Zhou1298 YIhanzhou

How to run the program

A.First open server(wich equals to the function of metaserver) on command line with $python3 EchoServer.py (the defauld port number of metaserver is 9000, which should be different of the client port number you enter)

B.Second open 5 client(which equals to the function of server) on command line with $python3 EchoClient.py -p(port number)
for eg.p $python3 EchoClient.py -p5001 (please remember to add -p(port number)) or else the default port number of client will be 9001
According to the instruction
First type 1 to register to the server
After register, you can type 2 to download file and input filename
C. To gracefully close the connection , press 4

Assumptions made:

A. Make 5 folders exiting in the system S1 S2 S3 S4 S5 when the user open the server. And for each server's id, he is mandatory to input S1 to S5. In this way, 5 folders corresponds to 5 server's id_name. When searching a file in a server , you can directly use the server's id_name as a part of the path, as it is the same name with the folder name
B. When download file, if you successfully download a file from the p2p network, it will have a hint that you download it, if you failed to download it, it will have nothing to show on this terminal and show “no file found” in other terminals, when user receive nothing, it means he failed to download the file.
C. The filenames in these folders are all lowercase.
D. For all the file you want to download, you should input the whole name, for eg
If you input pwd1, you will receive no response, means you do not find file
If you input pwd1.txt, you will find this file at S3 and able to downloads it

Program data flow

A. The server sent to metaserver <its host, itsreal port>
B. The metaserver give back server<assigned port, referred port>
C. The client server sent to server server referred port number <its id_name, its port,its assigned name>
D. If the server server reject, send back<one of its connect client server port >

E. One of server in p2p send<”topo”, its id_name > to its metaserver and only “topo” to its client_server (only meta_server need to know who sent him this “topo” and prevent to resent him again, for client_server, it just need to send its client_server)
F. Server who receive “topo” do same things a s e
G. One of server in p2p send<”file”, its idname, its port> to its meta_metaserver and only <”file”, its port> to its client. This port number is used for establish temporal connection when find file in one server
H. Server who receive “topo”, keep its_port unchanged and send <”file”, its idname, original port> to its meta_metaserver and only <”file”, original port> to its client
I. Once file find in one server, establish temporal connection with the original port and sent (its id_name first),after the server on the other side receive this message and send bank ack, the server send file contents to the server on the other side.

Details concerning the project.

A. For Metaserver, it is just a class waiting for connection and give back a referred port and assigned port number and close that connection ,waiting for another.
B. For Server, once it initialize, it has self.sock open and waiting for connection from other server. after it connects to another server, it has a self.meta_cock open to first connect to its global metaserver ip and metaserver will give him back a referred ip.
C. After the server finally connect to its referred server, this connection is open and at the same time, it should able to handle requests from its neighboring neighbors
D. Run-Client is used to handle requests from its server_server after the server establish connection. By using thread, it could handle multiple request from server.
E. Run_server is used to handle requests from its client after a connection establish
F. As python dose not distinguish uppercase and lowercase, I check if there is uppercase in the user input file name and ask user to re input if any uppercase happen.\