EasyChair Preprint

№ 10218

Comparative Analysis of Cryptographic Algorithms in Computer Network

Ankit Kumar Soni, Gautam Kumar, Priya Kumari, Arpita Nayak and Arpita Arpita

May 21, 2023

COMPARATIVE ANALYSIS OF CRYPTOGRAPHIC ALGORITHMS IN COMPUTER NETWORK

Dr.Vishal, Ankit Kumar Soni, Gautam Kumar, Priya Kumari, Arpita Nayak, Arpita Arpita
Department of Computer Application, Lovely Professional University, Punjab, 144001, India

ABSTRACT
One technique to ensure the security and privacy of the user's data as well as the secrecy, authentication, integrity, availability, and identification of the data user is through the use of cryptography. The most valuable term in the realm of communication systems nowadays is security. One of the most secure technologies for delivering data to a valid user is cryptography, which transforms regular plain text into cipher text. Security come with a variety of technologies and approaches. Regarding different parameters, we give performance and comparison. It is possible to express the encryption ratio as a minimal, moderate, or maximal value. The Visual Studio. All the results of the experiment are implemented using net packages. Based on the quantity of keys, cryptography techniques can be classified as either symmetric or asymmetric. While asymmetric algorithms employ different keys for encryption and decryption, symmetric algorithms use the same key for both operations. The most popular symmetric algorithms are Blowfish, Data Encryption Standard (DES), and Advanced Encryption Standard (AES). Elliptic Curve Cryptography, Diffie-Hellman, and RSA are examples of asymmetric algorithms (ECC). A cryptographic algorithm's key size, complexity, and attack resistance all affect how effective it is. In computer networks, cryptographic techniques are frequently used to protect data while it is in motion and at rest, such as during online transactions, mail delivery, and file storage.

Keywords: Plain Text, Cipher Text, Decryption, Encryption, Algorithms.

1. INTRODUCTION

A cryptographic algorithm's primary goal is to give two parties a safe and secret channel of communication. This is accomplished by encoding the original communication in a form that is only understandable by those with the proper authorization. A secret key that is only known to the parties with permission is used to decrypt the message and restore it to its original form. In today's digital age, information security has become a critical aspect of computer networks. Cryptography is the science of secure communication, and cryptographic algorithms are the fundamental building blocks of secure communication in computer networks. Data communicated via computer networks is kept confidential, authentic, and authentic using cryptographic methods. Data confidentiality, integrity, and authenticity all assure that the data is not accessible to unauthorised parties, that it has not been altered in transit, and that it has been sent by the intended sender. Cryptographic algorithms can be divided into two categories: symmetric and asymmetric (Erondu, 2022). While asymmetric algorithms employ different keys for encryption and decryption, symmetric algorithms use the same key for both operations (G. Wei, 2008). The choice of algorithm depends on the particular requirements of the application. Both types of algorithms offer benefits and drawbacks. Finally, it should be noted that cryptographic techniques are crucial for protecting the confidentiality and security of data sent through computer networks. They are widely used in various applications, such as e-commerce, online banking, and secure messaging.

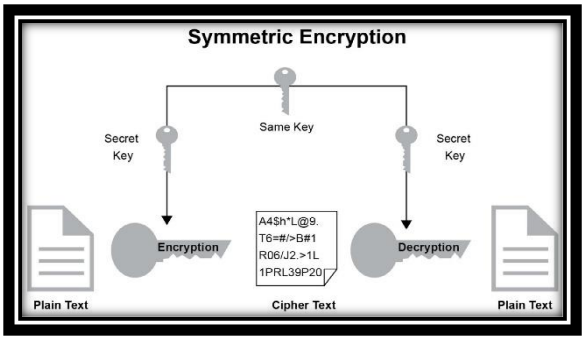

Figure 1. Symmetric Key Encryption

2. SYMMETRIC CRYPTOGRAPHYALGORITHM

The secret key encryption technique uses the same key for both encryption and decryption (Li Zhao, 2005). This section discusses each type of symmetric algorithm in detail, including how it operates and its benefits and drawbacks.

2.1 AES Algorithm

Rijndael originally introduced the AES (Advanced Encryption Standard) in October 2000. It was created in Belgium by Vincent Rijmen and Joan Daemen. It is a symmetric block cypher that supports blocks of 128, 192, and 256 bits in different block sizes. three block cipher of its own: AES-128, AES-192, and AES-256. Each of these types encrypts and decrypts data using cryptographic keys in blocks of 128 bits. Longer keys offer higher security. Data is encrypted and decrypted using the substitution-permutation network (SPN) design of AES, which employs a number of mathematical operations, including substitution, permutation, and XOR.The number of operations performed in a round of the AES algorithm depends on the key size. For instance, the AES-128 algorithm performs operations over 10 rounds, while the AES-256 method does operations over fourteen rounds (Vishal Choudhary, 2018). The procedures of substitution and permutation are combined in each round, and a round key is produced from the original key. The "key search problem," which refers to the challenge of undoing the mathematical operations utilised in the algorithm, is the foundation for AES' security. AES has undergone significant research and testing, and no real-world attacks have been discovered. AES is frequently used in many different applications, such as file encryption, secure chat, and online banking.[5]

2.2 DES Algorithm

A symmetric-key encryption method called the Data Encryption Standard (DES) was created by IBM in the 1970s and later standardised by the National Institute of Standards and Technology (NIST) in the United States. DES uses a Feistel network and a block cypher, which means it divides the block into two halves before modifying and merging them through a number of rounds. It encrypts data in fixed-size blocks of 64 bits. The key used by the algorithm is 56 bits, which is too small by modern standards and prone to brute-force assaults (Vishal Choudhary S. T., An Intrusion Detection Technique Using Frequency Analysis for Wireless Sensor Network, 2021). As a result more secure encryption techniques like Advanced Encryption Standard have taken their place (AES).It still has some applications, nevertheless, including in legacy systems and as a part of other cryptographic protocols. The DES algorithm's encryption and decryption speed is fast in terms of other symmetric algorithms. The basic steps of the DES algorithm are:

(i) Key Generation: The 56-bit key is transformed into 16 48-bit sub keys that will be used in the subsequent rounds of encryption and decryption.

(ii) Initial Permutation: The plaintext is permuted using a fixed table to rearrange the bits of the block.

(iii) Rounds: The block is divided into two halves, and each half goes through a series of 16 rounds, where the sub keys are used to perform a combination of substitution and permutation operations on the data.

(iv) Final Permutation: The final step of the algorithm is to permute the resulting block using another fixed table to produce the cipher text.To decrypt the cipher text, the process is reversed by using the same sub keys in the reverse order.

2.3 Triple-DES Algorithm

The triple data encryption algorithm, or 3DES, is an enhanced version of the previously revealed DES algorithm. TDEA is short for triple data encryption algorithm. In order to address the shortcomings of the DES algorithm, 3DES was created and started to be used in the late 1990s.3DES runs in two different modes:(i) Triple DES with EDE (Encrypt-Decrypt-Encrypt)mode: This mode applies three DES operations to each block of data, utilising two or three distinct keys. The algorithm first encrypts the plaintext with the first key, then decrypts the result using the second key, and lastly encrypts the decrypted result using the third key.(ii)Triple DES with DED (Decrypt-Encrypt-Decrypt) mode: This mode is similar to EDE mode, except the processes are conducted in reverse order. Because to its resistance to meet-in-the-middle attacks, 3DES in EDE mode offers greater security than 3DES in DED mode. Depending on whether two or three keys are employed, the key length for 3DES can be either 128 or 192 bits. Unfortunately, 3DES is becoming less used as more sophisticated encryption algorithms like AES gain popularity. Yet, it is still employed in some older programmes and systems that demand DES backward compatibility.

2.4 BLOWFISH Algorithm

As a quick, secure substitute for already-existing encryption techniques like DES, Bruce Schneider created the symmetric-key block cipher algorithm known as Blowfish in 1993. Using keys with varying lengths ranging from 32 bits to 448 bits, it is used to encrypt data in fixed-size blocks of 64 bits. The 64-bit blocks of plaintext are divided into pieces by the Blowfish method, which uses a combination of substitution and permutation operations to encrypt

each piece separately. The algorithm creates sequence of sub keys that are utilised in the encryption and decryption process using a 64-bit key. The basic steps of the Blowfish algorithm are as follows:
(i)Key Generation: The algorithm uses the user-supplied key to generate a series of sub keys, which Are used in the subsequent rounds of encryption and decryption.
(ii)Block Encryption: The plain text is divided into 64-bit blocks and encrypted using a Feistel network consisting of 16 rounds of substitution and permutation operations.
(iii) Block Decryption: The cipher text is decrypted using the same Feistel network in reverse order. Blowfish is renowned for its quickness, ease of use, and resistance to brute-force assaults. It has a solid security record and has undergone significant research. However, due to its smaller block size and worries about its security against more recent assaults, the Advanced Encryption Standard (AES) has essentially taken its place in many applications.[8]

2.5 MD5 Algorithm

MD5 (Message Digest 5) is a widely-used cryptographic hash function that generates a fixed-size output (128-bit) from an input message of any length. It was designed by Ron Rivest in 1991 as successor to MD4, with the goal of improving its security weaknesses. When using MD5, the input message is divided into 512-bit blocks and processed via a number of compression methods that combine and alter the data. The message digest, which is the final compression function's output and acts as a digital fingerprint or signature because it is specific to the input message, is produced. There are several uses for MD5, including message authentication, password hashing, and confirming the integrity of files. It has been discovered that it contains security flaws that leave it open to collision attacks, in which two different input messages result in the same message digest. It should not be used for cryptographic purposes as it is now regarded as being insecure. It is advised to use more secure hashing algorithms like SHA-256 and SHA-3.

3. ASYMMETRIC CRYPTOGRAPHY ALGORITHMS

This encryption method, also known as public-key encryption, employs various keys for encryption and decoding (Vishal Choudhary S. T., 2016). This section discusses each type of asymmetric algorithm in detail, including how it functions and its benefits and drawbacks.

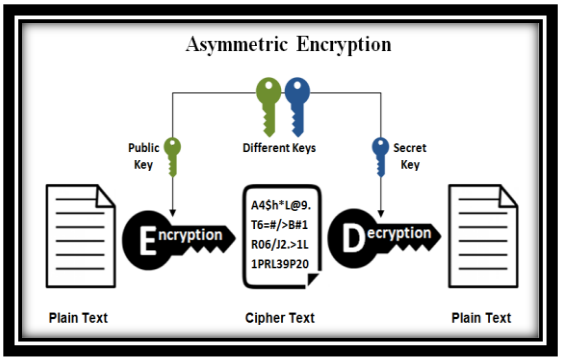

Figure 2. Asymmetric Key Encryption

3.1 RSA Algorithm

A public-key cryptography algorithm called RSA is used to transmit data securely via networks. It was created in 1977 and given the names Ron Rivest, Adi Shamir, and Leonard Adleman. An asymmetric cryptosystem is RSA. Two different keys are present. As one of them is shared by everyone, RSA is best known as a public key cryptosystem. As it is kept a secret, the other key is private. The mathematical foundation of RSA is solid. The RSA is regarded as reliable and secure due to its secrecy and privacy features. Moreover, RSA offers integrity, which keeps the content's original format. In the trading phase, One of RSA's drawbacks is that it requires the longest encryption times. Similar lengths are needed for c, which is a difficult requirement to achieve. In this situation, padding techniques are necessary, which increases processing time.RSA is mostly employed in hybrid encryption techniques and digital signatures, as well as in web browsers, chat programmers, email, VPNs, and various other types of communications that call for securely delivering data to servers or other users.

3.2ECC Algorithm

Elliptic Curve Cryptography (ECC) is a public key encryption algorithm that generates and exchanges keys by utilising the algebraic structure of elliptic curves over finite fields. ECC offers security that is comparable to that of other public key cryptography algorithms like RSA but with smaller key sizes, making it a more practical substitute for instances where resources are few, such as mobile devices and Internet of Things devices. With ECC, every user has both a private and public key. The public key is produced from a private key by multiplying it by an elliptic curve's specified point. The curve's final point becomes the public key, while the private key is kept private.ECC encryption and decryption use the following steps:

(i) Key Generation: The algorithm generates a private key and a corresponding public key based on a selected elliptic curve. Encryption:

(ii) To encrypt a message, the sender uses the recipient's public key to generate a shared secret point on the curve. The sender then uses the shared secret point to encrypt the message.

(iii) To decrypt the message, the recipient uses their private key to generate the shared secret point on the curve. The recipient then uses the shared secret point to decrypt the message. The Elliptic Curve Discrete Logarithm Problem (ECDLP), which involves determining the value of a private key from a public key, is the basis for the security of ECC (Vishal Choudhary S. T., he highly secure polynomial pool-based key pre-distribution scheme for wireless sensor network, 2020). ECC is a desirable alternative for secure communications in limited contexts duet its reduced key size and quicker computation tie when compared to RSA.

3.3 Digital Signature Algorithm

The National Institute of Standards and Technology (NIST) created the digital signature algorithm, or DSA, in the 1990s. Using a public-private key combination, it relies on the mathematical idea of modular exponentiation to sign and validate digital signatures. DSA does not encrypt or decrypt message digests using a private key or a public key, respectively. Rather, it makes a digital signature out of two 160-bit values using exceptional scientific abilities. In DSA The recipient should simply mark the communication as invalid if the general public key is unable to validate the digital signature. However, some governments and countries do not have any laws that address issues related to digital and innovation (Vishal, 2020). Although a digital signature provides validity, it does not provide confidentiality. As providing security is the ultimate goal, other techniques like encryption and unscrambling should be used. While transferring user data and information during email, DSA is employed in web applications.

4. PERFORMANCE ANALYSIS

Based on a variety of performance parameters, the performance results of numerous symmetric and asymmetric algorithms are analysed (Vishal Choudhary S. T., 2018). Which algorithm performs better than others is determined by these metrics. The ensuing performance indicators are examined.

FACTORS ANALYSED	**SYMMETRIC ENCRYPTION**				**ASYMMETRIC KEY ENCRYPTION**		
Algorithms	**AES**	**DES**	**3 DES**	**BLOWFISH**	**DSA**	**RSA**	**ECC**
Encryption Ratio	High	High	Moderate	High	High	High	High
Speed	Fast	Fast	Fast	Fast	Slow	Slow	Fast
Key Length	128– 192 Or 256- bit	56- bit Key	112 - 168 bits	32 bits to 448 bits.	2048 – 3072 bits	1024 - 2048 bits	160
Tunability	No	No	No	Yes	No	Yes	Yes
Security Against Attacks	Choose n-Plain Known- Plain text.	Brute Force	Brute Force Chosen – Plain text, Known Plain text	Dictionary Attacks	signature verification	Timing Attacks	Public parameters
Application	Wireless communication ,Bank	Image processing	Smart Card, epayment	Database Security, ECommer-ce Software	Web application and email verification	Internet Banking	Key exchange over web and Mobile.

4.1 Tunability
The ability to dynamically define the encrypted portion and the encryption parameters with regard to various applications and needs could be highly desired. The technique can only be used for a limited number of applications due to the static definition of the encrypted section and encrypted parameters.

4.2 Computational Speed
The encryption and decryption methods must be quick enough to meet real-time needs in many real-time applications.

4.3 Key Length Value
The crucial component of encryption approaches that demonstrates how data is encrypted is key management. Based on this key length, the encryption ratio for picture loss is calculated. The longer variable key length is used by the symmetric algorithm. Key management is therefore a crucial component of encryption processing.

4.4 Encryption Ratio
The amount of data that has to be encrypted is determined by the encryption ratio. Reduce the encryption ratio as mochas possible to simplify processing.

4.5 Security Issues
Whether an encryption method is secure against brute force or another plaintext-cipher text attack is determined by cryptographic security. It is crucial that the encryption strategy adhere to cryptographic security for highly valuable multimedia applications. We divide the three levels of cryptographic security used in our analysis into low, medium, and high.

4.6 Applications
Algorithm performance area for a certain function, either directly for the user or, in some situations, for application software. Find the optimal protocol for each application in the computer networking system.

5. RESULTS AND DISCUSSION

We can infer from the evaluation table above that symmetric key encryption algorithms have a high encryption ratio. The asymmetric encryption method has a higher tunability. As the asymmetric method of encryption uses a long key, breaking the code with RSA is difficult. The Symmetric key encryption is regarded as good in terms of speed. The blowfish method is listed as the second-best option in the symmetric key encryption approaches, followed by the AES algorithm. The RSA algorithm is more safe when using the asymmetric encryption method since it factors large prime numbers to create keys. As a result, the RSA algorithm is determined to be the preferable option in this manner.

6. CONCLUSION
The performance of many cryptographic algorithms is thoroughly analysed in this work to decide which method is optimal for a given sector of application. Performance is evaluated using the following parameters: encryption ratio, speed, key-length, tunability, and attack security. We came to the conclusion that symmetric cryptography algorithms, like AES, BLOWFISH, DES, and 3DES, are better suited for a variety of applications, including wireless communication, file, image processing, smart cards, and e-commerce type servers, based on our analysis of the results and discussion. However, for applications like Internet banking, web applications, email verification, key exchange over the web, and mobile, asymmetric cryptography algorithms like RSA, DSA, and ECC are the best option. This research can be further in the future by examining additional cryptographic schemes and techniques to determine potential application areas.

7. REFERENCES

1. Erondu, U. I. (2022). A Review on Different Encryption and Decryption Approaches for Securing Data. In A. K. Tyagi, *Handbook of , Research on Technical, Privacy, and Security Challenges in a Modern World* (pp. 357-370). IGI Global.

2. G. Wei, Z. G. (2008). Research and Realization of Random Encryption Algorithm. *International Conference on Internet Computing in Science and Engineering* (pp. 513-516). Harbin, China: IEEE.

3. Li Zhao, R. I. (2005). Anatomy and Performance of SSL Processing. *IEEE International Symposium on Performance Analysis of Systems and Software* (pp. 197-206). IEEE.

4. Vishal Choudhary, S. T. (2018). A Comparative Analysis of Cryptographic Keys and Security. *3rd International Conference and Workshops on Recent Advances and Innovations in Engineering (ICRAIE)* (pp. 1-8). Jaipur, India: IEEE.

5. Vishal Choudhary, S. T. (2018). A Distributed Key Management Protocol for Wireless Sensor Network. *International Conference on Advanced Informatics for Computing Research* (pp. 243–256). Shimla : Springer Nature Switzerland.

6. Vishal Choudhary, S. T. (2021). An Intrusion Detection Technique Using Frequency Analysis for Wireless Sensor Network. *International Conference on Computing, Communication, and Intelligent Systems (ICCCIS)* (pp. 206-210). Greater Noida: IEEE.

7. Vishal Choudhary, S. T. (2020). he highly secure polynomial pool-based key pre-distribution scheme for wireless sensor network. *Journal of Discrete Mathematical Sciences and Cryptography* , 95-114.

8. Vishal Choudhary, S. T. (2016). Improved Key Distribution and Management in Wireless Sensor Network. *Journal of Wireless Communications* , 16-22.

9. Vishal, S. T. (2020). An Efficient Quantum Key Management Scheme. *4th International Conference on Internet of Things and Connected Technologies (ICIoTCT)* (pp. 269–277). jaipur,India: Springer Nature Switzerland .