Lab exercises

OpenFlow
Mininet

Better if you connect to mininet via ssh
user: mininet
password: mininet

Set Network as Only host adapter

Start a basic topology

```
sudo mn --topo single,3 --mac --switch ovsk
--controller remote
```
Topology

OpenFlow Tutorial: 3 hosts-1 switch topology

- **s1**: OpenFlow Switch
  - s1-eth0
  - s1-eth1
  - s1-eth2
  - 127.0.0.1:6634

- **h2**: 10.0.0.2
- **h3**: 10.0.0.3
- **h4**: 10.0.0.4

- **c0**: Controller
  - port 6633
  - loopback (127.0.0.1)

- **dpctl**: (user-space process)
Controller

POX is a Python-based SDN controller platform geared towards research and education.
For more details on POX, see About POX or POX Documentation on NOXRepo.org.

Let’s try running a basic hub example:
$.pox.py log.level --DEBUG misc.of_tutorial

This tells POX to enable verbose logging and to start the of_tutorial component which you'll be using (which currently acts like a hub).
Controller

Wait until the application indicates that the OpenFlow switch has connected.

When the switch connects, POX will print something like this:

INFO:openflow.of_01:[Con 1/1] Connected to 00-00-00-00-00-01
DEBUG:samples.of_tutorial:Controlling [Con 1/1]

The first line is from the portion of POX that handles OpenFlow connections. The second is from the tutorial component itself.
Verify Hub Behavior with tcpdump

Create xterms for each host and view the traffic in each.

In the Mininet console, start up three xterms:

```
mininet> xterm h1 h2 h3
```

for h2 and h3:

```
tcpdump -XX -n -i h2-eth0
tcpdump -XX -n -i h3-eth0
```

for h1:

```
ping -c1 10.0.0.2
```
Build your own learning switch

On init, create a dict to store MAC to switch port mapping

```python
self.mac_to_port = {}
```

On packet_in,

- Parse packet to reveal src and dst MAC addr
- Map src_mac to the incoming port
- Lookup dst_mac in mac_to_port dict to find next hop
- If found, create flow_mod and send
- Else, flood like hub
Hints

To initialize a dictionary:
```python
mactable = {}
```

To add an element to a dictionary:
```python
mactable[0x123] = 2
```

To check for dictionary membership:
```python
if 0x123 in mactable:
    print 'element 2 is in mactable'
if 0x123 not in mactable:
    print 'element 2 is not in mactable'
```

To print a debug message in POX:
```python
log.debug('saw new MAC!')
```

To print an error message in POX:
```python
log.error('unexpected packet causing system meltdown!')
```

To print all member variables and functions of an object:
```python
print dir(object)
```