WWW.IJRPR.COM

International Journal of Research Publication and Reviews

Journal homepage: www.ijrpr.com ISSN 2582-7421

Enhanced Integer Factor Optimization in RSA Cryptography using Shor Quantum

[1]Padmapriya V, [2]Thanigaivelan A, [3]Harish J, [4]Surya G

[1]Department of Information Technology Sri Manakula Vinayagar Engineering College Puducherry, India ***padmapriya@smvec.ac.in***
[2]Department of Information Technology, Sri Manakula Vinayagar Engineering College, Puducherry, India ***tanigaivelashokkumar@gmail.com***
[3]Department of Information Technology,Sri Manakula Vinayagar Engineering College, Puducherry, India ***harishjs2707@gmail.com***
[4]Department of Information Technology, Sri Manakula Vinayagar Engineering College Puducherry, India surya.gunasekaran218@gmail.com

ABSTRACT—

The project titled "Enhanced Integer Factor Optimization in RSA Cryptography using Shor Quantum" proposes an innovative strategy to fortify the resilience of RSA cryptography against emerging quantum computing threats. In response to the escalating risk posed by quantum adversaries to conventional cryptographic methods, this research endeavors to amalgamate two forefront quantum technologies: the E91 Protocol and Shor's Algorithm. The E91 Protocol, rooted in the principles of quantum mechanics, facilitates secure key distribution, offering impregnable defense against eavesdropping attempts. Concurrently, Shor's Algorithm, a quantum algorithm renowned for its capability to efficiently factor large numbers, poses a significant challenge to widely-used asymmetric encryption schemes like RSA and ECC. By integrating these quantum technologies with RSA cryptography, this project aims to proactively mitigate the vulnerability of traditional cryptographic systems. The synergy between the E91 Protocol and Shor's Algorithm promises to establish a robust defense mechanism against quantum threats, ensuring the confidentiality, integrity, and authenticity of sensitive information in the face of evolving cybersecurity landscapes. Through rigorous theoretical analysis, simulations, and practical implementations, the efficacy and feasibility of the proposed approach will be scrutinized, with the ultimate goal of advancing quantum-resistant cryptography and fortifying the security posture of digital communication infrastructures.

Keywords—E91, Shor, RSA, ECC, Cryptography.

Introduction

In an era marked by unprecedented reliance on interconnected digital networks, the imperative to safeguard sensitive information against an ever-evolving array of cyber threats has never been more critical. The project at hand delves into the realm of network security with a comprehensive and forward-looking perspective, aiming to address the multifaceted challenges posed by malicious actors, emerging technologies, and the relentless evolution of cyber threats. With the proliferation of interconnected devices and the exponential growth of data transmission, the integrity, confidentiality, and availability of networked systems are continually under siege. This project embarks on a journey to fortify the foundations of network security, leveraging cutting-edge technologies, innovative methodologies, and rigorous analysis to devise holistic solutions that transcend conventional paradigms. By amalgamating theoretical insights with practical implementations, this endeavor seeks to not only mitigate existing vulnerabilities but also anticipate and preemptively counter emerging threats, thus fostering a resilient and secure digital ecosystem for the benefit of individuals, organizations, and society at large.

Cryptography

Cryptography, often regarded as the cornerstone of modern information security, is a multifaceted discipline that encompasses the art and science of securing digital communication and data. At its core, cryptography involves the transformation of plaintext information into ciphertext through various algorithms and techniques, rendering it unintelligible to unauthorized entities. This transformation process relies on cryptographic keys, which serve as the linchpin for both encryption and decryption operations, providing the means to protect sensitive information from unauthorized access or tampering.

Throughout history, cryptography has played a pivotal role in safeguarding sensitive communications, diplomatic correspondence, financial transactions, and military operations. From ancient civilizations employing rudimentary substitution ciphers to modern-day cryptographic systems

leveraging complex mathematical principles, cryptography has evolved significantly to meet the ever-changing demands of securing information in an interconnected digital world.

One of the fundamental concepts in cryptography is symmetric encryption, wherein a single key is used for both encryption and decryption operations. Algorithms like the Data Encryption Standard (DES), Advanced Encryption Standard (RSA), and Triple DES (3DES) exemplify the prowess of symmetric cryptography, offering robust protection against unauthorized access when implemented correctly.

In contrast, asymmetric encryption, also known as public-key cryptography, introduces the concept of key pairs: a public key for encryption and a private key for decryption. This paradigm shift not only simplifies key management but also facilitates secure communication over untrusted channels. Notable asymmetric encryption schemes such as RSA (Rivest–Shamir–Adleman) and Elliptic Curve Cryptography (ECC) underpin the security infrastructure of modern digital ecosystems, enabling secure online transactions, digital signatures, and key exchanges.

Beyond encryption, cryptography encompasses a myriad of cryptographic primitives and protocols designed to address various security requirements and use cases. Hash functions, for instance, play a crucial role in ensuring data integrity by generating fixed-size cryptographic hashes that uniquely represent the input data. Protocols like Secure Sockets Layer (SSL)/Transport Layer Security (TLS) and Internet Protocol Security (IPsec) provide secure communication channels through encryption, authentication, and integrity verification mechanisms.

The advent of quantum computing poses both opportunities and challenges for cryptography. While quantum computers hold the potential to break traditional cryptographic schemes like RSA and ECC through algorithms such as Shor's Algorithm, they also inspire the development of quantum-resistant cryptographic techniques. Post-quantum cryptography, characterized by algorithms like lattice-based cryptography, hash-based cryptography, and code-based cryptography, aims to withstand the computational power of quantum adversaries, ensuring the long-term security of digital communications.

Moreover, cryptography intersects with various domains, including blockchain technology, where cryptographic primitives like cryptographic hash functions and digital signatures underpin the security and immutability of distributed ledgers. In the realm of privacy-preserving technologies, cryptographic techniques like homomorphic encryption and zero-knowledge proofs enable secure data computation and verification without compromising data confidentiality.

Cryptography stands as an indispensable enabler of trust, confidentiality, and integrity in the digital age, empowering individuals, organizations, and societies to safeguard their most valuable assets against the pervasive threat of cybercrime and unauthorized surveillance. As technology continues to evolve, cryptography remains at the forefront of innovation, driving the quest for stronger, more resilient cryptographic solutions that uphold the principles of security and privacy in an ever-changing landscape of digital threats and challenges.

Types of Cryptography:

Cryptography encompasses various types of algorithms and mechanisms, each designed to fulfill specific security requirements and use cases. Below, I'll delve into the major types of cryptography, along with notable algorithms and their mechanisms:

Symmetric Cryptography:

Symmetric cryptography, also known as secret-key cryptography, employs a single shared key for both encryption and decryption operations. The primary advantage of symmetric encryption lies in its efficiency, making it suitable for encrypting large volumes of data.

Notable Algorithms:

Data Encryption Standard (DES):

DES is one of the earliest and most well-known symmetric encryption algorithms. It operates on 64-bit blocks of plaintext and uses a 56-bit key. Despite its widespread adoption, DES is now considered obsolete due to its vulnerability to brute-force attacks.

Advanced Encryption Standard (RSA):

RSA is a widely adopted symmetric encryption algorithm standardized by the National Institute of Standards and Technology (NIST). It supports key sizes of 128, 192, or 256 bits and operates on 128-bit blocks of data. RSA is renowned for its efficiency, security, and versatility, making it the de facto standard for symmetric encryption.

Asymmetric Cryptography:

Asymmetric cryptography, also referred to as public-key cryptography, utilizes a pair of mathematically related keys: a public key for encryption and a private key for decryption (or vice versa). This asymmetric key pair enables secure communication and digital signatures without necessitating the exchange of secret keys

Notable Algorithms:

RSA (Rivest–Shamir–Adleman)**: RSA is a cornerstone of asymmetric cryptography, named after its inventors. It relies on the computational difficulty of factoring large prime numbers. RSA is widely used for secure communication, digital signatures, and key exchange in various applications.

Elliptic Curve Cryptography (ECC):

ECC leverages the algebraic properties of elliptic curves to provide strong security with shorter key lengths compared to other asymmetric algorithms like RSA. ECC is particularly suitable for resource-constrained environments such as mobile devices and IoT devices.

Hash Functions:

Hash functions are cryptographic primitives that generate fixed-size output (hash) from variable-length input (message). Hash functions are crucial for ensuring data integrity, authentication, and digital signatures.

Notable Algorithms:

SHA-256 (Secure Hash Algorithm 256-bit):

SHA-256 is a member of the SHA-2 family of hash functions, standardized by NIST. It produces a 256-bit (32-byte) hash value and is widely used in blockchain technology, digital signatures, and data integrity verification.

MD5 (Message Digest Algorithm 5):

MD5 is a widely used hash function despite its vulnerabilities to collision attacks. It generates a 128-bit (16-byte) hash value and is commonly used for checksums and digital signatures. However, due to its weaknesses, it is now considered obsolete for cryptographic purposes.

Hybrid Cryptography:

Hybrid cryptography combines the strengths of symmetric and asymmetric encryption to achieve both efficiency and security. In hybrid schemes, symmetric encryption is typically used for encrypting the bulk of data, while asymmetric encryption is employed for securely exchanging the symmetric keys.

Notable Mechanisms:

SSL/TLS (Secure Sockets Layer/Transport Layer Security)

: SSL/TLS protocols utilize hybrid cryptography to secure communication over the internet. They establish a secure channel between a client and server using asymmetric encryption for key exchange and symmetric encryption for data transmission.

PGP (Pretty Good Privacy):

PGP is a popular encryption program that employs hybrid cryptography for secure email communication. It combines symmetric encryption (e.g., RSA) for encrypting message contents and asymmetric encryption (e.g., RSA) for securely transmitting the symmetric key.

RSA (Rivest Shamir Adleman):

RSA (Rivest-Shamir-Adleman) is a widely used asymmetric encryption algorithm named after its inventors: Ron Rivest, Adi Shamir, and Leonard Adleman. It's widely used in securing data transmission, digital signatures, and key exchange over insecure networks such as the Internet.

Key Generation:

Choose two distinct prime numbers

Calculate their product,This forms the modulus for the public and private keys.

Compute Euler's totient function, \

Choose an integer such that is coprime to (i.e., their greatest common divisor is 1). This is typically a small prime number like 65537 or randomly generated

Calculate the private exponent such that, i.eIt is the modular multiplicative inverse of modulo.

Public and Private Keys:

The public key consists of the modulus and the public exponent . It is used for encryption.

The private key consists of the modulus and the private exponent. It is used for decryption.

Encryption:

To encrypt a message , which is typically a number smaller than the modulus , the sender uses the recipient's public key. The sender computes and sends to the recipient.

Decryption:

The recipient, who possesses the private key, computes to recover the original message

Security:

The security of RSA relies on the difficulty of factoring the product of two large prime numbers, , into its prime factors.. As of my last update, RSA remains secure with sufficiently large key sizes. However, advances in quantum computing may pose a threat to RSA's security.

Usage:

RSA is commonly used in various cryptographic protocols, such as SSL/TLS for securing web communications, PGP for secure email, and SSH for secure remote access.

Padding:

To enhance security and prevent certain attacks like chosen ciphertext attacks, RSA typically uses padding schemes like PKCS#1 (RSRSA-OAEP for encryption and RSASSA-PSS for digital signatures) to ensure the security of encrypted data.

Drawback of RSA Algorithm

The integer factorization problem is at the core of the RSA algorithm's security. RSA relies on the presumed difficulty of factoring large composite numbers into their prime factors. The security of RSA rests on the assumption that it is computationally infeasible to factor a large composite number into its prime factors.

Here's an elaboration on this problem and its implications for RSA security:

Composite Number Generation:

In RSA, two large prime numbers,are chosen randomly and multiplied together to form the composite number
The security of RSA heavily relies on the fact that given, it's extremely difficult to determine efficiently.
Difficulty of Factoring:
Factoring large numbers is considered a computationally intensive task.
As of the time of my last update, the most efficient known algorithms for general-purpose integer factorization, such as the General Number Field Sieve (GNFS), have a high computational complexity, making it infeasible to factor large numbers with hundreds of digits within a reasonable timeframe using classical computers.
Security Implications:
The security of RSA depends on the assumption that no efficient algorithm exists for factoring large composite numbers.
If an efficient algorithm for factoring large composite numbers were discovered, it would undermine the security of RSA, as an attacker could efficiently compute the prime factors of the modulus and thereby break the RSA encryption scheme.
Cryptographers continuously monitor developments in integer factorization algorithms and adjust RSA key lengths accordingly to maintain the desired level of security. As computational power increases and new mathematical techniques emerge, longer RSA keys are periodically recommended to withstand potential attacks.
Quantum Computing Threat:
One of the most significant potential threats to RSA's security comes from quantum computers. Quantum computers, if sufficiently developed, could employ algorithms such as Shor's algorithm to efficiently factor large composite numbers, rendering RSA insecure. Therefore, the cryptographic community is actively researching post-quantum cryptography, which aims to develop encryption schemes that are resistant to attacks from quantum computers. Post-quantum cryptographic algorithms are being considered as potential replacements for RSA and other current cryptographic standards.
The integer factorization problem lies at the heart of RSA's security. The presumed difficulty of factoring large composite numbers into their prime factors forms the basis for RSA's security, and any significant breakthrough in factoring algorithms or the advent of quantum computing could potentially compromise the security of RSA. Therefore, ongoing research and advancements in both classical and quantum computing are crucial for understanding and addressing the security implications of the integer factorization problem in RSA.
Shor Algorithm
Shor's algorithm, discovered by mathematician Peter Shor in 1994, is a quantum algorithm that efficiently factors large composite numbers into their prime factors. Shor's algorithm poses a significant threat to the security of widely-used public-key cryptosystems like RSA, which rely on the presumed difficulty of factoring large numbers. Here's a detailed explanation of how Shor's algorithm works:
Quantum Computing Prerequisites:
Shor's algorithm exploits the parallelism inherent in quantum computing to perform certain computations significantly faster than classical computers.
Quantum computers leverage qubits, which can exist in superposition states, allowing them to represent multiple values simultaneously. This property enables quantum computers to explore many possible solutions to a problem simultaneously, leading to exponential speedup for certain algorithms.
Quantum Fourier Transform (QFT):
Shor's algorithm relies on the quantum Fourier transform, a quantum analogue of the classical Fourier transform.
The quantum Fourier transform efficiently computes the discrete Fourier transform of the amplitudes of a quantum state.
This transform is crucial for finding the period of a periodic function efficiently, which is the key step in Shor's algorithm for factoring large numbers.
Key Components of Shor's Algorithm:
Classical Preprocessing:
Shor's algorithm begins with some classical preprocessing steps. The number to be factored, , is chosen, and some additional parameters are selected.
Quantum Part:
Shor's algorithm operates in two quantum registers: the control register and the target register.
Superposition of States: The control register is prepared in a superposition of all possible input values.
Quantum Fourier Transform: The quantum Fourier transform is applied to the control register.
Modular Exponentiation: The result of the quantum Fourier transform is used to perform modular exponentiation in the target register.
Measurement: Finally, a measurement is made on the control register. With a high probability, the measured value will be the period of the modular exponentiation function.
Classical Postprocessing: After obtaining the measured period, classical algorithms are used to analyze this result and extract the prime factors..
Complexity and Efficiency::
Shor's algorithm achieves exponential speedup over the best-known classical algorithms for factoring large numbers.
The complexity of Shor's algorithm is polynomial in the number of bits of the input number , making it significantly faster than classical factoring algorithms like the General Number Field Sieve (GNFS) for large numbers.
For RSA encryption, which relies on the difficulty of factoring large composite numbers, Shor's algorithm represents a potential threat, as it could efficiently factor the modulus and break RSA encryption.

Implications for Cryptography:

The discovery of Shor's algorithm has significant implications for cryptography, particularly for public-key cryptosystems like RSA and Elliptic Curve Cryptography (ECC), whose security is based on the presumed difficulty of certain mathematical problems.If sufficiently large and scalable quantum computers are built, they could efficiently break RSA and other widely-used cryptographic schemes, necessitating the development and adoption of quantum-resistant cryptographic algorithms.In summary, the potential of quantum computing to solve certain problems exponentially faster than classical computers. Its efficient factorization of large numbers poses a threat to classical cryptographic systems like RSA, underscoring the importance of researching and developing post-quantum cryptographic algorithms resistant to quantum attacks.

LITERATURE SURVEY

Shor's Algorithm Using Efficient Approximate Quantum Fourier Transform

The paper titled "Shor's Algorithm Using Efficient Approximate Quantum Fourier Transform" introduces a novel optimization strategy for Shor's algorithm, a renowned quantum algorithm for integer factorization. By leveraging an

efficient approximate quantum Fourier transform (QFT), the study aims to streamline the algorithm's computational demands while maintaining its efficacy in factoring large composite numbers[1]. This optimization is particularly significant for advancing the practicality and scalability of Shor's algorithm on near-term quantum computing platforms. The proposed approach seeks to strike a balance between computational efficiency and accuracy, offering promising prospects for addressing the inherent challenges of quantum computing implementations. Furthermore, the optimization holds potential implications for accelerating research and development in quantum cryptography and other quantum computing applications reliant on integer factorization.

Fast Generation of RSA Keys Using Smooth Integers

The paper titled "Fast Generation of RSA Keys Using Smooth Integers" presents a novel method for expediting the generation of RSA cryptographic keys by leveraging smooth integers.[2] The term "smooth integers" refers to numbers that have only small prime factors, making them suitable candidates for RSA key generation. This study aims to streamline the process of key generation, which traditionally involves selecting large prime numbers and computing their product. By focusing on smooth integers, the proposed method accelerates key generation by reducing the search space and computational overhead. This optimization is particularly significant for scenarios requiring rapid deployment of RSA encryption, such as in secure communication protocols and cryptographic applications. The research demonstrates the feasibility of efficiently generating RSA keys using smooth integers, offering potential benefits in terms of speed, resource utilization, and overall cryptographic efficiency. Furthermore, this approach holds promise for enhancing the scalability and responsiveness of RSA-based cryptographic systems, contributing to advancements in secure data transmission and digital communication protocols.

A Lightweight Post-Quantum Lattice-Based RSA for Secure Communications

The paper titled "A Lightweight Post-Quantum Lattice-Based RSA for Secure Communications" introduces a novel cryptographic scheme aimed at addressing the potential threat posed by quantum computers to traditional RSA encryption.[3] This scheme is based on lattice-based cryptography, a class of cryptographic techniques believed to be resistant to attacks from quantum computers. The proposed method aims to provide a lightweight alternative to traditional RSA encryption while maintaining a high level of security against both classical and quantum adversaries. By leveraging lattice-based techniques, the cryptographic scheme offers robust security guarantees, making it suitable for secure communication applications in environments where protection against quantum attacks is paramount. The research highlights the feasibility of implementing lattice-based RSA encryption in resource-constrained environments, such as IoT devices or embedded systems, without compromising security or performance. Additionally, the paper contributes to the ongoing efforts in developing post-quantum cryptographic solutions capable of withstanding the potential advent of powerful quantum computing technologies.

Security Issues of Novel RSA Variant

The paper titled "Security Issues of Novel RSA Variant" investigates the security implications associated with a new variant of the RSA cryptographic algorithm.[4] This variant introduces modifications or enhancements to the traditional RSA scheme, potentially offering benefits such as improved efficiency or resistance against specific types of attacks. However, the study aims to scrutinize the security of this novel RSA variant, identifying potential vulnerabilities or weaknesses that may compromise its effectiveness in real-world applications. By conducting a thorough analysis of the variant's security properties, including its key generation, encryption, and decryption processes, the research sheds light on any potential pitfalls or vulnerabilities that adversaries could exploit. Understanding these security issues is crucial for ensuring the robustness and reliability of cryptographic systems deployed in sensitive or critical environments. The findings of the study contribute to the broader understanding of RSA variants' security properties, guiding practitioners and researchers in designing and implementing secure cryptographic solutions for various applications. Additionally, the paper underscores the importance of rigorous security evaluations and risk assessments in the development and deployment of novel cryptographic algorithms to mitigate potential threats and vulnerabilities effectively.

Implementation of the Shor Algorithm on a Quantum Circuit

The paper titled "Implementation of the Shor Algorithm on a Quantum Circuit" explores the practical realization of Shor's algorithm, a powerful quantum algorithm for integer factorization, on a quantum circuit. Shor's algorithm poses a significant challenge for implementation due to the intricate quantum operations involved in its execution.[5] This study aims to bridge the gap between theory and practice by demonstrating the

feasibility of executing Shor's algorithm on a quantum circuit architecture. By meticulously designing and simulating quantum circuits that emulate the steps of Shor's algorithm, the research showcases the practical considerations and challenges associated with implementing quantum algorithms in real-world quantum computing hardware. The findings provide valuable insights into the resource requirements, error mitigation strategies, and scalability issues that arise during the implementation of Shor's algorithm on quantum circuits. Moreover, the study contributes to the advancement of quantum algorithm development and quantum computing technologies, offering a roadmap for future research and development efforts in harnessing the power of quantum algorithms for practical applications such as integer factorization and cryptography.

Implementation of 5-Qubit approach-based Shor's Algorithm in IBM Qiskit

The paper titled "Implementation of 5-Qubit approach-based Shor's Algorithm in IBM Qiskit" delves into the practical realization of Shor's algorithm, a quantum algorithm for integer factorization, specifically focusing on a 5-qubit implementation using IBM Qiskit, a quantum computing framework.[6] Shor's algorithm has immense potential for revolutionizing cryptography and computational number theory, but its implementation on current quantum hardware poses significant challenges due to limitations such as qubit coherence times and error rates. This study aims to address these challenges by demonstrating a 5-qubit approximation of Shor's algorithm within the constraints of existing quantum computing platforms. By designing and simulating quantum circuits using IBM Qiskit, the research showcases the step-by-step execution of Shor's algorithm, highlighting the quantum gates, measurements, and error correction techniques utilized in the implementation process. The findings provide valuable insights into the practical considerations, resource requirements, and limitations of implementing Shor's algorithm on current quantum computing hardware. Furthermore, the study contributes to the ongoing efforts in advancing quantum algorithm development and quantum computing technologies, paving the way for future improvements in scalability, error mitigation, and algorithm optimization for practical applications in cryptography and computational mathematics.

Comment on "Efficient and Secure Outsourcing Scheme for RSA Decryption in Internet of Things"

The paper titled "Efficient and Secure Outsourcing Scheme for RSA Decryption in Internet of Things" introduces a novel scheme aimed at outsourcing RSA decryption tasks securely and efficiently within Internet of Things (IoT) environments. RSA decryption, which is computationally intensive, can pose challenges for resource-constrained IoT devices.[7] This scheme aims to alleviate this burden by outsourcing the decryption process to more powerful entities while ensuring security and efficiency. By leveraging cryptographic techniques and secure protocols, the proposed scheme facilitates the secure transmission of encrypted data from IoT devices to external servers or cloud platforms, where the decryption process is performed. The study emphasizes the importance of efficiency and security in IoT deployments, where devices often have limited computational capabilities and face potential security threats. By outsourcing RSA decryption, the scheme enables IoT devices to offload resource-intensive tasks without compromising data confidentiality or integrity. Additionally, the research contributes to the broader discourse on secure outsourcing schemes for IoT applications, addressing key considerations such as data privacy, authentication, and cryptographic protocols. Overall, the findings provide valuable insights into the design and implementation of efficient and secure RSA decryption outsourcing schemes tailored to the unique requirements of IoT environments, fostering advancements in IoT security and cryptographic techniques.

Implementation of RSA Signatures on GPU and CPU Architectures

The paper titled "Implementation of RSA Signatures on GPU and CPU Architectures" presents a study focusing on the efficient implementation of RSA signature generation and verification algorithms on both GPU and CPU architectures.[8] RSA signatures are widely used in cryptographic protocols for ensuring data integrity, authenticity, and non-repudiation. This study aims to explore the performance benefits of leveraging GPU acceleration for RSA signature operations compared to traditional CPU-based implementations. By implementing and benchmarking RSA signature algorithms on both GPU and CPU platforms, the research provides insights into the comparative performance, scalability, and resource utilization characteristics of each architecture. Furthermore, the study investigates optimization techniques tailored to GPU architectures to exploit parallelism and accelerate RSA signature operations. The findings contribute to the understanding of the trade-offs between GPU and CPU implementations for RSA signatures, considering factors such as throughput, latency, power consumption, and cost-effectiveness. Additionally, the research sheds light on the potential implications of GPU acceleration for improving the efficiency and scalability of cryptographic operations in various applications, including secure communication, digital signatures, and authentication protocols. Overall, this study advances the field by demonstrating the feasibility and advantages of GPU-based implementations for RSA signatures, paving the way for optimized cryptographic solutions in both GPU and CPU computing environments.

Systematic and Critical Review of RSA Based Public Key Cryptographic Schemes: Past and Present Status

The paper titled "Systematic and Critical Review of RSA Based Public Key Cryptographic Schemes: Past and Present Status" provides a comprehensive examination of RSA-based public key cryptographic schemes, encompassing their historical development and current status[9]. RSA is a foundational cryptographic algorithm widely used for secure communication, digital signatures, and key exchange. This systematic review aims to analyze the evolution of RSA-based cryptographic schemes over time, considering advancements, vulnerabilities, and emerging trends. By

conducting a critical assessment of past research, the paper evaluates the strengths, weaknesses, and practical implications of various RSA-based schemes, including RSA encryption, RSA signatures, and related protocols. The review covers key aspects such as security properties, computational efficiency, scalability, and compatibility with modern cryptographic requirements. Furthermore, the study highlights recent developments, challenges, and future directions in RSA-based cryptography, considering factors such as quantum computing threats, post-quantum cryptography, and the adoption of new cryptographic standards. By synthesizing existing literature and research findings, the paper offers valuable insights into the state-of-the-art RSA-based cryptographic schemes, informing practitioners, researchers, and policymakers about the current landscape and potential areas for further exploration and improvement. Overall, this systematic and critical review contributes to a deeper understanding of RSA-based cryptography, facilitating informed decision-making and advancements in secure communication and cryptographic protocols.

Double Counting in 2t -ary RSA Precomputation Reveals the Secret Exponent

The paper titled "Double Counting in \(2^t\)-ary RSA Precomputation Reveals the Secret Exponent" delves into a cryptographic vulnerability within the precomputation phase of RSA key generation, leading to the exposure of the secret exponent. RSA encryption relies on the difficulty of factoring large composite numbers into their prime factors.[10] In the precomputation phase of \(2^t\)-ary RSA, certain optimizations are employed to accelerate the key generation process. However, this study identifies a flaw in the precomputation process, termed "double counting," which inadvertently discloses information about the secret exponent. By exploiting this vulnerability, an attacker can deduce the secret exponent with reduced computational effort, compromising the security of RSA encryption. The paper meticulously dissects the mathematical underpinnings of \(2^t\)-ary RSA precomputation and elucidates the mechanism through which double counting occurs. Moreover, it discusses the implications of this vulnerability for cryptographic security and proposes mitigation strategies to address the issue. The findings underscore the importance of rigorous cryptographic analysis and highlight the need for robust implementation practices to safeguard against potential vulnerabilities in RSA-based systems. Furthermore, the research contributes to the ongoing discourse on cryptographic protocol design and underscores the necessity of continuously evaluating and fortifying cryptographic schemes against emerging threats and attacks. Overall, this paper sheds light on a critical security flaw in RSA precomputation, emphasizing the imperative of proactive measures to enhance the resilience of RSA encryption against potential exploits and vulnerabilities**.**

Conclusion

These papers give us a brief idea about the the limitation of RSA encryption and how to overcome the limitation using Quantum algorithms like Shor and Quantum Phase Evaluation thus rectify the limitation and making a hybrid robust encryption mechanism

Acknowledgement

I would like to thank the anonymous referees for their helpful guidance that has improved the quality of this paper. I would also like to express gratitude and sincere thanks to my guide for the valuable support and guidance in the completion of this paper

REFERENCES

[1] Kento Oonishi; Noboru Kunihiro “Shor's Algorithm Using Efficient Approximate Quantum Fourier Transfor”(IEEE Xplore Issue:23 Sep, 2023)

[2] Vasil Dimitrov; Luigi Vigneri; Vidal Attias “Fast Generaion of RSA Keys Using Smooth Integers” (IEEE Xplor Issue 2 July, 2022)

[3] Iqra Mustafa; Imran Ullah Khan; Sheraz Aslam; Ahthasham Sajid; “A Lightweight Post-Quantum Lattice-Based RSA for Secure Communications” (IEEE Xplore Issue: 19 May, 2020)

[4] Abderrahmane Nitaj Muhammad Rezal Bin Kamel Ariffin; Nurul Nur Hanisah Adenan; “Security Issues of Novel RSA Variant” (IEEE Xplore Issue: 16 May, 2022)

[5] Alexei Petrenko “2 Implementation of the Shor Algorithm on a Quantum Circuit
” (IEEE Xplore Issue: 2022)
[6] G. R. Mounica; G. Manimaran; L. B. Jerome; P. Bhattacharjee “Implementation of 5-Qubit approach-based Shor's Algorithm in IBM Qiskit” (IEEE Xplore Issue: 31 Jan, 2021)

[7] Damien Vergnaud “Comment on “Efficient and Secure Outsourcing Scheme for RSA Decryption in Internet of Things”” (IEEE Xplore Issue: 23 June, 2020)

[8] Eduardo Ochoa-Jiménez; Luis Rivera-Zamarripa; Nareli Cruz-Cortés; Francisco "Implementation of RSA Signatures on GPU and CPU Architectures" (IEEE Xplore Issue: 3 Jan, 2020)

[9] Raza Imam; Qazi Mohammad Areeb; Abdulrahman Alturki; Faisal Anwer "Systematic and Critical Review of RSA Based Public Key Cryptographic Schemes: Past and Present Status" (IEEE Xplore Issue: 18 Nov, 2021)

[10] Masahiro Kaminaga; Hideki Yoshikawa; Toshinori Suzuki "Double Counting in 2t -ary RSA Precomputation Reveals the Secret Exponent"(IEEE Xplore Issue:7 July, 2021)