# Name :- Sarthak Pagar

# Roll No. :- 40

# Class :- TE (IT)

# Practical B3 :- Implement a client and a server on different computers using python. Perform the encryption of message of sender between these two entities by using DES Algorithm and use Diffie Hellman method for exchange of keys.

**server.py**

import socket  
from Crypto.Cipher import DES  
from Crypto.Util.Padding import pad, unpad  
  
# Create socket object  
server\_socket = socket.socket(socket.AF\_INET, socket.SOCK\_STREAM)  
  
# Bind to a host and port  
server\_socket.bind(('localhost', 9999)) # You can change the port if needed  
  
# Listen for client connection  
server\_socket.listen(1)  
print("Server listening on port 9999...")  
  
# Accept client connection  
conn, addr = server\_socket.accept()  
print("Connected by", addr)  
  
# First receive client's name  
name = conn.recv(1024).decode()  
print(f"{name} has connected")  
  
# Send server's name to client  
s\_name = "Server"  
conn.send(s\_name.encode())  
  
#-------- Receiving P from Client to Server ----------------  
Pstr = conn.recv(1024)  
Pstr = Pstr.decode()  
print('Prime no P from Client : ', Pstr)  
P = int(Pstr)  
#------------------------------------------------------------------  
  
#-------- Sending G from Server to Client ------------------  
G = input("Enter the Value of Prime Number G : ")  
Gstr = str(G)  
print('Server sending G to Client : ', Gstr)  
conn.send(Gstr.encode())  
#--------------------------------------------------------------------  
  
# Server will choose the private key a  
a = input("Enter Server Private Key : ")  
a = int(a)  
print('\nThe Private Key A for Server is : ', a)  
  
# gets the generated key  
x = int(pow(int(G), a, int(P)))  
  
#-------- Receiving Y from Client to Server ----------------  
Ystr = conn.recv(1024)  
Ystr = Ystr.decode()  
print('\nY received from Client : ', Ystr)  
Y = int(Ystr)  
#------------------------------------------------------------------  
  
#-------- Sending X from Server to Client ------------------  
Xstr = str(x)  
print('\nServer sending X to Client : ', Xstr)  
conn.send(Xstr.encode())  
#--------------------------------------------------------------------  
  
# Secret key for Server  
ka = int(pow(Y, a, P))  
print('\nShared Secret key for the Server is : ', ka)  
print("\n-------------------------------------------------------\n")  
  
print("\nModifying the key to make it 64 bits")  
key = str(ka)  
for i in range(15):  
 key = key + str(ka)  
print("The modified key is : ", key)  
  
# Ensure the key is 8 bytes  
key\_bytes = key[:8].encode()  
  
#-------- Receiving Cipher Text and Decrypting it to Plain Text ----------------  
cipher\_text = conn.recv(1024)  
print('\nCipher Text Received From Client : ', cipher\_text)  
print("\nPerforming Decryption")  
des = DES.new(key\_bytes, DES.MODE\_ECB)  
plain\_text = unpad(des.decrypt(cipher\_text), DES.block\_size).decode()  
print("\nPlain Text : ", plain\_text)  
#--------------------------------------------------------------------  
  
#-------- Encrypting Plain Text and Sending Cipher Text to Client -------------------  
pt = input("\nEnter the Plain Text : ")  
print("\nPerforming Encryption")  
des = DES.new(key\_bytes, DES.MODE\_ECB)  
cipher\_text = des.encrypt(pad(pt.encode(), DES.block\_size))  
print("\nCipher Text : ", cipher\_text)  
conn.send(cipher\_text)  
print("\nSending Cipher text to Client\n")  
#--------------------------------------------------------------------

**client.py**

import socket  
from Crypto.Cipher import DES  
from Crypto.Util.Padding import pad, unpad  
  
# Create a socket object  
s = socket.socket()  
# Define the port on which you want to connect  
port = 9999  
print("Client program running...\n")  
  
# connect to the server on local computer  
s.connect(('127.0.0.1', port))  
name = input(str("\nEnter your name : "))  
print("Connected...to Server...\n")  
s.send(name.encode())  
s\_name = s.recv(1024)  
s\_name = s\_name.decode()  
print(s\_name, "has joined the Client\n")  
  
# Sending P from Client to Server  
P = input("Enter the Value of Prime Number P : ")  
Pstr = str(P)  
print('Client sending P to Server : ', Pstr)  
s.send(Pstr.encode())  
  
# Receiving G from Server to Client  
Gstr = s.recv(1024)  
Gstr = Gstr.decode()  
print('Prime no G from Server', Gstr)  
G = int(Gstr)  
  
# Client will choose the private key b  
b = input("\nEnter Client Private Key : ")  
b = int(b)  
print('\nThe Private Key B for Client is : ', b)  
  
# gets the generated key  
y = int(pow(int(G), b, int(P)))  
  
# Sending Y from Client to Server  
Ystr = str(y)  
print('\nClient sending Y to Server : ', Ystr)  
s.send(Ystr.encode())  
  
# Receiving X from Server to Client  
Xstr = s.recv(1024)  
Xstr = Xstr.decode()  
print('\nX received from Server : ', Xstr)  
X = int(Xstr)  
  
# Secret key for Client  
kb = int(pow(X, b, int(P)))  
print('Shared Secret Key for the Client is : ', kb)  
print("\n-------------------------------------------------------\n")  
  
print("\nModifying the key to make it 64 bits")  
key = str(kb)  
for i in range(15):  
 key = key + str(kb)  
print("The modified key is : ", key)  
  
# Ensure the key is 8 bytes  
key\_bytes = key[:8].encode()  
  
# Encrypting Plain Text and Sending Cipher Text from Client to Server  
pt = input("\nEnter the Plain Text : ")  
print("\nPerforming Encryption")  
des = DES.new(key\_bytes, DES.MODE\_ECB)  
cipher\_text = des.encrypt(pad(pt.encode(), DES.block\_size))  
print("Cipher Text : ", cipher\_text)  
s.send(cipher\_text)  
print("\nSending Cipher text to Server\n")  
  
# Receiving Cipher Text from Server and Decrypting it to Plain Text  
cipher\_text = s.recv(1024)  
print('\nCipher Text Received From Server : ', cipher\_text)  
print("\nPerforming Decryption")  
des = DES.new(key\_bytes, DES.MODE\_ECB)  
text = unpad(des.decrypt(cipher\_text), DES.block\_size).decode()  
print("Plain Text : ", text)

**Output :-**



