Zia Ur Rehman

1802034

Computer NEtwork

# Server

import pickle

import socket

from os import listdir

from os.path import isfile, join

import os

path = 'files/'

onlyfiles = [f for f in listdir(path) if isfile(join(path, f))]

# from commonthread import commonThread

sock = socket.socket(socket.AF\_INET, socket.SOCK\_DGRAM)

client\_req = []

client\_resp = []

udp\_host = "192.168.100.50"

udp\_port = 12345

sock.bind((udp\_host, udp\_port))

def send\_file\_chunk(addr):

    CHUNK\_SIZE = 100

    offset=0

    hex\_string="0x0012"

    client\_resp.clear()

    f = open(join(path,onlyfiles[0]), 'r')

    chunk = f.read(CHUNK\_SIZE)

    while chunk:

        client\_resp.append(hex\_string)

        client\_resp.append(hex(offset))

        client\_resp.append(chunk)

        print("file sending ....")

        sock.sendto(pickle.dumps(client\_resp), (addr[0], addr[1]))

        client\_resp.clear()

        chunk=f.read(CHUNK\_SIZE) #read the next chunk

        offset+=1

    #loop until the chunk is empty (the file is exhausted)

    print("file sending complete....")

    f.close()

def file\_list(addr):

    hex\_string = "0x0010"

    client\_resp = onlyfiles.copy()

    client\_resp.insert(0, hex(len(onlyfiles)))

    client\_resp.insert(0, hex\_string)

    sock.sendto(pickle.dumps(client\_resp), (addr[0], addr[1]))

def send\_file(addr):

    hex\_string="0x0011"

    client\_resp.append(hex\_string)

    client\_resp.append(client\_req[1])

    status=os.stat(join(path,client\_req[1]))

    client\_resp.append(status.st\_size)

    print(client\_resp)

    sock.sendto(pickle.dumps(client\_resp),(addr[0],addr[1]))

    send\_file\_chunk(addr)

while True:

    print("Waiting for client")

    data, addr = sock.recvfrom(1024)

    print("Receied Messages:", pickle.loads(data), "form ", addr)

    sock.sendto(pickle.dumps("Yes"), (addr[0], addr[1])) #ack

    data, addr = sock.recvfrom(1024)#file list req

    if(pickle.loads(data) == "0x0000"):

        file\_list(addr)

    data,addr=sock.recvfrom(1024)#file req

    client\_req=pickle.loads(data)

    print(client\_req)

    if(client\_req[0]=="0x0001" and client\_req[1] in onlyfiles):

        send\_file(addr)

    else:

        message=['0',"File Does not Exits"]

        sock.sendto(pickle.dumps("File Does not Exits"), (addr[0], addr[1]))

# Client

import socket

import pickle

sock = socket.socket(socket.AF\_INET, socket.SOCK\_DGRAM)

#

# socket.gethostname()

udp\_host = "192.168.100.50"

udp\_port = 12345

req\_file=[]

msg="Connection"

sock.sendto(pickle.dumps(msg), (udp\_host, udp\_port))

data,addr=sock.recvfrom(1024)

print("Receied Messages:", pickle.loads(data), "form ",addr)

# Server Responded Yes

hex\_string="0x0000"

sock.sendto(pickle.dumps(hex\_string), (udp\_host, udp\_port))

data,addr=sock.recvfrom(1024)

message=pickle.loads(data)

print(message)

# file req

req\_file.append("0x0001")

req\_file.append(message[2])

print(req\_file)

sock.sendto(pickle.dumps(req\_file),(udp\_host,udp\_port))

# file response

data,addr=sock.recvfrom(1024)

message=pickle.loads(data)

if(message[0]=='0'):

    print(message[1])

    input("Press Enter to Exit")

else:

    end=message[2]/100

    file=""

    while(end>=0):

        data,addr=sock.recvfrom(1024)

        message=pickle.loads(data)

        file=file+message[2]

        end-=1

    print(file)

    input("Press Enter to Exit")

# close the connection

input("Enter")

# Output

