Name: MANGUNDAYAO, PAULL ANDREI A.	**Date Performed:** 08/14/23
Course/Section: CPE31S4	**Date Submitted:** 08/15/23
Instructor: Dr. Jonathan Taylar	**Semester and SY:** 1st Semester – 2023-2024

Activity 1: Configure Network using Virtual Machines

1. Objectives:

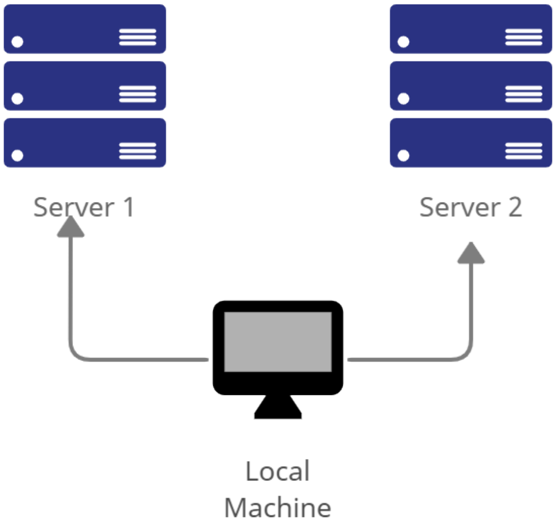
1.1. Create and configure Virtual Machines in Microsoft Azure or VirtualBox

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

2. Discussion:

Network Topology:

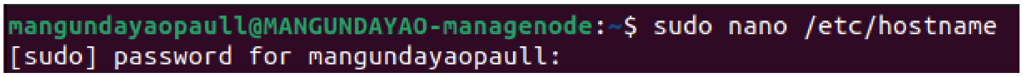
Assume that you have created the following network topology in Virtual Machines, *provide screenshots for each task*. (Note: *it is assumed that you have the prior knowledge of cloning and creating snapshots in a virtual machine*).



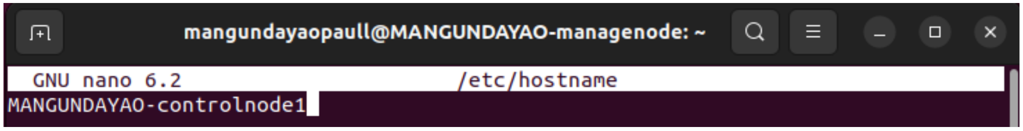
Task 1: Do the following on Server 1, Server 2, and Local Machine. In editing the file using nano command, press control + O to write out (save the file). Press enter when asked for the name of the file. Press control + X to end.

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

   1.1 Use server1 for Server 1

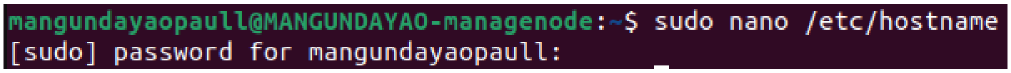


```
mangundayaopaull@MANGUNDAYAO-managenode:~$ sudo nano /etc/hostname
[sudo] password for mangundayaopaull:
```

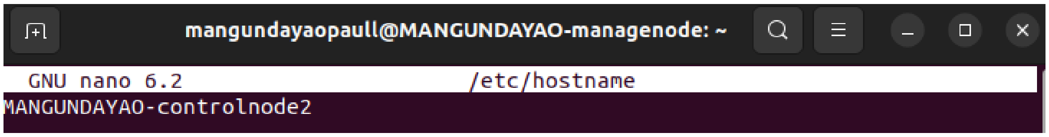


```
  GNU nano 6.2                      /etc/hostname
MANGUNDAYAO-controlnode1
```

1.2 Use server2 for Server 2

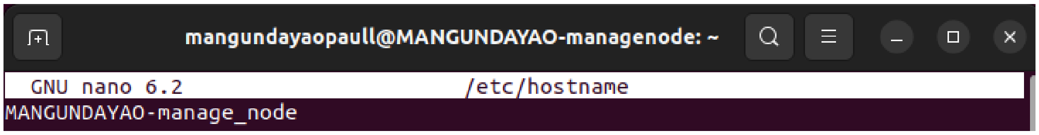


```
mangundayaopaull@MANGUNDAYAO-managenode:~$ sudo nano /etc/hostname
[sudo] password for mangundayaopaull:
```



```
  GNU nano 6.2                      /etc/hostname
MANGUNDAYAO-controlnode2
```

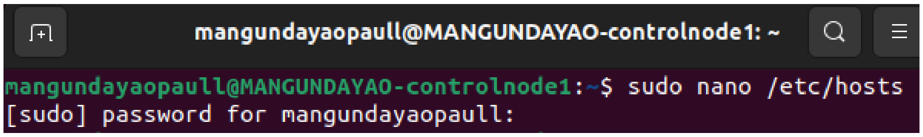
1.3 Use workstation for the Local Machine



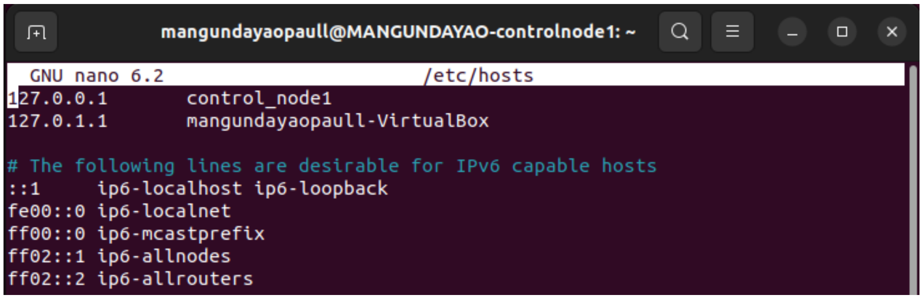
```
  GNU nano 6.2                      /etc/hostname
MANGUNDAYAO-manage_node
```

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

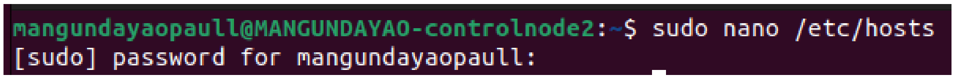


```
mangundayaopaull@MANGUNDAYAO-controlnode1:~$ sudo nano /etc/hosts
[sudo] password for mangundayaopaull:
```

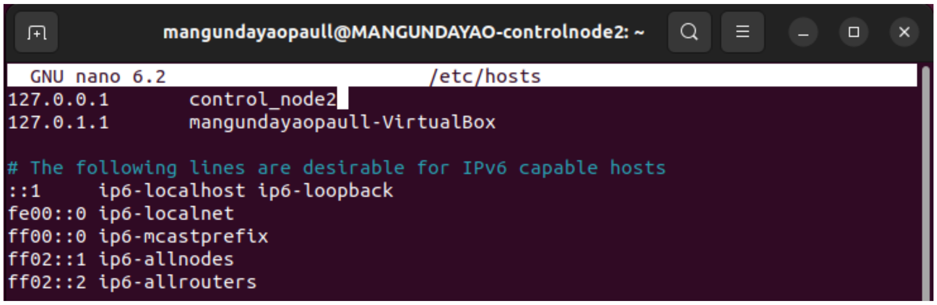


```
  GNU nano 6.2                      /etc/hosts
127.0.0.1       control_node1
127.0.1.1       mangundayaopaull-VirtualBox

# The following lines are desirable for IPv6 capable hosts
::1     ip6-localhost ip6-loopback
fe00::0 ip6-localnet
ff00::0 ip6-mcastprefix
ff02::1 ip6-allnodes
ff02::2 ip6-allrouters
```

2.2 Type 127.0.0.1 server 2 for Server 2

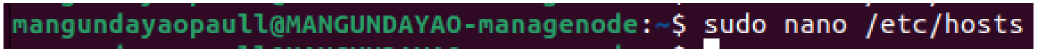


```
mangundayaopaull@MANGUNDAYAO-controlnode2:~$ sudo nano /etc/hosts
[sudo] password for mangundayaopaull:
```

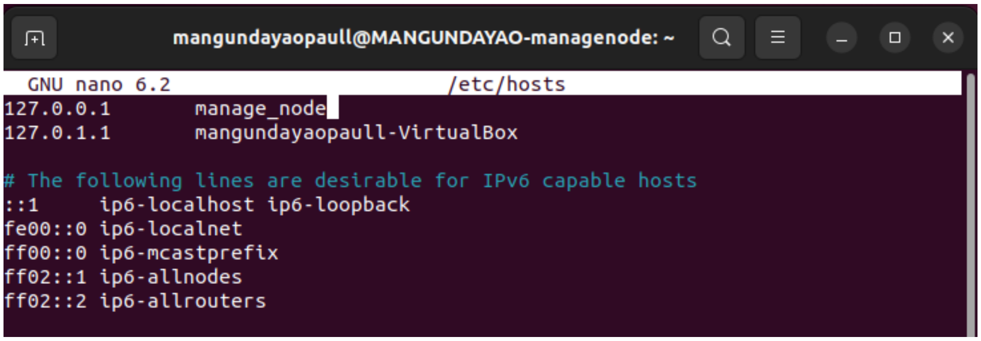


```
  GNU nano 6.2                      /etc/hosts
127.0.0.1       control_node2
127.0.1.1       mangundayaopaull-VirtualBox

# The following lines are desirable for IPv6 capable hosts
::1     ip6-localhost ip6-loopback
fe00::0 ip6-localnet
ff00::0 ip6-mcastprefix
ff02::1 ip6-allnodes
ff02::2 ip6-allrouters
```

2.3 Type 127.0.0.1 workstation for the Local Machine



```
mangundayaopaull@MANGUNDAYAO-managenode:~$ sudo nano /etc/hosts
```

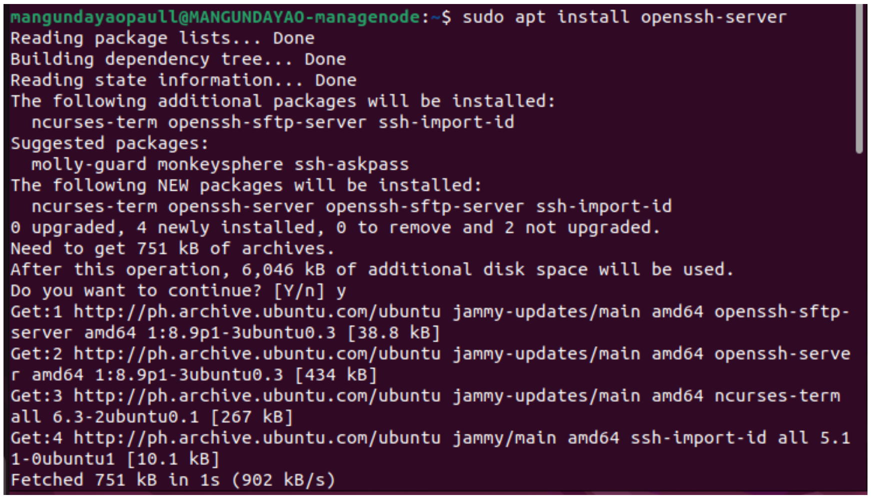


```
  GNU nano 6.2                            /etc/hosts
127.0.0.1       manage_node
127.0.1.1       mangundayaopaull-VirtualBox

# The following lines are desirable for IPv6 capable hosts
::1     ip6-localhost ip6-loopback
fe00::0 ip6-localnet
ff00::0 ip6-mcastprefix
ff02::1 ip6-allnodes
ff02::2 ip6-allrouters
```

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

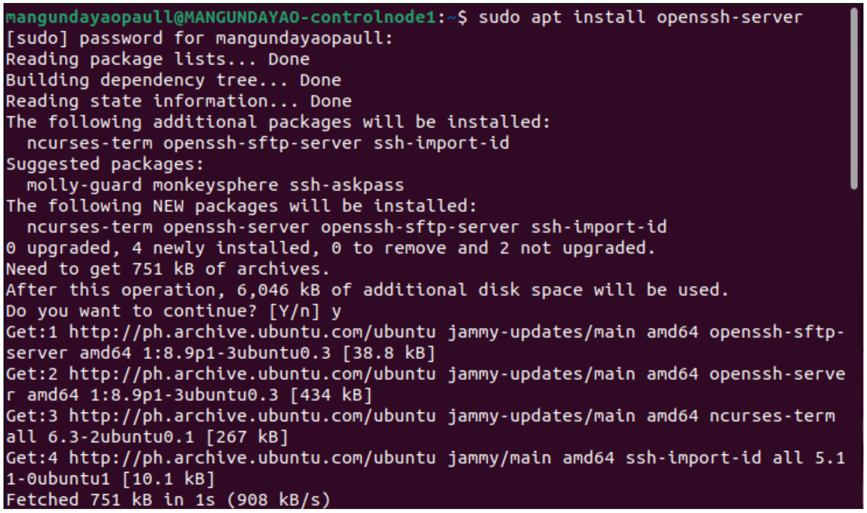
1. Upgrade the packages by issuing the command *sudo apt update* and *sudo apt upgrade* respectively.
2. Install the SSH server using the command *sudo apt install openssh-server*.

Manage Node:



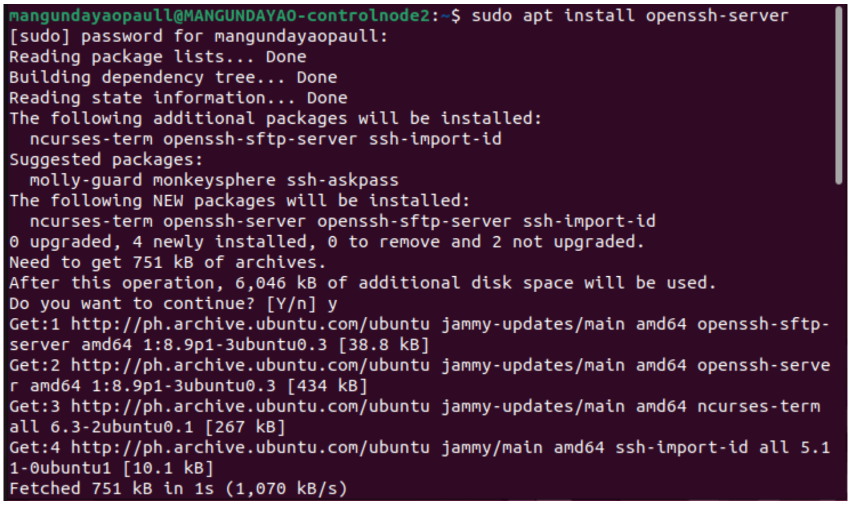
```
mangundayaopaull@MANGUNDAYAO-managenode:~$ sudo apt install openssh-server
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
The following additional packages will be installed:
  ncurses-term openssh-sftp-server ssh-import-id
Suggested packages:
  molly-guard 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 2 not upgraded.
Need to get 751 kB of archives.
After this operation, 6,046 kB of additional disk space will be used.
Do you want to continue? [Y/n] y
Get:1 http://ph.archive.ubuntu.com/ubuntu jammy-updates/main amd64 openssh-sftp-
server amd64 1:8.9p1-3ubuntu0.3 [38.8 kB]
Get:2 http://ph.archive.ubuntu.com/ubuntu jammy-updates/main amd64 openssh-serve
r amd64 1:8.9p1-3ubuntu0.3 [434 kB]
Get:3 http://ph.archive.ubuntu.com/ubuntu jammy-updates/main amd64 ncurses-term
all 6.3-2ubuntu0.1 [267 kB]
Get:4 http://ph.archive.ubuntu.com/ubuntu jammy/main amd64 ssh-import-id all 5.1
1-0ubuntu1 [10.1 kB]
Fetched 751 kB in 1s (902 kB/s)
```

Control Node 1:



```
mangundayaopaull@MANGUNDAYAO-controlnode1:~$ sudo apt install openssh-server
[sudo] password for mangundayaopaull:
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
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 2 not upgraded.
Need to get 751 kB of archives.
After this operation, 6,046 kB of additional disk space will be used.
Do you want to continue? [Y/n] y
Get:1 http://ph.archive.ubuntu.com/ubuntu jammy-updates/main amd64 openssh-sftp-
server amd64 1:8.9p1-3ubuntu0.3 [38.8 kB]
Get:2 http://ph.archive.ubuntu.com/ubuntu jammy-updates/main amd64 openssh-serve
r amd64 1:8.9p1-3ubuntu0.3 [434 kB]
Get:3 http://ph.archive.ubuntu.com/ubuntu jammy-updates/main amd64 ncurses-term
all 6.3-2ubuntu0.1 [267 kB]
Get:4 http://ph.archive.ubuntu.com/ubuntu jammy/main amd64 ssh-import-id all 5.1
1-0ubuntu1 [10.1 kB]
Fetched 751 kB in 1s (908 kB/s)
```

Control Node 2:



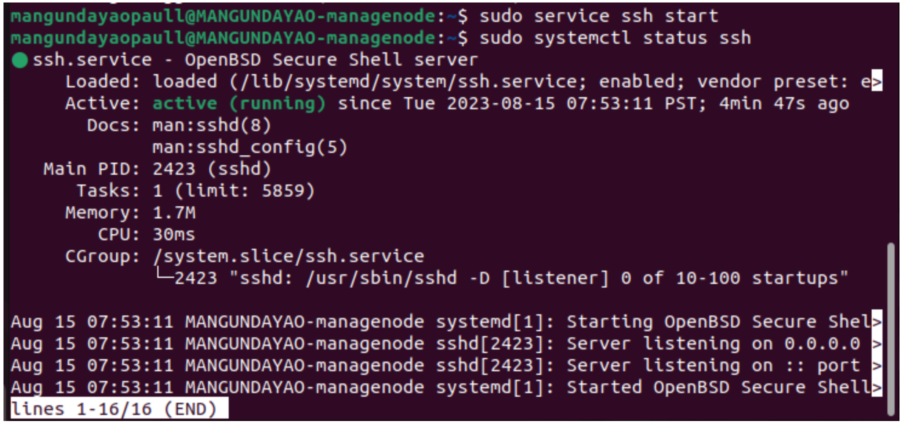
```
mangundayaopaull@MANGUNDAYAO-controlnode2:~$ sudo apt install openssh-server
[sudo] password for mangundayaopaull:
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
The following additional packages will be installed:
  ncurses-term openssh-sftp-server 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 2 not upgraded.
Need to get 751 kB of archives.
After this operation, 6,046 kB of additional disk space will be used.
Do you want to continue? [Y/n] y
Get:1 http://ph.archive.ubuntu.com/ubuntu jammy-updates/main amd64 openssh-sftp-
server amd64 1:8.9p1-3ubuntu0.3 [38.8 kB]
Get:2 http://ph.archive.ubuntu.com/ubuntu jammy-updates/main amd64 openssh-serve
r amd64 1:8.9p1-3ubuntu0.3 [434 kB]
Get:3 http://ph.archive.ubuntu.com/ubuntu jammy-updates/main amd64 ncurses-term
all 6.3-2ubuntu0.1 [267 kB]
Get:4 http://ph.archive.ubuntu.com/ubuntu jammy/main amd64 ssh-import-id all 5.1
1-0ubuntu1 [10.1 kB]
Fetched 751 kB in 1s (1,070 kB/s)
```

3. Verify if the SSH service has started by issuing the following commands:

   3.1 *sudo service ssh start*

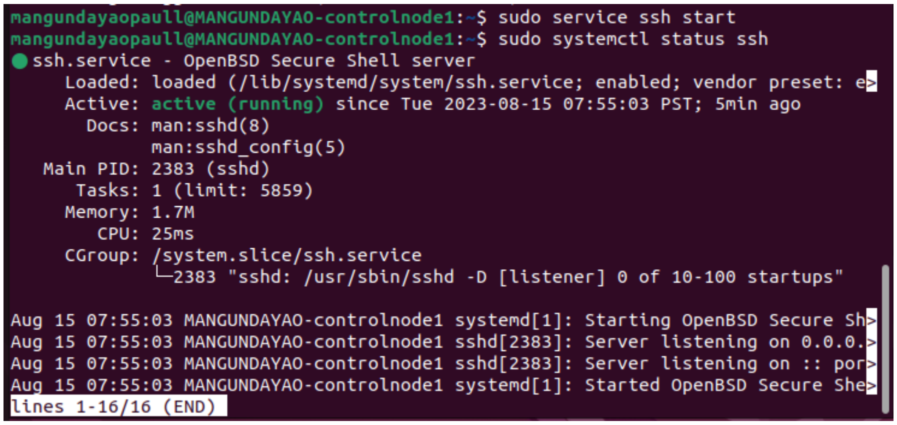
   3.2 *sudo systemctl status ssh*

Manage Node:



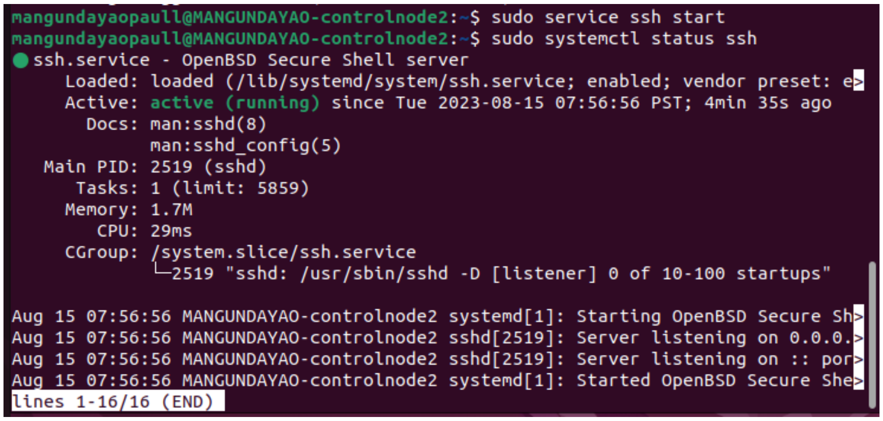
```
mangundayaopaull@MANGUNDAYAO-managenode:~$ sudo service ssh start
mangundayaopaull@MANGUNDAYAO-managenode:~$ sudo systemctl status ssh
● ssh.service - OpenBSD Secure Shell server
     Loaded: loaded (/lib/systemd/system/ssh.service; enabled; vendor preset: e>
     Active: active (running) since Tue 2023-08-15 07:53:11 PST; 4min 47s ago
       Docs: man:sshd(8)
             man:sshd_config(5)
   Main PID: 2423 (sshd)
      Tasks: 1 (limit: 5859)
     Memory: 1.7M
        CPU: 30ms
     CGroup: /system.slice/ssh.service
             └─2423 "sshd: /usr/sbin/sshd -D [listener] 0 of 10-100 startups"

Aug 15 07:53:11 MANGUNDAYAO-managenode systemd[1]: Starting OpenBSD Secure Shel>
Aug 15 07:53:11 MANGUNDAYAO-managenode sshd[2423]: Server listening on 0.0.0.0 >
Aug 15 07:53:11 MANGUNDAYAO-managenode sshd[2423]: Server listening on :: port >
Aug 15 07:53:11 MANGUNDAYAO-managenode systemd[1]: Started OpenBSD Secure Shell>
lines 1-16/16 (END)
```

Control Node 1:



```
mangundayaopaull@MANGUNDAYAO-controlnode1:~$ sudo service ssh start
mangundayaopaull@MANGUNDAYAO-controlnode1:~$ sudo systemctl status ssh
● ssh.service - OpenBSD Secure Shell server
     Loaded: loaded (/lib/systemd/system/ssh.service; enabled; vendor preset: e>
     Active: active (running) since Tue 2023-08-15 07:55:03 PST; 5min ago
       Docs: man:sshd(8)
             man:sshd_config(5)
   Main PID: 2383 (sshd)
      Tasks: 1 (limit: 5859)
     Memory: 1.7M
        CPU: 25ms
     CGroup: /system.slice/ssh.service
             └─2383 "sshd: /usr/sbin/sshd -D [listener] 0 of 10-100 startups"

Aug 15 07:55:03 MANGUNDAYAO-controlnode1 systemd[1]: Starting OpenBSD Secure Sh>
Aug 15 07:55:03 MANGUNDAYAO-controlnode1 sshd[2383]: Server listening on 0.0.0.>
Aug 15 07:55:03 MANGUNDAYAO-controlnode1 sshd[2383]: Server listening on :: por>
Aug 15 07:55:03 MANGUNDAYAO-controlnode1 systemd[1]: Started OpenBSD Secure She>
lines 1-16/16 (END)
```

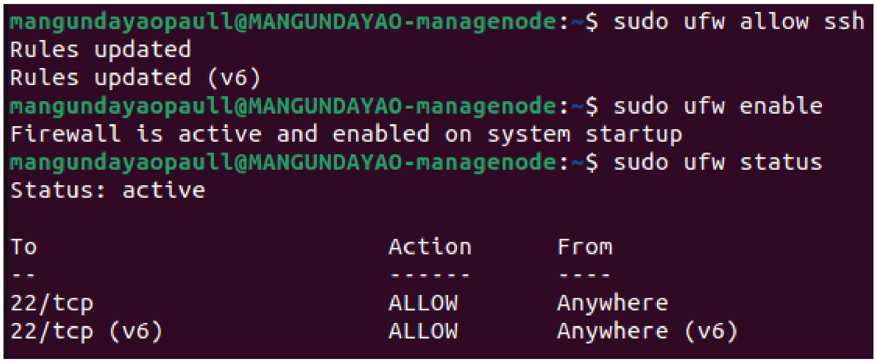
Control Node 2:



```
mangundayaopaull@MANGUNDAYAO-controlnode2:~$ sudo service ssh start
mangundayaopaull@MANGUNDAYAO-controlnode2:~$ sudo systemctl status ssh
● ssh.service - OpenBSD Secure Shell server
     Loaded: loaded (/lib/systemd/system/ssh.service; enabled; vendor preset: e>
     Active: active (running) since Tue 2023-08-15 07:56:56 PST; 4min 35s ago
       Docs: man:sshd(8)
             man:sshd_config(5)
   Main PID: 2519 (sshd)
      Tasks: 1 (limit: 5859)
     Memory: 1.7M
        CPU: 29ms
     CGroup: /system.slice/ssh.service
             └─2519 "sshd: /usr/sbin/sshd -D [listener] 0 of 10-100 startups"

Aug 15 07:56:56 MANGUNDAYAO-controlnode2 systemd[1]: Starting OpenBSD Secure Sh>
Aug 15 07:56:56 MANGUNDAYAO-controlnode2 sshd[2519]: Server listening on 0.0.0.>
Aug 15 07:56:56 MANGUNDAYAO-controlnode2 sshd[2519]: Server listening on :: por>
Aug 15 07:56:56 MANGUNDAYAO-controlnode2 systemd[1]: Started OpenBSD Secure She>
lines 1-16/16 (END)
```

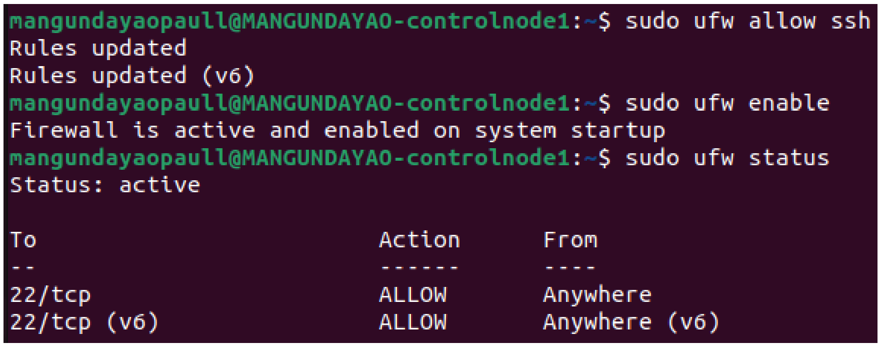
4. Configure the firewall to all port 22 by issuing the following commands:
   4.1 *sudo ufw allow ssh*
   4.2 *sudo ufw enable*
   4.3 *sudo ufw status*

Manage Node:



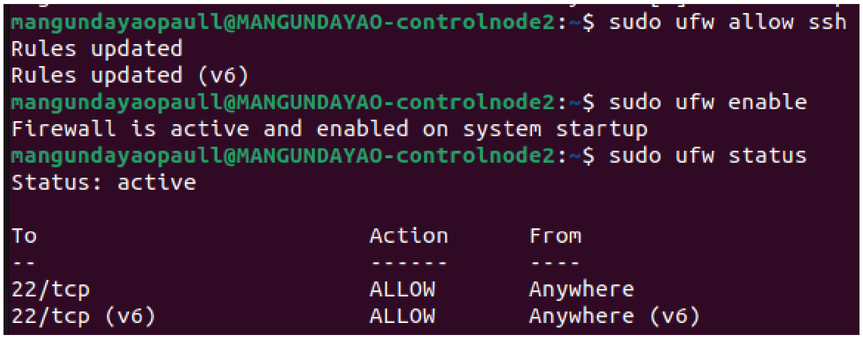
```
mangundayaopaull@MANGUNDAYAO-managenode:~$ sudo ufw allow ssh
Rules updated
Rules updated (v6)
mangundayaopaull@MANGUNDAYAO-managenode:~$ sudo ufw enable
Firewall is active and enabled on system startup
mangundayaopaull@MANGUNDAYAO-managenode:~$ sudo ufw status
Status: active

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

Control Node 1:



```
mangundayaopaull@MANGUNDAYAO-controlnode1:~$ sudo ufw allow ssh
Rules updated
Rules updated (v6)
mangundayaopaull@MANGUNDAYAO-controlnode1:~$ sudo ufw enable
Firewall is active and enabled on system startup
mangundayaopaull@MANGUNDAYAO-controlnode1:~$ sudo ufw status
Status: active

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

Control Node 2:

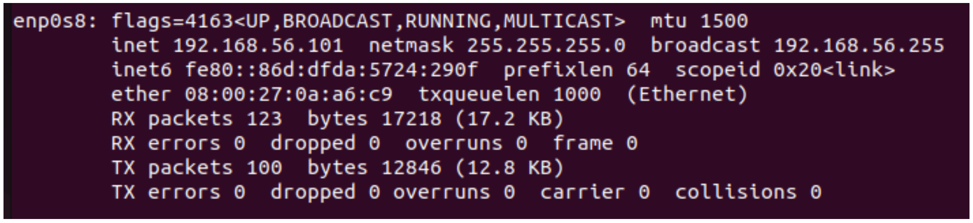


```
mangundayaopaull@MANGUNDAYAO-controlnode2:~$ sudo ufw allow ssh
Rules updated
Rules updated (v6)
mangundayaopaull@MANGUNDAYAO-controlnode2:~$ sudo ufw enable
Firewall is active and enabled on system startup
mangundayaopaull@MANGUNDAYAO-controlnode2:~$ sudo ufw status
Status: active

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:

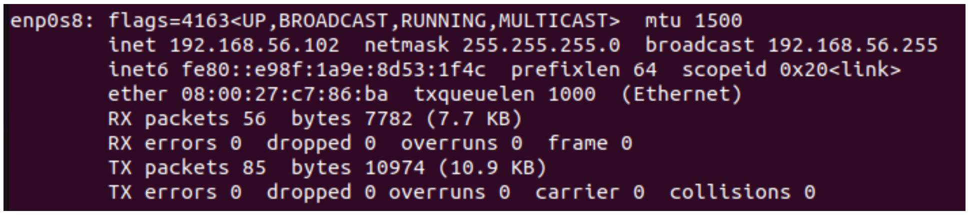
1. Record the ip address of Server 1, Server 2, and Local Machine. Issue the command *ifconfig* and check network settings. Note that the ip addresses of all the machines are in this network 192.168.56.XX.

1.1 **Manage Node** IP address: 192.168.56.**101**



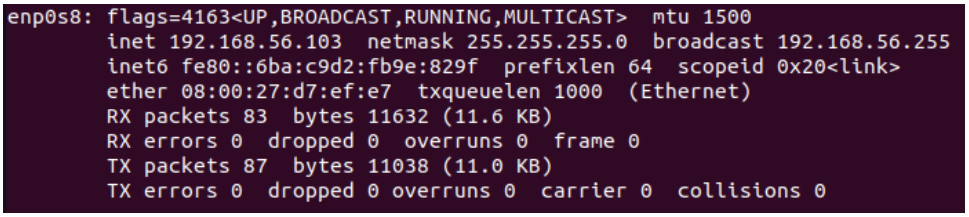
```
enp0s8: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1500
        inet 192.168.56.101  netmask 255.255.255.0  broadcast 192.168.56.255
        inet6 fe80::86d:dfda:5724:290f  prefixlen 64  scopeid 0x20<link>
        ether 08:00:27:0a:a6:c9  txqueuelen 1000  (Ethernet)
        RX packets 123  bytes 17218 (17.2 KB)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 100  bytes 12846 (12.8 KB)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0
```

1.2 **Control Node 1** IP address: 192.168.56.**102**



```
enp0s8: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1500
        inet 192.168.56.102  netmask 255.255.255.0  broadcast 192.168.56.255
        inet6 fe80::e98f:1a9e:8d53:1f4c  prefixlen 64  scopeid 0x20<link>
        ether 08:00:27:c7:86:ba  txqueuelen 1000  (Ethernet)
        RX packets 56  bytes 7782 (7.7 KB)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 85  bytes 10974 (10.9 KB)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0
```

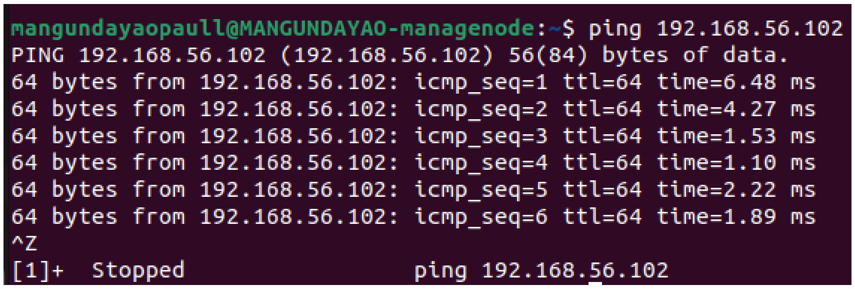
1.3 **Control Node 2** IP address: 192.168.56.**103**



```
enp0s8: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1500
        inet 192.168.56.103  netmask 255.255.255.0  broadcast 192.168.56.255
        inet6 fe80::6ba:c9d2:fb9e:829f  prefixlen 64  scopeid 0x20<link>
        ether 08:00:27:d7:ef:e7  txqueuelen 1000  (Ethernet)
        RX packets 83  bytes 11632 (11.6 KB)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 87  bytes 11038 (11.0 KB)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0
```

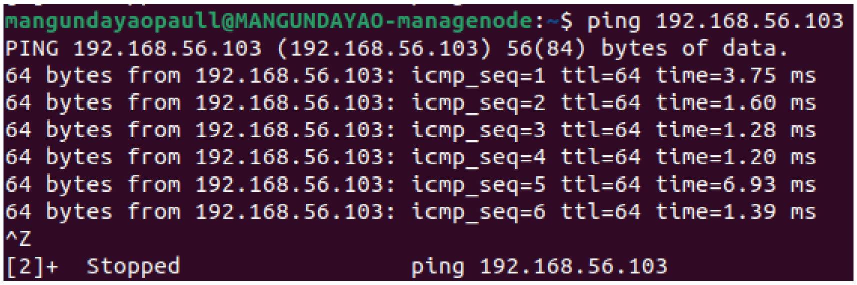
2. Make sure that they can ping each other.

2.1 Connectivity test for Local Machine 1 to Server 1: ☒ Successful ☐ Not Successful



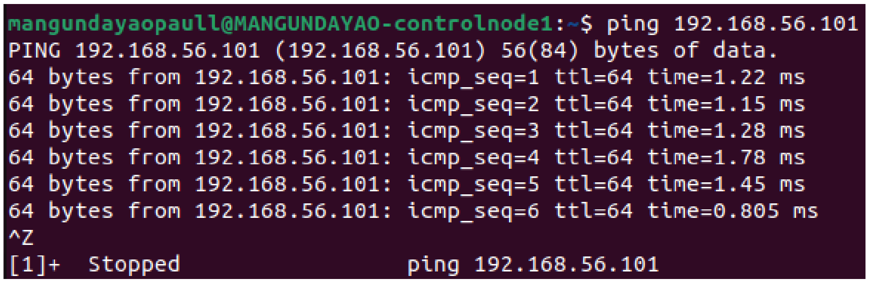
```
mangundayaopaull@MANGUNDAYAO-managenode:~$ ping 192.168.56.102
PING 192.168.56.102 (192.168.56.102) 56(84) bytes of data.
64 bytes from 192.168.56.102: icmp_seq=1 ttl=64 time=6.48 ms
64 bytes from 192.168.56.102: icmp_seq=2 ttl=64 time=4.27 ms
64 bytes from 192.168.56.102: icmp_seq=3 ttl=64 time=1.53 ms
64 bytes from 192.168.56.102: icmp_seq=4 ttl=64 time=1.10 ms
64 bytes from 192.168.56.102: icmp_seq=5 ttl=64 time=2.22 ms
64 bytes from 192.168.56.102: icmp_seq=6 ttl=64 time=1.89 ms
^Z
[1]+  Stopped                 ping 192.168.56.102
```

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



```
mangundayaopaull@MANGUNDAYAO-managenode:~$ ping 192.168.56.103
PING 192.168.56.103 (192.168.56.103) 56(84) bytes of data.
64 bytes from 192.168.56.103: icmp_seq=1 ttl=64 time=3.75 ms
64 bytes from 192.168.56.103: icmp_seq=2 ttl=64 time=1.60 ms
64 bytes from 192.168.56.103: icmp_seq=3 ttl=64 time=1.28 ms
64 bytes from 192.168.56.103: icmp_seq=4 ttl=64 time=1.20 ms
64 bytes from 192.168.56.103: icmp_seq=5 ttl=64 time=6.93 ms
64 bytes from 192.168.56.103: icmp_seq=6 ttl=64 time=1.39 ms
^Z
[2]+  Stopped                 ping 192.168.56.103
```

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



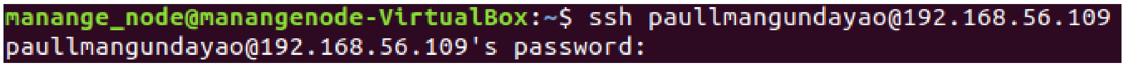
```
mangundayaopaull@MANGUNDAYAO-controlnode1:~$ ping 192.168.56.101
PING 192.168.56.101 (192.168.56.101) 56(84) bytes of data.
64 bytes from 192.168.56.101: icmp_seq=1 ttl=64 time=1.22 ms
64 bytes from 192.168.56.101: icmp_seq=2 ttl=64 time=1.15 ms
64 bytes from 192.168.56.101: icmp_seq=3 ttl=64 time=1.28 ms
64 bytes from 192.168.56.101: icmp_seq=4 ttl=64 time=1.78 ms
64 bytes from 192.168.56.101: icmp_seq=5 ttl=64 time=1.45 ms
64 bytes from 192.168.56.101: icmp_seq=6 ttl=64 time=0.805 ms
^Z
[1]+  Stopped                 ping 192.168.56.101
```

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

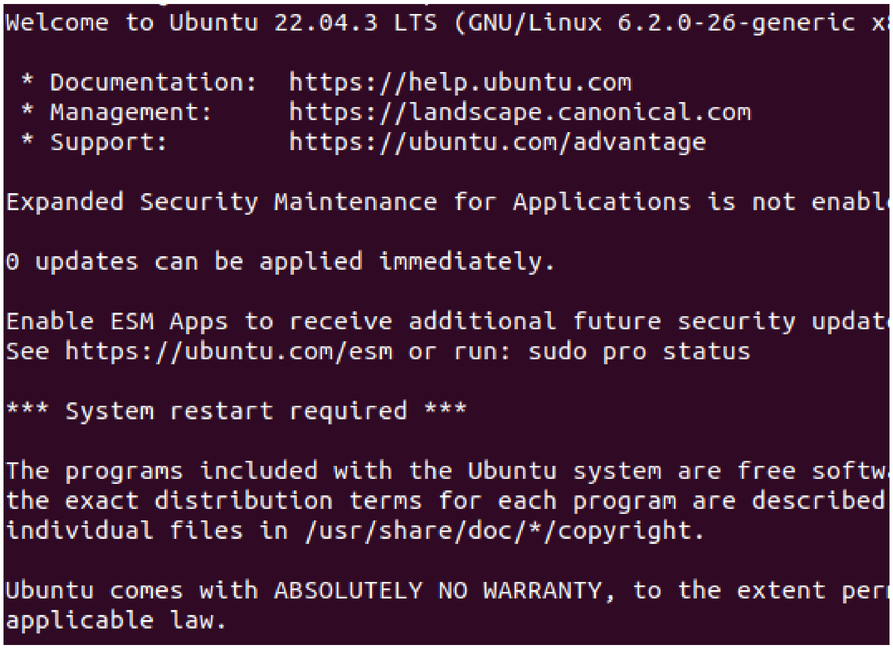
1. On the Local Machine, issue the following commands:

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



```
manange_node@manangenode-VirtualBox:~$ ssh paullmangundayao@192.168.56.109
paullmangundayao@192.168.56.109's password:
```

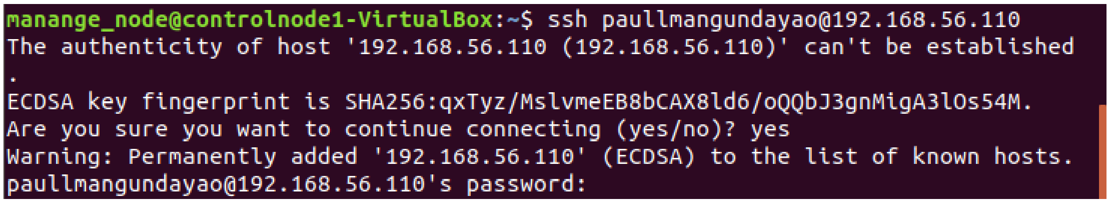


```
Welcome to Ubuntu 22.04.3 LTS (GNU/Linux 6.2.0-26-generic x

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

Expanded Security Maintenance for Applications is not enabl

0 updates can be applied immediately.

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

*** System restart required ***

The programs included with the Ubuntu system are free softw
the exact distribution terms for each program are described
individual files in /usr/share/doc/*/copyright.

Ubuntu comes with ABSOLUTELY NO WARRANTY, to the extent per
applicable law.
```

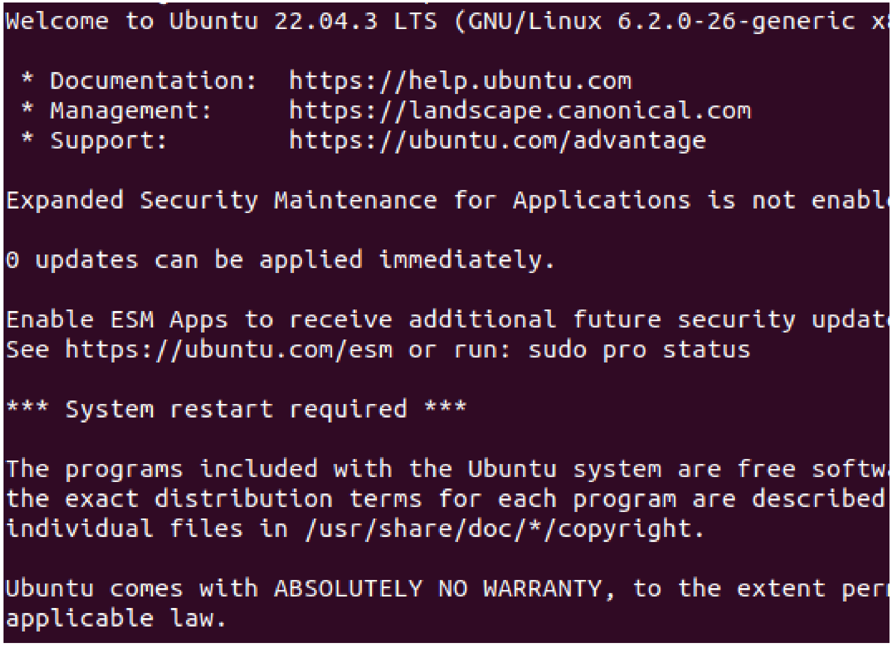
1.3 Verify that you are in server 1. The user should be in this format user@server1. For example, *jvtaylar@server1*

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

3. Do the same for Server 2.



```
manange_node@controlnode1-VirtualBox:~$ ssh paullmangundayao@192.168.56.110
The authenticity of host '192.168.56.110 (192.168.56.110)' can't be established
.
ECDSA key fingerprint is SHA256:qxTyz/MslvmeEB8bCAX8ld6/oQQbJ3gnMigA3lOs54M.
Are you sure you want to continue connecting (yes/no)? yes
Warning: Permanently added '192.168.56.110' (ECDSA) to the list of known hosts.
paullmangundayao@192.168.56.110's password:
```



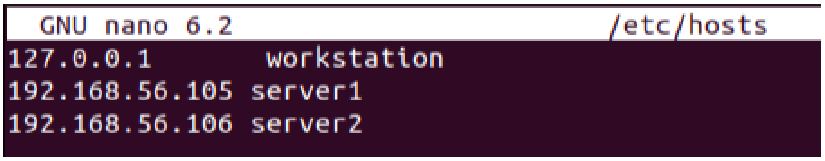
```
Welcome to Ubuntu 22.04.3 LTS (GNU/Linux 6.2.0-26-generic x

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

Expanded Security Maintenance for Applications is not enabl

0 updates can be applied immediately.

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

*** System restart required ***

The programs included with the Ubuntu system are free softw
the exact distribution terms for each program are described
individual files in /usr/share/doc/*/copyright.

Ubuntu comes with ABSOLUTELY NO WARRANTY, to the extent per
applicable law.
```

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)

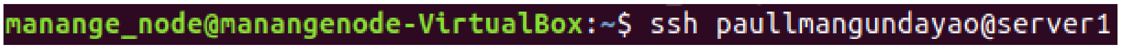
4.3 Save the file and exit.



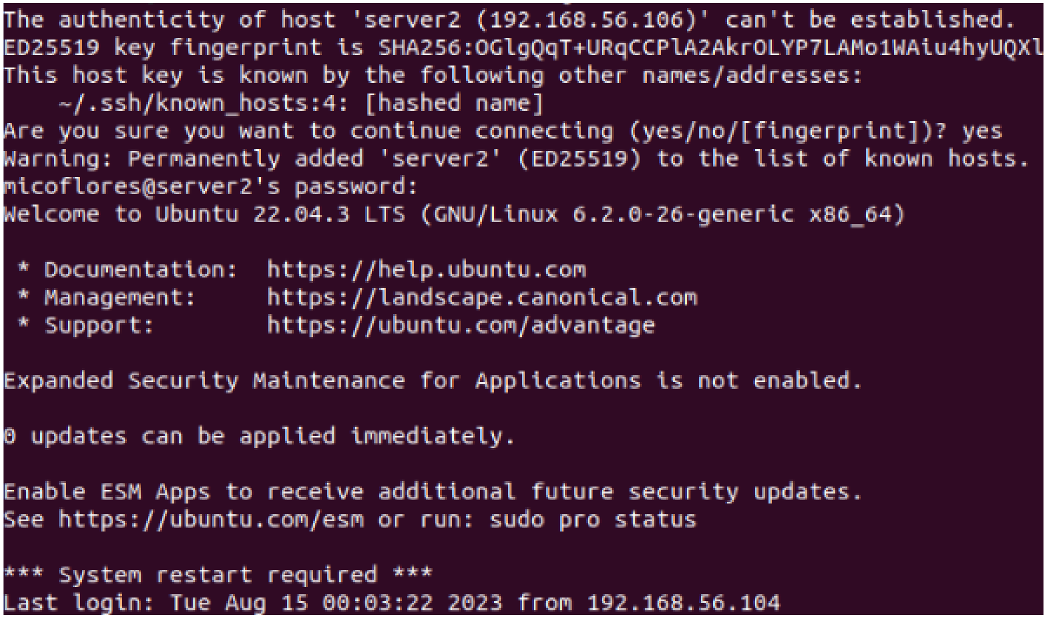
```
  GNU nano 6.2                          /etc/hosts
127.0.0.1       workstation
192.168.56.105 server1
192.168.56.106 server2
```

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.

Server 1:

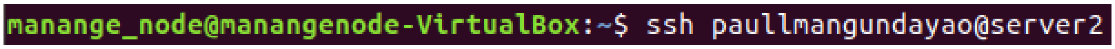


```
manange_node@manangenode-VirtualBox:~$ ssh paullmangundayao@server1
```

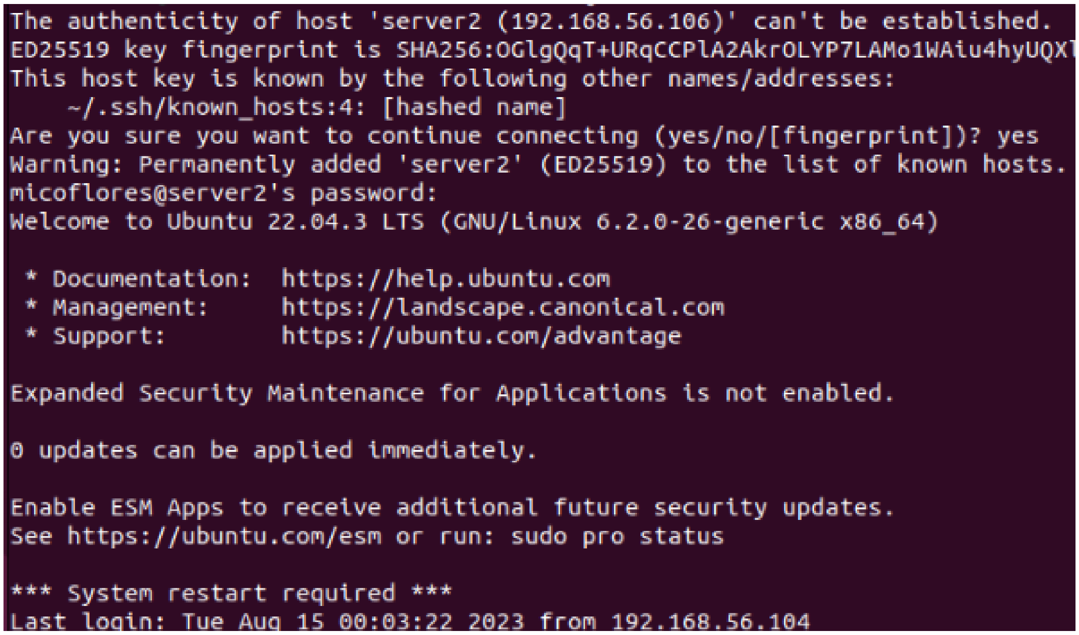


```
The authenticity of host 'server2 (192.168.56.106)' can't be established.
ED25519 key fingerprint is SHA256:OGlgQqT+URqCCPlA2AkrOLYP7LAMo1WAiu4hyUQXl
This host key is known by the following other names/addresses:
    ~/.ssh/known_hosts:4: [hashed name]
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added 'server2' (ED25519) to the list of known hosts.
micoflores@server2's password:
Welcome to Ubuntu 22.04.3 LTS (GNU/Linux 6.2.0-26-generic x86_64)

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

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

*** System restart required ***
Last login: Tue Aug 15 00:03:22 2023 from 192.168.56.104
```

Server 2:



```
manange_node@manangenode-VirtualBox:~$ ssh paullmangundayao@server2
```



```
The authenticity of host 'server2 (192.168.56.106)' can't be established.
ED25519 key fingerprint is SHA256:OGlgQqT+URqCCPlA2AkrOLYP7LAMo1WAiu4hyUQX
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.
micoflores@server2's password:
Welcome to Ubuntu 22.04.3 LTS (GNU/Linux 6.2.0-26-generic x86_64)

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

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

*** System restart required ***
Last login: Tue Aug 15 00:03:22 2023 from 192.168.56.104
```

Reflections:

With this activity, I was able to learn how to properly clone a virtual machine to be used as a virtual server. We also refreshed some of our knowledge we had during the last SysAd course, like renaming the hostname, and changing descriptions of the host's IP addresses, which helped to make it easier to recognize the different machines. This activity also helped us remember some of our knowledge regarding configuring networks.

Answer the following:

1. How are we able to use the hostname instead of IP address in SSH commands?
   I was able to use SSH instead of the IP addresses by first installing and enabling the necessary applications to make use of SSH; by using the commands sudo ufw allow ssh and sudo ufw enable, these commands enable the use of SSH in linux. After that creating a connection with other IP addresses will be possible by using the command “my_user@server1”.

2. How secure is SSH?

   SSH is said to be a secure protocol and the most common way to safely administer servers. Once the connection is established, SSH provides an encrypted way to exchange information safely regardless of the underlying network infrastructure.