# Assignment - Mininet Install and Test

**Total points: 100**

**Your name: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

This lab covers the basics of the Mininet network emulation environment under which your programming assignment will be carried out. The second part covers the basics of the POX controller that you need to program.

## Setup Mininet

The easiest way to get started is to use **a pre-packaged Mininet/Ubuntu Virtual Machine (VM)**. This VM contains the Mininet itself, all OpenFlow binaries and controllers, and tweaks to the kernel configuration to support larger Mininet networks.

### Download Virtualization System

You can use any virtualization system of your choice, but we recommend installing VirtualBox. It’s free and runs on Windows, Linux and OS X.

VirtualBox (Recommended): <https://www.virtualbox.org/wiki/Downloads> For other virtualizations systems, visit [here.](http://www.openflow.org/wk/index.php/OpenFlow_Tutorial#Set_up_Virtual_Machine)

### Download Mininet VM

We recommend using the latest Mininet image, which is in the package file we provide. You can also go to [www.mininet.org](http://www.mininet.org/) for more information. It comes with the latest version of Mininet and two OpenFlow Controllers (POX and Pyretic). The download will take some time. It’s ~ 800MB, compressed.

### Setup Virtual Machine

#### Start VirtualBox

* + **Select File>Import Appliance and select the .ova file**
  + **Press the "Import" button.**

o This will unpack and import the VM in your local machine. It will take a while, as the unpacked image is about 3 GB.

### Boot VM

Now, you are ready to start your VM. Press the "Start" arrow icon or double-click your VM within the VirtualBox window.

In the VM console window, log in with the user name and password for your VM. The username and password for this VM are:

* + User name - **mininet**
  + Password - **mininet**

Note that this user is a sudoer, so you can execute commands with root permissions by typing *sudo command*, where *command* is the command you wish to execute with root permission.

Another option is to install Mininet from source. You need to have a VM (Ubuntu 16.04 with GUI). Run the following command (make sure you have git installed):

* git clone git://github.com/mininet/mininet

Go to mininet directory and check out the latest version (in my case it is 2.2.2):

* cd mininet
* git tag # list available versions
* git checkout -b 2.2.2

Run install shell script. Option **-s /home/orkhans/mn**instructs installer where it should install Mininet files:

* ./util/install.sh -s /home/orkhans/mn -a

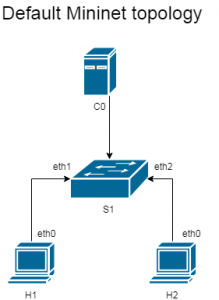
You can run the following command to make sure you have installed everything successfully:

* sudo mn --test pingall

Provide a screenshot of the screen showing mininet VM logged in or successful testing. **[50 points]**

## Mininet CLI

We will be using CLI to manage our virtual network.  The default topology includes two hosts (h1,h2), OpenFlow Switch(s1) and OpenFlow controller(c0).



Run **sudo mn** to enter CLI mode. Commands net and nodes give us information about our nodes and their interconnections:

* mininet> nodes
* available nodes are:
* c0 h1 h2 s1
* mininet> net
* h1 h1-eth0:s1-eth1
* h2 h2-eth0:s1-eth2
* s1 lo: s1-eth1:h1-eth0 s1-eth2:h2-eth0
* c0

Provide a screenshot of command line. **[25 points]**

We can find IP address of either host by running  **h1 ifconfig -a**or **h2 ifconfig -a**

We can test connectivity between them using **ping** command:

* mininet> h1 ping h2
* PING 10.0.0.2 (10.0.0.2) **56**(84) bytes of data.
* 64 bytes from 10.0.0.2: icmp\_seq=1 ttl=64 time=2.34 ms
* 64 bytes from 10.0.0.2: icmp\_seq=2 ttl=64 time=0.393 ms
* 64 bytes from 10.0.0.2: icmp\_seq=3 ttl=64 time=0.078 ms
* ^C
* --- 10.0.0.2 ping statistics ---
* 3 packets transmitted, 3 received, 0% packet loss, time 2031ms
* rtt min/avg/max/mdev = 0.078/0.938/2.345/1.003 ms

Another useful CLI command is **dump** which displays summary information about all devices:

* mininet>dump
* <Host h1: h1-eth0:10.0.0.1 pid=3300>
* <Host h2: h2-eth0:10.0.0.2 pid=3302>
* <OVSSwitch s1: lo:127.0.0.1,s1-eth1:None,s1-eth2:None pid=3307>
* <Controller c0: 127.0.0.1:6653 pid=3293>

Provide a screenshot of command line. **[25 points]**

## Appendix- Mininet

Mininet is a powerful network emulation tool. It creates a realistic virtual network, running real kernel, switch and application code on a single machine. You can easily interact with your network using Command Line Interface(CLI). Notice that to test connectivity, you need to run a POX controller that has switch functionality.

## Mininet Basics

Display Mininet Command Line Interface (CLI) commands: mininet> help

Display nodes:

mininet> nodes

If the first string of the CLI command is a host, switch or controller name, the command is executed on that node. For instance, to show the interface of host h1:

mininet> h1 ifconfig

Test connectivity between hosts. For example, test the connectivity between h1 and h2: mininet> h1 ping h2

Alternatively, you can test the connectivity between all hosts by typing: mininet> pingall

Exit Mininet:

mininet> exit Clean up:

$ sudo mn –c

### Mininet References:

* <http://mininet.org/>
* <https://github.com/mininet/mininet/wiki/Introduction-to-Mininet>