

Lab 1

1. See achen81.topo.py script that is attached to the Lab 1 submission
2. After inputting the dump command, the output shows which IP addresses has been assigned to h1 through h6 and a network interface eth0 which is connected to the switch on interface eth1. Also the three switches, s1, s2, s3, are shown to have the network interface eth1 that are connected with the interface eth3.

```
mininet@mininet-vm:~/Desktop$ sudo python achen81.topo.py
mininet> dump
<Host h1: h1-eth0:10.0.0.1 pid=2494>
<Host h2: h2-eth0:10.0.0.2 pid=2498>
<Host h3: h3-eth0:10.0.0.3 pid=2500>
<Host h4: h4-eth0:10.0.0.4 pid=2502>
<Host h5: h5-eth0:10.0.0.5 pid=2504>
<Host h6: h6-eth0:10.0.0.6 pid=2506>
<OVSSwitch s1: lo:127.0.0.1,s1-eth1:None,s1-eth2:None,s1-eth3:None pid=2511>
<OVSSwitch s2: lo:127.0.0.1,s2-eth1:None,s2-eth2:None,s2-eth3:None pid=2514>
<OVSSwitch s3: lo:127.0.0.1,s3-eth1:None,s3-eth2:None,s3-eth3:None,s3-eth4:None
pid=2517>
<Controller c0: 127.0.0.1:6633 pid=2487>
mininet>
```

After inputting the pingall command, the output shows the reachability of each host, h1 through h6, to all of the other hosts.

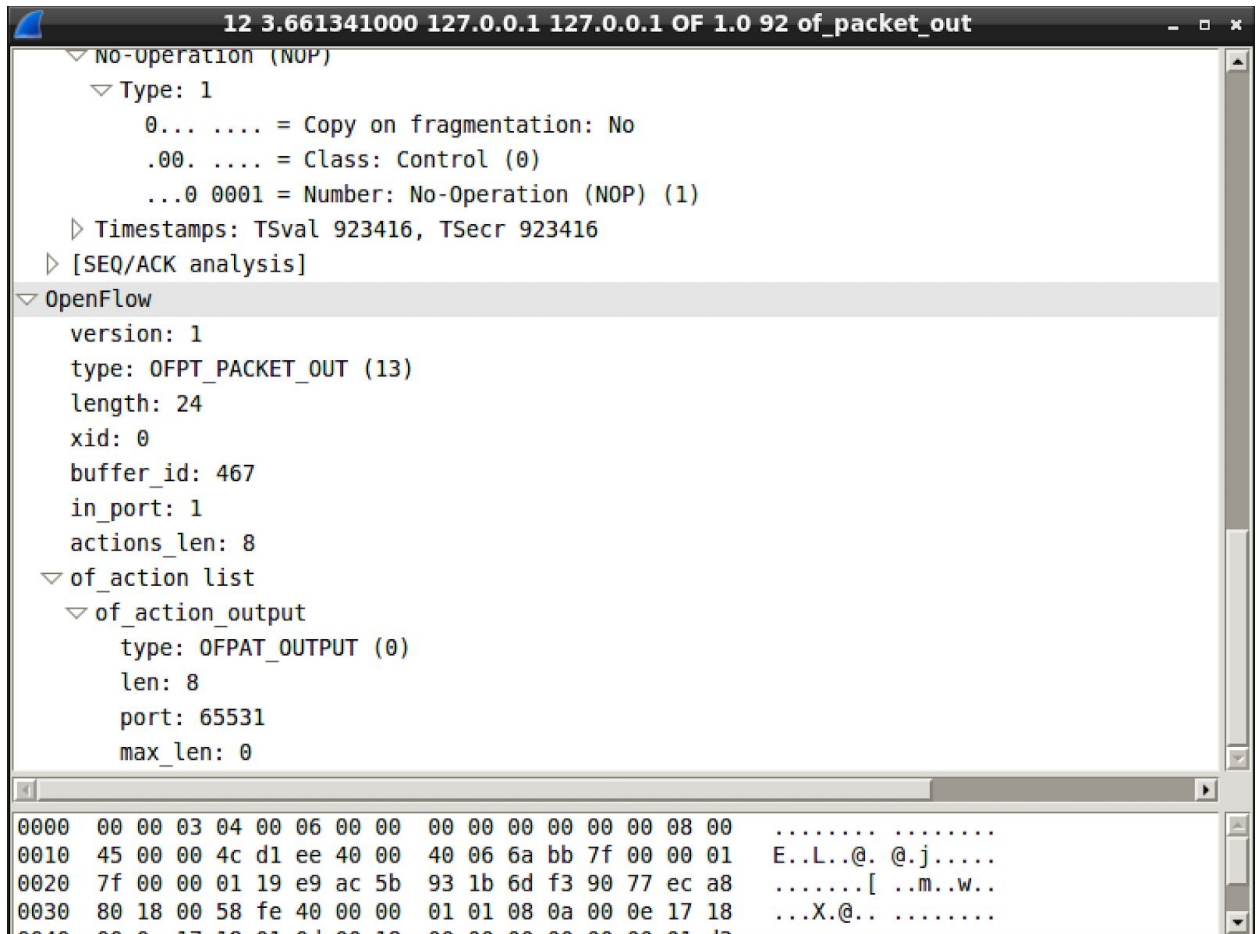
```
mininet> pingall
*** Ping: testing ping reachability
h1 -> h2 h3 h4 h5 h6
h2 -> h1 h3 h4 h5 h6
h3 -> h1 h2 h4 h5 h6
h4 -> h1 h2 h3 h5 h6
h5 -> h1 h2 h3 h4 h6
h6 -> h1 h2 h3 h4 h5
*** Results: 0% dropped (30/30 received)
mininet>
```

3. After running the iperf command, the connection between h1 and h6 is about 42.5 Gbits/sec.
4.
 - a. After running the ping command from h1 to h3, 15 of _packet_in messages showed up.

No.	Time	Source	Destination	Protocol	Length	Info
8	0.001250000	127.0.0.1	127.0.0.1	OF 1.0	76	of_echo_reply
11	3.661119000	10.0.0.1	10.0.0.4	OF 1.0	184	of_packet_in
12	3.661341000	127.0.0.1	127.0.0.1	OF 1.0	92	of_packet_out
18	3.661421000	10.0.0.1	10.0.0.4	OF 1.0	184	of_packet_in
19	3.661535000	127.0.0.1	127.0.0.1	OF 1.0	92	of_packet_out
26	3.661593000	10.0.0.1	10.0.0.4	OF 1.0	184	of_packet_in
27	3.661720000	127.0.0.1	127.0.0.1	OF 1.0	92	of_packet_out
33	3.661799000	10.0.0.4	10.0.0.1	OF 1.0	184	of_packet_in
34	3.661945000	127.0.0.1	127.0.0.1	OF 1.0	148	of_flow_add
37	3.662019000	10.0.0.4	10.0.0.1	OF 1.0	184	of_packet_in
38	3.662224000	127.0.0.1	127.0.0.1	OF 1.0	148	of_flow_add
41	3.662281000	10.0.0.4	10.0.0.1	OF 1.0	184	of_packet_in
42	3.662528000	127.0.0.1	127.0.0.1	OF 1.0	148	of_flow_add
48	4.662851000	10.0.0.1	10.0.0.4	OF 1.0	184	of_packet_in
49	4.663097000	127.0.0.1	127.0.0.1	OF 1.0	148	of_flow_add
53	4.663180000	10.0.0.1	10.0.0.4	OF 1.0	184	of_packet_in
54	4.663360000	127.0.0.1	127.0.0.1	OF 1.0	148	of_flow_add

No.	Time	Source	Destination	Protocol	Length	Info
58	4.663416000	10.0.0.1	10.0.0.4	OF 1.0	184	of_packet_in
59	4.663594000	127.0.0.1	127.0.0.1	OF 1.0	148	of_flow_add
105	8.666798000	2a:1a:f3:a3:f0:ab	76:44:94:3b:1c:d3	OF 1.0	128	of_packet_in
106	8.667045000	127.0.0.1	127.0.0.1	OF 1.0	148	of_flow_add
110	8.667176000	2a:1a:f3:a3:f0:ab	76:44:94:3b:1c:d3	OF 1.0	128	of_packet_in
111	8.667386000	127.0.0.1	127.0.0.1	OF 1.0	148	of_flow_add
115	8.667449000	2a:1a:f3:a3:f0:ab	76:44:94:3b:1c:d3	OF 1.0	128	of_packet_in
116	8.667629000	127.0.0.1	127.0.0.1	OF 1.0	148	of_flow_add
120	8.667708000	76:44:94:3b:1c:d3	2a:1a:f3:a3:f0:ab	OF 1.0	128	of_packet_in
121	8.667872000	127.0.0.1	127.0.0.1	OF 1.0	148	of_flow_add
124	8.667942000	76:44:94:3b:1c:d3	2a:1a:f3:a3:f0:ab	OF 1.0	128	of_packet_in
125	8.668110000	127.0.0.1	127.0.0.1	OF 1.0	148	of_flow_add
128	8.668164000	76:44:94:3b:1c:d3	2a:1a:f3:a3:f0:ab	OF 1.0	128	of_packet_in

- b. The source IP address for these entries is 10.0.0.1 and the destination IP address is 10.0.0.4. However, the source and destination IP address seem to be swapped at times. For the packet with the typefield set to OFPT_PACKET_OUT, the source IP address is 127.0.0.1 and the destination IP address is 127.0.0.1.



- c. After replacing the filter for “of” to “icmp && not of” and running pingall, 1102 packets are generated and the entry types consists of “100 Echo (ping) reply” and “100 Echo (ping) request”.

