

CLOUD SECURITY & MANAGEMENT  
LAB

Name: Harsh Ranjan

SAP ID: 500097019

Roll no: R2142211262

SUBMISSION TO:

Ms. Avita Katal

## **Experiment 2: To Install web server in virtual machine (On local machine and AWS EC2)**

### **1. ON UBUNTU VM**

**STEP 1: To install a web server on a virtual machine (VM) and serve a simple HTML webpage**, you can follow these general steps. I'll outline the process assuming you're using a Linux-based VM and the Apache web server, which is one of the most used web servers.

- Set Up a Virtual Machine:
- Install a Linux distribution such as Ubuntu Server on your VM.
- Ensure your VM has network connectivity.

#### **STEP 2: Install Apache Web Server:**

- Update your package index: `sudo apt update`
- Install Apache: `sudo apt install apache2`

```
harsranjan@harsranjan-virtual-machine:~$ sudo apt install apache2
[sudo] password for harsranjan:
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
apache2 is already the newest version (2.4.52-1ubuntu4.7).
0 upgraded, 0 newly installed, 0 to remove and 47 not upgraded.
harsranjan@harsranjan-virtual-machine:~$
```

#### **STEP 3: Create a Simple HTML Page and edit it by nano editor:**

- Create a directory for your HTML files: `sudo mkdir /var/www/html`
- Create an HTML file inside that directory. You can use a text editor like Nano or Vim. For example:
- `sudo nano /var/www/html/index.html`

```
harsranjan@harsranjan-virtual-machine: /var/www/html
Enabling module status.
Enabling module reqtimeout.
Enabling conf charset.
Enabling conf localized-error-pages.
Enabling conf other-vhosts-access-log.
Enabling conf security.
Enabling conf serve-cgi-bin.
Enabling site 000-default.
Created symlink /etc/systemd/system/multi-user.target.wants/apache2.service → /lib/systemd/system/apache2.service.
Created symlink /etc/systemd/system/multi-user.target.wants/apache-htcacheclean.service → /lib/systemd/system/apache-htcacheclean.service.
Processing triggers for ufw (0.36.1-4ubuntu0.1) ...
Processing triggers for man-db (2.10.2-1) ...
Processing triggers for libc-bin (2.35-0ubuntu3.6) ...
harsranjan@harsranjan-virtual-machine:~$ ls
Desktop Documents Downloads Music Pictures Public snap Templates Videos
harsranjan@harsranjan-virtual-machine:~$ cd ..
harsranjan@harsranjan-virtual-machine:/$ ls
harsranjan
harsranjan@harsranjan-virtual-machine:/$ cd ..
harsranjan@harsranjan-virtual-machine:/$ ls
bin cdrom etc lib lib64 lost+found mnt proc run snap swapfile tmp var
boot dev home lib32 libx32 media opt root sbin srv sys usr
harsranjan@harsranjan-virtual-machine:/$ cd var
harsranjan@harsranjan-virtual-machine:/var$ ls
backups cache crash lib local lock log mail metrics opt run snap spool tmp www
harsranjan@harsranjan-virtual-machine:/var$ cd www
harsranjan@harsranjan-virtual-machine:/var/www$ ls
html
harsranjan@harsranjan-virtual-machine:/var/www$ cd html
harsranjan@harsranjan-virtual-machine:/var/www/html$ ls
index.html
harsranjan@harsranjan-virtual-machine:/var/www/html$ nano index.html

GNU nano 6.2 index.html *
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN" "http://www.w3.org/1999/xhtml">
<html xmlns="http://www.w3.org/1999/xhtml">
<!--
  Modified from the Debian original for Ubuntu
  Last updated: 2022-03-22
  See: https://launchpad.net/bugs/1966004
-->
<head>
<meta http-equiv="Content-Type" content="text/html; charset=UTF-8" />
<title>Apache2 Ubuntu Default Page: It works</title>
<body>
<h1>Welcome to My Webpage</h1>
<p>My name is Harsh Ranjan.</p>
</body>

<style type="text/css" media="screen">
* {
margin: 0px 0px 0px 0px;
padding: 0px 0px 0px 0px;
}

^G Help      ^O Write Out ^W Where Is  ^K Cut       ^T Execute  ^C Location
^X Exit      ^R Read File ^_ Replace   ^U Paste     ^J Justify  ^_ Go To Line
```

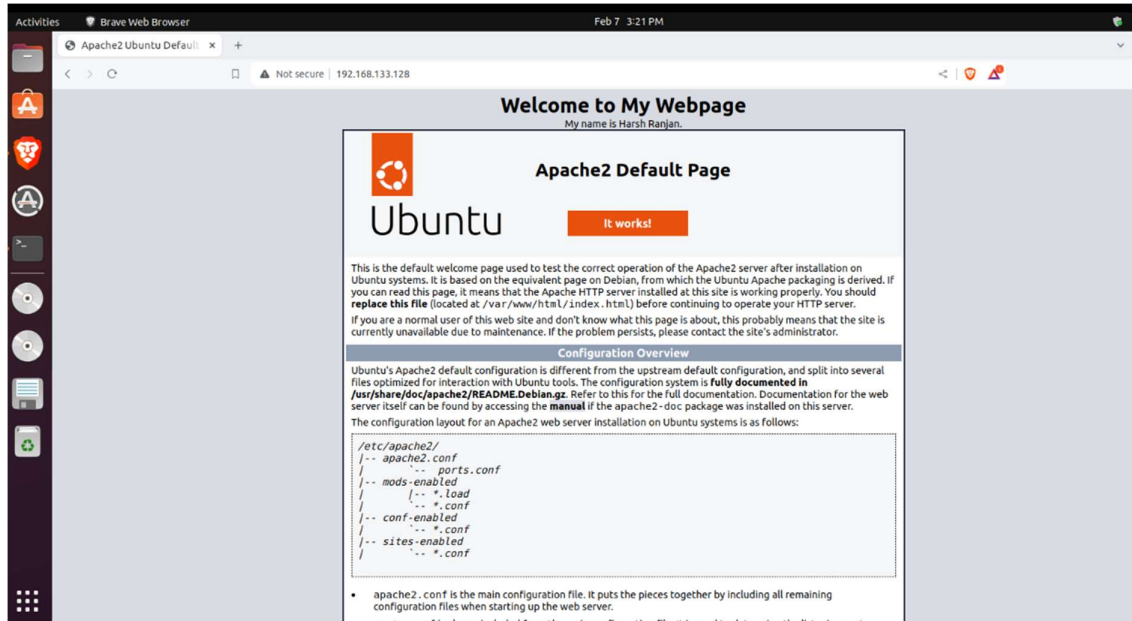
#### STEP 4: Restart Apache:

After making changes, it's good to restart Apache to apply the changes:

- `sudo systemctl restart apache2`

## STEP 5: Access Your Webpage:

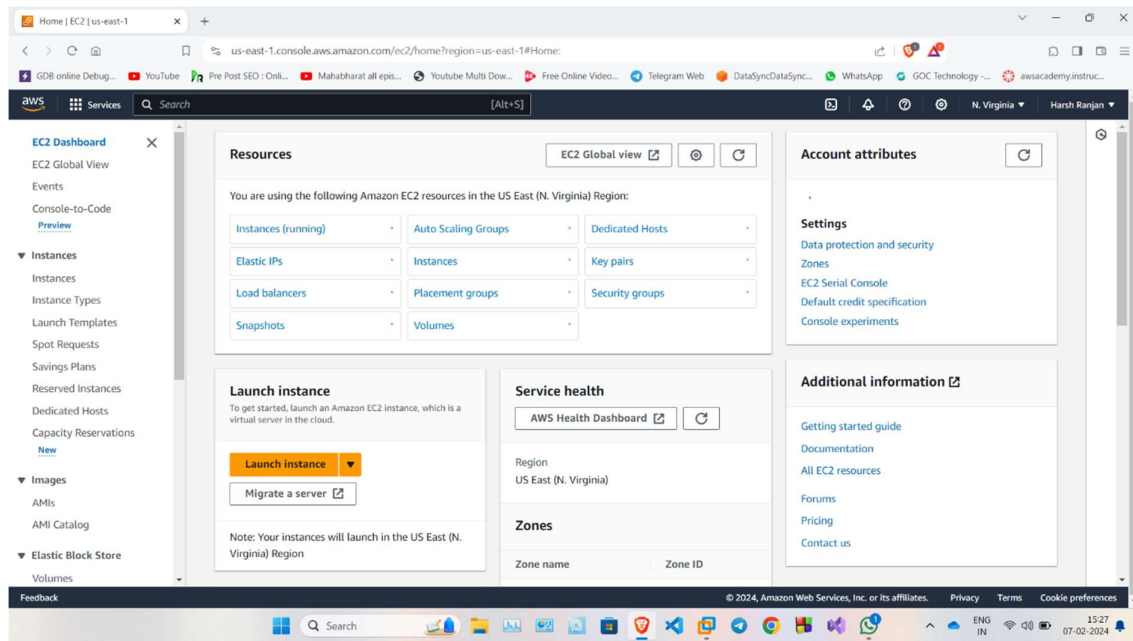
In a web browser, enter the IP address of your VM. You should see the webpage you created.



## 2. ON AWS EC2 Instance

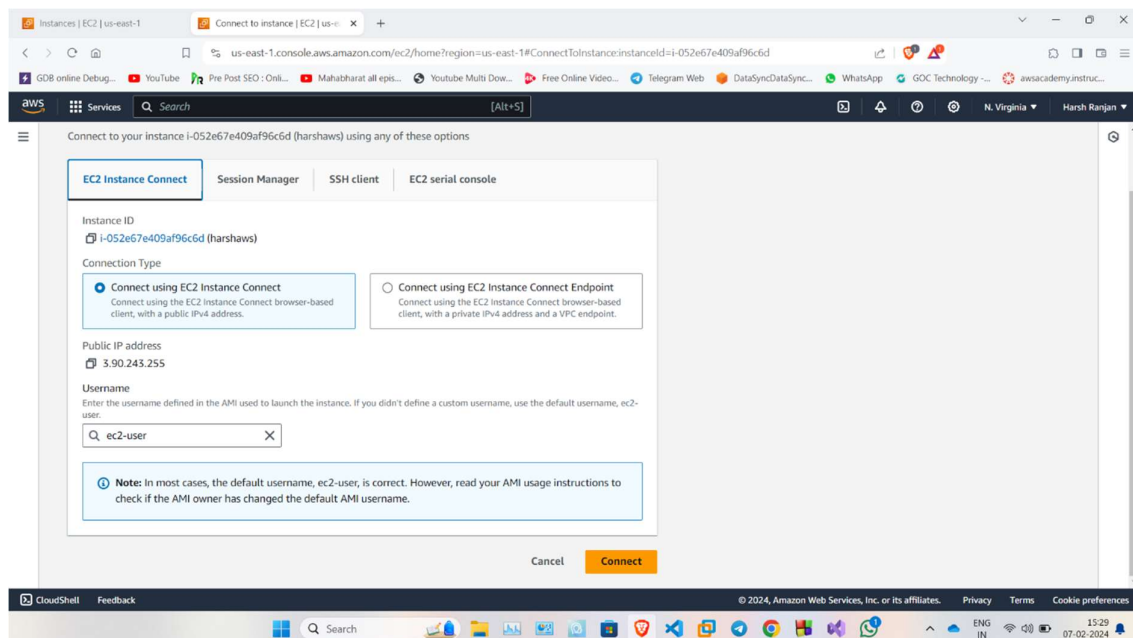
### STEP 1: Launch an EC2 Instance:

- Log in to the AWS Management Console.
- Go to the EC2 dashboard.
- Click on "Launch Instance" to create a new instance.



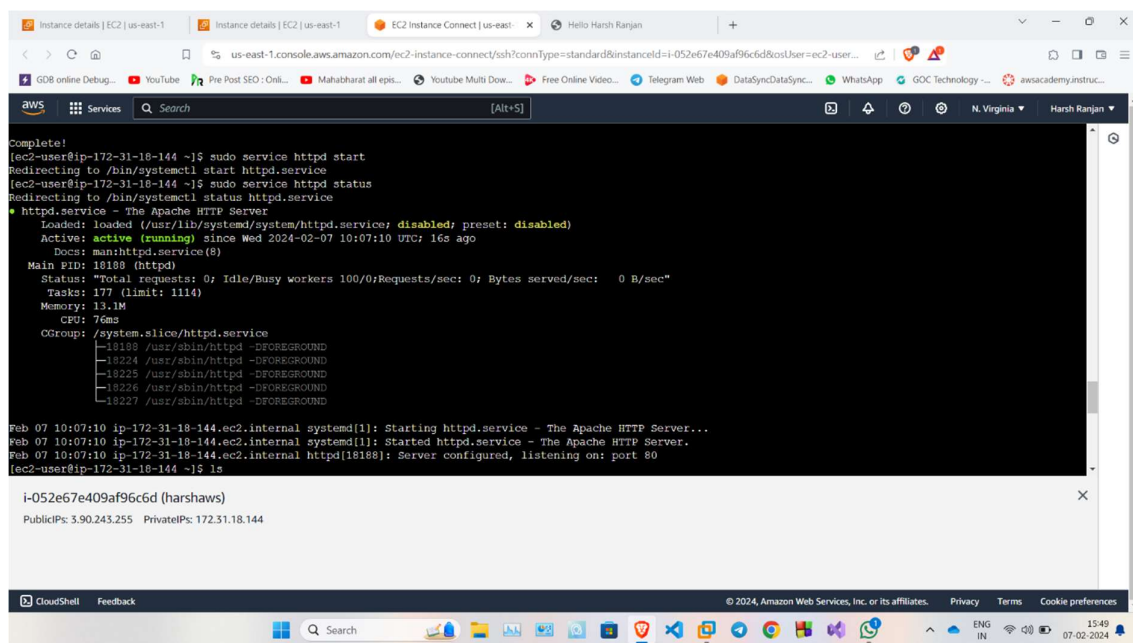
## STEP 2: Connect to Your EC2 Instance:

- Once your instance is running, connect to it using SSH. You can use a tool like PuTTY (on Windows) or Terminal (on macOS/Linux) to SSH into your instance.
- Use the private key associated with your instance to authenticate.



### STEP 3: Install Apache:

- Update the package index:  
`sudo yum update`
- Install Apache:  
`sudo yum install httpd`
- Start the Apache service:  
`sudo service httpd start`



```
Complete!
[ec2-user@ip-172-31-18-144 ~]$ sudo service httpd start
Redirecting to /bin/systemctl start httpd.service
[ec2-user@ip-172-31-18-144 ~]$ sudo service httpd status
Redirecting to /bin/systemctl status httpd.service
* httpd.service - The Apache HTTP Server
   Loaded: loaded (/usr/lib/systemd/system/httpd.service; disabled; preset: disabled)
   Active: active (running) since Wed 2024-02-07 10:07:10 UTC; 16s ago
     Docs: man:httpd.service(8)
  Main PID: 18188 (httpd)
    Status: "Total requests: 0; Idle/Busy workers 100/0;Requests/sec: 0; Bytes served/sec: 0 B/sec"
    Tasks: 177 (limit: 1114)
   Memory: 13.1M
      CPU: 76ms
   CGroup: /system.slice/httpd.service
           └─18188 /usr/sbin/httpd -DFOREGROUND
             └─18224 /usr/sbin/httpd -DFOREGROUND
               └─18225 /usr/sbin/httpd -DFOREGROUND
                 └─18226 /usr/sbin/httpd -DFOREGROUND
                   └─18227 /usr/sbin/httpd -DFOREGROUND

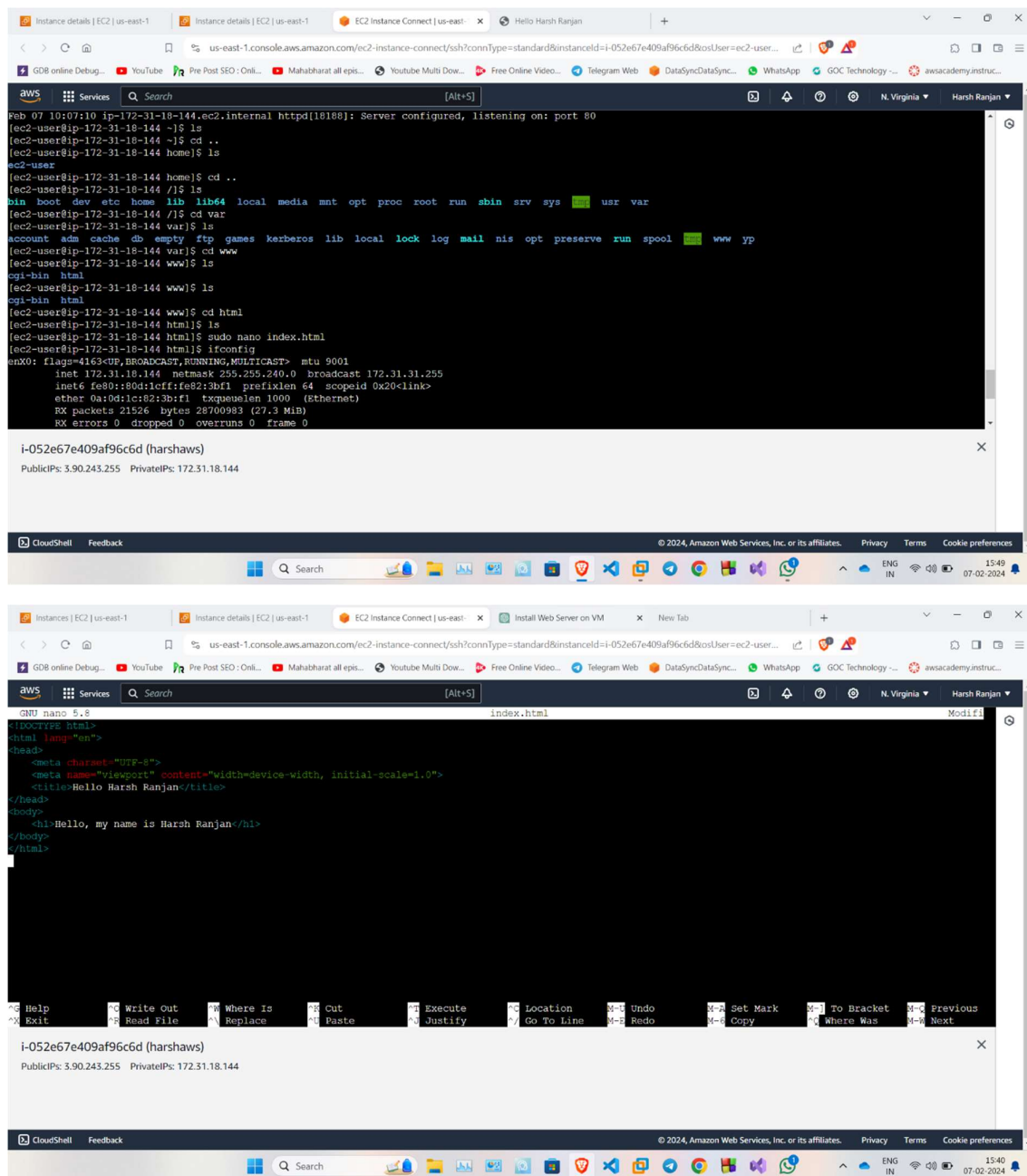
Feb 07 10:07:10 ip-172-31-18-144.ec2.internal systemd[1]: Starting httpd.service - The Apache HTTP Server...
Feb 07 10:07:10 ip-172-31-18-144.ec2.internal systemd[1]: Started httpd.service - The Apache HTTP Server.
Feb 07 10:07:10 ip-172-31-18-144.ec2.internal httpd[18188]: Server configured, listening on: port 80
[ec2-user@ip-172-31-18-144 ~]$ ls
```

i-052e67e409af96c6d (harshawshaws)  
PublicIPs: 3.90.243.255 PrivateIPs: 172.31.18.144

Open a web browser and enter the public IP address of your EC2 instance. You should see the default Apache page indicating that Apache is running.

### STEP 4: Serve Your Own Web Content:

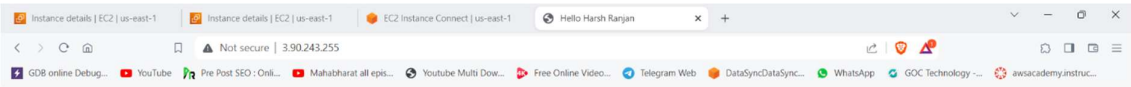
- Create your HTML files or copy your website files to the appropriate directory. The default directory for Apache on Amazon Linux is **`/var/www/html/`** and on Ubuntu is **`/var/www/html/`**.



## STEP 5: Access Your Website:

- Enter the public IP address of your EC2 instance into a web browser. You should now see your website served by Apache.





**Hello, my name is Harsh Ranjan**