Name: Flores, Joshua Mico A.	**Date Performed:08/14/23**
Course/Section: CPE31S4	**Date Submitted:08/15/23**
Instructor: Dr. Jonathan V. Taylar	**Semester and SY: 1st sem - 2023 - 2024**

Activity 1: Configure Network using Virtual Machines

1. **Objectives:**

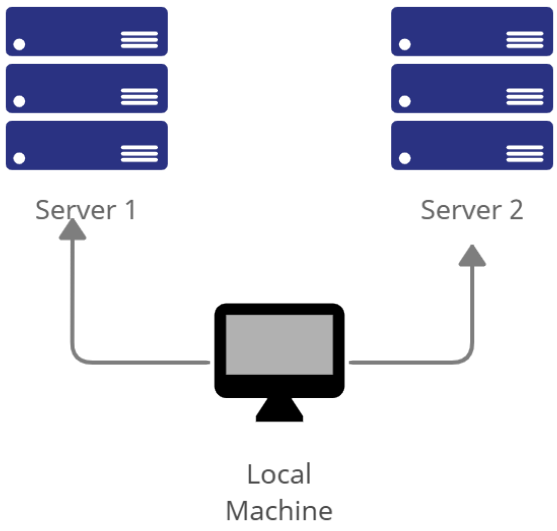
1.1. Create and configure Virtual Machines in Microsoft Azure or VirtualBox

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

2. **Discussion:**

Network Topology:

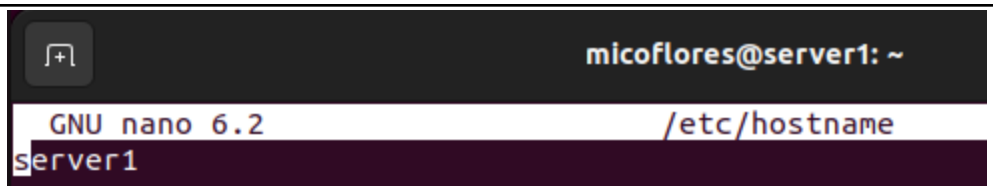
Assume that you have created the following network topology in Virtual Machines, *provide screenshots for each task*. (Note: *it is assumed that you have the prior knowledge of cloning and creating snapshots in a virtual machine*).



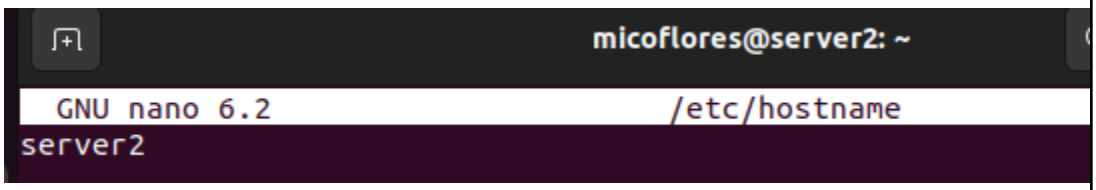
Task 1: Do the following on Server 1, Server 2, and Local Machine. In editing the file using nano command, press control + O to write out (save the file). Press enter when asked for the name of the file. Press control + X to end.

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

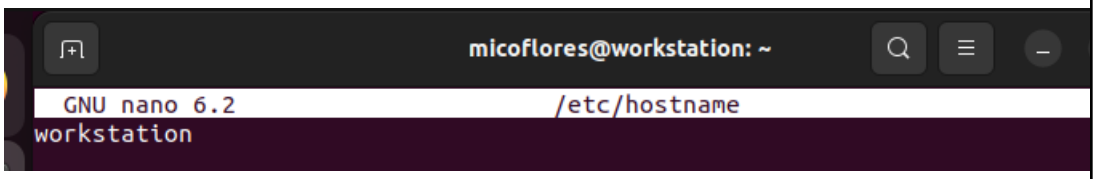
1.1 Use server1 for Server 1



1.2 Use server2 for Server 2

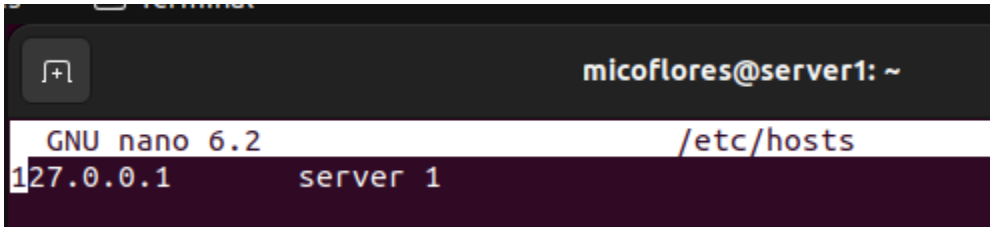


1.3 Use workstation for the Local Machine

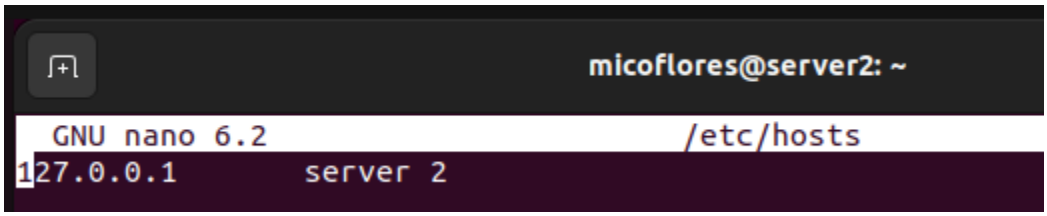


2. Edit the hosts using the command *sudo nano /etc/hosts*. Edit the second line.

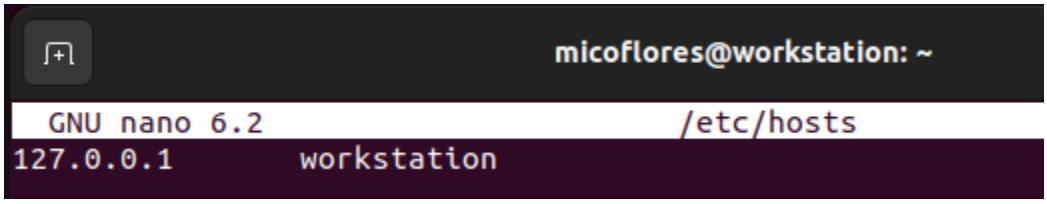
2.1 Type 127.0.0.1 server 1 for Server 1



2.2 Type 127.0.0.1 server 2 for Server 2

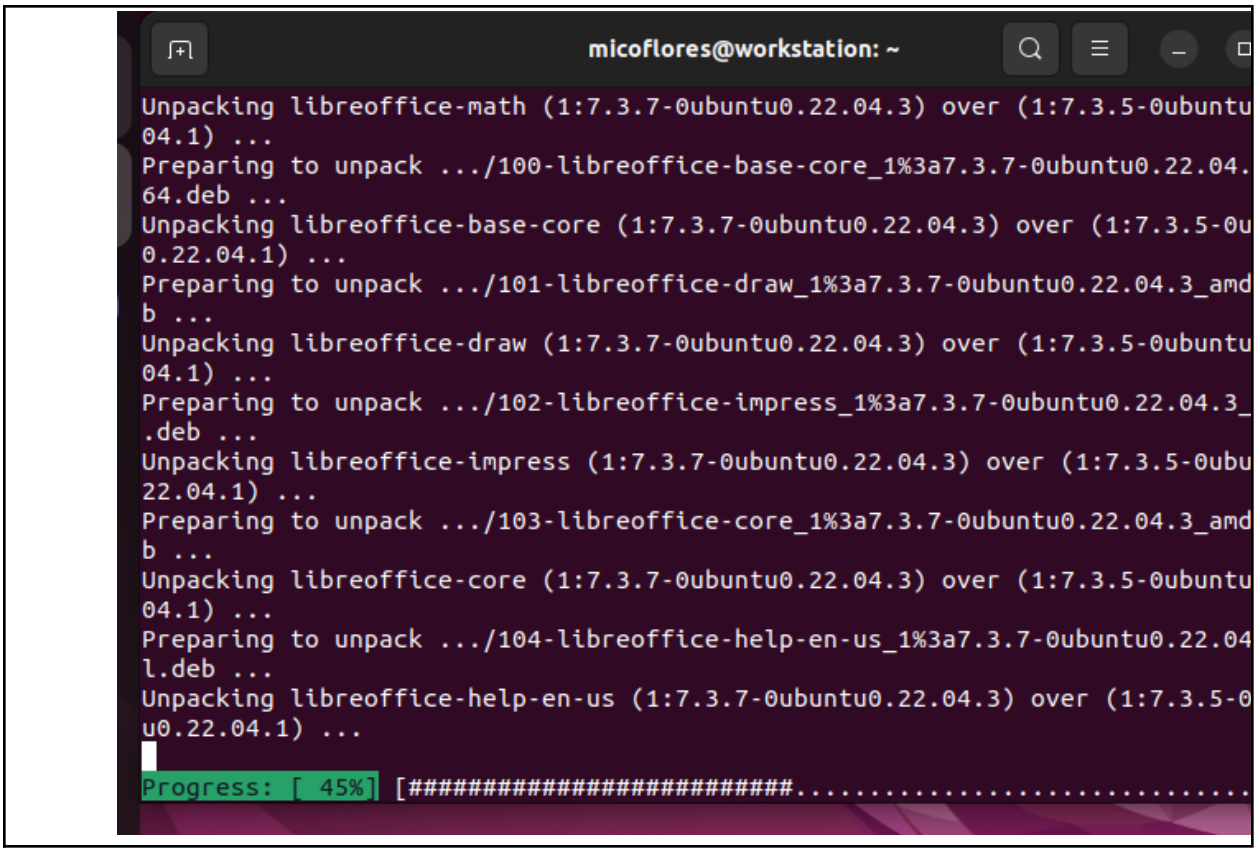


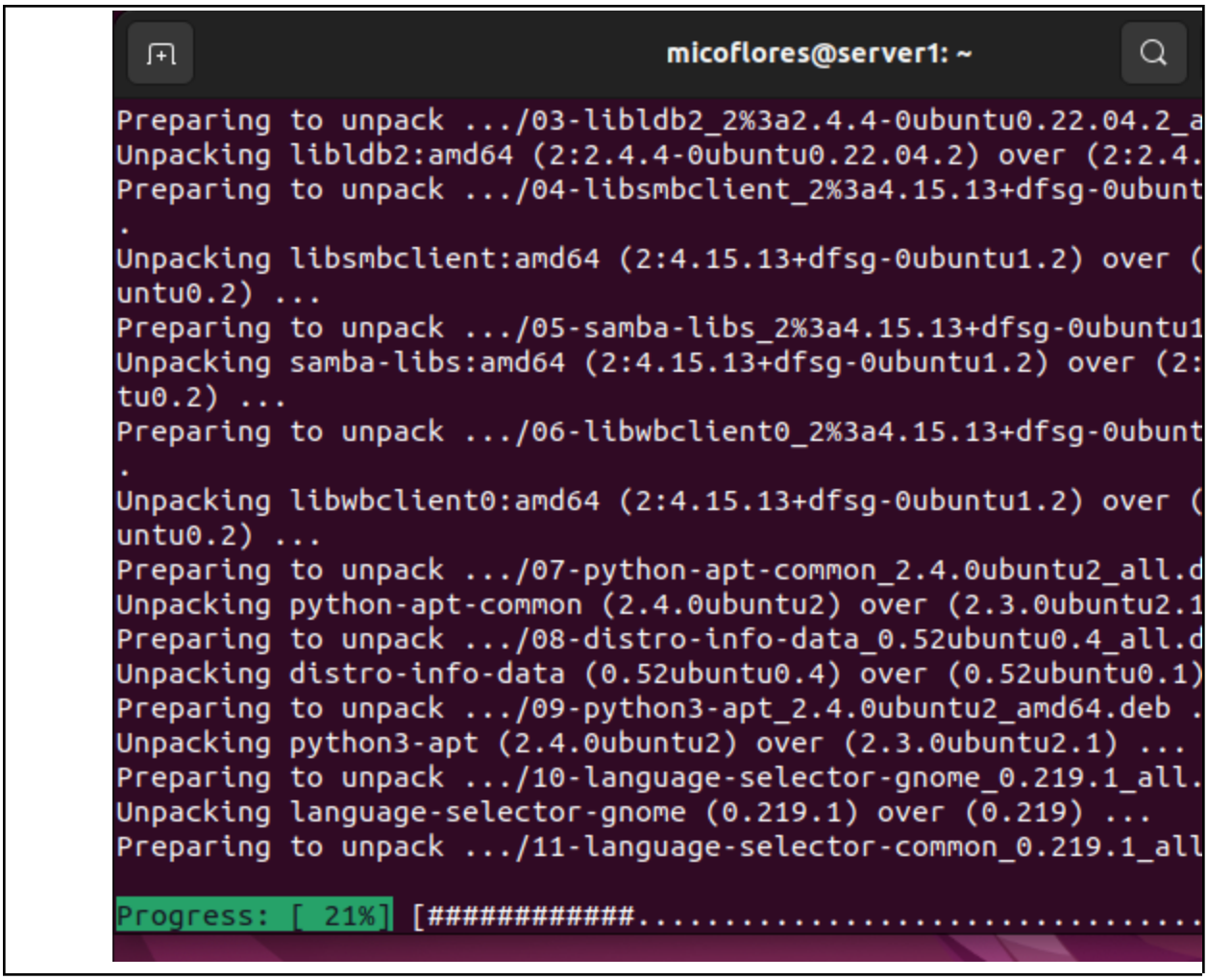
2.3 Type 127.0.0.1 workstation for the Local Machine

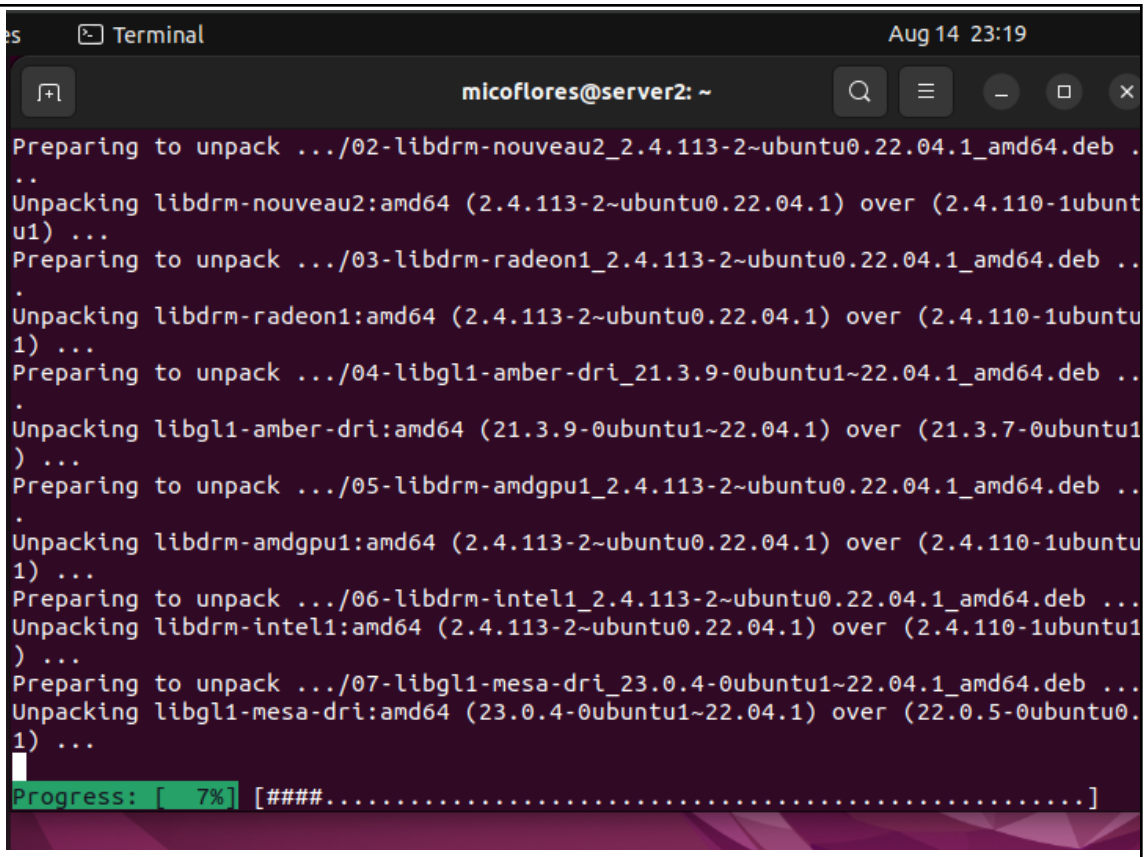


Task 2: Configure SSH on Server 1, Server 2, and Local Machine. Do the following:

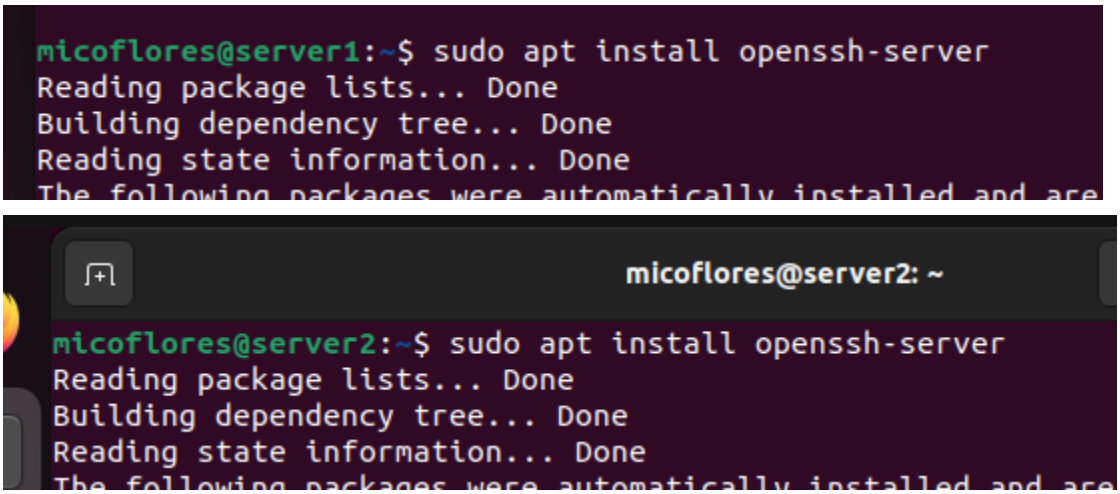
1. Upgrade the packages by issuing the command *sudo apt update* and *sudo apt upgrade* respectively.

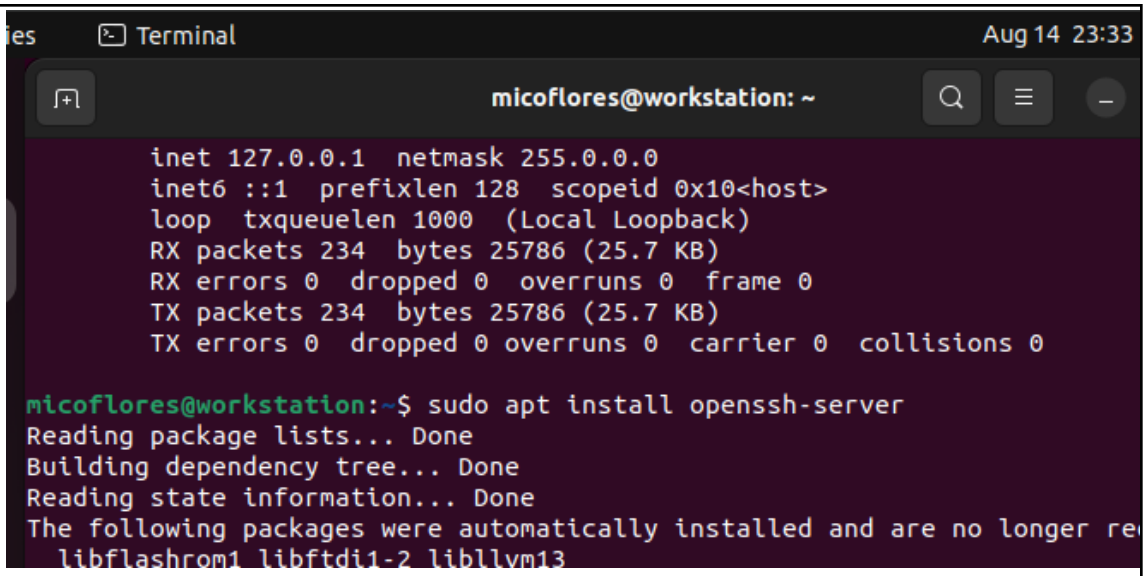

micoflores@workstation: ~
Unpacking libreoffice-math (1:7.3.7-0ubuntu0.22.04.3) over (1:7.3.5-0ubuntu
04.1) ...
Preparing to unpack .../100-libreoffice-base-core_1%3a7.3.7-0ubuntu0.22.04.
64.deb ...
Unpacking libreoffice-base-core (1:7.3.7-0ubuntu0.22.04.3) over (1:7.3.5-0u
0.22.04.1) ...
Preparing to unpack .../101-libreoffice-draw_1%3a7.3.7-0ubuntu0.22.04.3_amd
b ...
Unpacking libreoffice-draw (1:7.3.7-0ubuntu0.22.04.3) over (1:7.3.5-0ubuntu
04.1) ...
Preparing to unpack .../102-libreoffice-impress_1%3a7.3.7-0ubuntu0.22.04.3_
.deb ...
Unpacking libreoffice-impress (1:7.3.7-0ubuntu0.22.04.3) over (1:7.3.5-0ubu
22.04.1) ...
Preparing to unpack .../103-libreoffice-core_1%3a7.3.7-0ubuntu0.22.04.3_amd
b ...
Unpacking libreoffice-core (1:7.3.7-0ubuntu0.22.04.3) over (1:7.3.5-0ubuntu
04.1) ...
Preparing to unpack .../104-libreoffice-help-en-us_1%3a7.3.7-0ubuntu0.22.04
l.deb ...
Unpacking libreoffice-help-en-us (1:7.3.7-0ubuntu0.22.04.3) over (1:7.3.5-0
u0.22.04.1) ...
Progress: [45%] [#########################.....................


micoflores@server1: ~
Preparing to unpack .../03-libldb2_2%3a2.4.4-0ubuntu0.22.04.2_a
Unpacking libldb2:amd64 (2:2.4.4-0ubuntu0.22.04.2) over (2:2.4.
Preparing to unpack .../04-libsmbclient_2%3a4.15.13+dfsg-0ubunt
.
Unpacking libsmbclient:amd64 (2:4.15.13+dfsg-0ubuntu1.2) over (
untu0.2) ...
Preparing to unpack .../05-samba-libs_2%3a4.15.13+dfsg-0ubuntu1
Unpacking samba-libs:amd64 (2:4.15.13+dfsg-0ubuntu1.2) over (2:
tu0.2) ...
Preparing to unpack .../06-libwbclient0_2%3a4.15.13+dfsg-0ubunt
.
Unpacking libwbclient0:amd64 (2:4.15.13+dfsg-0ubuntu1.2) over (
untu0.2) ...
Preparing to unpack .../07-python-apt-common_2.4.0ubuntu2_all.d
Unpacking python-apt-common (2.4.0ubuntu2) over (2.3.0ubuntu2.1
Preparing to unpack .../08-distro-info-data_0.52ubuntu0.4_all.d
Unpacking distro-info-data (0.52ubuntu0.4) over (0.52ubuntu0.1)
Preparing to unpack .../09-python3-apt_2.4.0ubuntu2_amd64.deb .
Unpacking python3-apt (2.4.0ubuntu2) over (2.3.0ubuntu2.1) ...
Preparing to unpack .../10-language-selector-gnome_0.219.1_all.
Unpacking language-selector-gnome (0.219.1) over (0.219) ...
Preparing to unpack .../11-language-selector-common_0.219.1_all
Progress: [21%] [############.....................................

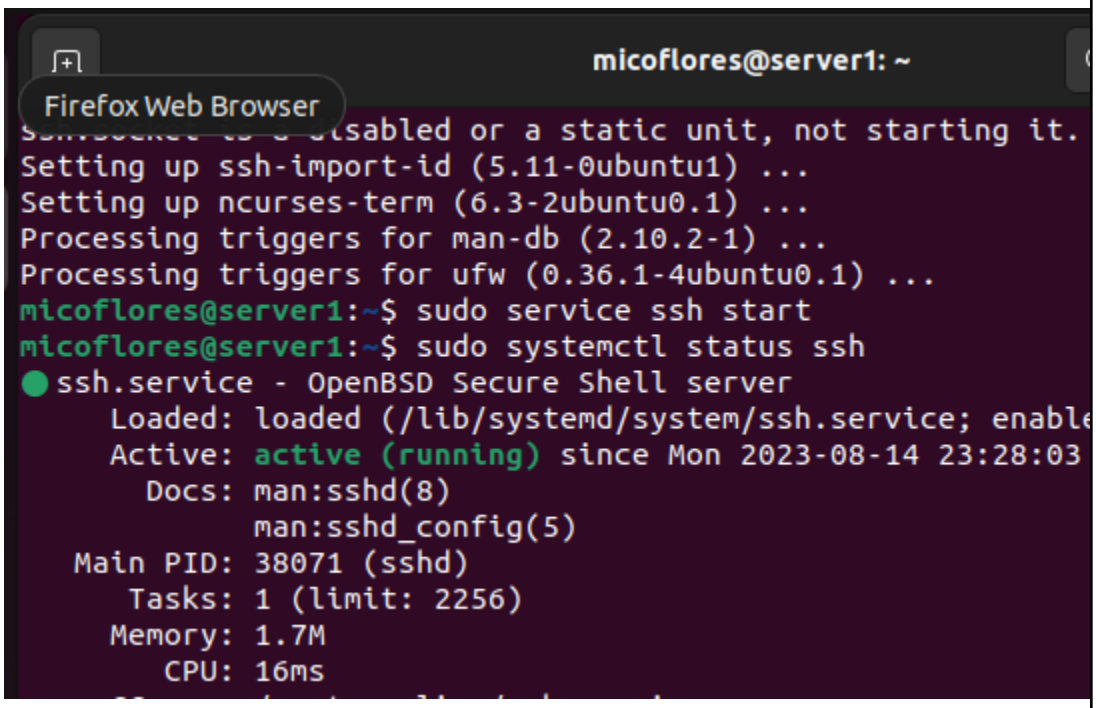


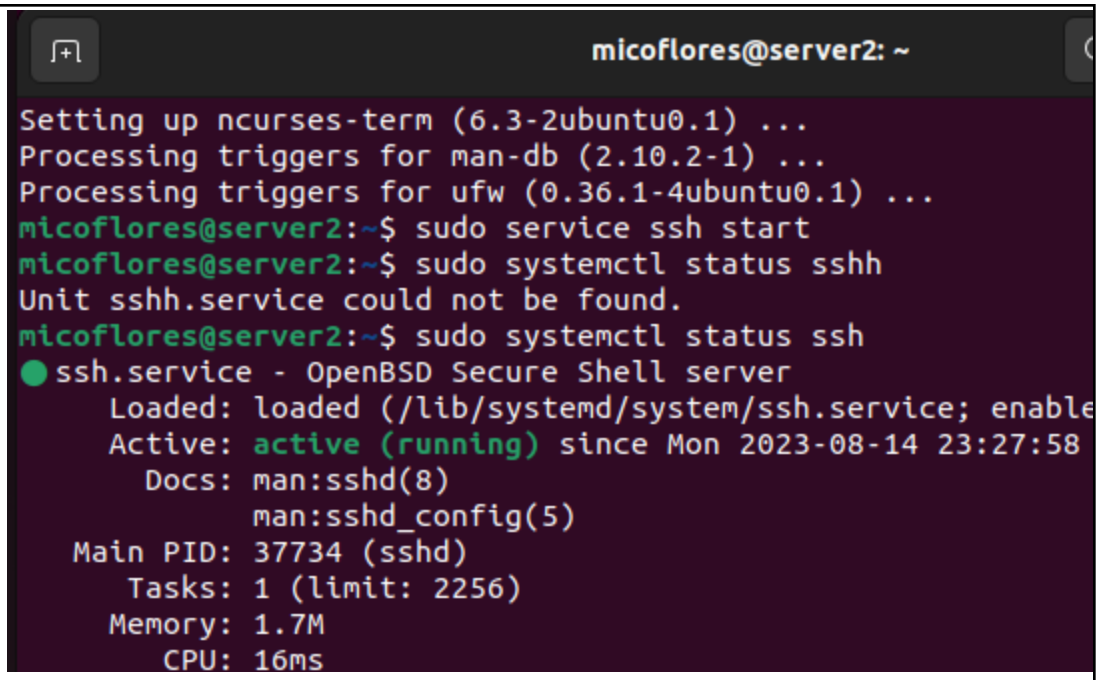
2. Install the SSH server using the command *sudo apt install openssh-server*.

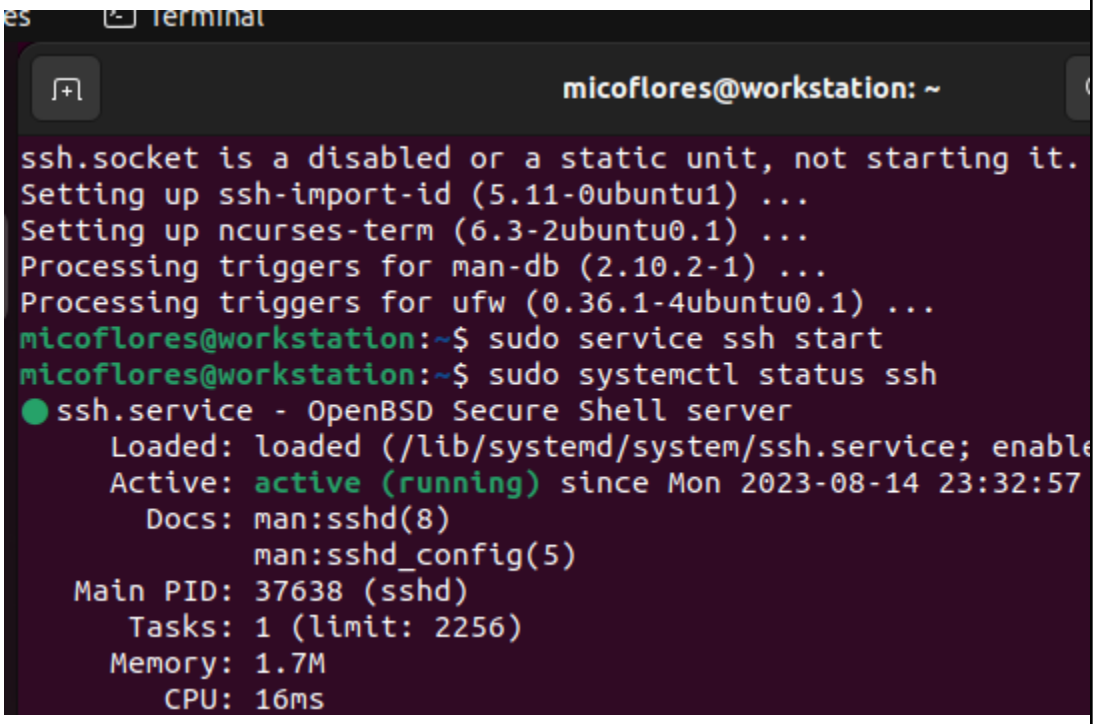




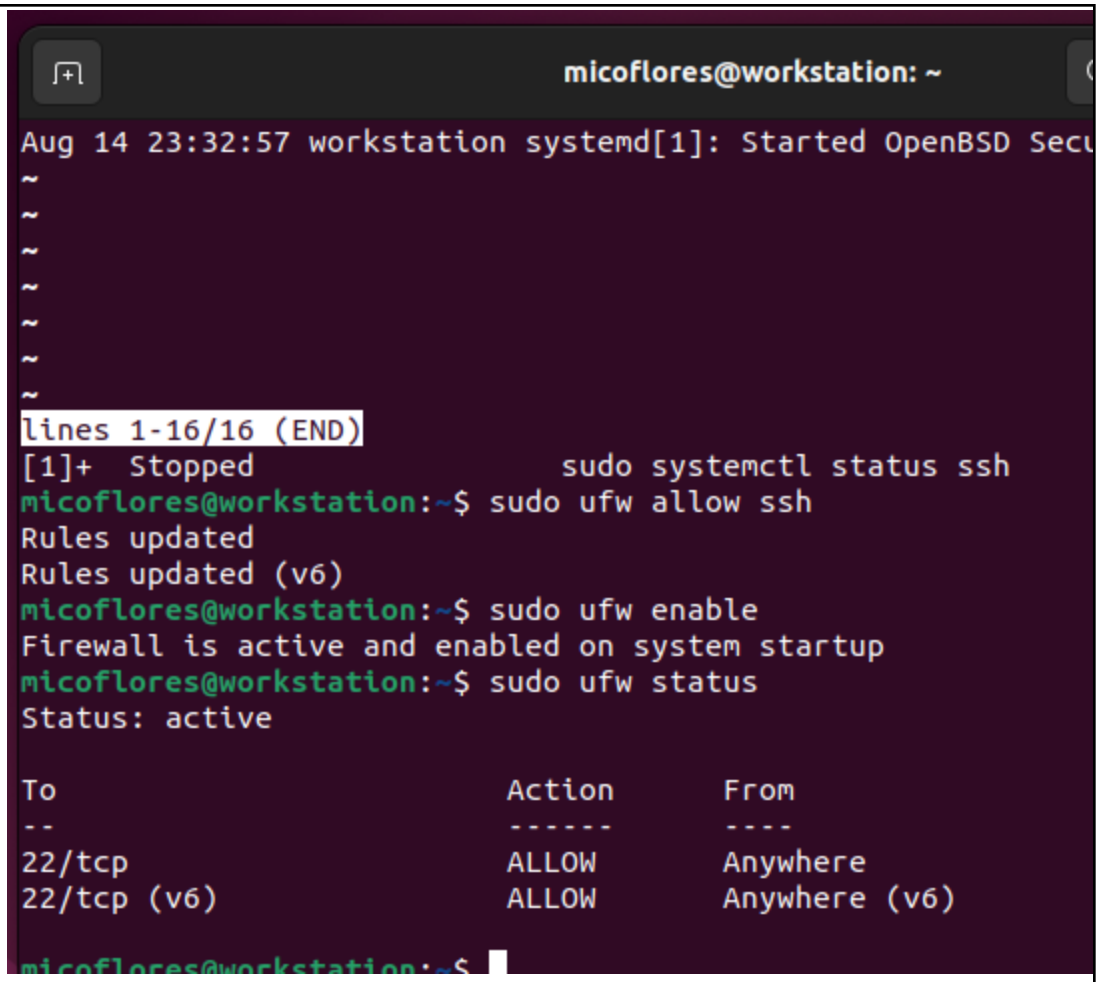
3. Verify if the SSH service has started by issuing the following commands:
   3.1 *sudo service ssh start*
   3.2 *sudo systemctl status ssh*

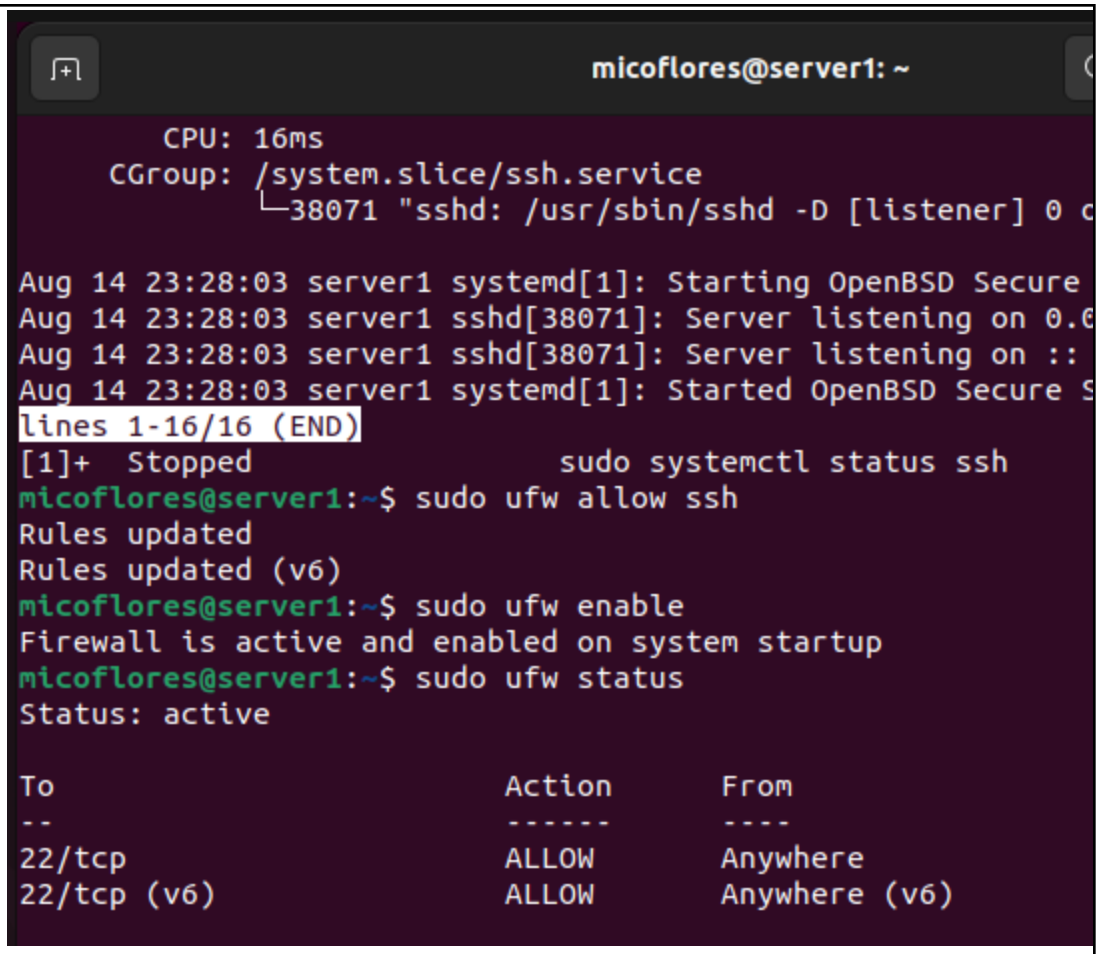


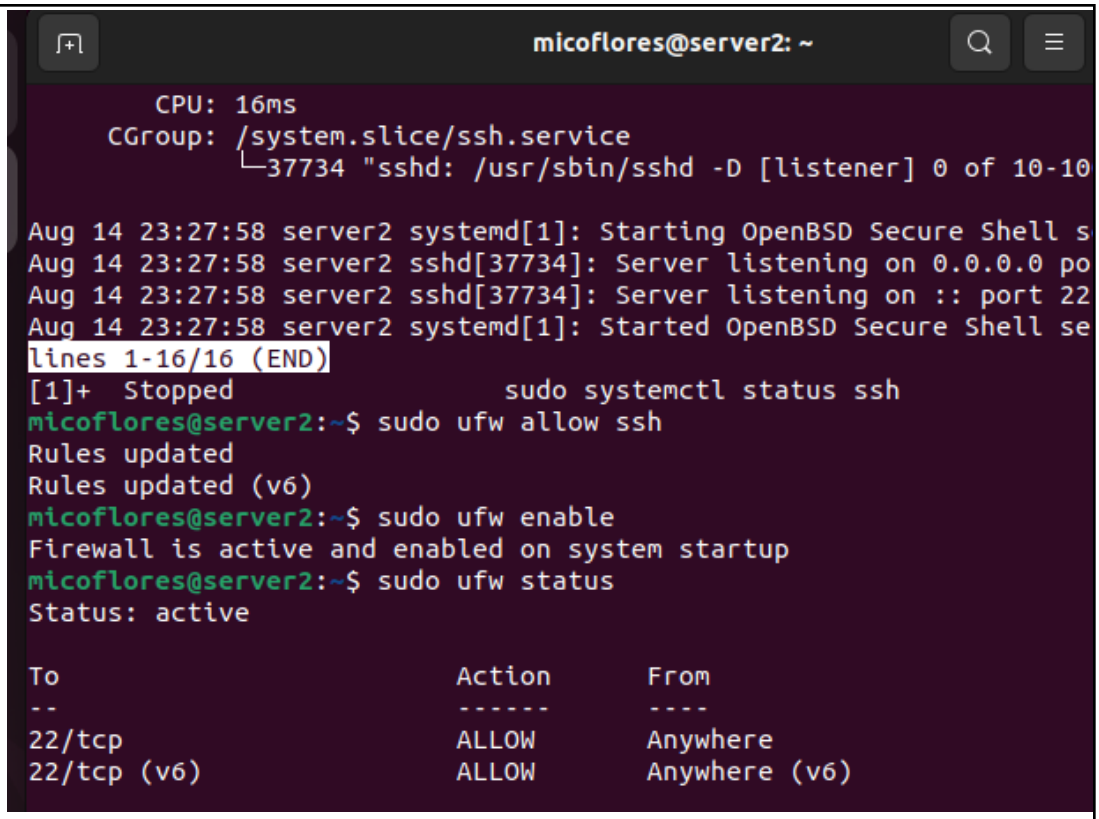




4. Configure the firewall to all port 22 by issuing the following commands:
   4.1 *sudo ufw allow ssh*
   4.2 *sudo ufw enable*
   4.3 *sudo ufw status*


micoflores@workstation: ~
Aug 14 23:32:57 workstation systemd[1]: Started OpenBSD Secu
~
~
~
~
~
~
~
lines 1-16/16 (END)
[1]+ Stopped sudo systemctl status ssh
micoflores@workstation:~$ sudo ufw allow ssh
Rules updated
Rules updated (v6)
micoflores@workstation:~$ sudo ufw enable
Firewall is active and enabled on system startup
micoflores@workstation:~$ sudo ufw status
Status: active
To Action From
-- ------ ----
22/tcp ALLOW Anywhere
22/tcp (v6) ALLOW Anywhere (v6)
micoflores@workstation:~$


micoflores@server1: ~
CPU: 16ms
CGroup: /system.slice/ssh.service
└─38071 "sshd: /usr/sbin/sshd -D [listener] 0 c
Aug 14 23:28:03 server1 systemd[1]: Starting OpenBSD Secure
Aug 14 23:28:03 server1 sshd[38071]: Server listening on 0.0
Aug 14 23:28:03 server1 sshd[38071]: Server listening on ::
Aug 14 23:28:03 server1 systemd[1]: Started OpenBSD Secure S
lines 1-16/16 (END)
[1]+ Stopped sudo systemctl status ssh
micoflores@server1:~$ sudo ufw allow ssh
Rules updated
Rules updated (v6)
micoflores@server1:~$ sudo ufw enable
Firewall is active and enabled on system startup
micoflores@server1:~$ 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

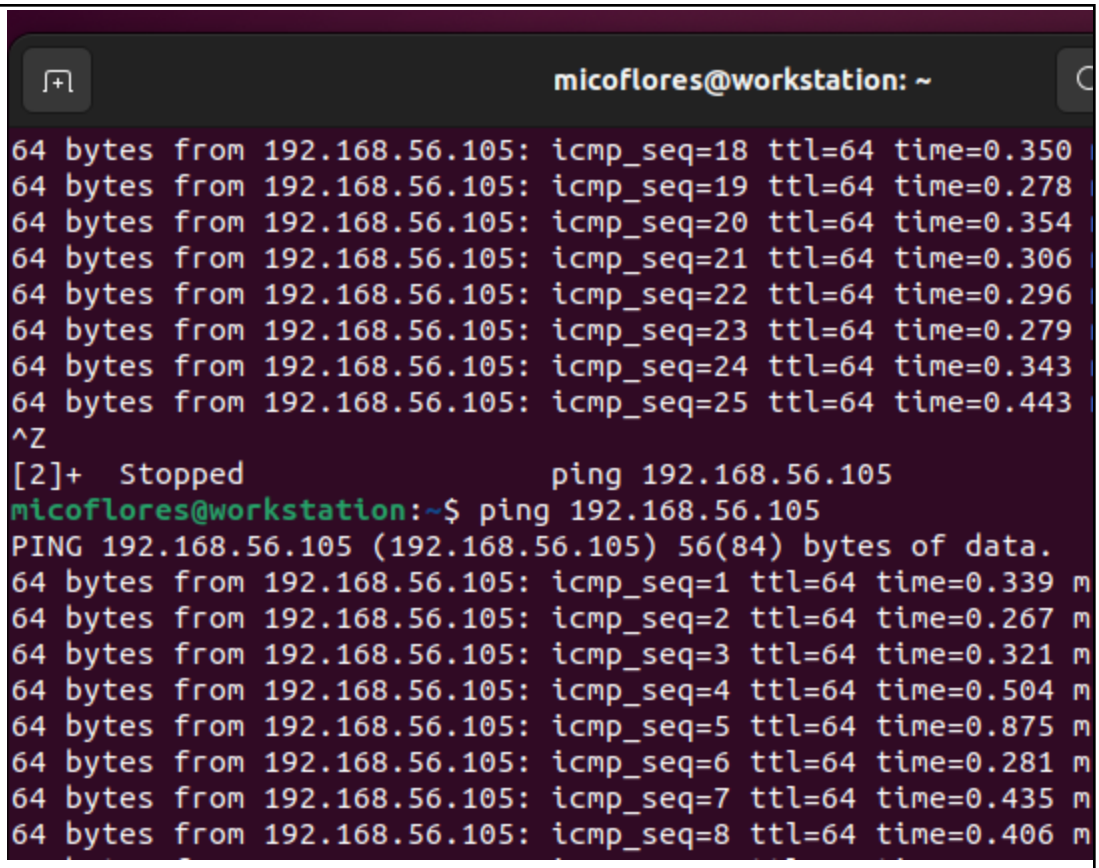
   `192.168.56.105`

   1.2 Server 2 IP address: 192.168.56.106

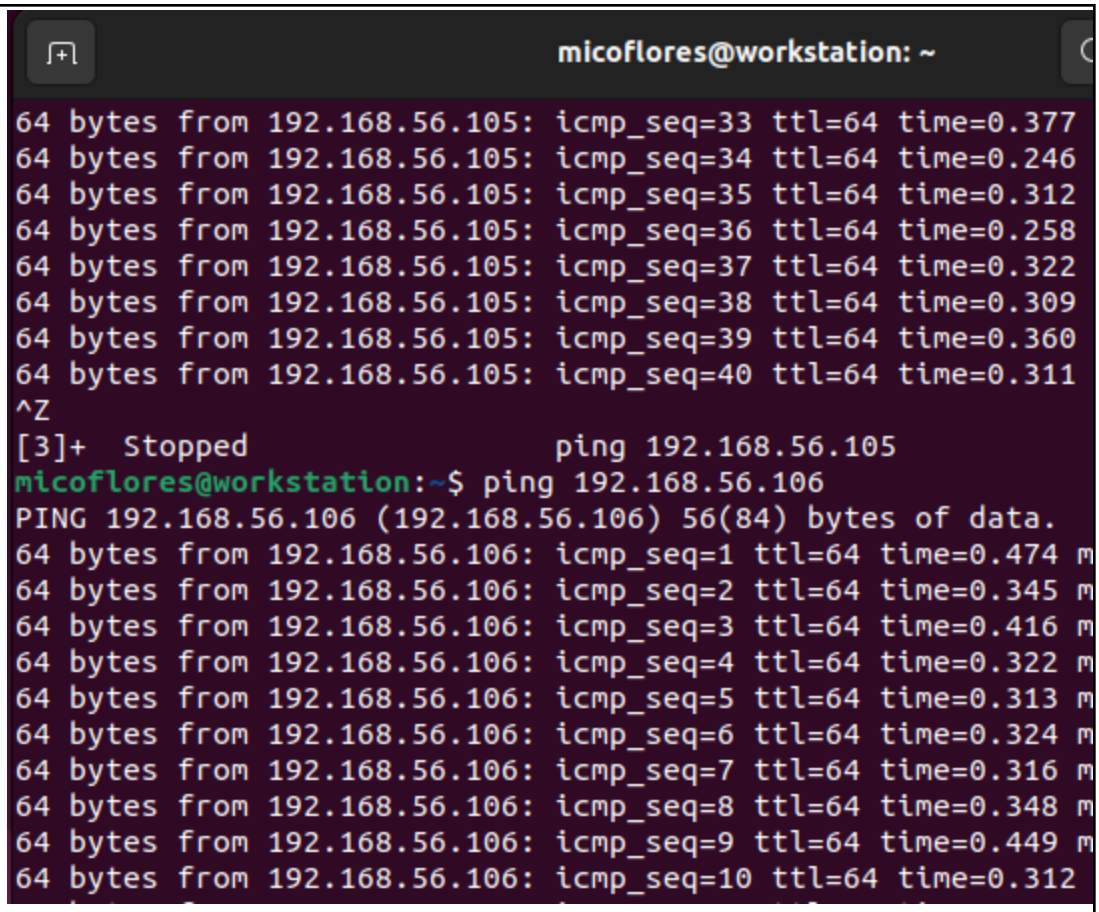
   `192.168.56.106`

   1.3 Server 3 IP address: 192.168.56.104

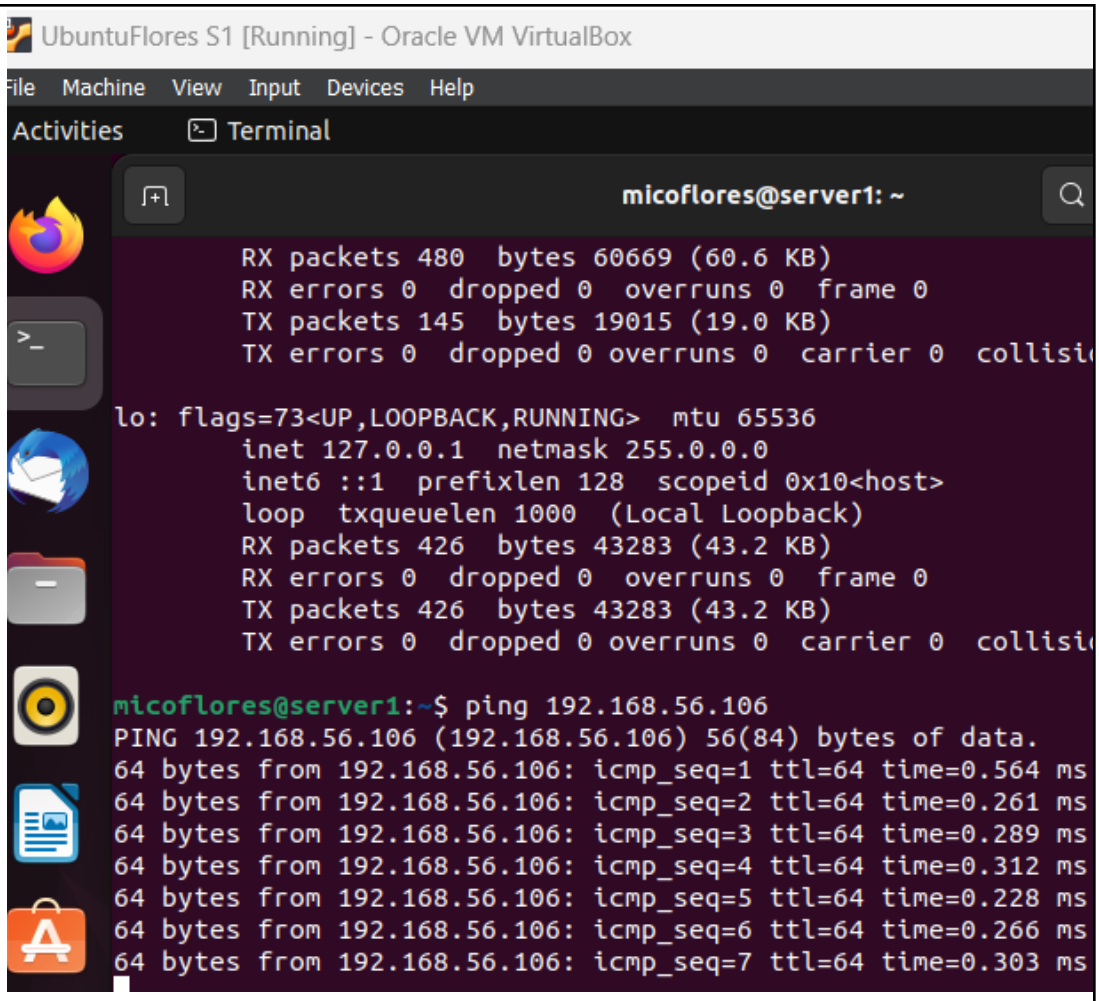
   `192.168.56.104`
2. Make sure that they can ping each other.
   2.1 Connectivity test for Local Machine 1 to Server 1: ✅ Successful ☐ Not Successful



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



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



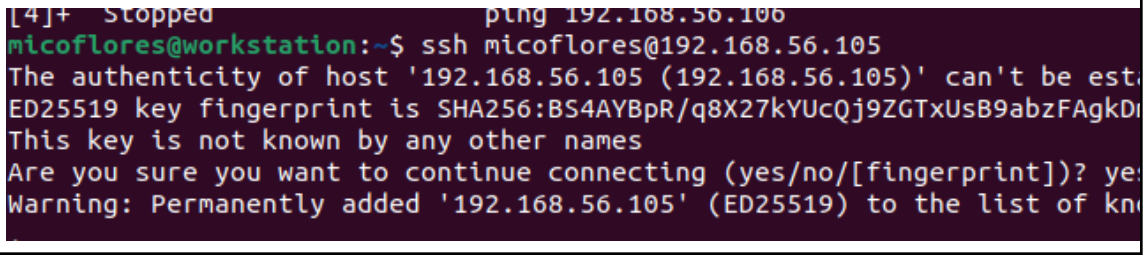
Task 4: Verify SSH connectivity on Server 1, Server 2, and Local Machine.

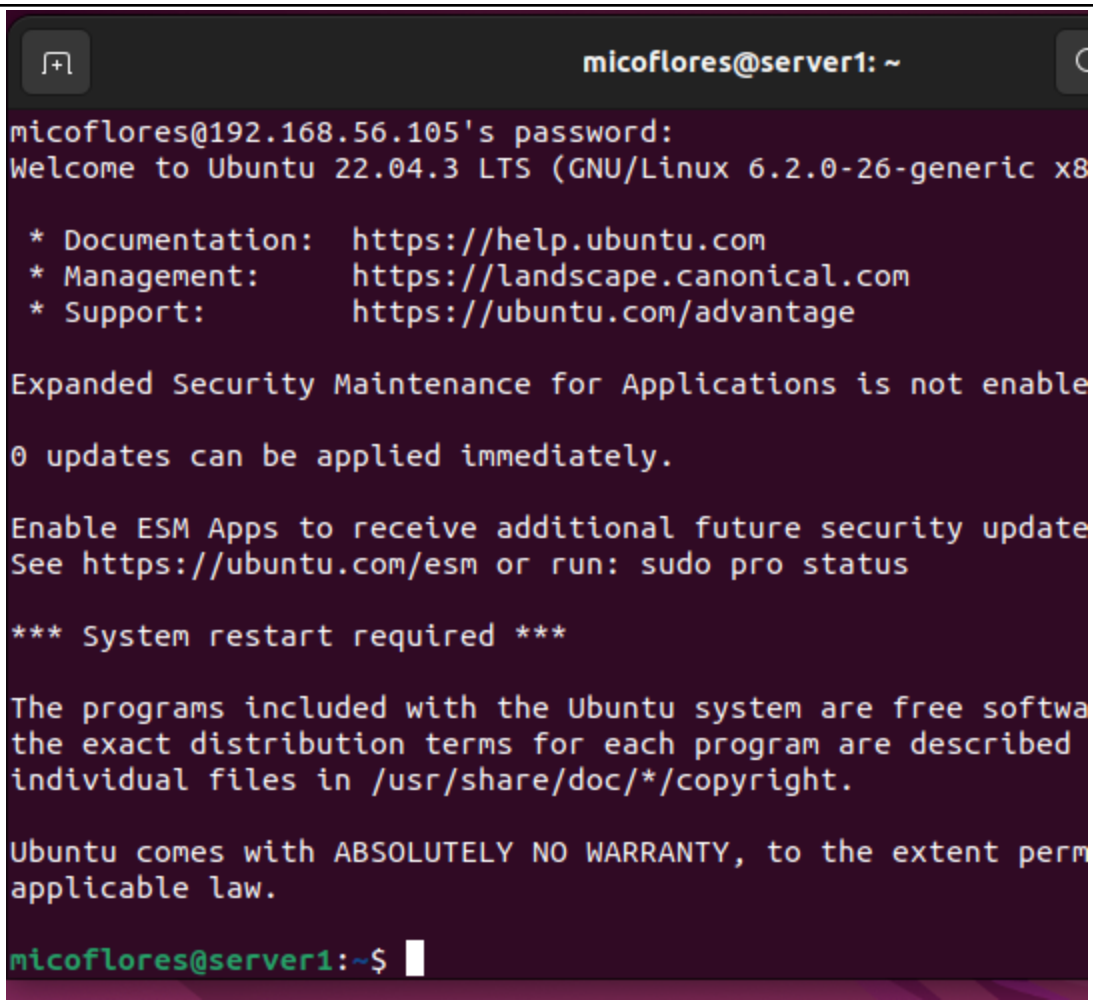
1. On the Local Machine, issue the following commands:

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

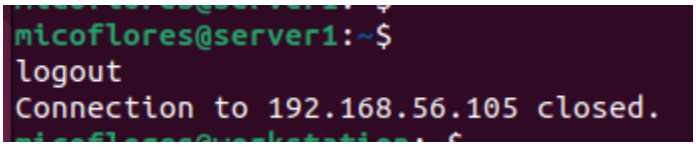
1.2 Enter the password for server 1 when prompted

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

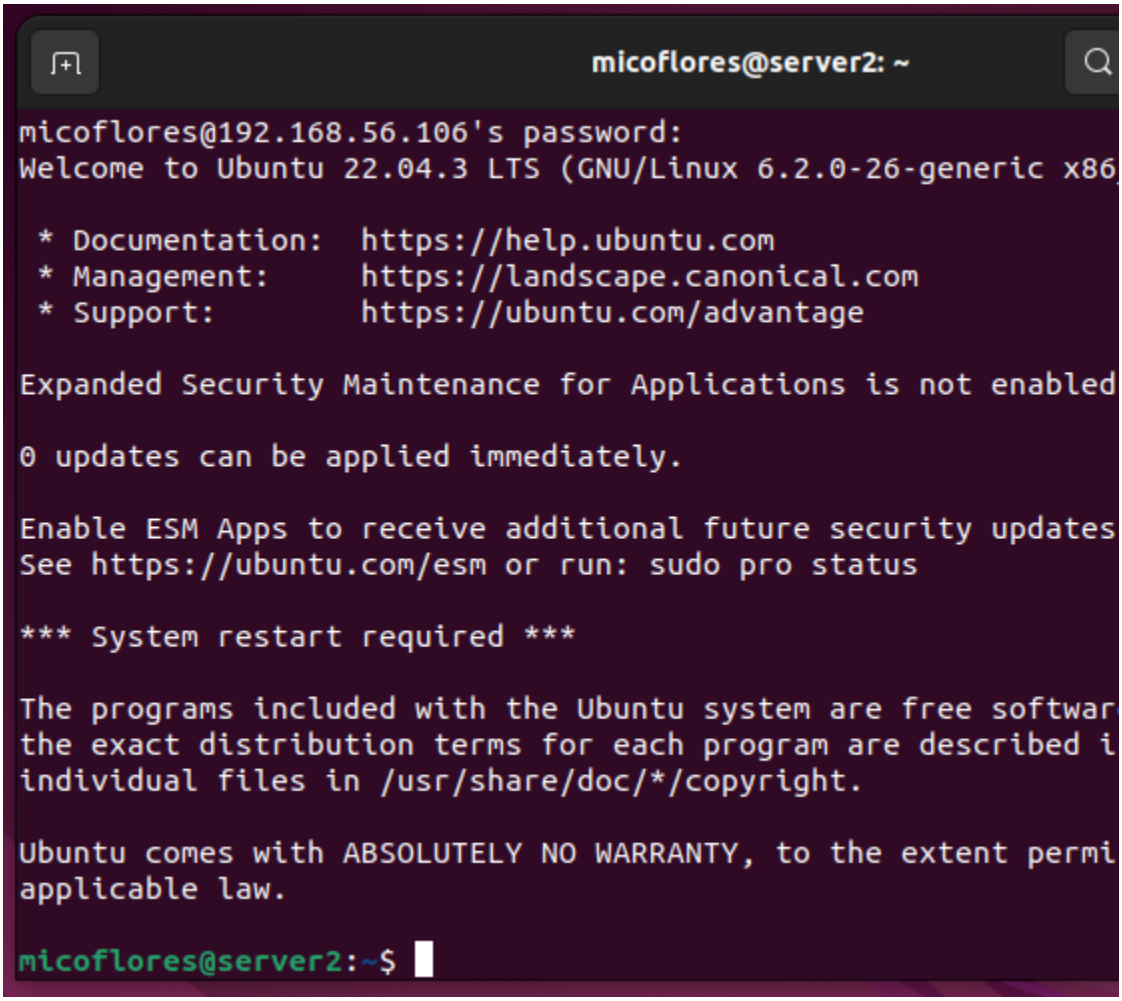




2. Logout of Server 1 by issuing the command *control + D.*



3. Do the same for Server 2.

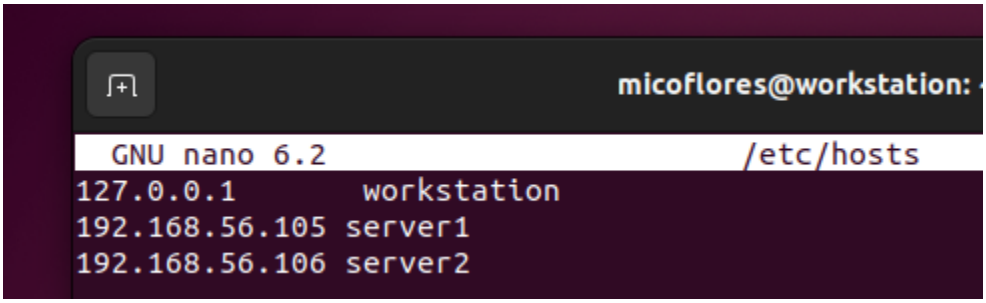


4. Edit the hosts of the Local Machine by issuing the command *sudo nano /etc/hosts.* Below all texts type the following:

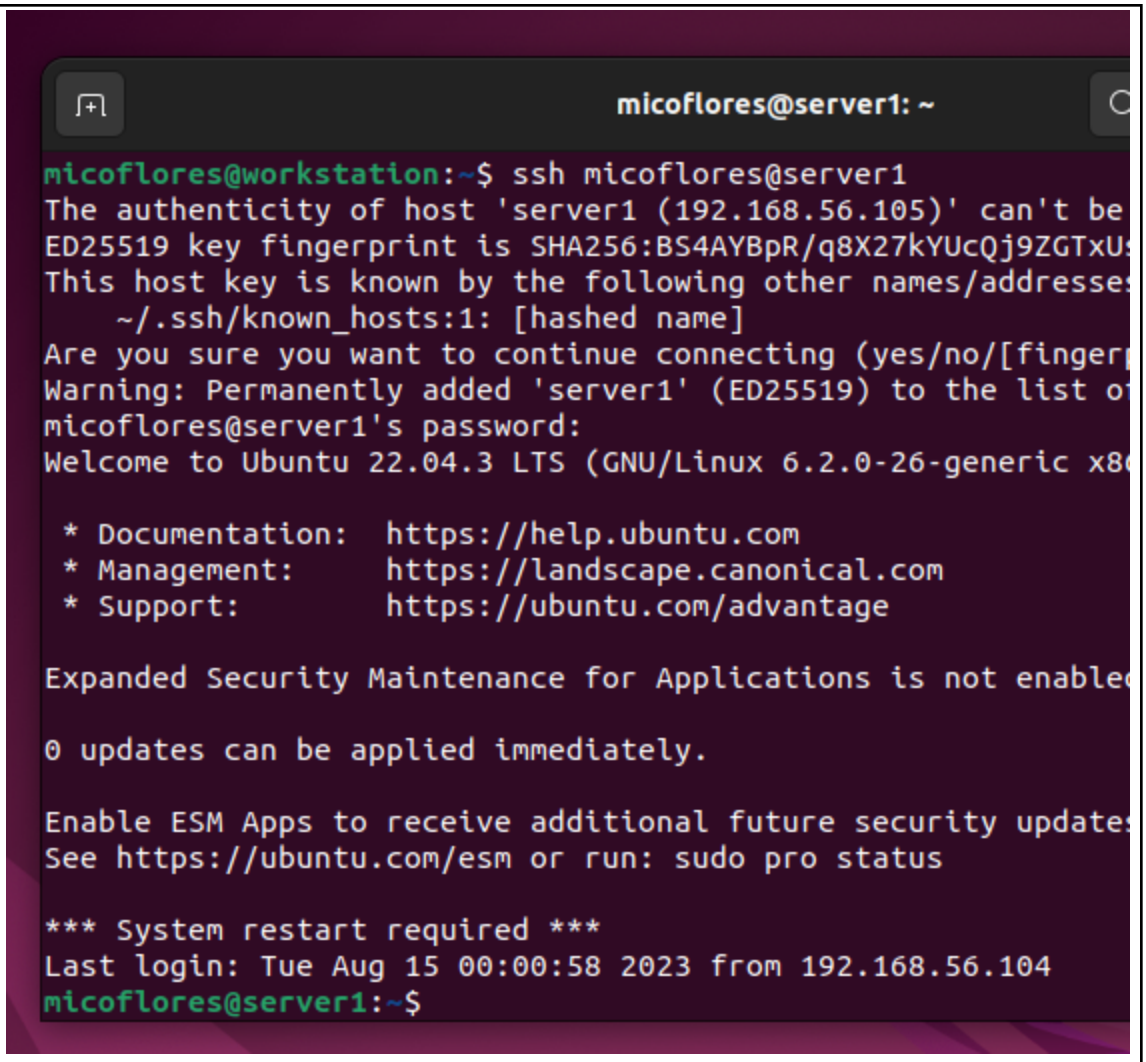
4.1 IP_address server 1 (provide the ip address of server 1 followed by the hostname)

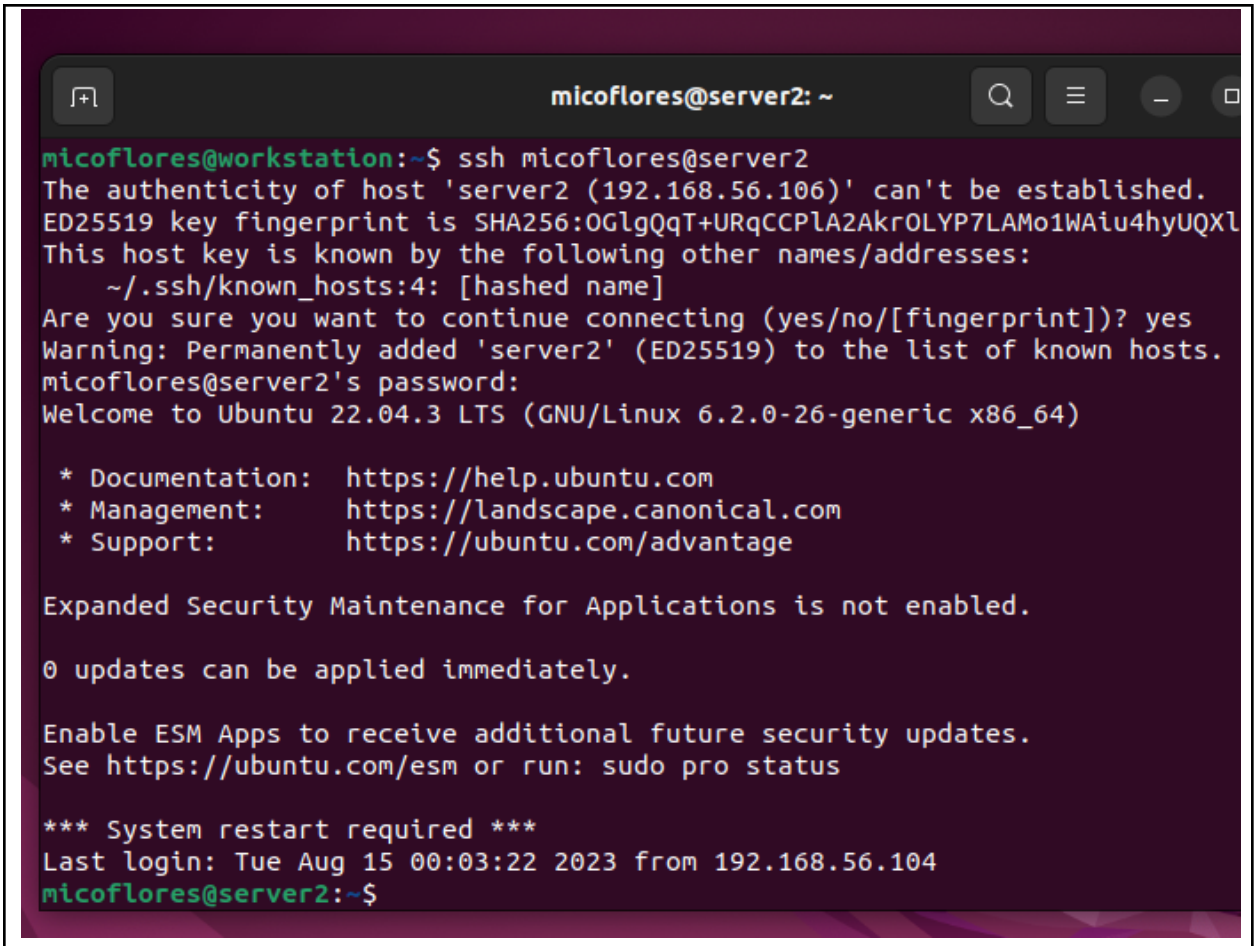
4.2 IP_address server 2 (provide the ip address of server 2 followed by the hostname)

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.


micoflores@server1: ~
micoflores@workstation:~$ ssh micoflores@server1
The authenticity of host 'server1 (192.168.56.105)' can't be
ED25519 key fingerprint is SHA256:BS4AYBpR/q8X27kYUcQj9ZGTxU
This host key is known by the following other names/addresse
~/.ssh/known_hosts:1: [hashed name]
Are you sure you want to continue connecting (yes/no/[finger
Warning: Permanently added 'server1' (ED25519) to the list o
micoflores@server1's password:
Welcome to Ubuntu 22.04.3 LTS (GNU/Linux 6.2.0-26-generic x8
* Documentation: https://help.ubuntu.com
* Management: https://landscape.canonical.com
* Support: https://ubuntu.com/advantage
Expanded Security Maintenance for Applications is not enable
0 updates can be applied immediately.
Enable ESM Apps to receive additional future security update
See https://ubuntu.com/esm or run: sudo pro status
*** System restart required ***
Last login: Tue Aug 15 00:00:58 2023 from 192.168.56.104
micoflores@server1:~$



Reflections:

After doing the tasks in this manual, I have learned how to rename hosts, update each device, install necessary tools, identify IP addresses and access servers using a local machine or one host. In addition to this, I have learned how to operate and run ssh on each server and have access by just putting their hostname instead of IP addresses which makes it more convenient for the admin. This knowledge will be beneficial for us because this gives us skills to be a proper computer engineering student especially for our elective.

Answer the following:

1. How are we able to use the hostname instead of IP address in SSH commands?
   - I am able to use the hostname because I input the IP address followed by the name of server 1 in the /etc/hosts. In doing so, instead of typing the ip address, I can just type in “my_user@server1” and it will connect to the server after entering the password. The same goes to server 2.

2. How secured is SSH?

   - It provides strong encryption which makes it safe to protect the data transmission between the host and the server, and it will be difficult for any attackers to interrupt. It has a logging and monitoring if there are any suspicious unauthorized attempts in trying to access the servers.