St. Francis Institute of Technology

Mount Poinsur, S.V.P. Road, Borivali (W), Mumbai-103

**Class:** BE CMPN A  **Academic year: 2017-2018**

**Experiment - 6**

**Parallel And Distributed Systems** 

**Aim:** Encryption and Decryption in Distributed Systems

**Code:**

**Client:**

import socket

client = socket.socket()

host = socket.gethostname()

port = 5555

client.connect((host, port))

print(client.recv(1024).decode("ASCII"))

key = client.recv(1024).decode("ASCII")

print("Public Key received ="+key)

temp = key.split(",")

e = int(temp[0])

n = int(temp[1])

plainText = input("Enter a message to encode and send\n")

tp = list(plainText)

enc = ""

for i in tp:

   enc = enc + str((ord(i)\*\*e) % n) + " "

enc = enc.rstrip()

print("Encoded message ="+enc)

client.send(enc.encode("ASCII"))

print("Message sent")

client.close()

**Server:**

import socket

import math

server=socket.socket()

port=5555

server.bind((socket.gethostname(),port))

server.listen(5)

while True:

   print("Waiting for connection")

   client,addr=server.accept()

   print("got a request for transmission")

   msg="Wait for the public key from server"

   client.send(msg.encode("ASCII"))

   p=int(input("Enter the first prime number\n"))

   q=int(input("Enter the second prime number\n"))

   n=p\*q

   print("n ="+str(n))

   phi=(p-1)\*(q-1);

   print("phi ="+str(phi))

   while True:

       e = int(input("enter the encoder value\n"))

       temp = math.gcd(phi,e)

       if temp == 1 :

           break

       print("entered value is not coprime with phi")

   print("Public key =("+str(e)+","+str(n)+")")

   msg=str(e)+","+str(n)

   client.send(msg.encode("ASCII"))

   d=0

   for i in range(1,100):

       d=phi\*i+1

       if(d % e == 0):

           d=int(d/e)

           break

   print("Private key =("+str(d)+","+str(n)+")")

   encmsg = str(client.recv(1024).decode("ASCII"))

   print("Received Message =" + encmsg)

   tp = encmsg.split(" ")

   decmsg = ""

   for i in tp:

       decmsg = decmsg + str(chr((int(i)\*\*d)%n))

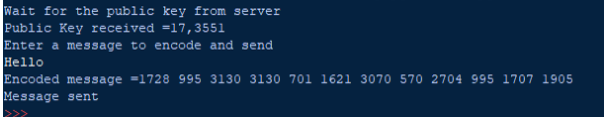
   print("Message is " + decmsg)

   client.close()

server.close();

**Output:**

Client output:



Server output:

