School of Computer Science

University of Petroleum and Energy Studies

System Provisioning & Configuration Management

Lab File (6th Sem)

Submitted By:-	Submitted To:-
Akshat Pandey 500101788 R2142220306 DevOps B1	Dr Hitesh Kumar Sharma

EXPERIMENT 1

Lab Exercise: Creating Static Host Inventory

Objective: To create a static host inventory for managing and automating infrastructure tasks efficiently across multiple servers using Ansible

Tools required: Ubuntu OS

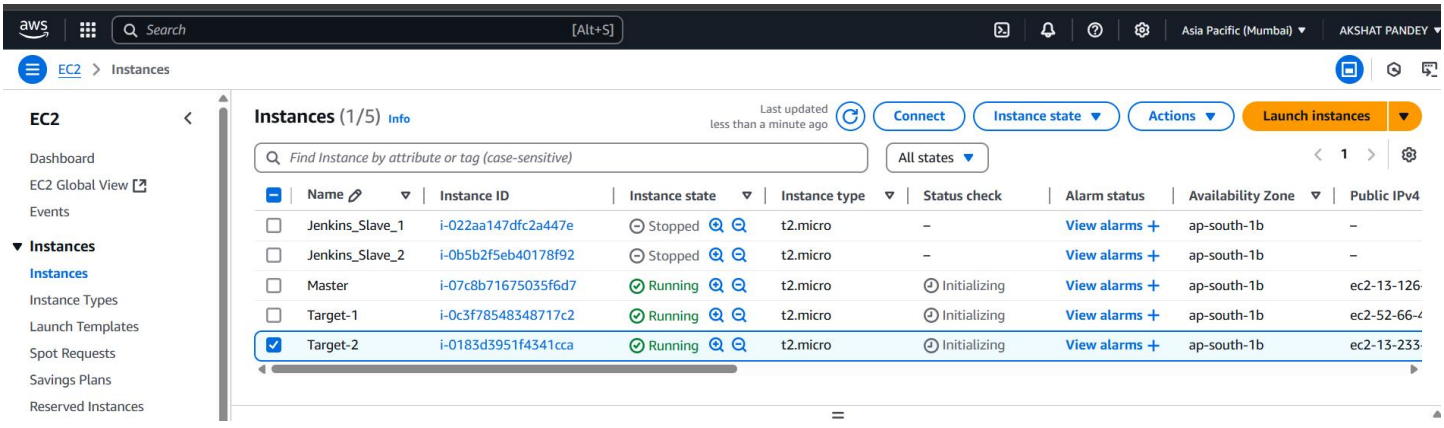
Prerequisites: You need to have Ansible installed to proceed with this demo

Steps to be followed:

1. Generate SSH key pair on the main node
2. Copy the SSH key to the two other nodes
3. Update the inventory or host file with the host IP address
4. Establish connectivity between the hosts specified in the host file and the Ansible server

Step 1: Launch EC2 Instance

1. Go to AWS Console → EC2 → Launch instance
2. OS: **Ubuntu 22.04** or similar
3. Instance type: **t2.micro** (Free Tier)
4. Enable **port 22 (SSH)** in Security Group



→ SSH into Your Instance: From the terminal (on your laptop): **ssh -i master_key.pem ubuntu@13.235.83.52**

```
The authenticity of host '13.235.83.52 (13.235.83.52)' can't be established.
ED25519 key fingerprint is SHA256:+jzyARS+SqHVcC734U4+YwArkTh8xgxIozfuKZRnes0.
This key is not known by any other names.
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added '13.235.83.52' (ED25519) to the list of known hosts.
Welcome to Ubuntu 24.04.2 LTS (GNU/Linux 6.8.0-1024-aws x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:     https://landscape.canonical.com
 * Support:        https://ubuntu.com/pro

 System information as of Fri Apr 11 10:59:01 UTC 2025

  System load:  0.0               Processes:             104
  Usage of /:   25.0% of 6.71GB   Users logged in:       0
  Memory usage: 19%               IPv4 address for enX0: 172.31.15.101
  Swap usage:   0%

Expanded Security Maintenance for Applications is not enabled.

0 updates can be applied immediately.

Enable ESM Apps to receive additional future security updates.
See https://ubuntu.com/esm or run: sudo pro status


The list of available updates is more than a week old.
To check for new updates run: sudo apt update
```

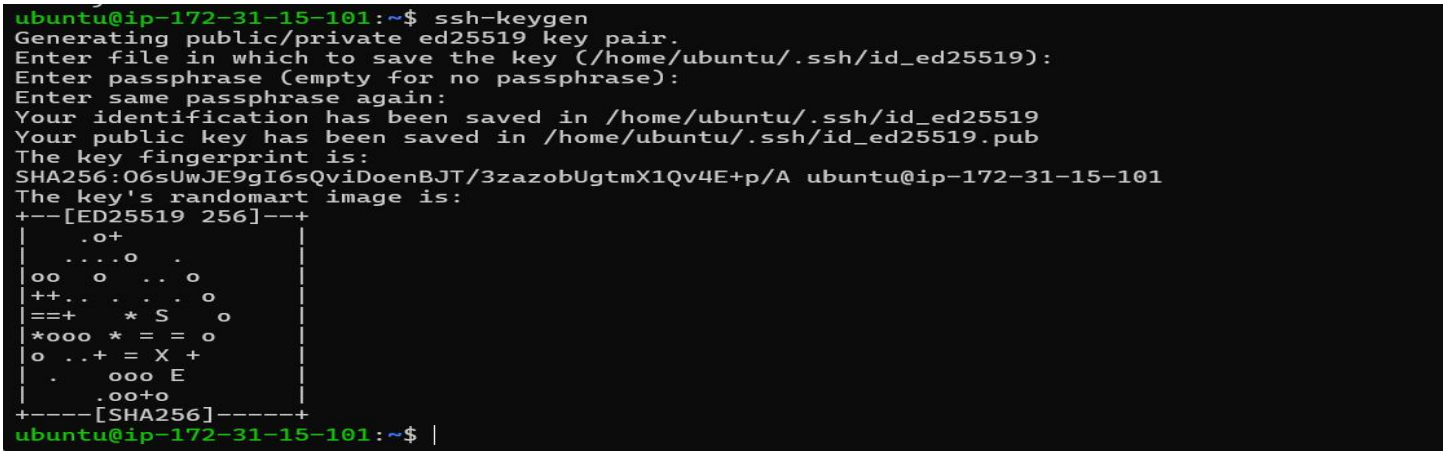
→Install Ansible on EC2: Once you're logged into the instance: **sudo apt update**

sudo apt install ansible -y

```
ubuntu@ip-172-31-15-101:~$ sudo apt update
sudo apt install ansible -y
Hit:1 http://ap-south-1.ec2.archive.ubuntu.com/ubuntu noble InRelease
Hit:2 http://ap-south-1.ec2.archive.ubuntu.com/ubuntu noble-updates InRelease
Hit:3 http://ap-south-1.ec2.archive.ubuntu.com/ubuntu noble-backports InRelease
Hit:4 http://security.ubuntu.com/ubuntu noble-security InRelease
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
59 packages can be upgraded. Run 'apt list --upgradable' to see them.
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
ansible is already the newest version (9.2.0+dfsg-0ubuntu5).
0 upgraded, 0 newly installed, 0 to remove and 59 not upgraded.
ubuntu@ip-172-31-15-101:~$ ansible --version
ansible [core 2.16.3]
  config file = None
  configured module search path = ['/home/ubuntu/.ansible/plugins/modules', '/usr/share/ansible/plugins/modules']
  ansible python module location = /usr/lib/python3/dist-packages/ansible
  ansible collection location = /home/ubuntu/.ansible/collections:/usr/share/ansible/collections
  executable location = /usr/bin/ansible
  python version = 3.12.3 (main, Feb  4 2025, 14:48:35) [GCC 13.3.0] (/usr/bin/python3)
  jinja version = 3.1.2
  libyaml = True
ubuntu@ip-172-31-15-101:~$
```

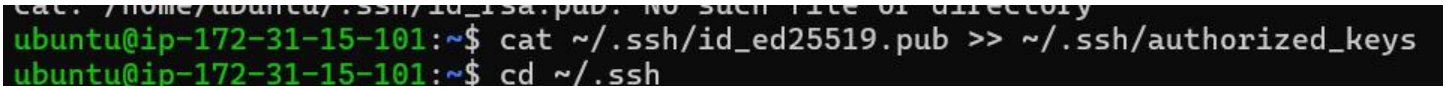
Step 2: Generate SSH key pair on the main node

2.1 Use the following command to generate the SSH key on the Ansible server: **ssh-keygen**



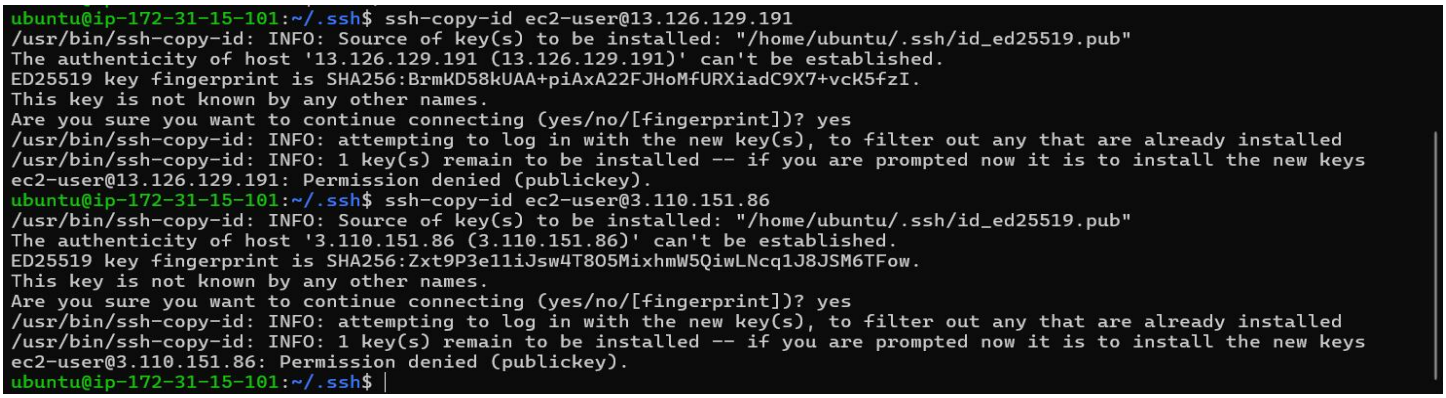
Step 3: Copy the SSH key to the other two nodes

3.1 Use the following command to copy the public key to a file named **authorized_keys** in localhost: **cat .ssh/id_rsa.pub >> .ssh/authorized_keys**



3.2 Run the following command to go to the **.ssh** directory of the Ansible server: **cd .ssh**

3.3Run the following command to copy the public key to another node that will connect to the Ansible server: **ssh-copy-id username@ip -p 22**



Note: You must use a **username@ip** with your node and IP username, which are provided in the lab credential.

3.4Execute the following command to exit the **.ssh** directory of the Ansible server: **cd**

```
ubuntu@ip-172-31-15-101:~/.ssh$ cd
ubuntu@ip-172-31-15-101:~$
```

Step 4: Update the inventory or host file with the host IP address

4.1Use the following command to open the Ansible inventory file and add the host localhost to it: **sudo vi /etc/ansible/hosts**

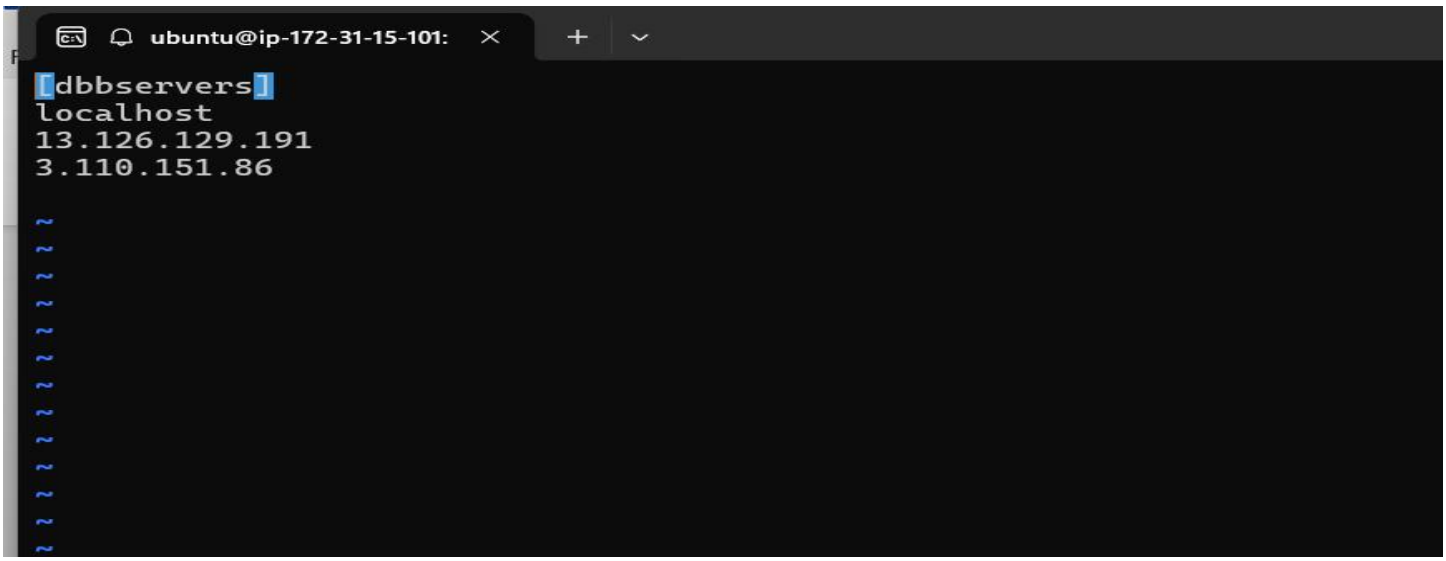
```
ubuntu@ip-172-31-15-101:~$ sudo mkdir -p /etc/ansible
ubuntu@ip-172-31-15-101:~$ sudo touch /etc/ansible/hosts
ubuntu@ip-172-31-15-101:~$ sudo vi /etc/ansible/hosts
```

4.2When the file opens, add the three lines of code below to the end of the file:

[dbbservers]

localhost:22

172.31.5.76:22



Note: Press **esc**, then write **:wq** and press **enter** to save the file.

Step 5: Establish connectivity between the hosts specified in the host file and the Ansible server

5.1Run the following command to copy the public key to another node that will connect to the Ansible server: **ansible -m ping dbbservers**

```
ubuntu@ip-172-31-15-101:~$ ansible -m ping dbbservers
localhost | SUCCESS => {
    "ansible_facts": {
        "discovered_interpreter_python": "/usr/bin/python3"
    },
    "changed": false,
    "ping": "pong"
}
3.110.151.86 | SUCCESS => {
    "ansible_facts": {
        "discovered_interpreter_python": "/usr/bin/python3"
    },
    "changed": false,
    "ping": "pong"
}
13.126.129.191 | SUCCESS => {
    "ansible_facts": {
        "discovered_interpreter_python": "/usr/bin/python3"
    },
    "changed": false,
    "ping": "pong"
}
ubuntu@ip-172-31-15-101:~$
```

5.2 Use the following command to check the number of hosts in the host file: **ansible all --list-hosts**

```
ubuntu@ip-172-31-15-101:~$ ansible all --list-hosts
  hosts (3):
    13.126.129.191
    3.110.151.86
    localhost
ubuntu@ip-172-31-15-101:~$
```

By following these steps, you have successfully created a static host inventory for managing and automating infrastructure tasks efficiently across multiple servers using Ansible.