

3. The output from dump seems to show all of the users and the IPs that they are hosted on and the IPs that each switch is on. Pingall seems to test the connection between all of the hosts to see if they can all reach each other using the network.

```
mininet@mininet-vm:~/Desktop/cse150/lab1$ date
Mon Oct 11 18:24:45 PDT 2021
mininet@mininet-vm:~/Desktop/cse150/lab1$ sudo python mrgasser-topo.py
mininet> dump
<Host Ipad: Ipad-eth0:10.0.0.1 pid=3767>
<Host Laptop: Laptop-eth0:10.0.0.2 pid=3771>
<Host Phone: Phone-eth0:10.0.0.3 pid=3773>
<Host Server1: Server1-eth0:10.0.0.4 pid=3775>
<Host Server2: Server2-eth0:10.0.0.5 pid=3777>
<Host User1: User1-eth0:10.0.0.6 pid=3779>
<Host User2: User2-eth0:10.0.0.7 pid=3781>
<OVSSwitch Switch1: lo:127.0.0.1,Switch1-eth1:None,Switch1-eth2:None,Switch1-eth3:None,Switch1-eth4:None pid=3786>
<OVSSwitch Switch2: lo:127.0.0.1,Switch2-eth1:None,Switch2-eth2:None,Switch2-eth3:None,Switch2-eth4:None pid=3789>
<OVSSwitch Switch3: lo:127.0.0.1,Switch3-eth1:None,Switch3-eth2:None,Switch3-eth3:None pid=3792>
>
<Controller c0: 127.0.0.1:6633 pid=3760>
mininet> pingall
*** Ping: testing ping reachability
Ipad -> Laptop Phone Server1 Server2 User1 User2
Laptop -> Ipad Phone Server1 Server2 User1 User2
Phone -> Ipad Laptop Server1 Server2 User1 User2
Server1 -> Ipad Laptop Phone Server2 User1 User2
Server2 -> Ipad Laptop Phone Server1 User1 User2
User1 -> Ipad Laptop Phone Server1 Server2 User2
User2 -> Ipad Laptop Phone Server1 Server2 User1
*** Results: 0% dropped (42/42 received)
mininet>
```

4. The iperf command tests the connection between two random hosts. In this case the connection between Ipad and User2 was 44.4 Gbits/sec. This speed seems to make sense.

```
mininet@mininet-vm:~/Desktop/cse150/lab1$ date
Mon Oct 11 18:26:40 PDT 2021
mininet@mininet-vm:~/Desktop/cse150/lab1$ sudo python mrgasser-topo.py
mininet> iperf
*** Iperf: testing TCP bandwidth between Ipad and User2
*** Results: ['44.4 Gbits/sec', '44.4 Gbits/sec']
mininet>
```

5a. These packets seem to be larger so that they can test the connection as well as the speed of the connection.

```

419 135.989520m 127.0.0.1 127.0.0.1 0F 1.0 76 of echo reply 64 bytes from 10.0.0.2: icmp_seq=1 ttl=64 time=0.239 ms
420 135.989172m 127.0.0.1 127.0.0.1 0F 1.0 76 of echo reply 64 bytes from 10.0.0.2: icmp_seq=2 ttl=64 time=0.709 ms
421 135.989376m 127.0.0.1 127.0.0.1 0F 1.0 76 of echo reply 64 bytes from 10.0.0.2: icmp_seq=3 ttl=64 time=0.065 ms
422 135.989522m 127.0.0.1 127.0.0.1 0F 1.0 76 of echo reply 64 bytes from 10.0.0.2: icmp_seq=4 ttl=64 time=0.041 ms
423 137.344459m 7e:a2:44:ce:91:67 Broadcast 0F 1.0 128 of packet in 64 bytes from 10.0.0.2: icmp_seq=5 ttl=64 time=0.058 ms
428 137.344788m 127.0.0.1 127.0.0.1 0F 1.0 92 of packet out
436 137.345846m 7e:a2:44:ce:91:67 Broadcast 0F 1.0 128 of packet in
437 137.345862m d6:c1:15:78:e1:7c 7e:a2:44:ce:91:67 0F 1.0 128 of packet in
438 137.345272m 127.0.0.1 127.0.0.1 0F 1.0 92 of packet out
440 137.345353m 127.0.0.1 127.0.0.1 0F 1.0 148 of flow add
448 137.345498m 10.0.0.6 10.0.0.2 0F 1.0 184 of packet in
449 137.345766m 7e:a2:44:ce:91:67 Broadcast 0F 1.0 128 of packet in
450 137.345991m 127.0.0.1 127.0.0.1 0F 1.0 148 of flow add
453 137.346287m 127.0.0.1 127.0.0.1 0F 1.0 92 of packet out
458 137.346374m 10.0.0.2 10.0.0.6 0F 1.0 184 of packet in

--- 10.0.0.2 ping statistics ---
5 packets transmitted, 5 received, 0% packet loss, time 4004ms
rtt min/avg/max/mdev = 0.041/0.654/2.399/0.908 ms
mininet: UserJ ping -c 5 Laptop
PING 10.0.0.2 (10.0.0.2) 56(84) bytes of data.
64 bytes from 10.0.0.2: icmp_seq=1 ttl=64 time=2.39 ms
64 bytes from 10.0.0.2: icmp_seq=2 ttl=64 time=0.709 ms
64 bytes from 10.0.0.2: icmp_seq=3 ttl=64 time=0.065 ms
64 bytes from 10.0.0.2: icmp_seq=4 ttl=64 time=0.041 ms
64 bytes from 10.0.0.2: icmp_seq=5 ttl=64 time=0.058 ms

--- 10.0.0.2 ping statistics ---
5 packets transmitted, 5 received, 0% packet loss, time 4004ms
rtt min/avg/max/mdev = 0.041/0.654/2.399/0.908 ms
mininet: UserJ ping -c 5 Laptop
PING 10.0.0.2 (10.0.0.2) 56(84) bytes of data.
64 bytes from 10.0.0.2: icmp_seq=1 ttl=64 time=0.354 ms
64 bytes from 10.0.0.2: icmp_seq=2 ttl=64 time=0.043 ms
64 bytes from 10.0.0.2: icmp_seq=3 ttl=64 time=0.028 ms
64 bytes from 10.0.0.2: icmp_seq=4 ttl=64 time=0.033 ms
64 bytes from 10.0.0.2: icmp_seq=5 ttl=64 time=0.026 ms

--- 10.0.0.2 ping statistics ---
5 packets transmitted, 5 received, 0% packet loss, time 3997ms
rtt min/avg/max/mdev = 0.026/0.096/0.354/0.129 ms
mininet:

```

5b. Source: 127.0.0.1 Destination: 127.0.0.1

```

> Frame 437: 128 bytes on wire (1024 bits), 128 bytes captured (1024 bits) on interface 0
> Linux cooked capture
> Internet Protocol Version 4, Src: 127.0.0.1 (127.0.0.1), Dst: 127.0.0.1 (127.0.0.1)
> Transmission Control Protocol, Src Port: 57529 (57529), Dst Port: 6633 (6633), Seq: 797, Ack: 125, Len
> OpenFlow

```

The image shows a Wireshark packet capture window. The top pane displays the packet list with details for Frame 437, which is a TCP Reset (RST) packet. The bottom pane shows the raw data in hexadecimal and ASCII format.

Offset	Hex	ASCII
0000	00 00 03 04 00 06 00 00 00 00 00 00 00 00 08 00	.....
0010	45 c0 00 70 f6 11 40 00 40 06 45 b4 7f 00 00 01	E..p..@. @.E....
0020	7f 00 00 01 e0 b9 19 e9 32 00 97 43 96 ba f9 ce	..... 2..C....
0030	80 18 00 56 fe 64 00 00 01 01 08 0a 00 04 8e 5e	...V.d.. .....^

5c. ICMP stands for internet control message protocol which is used for reporting errors and performing network diagnostics. There are both request and replies.

No.	Time	Source	Destination	Protocol	Length	Info
447	137.34559600	10.0.0.6	10.0.0.2	ICMP	100	Echo (ping) request id=0x1cd3, seq=1/256, ttl=64
451	137.34605400	10.0.0.6	10.0.0.2	ICMP	100	Echo (ping) request id=0x1cd3, seq=1/256, ttl=64 (reply in 452)
452	137.34606400	10.0.0.2	10.0.0.6	ICMP	100	Echo (ping) reply id=0x1cd3, seq=1/256, ttl=64 (request in 451)
460	137.34662800	10.0.0.2	10.0.0.6	ICMP	100	Echo (ping) reply id=0x1cd3, seq=1/256, ttl=64
462	138.34636500	10.0.0.6	10.0.0.2	ICMP	100	Echo (ping) request id=0x1cd3, seq=2/512, ttl=64
463	138.34655100	10.0.0.6	10.0.0.2	ICMP	100	Echo (ping) request id=0x1cd3, seq=2/512, ttl=64 (reply in 464)
464	138.34659800	10.0.0.2	10.0.0.6	ICMP	100	Echo (ping) reply id=0x1cd3, seq=2/512, ttl=64 (request in 463)
465	138.34701000	10.0.0.2	10.0.0.6	ICMP	100	Echo (ping) reply id=0x1cd3, seq=2/512, ttl=64
466	139.34813200	10.0.0.6	10.0.0.2	ICMP	100	Echo (ping) request id=0x1cd3, seq=3/768, ttl=64
467	139.34815000	10.0.0.6	10.0.0.2	ICMP	100	Echo (ping) request id=0x1cd3, seq=3/768, ttl=64 (reply in 468)
468	139.34816500	10.0.0.2	10.0.0.6	ICMP	100	Echo (ping) reply id=0x1cd3, seq=3/768, ttl=64 (request in 467)
469	139.34816900	10.0.0.2	10.0.0.6	ICMP	100	Echo (ping) reply id=0x1cd3, seq=3/768, ttl=64
472	140.35001200	10.0.0.6	10.0.0.2	ICMP	100	Echo (ping) request id=0x1cd3, seq=4/1024, ttl=64
473	140.35002300	10.0.0.6	10.0.0.2	ICMP	100	Echo (ping) request id=0x1cd3, seq=4/1024, ttl=64 (reply in 474)
474	140.35003300	10.0.0.2	10.0.0.6	ICMP	100	Echo (ping) reply id=0x1cd3, seq=4/1024, ttl=64 (request in 473)

> Frame 468: 100 bytes on wire (800 bits), 100 bytes captured (800 bits) on interface 0  
 > Linux cooked capture  
 > Internet Protocol Version 4, Src: 10.0.0.2 (10.0.0.2), Dst: 10.0.0.6 (10.0.0.6)  
 > Internet Control Message Protocol

6i.

```

mininet@mininet-vm:~/Desktop/cse150/lab1$ date
Mon Oct 11 18:43:20 PDT 2021
mininet@mininet-vm:~/Desktop/cse150/lab1$ sudo python mrgasser-topo.py

mininet>
mininet> Laptop ping -c 5 Server1
PING 10.0.0.4 (10.0.0.4) 56(84) bytes of data.
64 bytes from 10.0.0.4: icmp_seq=1 ttl=64 time=180 ms
64 bytes from 10.0.0.4: icmp_seq=2 ttl=64 time=120 ms
64 bytes from 10.0.0.4: icmp_seq=3 ttl=64 time=83.1 ms
64 bytes from 10.0.0.4: icmp_seq=4 ttl=64 time=92.3 ms
64 bytes from 10.0.0.4: icmp_seq=5 ttl=64 time=89.5 ms

--- 10.0.0.4 ping statistics ---
5 packets transmitted, 5 received, 0% packet loss, time 4021ms
rtt min/avg/max/mdev = 83.113/113.326/180.786/36.130 ms
mininet> User1 ping -c 5 Ipad
PING 10.0.0.1 (10.0.0.1) 56(84) bytes of data.
64 bytes from 10.0.0.1: icmp_seq=1 ttl=64 time=129 ms
64 bytes from 10.0.0.1: icmp_seq=2 ttl=64 time=62.9 ms
64 bytes from 10.0.0.1: icmp_seq=3 ttl=64 time=63.6 ms
64 bytes from 10.0.0.1: icmp_seq=4 ttl=64 time=62.7 ms
64 bytes from 10.0.0.1: icmp_seq=5 ttl=64 time=63.2 ms

--- 10.0.0.1 ping statistics ---
5 packets transmitted, 5 received, 0% packet loss, time 4007ms
rtt min/avg/max/mdev = 62.713/76.452/129.648/26.600 ms
mininet>
  
```

6iii. 1. The connection times have increased slightly, probably due to the delay.

2. Interestingly the connection times seems to have decreased even with the delay being added.

```
mininet@mininet-vm:~/Desktop/cse150/lab1$ date
Mon Oct 11 19:03:20 PDT 2021
mininet@mininet-vm:~/Desktop/cse150/lab1$ sudo python mrgasser-topo.py
mininet> Laptop ping -c 5 Server1
PING 10.0.0.4 (10.0.0.4) 56(84) bytes of data.
64 bytes from 10.0.0.4: icmp_seq=1 ttl=64 time=204 ms
64 bytes from 10.0.0.4: icmp_seq=2 ttl=64 time=142 ms
64 bytes from 10.0.0.4: icmp_seq=3 ttl=64 time=105 ms
64 bytes from 10.0.0.4: icmp_seq=4 ttl=64 time=112 ms
64 bytes from 10.0.0.4: icmp_seq=5 ttl=64 time=139 ms

--- 10.0.0.4 ping statistics ---
5 packets transmitted, 5 received, 0% packet loss, time 4028ms
rtt min/avg/max/mdev = 105.133/140.918/204.696/35.071 ms
mininet> User1 ping -c 5 Laptop
PING 10.0.0.2 (10.0.0.2) 56(84) bytes of data.
64 bytes from 10.0.0.2: icmp_seq=1 ttl=64 time=117 ms
64 bytes from 10.0.0.2: icmp_seq=2 ttl=64 time=51.8 ms
64 bytes from 10.0.0.2: icmp_seq=3 ttl=64 time=51.4 ms
64 bytes from 10.0.0.2: icmp_seq=4 ttl=64 time=52.1 ms
64 bytes from 10.0.0.2: icmp_seq=5 ttl=64 time=51.3 ms

--- 10.0.0.2 ping statistics ---
5 packets transmitted, 5 received, 0% packet loss, time 4006ms
rtt min/avg/max/mdev = 51.332/64.789/117.133/26.175 ms
mininet> █
```