Name: Christian Rey T. Rife	**Date Performed: January 23, 2024**
Course/Section: BSCPE	**Date Submitted:**
Instructor: Dr. Jonathan V. Taylar	**Semester and SY:** **2nd Semester** **2023 – 2024**

Activity 1: Configure Network using Virtual Machines

1. Objectives:

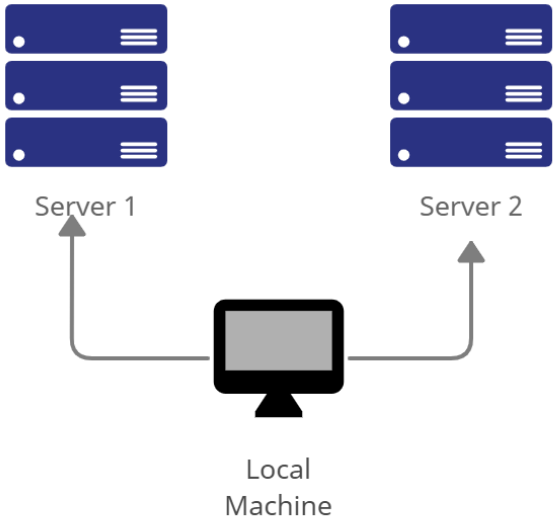
1.1. Create and configure Virtual Machines in Microsoft Azure or VirtualBox

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

2. Discussion:

Network Topology:

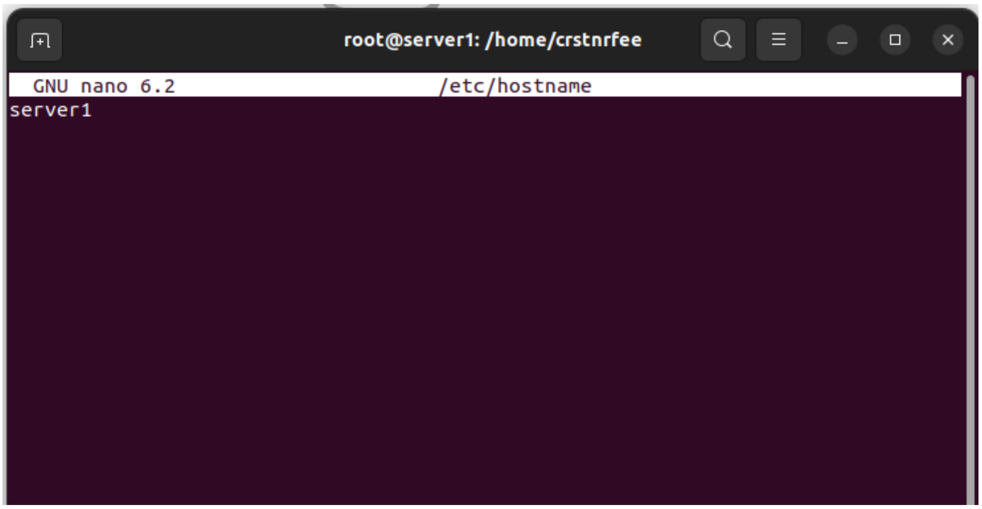
Assume that you have created the following network topology in Virtual Machines, *provide screenshots for each task*. (Note: *it is assumed that you have the prior knowledge of cloning and creating snapshots in a virtual machine*).



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.

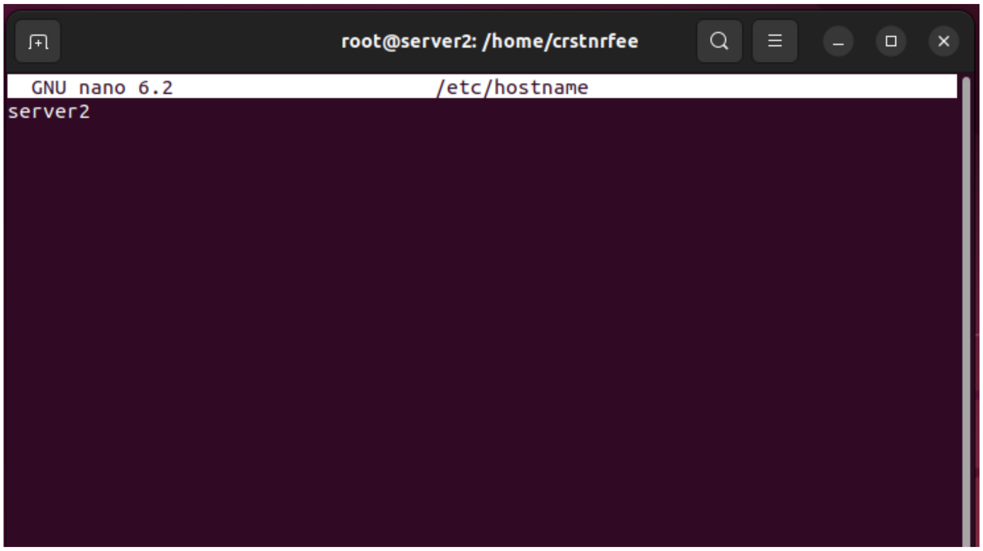
1. Change the hostname using the command *sudo nano /etc/hostname*

   1.1 Use server1 for Server 1



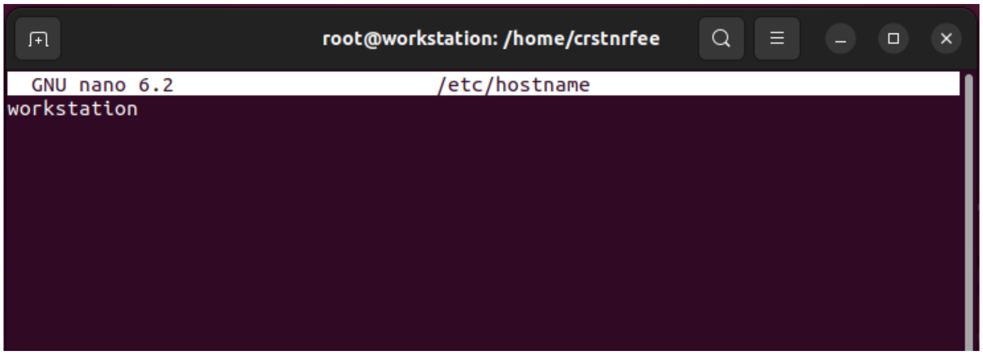
```
  GNU nano 6.2                                  /etc/hostname
server1
```

   1.2 Use server2 for Server 2



```
  GNU nano 6.2                                  /etc/hostname
server2
```

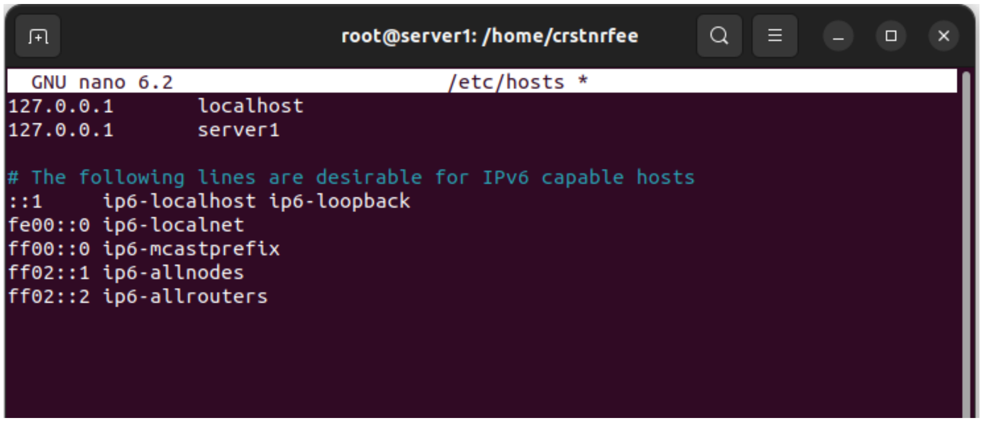
   1.3 Use workstation for the Local Machine



```
  GNU nano 6.2                                  /etc/hostname
workstation
```

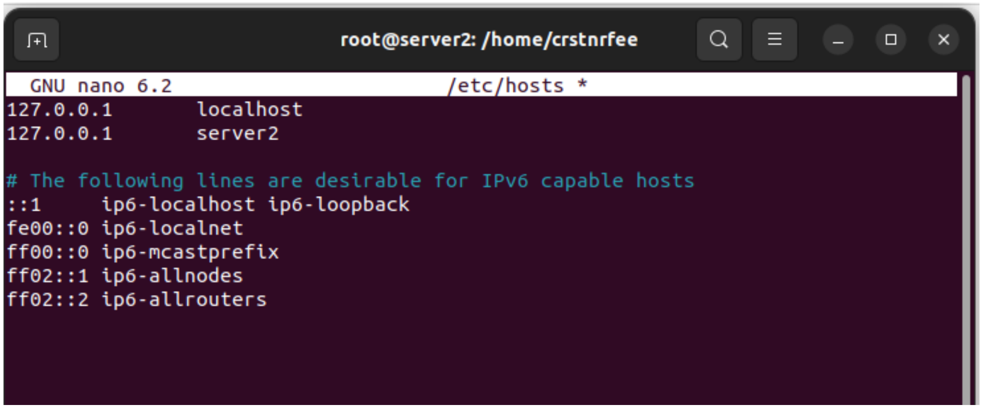
2. Edit the hosts using the command *sudo nano /etc/hosts.* Edit the second line.

2.1 Type 127.0.0.1 server 1 for Server 1



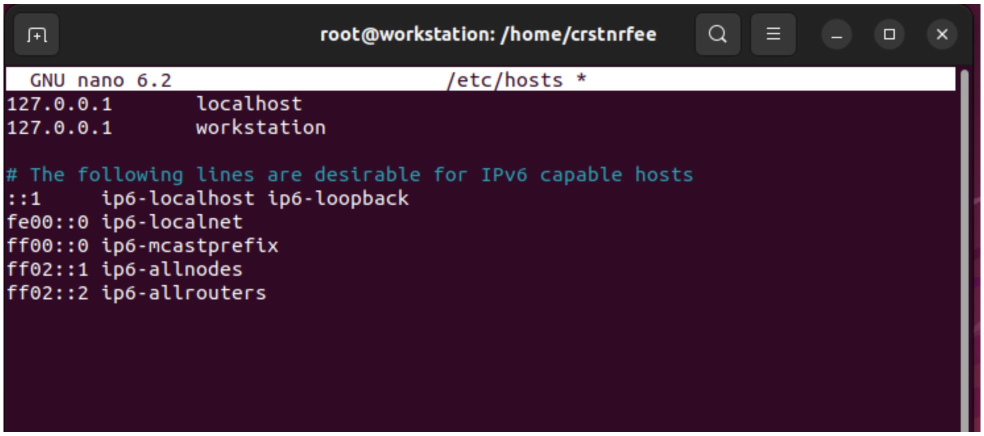
```
  GNU nano 6.2                     /etc/hosts *
127.0.0.1       localhost
127.0.0.1       server1

# 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



```
  GNU nano 6.2                     /etc/hosts *
127.0.0.1       localhost
127.0.0.1       server2

# 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
```

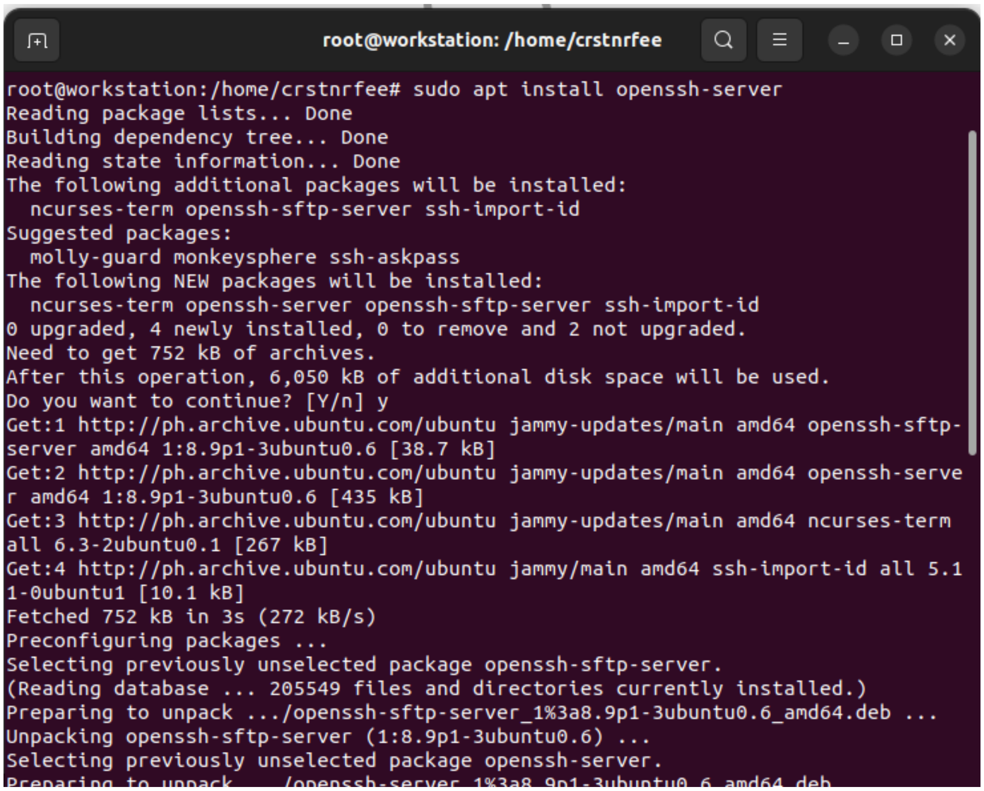
2.3 Type 127.0.0.1 workstation for the Local Machine



```
  GNU nano 6.2                     /etc/hosts *
127.0.0.1       localhost
127.0.0.1       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.
2. Install the SSH server using the command *sudo apt install openssh-server*.

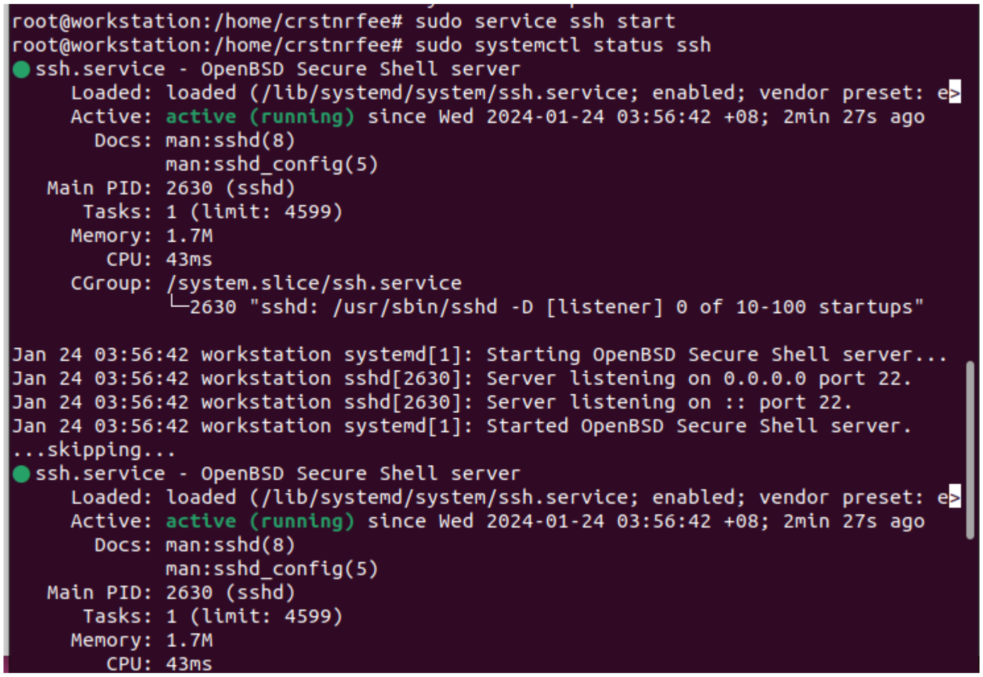


```
root@workstation:/home/crstnrfee# sudo apt install openssh-server
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
The following additional packages will be installed:
  ncurses-term openssh-sftp-server ssh-import-id
Suggested packages:
  molly-guard 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 752 kB of archives.
After this operation, 6,050 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 1:8.9p1-3ubuntu0.6 [38.7 kB]
Get:2 http://ph.archive.ubuntu.com/ubuntu jammy-updates/main amd64 openssh-serve
r amd64 1:8.9p1-3ubuntu0.6 [435 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 752 kB in 3s (272 kB/s)
Preconfiguring packages ...
Selecting previously unselected package openssh-sftp-server.
(Reading database ... 205549 files and directories currently installed.)
Preparing to unpack .../openssh-sftp-server_1%3a8.9p1-3ubuntu0.6_amd64.deb ...
Unpacking openssh-sftp-server (1:8.9p1-3ubuntu0.6) ...
Selecting previously unselected package openssh-server.
Preparing to unpack .../openssh-server_1%3a8.9p1-3ubuntu0.6_amd64.deb
```

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

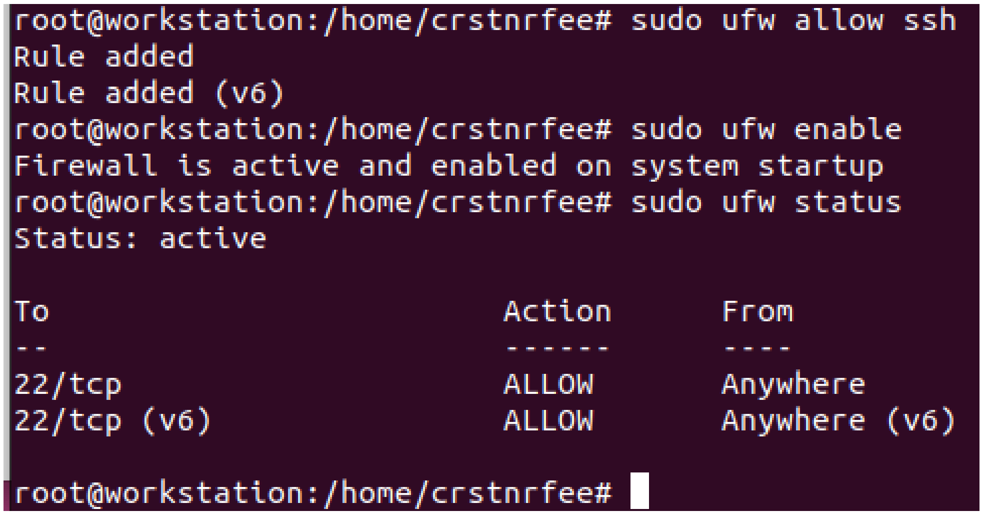
   3.1 *sudo service ssh start*

   3.2 *sudo systemctl status ssh*



```
root@workstation:/home/crstnrfee# sudo service ssh start
root@workstation:/home/crstnrfee# 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 Wed 2024-01-24 03:56:42 +08; 2min 27s ago
       Docs: man:sshd(8)
             man:sshd_config(5)
   Main PID: 2630 (sshd)
      Tasks: 1 (limit: 4599)
     Memory: 1.7M
        CPU: 43ms
     CGroup: /system.slice/ssh.service
             └─2630 "sshd: /usr/sbin/sshd -D [listener] 0 of 10-100 startups"

Jan 24 03:56:42 workstation systemd[1]: Starting OpenBSD Secure Shell server...
Jan 24 03:56:42 workstation sshd[2630]: Server listening on 0.0.0.0 port 22.
Jan 24 03:56:42 workstation sshd[2630]: Server listening on :: port 22.
Jan 24 03:56:42 workstation systemd[1]: Started OpenBSD Secure Shell server.
...skipping...
● ssh.service - OpenBSD Secure Shell server
     Loaded: loaded (/lib/systemd/system/ssh.service; enabled; vendor preset: e>
     Active: active (running) since Wed 2024-01-24 03:56:42 +08; 2min 27s ago
       Docs: man:sshd(8)
             man:sshd_config(5)
   Main PID: 2630 (sshd)
      Tasks: 1 (limit: 4599)
     Memory: 1.7M
        CPU: 43ms
```

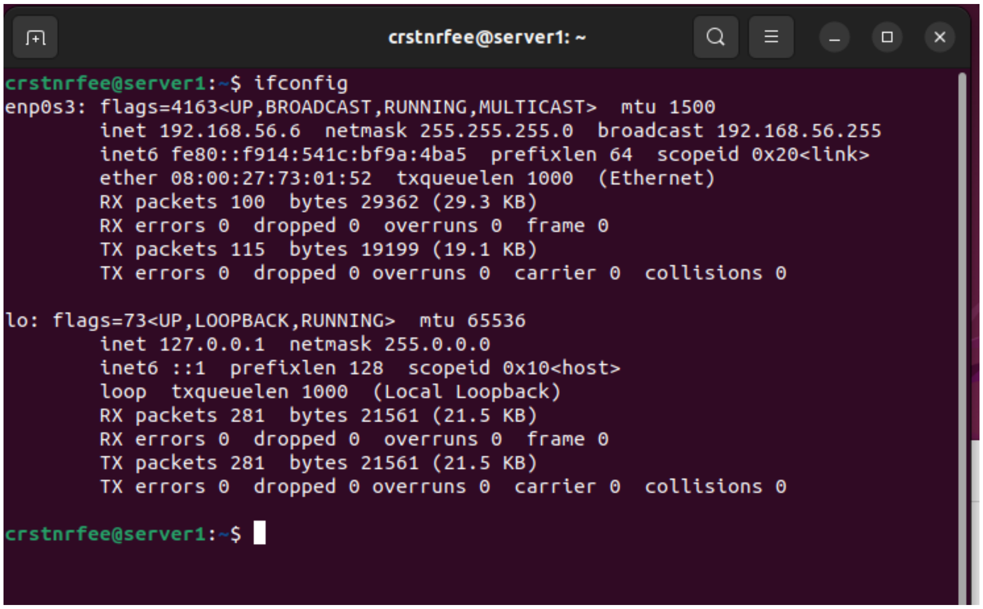
4. Configure the firewall to all port 22 by issuing the following commands:
    4.1 *sudo ufw allow ssh*
    4.2 *sudo ufw enable*
    4.3 *sudo ufw status*



```
root@workstation:/home/crstnrfee# sudo ufw allow ssh
Rule added
Rule added (v6)
root@workstation:/home/crstnrfee# sudo ufw enable
Firewall is active and enabled on system startup
root@workstation:/home/crstnrfee# sudo ufw status
Status: active

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

root@workstation:/home/crstnrfee# 
```

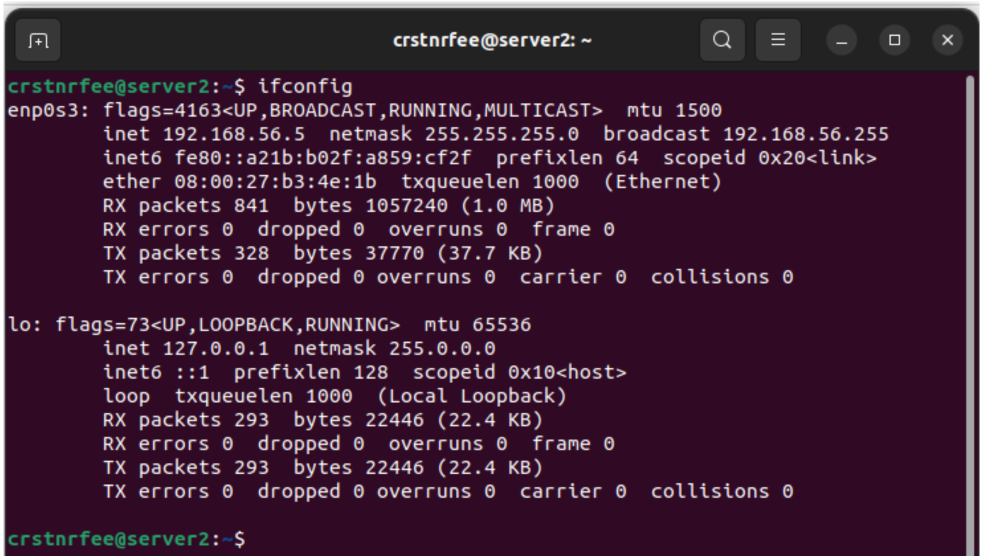
Task 3: Verify network settings on Server 1, Server 2, and Local Machine. On each device, do the following:

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: 192.168.56.6



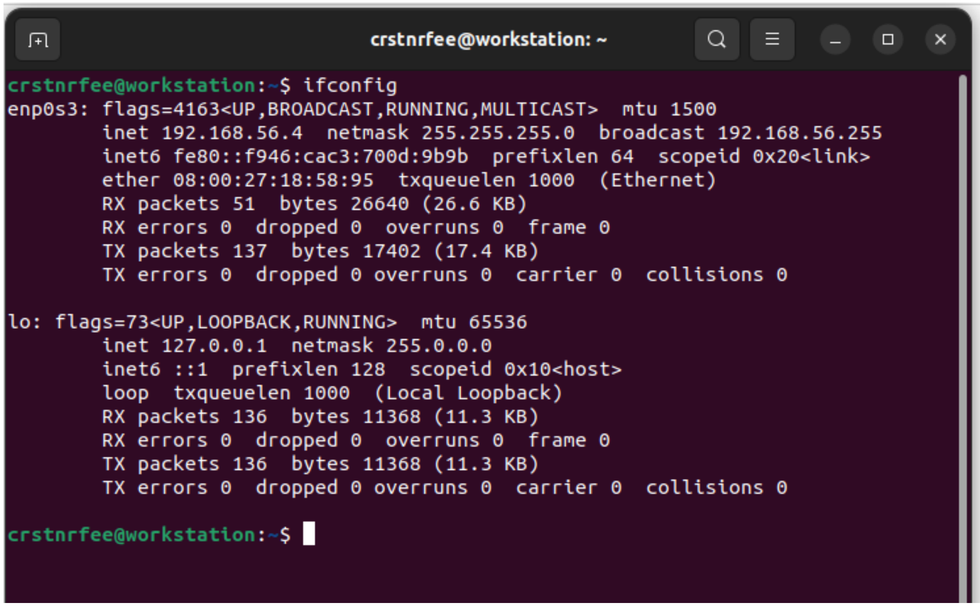
```
crstnrfee@server1:~$ ifconfig
enp0s3: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1500
        inet 192.168.56.6  netmask 255.255.255.0  broadcast 192.168.56.255
        inet6 fe80::f914:541c:bf9a:4ba5  prefixlen 64  scopeid 0x20<link>
        ether 08:00:27:73:01:52  txqueuelen 1000  (Ethernet)
        RX packets 100  bytes 29362 (29.3 KB)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 115  bytes 19199 (19.1 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 281  bytes 21561 (21.5 KB)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 281  bytes 21561 (21.5 KB)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0

crstnrfee@server1:~$ 
```

1.2 Server 2 IP address: 192.168.56.5



```
crstnrfee@server2:~$ ifconfig
enp0s3: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1500
        inet 192.168.56.5  netmask 255.255.255.0  broadcast 192.168.56.255
        inet6 fe80::a21b:b02f:a859:cf2f  prefixlen 64  scopeid 0x20<link>
        ether 08:00:27:b3:4e:1b  txqueuelen 1000  (Ethernet)
        RX packets 841  bytes 1057240 (1.0 MB)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 328  bytes 37770 (37.7 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 293  bytes 22446 (22.4 KB)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 293  bytes 22446 (22.4 KB)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0

crstnrfee@server2:~$
```

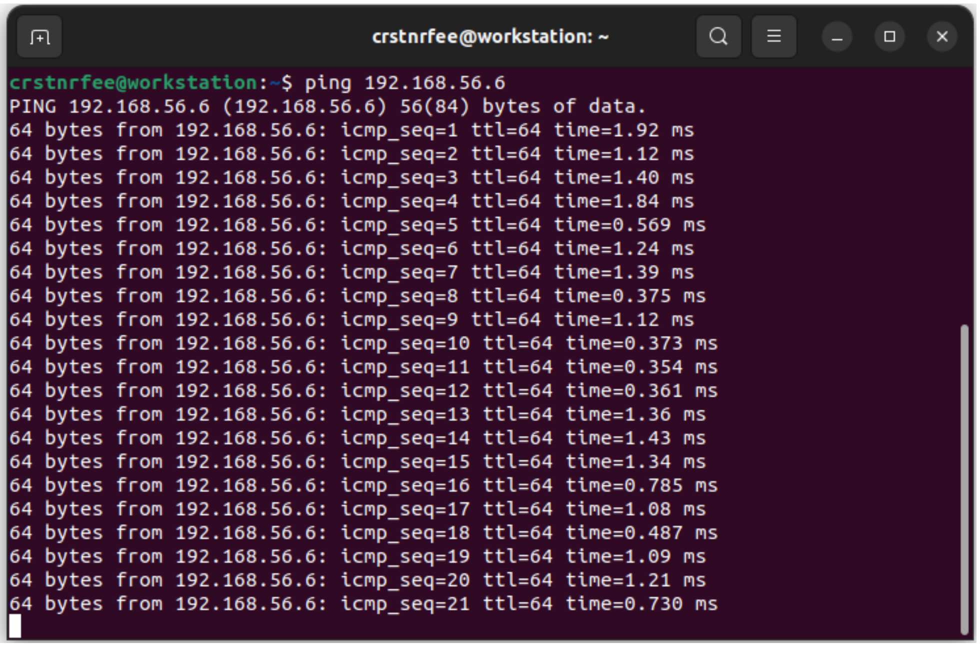
1.3 Server 3 IP address: 192.168.56.4



```
crstnrfee@workstation:~$ ifconfig
enp0s3: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1500
        inet 192.168.56.4  netmask 255.255.255.0  broadcast 192.168.56.255
        inet6 fe80::f946:cac3:700d:9b9b  prefixlen 64  scopeid 0x20<link>
        ether 08:00:27:18:58:95  txqueuelen 1000  (Ethernet)
        RX packets 51  bytes 26640 (26.6 KB)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 137  bytes 17402 (17.4 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 136  bytes 11368 (11.3 KB)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 136  bytes 11368 (11.3 KB)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0

crstnrfee@workstation:~$ 
```

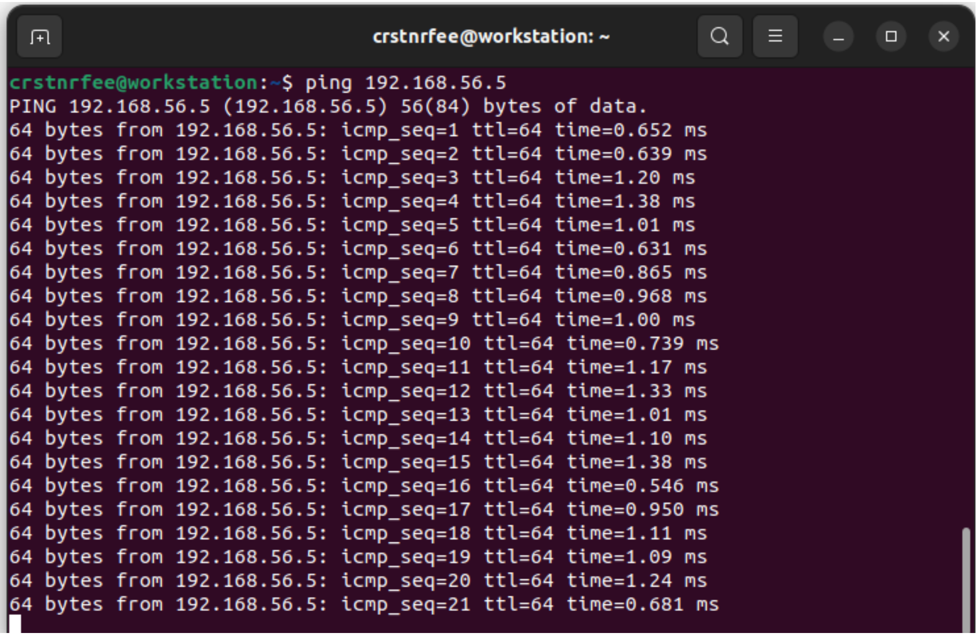
2. Make sure that they can ping each other.

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



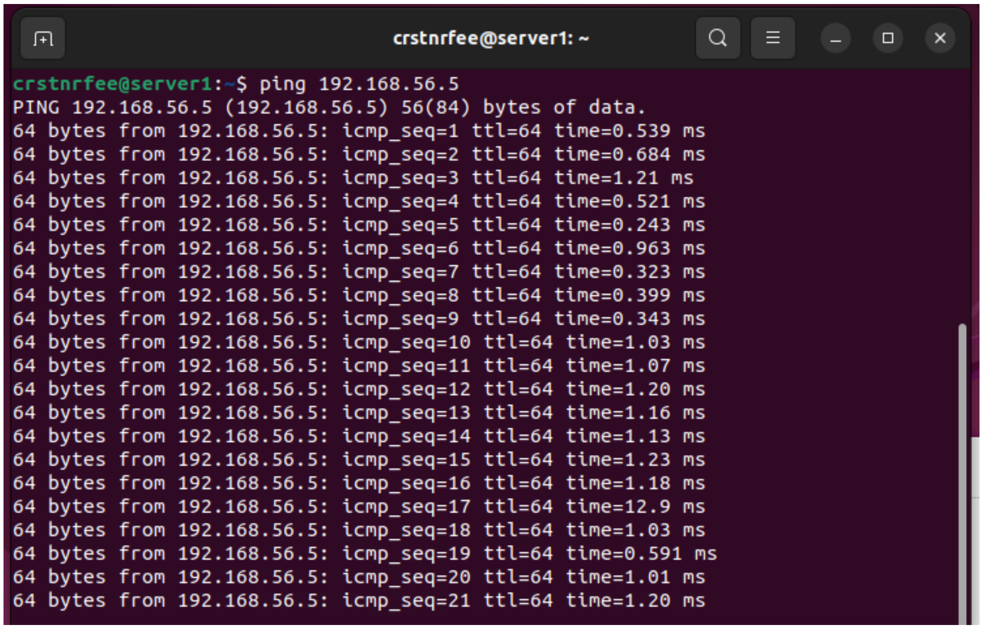
```
crstnrfee@workstation:~$ ping 192.168.56.6
PING 192.168.56.6 (192.168.56.6) 56(84) bytes of data.
64 bytes from 192.168.56.6: icmp_seq=1 ttl=64 time=1.92 ms
64 bytes from 192.168.56.6: icmp_seq=2 ttl=64 time=1.12 ms
64 bytes from 192.168.56.6: icmp_seq=3 ttl=64 time=1.40 ms
64 bytes from 192.168.56.6: icmp_seq=4 ttl=64 time=1.84 ms
64 bytes from 192.168.56.6: icmp_seq=5 ttl=64 time=0.569 ms
64 bytes from 192.168.56.6: icmp_seq=6 ttl=64 time=1.24 ms
64 bytes from 192.168.56.6: icmp_seq=7 ttl=64 time=1.39 ms
64 bytes from 192.168.56.6: icmp_seq=8 ttl=64 time=0.375 ms
64 bytes from 192.168.56.6: icmp_seq=9 ttl=64 time=1.12 ms
64 bytes from 192.168.56.6: icmp_seq=10 ttl=64 time=0.373 ms
64 bytes from 192.168.56.6: icmp_seq=11 ttl=64 time=0.354 ms
64 bytes from 192.168.56.6: icmp_seq=12 ttl=64 time=0.361 ms
64 bytes from 192.168.56.6: icmp_seq=13 ttl=64 time=1.36 ms
64 bytes from 192.168.56.6: icmp_seq=14 ttl=64 time=1.43 ms
64 bytes from 192.168.56.6: icmp_seq=15 ttl=64 time=1.34 ms
64 bytes from 192.168.56.6: icmp_seq=16 ttl=64 time=0.785 ms
64 bytes from 192.168.56.6: icmp_seq=17 ttl=64 time=1.08 ms
64 bytes from 192.168.56.6: icmp_seq=18 ttl=64 time=0.487 ms
64 bytes from 192.168.56.6: icmp_seq=19 ttl=64 time=1.09 ms
64 bytes from 192.168.56.6: icmp_seq=20 ttl=64 time=1.21 ms
64 bytes from 192.168.56.6: icmp_seq=21 ttl=64 time=0.730 ms
```

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



```
crstnrfee@workstation:~$ ping 192.168.56.5
PING 192.168.56.5 (192.168.56.5) 56(84) bytes of data.
64 bytes from 192.168.56.5: icmp_seq=1 ttl=64 time=0.652 ms
64 bytes from 192.168.56.5: icmp_seq=2 ttl=64 time=0.639 ms
64 bytes from 192.168.56.5: icmp_seq=3 ttl=64 time=1.20 ms
64 bytes from 192.168.56.5: icmp_seq=4 ttl=64 time=1.38 ms
64 bytes from 192.168.56.5: icmp_seq=5 ttl=64 time=1.01 ms
64 bytes from 192.168.56.5: icmp_seq=6 ttl=64 time=0.631 ms
64 bytes from 192.168.56.5: icmp_seq=7 ttl=64 time=0.865 ms
64 bytes from 192.168.56.5: icmp_seq=8 ttl=64 time=0.968 ms
64 bytes from 192.168.56.5: icmp_seq=9 ttl=64 time=1.00 ms
64 bytes from 192.168.56.5: icmp_seq=10 ttl=64 time=0.739 ms
64 bytes from 192.168.56.5: icmp_seq=11 ttl=64 time=1.17 ms
64 bytes from 192.168.56.5: icmp_seq=12 ttl=64 time=1.33 ms
64 bytes from 192.168.56.5: icmp_seq=13 ttl=64 time=1.01 ms
64 bytes from 192.168.56.5: icmp_seq=14 ttl=64 time=1.10 ms
64 bytes from 192.168.56.5: icmp_seq=15 ttl=64 time=1.38 ms
64 bytes from 192.168.56.5: icmp_seq=16 ttl=64 time=0.546 ms
64 bytes from 192.168.56.5: icmp_seq=17 ttl=64 time=0.950 ms
64 bytes from 192.168.56.5: icmp_seq=18 ttl=64 time=1.11 ms
64 bytes from 192.168.56.5: icmp_seq=19 ttl=64 time=1.09 ms
64 bytes from 192.168.56.5: icmp_seq=20 ttl=64 time=1.24 ms
64 bytes from 192.168.56.5: icmp_seq=21 ttl=64 time=0.681 ms
```

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



```
crstnrfee@server1:~$ ping 192.168.56.5
PING 192.168.56.5 (192.168.56.5) 56(84) bytes of data.
64 bytes from 192.168.56.5: icmp_seq=1 ttl=64 time=0.539 ms
64 bytes from 192.168.56.5: icmp_seq=2 ttl=64 time=0.684 ms
64 bytes from 192.168.56.5: icmp_seq=3 ttl=64 time=1.21 ms
64 bytes from 192.168.56.5: icmp_seq=4 ttl=64 time=0.521 ms
64 bytes from 192.168.56.5: icmp_seq=5 ttl=64 time=0.243 ms
64 bytes from 192.168.56.5: icmp_seq=6 ttl=64 time=0.963 ms
64 bytes from 192.168.56.5: icmp_seq=7 ttl=64 time=0.323 ms
64 bytes from 192.168.56.5: icmp_seq=8 ttl=64 time=0.399 ms
64 bytes from 192.168.56.5: icmp_seq=9 ttl=64 time=0.343 ms
64 bytes from 192.168.56.5: icmp_seq=10 ttl=64 time=1.03 ms
64 bytes from 192.168.56.5: icmp_seq=11 ttl=64 time=1.07 ms
64 bytes from 192.168.56.5: icmp_seq=12 ttl=64 time=1.20 ms
64 bytes from 192.168.56.5: icmp_seq=13 ttl=64 time=1.16 ms
64 bytes from 192.168.56.5: icmp_seq=14 ttl=64 time=1.13 ms
64 bytes from 192.168.56.5: icmp_seq=15 ttl=64 time=1.23 ms
64 bytes from 192.168.56.5: icmp_seq=16 ttl=64 time=1.18 ms
64 bytes from 192.168.56.5: icmp_seq=17 ttl=64 time=12.9 ms
64 bytes from 192.168.56.5: icmp_seq=18 ttl=64 time=1.03 ms
64 bytes from 192.168.56.5: icmp_seq=19 ttl=64 time=0.591 ms
64 bytes from 192.168.56.5: icmp_seq=20 ttl=64 time=1.01 ms
64 bytes from 192.168.56.5: icmp_seq=21 ttl=64 time=1.20 ms
```

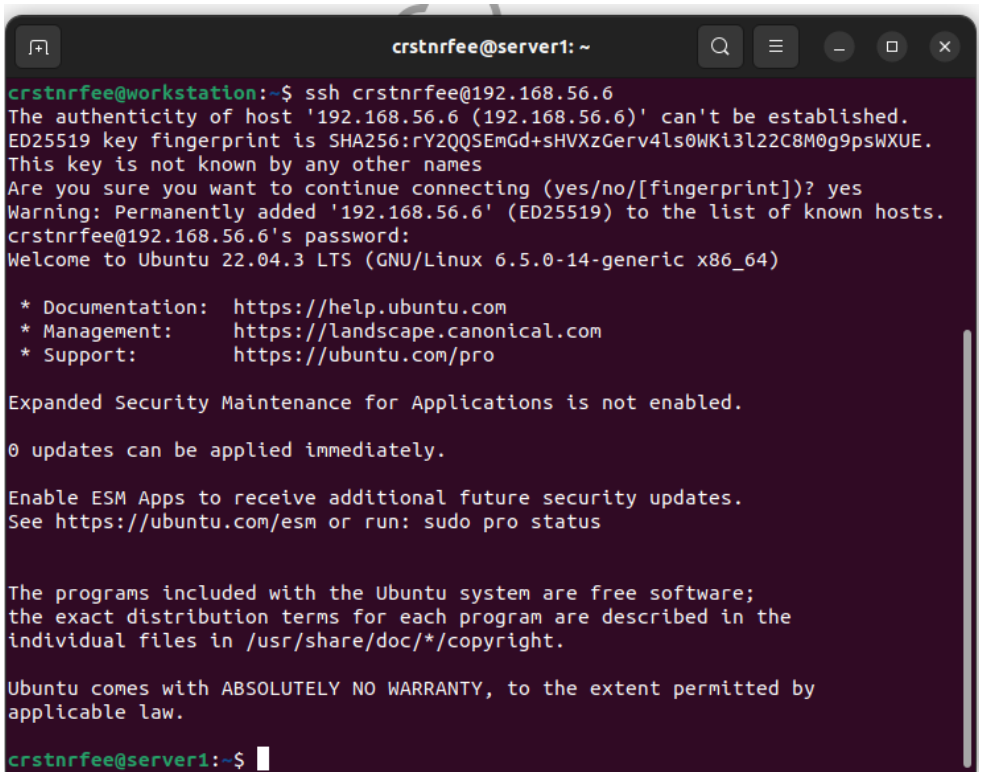
Task 4: Verify SSH connectivity on Server 1, Server 2, and Local Machine.

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

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

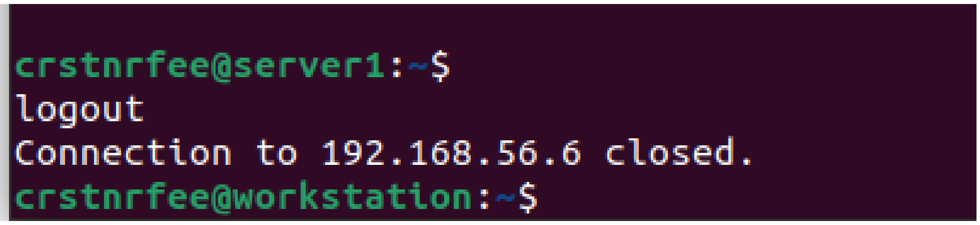
1.2 Enter the password for server 1 when prompted

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



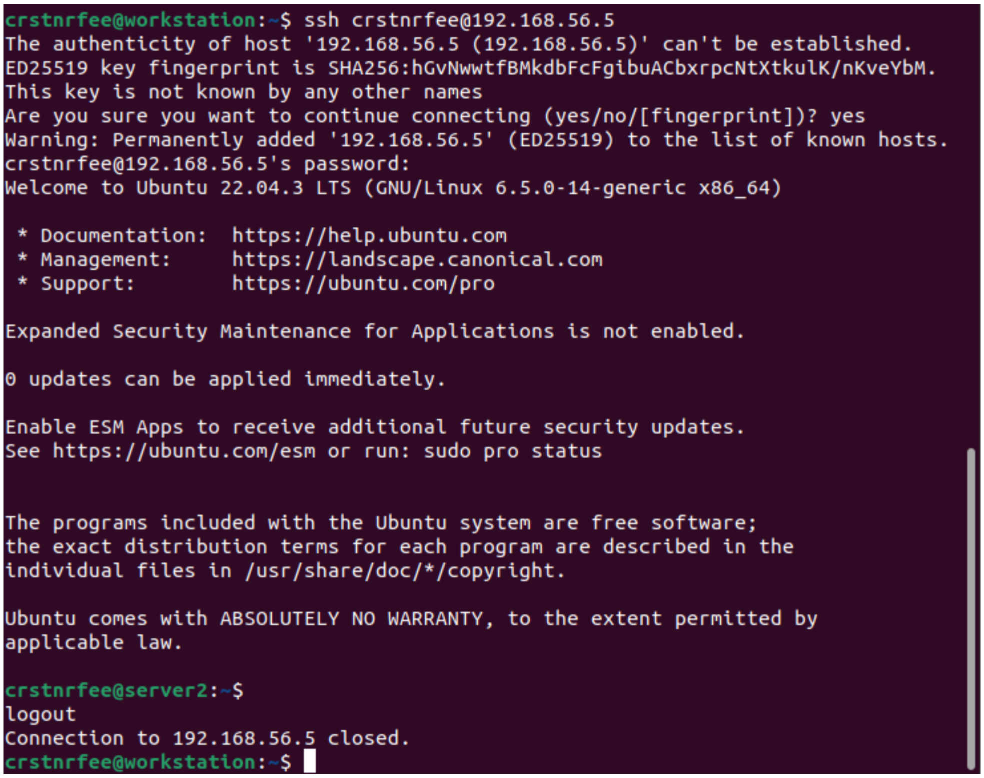
```
crstnrfee@workstation:~$ ssh crstnrfee@192.168.56.6
The authenticity of host '192.168.56.6 (192.168.56.6)' can't be established.
ED25519 key fingerprint is SHA256:rY2QQSEmGd+sHVXzGerv4ls0WKi3l22C8M0g9psWXUE.
This key is not known by any other names
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added '192.168.56.6' (ED25519) to the list of known hosts.
crstnrfee@192.168.56.6's password:
Welcome to Ubuntu 22.04.3 LTS (GNU/Linux 6.5.0-14-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.

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


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.

crstnrfee@server1:~$ 
```

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



```
crstnrfee@server1:~$
logout
Connection to 192.168.56.6 closed.
crstnrfee@workstation:~$
```

3. Do the same for Server 2.



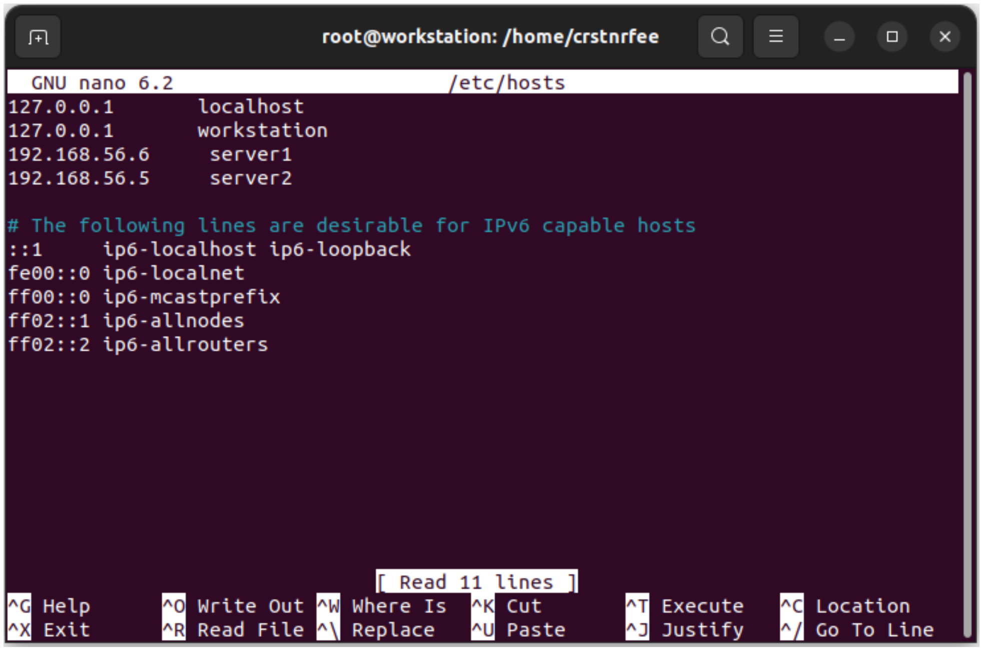
```
crstnrfee@workstation:~$ ssh crstnrfee@192.168.56.5
The authenticity of host '192.168.56.5 (192.168.56.5)' can't be established.
ED25519 key fingerprint is SHA256:hGvNwwtfBMkdbFcFgibuACbxrpcNtXtkulK/nKveYbM.
This key is not known by any other names
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added '192.168.56.5' (ED25519) to the list of known hosts.
crstnrfee@192.168.56.5's password:
Welcome to Ubuntu 22.04.3 LTS (GNU/Linux 6.5.0-14-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.

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


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.

crstnrfee@server2:~$
logout
Connection to 192.168.56.5 closed.
crstnrfee@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)

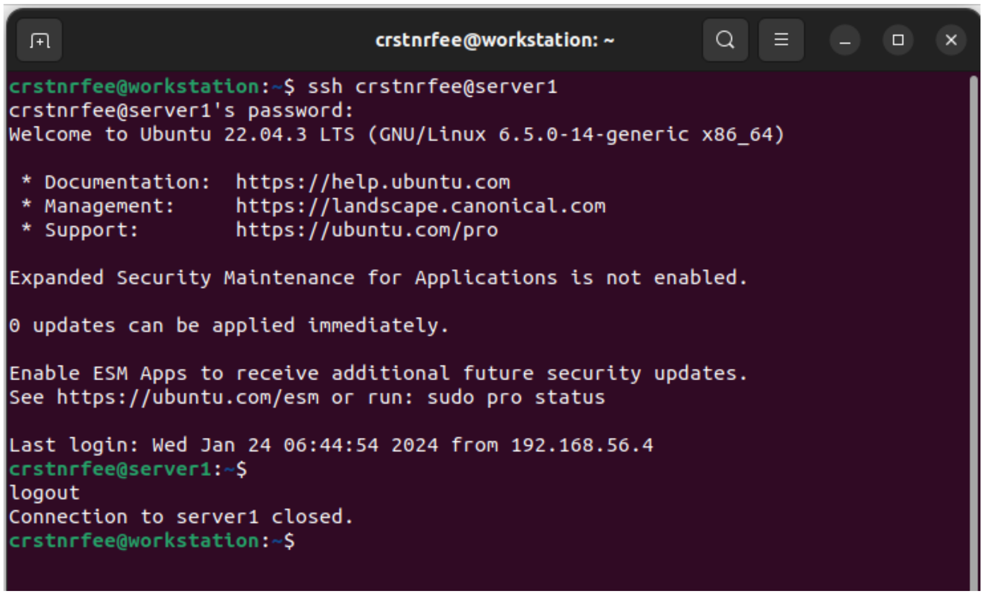
4.2 IP_address server 2 (provide the ip address of server 2 followed by the hostname)

4.3 Save the file and exit.



```
  GNU nano 6.2                     /etc/hosts
127.0.0.1       localhost
127.0.0.1       workstation
192.168.56.6     server1
192.168.56.5     server2

# 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








                              [ Read 11 lines ]
^G Help       ^O Write Out  ^W Where Is   ^K Cut        ^T Execute    ^C Location
^X Exit       ^R Read File  ^\ Replace    ^U Paste      ^J Justify    ^/ Go To Line
```

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.



```
crstnrfee@workstation:~$ ssh crstnrfee@server1
crstnrfee@server1's password:
Welcome to Ubuntu 22.04.3 LTS (GNU/Linux 6.5.0-14-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.

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

Last login: Wed Jan 24 06:44:54 2024 from 192.168.56.4
crstnrfee@server1:~$
logout
Connection to server1 closed.
crstnrfee@workstation:~$
```

Reflections:

Answer the following:

1. How are we able to use the hostname instead of IP address in SSH commands? By inputting the IP address and its corresponding hostname using “sudo nano /etc/hosts” command
2. How secured is SSH? It is secured but it depends on various factors, including its configuration, implementation, and the practices followed by users and administrators.