Lab 4

*CST 311, Introduction to Computer Networks, Spring 2021*

**READ INSTRUCTIONS CAREFULLY BEFORE YOU START THE LAB.**

This lab is due on Sunday, May 9, 2021.

Lab must be submitted electronically to iLearn on <https://ilearn.csumb.edu>  by 11:55 p.m. on the due date.  Late lab assignments will not be accepted.

Lab must in pdf format only. Any other formats will not be accepted.  The naming convention of the file should be Lab4\_yourlastname.pdf. **Put your name in the document as well.** Your lab submission should present the answers in the original order and be properly labeled.

This lab is worth 100 points. Each part of a question carries equal weight unless specified otherwise.

Name :   \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

## SDN and Openflow

Watch the “Introduction to OpenFlow” video as much as you need in order to be able to replicate what is demonstrated on your own Mininet/OpenDaylight networking setup at:

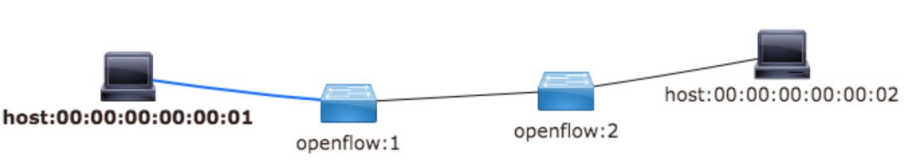
<https://www.youtube.com/watch?v=l25Ukkmk6Sk>

(This is just one of many online resources available for you to continue learning about computer networking.) Note that there are a few differences between what’s demonstrated in the video and what our versions of Mininet and Wireshark do:

● Filter on ‘openflow\_v1’ instead of ‘of’ in Wireshark

● OpenFlow protocol fields will have different names

1. Run a mininet linear network consisting of two switches, two hosts, and your OpenDaylight controller as in this diagram below (A good tour of Mininet’s options is available at <http://mininet.org/walkthrough/>).



1. Run a simple web server on host h2 with this command at the mininet> prompt:

h2 python -m SimpleHTTPServer 80 &

1. Run Wireshark on one of the switches to capture all of the packets that go between the hosts h1 & h2 through the switch, e.g.:

s1 wireshark &

and select either of the interfaces on either of the switches.

1. Make an HTTP request from h1 to the server running on h2:

h1 wget 10.0.0.2

1. Stop the Wireshark packet capture and examine all of the packets (except for the LLDP ones) involved in making the request. Look at the encapsulated packets and all header field values. Following this course’s tour down the network protocol stack you should now understand more of what you see in a Wireshark packet capture.
2. Perform steps 1 to 5 again, but this time with the goal of capturing the OpenFlow packets between the switch and the OpenDaylight controller – ensure that you run wireshark on the correct system and the appropriate interface in order to see the OpenFlow traffic, and not the web page request/response traffic from steps 1 to 5. Note the version of openflow that is needed to filter the packets in Wireshark.

## B. Extra Credit

1. Build a complicated mininet network that contains loops among the switches, e.g.:  
     
   sudo mn --topo torus,3,3 --mac \  
    --controller=remote,ip=192.168.56.101,port=6633 \  
    --switch ovs,protocols=OpenFlow13
2. View the Topology in OpenDaylight’s web-based visualization tool.
3. Make the hosts known to the OpenDaylight controller by executing a ‘pingall’ at the mininet> prompt.
4. Reload the Topology view in the visualization tool. Note the difference.
5. Try to create a larger torus network.
6. Repeat steps 1-4, but this time using mininet’s built in reference controller instead of the OpenDaylight controller:  
     
   sudo mn --topo torus,3,3 --mac \  
    --switch ovs,protocols=OpenFlow13

## C. What to turn in

1. Turn in the screen shots for each of Part A steps 1 to 6. [80 points]
2. Answer these questions:
   1. In Part A step 6 what system (i.e. at what prompt did you start wireshark) and interface did you use to see the OpenFlow traffic? What version of openflow did you need to select in Wireshark to filter the traffic to see just OpenFlow packets? [20 points]
   2. For Part B what is the largest torus network you can make? Include a screenshot of it. [Extra Credit: 15 points]
   3. For Part B step 6 state a possible reason why the results might be different between Mininet’s reference controller and OpenDaylight’s controller. [Extra Credit: 15 points]