

# cprime

AWS VPC Deployment with Public-Private Subnet

Segregation



NAME : T MANOJ

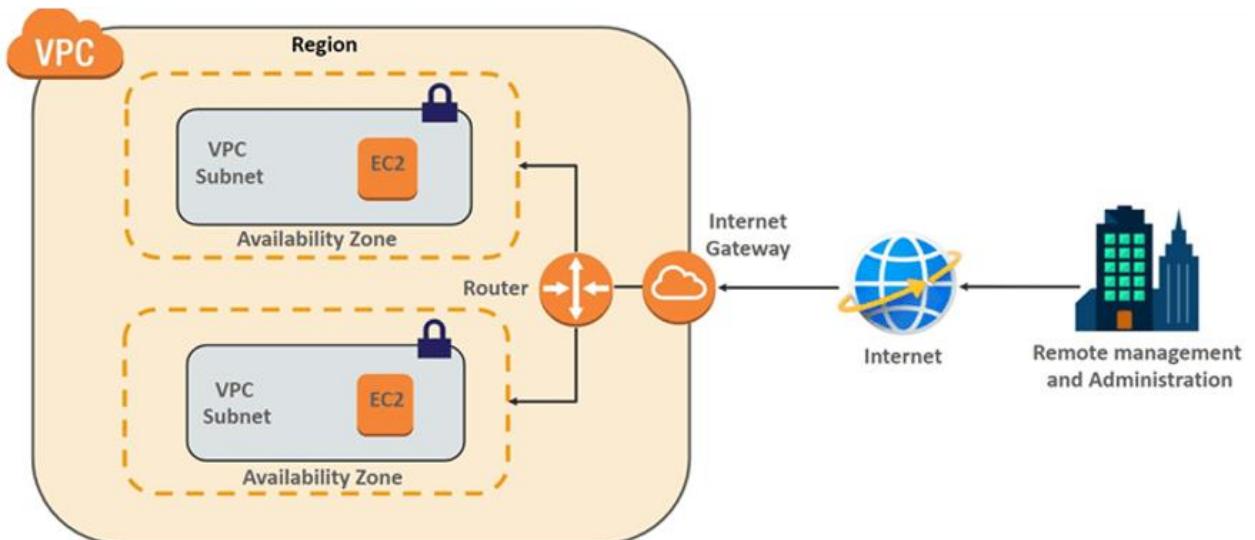
EMP ID : LYAKE2KHS

## Overview

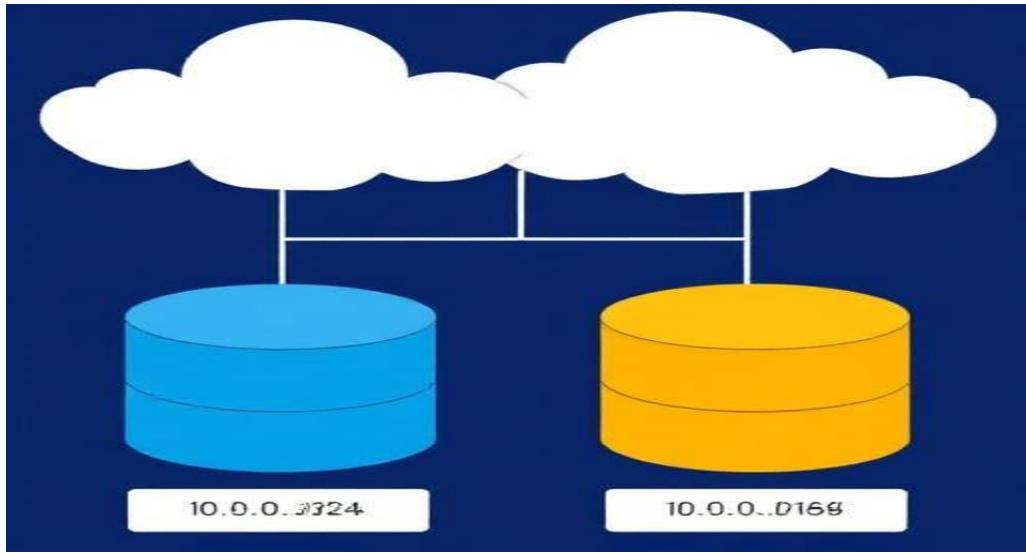
This project demonstrates the complete setup of a Virtual Private Cloud (VPC) in AWS, designed with both public and private subnets to securely deploy web applications. The public subnet hosts a home page EC2 instance with direct internet access, while the private subnet houses a login page EC2 instance without direct internet exposure. The private instance is accessible only through the public instance, ensuring enhanced security. The network is structured using proper CIDR allocations, route tables, an Internet Gateway for the public subnet, and a NAT Gateway for private subnet internet access. The project also integrates Apache HTTPD servers on both instances and uses a reverse proxy mechanism, allowing users to access the login page hosted on the private instance by clicking a button on the home page.

## Terminologies :

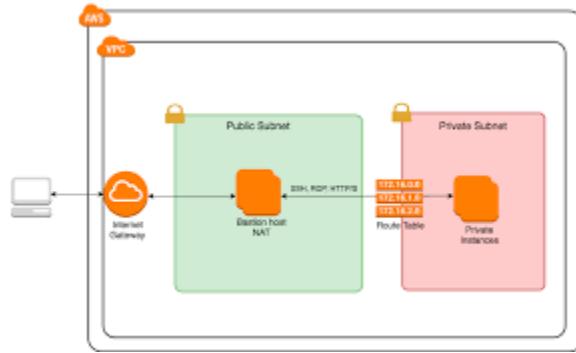
1. **Virtual Private Cloud (VPC):** A logically isolated network in AWS where you can launch resources in a secure environment. It provides full control over networking components, including IP address ranges, subnets, route tables, and gateways.



2. **Subnets:** Subdivisions of a VPC that allow you to group resources based on security and operational needs. Public subnets connect directly to the internet, while private subnets do not and are typically used for backend services.



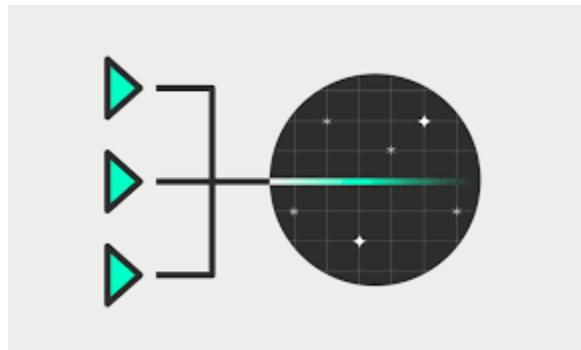
3. **Route Tables:** Define how network traffic is directed within the VPC. Each subnet must be associated with a route table to determine how instances communicate with each other, with gateways, and with the internet.



4. **Internet Gateway (IGW):** A VPC component that allows communication between instances in the VPC and the internet. It is attached to the VPC and connected to public subnets through appropriate route table entries.



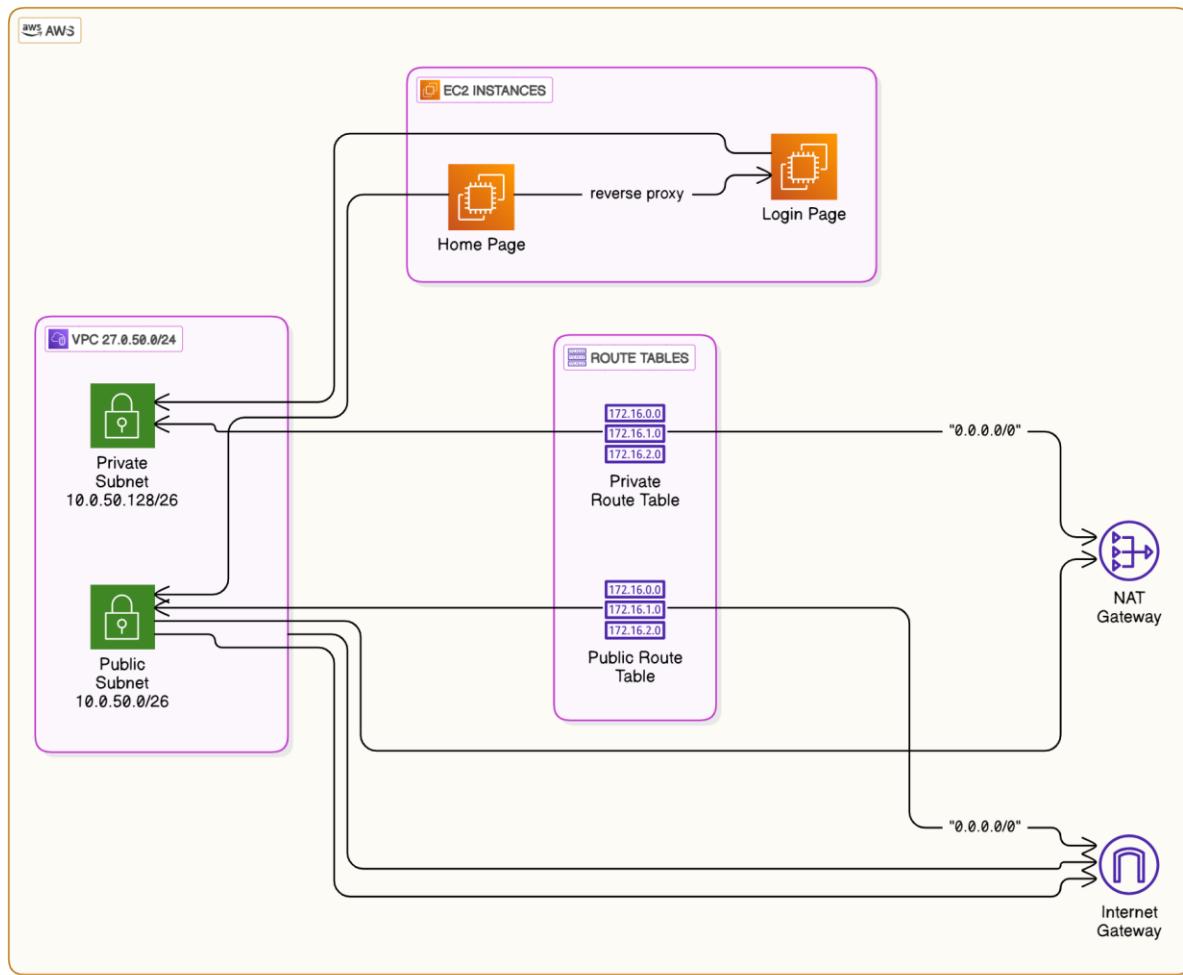
5. **NAT Gateway:** Enables instances in private subnets to access the internet for software updates or data retrieval while preventing inbound internet traffic, thus enhancing security.



6. **Security Groups:** Virtual firewalls associated with instances to control inbound and outbound traffic. Security groups ensure that only authorized traffic reaches resources in both public and private subnets.



## Architecture



## VPC Setup with Public and Private Subnets and EC2 Instances

### Step 1: Create VPC

1. Go to AWS Console → VPC → Your VPCs → Create VPC.
2. Name tag: MyVPC
3. IPv4 CIDR block: 27.0.50.0/24
4. Tenancy: Default
5. Click Create VPC.

AWS | Search [Alt+S] | United States (N. Virginia) ▾ | T MANOJ ▾

### VPC dashboard

EC2 Global View [ ] Filter by VPC: [ ]

**Create VPC** **Launch EC2 Instances**

Note: Your Instances will launch in the United States region.

#### Resources by Region

You are using the following Amazon VPC resources

VPCs	United States 1	NAT Gateways	United States 0
<a href="#">▶ See all regions</a>		<a href="#">▶ See all regions</a>	

Subnets	United States 11	VPC Peering Connections	United States 0
<a href="#">▶ See all regions</a>		<a href="#">▶ See all regions</a>	

Route Tables	United States 2	Network ACLs	United States 1
<a href="#">▶ See all regions</a>		<a href="#">▶ See all regions</a>	

Internet Gateways	United States 1	Security Groups	United States 6
<a href="#">▶ See all regions</a>		<a href="#">▶ See all regions</a>	

**Service Health**  
[View complete service health details \[ \]](#)

**Settings**  
[Block Public Access](#)  
[Zones](#)  
[Console Experiments](#)

**Additional Information**  
[VPC Documentation](#)  
[All VPC Resources](#)

CloudShell Feedback © 2025, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

AWS | Search [Alt+S] | United States (N. Virginia) ▾ | T MANOJ ▾

[VPC](#) > [Your VPCs](#) > Create VPC

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.

**IPv4 CIDR block** [Info](#)  
 IPv4 CIDR manual input  
 IPAM-allocated IPv4 CIDR block

**IPv4 CIDR**  
  
CIDR block size must be between /16 and /28.

**IPv6 CIDR block** [Info](#)  
 No IPv6 CIDR block.

CloudShell Feedback © 2025, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

The screenshot shows the 'Create VPC' wizard in the AWS VPC service. The 'Tenancy' dropdown is set to 'Default'. Under 'Tags', a single tag named 'MYVPC' is added. At the bottom, there are 'Cancel', 'Preview code', and 'Create VPC' buttons.

The screenshot shows the 'Details' page for the VPC 'vpc-0cfe6cabf6b6b9dd0 / MYVPC'. Key details include:

VPC ID	State	Block Public Access	DNS hostnames
vpc-0cfe6cabf6b6b9dd0	Available	Off	Disabled
DNS resolution	Tenancy	DHCP option set	Main route table
Enabled	default	dopt-0887c831259938960	rtb-02051ccf7856b178f
Main network ACL	Default VPC	IPv4 CIDR	IPv6 pool
acl-0fafcfbf41665939f	No	27.0.50.0/24	-
IPv6 CIDR (Network border group)	Network Address Usage metrics	Route 53 Resolver DNS Firewall rule groups	Owner ID
-	Disabled	-	491085415620

Below the table, tabs for 'Resource map', 'CIDRs', 'Flow logs', 'Tags', and 'Integrations' are visible.

## Step 2: Divide the Network into Public and Private Subnets

VPC CIDR Block: 27.0.50.0/24

- Public Subnet: 27.0.50.0/26
  - First IP: 27.0.50.1
  - Last IP: 27.0.50.62
  - Gateway IP: 27.0.50.63
- Private Subnet: 27.0.50.128/26
  - First IP: 27.0.50.129

- Last IP: 27.0.50.190
- Gateway IP: 27.0.50.191

### Step 3: Create Subnets

1. Go to Subnets → Create subnet.
2. Select VPC: MyVPC.
3. Subnet Name: Public-Subnet → CIDR block: 27.0.50.0/26 → AZ: Select any → Create.
4. Subnet Name: Private-Subnet → CIDR block: 27.0.50.128/26 → Create.

The screenshot shows the AWS VPC Create Subnet wizard with two main sections:

**VPC**

- VPC ID:** Associated subnets in this VPC. A dropdown menu shows "vpc-0cfe6cabf6b6b9dd0 (MYVPC)".
- Associated VPC CIDRs:** IPv4 CIDRs: 27.0.50.0/24.

**Subnet settings**

Specify the CIDR blocks and Availability Zone for the subnet.

**Subnet 1 of 1**

- Subnet name:** PVT\_SUB
- Availability Zone:** No preference
- IPv4 VPC CIDR block:** 27.0.50.0/24
- IPv4 subnet CIDR block:** 27.0.50.128/26 (64 IPs)
- Tags - optional:** Key: Name, Value: PVT\_SUB, Remove button.

VPC > Subnets > Create subnet

**Subnet 1 of 1**

**Subnet name**  
Create a tag with a key of 'Name' and a value that you specify.  
  
The name can be up to 256 characters long.

**Availability Zone** [Info](#)  
Choose the zone in which your subnet will reside, or let Amazon choose one for you.

**IPv4 VPC CIDR block** [Info](#)  
Choose the VPC's IPv4 CIDR block for the subnet. The subnet's IPv4 CIDR must lie within this block.

**IPv4 subnet CIDR block**  
 64 IPs

**Tags - optional**

© 2025, Amazon Web Services, Inc. or its affiliates. [Privacy](#) [Terms](#) [Cookie preferences](#)

AWS | Search [Alt+S] | United States (N. Virginia) | T MANOJ

**VPC dashboard**

**Subnets (2)** [Info](#) Last updated less than a minute ago

Name	Subnet ID	State	VPC
PUB_SUB	<a href="#">subnet-0dde49a8f0bb33c9b</a>	Available	<a href="#">vpc-0cfe6cabf6b6b9dd0   MYVPC</a>
PVT_SUB	<a href="#">subnet-08cb266551db1e61f</a>	Available	<a href="#">vpc-0cfe6cabf6b6b9dd0   MYVPC</a>

Select a subnet

© 2025, Amazon Web Services, Inc. or its affiliates. [Privacy](#) [Terms](#) [Cookie preferences](#)

#### Step 4: Create and Attach Internet Gateway (IGW)

1. Go to Internet Gateways → Create internet gateway.
2. Name tag: MyIGW → Create.
3. Select MyIGW → Actions → Attach to VPC → Select MyVPC → Attach.

Screenshot of the AWS VPC Internet Gateways creation page.

**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**     
You can add 49 more tags.

CloudShell Feedback © 2025, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

Screenshot of the AWS VPC Internet Gateways list page after creation.

**VPC dashboard** < **igw-0220cefaa66805f11**

The following internet gateway was created: igw-0220cefaa66805f11 - INTERNETGATEWAY. You can now attach to a VPC to enable the VPC to communicate with the internet.

**igw-0220cefaa66805f11 / INTERNETGATEWAY**

---

**Attach to VPC (igw-0220cefaa66805f11)** 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.

The screenshot shows the AWS VPC Internet Gateways page. The top navigation bar includes the AWS logo, search bar, and account information for United States (N. Virginia) and T MANOJ. The main content area displays the details of an Internet Gateway named "igw-0220cefaa66805f11". The "Details" section shows the Internet gateway ID (igw-0220cefaa66805f11), state (Attached), VPC ID (vpc-0cfe6cabf6b6b9dd0 | MYVPC), and owner (491085415620). A "Tags" section lists a single tag "Name: INTERNETGATEWAY". The left sidebar shows the VPC dashboard and a list of VPC components under "Virtual private cloud", including Internet gateways, which is currently selected. The bottom of the page includes standard AWS footer links.

## Step 5: Create Route Tables

1. Go to Route Tables → Create route table.
2. Name tag: Public-Route-Table → VPC: MyVPC → Create.
3. Name tag: Private-Route-Table → VPC: MyVPC → Create.

This screenshot is identical to the one above, showing the details of the same Internet Gateway "igw-0220cefaa66805f11". The "Details" section, tags, and sidebar are all the same. The bottom of the page includes standard AWS footer links.

AWS | Search [Alt+S] | United States (N. Virginia) | T MANOJ | Create route table

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.

PUB\_ROUTE

**VPC**  
The VPC to use for this route table.

vpc-0cfe6cabf6b6b9dd0 (MYVPC)

#### 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="button" value="X"/>	<input type="text" value="PUB_ROUTE"/> <input type="button" value="X"/> <input type="button" value="Remove"/>

Add new tag

You can add 49 more tags.

CloudShell Feedback © 2025, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

AWS | Search [Alt+S] | United States (N. Virginia) | T MANOJ | Create route table

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.

PVT\_ROUTE

**VPC**  
The VPC to use for this route table.

vpc-0cfe6cabf6b6b9dd0 (MYVPC)

#### 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="button" value="X"/>	<input type="text" value="PVT_ROUTE"/> <input type="button" value="X"/> <input type="button" value="Remove"/>

Add new tag

You can add 49 more tags.

CloudShell Feedback © 2025, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

The screenshot shows the AWS VPC Route Tables page. At the top, there is a success message: "Route table rtb-0f83e4e2edc5a6e59 | PVT\_ROUTE was created successfully." Below this, the title is "rtb-0f83e4e2edc5a6e59 / PVT\_ROUTE". On the left sidebar, under "Virtual private cloud", the "Route tables" option is selected. The main content area displays the "Details" tab for the route table. It shows the Route table ID (rtb-0f83e4e2edc5a6e59), which is Main and has No explicit subnet associations or edge associations. The VPC is associated with the owner ID 491085415620. Below this, the "Routes" tab is selected, showing one route entry: Destination 0.0.0.0/0, Target MyIGW, Status Active, and Propagated No. The bottom of the page includes standard AWS footer links like CloudShell, Feedback, Privacy, Terms, and Cookie preferences.

## Step 6: Configure Public Route Table

1. Select Public-Route-Table → Routes → Edit routes.
2. Add route: Destination: 0.0.0.0/0 → Target: Select Internet Gateway (MyIGW) → Save.

3. Subnet associations → Edit subnet associations → Select Public-Subnet → Save.

The screenshot shows the 'Edit routes' page for a specific route table. A route is listed for the destination 27.0.50.0/24, which is targeted to 'local'. The status is 'Active' and it is not propagated. Below this, there is a section for adding a new route with a placeholder '0.0.0.0/0'. At the bottom, there are buttons for 'Cancel', 'Preview', and 'Save changes'.

The screenshot shows the 'Edit subnet associations' page. It lists available subnets (PUB\_SUB and PVT\_SUB) and selected subnets (PUB\_SUB). The 'PUB\_SUB' checkbox is checked. At the bottom, there are buttons for 'Cancel' and 'Save associations'.

### Step 7: Configure Private Route Table

1. Select Private-Route-Table → Subnet associations → Select Private-Subnet → Save.

**Edit subnet associations**

Change which subnets are associated with this route table.

**Available subnets (1/2)**

Name	Subnet ID	IPv4 CIDR	IPv6 CIDR	Route table ID
PUB_SUB	subnet-0dde49a8f0bb33c9b	27.0.50.0/26	-	rtb-06cd344fe1f56b1d2 / PUB_RO...
PVT_SUB	subnet-08cb266551db1e61f	27.0.50.128/26	-	Main (rtb-02051ccf7856b178f)

**Selected subnets**

subnet-08cb266551db1e61f / PVT\_SUB X

**Buttons:** Cancel, Save associations

## Step 8: Create NAT Gateway for Private Subnet Internet Access

1. Go to Elastic IPs → Allocate Elastic IP → Allocate.
2. Go to NAT Gateways → Create NAT gateway.
3. Subnet: Public-Subnet → Elastic IP: Select the allocated EIP → Create NAT gateway.
4. Wait for the NAT gateway to become available.
5. Go back to Private-Route-Table → Routes → Edit routes.
  - a. Destination: 0.0.0.0/0 → Target: Select NAT Gateway → Save.

**NAT gateways** Info

**Create NAT gateway**

Name	NAT gateway ID	Connectivity...	State	State message	Primary
------	----------------	-----------------	-------	---------------	---------

aws | Search [Alt+S] | United States (N. Virginia) | T MANOJ

VPC > NAT gateways > Create NAT gateway

**Elastic IP address 3.218.94.6 (eipalloc-04bc8751ff0592a2e) allocated.**

**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](#)

[► Additional settings](#) [Info](#)

CloudShell Feedback © 2025, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

aws | Search [Alt+S] | United States (N. Virginia) | T MANOJ

VPC > Route tables > rtb-0f83e4e2edc5a6e59 > Edit routes

### Edit routes

Destination	Target	Status	Propagated
27.0.50.0/24	local	Active	No
<input type="text" value="0.0.0.0"/> <a href="#">X</a>	NAT Gateway	-	No
	<input type="text" value="nat-0bed03509fcf7a866"/> <a href="#">X</a>		<a href="#">Remove</a>

[Add route](#)

Cancel [Preview](#) [Save changes](#)

CloudShell Feedback © 2025, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

## Step 9: Configure Security Groups

- Public Security Group (Public-SG):
  - Inbound: All TCP, Port range: 0-65535, Source:  $0.0.0.0/0$
  - Outbound: All traffic,  $0.0.0.0/0$
- Private Security Group (Private-SG):
  - Inbound: All TCP, Source: Public-SG
  - Outbound: All traffic,  $0.0.0.0/0$

AWS | Search [Alt+S] | United States (N. Virginia) | T MANOJ

EC2 > Security Groups > Create security group

### Create security group Info

A security group acts as a virtual firewall for your instance to control inbound and outbound traffic. To create a new security group, complete the fields below.

**Basic details**

**Security group name Info**  
PUB-SG  
Name cannot be edited after creation.

**Description Info**  
SECURITY GROUP FOR PUBLIC USERS

**VPC Info**  
vpc-0cfe6cabf6b6b9dd0 (MYVPC)

**Inbound rules Info**

CloudShell Feedback © 2025, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

AWS | Search [Alt+S] | United States (N. Virginia) | T MANOJ

EC2 > Security Groups > Create security group

### Inbound rules Info

Type <small>Info</small>	Protocol <small>Info</small>	Port range <small>Info</small>	Source <small>Info</small>	Description - optional <small>Info</small>
All TCP	TCP	0 - 65535	A... <small>Info</small>	<input type="text" value="0.0.0.0/0"/> Delete
<a href="#">Add rule</a>				

⚠ Rules with source of 0.0.0.0/0 or ::/0 allow all IP addresses to access your instance. We recommend setting security group rules to allow access from known IP addresses only.

### Outbound rules Info

Type <small>Info</small>	Protocol <small>Info</small>	Port range <small>Info</small>	Destination <small>Info</small>	Description - optional <small>Info</small>

CloudShell Feedback © 2025, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

Search [Alt+S] United States (N. Virginia) ▾ T MANOJ ▾

EC2 > Security Groups > Create security group

## Create security group Info

A security group acts as a virtual firewall for your instance to control inbound and outbound traffic. To create a new security group, complete the fields below.

**Basic details**

**Security group name Info**  
PVT-SG  
Name cannot be edited after creation.

**Description Info**  
SECURITYGROUP FOR PRIVATE

**VPC Info**  
vpc-0c33a120e93d325f9

**Inbound rules Info**  
This security group has no inbound rules.

CloudShell Feedback © 2025, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

Search [Alt+S] United States (N. Virginia) ▾ T MANOJ ▾

EC2 > Security Groups > Create security group

## Create security group Info

A security group acts as a virtual firewall for your instance to control inbound and outbound traffic. To create a new security group, complete the fields below.

**Basic details**

**Security group name Info**  
PVT-SG  
Name cannot be edited after creation.

**Description Info**  
SECURITYGROUP FOR PRIVATE

**VPC Info**  
vpc-0cfe6cabf6b6b9dd0 (MYVPC)

**Inbound rules Info**  
This security group has no inbound rules.

CloudShell Feedback © 2025, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

The screenshot shows the AWS EC2 Security Groups interface. At the top, the navigation bar includes 'Search' and 'United States (N. Virginia)'. The main content area is titled 'Create security group' under 'Security Groups'. A dropdown menu shows 'vpc-0cfe6cabf6b6b9dd0 (MYVPC)'. The 'Inbound rules' section has a table with columns: Type, Protocol, Port range, Source, and Description - optional. One rule is listed: 'All TCP' on '0 - 65535' from 'sg-0e9499b5e3f3d553a' with the description 'ALLOWS FROM PUBLIC GROUP'. An 'Add rule' button is at the bottom. The 'Outbound rules' section is empty.

### Step 10: Launch EC2 Instances

1. Home Page Instance (Public EC2):
  - a. Go to EC2 → Launch instance.
  - b. Name: HomePageInstance
  - c. Network settings:
    - i. VPC: MyVPC
    - ii. Subnet: Public-Subnet
    - iii. Auto-assign Public IP: Enable
    - iv. Security group: Select Public-SG
  - d. Launch instance.
2. Login Page Instance (Private EC2):
  - a. Name: LoginPageInstance
  - b. VPC: MyVPC
  - c. Subnet: Private-Subnet
  - d. Auto-assign Public IP: Disable
  - e. Security group: Select Private-SG
  - f. Launch instance.

S | Search [Alt+S] | United States (N. Virginia) | T MANOJ

EC2 > Instances > Launch an instance

Login\_Page 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

Recents Quick Start

Amazon Linux macOS Ubuntu Windows Red Hat SUSE Linux

aws Mac ubuntu Microsoft Red Hat SUSE

Amazon Machine Image (AMI)

Software Image (AMI)  
Amazon Linux 2023 AMI 2023.6.2... [read more](#)  
ami-053a45ffff0a704a47

Virtual server type (instance type)  
t2.micro

Firewall (security group)  
New security group

Storage (volumes)

Cancel Launch instance Preview code

CloudShell Feedback © 2025, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

▼ 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*

RHLLINUX

Create new key pair

Screenshot of the AWS EC2 'Launch an instance' wizard.

**Step 1: Application and OS Images (Amazon Machine Image)**

The 'Amazon Machine Image (AMI)' section shows a grid of OS icons: Amazon Linux, macOS, Ubuntu, Windows, Red Hat, SUSE Linux, and a 'Browse more AMIs' link. A search bar at the top says 'Search our full catalog including 1000s of application and OS images'.

**Step 2: VPC - required**

The 'Subnet' dropdown is set to 'subnet-0dde49a8f0bb33c9b'. Below it, the 'Auto-assign public IP' dropdown is set to 'Disable'. The 'Firewall (security groups)' section shows a radio button for 'Select existing security group' (PVT-SG selected). The 'Common security groups' dropdown lists 'PVT-SG sg-0a13868a61c5e18fd'.

**Step 3: Summary**

The summary panel shows 1 instance being launched. It includes sections for 'Software Image (AMI)', 'Virtual server type (instance type) t2.micro', 'Firewall (security group) New security group', and 'Storage (volumes)'. The 'Launch instance' button is highlighted in orange.

Bottom navigation: CloudShell, Feedback, © 2025, Amazon Web Services, Inc. or its affiliates., Privacy, Terms, Cookie preferences.

The screenshot shows the AWS EC2 'Launch an instance' wizard. The first step, 'Name and tags', has a 'Name' field containing 'HomePage'. The second step, 'Application and OS Images (Amazon Machine Image)', includes a search bar and tabs for 'Recents' and 'Quick Start'. The third step, 'Summary', shows 1 instance being launched with the following details:

- Software Image (AMI)**: Provided by Red Hat, Inc. ami-0c7af5fe939f2677f
- Virtual server type (instance type)**: t2.micro
- Firewall (security group)**: New security group
- Storage (volumes)**: None

Buttons for 'Cancel', 'Launch instance', and 'Preview code' are visible.

The screenshot shows the 'Subnet' configuration step. A dropdown menu shows 'vpc-0fce6cabf6b6b9dd0 (MYVPC) 27.0.50.0/24'. The 'Subnet' section shows 'subnet-0dde49a8f0bb33c9b' (PUB\_SUB). The 'Auto-assign public IP' section is set to 'Enable'. The 'Firewall (security groups)' section shows 'Select existing security group' selected. The 'Common security groups' section lists 'PUB-SG sg-0e9499b5e3f3d553a' (VPC: vpc-0fce6cabf6b6b9dd0). Buttons for 'Create new subnet', 'Compare security group rules', and 'Cancel' or 'Launch instance' are present.

## Step 11: Connect Instances and Configure HTTPD

### 1. Connect to Public EC2:

```
ssh -i "/path/to/key.pem" ec2-user@<Public-Instance-Public-IP>
```

### 2. From Public EC2, SSH into Private EC2:

```
ssh -i "/path/to/key.pem" ec2-user@<Private-Instance-Private-IP>
```

3. Install HTTPD on Both Instances:

```
sudo yum update -y
sudo yum install httpd -y
sudo systemctl start httpd
sudo systemctl enable httpd
```

4. Add HTML Files:

- a. On Private Instance (/var/www/html/login.html):

```
<html><body><h2>Login Page</h2></body></html>
```

- b. On Public Instance (/var/www/html/index.html):

*Step 12: Configure Reverse Proxy on Public Instance*

1. Enable Proxy Modules:

```
sudo yum install mod_proxy -y
sudo yum install mod_proxy_http -y
```

2. Edit Apache Config:

```
sudo vi /etc/httpd/conf/httpd.conf

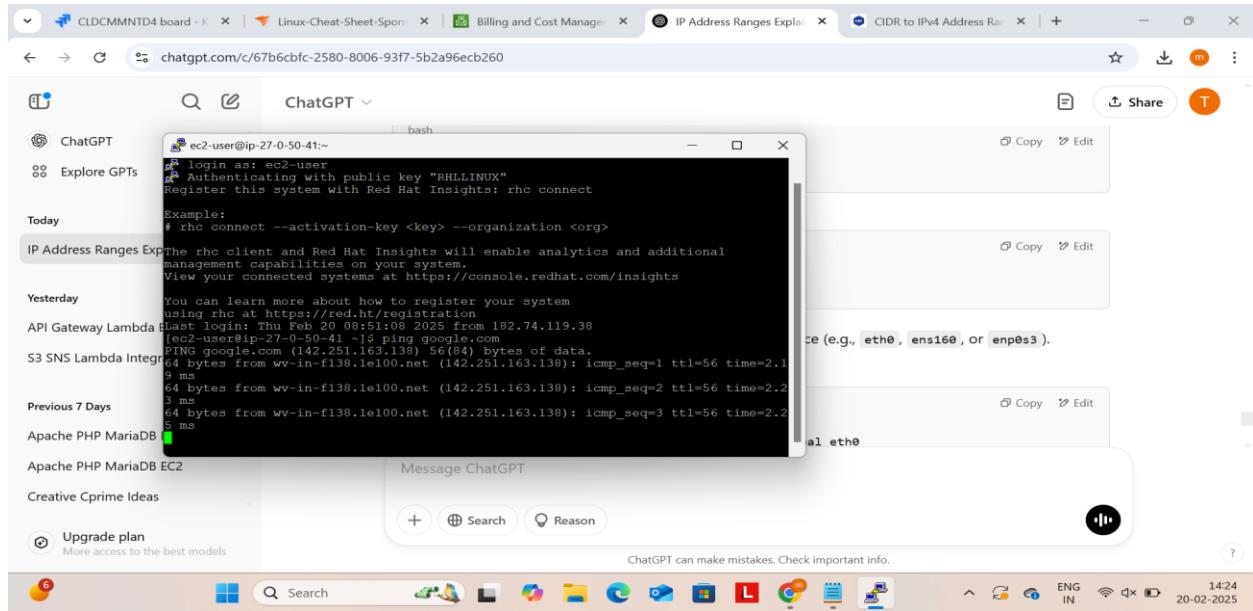
<VirtualHost *:80>
    ServerName PublicIP

    ProxyPreserveHost On
    ProxyPass '/login' 'http://privateIP/'
    ProxyPassReverse /login' 'http://privateIP/'

    ErrorLog /var/log/httpd/reverse-proxy-error.log
    CustomLog /var/log/httpd/reverse-proxy-access.log combined
</VirtualHost>
```

### 3. Restart HTTPD:

```
sudo systemctl restart httpd
```



## Advanced Site Settings

? X

### Environment

- Directories
- Recycle bin
- Encryption
- SFTP
- Shell

### Connection

- Proxy
- Tunnel

### SSH

- Key exchange
- Authentication

- Bugs

### Note

Bypass authentication entirely

#### Authentication options

Attempt authentication using Pageant

Attempt 'keyboard-interactive' authentication

Respond with a password to the first prompt

#### Authentication parameters

Allow agent forwarding

#### Private key file:

C:\Users\T Manoj\Downloads\RHLINUX.ppk



Display Public Key

Tools ▾

#### Certificate to use with the private key:



#### GSSAPI

Attempt GSSAPI authentication

Allow GSSAPI credential delegation

Color



OK

Cancel

Help

## Advanced Site Settings

?

X

### Environment

- Directories
- Recycle bin
- Encryption
- SFTP
- Shell

### Connection

- Proxy
- Tunnel

### SSH

- Key exchange
- Authentication**
- Bugs

### Note

Bypass authentication entirely

#### Authentication options

- Attempt authentication using Pageant
- Attempt 'keyboard-interactive' authentication
- Respond with a password to the first prompt

#### Authentication parameters

Allow agent forwarding

Private key file:

C:\Users\T Manoj\Downloads\RHLINUX.ppk



Display Public Key

Tools ▾

Certificate to use with the private key:



#### GSSAPI

Attempt GSSAPI authentication

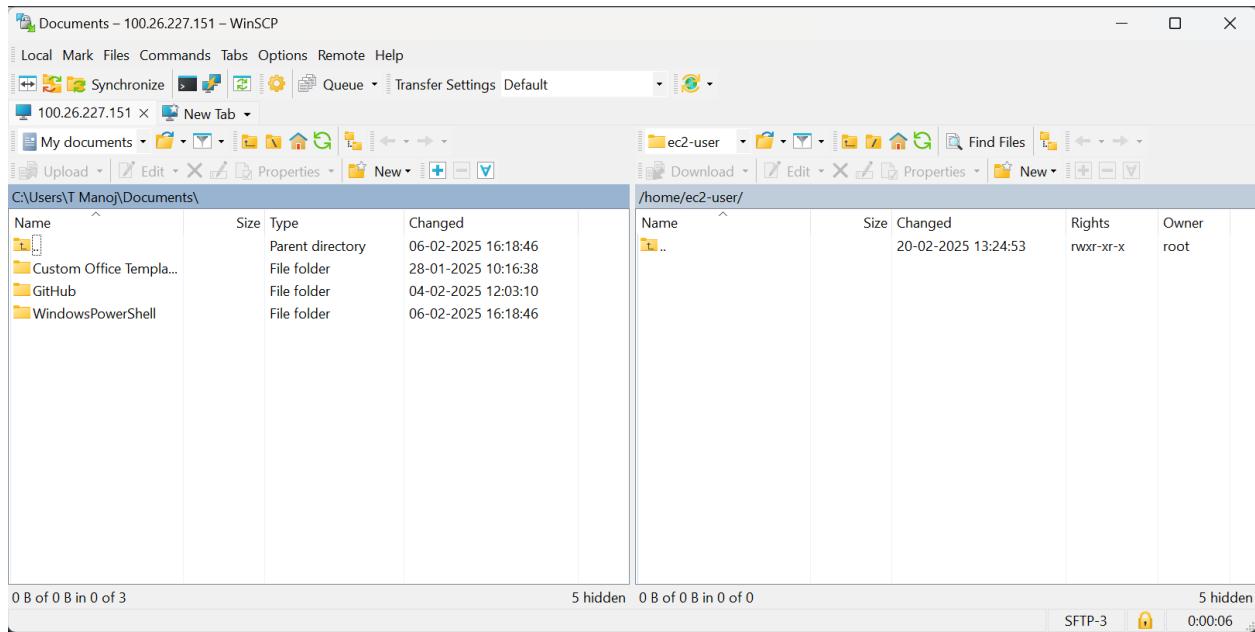
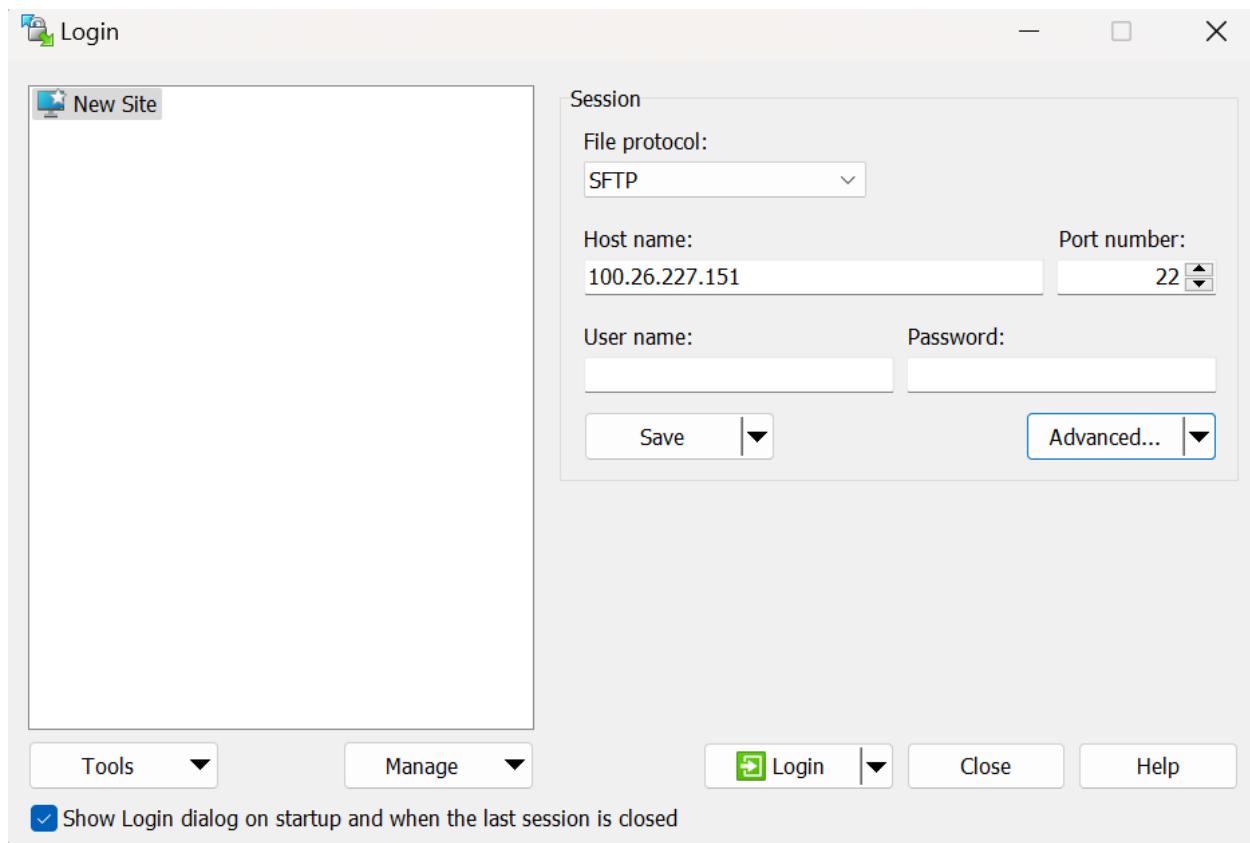
Allow GSSAPI credential delegation

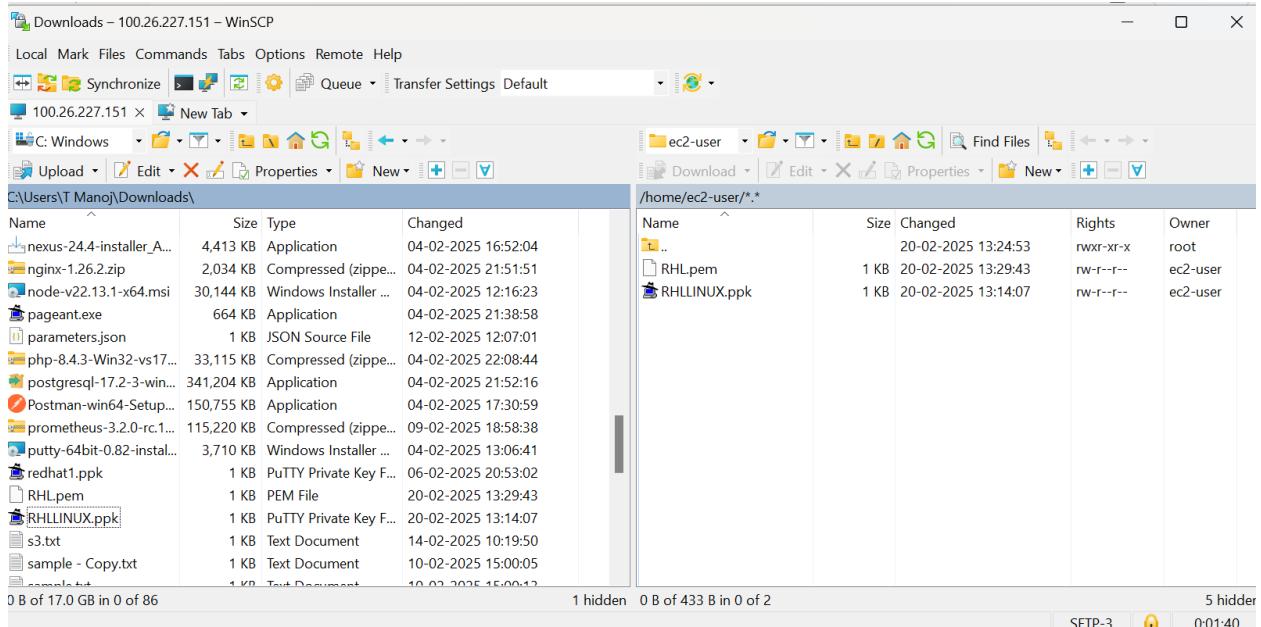
Color ▾

OK

Cancel

Help





```

-rw-rw-r--. 1 ec2-user ec2-user 1462 Feb 20 10:00 RSA.ppk
[ec2-user@ip-27-0-50-24 ~]$ ls
RSA.pem RSA.ppk
[ec2-user@ip-27-0-50-24 ~]$ sudo su -
[root@ip-27-0-50-24 ~]# ssh -i /home/ec2-user/RSA.pem ec2-user@27.0.50.163
The authenticity of host '27.0.50.163 (27.0.50.163)' can't be established.
ED25519 key fingerprint is SHA256:/2puSdghTlOo5vb906i4qdtd0Kvfxa0GtQXe2sxyUaA.
This key is not known by any other names
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added '27.0.50.163' (ED25519) to the list of known hosts.

  _#
  ~\_#####
  ~~\_\#####
  ~~ \#####
  ~~  \|/
  ~~ V~ '--->
  ~~  /
  ~~ ._/
  /`/-
  /m`-
[ec2-user@ip-27-0-50-163 ~]$ 

```

i-046162d54a6a240a3 (Homepages)  
Public IPs: 54.165.60.236 Private IPs: 27.0.50.24

```

/m/'
[ec2-user@ip-27-0-50-163 ~]$ ifconfig
enX0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 9001
        inet 27.0.50.163 netmask 255.255.255.192 broadcast 27.0.50.191
        inet6 fe80::47c:24ff:fe4e:8bd3 prefixlen 64 scopeid 0x20<link>
          ether 06:7c:24:4e:8b:d3 txqueuelen 1000 (Ethernet)
            RX packets 875 bytes 117059 (114.3 KiB)
            RX errors 0 dropped 0 overruns 0 frame 0
            TX packets 1076 bytes 119761 (116.9 KiB)
            TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

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

[ec2-user@ip-27-0-50-163 ~]$ 

```

```
TX packets 1076 bytes 119761 (116.9 KiB)
TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

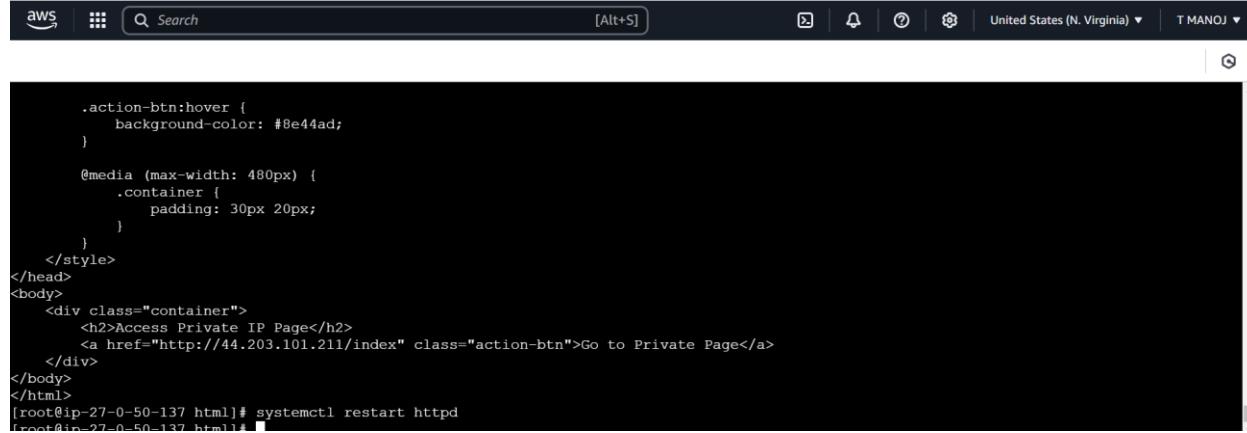
lo: flags=73<UP,LOOPBACK,RUNNING> mtu 65536
    inet 127.0.0.1 netmask 255.0.0.0
        inet6 ::1 prefixlen 128 scopeid 0x10<host>
            loop txqueuelen 1000 (Local Loopback)
            RX packets 12 bytes 1020 (1020.0 B)
            RX errors 0 dropped 0 overruns 0 frame 0
            TX packets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

[ec2-user@ip-27-0-50-163 ~]$ ping google.com
PING google.com (172.253.115.113) 56(84) bytes of data.
64 bytes from bg-in-f113.1e100.net (172.253.115.113): icmp_seq=1 ttl=103 time=2.65 ms
64 bytes from bg-in-f113.1e100.net (172.253.115.113): icmp_seq=2 ttl=103 time=2.23 ms
64 bytes from bg-in-f113.1e100.net (172.253.115.113): icmp_seq=3 ttl=103 time=5.00 ms
64 bytes from bg-in-f113.1e100.net (172.253.115.113): icmp_seq=4 ttl=103 time=2.26 ms
64 bytes from bg-in-f113.1e100.net (172.253.115.113): icmp_seq=5 ttl=103 time=2.26 ms
64 bytes from bg-in-f113.1e100.net (172.253.115.113): icmp_seq=6 ttl=103 time=2.27 ms
```

```
<head>
    <meta charset="UTF-8">
    <title>Home Page</title>
    <style>
        body { text-align: center; font-family: Arial, sans-serif; }
        button { padding: 10px 20px; font-size: 18px; border-radius: 10px; }
    </style>
</head>
<body>
    <h1>Welcome to the Home Page</h1>
    <p>Click the button to go to the Login Page.</p>
    <button onclick="window.location.href='http://27.0.50.163/index.html'">Go to Login</button>
</body>
</html>
[root@ip-27-0-50-24 ~]# ls
index.html
[root@ip-27-0-50-24 ~]# mv index.html /var/www/html
[root@ip-27-0-50-24 ~]# cd /var/www/html
[root@ip-27-0-50-24 html]# ls
index.html
[root@ip-27-0-50-24 html]#
```

### i-046162d54a6a240a3 (Homepages)

PublicIPs: 54.165.60.236 PrivateIPs: 27.0.50.24



A screenshot of a terminal window titled "aws" showing the creation of a static website. The terminal output includes the creation of an "index.html" file, its move to the "/var/www/html" directory, and its subsequent listing there. It also shows the command to restart the httpd service.

```
.action-btn:hover {
    background-color: #8e44ad;
}

@media (max-width: 480px) {
    .container {
        padding: 30px 20px;
    }
}
</style>
</head>
<body>
    <div class="container">
        <h2>Access Private IP Page</h2>
        <a href="http://44.203.101.211/index" class="action-btn">Go to Private Page</a>
    </div>
</body>
</html>
[root@ip-27-0-50-137 html]# systemctl restart httpd
[root@ip-27-0-50-137 html]#
```

```
GNU nano 5.8                               /etc/httpd/conf.d/reverse-proxy.conf
<VirtualHost *:80>
    ServerName 44.203.101.211
    ProxyPreserveHost On
    ProxyPass "/index" "http://27.0.50.4/"
    ProxyPassReverse "/index" "http://27.0.50.4/"

    ErrorLog /var/log/httpd/reverse-proxy-error.log
    CustomLog /var/log/httpd/reverse-proxy-access.log combined
</VirtualHost>
```

^G Help [O Write Out [W Where Is [C Cut [ Read 9 lines ]
^X Exit [R Read File [R Replace [V Paste ^T Execute [G Location M-1 Undo
 [J Justify [Y Go To Line M-2 Redo
 [A Set Mark M-Q Copy [Q Where Was
 [C Cut [V Paste
 [D Delete [B Backspace [H Home
 [F Forward [P Page Down [K End
 [L Page Up [B Backspace [H Home

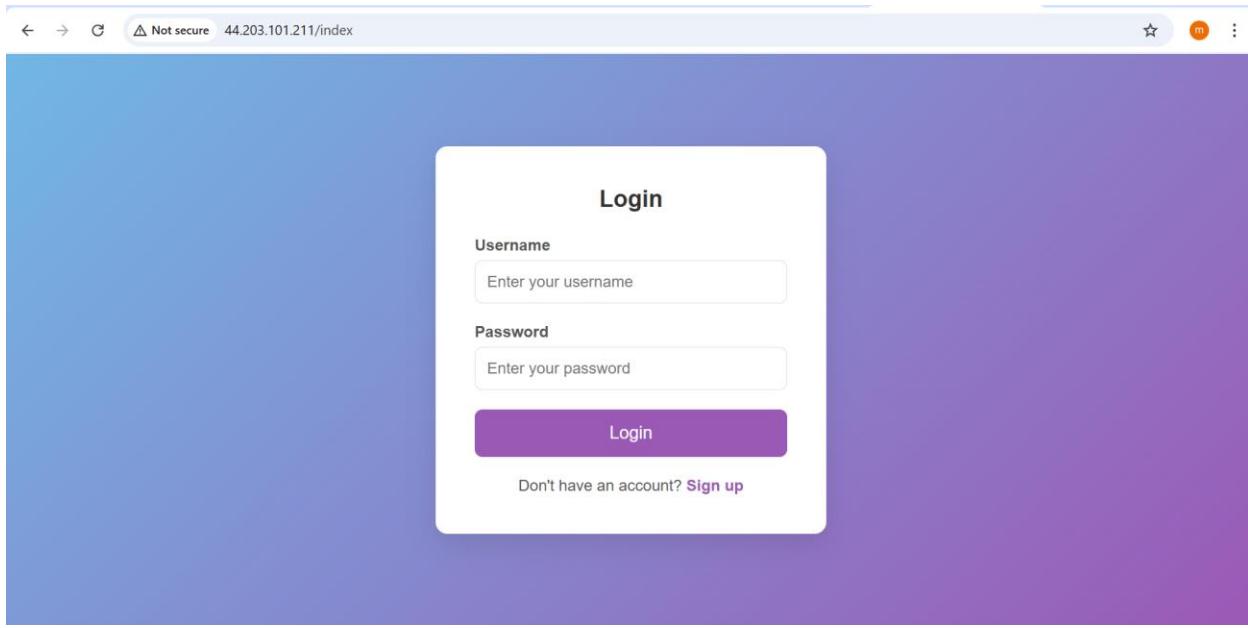
i-0060fec5f16461b16 (Homepage)

PublicIPs: 44.203.101.211 PrivateIPs: 27.0.50.137

### Step 13: Final Testing

- Open browser: <http://<Public-Instance-Public-IP>>
- Click Login button. It should load the login page from the private instance via reverse proxy.





## ***Advantages***

The primary advantage of this setup lies in its robust security model. By separating resources into public and private subnets, sensitive data and applications in the private subnet remain shielded from direct internet exposure. The use of a NAT Gateway provides controlled outbound internet access for private instances, ensuring updates without compromising security. Implementing a reverse proxy on the public instance enhances accessibility and reduces the attack surface by routing requests securely. Additionally, this architecture ensures scalability, allowing businesses to expand their applications seamlessly while maintaining a secure and efficient network infrastructure.

## ***Conclusion***

In conclusion, this VPC setup demonstrates best practices in building a secure, scalable, and highly available AWS environment. By effectively leveraging AWS components such as VPC, subnets, Internet and NAT Gateways, security groups, and EC2 instances, the project achieves a functional web application infrastructure. The reverse proxy configuration ensures that sensitive backend services remain protected while still being accessible when needed. This architecture is ideal for organizations seeking a balanced approach between

accessibility and security in their cloud deployments, making it suitable for web applications, microservices, and database-driven solutions.