Luis Valdivia - 1462385
levaldiv@ucsc.edu
Lab 3 - 11/20/19

# ***Lab Questions:***

1. Pingall: This should fail, since ICMP traffic should be blocked

```
mininet@mininet-vm:~$ sudo python ~/lab3.py
mininet> pingall
*** Ping: testing ping reachability
h1 -> X X X
h2 -> X X X
h3 -> X X X
h4 -> X X X
*** Results: 100% dropped (0/12 received)
mininet>
```

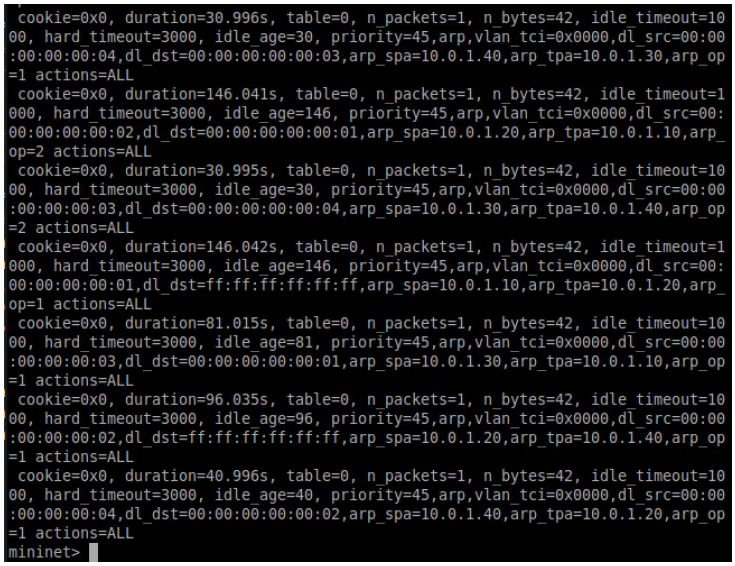
As expected, pingall fails for all transfers. In other words, my firewall allows ARP and TCP traffic to pass; however, any other type of traffic should be dropped. This is because pingall is an ICMPv4 protocol, while the filter only accepts TCP(/ipv4) and ARP connections, as stated above.

2. dpctl dump-flows: This should show a few entries. These are entries that you installed into the switch with “of_flow_mod.” You’ll need to do this within the timeout you specified in your “of_flow_mod” for the entries to show up

```
mininet> dpctl dump-flows
*** s1 ------------------------------------------------------------------------
NXST_FLOW reply (xid=0x4):
 cookie=0x0, duration=146.04s, table=0, n_packets=1, n_bytes=98, idle_timeout=1000, hard_timeout=3000, idle_age=146, icmp,vlan_tci=0x0000,dl_src=00:00:00:00:00:01,dl_dst=00:00:00:00:00:02,nw_src=10.0.1.10,nw_dst=10.0.1.20,nw_tos=0,icmp_type=8,icmp_code=0 actions=drop
 cookie=0x0, duration=126.043s, table=0, n_packets=1, n_bytes=98, idle_timeout=1000, hard_timeout=3000, idle_age=126, icmp,vlan_tci=0x0000,dl_src=00:00:00:00:00:01,dl_dst=00:00:00:00:00:04,nw_src=10.0.1.10,nw_dst=10.0.1.40,nw_tos=0,icmp_type=8,icmp_code=0 actions=drop
 cookie=0x0, duration=76.02s, table=0, n_packets=1, n_bytes=98, idle_timeout=1000, hard_timeout=3000, idle_age=76, icmp,vlan_tci=0x0000,dl_src=00:00:00:00:00:03,dl_dst=00:00:00:00:00:02,nw_src=10.0.1.30,nw_dst=10.0.1.20,nw_tos=0,icmp_type=8,icmp_code=0 actions=drop
 cookie=0x0, duration=66.018s, table=0, n_packets=1, n_bytes=98, idle_timeout=1000, hard_timeout=3000, idle_age=66, icmp,vlan_tci=0x0000,dl_src=00:00:00:00:00:03,dl_dst=00:00:00:00:00:04,nw_src=10.0.1.30,nw_dst=10.0.1.40,nw_tos=0,icmp_type=8,icmp_code=0 actions=drop
 cookie=0x0, duration=136.017s, table=0, n_packets=1, n_bytes=98, idle_timeout=1000, hard_timeout=3000, idle_age=136, icmp,vlan_tci=0x0000,dl_src=00:00:00:00:00:01,dl_dst=00:00:00:00:00:03,nw_src=10.0.1.10,nw_dst=10.0.1.30,nw_tos=0,icmp_type=8,icmp_code=0 actions=drop
 cookie=0x0, duration=45.994s, table=0, n_packets=1, n_bytes=98, idle_timeout=1000, hard_timeout=3000, idle_age=45, icmp,vlan_tci=0x0000,dl_src=00:00:00:00:00:04,dl_dst=00:00:00:00:00:02,nw_src=10.0.1.40,nw_dst=10.0.1.20,nw_tos=0,icmp_type=8,icmp_code=0 actions=drop
 cookie=0x0, duration=116.021s, table=0, n_packets=1, n_bytes=98, idle_timeout=1000, hard_timeout=3000, idle_age=116, icmp,vlan_tci=0x0000,dl_src=00:00:00:00:00:02,dl_dst=00:00:00:00:00:01,nw_src=10.0.1.20,nw_dst=10.0.1.10,nw_tos=0,icmp_type=8,icmp_code=0 actions=drop
 cookie=0x0, duration=86.008s, table=0, n_packets=1, n_bytes=98, idle_timeout=10
```

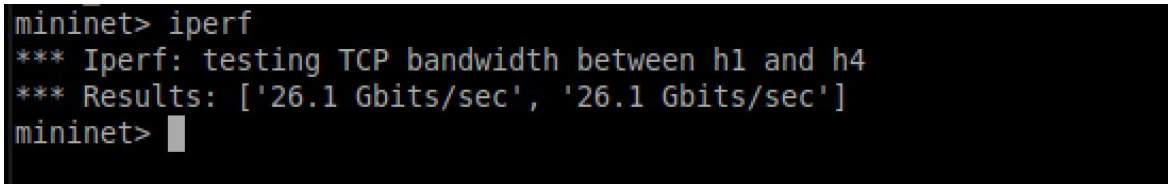
```
00, hard_timeout=3000, idle_age=86, icmp,vlan_tci=0x0000,dl_src=00:00:00:00:00:03,dl_dst=00:00:00:00:00:01,nw_src=10.0.1.30,nw_dst=10.0.1.10,nw_tos=0,icmp_type=8,icmp_code=0 actions=drop
 cookie=0x0, duration=56.043s, table=0, n_packets=1, n_bytes=98, idle_timeout=1000, hard_timeout=3000, idle_age=56, icmp,vlan_tci=0x0000,dl_src=00:00:00:00:00:04,dl_dst=00:00:00:00:00:01,nw_src=10.0.1.40,nw_dst=10.0.1.10,nw_tos=0,icmp_type=8,icmp_code=0 actions=drop
 cookie=0x0, duration=106.02s, table=0, n_packets=1, n_bytes=98, idle_timeout=1000, hard_timeout=3000, idle_age=106, icmp,vlan_tci=0x0000,dl_src=00:00:00:00:00:02,dl_dst=00:00:00:00:00:03,nw_src=10.0.1.20,nw_dst=10.0.1.30,nw_tos=0,icmp_type=8,icmp_code=0 actions=drop
 cookie=0x0, duration=35.99s, table=0, n_packets=1, n_bytes=98, idle_timeout=1000, hard_timeout=3000, idle_age=35, icmp,vlan_tci=0x0000,dl_src=00:00:00:00:00:04,dl_dst=00:00:00:00:00:03,nw_src=10.0.1.40,nw_dst=10.0.1.30,nw_tos=0,icmp_type=8,icmp_code=0 actions=drop
 cookie=0x0, duration=96.032s, table=0, n_packets=1, n_bytes=98, idle_timeout=1000, hard_timeout=3000, idle_age=96, icmp,vlan_tci=0x0000,dl_src=00:00:00:00:00:02,dl_dst=00:00:00:00:00:04,nw_src=10.0.1.20,nw_dst=10.0.1.40,nw_tos=0,icmp_type=8,icmp_code=0 actions=drop
 cookie=0x0, duration=126.043s, table=0, n_packets=1, n_bytes=42, idle_timeout=1000, hard_timeout=3000, idle_age=126, priority=45,arp,vlan_tci=0x0000,dl_src=00:00:00:00:00:04,dl_dst=00:00:00:00:00:01,arp_spa=10.0.1.40,arp_tpa=10.0.1.10,arp_op=2 actions=ALL
 cookie=0x0, duration=106.023s, table=0, n_packets=1, n_bytes=42, idle_timeout=1000, hard_timeout=3000, idle_age=106, priority=45,arp,vlan_tci=0x0000,dl_src=00:00:00:00:00:02,dl_dst=ff:ff:ff:ff:ff:ff,arp_spa=10.0.1.20,arp_tpa=10.0.1.30,arp_op=1 actions=ALL
 cookie=0x0, duration=111.023s, table=0, n_packets=1, n_bytes=42, idle_timeout=1000, hard_timeout=3000, idle_age=111, priority=45,arp,vlan_tci=0x0000,dl_src=00:00:00:00:00:01,dl_dst=00:00:00:00:00:02,arp_spa=10.0.1.10,arp_tpa=10.0.1.20,arp_op=2 actions=ALL
```

Luis Valdivia - 1462385
levaldiv@ucsc.edu
Lab 3 - 11/20/19




After I entered the pingall command, this was the result of dpctl dump-flows. These are the entries that I installed into the switch using “of_flow_mod.” I played around with the timeout settings and found out that if I had lower numbers, dump wouldn't show anything. But when I set timeout 1000 for idle and 3000 for hard, shown above were the result of using those numbers.

3. Iperf: this should succeed




As expected, the iperf command passes for the TCP connection. We can also see that we specify h1 and h4 because we want to rest the bandwidth between those two hosts. This is similar to pingall, except we have an exception for TCP connections to go through, so we get the above result with a successful transfer. The switch sees that the packet has a dl_type of 0x0800 and a nw_proto of 6 so it lets it through. Additionally, since the command iperf only works for TCP and UDP data streams, we get the results desired or the desired hosts.