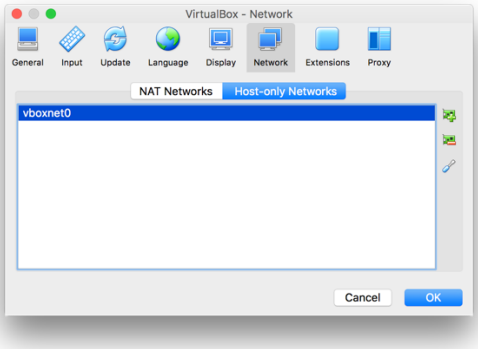
Lab 3: Mininet Hands-on Lab

Due date: 11:59pm, March 16th

In this lab, you will set up Mininet and run simple network experiments. Mininet is a virtual network tool developed by researchers at Stanford University. It allows creation of a realistic virtual network, running real kernel, switch and application code, on a single machine (VM, cloud or native). There is excellent information on mininet at http://mininet.org.

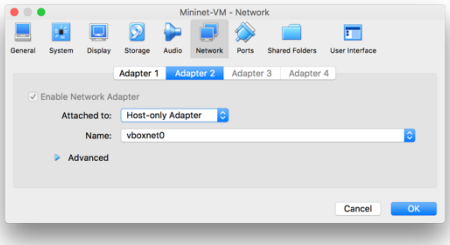
Mininet VM Installation

1. Download and install a virtualization system. We recommend VirtualBox (free, GPL) because it is **free** and works on OS X, Windows, and Linux
   Note: if you install VirtualBox on macOS high sierra (10.13.2) and experience installation failure, you need to allow "Oracle, America" from System -> Security & Privacy -> General/Allow… and select "Oracle, America" and then install Virtualbox again.
2. In VirtualBox, add additional network interface in the following steps:
   a. Click "preferences" and "network". You see there are two tabs "NAT NetworksThe "host-only networks" tab is initially empty.
   b. Click  on the right, you will see vboxnet0 being added to the list. The host-only network connects a virtual machine to the host virtual adaptor. Click  on the right, a pop up window allows you to specify the IPv4 address and the network mask of the host adaptor. By enabling DHCP server, the virtual machine's IP address can be automatically configured.



   Note: The host-only networks interface can also be enabled from command line using "VBoxManage hostonlyif create"

3. Download the Mininet VM image (Note this is *different* from the Mininet image from the official website).
4. Set up Mininet VM double-click on the **.ovf** file and import it.
   a. The default username/password is mininet/mininet
5. Select mininet-VM and click "Settings" and "Network". Under tab "Adapter 2", select "Host-only Adapter" and name "vboxnet0"



To activate both network interfaces each time when you restart the VM, make sure the following lines have been included in /etc/network/interfaces

```
# the primary network interface
auto etho
iface etho inet dhcp

# the secondary network interface
auto eth1
iface eth1 inet dhcp
```

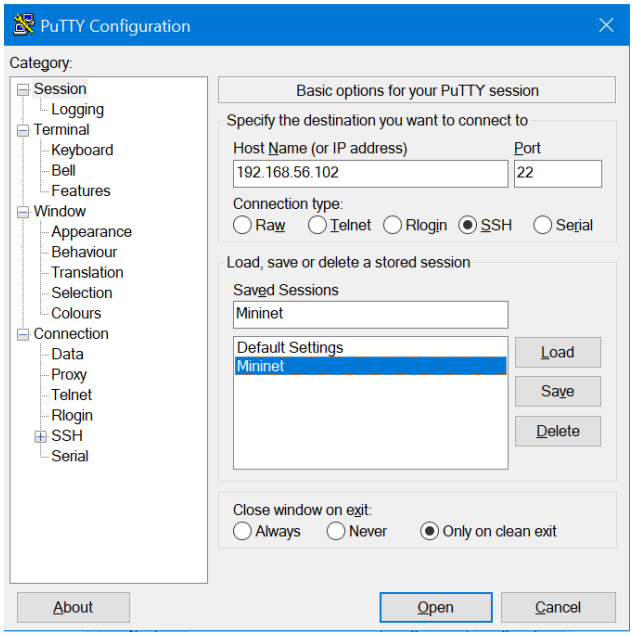
6. Install X Windows on your host computer. This is needed if you choose to run programs with a graphical interface by ssh to the Mininet virtual machine.
    a. For MacOS/OSX, you can download XQuartz from https://www.xquartz.org
    b. For Windows, you can download Xming X server from https://sourceforge.net/projects/xming/
7. To verify the setup, start the Mininet VM and start the X server. ssh to the VM with X forwarding enabled.
    a. The IP address of VM on the host-only network can be determined by "ifconfig" on the command line. Since we have set up two network interfaces, one for connecting the VM to the outside through NAT, and one for the the host-only network, you need to determine which interface is used for the later.
    b. If X Server successfully runs, upon logging in using *ssh*, you can run "xterm &" or "wireshark &" from the command line.

Note: It may be needed to run Wireshark with root access if it doesn't have permission to access you NIC: "sudo wireshark &"

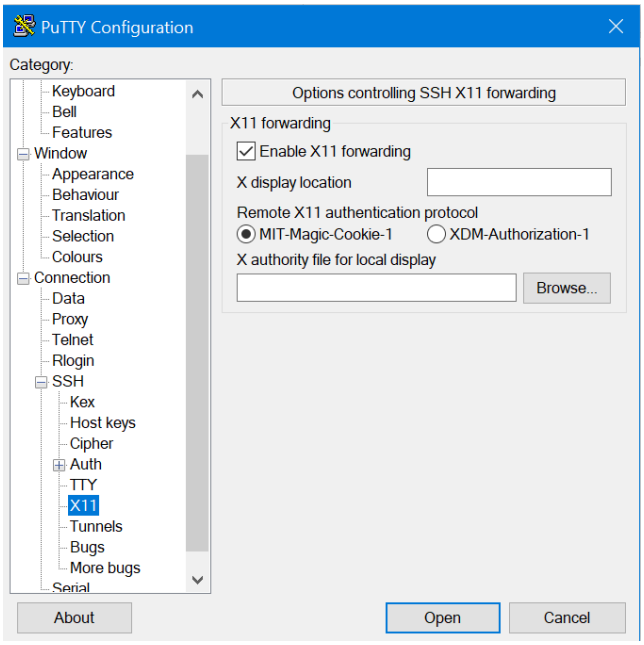
```
MacBook-Pro-2:Lectures rzheng$ ssh -X mininet@192.168.56.102
mininet@192.168.56.102's password:
Welcome to Ubuntu 14.04.4 LTS (GNU/Linux 4.2.0-27-generic x86_64)

 * Documentation:  https://help.ubuntu.com/
Last login: Tue Jan  2 15:53:41 2018 from 192.168.56.1
```

Figure 1 Screen shot of running ssh from a macOS terminal

    c. For Windows, download and use a SSH client software (like Putty):



Also enable “X11 forwarding” option before starting the session. This forwards the X sessions to XMing:



8. Go through the walkthrough (http://mininet.org/walkthrough/).

For those who have Ubuntu as the host system, you can install Mininet package directly using "apt-get install mininet".
Detailed instruction can be found at:
http://mininet.org/download/#option-3-installation-from-packages

After installing Mininet, you need to install the following packages if they are not yet in your system:

- **Netwox**
  - #sudo apt-get install netwox
- **Mininet Utils**
  - #git clone git://github.com/mininet/mininet
  - #mininet/util/install.sh -fw
- **Wireshark**
  - #sudo apt-get install wireshark
  - Wireshark is included in Mininet Utils, but the latest version is recommended
- **Telnet Service**
  - #sudo apt-get install xinetd telnetd

Practice Questions:

1. In Step 7, explain how to determine the IP address associated with the host-only network on the VM.
2. In a terminal, type *sudo mn --custom ~/mininet/custom/topo-2sw-2host.py --topo mytopo --link tc,bw=10,delay=100ms,* you will see

```
3.  mininet@mininet-vm:~/mininet/custom$ sudo mn --custom ~/mininet/custom/topo-2sw-2host.py -
    -topo mytopo --link tc,bw=10,delay=100ms
4.  *** Creating network
5.  *** Adding controller
6.  *** Adding hosts:
7.  h1 h2
8.  *** Adding switches:
9.  s3 s4
10. *** Adding links:
11. (10.00Mbit 100ms delay) (10.00Mbit 100ms delay) (h1, s3) (10.00Mbit 100ms delay)
    (10.00Mbit 100ms delay) (s3, s4) (10.00Mbit 100ms delay) (10.00Mbit 100ms delay) (s4, h2)
12. *** Configuring hosts
13. h1 h2
14. *** Starting controller
15. c0
16. *** Starting 2 switches
17. s3 s4 ...(10.00Mbit 100ms delay) (10.00Mbit 100ms delay) (10.00Mbit 100ms delay)
    (10.00Mbit 100ms delay)
18. *** Starting CLI:
19. mininet>
```

Now, at the mininet prompt > run 1) *links* 2) *h1 ping -c 5 h2* and 3) *iperf h1 h2*
Include the output from these command line inputs and explain why.
3. In the above example, we set all links to be of the same delay and bandwidth using the command link options. Now, modify the ~/mininet/custom/topo-2sw-2host.py so that 1) the

link between h1 and s3 and the link between s4 and h2 both have bandwidth of 10Mbps and delay 5ms, 2) the link between s3 and s4 has a bandwidth of 2 and delay of 2ms. Redo Question #2.

Hint: In mininet.topo.Topo.addlink (See http://mininet.org/api/classmininet_1_1topo_1_1Topo.html), one can specify the bw and delay as optional parameters in the form of bw=…, delay='…', where bandwidth is in unit of Mbps, and the unit of delay (e.g., ms or s) is included in the quotation marks.

Submission

Include your answers in a report and submit your report and the modified *topo-2sw-2host.py* through gitlab/Lab3.

Useful Resources

Mininet Python API http://mininet.org/api/index.html
Linux command cheat sheet https://www.cheatography.com/davechild/cheat-sheets/linux-command-line/pdf/

Appendix I Common Network Utilities

Details of all network utilities can be seen through "man XXX" in the Mininet VM terminal. In this appendix, we introduce a few commonly used ones.

ifconfig – configure a network interface

netstat – print network connections, routing tables, interface statistics

iperf – performance network throughput tests for TCP or UDP. iperf –s runs in server mode and iperf –c runs in client mode.

netwox – A network toolbox for traffic generation and hacking (See http://www.cis.syr.edu/~wedu/Teaching/cis758/netw522/netwox-doc_html/tools/index.html)

DO NOT RUN NETWOX TO HOSTS OUTSIDE YOUR VM!