Lab 1
Setup of VMs and the Virtual Network Environment

All the labs in this course will be conducted in Linux Virtual Machines (VMs). In the first lab, you will create two VMs running in a host (your laptop or workstation). In addition, you will create a bridged network to connect the VMs and the host so they can communicate with each other through the Internet TCP/IP Protocol stack.

The VMs in this course will run Ubuntu OS on a Virtualbox hypervisor. Both Ubuntu and Virtualbox are open source and available freely.

I. Download and installation procedure

1. Go to the link https://ubuntu.com/download/desktop and click on <Download> tag. After a successful download, you should see a file

   Ubuntu-20.04.1-desktop-adm64.iso

   in the destination folder. This file contains Ubuntu bootable image.

2. Go to the Virtualbox site https://www.virtualbox.org/ to download Virtualbox. The VM will run in the Virtualbox environment.

3. Use the instruction of the link below to install a Linux VM (running Ubuntu OS) in Virtualbox:

   https://docs.oracle.com/cd/E26217_01/E26796/html/qs-create-vm.html#:~:text=To%20create%20a%20new%20virtual,VirtualBox%20command%20in%20a%20terminal.

4. Repeat step 3 to create another VM.

II. Create a virtual bridged network and test connectivity

5. In the Virtualbox Manager, start a VM by clicking on <start> tag. During the start-up process, you can assign user ID and the associated password.

6. Once the VM is completely activated, log in to the VM and open a text terminal. If you do not see the terminal icon, click on <Activities> tag and search for the terminal icon. Later, you will need the text terminal to issue commands to test the connectivity between VMs.

7. Repeat the steps above to setup another VM.

8. After the completion of the above steps, there are two VMs running in the host as illustrated in Figure 1:

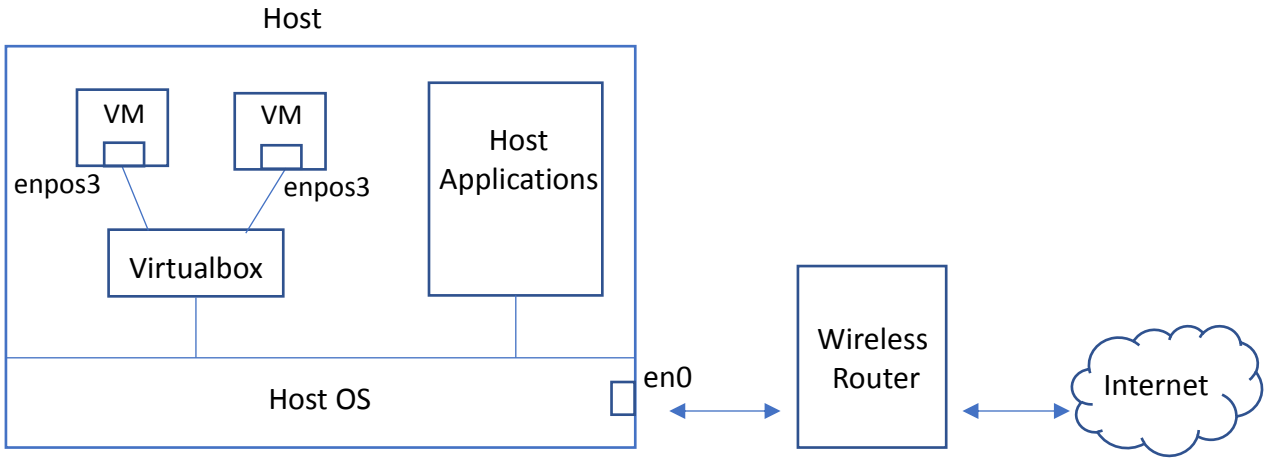

Fig 1

Each VM has one network interface enpos3, which is a logical interface. The traffic to and from the VM flows through this logical interface. The host has a wireless interface en0 (assuming you are using a MACbook as a host, host running window names the wireless interface as wireless), which is a physical interface. The VMs can access the Internet through interface en0. In other words, the Internet traffic to and from either VMs/host flows through this physical interface.

9. The default Virtualbox VM network setting is "NAT". Under this setting, the two VMs cannot communicate with each other. To enable the communication among VMs and the host, you can use the "bridged network" setting.

10. On the Virtualbox manager, click on <Settings> then <network>. Choose the option such that the network adaptor 1 is attached to a "Bridged" Adaptor. A bridge is a network device interconnecting end systems (host or VMs) such that they can communicate among themselves at the second layer. A bridge can be a physical or a virtual device.

11. After the above steps, the host and the VMs are logically connected to a bridge as shown in Figure 2. The VMs and the host have the same network ID probably in the IP address range of 192.168.0.xx/24 (the actual address range used depends on the settings of the DHCP server running in the wireless router of your home network. DHCP server is responsible to assign IP addresses to machines in your network). Figure 2 further illustrates logically how VMs and host are connected to the Internet.

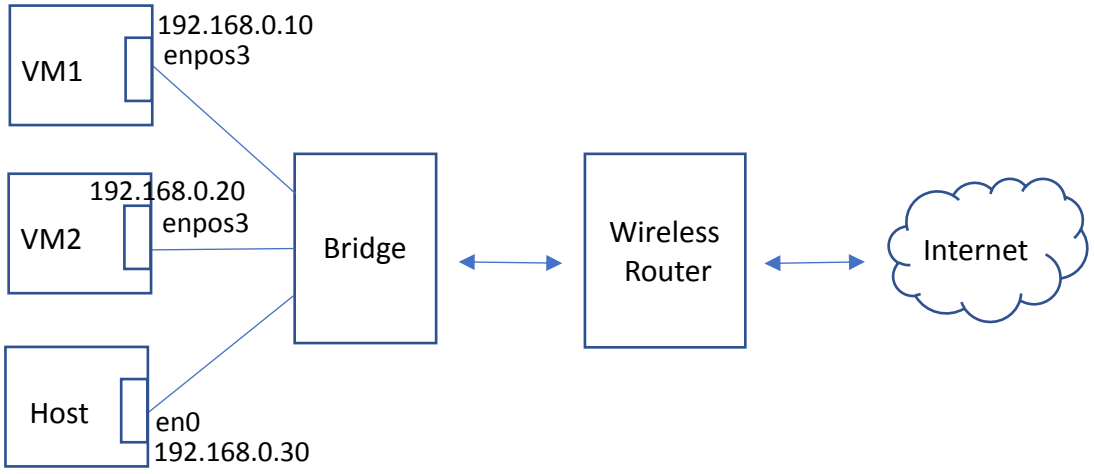

Fig 2

8. In this document, we assume that the first VM (VM1) has the IP address of 192.168.0.10/24, second VM (VM2) 192.168.0.20/24, host 192.168.0.30/24. You can find the actual IP address of the VM by typing

   $ifconfig enp0s3

   in the text terminal of the VM and look for the line similar to the following:

   inet 192.168.0.30 netmask 255.255.255.0 broadcast 192.168.0.255

   The number in the dot-decimal form (192.168.0.30 here) is the IP address. If your host is a MACbook, you can use the same command above to obtain the IP address of the host. If your host is running window OS, then open the command prompt terminal and type the command "ipconfig" to find the IP address.

9. Now you may test the connectivity of the three machines (2 VMs and the host) by using "Ping" command. To do that, suppose you want to test the connectivity between VM1 and VM2, type the following command at the terminal of VM1

   $ping 192.168.0.20

   This command instructs VM1 to send a series of probe (Ping) messages to VM2 (192.168.0.20). If there is a connectivity, VM2 will generate a response to each probe message. you should see a series of messages appear on the terminal window similar to the following message:

64 bytes from 192.168.0.20: icmp_seq=1 ttl=64 time=0.558 ms

Type <CTRL+C> to stop the "ping". You should also ping from any of the VMs to the host to test the connectivity between host and VMs.

10. Your VM can also initiate a connection outside of your home network (Assuming your home network has an Internet connection). Open a web browser in your VM and start browsing web sites over the Internet.

11. Run a simple web server in VM1 by typing this in the terminal of VM1

    $Python3 -m http.server 8000

    This command starts up a simple web server in VM1. The number 8000 is the port number assigned to the web server (more on that in the subsequent labs and lectures) and is part of the address of the server. You may reach this web server from VM2 and host through their web browsers. To do this, open a web browser in VM2/host and type in the complete address of the web server, which consists of the IP address and the port number:

    http://192.168.0.10:8000

    If the connection is successful, your browser should display a list of directories at the root directory of VM1. You may also use the smart phone to contact the web server as long as your phone uses the same WiFi (i.e. in the same network).
    (**Note**: the webserver setup in VM1 can also be reached from anywhere over the Internet. To do this, you need to setup port forwarding at your wireless router.)

What you needed to demonstrate to your TA

1. Show that you have 2 VMs powered up in the Virtualbox;
    a. Use Ifconfig command to show the IP address of each VM;
2. Show successful pings between VMs and between host and VM;
3. Start a web server and show that your host and smart phone can contact the web server.