(IJUSE) 2024, Vol. No. 10, Jan-Dec e-**ISSN: 2454-759X, p-ISSN: 2454-7581**

A Study Of The Importance Of Prime Numbers In Cryptographic Algorithms

Mohan Satvik Adusumalli
Disha Delphi Public School, Kota

[1]*Date of Receiving: 12 November 2023; Date of Acceptance: 02 January 2024; Date of Publication: 16 January 2024*

ABSTRACT

Cryptography is a concept of protecting information and conversations which are transmitted through a public source, so that the send and receive only read and process it. There are several encryption and decryption algorithm which involves mathematical concepts to provide more security to the text which has to be shared through a medium. This paper aims to explain the Rivest, Shamir and Adleman algorithm invented in 1977 used for asymmetric cryptography. In an asymmetric key encryption scheme, anyone can encrypt messages using the public key, but only the holder of the private key can decrypt the message. Security depends upon the secrecy of the private key. Importance of prime numbers in RSA cryptographic system is also explained in this paper.

In this paper, we will research the fundamental rules that cryptography is based on to better understand the mathematics standing behind it. Then we will describe a foundational algorithm in cryptography: the Diffie-Hellman key exchange protocol. This paper explores the significance of prime numbers in cryptographic algorithms. It examines how prime numbers are used to enhance security in various encryption schemes, focusing on their mathematical properties that make them ideal for such applications. The paper also discusses contemporary cryptographic protocols that leverage primes and addresses potential vulnerabilities associated with their use.

INTRODUCTION

Cryptography is essential for securing digital information and communication in the modern world. At the core of many cryptographic algorithms lie prime numbers, which provide a foundation for generating secure keys and encrypting data. This paper provides an overview of the importance of prime numbers in cryptographic algorithms, highlighting their properties and applications.

Throughout history, there has been a need for secure, or private, communication. In war, it would be devastating for the opposing side to intercept information during communication over an insecure line about the plan of attack. With all of the information that is stored on the internet today, it is important that sensitive information, such as a person's credit card number, is stored securely. Cryptosystems, or ciphers, use a key, or keys, for encrypting and decrypting information being communicated with the intention of keeping this information out of the hands of unwanted recipients. Prior to the 1970s, the key(s) used in cryptosystems had to be agreed upon and kept private between the originator, or sender, of a message and the intended recipient. This made it difficult to achieve secure communication for two parties far from one another because they would have to find another means of secure communication to agree upon the key(s) of the cryptosystem begin used. Such ciphers, where the encryption/decryption key(s) must be kept private between the originator and intended recipient, are called symmetric-key ciphers.

As technology increased, the security of symmetric-key ciphers became more vulnerable. It was not until the 1970s that a more secure cryptosystem, the RSA cryptosystem, was made public by Rivest, Shamir and Adleman, three MIT researchers. The RSA cipher, when implemented appropriately, is still considered to be an unbreakable cipher. It was the first specific type of asymmetric-key cipher, or public-key cipher, meaning that two parties could publicly agree upon encryption keys over an insecure communication line and not compromise the security of the cipher. The security of the RSA cipher comes from the general difficulties of factoring integers that are the product of two large prime

[1] *How to cite the article:* Adusumalli M.S..; (January 2024); A Study Of The Importance Of Prime Numbers In Cryptographic Algorithms; *International Journal of Universal Science and Engineering;* Vol 10, 1-10

numbers. The level of security for the RSA cipher increases as the size of the prime numbers used for determining the encryption key increases.

Modern implementations of the RSA cipher require that the prime numbers for determining the encryption key be very large, hundreds of digits in length. This paper discusses some of the algorithms, called primality tests, that are used to determine whether or not a given positive integer is prime, very likely prime, or composite. These tests are very helpful for determining the primarily of integers that are hundreds of digits long. There is no known factoring method for determining the prime factors in a feasible time frame for an integer that is the product of primes equal in size to those desired in the RSA cipher. However, to truly understand how the RSA cipher gets its security and the role prime numbers play in this security, one must understand the processes for factoring integers with prime factors.

PRIME NUMBER

Prime numbers are fundamental to the field of cryptography due to their unique mathematical properties, which provide a foundation for creating secure cryptographic algorithms. Here are several detailed reasons why prime numbers are extensively used in cryptography:

Integer Factorization Difficulty:

- Cryptographic systems often rely on the difficulty of factoring large composite numbers into their prime factors.
- Prime factorization is known to be a computationally complex problem, especially for large numbers with two large prime factors.

Public Key Cryptography:

- Public key cryptography, exemplified by the RSA algorithm, utilizes the mathematical relationship between two large prime numbers for secure communication.
- The public key is generated from the product of two large prime numbers, and the security of the system relies on the computational challenge of factoring the resulting composite number.

Discrete Logarithm Problem:

- The Diffie-Hellman key exchange and other cryptographic techniques depend on the difficulty of solving the discrete logarithm problem in a finite field.
- Choosing a large prime modulus enhances security by making it computationally infeasible to efficiently calculate discrete logarithms.

Primality Testing:

- Cryptographic protocols often require the generation of large prime numbers.
- Efficient primality testing algorithms, like the Miller-Rabin algorithm, are employed to ensure the generated numbers are prime with high probability.

Random Number Generation:

- Cryptographic applications demand the generation of random numbers for various purposes, such as key generation and initialization vectors.
- Large prime numbers can serve as the basis for generating pseudorandom numbers, contributing to the security of cryptographic systems.

(IJUSE) 2024, Vol. No. 10, Jan-Dec e-**ISSN: 2454-759X, p-ISSN: 2454-7581**

Security through Complexity:

- The use of prime numbers in cryptography is grounded in the complexity of mathematical problems associated with primes.

- Leveraging the complexity of operations involving prime numbers provides a layer of security, making it challenging for adversaries to compromise the cryptographic algorithms.

Prime numbers are integral to the security of cryptographic systems, offering a mathematical foundation for creating algorithms that resist attacks and ensure the confidentiality and integrity of sensitive information in modern communication and information security protocols.

PRIME NUMBERS AND CRYPTOGRAPHY

There is a remarkable disparity between the degree of difficulty of the task of multiplication and that of factorization. Multiplying integers together is a reasonable exercise for a young child if the integers are small, and it remains a very straightforward task even when the integers are very large. The reverse operation, however, that of resolving a given integer into factors, is cumbersome except for the very smallest integers and becomes near to impossible for large numbers. This asymmetry is exploited in a new kind of cryptosystem, called RSA after its discoverers, Rivest, Shamir and Adleman. In the RSA system secrecy is provided by placing a would-be codebreaker in a situation where in principle he commands all information necessary for reading the protected message but is confronted with an arithmetic task which in practice is prohibitively time-consuming.

Since these systems seem to achieve secrecy without keeping any key secret they are often referred to by the term Public Key Cryptosystems or Open Key Cryptosystems. We prefer the designation Open Encryption-Key Systems, since there also is a decryption key which, of course, must be kept secret.

Keys in Cryptography- Encryption may be seen as an operation on a segment *(coding unit)* of plaintext *T* yielding a corresponding segment of ciphertext *C* according to an encryption function f:

$$C = f(T). \tag{1}$$

Decryption is the inverse operation, performed by the function f^{-1}:

$$T = f^{-1}(C). \tag{2}$$

In most conventional cryptosystems, the mathematical relation between f and f^{-1} is trivial. For decryption, the encryption tables need only be read the other way round. If encryption is performed as word-by-word substitution of codes for

plaintext words, then the writer will require, say, an English-to-Crypto code-book while the reader will find it convenient to have a Crypto-to-English dictionary, if the crypto vocabulary is at all large. Either of these books can be obtained from the other by a mere sorting process.

The encryption function f is traditionally defined by an algorithm and one or more parameters for that algorithm. The parameters *(key of the day)* are changed more often than the algorithm, and different sets of parameters may be chosen for different correspondents sharing the same algorithm (and communications network and crypto machinery). It is customary for cryptographers to assume that the algorithm is known to illegitimate readers, *the enemy*, but that the parameters are not, and the set of parameters is therefore called the *key*. Similarly, f^{-1} is described by a different algorithm with its own parameters. Since these two sets of parameters, if at all different, can easily be derived, one set from the other, the term key is often used loosely for the set of parameters as well for encryption as for decryption.

(IJUSE) 2024, Vol. No. 10, Jan-Dec e-**ISSN: 2454-759X, p-ISSN: 2454-7581**

Now, the innovation introduced by RSA lies in the design of a system where f^{-1} may remain unknown to someone who possesses f. The existence of such a crypto opens up fascinating perspectives. Since f can be openly announced, crypto traffic is not hampered by the traditional difficulties of safely conveying the key from one place to another. The key for each addressee could in fact be published in a telephone directory or—to take a more practical example—in conjunction with the list of mailboxes in a computerized message system. The cryptosystem becomes less vulnerable, since every correspondent will have sole responsibility for the decryption key applicable to messages directed to him. Unauthorized readers will gain nothing if they come into the possession of what is otherwise lethal for most cryptosystems, a pair of plain and crypto texts. In fact, a sender of encrypted text will not be able to decode what he himself has written!

In this context we can no longer consider the distinction between the encryption and decryption keys as one involving merely a practical reformulation. The *practical* inconvenience for the unauthorized reader is precisely the heart of the matter. This is the reason why we shall prefer the term Open *Encryption*-Key Systems.

PUBLIC KEY CRYPTOGRAPHY

Public-key cryptography is a radical departure from all that has gone before. Right up to modern times all cryptographic systems have been based on the elementary tools of substitution and permutation. However, public-key algorithms are based on mathematical functions and are asymmetric in nature, involving the use of two keys, as opposed to conventional single key encryption. Several misconceptions are held about p-k:

1. That p-k encryption is more secure from cryptanalysis than conventional encryption. In fact the security of any system depends on key length and the computational work involved in breaking the cipher.

2. That p-k encryption has superseded single key encryption. This is unlikely due to the increased processing power required.

3. That key management is trivial with public key cryptography, this is not correct.

RSA Algorithm -

The RSA algorithm was developed by Ron **R**ivest, Adi **S**hamir and Len **A**dleman at MIT in 1978. Since this time it has reigned supreme as the most widely accepted and implemented general-purpose approach to public-key encryption.

The RSA scheme is a ***block cipher*** in which the plaintext and ciphertext are integers between 0 and ***n*** — 1 for some n. The scheme makes use of an expression with exponentials. Plaintext is encrypted in blocks having a binary value less than some number n. For some plaintext block M and ciphertext block C we have:

$$\begin{aligned} C &\equiv M^e \pmod{n} \\ M &\equiv C^d \pmod{n} \quad (M^e)^d \pmod{n} \\ M &\equiv M^{ed} \pmod{n} \end{aligned} \tag{3}$$

Both sender and receiver know n. The sender knows the value of e and only the receiver knows the value of d. To restate:

$$\begin{aligned} KU &= \{e, n\} \\ KR &= \{d, n\} \end{aligned} \tag{4}$$

For this algorithm to be satisfactory for public-key encryption, the following requirements must be met:

1. It is possible to find values of e, d and n such that $M^{ed} = M \pmod{n}$ for all

$M < n$

2. It is relatively easy to calculate M^e and C^d for all values of $M < n$.

3. It is infeasible to determine d given e and n.

Focusing initially on the first question we need to find a relationship of the form: $M^{ed} = M \pmod{n}$ If we recall that Euler's theorem states that

$$a^{\phi(m)} \equiv 1 \pmod{m} \qquad \gcd(a, m) = 1 \tag{5}$$

There is a corollary to this theorem that can be used to produce the required relationship. Given two prime numbers ***p*** and q and integers $n = pq$ and m, with $0 < m < n$,

the following relationship holds:

$$m^{\phi(n)+1} \equiv m^{(p-1)(q-1)+1} \equiv m \pmod{n} \tag{6}$$

If $\gcd(m, n) = 1$ then this holds by virtue of Euler's theorem. Suppose however that $\gcd(m, n) \neq 1$. What does this mean? Well, because $n = pq$, the equality $\gcd(m, n) = 1$ is equivalent to the logical expression (m is not a multiple of p) **AND** (m is not a multiple of q). If m is a multiple of p then n and m share the prime factor p and are not relatively prime (the same can be said for q). Therefore, the expression $\gcd(m, n) \neq 1$ must be equivalent to the negation of the foregoing logical expression. Therefore, $\gcd(m, n) \neq 1$ is equivalent to the logical expression (m is a multiple of p) **OR** (m is a multiple of q).

Looking at the case in which m is a multiple of p, so that the relationship $m = cp$ holds for some positive integer ***c.*** In this case we must have $\gcd(m, q) = 1$. Otherwise, we have m a multiple of p and m a multiple of q and yet $m < pq$. If $\gcd(m, q) = 1$ then Euler's theorem holds and

$$m^{\phi(q)} \equiv 1 \pmod{q}$$

But then, by the rules of modular arithmetic,

$$[m^{\phi(q)}]^{\phi(p)} \equiv 1 \pmod{q}$$
$$m^{\phi(n)} \equiv 1 \pmod{q}$$

Therefore, there is some integer k such that

$$m^{\phi(n)} \quad 1 + kq$$

Multiplying each side by $m \quad cp$,

$$m^{\phi(n)+1} = m + kcpq = m + kcn$$
$$m^{\phi(n)+1} \equiv m \pmod{n}$$

A similar line of reasoning is used for the case in which m is a multiple of q. Thus, equation 6 is proven. An alternative form of this corollary is directly relevant to RSA:

$$\begin{aligned} m^{k\phi(n)+1} &\equiv [(m^{\phi(n)})^k \times m)] \pmod{n} \\ &\equiv [(1)^k \times m)] \pmod{n} \text{ by Euler's theorem} \\ &\equiv m \pmod{n} \end{aligned} \quad (7)$$

We can now state the RSA scheme. The ingredients are the following:

p, q, two primes	(private, chosen)
$n = pq$	(public, calculated)
e, with $\gcd(\phi(n), e) \quad 1; 1 < e < \phi(n)$	(public, chosen)
$d \equiv e^{-1} \pmod{\phi(n)}$	(private, calculated)

The private key consists of $\{d, n\}$and publickey is$\{e, n\}$. Suppose that user A has published his public key and that user B wishes to send the message M to A.***B*** calculates $C = M^e \pmod{n}$and transmits C. On receipt of the ciphertextC user A decrypts by calculating the following: $M = C^d \pmod{n}$. Figure 1 summarises the algorithm.

Example:

1. Select p=7 ,q=17
2. Calculate $n = pq = 7 \times 17 = 11$
3. Calculate $\phi(n) = (p-1)(q-1) = 96$.
4. Select e, relatively prime to and less than $\phi(n)$, say $e \quad 5$.
5. Determine d such that$de = 1 \pmod{96}$and$d < 96$.

6. The correct value for d is 77 because 77 x 5 = 385 = 4 x 96 + 1 (can be calculated using the extended version of Euclid's algorithm).

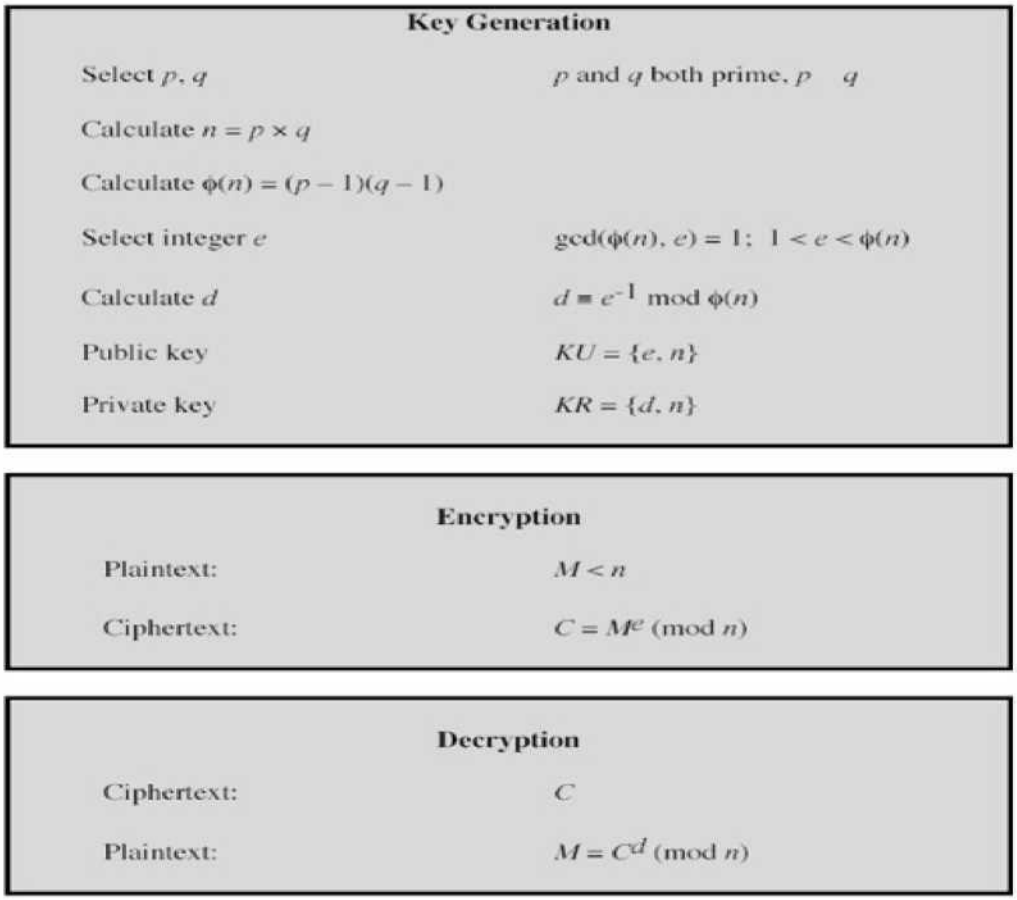

Fig 1: The RSA algorithm

The resulting public key is KU= {5,119} and private key is KR{77,119}. Say the plaintext is M = 19. For encryption 194 is raised to the 5th power, yielding 2, 476, 099. Upon division by 119, the remainder is 66, hence ciphertext sent is 66. For decryption it is determined using KR that 6677 = 19 (mod 119) so the recovered plaintext is 19.

DIFFIE HELLMAN KEY EXCHANGE

The first published P-K algorithm appeared in the paper by Diffie and Hellman that defined public key cryptography however it is limited to the secure exchange of a secret key and not of a message. The security of the scheme depends on the difficulty of computing discrete logarithms which were discussed earlier in the course. The Diffie-Hellman key exchange consists of two publicly known numbers: a prime number p and an integer α that is a primitive root of q. Suppose the users A and *B* wish to exchange a key. User *A* selects a random integer $X_A < q$ and computes $Y_A = \alpha^{X_A}$ mod *q*. Similarly user *B* independently selects a random integer $X_B < q$ and computes $Y_B = \alpha^{X_B}$ mod *q*. Each side keeps the *X* values private and makes the *Y* value available publicly to the other side. User *A* computes the key as $K = (Y_B)^{X_A}$ mod q and user *B* computes the key as $K = (Y_A)^{X_B}$ mod q. These two calculations produce identical results and the result is that the two sides have exchanged a secret key. This can be seen because:

$$\begin{aligned} K &= (Y_B)^{X_A} \bmod q \\ &= (\alpha^{X_B} \bmod q)^{X_A} \bmod q \\ &= (\alpha^{X_B})^{X_A} \bmod q \\ &= (\alpha^{X_A})^{X_B} \bmod q \\ &= (\alpha^{X_A} \bmod q)^{X_B} \bmod q \\ &= (Y_A)^{X_B} \bmod q \end{aligned}$$

Furthermore because X_A and X_B are private, an opponent is forced to take a discrete logarithm to determine the key. For example, attacking the secret key of user *B* the opponent must compute:

$$X_B = ind_{\alpha,q}(Y_B)$$

where $ind_{\alpha,q}(Y_B)$ is the discrete logarithm, or index, of Y_B for the base α mod *q*. The

scheme can be summarised as shown in figure 2

For example let's say we have the values $q = 353$ and a primitiveroot $\alpha = 3$.We can see that $\alpha = 3$ is a primitive root of $q = 353$ due to the following reasoning.

If α is a primitive root of a prime *q* then the set of numbers $\{\alpha, \alpha^2, \ldots, \alpha^{\phi(q)}\}$ are distinct modulo *q* and hence form the set $\{1, 2, \ldots (q-1)\}$ in some order. In this case $\alpha = 3$ and it can be seen to be a primitive root of $q = 353$ as $\{3 \bmod 353, 3^2 \bmod 353, \ldots, 3^{353} \bmod 353, \}$ which contains all the elements of $\{1, 2, \ldots 352\}$

Suppose A and B select the private keys $X_A = 97$ and $X_B = 233$ respectively.To calculate the secret key *K* user *A* calculates:

$$\begin{aligned} Y_A &= \alpha^{X_A} \bmod q \\ &= 3^{97} \bmod 353 \\ &= 40 \end{aligned}$$

Similarly user B calculates

$$\begin{aligned} Y_B &= \alpha^{X_B} \bmod q \\ &= 3^{233} \bmod 353 \\ &= 248 \end{aligned}$$

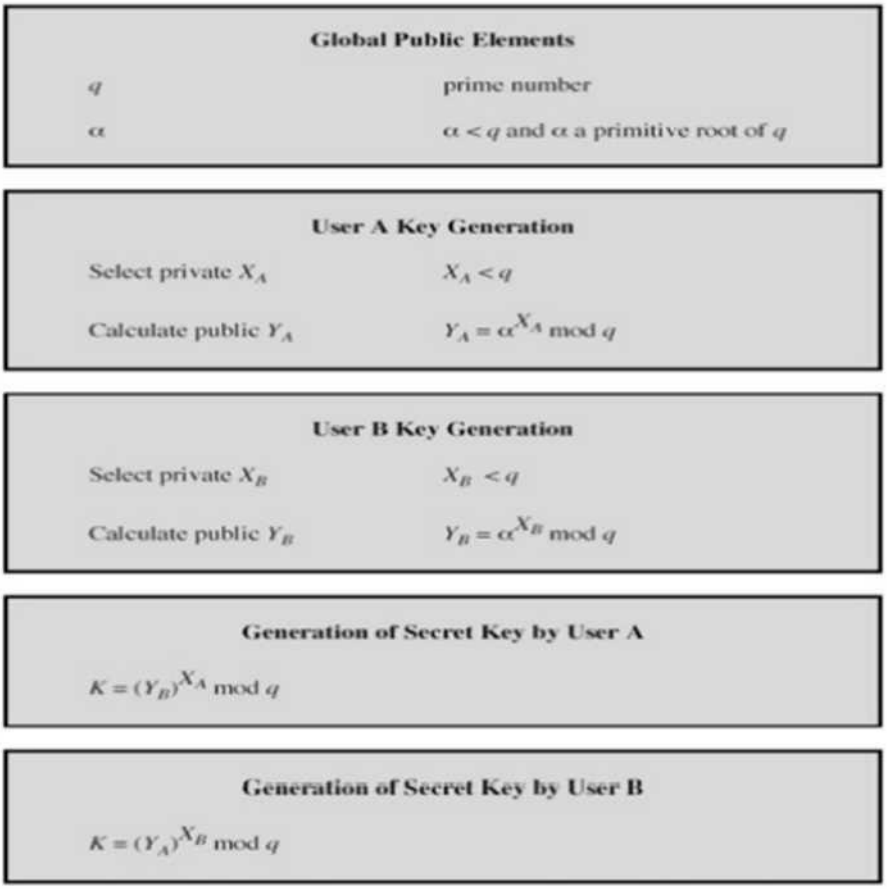

Fig. 2: The Diffie Heilman Key Exchange Algorithm.

Then we have $K = 248^{97} \bmod 353 = 40^{233} \bmod 353 = 160$.

We assume the attacker would have q, α, Y_A, Y_B which for this example might be enough using a brute force approach. However with large numbers this becomes impractical.

CONCLUSION

In conclusion, prime numbers play a pivotal role in modern cryptographic algorithms, serving as the backbone for various encryption and key exchange mechanisms. Their unique mathematical properties, such as their indivisibility and distribution characteristics, provide essential security features that underpin the robustness of many cryptographic systems. Prime numbers are integral to public-key cryptographic schemes like RSA, where their use in key generation and encryption/decryption processes is crucial. The security of these algorithms relies heavily on the difficulty of factoring large composite numbers or solving discrete logarithm problems, tasks that become increasingly challenging with larger primes.

In summary, prime numbers remain a cornerstone of cryptographic algorithms, providing the mathematical foundation for secure communication and data protection. Their continued importance underscores the need for rigorous research and innovation to ensure that cryptographic systems remain robust in the face of evolving threats and technological advancements.

REFERENCES

1. Boneh, D., & Brakerski, Z. (2023). "Advances in Cryptographic Algorithms Utilizing Large Primes." *Proceedings of the International Conference on Cryptography and Network Security (CANS)*.
2. Pomerance, C., & R. S. (2023). "A Study on the Computational Complexity of Prime Number Generation and Testing." *Journal of Cryptographic Engineering*, 13(2), 101-120.
3. Stern, J. (2023). "The Role of Prime Numbers in Secure Cryptographic Protocols." *IEEE Transactions on Information Forensics and Security*, 18(1), 45-60.
4. Boneh, D., & Shoup, V. (2022). "A Graduate Course in Applied Cryptography." *Cryptography and Security: An Introduction*. Stanford University.
5. The Science of Encryption: Prime Numbers and Mod N Arithmetic (n.d.): n. pag. Math.berkeley.edu. University of California, Berkeley. Web. 17 Oct. 2022.
6. F. Bahr, M. Boehm, J. Franke, T. Kleinjung. “rsa200”. May 9 202021.
7. Jeffrey Hoffstein, Jill Pipher, Joseph H. Silverman. An Introduction to Mathematical Cryptography. Undergraduate Texts in Mathematics, 2020.
8. Katz, J., & Lindell, Y. (2020). *Introduction to Modern Cryptography: Principles and Protocols* (3rd ed.). CRC Press.
9. Stinson, D. R. (2020). *Cryptography: Theory and Practice* (4th ed.). CRC Press.
10. Menon, Vijay. Deterministic Primality Testing - Understanding The AKS Algorithm. (2013): arXiv. Web. 1 Sept. 2016.