

Assignment 3

Trinity College

Xinying (Ashlyn) Li

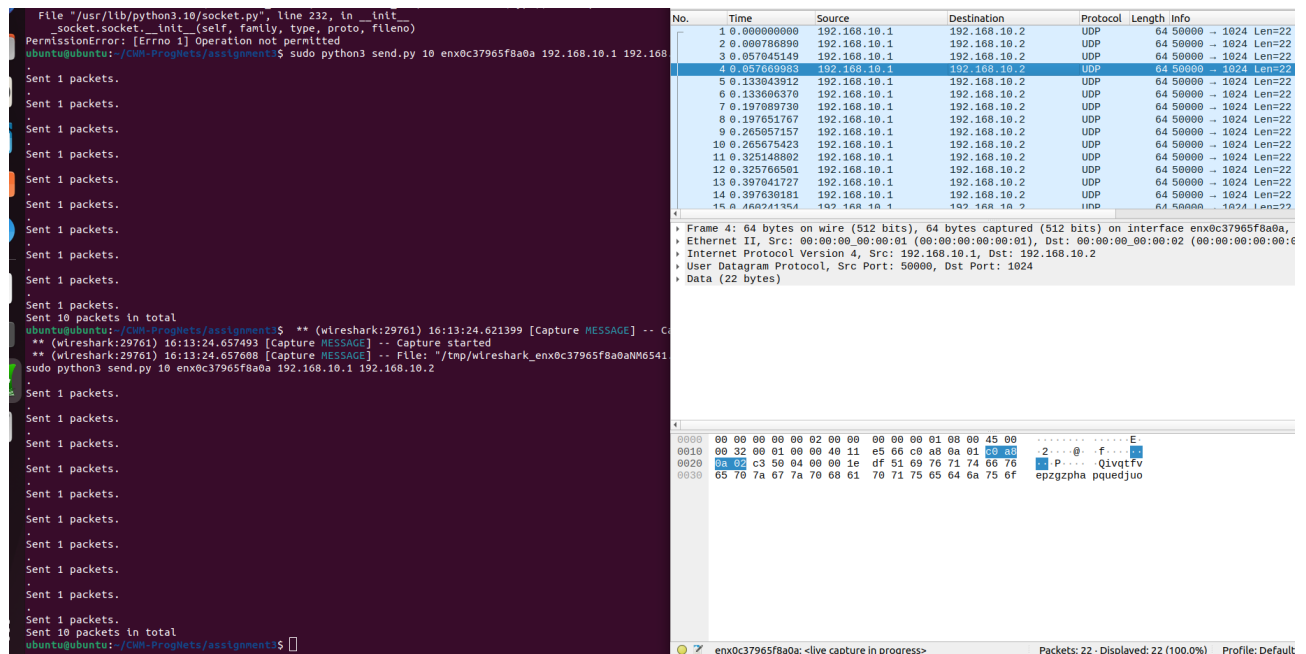


Figure 1 sending packets from the lab machine to the Raspberry Pi

The screenshot above shown the packets sent and ‘rebounded’ back for the lab machine. Even the destination and source shown on the tracking is unchanged (They are all from 192.168.10.1 to 192.168.10.2), from the bottom square we can track the change of the dsc_mac and dsc_src from the python code has swapped. The destination and the source address does not change since it’s just a simple switch and no ip change occurs.

Also, the wireshark captured 20 packets, which is exactly what we expected to get.

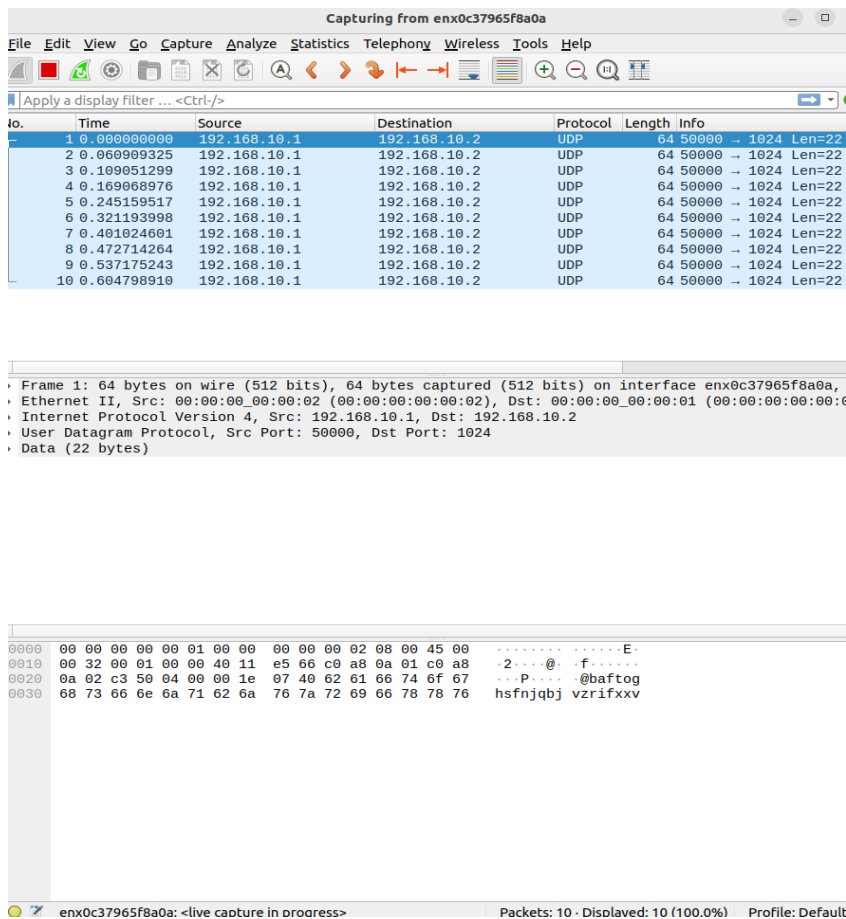


Figure 2. Traffic captured after adding table entry

```
RuntimeCmd: table_add MyIngress.src_mac_drop MyIngress.drop 00:00:00:00:00:02 =>
Adding entry to exact match table MyIngress.src_mac_drop
match key:          EXACT-00:00:00:00:00:02
action:             MyIngress.drop
runtime data:
Entry has been added with handle 2
RuntimeCmd: 
```

Figure 3. Table entry operation

To drop the packet sent, just adding a row entry. The code means, when the source MAC address is 00:00:00:00:02, the packets will be dropped.

```

[2] 25970
pi@p4pi:~/CWM-ProgNets/scripts $ Calling target program-options parser
Nanomsg returned a exception when trying to bind to address 'ipc:///tmp/bmv2-0-notifications.ipc'.
The exception is: Address already in use
This may happen if
1) the address provided is invalid,
2) another instance of bmv2 is running and using the same address, or
3) you have insufficient permissions (e.g. you are using an IPC socket on Unix, the file already exists and you do not have permissions to create it)
^C
[2]+  Exit 1                  sudo simple_switch -i 0@eth0 reflector.json
pi@p4pi:~/CWM-ProgNets/scripts $ ps aux | grep simple
root      13685  0.0  0.0 12172 4352 pts/0    S   01:59   0:00 sudo simple_switch -i 0@eth0 reflector.json
root      13686  0.0  0.1 664104 11632 pts/0    Sl  01:59   0:08 simple_switch -i 0@eth0 reflector.json
pi        26004  0.0  0.0   6044   640 pts/0    S+   06:09   0:00 grep --color=auto simple
pi@p4pi:~/CWM-ProgNets/scripts $ kill 13685
pi@p4pi:~/CWM-ProgNets/scripts $ ps aux | grep simple
pi        26006  0.0  0.0   5912   652 pts/0    S+   06:09   0:00 grep --color=auto simple
[1]+  Terminated              sudo simple_switch -i 0@eth0 reflector.json (wd: ~/CWM-ProgNets/assignment3)
(wd now: ~/CWM-ProgNets/scripts)
pi@p4pi:~/CWM-ProgNets/scripts $ sudo simple_switch -i 0@eth0 reflector.json &
[1] 26007
pi@p4pi:~/CWM-ProgNets/scripts $ Calling target program-options parser
JSON input file reflector.json cannot be opened
^C
[1]+  Exit 1                  sudo simple_switch -i 0@eth0 reflector.json
pi@p4pi:~/CWM-ProgNets/scripts $ ps aux | grep simple
pi        26031  0.0  0.0   5912   620 pts/0    S+   06:09   0:00 grep --color=auto simple
pi@p4pi:~/CWM-ProgNets/scripts $ ls
commands.txt  cwm  README.md
pi@p4pi:~/CWM-ProgNets/scripts $ cd ..
pi@p4pi:~/CWM-ProgNets $ ls
assignment1  assignment2  assignment3  assignment4  assignment5  LICENSE  README.md  scripts
pi@p4pi:~/CWM-ProgNets $ cd assignment3
pi@p4pi:~/CWM-ProgNets/assignment3 $ ls
reflector_class.p4  reflector.json  reflector.p4  reflector.p4i  send.py
pi@p4pi:~/CWM-ProgNets/assignment3 $ sudo simple_switch -i 0@eth0 reflector.json &
[1] 26046
pi@p4pi:~/CWM-ProgNets/assignment3 $ Calling target program-options parser
JSON input file reflector.json cannot be opened
^C
[1]+  Exit 1                  sudo simple_switch -i 0@eth0 reflector.json
pi@p4pi:~/CWM-ProgNets/assignment3 $ sudo simple_switch -i 0@eth0 reflector.json &
[1] 26078
pi@p4pi:~/CWM-ProgNets/assignment3 $ Calling target program-options parser
Adding interface eth0 as port 0
sudo simple_switch -i 0@eth0 reflector.json

```

Figure 4. Trying out with cwm scripts