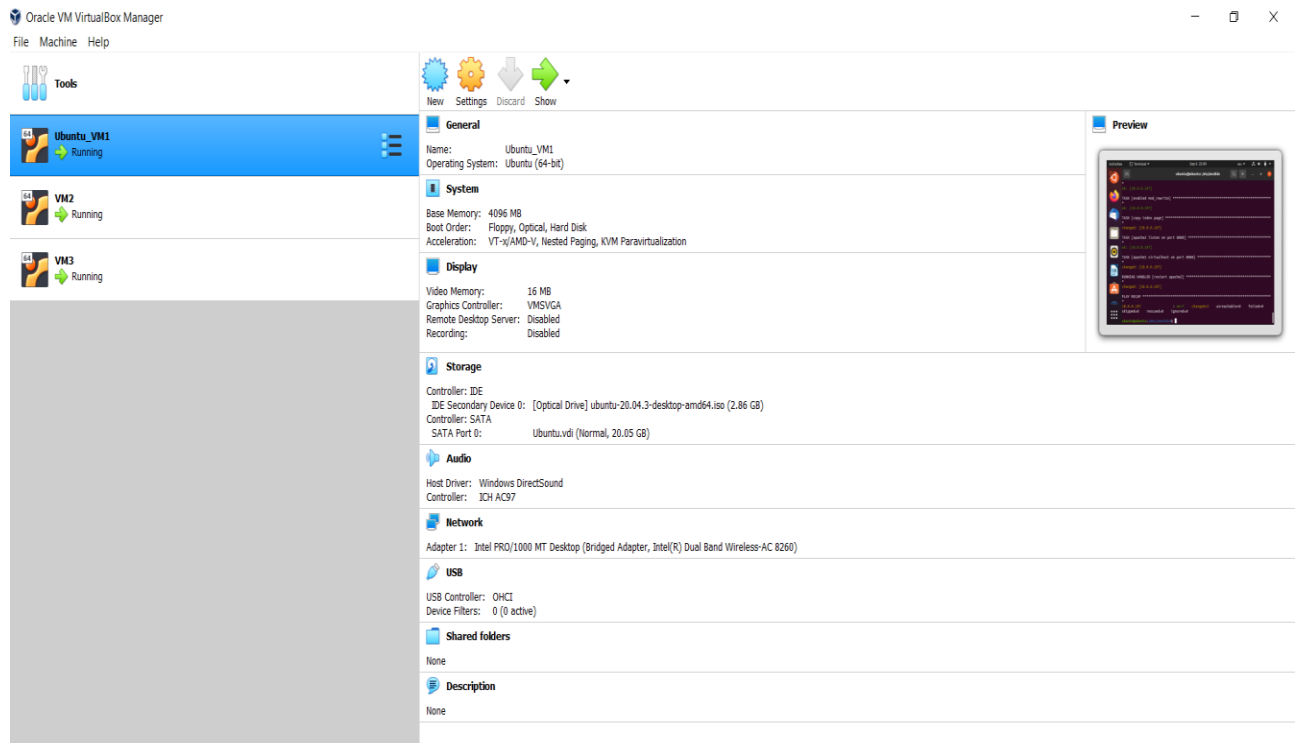


ASSIGNMENT 1: ANSIBLE

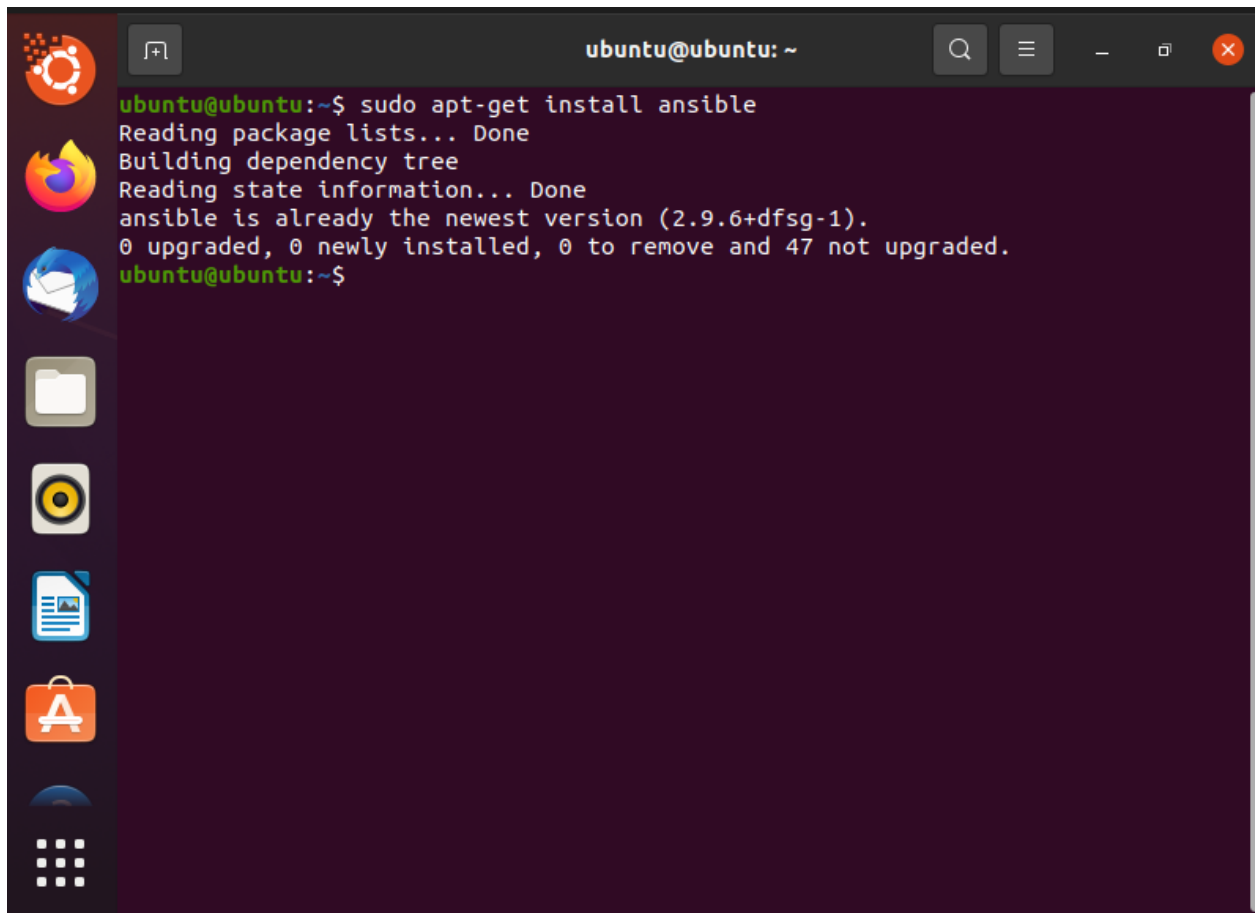
GitHub Repo:

[https://github.com/poojashreeNS/Assignments/tree/main/HW%231 Ansible/
HW%231 By Charu](https://github.com/poojashreeNS/Assignments/tree/main/HW%231%20Ansible/HW%231%20By%20Charu)

1. **Provision 3 VMs:** Ubuntu_VM1, will be acting as primary/deployer VM in this exercise and rest two VMs are going to be client VMs on which webserver will be deployed.



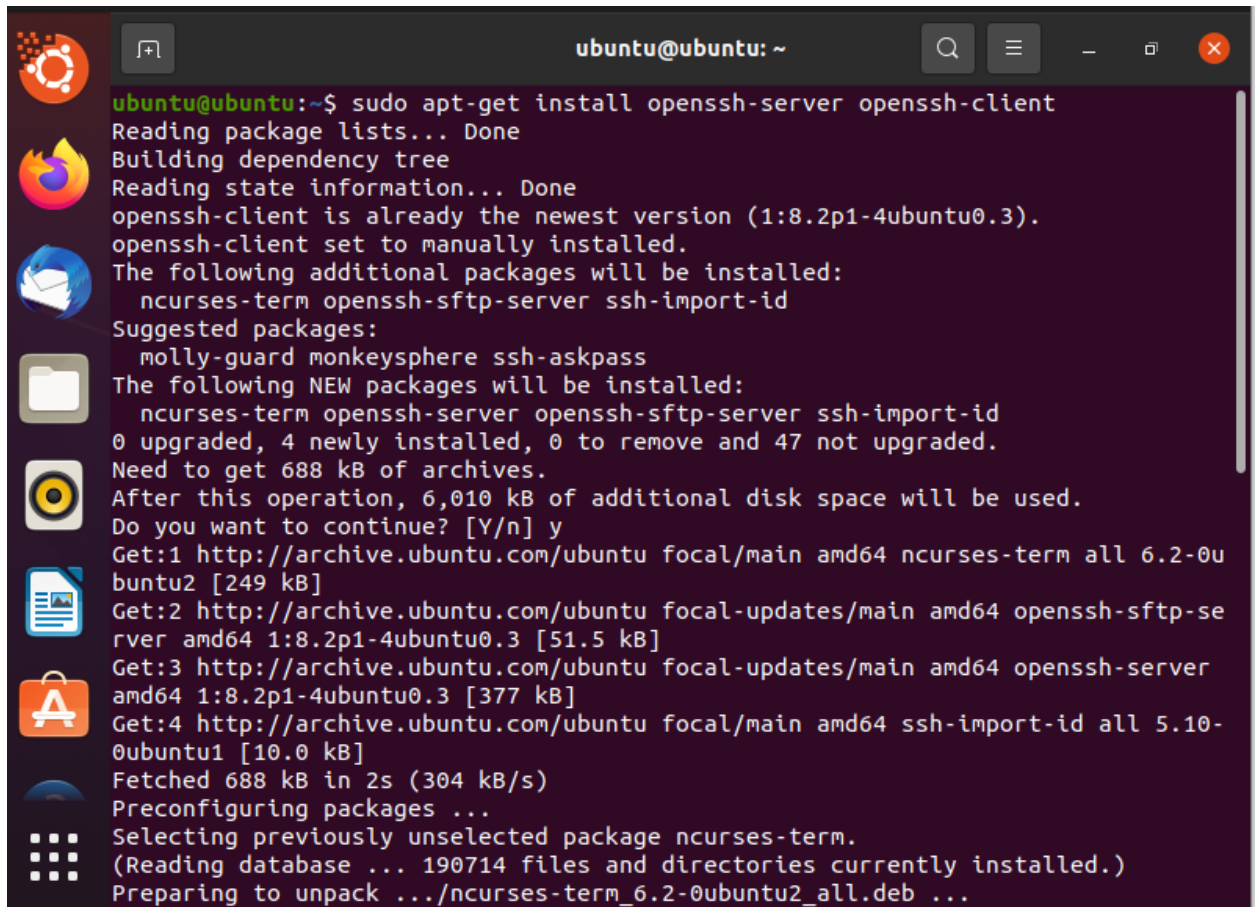
2. Install Ansible on primary VM.



A terminal window titled 'ubuntu@ubuntu: ~' with standard window controls. The terminal output shows the command 'sudo apt-get install ansible' being executed. The system reports that Ansible is already the newest version (2.9.6+dfsg-1) and that no packages need to be upgraded, installed, or removed. The prompt returns to 'ubuntu@ubuntu:~\$'.

```
ubuntu@ubuntu:~$ sudo apt-get install ansible
Reading package lists... Done
Building dependency tree
Reading state information... Done
ansible is already the newest version (2.9.6+dfsg-1).
0 upgraded, 0 newly installed, 0 to remove and 47 not upgraded.
ubuntu@ubuntu:~$
```

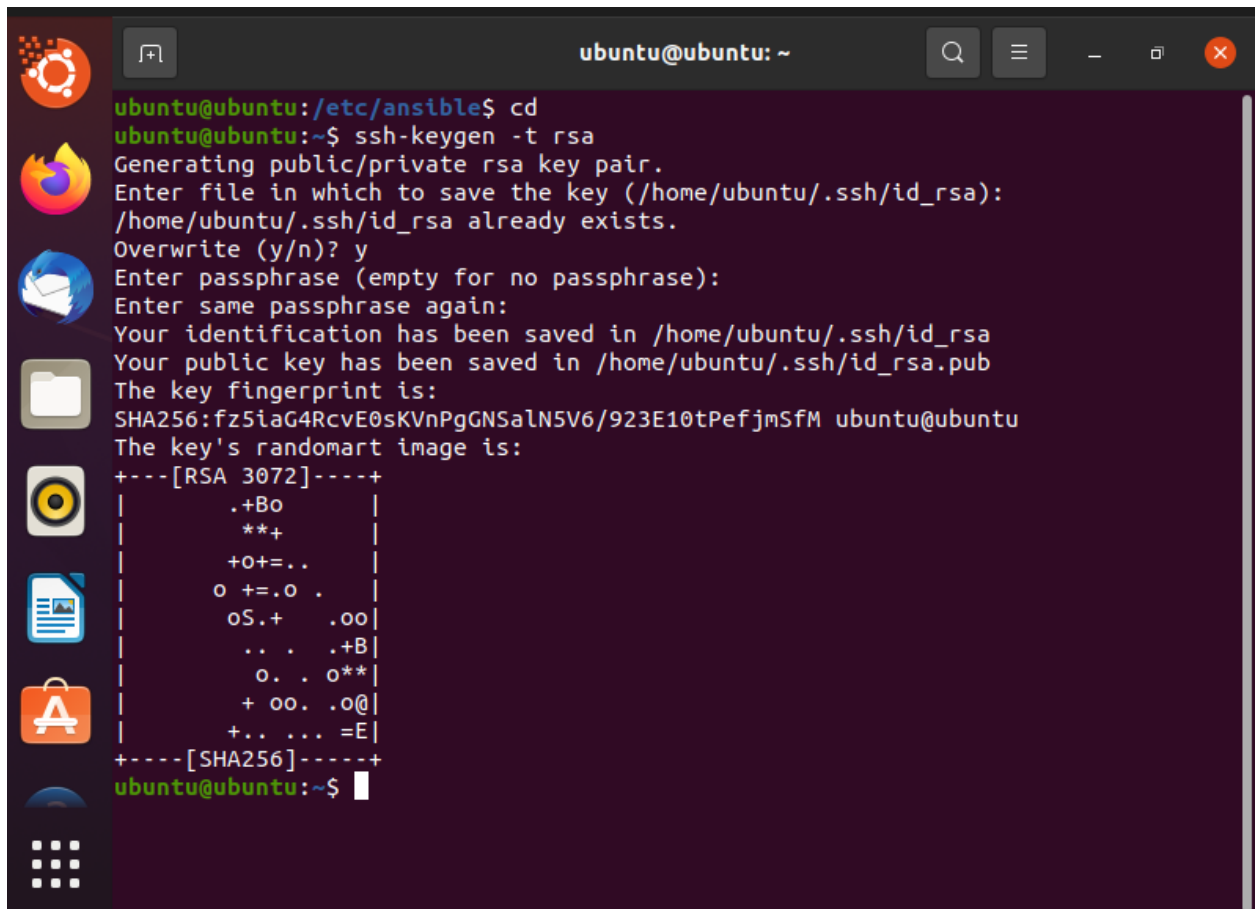
3. Install openssh server and client on all VMs.



The image shows a terminal window titled 'ubuntu@ubuntu: ~' with a search icon, a menu icon, and window control buttons. The terminal output shows the command 'sudo apt-get install openssh-server openssh-client' being executed. The output indicates that openssh-client is already installed and that several additional packages will be installed along with the requested ones. The packages to be installed are ncurses-term, openssh-server, openssh-sftp-server, and ssh-import-id. The terminal also shows the progress of downloading these packages from the Ubuntu archive.

```
ubuntu@ubuntu:~$ sudo apt-get install openssh-server openssh-client
Reading package lists... Done
Building dependency tree
Reading state information... Done
openssh-client is already the newest version (1:8.2p1-4ubuntu0.3).
openssh-client set to manually installed.
The following additional packages will be installed:
  ncurses-term openssh-sftp-server ssh-import-id
Suggested packages:
  molly-guard monkeysphere ssh-askpass
The following NEW packages will be installed:
  ncurses-term openssh-server openssh-sftp-server ssh-import-id
0 upgraded, 4 newly installed, 0 to remove and 47 not upgraded.
Need to get 688 kB of archives.
After this operation, 6,010 kB of additional disk space will be used.
Do you want to continue? [Y/n] y
Get:1 http://archive.ubuntu.com/ubuntu focal/main amd64 ncurses-term all 6.2-0ubuntu2 [249 kB]
Get:2 http://archive.ubuntu.com/ubuntu focal-updates/main amd64 openssh-sftp-server amd64 1:8.2p1-4ubuntu0.3 [51.5 kB]
Get:3 http://archive.ubuntu.com/ubuntu focal-updates/main amd64 openssh-server amd64 1:8.2p1-4ubuntu0.3 [377 kB]
Get:4 http://archive.ubuntu.com/ubuntu focal/main amd64 ssh-import-id all 5.10-0ubuntu1 [10.0 kB]
Fetched 688 kB in 2s (304 kB/s)
Preconfiguring packages ...
Selecting previously unselected package ncurses-term.
(Reading database ... 190714 files and directories currently installed.)
Preparing to unpack .../ncurses-term_6.2-0ubuntu2_all.deb ...
```

4. Generate keys on primary VM.



The image shows a terminal window titled 'ubuntu@ubuntu: ~'. The user has navigated to the directory `/etc/ansible` and then back to the home directory. They have run the command `ssh-keygen -t rsa` to generate an RSA key pair. The terminal output shows the following steps: 'Generating public/private rsa key pair.', 'Enter file in which to save the key (/home/ubuntu/.ssh/id_rsa): /home/ubuntu/.ssh/id_rsa already exists.', 'Overwrite (y/n)? y', 'Enter passphrase (empty for no passphrase):', 'Enter same passphrase again:', 'Your identification has been saved in /home/ubuntu/.ssh/id_rsa', 'Your public key has been saved in /home/ubuntu/.ssh/id_rsa.pub', 'The key fingerprint is: SHA256:fz5iaG4RcvE0sKVnPgGNSalN5V6/923E10tPefjmSfM ubuntu@ubuntu', and 'The key's randomart image is:'. The randomart image is displayed as a grid of characters representing the key's fingerprint. The terminal window has a sidebar on the left with icons for various applications like a file manager, web browser, and terminal.

```
ubuntu@ubuntu:~$ cd /etc/ansible
ubuntu@ubuntu:~$ ssh-keygen -t rsa
Generating public/private rsa key pair.
Enter file in which to save the key (/home/ubuntu/.ssh/id_rsa):
/home/ubuntu/.ssh/id_rsa already exists.
Overwrite (y/n)? y
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in /home/ubuntu/.ssh/id_rsa
Your public key has been saved in /home/ubuntu/.ssh/id_rsa.pub
The key fingerprint is:
SHA256:fz5iaG4RcvE0sKVnPgGNSalN5V6/923E10tPefjmSfM ubuntu@ubuntu
The key's randomart image is:
+---[RSA 3072]---+
|      .+Bo      |
|      **+       |
|      +O+=..    |
|      o +=.o .   |
|      oS.+ .oo   |
|      .. . .+B   |
|      o. . o**   |
|      + oo. .o@  |
|      +.. ... =E  |
+-----[SHA256]-----+
ubuntu@ubuntu:~$
```

5. Determine IPs of client machines (VM2 and VM3) using ifconfig command and copy primary machine's ssh key into client machine's authorized keys.

```
ubuntu@ubuntu: ~
..
Unpacking net-tools (1.60+git20180626.aebd88e-1ubuntu1) ...
Setting up net-tools (1.60+git20180626.aebd88e-1ubuntu1) ...
Processing triggers for man-db (2.9.1-1) ...
ubuntu@ubuntu:~$ ifconfig
enp0s3: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
    inet 10.0.0.222 netmask 255.255.255.0 broadcast 10.0.0.255
    inet6 2601:647:4001:4f90:bab1:5f6f:4dc0:decc prefixlen 64 scopeid 0x0
<global>
    inet6 2601:647:4001:4f90::93af prefixlen 128 scopeid 0x0<global>
    inet6 2601:647:4001:4f90:fcc8:723f:bf2b:7c0e prefixlen 64 scopeid 0x0
<global>
    inet6 fe80::817b:7b25:1991:336f prefixlen 64 scopeid 0x20<link>
    ether 08:00:27:e0:57:9a txqueuelen 1000 (Ethernet)
    RX packets 6196 bytes 6373313 (6.3 MB)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 2398 bytes 292461 (292.4 KB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

lo: flags=73<UP,LOOPBACK,RUNNING> mtu 65536
    inet 127.0.0.1 netmask 255.0.0.0
    inet6 ::1 prefixlen 128 scopeid 0x10<host>
    loop txqueuelen 1000 (Local Loopback)
    RX packets 292 bytes 25308 (25.3 KB)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 292 bytes 25308 (25.3 KB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0
```

```
ubuntu@ubuntu: ~
ubuntu@ubuntu:~$ ssh-copy-id -i ubuntu@10.0.0.222
/usr/bin/ssh-copy-id: INFO: Source of key(s) to be installed: "/home/ubuntu/.ssh/id_rsa.pub"
The authenticity of host '10.0.0.222 (10.0.0.222)' can't be established.
ECDSA key fingerprint is SHA256:D/QVqCgrA1f+uOruPCZ744Mx565b0yOZ20m59RTlGww.
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
/usr/bin/ssh-copy-id: INFO: attempting to log in with the new key(s), to filter
out any that are already installed
/usr/bin/ssh-copy-id: INFO: 1 key(s) remain to be installed -- if you are promp
ted now it is to install the new keys
ubuntu@10.0.0.222's password:

Number of key(s) added: 1

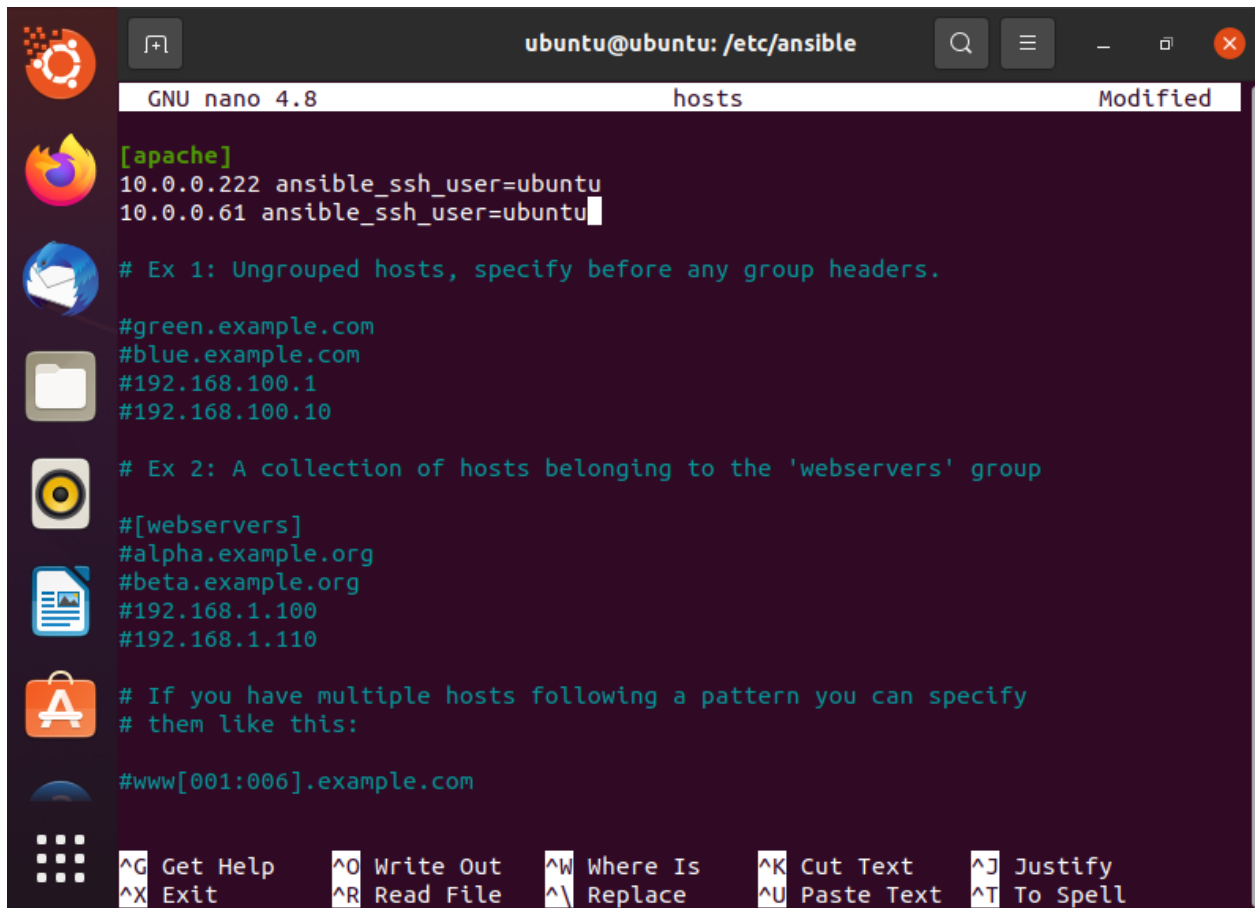
Now try logging into the machine, with:  "ssh 'ubuntu@10.0.0.222'"
and check to make sure that only the key(s) you wanted were added.

ubuntu@ubuntu:~$
```

```
ubuntu@ubuntu: ~  
..  
Unpacking net-tools (1.60+git20180626.aebd88e-1ubuntu1) ...  
Setting up net-tools (1.60+git20180626.aebd88e-1ubuntu1) ...  
Processing triggers for man-db (2.9.1-1) ...  
ubuntu@ubuntu:~$ ifconfig  
enp0s3: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500  
    inet 10.0.0.61 netmask 255.255.255.0 broadcast 10.0.0.255  
    inet6 fe80::d63d:bc6a:f80f:ccfc prefixlen 64 scopeid 0x20<link>  
    inet6 2601:647:4001:4f90:8ddd:aefe:1faf:418d prefixlen 64 scopeid 0x0  
<global>  
    inet6 2601:647:4001:4f90::a5e8 prefixlen 128 scopeid 0x0<global>  
    inet6 2601:647:4001:4f90:2ae0:8fa2:851b:7bf4 prefixlen 64 scopeid 0x0  
<global>  
    ether 08:00:27:22:bd:79 txqueuelen 1000 (Ethernet)  
    RX packets 6179 bytes 6374934 (6.3 MB)  
    RX errors 0 dropped 0 overruns 0 frame 0  
    TX packets 2420 bytes 300343 (300.3 KB)  
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0  
  
lo: flags=73<UP,LOOPBACK,RUNNING> mtu 65536  
    inet 127.0.0.1 netmask 255.0.0.0  
    inet6 ::1 prefixlen 128 scopeid 0x10<host>  
    loop txqueuelen 1000 (Local Loopback)  
    RX packets 285 bytes 25646 (25.6 KB)  
    RX errors 0 dropped 0 overruns 0 frame 0  
    TX packets 285 bytes 25646 (25.6 KB)  
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0
```

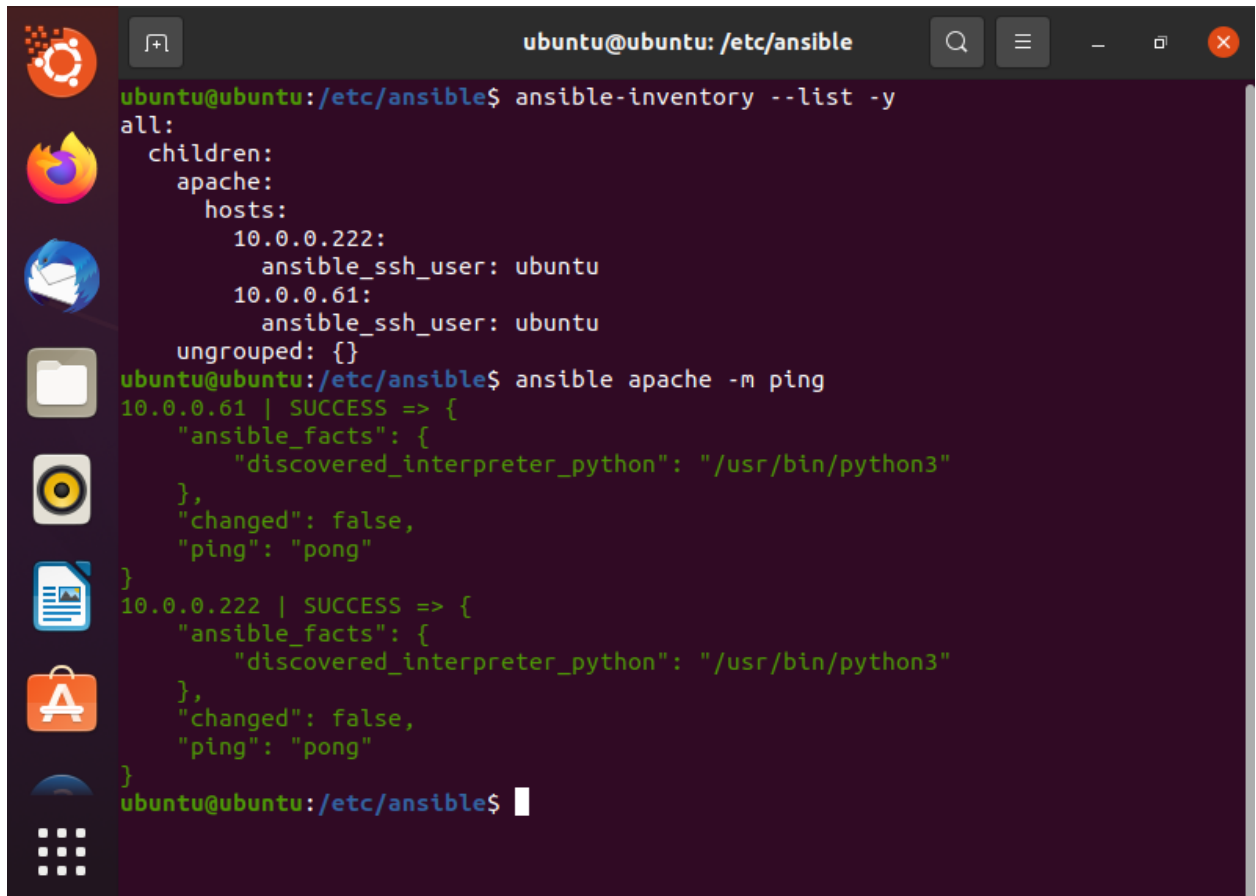
```
ubuntu@ubuntu:~$ ssh-copy-id -i ubuntu@10.0.0.61  
/usr/bin/ssh-copy-id: INFO: Source of key(s) to be installed: "/home/ubuntu/.ssh/id_rsa.pub"  
The authenticity of host '10.0.0.61 (10.0.0.61)' can't be established.  
ECDSA key fingerprint is SHA256:1dF7yVdADS1E7J9hJGJwLEyVwWlZoZxQe0qWT2a/yGo.  
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes  
/usr/bin/ssh-copy-id: INFO: attempting to log in with the new key(s), to filter  
out any that are already installed  
/usr/bin/ssh-copy-id: INFO: 1 key(s) remain to be installed -- if you are promp  
ted now it is to install the new keys  
ubuntu@10.0.0.61's password:  
  
Number of key(s) added: 1  
  
Now try logging into the machine, with: "ssh 'ubuntu@10.0.0.61'"  
and check to make sure that only the key(s) you wanted were added.  
  
ubuntu@ubuntu:~$
```

6. Update the hosts file in /etc/ansible directory to specify our client VMs IP address as part of a new group '[apache]' (Since I have used Apache webserver as part of this assignment).



```
ubuntu@ubuntu: /etc/ansible
GNU nano 4.8 hosts Modified
[apache]
10.0.0.222 ansible_ssh_user=ubuntu
10.0.0.61 ansible_ssh_user=ubuntu
# Ex 1: Ungrouped hosts, specify before any group headers.
#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
^G Get Help    ^O Write Out  ^W Where Is   ^K Cut Text   ^J Justify
^X Exit        ^R Read File  ^\ Replace    ^U Paste Text ^T To Spell
```


7. Verify that ansible inventory is updated to reflect newly added server group and make sure that all the host in '[apache]' group are pingable.



The image shows a terminal window titled 'ubuntu@ubuntu: /etc/ansible'. The terminal displays the output of the 'ansible-inventory --list -y' command, which shows a hierarchical inventory structure. The 'all' group contains a 'children' group, which includes an 'apache' group and an 'ungrouped' group. The 'apache' group contains two hosts: '10.0.0.222' and '10.0.0.61', both with 'ansible_ssh_user: ubuntu'. The 'ungrouped' group is empty. Below the inventory output, the terminal shows the output of the 'ansible apache -m ping' command. This command successfully pings both hosts, returning 'pong' for each. The output for '10.0.0.61' shows 'ansible_facts' with 'discovered_interpreter_python: /usr/bin/python3'. The output for '10.0.0.222' shows 'ansible_facts' with 'discovered_interpreter_python: /usr/bin/python3'. The terminal ends with the prompt 'ubuntu@ubuntu: /etc/ansible\$'.

```
ubuntu@ubuntu: /etc/ansible$ ansible-inventory --list -y
all:
  children:
    apache:
      hosts:
        10.0.0.222:
          ansible_ssh_user: ubuntu
        10.0.0.61:
          ansible_ssh_user: ubuntu
    ungrouped: {}
ubuntu@ubuntu: /etc/ansible$ ansible apache -m ping
10.0.0.61 | SUCCESS => {
  "ansible_facts": {
    "discovered_interpreter_python": "/usr/bin/python3"
  },
  "changed": false,
  "ping": "pong"
}
10.0.0.222 | SUCCESS => {
  "ansible_facts": {
    "discovered_interpreter_python": "/usr/bin/python3"
  },
  "changed": false,
  "ping": "pong"
}
ubuntu@ubuntu: /etc/ansible$
```

8. Deploy Apache webserver on both client machines by means of the curated playbook "deploy_apache_webserver.yml"
(https://github.com/poojashreeNS/Assignments/blob/main/HW%20Ansible/HW%20By%20Charu/deploy_apache_webserver.yml).



```
ubuntu@ubuntu: /etc/ansible
ubuntu@ubuntu:/etc/ansible$ ansible-playbook deploy_apache_webserver.yml

PLAY [apache] *****
*

TASK [Gathering Facts] *****
*
ok: [10.0.0.222]
ok: [10.0.0.61]

TASK [install apache2] *****
*
changed: [10.0.0.61]
changed: [10.0.0.222]

TASK [enabled mod_rewrite] *****
*
changed: [10.0.0.222]
changed: [10.0.0.61]

TASK [copy index page] *****
*
changed: [10.0.0.61]
changed: [10.0.0.222]

TASK [configure apache2 to listen on port 8080] *****
*
changed: [10.0.0.222]
changed: [10.0.0.61]

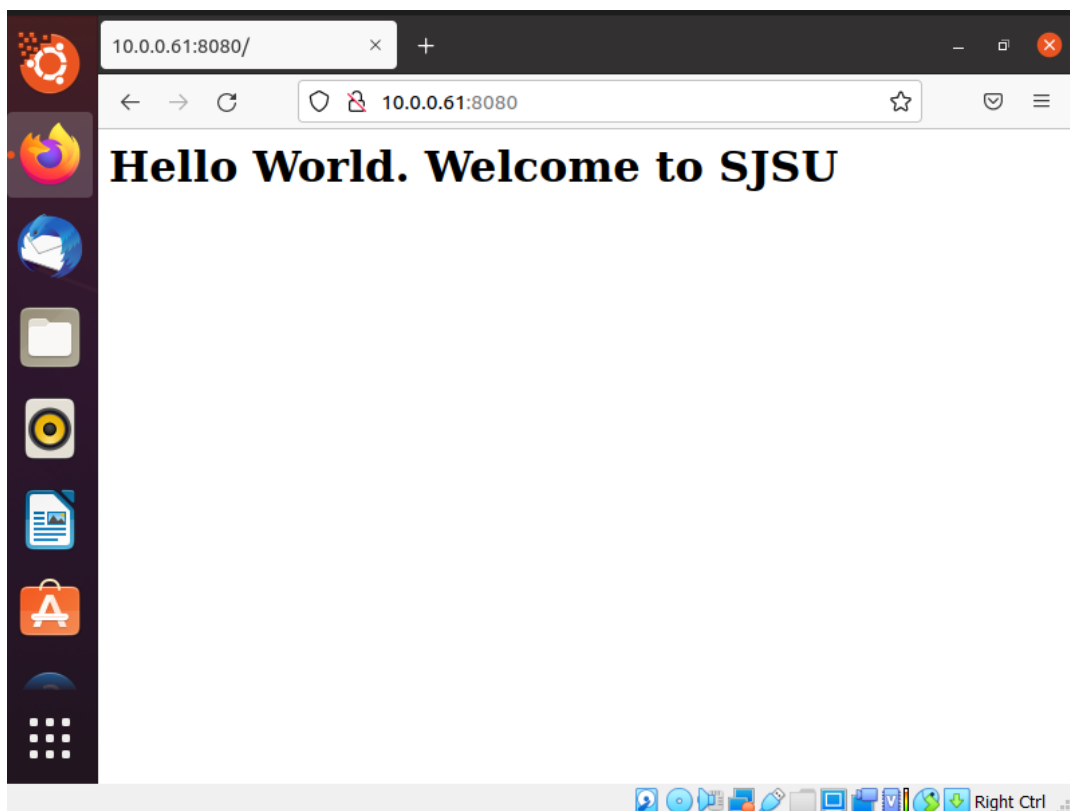
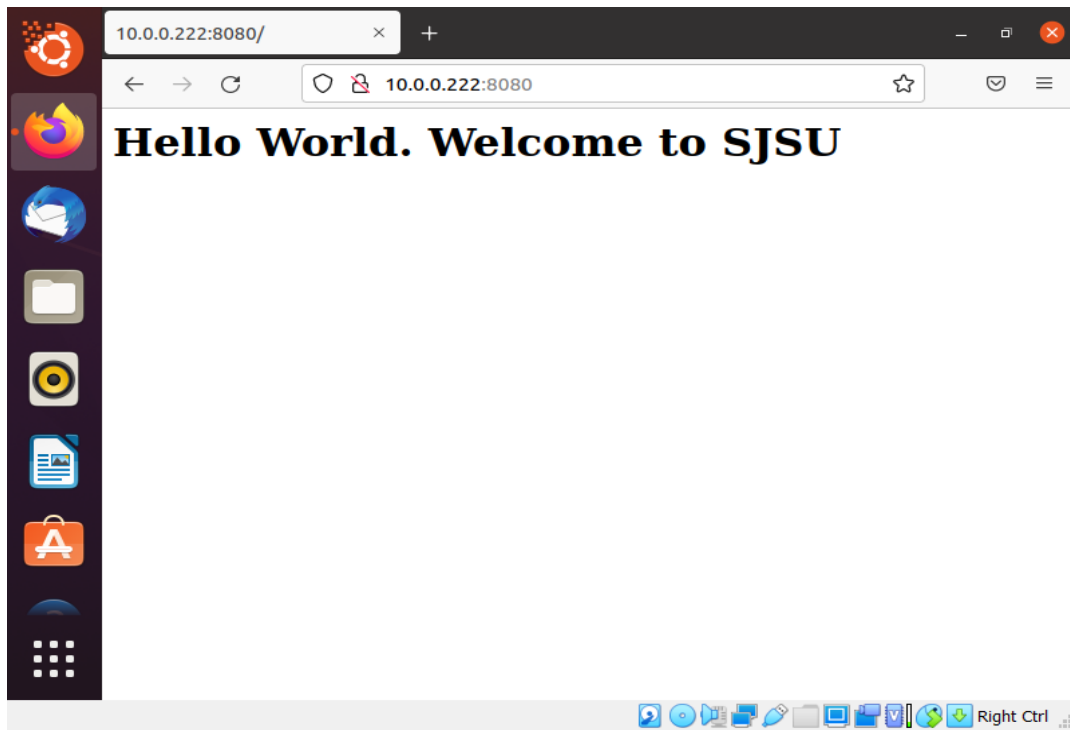
TASK [configure apache2 virtualhost on port 8080] *****
*
changed: [10.0.0.222]
changed: [10.0.0.61]

RUNNING HANDLER [restart apache2] *****
*
changed: [10.0.0.222]
changed: [10.0.0.61]

PLAY RECAP *****
10.0.0.222      : ok=7    changed=6    unreachable=0    failed=0
skipped=0      rescued=0    ignored=0
10.0.0.61      : ok=7    changed=6    unreachable=0    failed=0
skipped=0      rescued=0    ignored=0

ubuntu@ubuntu:/etc/ansible$
```

9. Verify that deployed webservers are up and running.



10. Un-deploy Apache server from client VMs by means of second curated playbook “undeploy_apache_webserver.yml”
([https://github.com/poojashreeNS/Assignments/blob/main/HW%20Ansible/HW%20231 By Charu/undeploy_apache_webserver.yml](https://github.com/poojashreeNS/Assignments/blob/main/HW%20Ansible/HW%20231%20By%20Charu/undeploy_apache_webserver.yml)).

```
ubuntu@ubuntu:/etc/ansible$ sudo nano undeploy_apache_webserver.yml
ubuntu@ubuntu:/etc/ansible$ ansible-playbook undeploy_apache_webserver.yml

PLAY [apache] *****
*

TASK [Gathering Facts] *****
*
ok: [10.0.0.61]
ok: [10.0.0.222]

TASK [Uninstall Apache server] *****
*
changed: [10.0.0.222]
changed: [10.0.0.61]

TASK [Remove leftover Apache2 packages] *****
*
changed: [10.0.0.222]
changed: [10.0.0.61]

PLAY RECAP *****
*
10.0.0.222      : ok=3    changed=2    unreachable=0    failed=0
skipped=0      rescued=0  ignored=0
10.0.0.61      : ok=3    changed=2    unreachable=0    failed=0
skipped=0      rescued=0  ignored=0

ubuntu@ubuntu:/etc/ansible$
```

11. Verify that webserver are not responding anymore.

