Name: Planta, Calvin Earl L.	**Date Performed: Aug 8, 2025**
Course/Section: CPE 212 - CPE31S2	**Date Submitted: Aug 8, 2025**
Instructor: Engr. Robin Valenzuela	**Semester and SY: 1st Semester S.Y 2025-2026**

Activity 1: Configure Network using Virtual Machines

1. **Objectives:**

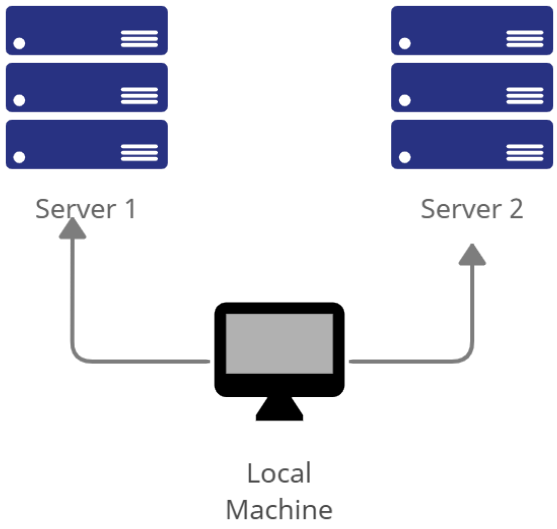
1.1. Create and configure Virtual Machines in Microsoft Azure or VirtualBox

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*).

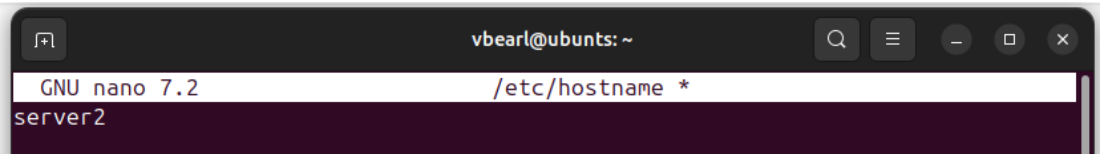


Task 1: Do the following on Server 1, Server 2, and Local Machine. In editing the file using nano command, press control + O to write out (save the file). Press enter when asked for the name of the file. Press control + X to end.

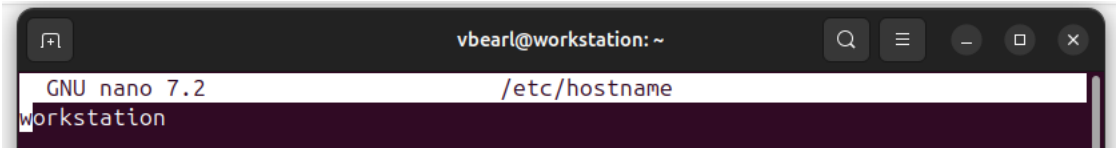
1. Change the hostname using the command *sudo nano /etc/hostname*

1.1 Use server1 for Server 1

1.2 Use server2 for Server 2

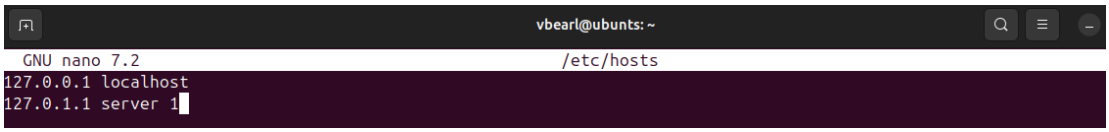


1.3 Use workstation for the Local Machine

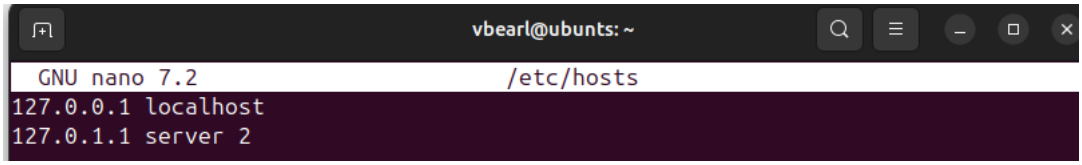


2. Edit the hosts using the command *sudo nano /etc/hosts*. Edit the second line.

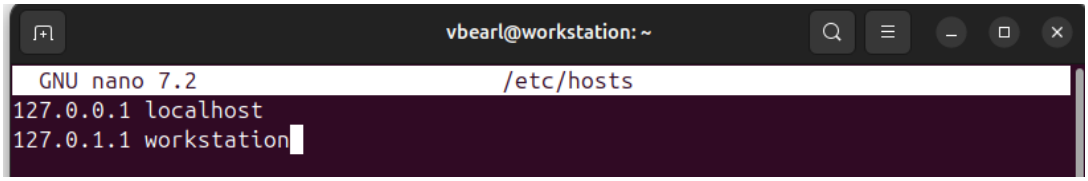
2.1 Type 127.0.0.1 server 1 for Server 1



2.2 Type 127.0.0.1 server 2 for Server 2

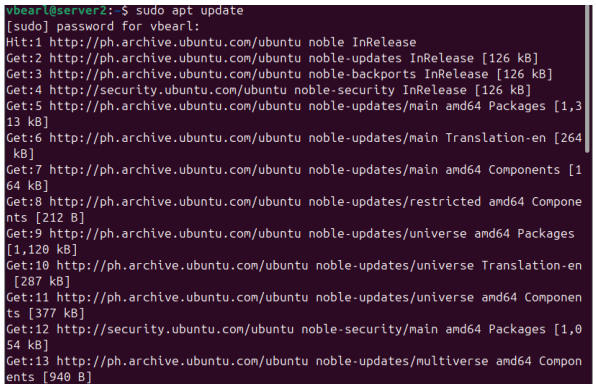


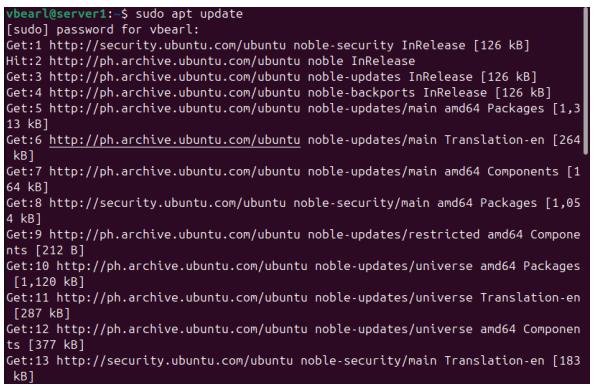
2.3 Type 127.0.0.1 workstation for the Local Machine



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.





```
vbearl@workstation:~$ sudo apt update
[sudo] password for vbearl:
Hit:1 http://ph.archive.ubuntu.com/ubuntu noble InRelease
Get:2 http://ph.archive.ubuntu.com/ubuntu noble-updates InRelease [126 kB]
Get:3 http://ph.archive.ubuntu.com/ubuntu noble-backports InRelease [126 kB]
Get:4 http://security.ubuntu.com/ubuntu noble-security InRelease [126 kB]
Get:5 http://ph.archive.ubuntu.com/ubuntu noble-updates/main amd64 Packages [1,3
13 kB]
Get:6 http://ph.archive.ubuntu.com/ubuntu noble-updates/main Translation-en [264
 kB]
Get:7 http://ph.archive.ubuntu.com/ubuntu noble-updates/main amd64 Components [1
64 kB]
Get:8 http://ph.archive.ubuntu.com/ubuntu noble-updates/restricted amd64 Compone
nts [212 B]
Get:9 http://ph.archive.ubuntu.com/ubuntu noble-updates/universe amd64 Packages
[1,120 kB]
Get:10 http://security.ubuntu.com/ubuntu noble-security/main amd64 Packages [1,0
54 kB]
Get:11 http://ph.archive.ubuntu.com/ubuntu noble-updates/universe Translation-en
 [287 kB]
Get:12 http://ph.archive.ubuntu.com/ubuntu noble-updates/universe amd64 Componen
ts [377 kB]
Get:13 http://ph.archive.ubuntu.com/ubuntu noble-updates/multiverse amd64 Compon
ents [940 B]
```

```
vbearl@workstation:~$ 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 libllvm17t64 python3-netifaces
Use 'sudo apt autoremove' to remove them.
The following NEW packages will be installed:
  libllvm19 libmalcontent-0-0 mesa-libgallium
The following packages have been kept back:
  libgl1-amber-dri libglapi-mesa
The following packages will be upgraded:
  acl alsa-ucm-conf apparmor apt apt-utils base-files bluez bluez-cups
  bluez-obexd bsdextrautils bsdutils cloud-init cups-browsed dbus dbus-bin
  dbus-daemon dbus-session-bus-common dbus-system-bus-common dbus-user-session
  dhcpcd-base distro-info-data dmidecode dmsetup dns-root-data dnsmasq-base
  dracut-install e2fsprogs e2fsprogs-l10n eject evince evince-common
  evolution-data-server evolution-data-server-common fdisk
  fonts-noto-color-emoji fwupd gdm3 gir1.2-gdesktopenums-3.0 gir1.2-gdm-1.0
  gir1.2-gnomebluetooth-3.0 gir1.2-gtk-3.0 gir1.2-gtk-4.0 gir1.2-mutter-14
  gir1.2-nm-1.0 gir1.2-packagekitglib-1.0 gir1.2-polkit-1.0
  gnome-bluetooth-3-common gnome-bluetooth-sendto gnome-calculator
  gnome-control-center gnome-control-center-data gnome-control-center-faces
```

```
vbearl@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 libllvm17t64 python3-netifaces
Use 'sudo apt autoremove' to remove them.
The following NEW packages will be installed:
  libllvm19 libmalcontent-0-0 mesa-libgallium
The following packages have been kept back:
  libgl1-amber-dri libglapi-mesa
The following packages will be upgraded:
  acl alsa-ucm-conf apparmor apt apt-utils base-files bluez bluez-cups
  bluez-obexd bsdextrautils bsdutils cloud-init cups-browsed dbus dbus-bin
  dbus-daemon dbus-session-bus-common dbus-system-bus-common dbus-user-session
  dhcpcd-base distro-info-data dmidecode dmsetup dns-root-data dnsmasq-base
  dracut-install e2fsprogs e2fsprogs-l10n eject evince evince-common
  evolution-data-server evolution-data-server-common fdisk
  fonts-noto-color-emoji fwupd gdm3 gir1.2-gdesktopenums-3.0 gir1.2-gdm-1.0
  gir1.2-gnomebluetooth-3.0 gir1.2-gtk-3.0 gir1.2-gtk-4.0 gir1.2-mutter-14
  gir1.2-nm-1.0 gir1.2-packagekitglib-1.0 gir1.2-polkit-1.0
  gnome-bluetooth-3-common gnome-bluetooth-sendto gnome-calculator
  gnome-control-center gnome-control-center-data gnome-control-center-faces
```

```
vbearl@server2:~$ 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 libllvm17t64 python3-netifaces
Use 'sudo apt autoremove' to remove them.
The following NEW packages will be installed:
  libllvm19 libmalcontent-0-0 mesa-libgallium
The following packages have been kept back:
  libgl1-amber-dri libglapi-mesa
The following packages will be upgraded:
  acl alsa-ucm-conf apparmor apt apt-utils base-files bluez bluez-cups
  bluez-obexd bsdextrautils bsdutils cloud-init cups-browsed dbus dbus-bin
  dbus-daemon dbus-session-bus-common dbus-system-bus-common dbus-user-session
  dhcpcd-base distro-info-data dmidecode dmsetup dns-root-data dnsmasq-base
  dracut-install e2fsprogs e2fsprogs-l10n eject evince evince-common
  evolution-data-server evolution-data-server-common fdisk
  fonts-noto-color-emoji fwupd gdm3 gir1.2-gdesktopenums-3.0 gir1.2-gdm-1.0
  gir1.2-gnomebluetooth-3.0 gir1.2-gtk-3.0 gir1.2-gtk-4.0 gir1.2-mutter-14
  gir1.2-nm-1.0 gir1.2-packagekitglib-1.0 gir1.2-polkit-1.0
  gnome-bluetooth-3-common gnome-bluetooth-sendto gnome-calculator
```

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

```
vbearl@workstation:~$ sudo apt install openssh-server
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
The following packages were automatically installed and are no longer required:
  libgl1-amber-dri libglapi-mesa libllvm17t64 python3-netifaces
Use 'sudo apt autoremove' to remove them.
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 832 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-sftp-
server amd64 1:9.6p1-3ubuntu13.13 [37.1 kB]
Get:2 http://ph.archive.ubuntu.com/ubuntu noble-updates/main amd64 openssh-serve
r amd64 1:9.6p1-3ubuntu13.13 [510 kB]
```

```
vbearl@server2:~$ sudo apt install openssh-server
[sudo] password for vbearl:
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
openssh-server is already the newest version (1:9.6p1-3ubuntu13.13).
The following packages were automatically installed and are no longer required:
  libgl1-amber-dri libglapi-mesa libllvm17t64 python3-netifaces
Use 'sudo apt autoremove' to remove them.
0 upgraded, 0 newly installed, 0 to remove and 2 not upgraded.
```

```
vbearl@server1:~$ sudo apt install openssh-server
[sudo] password for vbearl:
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
The following packages were automatically installed and are no longer required:
  libgl1-amber-dri libglapi-mesa libllvm17t64 python3-netifaces
Use 'sudo apt autoremove' to remove them.
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
```

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

   3.1 *sudo service ssh start*

```
vbearl@workstation:~$ sudo service ssh start
```

```
vbearl@server1:~$ sudo service ssh start
```

```
vbearl@server2:~$ sudo service ssh start
```

   3.2 *sudo systemctl status ssh*

```
vbearl@workstation:~$ sudo systemctl status ssh
● ssh.service - OpenBSD Secure Shell server
     Loaded: loaded (/usr/lib/systemd/system/ssh.service; disabled; preset: ena>
     Active: active (running) since Fri 2025-08-08 09:53:10 UTC; 2min 23s ago
TriggeredBy: ● ssh.socket
       Docs: man:sshd(8)
             man:sshd_config(5)
    Process: 26462 ExecStartPre=/usr/sbin/sshd -t (code=exited, status=0/SUCCES>
   Main PID: 26464 (sshd)
      Tasks: 1 (limit: 4603)
     Memory: 1.2M (peak: 1.6M)
        CPU: 18ms
     CGroup: /system.slice/ssh.service
             └─26464 "sshd: /usr/sbin/sshd -D [listener] 0 of 10-100 startups"
```

```
vbearl@server2:~$ sudo systemctl status ssh
● ssh.service - OpenBSD Secure Shell server
     Loaded: loaded (/usr/lib/systemd/system/ssh.service; disabled; preset: ena>
     Active: active (running) since Fri 2025-08-08 09:54:04 UTC; 1min 58s ago
TriggeredBy: ● ssh.socket
       Docs: man:sshd(8)
             man:sshd_config(5)
    Process: 2994 ExecStartPre=/usr/sbin/sshd -t (code=exited, status=0/SUCCESS)
   Main PID: 2996 (sshd)
      Tasks: 1 (limit: 4603)
     Memory: 2.1M (peak: 2.6M)
        CPU: 20ms
```

```
vbearl@server1:~$ sudo systemctl status ssh
● ssh.service - OpenBSD Secure Shell server
     Loaded: loaded (/usr/lib/systemd/system/ssh.service; disabled; preset: ena>
     Active: active (running) since Fri 2025-08-08 09:53:32 UTC; 3min 23s ago
TriggeredBy: ● ssh.socket
       Docs: man:sshd(8)
             man:sshd_config(5)
    Process: 26368 ExecStartPre=/usr/sbin/sshd -t (code=exited, status=0/SUCCES>
   Main PID: 26369 (sshd)
      Tasks: 1 (limit: 4603)
     Memory: 1.2M (peak: 1.6M)
        CPU: 21ms
     CGroup: /system.slice/ssh.service
             └─26369 "sshd: /usr/sbin/sshd -D [listener] 0 of 10-100 startups"
```

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

4.1 *sudo ufw allow ssh*

```
vbearl@workstation:~$ sudo ufw allow ssh
Rules updated
Rules updated (v6)
```

```
vbearl@server2:~$ sudo ufw allow ssh
Rules updated
Rules updated (v6)
```

```
vbearl@server1:~$ sudo ufw allow ssh
Rules updated
Rules updated (v6)
```

4.2 *sudo ufw enable*

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

```
vbearl@server1:~$ sudo ufw enable
Firewall is active and enabled on system startup
```

```
vbearl@server2:~$ sudo ufw enable
Firewall is active and enabled on system startup
```

4.3 *sudo ufw status*

```
vbearl@workstation:~$ sudo ufw status
Status: active

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

```
vbearl@server1:~$ sudo ufw status
Status: active

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

```
vbearl@server2:~$ 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.**108**

```
enp0s8: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1500
        inet 192.168.56.108  netmask 255.255.255.0  broadcast 192.168.56.255
```

   1.2 Server 2 IP address: 192.168.56.**107**

```
enp0s8: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1500
        inet 192.168.56.107  netmask 255.255.255.0  broadcast 192.168.56.255
```

   1.3 Server 3 IP address: 192.168.56.**106**

```
enp0s8: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1500
        inet 192.168.56.106  netmask 255.255.255.0  broadcast 192.168.56.255
```

2. Make sure that they can ping each other.

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

```
vbearl@workstation:~$ ping 192.168.56.107
PING 192.168.56.107 (192.168.56.107) 56(84) bytes of data.
64 bytes from 192.168.56.107: icmp_seq=1 ttl=64 time=0.466 ms
64 bytes from 192.168.56.107: icmp_seq=2 ttl=64 time=0.490 ms
64 bytes from 192.168.56.107: icmp_seq=3 ttl=64 time=0.367 ms
64 bytes from 192.168.56.107: icmp_seq=4 ttl=64 time=0.487 ms
^C
--- 192.168.56.107 ping statistics ---
4 packets transmitted, 4 received, 0% packet loss, time 3120ms
rtt min/avg/max/mdev = 0.367/0.452/0.490/0.050 ms
```

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

```
vbearl@workstation:~$ 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=1.11 ms
64 bytes from 192.168.56.106: icmp_seq=2 ttl=64 time=0.915 ms
64 bytes from 192.168.56.106: icmp_seq=3 ttl=64 time=1.36 ms
64 bytes from 192.168.56.106: icmp_seq=4 ttl=64 time=0.664 ms
^C
--- 192.168.56.106 ping statistics ---
4 packets transmitted, 4 received, 0% packet loss, time 6826ms
rtt min/avg/max/mdev = 0.664/1.011/1.356/0.254 ms
```

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

```
vbearl@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=1.12 ms
64 bytes from 192.168.56.106: icmp_seq=2 ttl=64 time=0.823 ms
64 bytes from 192.168.56.106: icmp_seq=3 ttl=64 time=1.77 ms
64 bytes from 192.168.56.106: icmp_seq=4 ttl=64 time=0.360 ms
^C
--- 192.168.56.106 ping statistics ---
4 packets transmitted, 4 received, 0% packet loss, time 4302ms
rtt min/avg/max/mdev = 0.360/1.018/1.765/0.509 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*

```
vbearl@workstation:~$ ssh vbearl@192.168.56.107
The authenticity of host '192.168.56.107 (192.168.56.107)' can't be established.
ED25519 key fingerprint is SHA256:9D2t23fa1VSbYaADtRoKK8W6xAW/r8xGUcM5VaNUKoY.
This key is not known by any other names.
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added '192.168.56.107' (ED25519) to the list of known hosts
```

1.2 Enter the password for server 1 when prompted

```
vbearl@192.168.56.107'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.

1 update can be applied immediately.
To see these additional updates run: apt list --upgradable

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

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

```
vbearl@server1:~$
```

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

```
vbearl@server1:~$
logout
Connection to 192.168.56.107 closed.
vbearl@workstation:~$
```

3. Do the same for Server 2.

```
vbearl@workstation:~$ ssh vbearl@192.168.56.106
The authenticity of host '192.168.56.106 (192.168.56.106)' can't be established.
ED25519 key fingerprint is SHA256:9D2t23fa1VSbYaADtRoKK8W6xAW/r8xGUcM5VaNUKoY.
This host key is known by the following other names/addresses:
    ~/.ssh/known_hosts:1: [hashed name]
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added '192.168.56.106' (ED25519) to the list of known hosts
.
```

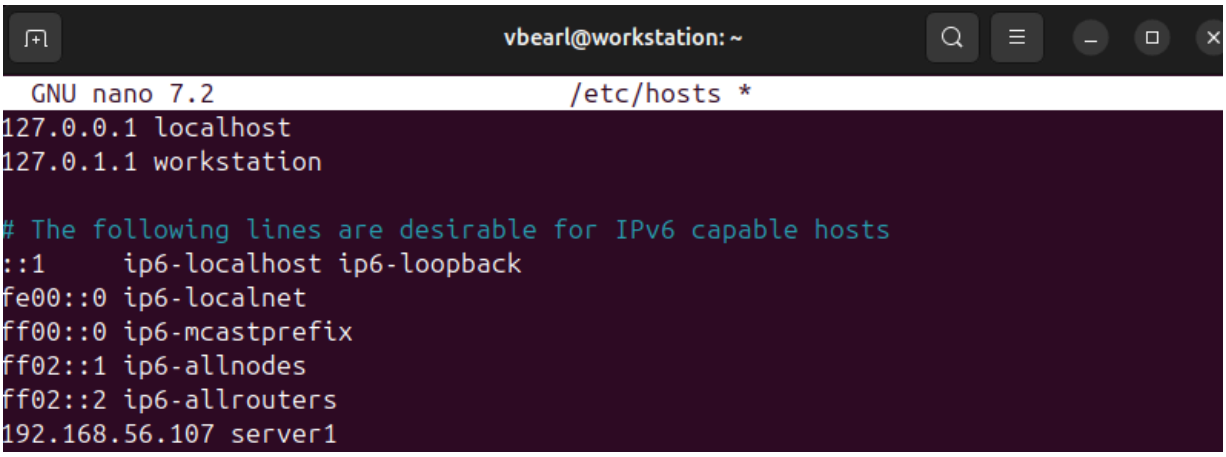
```
vbearl@192.168.56.106's password:
Welcome to Ubuntu 24.04.3 LTS (GNU/Linux 6.14.0-27-generic x86_64)

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

Expanded Security Maintenance for Applications is not enabled.

1 update can be applied immediately.
To see these additional updates run: apt list --upgradable

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

```
vbearl@server2:~$
```

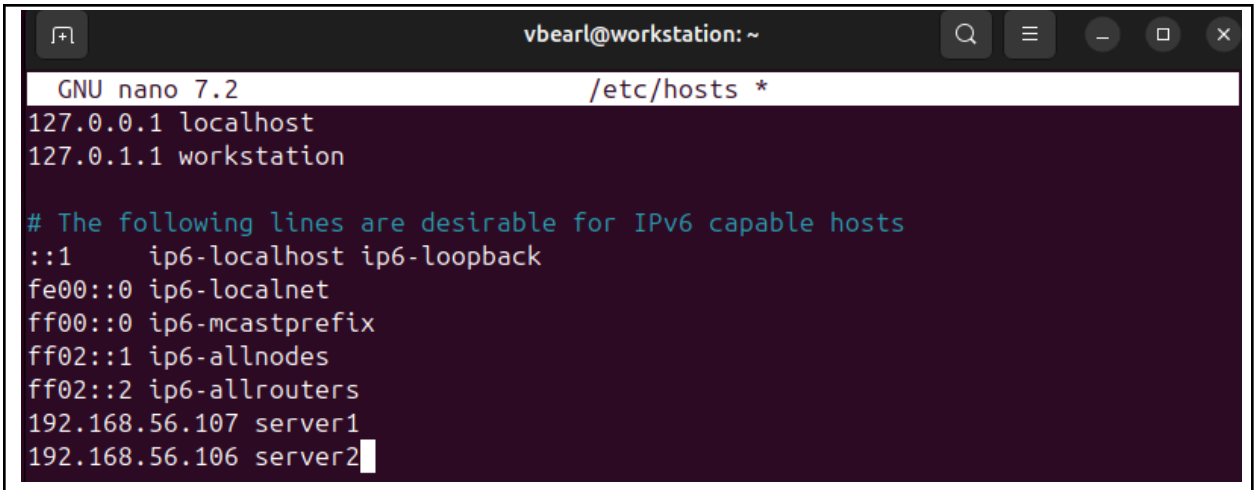
```
vbearl@server2:~$
logout
Connection to 192.168.56.106 closed.
vbearl@workstation:~$
```

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.

```
vbearl@workstation:~$ ssh vbearl@server2
The authenticity of host 'server2 (192.168.56.106)' can't be established.
ED25519 key fingerprint is SHA256:9D2t23fa1VSbYaADtRoKK8W6xAW/r8xGUcM5VaNUKoY.
This host key is known by the following other names/addresses:
    ~/.ssh/known_hosts:1: [hashed name]
    ~/.ssh/known_hosts:4: [hashed name]
    ~/.ssh/known_hosts:5: [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.
vbearl@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
```

Reflections:

Answer the following:

1. How are we able to use the hostname instead of IP address in SSH commands?
    - This is because we already wrote the ip addresses of the two servers within /etc/hosts. After including the ip addresses and the hostnames, the system can now identify the servers by their hostnames, allowing the local machine to ssh the servers by their hostnames instead of IP addresses as seen in the output above.
2. How secured is SSH?
    - SSH is a high level security protocol that was refined to replace Telnet for its lack of security. SSH allows us to encrypt passwords, provide strong authentication for logins, and overall provide a strong security for automated servers. Unlike Telnet, which uses unsecured plain text for its authentication, SSH can encrypt your passwords and provide public keys to ensure strong security and avoid unauthorized access and potential hacking.