**Practical Number 5 a.**

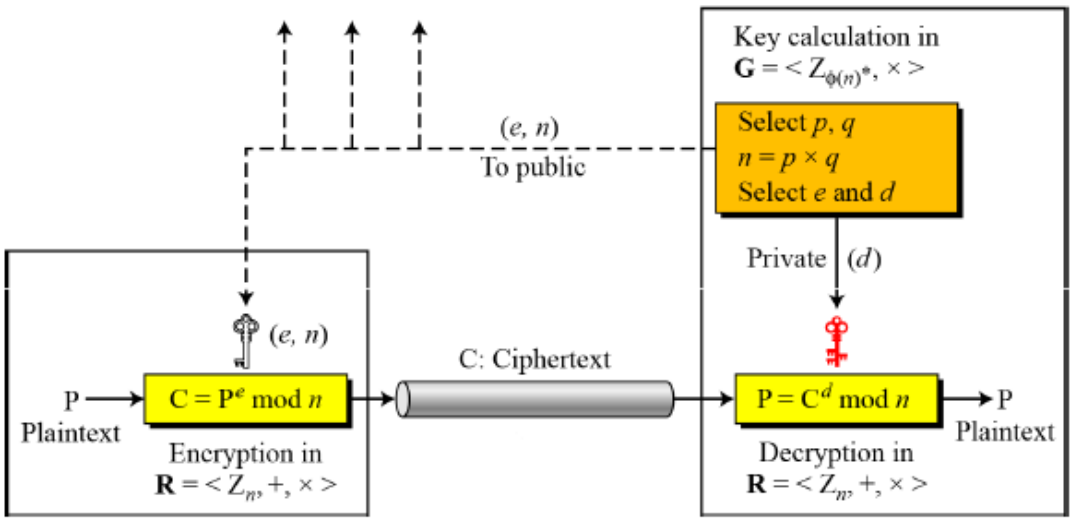
**Aim:** To implement RSA algorithm for providing confidentiality.

**Theory:** RSA cryptosystem is the most common public-key algorithm, named after its inventors Rivest, Shamir, and Adleman. Encryption and Decryption process uses modular exponentiation. Modular exponentiation is feasible in polynomial time using the fast exponentiation algorithm. However, modular algorithm is as hard as factoring the modulus, for which there is no polynomial time algorithm yet. The sender can encrypt in polynomial time, the receiver can decrypt in polynomial time, however, the attacker cannot decrypt in polynomial time as he/she would have to calculate the $e^{th}$ root of ciphertext using modular arithmetic where e is the public key of the receiver. In other words, the sender uses a one-way function (modular exponentiation) with a trapdoor known only to the receiver. The attacker, who does not know the trapdoor, cannot decrypt the message.

RSA uses two algebraic structures: a Ring and a Group.

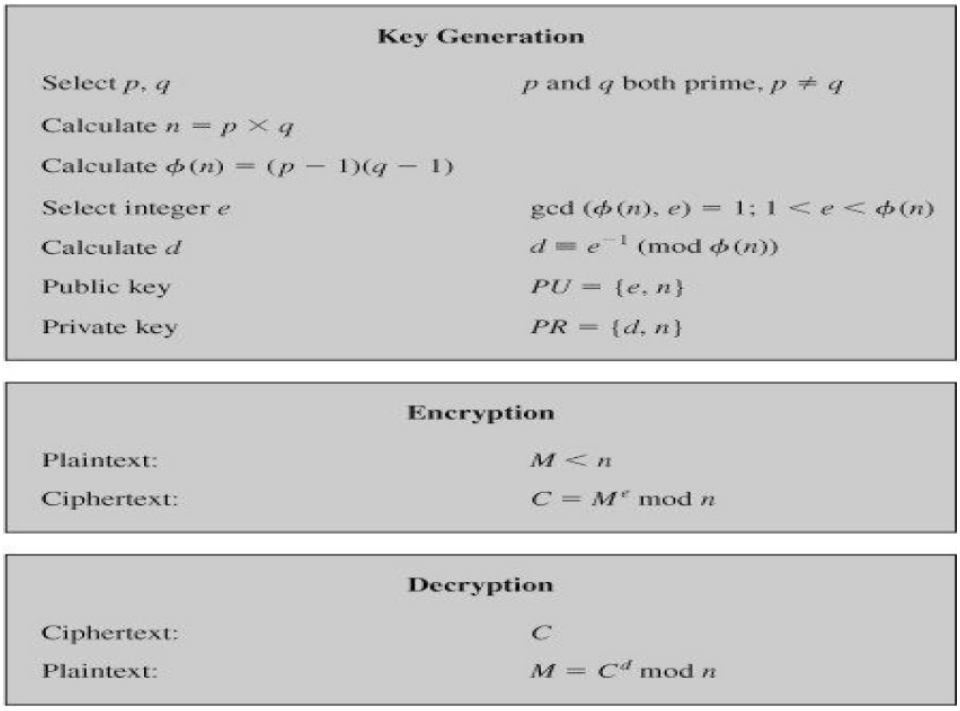
Encryption and Decryption are done using the commutative ring $R=<Z_n, +, x>$ with two arithmetic operations: addition and multiplication. In RSA, the ring is public because the modulus is public. Anyone can send a message to the receiver using this ring to do encryption.

RSA uses a multiplicative group $G=<Z_{\Phi(n)}^*, x>$ for key generation. This group supports only multiplication and division (using multiplicative inverse), which are needed for generating public and private keys. This group is hidden from the public because its modulus, $\Phi(n)$, is hidden from the public.




The receiver uses the key generation algorithm given below to create his public and private key. He announces the tuple (e, n) as his public key, keeps the integer d as his private key and discards p, q, and $\Phi(n)$. To be secure, the recommended size for each prime, p or q, is 512 bits. This makes the size of n, the modulus, 1024 bits. The sender

uses the encryption algorithm to encrypt the message M and the receiver uses the decryption algorithm to decrypt the ciphertext C.



| Key Generation | |
|---|---|
| Select $p, q$ | $p$ and $q$ both prime, $p \neq q$ |
| Calculate $n = p \times q$ | |
| Calculate $\phi(n) = (p-1)(q-1)$ | |
| Select integer $e$ | $\gcd(\phi(n), e) = 1; 1 < e < \phi(n)$ |
| Calculate $d$ | $d \equiv e^{-1} \pmod{\phi(n)}$ |
| Public key | $PU = \{e, n\}$ |
| Private key | $PR = \{d, n\}$ |

| Encryption | |
|---|---|
| Plaintext: | $M < n$ |
| Ciphertext: | $C = M^e \bmod n$ |

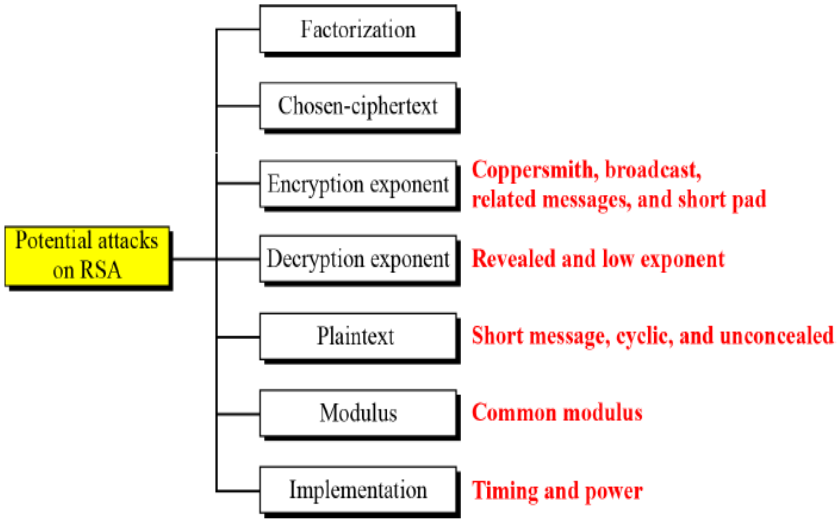
| Decryption | |
|---|---|
| Ciphertext: | $C$ |
| Plaintext: | $M = C^d \bmod n$ |

Encryption and Decryption is carried out using the " Square and Multiply" fast exponentiation algorithm.

**Square_and_Multiply(a, x, n)** // calculates $a^x$ mod n

```
{                                       // x is in binary form
    y =1
    for (i=0 to nb – 1)                 // nb is the no. of bits in x
    {
        if (xi = 1) y = a * y mod n     // multiply only if bit is 1
        a = a^2 mod n
    }
    return y
}
```

**Attacks on RSA:** Based on the weak plaintext, weak parameter selection, or inappropriate implementation, following potential attacks on RSA are possible.




**Recommendations:** To avoid these potential attacks following recommendations must be followed. These recommendations are based on theoretical and experimental results.

1. The number of bits in n should be at least 1024.
2. Two primes p and q must be 512 bit at least.
3. Values of p and q should not be close to each other.
4. Both p-1 and q-1 should have at least one large prime factor.
5. The ratio of p/q should not be close to a rational number with a small numerator or denominator.
6. Modulus n must not be shared.
7. If d is leaked, immediately change n, e and d.
8. The value of e should be $2^{16+1}$ or an integer close to it.
9. Message must be padded by Optimal Asymmetric Encryption Padding.

**Example:**

1. Let : $p$ =17 & $q$=11
2. Compute $n = p.q$ =17×11=187
3. Compute ø($n$)=($p$–1)($q$-1)=16×10=160
4. Select e *:* gcd(e,160)=1; e is member of $Z_{ø(n)*}$

   Let $e$ = 7
5. Determine d*:* $d=e^{-1}$ mod 160 and $d$ < 160

   d=23
6. Publish public key KU={7,187}
7. Keep secret private key KR={23,17,11}

8. Given message M = 88 (88<187)

   Encryption: $C = 88^7 \bmod 187$

   $= 11$

   Decryption: $M = 11^{23} \bmod 187$

   $= 88$

**Conclusion:** The concepts of RSA cryptosystem for providing confidentiality has been understood and successfully implemented.