 'aa'. Under 'Network settings', the VPC is 'vpc-0a647f3490b516713 (abc)' and the subnet is 'subnet-091df7b18b99f6681'. Auto-assign public IP is set to 'Enable'.
- Instance 2 (Right):** Name is set to 'private'. Under 'Key pair (login)', there is a link to 'Create new key pair'. Under 'Network settings', the VPC is 'vpc-0a647f3490b516713 (abc)' and the subnet is 'subnet-091df7b18b99f6681'. Auto-assign public IP is set to 'Enable'.

Step-5: Goto VPC Endpoint

The screenshot shows the 'Create endpoint' wizard in the AWS VPC console. It includes the following sections:

- Create endpoint**: Describes three types of endpoints: Interface endpoints, Gateway Load Balancer endpoints, and Gateway endpoints. Interface endpoints are typically used for private services.
- Endpoint settings**:
 - Name tag - optional**: A text input field containing 'vpccend'.
 - Service category**: A dropdown menu showing categories: 'AWS services', 'PrivateLink Ready partner services', 'EC2 Instance Connect Endpoint', and 'Other endpoint services'.

Services (1/2)

Search

Service Name = com.amazonaws.us-east-1.s3 X Clear filters

< 1 > |

Service Name	Owner	Type
com.amazonaws.us-east-1.s3	amazon	Gateway
com.amazonaws.us-east-1.s3	amazon	Interface

VPC

Select the VPC in which to create the endpoint

VPC

The VPC in which to create your endpoint.

vpc-0a647f5490b516713 (abc) C

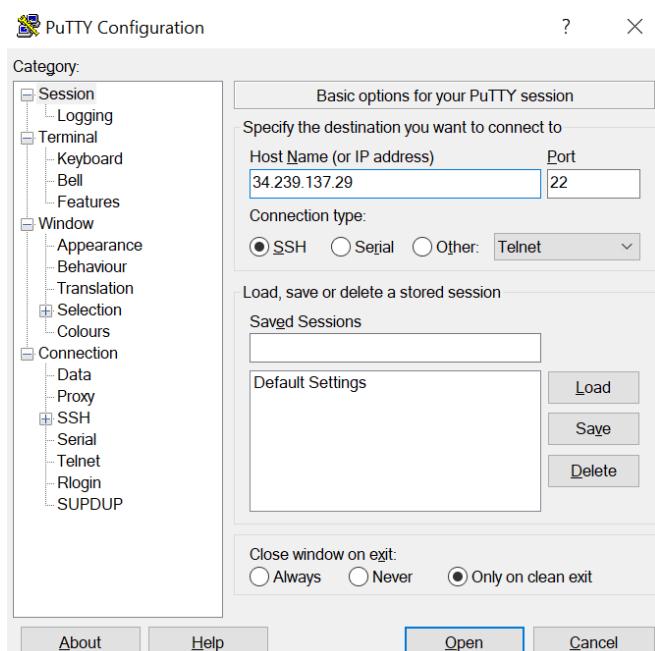
Route tables (1/1) Info

Search

< 1 > |

Name	Route Table ID	Main	Associated Id
-	rtb-01682d6b44dbf9b3f	Yes	2 subnets

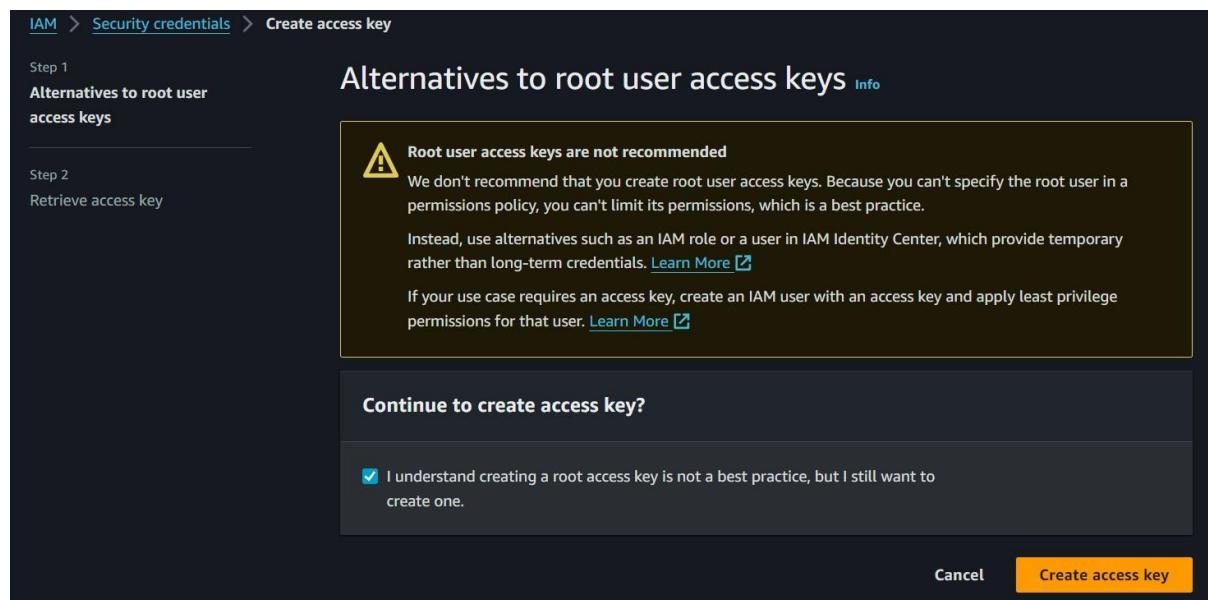
Step-6: Open PuTTY and enter public with the private key



```
ec2-user@login:~$ login as: ec2-user
[ec2-user@login:~]$ Authenticating with public key "imported-openssh-key"
[ec2-user@login:~]$ ,
[ec2-user@login:~]$ ~\_\_ #####
[ec2-user@login:~]$ ~~~ \_\_\#\#\#\_
[ec2-user@login:~]$ ~~~ \#\#\#|      Amazon Linux 2
[ec2-user@login:~]$ ~~~ \#\#/
[ec2-user@login:~]$ ~~~ V~'__->      AL2 End of Life is 2025-06-30.
[ec2-user@login:~]$ ~~~ /      A newer version of Amazon Linux is available!
[ec2-user@login:~]$ ~.._. /      Amazon Linux 2023, GA and supported until 2028-03-15.
[ec2-user@login:~]$ _/m' /      https://aws.amazon.com/linux/amazon-linux-2023/
[ec2-user@login:~]$
```

Step-7: Enter commands in the terminal after logging in it

Step-8: Goto AWS Security Credentials and create an access key



Step-9: Goto the terminal and configure AWS and paster the access key

```
[root@ip-10-0-0-30 ~]# aws configure
AWS Access Key ID [None]: AKIAZQ3DQCDTVQHYR47G
AWS Secret Access Key [None]: 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
Default region name [None]: us-east-1
Default output format [None]: json
[root@ip-10-0-0-30 ~]# aws s3 ls
```

Output:- Successfully created a VPC End-Point using AWS console.

Experiment-6

Aim:- To create a VPN using AWS console.

Procedure:-

Step-1: Create a VPC

The screenshot shows the 'Create VPC' settings page. At the top, it says 'A VPC is an isolated portion of the AWS Cloud populated by AWS objects, such as Amazon EC2 instances.' Below this is a section titled 'VPC settings' with the sub-section 'Resources to create'. It offers two options: 'VPC only' (selected) and 'VPC and more'. Under 'Name tag - optional', there is a text input field containing 'abc'. In the 'IPv4 CIDR block' section, 'IPv4 CIDR manual input' is selected, and the CIDR block '10.0.0.0/16' is specified. A note below states 'CIDR block size must be between /16 and /28.'

Step-2: Create a Subnet

The screenshot shows the 'Create subnet' settings page. It starts with a 'VPC' section where the 'VPC ID' is set to 'vpc-0a647f3490b516713 (abc)'. Below this is an 'Associated VPC CIDRs' section with the value '10.0.0.0/16'. The main part of the page is the 'Subnet settings' section, which includes a 'Subnet 1 of 2' configuration. It has fields for 'Subnet name' (set to 'public'), 'Availability Zone' (set to 'No preference'), and 'IPv4 VPC CIDR block' (set to '10.0.0.0/16'). There is also a 'IPv4 subnet CIDR block' dropdown showing '10.0.0.0/24'.

Step-3: Create an Internet gateway and attach it to the VPC created above and then edit route tables

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.

abcig

Tags - optional

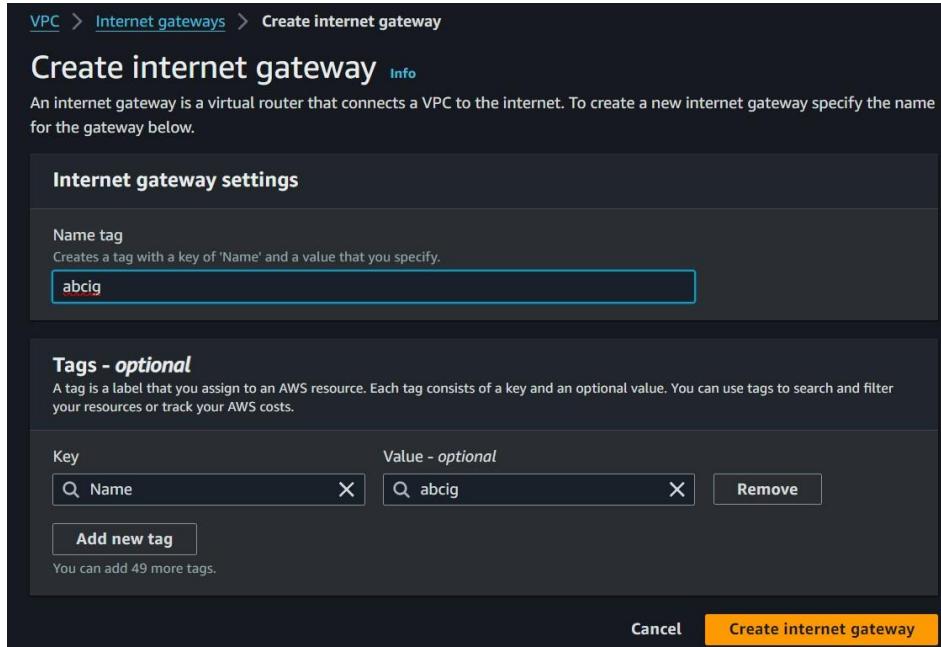
A tag is a label that you assign to an AWS resource. Each tag consists of a key and an optional value. You can use tags to search and filter your resources or track your AWS costs.

Key	Value - optional
Q Name	Q abcig

Add new tag

You can add 49 more tags.

Cancel **Create internet gateway**



VPC > Internet gateways > Attach to VPC (igw-02b8f7ce8499f3abc)

Attach to VPC (igw-02b8f7ce8499f3abc) Info

VPC

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

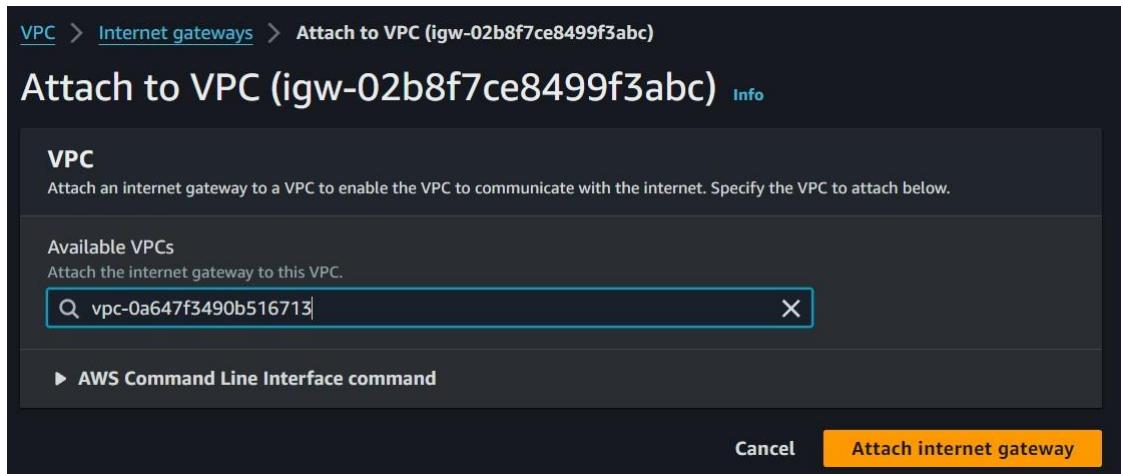
Available VPCs

Attach the internet gateway to this VPC.

Q vpc-0a647f3490b516713 X

▶ AWS Command Line Interface command

Cancel **Attach internet gateway**



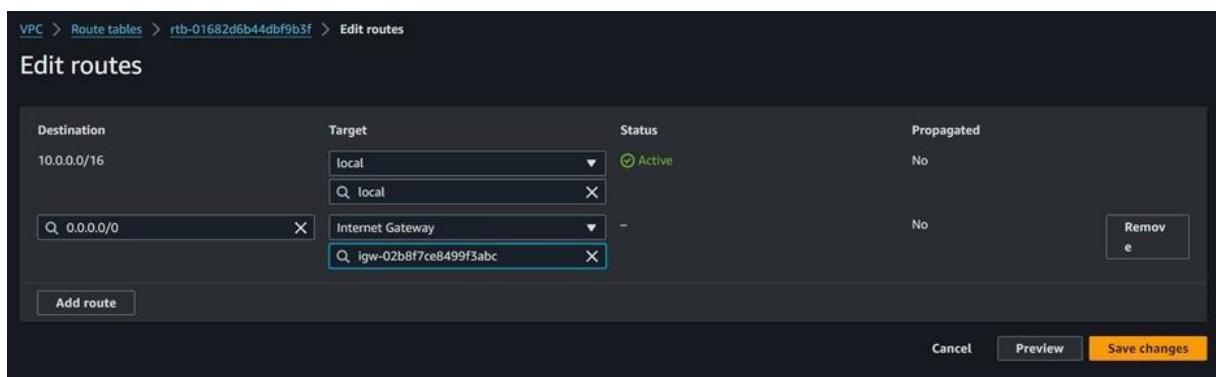
VPC > Route tables > rtb-01682d6b44dbf9b3f > Edit routes

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-02b8f7ce8499f3abc		Remove

Add route

Cancel **Preview** **Save changes**



Step-4: Go to EC2 and create an instance

The screenshot shows the 'Launch an instance' wizard in the AWS EC2 console. It consists of three main sections:

- Name and tags**: A 'Name' field contains 'server'. There is also a link to 'Add additional tags'.
- Key pair (login)**: A 'Key pair name - required' dropdown is set to 'aa'. A 'Create new key pair' button is available. A note states: "For Windows instances, you use a key pair to decrypt the administrator password. You then use the decrypted password to connect to your instance."
- Network settings**:
 - VPC - required**: A dropdown shows 'vpc-0a647f3490b516713 (abc)' and '10.0.0.0/16'.
 - Subnet**: A dropdown shows 'subnet-091df7b18b99f6681' with details: 'VPC: vpc-0a647f3490b516713 Owner: 654654312679 Availability Zone: us-east-1a IP addresses available: 251 CIDR: 10.0.0.0/24'. A 'Create new subnet' button is present.
 - Auto-assign public IP**: A dropdown is set to 'Enable'.A note at the bottom states: "Additional charges apply when outside of free tier allowance".

Step-5: Change the country and repeat the above steps

The screenshot shows the 'Subnet 1 of 1' configuration screen in the AWS VPC console. It includes the following fields:

- Subnet name**: A text input field contains 'client'. A note says: "Create a tag with a key of 'Name' and a value that you specify." Below it, a note says: "The name can be up to 256 characters long."
- Availability Zone**: A dropdown is set to 'No preference'.
- IPv4 VPC CIDR block**: A dropdown shows '172.16.0.0/16'.
- IPv4 subnet CIDR block**: A dropdown shows '172.16.0.0/24' with '256 IPs' indicated to its right.

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*

kk [▼](#) [Create new key pair](#)

Network settings [Info](#)

VPC - *required* [Info](#)

vpc-0482acf6a9bfc5fa7 (xyz)
172.16.0.0/16 [▼](#) [Create new VPC](#)

Subnet [Info](#)

subnet-06a1edebbf7c3b45 client [▼](#) [Create new subnet](#)

VPC: vpc-0482acf6a9bfc5fa7 Owner: 654654312679
Availability Zone: us-west-2c IP addresses available: 251 CIDR: 172.16.0.0/24

Step-6: Change back to the original country and goto VPG

[VPC](#) > [Virtual private gateways](#) > [Create virtual private gateway](#)

Create virtual private gateway [Info](#)

A virtual private gateway is the VPN concentrator on the Amazon side of the site-to-site VPN connection.

Details

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

server to client [▼](#)
Value must be 256 characters or less in length.

Autonomous System Number (ASN)

Amazon default ASN
 Custom ASN

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. Name tag helps you track your resources more easily. We recommend adding Name tag.

Key	Value - <i>optional</i>
<input type="text"/> Name X	<input type="text"/> server to client X Remove

[Add new tag](#)

Step-7: Create VPG and attach it to VPC

VPC > Virtual private gateways > vgw-0f4d5f108c655462b > Attach to VPC

Attach to VPC Info

Details

Virtual private gateway ID
 vgw-0f4d5f108c655462b

Available VPCs
Attach the virtual private gateway to this VPC.

▼

Cancel Attach to VPC

Step-8: Goto Customer Gateway and create one

Details

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

Value must be 256 characters or less in length.

BGP ASN Info
The ASN of your customer gateway device.

Value must be in 1 - 2147483647 range.

IP address Info
Specify the IP address for your customer gateway device's external interface.

Certificate ARN
The ARN of a private certificate provisioned in AWS Certificate Manager (ACM).

Device - optional
Enter a name for the customer gateway device.

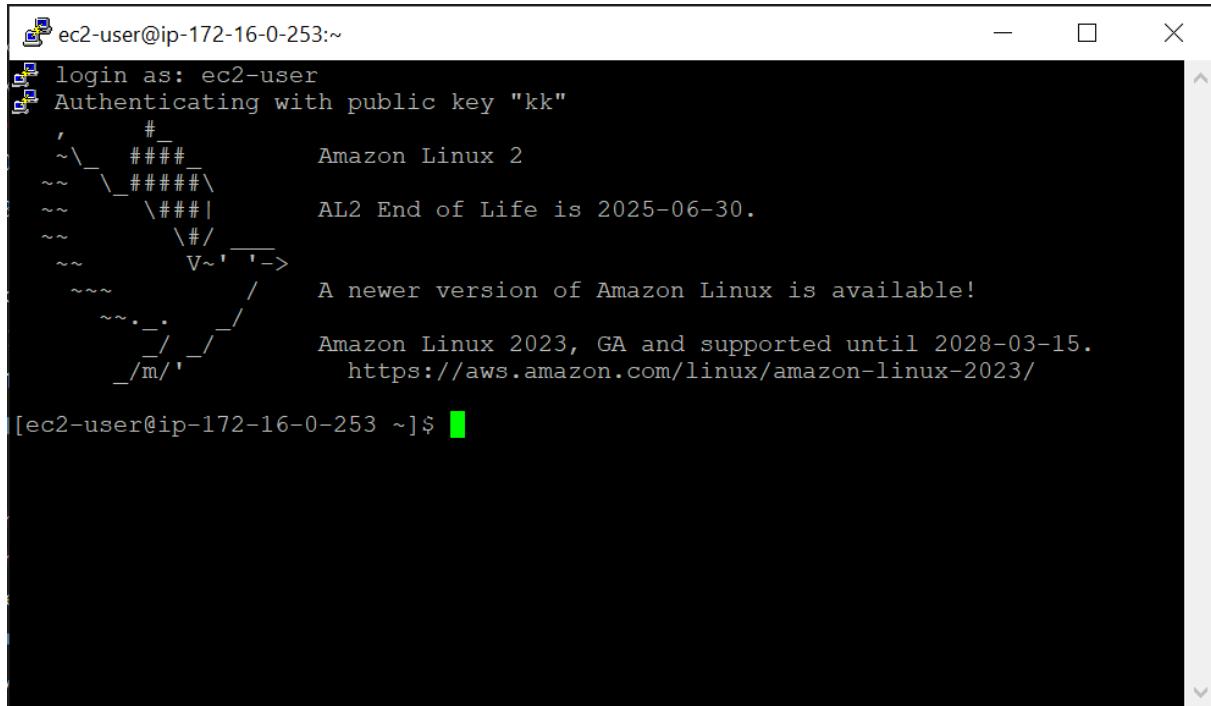
Step-9: Goto VPN and create one

The screenshot shows the 'Create VPN connection' page under 'VPC > VPN connections > Create VPN connection'. The 'Details' tab is selected. A 'Name tag - optional' field contains 'vpn'. Under 'Target gateway type', 'Virtual private gateway' is selected. In the 'Virtual private gateway' dropdown, 'vgw-0f4d5f108c655462b' is chosen. Under 'Customer gateway', 'Existing' is selected. The page includes a note: 'Creates a tag with a key of 'Name' and a value that you specify.'

Step-10: Select VPN which you created just now and download configuration of Openswan

The dialog box is titled 'Download configuration'. It contains fields for 'Vendor' (set to 'Openswan'), 'Platform' (set to 'Openswan'), 'Software' (set to 'Openswan 2.6.38+'), and 'IKE version' (set to 'ikev1'). At the bottom are 'Cancel' and 'Download' buttons.

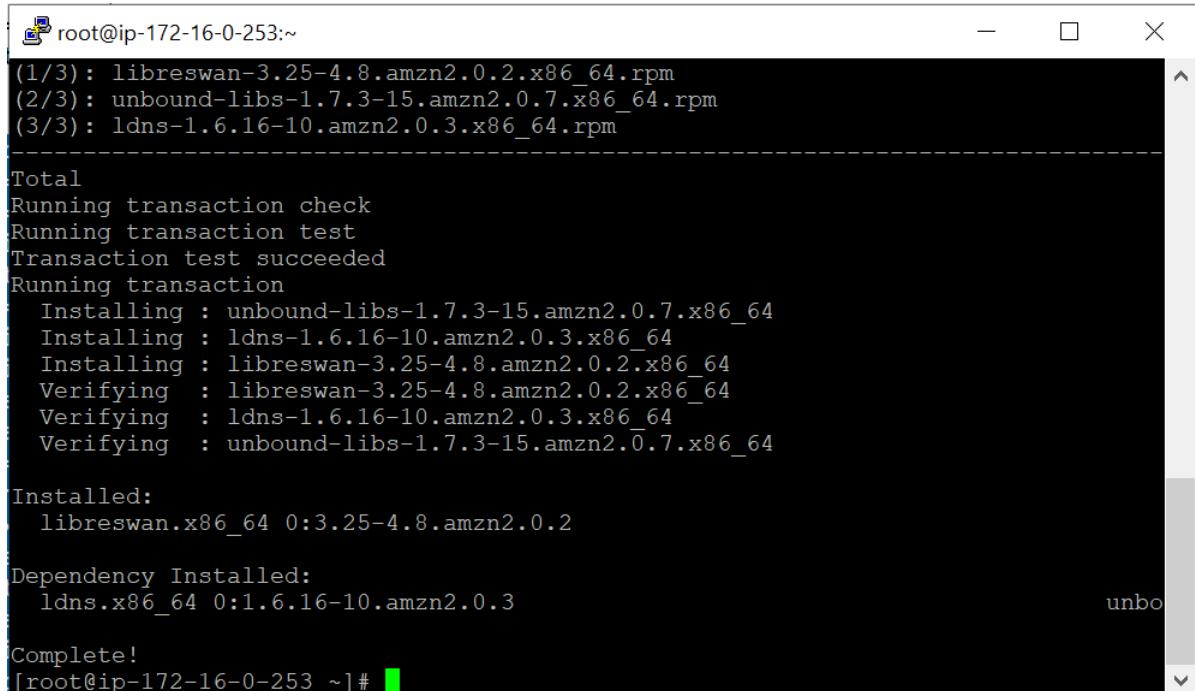
Step-11: Open PuTTY and paste client's public IP and login in it



The screenshot shows a PuTTY terminal window with the title bar "ec2-user@ip-172-16-0-253:~". The terminal displays the following text:

```
login as: ec2-user
Authenticating with public key "kk"
,
~\_ ##_
~~ \_\#\#\#\\
~~ \#\#\#|      Amazon Linux 2
~~ \#/          AL2 End of Life is 2025-06-30.
~~ V~' '-->
~~ /          A newer version of Amazon Linux is available!
~~*_. /_/
~/m/' _/      Amazon Linux 2023, GA and supported until 2028-03-15.
https://aws.amazon.com/linux/amazon-linux-2023/
[ec2-user@ip-172-16-0-253 ~]$
```

Step-12: Enter the commands in the terminal and configure the Openswan file



The screenshot shows a PuTTY terminal window with the title bar "root@ip-172-16-0-253:~". The terminal displays the following text:

```
(1/3): libreswan-3.25-4.8.amzn2.0.2.x86_64.rpm
(2/3): unbound-libs-1.7.3-15.amzn2.0.7.x86_64.rpm
(3/3): ldns-1.6.16-10.amzn2.0.3.x86_64.rpm
-----
Total
Running transaction check
Running transaction test
Transaction test succeeded
Running transaction
  Installing : unbound-libs-1.7.3-15.amzn2.0.7.x86_64
  Installing : ldns-1.6.16-10.amzn2.0.3.x86_64
  Installing : libreswan-3.25-4.8.amzn2.0.2.x86_64
  Verifying   : libreswan-3.25-4.8.amzn2.0.2.x86_64
  Verifying   : ldns-1.6.16-10.amzn2.0.3.x86_64
  Verifying   : unbound-libs-1.7.3-15.amzn2.0.7.x86_64
-----
Installed:
  libreswan.x86_64 0:3.25-4.8.amzn2.0.2
Dependency Installed:
  ldns.x86_64 0:1.6.16-10.amzn2.0.3
  unbo
Complete!
[root@ip-172-16-0-253 ~] #
```

Step-13: Check the status in the terminal if it's active now

```
[root@ip-172-16-0-253 ~]# systemctl start ipsec
Job for ipsec.service failed because the control process exited with error code.
See "systemctl status ipsec.service" and "journalctl -xe" for details.
[root@ip-172-16-0-253 ~]# vim /etc/ipsec.d/aws.conf
[root@ip-172-16-0-253 ~]# systemctl start ipsec
[root@ip-172-16-0-253 ~]# systemctl status ipsec
● ipsec.service - Internet Key Exchange (IKE) Protocol Daemon for IPsec
   Loaded: loaded (/usr/lib/systemd/system/ipsec.service; enabled; vendor preset
: disabled)
     Active: active (running) since Sun 2024-03-31 22:44:11 UTC; 5s ago
       Docs: man:ipsec(8)
              man:pluto(8)
              man:ipsec.conf(5)
   Process: 3975 ExecStopPost=/usr/sbin/ipsec --stopnflg (code=exited, status=0/
SUCCESS)
   Process: 3972 ExecStopPost=/sbin/ip xfrm state flush (code=exited, status=0/SU
CESS)
   Process: 3970 ExecStopPost=/sbin/ip xfrm policy flush (code=exited, status=0/S
UCCESS)
   Process: 4509 ExecStartPre=/usr/sbin/ipsec --checknflog (code=exited, status=0
/SUCCESS)
   Process: 4503 ExecStartPre=/usr/sbin/ipsec --checknss (code=exited, status=0/S
UCCESS)
   Process: 3992 ExecStartPre=/usr/libexec/ipsec/_stackmanager start (code=exited
, status=0/SUCCESS)
   Process: 3990 ExecStartPre=/usr/libexec/ipsec/addconn --config /etc/ipsec.conf
--checkconfig (code=exited, status=0/SUCCESS)
Main PID: 4527 (pluto)
  Status: "Startup completed."
  CGroup: /system.slice/ipsec.service
         └─4527 /usr/libexec/ipsec/pluto --leak-detective --config /etc/ips...
Mar 31 22:44:12 ip-172-16-0-253.us-west-2.compute.internal pluto[4527]: "Tunn...
Mar 31 22:44:12 ip-172-16-0-253.us-west-2.compute.internal pluto[4527]: "Tunn...
Mar 31 22:44:13 ip-172-16-0-253.us-west-2.compute.internal pluto[4527]: "Tunn...
Mar 31 22:44:15 ip-172-16-0-253.us-west-2.compute.internal pluto[4527]: "Tunn...
Hint: Some lines were ellipsized, use -l to show in full.
[root@ip-172-16-0-253 ~]#
```

Output:- Successfully created a VPN using AWS console.

Experiment-7

Aim:- To create an autoscaling using AWS console.

Procedure:-

Step-1: Create VPC

The screenshot shows the 'Create VPC' settings page. Under 'VPC settings', the 'Resources to create' section has 'VPC only' selected. A 'Name tag - optional' field contains 'abc'. Under 'IPv4 CIDR block', 'IPv4 CIDR manual input' is selected, and the CIDR block '10.0.0.0/16' is specified. A note at the bottom states 'CIDR block size must be between /16 and /28.'

Step-2: Create 2 Subnets with 2 different AZs preference

The screenshot shows the 'Create subnet' process. On the left, the 'VPC' tab is active, displaying the VPC ID 'vpc-0a647f3490b516713 (abc)' and the associated CIDR '10.0.0.0/16'. On the right, the 'Subnet settings' tab is active, showing 'Subnet 1 of 3'. It includes fields for 'Subnet name' (set to 'sub1'), 'Availability Zone' (set to 'US East (N. Virginia) / us-east-1a'), 'IPv4 VPC CIDR block' (set to '10.0.0.0/16'), and 'IPv4 subnet CIDR block' (set to '10.0.0.0/24'). A note indicates '256 IPs'. A second tab, 'Subnet 2 of 3', is visible on the right, showing 'Subnet name' (set to 'sub2'), 'Availability Zone' (set to 'US East (N. Virginia) / us-east-1b'), 'IPv4 VPC CIDR block' (set to '10.0.0.0/16'), and 'IPv4 subnet CIDR block' (set to '10.0.1.0/24'). A note indicates '256 IPs'.

Step-3: Create Internet gateway and attach it to the VPC created above and then edit route tables

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.

abcig

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
Q Name	Q abcig X Remove

Add new tag

You can add 49 more tags.

Cancel **Create internet gateway**

VPC > Internet gateways > Attach to VPC (igw-02b8f7ce8499f3abc)

Attach to VPC (igw-02b8f7ce8499f3abc) Info

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.

vpc-0a647f3490b516713 X

▶ AWS Command Line Interface command

Cancel **Attach internet gateway**

VPC > Route tables > rtb-01682d6b44dbf9b3f > 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-02b8f7ce8499f3abc X		Remove

Add route

Cancel Preview Save changes

Step-4: Go to EC2 and create 2 instances

The image consists of two side-by-side screenshots of the AWS EC2 'Launch an instance' wizard. Both screenshots show the 'Name and tags' step at the top, followed by 'Key pair (login)' and 'Network settings' sections.

Name and tags:

- Left screenshot: Name is set to "autoscaling".
- Right screenshot: Name is set to "my ami".

Key pair (login):

- Both screenshots show a dropdown menu for "Key pair name - required" with "aa" selected.
- Left screenshot: A note says "For Windows instances, you use a key pair to decrypt the administrator password. You then use the decrypted password to connect to your instance."
- Right screenshot: A "Create new key pair" button is visible.

Network settings:

- Both screenshots show a dropdown menu for "VPC - required" with "vpc-0a647f3490b516713 (abc)" selected.
- Left screenshot: Subnet is set to "subnet-091df7b18b99f6681" (sub1).
- Right screenshot: A "Create new subnet" button is visible.
- Both screenshots show an "Auto-assign public IP" dropdown set to "Enable".
- Left screenshot: A note says "Additional charges apply when outside of free tier allowance".

Step-5: Select the instance, then select image and template in action and then create image

The image shows the 'Create image' wizard for instance `i-061a7c51de97c6a12`. The wizard has the following steps:

- Create image**: An image (AMI) defines the programs and settings applied to a new instance.
- Instance ID**: `i-061a7c51de97c6a12` (autoscaling).
- Image name**: `xyz` (Maximum 127 characters,不可修改).
- Image description - optional**: `Image description` (Maximum 255 characters).
- No reboot**: Enable.

Step-6: Go to Launch Templates and create a template

The screenshot shows the 'Create launch template' wizard in the AWS EC2 console. The left sidebar lists steps: Step 1 (Choose launch template), Step 2 (Choose instance launch options), Step 3 - optional (Configure advanced options), Step 4 - optional (Configure group size and scaling), Step 5 - optional (Add notifications), Step 6 - optional (Add tags), and Step 7 (Review). The main panel is titled 'Create launch template' and contains fields for 'Launch template name - required' (xyz_usc_image) and 'Template version description' (A prod webserver for MyApp). A note says 'Must be unique to this account. Max 128 chars. No spaces or special characters like '&', '*', '@'.'. Below these are sections for 'Auto Scaling guidance' (Info) and 'Provide guidance to help me set up a template that I can use with EC2 Auto Scaling'. A checkbox for this option is checked. On the right, a 'Summary' section lists 'Software Image (AMI)', 'Virtual server type (instance type)', 'Firewall (security group)', and 'Storage (volumes)'. A tooltip for the 'Free tier' indicates it includes 750 hours of t2.micro or t3.micro usage per month. At the bottom are 'Cancel' and 'Create launch template' buttons.

Step-6: Go to AutoScaling Groups and create one using existing template

The screenshot shows the 'Create Auto Scaling group' wizard in the AWS Auto Scaling console. The left sidebar lists steps: Step 1 (Choose launch template), Step 2 (Choose instance launch options), Step 3 - optional (Configure advanced options), Step 4 - optional (Configure group size and scaling), Step 5 - optional (Add notifications), Step 6 - optional (Add tags), and Step 7 (Review). The main panel is titled 'Choose launch template' (Info) and asks to specify a launch template for all instances. It has a 'Name' field and an 'Auto Scaling group name' field (xyz). A note says 'Must be unique to this account in the current Region and no more than 255 characters.' Below this is a 'Launch template' section with a note about accounts created after May 31, 2023. It shows a dropdown menu with 'xyz_usc_image' selected and a 'Create a launch template' button. At the bottom are 'Create' and 'Cancel' buttons.

Step-7: Select both subnets, no load balancer, no VPC lattice services, no health checks, desired=2, max 4, min=1, no scaling policies, no policy in replacement behaviour, no emails/sms notifications, no tags given

Network Info

For most applications, you can use multiple Availability Zones and let EC2 Auto Scaling balance your instances across the zones. The default VPC and default subnets are suitable for getting started quickly.

VPC
Choose the VPC that defines the virtual network for your Auto Scaling group.

vpc-0a647f3490b516713 (abc) ▾ C
10.0.0.0/16

[Create a VPC](#)

Availability Zones and subnets
Define which Availability Zones and subnets your Auto Scaling group can use in the chosen VPC.

Select Availability Zones and subnets ▾ C

us-east-1b | subnet-0442770f24b81da76 (sub1) X
10.0.1.0/24

us-east-1c | subnet-073c7b7f328597bfa (sub2) X
10.0.2.0/24

[Create a subnet](#)

Load balancing Info

Use the options below to attach your Auto Scaling group to an existing load balancer, or to a new load balancer that you define.

No load balancer
Traffic to your Auto Scaling group will not be fronted by a load balancer.

Attach to an existing load balancer
Choose from your existing load balancers.

Attach to a new load balancer
Quickly create a basic load balancer to attach to your Auto Scaling group.

VPC Lattice integration options Info

To improve networking capabilities and scalability, integrate your Auto Scaling group with VPC Lattice. VPC Lattice facilitates communications between AWS services and helps you connect and manage your applications across compute services in AWS.

Select VPC Lattice service to attach

No VPC Lattice service
VPC Lattice will not manage your Auto Scaling group's network access and connectivity with other services.

Attach to VPC Lattice service
Incoming requests associated with specified VPC Lattice target groups will be routed to your Auto Scaling group.

[Create new VPC Lattice service](#)

Group size [Info](#)

Set the initial size of the Auto Scaling group. After creating the group, you can change its size to meet changes in demand by using CloudWatch Metrics or scheduled scaling.

Desired capacity type

Choose the unit of measurement for the desired capacity value. vCPUs and Memory(GiB) are only configured with a set of instance attributes.

Units (number of instances) ▾

Desired capacity

Specify your group size.

2

Scaling [Info](#)

You can resize your Auto Scaling group manually or automatically to meet changes in demand.

Scaling limits

Set limits on how much your desired capacity can be increased or decreased.

Min desired capacity 1 Equal or less than desired capacity

Max desired capacity 4 Equal or greater than desired capacity

Automatic scaling - optional

Choose whether to use a target tracking policy | [Info](#)

You can set up other metric-based scaling policies and scheduled scaling after creating your Auto Scaling group.

No scaling policies
Your Auto Scaling group will remain at its initial size and will not dynamically resize to meet demand.

Target tracking scaling policy
Choose a CloudWatch metric and target value and let the scaling policy adjust the desired capacity in proportion to the metric's value.

Instance maintenance policy [Info](#)

Control your Auto Scaling group's availability during instance replacement events. This includes health checks, instance refreshes, maximum instance lifetime features and events that happen automatically to keep your group balanced, called rebalancing events.

Choose a replacement behavior depending on your availability requirements

Mixed behavior

No policy
For rebalancing events, new instances will launch before terminating others. For all other events, instances terminate and launch at the same time.

Prioritize availability

Launch before terminating
Launch new instances and wait for them to be ready before terminating others. This allows you to go above your desired capacity by a given percentage and may temporarily increase costs.

Control costs

Terminate and launch
Terminate and launch instances at the same time. This allows you to go below your desired capacity by a given percentage and may temporarily reduce availability.

Flexible

Custom behavior
Set custom values for the minimum and maximum amount of available capacity. This gives you greater flexibility in setting how far below and over your desired capacity EC2 Auto Scaling goes when replacing instances.

Output:- Created an autoscaling using AWS console.

Experiment-8

Aim:- To create a load-balancer by application using AWS console.

Procedure:-

Step-1: Create VPC

The screenshot shows the 'Create VPC' configuration page. Under 'VPC settings', the 'Resources to create' section has 'VPC only' selected. A 'Name tag - optional' field contains 'abc'. Under 'IPv4 CIDR block', 'IPv4 CIDR manual input' is selected, and the CIDR block '10.0.0.0/16' is entered. A note at the bottom states 'CIDR block size must be between /16 and /28.'

Step-2: Create 3 Subnets with 3 different AZs preference

The screenshot shows the 'Create subnet' configuration page. Under 'VPC', the 'VPC ID' is set to 'vpc-0a647f3490b516713 (abc)'. In the 'Associated VPC CIDRs' section, '10.0.0.0/16' is listed. Under 'Subnet settings', 'Subnet 1 of 3' is being configured. The 'Subnet name' is 'sub1', and the 'Availability Zone' is 'US East (N. Virginia) / us-east-1a'. The 'IPv4 VPC CIDR block' is '10.0.0.0/16', and the 'IPv4 subnet CIDR block' is '10.0.0.0/24'.

Subnet 2 of 3

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

Subnet 3 of 3

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

Step-3: Create Internet gateway and attach it to the VPC created above and then edit route tables

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="abcig"/> X

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

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

VPC > [Internet gateways](#) > [Attach to VPC \(igw-02b8f7ce8499f3abc\)](#)

Attach to VPC (igw-02b8f7ce8499f3abc) [Info](#)

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](#)

VPC > Route tables > rtb-01682d6b44dbf9b3f > 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-02b8f7ce8499f3abc		Remove

Add route Cancel Preview Save changes

Step-4: Go to EC2 and create 3 instances

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: rload 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

Quick Start

Amazon Linux macOS Ubuntu Windows Red Hat SUSE Linux

Browse more AMIs Including AMIs from AWS, Marketplace and the Community

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: aa Create new key pair

For Windows instances, you use a key pair to decrypt the administrator password. You then use the decrypted password to connect to your instance.

Network settings Info

VPC - required Info

vpc-0a647f3490b516713 (abc) 10.0.0.0/16

Subnet Info

subnet-091df7b18b99f6681 sub1 VPC: vpc-0a647f3490b516713 Owner: 654654312679 Availability Zone: us-east-1a IP addresses available: 251 CIDR: 10.0.0.0/24 Create new subnet

Auto-assign public IP Info

Enable Additional charges apply when outside of free tier allowance

EC2 > Instances > Launch an instance

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
sload Add additional tags

EC2 > Instances > Launch an instance

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
mload Add additional tags

Step-5: Go to Target Groups and create target groups

Target group name
abc

A maximum of 32 alphanumeric characters including hyphens are allowed, but the name must not begin or end with a hyphen.

Protocol : Port
Choose a protocol for your target group that corresponds to the Load Balancer type that will route traffic to it. Some protocols now include anomaly detection for the targets and you can set mitigation options once your target group is created. This choice cannot be changed after creation

HTTP ▾ 80 1-65535

IP address type
Only targets with the indicated IP address type can be registered to this target group.

IPv4
Each instance has a default network interface (eth0) that is assigned the primary private IPv4 address. The instance's primary private IPv4 address is the one that will be applied to the target.

IPv6
Each instance you register must have an assigned primary IPv6 address. This is configured on the instance's default network interface (eth0). [Learn more](#)

VPC
Select the VPC with the instances that you want to include in the target group. Only VPCs that support the IP address type selected above are available in this list.

abc
vpc-0a647f3490b516713
IPv4 VPC CIDR: 10.0.0.0/16

Register targets

This is an optional step to create a target group. However, to ensure that your load balancer routes traffic to this target group you must register your targets.

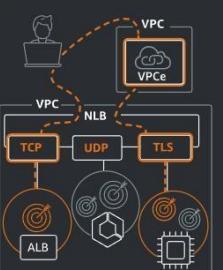
Available instances (3/3)				
<input type="checkbox"/> Filter instances				
Instance ID	Name	State	Security groups	
i-0b3747b35ebe772e8	mload	Running	launch-wizard-3	<input checked="" type="checkbox"/>
i-0b800f2e3b0ffc1f0	sload	Running	launch-wizard-2	<input checked="" type="checkbox"/>
i-097984cc886b06989	rload	Running	launch-wizard-1	<input checked="" type="checkbox"/>

Step-6: Go to Load Balancer and select application load balancer

EC2 > Load balancers > Compare and select load balancer type

Compare and select load balancer type

A complete feature-by-feature comparison along with detailed highlights is also available. Learn more [\[?\]](#)

Load balancer types		
Application Load Balancer Info  ALB	Network Load Balancer Info  VPC NLB TCP UDP TLS	Gateway Load Balancer Info  VPC GWLB

EC2 > Load balancers > Create Application Load Balancer

Create Application Load Balancer [Info](#)

The Application Load Balancer distributes incoming HTTP and HTTPS traffic across multiple targets such as Amazon EC2 instances, microservices, and AWS Lambda functions. You can define rules based on request attributes. When the load balancer receives a connection request, it evaluates the listener rules in priority order to determine which rule is applicable, it selects a target from the target group for the rule action.

▶ How Application Load Balancers work

Basic configuration

Load balancer name
Name must be unique within your AWS account and can't be changed after the load balancer is created.

A maximum of 32 alphanumeric characters including hyphens are allowed, but the name must not begin or end with a hyphen.

Scheme [Info](#)
Scheme can't be changed after the load balancer is created.

Internet-facing
An internet-facing load balancer routes requests from clients over the internet to targets. Requires a public subnet. [Learn more \[?\]](#)

Internal
An internal load balancer routes requests from clients to targets using private IP addresses.

IP address type [Info](#)
Select the type of IP addresses that your subnets use.

IPv4
Includes only IPv4 addresses.

Dualstack

VPC Info
Select the virtual private cloud (VPC) for your targets or you can [create a new VPC](#). Only VPCs with an internet gateway are enabled for selection. The selected VPC can't be changed after the load balancer is created. To confirm the VPC for your targets, view your [target groups](#).

Mappings | [Info](#)
Select at least two Availability Zones and one subnet per zone. The load balancer routes traffic to targets in these Availability Zones only. Availability Zones that are not supported by the load balancer or the VPC are not available for selection.

us-east-1a (use1-az2)
Subnet
subnet-091df7b18b99f6681 sub1 ▾
IPv4 address
Assigned by AWS

us-east-1b (use1-az4)
Subnet
subnet-0442770f24b81da76 sub2 ▾
IPv4 address
Assigned by AWS

us-east-1c (use1-az6)
Subnet
subnet-073c7b7f328597bfa sub3 ▾
IPv4 address
Assigned by AWS

Step-7: Open RDP clients for all three and host static website in it.

Connect to instance | [Info](#)
Connect to your instance i-097984cc886b06989 (rload) using any of these options

Session Manager | **RDP client** | **EC2 serial console**

Instance ID
 [i-097984cc886b06989 \(rload\)](#)

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](#)

Server Manager | Dashboard

Add Roles and Features Wizard

Installation progress

Before You Begin
Installation Type
Server Selection
Server Roles
Features

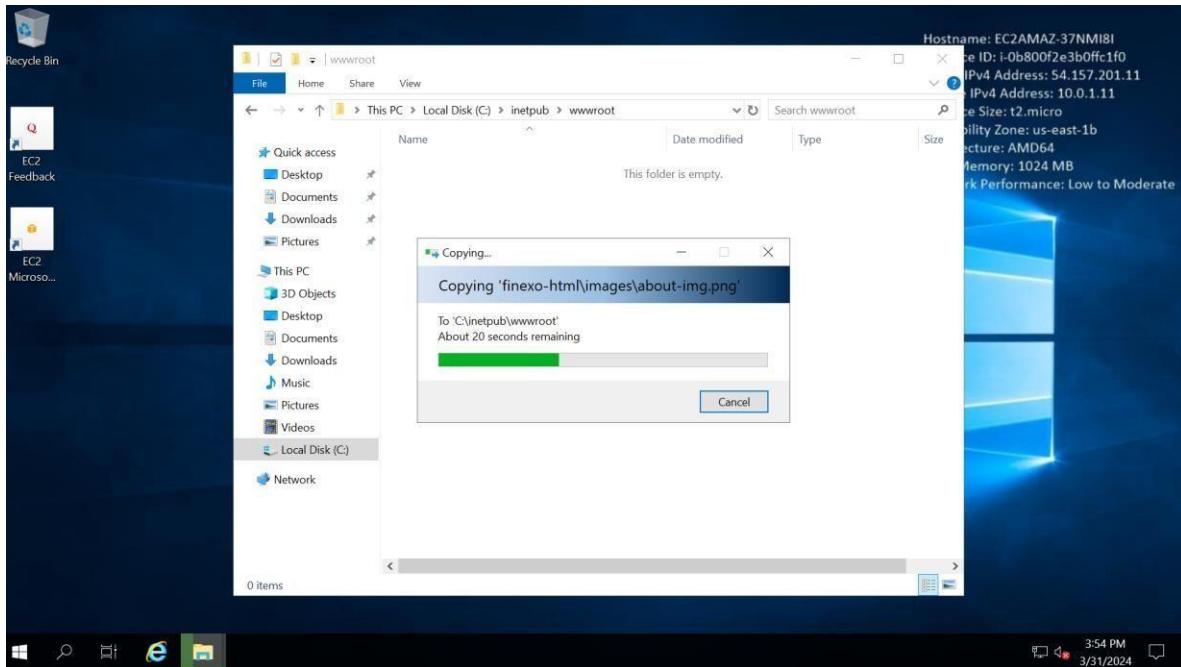
Web Server (IIS)
Role Services
Configuration
Results

Feature installation
Installation succeeded on EC2AMAZ-5B0V40K.

Web Server (IIS)
Web Server
Common HTTP Features
Static Content
Default Document
Directory Browsing
HTTP Errors
Security
Request Filtering
Health and Diagnostics
HTTP Logging

BPA results

3:56 PM 3/31/2024



Step-8: Goto load balancer and open DNS link

A screenshot of a web browser displaying the homepage of 'NEOGYM'. The page features a dark background with a woman in a red tank top and blue leggings performing a plank exercise. The word 'FITNESS' is written vertically above 'TRAINING' in white. Below that, 'NEOGYM' is written in large, bold, white letters. At the top, there's a navigation bar with links for 'HOME', 'WHY US', 'TRAINERS', and 'CONTACT US'. A small paragraph of placeholder text follows the main title. A 'CONTACT US' button is located near the bottom left of the main content area.

A screenshot of a web browser displaying the homepage of 'Kider'. The page features a large image of a young child with brown hair, wearing a blue patterned shirt, sitting at a table and looking down at something. To the left of the image, the text 'The Best Kindergarten School For Your Child' is displayed in large, bold, white font. Below this, a smaller paragraph of placeholder text is visible. The top navigation bar includes links for 'Home', 'About Us', 'Classes', 'Pages', 'Contact Us', and a 'Join Us' button.

Output:- Created a load-balancer by application using AWS console.

Experiment-9

Aim:- To create a load-balancer by networking using AWS console.

Procedure:-

Step-1: Create VPC

VPC > Your VPCs > Create VPC

Create VPC Info

A VPC is an isolated portion of the AWS Cloud populated by AWS objects, such as Amazon EC2 instances.

VPC settings

Resources to create Info
Create only the VPC resource or the VPC and other networking resources.

VPC only VPC and more

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

IPv4 CIDR block Info
 IPv4 CIDR manual input IPAM-allocated IPv4 CIDR block

IPv4 CIDR
10.0.0.0/16

CIDR block size must be between /16 and /28.

Step-2: Create 2 Subnets with 2 different AZs preference

VPC > Subnets > Create subnet

Create subnet Info

VPC

VPC ID
Create subnets in this VPC.
vpc-0a647f3490b516713 (abc)

Associated VPC CIDRs
IPv4 CIDRs
10.0.0.0/16

Subnet settings

Specify the CIDR blocks and Availability Zone for the subnet.

Subnet 1 of 3

Subnet name
Create a tag with a key of 'Name' and a value that you specify.
sub1
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

Subnet 2 of 3

Subnet name
Create a tag with a key of 'Name' and a value that you specify.
sub2
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-1b

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
10.0.1.0/24 256 IPs

Step-3: Create Internet gateway and attach it to the VPC created above and then edit route tables

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.

abcig

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
Q Name	Q abcig X Remove

Add new tag

You can add 49 more tags.

Cancel **Create internet gateway**

VPC > Internet gateways > Attach to VPC (igw-02b8f7ce8499f3abc)

Attach to VPC (igw-02b8f7ce8499f3abc) Info

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.

vpc-0a647f3490b516713 X

▶ AWS Command Line Interface command

Cancel **Attach internet gateway**

VPC > Route tables > rtb-01682d6b44dbf9b3f > 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-02b8f7ce8499f3abc X		Remove

Add route

Cancel Preview Save changes

Step-4: Go to EC2 and create 3 instances

The screenshot shows the 'Launch an instance' wizard in the AWS Management Console. It displays two parallel steps for launching instances n1 and n2. Both steps include sections for 'Name and tags', 'Key pair (login)', and 'Network settings'. In the 'Network settings' section, both instances are assigned to the same VPC (vpc-0a647f3490b516713) and subnet (subnet-091df7b18b99f6681). The 'Auto-assign public IP' option is enabled for both.

Step-5: Go to Target Groups and create target groups

The screenshot shows the 'Target Groups' configuration page. It lists three options: 'IP addresses' (selected), 'Lambda function', and 'Application Load Balancer'. The 'IP addresses' section includes a bulleted list of benefits and a note about IPv6 support. The 'Target group name' field is set to 'abc'. The 'Protocol : Port' section shows 'TCP' selected as the protocol and port 80. The 'IP address type' section shows 'IPv4' selected.

[EC2](#) > [Target groups](#) > [Create target group](#)

Step 1
[Specify group details](#)

Step 2
[Register targets](#)

Register targets

This is an optional step to create a target group. However, to ensure that your load balancer routes traffic to this target group you must register your targets.

IP addresses

Step 1: Choose a network
You can add IP addresses from the VPC selected for your target group or from outside the VPC. Note that you can assemble a mix of targets from multiple network sources by returning to this step and choosing another network.

Network

abc
vpc-0a647f3490b516713
IPv4 VPC CIDR: 10.0.0.0/16

Step 2: Specify IPs and define ports
You can manually enter IP addresses from the selected network.

Enter an IPv4 address from a VPC subnet.

10.0.1.200 Remove

10.0.2.171 Remove

[Add IPv4 address](#)

Step-6: Go to Load Balancer and select network load balancer

[EC2](#) > [Load balancers](#) > [Compare and select load balancer type](#)

Compare and select load balancer type

A complete feature-by-feature comparison along with detailed highlights is also available. [Learn more](#)

Load balancer types

Application Load Balancer	Network Load Balancer	Gateway Load Balancer
Info	Info	Info

[EC2](#) > [Load balancers](#) > [Create Network Load Balancer](#)

Create Network Load Balancer

The Network Load Balancer distributes incoming TCP and UDP traffic across multiple targets such as Amazon EC2 instances, microservices, and containers. When the load balancer receives a connection request, it selects a target based on the protocol and port that are specified in the listener configuration, and the routing rule specified as the default action.

► [How Network Load Balancers work](#)

Basic configuration

Load balancer name
Name must be unique within your AWS account and can't be changed after the load balancer is created.

A maximum of 32 alphanumeric characters including hyphens are allowed, but the name must not begin or end with a hyphen.

Scheme
Scheme can't be changed after the load balancer is created.

[Internet-facing](#)
An internet-facing load balancer routes requests from clients over the internet to targets. Requires a public subnet. [Learn more](#)

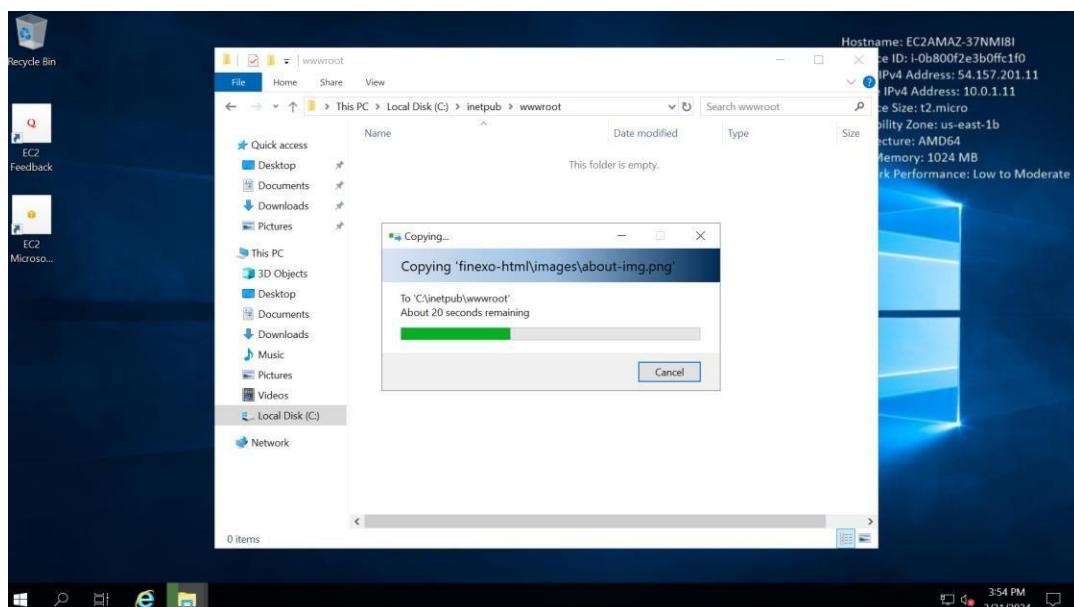
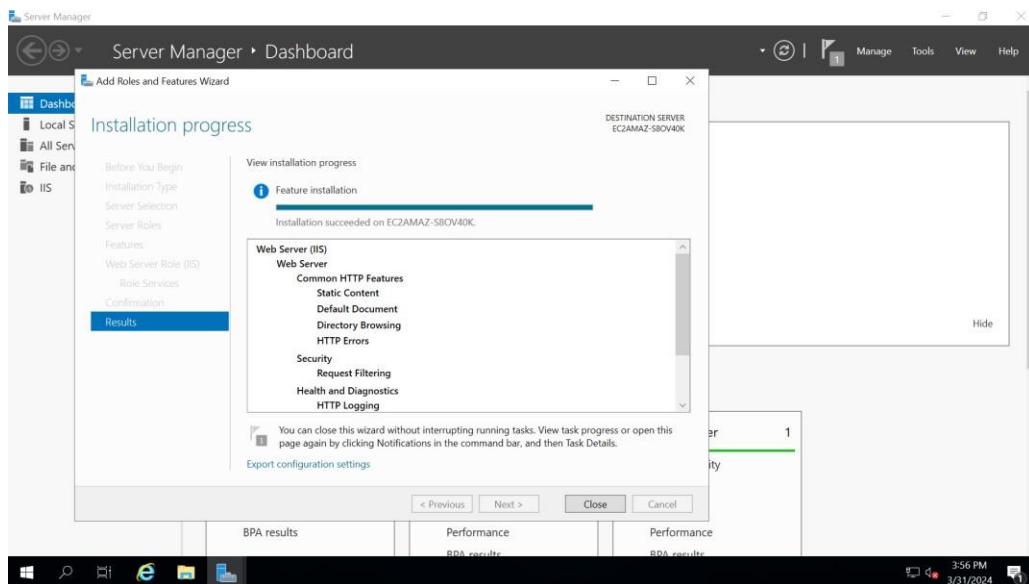
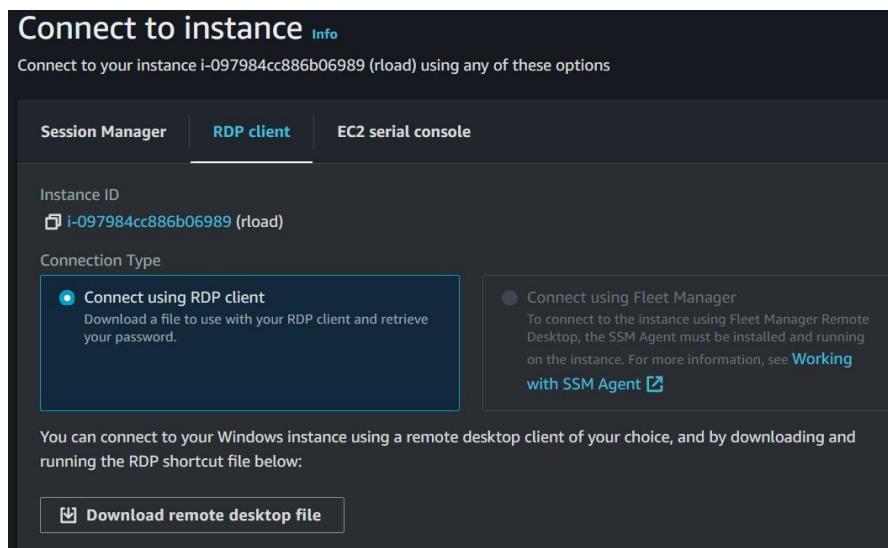
[Internal](#)
An internal load balancer routes requests from clients to targets using private IP addresses.

IP address type [Info](#)
Select the type of IP addresses that your subnets use.

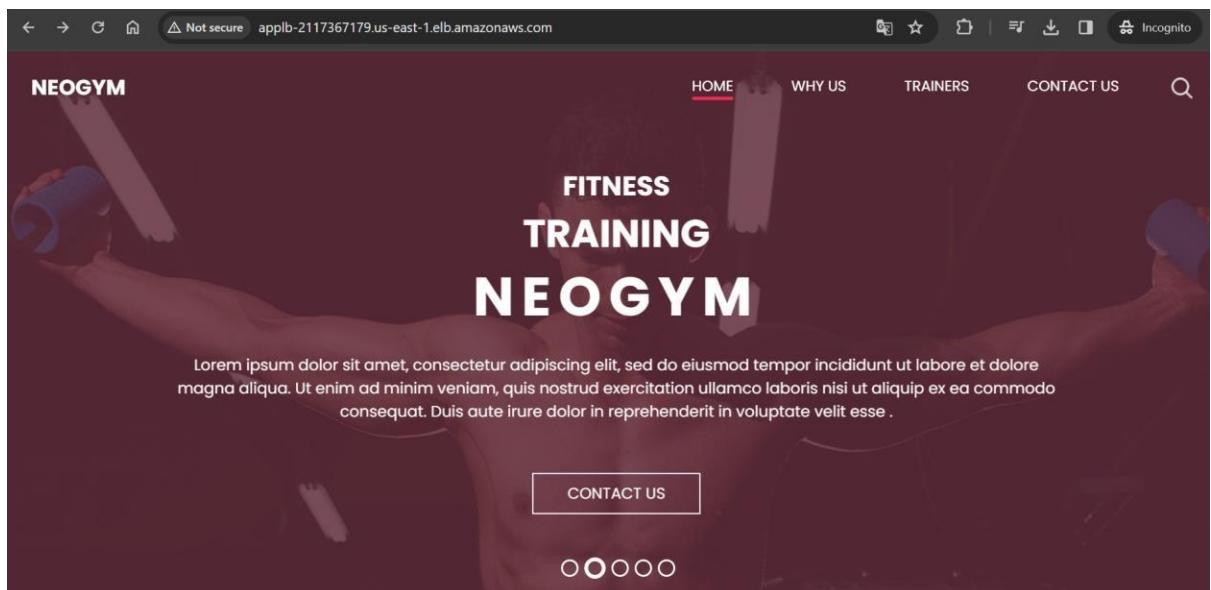
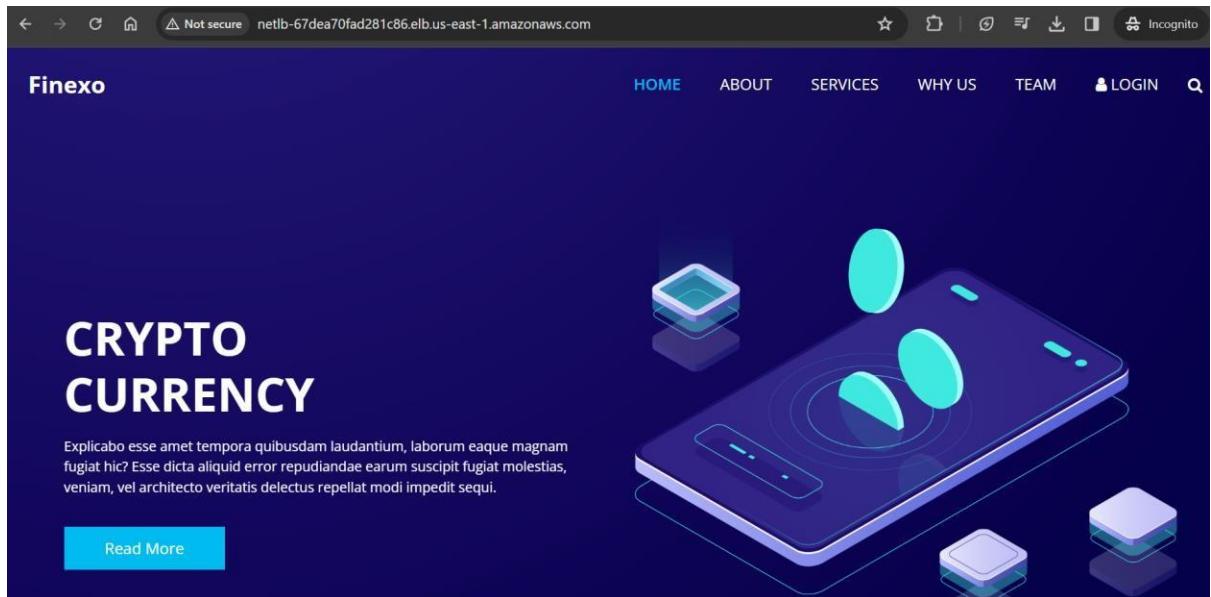
[IPv4](#)
Includes only IPv4 addresses.

[Dualstack](#)
Includes IPv4 and IPv6 addresses.

Step-7: Open RDP clients for the two instances and host static website in it.



Step-8: Goto load balancer and open DNS link



Output:- Created a load-balancer by networking using AWS console.