# Placement Empowerment Program

***Cloud Computing and DevOps Centre***

Implement DNS for Your Application: Set up a DNS record to map your web application’s IP or load balancer to a domain name.

Name:Shahana.M.S Department:ADS



**Introduction**

Domain Name System (DNS) is a crucial component of the internet that translates human-friendly domain names (e.g., [www.example.com](http://www.example.com)) into machine-readable IP addresses. Implementing DNS for your application ensures seamless access by mapping your web application's IP address or load balancer to a domain name. This makes it easier for users to access your application without needing to remember complex numerical IP addresses.

**Overview**

Setting up a DNS record for your web application involves configuring records in a DNS hosting service, such as AWS Route 53, Google Cloud DNS, or traditional domain registrars. The primary goal is to associate your application's public IP address or load balancer with a domain name, improving accessibility, scalability, and reliability. Different DNS record types, such as A (Address) records, CNAME (Canonical Name) records, and ALIAS records, help in directing traffic efficiently.

**Objective**

* Configure a DNS record to map a web application’s IP address or load balancer to a domain name.
* Ensure seamless access to the application by users through a friendly URL.
* Improve application performance and security through managed DNS solutions.
* Reduce downtime and enhance failover capabilities using load balancing and redundancy.

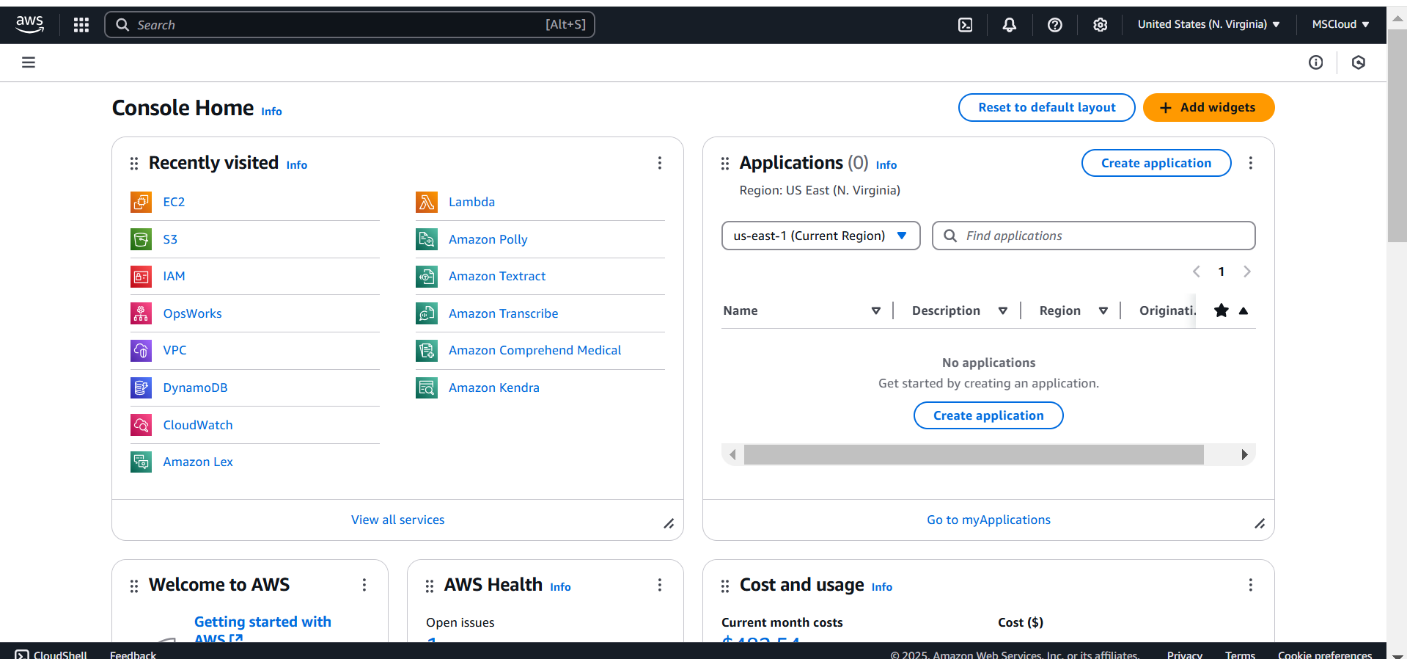
**Importance**

* **Improved User Experience:** Users can easily access the application using a domain name rather than a numerical IP.
* **Scalability:** Mapping a load balancer to a domain name enables the distribution of traffic across multiple servers.
* **Reliability:** DNS ensures that if an IP address changes, users can still reach the application without disruptions.
* **Security:** DNS management tools provide features like DDoS protection, traffic filtering, and domain authentication.
* **Branding and Credibility:** A custom domain name establishes a professional identity for businesses and applications.

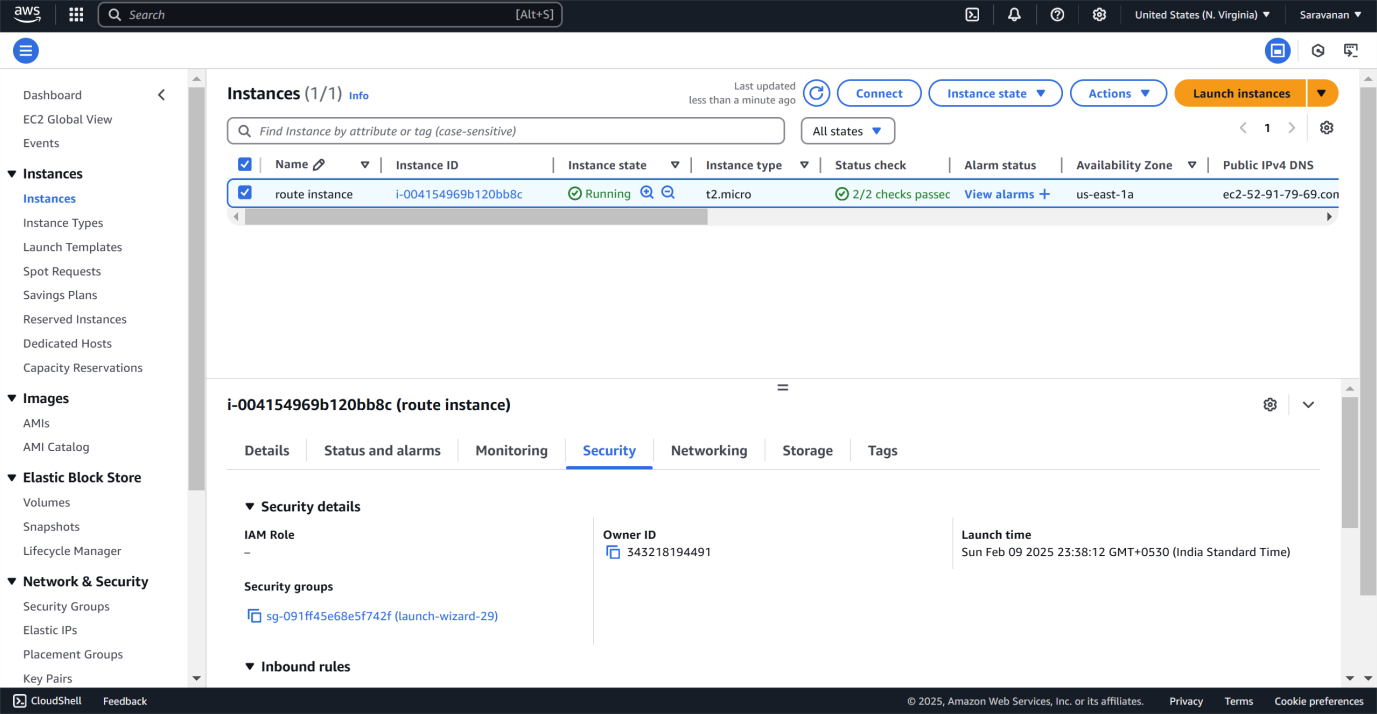
# Step-by-Step Overview

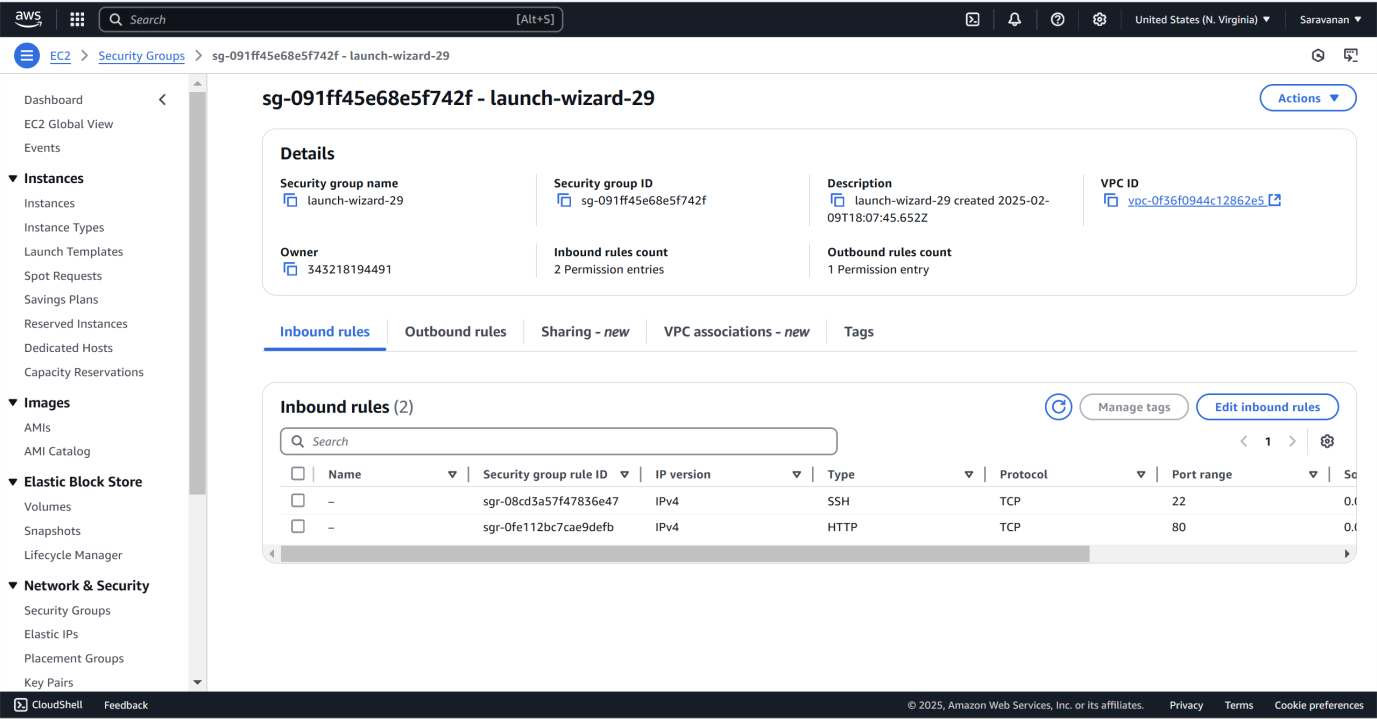
## Step 1:

1. Go to [AWS Management Console](https://aws.amazon.com/console/).
2. Enter your username and password to log in.



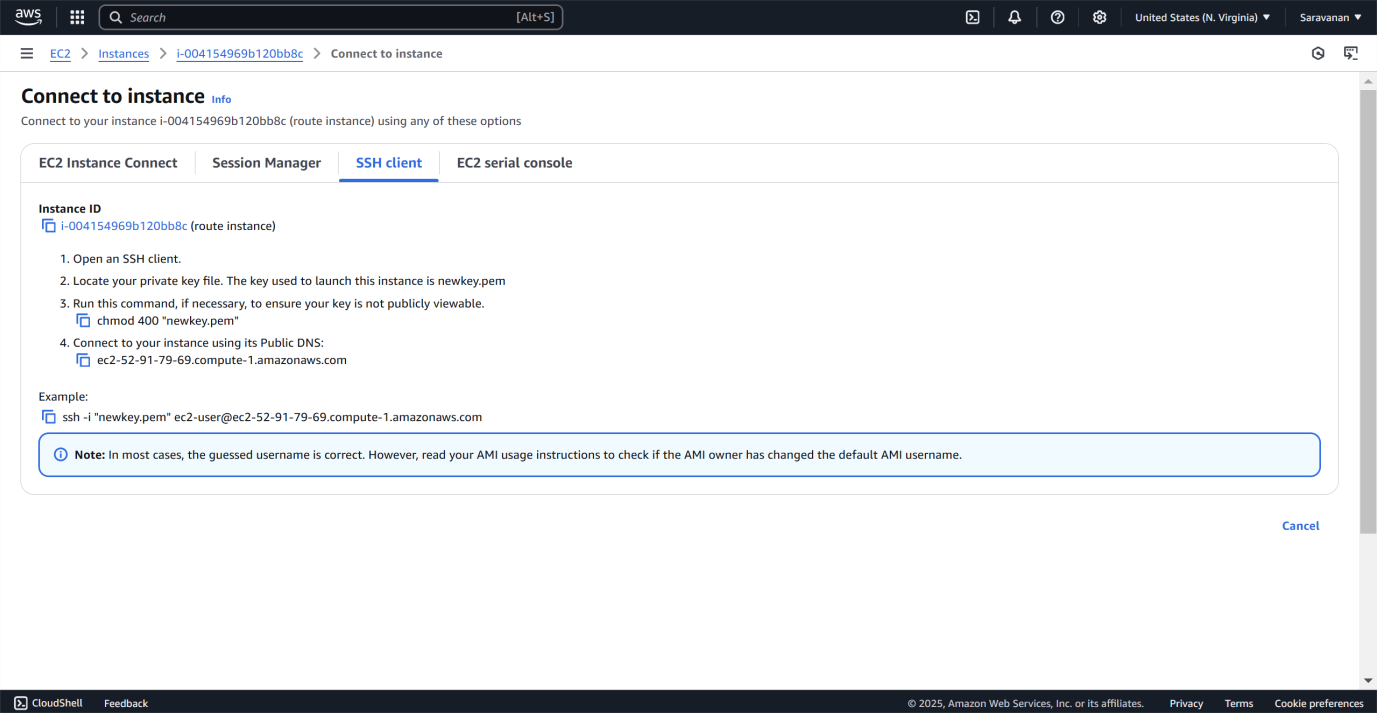
## Step 2:

Launch an instance and then configure SSH and HTTP whre you set 80 and 22

****

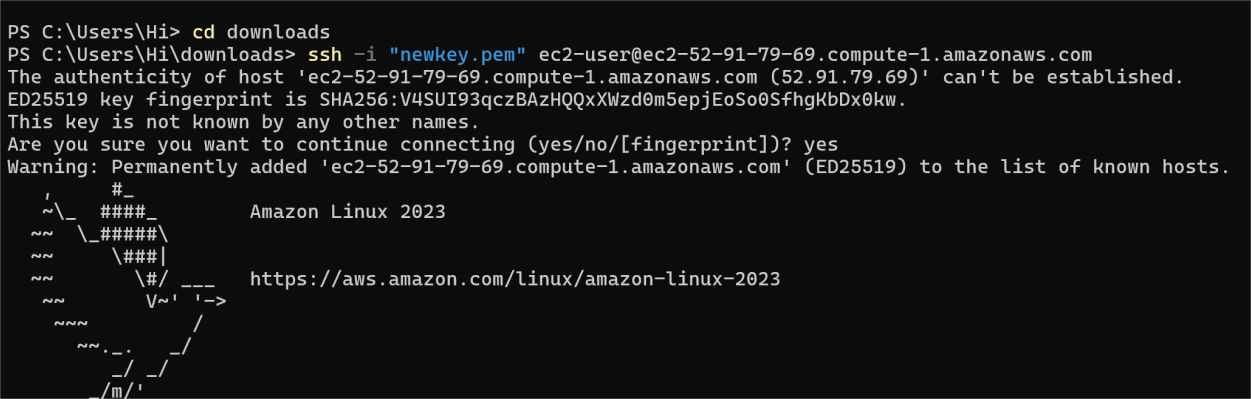
## Step 3:

Click the 'Connect' option on your launched instance, go to the SSH client section, and copy the command provided under the 'Example' section.



## Step 4:

Open PowerShell, and then use cd in order to give access into “Downloads” and paste SSH client code and then give ‘Y’ for confirmity



## Step 5:

Install Apache :

**sudo yum install httpd -y**

****

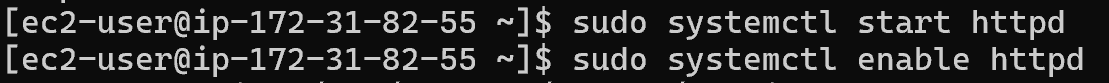
## Step 6:

Start Apache**:**

### sudo systemctl start httpd

Make Apache start on boot**:**

**sudo systemctl enable httpd**

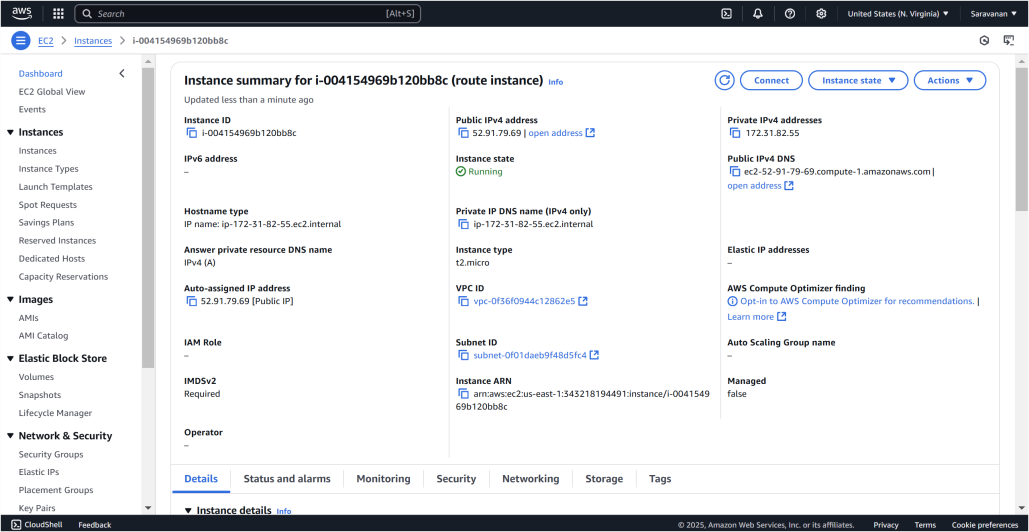
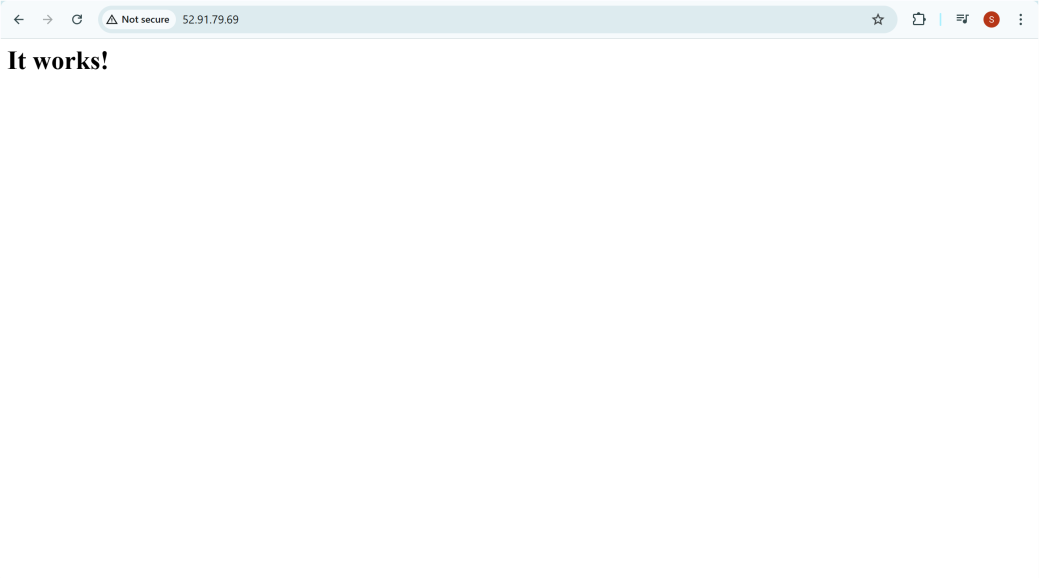
****

## Step 7:

### Verify Apache is running:

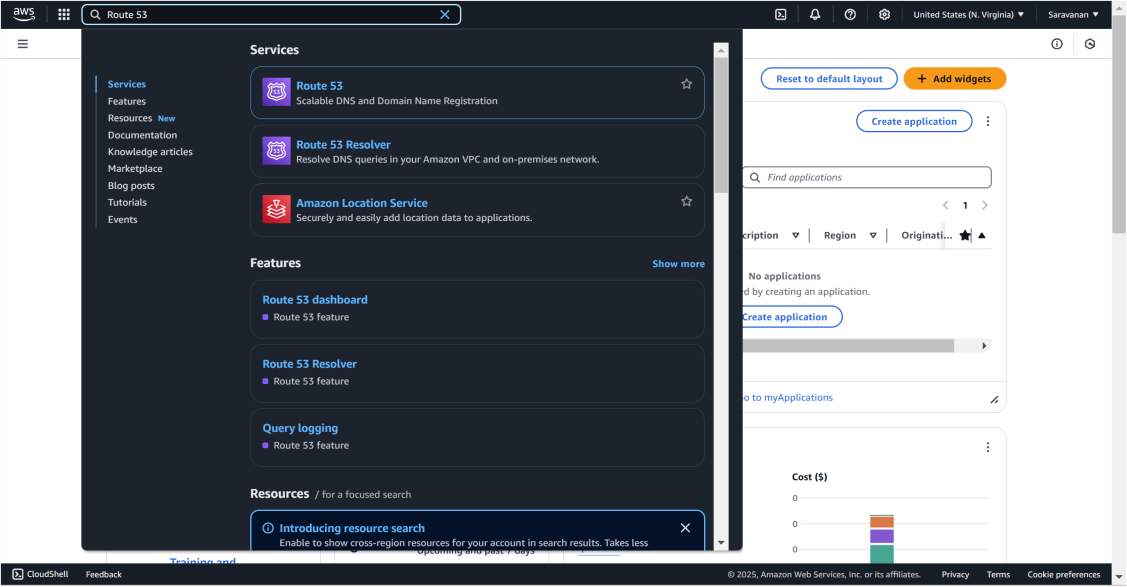
In your browser, enter the **EC2 public IP** (e.g., http://<your-ec2- public-ip>).

You should see the **Apache default page**. This means your EC2 instance is set up to serve websites.

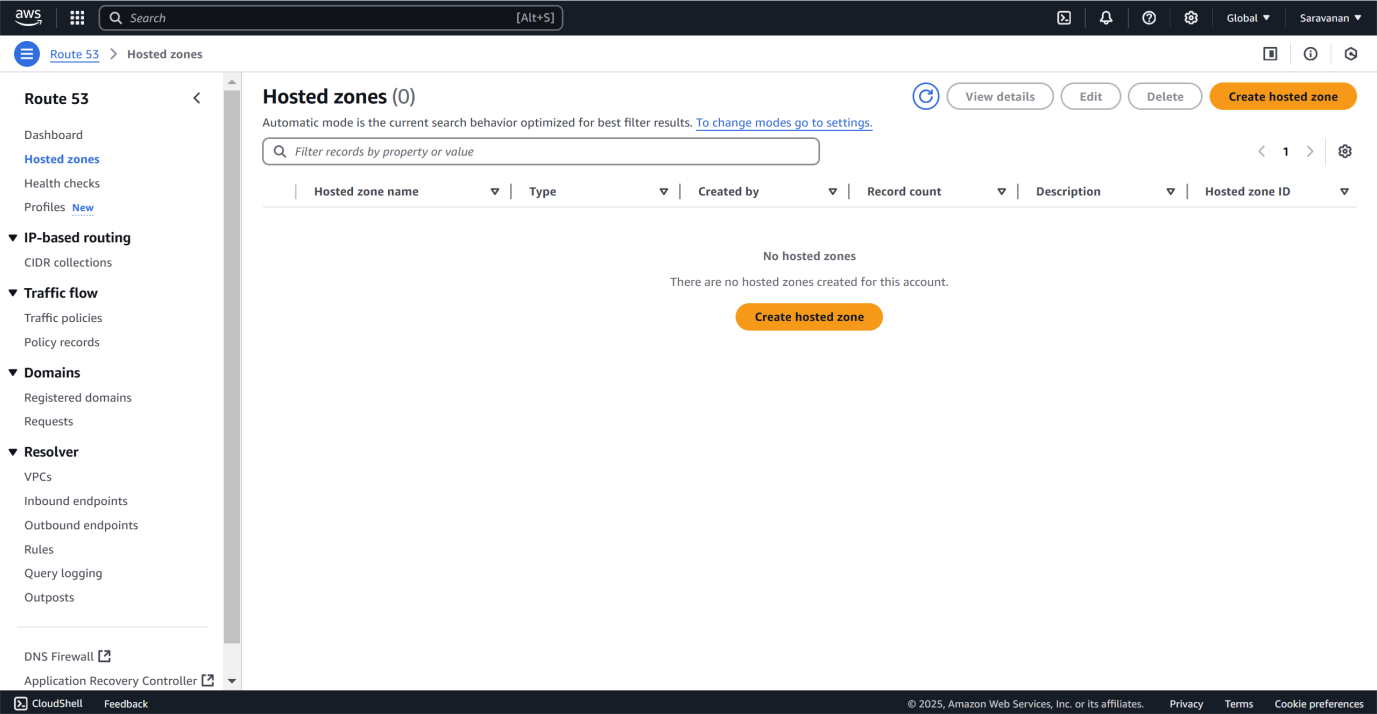


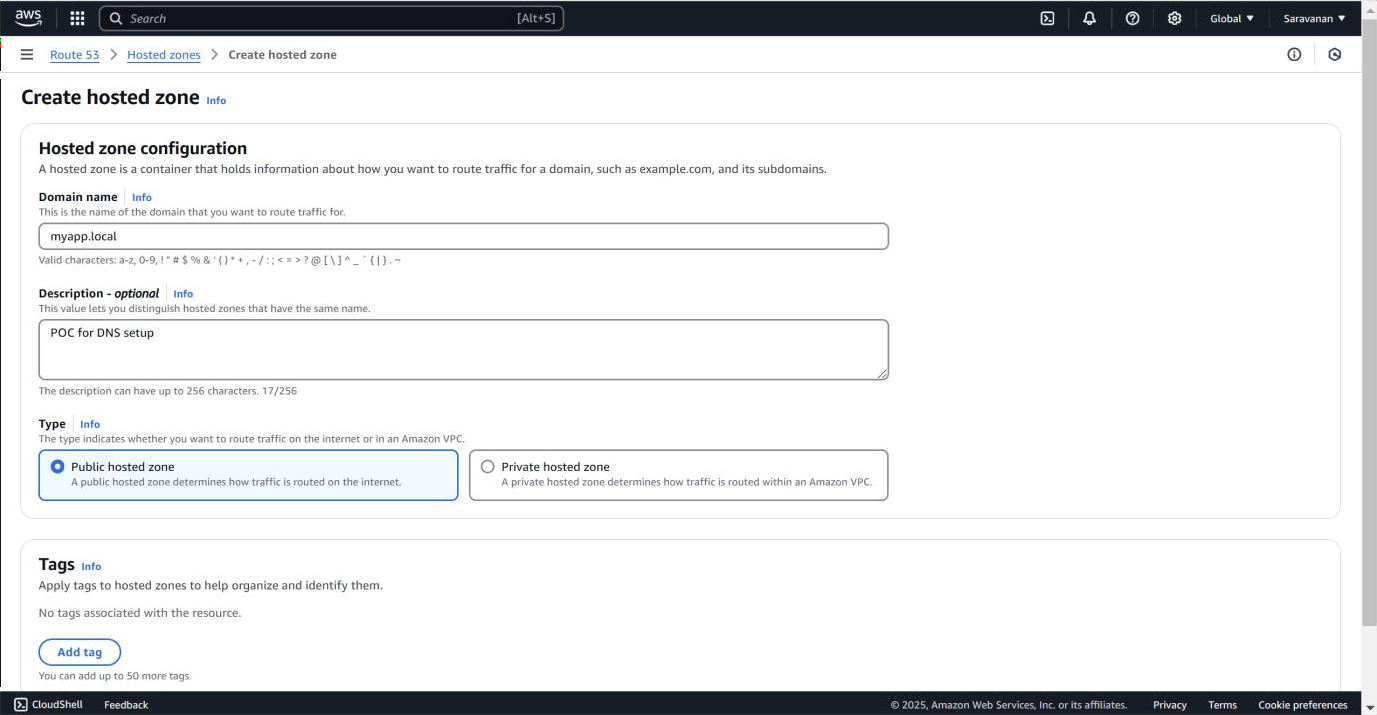
## Step 8:

In the AWS Console, search for **Route 53** and select it.( t is designed to route end-user requests to internet applications efficiently while providing robust security, availability, and performance features.)



## Step 9:

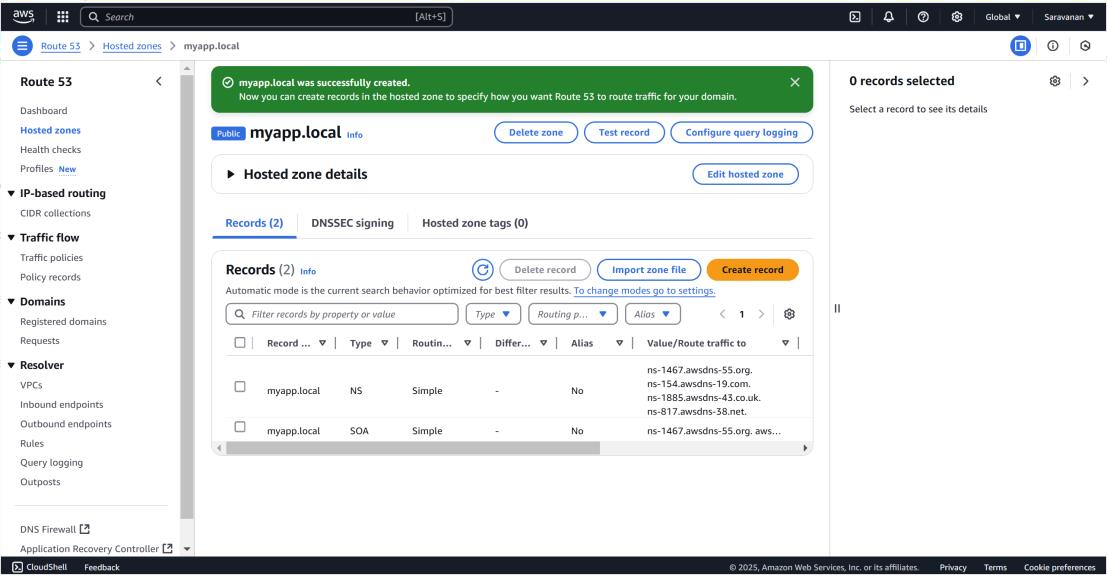
on the left choose host zone and create hosted zone with a name.

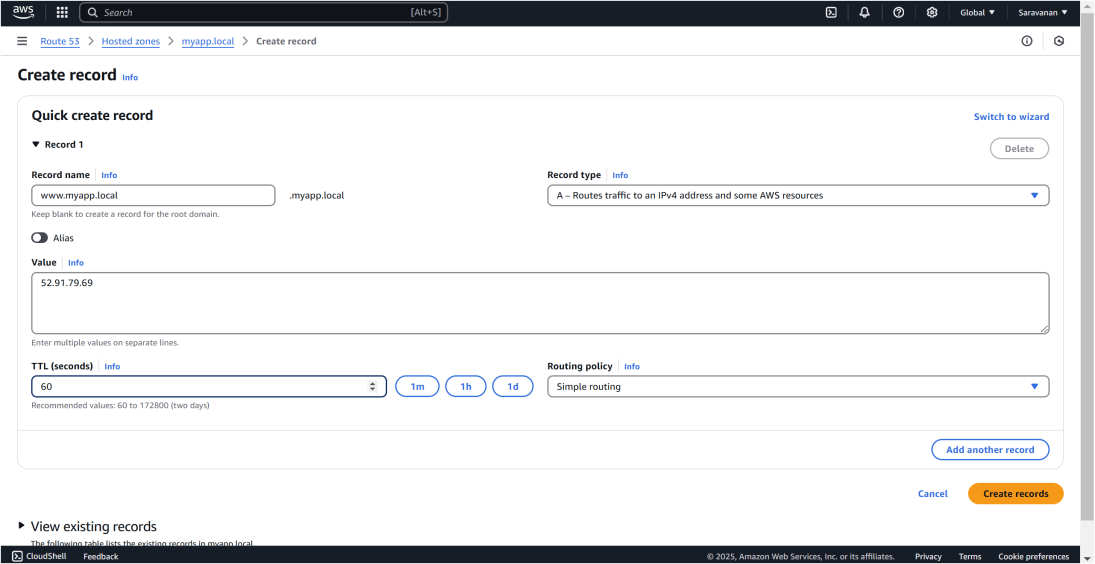


Click on public or private hosted zones in the type and click create hosted zones

## Step 10:

In your hosted zone, click **Create Record**. Click on IPV4 and set TTl to 60 seconds.

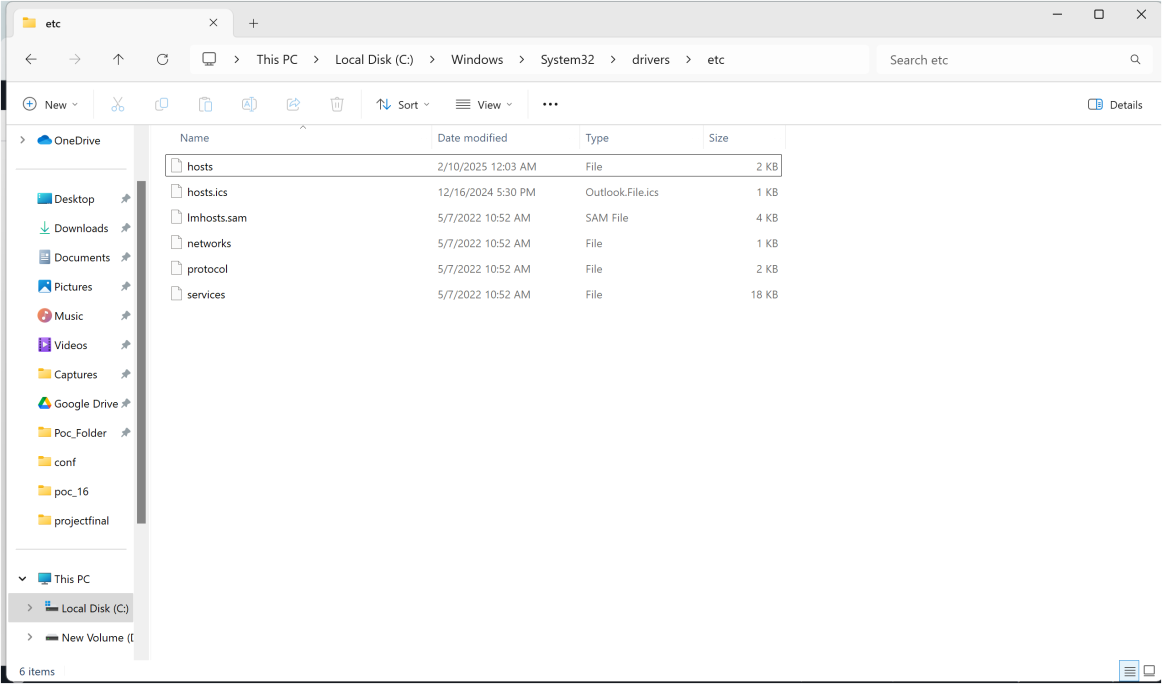




## Step 11:

Go to **FileExplorer > Open**, go to “C:\Windows\System32\drivers\etc.´ by using command cd a

In the file name field, type hosts and press **Enter**.



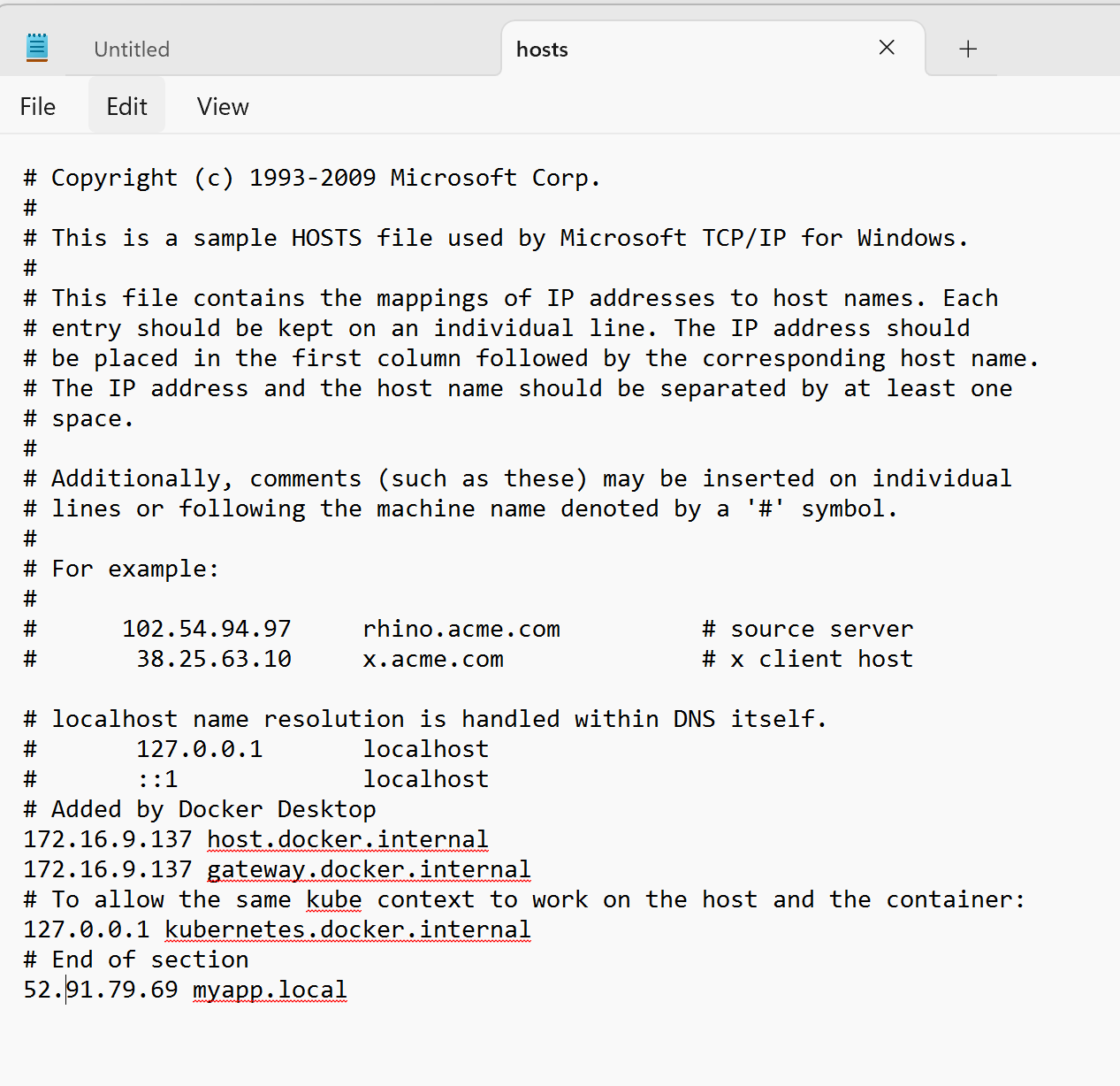
## Step 12:

* 1. At the bottom of the file, add:

<Your EC2 Public IP> myapp.local

Replace <Your EC2 Public IP> with the public IP you copied. (Eg: 52.91.79.69 myapp.local)

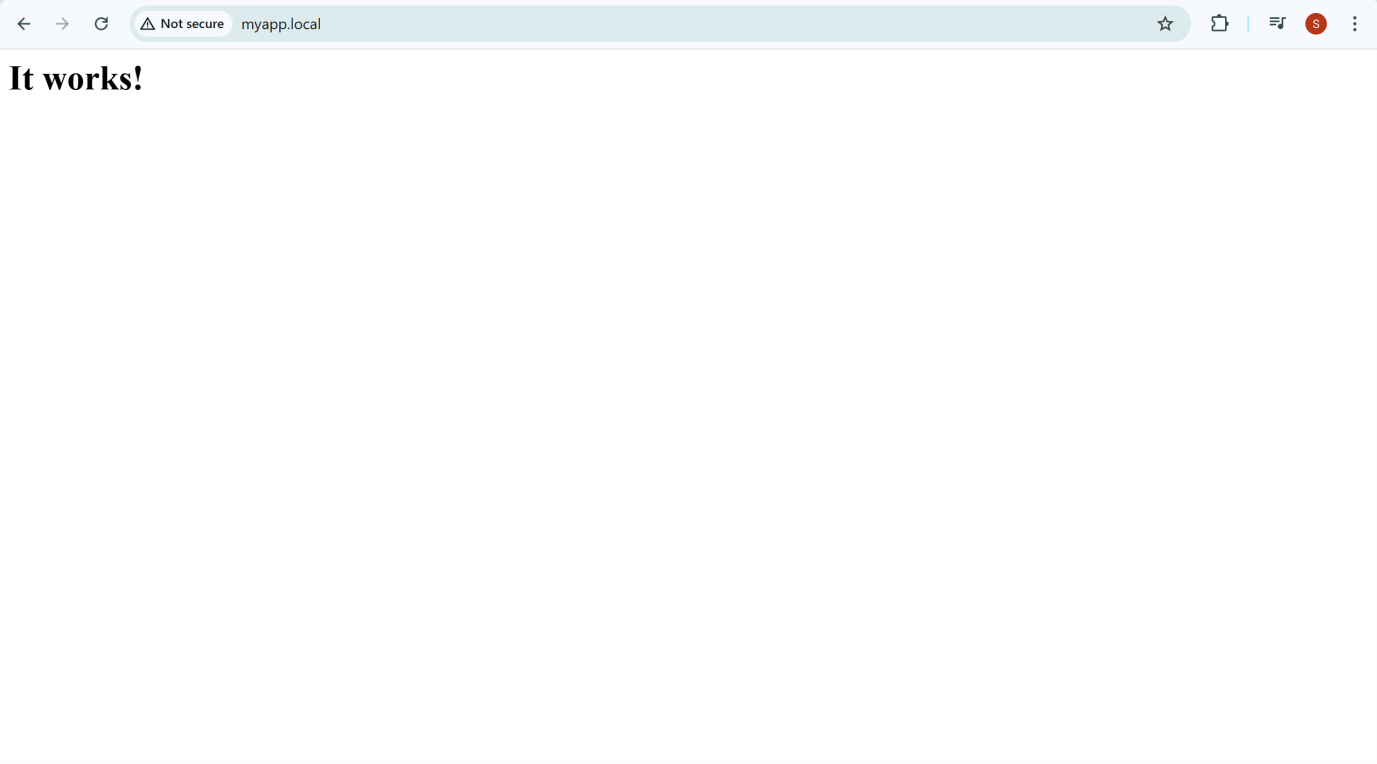
* 1. Save the file and close Notepad.



## Step 12:

Open your **web browser**.

Type myapp.local in the address bar and press **Enter**. You should see the Apache default page



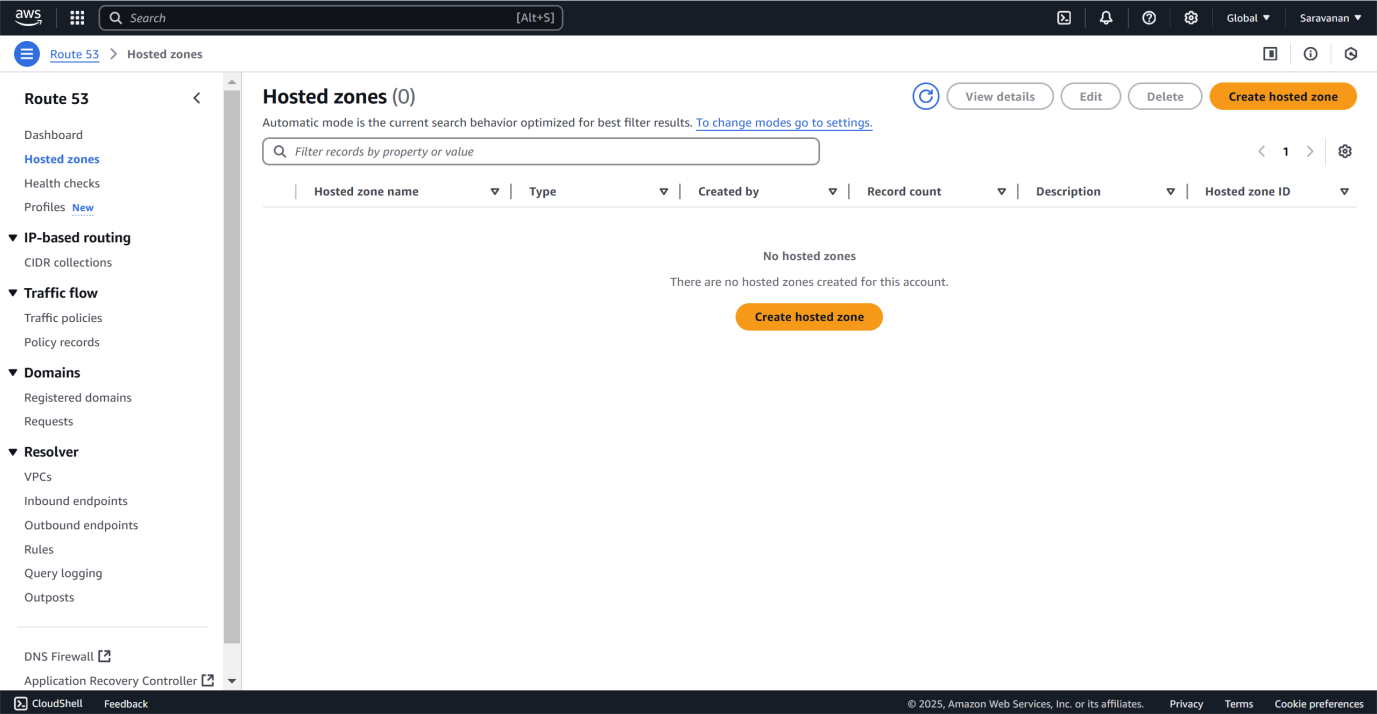
**Outcome**

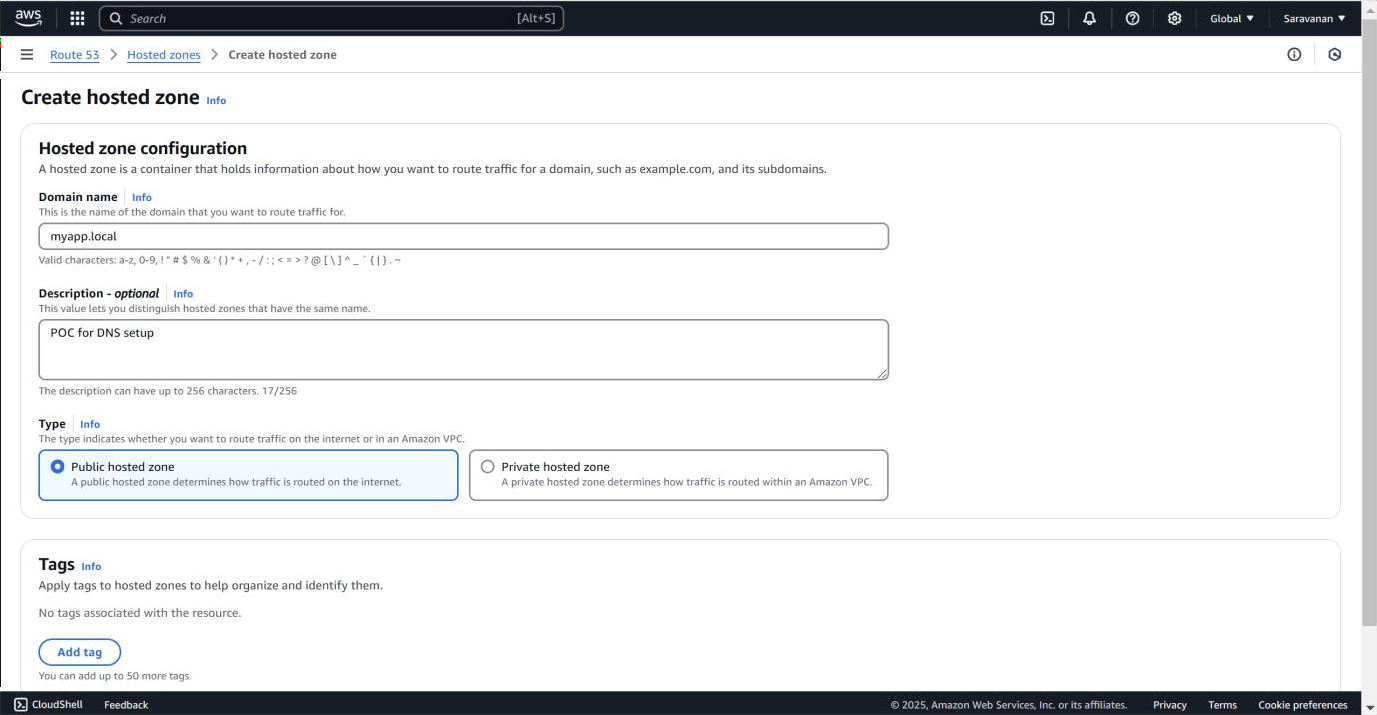
* Successfully set up DNS records that map the application’s IP or load balancer to a domain.
* Improved accessibility and performance of the web application.
* Enhanced traffic management and failover capabilities.
* A scalable and secure infrastructure that adapts to user demand and future application growth.

Would you like guidance on specific DNS services such a

## Step 9:

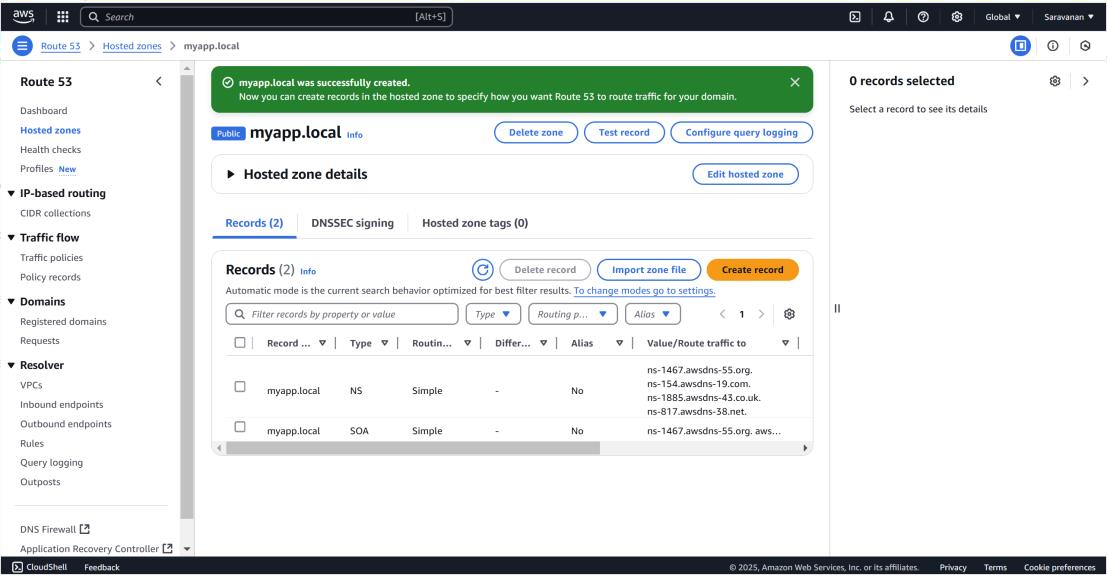
1. Click on **Create hosted zone**.
2. Enter a **Domain Name** (e.g., myapp.local).
3. Set the **Type** to **Public Hosted Zone**.
4. Click **Create hosted zone**.

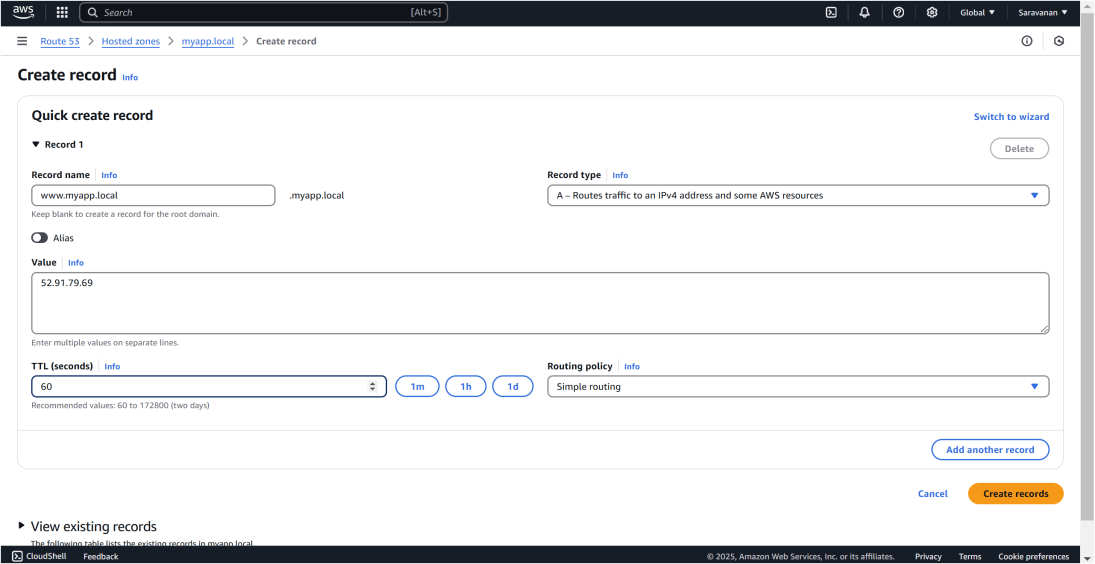




## Step 10:

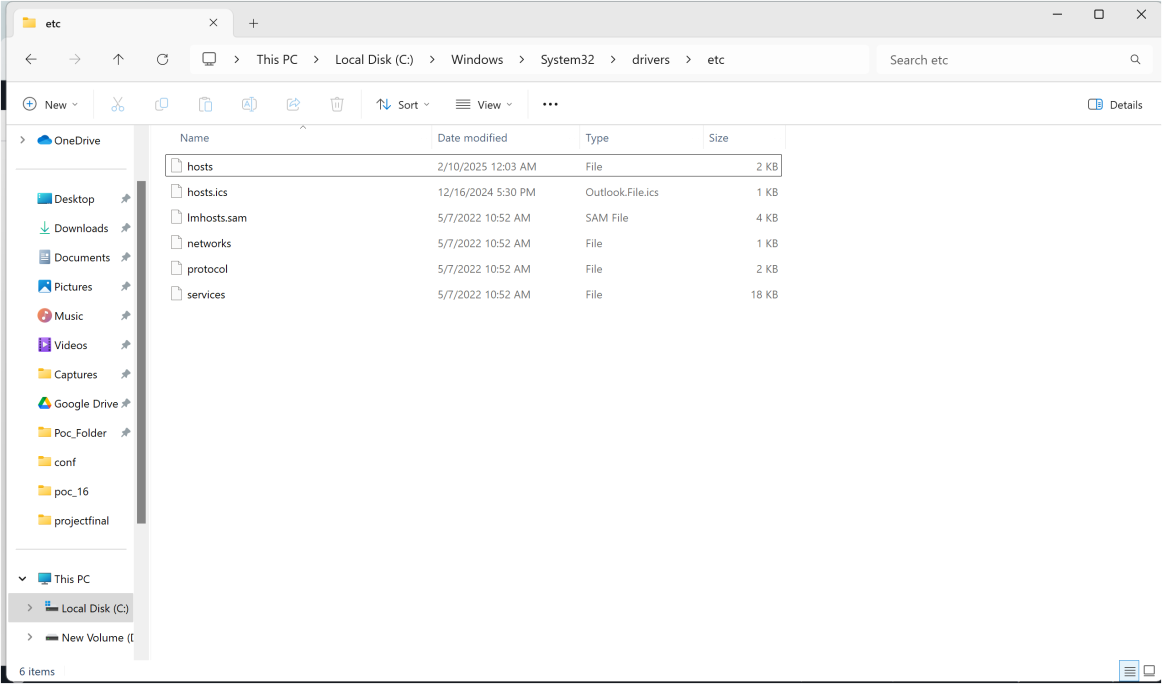
1. In your hosted zone, click **Create Record**.
2. **Record Name**: Leave it empty for the root domain (myapp.local),
3. **Record Type**: Select **A – IPv4 address**.
4. **Value**: Enter the **Public IP** of your EC2 instance.
5. **TTL**: Set to 60 seconds.
6. Click **Create records**.





## Step 11:

1. Go to **FileExplorer > Open**.
2. Navigate to: C:\Windows\System32\drivers\etc.
3. In the file name field, type hosts and press **Enter**.



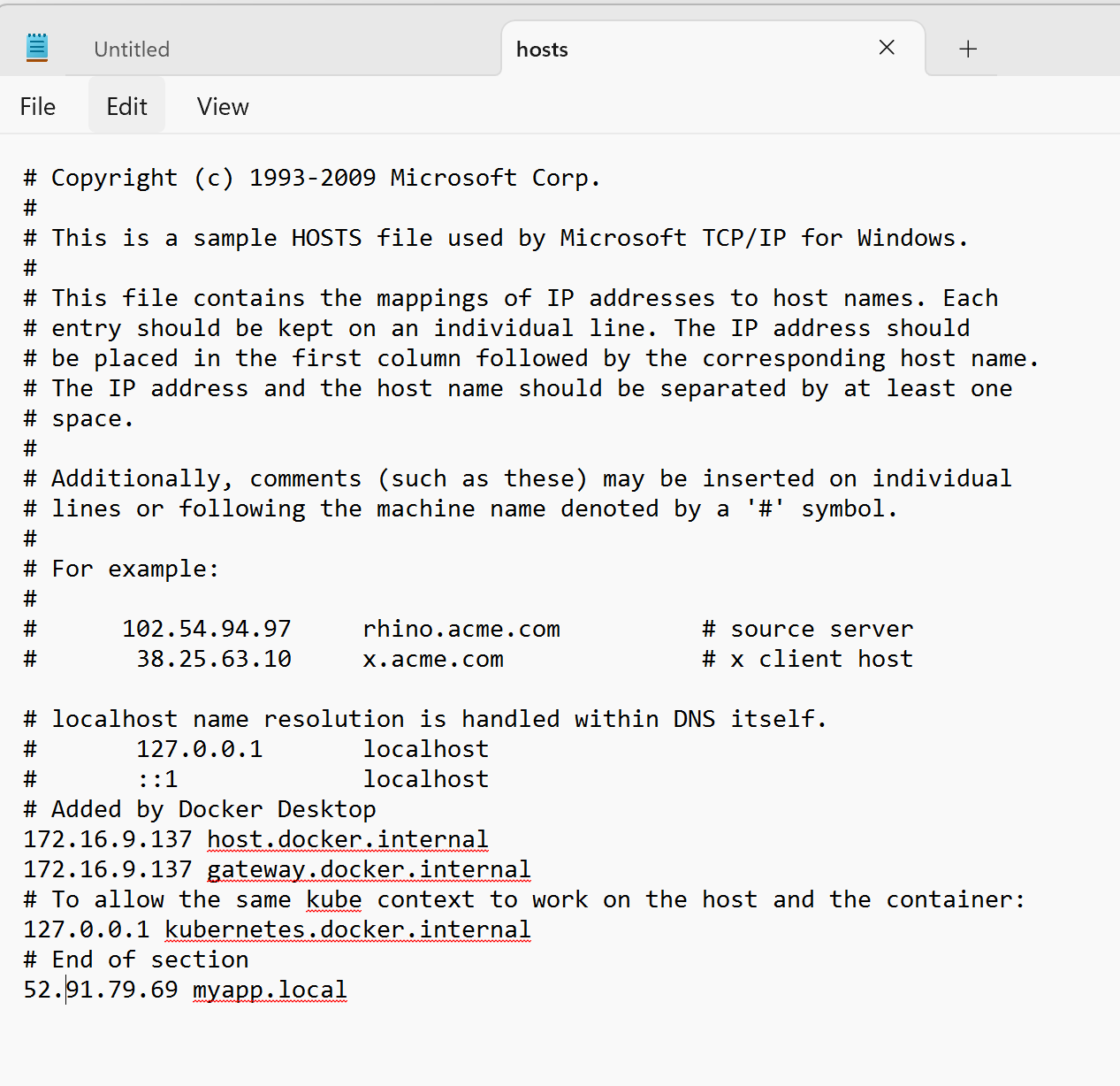
## Step 12:

* 1. At the bottom of the file, add:

<Your EC2 Public IP> myapp.local

Replace <Your EC2 Public IP> with the public IP you copied. (Eg: 52.91.79.69 myapp.local)

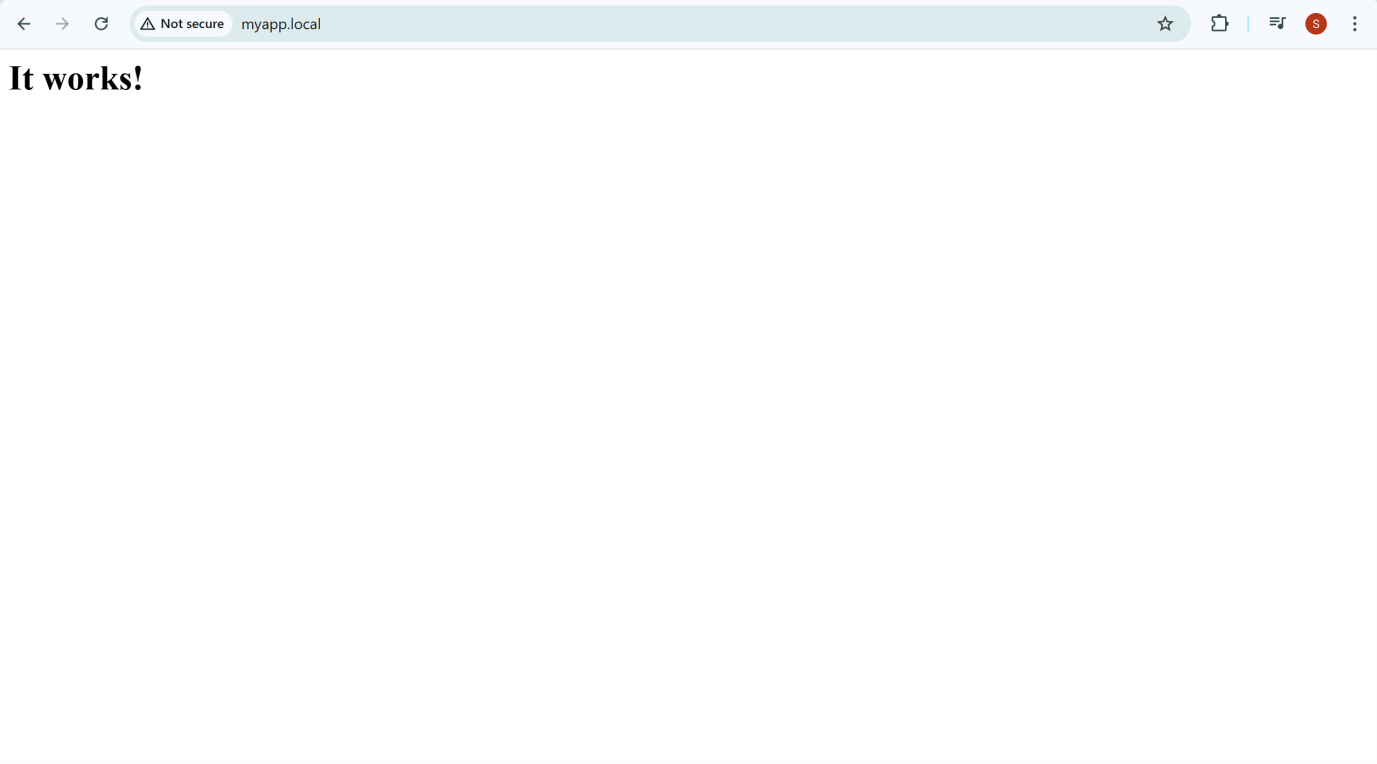
* 1. Save the file and close Notepad.



## Step 12:

Open your **web browser**.

Type myapp.local in the address bar and press **Enter**. You should see the Apache default page



**Outcome**

* Successfully set up DNS records that map the application’s IP or load balancer to a domain.
* Improved accessibility and performance of the web application.
* Enhanced traffic management and failover capabilities.
* A scalable and secure infrastructure that adapts to user demand and future application growth.

Would you like guidance on specific DNS services such a