

TRƯỜNG ĐẠI HỌC KHOA HỌC TỰ NHIÊN - TP HCM

KHOA CÔNG NGHỆ THÔNG TIN

LỚP CỬ NHÂN TÀI NĂNG 2016



Báo cáo đồ án Socket

XÂY DỰNG CHƯƠNG TRÌNH FTP-CLIENT

Phạm Minh Chiến - 1612052

Đoàn Quang Tuấn - 1612780

Lê Hoàng Sang - 1612554

TP Hồ Chí Minh, ngày 31 tháng 5 năm 2018

Mục lục

1. PHÂN CÔNG CÔNG VIỆC	2
2. NHỮNG CHỨC NĂNG CHÍNH VÀ CẤU TRÚC TỔ CHỨC	3
a. Chức năng chương trình:	3
b. Cấu trúc chương trình	3
i. Tổ chức chương trình::	3
ii. Cung cấp các hàm:	4
iii. Bảng mô tả các hàm chức năng:	4
c. Mô tả các hàm:.....	8
3. CHẠY THỬ VÀ KẾT QUẢ	9
a. Thử chế độ Active (mặc định):	10
b. Test chế độ Passive:	17
4. ĐÁNH GIÁ.....	20
5. BẮT GÓI TIN VÀ MÔ TẢ QUÁ TRÌNH GỬI NHẬN	21

1. PHÂN CÔNG CÔNG VIỆC

Thành phần nhóm:

Phạm Minh Chiến	1612052
Lê Hoàng Sang	1612554
Đoàn Quang Tuấn	1612780

Tên công việc	Người người thực hiện
Thiết kế lớp	Cả nhóm
Login()	Đoàn Quang Tuấn
Connect()	Đoàn Quang Tuấn
Send()	Đoàn Quang Tuấn
Receive()	Đoàn Quang Tuấn
displayMessage()	Đoàn Quang Tuấn
Action()	Đoàn Quang Tuấn
LS	Đoàn Quang Tuấn
DIR	Đoàn Quang Tuấn
PWD	Đoàn Quang Tuấn
CD	Phạm Minh Chiến
LCD	Lê Hoàng Sang
GET	Lê Hoàng Sang
MGET	Lê Hoàng Sang
PUT	Phạm Minh Chiến
MPUT	Phạm Minh Chiến
DEL	Phạm Minh Chiến
MDEL	Phạm Minh Chiến
MKDIR	Lê Hoàng Sang
RMDIR	Lê Hoàng Sang
PASV	Phạm Minh Chiến
QUIT	Lê Hoàng Sang
USER	Lê Hoàng Sang
PASS	Lê Hoàng Sang
CLEAR	Lê Hoàng Sang
HELP	Lê Hoàng Sang
standardizedCMD	Lê Hoàng Sang
defineOrder()	Lê Hoàng Sang
getClauses()	Lê Hoàng Sang
getCmd()	Lê Hoàng Sang
getDataPort()	Phạm Minh Chiến
getServerCode()	Phạm Minh Chiến
openPort()	Phạm Minh Chiến
openPassiveConnect()	Phạm Minh Chiến
openActiveConnect()	Phạm Minh Chiến
Active()	Đoàn Quang Tuấn

Passive()	Đoàn Quang Tuấn
Bắt và phân tích gói tin	Lê Hoàng Sang
Kiểm thử	Cả nhóm
Viết báo cáo	Cả nhóm

2. NHỮNG CHỨC NĂNG CHÍNH VÀ CẤU TRÚC TỔ CHỨC

a. Chức năng chương trình:

Hỗ trợ đầy đủ các tính năng của một FTP-Client, cả chế độ Active và Passive (mặc định chương trình sẽ ở chế độ active):

- Login : đăng nhập vào chương trình.
- Open : mở kết nối tới server.
- ls : liệt kê các tập tin và thư mục trên server.
- dir : liệt kê các tập tin và thư mục trên server nhưng chi tiết hơn ls.
- pwd : hiện đường dẫn hiện tại trên server .
- cd : thay đổi đường dẫn trên server (trường hợp cd <path> thì thay đổi đường dẫn, nếu trường hợp cd thì hiển thị đường dẫn hiện tại)
- lcd : thay đổi đường dẫn dưới client (trường hợp lcd <path> thì thay đổi đường dẫn, nếu trường hợp lcd thì hiển thị đường dẫn hiện tại)
- get : tải xuống một file về từ server.
- mget : tải xuống nhiều file về từ server.
- put : tải một file lên server.
- mput : tải nhiều file lên server.
- del : xóa tập tin, thư mục trên server.
- mdel : xóa nhiều tập tin, thư mục trên server.
- mkdir : tạo thư mục mới.
- rmdir : xóa thư mục.
- pasv : gửi lệnh command port ở chế độ passive.
- passive : chuyển qua chế độ Passive.
- active : chuyển qua chế độ Active .
- clear : xóa màn hình .
- help : hiện thông báo trợ giúp.

b. Cấu trúc chương trình

i. Tổ chức chương trình thành một **lớp FTPClient** chứa:

- cmdClient: CSocket - là command Client để gửi lệnh lên server.
- user: string - lưu tên người dùng.
- password: string - lưu mật khẩu người dùng.
- Request: string - chứa lệnh cần gửi lên server.
- Response: string - nơi chứa phản hồi của server.
- Mode: bool - cờ hiệu chế độ hiện tại 0 là passive, 1 là active.
- isLoggedIn: bool - cờ kiểm tra đăng nhập thành công.

- isConnected: bool – cờ kiểm tra kết nối thành công tới server.
- Argument: vector<string> - chứa danh sách tham số của lệnh cmd.
- **dataClient** được khởi tạo từ hàm openPort() tùy thuộc vào chế độ passive hay active mà phát sinh một dataClient tương ứng.

ii. Cung cấp các hàm:

Chỉ public một hàm duy nhất là hàm getCmd() nhận vào một chuỗi và xử lý chuỗi đó bằng những hàm chức năng tương ứng.

iii. Bảng mô tả các hàm chức năng:

Các tham số truyền vào chủ yếu là tên file sẽ được push vào vector argument nên class chỉ vào kiểm tra argument có những tham số nào và pop ra để sử dụng nên hàm chức năng sẽ không có tham số truyền vào.

Tên hàm	Chức năng
<code>void getCmd();</code>	Thực hiện một vòng lặp vô tận cho tới khi nhận được lệnh Quit. Đầu tiên yêu cầu đăng nhập, kiểm tra Login và nhận vào chuỗi người dùng nhập, gọi tiếp đến hàm defineOrder() để xác định hàm nào sẽ được gọi (hoặc báo không hợp lệ). Gọi tiếp hàm getClauses() để push các tham số (chủ yếu là tên files/folders) vào vector argument. Sau khi có được order của lệnh sẽ gọi đến lệnh đó.
<code>void getClauses(string cmd);</code>	Nhận vào một chuỗi và push các thành phần của chuỗi (tên file, tên thư mục) vào vector argument.
<code>int defineOrder(string cmd);</code>	Trả về một số nguyên xác định vị trí của chuỗi cmd truyền vào, nếu chuỗi cmd không hợp lệ trả về -1.
<code>String standardizedCMD(string cmd)</code>	Hàm này sẽ chuẩn hóa chuỗi cmd người dùng nhập vào như cắt bỏ khoảng trắng dư thừa, chuyển hết chữ cái về dạng lowercase. Trả về chuỗi đã chuẩn hóa.
<code>int getDataPort();</code>	Hàm này xử lý chuỗi this->response mà server gửi về có dạng “...(ipadress, a, b)” hàm sẽ tách a, b ra và trả về giá trị $a*256 + b$. Nếu chuỗi trong this->response không đúng cấu trúc, hàm trả về 0.

<code>int</code> getServerCode();	Hàm xử lí chuỗi this->response được nhận cmdClient nhận về từ server có dạng “xxx Chuỗi”. Hàm trả về giá trị xxx.
<code>CSocket*</code> openPort();	Hàm gọi đến 1 trong hai hàm openPassiveConnect() hoặc OpenActiveConnect() tùy thuộc vào cờ this->mode là 0 hay 1.
<code>CSocket*</code> openPassiveConnect();	Tạo một CSocket dataClient mới, cmdClient gửi lên pasv lên server. dataClient sẽ kết nối đến địa chỉ this->hostIP tại port có giá trị bằng this->getDataPort(). Hàm trả về địa chỉ của dataClient vừa tạo.
<code>CSocket*</code> openActiveConnect();	Tạo một CSocket dataClient mới. cmdClient lấy port của dataClient và gửi lên server để báo server là dataClient đang listen. Hàm trả về địa chỉ của dataClient vừa tạo.
<code>bool</code> connect();	Kiểm tra xem hiện tại có đang kết nối vào server nào hay không.
<code>bool</code> login();	Kiểm tra xem người dùng đã login vào server hay chưa.
<code>void</code> cmd_user();	Gửi lệnh USER lên server và nhận phản hồi.
<code>void</code> cmd_pass();	Gửi lệnh PASS lên server và nhận phản hồi.
<code>void</code> send();	Gửi lệnh được lưu trong this->request lên server.
<code>int</code> receive();	Nhận thông điệp từ server vào gán vào this->response.
<code>void</code> displayMessage();	Xử lí thông điệp trong response và xuất ra màn hình.
<code>void</code> action();	Gọi liên tiếp hàm this->send(); this->receive(); this->displayMessage();
<code>void</code> cmd_list_core(string cmd);	Mở port ở chế độ passive hoặc active, cmdClient gửi lệnh chứa trong string cmd lên server và dataClient nhận dữ liệu từ server và xuất ra màn hình.

<code>void cmd_ls();</code>	Gọi lại hàm <code>cmd_list_core("NLST")</code> ; để gửi lệnh NLST lên server.
<code>void cmd_dir();</code>	Gọi lại hàm <code>cmd_list_core("LIST")</code> ; để gửi lệnh NLST lên server.
<code>void cmd_pwd();</code>	<code>cmdClient</code> gửi lệnh PWD lên server và nhận phản hồi.
<code>void cmd_cd();</code>	<code>cmdClient</code> gửi lệnh CWD lên server và nhận phản hồi.
<code>void cmd_lcd();</code>	Thay đổi địa chỉ hiện tại của FTP-Client bằng hàm <code>setCurrentDirectory()</code> của <code>window</code> . Nếu không có tham số sẽ trả về địa chỉ hiện của Client.
<code>void cmd_get_core(string file);</code>	<ul style="list-style-type: none"> • Mở sẵn một file mới có tên "file" để ghi tại thư mục hiện hành của FTP Client. • <code>cmdClient</code> gửi lệnh RETR + file lên server và chờ phản hồi, nếu phản hồi thành công sẽ mở port. • <code>dataClient</code> sẽ nhận dữ liệu từ server (dạng stream) và ghi vào file đã mở sẵn.
<code>void cmd_get();</code>	Gọi lại hàm <code>cmd_get_put(argument[0])</code> .
<code>void cmd_mget();</code>	Gọi lại hàm <code>cmd_get</code> với tham số truyền vào là lần lượt các giá trị có trong vector <code>argument</code> . Sẽ có thông báo xác nhận xem có thực sự muốn tải xuống cho từng file, nhập N sẽ bỏ qua, Enter để xác nhận.
<code>bool cmd_put_core(string file);</code>	<ul style="list-style-type: none"> • Mở sẵn một file có tên "file" để đọc nhị phân tại thư mục hiện hành. • <code>cmdClient</code> gửi lệnh STOR + file lên server và chờ phản hồi, nếu phản hồi thành công sẽ mở port. • <code>dataClient</code> sẽ đọc cho tới hết dữ liệu trong file đã mở và truyền lên server bằng lệnh <code>send()</code>.
<code>void cmd_put();</code>	Gọi lại hàm <code>cmd_put_core</code> với tham số là <code>this->argument[0]</code> .
<code>void cmd_mput();</code>	Gọi lại hàm <code>cmd_put</code> với tham số là lần lượt các giá trị có trong vector <code>argument</code> . Sẽ có thông báo xác

	nhận xem có thực sự muốn tải lên cho từng file, nhập N sẽ bỏ qua, Enter để xác nhận.
<code>void cmd_del();</code>	Gửi lệnh DELE + tham số this->argument[0] lên server và chờ phản hồi.
<code>void cmd_mdel();</code>	Gọi lại n lần gàm cmd_del với mỗi lần là từng giá trị trong vector argument. Sẽ có thông báo xác nhận xem có thực sự muốn xóa từng file không, nhập N sẽ bỏ qua, Enter để xác nhận.
<code>void cmd_mkdir();</code>	Gửi lên server lệnh MKD + chuỗi this->argument[0] và chờ phản hồi.
<code>void cmd_rmdir();</code>	Gửi lên server lệnh MKD + chuỗi this->argument[0] và chờ phản hồi.
<code>void cmd_pasv();</code>	Gửi lệnh pasv lên server nếu là IPv4 hoặc epsv nếu là IPv6 và chờ phản hồi.
<code>void cmd_active();</code>	Nếu chế độ hiện tại là passive thì sẽ chuyển sang active, ngược lại sẽ xuất thông báo và không làm gì. Hàm bật cờ mode bằng ACTIVE.
<code>void cmd_passive();</code>	Nếu chế độ hiện tại là active thì sẽ chuyển sang passive, ngược lại sẽ xuất thông báo và không làm gì. Hàm bật cờ mode bằng PASSIVE.
<code>void cmd_quit();</code>	Gửi lệnh quit lên server và chờ phản hồi. Sau đó đóng chương trình.
<code>void cmd_help();</code>	Xuất ra tất cả lệnh mà chương trình hỗ trợ.
<code>void cmd_clear();</code>	Xóa màn hình hiện tại.

c. Mô tả các hàm:

Các hàm chức năng trong chương trình chia làm hai loại:

- Loại 1: các hàm command: del, mdel, pwd, cd, lcd, quit, help, user, pass.

CSocket cmdClient chỉ việc gửi lệnh lên và nhận về phản hồi.

- Loại 2: các hàm cần truyền tải dữ liệu: ls, get, put

Sau khi cmdClient gửi command lên server, nếu phản hồi thành công thì FPTClient sẽ thực hiện mở port, tạo dataClient để nhận dữ liệu từ server, hai dạng passive và active có đôi chút khác nhau ở chỗ port và lúc chuẩn bị truyền tải dữ liệu, còn kĩ thuật truyền/nhận thì giống nhau hoàn toàn. Passive thì sau khi mở port sẽ dùng ngay dataClient truyền tải ngay (vì đã kết nối với server rồi). Còn Active thì chưa, ở chế độ này dataClient chỉ mới có thể listen kết nối của server và không thể kết nối với server được, do đó cần có một CSocket Connector khác giữ kết nối với server mỗi khi dataClient accept thành công với server, Connector này mới là đối tượng trực tiếp truyền/nhận dữ liệu với server. Và một chú ý nữa là mỗi lần gọi ls, get, put ta phải mở một port mới vì port cũ sau khi dùng xong phía server sẽ đóng kết nối.

- **Hàm CSocket* openPassiveConnect():**

Như đã nêu bên trên, hàm này sẽ mở port ở chế độ passive. Đầu tiên ta tạo một CSocket mới có tên là dataClient với port do window xác định. Sau đó cmdClient sẽ gửi một lệnh PASV lên server để thông báo rằng chương trình muốn mở một kết nối passive với server, ta cần một thông điệp của server có nội dung "xxx stringyyy (ip address, a, b). Khi nhận được thông điệp này có nghĩa là server đã cấp cho ta một port dữ liệu, nhiệm vụ của ta là cho dataClient kết nối vào port có giá trị $(a*256+b)$ là xong.

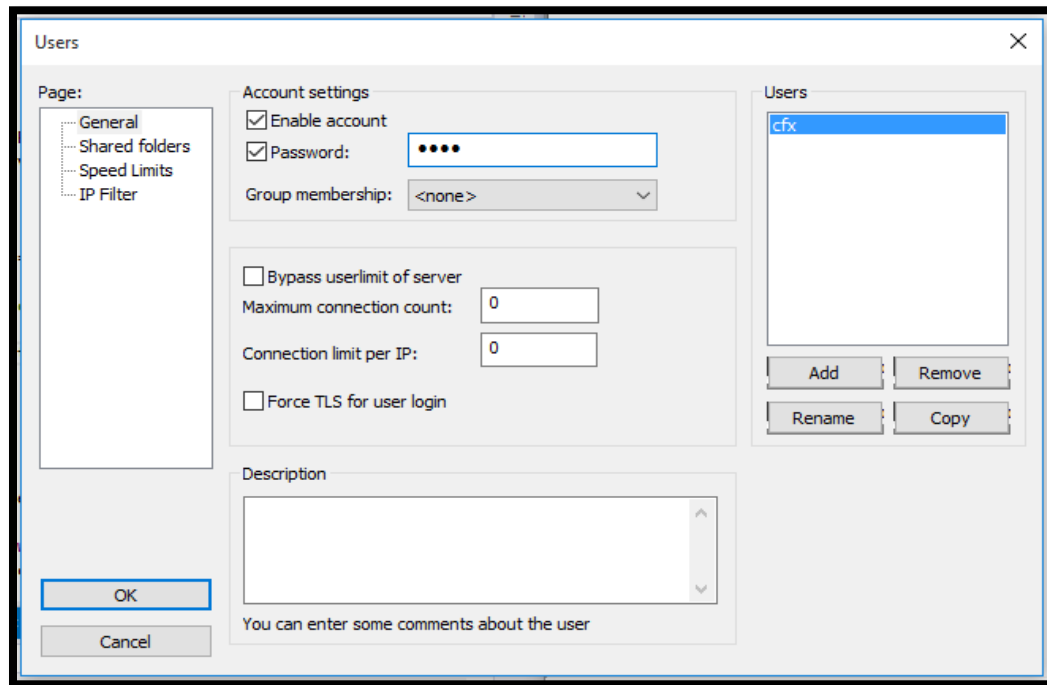
- **Hàm openActiveConnect():**

Ở chế độ Active ta mong muốn rằng server sẽ chủ động kết nối với ta. Tuy nhiên muốn vậy ta phải thông báo cho phía server biết ta đang chờ một kết nối từ server bằng lệnh command port "PORT ip, a, b" với ip là địa chỉ ip của Socket đang chờ server kết nối tới, $a = \text{port}/256$ và $b = \text{port} \% 256$, đó là quy tắc. Tóm lại ta sẽ tạo một dataClient với port do window xác định, dùng cmdClient gửi "thông tin" của dataClient cho server biết mà liên lạc với dataClient.

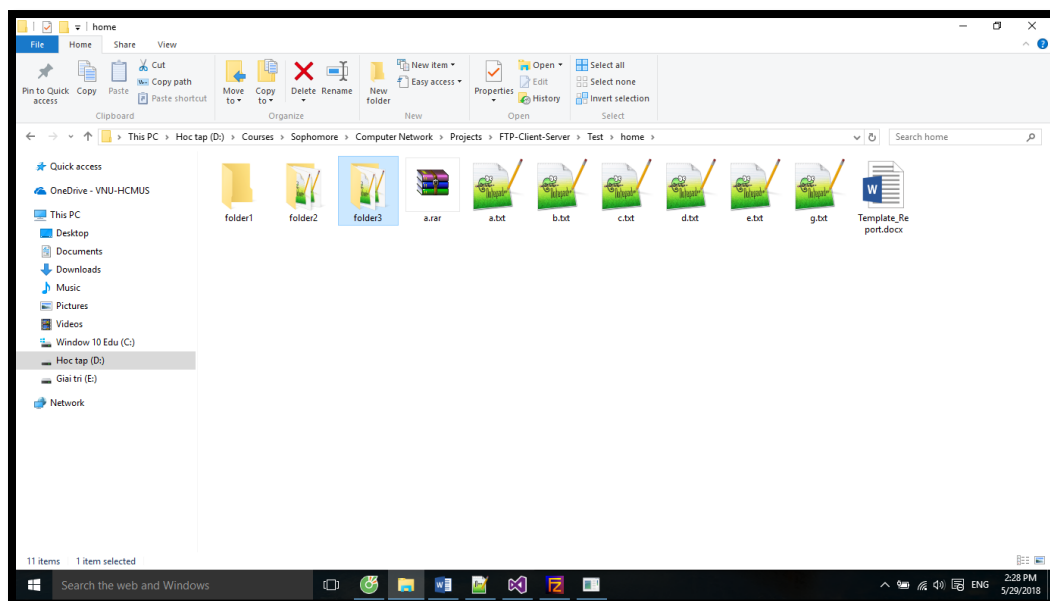
Khác với Passive, chế độ Active này dataClient chỉ có chức năng listen một kết nối từ server chứ tuyệt nhiên không thể truyền nhận trực tiếp với server bởi không thể connect tới server. Cho nên, mỗi lần truyền gửi (ls, get, put) ta sẽ tạo thêm một Socket Connector để giữ kết nối với server, sau lệnh accept(Connector) nếu thành công thì Connector giờ sẽ là đối tượng truyền dữ liệu trực tiếp với server, còn dataClient sẽ tiếp tục listen().

3. CHẠY THỬ VÀ KẾT QUẢ

- Dùng phần mềm **File Zilla Server**.
- Cấu hình user trên File Zilla Server:

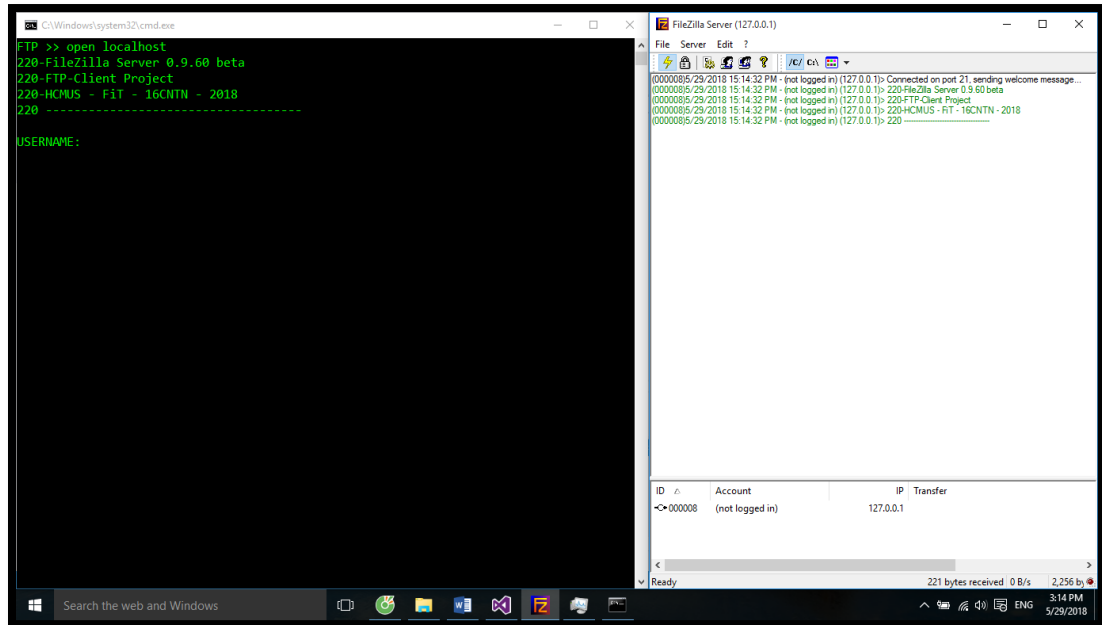


- + Tên đăng nhập: cfx
 - + Mật khẩu: 1234
 - + Quyền truy cập: Full access
- Nội dung thư mục localhost:

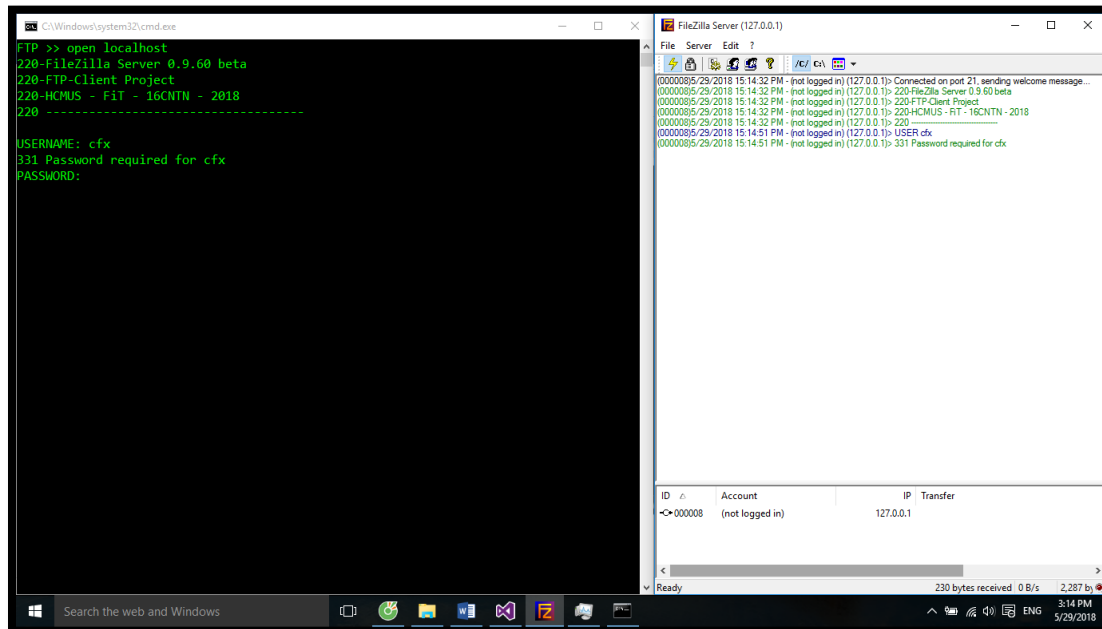


a. Thử chế độ active (mặc định):

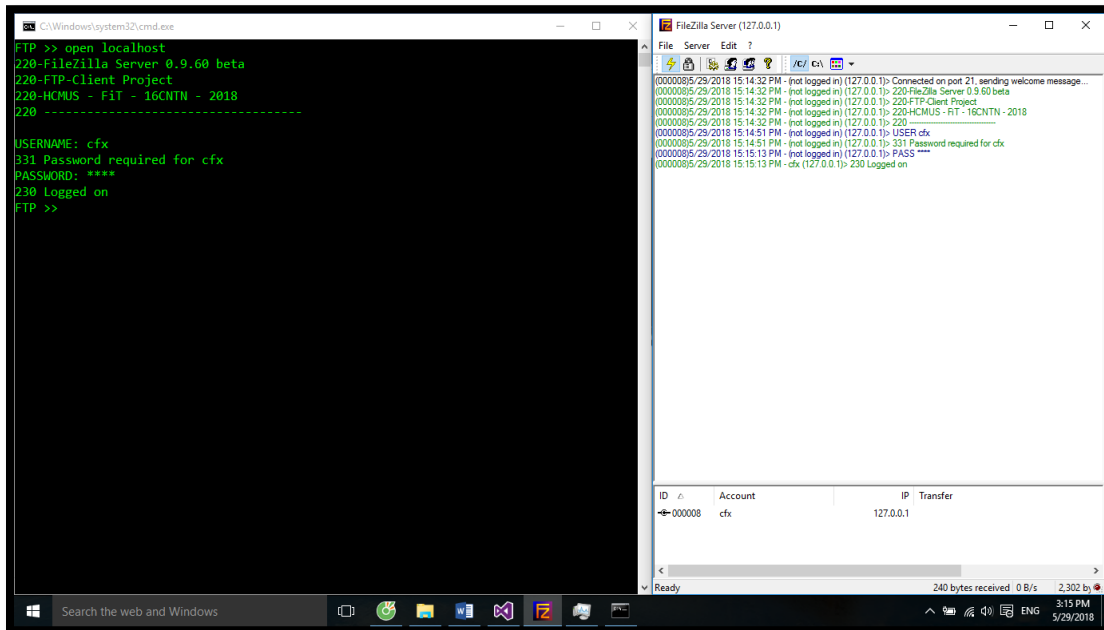
- Connect:



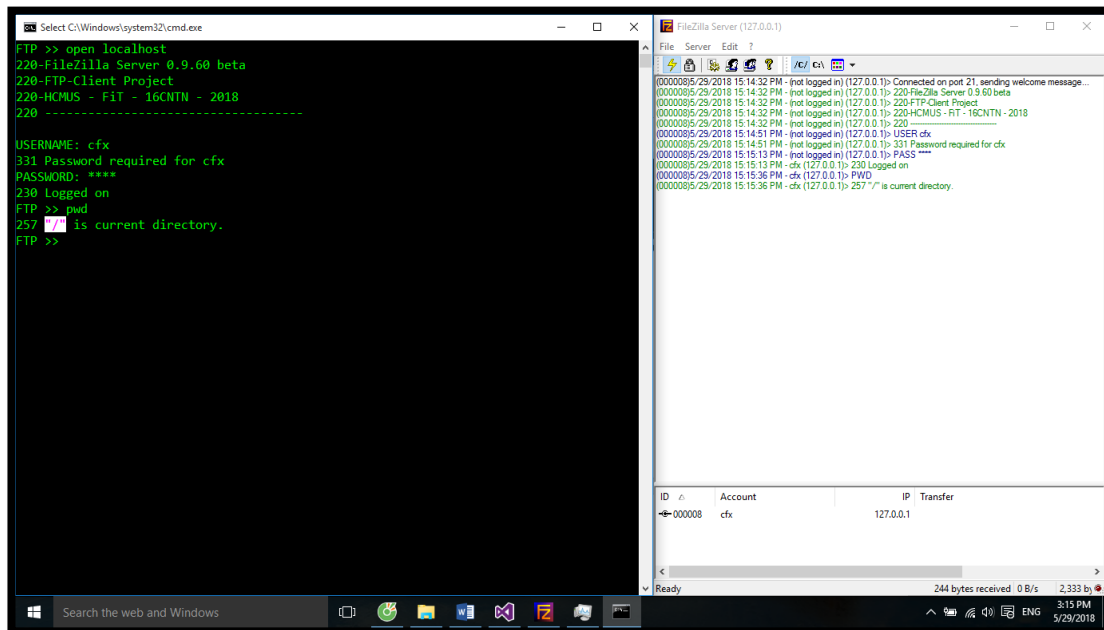
- Login user:



- Password:



- pwd



- cd:

The screenshot shows a Windows command prompt window titled "Select C:\Windows\system32\cmd.exe" and a FileZilla Server window titled "FileZilla Server (127.0.0.1)".

In the command prompt, the following commands and outputs are visible:

```

220-HCMUS - FiT - 16CNTN - 2018
220 -----
USERNAME: cfx
331 Password required for cfx
PASSWORD: ****
230 Logged on
FTP >> pwd
257 "/" is current directory.
FTP >> cd
550 CWD failed. "/cd": directory not found.
FTP >> ls
200 Port command successful
150 Opening data channel for directory listing of "/"
a.rar
b.txt
c.txt
d.txt
e.txt
folder1
folder2
folder3
g.txt
Template_Report.docx
226 Successfully transferred "/"
FTP >> cd folder1
250 CWD successful. "/folder1" is current directory.
FTP >>

```

In the FileZilla Server window, the log shows the following entries:

```

(000008/29/2018 15:14:32 PM - (not logged in) (127.0.0.1) Connected on port 21, sending welcome message...
(000008/29/2018 15:14:32 PM - (not logged in) (127.0.0.1) 220-FileZilla Server 0.9.60 beta
(000008/29/2018 15:14:32 PM - (not logged in) (127.0.0.1) 220-FTP-Client Project
(000008/29/2018 15:14:32 PM - (not logged in) (127.0.0.1) 220-HCMUS - FiT - 16CNTN - 2018
(000008/29/2018 15:14:32 PM - (not logged in) (127.0.0.1) 220 -----
(000008/29/2018 15:14:51 PM - (not logged in) (127.0.0.1) USER cfx
(000008/29/2018 15:14:51 PM - (not logged in) (127.0.0.1) 331 Password required for cfx
(000008/29/2018 15:15:13 PM - (not logged in) (127.0.0.1) PASS ****
(000008/29/2018 15:15:13 PM - cfx (127.0.0.1) 230 Logged on
(000008/29/2018 15:15:36 PM - cfx (127.0.0.1) PWD
(000008/29/2018 15:15:36 PM - cfx (127.0.0.1) 257 "/" is current directory.
(000008/29/2018 15:15:53 PM - cfx (127.0.0.1) 550 CWD failed. "/cd" directory not found.
(000008/29/2018 15:16:10 PM - cfx (127.0.0.1) PORT 127.0.0.1,239,51
(000008/29/2018 15:16:10 PM - cfx (127.0.0.1) 200 Port command successful
(000008/29/2018 15:16:10 PM - cfx (127.0.0.1) NLST
(000008/29/2018 15:16:10 PM - cfx (127.0.0.1) 150 Opening data channel for directory listing of "/"
(000008/29/2018 15:16:10 PM - cfx (127.0.0.1) 226 Successfully transferred "/"
(000008/29/2018 15:16:45 PM - cfx (127.0.0.1) 250 CWD folder1
(000008/29/2018 15:16:45 PM - cfx (127.0.0.1) 250 CWD successful. "/folder1" is current directory.

```

The FileZilla Server window also displays a table with the following data:

ID	Account	IP	Transfer
000008	cfx	127.0.0.1	

The status bar at the bottom of the FileZilla Server window shows "Ready", "294 bytes received 0 B/s", and "2.641 B/s".

- lcd

The screenshot shows a Windows command prompt window titled "C:\Windows\system32\cmd.exe" and a FileZilla Server window titled "FileZilla Server (127.0.0.1)".

In the command prompt, the following commands and outputs are visible:

```

FTP >> open localhost
220-FileZilla Server 0.9.60 beta
220-FTP-Client Project
220-HCMUS - FiT - 16CNTN - 2018
220 -----
USERNAME: cfx
331 Password required for cfx
PASSWORD: ****
230 Logged on
FTP >> pwd
257 "/" is current directory.
FTP >> cd folder1
folder1

250 CWD successful. "/folder1" is current directory.
FTP >> lcd
Local directory now D:\Courses\Sophomore\Computer Network\Projects\FTP-Client-Server\FTPClient\FTPClient
FTP >> lcd d:\
Local directory now d:\
FTP >>

```

In the FileZilla Server window, the log shows the following entries:

```

(000018/29/2018 23:09:01 PM - (not logged in) (127.0.0.1) Connected on port 21, sending welcome message...
(000018/29/2018 23:09:01 PM - (not logged in) (127.0.0.1) 220-FileZilla Server 0.9.60 beta
(000018/29/2018 23:09:01 PM - (not logged in) (127.0.0.1) 220-FTP-Client Project
(000018/29/2018 23:09:01 PM - (not logged in) (127.0.0.1) 220-HCMUS - FiT - 16CNTN - 2018
(000018/29/2018 23:09:01 PM - (not logged in) (127.0.0.1) 220 -----
(000018/29/2018 23:09:04 PM - (not logged in) (127.0.0.1) USER cfx
(000018/29/2018 23:09:04 PM - (not logged in) (127.0.0.1) 331 Password required for cfx
(000018/29/2018 23:09:04 PM - (not logged in) (127.0.0.1) PASS ****
(000018/29/2018 23:09:04 PM - cfx (127.0.0.1) 230 Logged on
(000018/29/2018 23:09:10 PM - cfx (127.0.0.1) PWD
(000018/29/2018 23:09:10 PM - cfx (127.0.0.1) 257 "/" is current directory.
(000018/29/2018 23:09:24 PM - cfx (127.0.0.1) CWD folder1
(000018/29/2018 23:09:24 PM - cfx (127.0.0.1) 250 CWD successful. "/folder1" is current directory.

```

The FileZilla Server window also displays a table with the following data:

ID	Account	IP	Transfer
000018	cfx	127.0.0.1	

The status bar at the bottom of the FileZilla Server window shows "Ready", "234 bytes received 0 B/s", and "85,518 bytes sent 0 B/s".

- ls

The screenshot shows the FileZilla Server (127.0.0.1) and FileZilla Client (0.9.60 beta) interface. The client window displays the following commands and output:

```

FTP >> open localhost
220-FileZilla Server 0.9.60 beta
220-FTP-Client Project
220-HCMUS - FIT - 16CNTN - 2018
220 -----
USERNAME: cfx
331 Password required for cfx
PASSWORD: ****
230 Logged on
FTP >> pwd
257 "/" is current directory.
FTP >> cd
550 CWD failed, "/cd": directory not found.
FTP >> ls
200 Port command successful
150 Opening data channel for directory listing of "/"
a.rar
b.txt
c.txt
d.txt
e.txt
folder1
folder2
folder3
g.txt
Template_Report.docx
226 Successfully transferred "/"
FTP >>

```

The server window shows the corresponding log entries, including the connection, login, and directory listing process.

- del

The screenshot shows the FileZilla Server (127.0.0.1) and FileZilla Client (0.9.60 beta) interface. The client window displays the following commands and output:

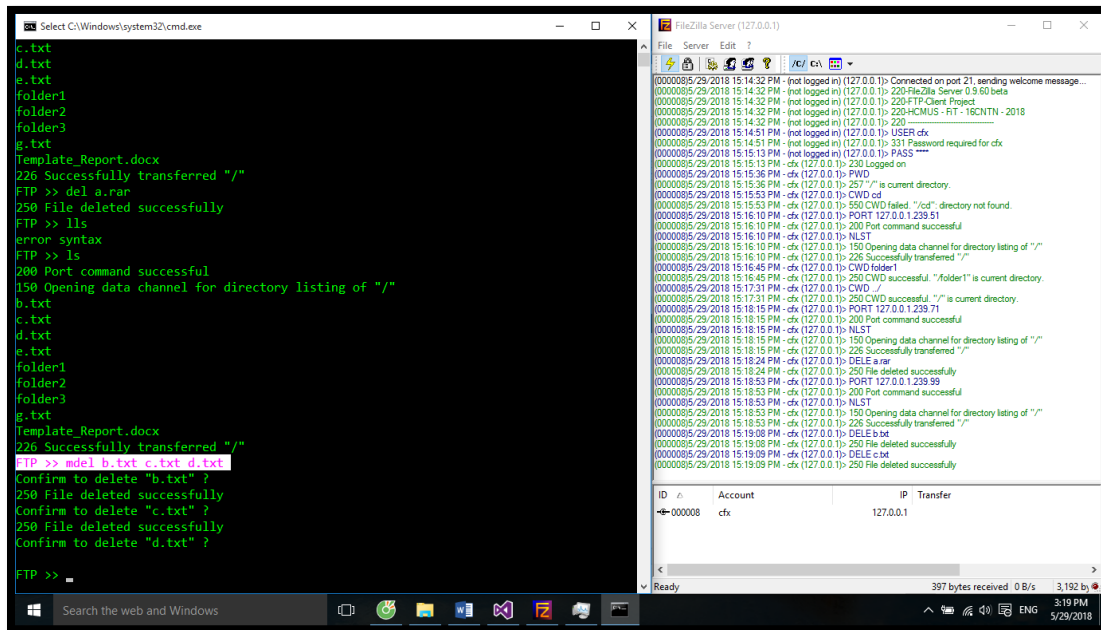
```

FTP >> ls
200 Port command successful
150 Opening data channel for directory listing of "/"
a.rar
b.txt
c.txt
d.txt
e.txt
folder1
folder2
folder3
g.txt
Template_Report.docx
226 Successfully transferred "/"
FTP >> del a.rar
250 File deleted successfully
FTP >>

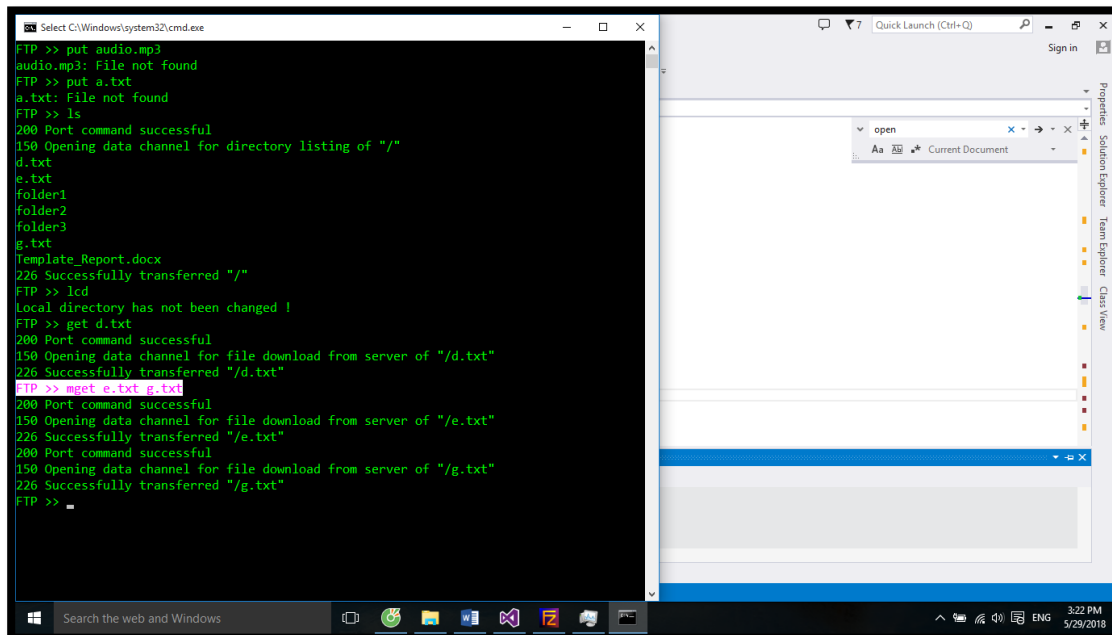
```

The server window shows the corresponding log entries, including the connection, login, and file deletion process.

- mdel (bỏ qua file d.txt nếu nhấn N)



- get & mget



- put: uploading file “audio.mp3” 382MB

The screenshot shows a Windows command prompt window with the following commands and output:

```

C:\Windows\system32\cmd.exe
FTP >> put audio.mp3
200 Port command successful
150 Opening data channel for file upload to server of "/audio.mp3"

```

Simultaneously, the FileZilla Server (127.0.0.1) window displays a log of activities. The relevant entries for the upload are:

```

000008/29/2018 15:24:00 PM - cfx (127.0.0.1) 150 Opening data channel for file upload to server of "/audio.mp3"
000008/29/2018 15:24:00 PM - cfx (127.0.0.1) 226 Successfully transferred "/audio.mp3"

```

The FileZilla Server window also shows a table of active connections:

ID	Account	IP	Transfer
000008	cfx	127.0.0.1	/audio.mp3

The status bar at the bottom of the FileZilla Server window indicates: Ready, 123,863,585 bytes received, 823.5 KB, 3:24 PM, 5/29/2018.

- mput

The screenshot shows a Windows command prompt window with the following commands and output:

```

C:\Windows\system32\cmd.exe
220-HCMUS - FIT - 16CNTN - 2018
220 -----
USERNAME: cfx
331 Password required for cfx
PASSWORD: ****
230 Logged on
FTP >> ls
200 Port command successful
150 Opening data channel for directory listing of "/"
audio.mp3
d.txt
e.txt
folder1
folder2
folder3
ftplibclient.cpp
g.txt
Template_Report.docx
226 Successfully transferred "/"
FTP >> mput a.rar anh.jpg g.txt
Upload "a.rar"?
200 Port command successful
150 Opening data channel for file upload to server of "/a.rar"
226 Successfully transferred "/a.rar"
Upload "anh.jpg"?
200 Port command successful
150 Opening data channel for file upload to server of "/anh.jpg"
226 Successfully transferred "/anh.jpg"
Upload "g.txt"?
200 Port command successful
150 Opening data channel for file upload to server of "/g.txt"
226 Successfully transferred "/g.txt"
FTP >>

```

The FileZilla Server (127.0.0.1) window displays a log of activities. The relevant entries for the mput command are:

```

000011/29/2018 22:28:57 PM - cfx (127.0.0.1) 150 Opening data channel for file upload to server of "/a.rar"
000011/29/2018 22:28:57 PM - cfx (127.0.0.1) 226 Successfully transferred "/a.rar"
000011/29/2018 22:28:57 PM - cfx (127.0.0.1) 150 Opening data channel for file upload to server of "/anh.jpg"
000011/29/2018 22:28:57 PM - cfx (127.0.0.1) 226 Successfully transferred "/anh.jpg"
000011/29/2018 22:28:57 PM - cfx (127.0.0.1) 150 Opening data channel for file upload to server of "/g.txt"
000011/29/2018 22:28:57 PM - cfx (127.0.0.1) 226 Successfully transferred "/g.txt"

```

The FileZilla Server window also shows a table of active connections:

ID	Account	IP	Transfer
000011	cfx	127.0.0.1	

The status bar at the bottom of the FileZilla Server window indicates: Ready, 937,956 bytes received, 0 B/s, 926 bytes sent, 0 B/s, 10:29 PM, 5/29/2018.

- Ls và passive (chuyển qua chế độ passive)

The screenshot shows the FileZilla Server interface on the right and the FileZilla Client terminal on the left. The client terminal displays the following commands and responses:

```

230 Logged on
FTP >> ls
200 Port command successful
150 Opening data channel for directory listing of "/"
a.rar
anh.jpg
audio.mp3
d.txt
e.txt
folder1
folder2
folder3
ftplib.cpp
g.txt
Template_Report.docx
226 Successfully transferred "/"
FTP >> passive
Switch to PASSIVE mode successful.
FTP >> ls
227 Entering Passive Mode (127,0,0,1,250,232)
150 Opening data channel for directory listing of "/"
a.rar
anh.jpg
audio.mp3
d.txt
e.txt
folder1
folder2
folder3
ftplib.cpp
g.txt
Template_Report.docx
226 Successfully transferred "/"
FTP >>

```

The FileZilla Server log on the right shows the corresponding server-side actions, including logging in, directory listing, and the transition to passive mode.

- dir

The screenshot shows the FileZilla Server interface on the right and the FileZilla Client terminal on the left. The client terminal displays the following commands and responses:

```

FTP >> dir
200 Port command successful
150 Opening data channel for directory listing of "/"
-rw-r--r-- 1 ftp ftp      854413 May 29 22:28 a.rar
-rw-r--r-- 1 ftp ftp      83344 May 29 22:28 anh.jpg
-rw-r--r-- 1 ftp ftp    393956875 May 29 15:24 audio.mp3
-rw-r--r-- 1 ftp ftp           0 May 29 15:35 d.txt
-rw-r--r-- 1 ftp ftp           0 May 29 15:28 e.txt
drwxr-xr-x 1 ftp ftp           0 May 15 21:49 folder1
drwxr-xr-x 1 ftp ftp           0 May 05 13:27 folder2
drwxr-xr-x 1 ftp ftp           0 May 15 22:03 folder3
-rw-r--r-- 1 ftp ftp           0 May 29 15:34 ftplib.cpp
-rw-r--r-- 1 ftp ftp           36 May 29 22:28 g.txt
-rw-r--r-- 1 ftp ftp      94469 May 31 00:28 img_1.png
-rw-r--r-- 1 ftp ftp      78897 May 31 00:28 img_2.png
-rw-r--r-- 1 ftp ftp      98111 May 31 00:28 img_3.png
-rw-r--r-- 1 ftp ftp    107119 Apr 24 2018 Template_Report.docx
226 Successfully transferred "/"
FTP >>

```

The FileZilla Server log on the right shows the corresponding server-side actions, including logging in, directory listing, and the transfer of the directory listing data.

b. Test chế độ Passive:

- Is

The screenshot shows a Windows command prompt window and a FileZilla Server window. The command prompt displays the following commands and output:

```
C:\Windows\system32\cmd.exe
226 Successfully transferred "/"
FTP >> passive
Switch to PASSIVE mode successful.
FTP >> ls
227 Entering Passive Mode (127,0,0,1,253,237)
150 Opening data channel for directory listing of "/"
a.rar
anh.jpg
audio.mp3
d.txt
e.txt
folder1
folder2
folder3
ftplib.cpp
g.txt
Template_Report.docx
226 Successfully transferred "/"
FTP >>
```

The FileZilla Server window shows the log of the connection. The log includes the following entries:

```
[000018/5/29/2018 23:09:01 PM - (not logged in) (127.0.0.1): Connected on port 21, sending welcome message...
[000018/5/29/2018 23:09:01 PM - (not logged in) (127.0.0.1): 220 FileZilla Server 0.9.60 beta
[000018/5/29/2018 23:09:01 PM - (not logged in) (127.0.0.1): 220-FTP-Client Project
[000018/5/29/2018 23:09:01 PM - (not logged in) (127.0.0.1): 220-HCMUS - FT - 16CNTN - 2018
[000018/5/29/2018 23:09:01 PM - (not logged in) (127.0.0.1): 220
[000018/5/29/2018 23:09:04 PM - (not logged in) (127.0.0.1): USER cfx
[000018/5/29/2018 23:09:04 PM - (not logged in) (127.0.0.1): 331 Password required for cfx
[000018/5/29/2018 23:09:04 PM - (not logged in) (127.0.0.1): PASS
[000018/5/29/2018 23:09:04 PM - cfx (127.0.0.1): 230 Logged on
[000018/5/29/2018 23:09:10 PM - cfx (127.0.0.1): PWD
[000018/5/29/2018 23:09:10 PM - cfx (127.0.0.1): 257 "/" is current directory.
[000018/5/29/2018 23:09:24 PM - cfx (127.0.0.1): CWD folder1
[000018/5/29/2018 23:09:24 PM - cfx (127.0.0.1): 250 CWD successful. "folder1" is current directory.
[000018/5/29/2018 23:10:34 PM - cfx (127.0.0.1): PORT 127.0.0.1,250,59
[000018/5/29/2018 23:10:34 PM - cfx (127.0.0.1): 200 Port command successful
[000018/5/29/2018 23:10:34 PM - cfx (127.0.0.1): NLST
[000018/5/29/2018 23:10:34 PM - cfx (127.0.0.1): 150 Opening data channel for directory listing of "folder1"
[000018/5/29/2018 23:10:34 PM - cfx (127.0.0.1): 226 Successfully transferred "folder1"
[000018/5/29/2018 23:11:03 PM - cfx (127.0.0.1): CWD /
[000018/5/29/2018 23:11:03 PM - cfx (127.0.0.1): 250 CWD successful. "/" is current directory.
[000018/5/29/2018 23:11:05 PM - cfx (127.0.0.1): 200 Port command successful
[000018/5/29/2018 23:11:05 PM - cfx (127.0.0.1): NLST
[000018/5/29/2018 23:11:05 PM - cfx (127.0.0.1): 150 Opening data channel for directory listing of "/"
[000018/5/29/2018 23:11:05 PM - cfx (127.0.0.1): 226 Successfully transferred "/"
[000018/5/29/2018 23:12:25 PM - cfx (127.0.0.1): PASS
[000018/5/29/2018 23:12:25 PM - cfx (127.0.0.1): 227 Entering Passive Mode (127.0.0.1,253,237)
[000018/5/29/2018 23:12:25 PM - cfx (127.0.0.1): NLST
[000018/5/29/2018 23:12:25 PM - cfx (127.0.0.1): 150 Opening data channel for directory listing of "/"
[000018/5/29/2018 23:12:25 PM - cfx (127.0.0.1): 226 Successfully transferred "/"
```

- Dir

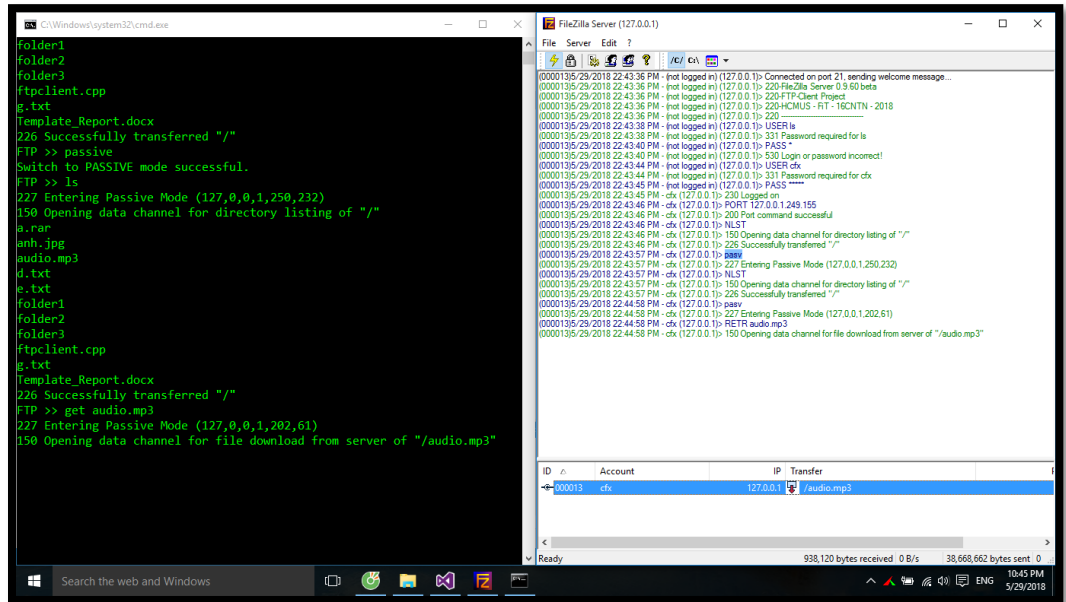
The screenshot shows a Windows command prompt window and a FileZilla Server window. The command prompt displays the following commands and output:

```
C:\Windows\system32\cmd.exe
FTP >> dir
227 Entering Passive Mode (127,0,0,1,211,234)
150 Opening data channel for directory listing of "/"
-rw-r--r-- 1 ftp ftp      854413 May 29 22:28 a.rar
-rw-r--r-- 1 ftp ftp      83344 May 29 22:28 anh.jpg
-rw-r--r-- 1 ftp ftp    393956875 May 29 15:24 audio.mp3
-rw-r--r-- 1 ftp ftp           0 May 29 15:35 d.txt
-rw-r--r-- 1 ftp ftp           0 May 29 15:28 e.txt
drwxr-xr-x 1 ftp ftp           0 May 15 21:49 folder1
drwxr-xr-x 1 ftp ftp           0 May 05 13:27 folder2
drwxr-xr-x 1 ftp ftp           0 May 15 22:03 folder3
-rw-r--r-- 1 ftp ftp           0 May 29 15:34 ftplib.cpp
-rw-r--r-- 1 ftp ftp           36 May 29 22:28 g.txt
-rw-r--r-- 1 ftp ftp      94469 May 31 00:28 img_1.png
-rw-r--r-- 1 ftp ftp      78897 May 31 00:28 img_2.png
-rw-r--r-- 1 ftp ftp      98111 May 31 00:28 img_3.png
-rw-r--r-- 1 ftp ftp      107119 Apr 24 2018 Template_Report.docx
226 Successfully transferred "/"
FTP >>
```

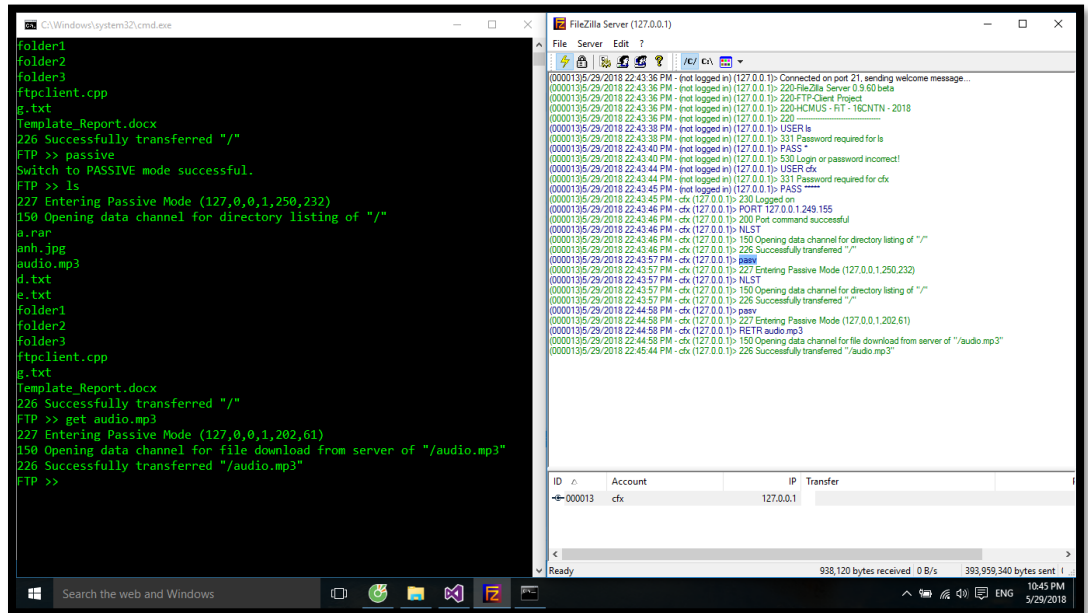
The FileZilla Server window shows the log of the connection. The log includes the following entries:

```
[000011/5/31/2018 9:26:56 AM - cfx (127.0.0.1): PASS
[000011/5/31/2018 9:26:56 AM - cfx (127.0.0.1): 227 Entering Passive Mode (127.0.0.1,211,234)
[000011/5/31/2018 9:26:56 AM - cfx (127.0.0.1): LIST
[000011/5/31/2018 9:26:56 AM - cfx (127.0.0.1): 150 Opening data channel for directory listing of "/"
[000011/5/31/2018 9:26:56 AM - cfx (127.0.0.1): 226 Successfully transferred "/"
```

- get



- Get done



- mget

```

C:\Windows\system32\cmd.exe
Template_Report.docx
226 Successfully transferred "/"
FTP >> passive
Switch to PASSIVE mode successful.
FTP >> ls
227 Entering Passive Mode (127,0,0,1,250,232)
150 Opening data channel for directory listing of "/"
a.rar
anh.jpg
audio.mp3
d.txt
e.txt
folder1
folder2
folder3
ftpclient.cpp
g.txt
Template_Report.docx
226 Successfully transferred "/"
FTP >> get audio.mp3
227 Entering Passive Mode (127,0,0,1,202,61)
150 Opening data channel for file download from server of "/audio.mp3"
226 Successfully transferred "/audio.mp3"
FTP >> mget anh.jpg a.rar e.txt
227 Entering Passive Mode (127,0,0,1,192,158)
150 Opening data channel for file download from server of "/anh.jpg"
226 Successfully transferred "/anh.jpg"
227 Entering Passive Mode (127,0,0,1,200,130)
150 Opening data channel for file download from server of "/a.rar"
226 Successfully transferred "/a.rar"
227 Entering Passive Mode (127,0,0,1,227,111)
150 Opening data channel for file download from server of "/e.txt"
226 Successfully transferred "/e.txt"
FTP >>
  
```

FileZilla Server (127.0.0.1)

File Server Edit ?

0000135/29/2018 22:43:36 PM - not logged in (127.0.0.1) 220 FileZilla Server 0.9.60 beta

0000135/29/2018 22:43:36 PM - not logged in (127.0.0.1) 220 FTP Client Project

0000135/29/2018 22:43:36 PM - not logged in (127.0.0.1) 220 HCMUS - FT - 16CNTN - 2018

0000135/29/2018 22:43:36 PM - not logged in (127.0.0.1) 220

0000135/29/2018 22:43:38 PM - not logged in (127.0.0.1) USER ls

0000135/29/2018 22:43:38 PM - not logged in (127.0.0.1) 331 Password required for ls

0000135/29/2018 22:43:40 PM - not logged in (127.0.0.1) PASS

0000135/29/2018 22:43:40 PM - not logged in (127.0.0.1) 332 Login or password incorrect!

0000135/29/2018 22:43:44 PM - not logged in (127.0.0.1) USER cfx

0000135/29/2018 22:43:44 PM - not logged in (127.0.0.1) 331 Password required for cfx

0000135/29/2018 22:43:45 PM - not logged in (127.0.0.1) PASS ****

0000135/29/2018 22:43:45 PM - cfx (127.0.0.1) 230 Logged on

0000135/29/2018 22:43:46 PM - cfx (127.0.0.1) PORT 127.0.0.1,249,155

0000135/29/2018 22:43:46 PM - cfx (127.0.0.1) 200 Port command successful

0000135/29/2018 22:43:46 PM - cfx (127.0.0.1) NLST

0000135/29/2018 22:43:46 PM - cfx (127.0.0.1) 150 Opening data channel for directory listing of "/"

0000135/29/2018 22:43:46 PM - cfx (127.0.0.1) 226 Successfully transferred "/"

0000135/29/2018 22:43:57 PM - cfx (127.0.0.1) mget

0000135/29/2018 22:43:57 PM - cfx (127.0.0.1) 227 Entering Passive Mode (127.0.0.1,250,232)

0000135/29/2018 22:43:57 PM - cfx (127.0.0.1) NLST

0000135/29/2018 22:43:57 PM - cfx (127.0.0.1) 150 Opening data channel for directory listing of "/"

0000135/29/2018 22:43:57 PM - cfx (127.0.0.1) 226 Successfully transferred "/"

0000135/29/2018 22:44:58 PM - cfx (127.0.0.1) RETR audio.mp3

0000135/29/2018 22:44:58 PM - cfx (127.0.0.1) 150 Opening data channel for file download from server of "/audio.mp3"

0000135/29/2018 22:44:58 PM - cfx (127.0.0.1) 226 Successfully transferred "/audio.mp3"

0000135/29/2018 22:46:17 PM - cfx (127.0.0.1) RETR anh.jpg

0000135/29/2018 22:46:17 PM - cfx (127.0.0.1) 150 Opening data channel for file download from server of "/anh.jpg"

0000135/29/2018 22:46:17 PM - cfx (127.0.0.1) 226 Successfully transferred "/anh.jpg"

0000135/29/2018 22:46:17 PM - cfx (127.0.0.1) RETR a.rar

0000135/29/2018 22:46:17 PM - cfx (127.0.0.1) 150 Opening data channel for file download from server of "/a.rar"

0000135/29/2018 22:46:17 PM - cfx (127.0.0.1) 226 Successfully transferred "/a.rar"

0000135/29/2018 22:46:18 PM - cfx (127.0.0.1) RETR e.txt

0000135/29/2018 22:46:18 PM - cfx (127.0.0.1) 150 Opening data channel for file download from server of "/e.txt"

0000135/29/2018 22:46:18 PM - cfx (127.0.0.1) 227 Entering Passive Mode (127.0.0.1,227,111)

ID Account IP Transfer

000013 cfx 127.0.0.1

Ready 938,176 bytes received 0 B/s 394,897,563 bytes sent

- put & mput

```

C:\Windows\system32\cmd.exe
150 Opening data channel for directory listing of "/"
226 Successfully transferred "/"
a.rar
anh.jpg
audio.mp3
d.txt
e.txt
folder1
folder2
folder3
ftpclient.cpp
g.txt
img_1.png
img_2.png
img_3.png
Template_Report.docx
FTP >> put img_3.png
227 Entering Passive Mode (127,0,0,1,237,45)
150 Opening data channel for file upload to server of "/img_3.png"
226 Successfully transferred "/img_3.png"
FTP >> mput img_1.png img_2.png img_3.png
Upload "img_1.png"?
227 Entering Passive Mode (127,0,0,1,206,220)
150 Opening data channel for file upload to server of "/img_1.png"
226 Successfully transferred "/img_1.png"
Upload "img_2.png"?
227 Entering Passive Mode (127,0,0,1,211,81)
150 Opening data channel for file upload to server of "/img_2.png"
226 Successfully transferred "/img_2.png"
Upload "img_3.png"?
227 Entering Passive Mode (127,0,0,1,237,64)
150 Opening data channel for file upload to server of "/img_3.png"
226 Successfully transferred "/img_3.png"
FTP >>
  
```

FileZilla Server (127.0.0.1)

File Server Edit ?

0000056/31/2018 0:23:38 AM - cfx (127.0.0.1) 227 Entering Passive Mode (127.0.0.1,203,195)

0000056/31/2018 0:24:22 AM - cfx (127.0.0.1) STOR img_1.png

0000056/31/2018 0:24:22 AM - cfx (127.0.0.1) 150 Opening data channel for file upload to server of "/img_1.png"

0000056/31/2018 0:26:45 AM - cfx (127.0.0.1) 226 Successfully transferred "/img_1.png"

0000056/31/2018 0:26:16 AM - cfx (127.0.0.1) disconnected

0000075/31/2018 0:27:02 AM - not logged in (127.0.0.1) Connected on port 21, sending welcome message...

0000075/31/2018 0:27:02 AM - not logged in (127.0.0.1) 220 FileZilla Server 0.9.60 beta

0000075/31/2018 0:27:02 AM - not logged in (127.0.0.1) 220 FTP Client Project

0000075/31/2018 0:27:02 AM - not logged in (127.0.0.1) 220 HCMUS - FT - 16CNTN - 2018

0000075/31/2018 0:27:02 AM - not logged in (127.0.0.1) 220

0000075/31/2018 0:27:04 AM - not logged in (127.0.0.1) USER cfx

0000075/31/2018 0:27:04 AM - not logged in (127.0.0.1) 331 Password required for cfx

0000075/31/2018 0:27:05 AM - not logged in (127.0.0.1) PASS ****

0000075/31/2018 0:27:05 AM - cfx (127.0.0.1) 230 Logged on

0000075/31/2018 0:27:11 AM - cfx (127.0.0.1) PORT

0000075/31/2018 0:27:11 AM - cfx (127.0.0.1) 227 Entering Passive Mode (127.0.0.1,200,130)

0000075/31/2018 0:27:11 AM - cfx (127.0.0.1) NLST

0000075/31/2018 0:27:11 AM - cfx (127.0.0.1) 150 Opening data channel for directory listing of "/"

0000075/31/2018 0:27:11 AM - cfx (127.0.0.1) 226 Successfully transferred "/"

0000075/31/2018 0:27:21 AM - cfx (127.0.0.1) STOR img_3.png

0000075/31/2018 0:27:21 AM - cfx (127.0.0.1) 150 Opening data channel for file upload to server of "/img_3.png"

0000075/31/2018 0:27:21 AM - cfx (127.0.0.1) 226 Successfully transferred "/img_3.png"

0000075/31/2018 0:27:47 AM - cfx (127.0.0.1) STOR img_1.png

0000075/31/2018 0:27:47 AM - cfx (127.0.0.1) 150 Opening data channel for file upload to server of "/img_1.png"

0000075/31/2018 0:27:47 AM - cfx (127.0.0.1) 226 Successfully transferred "/img_1.png"

0000075/31/2018 0:27:50 AM - cfx (127.0.0.1) STOR img_2.png

0000075/31/2018 0:27:50 AM - cfx (127.0.0.1) 150 Opening data channel for file upload to server of "/img_2.png"

0000075/31/2018 0:27:50 AM - cfx (127.0.0.1) 226 Successfully transferred "/img_2.png"

0000075/31/2018 0:27:52 AM - cfx (127.0.0.1) STOR img_3.png

0000075/31/2018 0:27:52 AM - cfx (127.0.0.1) 150 Opening data channel for file upload to server of "/img_3.png"

0000075/31/2018 0:27:52 AM - cfx (127.0.0.1) 226 Successfully transferred "/img_3.png"

ID Account IP Transfer

000007 cfx 127.0.0.1

Ready 1,023,193 bytes received 0 B/s 4,770 bytes sent

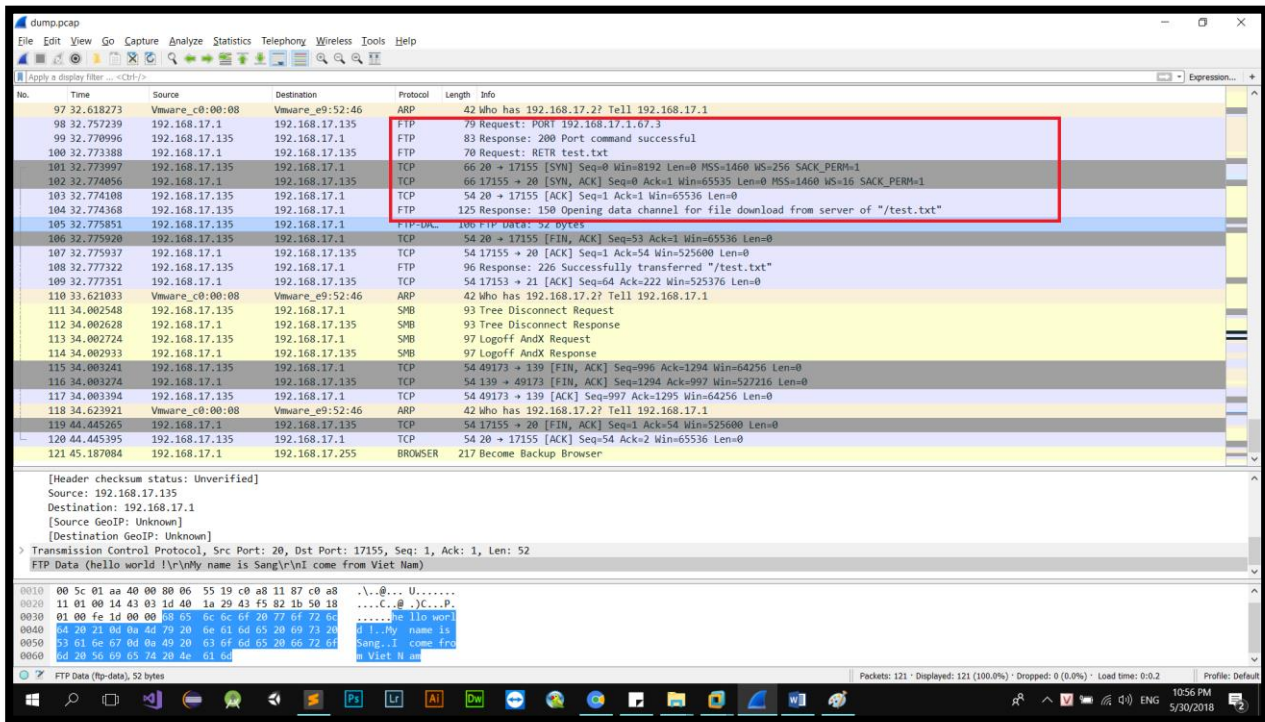
4. ĐÁNH GIÁ

ĐÁNH GIÁ TRÊN CẢ HAI CHẾ ĐỘ ACTIVE VÀ PASSIVE:

Tên hàm	Mức độ hoàn thành
<code>bool connect();</code>	100%
<code>bool login();</code>	100%
<code>void cmd_user();</code>	100%
<code>void cmd_pass();</code>	100%
<code>void cmd_ls();</code>	100%
<code>void cmd_dir();</code>	100%
<code>void cmd_pwd();</code>	100%
<code>void cmd_cd();</code>	100%
<code>void cmd_lcd();</code>	100%
<code>void cmd_get();</code>	100%
<code>void cmd_mget();</code>	100%
<code>void cmd_put();</code>	100%
<code>void cmd_mput();</code>	100%
<code>void cmd_del();</code>	100%
<code>void cmd_mdel();</code>	100%
<code>void cmd_mkdir();</code>	100%
<code>void cmd_rmdir();</code>	100%
<code>void cmd_pasv();</code>	100%
<code>void cmd_pasive();</code>	100%
<code>void cmd_active();</code>	100%
<code>void cmd_quit();</code>	100%
<code>void cmd_help();</code>	100%

5. BẮT GÓI TIN VÀ MÔ TẢ QUÁ TRÌNH GỬI NHẬN

- Thiết lập IP máy thật là 192.168.17.1 đặt client ở máy thật và máy ảo là 192.168.17.135 đặt server.



- Tiến hành yêu cầu **get test.txt** (yêu cầu download file test.txt từ server về client):

Ở chế độ **active** và **passive** thì client đều gửi port lên server, 2 bên thỏa thuận xong server thông báo **port command successful** về thì client tiếp tục gửi lệnh **RETR test.txt** (tệp cần download là file test.txt) lên server chấp nhận và thông báo về cho client mở data channel để download file về và gửi về dung lượng , nội dung gói tin cho client. Hoàn tất quá trình gửi dữ liệu server gửi 1 thông báo về cho client rằng đã gửi thành công.

```
FTP      79 Request: PORT 192.168.17.1.67.3
FTP      83 Response: 200 Port command successful
FTP      70 Request: RETR test.txt
TCP      66 20 → 17155 [SYN] Seq=0 Win=8192 Len=0 MSS=1460 WS=256 SACK_PERM=1
TCP      66 17155 → 20 [SYN, ACK] Seq=0 Ack=1 Win=65535 Len=0 MSS=1460 WS=16 SACK_PERM=1
TCP      54 20 → 17155 [ACK] Seq=1 Ack=1 Win=65536 Len=0
FTP      125 Response: 150 Opening data channel for file download from server of "/test.txt"
```

--Hết--