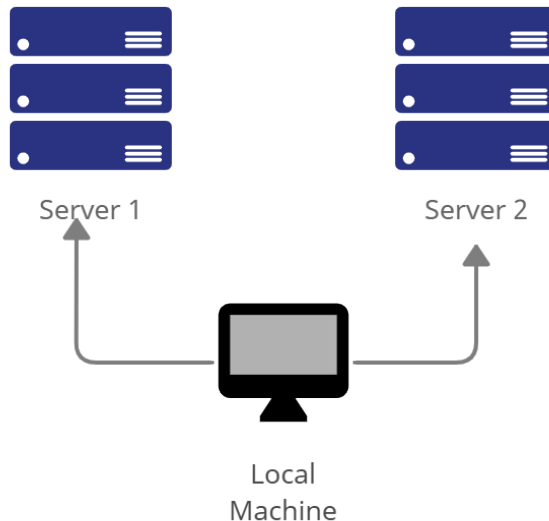


Name: Ashley L. Mallari	Date Performed: 08/17/23
Course/Section: CPE31S6	Date Submitted: 08/24/23
Instructor: Dr. Jonathan V. Taylar	Semester and SY: 1st Sem / 2023-2024
Activity 1: Configure Network using Virtual Machines	
1. Objectives: 1.1. Create and configure Virtual Machines in Microsoft Azure 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, <i>provide screenshots for each task</i> . (Note: <i>it is assumed that you have the prior knowledge of cloning and creating snapshots in a virtual machine</i>).	
 <pre> graph TD LocalMachine[Local Machine] --> Server1[Server 1] LocalMachine --> Server2[Server 2] </pre> <p>The diagram illustrates a network topology where a central 'Local Machine' (represented by a monitor icon) is connected to two separate server stacks. 'Server 1' on the left and 'Server 2' on the right each consist of three stacked server rack icons. Arrows point from the Local Machine to each of the two server stacks, indicating network connectivity.</p>	
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. <ol style="list-style-type: none"> Change the hostname using the command <i>sudo nano /etc/hostname</i> <ol style="list-style-type: none"> Use server1 for Server 1 	

```
File Edit View Search Terminal Help
GNU nano 2.9.3 /etc/hostname
server1
```

1.2 Use server2 for Server 2

```
File Edit View Search Terminal Help
GNU nano 2.9.3 /etc/hostname
server2
```

1.3 Use workstation for the Local Machine

```
ashley@workstation: ~
File Edit View Search Terminal Help
GNU nano 2.9.3 /etc/hostname
workstation
```

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

```
File Edit View Search Terminal Help
GNU nano 2.9.3 /etc/hosts

127.0.0.1    localhost
127.0.1.1    server 1

# 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

```
File Edit View Search Terminal Help
GNU nano 2.9.3 /etc/hosts

127.0.0.1    localhost
127.0.1.1    server 2

# 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

```
ashley@workstation: ~  
File Edit View Search Terminal Help  
GNU nano 2.9.3 /etc/hosts  
  
127.0.0.1    localhost  
127.0.1.1    workstation  
  
# 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.

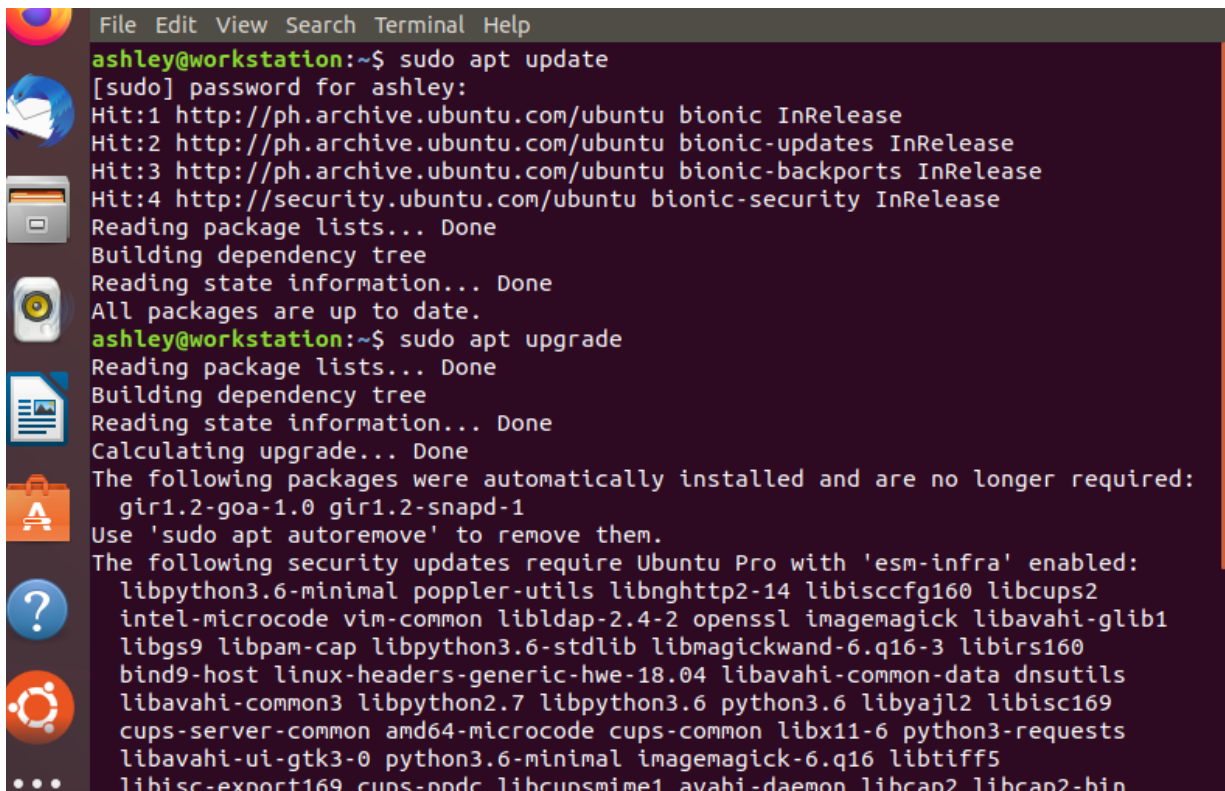
```
ashley@server2:~$ sudo apt update  
[sudo] password for ashley:  
Hit:1 http://ph.archive.ubuntu.com/ubuntu bionic InRelease  
Hit:2 http://ph.archive.ubuntu.com/ubuntu bionic-updates InRelease  
Hit:3 http://ph.archive.ubuntu.com/ubuntu bionic-backports InRelease  
Hit:4 http://security.ubuntu.com/ubuntu bionic-security InRelease  
Reading package lists... Done  
Building dependency tree  
Reading state information... Done  
All packages are up to date.  
ashley@server2:~$ sudo apt upgrade  
Reading package lists... Done  
Building dependency tree  
Reading state information... Done  
Calculating upgrade... Done  
The following packages were automatically installed and are no longer required:  
  gir1.2-goa-1.0 gir1.2-snapd-1  
Use 'sudo apt autoremove' to remove them.  
The following security updates require Ubuntu Pro with 'esm-infra' enabled:  
  libpython3.6-minimal poppler-utils libnghttp2-14 libisccfg160 libcups2  
  intel-microcode vim-common libldap-2.4-2 openssl imagemagick libavahi-glib1  
  libgs9 libpam-cap libpython3.6-stdlib libmagickwand-6.q16-3 libirs160  
  bind9-host linux-headers-generic-hwe-18.04 libavahi-common-data dnsutils
```

```

ashley@server1:~$ sudo apt update
[sudo] password for ashley:
Hit:1 http://ph.archive.ubuntu.com/ubuntu bionic InRelease
Hit:2 http://ph.archive.ubuntu.com/ubuntu bionic-updates InRelease
Hit:3 http://ph.archive.ubuntu.com/ubuntu bionic-backports InRelease
Hit:4 http://security.ubuntu.com/ubuntu bionic-security InRelease
Reading package lists... Done
Building dependency tree
Reading state information... Done
All packages are up to date.
ashley@server1:~$
ashley@server1:~$ sudo apt upgrade
Reading package lists... Done
Building dependency tree
Reading state information... Done
Calculating upgrade... Done
The following packages were automatically installed and are no longer required:
  gir1.2-goa-1.0 gir1.2-snapd-1
Use 'sudo apt autoremove' to remove them.
The following security updates require Ubuntu Pro with 'esm-infra' enabled:
  libpython3.6-minimal poppler-utils libnghttp2-14 libisccfg160 libcups2
  intel-microcode vim-common libldap-2.4-2 openssl imagemagick libavahi-glib1
  libgs9 libpam-cap libpython3.6-stdlib libmagickwand-6.q16-3 libirs160
  bind9-host linux-headers-generic-hwe-18.04 libavahi-common-data dnsutils

```

1



```

File Edit View Search Terminal Help
ashley@workstation:~$ sudo apt update
[sudo] password for ashley:
Hit:1 http://ph.archive.ubuntu.com/ubuntu bionic InRelease
Hit:2 http://ph.archive.ubuntu.com/ubuntu bionic-updates InRelease
Hit:3 http://ph.archive.ubuntu.com/ubuntu bionic-backports InRelease
Hit:4 http://security.ubuntu.com/ubuntu bionic-security InRelease
Reading package lists... Done
Building dependency tree
Reading state information... Done
All packages are up to date.
ashley@workstation:~$ sudo apt upgrade
Reading package lists... Done
Building dependency tree
Reading state information... Done
Calculating upgrade... Done
The following packages were automatically installed and are no longer required:
  gir1.2-goa-1.0 gir1.2-snapd-1
Use 'sudo apt autoremove' to remove them.
The following security updates require Ubuntu Pro with 'esm-infra' enabled:
  libpython3.6-minimal poppler-utils libnghttp2-14 libisccfg160 libcups2
  intel-microcode vim-common libldap-2.4-2 openssl imagemagick libavahi-glib1
  libgs9 libpam-cap libpython3.6-stdlib libmagickwand-6.q16-3 libirs160
  bind9-host linux-headers-generic-hwe-18.04 libavahi-common-data dnsutils
  libavahi-common3 libpython2.7 libpython3.6 python3.6 libyajl2 libisc169
  cups-server-common amd64-microcode cups-common libx11-6 python3-requests
  libavahi-ui-gtk3-0 python3.6-minimal imagemagick-6.q16 libtiff5
  libisc-export169 cups-pkcs libcupsmime1 avahi-daemon libcap2 libcap2-bin

```

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

```
ashley@server2:~$ sudo apt install openssh-server
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following packages were automatically installed and are no longer required:
  gir1.2-goa-1.0 gir1.2-snapd-1
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 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]
```

```
ashley@server1:~$ sudo apt install openssh-server
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following packages were automatically installed and are no longer required:
  gir1.2-goa-1.0 gir1.2-snapd-1
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 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]
```

```
ashley@workstation:~$ sudo apt install openssh-server
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following packages were automatically installed and are no longer required:
  gir1.2-goa-1.0 gir1.2-snapd-1
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 rssh ssh-askpass
```


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*

```
Processing triggers for systemd (237-5ubuntu10.37) ...
ashley@server2:~$ sudo service ssh start
ashley@server2:~$ 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 Mon 2023-08-21 14:09:02 PST; 52s ago
     Main PID: 4293 (sshd)
       Tasks: 1 (limit: 4657)
      CGroup: /system.slice/ssh.service
             └─4293 /usr/sbin/sshd -D

Aug 21 14:09:02 server2 systemd[1]: Starting OpenBSD Secure Shell server...
Aug 21 14:09:02 server2 sshd[4293]: Server listening on 0.0.0.0 port 22.
Aug 21 14:09:02 server2 sshd[4293]: Server listening on :: port 22.
Aug 21 14:09:02 server2 systemd[1]: Started OpenBSD Secure Shell server.
lines 1-12/12 (END)
[1]+  Stopped                  sudo systemctl status ssh
```

```
ashley@server1:~$ sudo service ssh start
ashley@server1:~$ 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 Mon 2023-08-21 13:55:09 PST; 19s ago
     Main PID: 2689 (sshd)
       Tasks: 1 (limit: 4657)
      CGroup: /system.slice/ssh.service
             └─2689 /usr/sbin/sshd -D

Aug 21 13:55:09 server1 systemd[1]: Starting OpenBSD Secure Shell server...
Aug 21 13:55:09 server1 sshd[2689]: Server listening on 0.0.0.0 port 22.
Aug 21 13:55:09 server1 sshd[2689]: Server listening on :: port 22.
Aug 21 13:55:09 server1 systemd[1]: Started OpenBSD Secure Shell server.
lines 1-12/12 (END)
[1]+  Stopped                  sudo systemctl status ssh
```

```
ashley@workstation:~$ sudo service ssh start
ashley@workstation:~$ 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 Mon 2023-08-21 13:30:13 PST; 30s ago
     Main PID: 2685 (sshd)
       Tasks: 1 (limit: 4657)
      CGroup: /system.slice/ssh.service
             └─2685 /usr/sbin/sshd -D

Aug 21 13:30:13 workstation systemd[1]: Starting OpenBSD Secure Shell server...
Aug 21 13:30:13 workstation sshd[2685]: Server listening on 0.0.0.0 port 22.
Aug 21 13:30:13 workstation sshd[2685]: Server listening on :: port 22.
Aug 21 13:30:13 workstation systemd[1]: Started OpenBSD Secure Shell server.
lines 1-12/12 (END)
[1]+  Stopped                  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*

```
ashley@server2:~$ sudo ufw allow ssh
Rules updated
Rules updated (v6)
ashley@server2:~$ sudo ufw enable
Firewall is active and enabled on system startup
ashley@server2:~$ sudo ufw status
Status: active

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

ashley@server2:~$
```

```
ashley@server1:~$ sudo ufw allow ssh
Rules updated
Rules updated (v6)
ashley@server1:~$ sudo ufw enable
Firewall is active and enabled on system startup
ashley@server1:~$ sudo ufw status
Status: active

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

ashley@server1:~$
```



```

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

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

ashley@workstation:~$ █

```

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: 192.168.56.102

```

File Edit View Search Terminal Help
ashley@server1:~$ ifconfig
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::807b:fbdd:56a9:d1af prefixlen 64 scopeid 0x20<link>
    ether 08:00:27:87:ae:90 txqueuelen 1000 (Ethernet)
    RX packets 1006 bytes 1279961 (1.2 MB)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 336 bytes 32959 (32.9 KB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

enp0s8: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
    inet 192.168.56.102 netmask 255.255.255.0 broadcast 192.168.56.255
    inet6 fe80::daed:8679:5cdb:d6d3 prefixlen 64 scopeid 0x20<link>
    ether 08:00:27:e2:ea:83 txqueuelen 1000 (Ethernet)
    RX packets 12 bytes 1590 (1.5 KB)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 65 bytes 7144 (7.1 KB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

```

1.2 Server 2 IP address: 192.168.56.103

```

File Edit View Search Terminal Help
ashley@server2:~$ ifconfig
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::7af1:4e3e:6619:e479 prefixlen 64 scopeid 0x20<link>
    ether 08:00:27:37:d8:8d txqueuelen 1000 (Ethernet)
    RX packets 999 bytes 1274414 (1.2 MB)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 340 bytes 32441 (32.4 KB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

enp0s8: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
    inet 192.168.56.103 netmask 255.255.255.0 broadcast 192.168.56.255
    inet6 fe80::a472:7d5b:ce03:5f56 prefixlen 64 scopeid 0x20<link>
    ether 08:00:27:50:ca:86 txqueuelen 1000 (Ethernet)
    RX packets 7 bytes 969 (969.0 B)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 60 bytes 6669 (6.6 KB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

```

1.3 Server 3 IP address: 192.168.56.101

```

ashley@workstation:~$ ifconfig
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::ef0e:6453:5894:d8ef prefixlen 64 scopeid 0x20<link>
    ether 08:00:27:1e:cd:2c txqueuelen 1000 (Ethernet)
    RX packets 31315 bytes 45029797 (45.0 MB)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 3663 bytes 264173 (264.1 KB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

enp0s8: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
    inet 192.168.56.101 netmask 255.255.255.0 broadcast 192.168.56.255
    inet6 fe80::1f23:656d:dc82:d83 prefixlen 64 scopeid 0x20<link>
    ether 08:00:27:b3:6b:32 txqueuelen 1000 (Ethernet)
    RX packets 19 bytes 3307 (3.3 KB)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 74 bytes 8492 (8.4 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

```
ashley@workstation:~$ 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=0.532 ms
64 bytes from 192.168.56.102: icmp_seq=2 ttl=64 time=0.377 ms
64 bytes from 192.168.56.102: icmp_seq=3 ttl=64 time=0.319 ms
64 bytes from 192.168.56.102: icmp_seq=4 ttl=64 time=0.340 ms
64 bytes from 192.168.56.102: icmp_seq=5 ttl=64 time=0.295 ms
64 bytes from 192.168.56.102: icmp_seq=6 ttl=64 time=0.312 ms
64 bytes from 192.168.56.102: icmp_seq=7 ttl=64 time=0.347 ms
64 bytes from 192.168.56.102: icmp_seq=8 ttl=64 time=0.348 ms
64 bytes from 192.168.56.102: icmp_seq=9 ttl=64 time=0.325 ms
64 bytes from 192.168.56.102: icmp_seq=10 ttl=64 time=0.477 ms
64 bytes from 192.168.56.102: icmp_seq=11 ttl=64 time=0.321 ms
64 bytes from 192.168.56.102: icmp_seq=12 ttl=64 time=0.288 ms
64 bytes from 192.168.56.102: icmp_seq=13 ttl=64 time=0.309 ms
64 bytes from 192.168.56.102: icmp_seq=14 ttl=64 time=0.301 ms
64 bytes from 192.168.56.102: icmp_seq=15 ttl=64 time=0.335 ms
64 bytes from 192.168.56.102: icmp_seq=16 ttl=64 time=0.323 ms
^Z
[1]+  Stopped                  ping 192.168.56.102
ashley@workstation:~$
```

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

```
ashley@workstation:~$ 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=0.924 ms
64 bytes from 192.168.56.103: icmp_seq=2 ttl=64 time=0.307 ms
64 bytes from 192.168.56.103: icmp_seq=3 ttl=64 time=0.305 ms
64 bytes from 192.168.56.103: icmp_seq=4 ttl=64 time=0.343 ms
64 bytes from 192.168.56.103: icmp_seq=5 ttl=64 time=0.288 ms
64 bytes from 192.168.56.103: icmp_seq=6 ttl=64 time=0.324 ms
64 bytes from 192.168.56.103: icmp_seq=7 ttl=64 time=0.247 ms
64 bytes from 192.168.56.103: icmp_seq=8 ttl=64 time=0.329 ms
64 bytes from 192.168.56.103: icmp_seq=9 ttl=64 time=0.301 ms
^Z
[1]+  Stopped                  ping 192.168.56.103
ashley@workstation:~$
```

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

```
ashley@server1:~$ 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=0.562 ms
64 bytes from 192.168.56.103: icmp_seq=2 ttl=64 time=0.373 ms
64 bytes from 192.168.56.103: icmp_seq=3 ttl=64 time=0.297 ms
64 bytes from 192.168.56.103: icmp_seq=4 ttl=64 time=0.378 ms
64 bytes from 192.168.56.103: icmp_seq=5 ttl=64 time=0.316 ms
64 bytes from 192.168.56.103: icmp_seq=6 ttl=64 time=0.314 ms
64 bytes from 192.168.56.103: icmp_seq=7 ttl=64 time=0.288 ms
64 bytes from 192.168.56.103: icmp_seq=8 ttl=64 time=0.324 ms
64 bytes from 192.168.56.103: icmp_seq=9 ttl=64 time=0.254 ms
^Z
[1]+  Stopped                  ping 192.168.56.103
ashley@server1:~$
```

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 jvtaylor@192.168.56.120`

1.2 Enter the password for server 1 when prompted

```
ashley@workstation:~$ ssh ashley@192.168.56.102
ashley@192.168.56.102'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.

85 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: Mon Aug 21 21:30:00 2023 from 192.168.56.101
```

1.3 Verify that you are in server 1. The user should be in this format `user@server1`.

For example, `jvtaylor@server1`

```
ashley@server1:~$
```

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

```
ashley@server1:~$ logout
Connection to 192.168.56.102 closed.
ashley@workstation:~$
```

3. Do the same for Server 2.

```
ashley@workstation:~$ ssh ashley@192.168.56.103
The authenticity of host '192.168.56.103 (192.168.56.103)' can't be established.
ECDSA key fingerprint is SHA256:3yqMnp8LqeI0hrTuDGhk+/Ici4jv4rGgfZC/RtmSvQ.
Are you sure you want to continue connecting (yes/no)? yes
Warning: Permanently added '192.168.56.103' (ECDSA) to the list of known hosts.
ashley@192.168.56.103'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.

85 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.
```

```
ashley@server2:~$
```

```
ashley@server2:~$ logout
Connection to 192.168.56.103 closed.
ashley@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)


```
File Edit View Search Terminal Help
GNU nano 2.9.3                               G                               Modified
127.0.0.1      localhost
127.0.1.1      workstation
192.168.56.102 server 1
192.168.56.103 server 2

# 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

█
```

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 jvtaylor@server1`. Enter the password when prompted. Verify that you have entered Server 1. Do the same for Server 2.

```
File Edit View Search Terminal Help
ashley@workstation:~$ ssh ashley@192.168.56.102
ashley@192.168.56.102'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.

85 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: Wed Aug 23 21:50:17 2023 from 192.168.56.101
ashley@server1:~$ █
```

```
ashley@workstation:~$ ssh ashley@192.168.56.103
ashley@192.168.56.103'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.

7 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: Thu Aug 24 10:27:00 2023 from 192.168.56.101
ashley@server2:~$
```

Reflections:

Answer the following:

1. How are we able to use the hostname instead of IP address in SSH commands?
 - Using hostnames in SSH commands is possible due to the Domain Name System (DNS), a distributed database that translates human-readable hostnames into IP addresses. When you input a hostname, your computer queries DNS servers to retrieve the associated IP address. Alternatively, local hosts files can also be configured to directly map hostnames to IP addresses, enabling more user-friendly and flexible remote server access.
2. How secured is SSH?
 - SSH is highly secure due to its encryption mechanisms, strong authentication methods, and data integrity checks. It encrypts the entire communication between client and server, protecting against eavesdropping and data manipulation. With features like public key authentication and optional two-factor authentication, SSH provides robust protection against unauthorized access, making it a trusted choice for secure remote access and data transfer.

Conclusions/Learnings:

- In this activity, I learned about how to use the SSH command to connect the two servers in the workstation. I also learned how to change the hosts and

hostname to avoid confusion. I tested the connectivity of server 1 and server 2 and it worked so that the workstation can be switched on different servers. Therefore, I find this activity fun yet challenging to do especially when your clone is not working properly and had the wrong ip address.