Mininet/Openflow

Objectives

In this lab, you will start by learning the basics of running Mininet in a virtual machine. Mininet facilitates creating and manipulating Software Defined Networking components. Through mininet you will explore OpenFlow, which is an open interface for controlling the network elements through their forwarding tables. A network element may be converted into a switch, router or even an access points via low-level primitives defined in OpenFlow. This lab is your opportunity to gain hands-on experience with the platforms and debugging tools most useful for developing network control applications on OpenFlow.

- Access Mininet on the hosted virtual machine
- Run the Ryu controller with a sample application
- Use various commands to gain experience with OpenFlow control of OpenvSwitch

Network Topology

The topology we are using is similar to the one in the Openflow Tutorial (https://github.com/osrg/ryu/wiki/OpenFlow_Tutorial). It has three hosts named h1, h2 and h3 respectively. Each host has an Ethernet interface called h1-eth0, h2-eth0 and h3-eth0 respectively. The three hosts are connected through a switch names s1. The switch s1 has three ports named s1-eth1, s1-eth2 and s1-eth3. The controller is connected on the loopback interface (in real life this may or may not be the case, it means the switch and controller are built in a single box). The controller is identified as c0 and connected through port 6633.

```
     +----------------------------+
     |                            |
     |      C0 - Controller       |
     |                            |
     +-------------+--------------+
                   |
     +-------------+--------------+
     |                            |
     |      S1 - OpenFlow         |
     |          Switch            |
     |                            |
     +-+----------+----------+----+
     s1-eth0    s1-eth1     s1-eth2
      +            +             +
      |            |             |
      |            |             |
      v            v             v
h1-eth0         h2-eth0          h3-eth0
+-+--+          +-+--+           +-+--+
| H1 |          | H2 |           | H3 |
+----+          +----+           +----+
```

You will need a Number

The instructor should have given everyone a different number. This will dictate which virtual machine you will be using. If the instructor happened to forget, then this is the time to remind them.

Write this number down on a piece of paper.

Anytime this lab mentions substitute it with the number you have written down.

Connect to your SDN VM with SSH

Open a terminal window on your machine. If you don't know how to do this ask an instructor for help.

Make sure

At the prompt type:

```
ssh ryu@192.168.122.<NUMBER>      (This is 100 + what your PC number was)
```

When you see the warning that looks something like this:

```
The authenticity of host '192.168.122.<NUMBER> (192.168.122.<NUMBER>)' can't be
established.
RSA key fingerprint is e8:05:43:d5:9a:4b:72:ad:c9:92:97:ca:df:32:86:ab.
Are you sure you want to continue connecting (yes/no)? yes

Type "yes" and press ENTER.
```

When prompted with "ryu@192.168.122.'s password:" enter the password ryu

That should be it. You are now connected to a terminal session on your machine.

```
$ ssh ryu@192.168.122.<NUMBER>
ryu@192.168.122.<NUMBER>'s password:
Welcome to Ubuntu 13.04 (GNU/Linux 3.8.0-19-generic x86_64)

* Documentation:  https://help.ubuntu.com/
Your Ubuntu release is not supported anymore.
For upgrade information, please visit:
http://www.ubuntu.com/releaseendoflife

New release '13.10' available.
Run 'do-release-upgrade' to upgrade to it.
```

```
Last login: Wed Sep  3 08:22:23 2015
sysadm@sdnXXX:~$
```

To log out you can type:

```
exit
```