

Static Website Deployment On Amazon Linux Using HTTPD

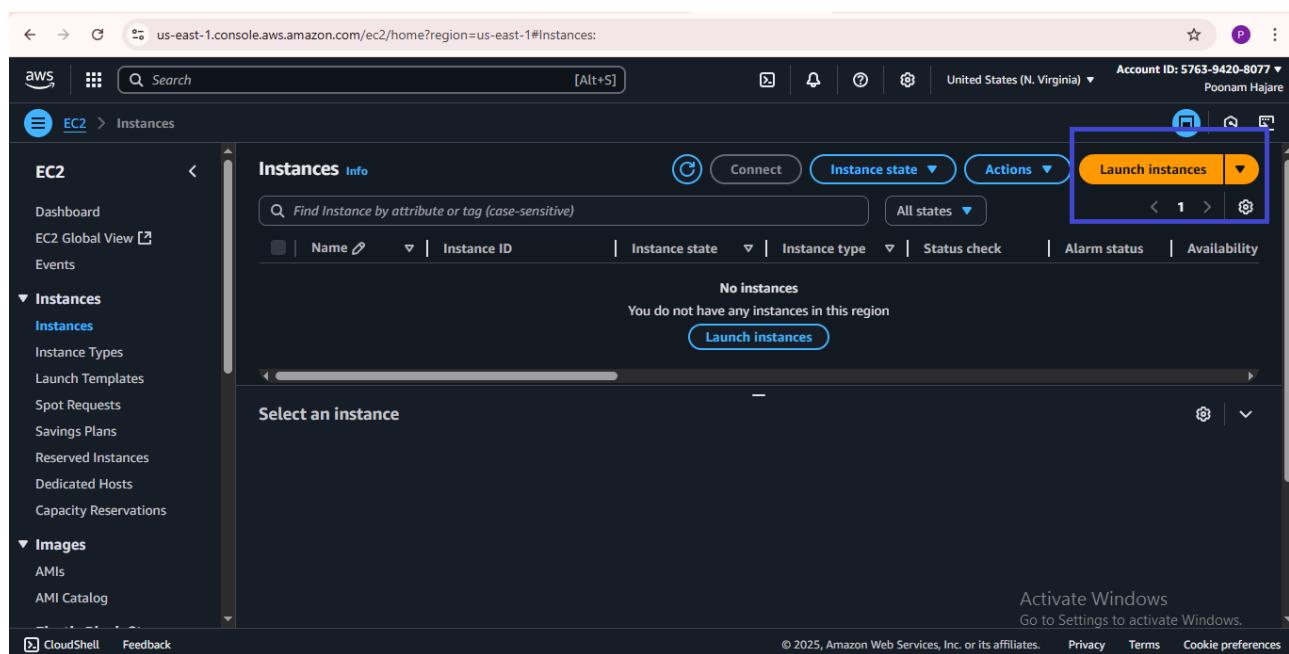
A simple, step-by-step guide to deploying a static website on **an Amazon Linux EC2 instance** using **Apache HTTP Server** (httpd).

Prerequisites

1. An **AWS Account** and appropriate permission
2. A **key pair** (.pem file) for SSH access.
3. A basic static website (HTML, CSS, JS files) ready to deploy.

Step1: Launch an EC2 Instance

1. Sign in to the **AWS Management Console** and go to the **EC2 Dashboard**.
2. Click **Launch Instance**.



3. Choose the **Amazon Linux 2 AMI**.

4. Select an instance type (e.g., **t3.micro** for free tier).

5. select the key pair you have created.

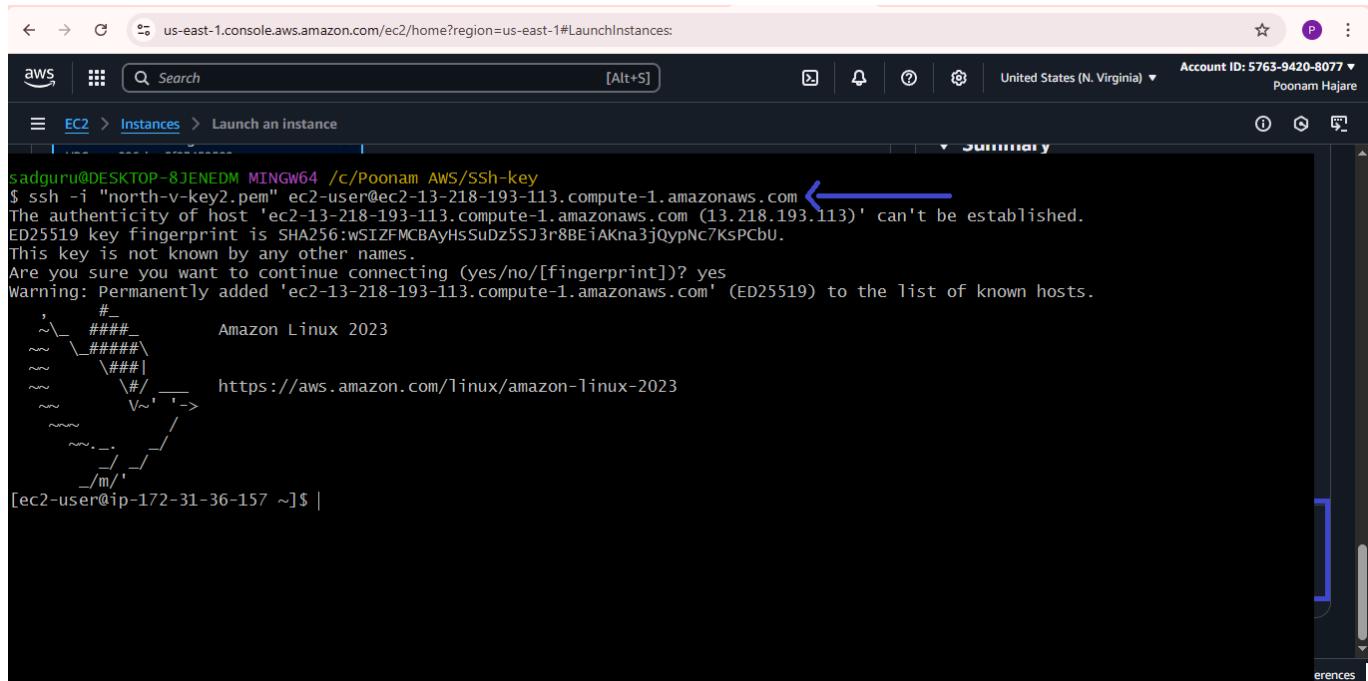
6. In **Security Group**, allow:

- SSH (port 22) from your IP.
- HTTP (port 80) from anywhere.

7. Launch the instance with your key pair.

Step 2: Connect to the Instance

Use your terminal (Git bash)to connect:
`ssh -i /path/to/your-key.pem ec2-user@YOUR_EC2_PUBLIC_DNS`

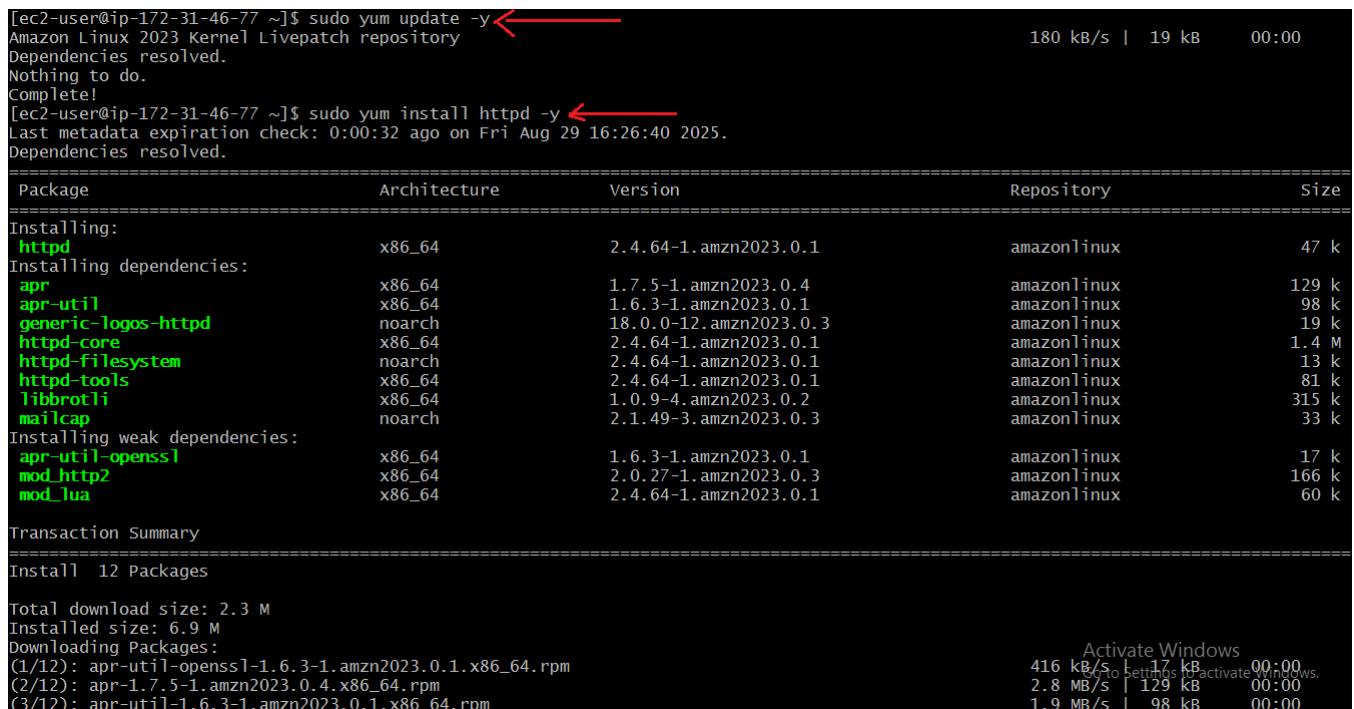


```
sadguru@DESKTOP-8JENEDM MINGW64 /c/Poonam AWS/SSH-key
$ ssh -i "north-v-key2.pem" ec2-user@ec2-13-218-193-113.compute-1.amazonaws.com
The authenticity of host 'ec2-13-218-193-113.compute-1.amazonaws.com (13.218.193.113)' can't be established.
ED25519 key fingerprint is SHA256:wSI2fMCBayHsSuDz55j3r8BE1Akna3jQypNc7KSPCbU.
This key is not known by any other names.
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added 'ec2-13-218-193-113.compute-1.amazonaws.com' (ED25519) to the list of known hosts.

[ec2-user@ip-172-31-36-157 ~]$ |
```

Step 3: Update your system and Install Apache webserver

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



```
[ec2-user@ip-172-31-46-77 ~]$ sudo yum update -y
Amazon Linux 2023 Kernel Livepatch repository
Dependencies resolved.
Nothing to do.
Complete!
[ec2-user@ip-172-31-46-77 ~]$ sudo yum install httpd -y
Last metadata expiration check: 0:00:32 ago on Fri Aug 29 16:26:40 2025.
Dependencies resolved.

=====
Package           Architecture     Version          Repository      Size
=====
Installing:
httpd            x86_64          2.4.64-1.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.amzn2023.0.1    amazonlinux   98 k
generic-logos-httpd    noarch        18.0.0-12.amzn2023.0.3  amazonlinux   19 k
httpd-core       x86_64          2.4.64-1.amzn2023.0.1    amazonlinux   1.4 M
httpd-filesystem noarch        2.4.64-1.amzn2023.0.1    amazonlinux   13 k
httpd-tools      x86_64          2.4.64-1.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.amzn2023.0.1    amazonlinux   17 k
mod_http2        x86_64          2.0.27-1.amzn2023.0.3    amazonlinux   166 k
mod_lua          x86_64          2.4.64-1.amzn2023.0.1    amazonlinux   60 k

Transaction Summary
=====
Install 12 Packages

Total download size: 2.3 M
Installed size: 6.9 M
Downloading Packages:
(1/12): apr-util-openssl-1.6.3-1.amzn2023.0.1.x86_64.rpm 416 kB/s | 17 kB 00:00
(2/12): apr-1.7.5-1.amzn2023.0.4.x86_64.rpm 2.8 MB/s | 129 kB 00:00
(3/12): apr-util-1.6.3-1.amzn2023.0.1.x86_64.rpm 1.9 MB/s | 98 kB 00:00
```

Step 4: Start,enable and status the Apache Service

```
sudo systemctl start httpd
sudo systemctl enable httpd
sudo systemctl status httpd
```

```
[ec2-user@ip-172-31-46-77 ~]$ sudo systemctl start httpd
[ec2-user@ip-172-31-46-77 ~]$ sudo systemctl enable httpd
Created symlink /etc/systemd/system/multi-user.target.wants/httpd.service → /usr/lib/systemd/system/httpd.service.
[ec2-user@ip-172-31-46-77 ~]$ 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 Fri 2025-08-29 16:34:54 UTC; 27s ago
     Docs: man:httpd.service(8)
 Main PID: 25641 (httpd)
   Status: "Total requests: 0; Idle/Busy workers 100/0;Requests/sec: 0; Bytes served/sec: 0 B/sec"
    Tasks: 177 (limit: 1057)
   Memory: 13.3M
      CPU: 81ms
     CGrouP: /system.slice/httpd.service
             └─25641 /usr/sbin/httpd -DFOREGROUND
                 ├─25642 /usr/sbin/httpd -DFOREGROUND
                 ├─25643 /usr/sbin/httpd -DFOREGROUND
                 ├─25644 /usr/sbin/httpd -DFOREGROUND
                 └─25645 /usr/sbin/httpd -DFOREGROUND

Aug 29 16:34:54 ip-172-31-46-77.ec2.internal systemd[1]: Starting httpd.service - The Apache HTTP Server...
Aug 29 16:34:54 ip-172-31-46-77.ec2.internal httpd[25641]: server configured, listening on: port 80
Aug 29 16:34:54 ip-172-31-46-77.ec2.internal systemd[1]: Started httpd.service - The Apache HTTP Server.
[ec2-user@ip-172-31-46-77 ~]$ |
```

INSTALLED SIZE: 0.9 M
 Downloading Packages:
 (1/12): apr-util-openssl-1.6.3-1.amzn2023.0.1.x86_64.rpm 416 kB/s | 17 kB 00:00
 (2/12): apr-1.7.5-1.amzn2023.0.4.x86_64.rpm 2.8 MB/s | 129 kB 00:00
 (3/12): apr-util-1.6.3-1.amzn2023.0.1.x86_64.rpm 1.9 MB/s | 98 kB 00:00
 Activate Windows

Step 5: Create Your First Web Page

1.Go to the web root directory:

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

2.Create an index.html file:

```
sudo vim index.html
```

```
[ec2-user@ip-172-31-46-77 ~]$ cd /var/www/html
[ec2-user@ip-172-31-46-77 html]$ sudo vim index.html
[ec2-user@ip-172-31-46-77 html]$ |
```

3.Add the following content:

```
<html>
<head>
<title>My First Web Server</title>
</head>
<body>
<h1>Hello, World! </h1>
<p>This is my first web server on Linux.</p>
</body>
</html>
```

```
<html>
  <head>
    <title>My First Web Server</title>
  </head>
  <body>
    <h1>Hello, World! </h1>
    <p>This is my first web server on Linux.</p>
  </body>
</html>

```
-- INSERT --
```

  Activate
  Go to Settings
```

Step 6: Check results

Open your browser and enter your server's public IP



Hello, World!

This is my first web server on Linux.

Conclusion:

I successfully hosted my first website on Linux using Apache.