

NETWORKING LAB (CSE1004)

EXPERIMENT-4

Shivam Batra

19BPS1131

Q1. Design a multi threaded Chat client and server with TCP socket 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 simultaneously

Server:

```
shivam23@shivam23-HP-EliteBook-840-G1: ~/Desktop/netlab/socket (1)/tcp$ python3 server.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...
█
```

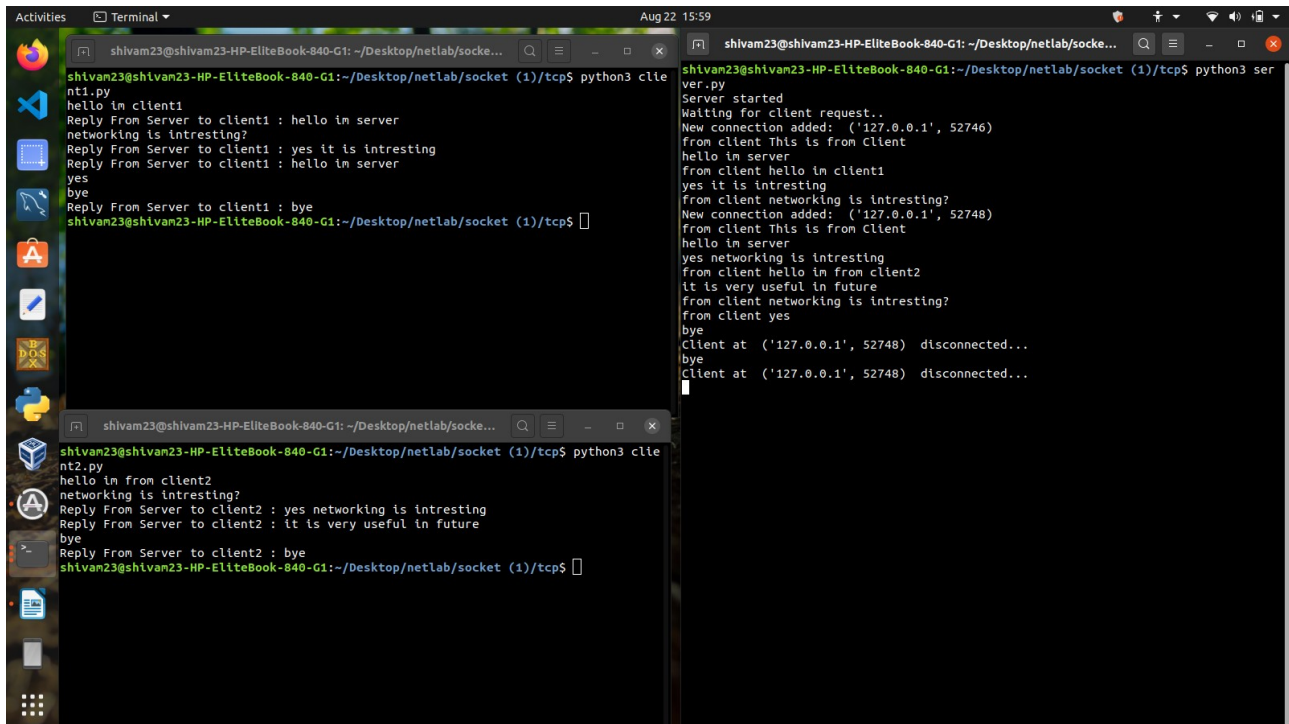
Client 1:

```
shivam23@shivam23-HP-EliteBook-840-G1: ~/Desktop/netlab/socket (1)/tcp$ python3 client1.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/socket (1)/tcp$ python3 client2.py
hello im from client2
networking is interesting?
Reply From Server to client2 : yes networking is interesting
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:



The screenshot shows a Linux desktop environment with a terminal window open. The terminal displays the output of a Python script running a multi-threaded TCP server. The server is listening on port 52746. It receives two connections from client1 and client2. Client1 sends 'hello in server', 'networking is interesting?', 'yes it is interesting', 'hello in server', and 'bye'. Client2 sends 'hello in from client2', 'networking is interesting?', 'yes networking is interesting', 'it is very useful in future', and 'bye'. The server responds to each client with the same messages. The terminal also shows the server starting and waiting for client requests.

```
shivan23@shivan23-HP-EliteBook-840-G1: ~/Desktop/netlab/socket (1)/tcp$ python3 client.py
hello in client1
Reply From Server to client1 : hello in server
networking is interesting?
Reply From Server to client1 : yes it is interesting
Reply From Server to client1 : hello in server
yes
bye
Reply From Server to client1 : bye
shivan23@shivan23-HP-EliteBook-840-G1:~/Desktop/netlab/socket (1)/tcp$

shivan23@shivan23-HP-EliteBook-840-G1:~/Desktop/netlab/socket (1)/tcp$ python3 server.py
Server started
Waiting for client request..
New connection added: ('127.0.0.1', 52746)
from client This is from Client
hello in server
from client hello in client1
yes it is interesting
from client networking is interesting?
New connection added: ('127.0.0.1', 52748)
from client This is from Client
hello in server
yes networking is interesting
from client hello in from client2
it is very useful in future
from client networking is interesting?
from client yes
bye
Client at ('127.0.0.1', 52748) disconnected...
bye
Client at ('127.0.0.1', 52748) disconnected...
```

Ques 2. Design a multi threaded File client and server with TCP socket 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:

```

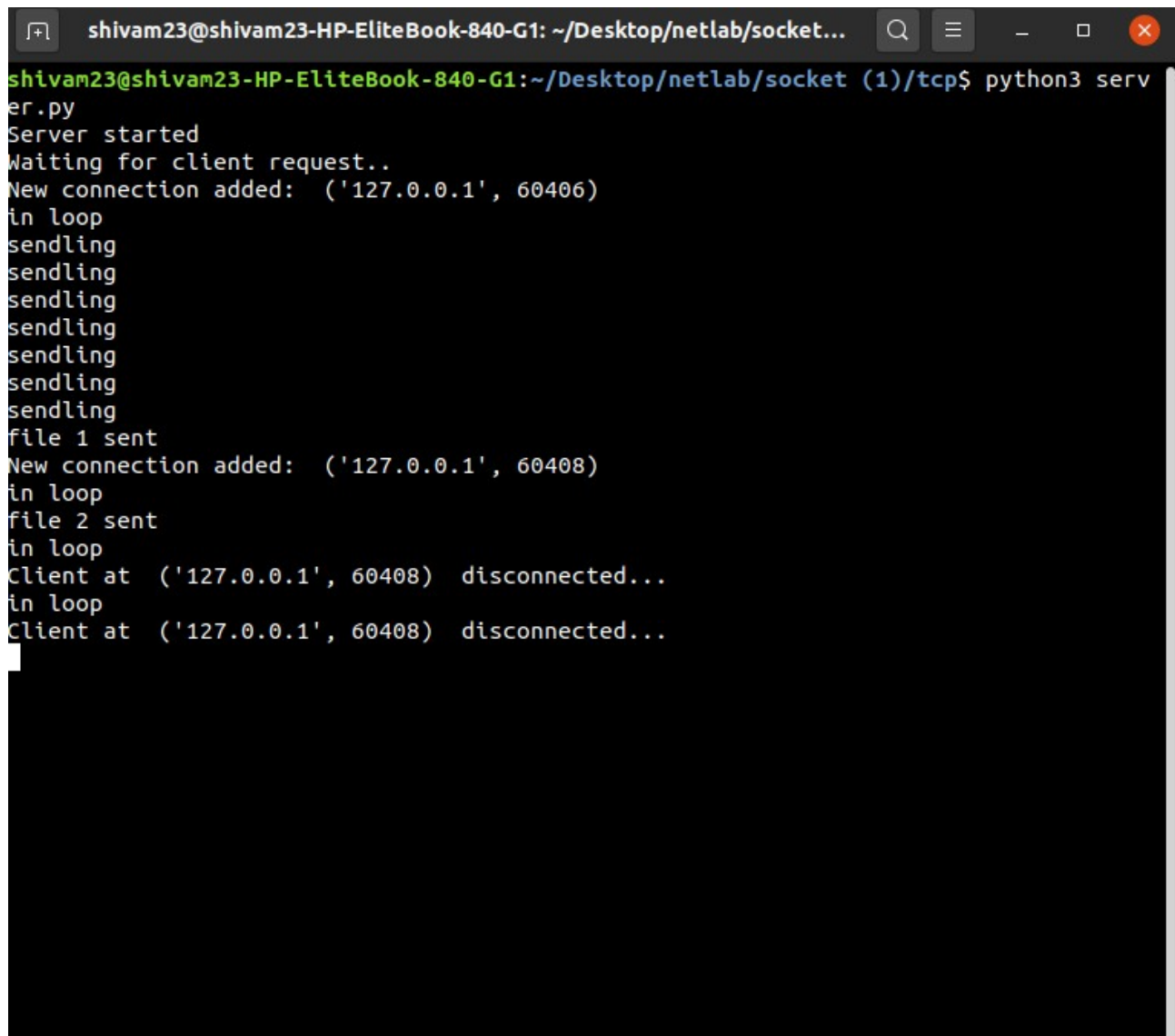
import socket
SERVER = "127.0.0.1"
PORT = 8080
client = socket.socket(socket.AF_INET, socket.SOCK_STREAM)
client.connect((SERVER, PORT))
while True:
    out_data = input()
    client.sendall(bytes(out_data,'UTF-8'))
    in_data = client.recv(1024)
    print("Reply From Server to client2: ")
    print(in_data.decode())

```

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

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

SERVER:

A screenshot of a terminal window titled "shivam23@shivam23-HP-EliteBook-840-G1: ~/Desktop/netlab/socket...". The terminal shows the output of a Python script. The script starts with "Server started" and "Waiting for client request..". It then receives a "New connection added: ('127.0.0.1', 60406)" and enters a loop where it repeatedly prints "sendling" (likely a typo for "sending") before printing "file 1 sent". A second connection is added: "New connection added: ('127.0.0.1', 60408)", and it prints "file 2 sent". Both clients then disconnect, with the terminal showing "Client at ('127.0.0.1', 60408) disconnected..." twice.

```
shivam23@shivam23-HP-EliteBook-840-G1: ~/Desktop/netlab/socket (1)/tcp$ python3 serv  
er.py  
Server started  
Waiting for client request..  
New connection added: ('127.0.0.1', 60406)  
in loop  
sendling  
sendling  
sendling  
sendling  
sendling  
sendling  
sendling  
sendling  
file 1 sent  
New connection added: ('127.0.0.1', 60408)  
in loop  
file 2 sent  
in loop  
Client at ('127.0.0.1', 60408) disconnected...  
in loop  
Client at ('127.0.0.1', 60408) disconnected...
```

Client 1:

```
shivam23@shivam23-HP-EliteBook-840-G1: ~/Desktop/netlab/s...
shivam23@shivam23-HP-EliteBook-840-G1:~/Desktop/netlab/socket (1)/tcp$ python3 client1.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/

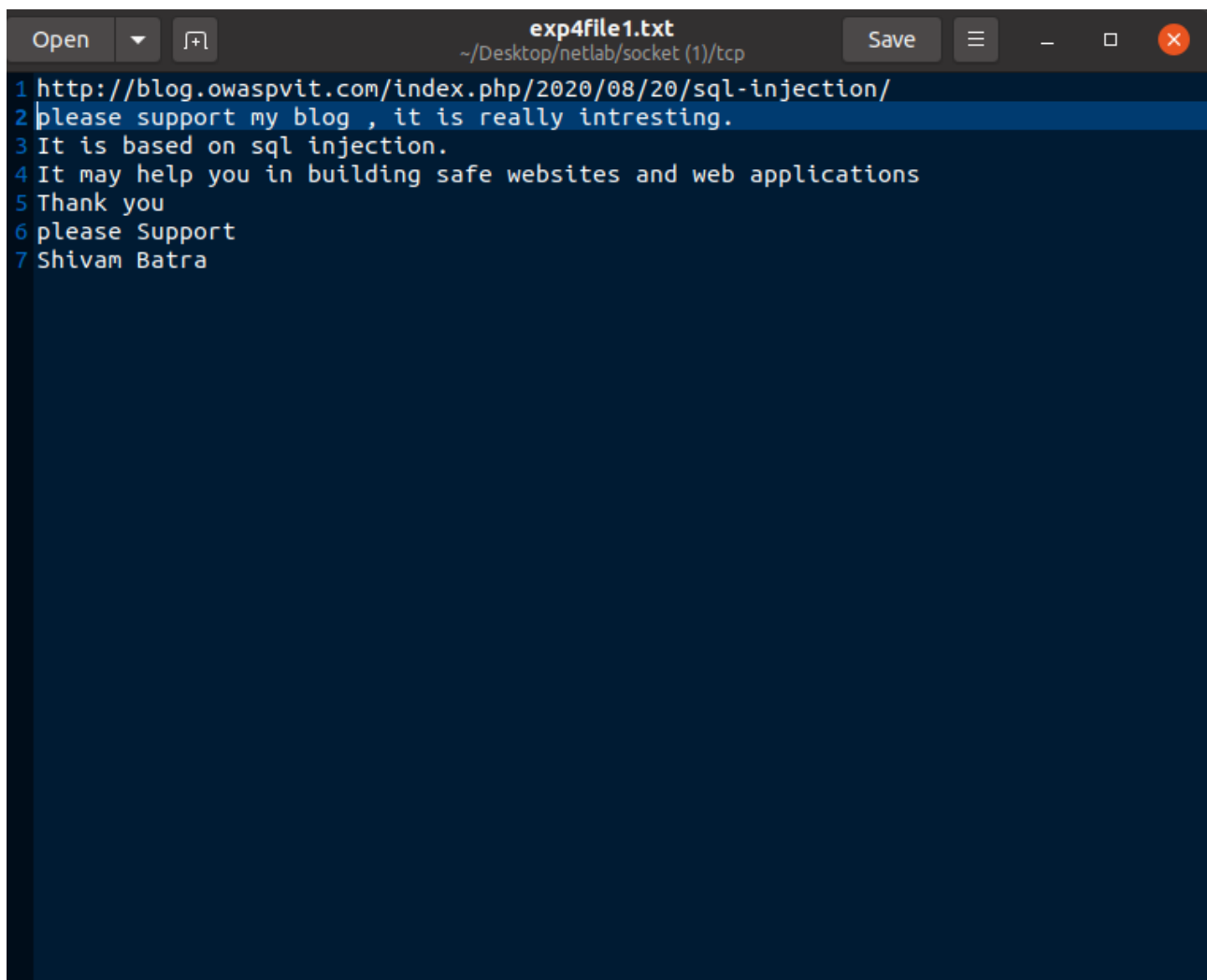
Reply From Server to client1:
please support my blog , it is really intresting.
It is based on sql injection.
It may help you in building safe websites and web applications
Thank you
please Support
Shivam Batra
please enter the name of file
bye
Reply From Server to client1:

shivam23@shivam23-HP-EliteBook-840-G1:~/Desktop/netlab/socket (1)/tcp$
```

Client 2:

[illegible]

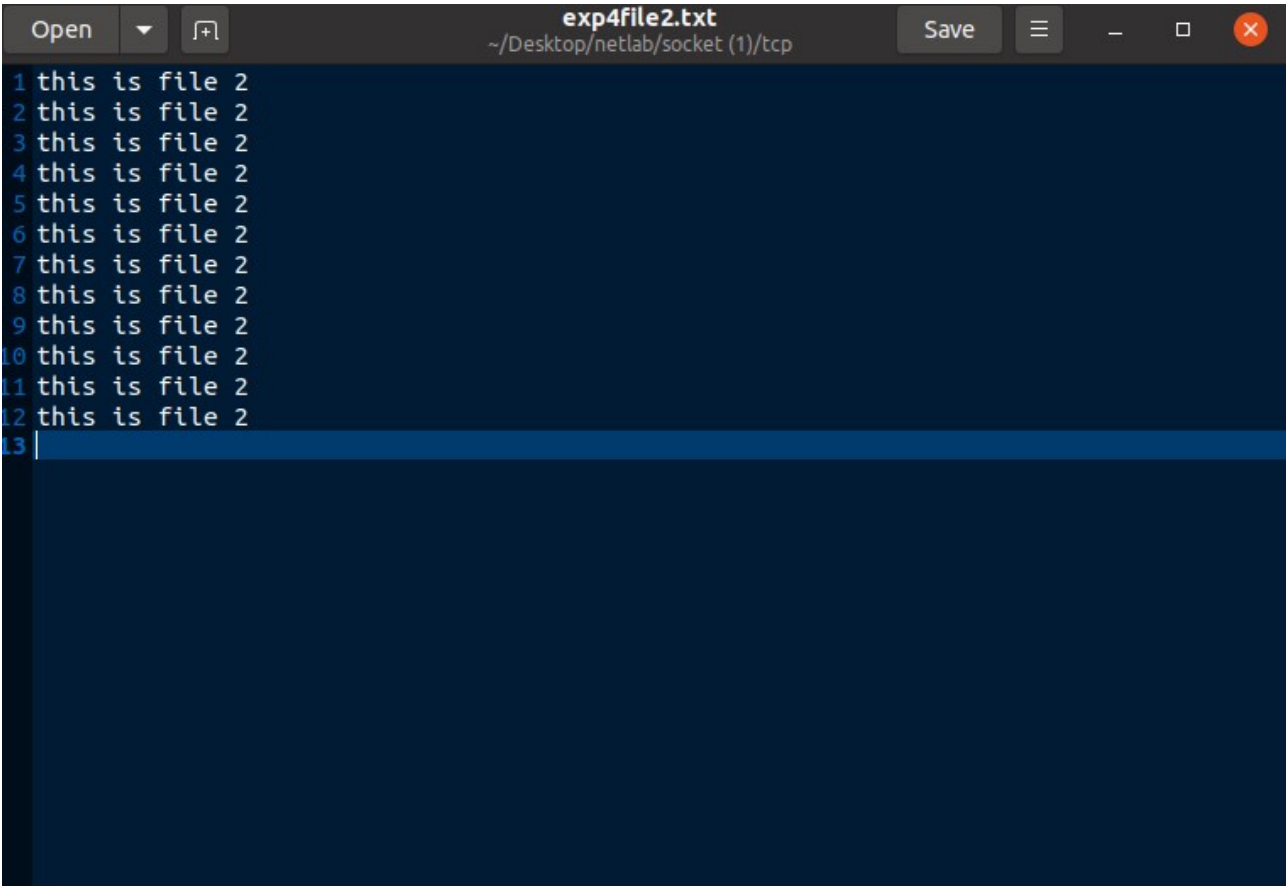
exp4file1.txt:



The image shows a text editor window with a dark theme. The title bar at the top reads 'exp4file1.txt' and the path below it is '~/Desktop/netlab/socket (1)/tcp'. The editor contains the following text:

```
1 http://blog.owaspvit.com/index.php/2020/08/20/sql-injection/  
2 please support my blog , it is really intresting.  
3 It is based on sql injection.  
4 It may help you in building safe websites and web applications  
5 Thank you  
6 please Support  
7 Shivam Batra
```

exp4file2.txt



SCREEN:

