# Group Members:

Tinh Van Phan (301384974):

* Implemented from\_json (in app\_fw.py) and from\_json (in app\_te.py)
* Implemented provision\_by\_paths, run command 2, 3 and answered to questions accordingly.

# Command 1

Text

Description automatically generated

Here is the flow table for all switches after executing the L2 command as the operator.

Text

Description automatically generatedText

Description automatically generatedText

Description automatically generated

Text

Description automatically generated

The flow tables are now populated. Analyzing the flow tables, we see that the ‘actions’ is set to forward packets to the output port that will lead them towards their destination based on their destination MAC address. For example, we see that for s2, a destination MAC address of 00:00:00:00:00:02 is forwarded along s2-eth1, which corresponds to the host for switch 2. This host has MAC address 00:00:00:00:00:02. S4 sends packets with destination MAC address 00:00:00:00:00:02 along output port s4-eth3. This is the output port that shares a link with s2, so again, this checks out. In general, this flow table makes it so packets are routed along the shorted path (in terms of number of links used) to reach their destination.

# Command 2

Output:

Graphical user interface, text, application

Description automatically generated

OpenFlow rule used:

Text

Description automatically generatedText

Description automatically generatedText

Description automatically generated

From the command, we know that h3 is trying to connect to h1 via port 80.

From the OpenFlow rules, we notice that h3 forwarded 5 packets however, h1 will drop any UDP packet coming into port 80. There is only 1 packet being dropped at h1.

# Command 3

Screenshots of the OpenFlow rules that have been used to forward packets of the iperf3 command.

Text

Description automatically generatedText

Description automatically generatedText

Description automatically generated

# Command 4