**Harita Shroff**

**CMPE 210**

**Prof.Younghee Park**

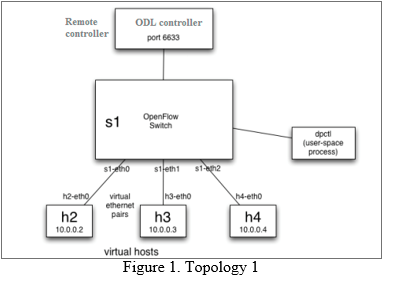
**Date:02/11/2018**

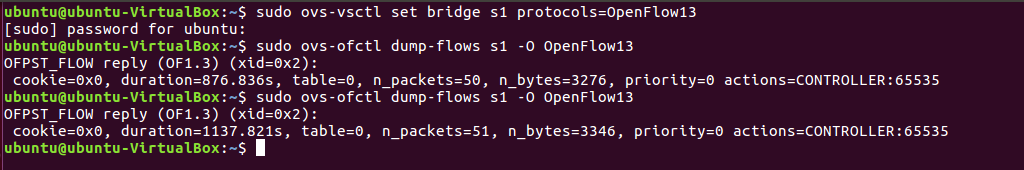
**LAB#1**

**Q1. [20 points] Install Mininet to explore Software Defined Networking(SDN) based on**

**the manual on Canvas. Create the topology in Figure 1 and print out the flow rules for the**

**topology. Explain each flow rules that you capture.**

****

****

Explanation: In the above flow rule, that I have captured,

cookie=0x0 is the opaque controller-issued identifier. This field is mainly used for MODIFY or DELETE flow rules, as this is a new flow value is set to 0 (OFPFC\_ADD).

Duration=1137.821s is the duration time flow has been alive in seconds

table=0 This is the table id from where flow came from.

n\_packets=51 this is the number of packets in the flow.

n\_bytes=3346 this is the number of bytes in the flow.

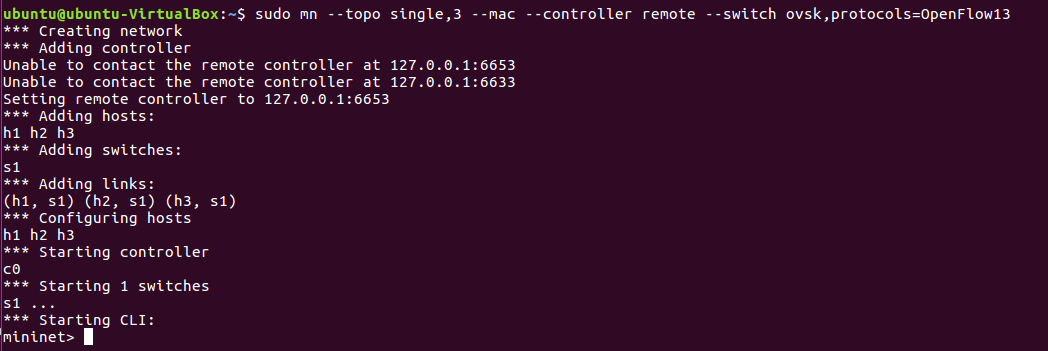
priority=0 Priority of the entry. This is used when there is not an exact-match entry.

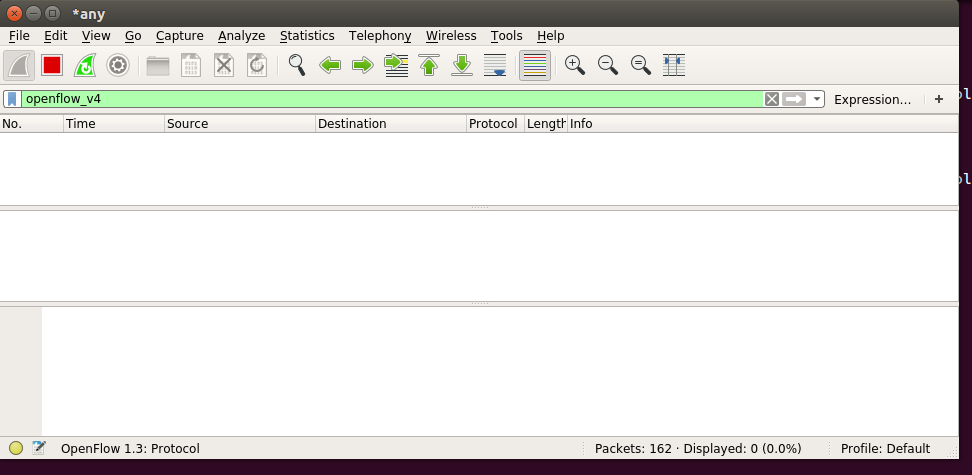
actions=CONTROLLER:65535 this indicates that forwarding will be done to the controller and first 65535 bytes will be sent to the controller.

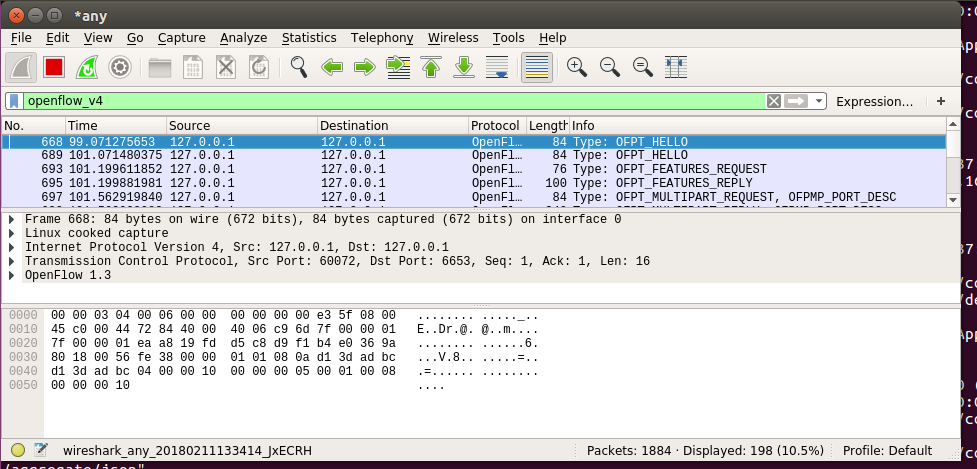
**Q2. [50 points] Using the Mininet instance with Figure 1, run the OpenFlow 1.3 tutorial located at the following website address:**

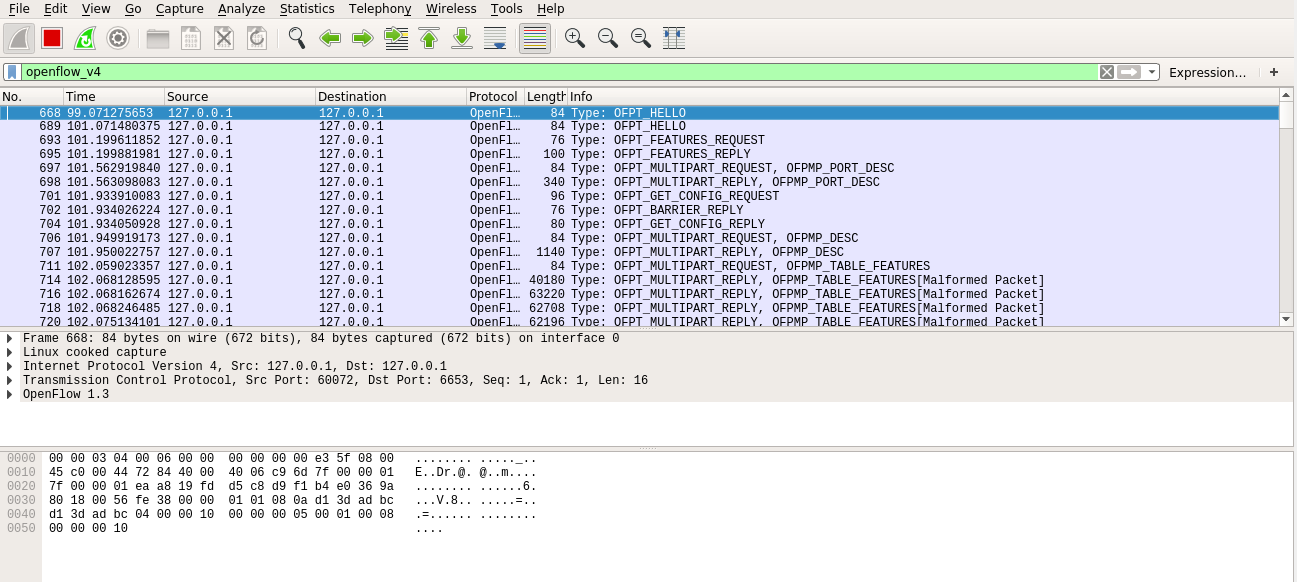
**<http://sdnhub.org/tutorials/openflow-1-3/>**

**Take screen shots of the terminal after executing the commands in the tutorial. This will account for 80% of the grade. 20% of the grade will be a write-up by the student describing any issues that they ran across in the tutorial.**

****

****

****

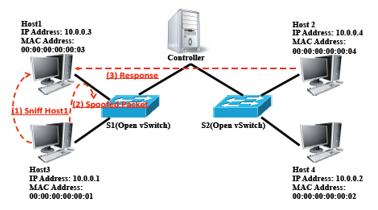
****

**Issue:** First, I started running floodlight controller and then started wireshark, So I was not able to see all the OpenFlow packets.

**Solution:**

I stopped the floodlight controller and started wireshark. Then I started floodlight controller.

Wireshark had captured all the OpenFlow packets. I was able to all the OpenFlow packets.

****

**Figure 2. Topology 2**

**Q3. [30 points] We can create the topology 2 in Figure 2 with the following settings. Please explain the installed network topology while explaining the two options: “tag” and “trunk”. Please print the flow rules of S1 and S2 for each and explain them.**

**#adding tags to the port of the switches configures them to belong to the corresponding vlan**

**ovs-vsctl set port s1-eth1 tag="100"**

**ovs-vsctl set port s2-eth2 tag="100"**

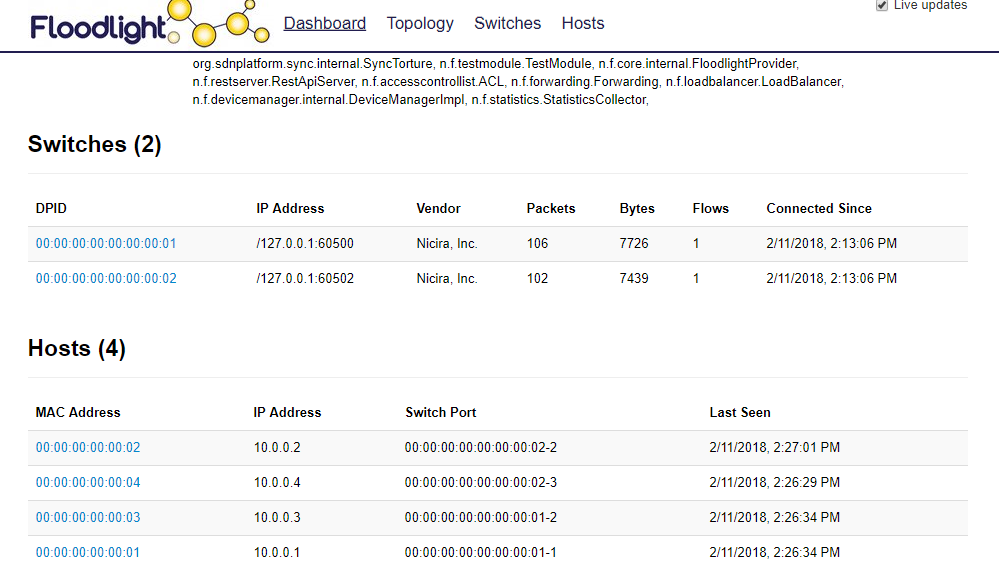
**ovs-vsctl set port s1-eth2 tag="200"**

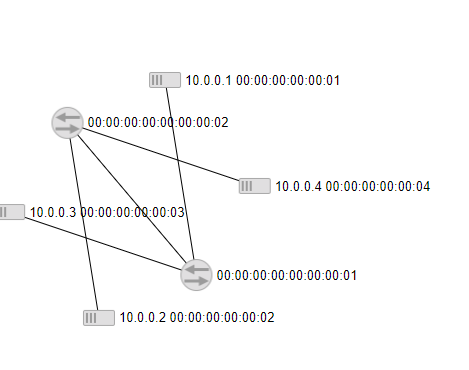
**ovs-vsctl set port s2-eth3 tag="200"**

**#Setting the ports of switches s1 and s2 connected to each other as a trunk port**

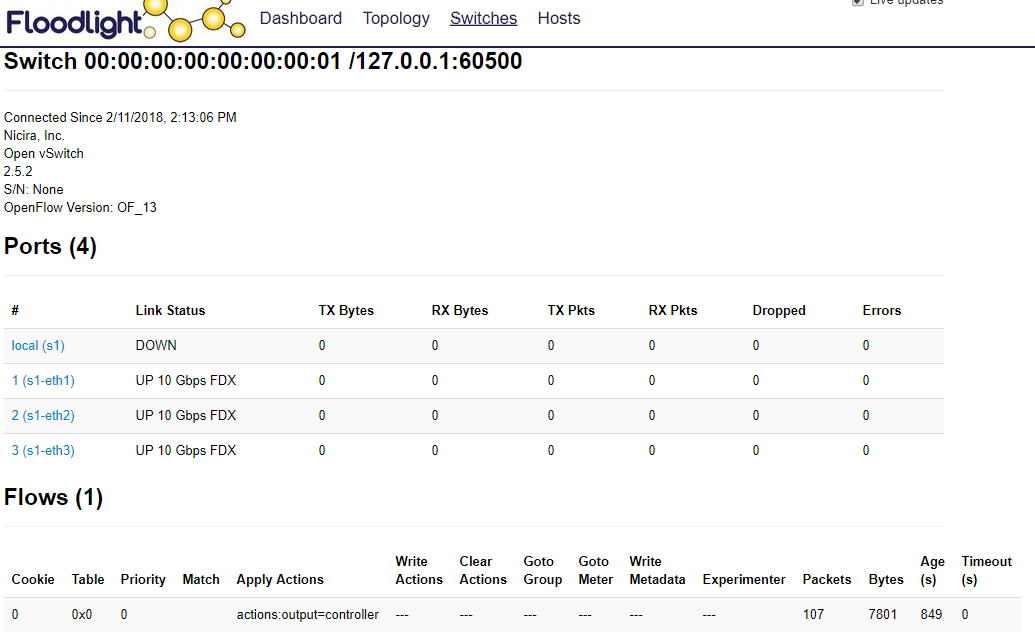
**ovs-vsctl set port s1-eth3 trunk="100,200"**

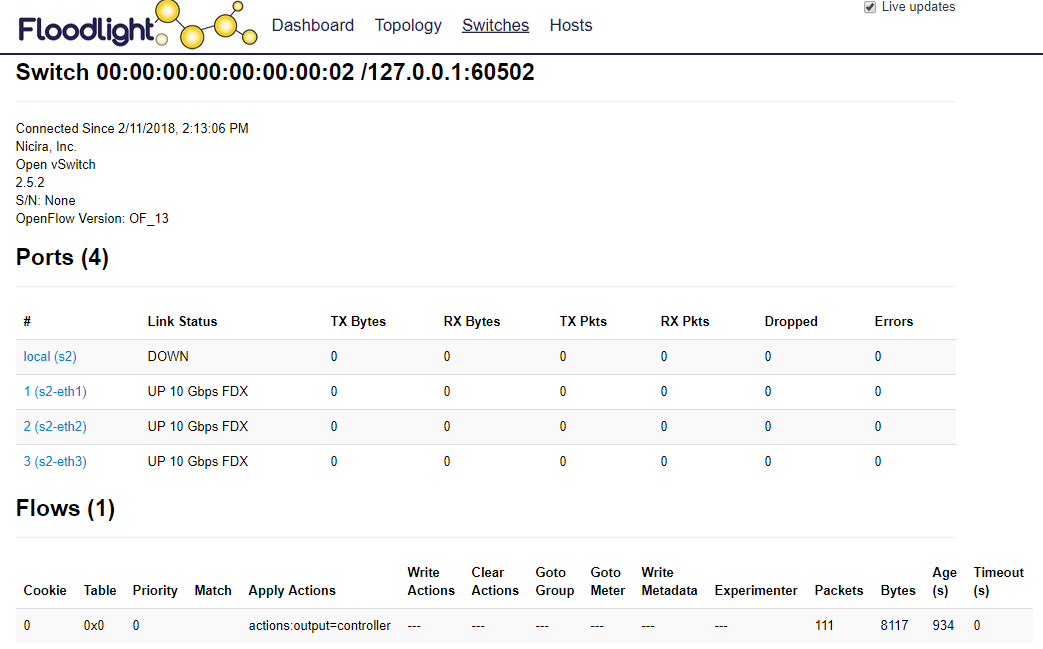
**ovs-vsctl set port s2-eth1 trunk="100,200"**

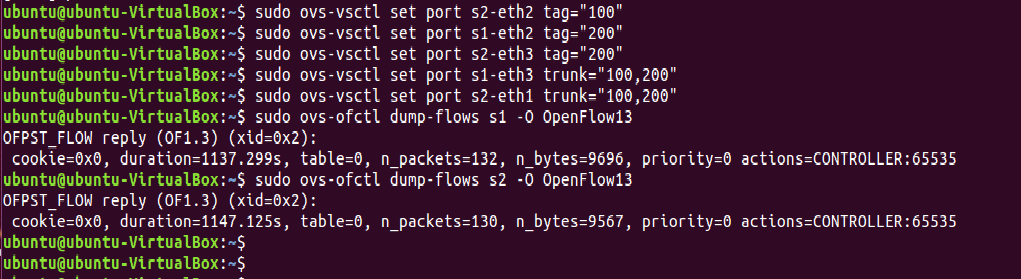
****

****

****

****

****

****

**Explanation:**

In this question, the below topology is used.

There are two switches and four hosts in the topology. The switch S1 has hosts H1 and H2 connected to it while the switch S2 has hosts H3 and H4 connected to it.

Both switches are using VLAN functionality to differentiate traffic between the hosts on the same switch. For S1, eth1 is tagged as VLAN-100 which means that each packet will be tagged with VLAN header 100. For S2, eth2 is tagged with same VLAN 100 and each packet from H3 will be tagged with VLAN 100 header. This means that H1 and H3 are on the same VLAN.

Similarly, S1, eth2 and S2, eth3 is tagged as VLAN 200 and each packet from this ports will be tagged with this VLAN header. This means that H2 and H4 are on the same VLAN.

While S1, eth3 and S2, eth1 is connected to the controller, they are allowing multiple VLAN tags through this port. S1 has 100 and 200 VLAN tags for two different ports, for this tagged packets to reach controller, the port must be marked trunk and should allow both VLAN tags through this port. Hence, S1 and S2 both have ports marked trunk to allow multiple VLAN tags to reach the controller.

Tag: This option is used to tag a switch port with specific VLAN Id. This option adds a VLAN header with a specific VLAN id to the packet.

Trunk: This option is used when multiple VLAN ID has to be allowed through a switch port. Trunk switch port can allow multiple VLAN IDs through them.

**Flow rule for the switch 1:**

cookie=0x0 is the opaque controller-issued identifier. This field is mainly used for MODIFY or DELETE flow rules, as this is a new flow value is set to 0 (OFPFC\_ADD).

Duration=1137.299s is the duration time flow has been alive in seconds

table=0 This is the table id from where flow came from.

n\_packets=132 this is the number of packets in the flow.

n\_bytes=9696 this is the number of bytes in the flow.

priority=0 Priority of the entry. This is used when there is not an exact-match entry.

actions=CONTROLLER:65535 this indicates that forwarding will be done to the controller and first 65535 bytes will be sent to the controller.

**Flow rule for the Switch 2:**

cookie=0x0 is the opaque controller-issued identifier. This field is mainly used for MODIFY or DELETE flow rules, as this is a new flow value is set to 0 (OFPFC\_ADD).

Duration=1147.125s is the duration time flow has been alive in seconds

table=0 This is the table id from where flow came from.

n\_packets=138 this is the number of packets in the flow.

n\_bytes=9567 this is the number of bytes in the flow.

priority=0 Priority of the entry. This is used when there is not an exact-match entry.

actions=CONTROLLER:65535 this indicates that forwarding will be done to the controller and first 65535 bytes will be sent to the controller.

**Q4. [20 points] Please explain the difference between Openflow 1.0.0 and Openflow 1.3.1.**

**OpenFlow 1.0.0:**

* In the Switch Hardware, there is only one flow table in the forwarding plane.
* Last field in the flow table is an Action.
* It can only have main single connection.
* Main connection between switch and controller is authenticated by exchanging certificates.
* Controller only has master role.

**OpenFlow1.3.1:**

* In the switch hardware, there are multiple flow tables based on the priority in the forwarding plane.There is also a Meter table and a Group table in the forwarding plane.
* Last field in the flow table is an Instructions.
* It can have multiple auxiliary connections while still have single connection.
* Auxiliary connections are not established using certificates, only main connection uses this method.
* Controller has multiple roles like master or slave.controller can change roles from master to slave or slve to master based on role change request