

```
mininet@mininet-vm: ~  
File Edit Tabs Help  
*** Cleanup complete.  
mininet@mininet-vm:~$ sudo python example-topo.py  
mininet> net  
h1 h1-eth0:s1-eth1  
h2 h2-eth0:s2-eth3  
h3 h3-eth0:s1-eth3  
h4 h4-eth0:s2-eth1  
h5 h5-eth0:s3-eth1  
h6 h6-eth0:s3-eth4  
s1 lo: s1-eth1:h1-eth0 s1-eth2:s3-eth2 s1-eth3:h3-eth0  
s2 lo: s2-eth1:h4-eth0 s2-eth2:s3-eth3 s2-eth3:h2-eth0  
s3 lo: s3-eth1:h5-eth0 s3-eth2:s1-eth2 s3-eth3:s2-eth2 s3-eth4:h6-eth0  
c0  
mininet> dump  
<Host h1: h1-eth0:10.0.0.1 pid=4280>  
<Host h2: h2-eth0:10.0.0.2 pid=4284>  
<Host h3: h3-eth0:10.0.0.3 pid=4286>  
<Host h4: h4-eth0:10.0.0.4 pid=4288>  
<Host h5: h5-eth0:10.0.0.5 pid=4290>  
<Host h6: h6-eth0:10.0.0.6 pid=4292>  
<OVSSwitch s1: lo:127.0.0.1,s1-eth1:None,s1-eth2:None,s1-eth3:None pid=4297>  
<OVSSwitch s2: lo:127.0.0.1,s2-eth1:None,s2-eth2:None,s2-eth3:None pid=4300>  
<OVSSwitch s3: lo:127.0.0.1,s3-eth1:None,s3-eth2:None,s3-eth3:None,s3-eth4:None pid=4303>  
<Controller c0: 127.0.0.1:6633 pid=4273>  
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> iperf  
*** Iperf: testing TCP bandwidth between h1 and h6  
*** Results: ['28.3 Gbits/sec', '28.4 Gbits/sec']  
mininet>
```

2. Dump lists information about the nodes, switches, and controllers in the simulated network. We are given the IP addresses of each host as well as the pids of each process. Pingall displays the connectivity between all hosts and displays which hosts are connected to each other.

3. 2.83 Gbits/sec, 28.4 Gbits/sec

4.

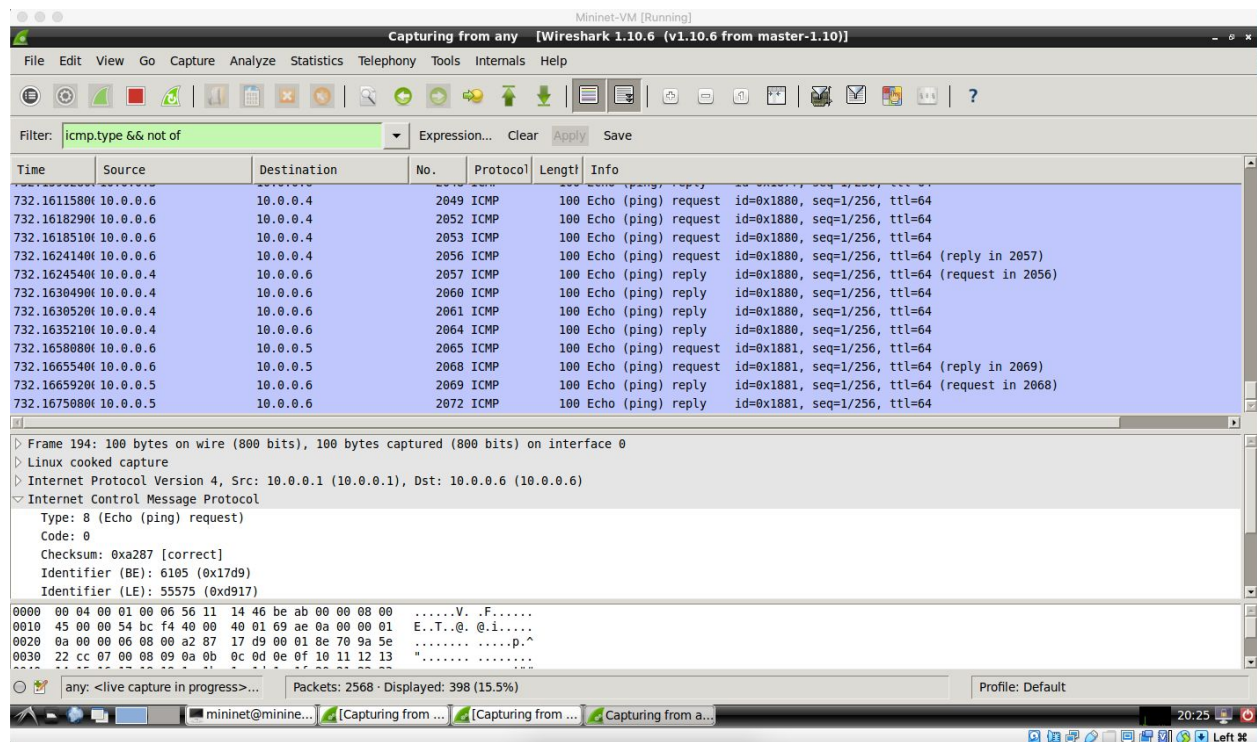
Time	Source	Destination	No.	Protocol	Length	Info
95.88801300	10.0.0.1	10.0.0.6	182	OF 1.0	184	of_packet_in
95.88884600	127.0.0.1	127.0.0.1	183	OF 1.0	92	of_packet_out
95.88919500	10.0.0.1	10.0.0.6	189	OF 1.0	184	of_packet_in
95.88953800	127.0.0.1	127.0.0.1	190	OF 1.0	92	of_packet_out
95.88988700	10.0.0.6	10.0.0.1	198	OF 1.0	184	of_packet_in
95.88991500	10.0.0.1	10.0.0.6	199	OF 1.0	184	of_packet_in
95.89027400	127.0.0.1	127.0.0.1	200	OF 1.0	92	of_packet_out
95.89200800	127.0.0.1	127.0.0.1	205	OF 1.0	148	of_flow_add
95.89247900	10.0.0.6	10.0.0.1	208	OF 1.0	184	of_packet_in
95.89280300	127.0.0.1	127.0.0.1	209	OF 1.0	148	of_flow_add
96.89032000	10.0.0.1	10.0.0.6	214	OF 1.0	184	of_packet_in
96.89086000	127.0.0.1	127.0.0.1	215	OF 1.0	148	of_flow_add
96.89124200	10.0.0.1	10.0.0.6	219	OF 1.0	184	of_packet_in
96.89151500	127.0.0.1	127.0.0.1	220	OF 1.0	148	of_flow_add
100.00032500	127.0.0.1	127.0.0.1	251	OF 1.0	76	of_echo_request
100.00093600	127.0.0.1	127.0.0.1	252	OF 1.0	76	of_echo_reply
100.89549600	3a:55:fd:a0:ac:88	56:11:14:46:be:ab	255	OF 1.0	128	of_packet_in
100.89682200	127.0.0.1	127.0.0.1	256	OF 1.0	148	of_flow_add
100.89741200	3a:55:fd:a0:ac:88	56:11:14:46:be:ab	260	OF 1.0	128	of_packet_in
100.89826000	127.0.0.1	127.0.0.1	261	OF 1.0	148	of_flow_add
100.89874800	56:11:14:46:be:ab	3a:55:fd:a0:ac:88	265	OF 1.0	128	of_packet_in
100.89910800	127.0.0.1	127.0.0.1	266	OF 1.0	148	of_flow_add
100.89947900	56:11:14:46:be:ab	3a:55:fd:a0:ac:88	269	OF 1.0	128	of_packet_in
100.89974900	127.0.0.1	127.0.0.1	270	OF 1.0	148	of_flow_add

0000 00 00 03 04 00 06 00 00 00 00 00 00 00 08 00
 any: <live capture in progress>... Packets: 363 · Displayed: 204 (56.2%) Profile: Default
 [Capturing from any [Wireshark]

A.) 11 of_packet_in show up

B.) The source is 10.0.0.1 / 10.0.0.6 as well as the destination. But for of_packet_out the destination is 127.0.0.1 as well as the destination

C.)



The kinds of entries are generated in wireshark after running pingall are Echo (ping) requests/ Echo (ping) replies