NETWORKING LAB (CSE1004)

EXPERIMENT-4

Shivam Batra

19BPS1131

Q1. Design a multi threaded Chat client and server with TCP scoket using Python.

(One server should be serving multiple clients concurrently)

Answer:

Code:

Server:

```
import socket, threading
class ClientThread(threading.Thread):
    def __init__(self,clientAddress,clientsocket):
        threading.Thread.__init__(self)
        self.csocket = clientsocket
        print ("New connection added: ", clientAddress)
    def run(self):
        #self.csocket.send(bytes("Hi, This is from Server..",'utf-8'))
        msg = ''
        while True:
            data = self.csocket.recv(2048)
            msg = data.decode()
            if msg=='bye':
              break
            print ("from client",msg)
            message=input()
            self.csocket.send(bytes(message,'UTF-8'))
        print ("Client at ", clientAddress , " disconnected...")
LOCALHOST = "127.0.0.1"
PORT = 8080
server = socket.socket(socket.AF_INET, socket.SOCK_STREAM)
#server.setsockopt(socket.SOL_SOCKET, socket.SO_REUSEADDR, 1)
server.bind((LOCALHOST, PORT))
print("Server started")
print("Waiting for client request..")
while True:
    server.listen(1)
    clientsock, clientAddress = server.accept()
    newthread = ClientThread(clientAddress, clientsock)
    newthread.start()
```

Client 1:

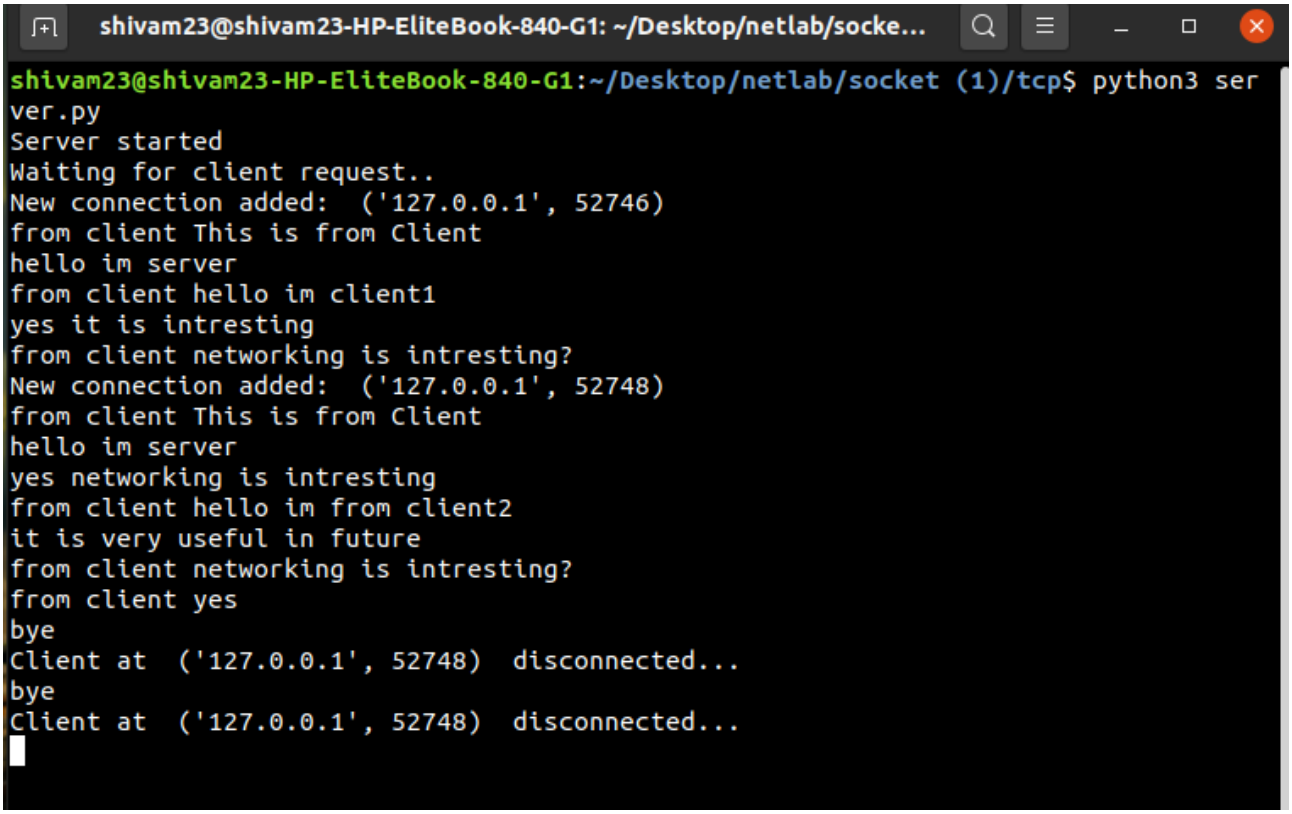
```
import socket
SERVER = "127.0.0.1"
PORT = 8080
client = socket.socket(socket.AF_INET, socket.SOCK_STREAM)
client.connect((SERVER, PORT))
client.sendall(bytes("This is from Client",'UTF-8'))
while True:
  in_data =  client.recv(1024)
  print("Reply From Server to client1 :" ,in_data.decode())
  out_data = input()
  client.sendall(bytes(out_data,'UTF-8'))
  if out_data=='bye':
        break
client.close()
```

Client 2:

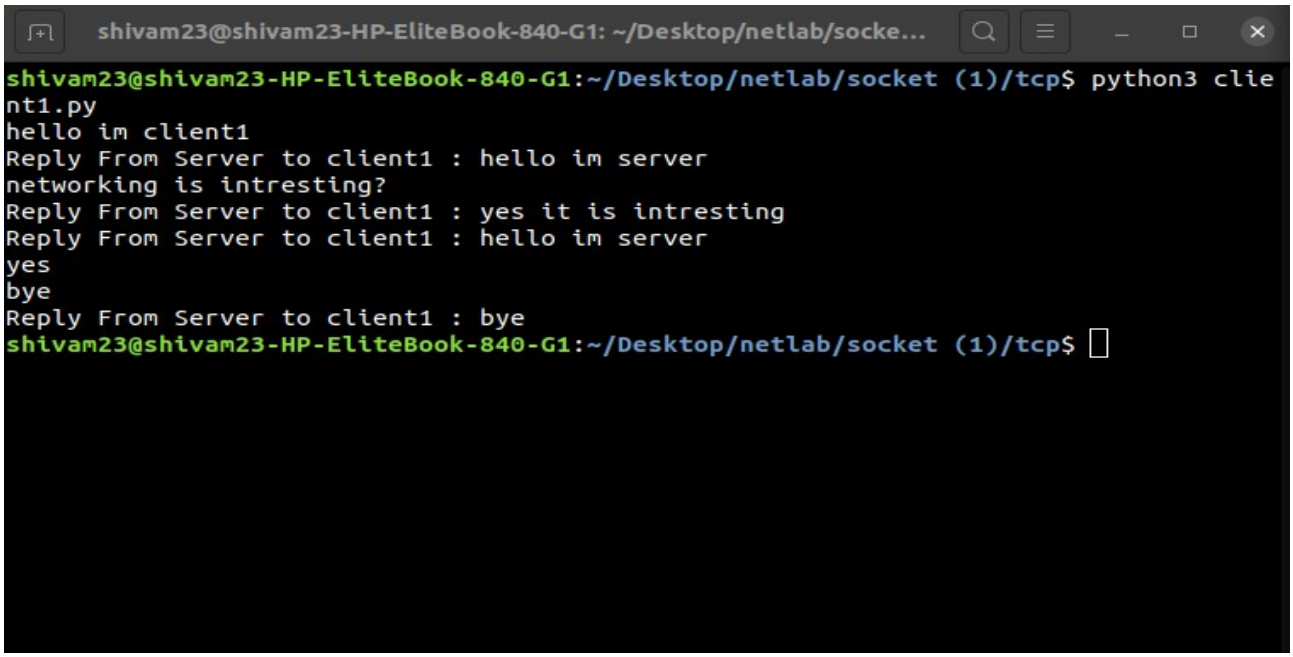
```
import socket
SERVER = "127.0.0.1"
PORT = 8080
client = socket.socket(socket.AF_INET, socket.SOCK_STREAM)
client.connect((SERVER, PORT))
client.sendall(bytes("This is from Client",'UTF-8'))
while True:
  in_data =  client.recv(1024)
  print("Reply From Server to client2 :" ,in_data.decode())
  out_data = input()
  client.sendall(bytes(out_data,'UTF-8'))
  if out_data=='bye':
        break
client.close()
```

SCREENSHOTS: The three terminals are running simaltanously

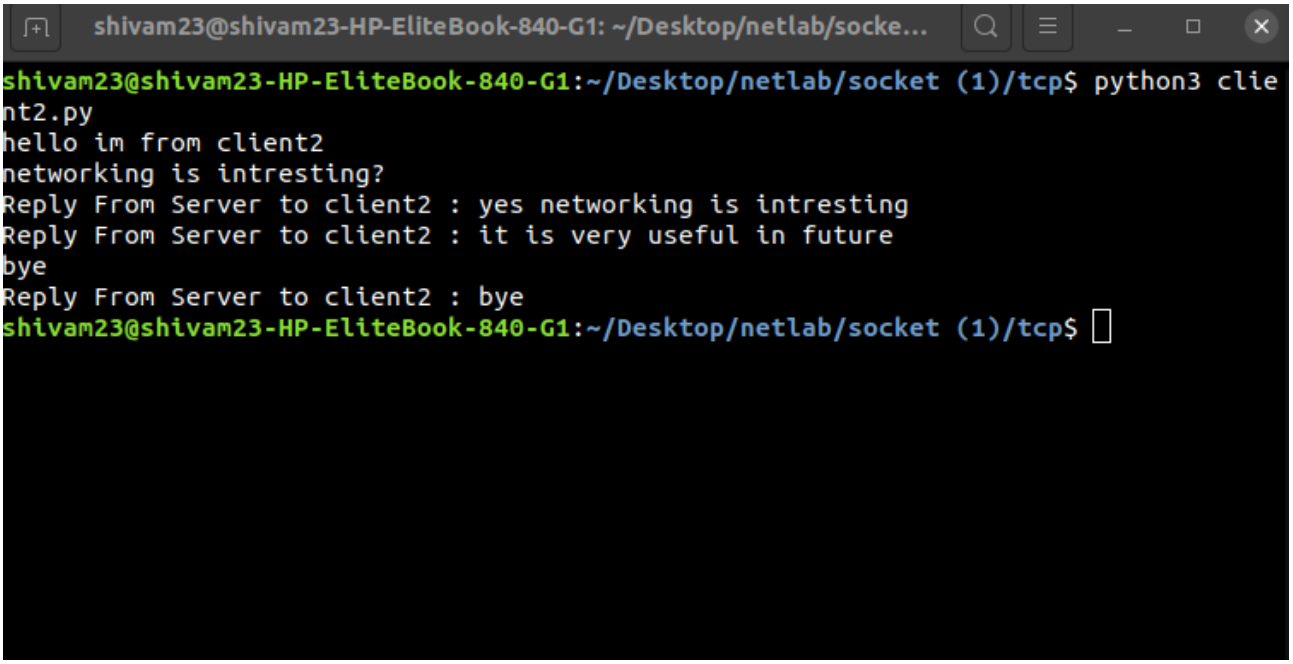
Server:


shivam23@shivam23-HP-EliteBook-840-G1: ~/Desktop/netlab/socke...
shivam23@shivam23-HP-EliteBook-840-G1:~/Desktop/netlab/socket (1)/tcp$ python3 ser
ver.py
Server started
Waiting for client request..
New connection added: ('127.0.0.1', 52746)
from client This is from Client
hello im server
from client hello im client1
yes it is intresting
from client networking is intresting?
New connection added: ('127.0.0.1', 52748)
from client This is from Client
hello im server
yes networking is intresting
from client hello im from client2
it is very useful in future
from client networking is intresting?
from client yes
bye
Client at ('127.0.0.1', 52748) disconnected...
bye
Client at ('127.0.0.1', 52748) disconnected...

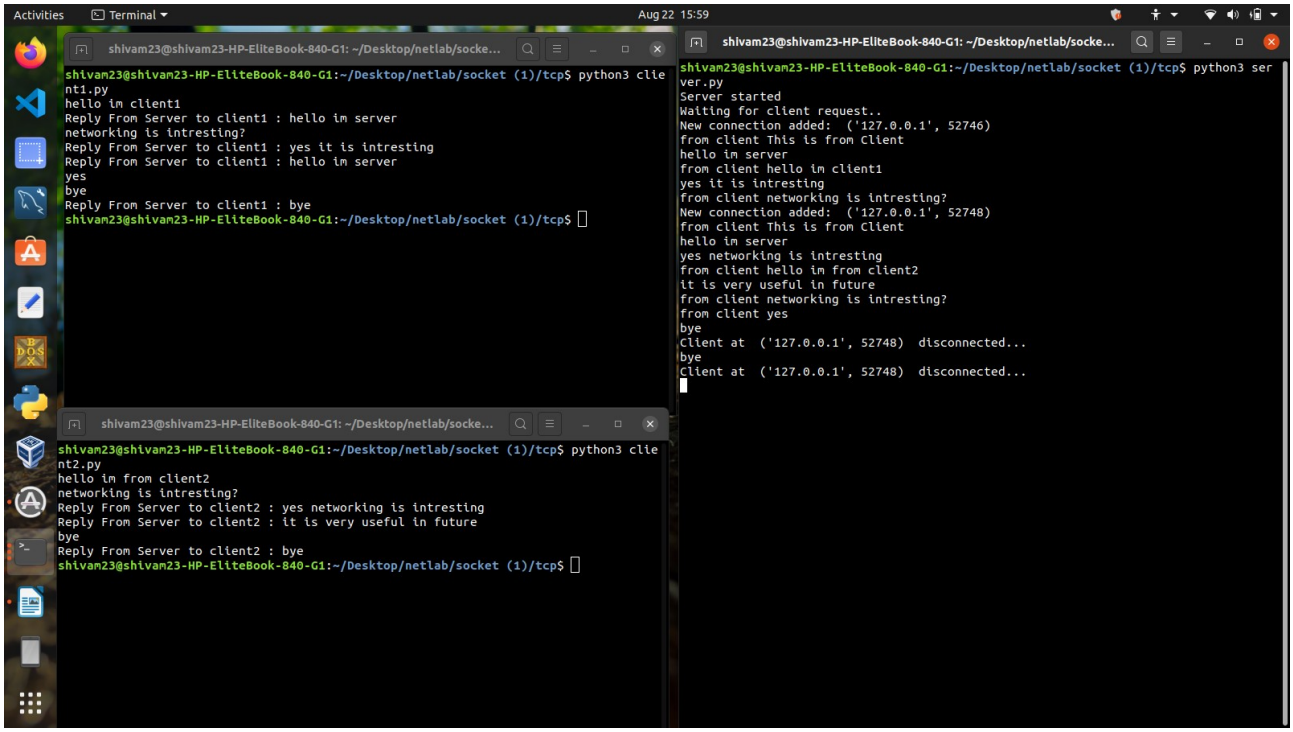
Client 1:


shivam23@shivam23-HP-EliteBook-840-G1: ~/Desktop/netlab/socke...
shivam23@shivam23-HP-EliteBook-840-G1:~/Desktop/netlab/socket (1)/tcp$ python3 clie
nt1.py
hello im client1
Reply From Server to client1 : hello im server
networking is intresting?
Reply From Server to client1 : yes it is intresting
Reply From Server to client1 : hello im server
yes
bye
Reply From Server to client1 : bye
shivam23@shivam23-HP-EliteBook-840-G1:~/Desktop/netlab/socket (1)/tcp$

Client 2:


shivam23@shivam23-HP-EliteBook-840-G1: ~/Desktop/netlab/socke...
shivam23@shivam23-HP-EliteBook-840-G1:~/Desktop/netlab/socket (1)/tcp$ python3 clie
nt2.py
hello im from client2
networking is intresting?
Reply From Server to client2 : yes networking is intresting
Reply From Server to client2 : it is very useful in future
bye
Reply From Server to client2 : bye
shivam23@shivam23-HP-EliteBook-840-G1:~/Desktop/netlab/socket (1)/tcp$

SCREEN:



Ques 2. Design a multi threaded File client and server with TCP scoket using Python. (One server should be serving multiple clients concurrently)

Answer

Server:

```
import socket, threading
class ClientThread(threading.Thread):
    def __init__(self,clientAddress,clientsocket):
        threading.Thread.__init__(self)
        self.csocket = clientsocket
        print ("New connection added: ", clientAddress)
    def run(self):

         msg = ''
         while True:
             self.csocket.send(bytes("please enter the name of file ",'UTF-8'))
        #self.csocket.send(bytes("Hi, This is from Server..",'utf-8'))
             data = self.csocket.recv(2048)
             msg = data.decode()
             line=" "
             print("in loop")
             if msg=='bye':
               print ("Client at ", clientAddress , " disconnected...")
               break
             elif msg =="exp4file1.txt" :
               f= open("exp4file1.txt","r")
               for line in f:
                  self.csocket.sendall(bytes(line,'UTF-8'))
```

```
                print("sendling")
              print("file 1 sent")
            elif msg=="exp4file2.txt":
              f= open("exp4file2.txt","r")
              for line in f:
                self.csocket.sendall(bytes(line,'UTF-8'))
              print("file 2 sent")
            else:
              self.csocket.sendall(bytes(" file not found ",'UTF-8'))
LOCALHOST = "127.0.0.1"
PORT = 8080
server = socket.socket(socket.AF_INET, socket.SOCK_STREAM)
#server.setsockopt(socket.SOL_SOCKET, socket.SO_REUSEADDR, 1)
server.bind((LOCALHOST, PORT))
print("Server started")
print("Waiting for client request..")
while True:
    server.listen(1)
    clientsock, clientAddress = server.accept()
    newthread = ClientThread(clientAddress, clientsock)
    newthread.start()
```

Client 1:

```
import socket
SERVER = "127.0.0.1"
PORT = 8080
client = socket.socket(socket.AF_INET, socket.SOCK_STREAM)
client.connect((SERVER, PORT))
while True:
  out_data = input()
  client.sendall(bytes(out_data,'UTF-8'))
  in_data =  client.recv(1024)
  print("Reply From Server to client1: ")
  print(in_data.decode())
  if out_data=='bye':
   break
client.close()
```

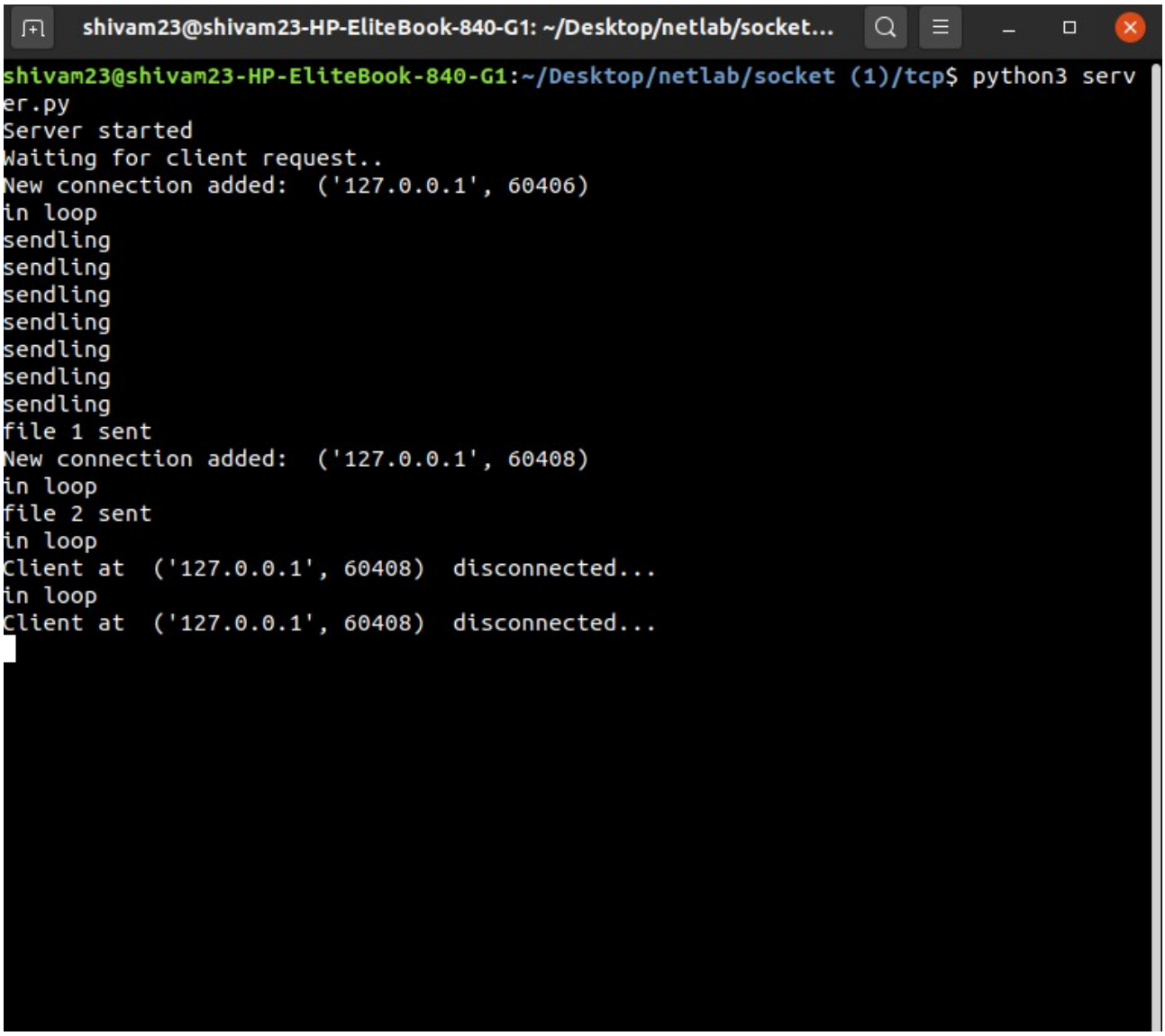
Client 2:

```
import socket
SERVER = "127.0.0.1"
PORT = 8080
client = socket.socket(socket.AF_INET, socket.SOCK_STREAM)
client.connect((SERVER, PORT))
while True:
  out_data = input()
  client.sendall(bytes(out_data,'UTF-8'))
  in_data =  client.recv(1024)
  print("Reply From Server to client2: ")
  print(in_data.decode())
```

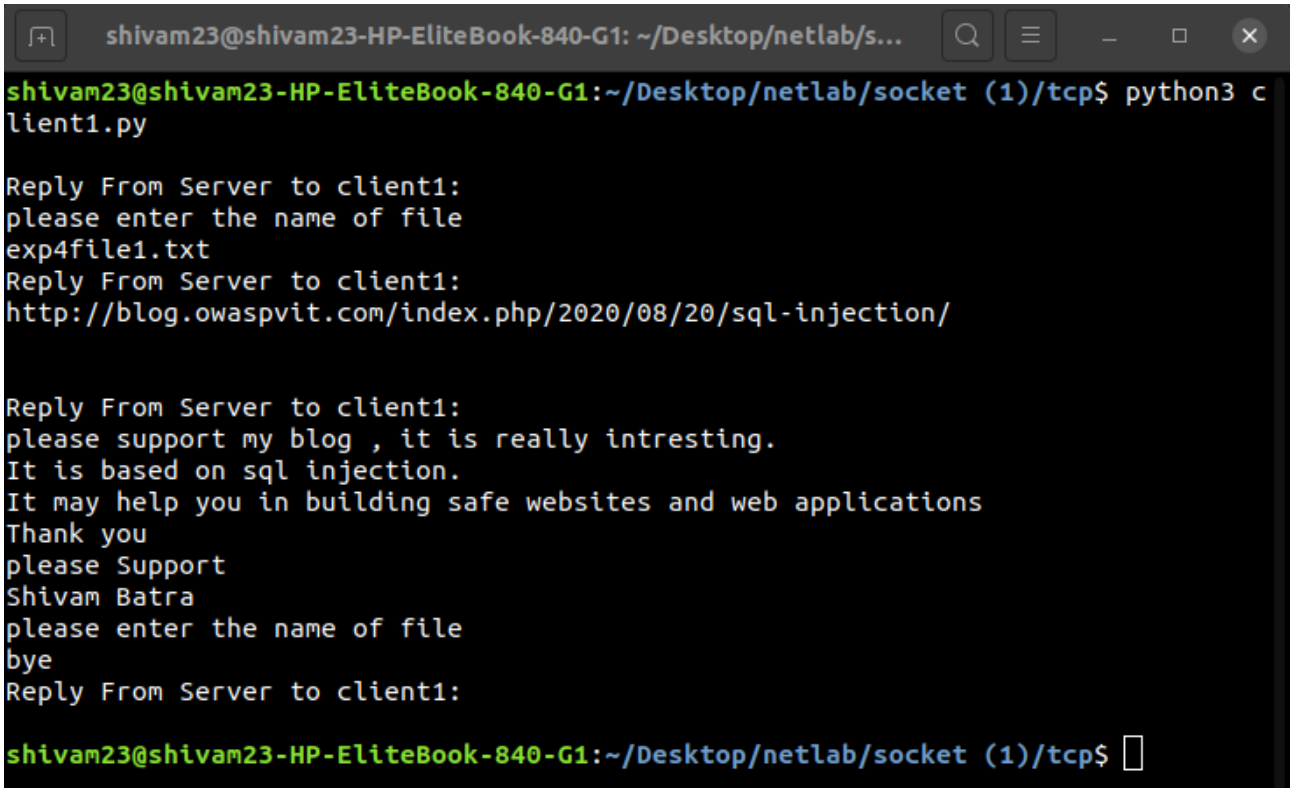
```
  if out_data=='bye':
   break
client.close()
```

SCREENSHOTS : The three terminals are running simaltanously. Client 1 asks for file named “exp4file1.txt” and Client2 asks for “exp4file2.txt” both client 1 and client 2 requests server simaltanously

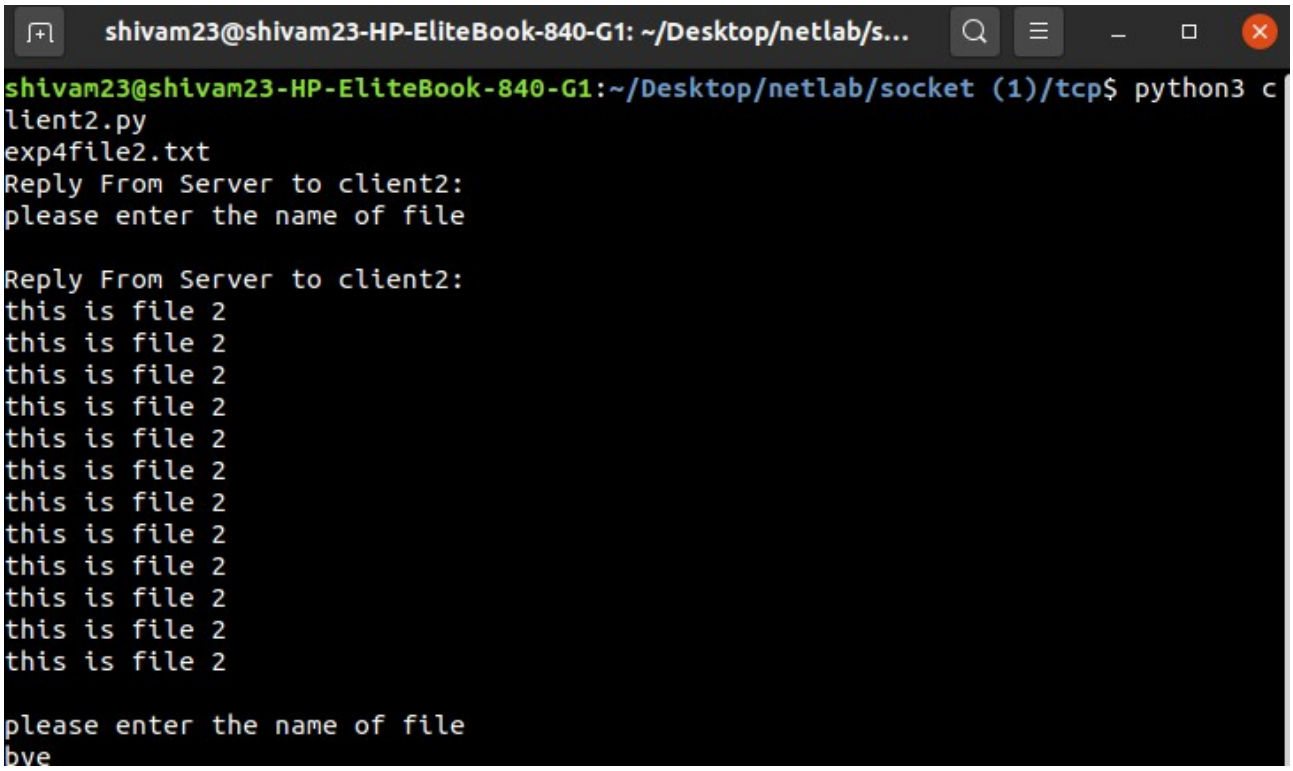
SERVER:



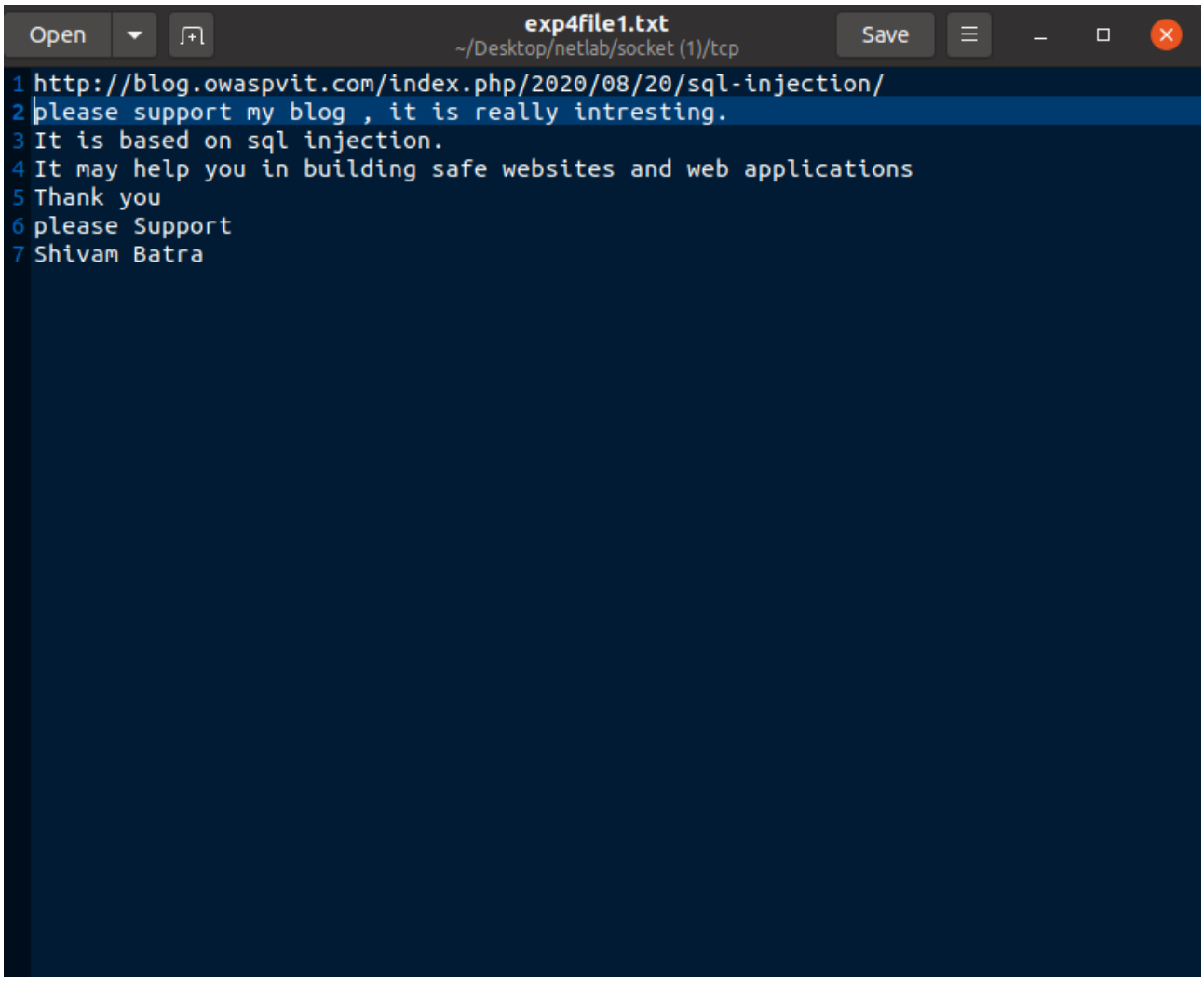
Client 1:


shivam23@shivam23-HP-EliteBook-840-G1: ~/Desktop/netlab/s...
shivam23@shivam23-HP-EliteBook-840-G1:~/Desktop/netlab/socket (1)/tcp$ python3 c
lient1.py
Reply From Server to client1:
please enter the name of file
exp4file1.txt
Reply From Server to client1:
http://blog.owaspvit.com/index.php/2020/08/20/sql-injection/
Reply From Server to client1:
please support my blog , it is really intresting.
It is based on sql injection.
It may help you in building safe websites and web applications
Thank you
please Support
Shivam Batra
please enter the name of file
bye
Reply From Server to client1:
shivam23@shivam23-HP-EliteBook-840-G1:~/Desktop/netlab/socket (1)/tcp$

Client 2:


shivam23@shivam23-HP-EliteBook-840-G1: ~/Desktop/netlab/s...
shivam23@shivam23-HP-EliteBook-840-G1:~/Desktop/netlab/socket (1)/tcp$ python3 c
lient2.py
exp4file2.txt
Reply From Server to client2:
please enter the name of file
Reply From Server to client2:
this is file 2
this is file 2
this is file 2
this is file 2
this is file 2
this is file 2
this is file 2
this is file 2
this is file 2
this is file 2
this is file 2
this is file 2
please enter the name of file
bye

exp4file1.txt:



exp4file2.txt

Open exp4file2.txt ~/Desktop/netlab/socket (1)/tcp Save

```
this is file 2
this is file 2
this is file 2
this is file 2
this is file 2
this is file 2
this is file 2
this is file 2
this is file 2
this is file 2
this is file 2
this is file 2
```

SCREEN: