Lab 6: Network Monitoring with SDN

# Xinmei Jin, NetID: xj573

# Zhaoxi Chen, NetID: zc1134

# Objectives

* Learn developing SDN traffic monitoring applications.
* Gain real time traffic measurement experience.
* Manage flows in real-time based on traffic measurement.

# Equipment

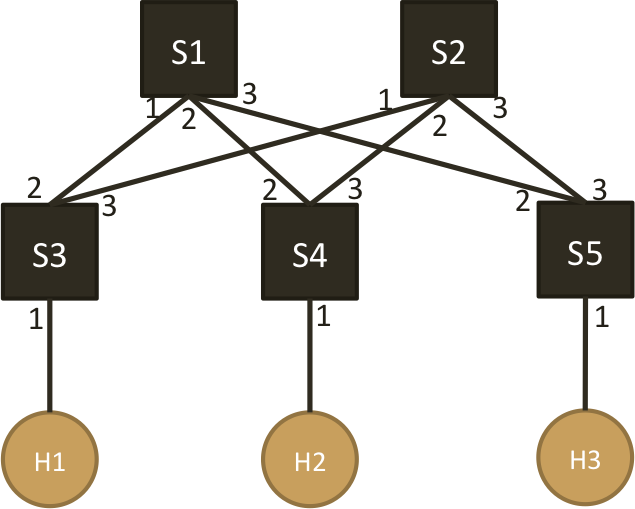
* Computer with Internet Connectivity.

# References

* RYU programming guide: <https://osrg.github.io/ryu-book/en/html/>

# Experiments

* 1. Go through the traffic monitoring example provided by RYU website: <https://osrg.github.io/ryu-book/en/html/traffic_monitor.html>
  2. Create the following topology using the topology file provided on NYU Classes to create 5 switches and 3 hosts. The topology .py file also includes a costumed command: “runTraffic” to use in the Mininet prompt. This command generates UDP traffic among H1, H2 and H3 for 10 mins.



* 1. Write two controllers to,
     1. Controller 1: When a new flow arrives, always manage the flow to follow the paths as instructed below,   
         H1 ➔ H2: S3 – S1 – S4   
         H1 ➔ H3: S3 – S1 – S5  
         H2 ➔ H3: S4 – S1 – S5  
         Measure (every 5 or 10 seconds) the traffic rate (in bps) of each link for 10 mins and draw a figure of ( time vs. traffic rate ) for each link
     2. Controller 2: Use the measured traffic (every 5 or 10 seconds) to manage flows so that all the links in the network is as evenly loaded as possible. Flow management can be done through either arranging new arriving flow’s path or re-locating existing flows. Draw a figure of ( time vs. traffic rate ) for each link.
  2. Start the controller, and run the “runTraffic” command in Mininet prompt to monitor the traffic and manage flows for 10 mins.

# Reports

1. **Explain your approach used in Controller 2 to evenly distribute the traffic onto different links.**

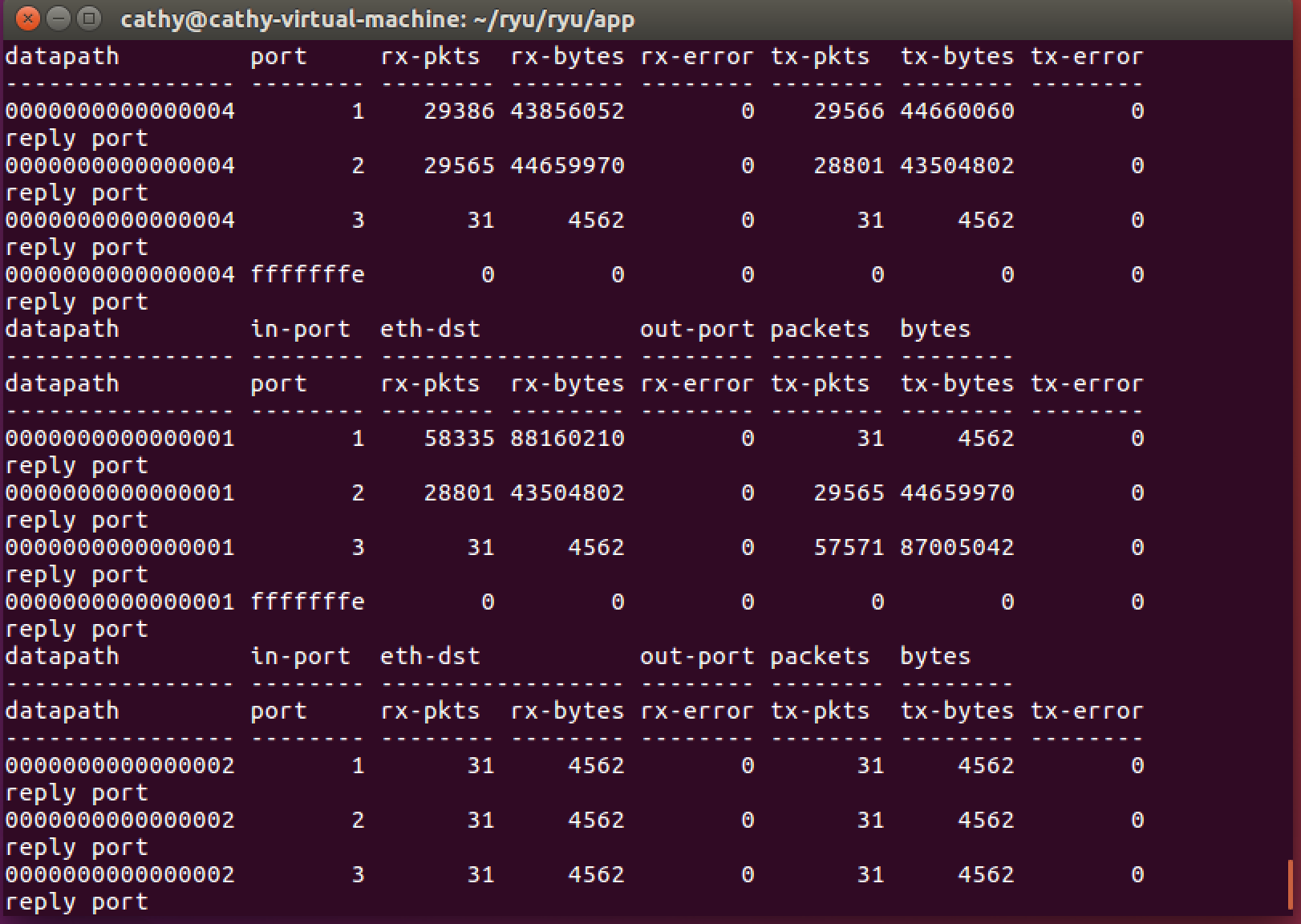
Record bytes for both ports. If the received and transmitted bytes is in port2 is larger than port3, designate packet to port 3, and if the received and transmitted bytes is in port3 is larger than port2, designate packet to port 2.

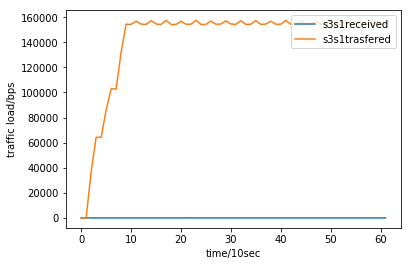
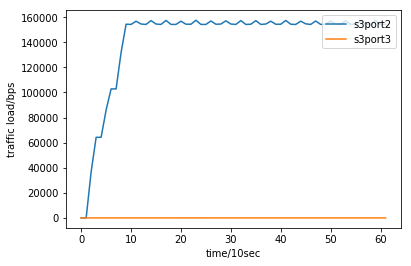
1. **Submit your controller codes for both controller 1 and controller 2**

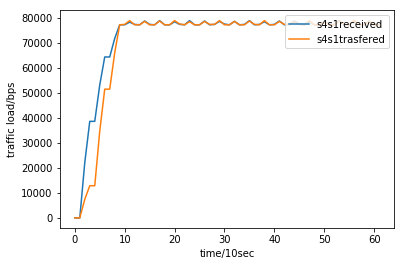
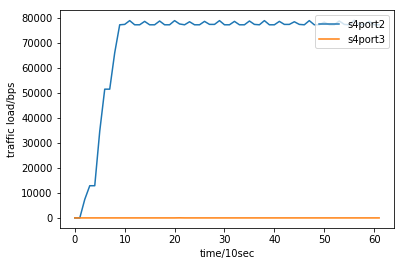
See in attachments controller1.py and controller2.py.

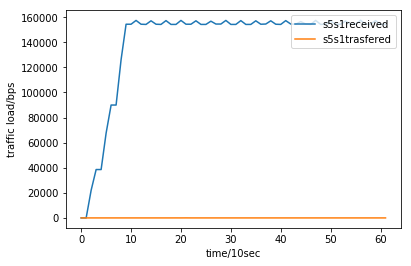
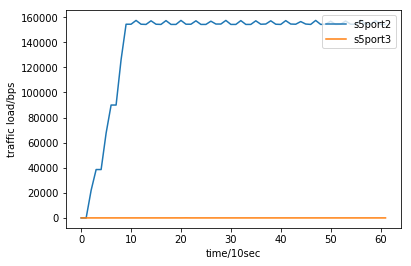
1. **Figures of ( Time vs. Traffic Rate (bps) ) on each link over 10 mins from both controller 1 and controller 2**

**i) Controller 1:**

****

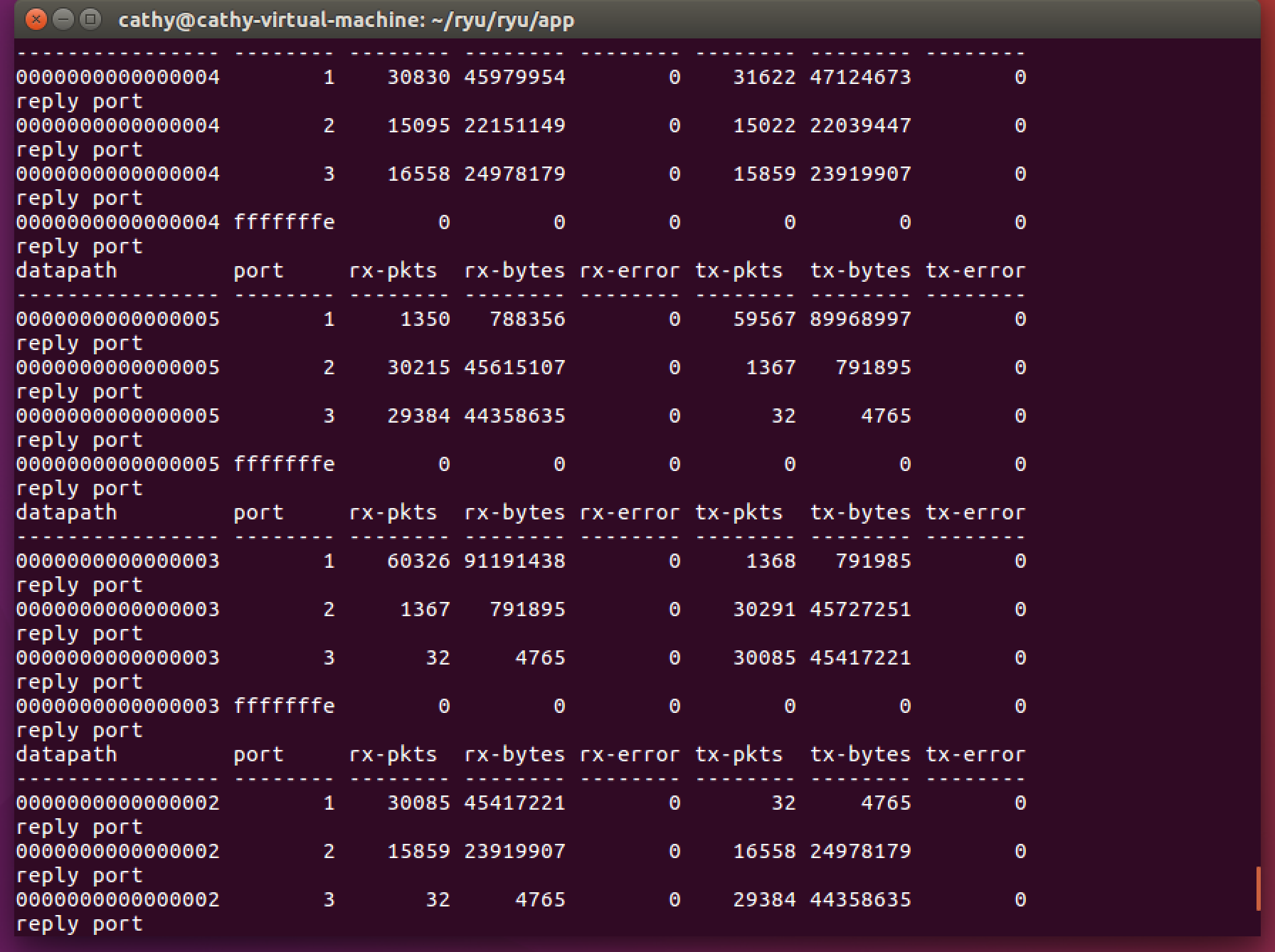
****

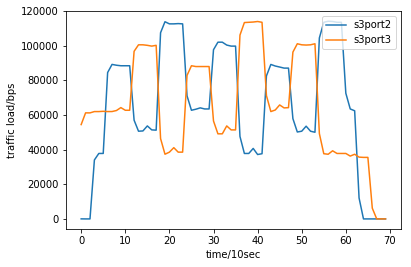


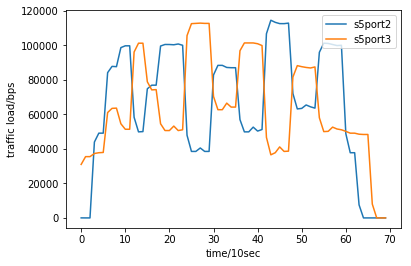
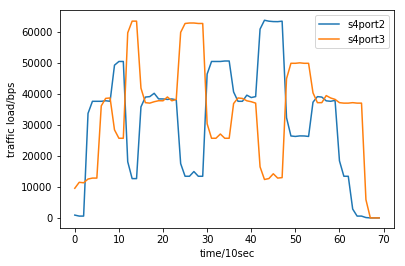


There is almost no flow goes through port3, since controller leads all flow trasfer through S1. All flow use port2, the rate goes up as traffic start to enter and the rate is roughly constant in the middle.

**ii) Contorller 2:**

****

****

****

Both ports have alternative peaks and valleys in rate-time chart. When port2 reach peak, port3 reach valley. When

port2 reach valley, port3 reach peak.

**We have zero tolerance to forged or fabricated data!!** A single piece of forged/fabricated data would bring the total score down to zero.