

AWS EC2 Web Server Deployment Practical Report

Name : Rituraj Singh

Qid : 24030835

1. Aim

To deploy a basic web server using Amazon EC2, connect to it securely using SSH via Windows PowerShell, install Apache Web Server, and host a static webpage displaying student details.

2. Objective

- To understand AWS EC2 service and cloud infrastructure deployment
 - To configure Security Groups (Firewall Rules)
 - To connect to EC2 using SSH
 - To install and manage Apache Web Server
 - To deploy and access a website using Public IP
 - To troubleshoot cloud environment connection issues
-

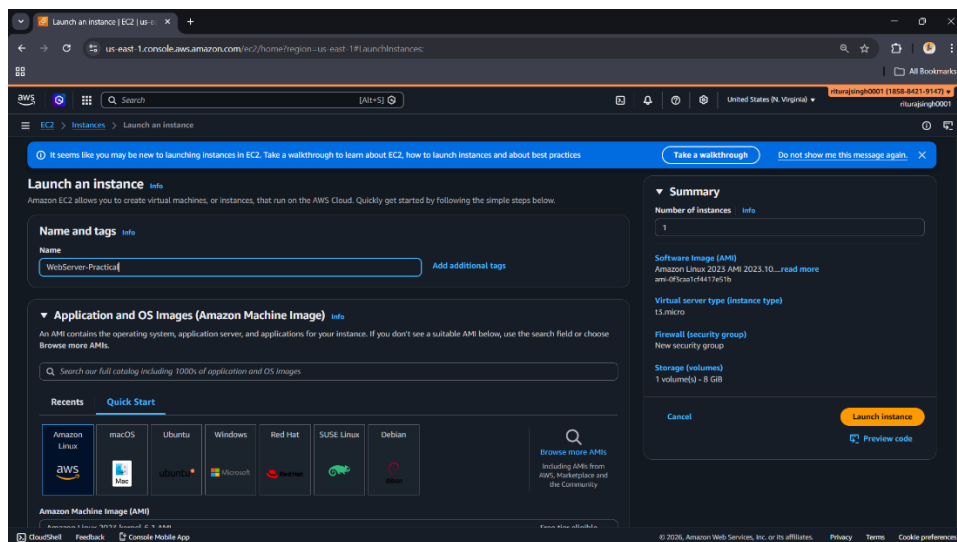
3. Tools & Technologies Used

- Amazon Web Services (AWS)
 - EC2 (Elastic Compute Cloud)
 - Amazon Linux 2023 AMI
 - Apache HTTP Server (httpd)
 - Windows PowerShell
 - SSH (Secure Shell Protocol)
 - Web Browser (Chrome/Edge)
-

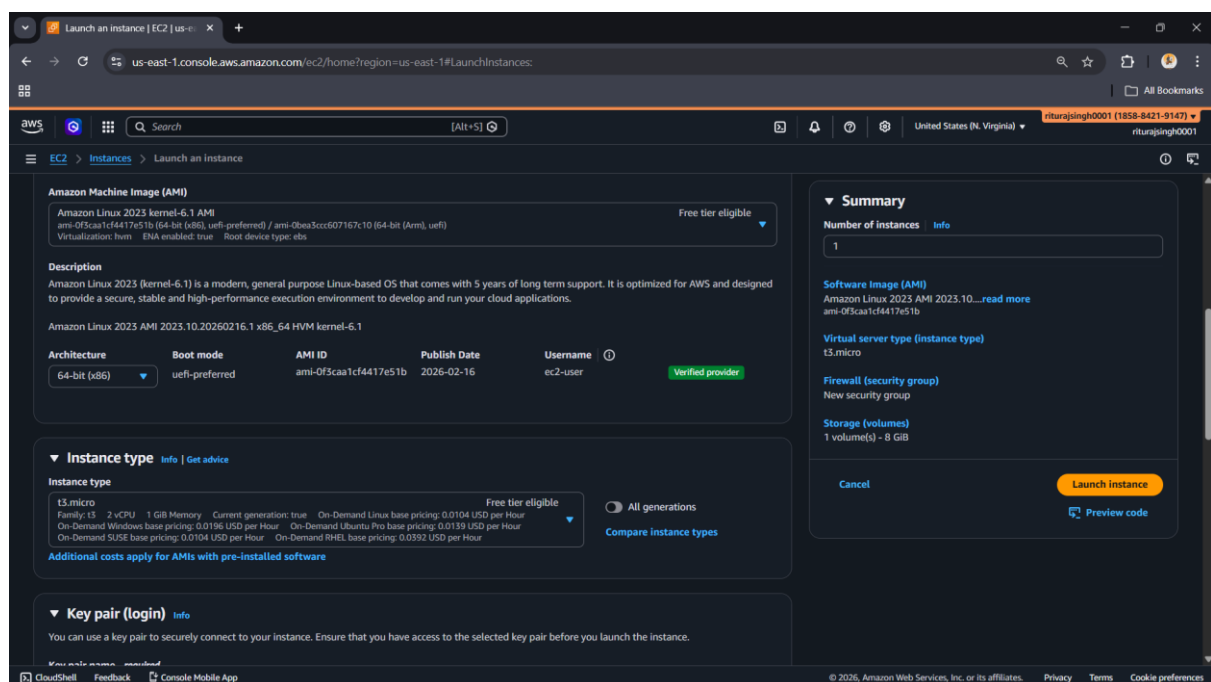
4. Procedure

Step 1: Launch EC2 Instance

1. Logged into AWS Management Console.
2. Navigated to EC2 Dashboard.
3. Clicked on **Launch Instance**.
4. Instance Name : WebServer-Practical



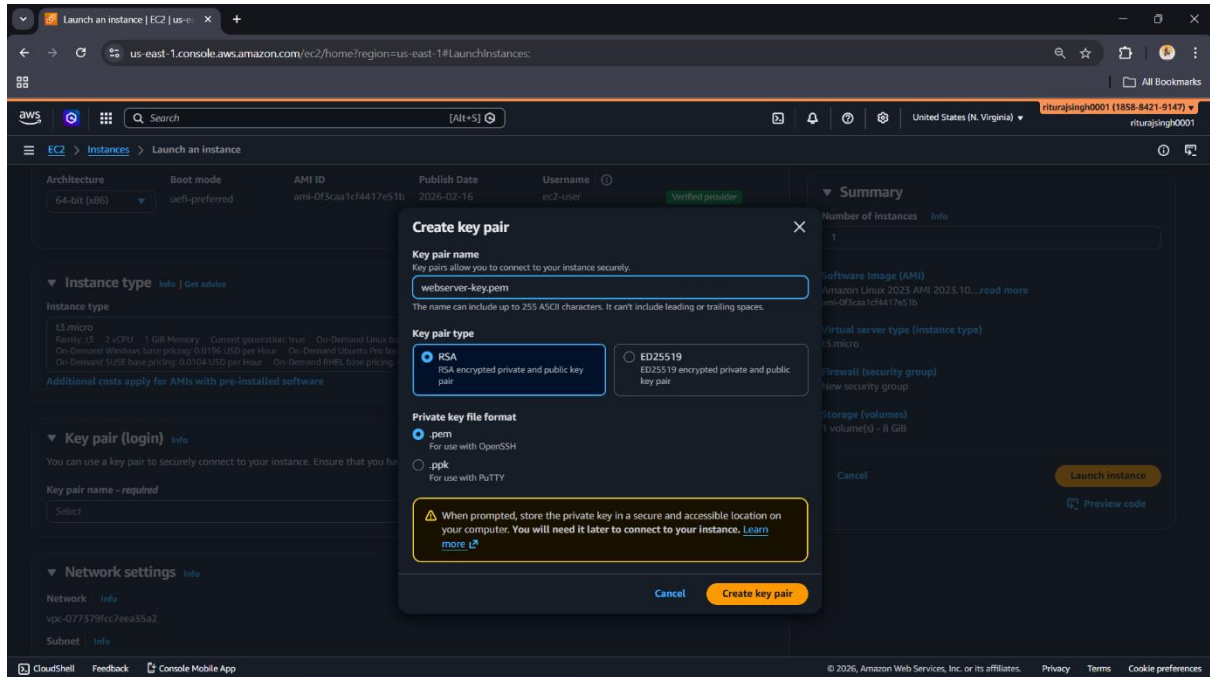
5. Selected:
 - AMI: Amazon Linux 2023
 - Instance Type: t3.micro



6. Created a new Key Pair (webserver-key.pem).

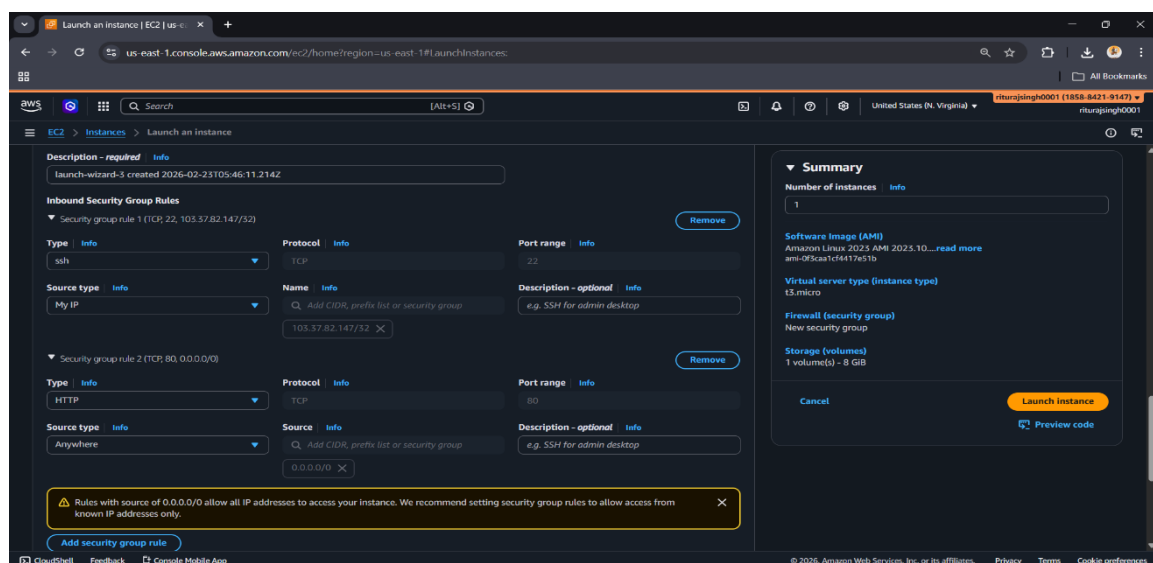
Under Key Pair:

1. Click **Create new key pair**
2. Name: webserver-key
3. Key pair type: RSA
4. File format: .pem
5. Click **Create Key Pair**

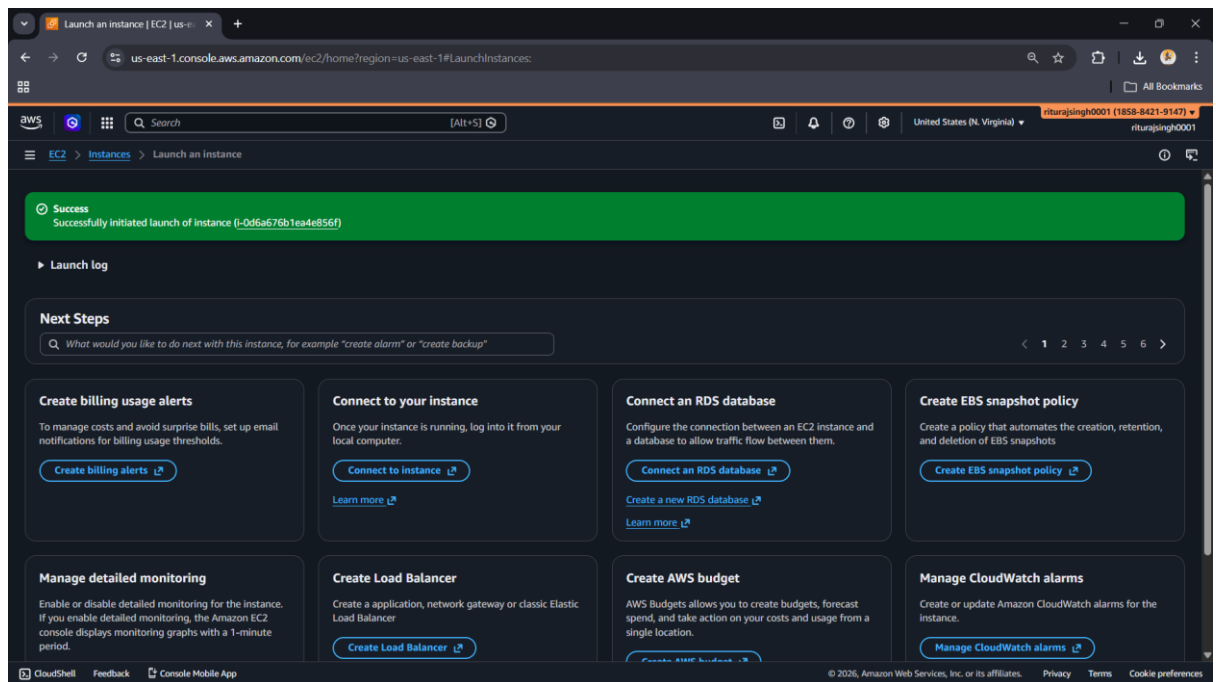


7. Configured Security Group:

- SSH (Port 22) – Source: My IP
- HTTP (Port 80) – Source: Anywhere



8. Launched the instance and waited until status showed:
Running (3/3 checks passed)

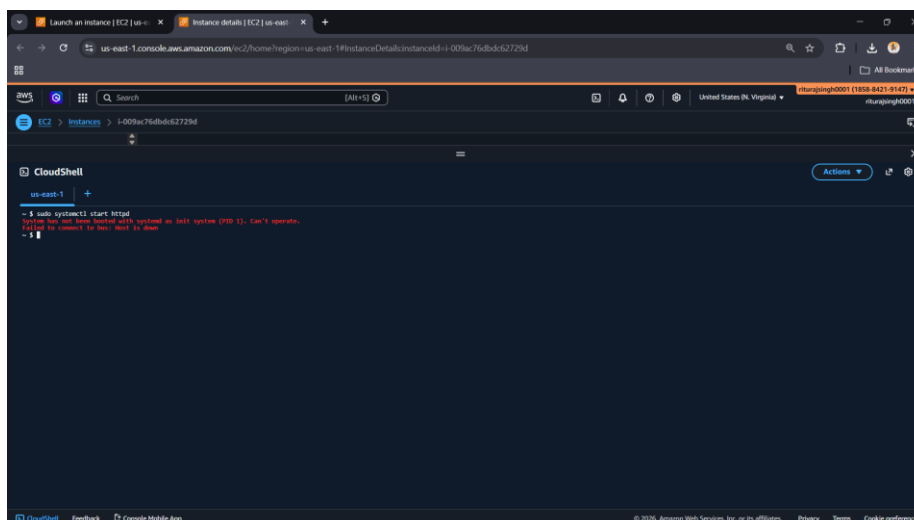


Step 2: Initial Attempt Using AWS CloudShell

Initially, Apache installation commands were executed inside **AWS CloudShell**.

When running:

`sudo systemctl start httpd`



The following error was encountered:

System has not been booted with systemd as init system (PID 1). Can't operate.

Failed to connect to bus: Host is down

Reason for Error:

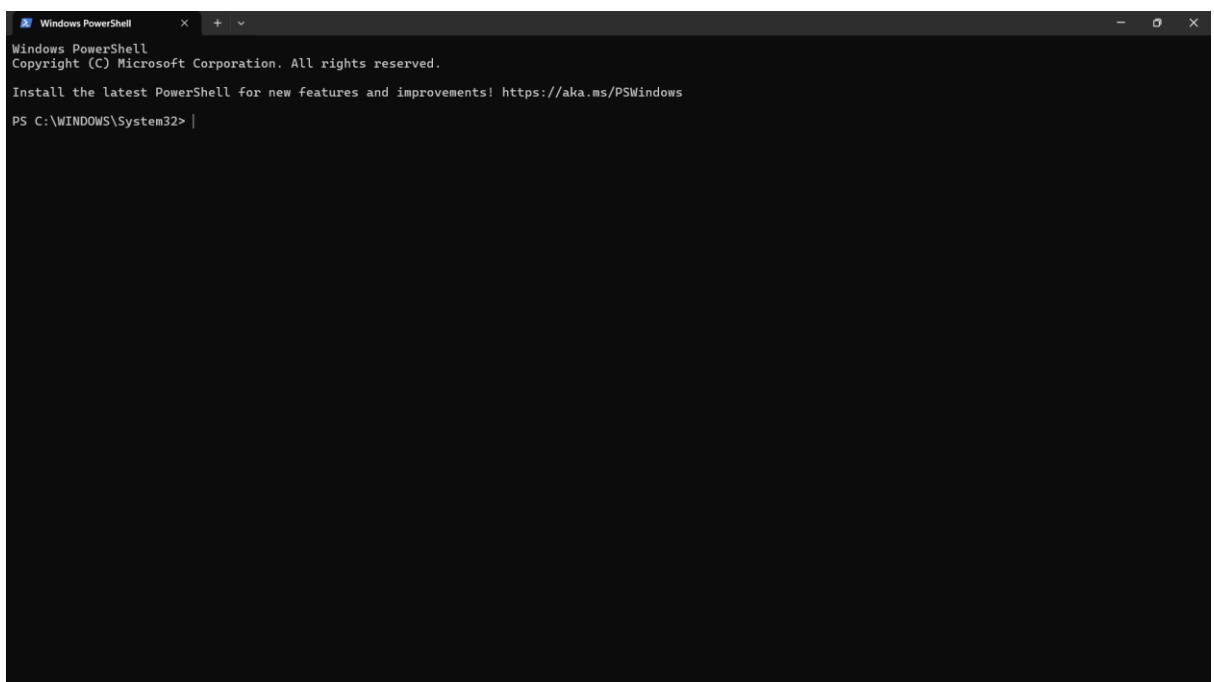
- AWS CloudShell is a managed shell environment.
- It does not operate as the EC2 instance operating system.
- It does not support system service control using systemctl.
- Apache service must be installed inside the actual EC2 instance.

This demonstrated that CloudShell is different from the EC2 server environment.

Step 3: Connecting to EC2 Using Windows PowerShell (Correct Method)

To resolve the issue, a direct SSH connection was established using Windows PowerShell.

1. Opened Windows PowerShell.

A screenshot of a Windows PowerShell terminal window. The title bar at the top reads "Windows PowerShell" with standard window controls. The terminal content shows the PowerShell version and copyright information: "Windows PowerShell", "Copyright (C) Microsoft Corporation. All rights reserved.", and a message to install the latest PowerShell: "Install the latest PowerShell for new features and improvements! https://aka.ms/PSWindows". The prompt "PS C:\WINDOWS\System32>" is visible at the bottom of the terminal.

2. Navigated to the folder where the key file was downloaded:

```
cd C:\Users\YourName\Downloads
```

```
Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.

Install the latest PowerShell for new features and improvements! https://aka.ms/PSWindows

PS C:\WINDOWS\System32> cd C:\Users\asusv\Downloads
PS C:\Users\asusv\Downloads> |
```

3. Connected using SSH:

`ssh -i webserver-key.pem ec2-user@Public-IP`

```
ec2-user@ip-172-31-26-223:~$ ssh -i webserver-key.pem ec2-user@13.221.111.128
The authenticity of host '13.221.111.128 (13.221.111.128)' can't be established.
ED25519 key fingerprint is SHA256:LaSbUx7+NiLNlFTWf9Cm0w8sv7y9t8LKwjXofCsSDaM.
This key is not known by any other names.
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added '13.221.111.128' (ED25519) to the list of known hosts.

      _#_
     _###_
    _####_
   _####_
  _####_
 _####_
_####_

Amazon Linux 2023

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

[ec2-user@ip-172-31-26-223 ~]$
```

4. Accepted the fingerprint confirmation by typing:
Yes

Successfully connected to the EC2 instance (Amazon Linux terminal appeared).

Step 4: Install Apache Web Server

Inside the EC2 terminal, the following commands were executed:

```
sudo yum update -y
```

[illegible]

```
sudo yum install httpd -y
```

```

ec2-user@ip-172-31-26-223: ~ % sudo yum install httpd -y
Complete!
[ec2-user@ip-172-31-26-223 ~]$ sudo yum install httpd -y
Last metadata expiration check: 0:00:31 ago on Mon Feb 23 07:00:40 2026.
Dependencies resolved.

=====
Package                               Architecture      Version            Repository          Size
=====
Installing:
httpd                                 x86_64            2.4.66-1.amzn2023.0.1  amazonlinux        47 k
Installing dependencies:
apr                                   x86_64            1.7.5-1.amzn2023.0.4   amazonlinux        129 k
apr-util                             x86_64            1.6.3-1.amzn2023.0.2   amazonlinux        97 k
apr-util-ldap                        x86_64            1.6.3-1.amzn2023.0.2   amazonlinux        13 k
generic-logos-httpd                 noarch            18.0-8.12.amzn2023.0.3  amazonlinux        19 k
httpd-core                           x86_64            2.4.66-1.amzn2023.0.1  amazonlinux        1.4 M
httpdfilesystem                     noarch            2.4.66-1.amzn2023.0.1  amazonlinux        13 k
httpd-tools                         x86_64            2.4.66-1.amzn2023.0.1  amazonlinux        81 k
libbrotli                           x86_64            1.0.9-4.amzn2023.0.2   amazonlinux        315 k
mailcap                             noarch            2.1.49-3.amzn2023.0.3  amazonlinux        33 k
Installing weak dependencies:
apr-util-openssl                    x86_64            1.6.3-1.amzn2023.0.2   amazonlinux        15 k
mod_http2                           x86_64            2.0.27-1.amzn2023.0.1  amazonlinux        166 k
mod_lua                              x86_64            2.4.66-1.amzn2023.0.1  amazonlinux        60 k
Transaction Summary
=====
Install 13 Packages

Total download size: 2.4 M
Installed size: 6.9 M
Downloading Packages:
(1/13): apr-util-1.6.3-1.amzn2023.0.2.x86_64.rpm                2.7 MB/s | 97 kB | 00:00
(2/13): apr-util-ldap-1.6.3-1.amzn2023.0.2.x86_64.rpm           354 kB/s | 13 kB | 00:00
(3/13): apr-1.7.5-1.amzn2023.0.4.x86_64.rpm                     3.0 MB/s | 120 kB | 00:00
(4/13): generic-logos-httpd-18.0-8.12.amzn2023.0.3.noarch.rpm   693 kB/s | 19 kB | 00:00
(5/13): apr-util-openssl-1.6.3-1.amzn2023.0.2.x86_64.rpm        490 kB/s | 15 kB | 00:00
(6/13): httpd-2.4.66-1.amzn2023.0.1.x86_64.rpm                 1.6 MB/s | 47 kB | 00:00
(7/13): httpdfilesystem-2.4.66-1.amzn2023.0.1.noarch.rpm        552 kB/s | 13 kB | 00:00
(8/13): httpd-core-2.4.66-1.amzn2023.0.1.x86_64.rpm             37 MB/s | 1.4 MB | 00:00
(9/13): httpd-tools-2.4.66-1.amzn2023.0.1.x86_64.rpm           2.2 MB/s | 81 kB | 00:00
(10/13): libbrotli-1.0.9-4.amzn2023.0.2.x86_64.rpm             13 MB/s | 315 kB | 00:00
(11/13): mailcap-2.1.49-3.amzn2023.0.3.noarch.rpm              1.4 MB/s | 33 kB | 00:00
(12/13): mod_http2-2.0.27-1.amzn2023.0.3.x86_64.rpm            6.1 MB/s | 166 kB | 00:00
(13/13): mod_lua-2.4.66-1.amzn2023.0.1.x86_64.rpm              2.0 MB/s | 60 kB | 00:00

```

sudo systemctl start httpd

sudo systemctl enable httpd

```
Transaction test succeeded.
Running transaction
  Preparing      :                                1/1
  Installing     : apr-1.7.5-1.amzn2023.0.4.x86_64 1/13
  Installing     : apr-util-lmdb-1.6.3-1.amzn2023.0.2.x86_64 2/13
  Installing     : apr-util-openssl-1.6.3-1.amzn2023.0.2.x86_64 3/13
  Installing     : apr-util-1.6.3-1.amzn2023.0.2.x86_64 4/13
  Installing     : mailcap-2.1.49-3.amzn2023.0.3.noarch 5/13
  Installing     : httpd-tools-2.4.66-1.amzn2023.0.1.x86_64 6/13
  Installing     : libbrotli-1.0.9-4.amzn2023.0.2.x86_64 7/13
  Running scriptlet: httpd-filesystem-2.4.66-1.amzn2023.0.1.noarch 8/13
  Installing     : httpd-filesystem-2.4.66-1.amzn2023.0.1.noarch 8/13
  Installing     : httpd-core-2.4.66-1.amzn2023.0.1.x86_64 9/13
  Installing     : mod_http2-2.0.27-1.amzn2023.0.3.x86_64 10/13
  Installing     : mod_lua-2.4.66-1.amzn2023.0.1.x86_64 11/13
  Installing     : generic-logos-httpd-18.0.0-12.amzn2023.0.3.noarch 12/13
  Installing     : httpd-2.4.66-1.amzn2023.0.1.x86_64 13/13
  Running scriptlet: httpd-2.4.66-1.amzn2023.0.1.x86_64 13/13
  Verifying      : apr-1.7.5-1.amzn2023.0.4.x86_64 13/13
  Verifying      : apr-util-1.6.3-1.amzn2023.0.2.x86_64 2/13
  Verifying      : apr-util-lmdb-1.6.3-1.amzn2023.0.2.x86_64 3/13
  Verifying      : apr-util-openssl-1.6.3-1.amzn2023.0.2.x86_64 4/13
  Verifying      : generic-logos-httpd-18.0.0-12.amzn2023.0.3.noarch 5/13
  Verifying      : httpd-2.4.66-1.amzn2023.0.1.x86_64 6/13
  Verifying      : httpd-core-2.4.66-1.amzn2023.0.1.x86_64 7/13
  Verifying      : httpd-filesystem-2.4.66-1.amzn2023.0.1.noarch 8/13
  Verifying      : httpd-tools-2.4.66-1.amzn2023.0.1.x86_64 9/13
  Verifying      : libbrotli-1.0.9-4.amzn2023.0.2.x86_64 10/13
  Verifying      : mailcap-2.1.49-3.amzn2023.0.3.noarch 11/13
  Verifying      : mod_http2-2.0.27-1.amzn2023.0.3.x86_64 12/13
  Verifying      : mod_lua-2.4.66-1.amzn2023.0.1.x86_64 13/13

Installed:
apr-1.7.5-1.amzn2023.0.4.x86_64      apr-util-1.6.3-1.amzn2023.0.2.x86_64      apr-util-lmdb-1.6.3-1.amzn2023.0.2.x86_64
apr-util-openssl-1.6.3-1.amzn2023.0.2.x86_64      generic-logos-httpd-18.0.0-12.amzn2023.0.3.noarch      httpd-2.4.66-1.amzn2023.0.1.x86_64
httpd-core-2.4.66-1.amzn2023.0.1.x86_64      httpd-filesystem-2.4.66-1.amzn2023.0.1.noarch      httpd-tools-2.4.66-1.amzn2023.0.1.x86_64
libbrotli-1.0.9-4.amzn2023.0.2.x86_64      mailcap-2.1.49-3.amzn2023.0.3.noarch      mod_http2-2.0.27-1.amzn2023.0.3.x86_64
mod_lua-2.4.66-1.amzn2023.0.1.x86_64

Complete!
[ec2-user@ip-172-31-26-223 ~]$ sudo systemctl start httpd
[ec2-user@ip-172-31-26-223 ~]$ sudo systemctl enable httpd
Created symlink /etc/systemd/system/multi-user.target.wants/httpd.service → /usr/lib/systemd/system/httpd.service.
[ec2-user@ip-172-31-26-223 ~]$
```

Apache service status was verified using:

sudo systemctl status httpd

```
Verifying      : apr-util-lmdb-1.6.3-1.amzn2023.0.2.x86_64 3/13
Verifying      : apr-util-openssl-1.6.3-1.amzn2023.0.2.x86_64 4/13
Verifying      : generic-logos-httpd-18.0.0-12.amzn2023.0.3.noarch 5/13
Verifying      : httpd-2.4.66-1.amzn2023.0.1.x86_64 6/13
Verifying      : httpd-core-2.4.66-1.amzn2023.0.1.x86_64 7/13
Verifying      : httpd-filesystem-2.4.66-1.amzn2023.0.1.noarch 8/13
Verifying      : httpd-tools-2.4.66-1.amzn2023.0.1.x86_64 9/13
Verifying      : libbrotli-1.0.9-4.amzn2023.0.2.x86_64 10/13
Verifying      : mailcap-2.1.49-3.amzn2023.0.3.noarch 11/13
Verifying      : mod_http2-2.0.27-1.amzn2023.0.3.x86_64 12/13
Verifying      : mod_lua-2.4.66-1.amzn2023.0.1.x86_64 13/13

Installed:
apr-1.7.5-1.amzn2023.0.4.x86_64      apr-util-1.6.3-1.amzn2023.0.2.x86_64      apr-util-lmdb-1.6.3-1.amzn2023.0.2.x86_64
apr-util-openssl-1.6.3-1.amzn2023.0.2.x86_64      generic-logos-httpd-18.0.0-12.amzn2023.0.3.noarch      httpd-2.4.66-1.amzn2023.0.1.x86_64
httpd-core-2.4.66-1.amzn2023.0.1.x86_64      httpd-filesystem-2.4.66-1.amzn2023.0.1.noarch      httpd-tools-2.4.66-1.amzn2023.0.1.x86_64
libbrotli-1.0.9-4.amzn2023.0.2.x86_64      mailcap-2.1.49-3.amzn2023.0.3.noarch      mod_http2-2.0.27-1.amzn2023.0.3.x86_64
mod_lua-2.4.66-1.amzn2023.0.1.x86_64

Complete!
[ec2-user@ip-172-31-26-223 ~]$ sudo systemctl start httpd
[ec2-user@ip-172-31-26-223 ~]$ sudo systemctl enable httpd
Created symlink /etc/systemd/system/multi-user.target.wants/httpd.service → /usr/lib/systemd/system/httpd.service.
[ec2-user@ip-172-31-26-223 ~]$ sudo systemctl status httpd
● httpd.service - The Apache HTTP Server
   Loaded: loaded (/usr/lib/systemd/system/httpd.service; enabled; preset: disabled)
   Active: active (running) since Mon 2026-02-23 07:02:19 UTC; 1min 8s ago
     Docs: man:httpd.service(8)
  Main PID: 25448 (httpd)
    Status: "Total requests: 0; Idle/Busy workers 100/0; Requests/sec: 0; Bytes served/sec: 0 B/sec"
    Tasks: 177 (limit: 1067)
   Memory: 13.4M
      CPU: 116ms
   CGroup: /system.slice/httpd.service
           └─25448 /usr/sbin/httpd -DFOREGROUND
           └─25473 /usr/sbin/httpd -DFOREGROUND
           └─25477 /usr/sbin/httpd -DFOREGROUND
           └─25478 /usr/sbin/httpd -DFOREGROUND
           └─25481 /usr/sbin/httpd -DFOREGROUND

Feb 23 07:02:19 ip-172-31-26-223.ec2.internal systemd[1]: Starting httpd.service - The Apache HTTP Server...
Feb 23 07:02:19 ip-172-31-26-223.ec2.internal systemd[1]: Started httpd.service - The Apache HTTP Server.
Feb 23 07:02:19 ip-172-31-26-223.ec2.internal httpd[25448]: Server configured, listening on: port 80
[ec2-user@ip-172-31-26-223 ~]$
```


Service showed:
active (running)

Step 5: Create Webpage

Navigated to the web root directory:

```
cd /var/www/html
```

Created and edited index file:

```
sudo nano index.html
```

Added HTML content including:

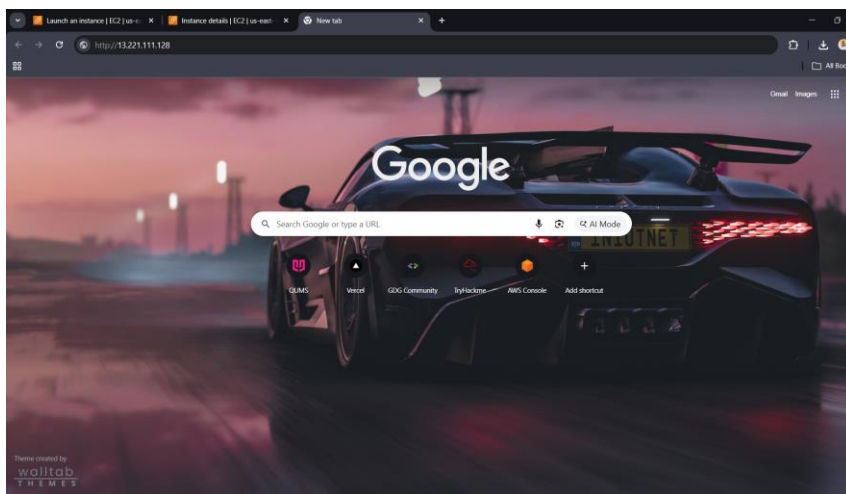
- Title: AWS EC2 Web Server Practical
- Student Name
- Student ID

Saved and exited the file.

Step 6: Access Website

1. Copied Public IPv4 address from EC2 dashboard.
2. Opened web browser.
3. Entered:

<http://Public-IP>



The webpage loaded successfully displaying the student details.

5. Security Configuration

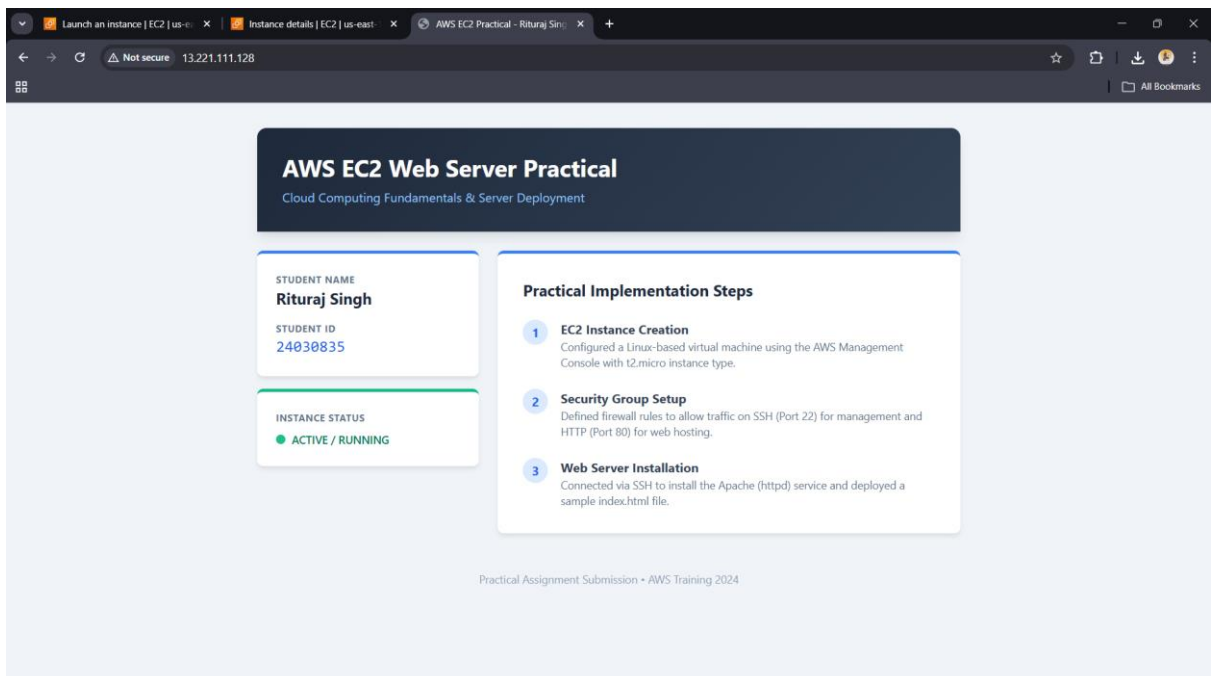
- SSH access restricted to "My IP"
- HTTP (Port 80) allowed for public access
- Instance health checks: 3/3 passed

6. Output

The Apache Web Server was successfully deployed on Amazon EC2.

The website was accessible through the public IP address and displayed:

- AWS EC2 Web Server Practical
- Student Name
- Student ID



7. Troubleshooting & Learning

During the practical, an error was encountered while using AWS CloudShell for service management. The issue was analyzed and resolved by switching to a direct SSH connection via Windows PowerShell.

This demonstrated:

- Understanding of difference between CloudShell and EC2
- Troubleshooting ability
- Practical knowledge of system service management
- Real-time cloud server handling

8. Conclusion

The practical was successfully completed. An EC2 instance was launched, Apache Web Server was installed and configured, and a static webpage was deployed and accessed via public IP.

The task enhanced understanding of cloud-based server deployment, Linux command-line operations, SSH connectivity, and troubleshooting in AWS environments.