NET100-M2-1-Lab-Configure NICs

Due No due date   **Points** 10   **Questions** 10   **Time Limit** None
Allowed Attempts 3

Instructions

Abstract

Networking begins on the physical layer. The component that accomplishes this is the Network Interface Card. The NIC will contain some necessary information, for example, the Media Access Control address (MAC). Using configuration files, the rest of the information necessary to establish a full connection will be established.

We will explore the basics of how to configure a NIC, as well as the hostname and hosts file to establish basic network connectivity.

Learning Objective(s)

By the end of this lab, you should be able to:

> EO1: Configure a Network Interface Card (NIC)

> EO2: Configure hostname

> EO3: Configure hosts file

System Requirements & Configuration

System Requirements

Your laptop and Cyber Range. This lab uses the NET100-M1-1-Cyber Range-Configure NICs lab environment in Cyber Range. These will be Lubuntu 16.04 Virtual Machines.

If you are doing this on your own, Ubuntu, Lubuntu or Xubuntu 16.04, or later will also work.
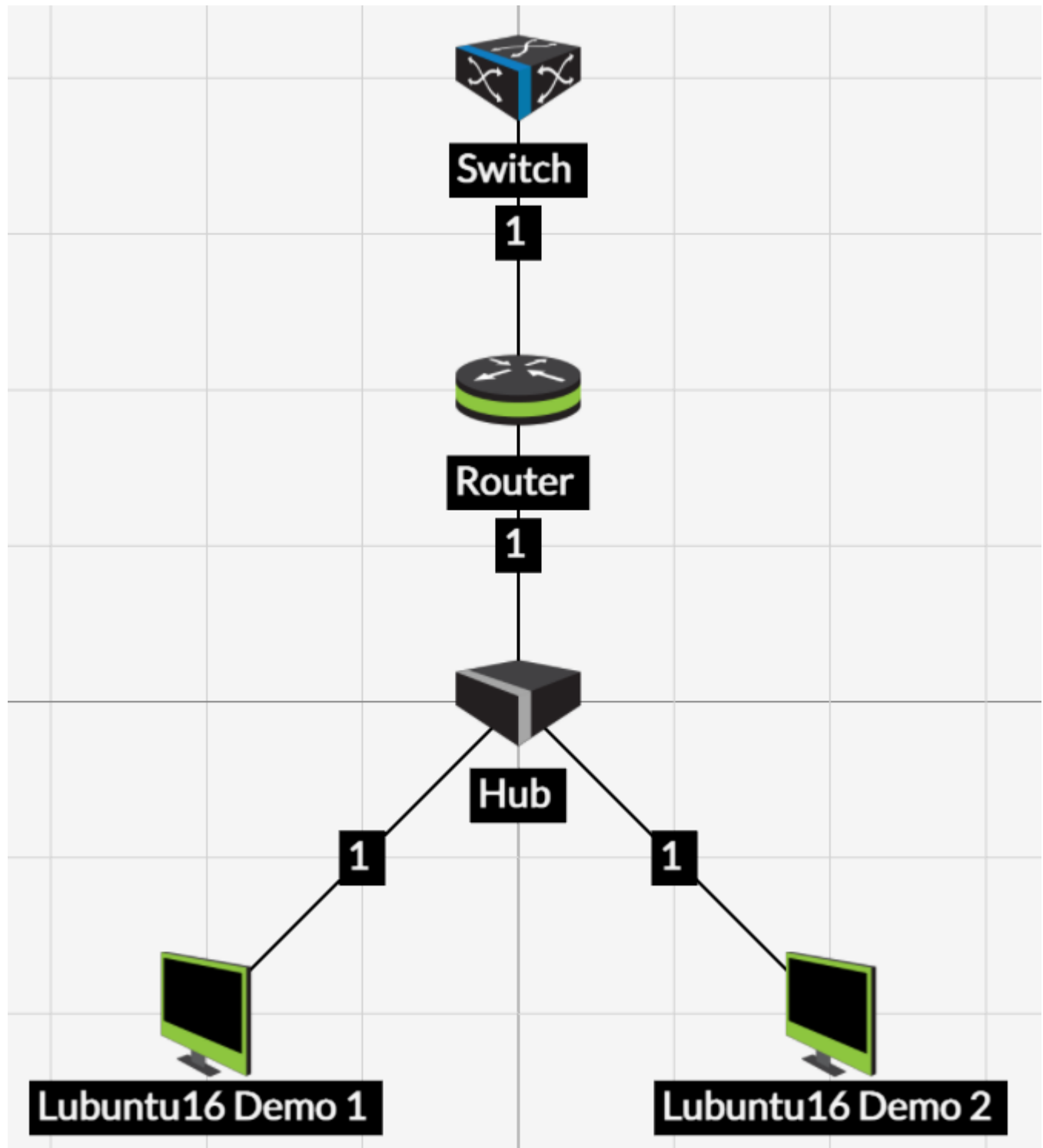
Network Requirements

Internet connectivity should be available.

Procedure – Detailed Lab Steps

Chat

Setup

Log into the NET100-M1-1-Cyber Range-Configure NICs in Canvas. You should see the following environment:

Ensure you have started the environment and that all of the machines are green, as shown in the diagram above.

The VyOS router is only to route traffic externally through the switch. We will wor' future labs.

Chat

For each Lubuntu machine:

      Username: **student**

      Password: **Password1!**

Lab Execution

Basic Networking

Double click on Lubuntu16 Demo 1 and log in with the credentials listed above. Open a Terminal, by clicking on the green Terminal icon in the tray at the bottom of the window.

Once on the command line, issue the following commands:

      `hostname`

      `ifconfig`

The `hostname` command will display hostname for the computer. Note that depending on the configuration, you may also see **student@<hostname>** for the prompt.

The **hostname** is the name given to the local computer and should be unique on the network. The hostname becomes part of the **Fully Qualified Domain Name**.

The `ifconfig` command displays the information for the interface cards present. This includes the IP Address, Media Access Control (MAC) address and other information for network interactions. Notice that the IP address is not currently present.

The following screenshot is for demonstration only and your information may be different.

Chat

The **Media Access Control address (MAC)** is also known as the **HWaddr (Hardware address)**. The name of the NIC is *enp4s1* in this example, but yours may be different. Look at the `ifconfig` to get this information (see the screenshot above). The **hostname** is *demo1* and can be seen in the prompt, **student@demo1**.

You can get similar information using the `ip addr` command.

Here you can see the loopback defined as device 1 and the NIC as device 2 and designated *enp4s1*. Again, notice that there is no IPv4 address assigned to *enp4s1*. If you look at the loopback, you can see it is assigned **127.0.0.1/8** as indicated on the line starting with **inet** above.

You should be able to issue the following command and get an active response.
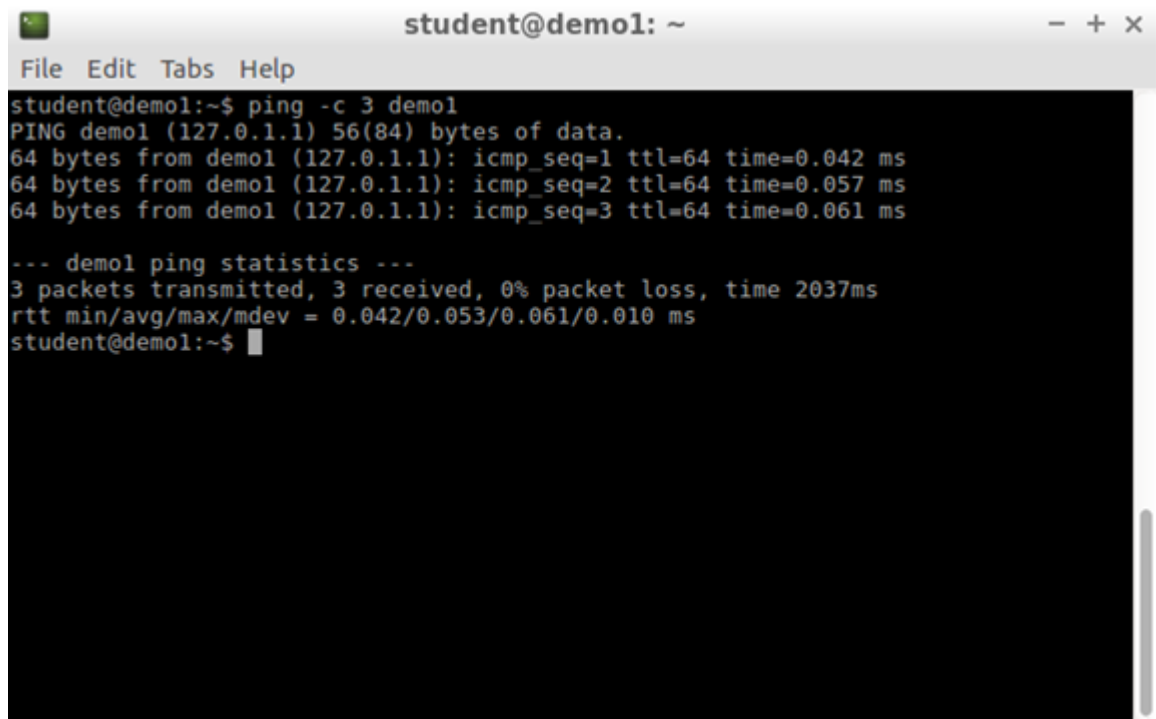
```
ping -c 3 <hostname>
```

Substitute the hostname for **<hostname>** in the above command, for example `ping -c 3 demo1`. If you receive an error, for example **Name or service not known**, verify that you sp   Chat

correctly, including capitalization. If it is correct, you may need to see the troubleshooting section in the appendix.

If it is successful, you should see something like the following:

```
student@demo1: ~                                            – + ×
File  Edit  Tabs  Help
student@demo1:~$ ping -c 3 demo1
PING demo1 (127.0.1.1) 56(84) bytes of data.
64 bytes from demo1 (127.0.1.1): icmp_seq=1 ttl=64 time=0.042 ms
64 bytes from demo1 (127.0.1.1): icmp_seq=2 ttl=64 time=0.057 ms
64 bytes from demo1 (127.0.1.1): icmp_seq=3 ttl=64 time=0.061 ms

--- demo1 ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 2037ms
rtt min/avg/max/mdev = 0.042/0.053/0.061/0.010 ms
student@demo1:~$
```

The `-c 3` , tells ping to issue exactly 3 pings. If you leave off the `-c 3` , you will continue to `ping` indefinitely. To stop the pinging, use the key combination `ctrl-c` to cancel the command.

The ping command is used to verify connectivity between two machines.

When you **ping** the hostname of the current machine, it should resolve to the **localhost**, a special designate for the internal connection. Notice the IPv4 address that got resolved is **127.0.1.1**. The technical definition of the localhost is any IPv4 address in the range **127.0.0.0** through **127.255.255.255**. Normally, **localhost** is defined as **127.0.0.1**. This address is also called the **loopback**. Any communications on **localhost** (**loopback**) will not leave the computer it is on. It is not routable externally to the machine.

Now try this command:

```
ping -c 3 localhost
```

▾ Did it ping successfully? Did it ping the same address as before? [Click to expand the answer]

You can also ping the IPv4 address directly:

```
ping -c 3 127.0.0.1
```

Chat

```
ping -c 3 127.0.1.1

ping -c 3 127.1.1.1
```

Configure IPv4 address

Lubuntu uses Network Manager to manage the IP configurations. You can access a Graphical-like User Interface with the command `nmtui` . This acts like a Graphical interface, but you cannot use your mouse. Instead, you will use the arrow keys, tab key and the enter key to access the functions. The tab key, in particular, is useful to move quickly down the choices on a screen. From the terminal window type `nmtui` and hit enter. You will be presented with the following menu.

```
┌─┤ NetworkManager TUI ├─┐

  Please select an option

  Edit a connection
  Activate a connection
  Set system hostname

  Quit

                    <OK>

```
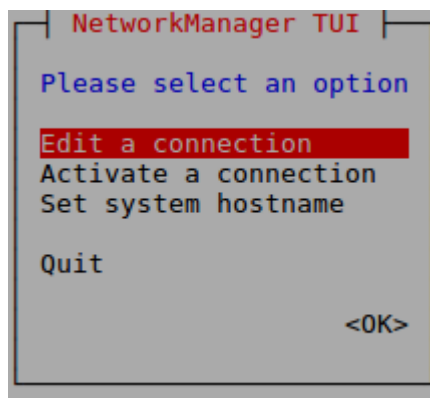
Arrow down to **Edit a connection**, select the **Wired connection 1** and then hit enter (or arrow right to highlight **Edit**). Hit the enter key to confirm.

```
  Ethernet                 ↑      <Add>
     Wired connection 1

                                  <Edit...>

                                  <Delete>




                          ↓      <Quit>
```
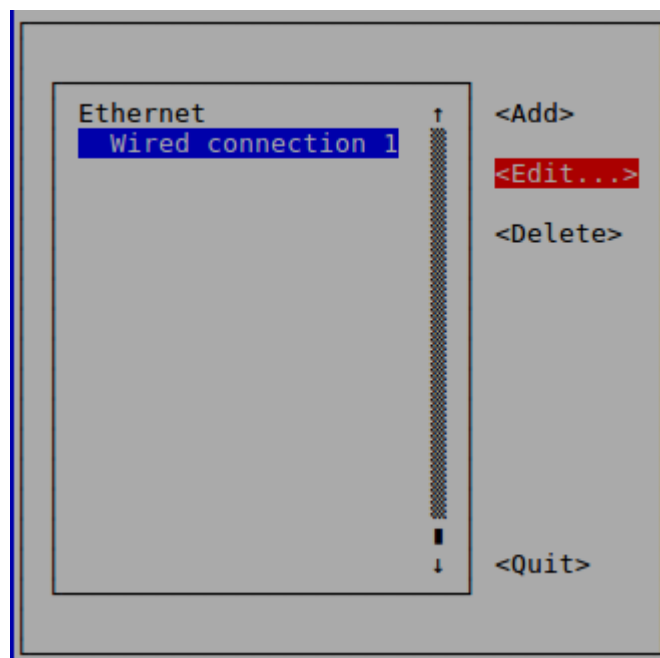
If your selection menu had a different name for it, that's ok. Use whatever is displayed.

If nothing is displayed on the selection menu. Then click Add and select Ethernet.   Chat
presented with the next screen, but it will be missing information, namely the Device name. You

will need to manually enter this information (you can get this from the `ip addr` command above.)

You should be presented with a screen like this:

```
┤ Edit Connection ├
        Profile name Wired connection 1_____
              Device enp4s1 (52:54:00:42:25:2A)

 = ETHERNET                                              <Show>

 = IPv4 CONFIGURATION <Automatic>                       <Show>
 = IPv6 CONFIGURATION <Automatic>                       <Show>

 [X] Automatically connect
 [X] Available to all users

                                              <Cancel> <OK>
```
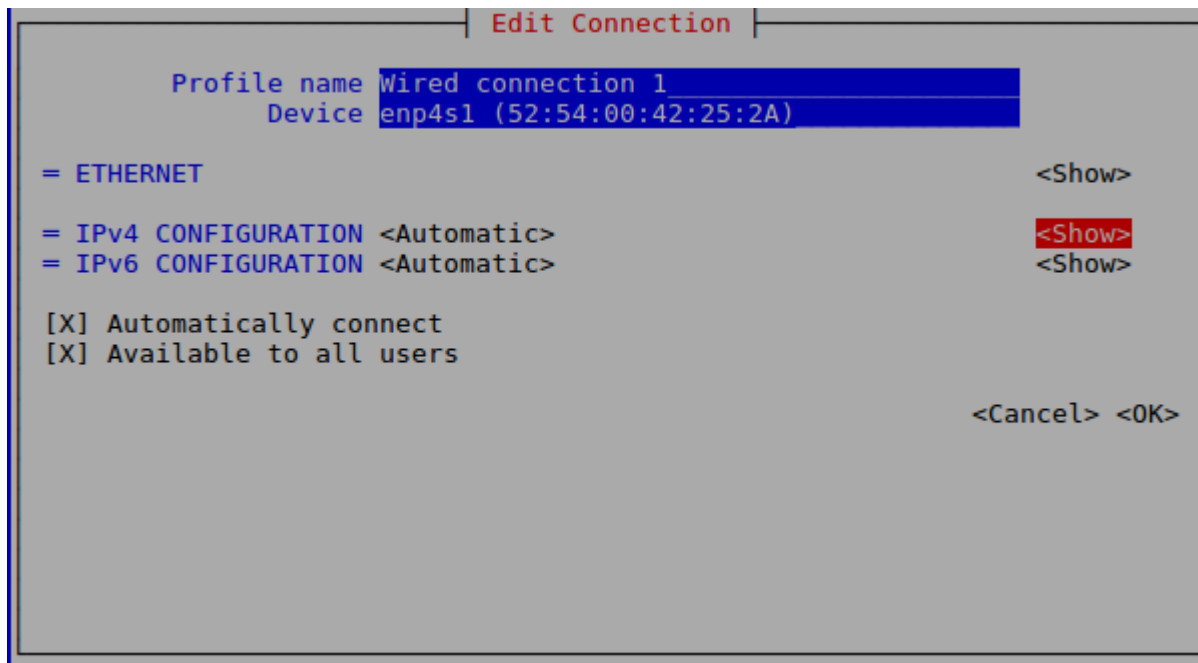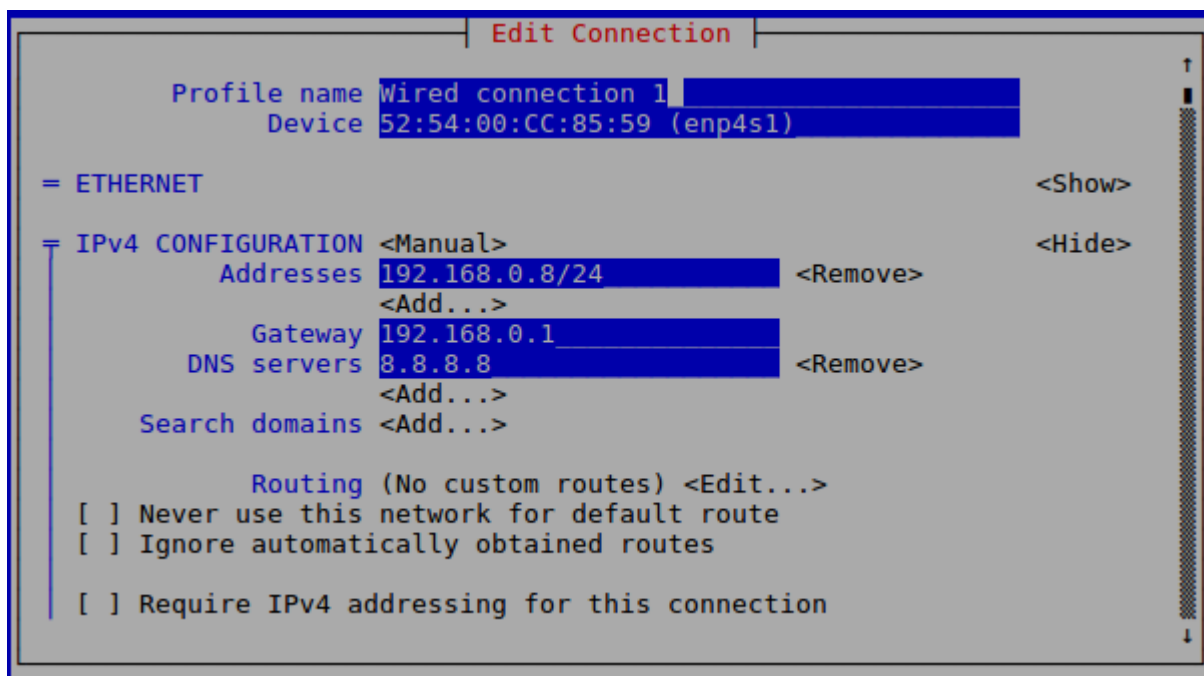
Initially, the settings for the IPv4 CONFIGURATION is set to **Automatic**, meaning DHCP. If you expand the choices using **Show**, you'll see the additional settings. You can change settings for IPv4 CONFIGURATION (**Automatic** vs **Manual**), **Addresses**, **Gateway**, **DNS Servers**, and **Search Domains**.

Change the IPv4 CONFIGURATION from **Automatic** to **Manual**. Arrow or tab down to Addresses and change this to **192.168.0.8/24**, the Gateway to **192.168.0.1** and the DNS servers to **8.8.8.8**.

```
┤ Edit Connection ├                                          ↑
        Profile name Wired connection 1_____ ▮
              Device 52:54:00:CC:85:59 (enp4s1)

 = ETHERNET                                              <Show>

 ╤ IPv4 CONFIGURATION <Manual>                           <Hide>
         Addresses 192.168.0.8/24            <Remove>
                   <Add...>
           Gateway 192.168.0.1_____
       DNS servers 8.8.8.8                    <Remove>
                   <Add...>
     Search domains <Add...>

           Routing (No custom routes) <Edit...>
 [ ] Never use this network for default route
 [ ] Ignore automatically obtained routes

 [ ] Require IPv4 addressing for this connection
                                                             ↓
```

After making these changes, move down to the **Ok** selection and hit the enter key   Chat
choose **Quit** to exit from nmtui.

On the command line, check `ifconfig` and ensure the changes have taken place. That is, you should see i**net addr: 192.168.0.8** under the **enp4s1** device. Also compare it with the information using `ip addr`.

If you do not see the information update, issue the following command:

> `nmcli network off && nmcli network on`

This command can be used to take down all interfaces and then bring them back up. It's very important to type it exactly as indicated above. If you are using an **SSH** connection (covered later), you can lock yourself out of a machine if you don't do this.
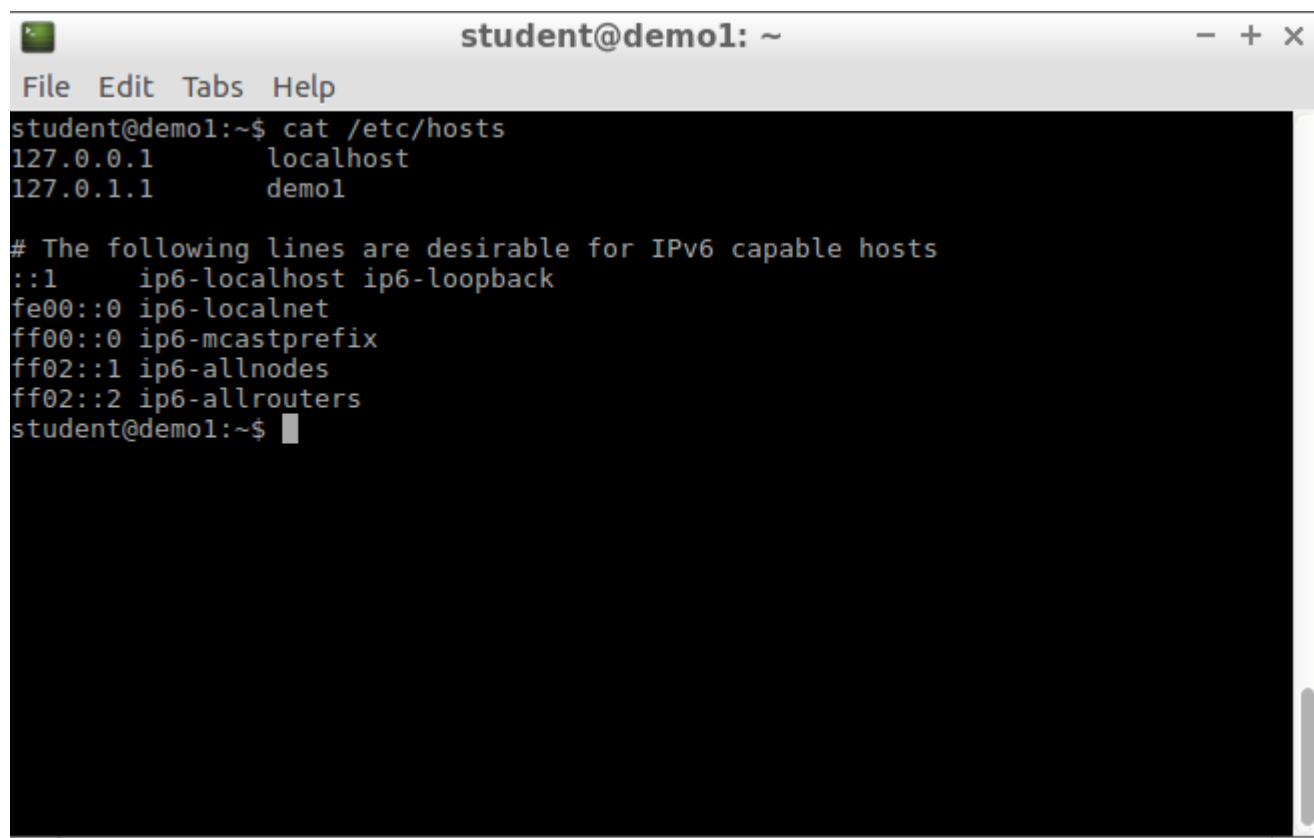
You should be able to ping the Gateway (192.168.0.1) using the IP address, `ping 192.168.0.1`. You should also be able to `ping google.com`.

Updating the hosts file

The **localhost** is only able to be resolved to the internal IPv4 address, `ping -c 3 localhost` will resolve to one of the IPv4 addresses listed above. This is not being done through DNS (more on this later). Instead, it uses a special file called the **hosts**. This file is found at **/etc/hosts**. You can use the `cat` command to view the contents of this file.

> `cat /etc/hosts`

```
student@demo1: ~                                    – + x
File  Edit  Tabs  Help
student@demo1:~$ cat /etc/hosts
127.0.0.1       localhost
127.0.1.1       demo1

# 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
student@demo1:~$
```

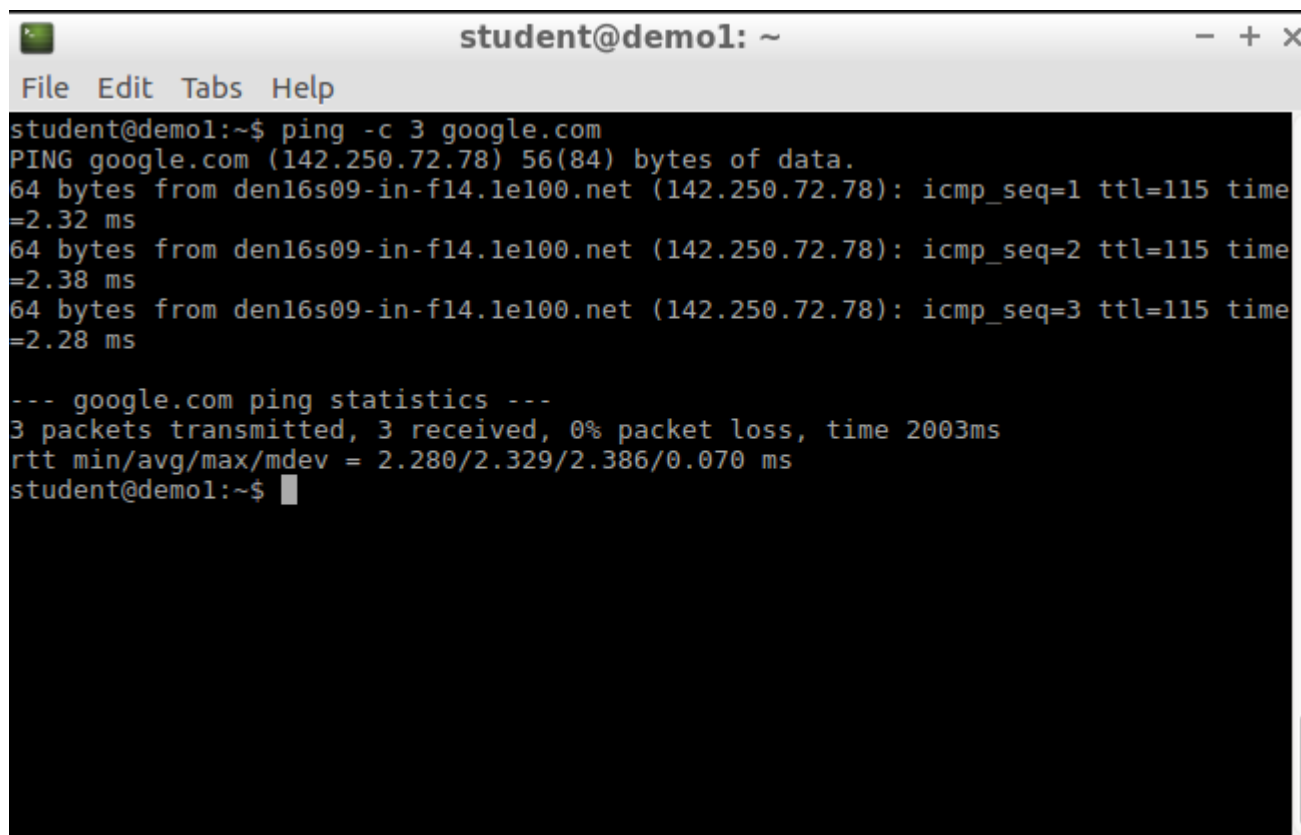The format of this file is:

Chat

IP address   hostname(s)

You can have multiple hostnames listed for a single IP.

When the computer is looking for a DNS resolution for a hostname or FQDN, the system will first check the **/etc/hosts** file. If it can find a corresponding entry, it will resolve to the specific IP address defined.

Try this:

> `ping -c 3 google.com`

```
student@demo1:~$ ping -c 3 google.com
PING google.com (142.250.72.78) 56(84) bytes of data.
64 bytes from den16s09-in-f14.1e100.net (142.250.72.78): icmp_seq=1 ttl=115 time
=2.32 ms
64 bytes from den16s09-in-f14.1e100.net (142.250.72.78): icmp_seq=2 ttl=115 time
=2.38 ms
64 bytes from den16s09-in-f14.1e100.net (142.250.72.78): icmp_seq=3 ttl=115 time
=2.28 ms

--- google.com ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 2003ms
rtt min/avg/max/mdev = 2.280/2.329/2.386/0.070 ms
student@demo1:~$ 
```

▼ Did it resolve? What was the IP? [Click to expand the answer]

Now edit the **/etc/hosts** file. Note that we will need to have elevated privileges for this action, so we will need to use the sudo command to run this as root.

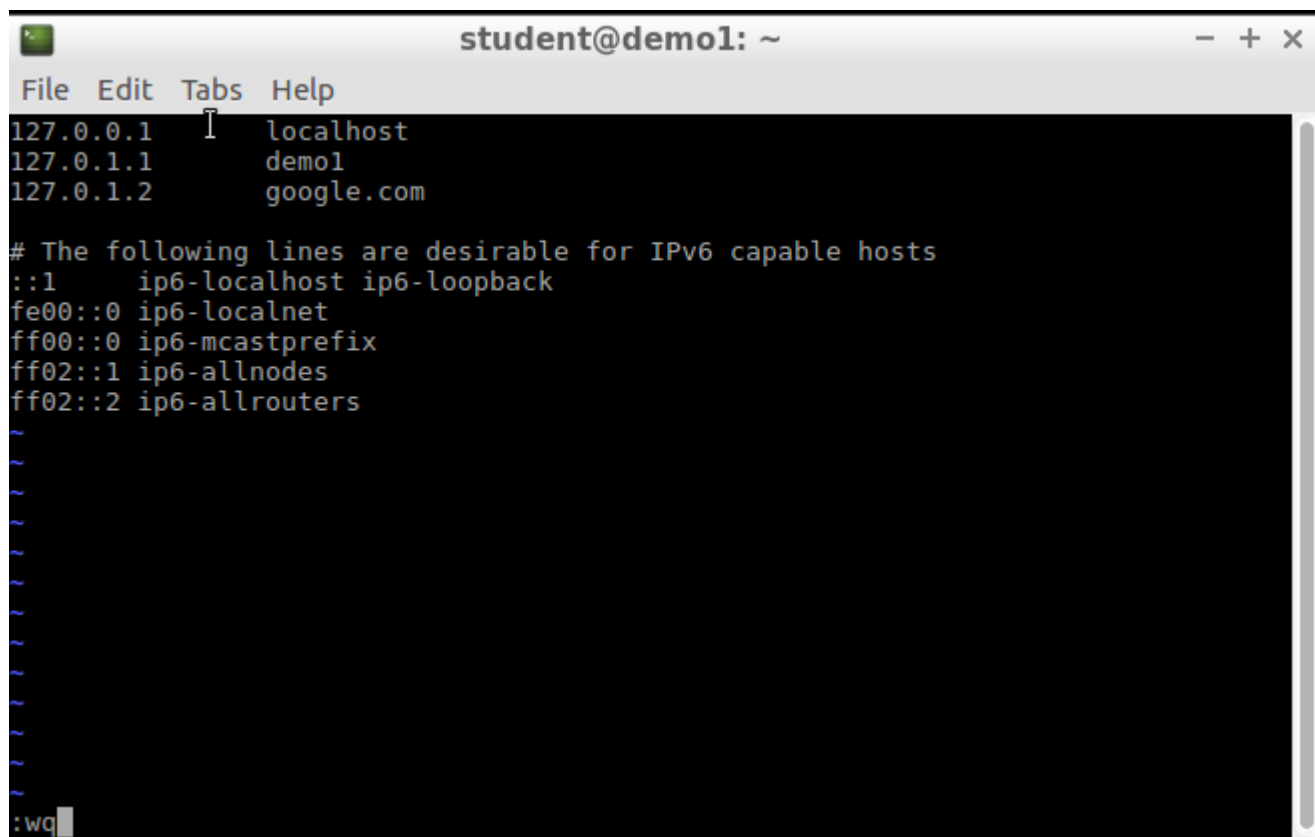> `sudo vi /etc/hosts`

Position the cursor on the line that starts 127.0.1.1 and type the letter o (the letter 'oh' and not the number 0). This will add a line below the cursor and put it into edit mode. Add the following:

127.0.1.2   google.com

Hit the escape key (`esc`) and then type   `:wq`   to save and quit the vi editor. Now repeat the ping command for google.

Chat

```
┌─────────────────────────────────────────────────────────────────┐
│  ▣              student@demo1: ~                    – + ×          │
├─────────────────────────────────────────────────────────────────┤
│  File  Edit  Tabs  Help                                           │
│ 127.0.0.1      I   localhost                                      │
│ 127.0.1.1          demo1                                          │
│ 127.0.1.2          google.com                                     │
│                                                                   │
│ # 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                                            │
│ ~                                                                 │
│ ~                                                                 │
│ ~                                                                 │
│ ~                                                                 │
│ ~                                                                 │
│ ~                                                                 │
│ ~                                                                 │
│ ~                                                                 │
│ ~                                                                 │
│ ~                                                                 │
│ :wq▮                                                              │
└─────────────────────────────────────────────────────────────────┘
```

▾ Did it successfully ping? Was it the correct address? [Click to expand the answer]

Remember to go back into the **/etc/hosts** file and remove that entry for **google.com** or you will not be able to browse to Google! Use `sudo vi /etc/hosts` , as before. To delete a line, simply put the cursor on the line using the arrow keys and type `dd` . This will remove just that line. You can save again using `:wq` , as before.

Lubuntu16 Demo 2 should have a hostname of **demo2**. Try to ping it as before from demo1, `ping -c 3 demo2` .

▾ Did it work? [Click to expand the answer]

Configure Demo 2

Open the console for Lubuntu Demo 2 and open a terminal. Use the `hostname` command to validate that the hostname is set correctly on Demo 2.

Because the hostname is currently set to **localhost** instead of **demo2** we will need to change the hostname. To do this we can issue the command `hostname <newhostname>` , for example, `hostname demo2` . However, this will only change the hostname during the current session. If we reboot the machine, it will not persist, and the machine will revert back to **localhost**.

Instead, we need to modify a file. We again need root privileges and the file is **/etc/hostname**. So, type the following command:
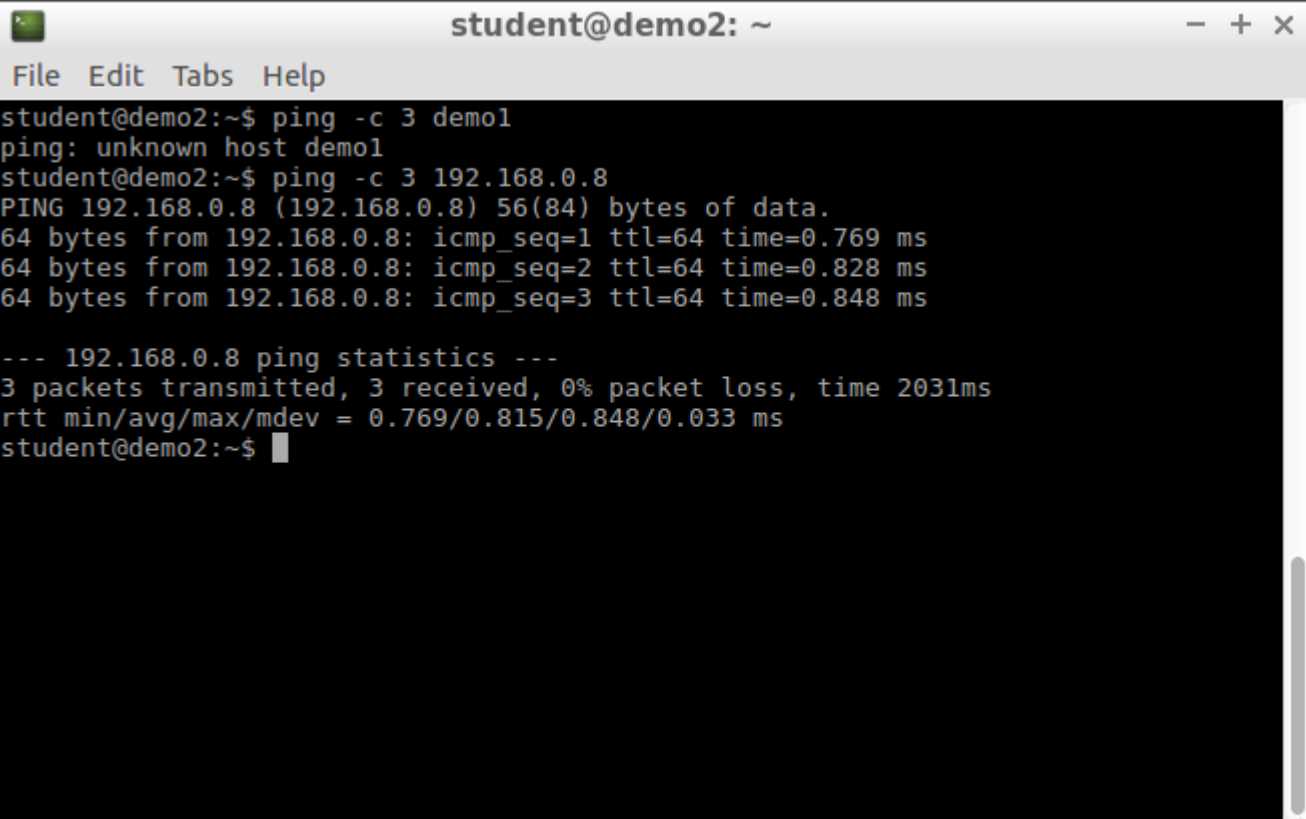
Chat

    `sudo vi /etc/hostname`

Notice the hostname is the only thing in this file. We will need to replace it with the new hostname.
First delete the current line using `dd` . Then type the letter `i` to go to edit (insert) mode and type
`demo2` . This should be the only entry in the file.
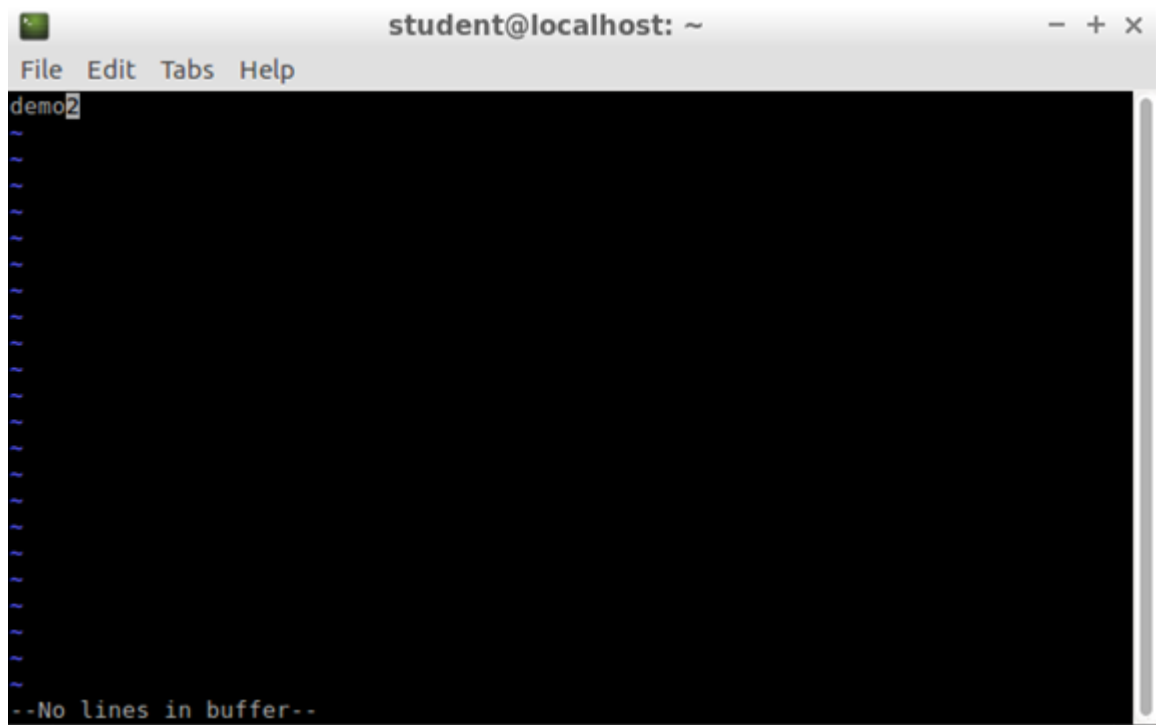
Use the escape key (`esc`) to get out of edit mode and `:wq` to save and quit the editor.

Now try to ping the other host, `ping host1`

Notice that the hostname did not resolve. Try to ping the IP address directly, `ping 192.168.0.8` .
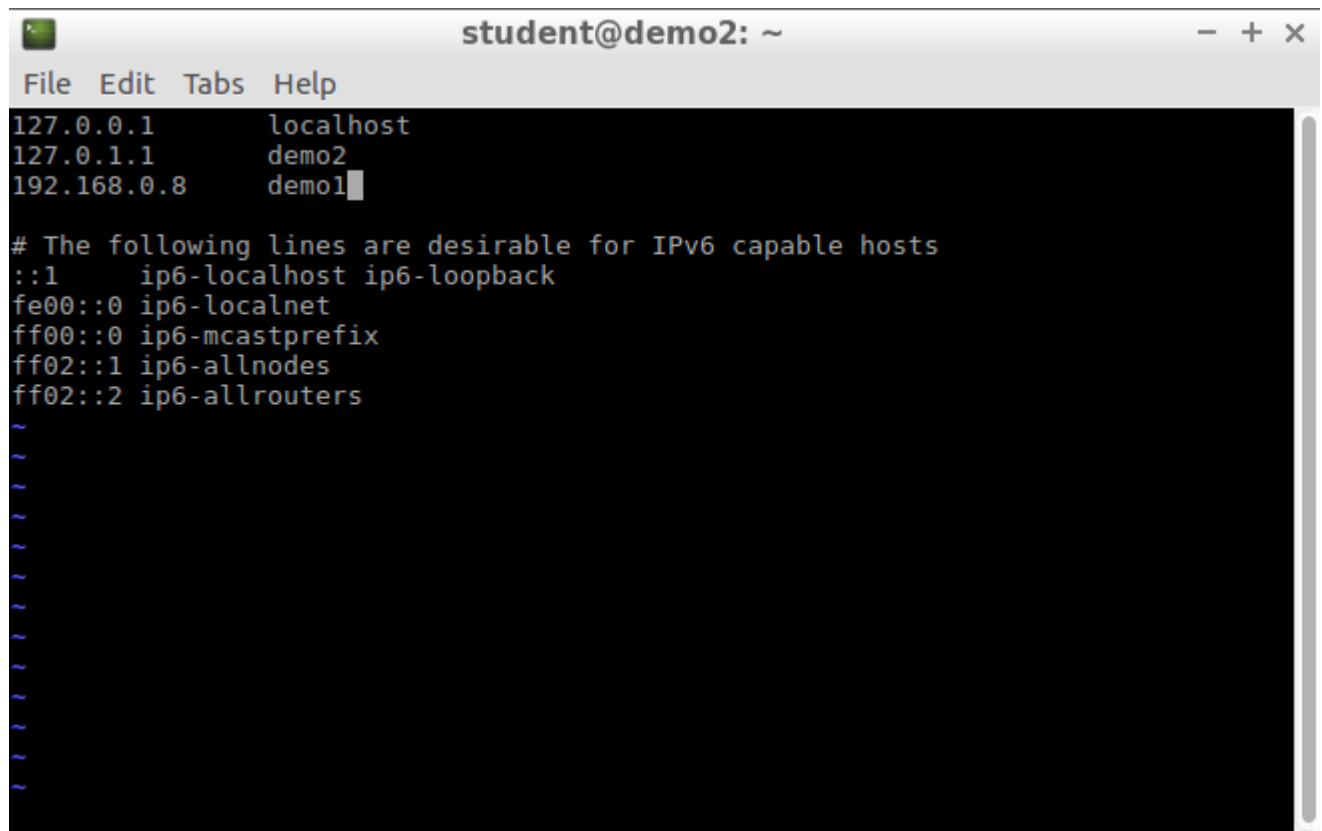Was this successful?

Chat

```
                            student@demo2: ~                    − + ×

 File  Edit  Tabs  Help
 student@demo2:~$ ping -c 3 demo1
 ping: unknown host demo1
 student@demo2:~$ ping -c 3 192.168.0.8
 PING 192.168.0.8 (192.168.0.8) 56(84) bytes of data.
 64 bytes from 192.168.0.8: icmp_seq=1 ttl=64 time=0.769 ms
 64 bytes from 192.168.0.8: icmp_seq=2 ttl=64 time=0.828 ms
 64 bytes from 192.168.0.8: icmp_seq=3 ttl=64 time=0.848 ms

 --- 192.168.0.8 ping statistics ---
 3 packets transmitted, 3 received, 0% packet loss, time 2031ms
 rtt min/avg/max/mdev = 0.769/0.815/0.848/0.033 ms
 student@demo2:~$
```

We do not have an internal DNS server set up, which means any machine on the **192.168.0.x** network will not resolve correctly. Instead, we need to add a line to the **/etc/hosts** file to map the IP address to the hostname. Using the IPv4 address for **demo1**, add this information to the **/etc/hosts** just like we did with Google above:
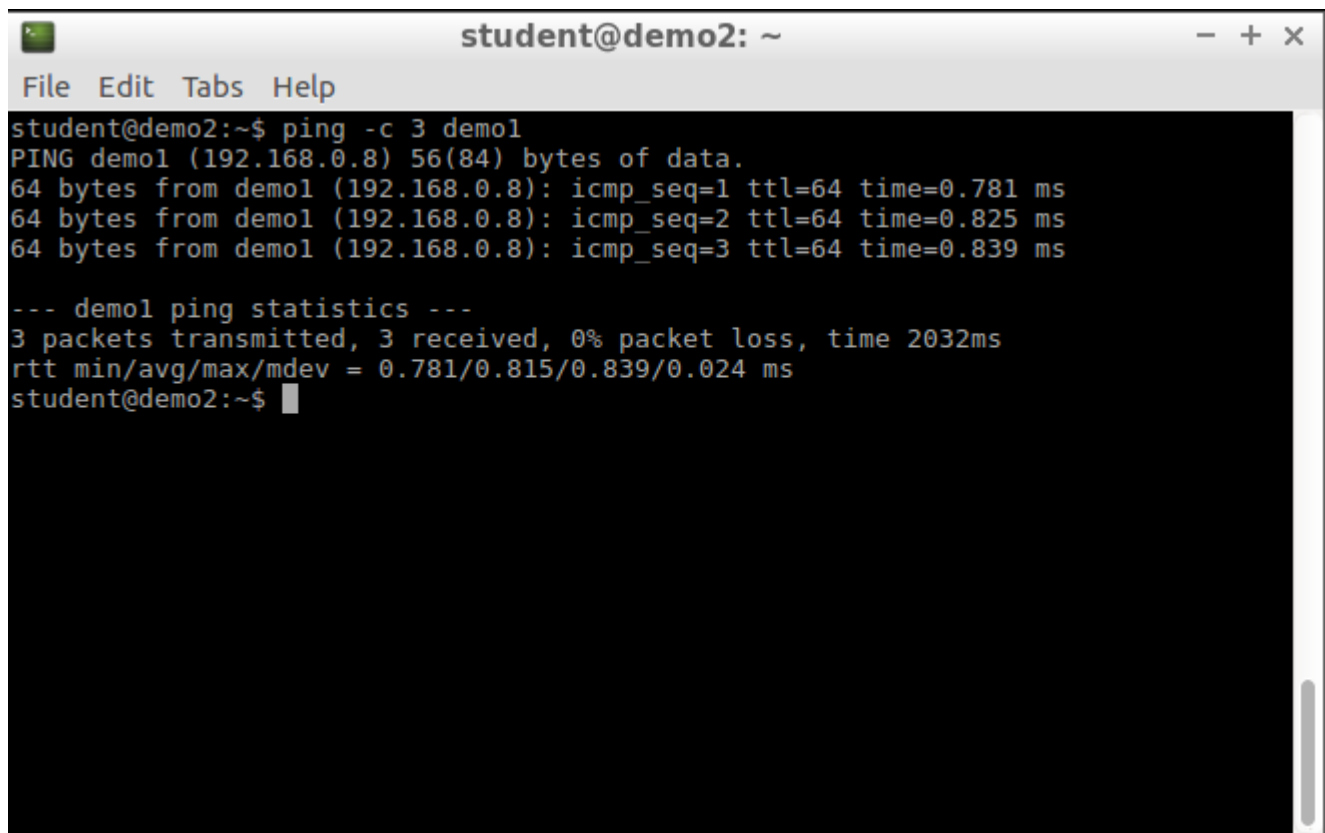
192.168.0.8   demo1

Note that your IP and hostname may be different.

```
                     student@demo2: ~                    − + ×
File  Edit  Tabs  Help
127.0.0.1        localhost
127.0.1.1        demo2
192.168.0.8      demo1

# 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
~
~
~
~
~
~
~
~
~
~
~
```

Try to ping demo1 again. It should be successful this time.

```
                     student@demo2: ~                    − + ×
File  Edit  Tabs  Help
student@demo2:~$ ping -c 3 demo1
PING demo1 (192.168.0.8) 56(84) bytes of data.
64 bytes from demo1 (192.168.0.8): icmp_seq=1 ttl=64 time=0.781 ms
64 bytes from demo1 (192.168.0.8): icmp_seq=2 ttl=64 time=0.825 ms
64 bytes from demo1 (192.168.0.8): icmp_seq=3 ttl=64 time=0.839 ms

--- demo1 ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 2032ms
rtt min/avg/max/mdev = 0.781/0.815/0.839/0.024 ms
student@demo2:~$
```

Appendix

Terminology

These terms will be covered in more detail in future classes.

IPv4	Internet Protocol version 4 – a 32-bit routable address on the network / internet
IPv6	Internet Protocol version 6 – a 128-bit routable address on the network / internet
DNS	Domain Name Service – Allows for a more efficient way of routing public IP addresses on the internet
NIC	Network Interface Card – The layer 1 device that allows for network communications
FQDN	Fully Qualified Domain Name – Full name of this computer on the current network. This is usually a routable name
MAC	The Media Access Control – Physical addressed assigned to the NIC, usually by the manufacturer

Home Lab

Building a home lab that mirrors this lab can be done easily.

The downloads needed will be:

> VirtualBox (https://www.virtualbox.org/wiki/Downloads)

> Lubuntu 16 ISO (http://cdimage.ubuntu.com/lubuntu/releases/16.04/release/)

Note that if the links above do not work, just google for the appropriate term.

Before setting up the machines, first create a NAT Network by going to **File->Preferences**. In the **Preferences** window, choose **Network**. On the righthand side, choose the **Add** button ( ). This will add a **NatNetwork** or **NatNetwork1**, if **NatNetwork** already existed. Select the newly create network and click on the Edit button ( ).

Chat
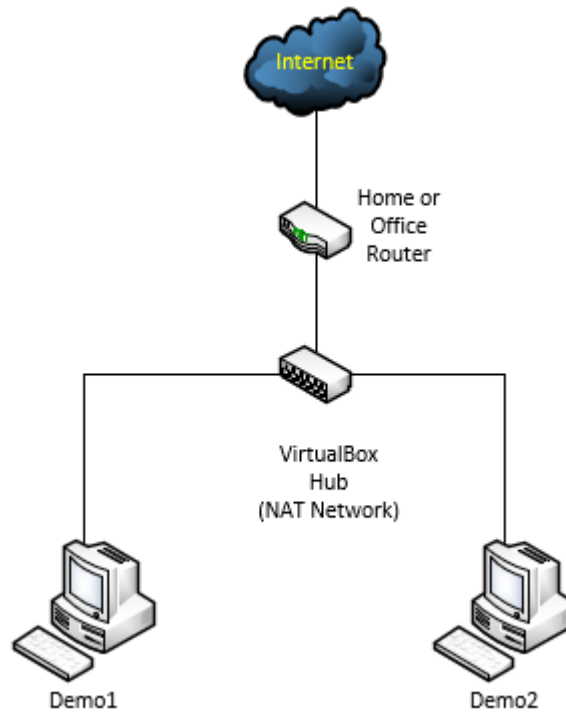
Change the Network Name to something useful, like **Net100Lab1** or something similar. For the Network **CIDR**, change it to be **192.168.0.0/24**.
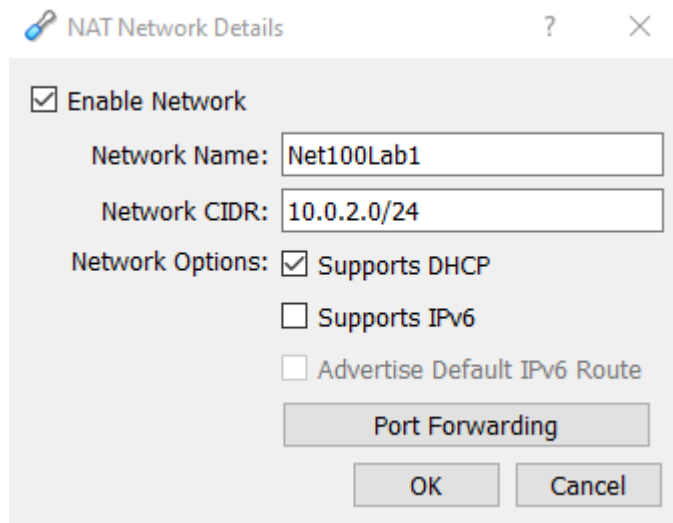
Build and Install the Lubuntu VMs

These machines will have minimum specs. Recommended:

RAM: 1024 KB

Hard drive: 20 GB

CPU: 1 Single core

Chat

Create the base VM and select Settings. Click on Network and choose NAT Network in the Attached to dropdown. Choose the network you created above.

Once you mount the Lubuntu16 ISO to the two Lubuntu boxes, go ahead and start them. Ensure you give them unique hostnames (recommend: demo1 and demo2, as indicated above). Once the installation is complete, run `sudo apt update && sudo apt upgrade` to ensure they are up to date.

From here you can begin the lab as above. Note that you do not have the VyOS router and instead VirtualBox will be the hub and the router will be the actual router on your network. We will study the details of this in a future lab.

Questions

Take the Quiz Again

⚠ Answers will be shown after your last attempt

Score for this attempt: **10** out of 10
Submitted Apr 20 at 4:25pm
This attempt took 36 minutes.

Question 1   1 / 1 pts

What does NIC stand for?

○ Network Insertion Code

◉ Network Interface Card

○ Not In Computer

○ Network Intercept Communications

Question 2   1 / 1 p   Chat

What does MAC stand for?

○ Motherboard And Computer

○ Monitor Address Connection

◉ Media Access Control

○ Multiple Authentication Connections

Question 3   1 / 1 pts

The loopback address is a very important address for computer communications. Which of the following is a loopback address?

◉ 127.0.0.1

○ 10.0.0.0

○ 192.168.1.0/24

○ 192.168.0.1

Question 4   1 / 1 pts

What does the ping command do?

○ Resets the NIC to enable network connectivity

○ Causes the computer to issue an audible beep

Chat

⦿ Verifies connectivity between two systems using the ICMP protocol

> Note the the ICMP protocol will be discussed in more detail in future classes.

○ Verifies the configuration of a NIC

Question 5   1 / 1 pts

Which of the following is a valid IPv4 address?

○ 275.10.10.0

○ 1.1.1.1.1

⦿ 192.168.10.50

○ 52:54:00:42:25:2a

Question 6   1 / 1 pts

Which of the following would give 1 ping only to the specified address?

○ ping -1 192.168.10.50

⦿ ping -c 1 192.168.10.50

○ ping 1 192.168.10.50

○ ping 192.168.10.50

Chat

Question 7   1 / 1 pts

You use the cat command on an /etc/hosts files and see the following entry:

    127.0.0.1   google.com

What will happen if you ping google.com?

- ⦿ You will get a successful ping result, but it will be coming from the local computer

- ◯ Nothing, the configuration file is wrong

- ◯ You will get a "host not found" error

- ◯ You will get a successful ping result coming from google.com

Question 8   1 / 1 pts

What is the decimal 192 in Hex?

- ◯ FF

- ⦿ C0

- ◯ AF

- ◯ 1E

Chat

Question 9   1 / 1 pts

What is 0xFF in binary?

○ 0000 0000

○ 1010 1010

◉ 1111 1111

○ 0101 0101

Question 10   1 / 1 pts

What does the command **ip a** do?

○ Reconfigures a NIC

○ Shows the path between nodes

○ Determines if we have connectivity between nodes

◉ Shows the IP Addresses associated with all the NICs

Quiz Score: 10 out of 10

How do you feel about this lesson?

👍👎

Have specific feedback?   Chat

Tell us here! **(https://flatironschoolforms.formstack.com/forms/canvas_feedback?**
CourseID=5336&LessonID=28215&LessonType=quizzes&CanvasUserID=10700&Course=None)

Chat