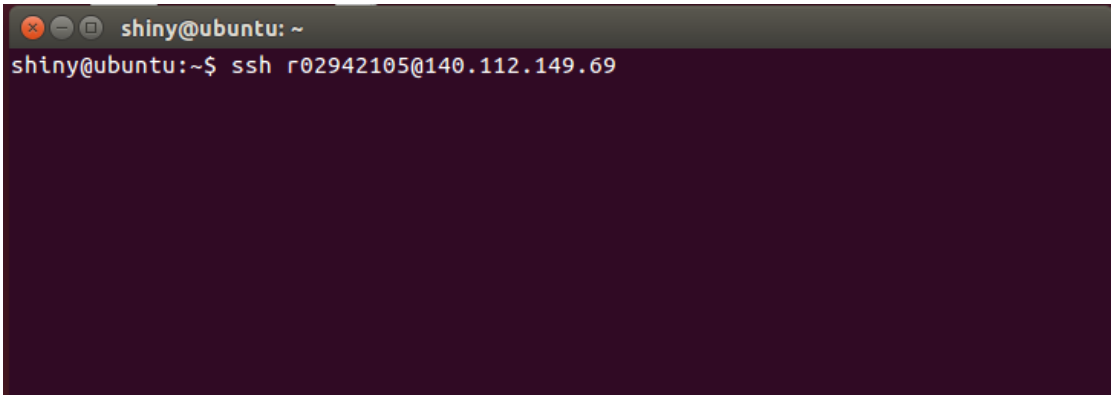
[SDN-Lab2]How to upload your sever.py

1. **Server ip : 140.112.149.69**

2. **ssh to the sever (default port:22)**
   a. terminal (Ubuntu, your vm) command: ssh stu_id@140.112.149.69



3. **ID & Password**

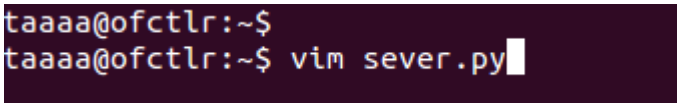
ID : your student_ID

Pwd: your student_ID

sever port of the socket: reference it in the list.xls file (server application port)

4. **Open your sever server.py**

Command: vim server.py



5. **Copy your code to the sever**

```
taaaa@ofctlr: ~
mport socket

HOST = ''                 # Symbolic name meaning all available interfaces
PORT = 50007              # Arbitrary non-privileged port
s = socket.socket(socket.AF_INET, socket.SOCK_STREAM)
s.bind((HOST, PORT))
s.listen(1)
conn, addr = s.accept()
print 'Connected by', addr
while 1:
    data = conn.recv(1024)
    if not data: break
    conn.sendall(data)
conn.close()

~
~
~
~
~
~
~
~
:wq
```

Then press “Esc” and enter “:wq” (which means write and quit)

6. **Then you could see your file under the directory**

Command “ll”

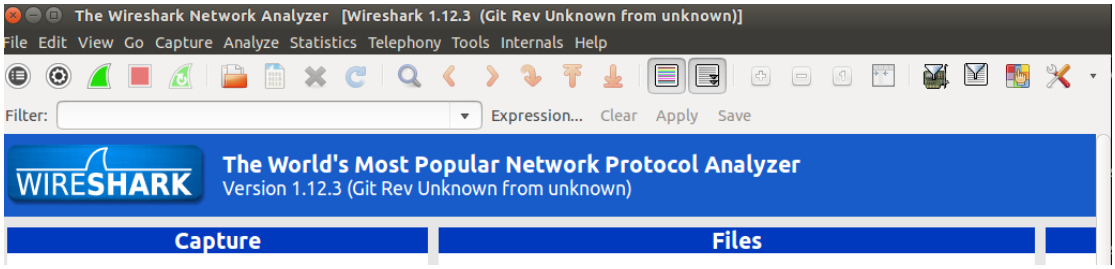
```
taaaa@ofctlr:~$ ll
total 36
drwxr-xr-x  3 taaaa taaaa 4096 May  1 16:41 ./
drwxr-xr-x 44 root  root  4096 Apr 14 16:41 ../
-rw-r--r--  1 taaaa taaaa  220 Apr 14 16:41 .bash_logout
-rw-r--r--  1 taaaa taaaa 3637 Apr 14 16:41 .bashrc
drwx------  2 taaaa taaaa 4096 May  1 16:00 .cache/
-rw-r--r--  1 taaaa taaaa  675 Apr 14 16:41 .profile
-rw-rw-r--  1 taaaa taaaa  381 May  1 16:18 r02942105.py
-rw-rw-r--  1 taaaa taaaa  382 May  1 16:41 sever.py
-rw-------  1 taaaa taaaa  872 May  1 16:41 .viminfo
taaaa@ofctlr:~$
```

7. **Run your socket code on the remote server**

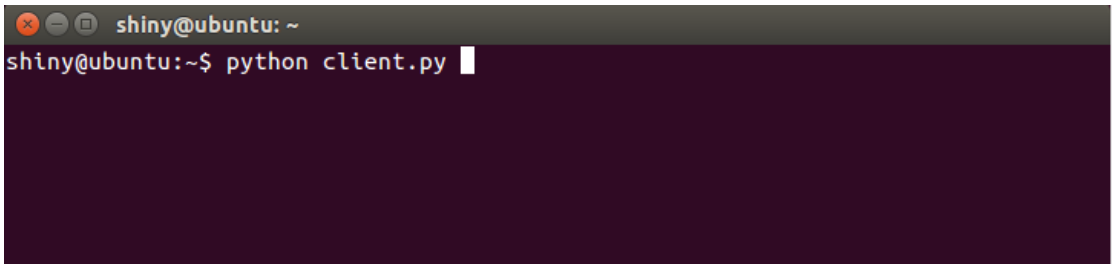
```
taaaa@ofctlr:~$ python r02942105.py
```

8. **Run “wireshark” in your computer (client side) to observe the whole process of communication between sockets and write what you observe in the report.**

```
shiny@ubuntu: ~
shiny@ubuntu:~$ sudo wireshark
[sudo] password for shiny:
```



9. **Open "another" terminal to run your client.py to send the txt file to the remote server.**



10. **The server should send the correct data back to the client. Then when client receive it, you should save it as stu_id.txt (correct txt file)**