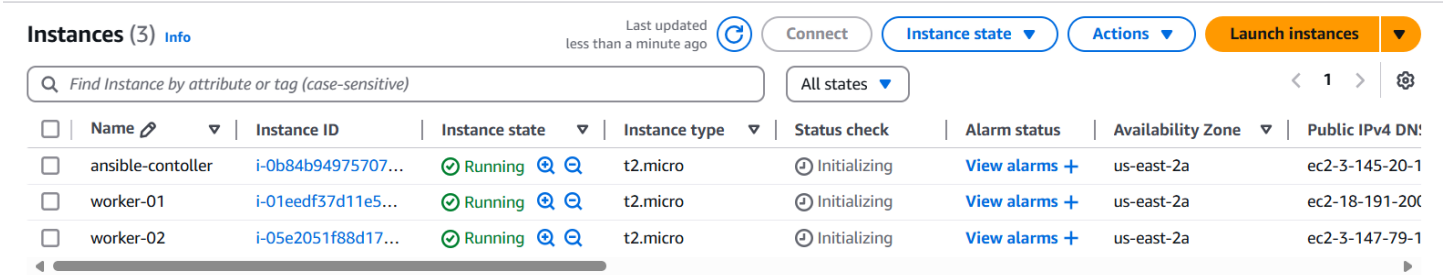
ANSIBLE-01

1) Watch ansible-01 video and write down notes. 2) Setup one master and two worker nodes in ansible.

→CREATE THREE INSTANCES WITH UBUNTU VERSION 22.04 AMI



→PYTHON IS PRE REQUISITE FOR ANSIBLE CHECK IF PYTHON IS INSTALLED ON SERVER

```
root@ip-172-31-2-241:~# python3 --version
Python 3.10.12
```

→UPDATE THE SERVER AND INSTALL ANSIBLE

apt update

apt install -y ansible

ansible –version

```
root@ip-172-31-2-241:~# ansible --version
ansible 2.10.8
  config file = None
  configured module search path = ['/root/.ansible/plugins/modules', '/usr/share/ansible/plugins/modules']
  ansible python module location = /usr/lib/python3/dist-packages/ansible
  executable location = /usr/bin/ansible
  python version = 3.10.12 (main, Sep 11 2024, 15:47:36) [GCC 11.4.0]
```

→CREATE PASSWORD LESS AUTHENTICATION FOR WORKER SERVERS

<ON ANSIBLE CONTROLLER SERVER>

sudo su –

cd

ssh-keygen -t rsa

cat /root/.ssh/id_rsa.pub

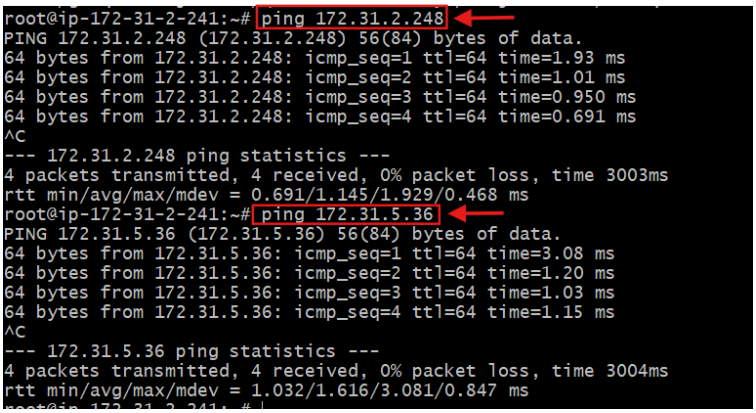
```
root@ip-172-31-2-241:~# cat /root/.ssh/id_rsa.pub
ssh-rsa AAAAB3NzaC1yc2EAAAADAQABAAABgQC51MXjZ9mD4EnTElhxSfSygvYs0VQvKlLhQXoQops1A8SrAFs6gbS4FChM4C5KBY3v9ueuRUmibDe8Xa
+O9viq8CRFNA/vR6wYBOmbtfP6kOhwsieW67ArsQR8g7n/i6v6ItdyZj8JGII5nb1tLk1XmtsnMxZqujWrCWX2UV2k+LSE1CSbrb47/znCqhvARPSNTDnc
6VF07/gzCq4O9OGZjW6YhrX/qFMrT8OesszV5CPbyx/ksojw9AXVsSK/Ku7Zqdto18zwEETwk2T4p3HYR7x7eK50PjaT5pNWQfDz0yPM8q29we+T8kf1z3
```

<ON WORKER-01>

ubuntu@ip-172-31-5-36:~$ sudo su
root@ip-172-31-5-36:/home/ubuntu# cd
root@ip-172-31-5-36:~# cd .ssh
root@ip-172-31-5-36:~/.ssh# vi authorized_keys
save the key
<ON WORKER-02>
UBUNTU@IP-172-31-2-248:~$ SUDO SU
ROOT@IP-172-31-2-248:/HOME/UBUNTU# CD
ROOT@IP-172-31-2-248:~# CD .SSH
ROOT@IP-172-31-2-248:~/.SSH# VI AUTHORIZED_KEYS

save the key

→CHECK THE CONNECTIVITY WITH PRIVATE IP FROM ANSIBLE CONTROLLER SERVER



NOTE: AFTER INSTALLATION IF ANSIBLE DIRECTORY IS MISSING IN /etc **THEN FOLLOW THE NEXT STEP WITH "→"**

→CREATE A ANSIBLE DIRECTORY WITH CONFIGURATION FILE AND HOSTS FILE

mkdir /etc/ansible
cd /etc/ansible
touch ansible.cfg hosts
vi /etc/ansible/ansible.cfg
<**add the following content and save the file>**

```
[defaults]
inventory = /etc/ansible/hosts
remote_user = root
host_key_checking = False
retry_files_enabled = False

[privilege_escalation]
become = True
become_method = sudo
become_user = root

[ssh_connection]
ssh_args = -o ControlMaster=auto -o ControlPersist=60s

[paramiko_connection]
pipelining = True

[inventory]
enable_plugins = host_list, script, yaml, ini, auto

[logging]
log_path = /var/log/ansible.log
```

→ADD THE INVENTORY IN HOSTS FILE
vi /etc/ansible/hosts
[all]
Your-Ips
[worker_01]
Your-ip
[worker_02]
Your-ip ***SAVE IT***

→NOW TRY THE ADHOC COMMAND TO PING THE WORKERS

ansible all -m ping

```
root@ip-172-31-2-241:/etc/ansible# ansible all -m ping
172.31.2.248 | SUCCESS => {
    "ansible_facts": {
        "discovered_interpreter_python": "/usr/bin/python3"
    },
    "changed": false,
    "ping": "pong"
}
172.31.5.36 | SUCCESS => {
    "ansible_facts": {
        "discovered_interpreter_python": "/usr/bin/python3"
    },
    "changed": false,
    "ping": "pong"
}
```

3) Execute the adhoc command

```
# To check the connetivity between slaves
ansible all -m ping
ansible all -m ping -i ansible_hosts
# How to gather facts of slave machine
ansible all -m setup
# Tips: -i is used if we are using more than one invertory file
# What if you do not have SSH key-based? How to pass username and password?
ansible all -m ping --user=ansadm --ask-pass
# TO checkthe uptime of a slave machine
ansible all -a uptime
# Tips: -m is the module and -a should contain the command it should run which goes as an argument to command and shell.
# check the free memory or memory usage of  host
ansible all -a "free -m"
# Execute a command as root user (sudo) on host
ansible all -m shell -a "cat /etc/passwd|grep -i ansadm" -s --ask-sudo-pass
# Execute a command using become module
ansible all -m shell -a "cat /etc/passwd|grep -i ansadm" -b -K
# Tips: -b is the option for become and by default it will become root user
        -K is to tell ansible to ask for SUDO password
#Execute a command as a different user  (sudo su)
ansible all -m file -a "path=/home/ansadm/test state=directory mode=0755" -b --become-user=ansadm
# Create a Linux user group
ansible all -s -m group -a "name=test state=present" -b -K
# Create a file with 755 permission
ansible all -m file -a "path=/home/asnadm/testfile state=touch mode=0755"
# Change ownership of a file
ansible all -m file -a "path=/home/ansadm group=weblogic owner=weblogic" -b
# Install a package using yum command
ansible all -m yum -a "name=httpd state=installed"
# Start or stop the service
# To Start
ansible all -m service -a "name=httpd state=started enabled=yes"
# To Stop
ansible all -m service -a "name=httpd state=stop enabled=yes"
```

→UPDATING BOTH WORKER SERVER WITH ANSIBLE CONTROLLER

ansible all -m shell a “apt update -y”

```
WARNING: apt does not have a stable CLI interface. Use with caution in scripts.
172.31.5.36 | CHANGED | rc=0 >>
Hit:1 http://us-east-2.ec2.archive.ubuntu.com/ubuntu jammy InRelease
Get:2 http://us-east-2.ec2.archive.ubuntu.com/ubuntu jammy-updates InRelease [128 kB]
```

```
root@ip-172-31-2-241:/home/ubuntu# ansible all -m shell -a "apt update -y"
172.31.2.248 | CHANGED | rc=0 >>
Hit:1 http://us-east-2.ec2.archive.ubuntu.com/ubuntu jammy InRelease
Get:2 http://us-east-2.ec2.archive.ubuntu.com/ubuntu jammy-updates InRelease [128 kB]
Get:3 http://us-east-2.ec2.archive.ubuntu.com/ubuntu jammy-backports InRelease [127 kB]
```

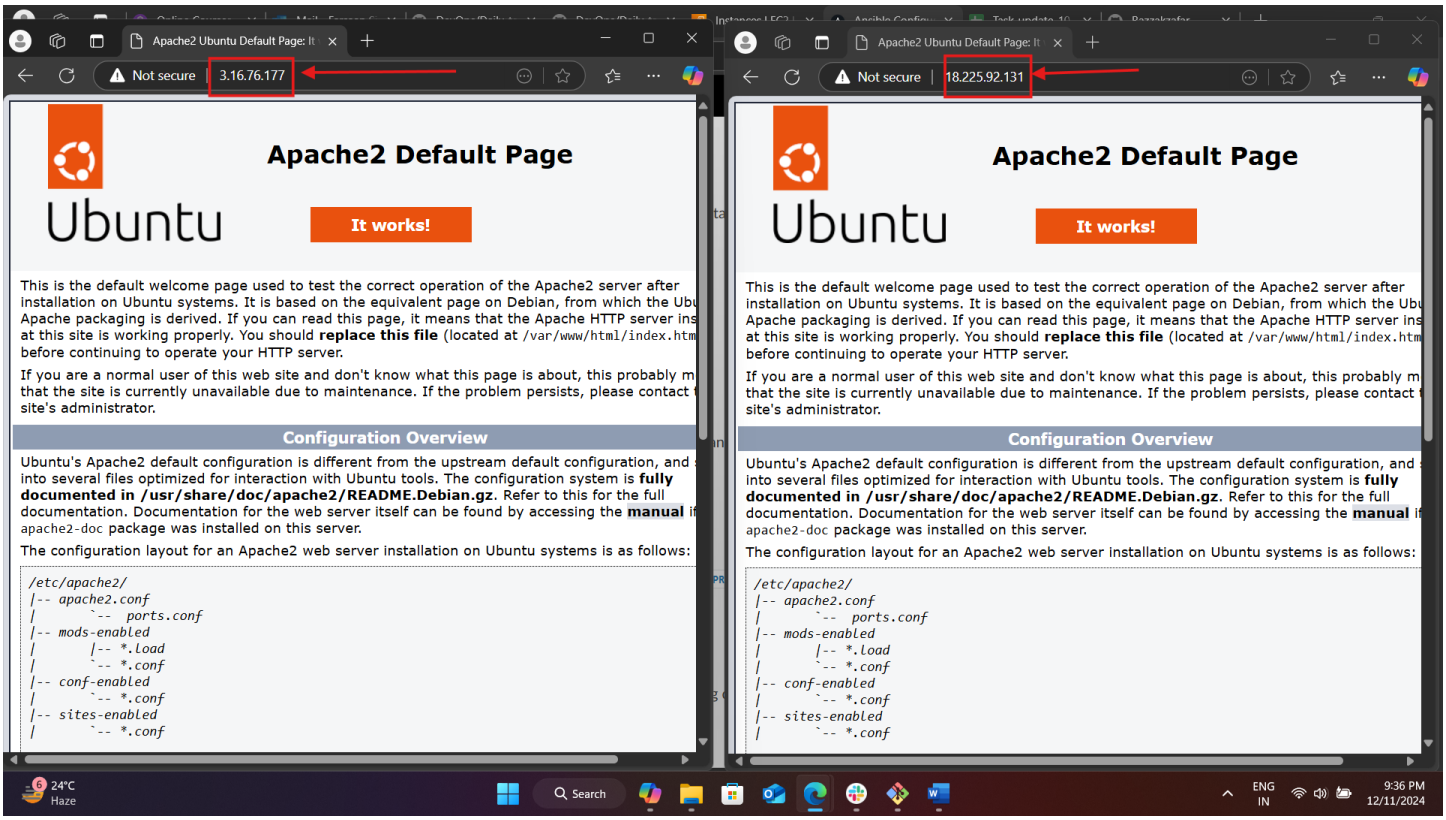
→INSTALLING APACHE IN BOTH WORKER SERVER

ansible all -m apt -a "name=apache2 state=latest"

```
172.31.2.248 | CHANGED => {
    "ansible_facts": {
        "discovered_interpreter_python": "/usr/bin/python3"
    },
    "cache_update_time": 1733932444,
    "cache_updated": false,
    "changed": true,
    "stderr": "",
    "stderr_lines": [],
```

```
172.31.5.36 | CHANGED => {
    "ansible_facts": {
        "discovered_interpreter_python": "/usr/bin/python3"
    },
    "cache_update_time": 1733932456,
    "cache_updated": false,
    "changed": true,
    "stderr": "",
    "stderr_lines": [],
```

→CHECKING ON BROWSER IF THE SERVICE IS RUNNING



→CREATING MY OWN INVENTORY IN DIFFERENT LOCATION

vi /tmp/farsaan_inventory

[worker_01]

Your-ip

ansible -i /tmp/farsaan_inventory worker_01 -m ping

```
root@ip-172-31-2-241:/home/ubuntu# ansible -i /tmp/farsaan_inventory worker_01 -m ping
172.31.2.248 | SUCCESS => {
    "ansible_facts": {
        "discovered_interpreter_python": "/usr/bin/python3"
    },
    "changed": false,
    "ping": "pong"
}
```