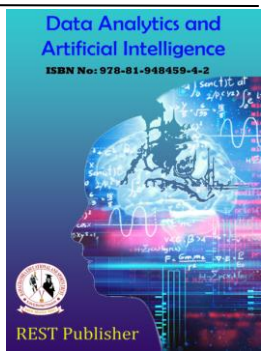
Data Analytics and Artificial Intelligence

Vol: 3(2), 2023

REST Publisher; ISBN: 978-81-948459-4-2

Website: http://restpublisher.com/book-series/daai/

DOI: https://doi.org/10.46632/daai/3/2/15



Overview of Cryptography

Archana B U, V Niranjana

St. Joseph's College of Arts and Science for Women, Hosur, Tamilnadu, India

*Corresponding Author Email: archanabu3129@gmail.com

Abstract. *Until very recently, the term "cryptography" was almost always used to refer to "encryption," which is the act of transforming plaintext or conventional information into an unintelligible form (called cipher text). Decryption is the opposite, or the process of returning plaintext from the unintelligible cipher text. The encryption and reversing decryption operations are carried out by a pair of algorithms known as cyphers (or cyphers). The algorithm and, in each case, a "key" work together just to regulate the precise operation of a cypher. The key, which is required to decrypt the cipher text, is a secret that should ideally only be known by the communicants. It typically takes the form of a short string of characters that the user can remember. A "cryptosystem" is the ordered list of elements with a finite number of potential combinations in formal mathematics.*

Keywords: *Cryptography, Encryption, Cryptosystem, Cyphers.*

1. INTRODUCTION

The art and science of concealing the messages to introduce secrecy in information security is recognized as cryptography. A message is plaintext (sometimes called clear text). The process of disguising a message in such a way as to hide its substance is encryption. An encrypted message is cipher text. Cryptography is the study and practice of techniques for secure communication in the presence of third parties called adversaries. It deals with developing and analyzing Protocols which prevents malicious third parties from retrieving information being shared between two entities thereby following the various aspects of information security. Secure Communication refers to the scenario where the message or data shared between two parties can't be accessed by an adversary. In Cryptography, an Adversary is a malicious entity, which aims to retrieve precious information or data thereby undermining the principles of information security. Data Confidentiality, Data Integrity, Authentication and Non-repudiation are core principles of modern-day cryptography. Cryptography can be strong or weak. Cryptographic strength is measured in the time and resources it would require to recover the plaintext. The result of strong cryptography is cipher text that is very difficult to decipher without possession of the appropriate decoding tool. How difficult? Given all of today's computing power and available time—even a billion computers doing a billion checks a second—it is not possible to decipher the result of strong cryptography before the end of the universe.

2. THE PURPOSE OF CRYPTOGRAPHY

The purpose of cryptography is to protect data transmitted in the likely presence of an adversary a cryptography transformation of data is a procedure by which plain text data is disguised, or encrypted, resulting in an altered text, called cipher text, that does not reveal the original input. The cipher text can be reverse-transformed by a designated recipient so that the original plaintext can be recovered. Cryptography plays an essential role in Modern cryptography concerns itself with the following four objectives:

- Confidentiality. The information cannot be understood by anyone for whom it was unintended.
- Integrity. The information cannot be altered in storage or transit between sender and intended receiver without the alteration being detected.
- Non-repudiation. The creator/sender of the information cannot deny at a later stage their intentions in the creation or transmission of the information.
- Authentication. The sender and receiver can confirm each other's identity and the origin/destination of the information.

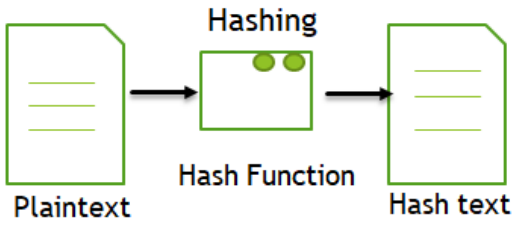
Procedures and protocols that meet some or all of the above criteria are known as cryptosystems. Cryptosystems are often thought to refer only to mathematical procedures and computer programs; however, they also include the regulation

of human behavior, such as choosing hard-to-guess passwords, logging off unused systems and not discussing sensitive procedures with outsiders.

3. TYPES OF CRYPTOGRAPHIC ALGORITHM

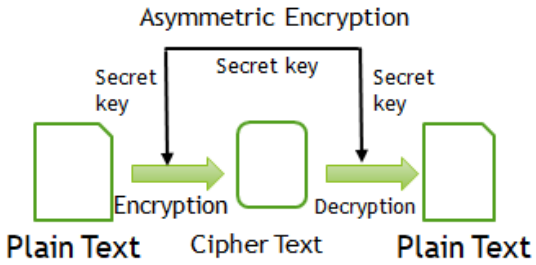
There are three general classes of NIST-approved cryptographic algorithms, which are defined by the number or types of cryptographic keys that are used with each.

Hash function: A cryptographic hash function does not use keys for its basic operation. This function creates a small digest or "hash value" from often large amounts of data through a one-way process. Hash functions are generally used to create the building blocks that are used in key management and provide security services such as:



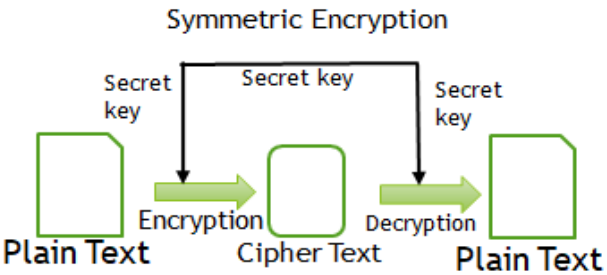
- Providing source and integrity authentication services by generating message authentication codes (MACs)
- Compressing messages for generating and verifying digital signatures
- Deriving keys in key-establishment algorithms
- Generating deterministic random numbers

Asymmetric-key algorithms: Also referred to as public-key algorithms, asymmetric-key algorithms use paired keys (a public and a private key) in performing their function. The public key is known to all, but the private key is controlled solely by the owner of that key pair. The private key cannot be mathematically calculated through the use of the public key even though they are cryptographically related. Asymmetric algorithms are used for:



- Computing digital signatures
- Establishing cryptographic keying material
- Identity Management

Symmetric-key algorithms: Also referred to as a secret-key algorithm, a symmetric-key algorithm transforms data to make it extremely difficult to view without possessing a secret key. The key is considered symmetric because it is used for both encrypting and decrypting. These keys are usually known by one or more authorized entities. Symmetric key algorithms are used for:



- Providing data confidentiality by using the same key for encrypting and decrypting data.
- Providing Message Authentication Codes (MACs) for source and integrity authentication services. The key is used to create the MAC and then to validate it.
- Establishing keys during key-establishment processes
- Generating deterministic random numbers

There are two types of symmetric encryption algorithms are Block algorithms. Set lengths of bits are encrypted in blocks of electronic data with the use of a specific secret key. As the data is being encrypted, the system holds the data in its memory as it waits for complete blocks. Stream algorithms. Data is encrypted as it streams instead of being retained in the system's memory. Some examples of symmetric encryption algorithms include:

- AES (Advanced Encryption Standard)
- DES (Data Encryption Standard)
- IDEA (International Data Encryption Algorithm)
- Blowfish (Drop-in replacement for DES or IDEA)
- RC4 (Rivest Cipher 4)
- RC5 (Rivest Cipher 5)
- RC6 (Rivest Cipher 6)

4 .TRIPLE DES ALGORITHM

Although it's officially known as the Triple Data Encryption Algorithm (3DEA), it is most commonly referred to as 3DES. This is because the 3DES algorithm uses the Data Encryption Standard (DES) cipher three times to encrypt its data. DES is a symmetric-key algorithm based on a Feistel network. As a symmetric key cipher, it uses the same key for both the encryption and decryption processes. The Feistel network makes both of these processes almost exactly the same, which results in an algorithm which is more efficient to implement. DES has both a 64-bit block and key size, but in practice, the key only grants 56-bits of security. 3DES was developed as a more secure alternative because of DES's small key length. In 3DES, the DES algorithm is run through three times with three keys; however, it is only considered secure if three separate keys are used.

5. THE USE OF 3DES

Once the weaknesses of normal DES became more apparent, 3DES was adopted in a wide range of applications. It was one of the more commonly used encryption schemes before the rise of AES. Some examples of its implementations included:

- Microsoft Office
- Firefox
- EMV payment systems

Many of these platforms no longer use 3DES because there are better alternatives. The National Institute of Standards and Technology (NIST) has released a draft proposal saying that all forms of 3DES will be deprecated through 2023 and disallowed from 2024 onward. Although it's just a draft, the proposal signifies the end of an era, and it is well past the time to move onto other, more secure algorithms.

6. UNDERSTANDING THE DES ALGORITHM

Before we can talk about the details of 3DES, it's important to understand the DES algorithm that it's derived from. So let's start right at the beginning. We use encryption to turn our plaintext data into cipher text, which is information that cannot be accessed by attackers (as long as we are using appropriate algorithms). Encryption algorithms are essentially complex mathematical formulas. When it comes to encrypting something like "Let's go to the beach", many people get confused. After all, how can you apply math to things like letters and characters?

Encoding the text: The reality is that computers don't deal in letters and characters. Instead, they work on a system of 1s and 0s known as binary. Each 1 or 0 is known as a bit, and a collection of eight of them is known as a byte. You can either look it up manually or use an online converter to see that in binary, "Let's go to the beach" becomes:

01001100 01100101 01110100 00100111 01110011 00100000 01100111 01101111 00100000 01110100 01101111 00100000 01110100 01101000 01100101 00100000 01100010 01100101 01100001 01100011 01101000

Blocks: When data is encrypted, it's divided into separate blocks for processing. DES has a 64-bit block size, which essentially means that each block fits a mix of 64 ones and zeros. Our first block (the first 64 digits of the binary shown above) would be:

01001100 01100101 01110100 00100111 01110011 00100000 01100111 01101111

Our second would be:

00100000 01110100 01101111 00100000 01110100 01101000 01100101 00100000

And our final block would be:

01100010 01100101 01100001 01100011 01101000

Padding: You may have noticed that our third block is only 40 bits long. Before it can be encrypted, it needs to be build up to a 64-bit block size. This is done with **padding**, which involves adding extra information to a block in order to complete it. This can be done with a number of different schemes, and it can also serve to make encrypted information harder to crack, but we won't get into that in this article.

7. 3-DES

As the security weaknesses of DES became more apparent, 3DES was proposed as a way of extending its key size without having to build an entirely new algorithm. Rather than using a single key as in DES, 3DES runs the DES algorithm three times, with three 56-bit keys:

- Key one is used to **encrypt** the plaintext.
- Key two is used to **decrypt** the text that had been encrypted by key one.
- Key three is used to **encrypt** the text that was decrypted by key two.

In each stage, the complete DES process is followed as outlined above. Now, you may be wondering "How can applying decryption in the second step enhance security?"The answer is that it uses a separate key. If the first key was also used to decrypt the data in the second step, then the data would be right back where it started. However, since it uses a different key, the decryption process doesn't actually serve to decrypt the data. It may seem logically perverse, but decrypting with a separate key only serves to jumble up the data even further. Once the second key has "decrypted" the data, the third key is applied to encrypt it again. The result is the 3DES cipher text. 3DES is structured this way because it allows implementations to be compatible with single key DES, two key DES, and three key DES (these are covered in the following section). This would not work if encryption was used in all three steps.

8. 3-DES KEYING OPTIONS

Technically, 3DES can be implemented with three different key configurations. Despite this, the second and third options are insecure and should never be implemented.

- **Keying option one** – This option uses three independent keys and is the most secure
- **Keying option two** – In this configuration, the first and third keys are the same
- **Keying option three** – This uses three identical keys. When identical keys are used, the decryption process in the second stage cancels out the first encryption, leaving only the final encryption to alter the data. This makes the result the same as ordinary DES

The 3DES process: Keying option one: Let's be honest, the entirety of the 3DES process can make your head spin, especially if you are new to cryptography. To help it sink in, here's a brief summary of the entire encryption scheme of the 3DES algorithm: The plaintext enters the 3DES algorithm and is first **encrypted with key one** in the following steps

- Key schedule – the 16 sub keys are derived from key one
- Initial permutation
- The block is split into left and right halves
- The right half is sent through the F function
- Expansion permutation
- XOR with the sub key for the round
- Substitution
- Permutation
- XOR the result of the F function with the left side
- Make the old right side the new left side, and the result the new right side

Repeat the above steps 14 times

- The right half is sent through the F function
- Expansion permutation
- XOR with the sub key for the 16^{th} round
- Substitution
- Permutation
- XOR the result of the F function with the left side
- Combine the left and right sides of the block together
- Final permutation

Take the text that has been encrypted with key one, then send it through the **"decryption" process with key two**:

- Key schedule – the 16 sub keys are derived from key two
- Initial permutation
- The block is split into left and right halves
- The right half is sent through the F function
- Expansion permutation
- XOR with the sub key for the round (starting from the 16^{th} sub key for decryption)
- Substitution
- Permutation
- XOR the result of the F function with the left side
- Make the old right side the new left side, and the result the new right side

Repeat the above steps 14 times

- The right half is sent through the F function
- Expansion permutation
- XOR with the sub key for the first round
- Substitution
- Permutation
- XOR the result of the F function with the left side
- Combine the left and right sides of the block together
- Final permutation

Take the data that has been "decrypted" by key two, then send it through the **encryption process with key three**:

- Key schedule – the 16 sub keys are derived from key three
- Initial permutation
- The block is split into left and right halves
- The right half is sent through the F function
- Expansion permutation
- XOR with the sub key for the round
- Substitution
- Permutation
- XOR the result of the F function with the left side
- Make the old right side the new left side, and the result the new right side

Repeat the above steps 14 times

- The right half is sent through the F function
- Expansion permutation
- XOR with the sub key for the 16^{th} round
- Substitution
- Permutation

XOR the result of the F function with the left side

- Combine the left and right sides of the block together
- Final permutation
- The result is the 3DES cipher text.

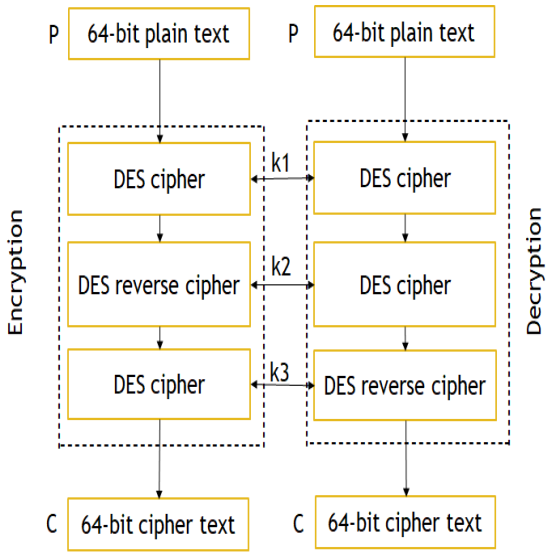
9. THE SECURITY OF 3-DES

The security of 3DES depends on which keying option is being used. Keying option one involves three different 56-bit keys, which gives it a total key length of 168 bits. The effective length is reduced considerably by meet-in-the-middle attacks, which bring its real-world security down to 112 bits. Meet-in-the-middle attacks are useful against encryption schemes that repeat the same algorithm several times. The technique stores the immediate values from each encryption stage, then uses this information to radically improve the time that it would take to brute force the algorithm. Options two and three have significantly smaller keys and are vulnerable to both known-plaintext, and chosen-plaintext attacks, as well as others. Known-plaintext attacks are possible when an adversary has access to both the plaintext and cipher text of a message. If an algorithm is susceptible to these attacks, the attacker can use this information to deduce the key, which allows them to crack all of the other data that has been encrypted by the same key. A chosen-plaintext attack is similar, but it involves the attacker uncovering the key by comparing cipher texts to arbitrary plaintexts. Because of these vulnerabilities and the overall small key-sizes involved, keying options two and three are insecure and should not be implemented.

10.ALGORITHM

The encryption-decryption process is as follows −

- Encrypt the plaintext blocks using single DES with key K_1.
- Now decrypt the output of step 1 using single DES with key K_2.
- Finally, encrypt the output of step 2 using single DES with key K_3.
- The output of step 3 is the cipher text.
- Decryption of a cipher text is a reverse process. User first decrypt using K_3, then encrypt with K_2, and finally decrypt with K_1.



11.CONCLUSION

Cryptography is the practice of secure communication in the presence of third parties. Its objective is to make it difficult for an eavesdropper to understand the communication. Cryptography is used in a variety of applications, including email, file sharing, and secure communications. The conclusion of cryptography is that it is a powerful tool for secure communication, but it is not perfect. There are a number of ways to attack a cryptographic system, and new attacks are constantly being discovered. Cryptography is an important part of security, but it is not the only factor to consider.

ACKNOWLEDGMENT

I am delighted to express my heartfelt appreciation to our department's head and staff, as well as family and friends. This paper is made possible by their encouragement, assistance, and support.

REFERENCES

[1]. van Tilborg, H.C.A.: Encyclopedia of Cryptography and Security. Springer-Verlag
[2]. https://www.alibabacloud.com
[3]. National Institute of Standards and echnology: FIPS PUB 46-3: Data Encryption Standard (DES)
[4]. Gomes, O., Moreno, R., Pimenta, T.: Afast cryptography pipelined hardware developed in FPGA with VHDL. In: The 3rd International Congress on Ultra
[5]. National Institute of Standards and Technology, http://www.nist.gov/index.html
[6]. https://www.researchgate.net/publication/334418542 A Review Paper on Cryptography
[7]. https://www.garykessler.net/library/crypto.html