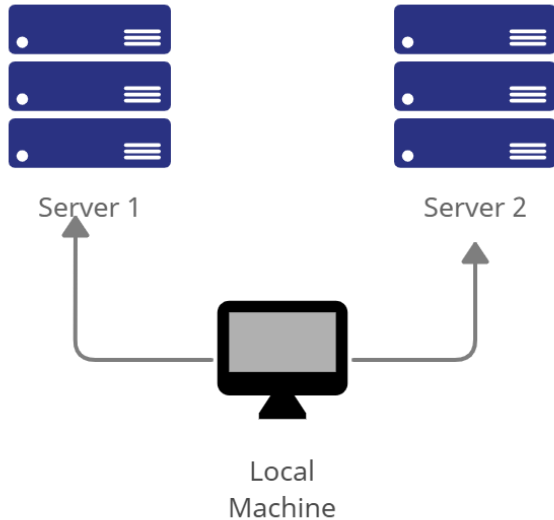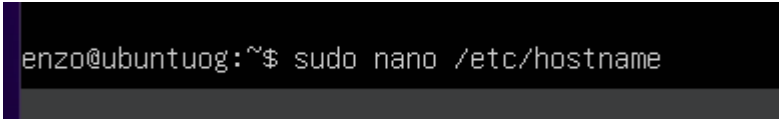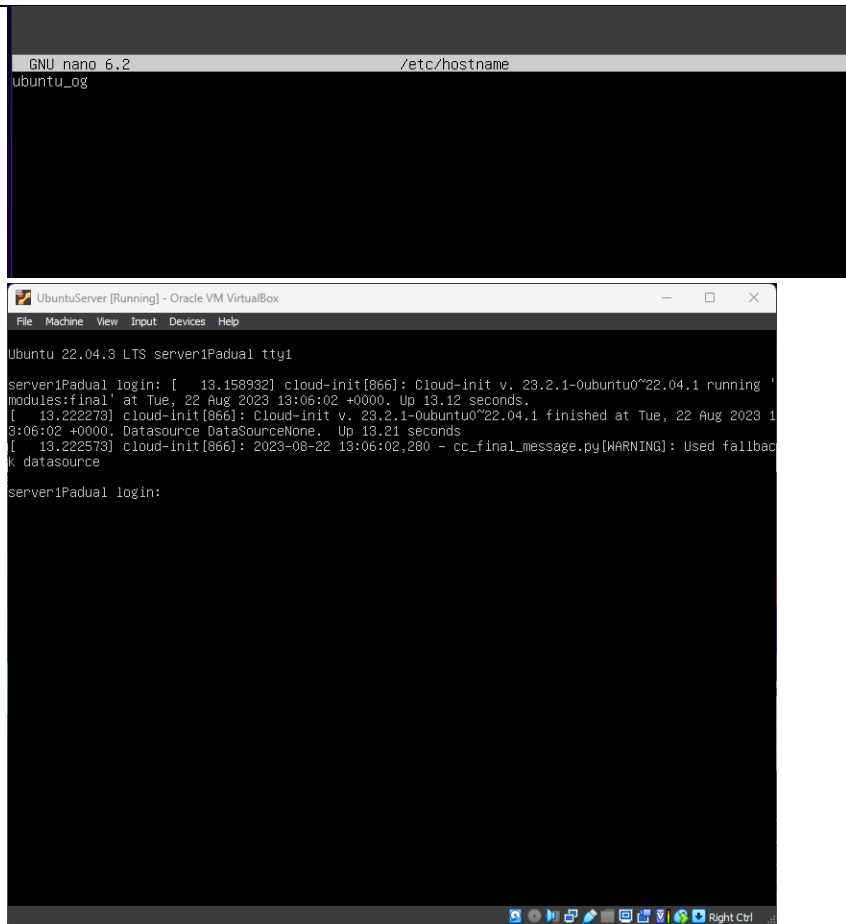| Name: Enzo Daren B. Padual	Date Performed: 08/22/2023
Course/Section: CPE31S5	Date Submitted: 08/23/2023
Instructor: Engr. Roman Richard	Semester and SY: 2nd, 2023-2024

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*
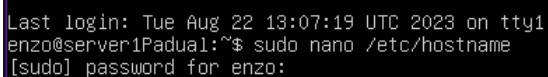
    1.1 Use server1 for Server 1

```
enzo@ubuntuog:~$ sudo nano /etc/hostname
```

```
GNU nano 6.2                          /etc/hostname
ubuntu_og
```

```
UbuntuServer [Running] - Oracle VM VirtualBox                    —    □    ✕
File  Machine  View  Input  Devices  Help

Ubuntu 22.04.3 LTS server1Padual tty1

server1Padual login: [   13.158932] cloud-init[866]: Cloud-init v. 23.2.1-0ubuntu0~22.04.1 running '
modules:final' at Tue, 22 Aug 2023 13:06:02 +0000. Up 13.12 seconds.
[   13.222273] cloud-init[866]: Cloud-init v. 23.2.1-0ubuntu0~22.04.1 finished at Tue, 22 Aug 2023 1
3:06:02 +0000. Datasource DataSourceNone.  Up 13.21 seconds
[   13.222573] cloud-init[866]: 2023-08-22 13:06:02,280 - cc_final_message.py[WARNING]: Used fallbac
k datasource

server1Padual login:
```

After restarting the virtual machine server, the hostname has been changed to what I've specifically inputted.

1.2 Use server2 for Server 2

```
Last login: Tue Aug 22 13:07:19 UTC 2023 on tty1
enzo@server1Padual:~$ sudo nano /etc/hostname
[sudo] password for enzo:
```

After restarting the virtual machine server, the hostname has been changed to what I've specifically inputted.

1.3 Use workstation for the Local Machine

```
enzopadual@ubuntu:~$ sudo nano /etc/hostname
[sudo] password for enzopadual:
```
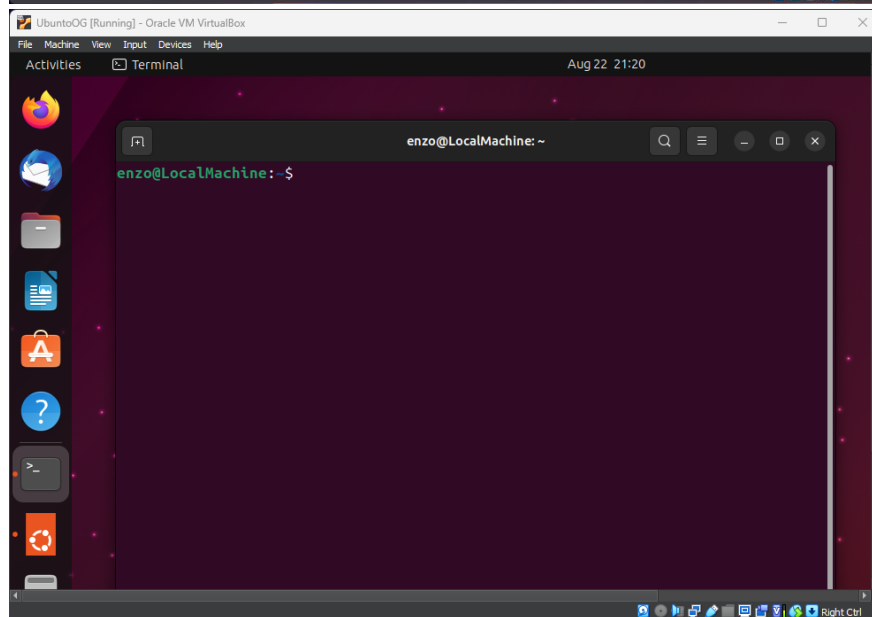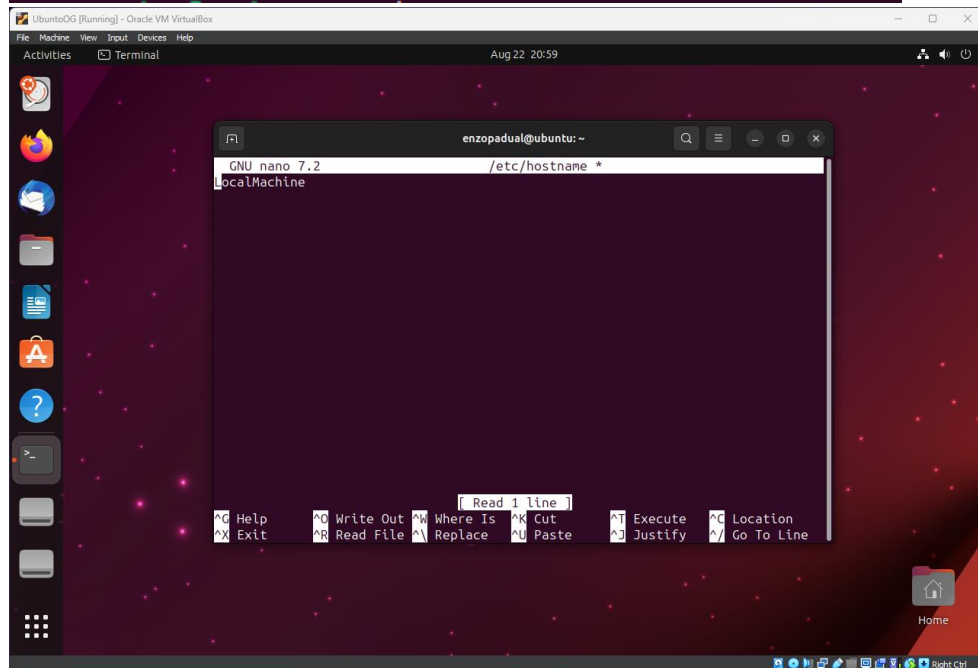

After restarting the Virtual Machine the hostname is changed to "Local Machine".
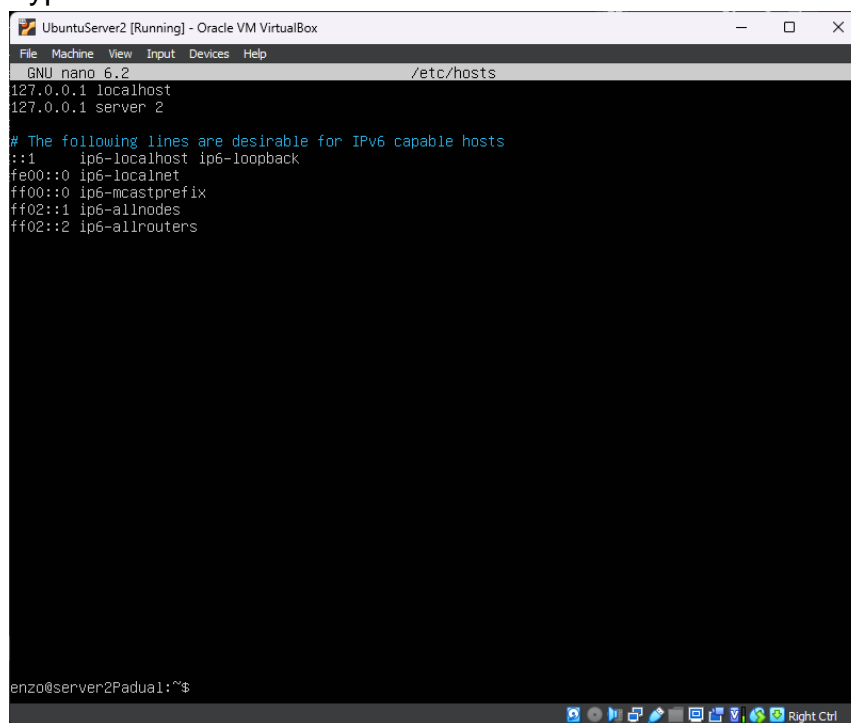
2. Edit the hosts using the command *sudo nano /etc/hosts.* Edit the second line.
    2.1 Type 127.0.0.1 server 1 for Server 1

2.2 Type 127.0.0.1 server 2 for Server 2
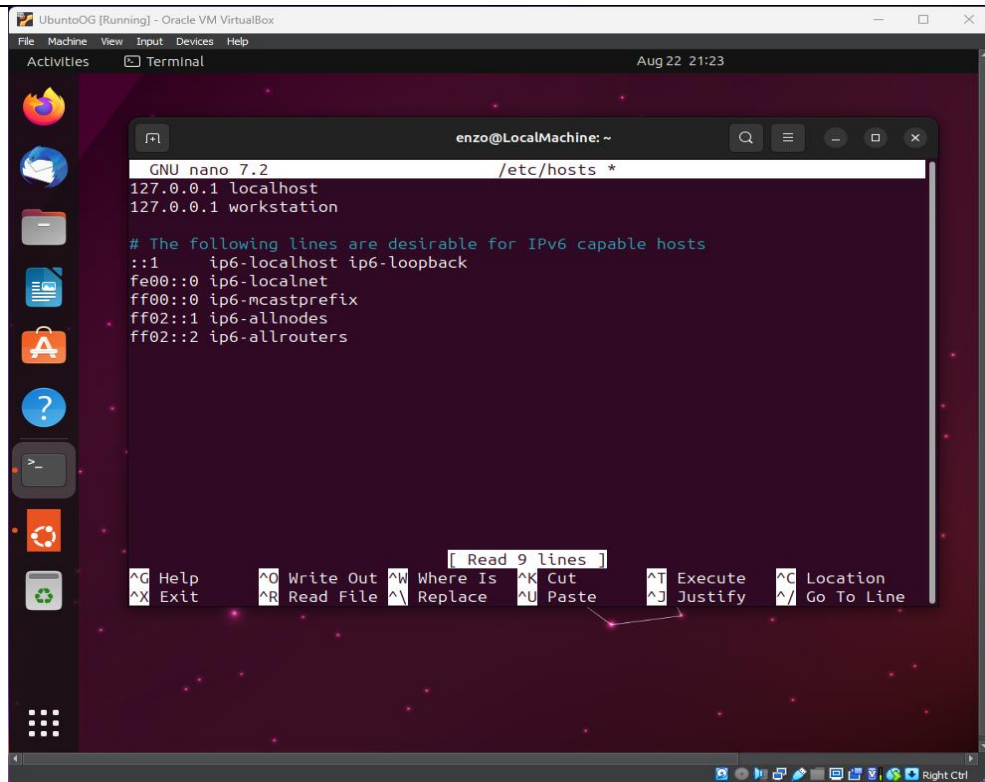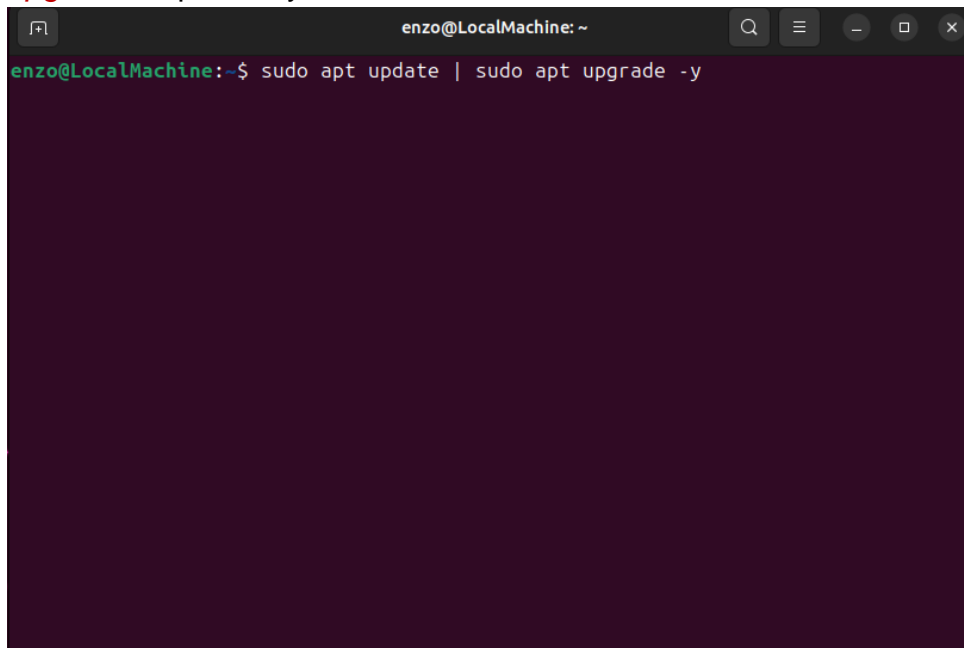
2.3 Type 127.0.0.1 workstation for the Local Machine

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.

This command is supposed to update the system, but I recently updated it, so I skipped this step.

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

I used this command to install the SSH server on all three systems. I used the word -y to directly respond y to the installation questions.

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*

```
enzo@LocalMachine: ~

Processing triggers for man-db (2.11.2-1) ...
Processing triggers for ufw (0.36.1-4.1ubuntu0.1) ...
enzo@LocalMachine:~$ sudo service ssh start
enzo@LocalMachine:~$ sudo systemctl status ssh
● ssh.service - OpenBSD Secure Shell server
     Loaded: loaded (/lib/systemd/system/ssh.service; disabled; preset: enabled)
    Drop-In: /etc/systemd/system/ssh.service.d
             └─00-socket.conf
     Active: active (running) since Tue 2023-08-22 21:33:50 PST; 1min 13s ago
 TriggeredBy: ● ssh.socket
       Docs: man:sshd(8)
             man:sshd_config(5)
    Process: 27344 ExecStartPre=/usr/sbin/sshd -t (code=exited, status=0/SUCCES>
   Main PID: 27345 (sshd)
      Tasks: 1 (limit: 5683)
     Memory: 1.7M
        CPU: 14ms
     CGroup: /system.slice/ssh.service
             └─27345 "sshd: /usr/sbin/sshd -D [listener] 0 of 10-100 startups"

Aug 22 21:33:50 LocalMachine systemd[1]: Starting ssh.service - OpenBSD Secure >
Aug 22 21:33:50 LocalMachine sshd[27345]: Server listening on :: port 22.
Aug 22 21:33:50 LocalMachine systemd[1]: Started ssh.service - OpenBSD Secure S>
lines 1-19/19 (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*

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

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

Using these commands, I upgraded the firewall on all three systems.

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**

```
enzo@server1Padual:~$ ifconfig
enp0s3: 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::a00:27ff:fecd:8600  prefixlen 64  scopeid 0x20<link>
        ether 08:00:27:cd:86:00  txqueuelen 1000  (Ethernet)
        RX packets 4  bytes 1326 (1.3 KB)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 10  bytes 1346 (1.3 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 84  bytes 6244 (6.2 KB)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 84  bytes 6244 (6.2 KB)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0
```

```
enzo@server1Padual:~$ ip ad
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default qlen 1000
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
       valid_lft forever preferred_lft forever
    inet6 ::1/128 scope host
       valid_lft forever preferred_lft forever
2: enp0s3: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP group default qlen 100
0
    link/ether 08:00:27:cd:86:00 brd ff:ff:ff:ff:ff:ff
    inet 192.168.56.102/24 metric 100 brd 192.168.56.255 scope global dynamic enp0s3
       valid_lft 557sec preferred_lft 557sec
    inet6 fe80::a00:27ff:fecd:8600/64 scope link
       valid_lft forever preferred_lft forever
```

1.2 Server 2 IP address: 192.168.56.**103**

```
enzo@server2Padual:~$ ip ad
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default qlen 1000
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
       valid_lft forever preferred_lft forever
    inet6 ::1/128 scope host
       valid_lft forever preferred_lft forever
2: enp0s3: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP group default qlen 100
0
    link/ether 08:00:27:c7:7d:bf brd ff:ff:ff:ff:ff:ff
    inet 192.168.56.103/24 metric 100 brd 192.168.56.255 scope global dynamic enp0s3
       valid_lft 566sec preferred_lft 566sec
    inet6 fe80::a00:27ff:fec7:7dbf/64 scope link
       valid_lft forever preferred_lft forever
```

1.3 Server 3 IP address: 192.168.56.**101**

```
[⊞]                         enzo@LocalMachine: ~              [Q] [≡] [—] [□] [×]

enzo@LocalMachine:~$ ip ad
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group defaul
t qlen 1000
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
       valid_lft forever preferred_lft forever
    inet6 ::1/128 scope host
       valid_lft forever preferred_lft forever
2: enp0s3: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc pfifo_fast state UP
group default qlen 1000
    link/ether 08:00:27:70:e4:c3 brd ff:ff:ff:ff:ff:ff
    inet 10.0.2.15/24 brd 10.0.2.255 scope global dynamic noprefixroute enp0s3
       valid_lft 84634sec preferred_lft 84634sec
    inet6 fe80::5f76:9a60:5c22:87c2/64 scope link noprefixroute
       valid_lft forever preferred_lft forever
3: enp0s8: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc pfifo_fast state UP
group default qlen 1000
    link/ether 08:00:27:ec:29:f0 brd ff:ff:ff:ff:ff:ff
    inet 192.168.56.101/24 brd 192.168.56.255 scope global dynamic noprefixroute
 enp0s8
       valid_lft 335sec preferred_lft 335sec
    inet6 fe80::1b9d:4874:b961:1d63/64 scope link noprefixroute
       valid_lft forever preferred_lft forever
enzo@LocalMachine:~$ █
```

```
[⊞]                         enzo@LocalMachine: ~              [Q] [≡] [—] [□] [×]

enzo@LocalMachine:~$ 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::5f76:9a60:5c22:87c2  prefixlen 64  scopeid 0x20<link>
        ether 08:00:27:70:e4:c3  txqueuelen 1000  (Ethernet)
        RX packets 812  bytes 1047778 (1.0 MB)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 317  bytes 31444 (31.4 KB)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0

enp0s8: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1500
        inet 192.168.56.101  netmask 255.255.255.0  broadcast 192.168.56.255
        inet6 fe80::1b9d:4874:b961:1d63  prefixlen 64  scopeid 0x20<link>
        ether 08:00:27:ec:29:f0  txqueuelen 1000  (Ethernet)
        RX packets 4  bytes 870 (870.0 B)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 49  bytes 6791 (6.7 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 128  bytes 11228 (11.2 KB)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 128  bytes 11228 (11.2 KB)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0

enzo@LocalMachine:~$
```

2. Make sure that they can ping each other.

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

```
enzo@LocalMachine:~$ 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=0.501 ms
64 bytes from 192.168.56.102: icmp_seq=2 ttl=64 time=0.300 ms
64 bytes from 192.168.56.102: icmp_seq=3 ttl=64 time=0.344 ms
64 bytes from 192.168.56.102: icmp_seq=4 ttl=64 time=0.286 ms
64 bytes from 192.168.56.102: icmp_seq=5 ttl=64 time=0.257 ms
64 bytes from 192.168.56.102: icmp_seq=6 ttl=64 time=0.321 ms
64 bytes from 192.168.56.102: icmp_seq=7 ttl=64 time=0.265 ms
64 bytes from 192.168.56.102: icmp_seq=8 ttl=64 time=0.254 ms
64 bytes from 192.168.56.102: icmp_seq=9 ttl=64 time=0.270 ms
64 bytes from 192.168.56.102: icmp_seq=10 ttl=64 time=0.299 ms
64 bytes from 192.168.56.102: icmp_seq=11 ttl=64 time=0.285 ms
64 bytes from 192.168.56.102: icmp_seq=12 ttl=64 time=0.304 ms
64 bytes from 192.168.56.102: icmp_seq=13 ttl=64 time=0.312 ms
38 packets transmitted, 38 received, 0% packet loss, time 44117ms
rtt min/avg/max/mdev = 0.190/0.365/2.112/0.295 ms
```

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

```
enzo@LocalMachine:~$ 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=0.365 ms
64 bytes from 192.168.56.103: icmp_seq=2 ttl=64 time=0.241 ms
64 bytes from 192.168.56.103: icmp_seq=3 ttl=64 time=0.213 ms
64 bytes from 192.168.56.103: icmp_seq=4 ttl=64 time=0.231 ms
64 bytes from 192.168.56.103: icmp_seq=5 ttl=64 time=0.340 ms
64 bytes from 192.168.56.103: icmp_seq=6 ttl=64 time=0.192 ms
64 bytes from 192.168.56.103: icmp_seq=7 ttl=64 time=0.234 ms
64 bytes from 192.168.56.103: icmp_seq=8 ttl=64 time=0.256 ms
64 bytes from 192.168.56.103: icmp_seq=9 ttl=64 time=0.298 ms
64 bytes from 192.168.56.103: icmp_seq=10 ttl=64 time=0.221 ms
64 bytes from 192.168.56.103: icmp_seq=11 ttl=64 time=0.229 ms
--- 192.168.56.103 ping statistics ---
24 packets transmitted, 24 received, 0% packet loss, time 23874ms
rtt min/avg/max/mdev = 0.192/0.257/0.416/0.052 ms
```

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

```
enzo@server1Padual:~$ 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=0.433 ms
64 bytes from 192.168.56.103: icmp_seq=2 ttl=64 time=0.264 ms
64 bytes from 192.168.56.103: icmp_seq=3 ttl=64 time=0.246 ms
64 bytes from 192.168.56.103: icmp_seq=4 ttl=64 time=0.261 ms
64 bytes from 192.168.56.103: icmp_seq=5 ttl=64 time=0.215 ms
64 bytes from 192.168.56.103: icmp_seq=6 ttl=64 time=0.283 ms
64 bytes from 192.168.56.103: icmp_seq=7 ttl=64 time=0.292 ms
64 bytes from 192.168.56.103: icmp_seq=8 ttl=64 time=0.212 ms
64 bytes from 192.168.56.103: icmp_seq=9 ttl=64 time=0.248 ms
64 bytes from 192.168.56.103: icmp_seq=10 ttl=64 time=0.295 ms
64 bytes from 192.168.56.103: icmp_seq=11 ttl=64 time=0.253 ms
64 bytes from 192.168.56.103: icmp_seq=12 ttl=64 time=0.268 ms
64 bytes from 192.168.56.103: icmp_seq=13 ttl=64 time=0.283 ms
64 bytes from 192.168.56.103: icmp_seq=14 ttl=64 time=0.319 ms
64 bytes from 192.168.56.103: icmp_seq=15 ttl=64 time=0.267 ms
64 bytes from 192.168.56.103: icmp_seq=16 ttl=64 time=0.266 ms
64 bytes from 192.168.56.103: icmp_seq=17 ttl=64 time=0.279 ms
64 bytes from 192.168.56.103: icmp_seq=18 ttl=64 time=0.241 ms
64 bytes from 192.168.56.103: icmp_seq=19 ttl=64 time=0.323 ms
64 bytes from 192.168.56.103: icmp_seq=20 ttl=64 time=0.264 ms
^C
--- 192.168.56.103 ping statistics ---
20 packets transmitted, 20 received, 0% packet loss, time 20174ms
rtt min/avg/max/mdev = 0.212/0.275/0.433/0.045 ms
```

All three tests were successful, and all three servers are now accessible.

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*

```
enzo@LocalMachine:~$ ssh enzo@192.168.56.102
The authenticity of host '192.168.56.102 (192.168.56.102)' can't be established.
ED25519 key fingerprint is SHA256:ah0RQEnWDbiuxoGDvhXHOV3Ja8u6i3wP0xqdWkhRUho.
This key is not known by any other names
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added '192.168.56.102' (ED25519) to the list of known hosts.
enzo@192.168.56.102's password:
Welcome to Ubuntu 22.04.3 LTS (GNU/Linux 5.15.0-79-generic x86_64)

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

  System information as of Tue Aug 22 03:01:05 PM UTC 2023

  System load:  0.1826171875      Processes:             112
  Usage of /:   45.8% of 11.21GB  Users logged in:       1
  Memory usage: 6%                IPv4 address for enp0s3: 192.168.56.102
  Swap usage:   0%


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
```

```
 Usage of /:   45.8% of 11.21GB   Users logged in:        1
 Memory usage: 6%                  IPv4 address for enp0s3: 192.168.56
 Swap usage:   0%


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: Tue Aug 22 14:48:42 2023
enzo@server1Padual:~$ ▯
```

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

```
enzo@server1Padual:~$ logout
Connection to 192.168.56.102 closed.
```

3. Do the same for Server 2.

```
enzo@LocalMachine:~$ ssh enzo@192.168.56.103
The authenticity of host '192.168.56.103 (192.168.56.103)' can't be established.
ED25519 key fingerprint is SHA256:8ljDSgJUfpuiYpuzH599h+RnOTrmEfNeLZS60fwIh8o.
This key is not known by any other names
Are you sure you want to continue connecting (yes/no/[fingerprint])? y
Please type 'yes', 'no' or the fingerprint: yes
Warning: Permanently added '192.168.56.103' (ED25519) to the list of known hosts.
enzo@192.168.56.103's password:
Welcome to Ubuntu 22.04.3 LTS (GNU/Linux 5.15.0-79-generic x86_64)

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

  System information as of Tue Aug 22 03:05:34 PM UTC 2023

  System load:  0.0               Processes:             112
  Usage of /:   25.5% of 11.21GB  Users logged in:       1
  Memory usage: 4%                IPv4 address for enp0s3: 192.168.56.103
  Swap usage:   0%
```

```
Expanded Security Maintenance for Applications is not enabled.

12 updates 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



Last login: Tue Aug 22 14:53:27 2023
enzo@server2Padual:~$
```

```
enzo@server2Padual:~$ logout
Connection to 192.168.56.103 closed.
```

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.
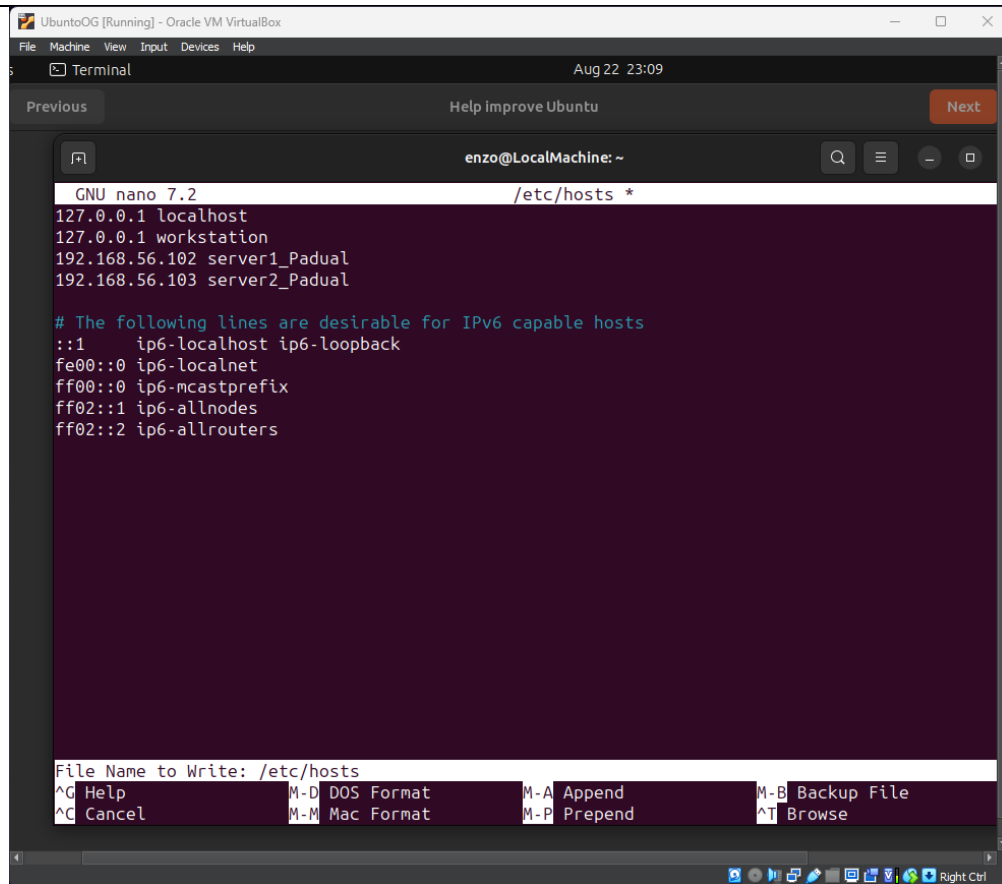
The purpose of this editing is to access the servers just by typing the server's name because the ip is already associated with the server name.

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.

```
enzo@LocalMachine:~$ ssh enzo@server1_Padual
The authenticity of host 'server1_padual (192.168.56.102)' can't be established.
ED25519 key fingerprint is SHA256:ah0RQEnWDbiuxoGDvhXHOV3Ja8u6i3wP0xqdWkhRUho.
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 'server1_padual' (ED25519) to the list of known hosts.
enzo@server1_padual's password:
Welcome to Ubuntu 22.04.3 LTS (GNU/Linux 5.15.0-79-generic x86_64)

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

  System information as of Tue Aug 22 03:10:36 PM UTC 2023

  System load:  0.15869140625     Processes:              114
  Usage of /:   45.8% of 11.21GB  Users logged in:        1
  Memory usage: 6%                IPv4 address for enp0s3: 192.168.56.102
  Swap usage:   0%


Expanded Security Maintenance for Applications is not enabled.
```

UbuntoOG [Running] - Oracle VM VirtualBox

File   Machine   View   Input   Devices   Help

Terminal   Aug 22 23:11

Previous   Help improve Ubuntu   Next

enzo@server1Padual: ~

```
ED25519 key fingerprint is SHA256:ah0RQEnWDbiuxoGDvhXHOV3Ja8u6i3wP0xqdWkhRUho.
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 'server1_padual' (ED25519) to the list of known hosts.
enzo@server1_padual's password:
Welcome to Ubuntu 22.04.3 LTS (GNU/Linux 5.15.0-79-generic x86_64)

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

  System information as of Tue Aug 22 03:10:36 PM UTC 2023

  System load:  0.15869140625     Processes:              114
  Usage of /:   45.8% of 11.21GB  Users logged in:        1
  Memory usage: 6%                IPv4 address for enp0s3: 192.168.56.102
  Swap usage:   0%


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: Tue Aug 22 15:01:06 2023 from 192.168.56.101
enzo@server1Padual:~$
```
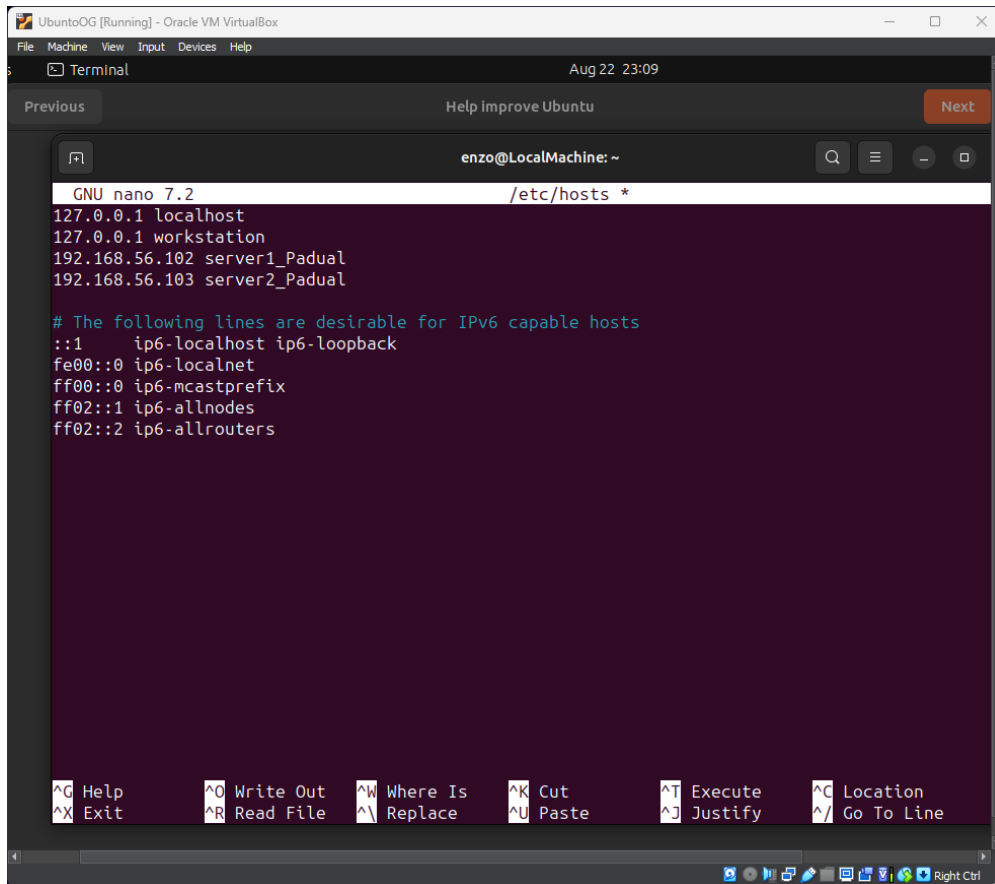
Right Ctrl

```
enzo@LocalMachine:~$ ssh enzo@server2_Padual
The authenticity of host 'server2_padual (192.168.56.103)' can't be established.
ED25519 key fingerprint is SHA256:8ljDSgJUfpuiYpuzH599h+RnOTrmEfNeLZS60fwIh8o.
This host key is known by the following other names/addresses:
    ~/.ssh/known_hosts:4: [hashed name]
Are you sure you want to continue connecting (yes/no/[fingerprint])? ^[[Ayes
Please type 'yes', 'no' or the fingerprint: yes
Warning: Permanently added 'server2_padual' (ED25519) to the list of known hosts.
enzo@server2_padual's password:
Welcome to Ubuntu 22.04.3 LTS (GNU/Linux 5.15.0-79-generic x86_64)

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

  System information as of Tue Aug 22 03:12:18 PM UTC 2023

  System load:  0.00146484375      Processes:             114
  Usage of /:   25.5% of 11.21GB   Users logged in:       1
  Memory usage: 4%                 IPv4 address for enp0s3: 192.168.56.103
  Swap usage:   0%
```

```
  Swap usage:   0%


Expanded Security Maintenance for Applications is not enabled.

12 updates 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


Last login: Tue Aug 22 15:05:34 2023 from 192.168.56.101
enzo@server2Padual:~$
```

I was now able to access the servers' terminals and update their settings and configurations from the original server by using their names (server1 & server2).

Reflections:

Answer the following:

1. **How are we able to use the hostname instead of IP address in SSH commands?**

   When you use a hostname in an SSH command, your computer sends a DNS query to a DNS server in order to obtain the IP address associated with that hostname. Following that, the DNS server provides the appropriate IP address, allowing your computer to launch an SSH connection using the specified IP address.

   The efficacy of this technique is contingent upon the presence of a functional DNS infrastructure encompassing DNS servers endowed with the capability to decipher hostnames into corresponding IP addresses. DNS resolution not only facilitates the

utilization of familiar and meaningful hostnames for linking with remote servers, but it also streamlines the management and recollection of connections.

Local Hostname Resolution:

In conjunction with DNS, operating systems frequently uphold a localized mechanism for hostname resolution. One prevalent approach entails employing the /etc/hosts file on Unix-like systems (including Linux) and the analogous "hosts" file on Windows. This file encompasses a roster of mappings between hostnames and IP addresses. Subsequently, your computer cross-references this file prior to executing a DNS query. Should the hostname be found within this file, the associated IP address is harnessed for the SSH connection.

This mode of local hostname resolution proves advantageous, especially within expedited and compact setups. It is apt for preserving an uncomplicated index of hostname-to-IP mappings sans reliance on external DNS servers.

In summation, the viability of substituting IP addresses with hostnames in SSH commands is predicated upon the orchestration of DNS resolution or local hostname resolution mechanisms. These mechanisms facilitate the conversion of human-readable hostnames into IP addresses usable by the computer to instantiate the SSH connection. While DNS represents the more extensively adopted methodology essential for internet-scale hostname resolution, local hostname resolution offers a more straightforward avenue suitable for modest networks and localized configurations.

2. **How secure is SSH?**

SSH (Secure Shell) is a secure protocol widely used for remote access and data transfer. It ensures security through encryption, using robust algorithms to safeguard data during transmission. Both password-based and public key authentication are supported by SSH, with the latter being the most secure. The protocol establishes secure connections using a key exchange procedure akin to Diffie-Hellman. Host key verification prevents man-in-the-middle attacks by verifying server authenticity. SSH supports port forwarding, enabling secure tunneling of network traffic. It provides detailed audit trails for monitoring login attempts and authentication methods. Despite its security, weak passwords and misconfigured servers can pose risks. Regular updates, strong authentication practices, and proper configuration are essential for SSH security.