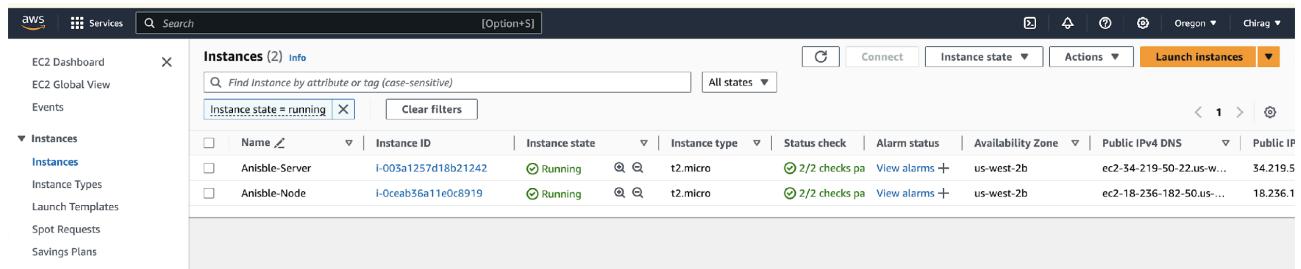


**Shivam Arora**  
**500094155**  
**R2142210733**

## **SPCM LAB** **EXPERIMENT - 11**

### **Step 1 :- Setup two EC2 Instances**



The screenshot shows the AWS EC2 Instances page. The left sidebar includes links for EC2 Dashboard, EC2 Global View, Events, Instances (selected), Instance Types, Launch Templates, Spot Requests, and Savings Plans. The main content area displays a table titled 'Instances (2) Info' with the following data:

Name	Instance ID	Instance state	Instance type	Status check	Alarm status	Availability Zone	Public IPv4 DNS	Public IP
Ansible-Server	i-005a1257d18b21242	Running	t2.micro	2/2 checks pa	<a href="#">View alarms</a>	us-west-2b	ec2-34-219-50-22.us-w...	34.219.5
Ansible-Node	i-0ceab36a11e0c8919	Running	t2.micro	2/2 checks pa	<a href="#">View alarms</a>	us-west-2b	ec2-18-236-182-50.us-...	18.236.1

### **Step 2 :- Install Ansible on both Instances**

```

=====
Install 1 Package (+3 Dependent packages)

Total download size: 17 M
Installed size: 105 M
Is this ok [y/d/N]: y
Downloading packages:
(1/4): ansible-2.9.27-1.el7.noarch.rpm | 17 MB 00:00:00
(2/4): python-paramiko-2.1.1-0.10.el7.noarch.rpm | 269 kB 00:00:00
(3/4): python2-httpplib2-0.18.1-3.el7.noarch.rpm | 125 kB 00:00:00
(4/4): sshpass-1.06-1.el7.x86_64.rpm | 21 kB 00:00:00
                                           37 MB/s | 17 MB 00:00:00

Total
Running transaction check
Running transaction test
Transaction test succeeded
Running transaction
  Installing : python2-httpplib2-0.18.1-3.el7.noarch
  Installing : sshpass-1.06-1.el7.x86_64
  Installing : python-paramiko-2.1.1-0.10.el7.noarch
  Installing : ansible-2.9.27-1.el7.noarch
  Verifying   : python-paramiko-2.1.1-0.10.el7.noarch
  Verifying   : sshpass-1.06-1.el7.x86_64
  Verifying   : python2-httpplib2-0.18.1-3.el7.noarch
  Verifying   : ansible-2.9.27-1.el7.noarch
                                           1
                                           2
                                           3
                                           4
                                           1
                                           2
                                           3
                                           4

Installed:
  ansible.noarch 0:2.9.27-1.el7

Dependency Installed:
  python-paramiko.noarch 0:2.1.1-0.10.el7           python2-httpplib2.noarch 0:0.18.1-3.el7
                                                       sshpass.x86_64 0:1.06-1.el7

Complete!
[root@ip-172-31-30-251 ~]#

```

## Step 3 :- Configure /etc/ansible/hosts file on ansible server instance

```

# This is the default ansible 'hosts' file.
#
# It should live in /etc/ansible/hosts
#
# - Comments begin with the '#' character
# - Blank lines are ignored
# - Groups of hosts are delimited by [header] elements
# - You can enter hostnames or ip addresses
# - A hostname/ip can be a member of multiple groups

# Ex 1: Ungrouped hosts, specify before any group headers.
#       A hostname/ip can be a member of multiple group
[unspec]
192.236.182.50
## green.example.com
## blue.example.com
## 192.168.100.1
## 192.168.100.10

# Ex 2: A collection of hosts belonging to the 'webservers' group
## [webservers]
## alpha.example.org
## beta.example.org
## 192.168.1.100
## 192.168.1.110

# If you have multiple hosts following a pattern you can specify
# them like this:
## www[001:006].example.com

# Ex 3: A collection of database servers in the 'dbservers' group
## [dbservers]
## /etc/ansible/hosts" [readonly] 46L, 1088B

```

## Step 4:- Create a ansible user and password and update the root permissions for both the users inside visudo file

```
# Defaults env_keep += "HOME"
Defaults secure_path = /sbin:/bin:/usr/sbin:/usr/bin

## Next comes the main part: which users can run what software on
## which machines (the sudoers file can be shared between multiple
## systems).
## Syntax:
##
##     user    MACHINE=COMMANDS
##
## The COMMANDS section may have other options added to it.
##
## Allow root to run any commands anywhere
root    ALL=(ALL)        ALL
ansible ALL=(ALL)        NOPASSWD: ALL
## Allows members of the 'sys' group to run networking, software,
## service management apps and more.
# tsys ALL = NETWORKING, SOFTWARE, SERVICES, STORAGE, DELEGATING, PROCESSES, LOCATE, DRIVERS
## Allows people in group wheel to run all commands
wheel   ALL=(ALL)        ALL

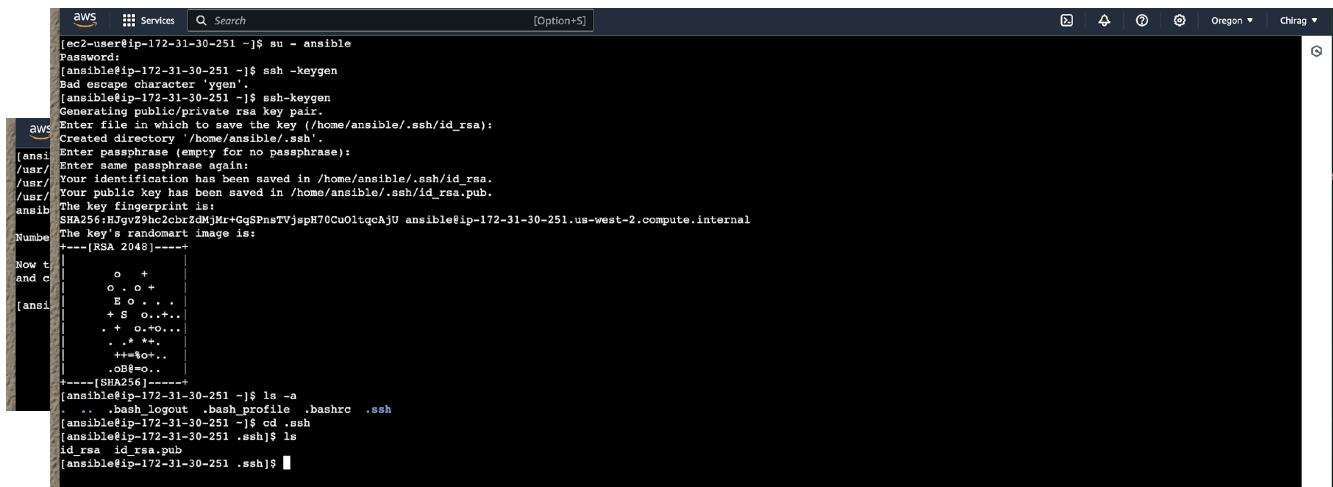
## Same thing without a password
# twheel  ALL=(ALL)        NOPASSWD: ALL
## Allows members of the users group to mount and umount the
## cdrom as root
# tusers  ALL=/sbin/mount /mnt/cdrom, /sbin/umount /mnt/cdrom
## Allows members of the users group to shutdown this system
# tusers  localhost=/sbin/shutdown -h now

## Read drop-in files from /etc/sudoers.d (the # here does not mean a comment)
#includedir /etc/sudoers.d
"/etc/sudoers.tmp" 120L, 4361B
```

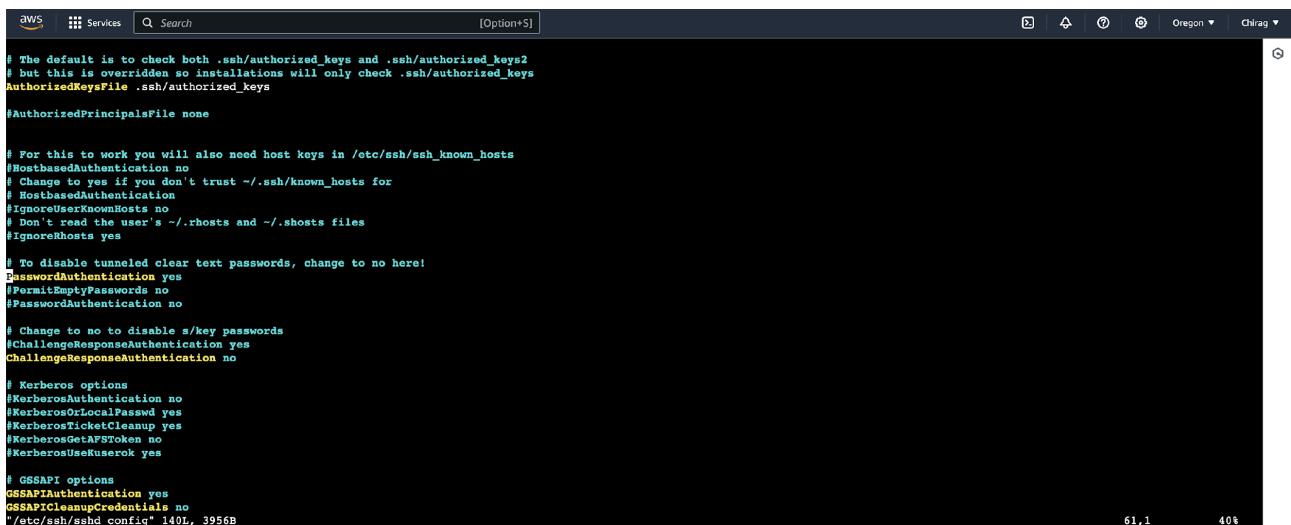
104, 33 Bot

## Step 5 Create a ssh key in ansible server

## Step 6: Copy the ssh key to connect with Ansible node



```
[ec2-user@ip-172-31-30-251 ~]$ su - ansible
Password:
[ansible@ip-172-31-30-251 ~]$ ssh-keygen
Bad escape character 'ygen'.
[ansible@ip-172-31-30-251 ~]$ ssh-keygen
Generating public/private rsa key pair.
Enter file in which to save the key (/home/ansible/.ssh/id_rsa):
Created directory '/home/ansible/.ssh'.
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in /home/ansible/.ssh/id_rsa.
Your public key has been saved in /home/ansible/.ssh/id_rsa.pub.
The key fingerprint is:
SHA256:HJgvZ9hc2cbrxDNjMr+GgSPnsTVjspH70Cu0ltqcAJU ansible@ip-172-31-30-251.us-west-2.compute.internal
Number of bits: 2048
Now to add the public key to your .ssh/authorized_keys file.
-----[RSA 2048]-----
Now to add the public key to your .ssh/authorized_keys file.
-----[SHA256]-----
[ansible@ip-172-31-30-251 ~]$ ls -a
. .. .bash_logout .bash_profile .bashrc .ssh
[ansible@ip-172-31-30-251 ~]$ cd .ssh
[ansible@ip-172-31-30-251 .ssh]$ ls
id_rsa id_rsa.pub
[ansible@ip-172-31-30-251 .ssh]$
```



```
# The default is to check both .ssh/authorized_keys and .ssh/authorized_keys2
# but this is overridden so installations will only check .ssh/authorized_keys
AuthorizedKeysFile .ssh/authorized_keys

#AuthorizedPrincipalsFile none

# For this to work you will also need host keys in /etc/ssh/ssh_known_hosts
#HostbasedAuthentication no
# Change to yes if you don't trust ~/.ssh/known_hosts for
# HostbasedAuthentication
#IgnoreUserKnownHosts no
# Don't read the user's ~/.rhosts and ~/.shosts files
#IgnoreRhosts yes

# To disable tunneled clear text passwords, change to no here!
#PasswordAuthentication yes
#PermitEmptyPasswords no
#PasswordAuthentication no

# Change to no to disable s/key passwords
#ChallengeResponseAuthentication yes
ChallengeResponseAuthentication no

# Kerberos options
#KerberosAuthentication no
#KerberosOrLocalPasswd yes
#KerberosTicketCleanup yes
#KerberosGetTGTFromKDC no
#KerberosUseKuserok yes

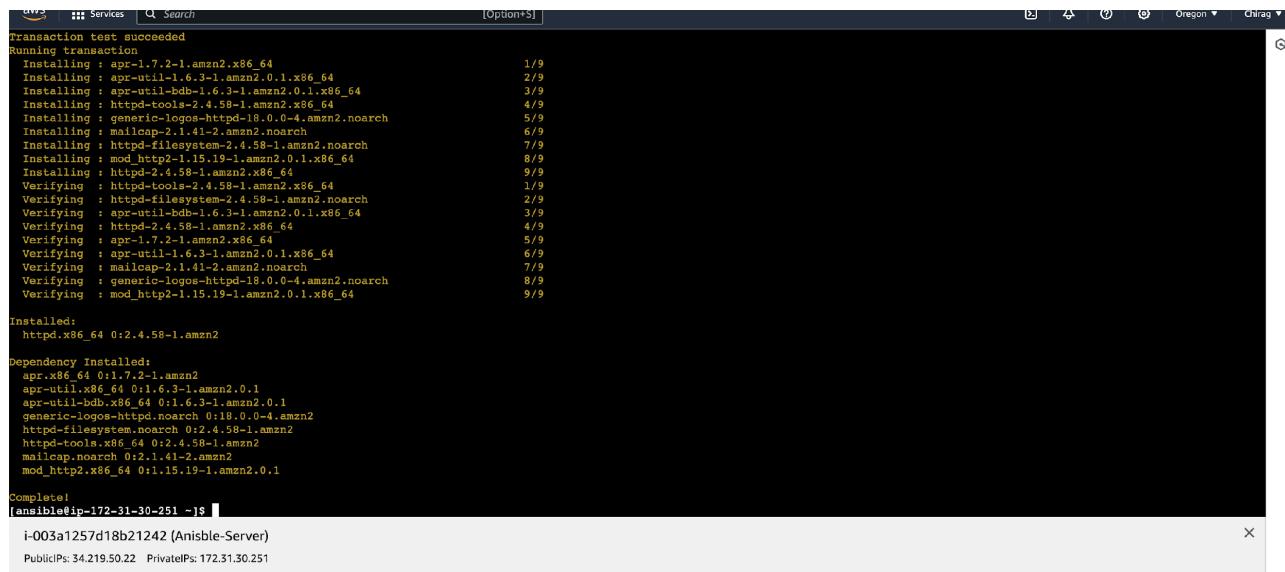
# GSSAPI options
#GSSAPIAuthentication yes
#GSSAPICleanupCredentials no
"/etc/ssh/sshd_config" 140L, 3956B
```

## Step 7 :- Connect with Node instance from Server instance

```
Last login: Fri Apr 26 10:04:34 2024 from ec2-18-237-140-164.us-west-2.compute.amazonaws.com
[ec2-user@ip-172-31-30-251 ~]$ su - ansible
Password:
Last login: Fri Apr 26 10:04:51 UTC 2024 on pts/0
[ansible@ip-172-31-30-251 ~]$ ssh 18.236.182.50
Last login: Fri Apr 26 09:52:02 2024 from 34.219.50.22
[ansible@ip-172-31-17-175 ~]$
```

## EXPERIMENT 12

### Step 1 :- Install httpd on ansible server instance

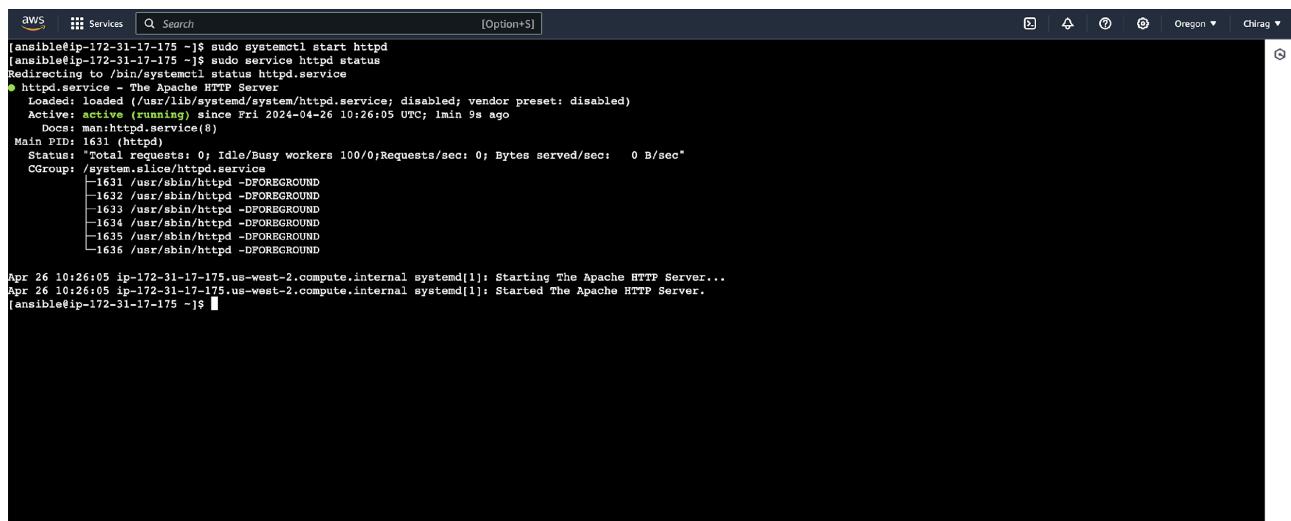


```
aws Services Search [Option+S] Oregon Chirag
Transaction test succeeded
Running transaction
  Installing : apr-1.7.2-1.amzn2.x86_64 1/9
    apr-util-1.6.3-1.amzn2.0.1.x86_64 2/9
    apr-util-bdb-1.6.3-1.amzn2.0.1.x86_64 3/9
    httpd-tools-2.4.58-1.amzn2.x86_64 4/9
    generic-logos-httdp-18.0.0-4.amzn2.noarch 5/9
    mailcap-2.1.41-2.amzn2.noarch 6/9
    httpd-fslemntdm-2.4.58-1.amzn2.noarch 7/9
    mod_http2-1.15.19-1.amzn2.0.1.x86_64 8/9
    httpd-2.4.58-1.amzn2.x86_64 9/9
  Verifying : httpd-tools-2.4.58-1.amzn2.x86_64 1/9
  Verifying : httpd-fslemntdm-2.4.58-1.amzn2.noarch 2/9
  Verifying : apr-util-1.6.3-1.amzn2.0.1.x86_64 3/9
  Verifying : httpd-2.4.58-1.amzn2.x86_64 4/9
  Verifying : apr-1.7.2-1.amzn2.x86_64 5/9
  Verifying : apr-util-1.6.3-1.amzn2.0.1.x86_64 6/9
  Verifying : mailcap-2.1.41-2.amzn2.noarch 7/9
  Verifying : generic-logos-httdp-18.0.0-4.amzn2.noarch 8/9
  Verifying : mod_http2-1.15.19-1.amzn2.0.1.x86_64 9/9
Installed:
  httpd.x86_64 0:2.4.58-1.amzn2

Dependency Installed:
  apr.x86_64 0:1.7.2-1.amzn2
  apr-util.x86_64 0:1.6.3-1.amzn2.0.1
  apr-util-bdb.x86_64 0:1.6.3-1.amzn2.0.1
  generic-logos-httdp.noarch 0:18.0.0-4.amzn2
  httpd-fslemntdm.noarch 0:2.4.58-1.amzn2
  httpd-tools.x86_64 0:2.4.58-1.amzn2
  mailcap.noarch 0:2.1.41-2.amzn2
  mod_http2.x86_64 0:1.15.19-1.amzn2.0.1

Complete!
[ansible@ip-172-31-30-251 ~]$ i-003a1257d18b21242 (Ansible-Server)
PublicIPs: 34.219.50.22 PrivateIPs: 172.31.30.251
```

### Step 2 :- Start the httpd service on node instance



```
aws Services Search [Option+S] Oregon Chirag
[ansible@ip-172-31-17-175 ~]$ sudo systemctl start httpd
[ansible@ip-172-31-17-175 ~]$ 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; vendor preset: disabled)
     Active: active (running) since Fri Apr 26 10:26:05 UTC 2024; 1min 9s ago
       Docs: man:systemctl(8)
 Main PID: 1631 (httpd)
   Status: "Total requests: 0; Idle/Busy workers 100/0;Requests/sec: 0; Bytes served/sec: 0 B/sec"
    CGroup: /system.slice/httpd.service
           ├─1631 /usr/sbin/httpd -DFOREGROUND
           ├─1632 /usr/sbin/httpd -DFOREGROUND
           ├─1633 /usr/sbin/httpd -DFOREGROUND
           ├─1634 /usr/sbin/httpd -DFOREGROUND
           ├─1635 /usr/sbin/httpd -DFOREGROUND
           └─1636 /usr/sbin/httpd -DFOREGROUND

Apr 26 10:26:05 ip-172-31-17-175.us-west-2.compute.internal systemd[1]: Starting The Apache HTTP Server...
Apr 26 10:26:05 ip-172-31-17-175.us-west-2.compute.internal systemd[1]: Started The Apache HTTP Server.
[ansible@ip-172-31-17-175 ~]$
```

## Step 3 :- Home Page can be seen on Node public IP



This page is used to test the proper operation of the Apache HTTP server after it has been installed. If you can read this page, it means that the Apache HTTP server installed at this site is working properly.

### If you are a member of the general public:

The fact that you are seeing this page indicates that the website you just visited is either experiencing problems, or is undergoing routine maintenance.

If you would like to let the administrators of this website know that you've seen this page instead of the page you expected, you should send them e-mail. In general, mail sent to the name "webmaster" and directed to the website's domain should reach the appropriate person.

For example, if you experienced problems while visiting www.example.com, you should send e-mail to "webmaster@example.com".

### If you are the website administrator:

You may now add content to the directory /var/www/html/. Note that until you do so, people visiting your website will see this page, and not your content. To prevent this page from ever being used, follow the instructions in the file /etc/httpd/conf.d/welcome.conf.

You are free to use the image below on web sites powered by the Apache HTTP Server:



## Step 4 :- Remove the httpd manually using command

```
AWS Services Search [Option+S] Oregon ▾ Chicago ▾
--> Package mod_http2.x86_64 0:1.15.19-1.amzn2.0.1 will be erased
--> Finished Dependency Resolution
Dependencies Resolved

Package          Arch      Version           Repository      Size
=====
Removing:
httpd           x86_64   2.4.58-1.amzn2          @amzn2-core    4.2 M
Removing for dependencies:
mod_http2       x86_64   1.15.19-1.amzn2.0.1    @amzn2-core    382 k

Transaction Summary
=====
Remove 1 Package (+1 Dependent package)

Installed size: 4.5 M
Downloading packages:
Running transaction check
Running transaction test
transaction test succeeded
Running transaction
  Erasing   : httpd-2.4.58-1.amzn2.x86_64                                1/2
  Erasing   : mod_http2-1.15.19-1.amzn2.0.1.x86_64                         2/2
  Verifying : mod_http2-1.15.19-1.amzn2.0.1.x86_64                         1/2
  Verifying : httpd-2.4.58-1.amzn2.x86_64                                2/2

Removed:
httpd.x86_64 0:2.4.58-1.amzn2

Dependency Removed:
mod_http2.x86_64 0:1.15.19-1.amzn2.0.1

Complete!
[ansible@ip-172-31-30-251 ~]$
```

```
AWS Services Search [Option+S] Oregon Chirag ⓘ
[ansible@ip-172-31-17-175 ~]$ sudo systemctl start httpd
[ansible@ip-172-31-17-175 ~]$ 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; vendor preset: disabled)
     Docs: man:httpd.service(8)
   Active: active (running) since Fri 2024-04-26 10:26:05 UTC; 1min 9s ago
     Docs: man:httpd.service(8)
Main PID: 1631 (httpd)
   Status: "Total requests: 0; Idle/Busy workers 100/0;Requests/sec: 0; Bytes served/sec: 0 B/sec"
   CGroup: /system.slice/httpd.service
           ├─1631 /usr/sbin/httpd -DFOREGROUND
           ├─1632 /usr/sbin/httpd -DFOREGROUND
           ├─1633 /usr/sbin/httpd -DFOREGROUND
           ├─1634 /usr/sbin/httpd -DFOREGROUND
           ├─1635 /usr/sbin/httpd -DFOREGROUND
           ├─1636 /usr/sbin/httpd -DFOREGROUND

Apr 26 10:26:05 ip-172-31-17-175.us-west-2.compute.internal systemd[1]: Starting The Apache HTTP Server...
Apr 26 10:26:05 ip-172-31-17-175.us-west-2.compute.internal systemd[1]: Started The Apache HTTP Server.
[ansible@ip-172-31-17-175 ~]$ sudo service httpd status
Redirecting to /bin/systemctl status httpd.service
Unit httpd.service could not be found.
[ansible@ip-172-31-17-175 ~]$ which httpd
/usr/bin/which: no httpd in (/usr/local/bin:/usr/bin:/usr/local/sbin:/usr/sbin:/home/ansible/.local/bin:/home/ansible/bin)
[ansible@ip-172-31-17-175 ~]$
```

## Step 5 :- Using all the task using playbook

```
AWS Services Search [Option+S] Oregon Chirag ⓘ
[ec2-user@ip-172-31-30-251 ~]$ su - ansible
Password:
Last login: Fri Apr 26 10:36:58 UTC 2024 on pts/0
[ansible@ip-172-31-30-251 ~]$ ls
1 ansible1.yml
[ansible@ip-172-31-30-251 ~]$ ansible-playbook playbook1.yml
ERROR! The playbook: playbook1.yml could not be found
[ansible@ip-172-31-30-251 ~]$ ls
1 ansible1.yml
[ansible@ip-172-31-30-251 ~]$ ansible-playbook ansible1.yml

PLAY [upes] ****
*****
TASK [Gathering Facts] ****
[WARNING]: Platform linux on host 18.236.182.50 is using the discovered Python interpreter at '/usr/bin/python', but future installation of another Python interpreter could change this. See https://docs.ansible.com/ansible/2.9/reference_appendices/interpreter_discovery.html for more information.
ok: [18.236.182.50]

TASK [Un-Install HTTPD] ****
ok: [18.236.182.50]

PLAY RECAP ****
18.236.182.50 : ok=2    changed=0    unreachable=0    failed=0    skipped=0    rescued=0    ignored=0
[ansible@ip-172-31-30-251 ~]$
```

```

AWS Services Search [Option+S]
(ansible@ip-172-31-17-175 ~)$ sudo systemctl start httpd
(ansible@ip-172-31-17-175 ~)$ 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; vendor preset: disabled)
   Active: active (running) since Fri 2024-04-26 10:26:05 UTC; 1min 9s ago
     Docs: man:httpd.service(8)
Main PID: 1631 (httpd)
  Status: "Total requests: 0; Idle/Busy workers 100/0;Requests/sec: 0; Bytes served/sec: 0 B/sec"
 CGroup: /system.slice/httpd.service
        └─1631 /usr/sbin/httpd -DFOREGROUND
           ├─1632 /usr/sbin/httpd -DFOREGROUND
           ├─1633 /usr/sbin/httpd -DFOREGROUND
           ├─1634 /usr/sbin/httpd -DFOREGROUND
           ├─1635 /usr/sbin/httpd -DFOREGROUND
           └─1636 /usr/sbin/httpd -DFOREGROUND

Apr 26 10:26:05 ip-172-31-17-175.us-west-2.compute.internal systemd[1]: Starting The Apache HTTP Server...
Apr 26 10:26:05 ip-172-31-17-175.us-west-2.compute.internal systemd[1]: Started The Apache HTTP Server.
(ansible@ip-172-31-17-175 ~)$ sudo service httpd status
Redirecting to /bin/systemctl status httpd.service
Unit httpd.service could not be found.
(ansible@ip-172-31-17-175 ~)$ which httpd
/usr/bin/which: no httpd in (/usr/local/bin:/usr/bin:/usr/local/sbin:/usr/sbin:/home/ansible/.local/bin:/home/ansible/bin)
(ansible@ip-172-31-17-175 ~)$

```

Gmail YouTube Microsoft Office H... JetBrains Account MERN full stack w... 3Blue1Brown Xebia Academy: L... All Bookmarks

## Test Page

This page is used to test the proper operation of the Apache HTTP server after it has been installed. If you can read this page, it means that the Apache HTTP server installed at this site is working properly.

**If you are a member of the general public:**

The fact that you are seeing this page indicates that the website you just visited is either experiencing problems, or is undergoing routine maintenance.

If you would like to let the administrators of this website know that you've seen this page instead of the page you expected, you should send them e-mail. In general, mail sent to the name "webmaster" and directed to the website's domain should reach the appropriate person.

For example, if you experienced problems while visiting [www.example.com](http://www.example.com), you should send e-mail to ["webmaster@example.com"](mailto:webmaster@example.com).

**If you are the website administrator:**

You may now add content to the directory `/var/www/html/`. Note that until you do so, people visiting your website will see this page, and not your content. To prevent this page from ever being used, follow the instructions in the file `/etc/httpd/conf.d/welcome.conf`.

You are free to use the image below on web sites powered by the Apache HTTP Server:

