Computer Networking CSE 5344

Project 1

Simple Web Server & Client

Instructor: Sajib Datta

Name: - Ravi Prakasha

UTA ID: - 1002026832

Objectives:

1) Gaining knowledge of multithreaded client-server communication utilizing sockets.

2) To understand the HTTP message structures.

3) To comprehend how a web client and server operate.

Project Description:

(A) You will be developing a multi-threaded Web server which interacts with any standard Web Clients (You may use any web browser of your choice to test the functionality however you should also submit the a client as given in (B) below). The Web server and Web client communicate using a text - based protocol called HTTP (Hypertext Transfer Protocol.

(B) Build a single threaded Web Client on your own which interacts with your Web Server and downloads a file from the server.

(C) Display the essential connection parameters of connection for both the Web client (on the server side) and for the Web Server (on the client side).

Code Compilation Instruction:

- Language - Python 3.9.7
- IDE - VS code
- Web Browser: Chrome

Steps to run the server:

1. Run server program i.e., servy.py on 1st Terminal.
2. python3 servy.py port-number
3. E.g. - python3 servy.py 8080

Note: If a port number is not given, the default value of 8080 will be used.

Default Values Provided (Server):

1. IP Address: 127.0.0.1 (localhost)
2. Port Number: 8080

Running Steps for Client:

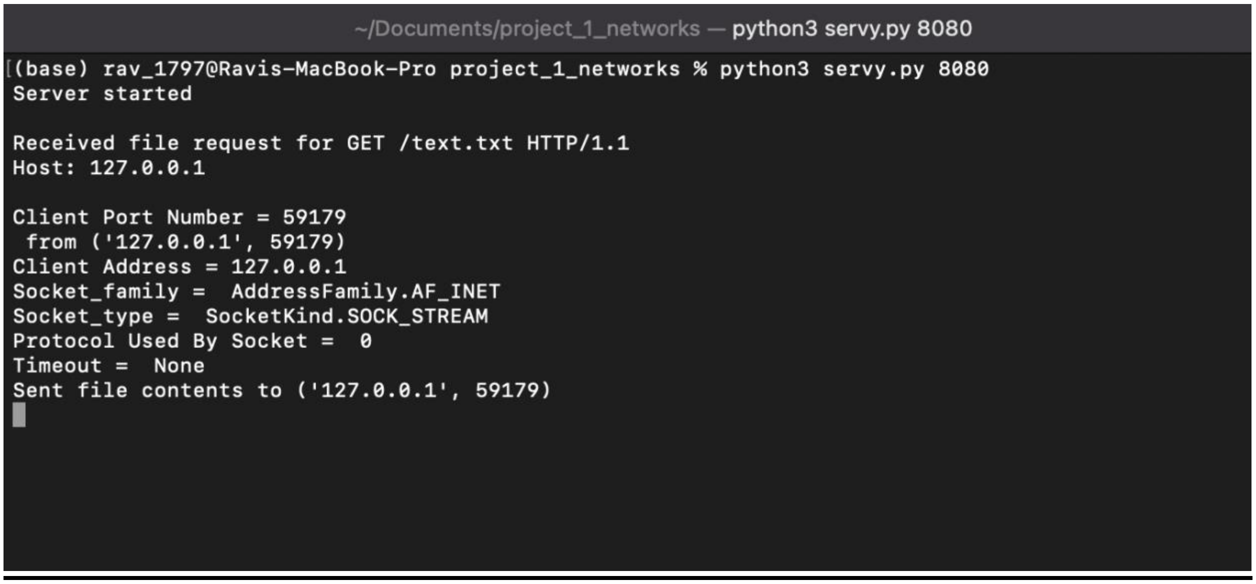
1. Run the client program i.e., clieny.py 2nd Terminal.
2. python3 client.py IP-Address Port-Number Filename
3. E.g. - python3 client.py 127.0.0.1 8080 text.txt

Note: There is no requirement for IP address, port number, or filename. If nothing is entered, the default value will be used.

Default Values(Client):

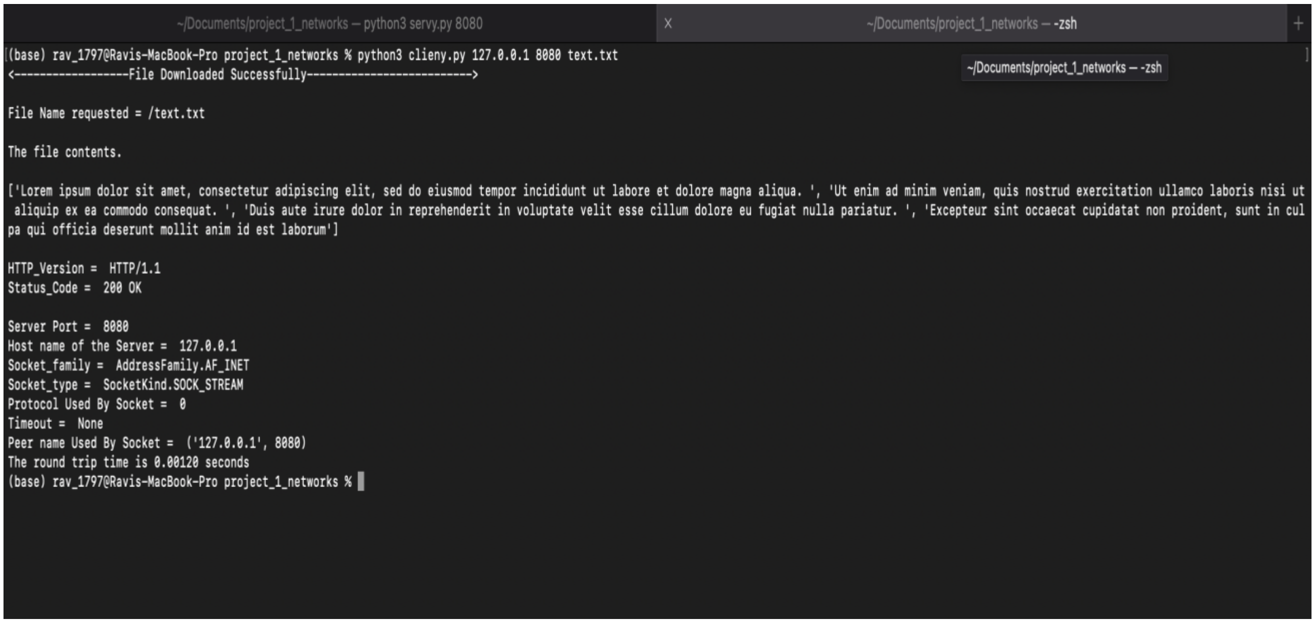
1. IP Address: 127.0.0.1(localhost)
2. Port Number: 8080
3. Filename: text.txt

Local :



```
[(base) rav_1797@Ravis-MacBook-Pro project_1_networks % python3 servy.py 8080
 Server started

 Received file request for GET /text.txt HTTP/1.1
 Host: 127.0.0.1

 Client Port Number = 59179
  from ('127.0.0.1', 59179)
 Client Address = 127.0.0.1
 Socket_family =  AddressFamily.AF_INET
 Socket_type =  SocketKind.SOCK_STREAM
 Protocol Used By Socket =  0
 Timeout =  None
 Sent file contents to ('127.0.0.1', 59179)
```

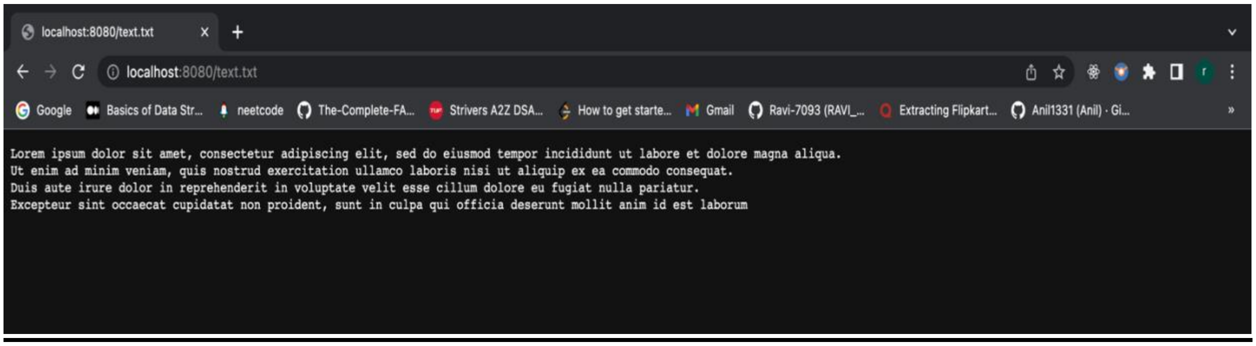
Fig.1 servy.py


~/Documents/project_1_networks — python3 servy.py 8080
X
~/Documents/project_1_networks — -zsh
+
~/Documents/project_1_networks — -zsh

```
[(base) rav_1797@Ravis-MacBook-Pro project_1_networks % python3 clieny.py 127.0.0.1 8080 text.txt
<------------------File Downloaded Successfully------------------------->

File Name requested = /text.txt

The file contents.

['Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. ', 'Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut
 aliquip ex ea commodo consequat. ', 'Duis aute irure dolor in reprehenderit in voluptate velit esse cillum dolore eu fugiat nulla pariatur. ', 'Excepteur sint occaecat cupidatat non proident, sunt in cul
pa qui officia deserunt mollit anim id est laborum']

HTTP_Version =  HTTP/1.1
Status_Code =  200 OK

Server Port =  8080
Host name of the Server =  127.0.0.1
Socket_family =  AddressFamily.AF_INET
Socket_type =  SocketKind.SOCK_STREAM
Protocol Used By Socket =  0
Timeout =  None
Peer name Used By Socket =  ('127.0.0.1', 8080)
The round trip time is 0.00120 seconds
(base) rav_1797@Ravis-MacBook-Pro project_1_networks %
```

Fig.2 clieny.py

Web:



```
Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua.
Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea commodo consequat.
Duis aute irure dolor in reprehenderit in voluptate velit esse cillum dolore eu fugiat nulla pariatur.
Excepteur sint occaecat cupidatat non proident, sunt in culpa qui officia deserunt mollit anim id est laborum
```

Fig.3 Output from the Browser

Reference:

- https://www.codingninjas.com/codestudio/library/socket-programming-with-multithreading-in-python
- https://stackoverflow.com/questions/23828264/how-to-make-a-simple-multithreaded-socket-server-in-python-that-remembers-client
- https://www.pubnub.com/blog/socket-programming-in-python-client-server-p2p/
- https://www.geeksforgeeks.org/socket-programming-python/
- https://www.geeksforgeeks.org/socket-programming-multi-threading-python/
- https://github.com/biswaranjannanda/CSE-5344-Computer-Networks