

Figure 1

- 2. Packets are forwarded differently in the accept() function for this lab because they require an end port and are therefore directed to a single host instead of being flooded to the ports of all hosts like in the last lab.
- 3. The screenshot of the pingall verifies that the rule #1 for ICMP is working as expected such that only communication between the faculty subnet, the IT subnet, and student subnet are allowed, as well as ICMP communication within the same subnet as observable through the University Data Center hosts being able to ping each other. Internet hosts are not part of a subnet in this network and therefore are blocked for all

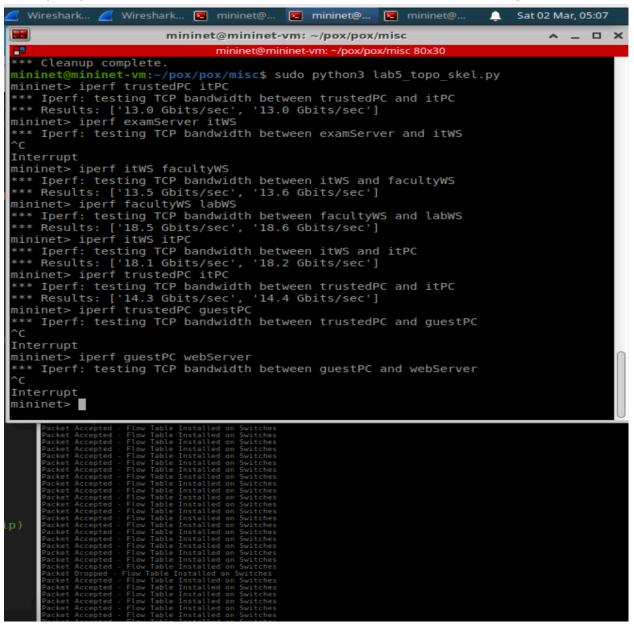
1.

pings on this network.

```
mininet@..
                                         mininet@...
                                                        mininet@.
                                                                            Sat 02 Mar, 04:47
                                                                                   _ D X
                       mininet@mininet-vm: ~/pox/pox/misc
pkill -9 -f mininet:
 ** Shutting down stale tunnels
pkill -9 -f Tunnel=Ethernet
pkill -9 -f .ssh/mn
rm -f ~/.ssh/mn/*
  ** Cleanup complete.
mininet@mininet-vm:~/pox/pox/misc$
mininet@mininet-vm:~/pox/pox/misc$ sudo python3 lab5_topo_skel.py
mininet> pingall
*** Ping: testing ping reachability
dnsServer -> examServer <u>X X X X X</u>
examserver -> dnsserver X X X X X X X webserver
facultyPC -> X X facultyWS X itPC itWS labWS printer studentPC X X facultyPC X itPC itWS labWS printer studentPC X X
*** Results: 63% dropped (48/132 receiv
mininet>
 ip
```

4. The screenshot shows that rule #2 for TCP packets is working as expected because iperf is working between all subnets and hosts within the same subnet, as well as with trustedPC, but it is not working between examServer and itWS because only hosts from the faculty subnet are allowed to send TCP packets to the exam server. It is also not

working for guestPC because the rule does not allow TCP packets to and from guestPC.



5. The individual results in the screenshot show that Rule #3 for UDP is working as expected such that all subnets can communicate with each other and hosts can communicate with each other within the same subnet. However, hosts on the internet cannot use iperfudp because the rule does not allow Internet hosts to send or receive

## UDP packets.

```
Sat 02 Mar, 05:32
                                                      Wireshark... 🗷 mininet@... 🔽 mininet@...
                                                                             mininet@mininet-vm: ~/pox/pox/misc
                                                                                                                                                                                                                                                                                          0 ×
 Interrupt
Interrupt
mininet> iperfudp 10M examServer itWS
*** Iperf: testing UDP bandwidth between examServer and itWS
*** Results: ['10M', '10.5 Mbits/sec', '10.5 Mbits/sec']
mininet> iperfudp 10M itWS facultyWS
*** Iperf: testing UDP bandwidth between itWS and facultyWS
*** Results: ['10M', '10.5 Mbits/sec', '10.5 Mbits/sec']
mininet> iperfudp 10M facultyWS labWS
*** Iperf: testing UDP bandwidth between facultyWS and labWS
*** Results: ['10M', '10.6 Mbits/sec', '10.6 Mbits/sec']
mininet> iperfudp 10M itWS itPC
**** Iperf: testing UDP bandwidth between itWS and itPC
**** Results: ['10M', '10.5 Mbits/sec', '10.5 Mbits/sec']
mininet> iperfudp 10M trustedPC itPC
**** Iperf: testing UDP bandwidth between trustedPC and itPC
^C
  ^C
Interrupt
 mininet> iperfudp 10M trustedPC guestPC
*** Iperf: testing UDP bandwidth between trustedPC and guestPC
 Interrupt
  mininet>
```

## Discord host communicates tcp, icmp, and udp only to and from student\_subnet

