

Instruction to setup Mininet VM for Programming Assignments

IF you are going to be working on the Lab Machines in DH 2020:

Step 1 : (Assuming you have logged on) Navigate to /virtual (Notice this is the root but not your home directory) and create a directory named by `csc358_yourutorid`.

Step 2: Run command `'chmod 700 csc358_yourutorid'` this will prevent other users from accessing your VMs and other files.

Step 3: Copy over the `mininet.zip` file from /virtual/joe to /virtual/`csc358_yourutorid`
You can unzip this by running command `'unzip mininet.zip'` in /virtual/`csc358_yourutorid`
Or through the Ubuntu GUI.

Step 4: After unzipping, you should see a list of vmdk files and vmx files. In Ubuntu, search and start VMarePlayer. Once prompted, chose to "Open a Virtual Machine" and select the vmx file you the directory you just unzipped.

Step 5: Power on the machine and login with user name `mininet` and password `mininet`

IF you are going to be working on your own machine:

Step 1: Download the mininet.zip file from this link (Also posted on portal)

<https://mcs.utm.utoronto.ca/csc358h5/index.php>

You will need to login with your UTorid

Step 2: Unzip the file to a location that you chose

Step 3: Use your own virtual machine runner (VMware Fusion or VirtualBox)

VirtualBox is free and VMware Fusion will require a license and setup procedure you can get from Andrew Wang from Mathematics and Computer Science Department

Step 4: (Optional)

If you want, you can setup port forwarding to forward guest os's port 22 to something else (Ex, 3022 or 8022) so you can ssh into your machine using `'ssh mininet@localhost'` instead of needing to find out its IP address.

Step 5: Power on the machine and login with username `mininet` and password `mininet`

IF you are going to ssh into the lab machine to use VM:

Step 1: Make sure you have X forwarding enabled (Install XQuartz on Mac or PuTTY on Win)

Step 2: ssh into the lab machine with `-X` flag:

On Mac: In XQuartz Terminal or Terminal,

`'ssh -X utorid@dh2020pcxx.utm.utoronto.ca'`

where xx is the machine number of which you created your VM

Step 3: In the ssh terminal, type 'vmplayer' to invoke vmplayer

Step 4: Chose to "Open a Virtual Machine" and select the vmx file in the directory you just unzipped.

Step 5: Power on the machine and login with user name **mininet** and password **mininet**

Remark: The mininet.zip and the unzipped vmdk/vmx files are stored local on the machines in DH2020 and will not be moving with your account. This means logging into your account on another computer (even in the same lab) will not provide you access to your vm and code.

We strongly recommend you protect the VM you created in the lab by changing the access control as mentioned above and BACK UP YOUR CODE REGULARLY by exporting them and save them to a secure place either physically or on the cloud.

Please REMEMBER, whichever way you chose to run the VM, your code MUST BE able to compile on UTM's system. You might receive a 0 if your code runs fine on your computer but not on the UTM's system.

Instruction to setup your testing/debug environment

- Step 1: On your VM, navigate to `~/cs144_lab3` and you should see files named as:
`run_mininet.sh`, `run_pox.sh`, `sr_solution.sh` and a bunch of other files
- Step 2: It is suggested that you have multiple terminal windows open at the same time so it would be easier for you to investigate packets and their flows through your router.
- Step 3: In addition to the VM window you already have opening, open three terminal sessions and login as username **mininet** and password **mininet**.
(If you have not set port forwarding, you need to find out the machine address by typing `ifconfig` and get the ip address of adapter `eth0`)

Without Port Forwarding: `'ssh mininet@ipaddress'`

With Port Forwarding: `'ssh mininet@localhost -p xxxx'`

Where `xxxx` is the port that you forward to

- Step 4: In the first window, type `'./run_mininet.sh'` and wait for the following messages to show

```
{ 'server1': '192.168.2.2', 'sw0-eth3': '10.0.1.1', 'sw0-eth1': '192.168.2.1', '
sw0-eth2': '172.64.3.1', 'client': '10.0.1.100', 'server2': '172.64.3.10' }
*** Creating network
*** Creating network
*** Adding controller
Unable to contact the remote controller at 127.0.0.1:6653
Unable to contact the remote controller at 127.0.0.1:6633
Setting remote controller to 127.0.0.1:6653
*** Adding hosts:
client server1 server2
*** Adding switches:
sw0
*** Adding links:
(client, sw0) (server1, sw0) (server2, sw0)
*** Configuring hosts
client server1 server2
*** Starting controller
c0
*** Starting 1 switches
sw0 ...
*** setting default gateway of host server1
server1 192.168.2.1
*** setting default gateway of host server2
server2 172.64.3.1
*** setting default gateway of host client
client 10.0.1.1
*** Starting SimpleHTTPServer on host server1
*** Starting SimpleHTTPServer on host server2
*** Starting CLI:
mininet>
```

Step 5: In the second window, type './run_pox.sh' and wait for the following messages:

```
mininet@mininet-vm:~/cs144_lab3$ ./run_pox.sh
POX 0.0.0 / Copyright 2011 James McCauley
DEBUG:.home.mininet.cs144_lab5.pox_module.cs144.ofhandler:*** ofhandler: Successfully loaded ip settings for
hosts
{'server1': '192.168.2.2', 'sw0-eth3': '10.0.1.1', 'sw0-eth1': '192.168.2.1', 'sw0-eth2': '172.64.3.1', 'cl
ient': '10.0.1.100', 'server2': '172.64.3.10'}

INFO:.home.mininet.cs144_lab5.pox_module.cs144.srhandler:created server
DEBUG:.home.mininet.cs144_lab5.pox_module.cs144.srhandler:SRServerListener listening on 8888
DEBUG:core:POX 0.0.0 going up...
DEBUG:core:Running on CPython (2.7.6/Oct 26 2016 20:30:19)
INFO:core:POX 0.0.0 is up.
This program comes with ABSOLUTELY NO WARRANTY. This program is free software,
and you are welcome to redistribute it under certain conditions.
Type 'help(pox.license)' for details.
DEBUG:openflow.of_01:Listening for connections on 0.0.0.0:6633
INFO:openflow.of_01:[Con 1/1] Connected to 00-00-00-00-00-01
DEBUG:.home.mininet.cs144_lab5.pox_module.cs144.ofhandler:Connection [Con 1/1]
INFO:.home.mininet.cs144_lab5.pox_module.cs144.ofhandler:Creating learning switch 00-00-00-00-00-01
Ready.
POX> █
```

These messages indicates that your mininet is setup correctly and pox had established connections

Step 6: In another window, you can start your reference solution by running command './sr_solution' and expect a display similar to this:

```
-----
^C
mininet@mininet-vm:~/cs144_lab3$ ./sr_solution
Using VNS sr stub code revised 2009-10-14 (rev 0.20)
Loading routing table from server, clear local routing table.
Loading routing table
-----
Destination    Gateway      Mask        Iface
10.0.1.100     10.0.1.100  255.255.255.255 eth3
192.168.2.2    192.168.2.2  255.255.255.255 eth1
172.64.3.10    172.64.3.10  255.255.255.255 eth2
-----
Client mininet connecting to Server localhost:8888
Requesting topology 0
successfully authenticated as mininet
Loading routing table from server, clear local routing table.
Loading routing table
-----
Destination    Gateway      Mask        Iface
10.0.1.100     10.0.1.100  255.255.255.255 eth3
192.168.2.2    192.168.2.2  255.255.255.255 eth1
172.64.3.10    172.64.3.10  255.255.255.255 eth2
-----
Router interfaces:
eth3  HWaddr26:01:e4:bf:36:8a
      inet addr 10.0.1.1
eth2  HWaddr42:4a:67:cc:6f:83
      inet addr 172.64.3.1
eth1  HWaddr6:15:d1:e0:bd:57
      inet addr 192.168.2.1
<-- Ready to process packets -->
```

* If you did not see the <-- Ready to process packets --> line, try exit mininet and restart it then interrupt your sr_solution and restart sr_solution.

Step 7: Test your code by typing commands in the mininet command line and see changes on Pox window and output of sr_solution. Pay close attention to packets received and sent by the router and the header changes.

Commands available for testing is included in your PA description PDF,
ex 'client ping -c 3 10.0.1.1'

* Recall that we have one more terminal window opened that was not utilized. This window is for you to transfer file from your computer in and out of the VM. You could easily use the scp command to copy files both directions and chose your preferred IDE to code.

* Inspect the Makefile content to see how your code gets compiled and made into the executable, you will need to modify this if you wish to add files to the existing structure.

Before we are done with setting up the discussion board, feel free to email me if you have any questions. However, any email that IS NOT sent through a UofT email address will be disregarded.

My email address is: zhancheng.song@mail.utoronto.ca