tcet IT ENGINEERS

TCET
DEPARTMENT OF INFORMATION TECHNOLOGY (IT)
(Accredited by NBA for 3 years, 4th Cycle Accreditation w.e.f. 1st July 2022)
Choice Based Credit Grading System (CBCGS)
Under TCET Autonomy

Name: Aaradhya Gawali
TE_IT_A_29

AIM: To implement RSA algorithm

Lab objectives:

1. Understanding RSA Algorithm: To gain a deep understanding of how the RSA encryption algorithm works, including its key generation, encryption, and decryption processes.

2. Implementing RSA: To implement the RSA algorithm in a programming language of your choice to gain hands-on experience with the encryption and decryption processes.

3. Security Analysis: To assess the security of RSA encryption, including its vulnerability to attacks like brute force, factoring, and chosen ciphertext attacks.

4. Key Management: To learn about key management practices, including the generation, storage, and distribution of public and private keys.

5. Performance Analysis: To evaluate the performance of RSA encryption in terms of computation time and resource usage for various key sizes.

6. Practical Use Cases: To explore and develop practical use cases for RSA encryption in information security, such as secure email communication, digital signatures, and secure file transfer.

7. With Other Cryptographic Algorithms: To compare the RSA algorithm with other cryptographic algorithms, like ECC (Elliptic Curve Cryptography) or AES (Advanced Encryption Standard), in terms of security, performance, and key size.

8. Security Parameter Selection: To determine appropriate key sizes and security parameters for different use cases and scenarios.

9. Educational Purposes: To create a learning resource for others, such as a tutorial or documentation on how to use RSA encryption.

10. Hands-On Cryptography Experience: To provide students or participants with hands-on experience in implementing cryptographic algorithms, enhancing their knowledge of information security.

The specific objectives of your lab will depend on your goals and the level of expertise of the participants. You can tailor the objectives to match your educational or research objectives.

Theory:

1. Key Generation:

Selecting Prime Numbers (p and q): The RSA algorithm begins with the selection of two large prime numbers, typically denoted as p and q. The security of the encryption relies on the difficulty of factoring the product of these two prime numbers.

Calculating the Modulus (n): Compute the modulus (n) as the product of p and q, i.e., n = p * q. This modulus is used in both the public and private keys.

Name: Aaradhya Gawali
TE_IT_A_29

Computing the Totient ($\varphi(n)$): Calculate the totient of n, denoted as $\varphi(n)$, where $\varphi(n) = (p-1)(q-1)$. The totient is used in key generation.

Selecting the Encryption Exponent (e): Choose an encryption exponent (e), which is typically a small prime number and relatively prime to $\varphi(n)$ (i.e., $\gcd(e, \varphi(n)) = 1$). This exponent is part of the public key.

Calculating the Decryption Exponent (d): Calculate the decryption exponent (d) such that $(e * d) \% \varphi(n) = 1$. This ensures that the encryption and decryption operations are inverses of each other. The exponent d is part of the private key.

Public Key and Private Key: ** The public key consists of (n, e), and the private key consists of (n, d). The public key is used for encryption, while the private key is used for decryption.

2. Encryption:

Message Conversion: To encrypt a message (plaintext), convert it into a numerical representation, typically using a padding scheme (e.g., PKCS#1) to ensure uniform message lengths.

Encryption Process: Encrypt the numerical message (M) using the recipient's public key (n, e). The encryption operation is defined as $C \equiv M^e \pmod{n}$, where C is the ciphertext.

3. Decryption:

Decryption Process: The recipient uses their private key (n, d) to decrypt the ciphertext (C). The decryption operation is defined as $M \equiv C^d \pmod{n}$, where M is the original plaintext message.

4. Security:

Security of RSA: The security of RSA relies on the difficulty of factoring the modulus n into its prime factors, p and q. It is crucial to select large prime numbers to make this factoring computationally infeasible.

Key Size: The security of RSA increases with the size of the modulus (n). Larger key sizes are generally considered more secure, but they also require more computational resources.

5. Use Cases

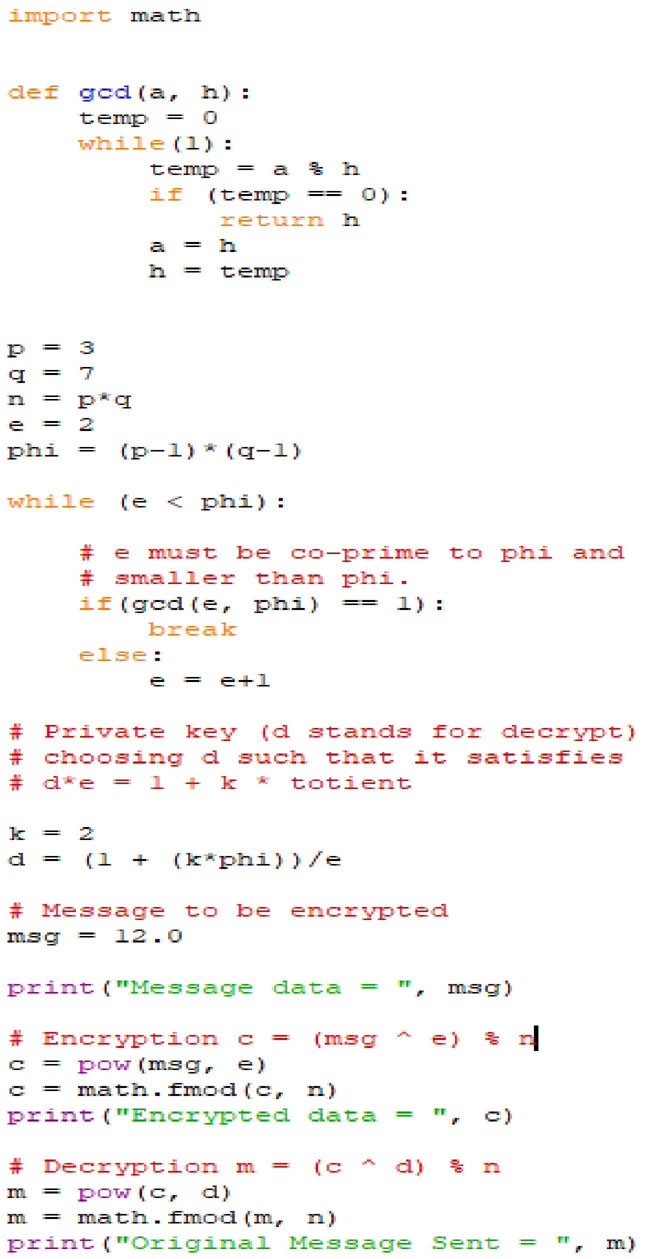
- RSA encryption is widely used for securing communications, including secure email (S/MIME), SSL/TLS for secure web browsing, digital signatures, and many other applications that require data confidentiality, integrity, and authentication.

In summary, the RSA encryption algorithm is a widely used asymmetric cryptographic system that relies on the mathematical properties of large prime numbers for secure data transmission

Name: Aaradhya Gawali
TE_IT_A_29

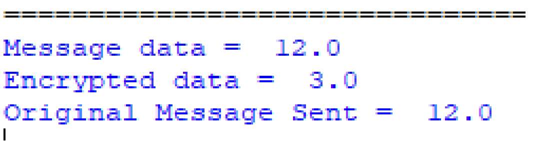
and digital signatures. It is a fundamental building block of modern cryptography and plays a crucial role in securing digital communication and data

SourceCode:



```
import math


def gcd(a, h):
    temp = 0
    while(1):
        temp = a % h
        if (temp == 0):
            return h
        a = h
        h = temp


p = 3
q = 7
n = p*q
e = 2
phi = (p-1)*(q-1)

while (e < phi):

    # e must be co-prime to phi and
    # smaller than phi.
    if(gcd(e, phi) == 1):
        break
    else:
        e = e+1

# Private key (d stands for decrypt)
# choosing d such that it satisfies
# d*e = 1 + k * totient

k = 2
d = (1 + (k*phi))/e

# Message to be encrypted
msg = 12.0

print("Message data = ", msg)

# Encryption c = (msg ^ e) % n
c = pow(msg, e)
c = math.fmod(c, n)
print("Encrypted data = ", c)

# Decryption m = (c ^ d) % n
m = pow(c, d)
m = math.fmod(m, n)
print("Original Message Sent = ", m)
```

Output:



```
==============================
Message data =  12.0
Encrypted data =  3.0
Original Message Sent =  12.0
```

Name: Aaradhya Gawali
TE_IT_A_29

Conclusion: In conclusion, the RSA encryption algorithm is a fundamental component of modern cryptography, widely used for securing digital communication, data transmission, and digital signatures. It is based on the mathematical properties of large prime numbers and offers several key advantages, including:

1. Security: The security of RSA encryption relies on the difficulty of factoring the product of two large prime numbers. As long as the key size is appropriately chosen, RSA encryption can provide strong security.

2. Public-Key Infrastructure: RSA forms the basis of public-key infrastructure (PKI), allowing secure communication and authentication over the internet.

3. Key Management: RSA involves the generation and management of public and private keys, enabling secure data exchange between parties.

4. Versatility: RSA can be used for both encryption and digital signatures, providing data