| Name: Allen Jerome B. Gonzales	Date Performed:8-15-2023
Course/Section: CPE31S4	Date Submitted: 8-15-2023
Instructor: Sir J. 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*).

Server 1   Server 2
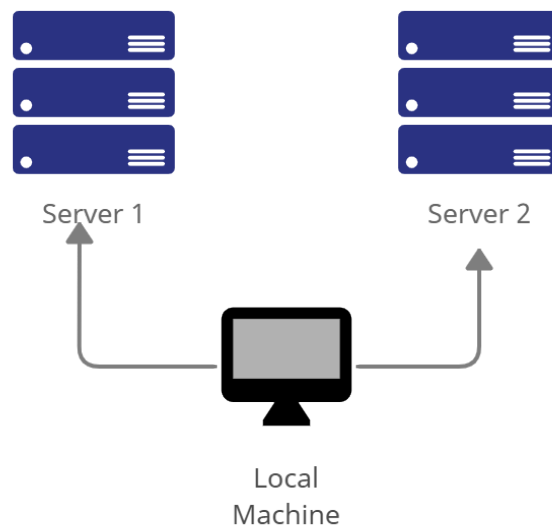
Local
Machine

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

```
GNU nano 2.9.3                              /etc/hostname

controlNode1
```

1.2 Use server2 for Server 2

```
GNU nano 2.9.3                              /etc/hostname

controlNode2
```

1.3 Use workstation for the Local Machine

```
GNU nano 2.9.3                              /etc/hostname

manageMode
```
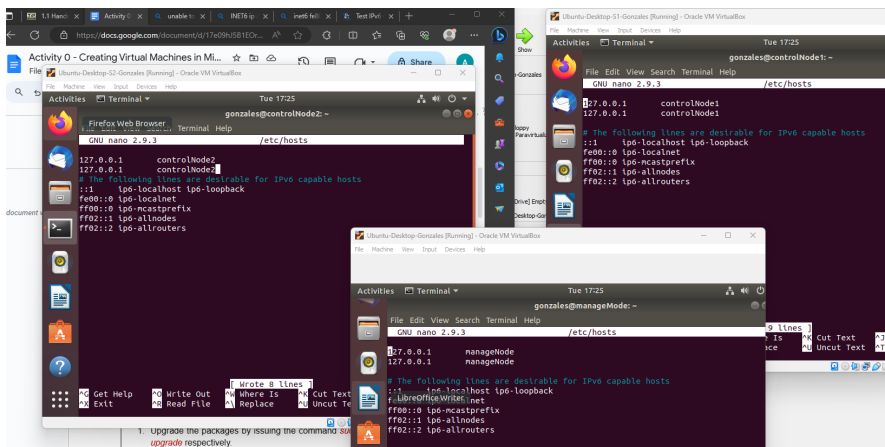
2. Edit the hosts using the command *sudo nano /etc/hosts.* Edit the second line.
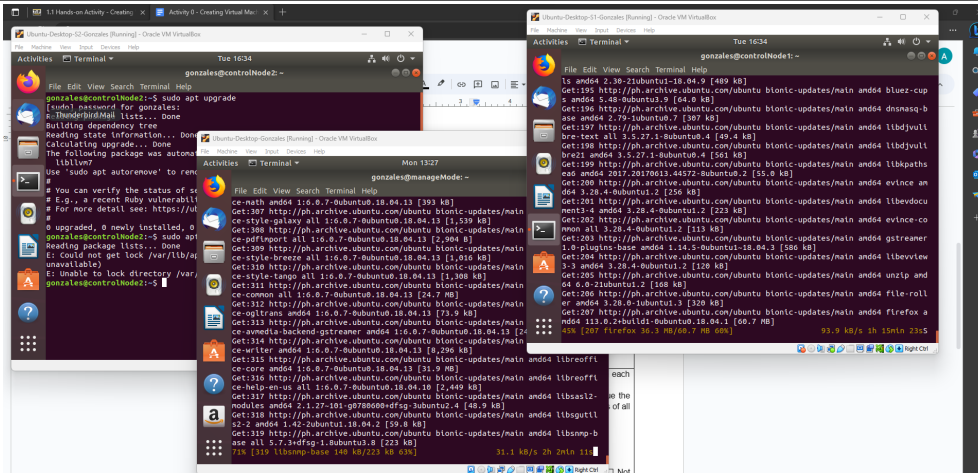   2.1 Type 127.0.0.1 server 1 for Server 1
   2.2 Type 127.0.0.1 server 2 for Server 2
   2.3 Type 127.0.0.1 workstation for the Local Machine

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.

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


```
gonzales@manageMode:~$ sudo apt install openssh-server
[sudo] password for gonzales:
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following package was automatically installed and is no longer required:
  libllvm7
Use 'sudo apt autoremove' to remove it.
The following additional packages will be installed:
  ncurses-term openssh-sftp-server ssh-import-id
Suggested packages:
  molly-guard monkeysphere rssh 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 0 not upgraded.
Need to get 637 kB of archives.
After this operation, 5,320 kB of additional disk space will be used.
Do you want to continue? [Y/n] S
```

```
gonzales@controlNode2:~$ sudo apt install openssh-server
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following package was automatically installed and is no longer required:
  libllvm7
Use 'sudo apt autoremove' to remove it.
The following additional packages will be installed:
  ncurses-term openssh-sftp-server ssh-import-id
Suggested packages:
  molly-guard monkeysphere rssh 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 0 not upgraded.
Need to get 637 kB of archives.
After this operation, 5,320 kB of additional disk space will be used.
Do you want to continue? [Y/n] y
Get:1 http://ph.archive.ubuntu.com/ubuntu bionic-updates/main amd64 ncurses-ter
m all 6.1-1ubuntu1.18.04.1 [248 kB]
Get:2 http://ph.archive.ubuntu.com/ubuntu bionic-updates/main amd64 openssh-sft
p-server amd64 1:7.6p1-4ubuntu0.7 [45.5 kB]
Get:3 http://ph.archive.ubuntu.com/ubuntu bionic-updates/main amd64 openssh-ser
ver amd64 1:7.6p1-4ubuntu0.7 [332 kB]
63% [3 openssh-server 131 kB/332 kB 39%]
```

```
Processing triggers for dbus (1.12.2-1ubuntu1.4) ...
gonzales@controlNode1:~$ sudo apt install openssh-server
[sudo] password for gonzales:
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following package was automatically installed and is no longer required:
  libllvm7
Use 'sudo apt autoremove' to remove it.
The following additional packages will be installed:
  ncurses-term openssh-sftp-server ssh-import-id
Suggested packages:
  molly-guard monkeysphere rssh 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 0 not upgraded.
Need to get 637 kB of archives.
After this operation, 5,320 kB of additional disk space will be used.
Do you want to continue? [Y/n]
```

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*

```
gonzales@manageMode:~$ sudo service ssh start
gonzales@manageMode:~$ sudo systemctl status ssh
● ssh.service - OpenBSD Secure Shell server
   Loaded: loaded (/lib/systemd/system/ssh.service; enabled; vendor preset: ena
   Active: active (running) since Tue 2023-08-15 17:02:28 +08; 18s ago
 Main PID: 21368 (sshd)
    Tasks: 1 (limit: 2318)
   CGroup: /system.slice/ssh.service
           └─21368 /usr/sbin/sshd -D

Aug 15 17:02:28 manageMode systemd[1]: Starting OpenBSD Secure Shell server...
Aug 15 17:02:28 manageMode sshd[21368]: Server listening on 0.0.0.0 port 22.
Aug 15 17:02:28 manageMode sshd[21368]: Server listening on :: port 22.
Aug 15 17:02:28 manageMode systemd[1]: Started OpenBSD Secure Shell server.
```

```
gonzales@controlNode1:~$ sudo service ssh start
gonzales@controlNode1:~$ sudo systemctl status ssh
● ssh.service - OpenBSD Secure Shell server
   Loaded: loaded (/lib/systemd/system/ssh.service; enabled; vendor preset: ena
   Active: active (running) since Tue 2023-08-15 17:08:23 +08; 34s ago
 Main PID: 21264 (sshd)
    Tasks: 1 (limit: 2318)
   CGroup: /system.slice/ssh.service
           └─21264 /usr/sbin/sshd -D

Aug 15 17:08:22 controlNode1 systemd[1]: Starting OpenBSD Secure Shell server..
Aug 15 17:08:23 controlNode1 sshd[21264]: Server listening on 0.0.0.0 port 22.
Aug 15 17:08:23 controlNode1 sshd[21264]: Server listening on :: port 22.
Aug 15 17:08:23 controlNode1 systemd[1]: Started OpenBSD Secure Shell server.
gonzales@controlNode1:~$
```

```
gonzales@controlNode2:~$ sudo service ssh start
gonzales@controlNode2:~$ sudo systemctl status ssh
● ssh.service - OpenBSD Secure Shell server
   Loaded: loaded (/lib/systemd/system/ssh.service; enabled; vendor preset: ena
   Active: active (running) since Tue 2023-08-15 16:38:26 +08; 4min 1s ago
 Main PID: 4087 (sshd)
    Tasks: 1 (limit: 2318)
   CGroup: /system.slice/ssh.service
           └─4087 /usr/sbin/sshd -D

Aug 15 16:38:26 controlNode2 systemd[1]: Starting OpenBSD Secure Shell server..
Aug 15 16:38:26 controlNode2 sshd[4087]: Server listening on 0.0.0.0 port 22.
Aug 15 16:38:26 controlNode2 sshd[4087]: Server listening on :: port 22.
Aug 15 16:38:26 controlNode2 systemd[1]: Started OpenBSD Secure Shell server.
lines 1-12/12 (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*

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

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

gonzales@manageMode:~$ 
```

```
Aug 15 17:08:25 controlNode1 systemd[1]: Started OpenBSD
gonzales@controlNode1:~$ sudo ufw allow ssh
Rules updated
Rules updated (v6)
gonzales@controlNode1:~$ sudo ufw enable
Firewall is active and enabled on system startup
gonzales@controlNode1:~$ sudo ufw status
Status: active

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

gonzales@controlNode1:~$ 
```

```
gonzales@controlNode2:~$ sudo ufw allow ssh
Rules updated
Rules updated (v6)
gonzales@controlNode2:~$ sudo ufw enable
Firewall is active and enabled on system startup
gonzales@controlNode2:~$ sudo ufw status
Status: active

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

gonzales@controlNode2:~$ S
```

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

```
enp0s3: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1500
        inet 10.0.2.15  netmask 255.255.255.0  broadcast 10.0.2.255
        inet6 fe80::6b70:b45c:6b13:7483  prefixlen 64  scopeid 0x20<link>
        ether 08:00:27:27:b3:8e  txqueuelen 1000  (Ethernet)
        RX packets 1066  bytes 1373254 (1.3 MB)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 514  bytes 49014 (49.0 KB)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0

enp0s8: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1500
        inet 192.168.56.105  netmask 255.255.255.0  broadcast 192.168.56.255
        inet6 fe80::9f5d:789d:6fb7:4313  prefixlen 64  scopeid 0x20<link>
        ether 08:00:27:d4:6b:83  txqueuelen 1000  (Ethernet)
        RX packets 9  bytes 1619 (1.6 KB)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 61  bytes 7002 (7.0 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 185  bytes 14815 (14.8 KB)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 185  bytes 14815 (14.8 KB)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0

gonzales@manageMode:~$
```

1.2 Server 2 IP address: 192.168.56.106

```
enp0s3: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1500
        inet 10.0.2.15  netmask 255.255.255.0  broadcast 10.0.2.255
        inet6 fe80::1376:50ef:5be0:7bef  prefixlen 64  scopeid 0x20<link>
        ether 08:00:27:a9:b9:c9  txqueuelen 1000  (Ethernet)
        RX packets 19  bytes 3035 (3.0 KB)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 73  bytes 7552 (7.5 KB)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0

enp0s8: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1500
        inet 192.168.56.106  netmask 255.255.255.0  broadcast 192.168.56.255
        inet6 fe80::6b34:6a9a:679a:34a6  prefixlen 64  scopeid 0x20<link>
        ether 08:00:27:21:22:37  txqueuelen 1000  (Ethernet)
        RX packets 8  bytes 1559 (1.5 KB)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 52  bytes 6151 (6.1 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 158  bytes 12257 (12.2 KB)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 158  bytes 12257 (12.2 KB)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0
```

1.3 Server 3 IP address: 192.168.56.107

```
enp0s3: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1500
        inet 10.0.2.15  netmask 255.255.255.0  broadcast 10.0.2.255
        inet6 fe80::5e4:8806:6b75:70bb  prefixlen 64  scopeid 0x20<link>
        ether 08:00:27:59:69:ef  txqueuelen 1000  (Ethernet)
        RX packets 821  bytes 1054651 (1.0 MB)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 402  bytes 34914 (34.9 KB)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0

enp0s8: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1500
        inet 192.168.56.107  netmask 255.255.255.0  broadcast 192.168.56.255
        inet6 fe80::2aa8:139a:1b96:fe08  prefixlen 64  scopeid 0x20<link>
        ether 08:00:27:21:fd:80  txqueuelen 1000  (Ethernet)
        RX packets 17  bytes 2331 (2.3 KB)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 54  bytes 6438 (6.4 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 163  bytes 12875 (12.8 KB)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 163  bytes 12875 (12.8 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: Yes Successful ☐ Not Successful

```
gonzales@manageMode:~$ 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=1.19 ms
64 bytes from 192.168.56.106: icmp_seq=2 ttl=64 time=1.53 ms
64 bytes from 192.168.56.106: icmp_seq=3 ttl=64 time=1.09 ms
64 bytes from 192.168.56.106: icmp_seq=4 ttl=64 time=1.35 ms

64 bytes from 192.168.56.106: icmp_seq=5 ttl=64 time=0.509 ms
64 bytes from 192.168.56.106: icmp_seq=6 ttl=64 time=1.94 ms
q
64 bytes from 192.168.56.106: icmp_seq=7 ttl=64 time=0.975 ms
^[[C64 bytes from 192.168.56.106: icmp_seq=8 ttl=64 time=0.673 ms
64 bytes from 192.168.56.106: icmp_seq=9 ttl=64 time=1.03 ms
64 bytes from 192.168.56.106: icmp_seq=10 ttl=64 time=1.50 ms
64 bytes from 192.168.56.106: icmp_seq=11 ttl=64 time=1.60 ms
^Z
[1]+  Stopped                 ping 192.168.56.106
gonzales@manageMode:~$
```

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

```
gonzales@manageMode:~$ 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=1.57 ms
64 bytes from 192.168.56.107: icmp_seq=2 ttl=64 time=1.16 ms
64 bytes from 192.168.56.107: icmp_seq=3 ttl=64 time=0.796 ms
64 bytes from 192.168.56.107: icmp_seq=4 ttl=64 time=0.954 ms
^Z
[2]+  Stopped                 ping 192.168.56.107
gonzales@manageMode:~$
```

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

```
gonzales@controlNode1:~$ 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=0.905 ms
64 bytes from 192.168.56.107: icmp_seq=2 ttl=64 time=0.413 ms
64 bytes from 192.168.56.107: icmp_seq=3 ttl=64 time=0.582 ms
^Z
[1]+  Stopped                 ping 192.168.56.107
```

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*

```
gonzales@manageMode:~$ ssh gonzales@192.168.56.106
The authenticity of host '192.168.56.106 (192.168.56.106)' can't be established
.
ECDSA key fingerprint is SHA256:Dy4AIu9OADks0I5vHmq9VklBL+timB0jagDOvXT8RyY.
Are you sure you want to continue connecting (yes/no)? y
Please type 'yes' or 'no': yes
```

1.2 Enter the password for server 1 when prompted

```
Warning: Permanently added '192.168.56.106' (ECDSA) to the list of known hosts.
gonzales@192.168.56.106's password:
```

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

```
Welcome to Ubuntu 18.04.6 LTS (GNU/Linux 5.4.0-150-generic x86_64)

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

Expanded Security Maintenance for Infrastructure is not enabled.

0 updates can be applied immediately.

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

Your Hardware Enablement Stack (HWE) is supported until April 2023.

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

Ubuntu comes with ABSOLUTELY NO WARRANTY, to the extent permitted by
applicable law.

gonzales@controlNode1:~$
```

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

```
gonzales@controlNode1:~$ logout
Connection to 192.168.56.106 closed.
gonzales@manageMode:~$
```

3. Do the same for Server 2.

```
gonzales@manageMode:~$ ssh gonzales@192.168.56.107
The authenticity of host '192.168.56.107 (192.168.56.107)' can't be establishe
.
ECDSA key fingerprint is SHA256:27Hlt7cmHa7jBFcruVAgfLHMAS25A0QyFemCFGqbNzs.
Are you sure you want to continue connecting (yes/no)? yes
Warning: Permanently added '192.168.56.107' (ECDSA) to the list of known host
gonzales@192.168.56.107's password:
Welcome to Ubuntu 18.04.6 LTS (GNU/Linux 5.4.0-150-generic x86_64)

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

Expanded Security Maintenance for Infrastructure is not enabled.

0 updates can be applied immediately.

77 additional security updates can be applied with ESM Infra.
Learn more about enabling ESM Infra service for Ubuntu 18.04 at
https://ubuntu.com/18-04

Your Hardware Enablement Stack (HWE) is supported until April 2023.

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

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

```
gonzales@controlNode2:~$ logout
Connection to 192.168.56.107 closed.
gonzales@manageMode:~$
```

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 2.9.3                         /etc/hosts                              Modi
127.0.0.1          manageNode
127.0.0.1          manageNode

192.168.56.106 controlNode1
192.168.56.107 controlNode2

# 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
```

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.

```
gonzales@manageMode:~$ ssh gonzales@controlNode1
The authenticity of host 'controlnode1 (192.168.56.106)' can't be established.
ECDSA key fingerprint is SHA256:Dy4AIu9OADks0I5vHmq9VklBL+timB0jagDOvXT8RyY.
Are you sure you want to continue connecting (yes/no)? yes
Warning: Permanently added 'controlnode1' (ECDSA) to the list of known hosts.
gonzales@controlnode1's password:
Welcome to Ubuntu 18.04.6 LTS (GNU/Linux 5.4.0-150-generic x86_64)

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

Expanded Security Maintenance for Infrastructure is not enabled.

0 updates can be applied immediately.

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

New release '20.04.6 LTS' available.
Run 'do-release-upgrade' to upgrade to it.

Your Hardware Enablement Stack (HWE) is supported until April 2023.
Last login: Tue Aug 15 17:47:57 2023 from 192.168.56.105
gonzales@controlNode1:~$ S
```

```
gonzales@manageMode:~$ ssh gonzales@controlNode2
The authenticity of host 'controlnode2 (192.168.56.107)' can't be established.
ECDSA key fingerprint is SHA256:27Hlt7cmHa7jBFcruVAgfLHMAS25A0QyFemCFGqbNzs.
Are you sure you want to continue connecting (yes/no)? yes
Warning: Permanently added 'controlnode2' (ECDSA) to the list of known hosts.
gonzales@controlnode2's password:
Welcome to Ubuntu 18.04.6 LTS (GNU/Linux 5.4.0-150-generic x86_64)

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

Expanded Security Maintenance for Infrastructure is not enabled.

0 updates can be applied immediately.

77 additional security updates can be applied with ESM Infra.
Learn more about enabling ESM Infra service for Ubuntu 18.04 at
https://ubuntu.com/18-04

New release '20.04.6 LTS' available.
Run 'do-release-upgrade' to upgrade to it.

Your Hardware Enablement Stack (HWE) is supported until April 2023.
Last login: Tue Aug 15 17:39:02 2023 from 192.168.56.105
gonzales@controlNode2:~$
```

Reflections:

Answer the following:

1. How are we able to use the hostname instead of IP address in SSH commands?
   You can use the hostname instead of IP address in SSH commands because you
   inserted the IP address and hostname of the servers in the /etc/hosts

2. How secured is SSH?

   SSH or known as Secure Shell is a secure protocol that provides a mechanism for
   establishing a secure connection between two parties. It is key based encryption
   and authentication that is not prone to brute-force attack