





Client

# -\*- coding: utf-8 -\*-

"""

Created on Wed Feb 1 12:56:12 2023

@author: STUDENT

"""

#hashed code

import socket

import json

import hashlib

client = socket.socket(socket.AF\_INET, socket.SOCK\_STREAM)

client.connect(("192.168.56.163", 1234))

val=1000

message = input("Enter your message: ")

# Generate the hash value using SHA-256

hash\_msg = 'mustafa'

sha256 = hashlib.sha256()

sha256.update(hash\_msg.encode())

hash\_value = sha256.hexdigest()

print(hash\_value)

# save the json data

try:

if val <= 0:

print("your balance is not enough")

else:

datalist = []

val-=int(message)

with open("messages2.json","a") as f:

print("remaining " ,val)

sha256 = hashlib.sha256()

sha256.update(hash\_value.encode())

hash\_value = sha256.hexdigest()

print(hash\_value)

data={"remaining\_value": val, "hash\_value": hash\_value}

datalist.append(data)

json.dump(datalist, f)

except FileNotFoundError:

print("filenotfound")

client.send(message.encode())

#client.send(hash\_value.encode())

response = client.recv(1024)

print(response)

client.close()

server

import json

import socket

import threading

import os

import hashlib

def handle\_client(client\_socket):

request = client\_socket.recv(1024)

request = request.decode()

print(f"[\*] Received: {request}")

try:

with open("messages2.json", "r") as f:

datalist=json.load(f)

for item in datalist:

value = item.get("value")

hash\_value = item.get("hash\_value")

except FileNotFoundError:

hash\_msg="munirabbasi"

sha256 = hashlib.sha256()

sha256.update(hash\_msg.encode())

hash\_value = sha256.hexdigest()

value=300

datalist = []

try:

value+=int(request)

with open("messages.json2", "a") as f:

data = {

"value" : value,

"hash\_value" : hash\_value

}

datalist.append(data)

json.dump(datalist, f)

except FileNotFoundError:

print("filenotopened")

client\_socket.send(b"ACK!")

client\_socket.close()

server = socket.socket(socket.AF\_INET, socket.SOCK\_STREAM)

server.bind(("0.0.0.0", 1234))

server.listen(5)

print("[\*] Listening on 0.0.0.0:1234")

while True:

client, addr = server.accept()

print(f"[\*] Connection from {addr[0]}:{addr[1]}")

client\_handler = threading.Thread(target=handle\_client, args=(client,))

client\_handler.start()