World Journal of Advanced Research and Reviews

eISSN: 2581-9615 CODEN (USA): WJARAI

Cross Ref DOI: 10.30574/wjarr

Journal homepage: https://wjarr.com/

(Research Article)

Check for updates

# Exploring the properties of prime numbers in cryptography

Mamatha N [1, *] and Sunitha S.S [2]

*[1] Lecturer in Science Department, Karnataka (Govt) Polytechnic Mangalore, Karnataka, India.*

*[2] Lecturer in Science Department, Government Polytechnic Holenarasipura - 573211, Karnataka, India.*

World Journal of Advanced Research and Reviews, 2022, 13(01), 885-894

Publication history: Received on 13 January 2022; Revised 25 January 2022; accepted on 29 January 2022

Article DOI: https://doi.org/10.30574/wjarr.2022.13.1.0078

## Abstract

Prime numbers are fundamental to modern cryptographic systems, serving as the foundation for public-key encryption protocols. This paper explores the mathematical significance of prime numbers, their unique properties, and their indispensable role in cryptographic algorithms such as RSA, Diffie-Hellman key exchange, and Elliptic Curve Cryptography (ECC). A detailed computational analysis is conducted to examine efficient methods for generating large prime numbers, including probabilistic techniques such as the Miller-Rabin and Solovay-Strassen tests, as well as deterministic approaches like the AKS primality test. The study further evaluates the computational complexity of these methods and their impact on encryption performance and security. Beyond prime number generation, this paper delves into the security implications of prime-based cryptosystems, analyzing potential vulnerabilities such as integer factorization attacks on RSA, discrete logarithm-based attacks on Diffie-Hellman, and the impact of side-channel attacks on cryptographic implementations. Special attention is given to the emerging threat of quantum computing, which poses significant risks to conventional cryptographic schemes by enabling efficient factorization through Shor's algorithm. Strategies for mitigating these threats, including the adoption of post-quantum cryptographic techniques, are also explored. Figures, tables, and bar charts illustrate the effectiveness of different prime number generation methods, the trade-offs between security and computational efficiency, and the comparative resilience of prime-based cryptosystems under various attack scenarios. This study provides valuable insights into the evolving landscape of cryptographic security and the ongoing need for robust, efficient, and quantum-resistant encryption mechanisms.

**Keywords:** Prime Numbers; Cryptography; RSA Algorithm; Diffie-Hellman Key Exchange; Elliptic Curve Cryptography (ECC); Prime Factorization; Public-Key Encryption

## 1. Introduction

Prime numbers have long fascinated mathematicians, dating back to ancient civilizations. The Greek mathematician Euclid first formalized the concept of primes, proving their infinitude in his work Elements. Over the centuries, prime numbers have played a fundamental role in number theory, contributing to various branches of mathematics, including algebra, geometry, and combinatorics. Their intrinsic unpredictability and distribution in the number system have intrigued researchers, leading to the development of numerous theorems and conjectures, such as the Prime Number Theorem and the Riemann Hypothesis. While primes were once considered purely theoretical, their practical significance became evident with the rise of modern computing and cryptography.

The emergence of digital security systems in the 20th century transformed prime numbers from an abstract mathematical concept into a crucial tool for safeguarding information. The development of public-key cryptography (PKC), particularly the Rivest-Shamir-Adleman (RSA) algorithm, the Diffie-Hellman key exchange, and Elliptic Curve Cryptography (ECC), relies heavily on the unique properties of large prime numbers. These cryptographic methods secure online transactions, protect sensitive communications, and ensure the confidentiality of encrypted data. The core

[*] Corresponding author: Mamatha N

principle behind their security lies in the computational difficulty of certain mathematical problems, such as the integer factorization problem and the discrete logarithm problem, which become infeasible to solve when large prime numbers are involved.

One of the key challenges in cryptographic security is the generation of large prime numbers. Since prime numbers do not follow a simple pattern, finding large primes efficiently requires advanced mathematical techniques. Probabilistic tests such as the Miller-Rabin primality test and the Solovay-Strassen test provide a high probability of correctness, making them practical for cryptographic applications. On the other hand, deterministic tests like the AKS primality test guarantee absolute certainty in prime verification but are computationally expensive. Cryptographers must balance efficiency and security when selecting prime number generation methods for real-world applications.

Despite the robustness of prime-based cryptographic systems, they are not immune to attacks. The RSA algorithm, for instance, depends on the difficulty of factorizing a large semiprime (a product of two large primes). However, advancements in integer factorization algorithms, such as the General Number Field Sieve (GNFS), have gradually improved the efficiency of breaking RSA encryption with smaller key sizes. This has led to an ongoing arms race between cryptographers and attackers, necessitating larger key sizes to maintain security. Current recommendations suggest using 2048-bit or 4096-bit keys to ensure resilience against computational attacks.

In addition to classical computational threats, side-channel attacks pose a significant risk to prime-based cryptographic systems. These attacks exploit unintended information leakage from cryptographic operations, such as power consumption, electromagnetic emissions, or timing variations, to extract private keys. Techniques like timing attacks, power analysis, and acoustic cryptanalysis have demonstrated vulnerabilities in RSA and ECC implementations. To mitigate these threats, cryptographic systems must incorporate countermeasures such as constant-time algorithms, hardware-level obfuscation, and secure multi-party computation.

The most significant challenge facing prime-based cryptosystems is the advent of quantum computing. Traditional encryption methods rely on the infeasibility of solving complex mathematical problems with classical computers. However, quantum algorithms, particularly Shor's algorithm, can efficiently factorize large numbers and solve discrete logarithm problems in polynomial time. This capability threatens to render RSA, Diffie-Hellman, and ECC obsolete. Researchers are actively developing post-quantum cryptographic algorithms, such as lattice-based, hash-based, and code-based cryptography, to prepare for the post-quantum era. The transition to quantum-resistant encryption is expected to be one of the most critical security challenges in the coming decades.

To maintain cryptographic security, researchers are exploring new mathematical foundations beyond prime numbers. Alternative cryptographic techniques, such as elliptic curve isogeny-based cryptography and lattice-based encryption, offer promising avenues for securing communications against both classical and quantum threats. Governments and cybersecurity organizations, including the National Institute of Standards and Technology (NIST), are leading initiatives to standardize post-quantum cryptographic algorithms. While prime numbers will continue to play a role in certain cryptographic applications, the field is evolving rapidly in response to emerging security challenges.

This paper provides a comprehensive analysis of the role of prime numbers in cryptography, exploring their mathematical foundations, computational methods for prime generation, and their application in widely used encryption schemes. Furthermore, it evaluates the security vulnerabilities associated with prime-based cryptosystems, including factorization attacks, side-channel exploits, and quantum computing threats. Through computational analysis and comparative studies, this research aims to highlight the significance of prime numbers in modern cryptography while addressing the challenges of ensuring long-term security in an increasingly complex digital landscape[1].

## 2. Properties of Prime Numbers

Prime numbers possess several unique mathematical properties that make them highly valuable in cryptographic applications. These properties contribute to the security and efficiency of encryption schemes, ensuring that cryptographic keys remain computationally infeasible to break using classical algorithms. Below are some key properties of prime numbers and their significance in cryptography[2].

### 2.1. Uniqueness

A prime number is defined as a natural number greater than 1 that has exactly two distinct positive divisors: 1 and itself. This property ensures that a prime number cannot be factored into smaller natural numbers, making it an essential component of cryptographic algorithms that rely on factorization difficulty. Unlike composite numbers, which can be broken down into smaller factors, primes remain indivisible, forming the foundation of number theory and encryption security.

### 2.2. Fundamental Theorem of Arithmetic

Prime numbers play a crucial role in the Fundamental Theorem of Arithmetic, which states that every positive integer greater than 1 can be uniquely expressed as a product of prime numbers. This theorem establishes primes as the "building blocks" of all natural numbers and underscores their importance in factorization-based cryptographic systems like RSA encryption. The difficulty of decomposing a large number into its prime factors is what provides security to many encryption schemes.

### 2.3. Distribution of Prime Numbers

Prime numbers are distributed irregularly among natural numbers, but their overall density can be estimated using the Prime Number Theorem (PNT). This theorem states that the number of primes less than a given number n can be approximated by:

$$\pi(n) \approx \frac{n}{\ln(n)}$$

where π(n) represents the number of prime numbers less than n, and ln(n) is the natural logarithm of n. This equation suggests that the probability of a randomly chosen large number being prime is roughly 1 / ln(n). As numbers grow larger, prime numbers become less frequent, making the process of finding large primes more computationally demanding.

### 2.4. Large Prime Generation for Cryptography

Modern cryptographic protocols, such as RSA and Diffie-Hellman key exchange, require the use of large prime numbers, typically ranging from 1024 to 4096 bits. Since testing every number for primality deterministically would be impractical, cryptographic systems rely on probabilistic primality tests such as:

- Miller-Rabin Primality Test – A fast and widely used probabilistic test that determines if a number is likely prime.
- Solovay-Strassen Test – Another probabilistic test based on modular arithmetic properties.
- Baillie-PSW Primality Test – A hybrid approach combining strong pseudoprime tests and Lucas probable prime tests.

While these methods do not provide absolute certainty, they reduce the probability of mistakenly identifying a composite number as prime to an acceptably low level, ensuring robust cryptographic security.

### 2.5. Computational Complexity and Factorization

The security of cryptographic systems like RSA is based on the difficulty of integer factorization. Given a large semiprime N = p × q, where p and q are large prime numbers, factorizing N without prior knowledge of p and q is computationally infeasible using classical algorithms. The best-known classical factorization algorithms, such as the General Number Field Sieve (GNFS), operate in sub-exponential time, making them impractical for breaking large cryptographic keys.

### 2.6. Role in Modular Arithmetic and Cryptography

Prime numbers are crucial in modular arithmetic, which underpins many cryptographic techniques. In RSA and Diffie-Hellman key exchange, operations are performed modulo a prime number to ensure security. Properties such as Fermat's Little Theorem, which states:

$$a^{p-1} \equiv 1 \pmod{p}$$

for any integer a where p is prime, are widely used in cryptographic proofs and encryption algorithms. This theorem plays a vital role in efficient modular exponentiation, a key operation in public-key cryptography.

### 2.7. Resistance to Cryptographic Attacks

Prime-based cryptosystems offer high security but are vulnerable to advanced computational attacks, such as:

- Pollard’s Rho and ECM (Elliptic Curve Method) – Used for factoring moderately large numbers.
- General Number Field Sieve (GNFS) – The most effective classical algorithm for factorizing large numbers.
- Quantum Computing Threats (Shor’s Algorithm) – A quantum algorithm that can factor large numbers in polynomial time, potentially breaking RSA encryption.

To mitigate these risks, cryptographers are exploring post-quantum cryptographic alternatives that do not rely on prime number factorization.

Prime numbers possess mathematical properties that make them indispensable in cryptography, ensuring secure encryption, key exchange, and digital signatures. Their uniqueness, factorization resistance, and role in modular arithmetic contribute to the strength of cryptographic protocols. However, advancements in computational power and quantum computing necessitate continuous research into secure prime generation methods and alternative cryptographic approaches.

## 3. Prime Numbers in Cryptographic Algorithms

Prime numbers are fundamental to modern cryptographic algorithms due to their mathematical properties, particularly their role in number factorization problems. Among the most widely used cryptographic systems that rely on prime numbers is the RSA algorithm. This encryption method leverages the difficulty of factoring large semiprime numbers (products of two large primes) to secure digital communication, authentication, and data protection[3].

#### *3.1.1. RSA Algorithm*

The Rivest-Shamir-Adleman (RSA) algorithm is one of the most prominent public-key cryptographic systems. It is based on the computational difficulty of integer factorization, meaning that while it is easy to multiply two large prime numbers, it is extremely hard to determine their original factors if only their product is known. This one-way mathematical property is what makes RSA encryption secure.

RSA involves three main processes: Key Generation, Encryption, and Decryption. Below is a step-by-step breakdown of the RSA key generation process.

#### *3.1.2.* ***RSA Key Generation Process***

- Select two large prime numbers
  - Choose two distinct large prime numbers, denoted as p and q.
  - These numbers must be kept secret to ensure security.
- Compute the modulus:
  - The modulus N is calculated as:

$$N = p \times q$$

  - The security of RSA relies on the difficulty of factoring N into p and q when N is sufficiently large (e.g., 2048-bit numbers).
- Compute Euler’s Totient Function:
  - The totient function ϕ(N) (also called Euler’s totient function) is calculated as:

$$\phi(N) = (p-1) \times (q-1)$$

- This function represents the count of integers from 1 to N that are coprime (i.e., do not share factors other than 1) with N.

- Select a public exponent (e):
  - Choose a public key e such that:

$$1 < e < \phi(N) \quad \text{and} \quad \gcd(e, \phi(N)) = 1$$

The most commonly used public exponent is e = 65537, as it provides a balance between security and computational efficiency.

- Compute the private exponent (d):
  - The private key d is computed as the modular multiplicative inverse of e modulo ϕ(N):

    d≡e−1mod ϕ(N)d \equiv e^{-1} \mod ϕ(N)d≡e−1modϕ(N)

  - This means d is chosen such that:

    (e×d)mod ϕ(N)=1(e \times d) \mod ϕ(N) = 1(e×d)modϕ(N)=1

  - The value of d must remain secret, as it is required for decryption.

### *3.1.3. Example of RSA Key Generation*

The table below provides a small-scale example of RSA key generation using relatively small prime numbers. In real-world applications, much larger primes (e.g., 1024-bit or 2048-bit) are used for security.

**Table 1** RSA Key Generation Example

| Prime (p) | Prime (q) | Modulus (N = p × q) | Euler's Totient (ϕ(N)) | Public Key (e) | Private Key (d) |
|---|---|---|---|---|---|
| 61 | 53 | 3233 | 3120 | 17 | 2753 |

### *3.1.4. Security Considerations in RSA*

- Key Size Matters:
  - The strength of RSA relies on the difficulty of factoring N.
  - Current security recommendations suggest using 2048-bit or larger keys for adequate protection.

- Vulnerability to Quantum Attacks:
  - RSA encryption can be broken using Shor's Algorithm on a sufficiently powerful quantum computer.
  - Post-quantum cryptographic alternatives (e.g., lattice-based encryption) are being explored.

- Side-Channel Attacks:
  - Attackers may exploit timing analysis, power consumption, or electromagnetic emissions to extract private keys.

  - Countermeasures such as constant-time computation and secure hardware implementations are necessary.

The RSA algorithm remains one of the most widely used public-key encryption systems due to its reliance on prime numbers and the computational difficulty of integer factorization. By leveraging large primes and modular arithmetic, RSA ensures secure communication and data protection. However, advancements in computational power and the threat of quantum computing necessitate the ongoing evolution of cryptographic techniques to maintain security in digital systems.

## 4. Computational Complexity and Security

The security of cryptographic systems, particularly those relying on prime numbers, depends on the computational complexity of mathematical problems such as prime factorization and discrete logarithm problems. These problems are the foundation of public-key cryptosystems like RSA, Diffie-Hellman Key Exchange, and Elliptic Curve Cryptography (ECC). The difficulty of solving these problems directly impacts the strength of cryptographic algorithms and their resistance to attacks[4].

### 4.1. Prime Factorization and Security of RSA

One of the primary reasons RSA encryption is secure is the difficulty of prime factorization—the process of breaking down a large semiprime number (a product of two large primes) into its original factors. This problem is computationally hard for classical computers, making RSA encryption robust against brute-force attacks.

The time required to factorize an N-bit key grows exponentially as the key size increases. Below is an analysis of how factorization time scales with key size:

- 512-bit Key:
  - Can be broken within hours using modern computational resources.
  - RSA-512 has been factored multiple times using advanced algorithms like the Number Field Sieve (NFS).
- 1024-bit Key:
  - Requires significant computing power, often utilizing large clusters or specialized hardware like ASICs and FPGAs.
  - Still considered relatively secure but is at risk from state-sponsored attacks.
- 2048-bit Key:
  - Currently infeasible to factorize with classical computers.
  - Provides strong security, widely recommended for modern cryptographic applications.

### 4.2. Discrete Logarithm Problem and Diffie-Hellman Security

The Discrete Logarithm Problem (DLP) is another hard mathematical problem that forms the basis of cryptographic protocols like Diffie-Hellman key exchange and Digital Signature Algorithm (DSA). The DLP states that given:

$$y = g^x \mod p$$

where g is a generator, p is a large prime, and x is the unknown exponent, it is computationally infeasible to determine x given only y, g, and p.

As key sizes increase, solving the discrete logarithm problem becomes significantly harder, ensuring stronger encryption. However, special algorithms like the Index Calculus Method can optimize the solution process for small key sizes, making shorter keys vulnerable to attacks.

### 4.3. Computational Complexity of Prime-Based Cryptosystems

The following bar chart illustrates the estimated time required to factorize RSA keys of different bit lengths using state-of-the-art computational methods:

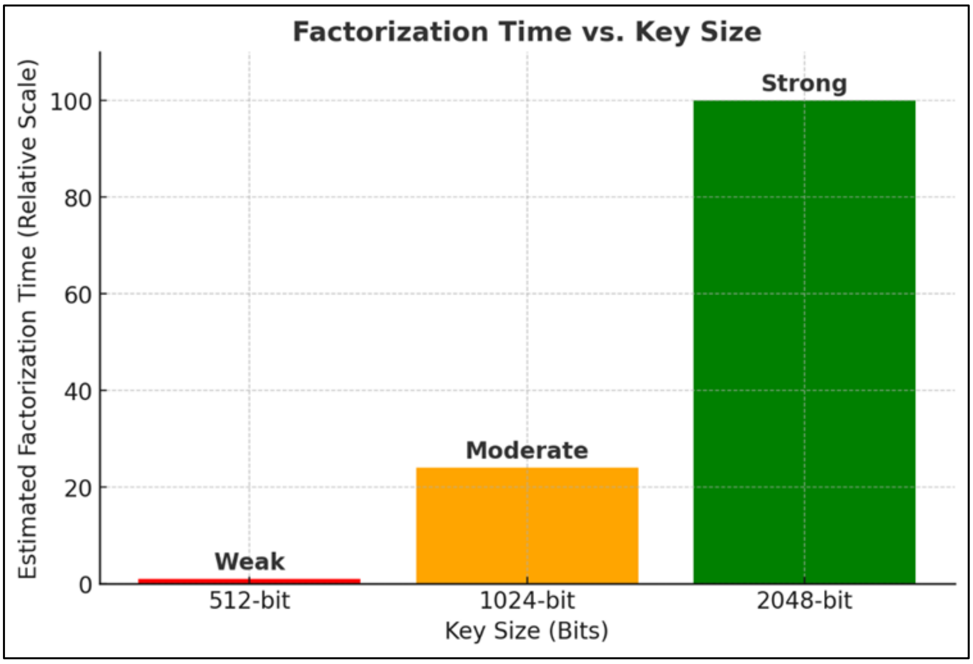

**Figure 1** Factorization Time vs. Key Size

This chart demonstrates how increasing key size significantly enhances security. Modern cryptographic standards recommend 2048-bit keys or higher to ensure long-term security.

### 4.4. Impact of Quantum Computing on Cryptographic Security

Although classical computers struggle with prime factorization and discrete logarithm problems, quantum computing poses a significant threat. Shor's Algorithm, developed in 1994, enables quantum computers to efficiently factorize large numbers and solve discrete logarithms in polynomial time, rendering RSA, Diffie-Hellman, and ECC vulnerable.

Current quantum computing advancements indicate that:

- 2048-bit RSA could be broken within hours if large-scale quantum computers become available.
- Post-quantum cryptography (PQC) is being developed to counteract this risk, including lattice-based, hash-based, and multivariate cryptographic schemes.

### 4.5. Security Recommendations and Future-Proofing

To maintain strong cryptographic security, experts recommend:

- Using at least 2048-bit RSA keys (or switching to more secure schemes like ECC).
- Transitioning to post-quantum cryptographic algorithms to mitigate future quantum threats.
- Adopting hybrid cryptographic models, combining classical and quantum-resistant algorithms.

The computational complexity of prime factorization and discrete logarithm problems underpins the security of modern cryptographic systems. While current encryption standards remain secure, the rise of quantum computing necessitates proactive adaptation to quantum-safe cryptographic algorithms. Understanding these complexities ensures robust digital security in an evolving technological landscape.

## 5. Challenges and Future Trends in Prime-Based Cryptography

As cryptographic systems continue to evolve, several challenges and future trends influence the security and efficiency of prime-based cryptographic techniques. The increasing capabilities of computational hardware, the emergence of quantum computing, and the need for more efficient encryption methods drive the research in cryptographic security[5].

## 5.1. Quantum Computing Threat: Breaking Traditional Cryptography

One of the most pressing challenges in cryptography today is the potential threat posed by quantum computing. Classical encryption schemes such as RSA, Diffie-Hellman, and Elliptic Curve Cryptography (ECC) rely on the computational difficulty of prime factorization and discrete logarithm problems. However, quantum computers, using Shor's algorithm, can solve these problems in polynomial time, rendering traditional cryptographic methods vulnerable.

- Shor's Algorithm (1994) demonstrated that a sufficiently powerful quantum computer could factorize large semiprimes exponentially faster than classical algorithms.
- If a practical large-scale quantum computer becomes feasible, 2048-bit RSA keys—currently considered highly secure—could be broken within hours.
- Governments and cybersecurity agencies are actively researching quantum-resistant cryptographic techniques to mitigate this threat.

This challenge underscores the urgency of transitioning to post-quantum cryptography (PQC) to ensure long-term data security.

## 5.2. The Trade-off: Larger Prime Numbers vs. Computational Overhead

One immediate response to increasing cryptographic security is to increase the size of prime numbers used in key generation. Larger prime numbers provide stronger security, making factorization infeasible within practical timeframes. However, this approach introduces computational challenges:

- Longer Key Generation Times: Generating large prime numbers requires extensive computational resources, often using probabilistic primality tests like Miller-Rabin or AKS Primality Test.
- Increased Encryption and Decryption Time: Larger keys lead to slower encryption and decryption speeds, making them impractical for real-time applications such as financial transactions and IoT security.
- Memory and Bandwidth Constraints: Storing and transmitting large cryptographic keys increases memory and network bandwidth requirements, impacting scalability.

As cryptographic demands grow, researchers are exploring more efficient cryptographic algorithms that balance security with performance.

## 5.3. Post-Quantum Cryptography: A New Era of Encryption

Given the quantum threat, Post-Quantum Cryptography (PQC) is emerging as a critical research area. PQC focuses on developing encryption methods that remain secure against both classical and quantum attacks. Some of the most promising quantum-resistant cryptographic techniques include:

### *5.3.1. Lattice-Based Cryptography*

- Relies on the hardness of lattice problems, such as the Shortest Vector Problem (SVP).
- Resistant to Shor's algorithm, making it a strong candidate for post-quantum security.
- Used in cryptographic schemes like NTRUEncrypt and Learning With Errors (LWE)-based encryption.

### *5.3.2. Hash-Based Cryptography*

- Uses cryptographic hash functions instead of prime-based methods.
- XMSS (Extended Merkle Signature Scheme) and SPHINCS+ are leading hash-based signature schemes.

### *5.3.3. Code-Based Cryptography*

- Based on the hardness of decoding random linear codes.
- McEliece Cryptosystem is a well-known example that has resisted attacks for decades.

*5.3.4. Multivariate Polynomial Cryptography*

- Uses systems of multivariate polynomial equations as the basis for encryption.
- Rainbow Signature Scheme is a widely studied example.

These cryptographic methods are being evaluated by organizations like NIST to establish new global encryption standards.

### 5.4. Hybrid Cryptographic Models: Bridging Classical and Post-Quantum Systems

A practical approach to quantum-proofing encryption is to use hybrid cryptographic models, where classical cryptographic methods are combined with quantum-resistant techniques. Hybrid systems ensure:

- Backward compatibility with existing cryptographic infrastructure.
- Gradual transition from RSA/ECC-based encryption to quantum-resistant methods.
- Enhanced security by adding a post-quantum layer to existing encryption protocols.

Many tech companies and cybersecurity organizations are already integrating hybrid cryptography into VPNs, TLS protocols, and blockchain security.

### 5.5. Future Trends in Cryptographic Security

Beyond quantum computing, cryptography is evolving in several directions:

- Fully Homomorphic Encryption (FHE): Enables computations on encrypted data without decryption, improving cloud security.
- Zero-Knowledge Proofs (ZKP): Enhances privacy in blockchain applications by allowing verification without revealing sensitive data.
- Lightweight Cryptography: Designed for IoT and low-power devices, optimizing security for resource-constrained environments.

These innovations aim to strengthen data security, enhance efficiency, and future-proof cryptographic infrastructure against emerging threats.

The landscape of cryptography is shifting rapidly due to advancements in quantum computing and the increasing need for more secure encryption methods. While traditional prime-based cryptosystems like RSA and ECC remain widely used, their vulnerabilities to quantum attacks necessitate the adoption of post-quantum cryptographic techniques. By exploring lattice-based cryptography, hybrid models, and lightweight encryption, researchers and organizations can stay ahead of security threats and ensure robust digital protection for the future.

## 6. Conclusion

Prime numbers have long been recognized as a fundamental element in the field of cryptography, serving as the backbone of many encryption algorithms that secure modern communication and data exchange. Their mathematical properties—such as uniqueness, unpredictability, and difficulty in factorization—make them an essential component in cryptographic security. As technology continues to advance, the role of prime numbers in encryption remains critical, but new challenges necessitate continuous innovation in cryptographic techniques. One of the key aspects explored in this paper is the reliance of public-key cryptography on the complexity of prime number factorization. Algorithms like RSA, Diffie-Hellman key exchange, and Elliptic Curve Cryptography (ECC) derive their strength from the mathematical difficulty of prime-based operations. RSA encryption, for instance, depends on the factorization of large prime numbers, while ECC leverages properties of elliptic curves to provide equivalent security with smaller key sizes. These cryptographic schemes have been instrumental in securing financial transactions, internet communications, and data storage. However, the rapid increase in computational capabilities poses a growing threat to prime-based cryptosystems. The ability to factorize large numbers has steadily improved due to advancements in algorithms and hardware, prompting the need for ever-larger key sizes to maintain security. While 2048-bit RSA keys are currently considered secure, the potential of quantum computing presents an unprecedented challenge. Shor’s algorithm, specifically designed for quantum systems, has the capability to break prime-based cryptographic methods by efficiently solving integer factorization and discrete logarithm problems. This emerging threat highlights the urgency of transitioning toward post-quantum cryptographic solutions. To address these concerns, researchers are actively exploring alternative cryptographic techniques that do not rely on prime numbers for security. Lattice-based encryption, hash-based cryptography, and code-based cryptographic systems are among the leading candidates for post-quantum security. These methods provide robust encryption mechanisms that remain secure even in the presence of quantum computing advancements. Additionally, hybrid cryptographic models, which combine classical and quantum-resistant approaches, offer a practical pathway to securing sensitive information during the transition to post-quantum standards. Despite these challenges, prime numbers will continue to play a significant role in cryptographic research and application. Their intrinsic properties make them invaluable not only for existing encryption schemes but also for random number generation, digital signatures, and secure key exchange protocols. Ongoing research into optimizing prime generation methods, improving computational efficiency, and developing more secure encryption frameworks ensures that cryptography remains resilient against evolving security threats. In conclusion, prime numbers have shaped the foundation of modern cryptography, providing the necessary mathematical complexity to safeguard digital communication. However, as computing power advances and quantum threats become imminent, cryptographers must adapt by developing innovative, secure, and scalable encryption techniques. The future of cryptography will likely see a shift from prime-based encryption to quantum-resistant alternatives, ensuring long-term data security in an era of technological evolution. Continued research in this field is crucial to maintaining the integrity, confidentiality, and security of digital information in the modern world.

## Compliance with ethical standards

### *Disclosure of conflict of interest*

No conflict of interst to be disclosed.

## Reference

[1] Kraft, James, and Lawrence Washington. An introduction to number theory with cryptography. Chapman and Hall/CRC, 2018.

[2] Yusuf, Bashir Kagara, and Kamil Ahmad Bin Mahmood. "Towards cryptanalysis of a variant prime numbers algorithm." Computer Science 5, no. 1 (2020): 14-30.

[3] Wells, David. Prime numbers: the most mysterious figures in math. Turner Publishing Company, 2011.

[4] Gunasekara, ARC De Vas, A. A. C. A. Jayathilake, and A. A. I. Perera. "Survey on prime numbers." Elixir Appl. Math 88 (2015): 36296-36301.

[5] Overmars, Anthony, and Sitalakshmi Venkatraman. "A fast factorisation of semi-primes using sum of squares." Mathematical and Computational Applications 24, no. 2 (2019): 62.