Pre-lab Questions:

1. [5] What command will show you which groups you are a member of?
‘groups’

2. [5] What does the environmental variable “$?” hold? (Hint: the command ‘echo $?’ will show you this on your screen)
Holds return value of last executed command. Returns integer. Usually returns 0 for success and non zero for error.

3. [5] What key combination will suspend a currently running process and place it as a background process?
Control z and then bg. Control z suspends the running process and then bg runs it in the background.

4. [5] With what command (and arguments) can you find out your kernel version and the “nodename”? [The output should not include any other information]
uname -vn

5. [5] What is the difference between the paths “.”, “..”, and “~”? What does the path “/” refer to when not preceded by anything?
‘~’ is home directory. ‘.’ is current directory. ‘..’ is parent directory. ‘/’ is root directory.

6. [5] What is a pid? Which command would you use to find the “pid” for a running Process?
pid is a process id, unique number. ‘ps’ finds the “pid” for a running process.

7. [20] Write a single command that will return every user’s default shell. [You may chain commands using piping and redirects] (Hint: See ‘Chapter 19: filters’ of linux-training.be as well as the man page for the /etc/passwd file: https://linux.die.net/man/5/passwd)
‘getent passwd | cut -d: -f1,7’

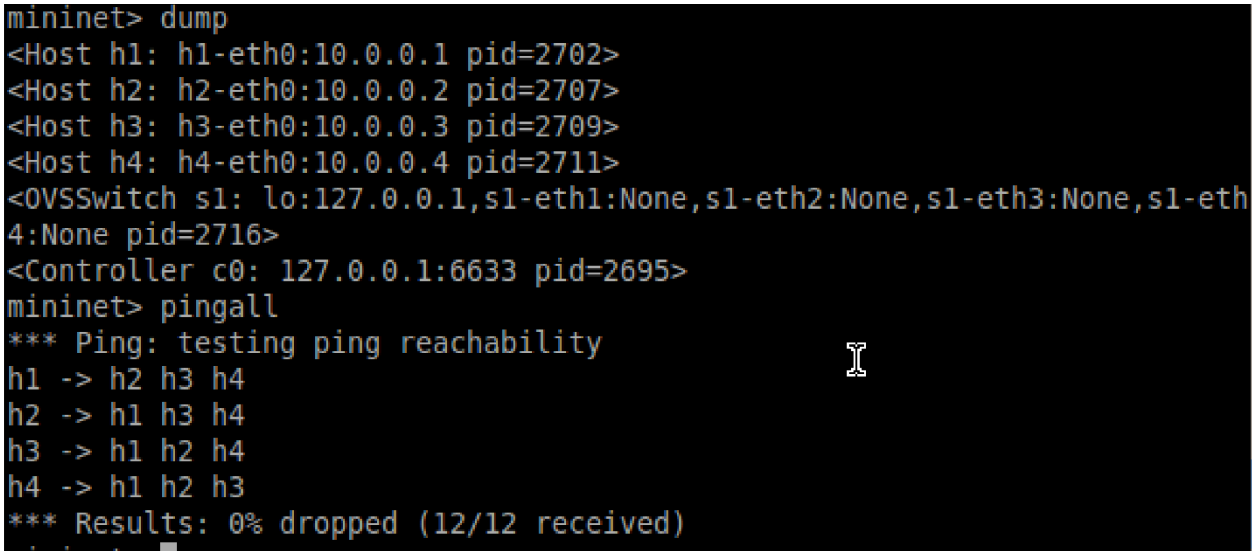
8. [10] What is the difference between “sudo” and “su root”?
su switches to root user and requires the root users password. sudo runs a single command with root privileges without needing the root users password.

9. [10] How would you tell your computer to run a program or script on a schedule or set interval on Linux? E.g. Run this program once every 30 minutes.
Use crontab. Use command ‘crontab -e’ and edit the crontab file. Add the entry in this format:
*/30 * * * * /path/to/your/script
For executing a program every 30 minutes.

*/30 → Every 30 minutes
* * * → Every hour, every day, every month, etc.

10. [30] Write a shell script that only prints the even numbered lines of each file in the current directory. The output should be filename: line for each even numbered line. You do not need to print line numbers.
Submitted file as CollinWen-script.sh

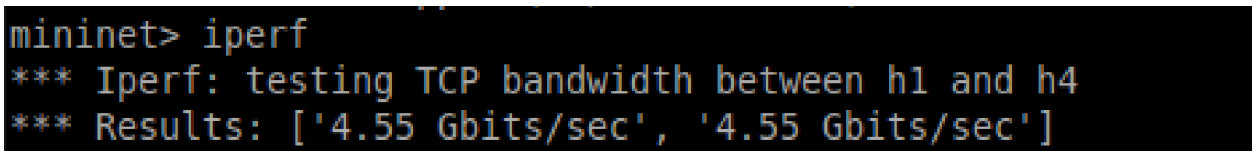
Lab Questions:
1. In Mininet change the default configuration to have 4 hosts connected to a switch.
2. [30 pts] Save a screenshot of dump and pingall output. Explain what is being shown in the screenshot.



Dump shows the IP addresses and pids of the hosts. Shows s1 is OVSSwitch with interfaces eth1 to eth4, all connected to one of the hosts and its pid.. Shows controller running locally at 127.0.0.1 on port 6633 and its pid.

pingall shows the network connectivity between all of the hosts. 12 ping attempts successful.

3. [10 pts] Run the iperf command as well, and screenshot the output, how fast is the Connect?



4.55 Gbits/sec

4. Run wireshark, and using the display filter, typing “openflow_v1” in filter line and

remember to click Apply after you start. Note: When you run wireshark you should do so as “sudo wireshark” in a new terminal. When you choose an interface to capture on, you should select “any”.

a. [20 pts] Run ping from a host to any other host using hX ping -c 5 hY. How many of_packet_in messages show up? Take a screenshot of your results.
I did h1 ping -c 5 h2

300(	127.0.0.1	127.0.0.1	OF 1.0	76 of_echo_reply
500(	10.0.0.1	10.0.0.2	OF 1.0	184 of_packet_in
700(	127.0.0.1	127.0.0.1	OF 1.0	92 of_packet_out
200(	10.0.0.2	10.0.0.1	OF 1.0	184 of_packet_in
900(	127.0.0.1	127.0.0.1	OF 1.0	148 of_flow_add
100(	10.0.0.1	10.0.0.2	OF 1.0	184 of_packet_in
900(	127.0.0.1	127.0.0.1	OF 1.0	148 of_flow_add
400(	be:a0:b7:56:35:ae	0a:c6:6a:4e:96:f0	OF 1.0	128 of_packet_in
100(	127.0.0.1	127.0.0.1	OF 1.0	148 of_flow_add
500(	0a:c6:6a:4e:96:f0	be:a0:b7:56:35:ae	OF 1.0	128 of_packet_in

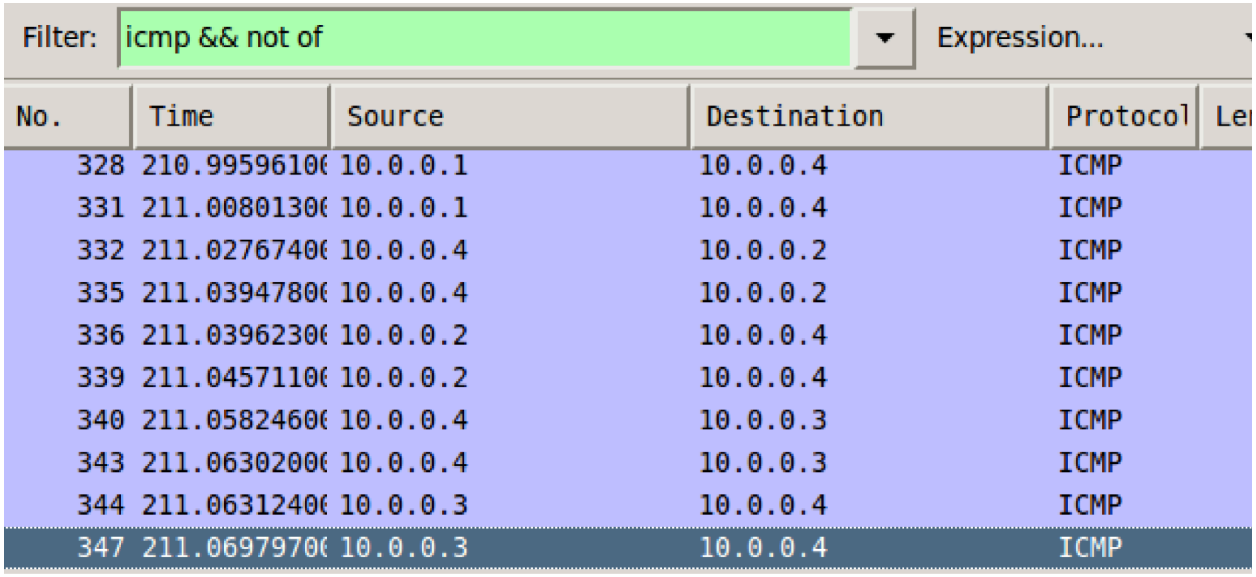
5

b. [20 pts] What is the source and destination IP addresses for these entries? Find another packet that matches the “of” filter with the OpenFlow typefield set to OF_PACKET_OUT. What is the source and destination IP address for this entry? Take screenshots showing your results.

Source Destination

127.0.0.1	127.0.0.1	OF 1.0	76 of_echo_reply
10.0.0.1	10.0.0.2	OF 1.0	184 of_packet_in
127.0.0.1	127.0.0.1	OF 1.0	92 of_packet_out
10.0.0.2	10.0.0.1	OF 1.0	184 of_packet_in
127.0.0.1	127.0.0.1	OF 1.0	148 of_flow_add
10.0.0.1	10.0.0.2	OF 1.0	184 of_packet_in
127.0.0.1	127.0.0.1	OF 1.0	148 of_flow_add
be:a0:b7:56:35:ae	0a:c6:6a:4e:96:f0	OF 1.0	128 of_packet_in
127.0.0.1	127.0.0.1	OF 1.0	148 of_flow_add
0a:c6:6a:4e:96:f0	be:a0:b7:56:35:ae	OF 1.0	128 of_packet_in

c. [20 pts] Replace the display filter for “of” to “icmp && not of”. Run pingall again, how many entries are generated in wireshark? What types of icmp entries show up? Take a screenshot of your results.
81 Entries


Filter: icmp && not of
Expression...
No. Time Source Destination Protocol Le
328 210.9959610(10.0.0.1 10.0.0.4 ICMP
331 211.0080130(10.0.0.1 10.0.0.4 ICMP
332 211.0276740(10.0.0.4 10.0.0.2 ICMP
335 211.0394780(10.0.0.4 10.0.0.2 ICMP
336 211.0396230(10.0.0.2 10.0.0.4 ICMP
339 211.0457110(10.0.0.2 10.0.0.4 ICMP
340 211.0582460(10.0.0.4 10.0.0.3 ICMP
343 211.0630200(10.0.0.4 10.0.0.3 ICMP
344 211.0631240(10.0.0.3 10.0.0.4 ICMP
347 211.0697970(10.0.0.3 10.0.0.4 ICMP