| | |
|---|---|
| **Name:** MANUEL, HAZEL AILLSON T. | **Date Performed:** Aug. 8, 2024 |
| **Course/Section:** CPE212 - CPE31S4 | **Date Submitted:** Aug. 8, 2024 |
| **Instructor:** ENGR. ROBIN VALENZUELA | **Semester and SY:** 1st Sem - 2025-2026 |

**Activity 1: Configure Network using Virtual Machines**

1. **Objectives:**

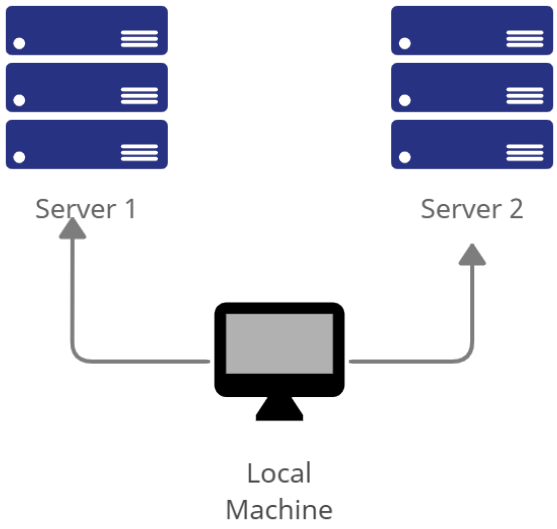
1.1. Create and configure Virtual Machines in Microsoft Azure or VirtualBox

1.2. Set-up a Virtual Network and Test Connectivity of VMs

2. **Discussion:**

**Network Topology:**

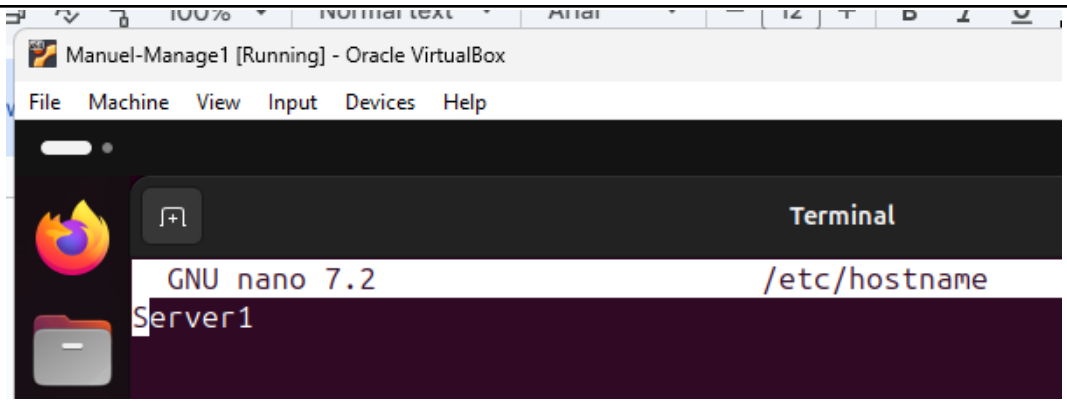
Assume that you have created the following network topology in Virtual Machines, *provide screenshots for each task*. (Note: *it is assumed that you have the prior knowledge of cloning and creating snapshots in a virtual machine*).



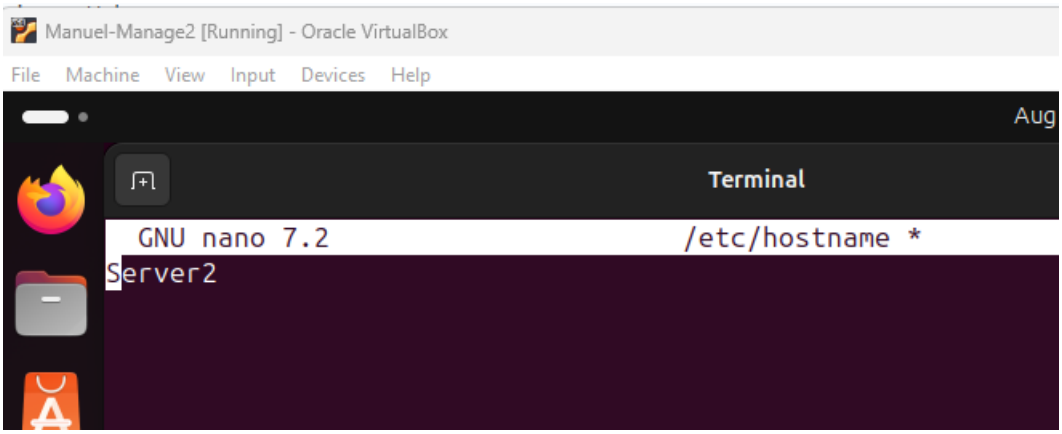

**Task 1**: Do the following on Server 1, Server 2, and Local Machine. In editing the file using nano command, press control + O to write out (save the file). Press enter when asked for the name of the file. Press control + X to end.

1. Change the hostname using the command *sudo nano /etc/hostname*

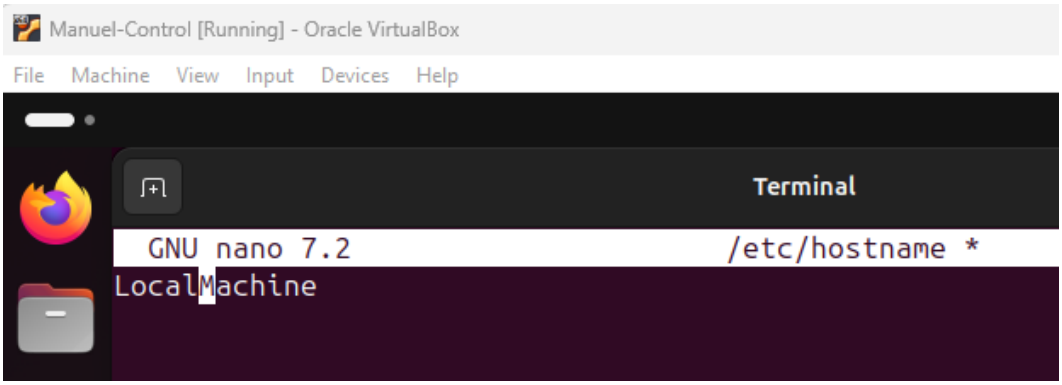
1.1 Use server1 for Server 1



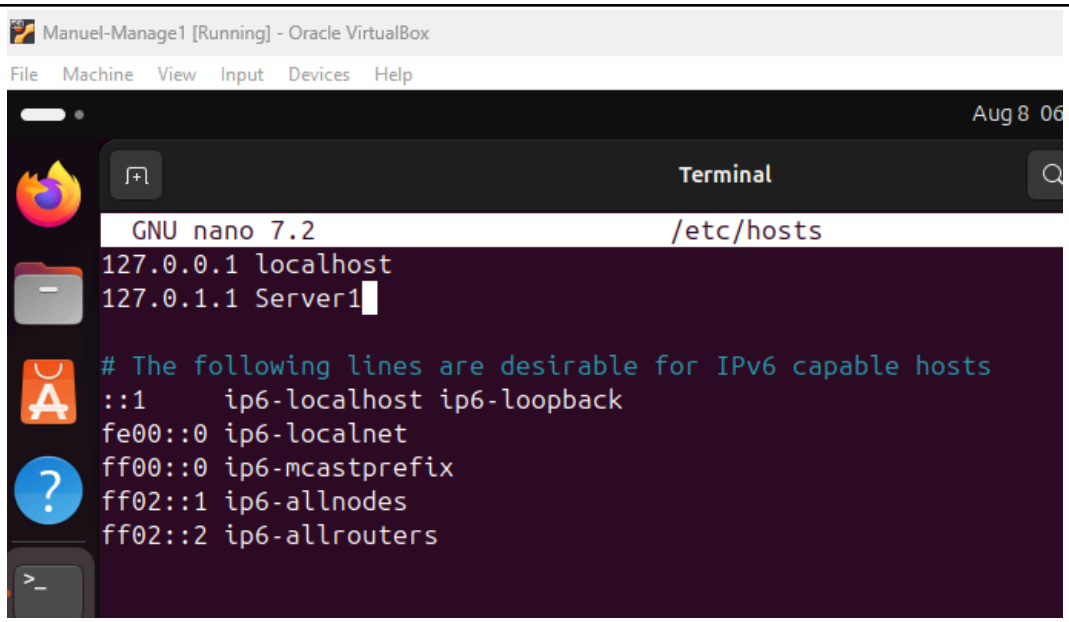

1.2 Use server2 for Server 2



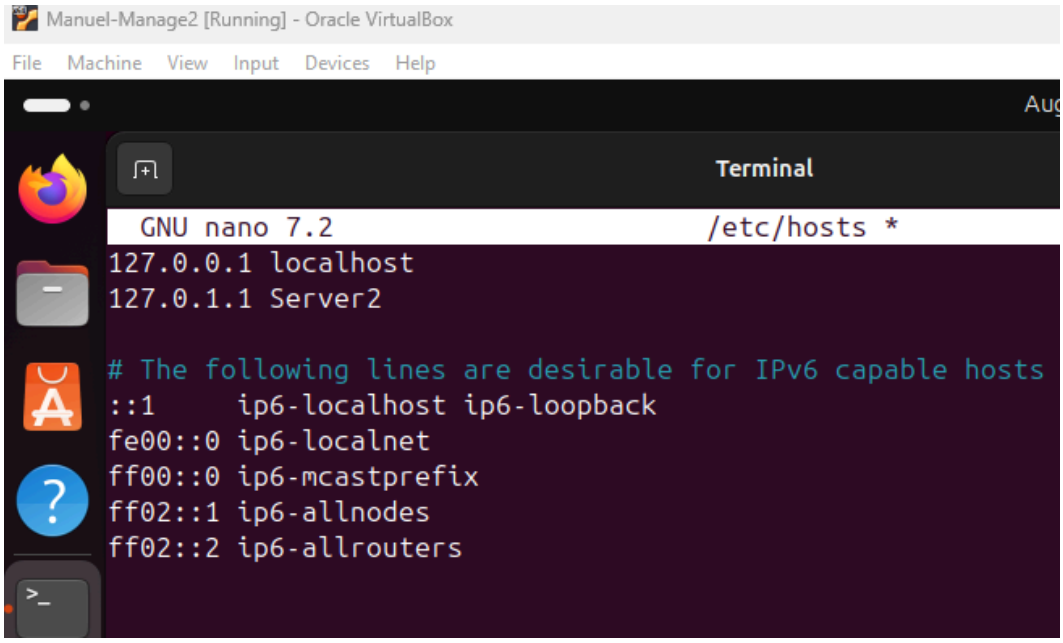

1.3 Use workstation for the Local Machine



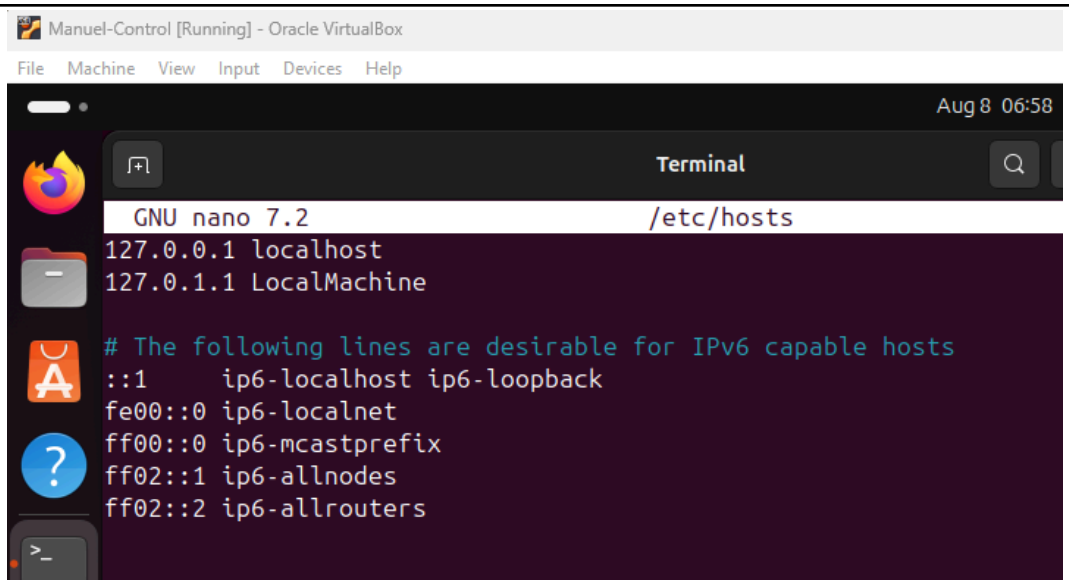

2. Edit the hosts using the command *sudo nano /etc/hosts*. Edit the second line.
   2.1 Type 127.0.0.1 server 1 for Server 1




2.2 Type 127.0.0.1 server 2 for Server 2




2.3 Type 127.0.0.1 workstation for the Local Machine




**Task 2**: Configure SSH on Server 1, Server 2, and Local Machine. Do the following:

1. Upgrade the packages by issuing the command *sudo apt update* and *sudo apt upgrade* respectively.

### *4.1.1 Server1 Update & Upgrade*

```
hazel@Server1:~$ sudo apt update
[sudo] password for hazel:
Hit:1 http://ph.archive.ubuntu.com/ubuntu noble InRelease
Hit:2 http://ph.archive.ubuntu.com/ubuntu noble-updates InRelease
Hit:3 http://security.ubuntu.com/ubuntu noble-security InRelease
Hit:4 http://ph.archive.ubuntu.com/ubuntu noble-backports InRelease
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
168 packages can be upgraded. Run 'apt list --upgradable' to see them.
```

### *4.1.2 Server2 Update & Upgrade*

```
hazel@Server2:~$ sudo apt update
[sudo] password for hazel:
Hit:1 http://ph.archive.ubuntu.com/ubuntu noble InRelease
Hit:2 http://ph.archive.ubuntu.com/ubuntu noble-updates InRelease
Hit:3 http://ph.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
168 packages can be upgraded. Run 'apt list --upgradable' to see them.
```

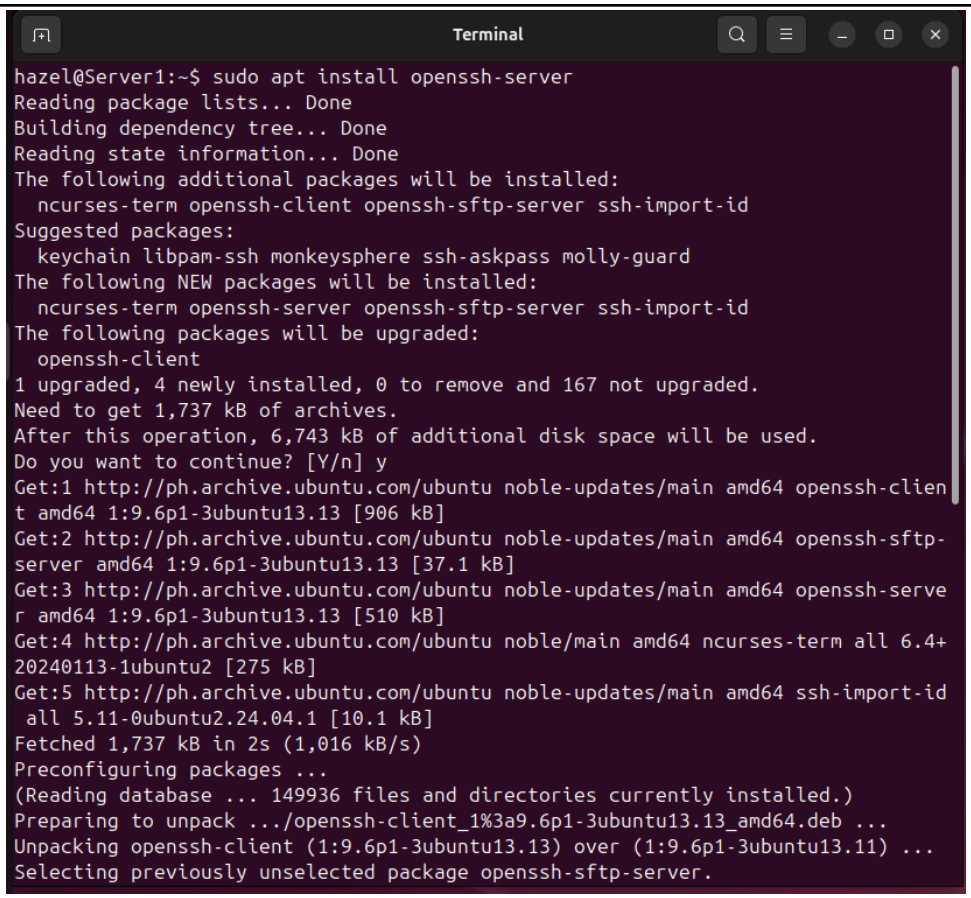
### *4.1.3 Local Machine Update & Upgrade*

```
hazel@TRY:~$ sudo apt update
Hit:1 http://ph.archive.ubuntu.com/ubuntu noble InRelease
Hit:2 http://ph.archive.ubuntu.com/ubuntu noble-updates InRelease
Hit:3 http://ph.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
168 packages can be upgraded. Run 'apt list --upgradable' to see them.
```

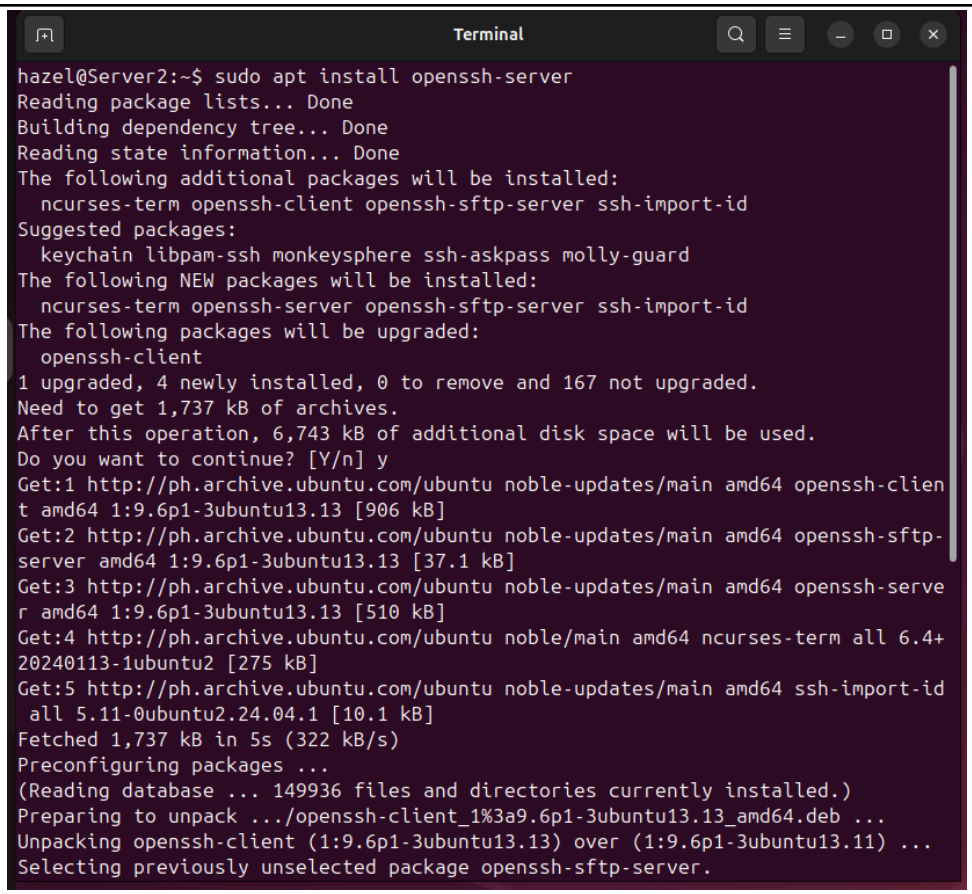
```
hazel@Server1:~$ sudo apt upgrade
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
Calculating upgrade... Done
The following packages were automatically installed and are no longer required:
  libgl1-amber-dri libglapi-mesa
Use 'sudo apt autoremove' to remove them.
The following packages have been kept back:
  libgl1-amber-dri
The following packages will be upgraded:
  alsa-ucm-conf apparmor apt apt-utils base-files bluez bluez-cups bluez-obexd bsdextrautils bsdutils
  distro-info-data dmsetup dns-root-data dnsmasq-base dracut-install eject fdisk
  fonts-noto-color-emoji fwupd gir1.2-gtk-3.0 gir1.2-gtk-4.0 gir1.2-mutter-14
  gir1.2-packagekitglib-1.0 gnome-calculator gnome-control-center gnome-control-center-data
  gnome-control-center-faces gnome-initial-setup gnome-shell gnome-shell-common
  gnome-shell-extension-desktop-icons-ng gnome-shell-extension-ubuntu-dock gstreamer1.0-packagekit
  gstreamer1.0-pipewire gtk-update-icon-cache gvfs gvfs-backends gvfs-common gvfs-daemons gvfs-fuse
  gvfs-libs gzip ibverbs-providers initramfs-tools initramfs-tools-bin initramfs-tools-core iproute2
```

2. Install the SSH server using the command *sudo apt install openssh-server*.

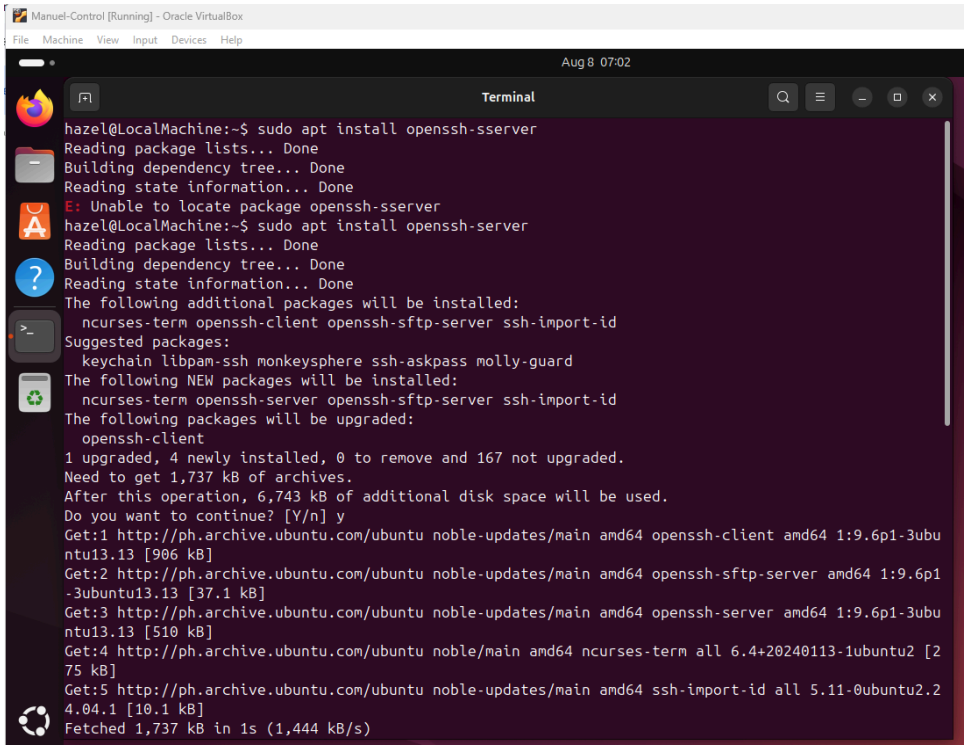
### 4.2.1 Server 1


Terminal
hazel@Server1:~$ sudo apt install openssh-server
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
The following additional packages will be installed:
ncurses-term openssh-client openssh-sftp-server ssh-import-id
Suggested packages:
keychain libpam-ssh monkeysphere ssh-askpass molly-guard
The following NEW packages will be installed:
ncurses-term openssh-server openssh-sftp-server ssh-import-id
The following packages will be upgraded:
openssh-client
1 upgraded, 4 newly installed, 0 to remove and 167 not upgraded.
Need to get 1,737 kB of archives.
After this operation, 6,743 kB of additional disk space will be used.
Do you want to continue? [Y/n] y
Get:1 http://ph.archive.ubuntu.com/ubuntu noble-updates/main amd64 openssh-client amd64 1:9.6p1-3ubuntu13.13 [906 kB]
Get:2 http://ph.archive.ubuntu.com/ubuntu noble-updates/main amd64 openssh-sftp-server amd64 1:9.6p1-3ubuntu13.13 [37.1 kB]
Get:3 http://ph.archive.ubuntu.com/ubuntu noble-updates/main amd64 openssh-server amd64 1:9.6p1-3ubuntu13.13 [510 kB]
Get:4 http://ph.archive.ubuntu.com/ubuntu noble/main amd64 ncurses-term all 6.4+20240113-1ubuntu2 [275 kB]
Get:5 http://ph.archive.ubuntu.com/ubuntu noble-updates/main amd64 ssh-import-id all 5.11-0ubuntu2.24.04.1 [10.1 kB]
Fetched 1,737 kB in 2s (1,016 kB/s)
Preconfiguring packages ...
(Reading database ... 149936 files and directories currently installed.)
Preparing to unpack .../openssh-client_1%3a9.6p1-3ubuntu13.13_amd64.deb ...
Unpacking openssh-client (1:9.6p1-3ubuntu13.13) over (1:9.6p1-3ubuntu13.11) ...
Selecting previously unselected package openssh-sftp-server.

### 4.2.2 Server 2

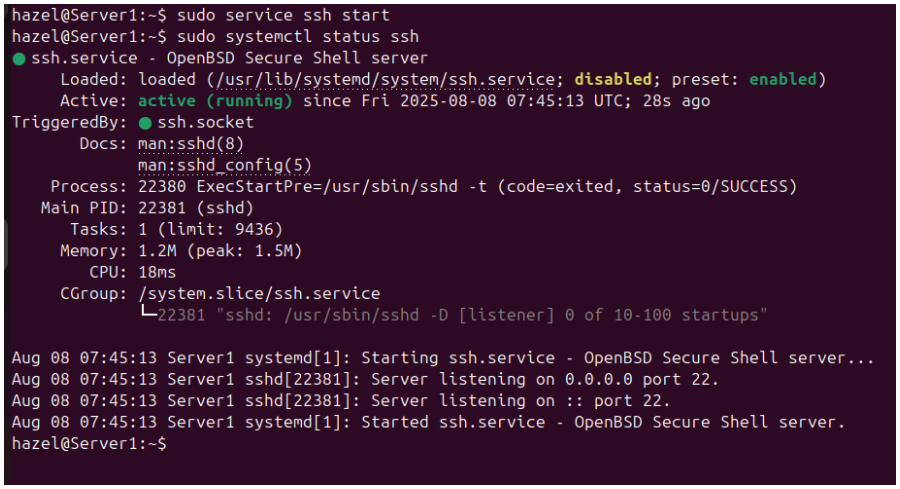



### 4.2.3 Local Machine

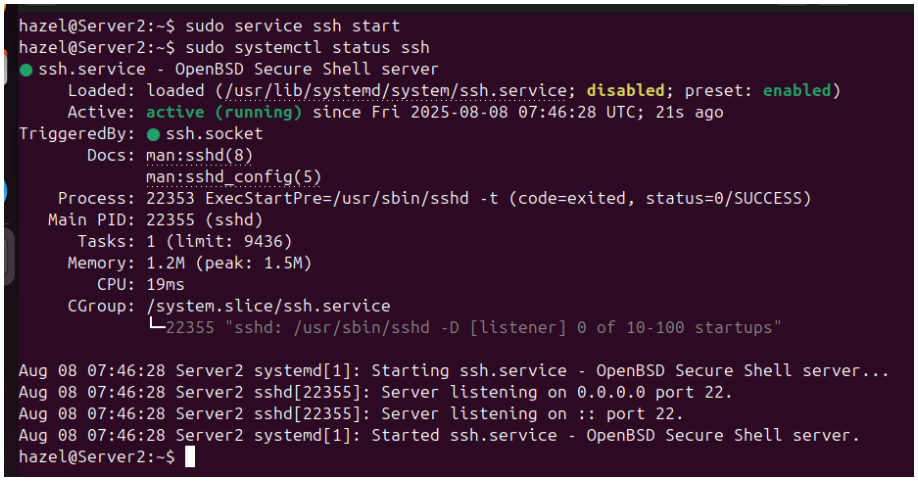


3. Verify if the SSH service has started by issuing the following commands:
   3.1 *sudo service ssh start*
   3.2 *sudo systemctl status ssh*

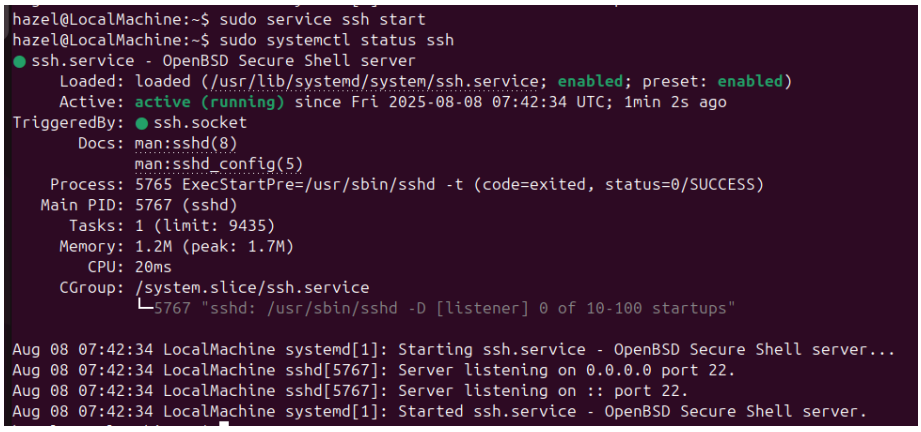
### *3.2.1 Server 1*




### *3.2.2 Server 2*




### *3.2.3 Local Machine*




4. Configure the firewall to all port 22 by issuing the following commands:

4.1 *sudo ufw allow ssh*

*4.1.1 Server 1*

```
hazel@Server1:~$ sudo ufw allow ssh
Rules updated
Rules updated (v6)
```

*4.1.2 Server 2*

```
hazel@Server2:~$ sudo ufw allow ssh
Rules updated
Rules updated (v6)
```

*4.1.3 Local Machine*

```
hazel@LocalMachine:~$ sudo ufw allow ssh
Rules updated
Rules updated (v6)
```

4.2 *sudo ufw enable*

*4.2.1 Server 1*

```
hazel@Server1:~$ sudo ufw enable
Firewall is active and enabled on system startup
```

*4.2.2 Server 2*

```
hazel@Server2:~$ sudo ufw enable
Firewall is active and enabled on system startup
```

*4.2.3 Local Machine*

```
hazel@LocalMachine:~$ sudo ufw enable
Firewall is active and enabled on system startup
```

4.3 *sudo ufw status*

*4.3.1 Server 1*

```
hazel@Server1:~$ sudo ufw status
Status: active

To                         Action      From
--                         ------      ----
22/tcp                     ALLOW       Anywhere
22/tcp (v6)                ALLOW       Anywhere (v6)
```

*4.3.2 Server 2*

```
hazel@Server2:~$ sudo ufw status
Status: active

To                         Action      From
--                         ------      ----
22/tcp                     ALLOW       Anywhere
22/tcp (v6)                ALLOW       Anywhere (v6)
```

*4.3.3 Local Machine*

```
hazel@LocalMachine:~$ sudo ufw status
Status: active

To                         Action      From
--                         ------      ----
22/tcp                     ALLOW       Anywhere
22/tcp (v6)                ALLOW       Anywhere (v6)
```

**Task 3:** Verify network settings on Server 1, Server 2, and Local Machine. On each device, do the following:

1. Record the ip address of Server 1, Server 2, and Local Machine. Issue the command *ifconfig* and check network settings. Note that the ip addresses of all the machines are in this network 192.168.56.XX.
   1.1 Server 1 IP address: 192.168.56.**105**
   1.2 Server 2 IP address: 192.168.56.**106**
   1.3 Server 3 IP address: 192.168.56.**107**
2. Make sure that they can ping each other.
   2.1 Connectivity test for Local Machine 1 to Server 1: ✔ **Successful** ☐ Not Successful

```
hazel@LocalMachine:~$ ping 192.168.56.105
PING 192.168.56.105 (192.168.56.105) 56(84) bytes of data.
64 bytes from 192.168.56.105: icmp_seq=1 ttl=64 time=1.09 ms
64 bytes from 192.168.56.105: icmp_seq=2 ttl=64 time=0.705 ms
64 bytes from 192.168.56.105: icmp_seq=3 ttl=64 time=0.545 ms
64 bytes from 192.168.56.105: icmp_seq=4 ttl=64 time=4.05 ms
64 bytes from 192.168.56.105: icmp_seq=5 ttl=64 time=0.453 ms
^C
--- 192.168.56.105 ping statistics ---
5 packets transmitted, 5 received, 0% packet loss, time 6941ms
rtt min/avg/max/mdev = 0.453/1.367/4.045/1.356 ms
```

   2.2 Connectivity test for Local Machine 1 to Server 2: ✔ **Successful** ☐ Not Successful

```
hazel@LocalMachine:~$ ping 192.168.56.106
PING 192.168.56.106 (192.168.56.106) 56(84) bytes of data.
64 bytes from 192.168.56.106: icmp_seq=1 ttl=64 time=0.414 ms
64 bytes from 192.168.56.106: icmp_seq=2 ttl=64 time=0.503 ms
64 bytes from 192.168.56.106: icmp_seq=3 ttl=64 time=0.531 ms
64 bytes from 192.168.56.106: icmp_seq=4 ttl=64 time=0.444 ms
64 bytes from 192.168.56.106: icmp_seq=5 ttl=64 time=0.469 ms
^C
--- 192.168.56.106 ping statistics ---
5 packets transmitted, 5 received, 0% packet loss, time 4109ms
rtt min/avg/max/mdev = 0.414/0.472/0.531/0.041 ms
```

2.3 Connectivity test for Server 1 to Server 2: ✔ **Successful** ☐ Not Successful

```
hazel@Server1:~$ ping 192.168.56.106
PING 192.168.56.106 (192.168.56.106) 56(84) bytes of data.
64 bytes from 192.168.56.106: icmp_seq=1 ttl=64 time=1.08 ms
64 bytes from 192.168.56.106: icmp_seq=2 ttl=64 time=0.500 ms
64 bytes from 192.168.56.106: icmp_seq=3 ttl=64 time=0.421 ms
^C
--- 192.168.56.106 ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 2063ms
rtt min/avg/max/mdev = 0.421/0.666/1.078/0.292 ms
```

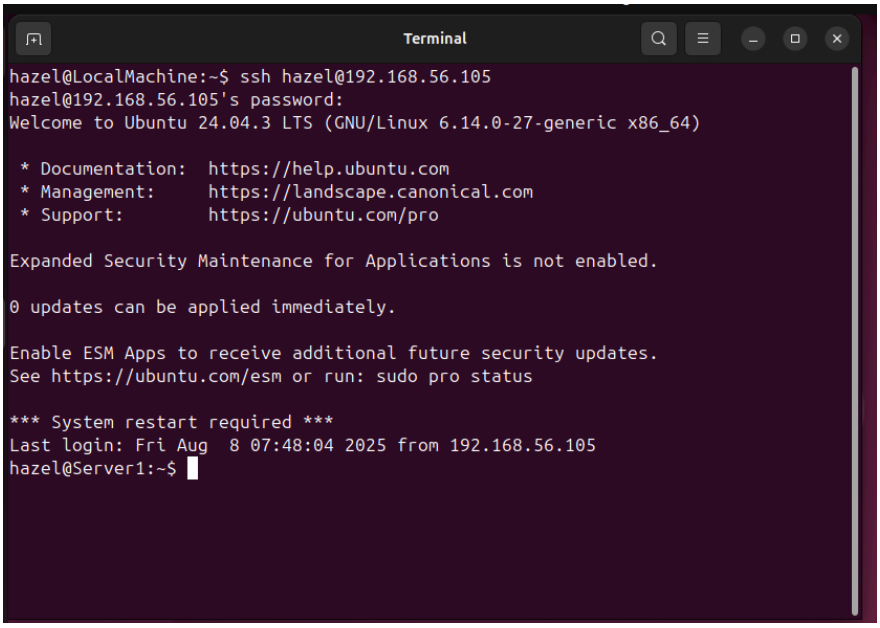
**Task 4:** Verify SSH connectivity on Server 1, Server 2, and Local Machine.

1. On the Local Machine, issue the following commands:

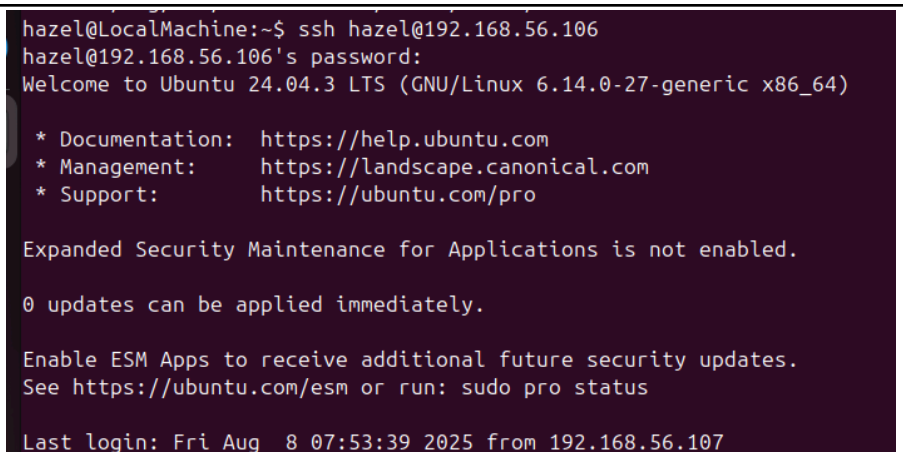
1.1 ssh username@ip_address_server1 for example, *ssh jvtaylar@192.168.56.120*

1.2 Enter the password for server 1 when prompted

**A. LOCAL MACHINE TO SERVER 1**




B. LOCAL MACHINE TO SEVER 2



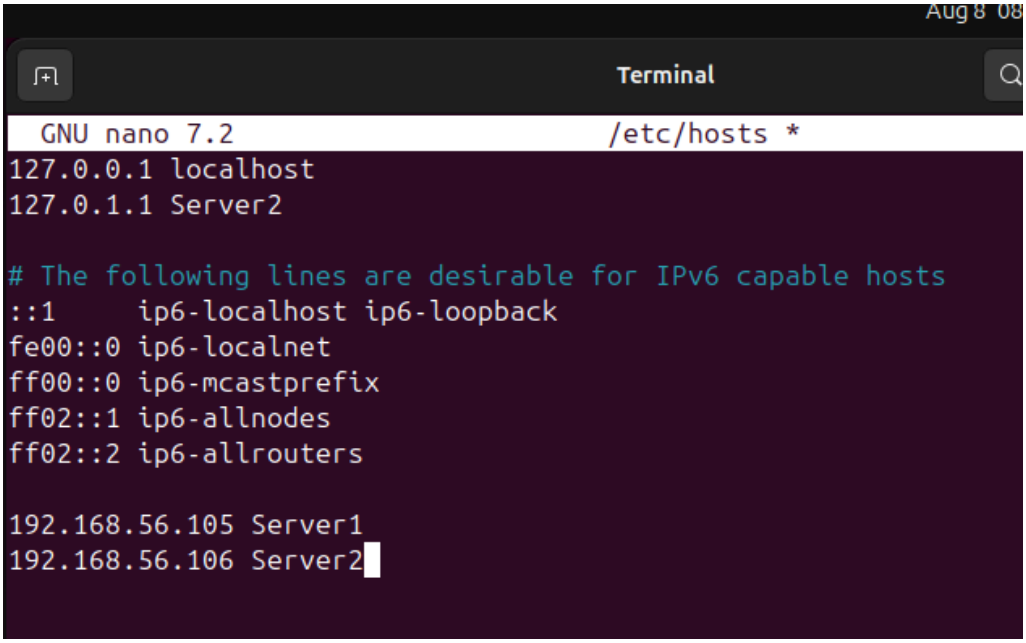

1.3 Verify that you are in server 1. The user should be in this format user@server1. For example, *jvtaylar@server1*

2. Logout of Server 1 by issuing the command *control + D.*
3. Do the same for Server 2.
4. Edit the hosts of the Local Machine by issuing the command *sudo nano /etc/hosts.* Below all texts type the following:

4.1 IP_address server 1 (provide the ip address of server 1 followed by the hostname)

4.2 IP_address server 2 (provide the ip address of server 2 followed by the hostname)

4.3 Save the file and exit.




5. On the local machine, verify that you can do the SSH command but this time, use the hostname instead of typing the IP address of the servers. For example,

try to do *ssh jvtaylar@server1*. Enter the password when prompted. Verify that you have entered Server 1. Do the same for Server 2.

SERVER1

```
hazel@Server2:~$ ssh hazel@Server1
The authenticity of host 'server1 (192.168.56.105)' can't be established.
ED25519 key fingerprint is SHA256:Yg7ZsPaRANg1nxq3MmM1ZMkcttV82oYkcM9KQAkonoM.
This key is not known by any other names.
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added 'server1' (ED25519) to the list of known hosts.
hazel@server1's password:
Welcome to Ubuntu 24.04.3 LTS (GNU/Linux 6.14.0-27-generic x86_64)

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

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

Last login: Fri Aug  8 07:49:37 2025 from 192.168.56.107
```

SERVER 2

```
hazel@Server1:~$ ssh hazel@Server2
The authenticity of host 'server2 (192.168.56.106)' can't be established.
ED25519 key fingerprint is SHA256:PqiA4qU1NCtx6dFlb5V1D/CLLYp+sM3kK6yxNsqHjw4.
This host key is known by the following other names/addresses:
    ~/.ssh/known_hosts:4: [hashed name]
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added 'server2' (ED25519) to the list of known hosts.
hazel@server2's password:
Welcome to Ubuntu 24.04.3 LTS (GNU/Linux 6.14.0-27-generic x86_64)

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

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

Last login: Fri Aug  8 08:09:05 2025 from 192.168.56.107
```

**Reflections:**

Answer the following:

1. How are we able to use the hostname instead of IP address in SSH commands?
   **We are able to use the hostname as we established an ssh connectivity to the other hosts. We also ensured that in the /etc/hosts that the hostname is**

**paired with the IP Address. Thus, when calling it with the hostname, we are able to use SSH Commands without the need to use the hosts IP addresses.**

2. How secured is SSH?

   **SSH is considered to be secure as this basically encrypts all of the data that's being transmitted between the client and the server. This ensures that all of the files, passwords, usernames, etc. are encrypted. This ensures that all of the files will not be accessed to those unauthorized.**