Original Article
ISSN (Online): 2582-7472

ShodhKosh: Journal of Visual and Performing Arts
June 2024 5(6), 331–335

A BRIEF OVERVIEW OF CRYPTOGRAPHIC TECHNIQUES: ENCRYPTION, DECRYPTION, RSA AND MORE

Dr. Poonam Chaudhary [1], Dr. Vishal Kumar [2]

[1] Assistant Professor, Department of Mathematics, Multanimal Modi PG College, Modinagar, Ghaziabad, Uttar Pradesh
[2] Professor, Department of Physics, Government Women PG College, Kandhla, Shamli, Uttar Pradesh

DOI
10.29121/shodhkosh.v5.i6.2024.3916

Funding: This research received no specific grant from any funding agency in the public, commercial, or not-for-profit sectors.

ABSTRACT

Communication plays a crucial role in today's information age, with the advancement of new technologies, altogether making security a vital consideration. In order to protect the information being transmitted, having a specific mechanism is of utmost importance. This current era of digitization involves the process of converting original data into an unreadable format, known as encryption, and then reverting it back to its original form, called decryption. The study encompassing both encryption and decryption is referred to as cryptography. This paper aims to analyze various types of cryptography, as well as the concepts of encryption and decryption, and provides a brief overview of cryptographic techniques. When discussing information security, several services come to mind, including confidentiality (information privacy), authentication, and integrity (ensuring data has not been altered). This paper offers a comprehensive description of these cryptographic techniques and focuses on the public key cryptography algorithm, RSA.

Keywords: Cryptography, Encryption, Transmission, Technology, Data, Communication, Digitization

1. INTRODUCTION

For a mathematician, cryptography primarily refers to public-key encryption based on one-way trapdoor functions. It's often taught as a sophisticated application of number theory, including concepts like the Euler–Fermat Theorem, which underpins the RSA cryptosystem, or the discrete logarithm problem linked to primitive roots of large primes or elliptic curves, foundational for the ElGamal and Diffie–Hellman systems. Conversely, engineers view public-key encryption as just a small aspect of cryptography. Their focus is on efficiently encrypting, transmitting, and decrypting large volumes of data in real-time using binary strings. The preferred method is symmetric-key encryption, where the sender encrypts the plaintext by combining it with a secret pseudo random key of the same length, and the receiver decrypts the ciphertext by adding the same key to retrieve the plaintext. However, the sender and receiver don't share a secret key; instead, they generate one dynamically during encryption or decryption. Specifically, the sender encrypts a short seed (128 bits for standard security or 256 bits for top security) using public-key encryption and sends it to Bob, who then retrieves it. They proceed to encrypt and decrypt the plaintext blocks using this seed.

For subsequent blocks, the sender employs a public algorithm like AES (Advanced Encryption Standard) to permute the ciphertext blocks before encrypting the next plaintext block, with the receiver mirroring this process for decryption. The key takeaway is that symmetric-key cryptography is fast and dependable, while public-key methods are too slow for mobile communications, financial transactions, or pay TV. Beyond just encryption and decryption, cryptography plays a crucial role in ensuring that the receiver can verify the authenticity of messages from the sender, which involves digital signatures, message authentication codes, and timestamps. The sender also needs to confirm the receiver's identity. Additionally, cryptography is essential for securely storing passphrases, identification codes, and PINs for smart cards, with hash functions being a vital tool for cryptographers. A significant part of practical cryptography is dedicated to maintaining data integrity and effective key management. Recent hacking incidents involving email storage and mobile devices highlight that the failures often stem not from weak cryptographic security, but from human lapses in following proper protocols.

2. UNDERSTANDING THE PROCESS

Plain text, or normal text, sent over a network is first converted into cipher text so that only the sender and recipient can access the information. This encoding process, which transforms plain text into cipher text, is known as encryption. The reverse process, where cipher text is converted back into plain text, is called decryption, and it is the opposite of encryption. In computer-to-computer communication, the sender's computer typically encrypts the plain text message into cipher text before transmitting it over the network. The recipient's computer then decrypts the received encrypted message to retrieve the original plain text. The combined processes of encryption and decryption fall under the umbrella of cryptography.

In essence, cryptography is both an art and a science that provides security by encoding messages to make them unreadable to unauthorized individuals. It can obscure the meaning of information in various forms, including software, graphics, or voice. The primary function of cryptography is to enable secure information exchange among participants, ensuring that others cannot read it. While its main goal is to ensure confidentiality, cryptography also addresses other challenges such as data integrity, authentication, and non-repudiation. It encompasses methods that allow information to be transmitted securely so that only the intended recipient can access it. Ongoing research continues to develop new cryptographic algorithms. However, finding the right algorithm is complex, as it must take into account several factors, including security, algorithm features, time complexity, and space complexity.

3. PURPOSE AND APPROACH

In data and telecommunications, cryptography is essential for communication over any non-trusted medium, which encompasses nearly all networks, especially the Internet. For application-to-application communication, there are specific security requirements, including:

- **Authentication:** This is the process of verifying one's identity. Currently, the main forms of host-to-host authentication on the Internet are name-based or address-based, both of which are known to be weak.
- **Privacy/Confidentiality:** This ensures that only the intended recipient can read the message.
- **Integrity:** This guarantees that the message received has not been altered in any way from its original form.
- **Non-repudiation:** This provides a means to prove that the sender actually sent the message. Cryptography, therefore, not only safeguards data against theft or modification but can also facilitate user authentication.

There are generally three types of cryptographic schemes used to achieve these objectives: secret key (or symmetric) cryptography, public-key (or asymmetric) cryptography, and hash functions. In all scenarios, the original unencrypted data is referred to as plaintext, which is then encrypted into cipher text, typically requiring decryption back into usable plaintext. In an encryption scheme, the message, or plaintext, is processed using an encryption algorithm to create cipher text, which can only be read when decrypted. Historically, encryption has been utilized by militaries and governments for secret communications, and it is now widely used to protect information across various civilian applications.

It secures data in transit, such as information transferred over networks (e.g., the Internet, e-commerce), mobile phones, wireless microphones, intercom systems, Bluetooth devices, and ATMs. In the proposed technique, a common key known as a private key is shared between the sender and receiver. The private key is essentially part of symmetric key concepts, where plaintext is converted into encrypted text (cipher text) using the private key, and the same key is used to decrypt the cipher-text back into plaintext. The encryption key is closely related to the decryption key.

4. SYMMETRIC AND ASYMMETRIC KEY CRYPTOGRAPHY

Symmetric encryption is the oldest and most well-known technique. It uses a secret key— which can be a number, a word, or a random string of letters— to modify the content of a message in a specific way. In contrast, asymmetric encryption involves two related keys, known as a key pair. A public key is openly shared with anyone who may wish to send you a message, while a second, private key is kept secret, known only to you. Symmetric-key cryptography refers to methods where both the sender and receiver use the same key. Symmetric key ciphers can be implemented as either block ciphers or stream ciphers. A block cipher encrypts data in blocks of plaintext, while a stream cipher processes input one character at a time. Symmetric methods are generally faster than asymmetric cryptography.

Asymmetric-key cryptography, on the other hand, involves different keys for the sender and receiver, one for encryption and another for decryption, providing greater security compared to symmetric systems. Various algorithms are employed for both symmetric and asymmetric techniques. For symmetric key algorithms, examples include DES (Data Encryption Standard) and AES (Advanced Encryption Standard). Public key algorithms include RSA (Rivest-Shamir-Adleman) and Diffie-Hellman. RSA, named after its inventors Ron Rivest, Adi Shamir, and Leonard Adleman, is one of the first practical public-key cryptosystems and is widely used for secure data transmission. In this cryptosystem, the encryption key is public, while the decryption key remains secret. The security of RSA relies on the difficulty of factoring the product of two large prime numbers, a challenge known as the factoring problem. The RSA algorithm was first publicly introduced in 1977.

5. MECHANISM AND DEMERITS OF RSA

RSA utilizes a public key and a private key. The public key is accessible to everyone and is used for encrypting messages. Messages that are encrypted with the public key can only be decrypted in a reasonable time frame using the private key. The keys for the RSA algorithm are generated through the following steps:

- Select two distinct prime numbers, p and q.
- For added security, p and q should be chosen randomly.
- Calculate n=pq.
- The value n serves as the modulus for both the public and private keys. Its length, typically measured in bits, determines the key length.
- Compute $\phi(n)=\phi(p)\phi(q)=(p-1)(q-1)=n-(p+q-1)$, where ϕ is Euler's totient function. This value is kept private.
- Select an integer e such that $1<e<\phi(n)$ and $\gcd(e,\phi(n))=1$; this means e and $\phi(n)$ are coprime.
- The value e is made public as the public key exponent.
- Calculate d as $d \equiv e^{-1} \pmod{\phi(n)}$; d is the modular multiplicative inverse of *e* e modulo $\phi(n)$.
- The value d is kept as the private key exponent.

It is worth noting that, in RSA encryption, which is a deterministic algorithm (meaning it does not incorporate randomness), an attacker can effectively execute a chosen plaintext attack against the cryptosystem. RSA has the characteristic that the product of two ciphertexts equals the encryption of the product of the corresponding plaintexts, expressed as $m_1^e m_2^e \equiv (m_1 m_2)^e \pmod{n}$ m 1 e m 2 e ≡(m 1 m 2) e (modn). This multiplicative property enables the possibility of a chosen ciphertext attack.

6. EXPLORING OTHER VULNERABILITIES

The security of RSA encryption heavily relies on the size of the keys used. As computational power increases, especially with advancements in hardware and algorithms, the previously secure key lengths may become vulnerable to brute-force attacks. For example, keys that were considered secure a decade ago may no longer be adequate today. Current best practices recommend using key lengths of at least 2048 bits to ensure adequate security against potential attacks. However, longer key lengths can also result in increased computational overhead, leading to slower encryption and decryption processes. RSA's security is fundamentally based on the difficulty of factoring large prime numbers. However, if an attacker can exploit weaknesses in the implementation of RSA, they may be able to factor the modulus n n more easily. For instance, using poorly chosen or predictable primes can lead to vulnerabilities. Moreover, research in number theory and advancements in algorithms could potentially weaken RSA's foundational assumptions, making it easier for attackers to break the encryption.

RSA is susceptible to side-channel attacks, which exploit information leaked during the encryption or decryption processes. For instance, variations in timing, power consumption, or electromagnetic emissions can provide clues about the private key or the plaintext being processed. Attackers can analyze these side effects to deduce critical information, making it essential to implement countermeasures to mitigate such risks, such as constant-time algorithms and secure hardware. As mentioned earlier, RSA's multiplicative property can be exploited through chosen ciphertext attacks. In a CCA, an attacker can submit a chosen ciphertext to be decrypted and analyze the returned plaintext. If RSA is not implemented with appropriate padding schemes, such as Optimal Asymmetric Encryption Padding (OAEP), it can be vulnerable to this type of attack. This necessitates the use of robust padding mechanisms to add randomness and protect against potential exploitation. Effective key management is vital for maintaining the security of RSA. This includes the generation, distribution, storage, and revocation of keys.

If private keys are mishandled or stored insecurely, they may be exposed to unauthorized individuals. Additionally, if public keys are not verified properly, attackers could substitute their own public key, allowing them to intercept and decrypt messages intended for legitimate recipients. Organizations must establish comprehensive key management policies to mitigate these risks. One of the most significant challenges facing RSA and other classical encryption methods is the potential rise of quantum computing. Quantum computers could leverage Shor's algorithm to factor large integers efficiently, thereby breaking RSA encryption much faster than classical computers can. This looming threat has prompted researchers to explore post-quantum cryptography solutions, which would remain secure even in a world dominated by quantum computing. Security vulnerabilities often arise not from the cryptographic algorithms themselves but from flaws in their implementation. Incorrectly implemented libraries or faulty software can introduce weaknesses that are exploitable. Developers must be vigilant in following best practices and thoroughly testing their implementations to avoid such pitfalls.

7. IN CONCLUSION

Cryptography is a fascinating field within computer science, primarily because much of the work involved remains confidential. Researchers explore a variety of techniques and algorithms, leading to diverse studies and findings in this domain. The most effective algorithms are typically those that are well-documented and widely recognized, as they have undergone extensive testing and scrutiny, ensuring their reliability and security. This paper delves deeper into the comparison between symmetric key cryptography and asymmetric key cryptography. It has been observed that symmetric key systems tend to be faster than their asymmetric counterparts. This speed is beneficial for applications requiring quick encryption and decryption processes, making symmetric algorithms advantageous in scenarios where performance is critical. However, asymmetric key cryptography offers distinct advantages, particularly in terms of scalability and enhanced security features such as authentication and non-repudiation.

Asymmetric systems utilize two keys, a public key for encryption and a private key for decryption, allowing for greater flexibility in user interactions and establishing trust between parties. This characteristic makes asymmetric cryptography particularly valuable for online communications and transactions, where secure identity verification is essential. Despite the strengths of both symmetric and asymmetric cryptography, there remains a pressing need to develop new algorithms that simplify the encryption and decryption processes. Current algorithms, including RSA and DES, while effective, can sometimes be cumbersome or resource-intensive. Innovations in cryptographic techniques

could lead to more efficient methods that maintain robust security while enhancing usability. Ongoing research in this area is crucial, as the landscape of cybersecurity continues to evolve, and the demand for faster, more reliable cryptographic solutions grows. Exploring new approaches and refining existing ones will be essential for advancing the field of cryptography and ensuring secure communications in an increasingly interconnected world.

CONFLICT OF INTERESTS

None.

ACKNOWLEDGMENTS

None.

REFERENCES

Chatterjee, Dripto, Joyshree Nath, Suvadeep Dasgupta, and Asoke Nath. "A New Symmetric Key Cryptography Algorithm Using Extended MSA Method: DJSA Symmetric Key Algorithm." *2023 International Conference on Communication Systems and Network Technologies*, IEEE, 2023, doi:978-0-2023-2024-3/11.

Nath, A., S. Das, and A. Chakrabarti. "Data Hiding and Retrieval." *Proceedings of the IEEE International Conference on Computer Intelligence and Computer Network*, Bhopal, 2023-2024 Nov. 2023.

Koblitz, Neal. *A Course in Number Theory and Cryptography*. 2nd ed., Springer-Verlag, 2023.

Morkel, T., and JHP Eloff. "Encryption Techniques: A Timeline Approach." *Information and Computer Security Architecture (ICSA) Research Group Proceedings*, 2023.

Stallings, William. *Data and Computer Communications*. 6th ed., Pearson, 2024.

Islam, Md. Nazrul, Md. Monir Hossain Mia, Muhammad F. I. Chowdhury, and M. A. Matin. "Effect of Security Increment to Symmetric Data Encryption Through AES Methodology." *Ninth ACIS International Conference on Software Engineering, Artificial Intelligence, Networking, and Parallel/Distributed Computing*, IEEE, 2023, doi:10.2/SNPD.2023.2024.

Daemen, Joan, and Vincent Rijmen. "AES Submission Document on Rijndael." Version 2, Sept. 2023.

Nath, A., S. Ghosh, and M. A. Mallik. "Symmetric Key Cryptography Using Random Key Generator." *Proceedings of the International Conference on SAM 2023*, Las Vegas, USA, 12-15 June 2024, vol. 2, pp. 2023-2024.