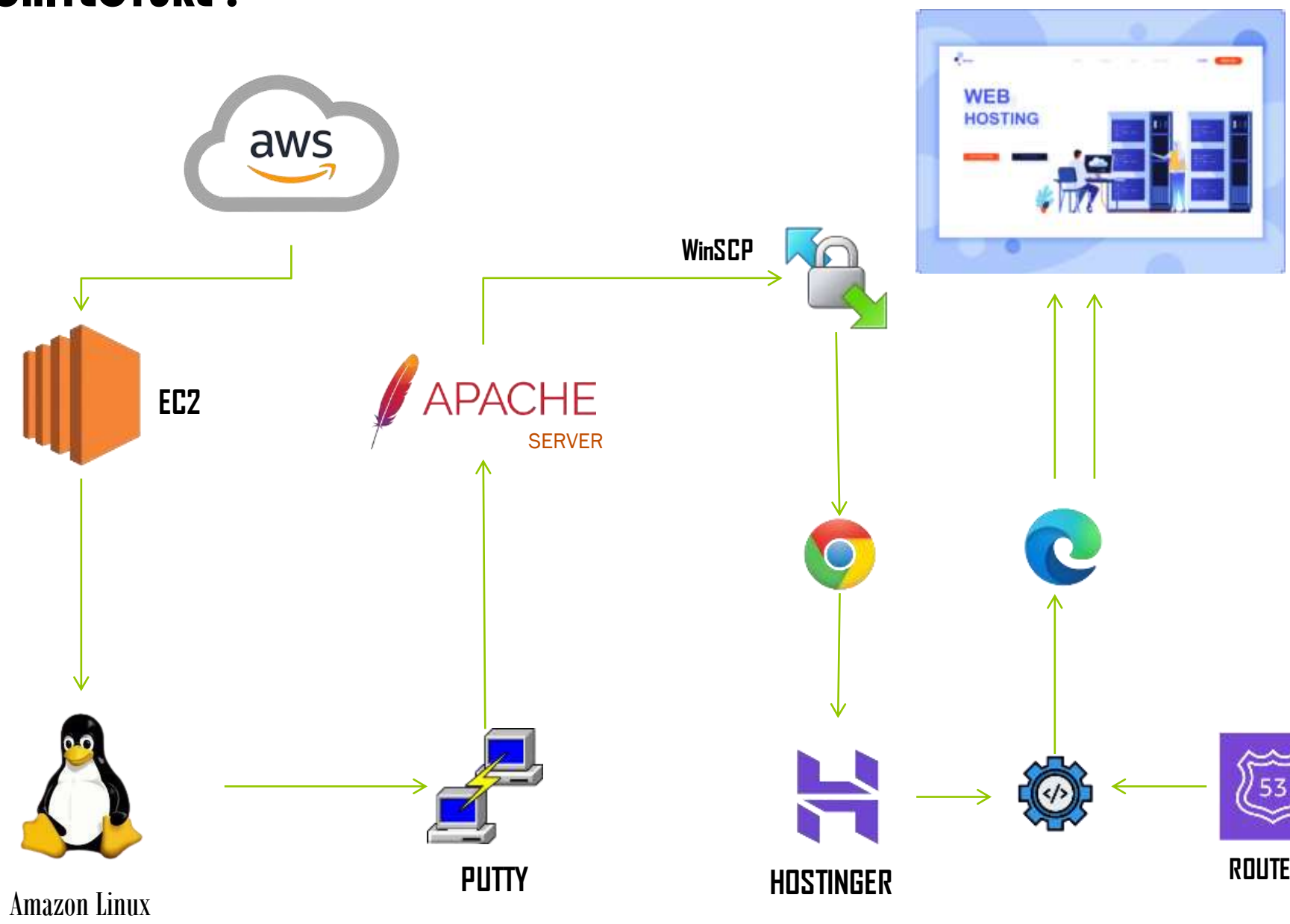


# HOSTING A **STATIC** WEBSITE **USING** AWS



ARCHITECTURE :-

OUTPUT



# EXPLANATION :

Deploy a static webpage using AWS, by launching an EC2 instance with the Amazon Linux 2 operating system. EC2 allows us to remotely run and manage servers. After launching the instance, we install the Apache HTTP Server, which is open-source software widely used to host web content. Apache serves our static website files (HTML, CSS, etc.) to users via HTTP. To transfer website assets from a Windows machine to the Linux server, we use WinSCP, a tool that enables secure file transfer using the EC2 instance's public IP and private key. Once the website files are in place, we can access the site through the EC2's public IP. To link a custom domain, we purchase one through Hostinger and configure it using Amazon Route 53. Route 53 handles DNS and maps the domain to the EC2 instance, making the site accessible by name instead of IP. We update the Hostinger domain's nameservers to point to Route 53. Finally, we can add routing policies in Route 53 to reduce server downtime and improve performance.



# TECHNOLOGY USED :



**Amazon Web Service**



**HOSTINGER**



**HTML**



**EC2**



**Amazon Linux**



**WinSCP**



**Route 53**

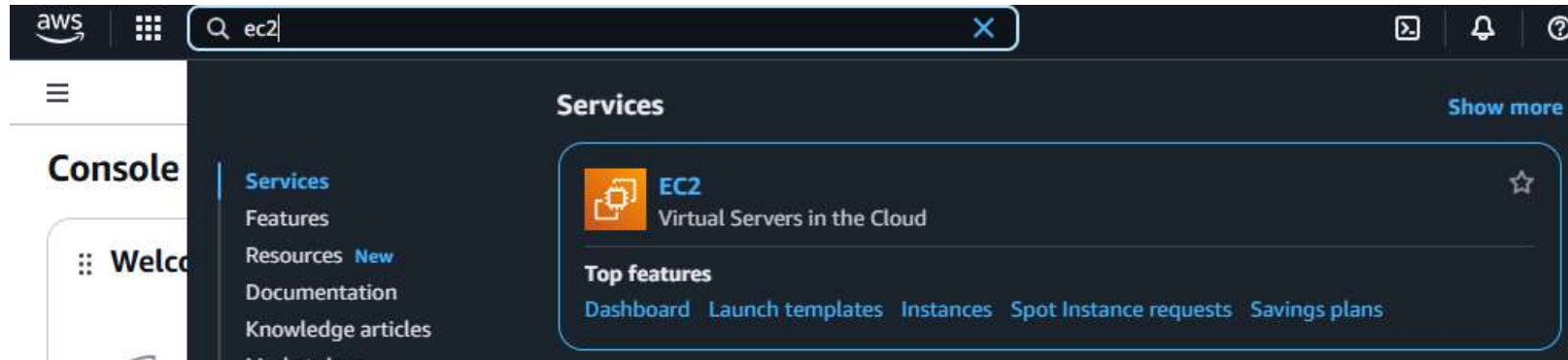


**PUTTY**



**APACHE  
SERVER**

# WORKING SAMPLE :



**Instances (1/1)** [Info](#) Last updated less than a minute ago [Connect](#) [Instance state](#) [Actions](#) [Launch instances](#)

[All states](#)

<input checked="" type="checkbox"/>	Name <a href="#">↗</a>	Instance ID	Instance state <a href="#">▼</a>	Instance type <a href="#">▼</a>	Status check	Alarm status	Availability Zone
<input checked="" type="checkbox"/>	Linux	i-044686fed0d4a6d4b	<span>Running</span> <a href="#">ⓘ</a> <a href="#">🔍</a>	t2.micro	<span>⌚</span> Initializing	<a href="#">View alarms +</a>	ap-south-1b

---

**i-044686fed0d4a6d4b (Linux)**

<b>Instance ID</b> <a href="#">🔗</a> i-044686fed0d4a6d4b	<b>Public IPv4 address</b> <a href="#">🔗</a> 3.109.213.224   <a href="#">open address</a> <a href="#">🔗</a>	<b>Private IPv4 addresses</b> <a href="#">🔗</a> 172.31.9.252
<b>IPv6 address</b> -	<b>Instance state</b> <span>Running</span>	<b>Public IPv4 DNS</b> <a href="#">🔗</a> ec2-3-109-213-224.ap-south-1.compute.amazonaws.com

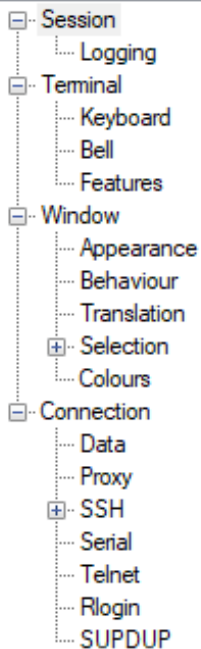


## PuTTY Configuration

?



Category:



## Basic options for your PuTTY session

Specify the destination you want to connect to

Host Name (or IP address)

Port

3.109.213.224

22

Connection type:

☒ SSH ☐ Serial ☐ Other:

- Load, save or delete a stored session

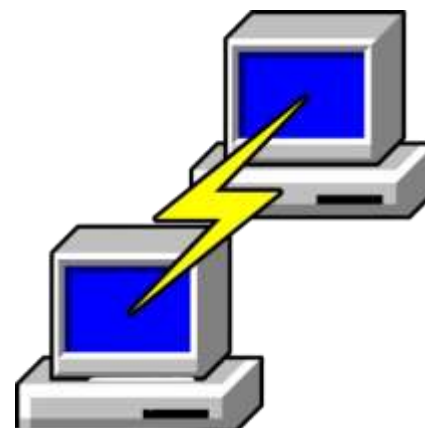
## Saved Sessions

### Default Settings

Load

Save

Delete



```
login as: ec2-user
```

```
Authenticating with public key "NEWKEY"
```

# Amazon Linux 2023

<https://aws.amazon.com/linux/amazon-linux-2023>

```
[ec2-user@ip-172-31-9-252 ~]$ cd /
[ec2-user@ip-172-31-9-252 /]$
```



# APACHE INSTALLATION :

```
root@ip-172-31-15-126:/  
[root@ip-172-31-15-126 /]# systemctl start httpd  
[root@ip-172-31-15-126 /]# systemctl status httpd  
● httpd.service - The Apache HTTP Server  
   Loaded: loaded (/usr/lib/systemd/system/httpd.service; disabled; preset: disabled)  
   Active: active (running) since Sat 2025-03-15 07:47:55 UTC; 35s ago  
     Docs: man:httpd.service(8)  
  Main PID: 25872 (httpd)  
    Status: "Total requests: 0; Idle/Busy workers 100/0; Requests/sec: 0; Bytes served/sec: 0 B/sec"  
    Tasks: 177 (limit: 1111)  
   Memory: 12.9M
```

## FILE PERMISSION :

```
[root@ip-172-31-9-252 var]# ls  
account  adm  cache  db  empty  ftp  games  kerberos  lib  loca  
[root@ip-172-31-9-252 var]# cd www  
[root@ip-172-31-9-252 www]# ls  
cgi-bin  html  
[root@ip-172-31-9-252 www]# cd html  
[root@ip-172-31-9-252 html]# chmod 777 /var/www/html
```



# APACHE INSTALLATION

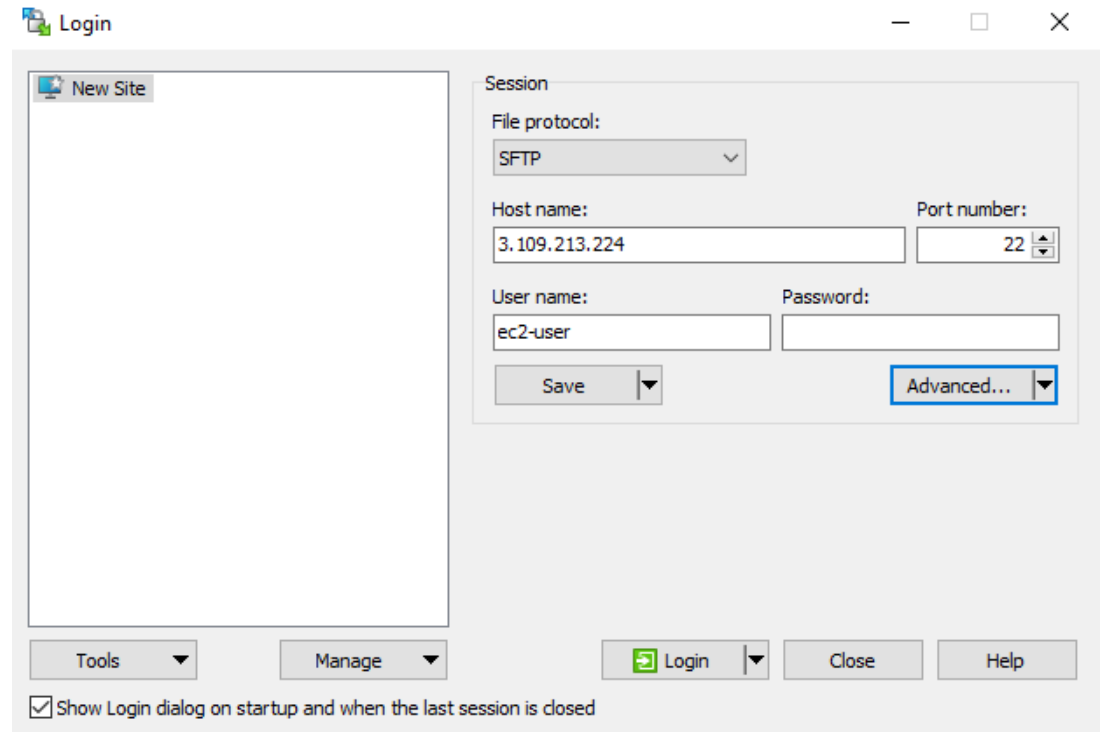
<code>sudo -i</code>	→	Changing as Root User
<code>cd /</code>	→	Change Directory
<code>Yum install httpd -y</code>	→	Installing Apache
<code>systemctl start httpd</code>	→	Starting Apache server
<code>systemctl status httpd</code>	→	Status check for apache
<code>Chmod 777 /var/html/www</code>	→	File Permission for apache directory



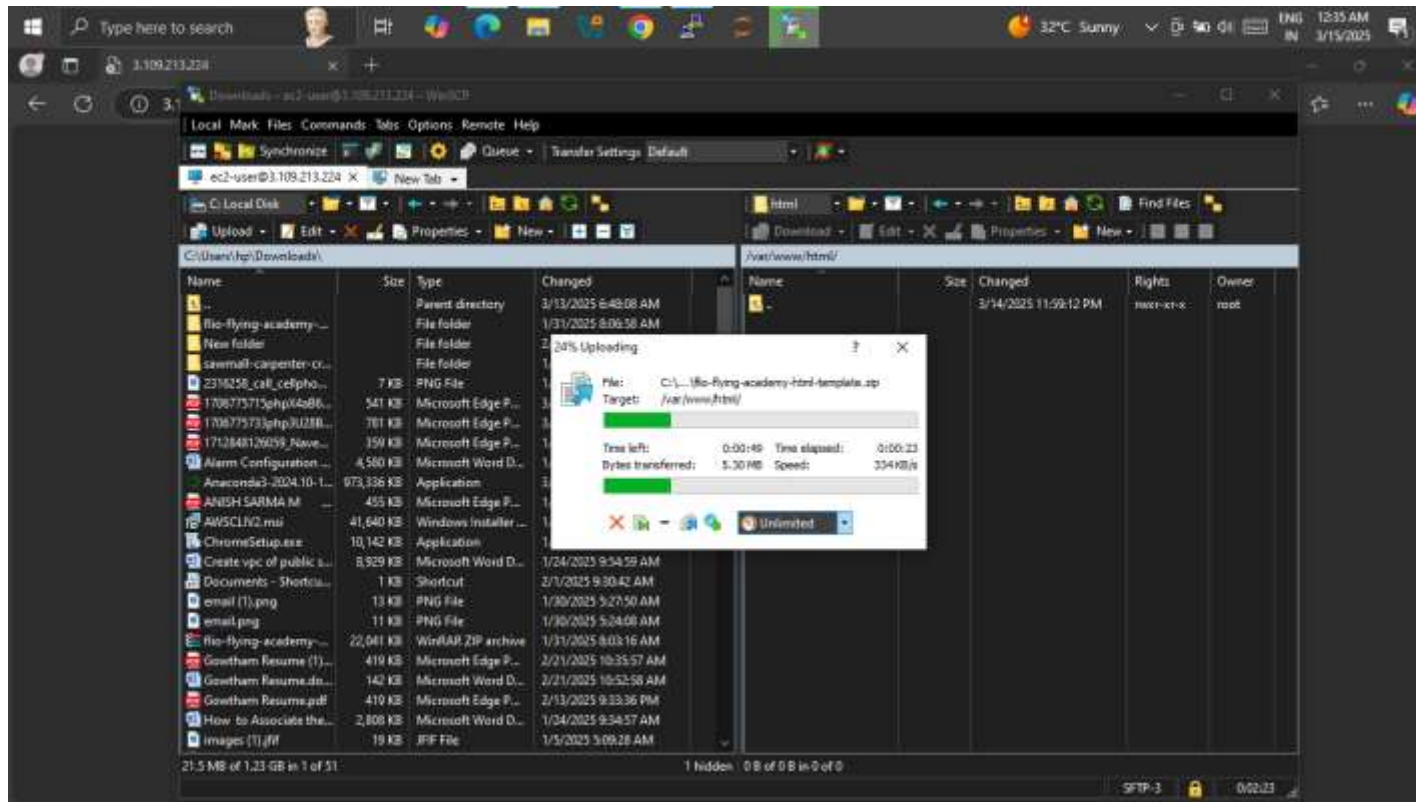


# CONNECTING WINSCP

- WinScp is a Tool used to Transfer data and files among different OS without any difficulties.



# ASSETS TRANSFER VIA WINSCP



# SOURCE CODE

```
<!DOCTYPE html>
<html lang="en">

<head>
  <meta charset="UTF-8" />
  <meta name="viewport" content="width=device-width, initial-scale=1.0" />
  <title>Vels University – Home</title>

  <!-- Google Fonts -->
  <link href="https://fonts.googleapis.com/css2?family=Poppins:wght@400;600&display=swap"
        rel="stylesheet">
  <link href="https://fonts.googleapis.com/css2?family=Cinzel:wght@700&display=swap"
        rel="stylesheet">
```



```
<style>
* {
  margin: 0;
  padding: 0;
  box-sizing: border-box;
  font-family: 'Poppins', sans-serif;
}

body {
  background-color: #000000;
  color: #ffffff;
  font-size: 16px;
  line-height: 1.6;
}

header {
  background-color: #111111;
  padding: 20px;
  text-align: center;
  box-shadow: 0 2px 5px rgba(255, 165, 0, 0.2);
}

.logo {
  max-width: 200px;
  height: auto;
  margin-bottom: 20px;
}
```



```
@keyframes fadeInSlideUp {  
  0% {  
    opacity: 0;  
    transform: translateY(30px);  
  }  
  
  100% {  
    opacity: 1;  
    transform: translateY(0);  
  }  
}  
  
.welcome-message {  
  background-color: #ff6600;  
  padding: 50px 20px;  
  text-align: center;  
  opacity: 0;  
  animation: fadeInSlideUp 2s ease-out forwards;  
}
```



```
.aws-title {  
  font-family: 'Cinzel', serif;  
  font-weight: 700;  
  letter-spacing: 1px;  
  font-size: 2.8em;  
  text-transform: uppercase;  
  margin-bottom: 20px;  
  color: #000000;  
}
```

```
.nav-bar {  
  background-color: #222222;  
  padding: 15px;  
  text-align: center;  
  display: flex;  
  justify-content: center;  
  gap: 15px;  
  flex-wrap: wrap;  
}
```



```
.nav-bar a {  
  color: #ffffff;  
  background-color: #ff6600;  
  padding: 10px 20px;  
  text-decoration: none;  
  border-radius: 5px;  
  font-weight: 600;  
  font-size: 1em;  
  transition: background-color 0.3s, color 0.3s;  
  display: inline-block;  
  cursor: pointer;  
}
```

```
.nav-bar a:hover {  
  background-color: #ffa347;  
  color: #000;  
}  
.content {  
  display: none;  
  padding: 30px;  
  background-color: #111111;  
  border-top: 2px solid #ff6600;  
}
```



```
.content.active {  
  display: block;  
  animation: slideFade 1s ease;  
}
```

```
@keyframes slideFade {  
  0% {  
    opacity: 0;  
    transform: translateY(20px);  
  }
```

```
  100% {  
    opacity: 1;  
    transform: translateY(0);  
  }  
}
```





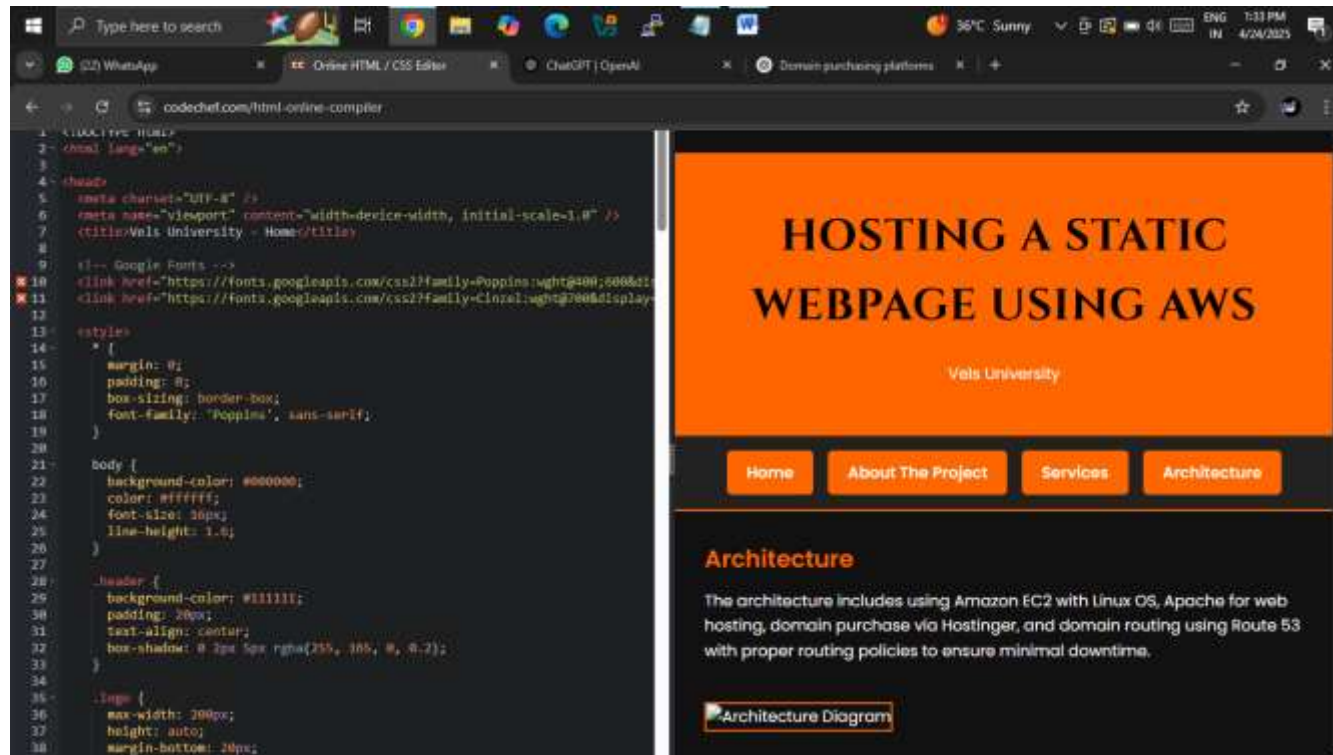
```
pre {  
  background-color: #222222;  
  padding: 10px;  
  border-left: 4px solid #ff6600;  
  overflow-x: auto;  
}
```

```
.architecture-image {  
  width: 100%;  
  max-width: 800px;  
  height: auto;  
  border: 2px solid #ff6600;  
  margin-top: 20px;  
}
```

```
.tool-icons {  
  display: flex;  
  flex-wrap: wrap;  
  justify-content: center;  
  gap: 20px;  
  margin-top: 20px;  
}
```

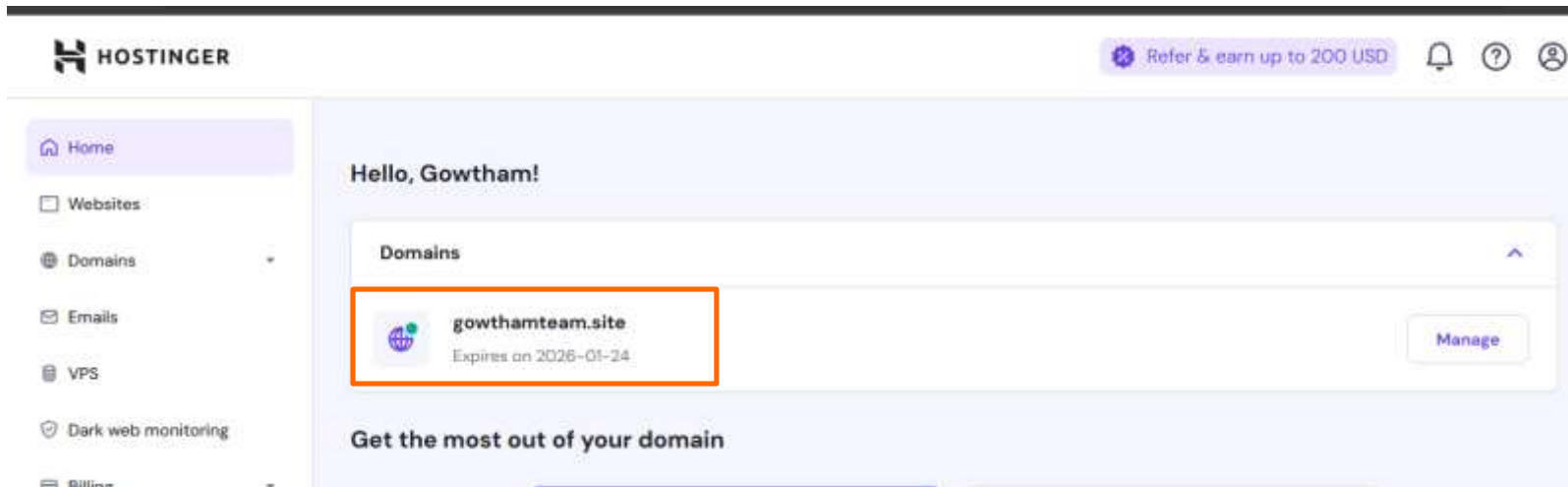


# TEST PAGE :



# PRIVATE DOMAIN :

Purchasing Private Domain “gowthamteam.site” from hostinger



# CONFIGURING PRIVATE DOMAIN VIA ROUTE 53

The screenshot displays the AWS Route 53 console interface. At the top, a green notification banner states: "gowthamteam.site was successfully created. Now you can create records in the hosted zone to specify how you want Route 53 to route traffic for your domain." Below this, the "Records (1/2)" section shows a table of existing records. The table has columns for Record, Type, Routing policy, Differ..., Alias, and Value/Route traffic. Two records are listed: an NS record for "gowthamteam.site" and an SOA record for "gowthamteam.site". The NS record is selected, and its details are shown on the right. The record details include the Record name "gowthamteam.site", Record type "NS", Value "ns-1536.awsdns-00.co.uk, ns-0.awsdns-00.com, ns-1024.awsdns-00.org, ns-512.awsdns-00.net", Alias "No", TTL (seconds) "172800", and Routing policy "Simple".

**Route 53**

- Dashboard
- Hosted zones**
- Health checks
- Profiles [New](#)
- ▼ **IP-based routing**
  - CIDR collections
- ▼ **Traffic flow**
  - Traffic policies
  - Policy records
- ▼ **Domains**
  - Registered domains
  - Requests
- ▼ **Resolver**
  - VPCs
  - Inbound endpoints

**Records (1/2)** [info](#)

Now you can create records in the hosted zone to specify how you want Route 53 to route traffic for your domain.

[Delete record](#) [Import zone file](#) [Create record](#)

Filter records by property or: [Type](#) [Routing policy](#) [Alias](#)

Record	Type	Routing policy	Differ...	Alias	Value/Route traffic
<input checked="" type="checkbox"/> gowthamteam.site	NS	Simple	-	No	ns-1536.awsdns-00.co.uk, ns-0.awsdns-00.com, ns-1024.awsdns-00.org, ns-512.awsdns-00.net
<input type="checkbox"/> gowthamteam.site	SOA	Simple	-	No	ns-1536.awsdns-00.co.uk

**Record details** [Edit record](#)

Record name: gowthamteam.site

Record type: NS

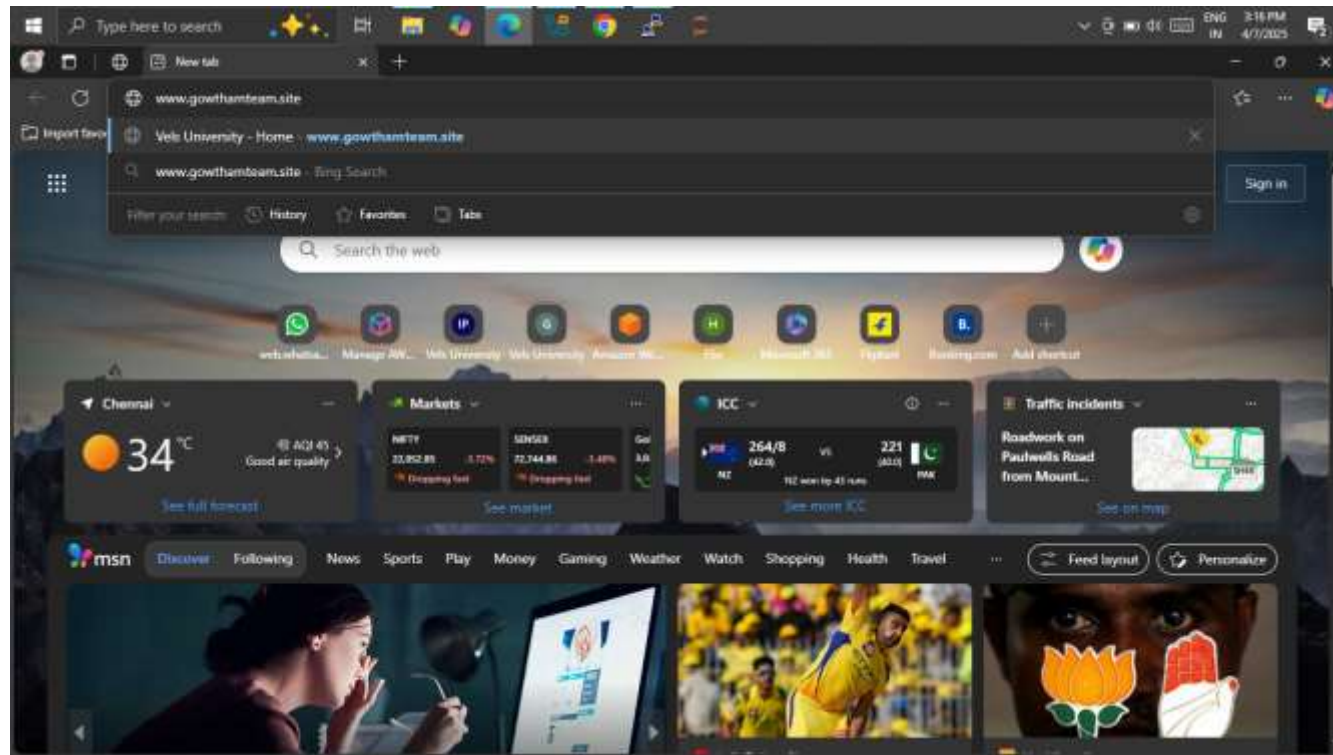
Value: ns-1536.awsdns-00.co.uk, ns-0.awsdns-00.com, ns-1024.awsdns-00.org, ns-512.awsdns-00.net

Alias: No

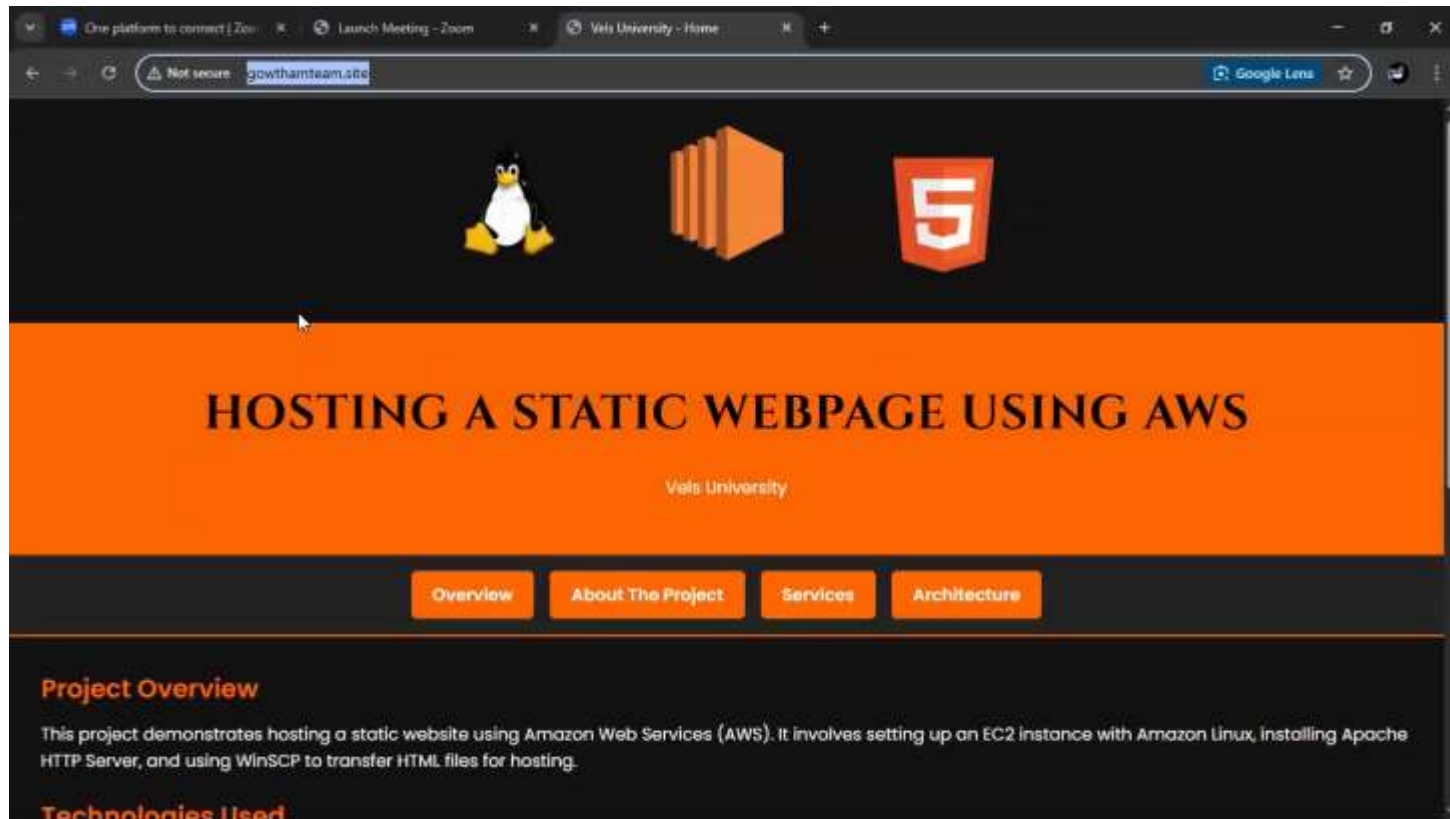
TTL (seconds): 172800

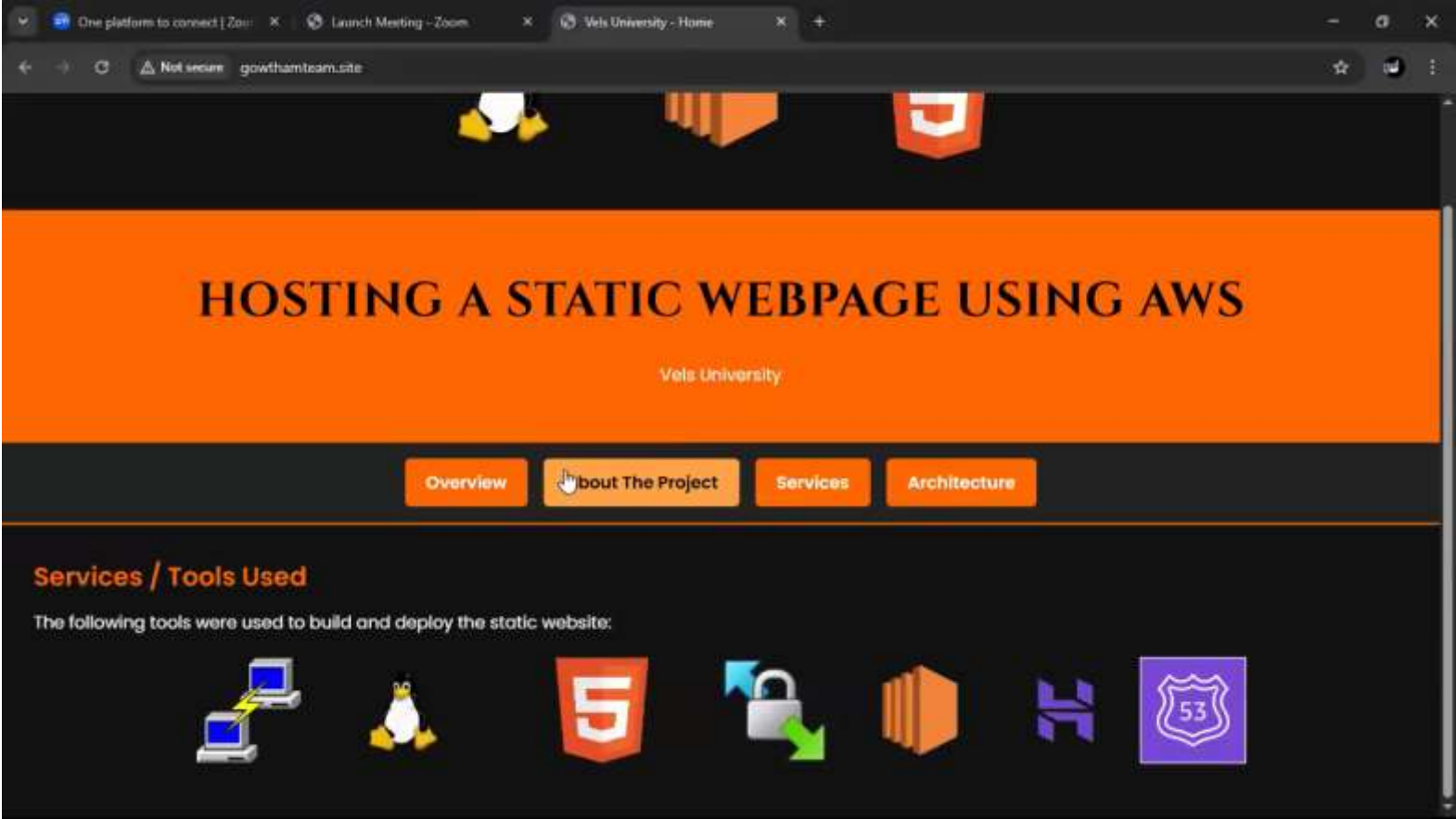
Routing policy: Simple

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



# OUTPUT :





One platform to connect | Zoom x Launch Meeting - Zoom x Vels University - Home x +

Not secure: gowthamteam.site


# HOSTING A STATIC WEBSITE USING AWS

Vels University

Overview About The Project **Services** Architecture

## Architecture

The architecture includes using Amazon EC2 with Linux OS, Apache for web hosting, domain purchase via Hostinger, and domain routing using Route 53 with proper routing policies to ensure minimal downtime.



The diagram illustrates the architecture for hosting a static website on AWS. It shows an Amazon EC2 instance (represented by orange server icons) connected to an Apache Server (represented by a red feather icon). The Apache Server is connected to a Web Hosting interface (represented by a blue box with a grid of icons). The Web Hosting interface is connected to a domain (represented by a blue box with a grid of icons). The domain is connected to a Route 53 icon (represented by a blue box with a grid of icons). The Route 53 icon is connected to a WnSCP icon (represented by a blue box with a grid of icons). The WnSCP icon is connected to a lock icon (represented by a blue box with a grid of icons). The lock icon is connected to a green arrow pointing to the Web Hosting interface. The diagram also shows an AWS cloud icon at the top left, connected to the EC2 instance.

Search 30°C Mostly clear ENG 1:16 PM





**THANK YOU**