**DEPARTMENT OF INFORMATION TECHNOLOGY**

**COURSE CODE: DJ19ITL501 DATE: 22/10/22**

**COURSE NAME: Cryptography and Network Security Lab Class: A3**

**LAB EXPERIMENT NO. 1**

**AIM:** Design and Implementation of RSA

**DESCRIPTION OF EXPERIMENT:**

The RSA cryptosystem is a public key cryptography algorithm used to encrypt a message without the need to exchange a secret key separately. The RSA algorithm can be used for both public key encryption and digital signatures. Its security is based on the difficulty of factoring large integers. Party A can send an encrypted message to party B without any prior exchange of secret keys. A just uses B's public key to encrypt the message and B decrypts it using his own private key, which only he knows. RSA can also be used to sign a message, so A can sign a message using their private key and B can verify it using A's public key.

**ALGORITHM**:

**Key Generation Algorithm:**

1.Choose prime numbers p and q.[private, chosen] p!=q

1.Recommended size 512 bits (almost 154 decimal digits)

2.Compute n= pq. [public, calculated]1.1024 bits (309 digits)

3.Compute φ(n) = φ(p)φ(q) = (p− 1)(q− 1) [Euler Totient function]

4.Choose an integer e such that 1 < e< φ(n) and gcd(e, φ(n)) = 1 (relatively prime, mutually prime, or co-prime) [public, chosen]

5.Determine d(<φ(n) )as d≡ e−1(mod φ(n)), i.e., d is the multiplicative inverse of e(modulo φ(n)).This is more clearly stated as: solve for,d given d⋅e≡ 1 (mod φ(n)). d is kept as the private key exponent. (Known to receiver only)[private, calculated]

6.Public key PU= {e, n}

7.Private key PR={d,n}

**Encryption**

1.Plaintext M<n

2.Turns M into an integer m, such that 0≤m<n by using an agreed-upon reversible protocol known as a padding scheme.

3.Compute the ciphertext C=m^e mod n

**Decryption**

1. Cipher text C

2. Plaintext M=C^d mod n

**TECHNOLOGY STACK** : Python

**DESIGN AND IMPLEMENTATION CODE:**

1. **RSA:**

import math

import random

'''

\*\*\*key generation\*\*\*

p=int(input("enter 1st prime number : "))

q=int(input("enter 2nd prime number : "))

n=p\*q

phi=(p-1)\*(q-1)

alle=[]

for i in range(2,phi):

    if(math.gcd(i,phi)==1):

        alle.append(i)

#print(alle)

e=random.choice(alle)

#e=3

#print("e = ",e)

ei=pow(e, -1, phi)

#print("einv = ",ei)

d=ei%phi

#print("d = ",d)

message="p"

'''

def encrypt(message,e,n):

    message=message.replace(" ","")

    m=""

    for i in message:

        if(len(str(ord(i)-ord("a")))==1):

            ele="0"+str(ord(i)-ord("a"))

            m=m+ele

        else:

            ele=str(ord(i)-ord("a"))

            m=m+ele

    #print(m)

    c=pow(int(m),e)%n

    #print("cypertext : ",c)

    return c

def decrypt(c,d,n):

    p=pow(int(c),d)%n

    o=ord("a")+p

    #print("plaintext : " ,p)

    return chr(o)

#c=encrypt(message,e,n)

#d=decrypt(c,d,n)

1. **Client:**

import socket

import deffieHellman as dh

q=23

alpha=9

a=4

def Main():

    host='192.168.111.1' #client ip

    port = 4005

    server = ('192.168.111.1', 4000)

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

    s.bind((host,port))

    m=dh.send(q,alpha,a)

    m=str(m[0])

    s.sendto(m.encode('utf-8'), server)

    print("sent text : ",m)

    data= 16

    print("key From connected user: " ,dh.get(data,q,a))

    '''

    while True:

        data, addr = s.recvfrom(1024)

        data = data.decode('utf-8')

        print("cipher text received : ",data)

        data= dh.get(int(data),q,a)

        print("Message from: " + str(addr))

        print("key From connected user: " ,data)

        s.close()

        '''

if \_\_name\_\_=='\_\_main\_\_':

    Main()

1. **Server:**

import socket

import rsa as r

def Main():

    host = '192.168.111.1' #Server ip

    port = 4000

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

    s.bind((host, port))

    print("Server Started")

    while True:

        data, addr = s.recvfrom(1024)

        n=33

        d=3

        data = data.decode('utf-8')

        print("cipher text received : ",data)

        data= r.decrypt(int(data),d,n)

        print("Message from: " + str(addr))

        print("From connected user: " ,data)

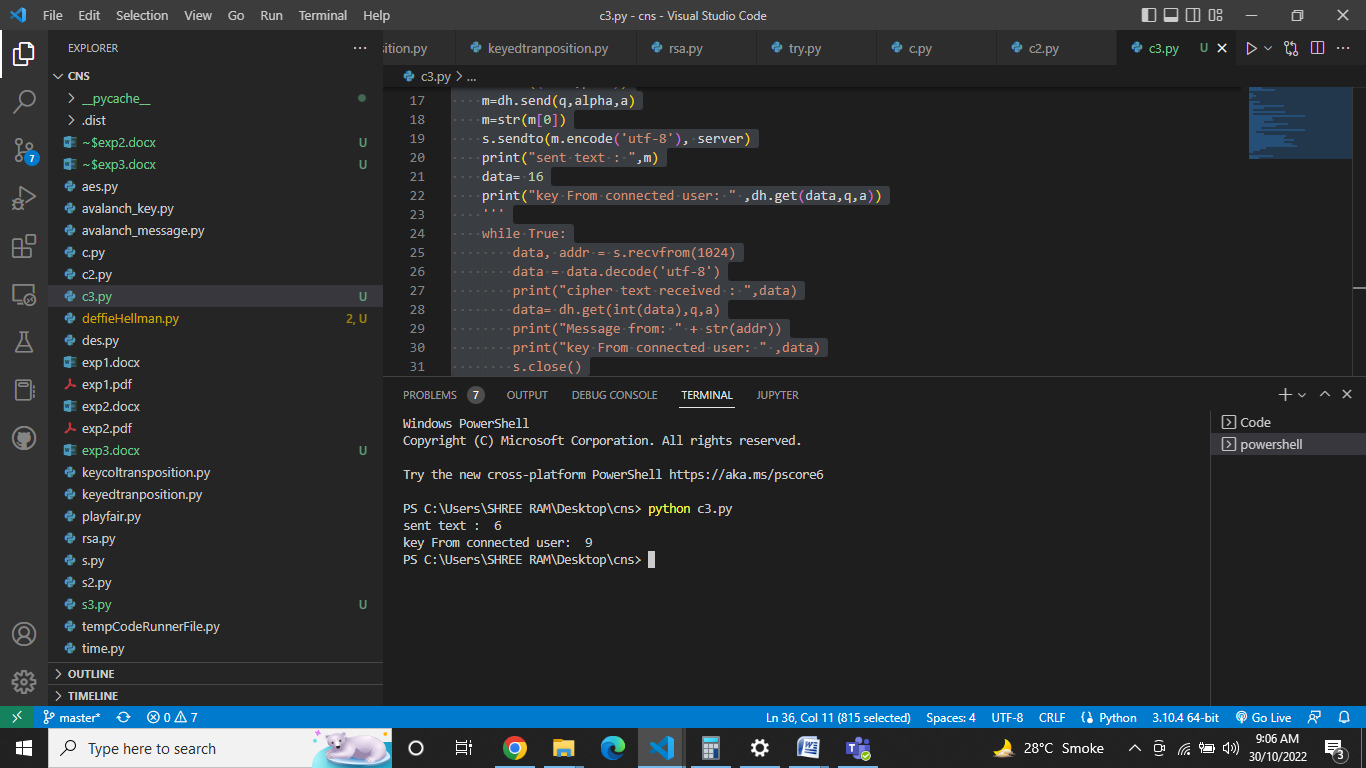
    c.close()

if \_\_name\_\_=='\_\_main\_\_':

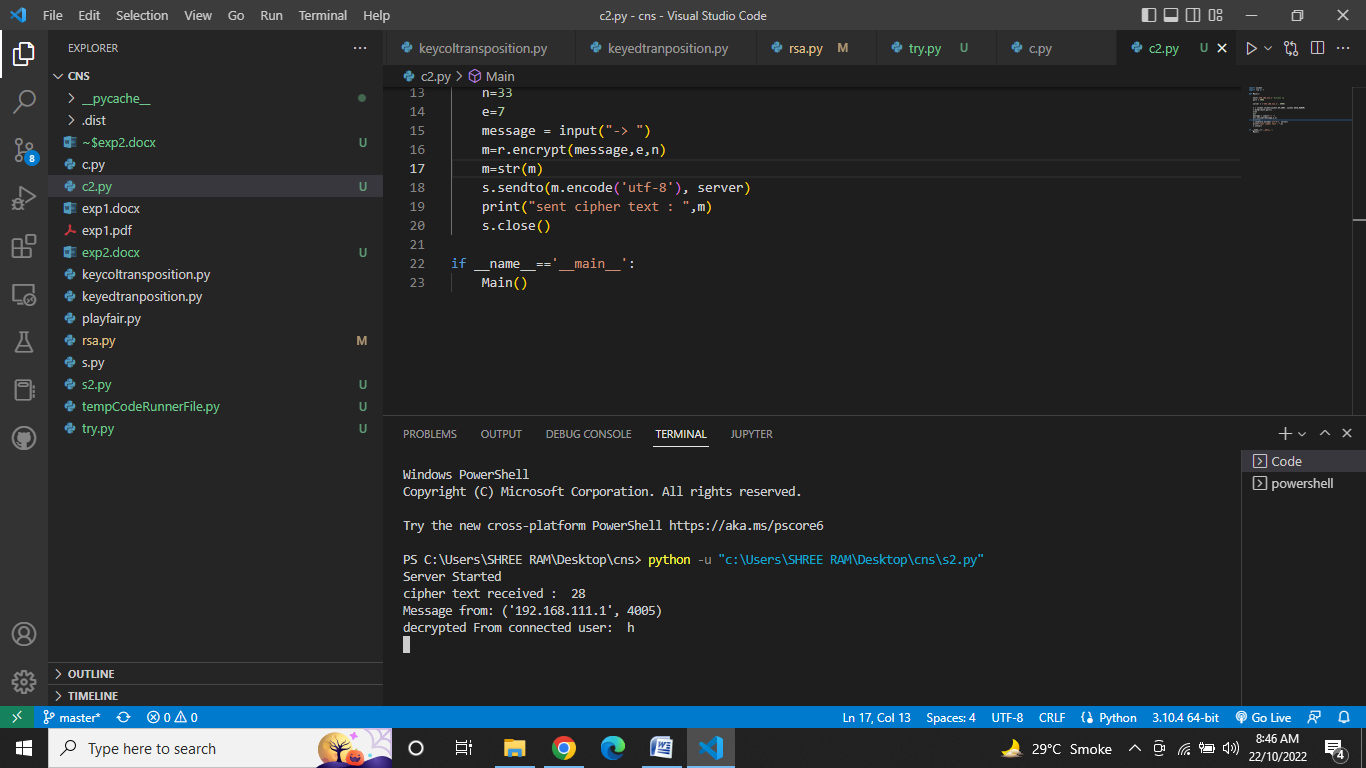
    Main()

**OUTPUT:**

**Client:**

****

**Server:**

****

**CONCLUSION:**

RSA is a cryptosystem, which is known as one of the first practicable public-key cryptosystems and is widely used for secure data transmission. Thus, we have studied and implemented RSA algorithm.

**​**